Juan J. Cuadrado-Gallego
Yuri Demchenko

Data Analytics

A Theoretical and Practical View from the EDISON Project

Springer

Data Analytics

Juan J. Cuadrado-Gallego • Yuri Demchenko

Data Analytics

A Theoretical and Practical View from the EDISON Project

With contributions by Josefa Gómez Pérez and Abdelhamid Tayebi Tayebi

Springer

Juan J. Cuadrado-Gallego
Department of Computer Science
University of Alcalá
Madrid, Spain

With Contrib. by
Josefa Gómez Pérez
Department of Computer Science
University of Alcalá
Madrid, Spain

With Contrib. by
Abdelhamid Tayebi Tayebi
Department of Computer Science
University of Alcalá
Madrid, Spain

Yuri Demchenko
Faculty of Science
Universiteit van Amsterdam
Amsterdam, The Netherlands

ISBN 978-3-031-39128-6 ISBN 978-3-031-39129-3 (eBook)
https://doi.org/10.1007/978-3-031-39129-3

This Springer imprint is published by the registered company Springer Nature Switzerland AG
The registered company address is: Gewerbestrasse 11, 6330 Cham, Switzerland

Paper in this product is recyclable.

Preface

From Juan J. Cuadrado-Gallego and Yuri Demchenko

This is the second book of a series that started in 2020 with the publication of the first book *The Data Science Framework: A View from the EDISON Project*. That first book was about the EDISON Data Science Framework (EDSF) developed by the EDISON project, whose definition of Data Science and Data Scientist as a profession that became widely accepted by the academic and professional communities. Since the publication of the first book, the EDSF has been developed by the practitioner community, including EDSF application to new data-related professions, such as Data Stewardship, and new (reference) educational and training courses development. The presented second book provides an important development and experience with establishing the theoretical and practical foundation for mastering the Data Science that is related to the first Data Science Analytics Knowledge Area Group as defined in the EDISON Data Science Body of Knowledge.

The book has been thought to help to start the learning of the techniques and algorithms used in data analytics and start dealing with their computational implementations. The book is intended to be used, both, as a text book to teach the concepts in courses about data analytics at graduate or postgraduate levels and to learn the data analytics knowledge by the practitioner readers by themselves. The book provides suggestions on how to use it for both purposes.

As in the first book, we acknowledge initial EU funding for the EDISON project (grant number 675419 in 2015–2017), and continuing efforts of the EDISON Community Initiative for EDSF maintenance and recent publication of the EDSF Release 4 (2022). Additionally, we would also like to acknowledge our universities, Universiteit van Amsterdam, Amsterdam, the Netherlands, and the Universidad de Alcalá, Madrid, Spain, for their support during the realization of this book. We acknowledge the researchers who have helped us in the realization of the book and who are coauthors in three of the chapters: Josefa Gomez and Abdelhamid Tayebi, from the University of Alcalá, Madrid, Spain. Finally, we want to give our

acknowledgment to our editor Paul Drougas for his trust in our proposal, his patience, and his help in the production of the book.

From Juan J. Cuadrado-Gallego

I want to dedicate this book to my wife Ana and my daughters Ana and Cris. Ganz lieben Dank für alles.

From Yuri Demchenko

I dedicate this book to my lovely women, my wife Natalia, my daughter Anastasia, and my granddaughter Sonia. All of them have an important role in what I do, what I write, what I learn, and how I live.

Contents

List of Figures

Introduction to Data Science and Data Analytics

This initial chapter, "Introduction to Data Science and Data Analytics", presents the main concepts related to the subject of the book. As happened in the first book of the series, "Introduction to the Data Science Framework: A View from the EDISON Project", the chapter uses the common word *about* to start all the sections that present the introduction to what is Data Analytics in the framework of Data Science. The chapter presents a brief introduction to Data Science that can be amplified by reading the previous book, an introduction to EDISON, the European Union (EU)-funded project under which the framework for Data Science, and specifically, the Data Analytics body of knowledge treated in this book, was developed. The chapter also presents an introduction to Data Analytics from the four different perspectives developed in the EDISON project, that is, the Data Analytics competences, its body of knowledge, its curriculum, and its related professional profiles. Finally, the chapter ends with a last about, in this case, about the book itself, in which the contents and structure of the book will be introduced.

About Data Science

What is Data Science?

All the content of this book has been created with the goal of providing the reader the foundational knowledge of Data Analytics, and the first notion that must be known is that Data Analytics is a part of Data Science. Consequently, to start the study of Data Analytics, we define what Data Science is.

There are multiple definitions of the data science discipline and technology that stress/put in the centre one of the four flavours/goals of data analysis:

- *Data Analytics* is a process of inspecting, transforming, and modelling data with the goal of discovering trends, patterns, or relations that describe observable real-life phenomena and can be used for informed decision-making.

J. J. Cuadrado-Gallego, Y. Demchenko, *Data Analytics*,
https://doi.org/10.1007/978-3-031-39129-3_1

- *Data Science* involves the systematic study of the structure and behaviour of data to understand past and current occurrences and predict the future behaviour of those data. Data science is an interdisciplinary field that uses scientific methods, processes, algorithms, and systems to extract knowledge and insights from structured and unstructured data.
- *Machine Learning* deals with the development of algorithms, some of which are based on statistical models, with the objective that their computational implementation allows the computer not only to carry out the tasks without supervision but also to learn the results for continuous improvement. Within machine learning, deep learning is the set of predictive methodologies that use artificial neural networks to progressively extract higher-level features from unstructured raw data. This class of methods is particularly effective for making predictions from large amounts of data generated by real-life behavioural processes or sensors. Machine learning and deep learning are considered subfields of data science focused on specific tasks, whereas data science provides a general methodology for working with a wide variety of data using different methods and tools.
- *Artificial Intelligence* is a machine or application with the capability to autonomously execute predictions from data, where prediction is made based on data science and analytics methods.

It is important to clarify the relation of data science to other closely related scientific disciplines and technology domains, such as *big data*, *artificial intelligence*, *machine learning,* and *statistics*. Despite the fact that some authors may refer to historical facts mentioning these many years ago, we refer to the current data-driven technology development that made data science a central component of all other data-related and data-driven technology developments. We identify that such technology fusion and consolidation took place in 2011–2013 with the advent of cloud computing and big data, which also aligned with the US National Institute of Standards and Technologies' (NIST) definition of cloud computing in 2011 and big data definition in 2013.

Big Data serves as a technology platform to allow the data science and analytics solutions and applications to work with modern data that are of the *big data 3 V scale*: *volume*, amount of data processed; *velocity*, speed of growth of data processed; and *variety*, number of different types of data processed. Big data technology platforms include large-scale computation, storage, and network facilities, typically cloud based, such as Hadoop, Spark, NoSQL databases, data lakes, and others.

In the whole digital economy ecosystem, data science integrates all multiple components from other scientific and technology domains to drive data-intensive research and emerging digital technology development.

Different proposals can be found in the literature to answer the question stated at the beginning of this section, but from the experience of the EDISON Data Science Framework development and with the purpose of having a brief/actionable definition to answer the question, the authors can give the following answer to the question:

What is Data Science in practice?

> Data Science is a complex discipline that uses conceptual and mathematical abstractions and models, statistical methods, together with modern computational tools to obtain knowledge and derive insight from data to (uncover correlations and causations in business data) support decision making in scientific research and business activity. (Yuri Demchenko and Juan J. Cuadrado-Gallego)

If we must define data science in only one sentence:

> Science that studies how to obtain knowledge from Data. (Juan J. Cuadrado-Gallego and Yuri Demchenko)

About the EDISON Project and Data Science Framework

This book is entitled *Data Analytics: From the EDISON Project to the Practice*, and for that reason, as we have done with the concept of Data Science, it is beneficial, before going to the knowledge of Data Analytics, to introduce what is the EDISON Project and its main result the EDISON Data Science Framework (EDSF).

The EDISON Project

The EDISON Project was the EU-funded Horizon 2020 project with Grant 675419, which was developed from 2015 until 2017, and its goal was to create the foundation for the data science profession in Europe. The EDISON project originated from the community initiative started at the Research Data Alliance (RDA), with the creation of the RDA Interest Group on Education and Training on Handling Research Data (IG-ETHRD) in 2014, and joined experts and practitioners in research data management to address the demand for data specialists that would be capable of bringing value from data explosion at that time. From its start, in September 2015, the project became involved in the European Digital Skills Initiative, which included the whole complex of activities addressing the growing demand for digital and data skills in Europe.

During its term, the EDISON project undertook multiple initiatives and organized multiple community activities/events and conducted important studies to involve data experts and practitioners from academia, research, and industry to define the foundation of the new profession of the data scientist.

The main outcome of the EDISON project was, until the publication of this book, the EDISON Data Science Framework (EDSF), which was a product of a wide professional community facilitated by the EDISON project. The project published EDSF Release 2 as its final deliverable in 2017. Since the project's end, the EDSF has been maintained by the EDISON Community Initiative, coordinated by the University of Amsterdam, which involves former project partners and numerous

contributors from academia, research, and industry. EDSF Release 3 was published in 2018, and the new EDSF Release 4 was published in 2020 and updated in 2022. This book includes all the knowledge included in the last EDSF Release 4. In particular, EDSF Release 4 includes recent contributions from the MATES project[1] on digital and data skills for Industry 4.0 and the definition of the Data Stewardship and FAIR competences developed in the FAIRsFAIR project.[2]

In addition to the multidimensional definition of the data science profession, the EDSF created a comprehensive and effective methodology that can be used for other professional domains to address multiple aspects of organizational human resources management and capacity building that include competences and skills definition and assessment, education and training, customized curriculum design, knowledge assessment and certification, individual professional development, and career path building.

The EDISON Data Science Framework (EDSF)

The EDISON Data Science Framework provides the basis for the definition of the data science profession and enables the definition of the other components related to data science education, training, organizational role definition and skills management, as well as professional certification.

Figure 1 illustrates the main components of the EDISON Data Science Framework and their interrelations that provide the conceptual basis for the development of the data science profession:

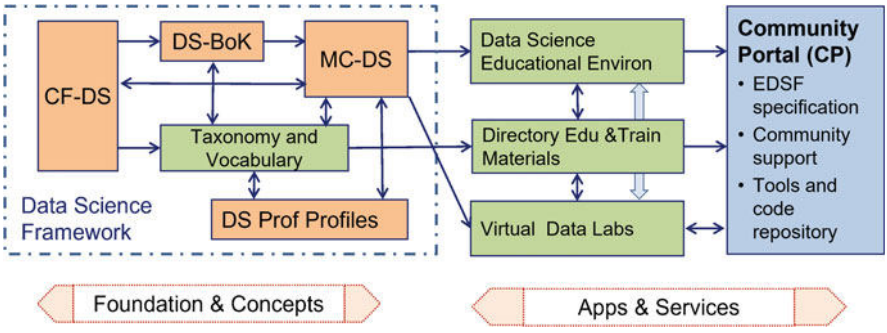


Fig. 1 EDISON Data Science Framework components

[1] Erasmus+ Project MATES (grant number 591889) –https://www.projectmates.eu/

[2] H2020 Project FAIRsFAIR (grant number 831558) –https://www.fairsfair.eu/

- Data Science Competence Framework (CF-DS). EDSF Part 1.
- Data Science Body of Knowledge (DS-BoK). EDSF Part 2.
- Data Science Model Curriculum (MC-DS). EDSF Part 3.
- Data Science Professional Profiles and Occupations Taxonomy (DSPP). EDSF Part 4.

The proposed framework provides the basis for other components of the data science professional ecosystem (defined and piloted in the EDISON project and constituting the project legacy that can be reused and followed by the community), such as:

- Data Science Education Environment (DSEE)
- Directory of Education and Training Materials
- Virtual Data Labs (templates)
- Data Science Community Portal (DSCP), which provides community support and contains essential community-maintained information about EDSF, code repository, and tools for curriculum design and competences assessment

The *Competences Framework for Data Science (CF-DS)* provides the overall basis for the whole EDSF. The core CF-DS includes common competences required for the successful work of data scientists in different work environments in industry and in research and through the whole career path. The future CF-DS development will include coverage of the domain-specific competences and skills and will involve domain and subject matter experts.

The *Data Science Body of Knowledge (DS-BoK)* defines the knowledge areas for building data science curricula that are required to support identified data science competences. DS-BoK is organized by knowledge area groups (KAGs) that correspond to the CF-DS competence groups. Each KAG is composed of knowledge areas (KAs). Each KA is composed of a number of knowledge units (KUs), which are currently the lowest component of the DS-BoK. DS-BoK incorporates best practices in computer science and domain-specific bodies of knowledge and includes KAs and KUs defined, where possible, based on the classification of computer science components taken from other bodies of knowledge and proposes new KAs/KUs to incorporate new technologies used in data science and their recent developments.

The *Model Curriculum for Data Science (MC-DS)* is built based on CF-DS and DS-BoK, where learning outcomes (LOs) are defined based on CF-DS competences and learning units (LUs) are mapped to knowledge units in DS-BoK. Three mastery, or proficiency, levels are defined for each learning outcome to allow for flexible curricula development and profiling for different data science professional profiles. The proposed learning outcomes are enumerated to have a direct mapping to the enumerated competences in CF-DS.

The *Data Science Professional Profiles (DSPP)* are defined as an extension to the European Skills, Competences, Occupations, and Qualifications (ESCO) taxonomy using the ESCO top classification groups. The DSPP definition provides an important instrument to define effective organizational structures and roles related to data

science positions (e.g., building data science teams) and can also be used for building individual career paths and corresponding competences and skills transferability between organizations and sectors.

The Data Science Taxonomy and Scientific Disciplines Classification ensures consistency between four core components of EDSF: CF-DS, DS-BoK, MC-DS, and DSPP. To ensure consistency and linking between EDSF components, all individual elements of the framework are enumerated, in particular competences, skills, and knowledge topics in CF-DS; knowledge groups, areas, and units in DS-BoK; learning outcomes and learning units in MC-DS; and professional profiles in DSPP.

The EDISON data science professional ecosystem illustrated in Fig. 1 uses core EDSF components to specify the potential services that can be offered for the professional data science community and provide the basis for the sustainable data science competences and skills management by organizations, in particular in conditions of emerging Industry 4.0, growing digitalizations and artificial intelligence development. As an example of practical use, CF-DS and DS-BoK can be used for individual competences and knowledge benchmarking and play an instrumental role in constructing personalized learning paths and professional (up/re) skilling programmes based on MC-DS.

The recent EDSF Release 4 is the result of cooperation and contribution by the wide community of academicians, researchers, and practitioners that are practically involved in data science and data analytics education and training, competences and skills management in organizations, and standardization in the area of competences, skills, occupations, and digital technologies.

The EDSF provides the conceptual basis for the data science profession definition, targeted education and training, professional certification, organizational capacity building and organization and individual skills management and career transferability.

The EDSF Part 5 document, part of the EDSF2020 Release, defines the EDSF use cases and applications:

- Digital competences and data literacy training
- Data science competences analysis and curriculum design
- Assessment of individual and team competences, as well as balanced data science team composition
- Development of the tailored curriculum for academic education or professional training to bridge the skills gap and staff up/reskilling

The EDSF Part 5 is intended to provide guidance and the basis for universities, training organizations, data management and data steward teams, and practitioners to define their data science curricula and course selection, on the one hand, and for companies to better define a set of required competences and skills for their specific industry domain in their search for data science talents, on the other hand.

About Data Analytics

Once we have introduced what Data Science and the EDISON Project are, we will introduce extensively in this section what Data Analytics are. We are going to do this using the EDISON Data Science Framework, EDSF, defined by the EDISON Project. We start with the introduction of the Data Analytics competences defined in the EDSF, and from that, we will introduce the Data Analytics Body of Knowledge. Once we know both aspects, we will introduce the Data Analytics Model Curriculum to finalize with the Data Analytics professional profiles. We are going to use the EDSF to introduce the Data Analytics subject in two ways: first, to use it as a way to introduce the main related knowledge, from the definition of the competences to the definition of the professional profiles; and second, as the source for that knowledge, that is, for example, the data analytics body of knowledge introduced later has been obtained from the Data Science Body of Knowledge defined in the EDSF.

Data Analytics Competences

Before defining the Data Analytics Competences, we will establish what the concepts of Data Science Competences, Skills, and Knowledge are within the framework of the EDISON Project.

The competences definition in the EDSF has a strong foundation and roots in the existing frameworks and best practices that were used for defining the proposed set of data science competences and skills. In particular, similar to e-CF 3.0, the CF-DS is defined as a three-dimensional model with dimensions: *competences*, *skills,* and knowledge. The relation between them is illustrated in Fig. 2.

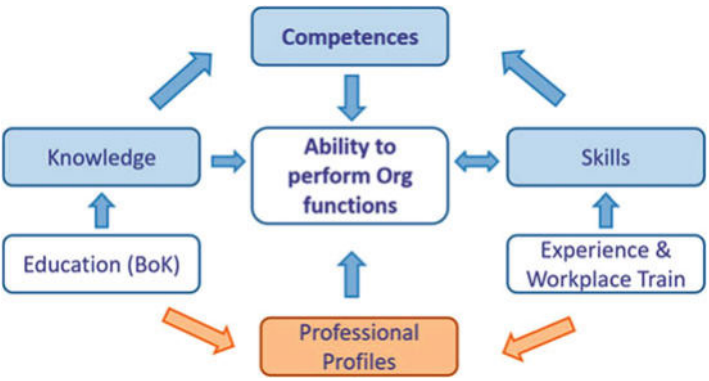


Fig. 2 Relation between competences, skills, knowledge, and education

Competences ensure the ability to perform required organizational functions that are defined for a specific organizational role that the worker/employee performs in the organization. Competences must be supported by knowledge acquired in the process of education or training and by specific skills that are acquired/obtained as a result of practical activity or previous work in a similar role or profession. Knowledge and skills add to the ability and performance of organizational functions.

CF-DS adopts a holistic e-CF definition: Competence is a demonstrated ability to apply knowledge, skills, and attributes to achieve desirable results in organizational or role contexts. CF-DS should work as an enabler for multiple applications that can be used by different types of users from individual to organizational; it should support common understanding and not mandate specific implementation.

In the following, the three dimensions of the CF-DS are explained:

- The first dimension of CF-DS is *Competences*. The following CF-DS five competence and skills groups have been identified:

  1. *DSDA, Data analytics*. This group includes statistical analysis, machine learning, data mining, business analytics, and others.
  2. *DSENG, Data engineering*. This group includes software and applications engineering, data warehousing, big data infrastructure, and tools.
  3. *DSDM, Data management and governance*. This group includes data stewardship, curation, and preservation.
  4. *DSRMP, Research methods and project management*. This group includes research methods and project management for research-related professions and business process management for business-related professions.
  5. *DSDK, Domain-specific knowledge and expertise* (subject/scientific domain related). Additionally, will be named, indistinctly, business analytics, DSDA. This group includes domain-specific knowledge and expertise.

  DSDA, DSENG, and DSDM competence groups constitute the core data science competences that actually define the main Data Science Professional profiles and roles, including those related to different application domains. DSDM and DSRMP competence groups are considered commonly required for all Data Science Professional Profiles to ensure effective work with modern data-driven technologies and in modern data-driven organizations. Data management, curation, and preservation competences are already attributed to the existing (research) data-related professions such as data stewards, data managers, data librarians, data archivists, and others. Data management is an important component of the European research area and open data and open access policies. It is extensively addressed by the Research Data Alliance (RDA) and supported by numerous projects, initiatives, and training programmes.

  DSRMP Knowledge of the research methods and techniques is something that makes the data scientist profession different from all previous professions. It should also be coupled with basic project management competences and skills. The research methods typically include the following stages: 1. design

experiment; 2. collect data; 3. analyse data; 4. identify patterns; 5. hypothesis explanation; 6. test hypothesis.

The reason of the DSDK (DSBA) Knowledge is based on the fact that an important part of the research process is theory building, but this activity is attributed to the domain or subject matter researcher. The data scientist (or related role) should be aware of domain-related research methods and theory as a part of their domain-related knowledge and team or workplace communications. There are a number of business process operations models depending on their purpose, but typically, they contain the following stages that are generally similar to those for scientific methods, in particular in collecting and processing data: 1. Design; 2. Model/plan; 3. Deploy and execute; 4. Monitor and control; 5. Optimize and redesign.

The identified demand for general competences and knowledge on data management and research methods needs to be implemented in future data science education and training programmes, as well as to be included in reskilling training programmes. It is important to mention that knowledge of research methods does not mean that all data scientists must be talented scientists; however, they need to know the general research methods, such as formulating hypotheses, applying research methods, producing artefacts, and evaluating hypotheses (the so-called 4-step model). Research methods training is already included in master's programmes and graduate students of many master's programmes.

From the education and training point of view, the identified competences can be treated or linked to the expected learning or training outcome. This aspect is discussed below in relation to the definition of the Data Science Body of Knowledge and Data Science Model Curriculum. The five identified data science-related competence groups provide the basis for defining consistent and balanced education and training programmes for data science-related jobs, reskilling, and professional certification.

The proposed data science competences definition in the EDISON Project for different groups are supported by the data extracted from the job market analysis for the demanded competences, skills, knowledge, and attitude. The presented competences definition has been reviewed by a number of expert groups and individual experts as a part of the project EDISON engagement and network activities. The presented competences are required for different professional profiles, organizational roles and throughout the whole data life cycle but do not need to be provided by a single role or individual. The presented competences are enumerated to allow easy use and linking between the parts of the data science framework: CF-DS, DS-BoK, MC-DS, and DSPP.

- The second dimension of the CF-DS is *Skills*. The identified skills can be organized into the following three groups:

  1. *Group A Skills*. They refer to data science skills related to the main competence groups that cover knowledge and experience related to effectively realizing defined competences and related organizational functions. The

identified data science skills associated with the main competence groups are as follows:

1.1 *Data analytics skills*. Covering extensive skills related to using different machine learning, data mining, statistical methods, and algorithms
1.2 *Data engineering skills*. Related to design, implementation, and operation of the data science (or big data) infrastructure, platforms, and applications
1.3 *Data management and governance skills.* Including both general data management and research data management)
1.4 *Research methods and project management skills*.
1.5 *Business analytics as an example of domain-related skills*.

The data analytics and data engineering groups are the most populated, which reflect a wide spectrum of required skills in these groups as a core for data science because it is mandatory for the data scientist to have the ability to implement effective data analytics solutions and applications. In addition, it is important to mention that the whole complex of data science-related competences, skills, and knowledge is strongly based on the mathematical foundation that should include knowledge of mathematics, including linear algebra, calculus, statistics, probability theory, and other mathematical subjects.

2. *Group B Skills*. They refer to data analytics and data handling languages, tools, platforms, and applications, including SQL- and NoSQL-based applications and data management tools and knowledge and experience with big data infrastructure platforms and tools.
3. *Transversal Skills*. Separately defined are personal and attitude skills, also referred to as transversal, the twenty-first-century skills and data science professional skills that define specific (personal) skills that the data scientist needs to develop to successfully work as a data scientist in different organizational roles along their career.

- The third dimension of the CF-DS is knowledge topics. Knowledge or knowledge topics are the required knowledge to support corresponding competence groups. There is no direct mapping between individual competences and knowledge topics; a single competence may be mapped to multiple knowledge topics and vice versa. CF-DS provides mapping between knowledge topics defined for individual competences and knowledge units defined in DS-BoK.

After introducing three dimensions of competence in the EDSF, we are now going to see the application of the model over the specific subject of this book, that is, we are going to introduce the Data Analytics Competences, Knowledge, and Skills (DSDA):

- Data Analytics (DSDA) Competences

  Data analytics competences deal with the use of appropriate data analytics and statistical techniques on available data to discover new relations and deliver insights into research problems or organizational processes and support

decision-making and cover extensive skills related to using different machine learning, data mining, statistical methods, and algorithms. The following are the six DSDA identified competences:

1. DSDA01. Effectively use a variety of data analytics techniques, such as machine learning (including supervised, unsupervised, and semi-supervised learning), data mining, prescriptive and predictive analytics, for complex data analysis through the whole data life cycle.
2. DSDA02. Apply designated quantitative techniques, including statistics, time series analysis, optimization, and simulation, to deploy appropriate models for analysis and prediction.
3. DSDA03. Identify, extract, and pull together available and pertinent heterogeneous data, including modern data sources such as social media data, open data, and governmental data, and verify data quality.
4. DSDA04. Understand and use different performance and accuracy metrics for model validation in analytics projects, hypothesis testing, and information retrieval.
5. DSDA05. Develop required data analytics for organizational tasks, integrate data analytics and processing applications into organizational workflows and business processes to enable agile decision-making.
6. DSDA06. Visualize the results of data analysis, design dashboards, and use storytelling methods.

- Data Analytics Knowledge Topics (KDSDA)

  The following are the eighteen data science and data analytics knowledge (KDSDA) required to support the identified competences in this subject:

  1. KDSDA01. Machine learning supervised: decision trees, naïve Bayes classification, ordinary least square regression, logistic regression, neural networks, SVM (support vector machine), ensemble methods, and others.
  2. KDSDA02. Machine learning unsupervised: clustering algorithms, principal component analysis (PCA), singular value decomposition (SVD), independent component analysis (ICA)
  3. KDSDA03. Machine learning (reinforced): Q-learning, TD-learning, genetic algorithms)
  4. KDSDA04. Data mining (text mining, anomaly detection, regression, time series, classification, feature selection, association, clustering)
  5. KDSDA05. Text data mining: statistical methods, NLP, feature selection, a priori algorithm, etc.
  6. KDSDA06. General statistical analysis methods and techniques, descriptive analytics
  7. KDSDA07. Quantitative analytics
  8. KDSDA08. Qualitative analytics
  9. KDSDA09. Predictive analytics
  10. KDSDA10. Prescriptive analytics

11. KDSDA11. Graph data analytics: path analysis, connectivity analysis, community analysis, centrality analysis, subgraph isomorphism, etc.
12. KDSDA12. Natural language processing
13. KDSDA13. Data preparation and preprocessing
14. KDSDA14. Performance and accuracy metrics
15. KDSDA15. Markov models, conditional random fields
16. KDSDA16. Operations research
17. KDSDA17. Optimization
18. KDSDA18. Simulation

Data Science Analytics skills include three groups: skills group A, which are related to competences; skills group B, which includes skills related to practical skills related to using computational platforms, programming languages, and tools; and transversal skills, which include data science professional skills and workplace skills. These three groups of skills are described below in detail.

- Data Analytics Skills Group A (SDSDA)

The following are the sixteen data science and data analytics identified Group A skills, SDSDA:

1. SDSDA01. Use machine learning technology, algorithms, tools, including supervised, unsupervised, or reinforced learning.
2. SDSDA02. Use data mining techniques.
3. SDSDA03. Use text data mining techniques.
4. SDSDA04. General statistical analysis methods and techniques, descriptive analytics.
5. SDSDA05. Use quantitative analytics methods.
6. SDSDA06. Use qualitative analytics methods.
7. SDSDA07. Apply predictive analytics methods.
8. SDSDA08. Apply prescriptive analytics methods.
9. SDSDA09. Use graph data analytics for organizational network analysis, customer relations, and other tasks.
10. SDSDA10. Analytics and statistical methods were applied for data preparation and preprocessing.
11. SDSDA11. Be able to use performance and accuracy metrics for data analytics assessment and validation.
12. SDSDA12. Use effective visualization and storytelling methods to create dashboards and data analytics reports.
13. SDSDA13. Use natural language processing methods.
14. SDSDA14. Operations research.
15. SDSDA15. Optimization.
16. SDSDA16. Simulation.

- Data Analytics Skills Group B

  Group B skills are common practical skills related to using computational and data management platforms, programming languages, and tools. Group B skills are all related to data analytics and data handling languages, tools, platforms, and applications, including SQL- and NoSQL-based applications and data management tools and knowledge and experience with big data infrastructure platforms and tools.

  The identified skills related to the data analytics languages, tools, platforms, and big data infrastructure are split into six subgroups. The groups and their associated skills are as follows:

  1. Data analytics and statistical languages and tools
  2. Databases and query languages
  3. Data/application visualization
  4. Data management and curation platform
  5. Big data analytics platforms
  6. Development and project management frameworks, platforms, and tools

  The Data Analytics and statistical languages and tools skills group (DSDALANG) includes popular languages and tools for data analytics. It is important to know many of them even if someone is dedicated only to Data Analytics, but specifically, the following are the twelve data analytics and statistical languages and tools Group B skills, DSDALANG:

  1. DSDALANG01. R and data analytics libraries (CRAN, ggplot2, dplyr, reshap2, etc.)
  2. DSDALANG02. Python and data analytics libraries (pandas, pandas profiling, numpy, mathplotlib, scipy, scikit-learn, seaborn, beautifulsoup4, etc.)
  3. DSDALANG03. SAS
  4. DSDALANG04. IBM SPSS
  5. DSDALANG05. Julia
  6. DSDALANG06. RapidMiner
  7. DSDALANG07. Other analytics, statistical, and programming languages (WEKA, KNIME, Scala, Stata, Orange, etc.)
  8. DSDALANG08. Scripting language, e.g., Octave, PHP, Pig, HiveQL, others
  9. DSDALANG09. MATLAB data analytics
  10. DSDALANG10. Analytics tools (R/R Studio, Python/Anaconda, SPSS, MATLAB, etc.)
  11. DSDALANG11. Data mining tools: RapidMiner, Orange, R, WEKA, NLTK, and others
  12. DSDALANG12. Excel data analytics (Analysis ToolPack, PivotTables, etc.)

Among the six skills groups listed above, the data/application visualization group and some of the big data analytics platform group B skills are also relevant to data analytics.

The following are essential data/application visualization skills (DSVIZ):

1. DSVIZ01. Data visualization libraries (mathpoltlib, seaborn, D3.js, FusionCharts, Chart.js, and others)
2. DSVIZ02. Visualization software (D3.js, Processing, Tableau, Raphael, Gephi, etc.)
3. DSVIZ03. Online visualization tools (Datawrapper, Google Visualization API, Google Charts, Flare, etc.)

The EDSF defines the following big data analytics platform skills related to data analytics (DSBDA):

1. DSBDA05. Azure data analytics platforms (HDInsight, Data Lake Analytics, PowerBI, Team Data Science Process/MLOps, Machine Learning Studio, etc.)
2. DSBDA06. The Amazon Data Analytics platform (SageMaker, EMR, Kinesis, Data Pipeline, Machine Learning Services and tools, etc.)
3. DSBDA07. Google Analytics platform (Google Data Studio, Machine Learning, TensorFlow, others)
4. DSBDA09. Other cloud-based data analytics platforms (Cloudera/HortonWorks Data Platform, Vertica, LexisNexis HPCC System, etc.)
5. DSBDA10. Cognitive platforms (such as IBM Watson, Microsoft Cortana, and others)
6. DSBDA11. Kaggle competition, resources, and community platform

It is also important for data scientists to be familiar with multiple data analytics languages and demonstrate proficiency in one or a few of the most popular languages (which should be supported with several years of practical experience), such as:

- R, including extensive data analysis libraries
- Python and related data analytics libraries
- Julia
- SPSS
- KNIME, Orange, WEKA, and others

Finally, referring to Group B skills, any data science practitioner and, consequently, a data analytics practitioner must be familiar and have experience with general programming languages, software versioning, and project management environments such as the following:

- Java, JavaScript and/or C/C++ as general application programming languages
- Git versioning system as a general platform for software development
- Scrum agile software development and management methodology and platform, in particular, applied to Data Science projects MLOps and DataOps supported by major Big Data platforms.

  It is essential to mention that all modern big data platforms and general data storage and management platforms are cloud based. The knowledge of cloud computing and related platforms for application deployment and data management are included in the table. The use of cloud-based data analytics tools is

growing, and most large cloud service providers provide whole suites of platforms and tools for enterprise data management from enterprise data warehouses, data backup and archiving to business data analytics, data visualization, and content streaming.

- Data Analytics Transversal Skills (Applied to Data Science Analytics)

  It is commonly agreed on the importance of soft skills for data scientists and, consequently, for data analysts because a data analyst is a specific data scientist. The job market analysis clearly confirmed the importance of workplace and attitude skills and identified a number of specific data science professional skills that are required for data scientists to effectively work in modern agile data-driven organizations and project teams. These should also be complemented with general professional skills referred to as twenty-first-century skills. The importance of such skills for data scientists, or data analysts, is defined by their cross-organizational functions and responsibilities in collecting and analysing organizational data to provide insight for decision-making.

  In such a role, the data scientist often reports to the executive level or to other departments and teams. These skills extend beyond traditionally required communication or team skills. In addition, the ideal data scientist is expected to bring and spread new knowledge to the organization and contribute to the processes related to data collection, analysis, and exploitation. Consequently, the main two transversal skills and their associated skills are as follows:

- Data Science Professional or Attitude Skills (DSPS) (Thinking and Acting Like a Data Scientist)

  Data science is growing as a distinct profession and consequently will need professional identification via the definition of the specific professional skills and code of conduct that can be defined as "Thinking and acting like Data Scientist". Understanding, recognizing, and acquiring such skills are essential for the data scientist to successfully progress along their career. It is also important for team leaders to correctly build relations in the team or the project group. Below are listed the data science professional (or attitude) skills (DSPS) that are identified by the data science practitioners and educators. Although some of the skills are common to the twenty-first-century skills, it is important to provide the whole list of skills that can provide guidance for future data scientists regarding what skills are expected from them and need to be developed along their careers. The fifteen skills are as follows:

  1. DSPS01. Accept/be ready for iterative development, know when to stop, comfortable with failure, accept the symmetry of outcome (both positive and negative results are valuable).
  2. DSPS02. Ask the right questions.
  3. DSPS03. Recognize what things are important and what things are not important.
  4. DSPS04. Respect domain/subject matter knowledge in the area of data science.
  5. DSPS05. Data-driven problem solver and impact-driven mindset.

6. DSPS06. Recognize value of data, work with raw data, exercise good data intuition.
7. DSPS07. Good sense of metrics, understand importance of the results validation, never stop looking at individual examples.
8. DSPS08. Be aware of the power and limitations of the main machine learning and data analytics algorithms and tools.
9. DSPS09. Understand that most of data analytics algorithms are statistics- and probability-based, so any answer or solution has some degree of probability and represents an optimal solution for a number of variables and factors.
10. DSPS10. Working in an agile environment and coordinating with other roles and team members.
11. DSPS11. Work in a multidisciplinary team, ability to communicate with the domain and subject matter experts.
12. DSPS12. Embrace online learning, continuously improve your knowledge, use professional networks and communities.
13. DSPS13. Storytelling: Deliver actionable result of your analysis.
14. DSPS14. Attitude: Creativity, curiosity (willingness to challenge the status quo), commitment to finding new knowledge and progress to completion.
15. DSPS15. Ethics and responsible use of data and insight delivered, awareness of dependability (data scientist is a feedback loop in data-driven companies).

- Twenty-First-Century Skills (SK21) (Aka "Soft" Skills)

  Twenty-first-century skills comprise a set of general workplace skills that include critical thinking, creativity, communication, collaboration, organizational awareness, ethics, and others. The importance of this kind of skill is motivated by fast technology development and the ongoing digital transformation of the modern economy and Industry 4.0.

  Below are listed the twenty-first-century skills (SK 21) defined based on the recommendations of the DARE Project, OECD Report on industry digitalization, and P21's Framework for twenty-first-century learning.

  1. SK21C01. Critical Thinking: Demonstrating the ability to apply critical thinking skills to solve problems and make effective decisions.
  2. SK21C02. Communication: Understanding and communicating ideas.
  3. SK21C03. Collaboration: Working with others, appreciation of multicultural difference.
  4. SK21C04. Creativity and attitude: Deliver high-quality work and focus on final results, initiative, and intellectual risk.
  5. SK21C05. Planning and organizing: Planning and prioritizing work to manage time effectively and accomplish assigned tasks.
  6. SK21C06. Business fundamentals: Having fundamental knowledge of the organization and the industry.
  7. SK21C07. Customer focus: Actively look for ways to identify market demands and meet customer or client needs.

8. SK21C08. Working with tools and technology: Selecting, using, and maintaining tools and technology to facilitate work activity.
9. SK21C09. Dynamic (self-) reskilling: Continuously monitor individual knowledge and skills as a shared responsibility between employer and employee, ability to adapt to changes.
10. SK21C10. Professional network: Involvement and contribution to professional network activities.
11. SK21C11. Ethics: Adhere to high ethical and professional norms, responsible use of power data-driven technologies, avoid and disregard unethical use of technologies and biased data collection and presentation.

Data Analytics Body of Knowledge

Once we defined the competences that a Data Analyst must have, the next step is to introduce the Body of Knowledge defined in the EDSF for Data Analytics as a part of the whole Data Science Body of Knowledge, DS-BoK.

DS-BoK has been developed with three main objectives: 1. support the competence groups defined in the Competences Framework for Data Science (CF-DS) presented in the previous subsection; 2. reflect the data-lifecycle management where different organizational roles, functions, competences, and knowledge are needed; and 3. ensure knowledge transferability and education programme compatibility. Extending this third objective, the DS-BoK can also be used as the basis for defining data science-related curricula, courses, instructional methods, educational/course materials, and necessary practices for university undergraduate and postgraduate programmes and professional training courses.

The DS-BoK is also intended to be used for defining certification programmes and certification exam questions. Although the CF-DS (comprising competences, skills, and knowledge) can be used for defining job profiles (and correspondingly the content of job advertisements), the DS-BoK can provide the basis for interview questions and evaluation of the candidate's knowledge and related skills, as well as for professional certification exams and training.

The DS-BoK is organized into knowledge area groups, KAGs, each one of them constituted by Knowledge Areas, and each Knowledge Area, KA, is formed by Knowledge Units, KUs. The DS-BoK contains the following five knowledge area groups (KAGs) that follow the competence groups defined in the previous subsection:

1. KAG1-DSDA, Data Analytics Area Group.
2. KAG2-DSENG, Data Engineering Area Group.
3. KAG3-DSDM, Data Management Area Group.
4. KAG4-DSRMP, Research methods and project management for research-related professions and business process management for business-related professions Area Group.

5. KAG5-DSBA, Business Analytics. This subject domain-related knowledge group (scientific or business) KAG*-DSBA is recognized as essential for the practical work of data scientists, which in fact means not professional work in a specific subject domain but understanding the domain-related concepts, models, and organization and corresponding data analysis methods and models. These knowledge areas will be a subject for future development in tight cooperation with subject domain specialists.

For the purpose of this book, we will look closer at the DS-BoK part related to Data Analytics, which is defined as the Data Analytics Knowledge Area Group, KAG1-DSDA.

The KAG1-DSDA Data Analytics Knowledge Area Group is key and distinguishes KAG for DS-BoK. It includes different methods and algorithms, primarily statistical, machine learning, and data mining, to enable data processing, modelling, analysis, and inspection with the goal of discovering useful information, providing insight and recommendations, and supporting decision-making. The following are the six commonly defined data science analytics knowledge areas, KAs:

1. KA01.01 (DSDA.01/SMA) Statistical methods for data analysis
2. KA01.02 (DSDA.02/ML) Machine learning
3. KA01.03 (DSDA.03/DM) Data mining
4. KA01.04 (DSDA.04/TDM) Text data mining
5. KA01.05 (DSDA.05/PA) Predictive analytics
6. KA01.06 (DSDA.06/MSO) Computational modelling, simulation, and optimization

We are going now to enumerate the Knowledge Units, KUs, in each one of the Knowledge Areas, KAs, in the KAG1-DSDA Data Analytics Knowledge Area Group.

- KA01.01 (DSDA.01/SMA) Statistical methods for data analysis. Starting with an initial KU about a general overview and main concepts, the sixteen suggested specific knowledge units, KUs, for statistical methods knowledge are as follows:

  0. KU1.01.00. General overview and main concepts in statistical methods for data analysis.
  1. KU1.01.01. Probability and statistics.
  2. KU1.01.02. Statistical paradigms (regression, time series, dimensionality, clusters).
  3. KU1.01.03. Probabilistic representations (causal networks, Bayesian analysis, Markov nets).
  4. KU1.01.04. Frequentist and Bayesian statistics.
  5. KU1.01.05. Probabilistic reasoning.
  6. KU1.01.06. Exploratory and confirmatory data analysis.
  7. KU1.01.07. Quantitative analytics.
  8. KU1.01.08. Qualitative analytics.
  9. KU1.01.09. Data preparation and preprocessing.

10. KU1.01.10. Performance analysis.
11. KU1.01.11. Markov models, Markov networks.
12. KU1.01.12. Operations research.
13. KU1.01.13. Information theory.
14. KU1.01.14. Discrete mathematics and graph theory.
15. KU1.01.15. Mathematical analysis.
16. KU1.01.16. Mathematical software and tools.

- KA01.02 (DSDA.02/ML) Machine learning knowledge area. Machine learning and related methods for information search, image recognition, decision support, classification. Starting with an initial KU about a general overview and main concepts, the thirteen suggested specific knowledge units, KU, for the machine learning methods knowledge are:

0. KU1.02.00. General overview and main concepts in machine learning
1. KU1.02.01. Machine learning theory and algorithms
2. KU1.02.02. Supervised machine learning
3. KU1.02.03. Unsupervised machine learning
4. KU1.02.04. Reinforced learning
5. KU1.02.05. Classification methods
6. KU1.02.06. Design and analysis of algorithms
7. KU1.02.07. Game theory and mechanism design
8. KU1.02.08. Artificial intelligence
9. KU1.01.02. Statistical paradigms (regression, time series, dimensionality, clusters)
10. KU1.01.03. Probabilistic representations (causal networks, Bayesian analysis, Markov nets)
11. KU1.01.04. Frequentist and Bayesian statistics
12. KU1.01.05. Probabilistic reasoning
13. KU1.01.08. Performance analysis

- KA01.03 (DSDA.03/DM) Data mining knowledge area. It is a particular data analysis technique that focuses on modelling and knowledge discovery for predictive rather than purely descriptive purposes. Starting with an initial KU about a general overview and main concepts, the thirteen suggested specific knowledge units (Kus) for the data mining knowledge are as follows:

0. KU1.03.00. General overview and main concepts in data mining
1. KU1.03.01. Data mining and knowledge discovery
2. KU1.03.02. Knowledge representation and reasoning
3. 4. KU1.03.03. CRISP-DM and data mining stages
4. KU1.03.04. Anomaly detection
5. KU1.03.05. Time series analysis
6. KU1.03.06. Feature selection, a priori algorithm
7. KU1.03.07. Graph data analytics
8. KU1.01.08. Performance analysis
9. KU1.02.01. Machine learning theory and algorithms

10. KU1.02.02. Supervised machine learning
11. KU1.02.03. Unsupervised machine learning
12. KU1.02.04. Reinforced learning
13.KU1.02.05. Classification methods

- KA01.04 (DSDA.04/TDM) Text analytics applies statistical, linguistic, and structural techniques to extract and classify information from textual sources, a species of unstructured data. Starting with an initial KU about a general overview and main concepts, the seven suggested specific knowledge units (KUs) for text data mining knowledge are as follows:

  0. KU1.04.00. General overview and main concepts in text data mining
  1. KU1.04.01. Text analytics including statistical, linguistic, and structural techniques to analyse structured and unstructured data
  2. KU1.04.02. Data mining and text analytics
  3. KU1.04.03. Natural language processing
  4. KU1.04.04. Predictive models for text
  5. KU1.04.05. Retrieval and clustering of documents
  6. KU1.04.06. Information extraction
  7. KU1.04.07. Sentiment analysis

- KA01.05 (DSDA.05/PA) Predictive analytics knowledge area. It focuses on the application of statistical models for predictive forecasting or classification. Starting with an initial KU about a general overview and main concepts, the seven suggested specific knowledge units (KU) for predictive analytics knowledge are as follows:

  0. KU1.05.00. General overview and main concepts in predictive analytics
  1. KU1.05.01. Predictive modelling and analytics
  2. KU1.05.02. Inferential and predictive statistics
  3. KU1.05.03. Machine learning for predictive analytics
  4. KU1.05.04. Regression and multianalysis
  5. KU1.05.05. Generalized linear models
  6. KU1.05.06. Time series analysis and forecasting
  7. KU1.05.07. Deploying and refining predictive models

- KA01.06 (DSDA.06/MSO) Computational modelling, simulation, and optimization knowledge area. Starting with an initial KU about a general overview and main concepts, the five suggested specific knowledge units (KUs) for business analytics and business intelligence knowledge are as follows:

  0. KU1.06.00. General overview and main concepts in computational modelling, simulation, and optimization
  1. KU1.06.01. Modelling and simulation theory and techniques (general and domain-oriented)
  2. KU1.06.02. Operations research and optimization
  3. KU1.06.03. Large-scale modelling and simulation systems
  4. KU1.06.04. Network optimization
  5. KU1.06.05. Risk simulation and queueing

The above presented enumerated Knowledge Units for each Knowledge Area of the Data Analytics Knowledge Area Group reused where possible the ACM Computing Classification System, ACM CCS2012, providing extension where necessary based on the existing university curricula and new scientific disciplines definitions. That is, we will present/describe the subset of the ACM CCS2012 taxonomy that provided the initial structure for the DS-BoK that was further extended with a full set of knowledge areas and knowledge units related to data science that in return can be partly mapped to ACM CCS2012. The subset of ACM CCS2012 classification defined below can provide the basis for future ACM CCS2012 extension with a new classification group related to data science and individual disciplines that are missing in the current ACM-IEEE classification.

The ACM CCS2012 has been developed as a polyhierarchical ontology that can be utilized in semantic web applications. It replaces the traditional 1998 version of the ACM CCS, which has served as the de facto standard classification system for the computing field for many years (it has also been more human-readable). The ACM CCS2012 is being integrated into the search capabilities and visual topic displays of the ACM Digital Library. It relies on a semantic vocabulary as the single source of categories and concepts that reflect the state of the art of the computing discipline and is receptive to structural change as it evolves in the future. ACM provides a tool within the visual display format to facilitate the application of 2012 CCS categories to forthcoming papers and a process to ensure that the CCS stays current and relevant. However, at the moment, none of the data science, big data or data-intensive science technologies are reflected in the ACM classification.

The following is an extraction of the relevant classification facets from ACM CCS2012 related to data science, which reflects the multisubject area nature of data science. As an example, cloud computing, which is also a new technology and closely related to big data technologies, is currently classified in ACM CCS2012 into 3 groups:

- Networks:: Network services:: Cloud computing
- Computer systems organization:: Architectures:: Distributed Architectures:: Cloud computing
- Software and its engineering:: Software organization and properties:: Software systems structures:: Distributed systems organizing principles:: Cloud computing

Taxonomy is required to consistently present information about scientific disciplines and knowledge areas related to data science. Taxonomy is an important component to link components such as data science competences and knowledge areas, body of knowledge, and corresponding academic disciplines. From a practical point of view, taxonomy includes the vocabulary of names (or keywords) and the hierarchy of their relations.

The presented ACM CCS2012 subsets/subtrees contain scientific disciplines related to three data science knowledge area groups as defined in DS-BoK:

- KAG1-DSDA: Data analytics group including machine learning, statistical methods, and business analytics

- KAG2-DSENG: Data science engineering group including software engineering and infrastructure engineering
- KAG3-DSDM: Data management group including data curation, preservation, and data infrastructure

Two other groups, KAG4-DSRMP, research methods and project management, and KAG5-DSDK, do not have a direct mapping to ACM CCS2012, and their taxonomies are defined based on other domain-specific bodies of knowledge. It is important to note that ACM CCS2012 provides a top-level classification entry "Applied Computing" that can be used as an extension point for the domain-related knowledge area group KAG5-DSDK.

The following approach was used when constructing the proposed taxonomy:

- ACM CCS2012 provides almost full coverage of data science-related knowledge areas or disciplines related to KAG1, KAG2, and KAG3. The following top-level classification groups are used:
  - Theory of computation
  - Mathematics of computing
  - Computing methodologies
  - Information systems
  - Computer systems organization
  - Software and its engineering
- Each of the KAGs includes subsets from a few ACM CCS2012 classification groups to cover theoretical, technology, engineering, and technical management aspects. Extension points are suggested for possible future extensions of related KAGs together with their hierarchies.
- KAG3-DSDM: The data management group is extended with new concepts and technologies developed by the Research Data Alliance community and documented in community best practices.

In the following lists, the ACM CCS2012 classification facets related to data analytics grouped by DS-BoK knowledge area groups and knowledge areas are presented. The ACM CCS2012 Subjects used to develop the DS-BoK Data science analytics-related scientific subjects from CCS2012 are as follows:

- CCS2012: Computing methodologies
- CCS2012: Mathematics of computing

For each Data Analytics Knowledge Area Group the knowledge areas are:

1. Statistical Methods Knowledge Area. Data science statistical methods related to scientific subjects from CCS2012 are as follows:
   - Mathematics of computing
     - Discrete mathematics

       Graph theory
       Probability and statistics

Probabilistic representations
Probabilistic inference problems
Probabilistic reasoning algorithms
Probabilistic algorithms

- Statistical paradigms
- Mathematical software
- Information theory
- Mathematical analysis

2. Machine Learning Methods Knowledge Area. Data science machine learning methods related to scientific subjects from CCS2012 are as follows:
   For KU1.02.00 to KU1.02.08:

- Computing methodologies
  - Artificial intelligence

    Machine learning
    Learning paradigms

  - Supervised learning
  - Unsupervised learning
  - Reinforcement learning
  - Multitask learning
  - Machine learning approaches

    Machine learning algorithms

   For KU1.01.02, KU1.01.03, KU1.01.04, KU1.01.05, and KU1.01.08:

- Theory of computation
  - Design and analysis of algorithms

    Data structure design and analysis

  - Theory and algorithms for application domains

    Machine learning theory
    Algorithmic game theory and mechanism design

  - Semantics and reasoning

3. Data Mining Knowledge Area
   Data Science data mining-related scientific subjects from CCS2012 are as follows:

- Theory of computation
  - Design and analysis of algorithms

    Data structure design and analysis

  - Theory and algorithms for application domains

    Machine learning theory
    Algorithmic game theory and mechanism design

  - Semantics and reasoning

4. Text Data Mining Knowledge Area
   Data science text mining-related scientific subjects from CCS2012 are as follows:

   - Computing methodologies
     - Artificial intelligence

       Natural language processing
       Knowledge representation and reasoning
       Search methodologies

5. Predictive Analytics Knowledge Area analytics knowledge area. Data science predictive analytics-related scientific subjects from CCS2012 are as follows:

   - Computing methodologies
     - Artificial intelligence

       Natural language processing
       Knowledge representation and reasoning
       Search methodologies

6. Computational Modelling, Simulation, and Optimization Knowledge Area. Data science computational modelling, simulation, and optimization-related scientific subjects from CCS2012 are as follows:

   - Computing methodologies
     - Modelling and simulation
     - Model development and analysis
     - Simulation theory
     - Simulation types and techniques
     - Simulation support systems

On the other hand, the ACM CCS2012 Extension Points from the DS-BoK Data Analytics are:

1. Theory of computation. The ACM CCs 2012 Theory of computation extension point from DS-Bok is:

   - Algorithms for big data computation

2. Mathematics of Computing. ACM CCsCC 2012 Mathematics of computing extension point from DS-Bok is:
   - Mathematical software for big data computation
3. Computing methodologies. The ACM CCs 2012 Computing methodologies extension point from DS-Bok is:
   - New DSA computing
4. Information Systems. The ACM CCs 2012 Information systems extension point from DS-Bok is:
   - Big data systems (e.g., cloud based)

     The ACM CCs 2012 Information systems applications extension points from DS-Bok are as follows:
   - Big data applications
   - Domain-specific data applications

Data Analytics Model Curriculum Approach

This subsection presents the definition of the EDISON Data Science Model Curriculum that is primarily based on mapping between DS-BoK knowledge areas and Data Science Model Curriculum, MC-DS, learning units, which may represent academic courses and training modules, for required competence groups using a competence-based learning model.

The proposed MC-DS can be used for defining individual curricula for specific data science professional profiles or customized individual curricula for practitioners who want to obtain a data science qualification or certification. The proposed methods can be used for developing tools for customizing or profiling training and/or education programmes for students or individual trainees.

The model curriculum is organized as core and elective topics, following the ACM definition. Core topics are required for every data science programme, whereas elective topics aim to cover in depth the knowledge on a specific area of data science. The last step identifies the learning outcomes associated with each core or elective topic. The approach to defining the Data Science Model Curriculum in the EDISON Project has followed a competence-based education model and can be summarized in the following five steps:

1. For each enumerated competence from the CF-DS, define learning outcome according to knowledge or mastery level (defined as familiarity, usage, assessment for current MC-DS version).
2. Each knowledge area group of DS-BoK (which includes both KAGs from existing BoKs and those defined based on the ACM Classification Computer Science CCS2012) is mapped to existing academic subject classification groups

that are primarily based on ACM CS2012 and complemented with domain- or technology-specific classifications such as BABOK, ACM-BOK, DAMABOK, PM-BOK, and others to be defined by subject matter experts.
3. For each KAG or knowledge unit, specify related learning units defined according to academic subject classification or following current practices by universities.
4. For each learning unit, assign/suggest its category as core/mandatory (Tier 1 or Tier 2), elective, or prerequisite.
5. For both core and elective, define a list of learning outcomes.

The MC-DS learning units, LUs, or courses of step 3 can be defined based on the knowledge area groups and knowledge units defined in the DS-BoK. The individual learning units or courses are defined in accordance with the existing classification of academic disciplines, in particular the ACM CCS2012, and are verified with the existing offered courses at universities. In addition, the proposed LUs are grouped according to ACM CCS2012 classification or DS-BoK knowledge groups/units that can be used as context information for future data science curricula development, modification, or enhancement with the linked courses and disciplines.

In the following, we present the six learning outcomes related to enumerated CF-DS competences for Data Analytics and the different knowledge/proficiency levels defined based on Bloom's taxonomy, with the general learning outcomes defined after CF-DS competences that are in most cases split into 3 knowledge levels and use specific verbs that reflect necessary comprehension or mastery level. The data analytics learning outcomes are as follows:

1. Data Analytics Learning Outcome (LO1-DA). Learning outcome 1, Data analytics, DSDA. Its acronym is LO1-DA. The global learning outcome (LO) of data analytics (DSDA-DA) is the use of appropriate data analytics and statistical techniques on available data to discover new relations, deliver insights into research problems or organizational processes, and support decision-making. The learning outcomes (LOs) for the whole DSDA are denoted as LO1-DA and specified at three levels:

    - Familiarity: Choose an appropriate existing analytical method and operate existing tools to perform specified data analysis. Present data in the required form.
    - Usage: Develop data analysis applications for specific datasets and tasks or processes. Identify necessary methods and use them in combination if necessary. Identify relations and provide consistent reports and visualizations.
    - Assessment: Create a formal model for the specific organizational tasks and processes and use it to discover hidden relations, propose optimization and improvements. Develop new models and methods if necessary. Recommend and influence organizational improvement based on continuous data analysis.

Once we have specified the general learning outcome, the learning outcomes for specific DSDA competences are:

1.1. LO1.01 based on DSDA01. Effectively use a variety of data analytics techniques, such as machine learning (including supervised, unsupervised, semisupervised learning), data mining, prescriptive and predictive analytics, for complex data analysis through the whole data life cycle.

   (a) Familiarity: Choose and execute existing data analytics and predictive analytics tools.
   (b) Usage: Identify existing requirements and develop predictive analysis tools.
   (c) Assessment: Design and evaluate predictive analysis tools to discover new relations.

1.2. LO1.02 based on DSDA02. Apply designated quantitative techniques, including statistics, time series analysis, optimization and simulation, to deploy appropriate models for analysis and prediction.

   (a) Familiarity: Choose and execute standard methods from existing statistical libraries to provide an overview.
   (b) Usage: Select the most appropriate statistical techniques and model available data to deliver insights.
   (c) Assessment: Assess and optimize organization processes using statistical techniques.

1.3. LO1.03 based on DSDA03. Identify, extract, and pull together available and pertinent heterogeneous data, including modern data sources such as social media data, open data, and governmental data.

   (a) Familiarity: Operate tools for complex data handling.
   (b) Usage: Analyse available data sources and develop tools that work with complex datasets.
   (c) Assessment: Assess, adapt and combine data sources to improve analytics.

1.4. LO1.04 based on DSDA04. Understand and use different performance and accuracy metrics for model validation in analytics projects, hypothesis testing, and information retrieval.

   (a) Familiarity: Name and use basic performance assessment metrics and tools
   (b) Usage: Use multiple performance and accuracy metrics and select and use the most appropriate metric for a specific type of data analytics application.
   (c) Assessment: Evaluate and recommend the most appropriate metrics and propose new methods for new applications.

1.5. LO1.05 based on DSDA05. Develop required data analytics for organizational tasks, integrate data analytics and processing applications into organization workflow and business processes to enable agile decision-making.

   (a) Familiarity: Define data elements necessary to develop specified data analytics.
   (b) Usage: Develop specialized analytics to enable decision-making.
   (c) Assessment: Design specialized analytics to improve decision-making.

1.6. LO1.06 based on DSDA06. Visualize the results of data analysis, design dashboards, and use storytelling methods.

   (a) Familiarity: Choose and execute standard visualization.
   (b) Usage: Build visualizations for complex and variable data.
   (c) Assessment: Create and optimize visualizations to influence executive decisions.

Now, after defining the Data Analytics Learning Outcomes, we will present the Organization and Application of the Data Science Model Curriculum before presenting the Data Analytics-related courses. As has been explained, each knowledge area group of DS-BoK is mapped to existing academic subject classification groups that are primarily based on ACM Classification Computer Science CCS2012 complemented with domain- or technology-specific classifications such as those defined in the existing BoK's ACM CS-BOK, BABOK, SWEBOK, DM-BoK, PM-BOK, and others that should be defined by subject matter experts. For each KAG, the MC-DS specifies learning outcomes and mastery levels following Bloom's taxonomy verb usage. Learning outcomes are also linked to a set of learning units, which are examples of the practical application of knowledge units. ECTS points are provided for professional profile groups and divided into Tier 1, Tier 2, elective and prerequisite categories to help create detailed tracks and specializations for academic programmes and professional training.

Figure 3 illustrates the relation between different EDSF components when defining specific academic or professional training programmes that can be tailored for specific target data science professional groups or target competences.

Figure 4 illustrates a general approach to the application of the model curriculum to create an educational programme. The work starts by deciding on a target Data Science Professional Profiles group (that is defined by the required competences, skills, and knowledge) the programme that should cover and the level of the programme, usually bachelor's or master's. These elements allow us to identify a set of competencies to be addressed in the programme. To identify relevant knowledge units and to what extent they should be covered in the new programme, the programme designer can consult tables with ECTS points, which are defined for each professional profile. ECTS point specifications include a degree of flexibility to adjust to particular needs. For each knowledge area, MC-DS defines a set of knowledge units based on BoK and a set of learning outcomes based on the competence framework. Topics and learning outcomes become a basis for the definition of new courses or the use of existing courses. It is important to note that

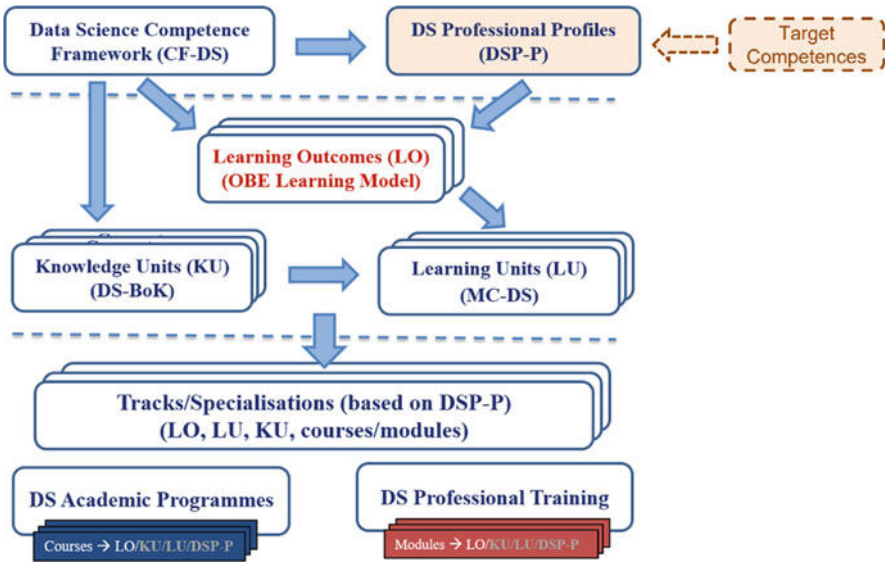


Fig. 3 Interaction between different components of EDSF when using model curriculum for defining academic of professional training programme for target professional group (or target competences)

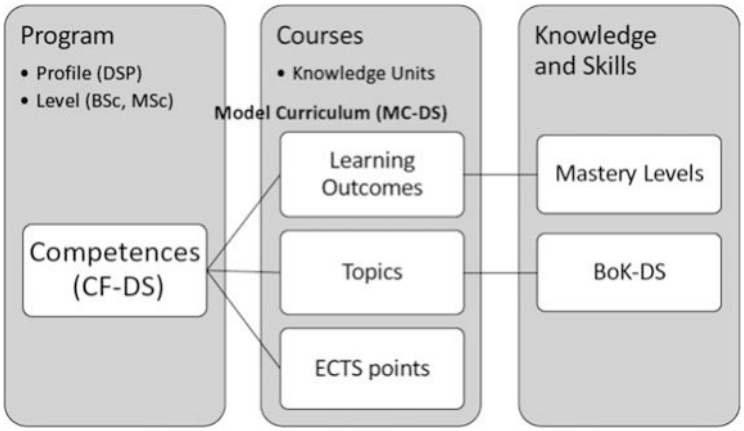


Fig. 4 Visualization of the model curriculum application for programmes and courses

when designing a specific course, it may include elements from several knowledge areas to ensure consistency of the whole data science programme.

Adjustment of learning outcome levels for different proficiency levels can be done based on the full list of learning outcomes for all CF-DS competences and for all mastery/proficiency levels. Learning outcomes can repeat between subgroups within the same KAG but can be adjusted to a specific course and topic context.

Now, we are ready to see a proposal of Data Analytics-Related Courses based on the previous definitions. For them, we know that the data analytics knowledge group builds the ability to use appropriate statistical and data analytics techniques on available data to deliver insights and discover information, provide recommendations, and support decision-making. It includes knowledge areas that cover data mining, supervised and unsupervised machine learning, statistical modelling, and predictive analytics. In addition, we are going to remember here the six Knowledge areas of the Data Analytics Area Group.

1. KA01.01 (DSDA.01/SMA) Statistical methods, including descriptive statistics, exploratory data analysis (EDA) focused on discovering new features in the data and confirmatory data analysis (CDA) dealing with validating formulated hypotheses.
2. KA01.02 (DSDA.02/ML) Machine learning and related methods for information search, image recognition, decision support, classification.
3. KA01.03 (DSDA.03/DM) Data mining as a particular data analysis technique that focuses on modelling and knowledge discovery for predictive rather than purely descriptive purposes.
4. KA01.04 (DSDA.04/TDM) Text analytics applies statistical, linguistic, and structural techniques to extract and classify information from textual sources, a species of unstructured data.
5. KA01.05 (DSDA.05/PA) Predictive analytics focuses on application of statistical models for predictive forecasting or classification.
6. KA01.06 (DSDA.06/MSO) Computational modelling, simulation, and optimization

Once we have remembered the Data Analytics Knowledge Areas, the proposed topics for the courses and learning outcomes for each are specified in the following points:

1. DSDA.01/SMA. Statistical Methods and Data Analysis Statistics and probability theory are foundational components of data analytics and constitute a significant part of data science competences and knowledge. This module provides insights into major statistical and data analytics paradigms and schools of thought. They can be taught separately or as a part of other data analytics-related modules or courses.

   - Topics:

     – Statistical paradigms (regression, time series, dimensionality, clusters).
     – Probabilistic representations (causal networks, Bayesian analysis, Markov nets).
     – Frequentist and Bayesian statistics.
     – Exploratory and confirmatory data analysis.
     – Information theory.
     – Graph theory.

- Learning outcomes
  - Choose and execute standard methods from existing statistical libraries to provide an overview (LODA.02 L1)
  - Select the most appropriate statistical techniques and model available data to deliver insights (LODA.02 L2)
  - Identify requirements and develop analysis approaches (LODA.01 L2)
  - Assess and optimize organization processes using statistical techniques and simulation (LODA.02 L3)

2. DSDA.02/ML. Machine Learning. Data scientists have a wide range of ready machine learning libraries available. Nevertheless, they also need to go beyond the simple application of algorithms to achieve the expected results. New problems they face might require an in-depth understanding of the theoretical underpinning of both simple and advanced algorithms. This module covers the use, analysis and design of machine learning algorithms.
   - Topics:
     Machine learning theory (supervised, unsupervised, reinforced learning, deep learning, kernel methods, Markov decision processes)
     - Design and analysis of algorithms (graph algorithms, data structure design and analysis, online algorithms, Bloom filters and hashing, MapReduce algorithms)
     - Game theory and mechanism design
     - Classification methods
     - Ensemble methods
     - Cross-validation
   - Learning outcomes
     - Choose and execute existing analytic techniques and tools (LODA.01 L1)
     - Identify requirements and develop analysis approaches (LODA.01 L2)
     - Develop specialized analytics to enable agile decision-making and integrate them into organizational workflows (L0DA.05 L2)
     - Design and evaluate analysis techniques and tools to discover new relations (LODA.01 L3)

3. DSDA.03/DM. Data Mining. Mathematical and theoretical aspects of data analytics must be implemented in a computational form appropriate for both the problem at hand and the data size. This module builds familiarity with the most relevant data mining algorithms and related methods for knowledge representation and reasoning.
   - Topics:
     - Data mining and knowledge discovery
     - Knowledge representation and reasoning

  - CRISP-DM and data mining stages
  - Anomaly detection
  - Time series analysis
  - Feature selection, a priori algorithm
  - Graph data analytics
- Learning outcomes
  - Choose and execute standard methods from statistical libraries to provide an overview (LODA.02 L1)
  - Select the most appropriate statistical techniques and model available data to deliver insights (LODA.02 L2)
  - Analyse available data sources and develop a tool that works with complex datasets (LODA.03 L2)
  - Develop specialized analytics to enable agile decision-making and integrate them into organizational workflows (LODA.05 L2)
  - Evaluate and recommend data analytics organizational strategy (LODA.05 L3)

4. DSDA.04/TDM. Text Data Mining. Text data mining can be considered a subset of data mining, but it is worth a separate consideration due to the amount of text data available and particular methods developed over the years to analyse it.
   - Topics
     - Text analytics including statistical, linguistic and structural techniques to analyse structured and unstructured data
     - Data mining and text analytics
     - Natural language processing
     - Predictive models for text
     - Retrieval and clustering of documents
     - Information extraction
     - Sentiment analysis
   - Learning outcomes
     - Choose and execute standard methods from statistical libraries to provide an overview (LODA.02 L1)
     - Analyse available data sources and develop a tool that works with complex datasets (LODA.03 L2)
     - Evaluate and recommend data analytics organizational strategy (LODA.05 L3)

5. DSDA.05/PA. Predictive Analytics Predictive analytics are commonly used to foresee future events to avoid them or act ahead. This module covers both traditional approaches based on time series and newer approaches based on deep learning. Anomaly detection is a particular focus since it is one of most common application areas.

- Topics
  - Predictive modelling and analytics
  - Inferential and predictive statistics
  - Machine learning for predictive analytics
  - Regression and multianalysis
  - Generalized linear models
  - Time series analysis and forecasting
  - Deploying and refining predictive models
- Learning outcomes
  - Choose and execute existing analytic techniques and tools (LODA.01 L1)
  - Identify requirements and develop analysis approaches (LODA.01 L2)
  - Create stories and optimize visualizations to influence executive decisions

6. DSDA.06/MSO. Computational Modelling, Simulation, and Optimization. Modelling and simulation are essential approaches to handle the complexity of some systems and event chains. This module provides an introduction in both theoretical and practical aspects of model development and simulation techniques.
   - Topics:
     - Modelling and simulation theory and techniques (general and domain-oriented)
     - Operations research and optimization
     - Large-scale modelling and simulation systems
     - Network optimization
     - Risk simulation and queuing
   - Learning outcomes
     - Describe and execute different performance and accuracy metrics (LODA.04 L1)
     - Compare and choose performance and accuracy metrics (LODA.04 L2)
     - Assess and optimize organization processes using statistical techniques and simulation (LODA.02 L3)

Data Analytics Professional Profiles

This section presents a description of the professional profiles, defined by the EDISON project, that can be performed when a Data Scientist is working on the Data Analytics Knowledge Area Group.

The EDISON project presents the following Data Science Professional Profiles, which are also called the data-related occupations family. They are defined as an

extension to the ESCO occupations taxonomy. The proposed occupations for data science are placed in four top classification groups:

- Managers, for managerial roles
- Professionals, applications developers, and infrastructure engineers
- Technicians and associate professionals, for operators and technicians
- Clerical support workers, for data curators and stewards

In the following, the data science-related occupation extension to ESCO classification is presented for the four top classifications of occupations: managers, professionals, technicians and associate professionals, and clerical support workers. The occupations are presented from the top level (TL) to the occupations (O) with the existing (EH) and new (NH) hierarchies for each top level, and the occupations group (OG):

- Managers (TL)
  - Production and specialized services managers (EH)
  - Data science/big data infrastructure managers (NH)

    Research infrastructure managers (OG)
    DSP01. Data science (group) manager (O)
    DSP02. Data science infrastructure manager (O)
    DSP03. Research infrastructure manager (O)

- Professionals (TL)
  - Science and engineering professionals (EH)
  - Data science professionals (NH)

    Data science professionals not elsewhere classified (OG)
    DSP04. Data scientist (O)
    DSP05. Data science researcher (O)
    DSP06. Data science architect (O)
    DSP07. Data science (application) programmer/engineer (O)
    DSP08. (Big) data analyst (O)
    DSP09. Business analyst (O)

  - Information and communications technology professionals (EH)
  - Data science technology professionals (NH)

    Data handling professionals not elsewhere classified (OG)
    DSP10. Data steward (O)
    DSP11. Digital data curator (O)
    DSP12. Data librarian (O)
    DSP13. Data archivist (O)

  - Science and engineering professionals (EH)
  - Database and network professionals (NH)

Large-scale (cloud) data storage designers and administrators (OG)
DSP14. Large-scale (cloud) database designer1 (O)
Large-scale (cloud) data storage designers and administrators (OG)
DSP15. Large-scale (cloud) database administrator (O)
Database and network professionals not elsewhere classified (OG)
DSP16. Scientific database administrator (O)

- Technicians and associate professionals (TL)
  - Science and engineering associate professionals (EH)
  - Data science technology professionals (NH)

    Data infrastructure engineers and technicians (OG)
    DSP17. Big data facilities operators (O)
    DSP18. Large-scale (cloud) data storage operators (O)
    Database and network professionals not elsewhere classified (OG)
    DSP19. Scientific database operator (O)
    Clerical support workers (TL)

  - General and keyboard clerks (EH)
  - Data handling and support workers (NH)

    Data and information entry and access (OG)
    DSP20. Data entry/access desk/terminal workers (O)
    DSP21. Data entry field workers (O)
    DSP22. User support data services (O)

The following are the commonly used definition of digital librarian responsibilities and functions: selection, acquisition, organization, accessibility, and preservation of digital information/libraries. Manages digital materials; takes a lead role in the creation, maintenance and stewardship of digital collections, including the digitization of special collections; and develops strategies for the effective management and preservation of library digital assets.

From the classification presented above, we now present the definition of the Data Science Professional Profiles by defining their competences and organizational roles. The proposed definition can be instrumental in defining education and training profiles for students and for practitioners to acquire necessary competences and knowledge for specific professional profiles or occupations. It can also be used for defining certification profiles or career path building. The presented DSPP together with CF-DS and other EDSF documents that provide the basis for multiple practical uses include but are not limited to the following:

- Assessment of individual and team competences, as well as balanced data science team composition comprising the data science-related roles that together provide the necessary set of skills.

- Developing tailored curriculum for academic education or professional training, in particular, to bridge skills gap and staff up/reskilling
- Professional certification and self-training.

As has been said, the data science occupation groups are placed in the following top-level ESCO hierarchies: managers; professionals; technicians and associate professionals; and optionally, some data management occupations can also be placed into the clerical support workers group such as digital data archivist and digital librarians.

Correspondingly, the following new third-level occupation groups are proposed:

- Data science/big data infrastructure managers
- Data science professionals
- Data science technology professionals
- Data and information entry and access (this is a candidate group under clerical support workers' top-level hierarchy)

It is proposed that the existing ESCO group "database and network professionals" should be extended with new occupations (or professions) related to big data or cloud-based databases: large-scale (cloud) database administrator/operator and scientific database administrator/operator; however, further identification of such occupations needs to be done.

To ensure smooth data science professional acceptance by industry and employment bodies, the proposed profiles should be compatible with the relevant standards ESCO, CWA 16458 2012 ICT Profiles, eCFv3.0 (future CEN standard EN 16324). Figure 5 graphically illustrates the existing ESCO hierarchy and the proposed new data science classification groups and corresponding new data science-related

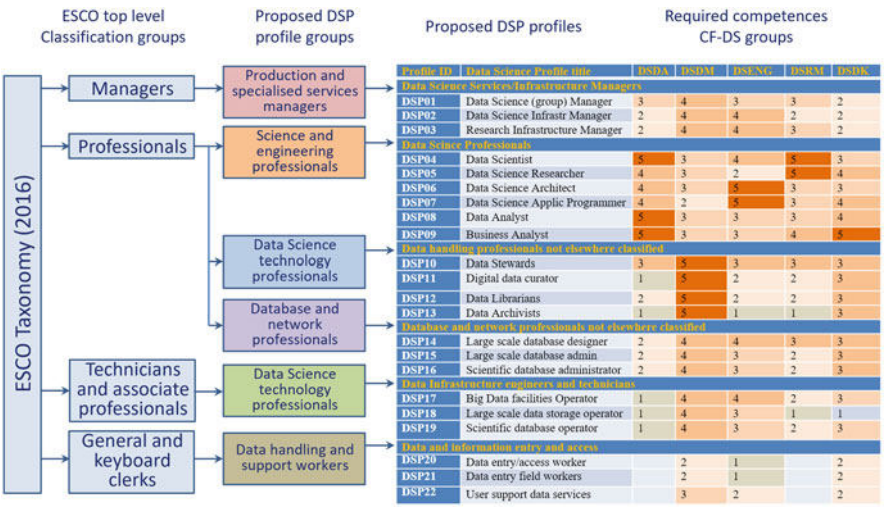


Fig. 5 Proposed data science-related extensions to the ESCO classification hierarchy and corresponding DSPP by classification groups

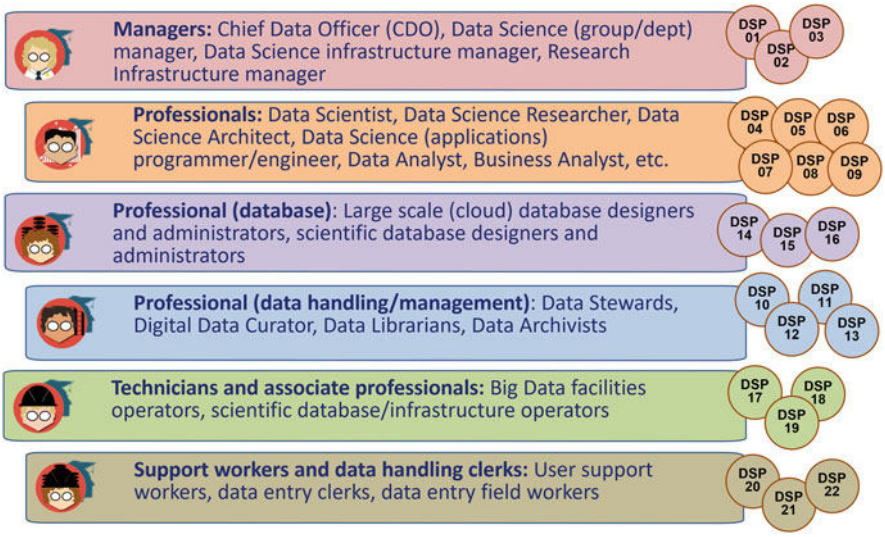


Fig. 6 Data Science Professional Profiles and their grouping by the proposed new professional groups compliant with the ESCO taxonomy

profiles. The table in the figure illustrates what competence groups are relevant to each profile by indicating competence relevance from 0 to 5 (0—not relevant, 5—very important), where information is taken from Table 4.3 that will be presented later. Figure 6 provides a visual presentation of the identified DSPP and their grouping by the proposed high-level classification groups.

In the following, a definition of the profile summary statement of the Data Science Professional Profiles defined in the taxonomy of 5.2.2. for Data Analytics is presented (between brackets are the alternative titles and legacy titles):

- Managers
  - Data science/big data infrastructure managers

    Research infrastructure managers

    DSP01. Data science (group) manager (data analytics department manager) Proposes, plans, and manages functional and technical evolutions of the data science operations within the relevant domain (technical, research, business)

- Professionals
  - Data science professionals

    Data science professionals not elsewhere classified

    DSP04. Data scientist (data analyst). Data scientists find and interpret rich data sources, manage large amounts of data, merge data sources, ensure

consistency of datasets and create visualizations to aid in understanding data. Build mathematical models, present and communicate data insights and findings to specialists and scientists, and recommend ways to apply the data.

DSP05. Data science researcher (data analyst). Data science researcher applies scientific discovery research/process, including hypothesis and hypothesis testing, to obtain actionable knowledge related to scientific problem, business

DSP07. Data science (application) programmer/engineer (scientific programmer, data engineer). Designs/develops/codes large data analytics applications to support scientific or enterprise/business processes.

DSP08. (Big) Data analyst. Analyses a large variety of data to extract information about system, service or organization performance and presents them in usable/actionable form.

DSP09. Business analysts (business development managers (data science roles)) analyse a large variety of data information systems for improving business performance.

Given the different professional profiles in which data analytics are the core knowledge, we will present the role of the experts in data analytics in a data science team. Data science team composition and competence matching is one of the intended uses of the EDSF and DSPP in particular. Figure 7 illustrates a case of creating a data science team or group for an average size of the research organization with an affiliated number of researchers of 200–300, which would require a data science team of 10–15 members whose responsibility would include supporting all main stages of the data life cycle: data collection, data input/ingest, data analysis,

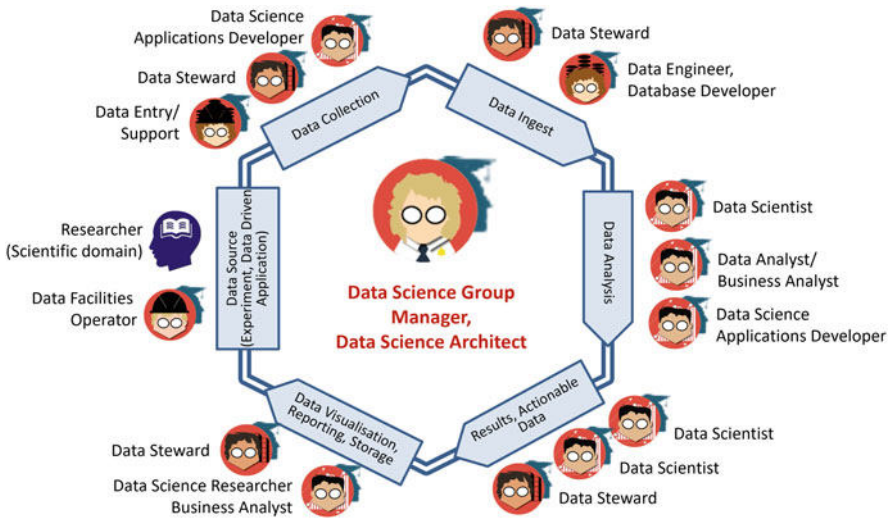


Fig. 7 Matching the candidate's competences for the data scientist competence profile

reporting, visualization, and storage. The figure also illustrates possible roles that may be assigned to perform different functions at different data workflow stages.

To support all data-related research or production stages, the following roles may be required (including suggested staffing for the team of 10–12 members):

- (Managing) Data science architect (1)
- Data scientist (1), data analyst (1)
- Data science application architect/developer/programmer (2)
- Data infrastructure/facilities administrator/operator: storage, cloud, computing (1)
- Data stewards, curators, archivists (3–5)

It is possible that some of the above roles can be redefined and reallocated to the data science team from the previous ICT and IT infrastructure groups or departments. In this case, some basic data science training will be required for not initially data-related professions.

It also suggested a distinct role of the data steward, a new emerging role for data-driven research organizations and projects. Data stewards should play a bridging role between the subject domain researcher and the data science team or data scientist in particular cases to help translate between the subject domain and data science or data analytics domain. Data stewards can have both backgrounds, either ICT and computer or digital curation/librarian.

A similar approach to data science and data governance role definition and team building was used in IBM enterprise consulting practice.

Finally, we present the data science-enabled professions. Recent studies by BHEF, PwC and IBM, BGT and BHEF identified strong growth of data science and analytics (DSA)-enabled jobs that are not pure data scientists but require extensive DSA knowledge to work in specific industry sectors. Figure 8 from the PwC and BHEF study provides an illustration of currently highly demanded DSA-enabled jobs in multiple industry and business sectors: finance and insurance; healthcare and social assistance; information; manufacturing; professional, scientific, and technical services; and retail trade.

The study provides data on 2.35 million job postings in the United States in 2017: 23% data scientist and 67% DSA-enabled jobs. There is also a strong demand for managers and decision-makers with data science (data analytics) skills/understanding. This creates a new challenge to deliver actionable knowledge and competences to CEO-level managers.

About This Book

This book is the second of a series of books written as a result of the EDISON Project, which has been described in the second section of this chapter. The first of the series was the book entitled *The Data Science Framework: A View from the EDISON Project*, published by Springer in 2020. That first book was written with a double purpose: the first one was to gather all the information and knowledge

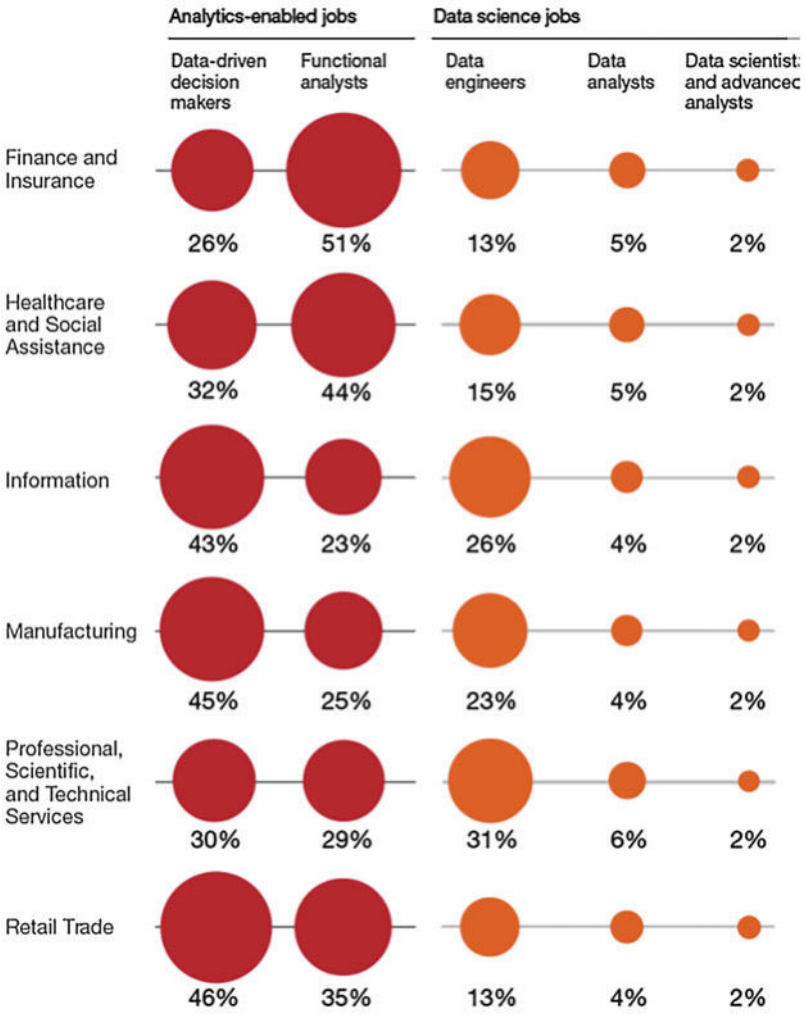


Fig. 8 There is a strong demand for business people with analytics skills, not just data scientists in multiple industry sectors

obtained during the development of the EDISON project in a single document that allows a much easier handling by researchers, practitioners, teachers, and all those interested in the data science; and the second was to go a step beyond what was obtained in the project and present the knowledge in a more elaborate and expanded way, which would allow an easier and deeper assimilation of them. Consequently, the topics were presented in more depth than the reader can find in the documentation resulting from the project, with a substantially revised structure and with a large amount of additional information. Most of the contents that have been introduced in

this introductory lesson, or chapter, are presented in depth in the first book; for that reason, we want to encourage the reader of this second book to read and consult the first to advance the concepts related to the Data Science Framework.

The content of the first book was structured as follows. After an initial chapter of introduction, Chapter "Data Science Competences" presents the set of competences that a data scientist must have. Starting from these lists of competences, the body of knowledge that the discipline must have to allow us to obtain them was presented in Chapter "Data Science Body of Knowledge". This body of knowledge was used to define, in Chapter "Data Science Curriculum", an approach to the development of data science curricula; after the treatment of the knowledge and education of data science seen in the previous chapters, the development of the profession of data science was exposed in Chapter "Data Science Profesional Profiles", in which the Data Science Professional Profiles are presented. Chapter "Use Cases and Applications" presents a set of four real successful use cases and applications of the EDISON Data Science Framework that can be very useful for the reader in his or her application of the knowledge acquired in the book. The book ends with an Annex in which some models of processes related to data science are presented.

This second book follows the series started in the first book, but its conception and development are totally different from the first one: if the first book is a theoretical book that was thought to present the framework of the whole data science discipline from an absolutely theoretical point of view, this second book does not present the whole data science discipline but only one of its six knowledge area groups, the data analytics knowledge area group, and presents it from an absolutely practical point of view. if in the first book there is neither solved nor even proposed any practical exercise, in this second book have been conceived and written with the practical exercise's resolution as the main structural element of the book.

Following the previous, the conception and construction of this book is based on the maxim that the authors read, many years ago, in an old book on electromagnetism problems, and whose veracity we have been able to verify throughout our academic lives, first as students and later as teachers. The maxim is as follows: "*I forget what I hear, I remember what I see, and I learn what I do.*" Consequently, the book will demand that the reader *to do* in order to *learn*. And,

what is the reader to do?

The answer is Exercises. Exercises, for the practical application of each of the theoretical concepts taught. The book will provide the reader with its complete detailed solution of all the exercises stated. However, the authors strongly ask the readers not to look at these solutions until they have solved the exercises themselves. If you look at the solutions before solving the exercise, it loses all its effectiveness as a learning element, and if this is done repeatedly, the book loses an important part of its value. However, it is important to make it very clear that this is not a "problems book", since the theoretical concepts are exposed with length and depth.

This book is about *Data Analytics*, and, how the reader knows now, after reading this introduction, it is a very extensive subject, for that reason, it is impossible to include all of its body of knowledge in only one book, and less if is intended to

describe and explain them with the high level of detail that we want to apply in the book. For this reason, only some knowledge units of some knowledge areas of the Data Science Knowledge Area have been chosen to be included. The selection criteria applied to choose those contents have been double: first, they have been selected because of their direct relevance to the Data Analytics domain (knowledge area) and presentation of the main paradigm in data analysis; the second criteria have been their instructional value in building consistent foundational knowledge by learners. This is also a goal that the reader can use the example presented in this book as a guide to follow the same procedure when they need to study and learn other Data Analytics knowledge units not included in the book.

Once this has been clarified, the book is structured in seven chapters or lessons because each presents a different lesson. The contents of the book are as follows: After an initial chapter of introduction, Chapter "Data" presents the fundamentals of the knowledge related to the concept of *Data*. Starting with the definition of the concepts Characteristic and Data, the chapter introduces to the reader all the knowledge, fundamentally of statistical techniques, related to the initial analysis that must be done over the set of data that are the object of the intended study with the objective of obtaining a deep knowledge of them and their main features. To facilitate the reader to pass to do only description of the data to make inferences over them, the book presents a novel approach over their observation and analysis and uses the concept of *event* for each observation, with the objective of facilitating a probabilistic approach over their treatment. This is quite different from the way that the books published about the subject describe and teach it, but we think that it will be very useful for the reader; for that reason, we have dedicated the whole Chapter "Probability" to introduce the fundamentals of *Probability*. Once we know how to do, from the contents of Chapter "Data", an initial analysis and description of the data, and we are able to see each observation as an event, from the contents of Chapter "Probability", and from this, their probabilistic character, in Chapter "Anomaly Detection" the book starts to apply Data Science Analytics to obtain "*Knowledge from the Data*", and the first one, introduced in this chapter is the *Anomaly Detection*, an analysis that tries to identify those data in the studied set, that are enough different from the others to be considered anomalies, and its identification can be very interesting and important, because those anomalies can be only errors in the data, that must be removed but can also be very important data that can give us a lot of information. After determining how individual data that are different from the others can be identified, in Chapter "Unsupervised Classification", we introduce how to identify groups of data that are sufficiently different from the others in the set and sufficiently similar between them to be considered as belonging to the group. This analysis is called *Unsupervised Classification* or Outliers Detection. In Chapter 6 "*Supervised Classification*" is introduced to teach to the reader how to use the studied dataset to try to obtain a classifier that allows, for future observed sets of data with the same characteristics, knowing the value of a specific characteristic whose value is unknown from the values of the characteristics that compose the classifier. Finally, the last chapter, Chapter "Association", introduces to the reader the foundations of the *Association Analysis*, which tries to identify which

of the possible sets of characteristics that can be observed for an object in different observations, or events, appears more times together to try to establish that those characteristics are associated.

All the chapters, or lessons, with the exception of the first one, are constructed and structured in the same manner, and all of them will have three sections:

- Section A introduces, in a theoretical and, at the same time, practical way, all the basic theoretical knowledge that a data analyst must know in depth.
- Section B presents computer-based cases solving of the same contents that have been presented in the previous part.
- Finally, Section C will consist of a set of statements of exercises about the contents presented in part A, in which detailed solutions can also be found in this part of the lesson. This will be remembered several times in other parts of the book, but it is very important to obtain the best results for the learning process throughout the use of the book, that the reader tries to solve the exercises by himself before seeing their solutions and that only once solved check if the obtained solutions are correct.

With this structure, after starting with an introduction, the contents of the lesson are presented in a theoretical-practical manner; that is, after each theoretical concept is introduced, an exercise to apply that concept is presented and solved in detail without the help of any software tool. Then, the examples that have been previously solved with the R software tool. Finally, all the learning of the contents of all the lessons is reinforced with the resolution of a set of proposed exercises, in which solutions, with and without the use of software tools, are explained in depth.[3]

In addition to introducing the Data Analytics Knowledge Area Group of the EDISON Data Science Framework, this book has also been conceived as a textbook for the Data Analytics Fundamentals course of 64 face-to-face, or contact, hours in the classroom, which could be taught during a term, that is, 16 weeks, with a theoretical class and another practical per week, both of 2-h duration.[4] Each chapter corresponds to a lesson and is taught in 2 weeks with a delay between the theoretical and the practical contents of the same lesson, which means that the first 3 weeks of the practical lesson must be dedicated to introduce the R environment because the first 2 weeks are to teach lesson 1, Introduction, which is theoretical, and the third week the lesson Data are started and there are no content for practices; in the fourth week, the first practical lesson about Data with R is held. As there are 7 lessons with 2 weeks for each lesson, there are 14 weeks of lessons, and the 2 weeks remaining can be dedicated to examinations or/and the resolution of a data analysis of a complete practical case. In the theory class, the theoretical concepts of each topic

[3] In this point, we recommend again to the reader to try solve alone all the exercises previously to see their solutions in the book.

[4] In the case of having fewer weeks, you could either increase the number of hours per week to 5, for 12 weeks, or not teach any lessons. The lessons removed must be any of the 4, 5, 6, or 7, because all of them are not the basis of Data Analytics and are self-contained. In the event that the time constraints are severe, both solutions could be combined.

will be exposed, and all of them will be seen through a methodology consisting of two steps, the first of which will be to state and explain the theoretical concept; then, immediately after following the construction principle that governs the entire book and that has been exposed at the beginning of it, a practical exercise will be carried out, which allows the student to consolidate the knowledge acquired. This practical exercise must be solved on paper with the help of a hand calculator. During the laboratory session of the same week, you will learn how to solve with the use of the environment and the R language the same exercises that were solved in class in the theory session. Learning and deepening the knowledge of R will occur in parallel with that of Data Science.

Data

In this second chapter, we will see the essential aspects related to the concept of *Data* or *Datum.*[1] As explained in the Introduction, this chapter is structured in three sections. This structure of three sections will also be presented for all the following chapters.

Section A introduces, in a theoretical and, at the same time, practical way, all the basic theoretical knowledge related to the concept of Data that a Data Analyst should know, from the definition of the concept of Data and the related ones to the initial description process of the data set under study. Data Science is generally linked to Statistics, which is the reason why there are many concepts and definitions that need to be clarified and matched/harmonized between both domains to bring synergy and allow specialists from both to understand, speak, communicate, and cooperate better.

Section B presents the computer-based solving of the same examples used in section A to introduce the theoretical knowledge. As it is the first time that this part appears in a lesson, the basics of the programming language and environment that, from here, will be used in the whole book, R, are introduced in detail. After that, the first problem solved with R will be solved in detail.

Section C consists of a set of statements of exercises about Data for which detailed solutions can also be found in this section of the chapter.[2]

[1] In English Language, a single datum is called *Datum,* whereas a set of datum is called *Data*, but in day-to-day work in Data Science, it is usually used the term *Data* for referring to both a single datum and a set of data. There is not a common recipe to know when the term *Data* is referring to a single datum of a set of data and it is the context that establishes whether the meaning is one or the other. In the book, we are going to use only the term *Data* for refereeing both single and plural and the same criterium will be followed, it will be the context that establishes whether it is single or plural.

[2] As has been said in other parts of the book, it is very important to obtain the best results for the learning process throughout the use of the book, that the reader tries to solve the exercises by himself before seeing their solutions, and that only once solved, check if the obtained solutions are correct.

J. J. Cuadrado-Gallego, Y. Demchenko, *Data Analytics*,
https://doi.org/10.1007/978-3-031-39129-3_2

In this lesson, the reader can also find an Annex[3] with extended concepts for contents treated in the lesson.

A. Theory

This first section of the chapter is structured in 7 Subsections: (1) Introduction, (2) Characteristic, (3) Data, (4) Available Data, (5) Frequency, (6) Mean, and (7) Median. In which the basic knowledge related to the concept of data and with the description process of the available data are presented in detail.

Introduction

As stated in the first introductory chapter, the raw material of Data Science and, consequently, of Data Science Analytics is data, so we begin our study of the subject by studying in depth the concept of *data* and all the related concepts. To do that, the lesson can be divided into two blocks:

In the first block, the most important concepts, which must be known in depth, related to the concept of *Data* are introduced. It starts with the concept of *Characteristic,* its definition, its types, and its difference and relationship with the *Data* concept will be explained. Next, the concept of *data*, its definition, and its types, from the point of view of their nature, and from their storage, that is, the type of data that are usually handled when the process of acquisition and preprocessing is being performed. The last contents introduced in this block are related to different concepts that have been grouped under the name of *Available Data* and they are focused on features of specific sets of data to be analysed in each study, which are the concepts of *Experiment*, *Population*, *Sample,* and data *Quality*.

The second block introduces the main parameters[4] used to perform the first analysis and description of the data that will be carried out, which allow us to better understand the available data set object of study. The first parameter that is usually calculated is the *Frequency*, which is fundamental in data observation. Its definition and types, its application to grouped[5] data and a related measure called *Mode* are introduced. Next, a second parameter, the *Mean*, is introduced, and its definition, the definition of the *Arithmetic Mean*, and the related parameters of *Variance* and

[3] No all the lessons are going to have an annex, only in those ones for which the contents treated in the annex have been considered to include there to increase the readability of the book.

[4] Parameter can be defined as: "a quantity (such as a mean or variance) that describes a statistical population".

[5] The grouping of data will be, in certain cases, a very important tool that will facilitate their analysis.

Standard Deviation are presented. Finally, the last parameter of the data introduced is the *Median*, and its related parameters *Quantiles* and *Range*.

Characteristic

This section introduces the concept of a *characteristic*, its definition, and its types.

Definition of Characteristic

In the introductory lesson, we define Data Science as follows:

Science that studies how to obtain knowledge from Data.

From this definition, we can ask ourselves: obtain knowledge from Data, about what?

We can answer the following:

About those that we observe of things and events.

And those that we observe of things and events are their characteristics. For that reason, before introducing the concept of data, it is very important to have knowledge of what is a *characteristic*.

If we take the definitions that can be found in the dictionaries about the terms *property*, *attribute*, *quality,* and *characteristic*, we have for property: "Attribute or essential quality of someone or something"; for attribute: "Each one of the qualities or properties of a being"; for quality: "Each one of the characteristics, natural or acquired, that distinguish people, living beings in general or things"; and for characteristic: "Said of a quality: That gives character or serves to distinguish someone or something from their peers". From these definitions, two immediate conclusions can be reached: the first one is that they are synonymous with each other, so any of them can be used to name what data science is going to focus their studies on; and the second one is that a collection of characteristics, which can also be called an *instance, record*, or *case*, describes an object. In this text, from all of them, we have selected the term *characteristic*.

It is commonly accepted to use the term *Variable* interchangeably to refer to the term characteristic, although their definitions are not equal. We introduce the definition of Variable here, but we recommend that the reader return to its definition when the *Experiment, Population,* and *Frequency* concepts had been defined, later in this lesson, and even later, after studying the *Probability* lesson, for a better understanding. *Variable* can be defined as "Magnitude that can have any value of those included in a set" or "A factor in a scientific experiment that may be subject to change". And a *Random Variable* can be defined as "A variable associated with a certain probability law or distribution, in which each of the values it can take corresponds to a specific relative or probability frequency". And a *Statistical*

Variable can be defined as "Function defined on a finite population or a sample, which takes the values of each one of the modalities of an attribute, and to which it associates a frequency distribution".

In this book, we will start using the term characteristic to gradually switch to using variable[6] in future lessons.

Types of Characteristics

The characteristics can be of two types:

1. *Quantitative*, or measurable,[7] means measurable that can be measured.[8] Consequently, their values are going to be numbers in which arithmetic operations[9] can be applied.

Quantitative, or measurable, characteristic. To introduce the concept of quantitative, or measurable, characteristics through an example, we can use the distances, measured in kilometres, km, between their homes and the University for the group of students of the Data Science subject of one academic course; the time each of them takes to travel the previous distance; the number of different subjects in which each one of them is enrolled in the first semester of the academic course; or the number or social networks used for every student.

2. *Qualitative*, which cannot be measured, and consequently arithmetic operations cannot be performed with their values. Qualitative characteristics are also usually called A*ttributes*.

[6] When the term *Characteristic* is replaced by the term *Variable*, it is important to note that in some experiments, the values of the data for that characteristic can be constants and it is possible to call a variable to a constant, but in that cases, the term variable is only replacing the term characteristic and it has not its etymological meaning.

[7] Quantitative data, to facilitate their mathematical treatment, can be rounded and truncated. Rounding can be defined as: "Dispense, in quantities, of small differences in more or less, to take into account only units of a higher order"; and defines truncate as: "From lat. truncāre. Cut a part to something". Both rounding and truncation applied to numbers replace a number with a shorter one, whose value is not exactly the same but is very approximate. If $x \in \mathbb{R}$ and it is expressed in decimal form, it can be rounded by following the next three rules: 1. If the first digit removed is less than 5, then the value of the first digit not removed is kept. 2. If the first digit removed is greater than 5, or is 5 followed by digits greater than 0, then the value of the first digit not removed is increased by one. 3. If the first digit removed is 5, or is 5 followed by digits equal to 0, then the value of the first digit not removed is changed to the even number that makes it closest to the number resulting from the rounded number. To truncate a number, the rule is followed: If $x \in \mathbb{R}$ and it is expressed in decimal form, it can be truncated by eliminating the digits to be eliminated. Digits not removed remain unchanged.

[8] Measure can be defined as: "Compare a quantity with its respective unit, in order to determine how many times the second is contained in the first".

[9] The arithmetic operations are addition, subtraction, multiplication, and division.

Qualitative characteristic. To introduce the concept of qualitative characteristic through an example, we will again use the group of students of the Data Science subject of one academic course, and in this case, we can use the name of their residence towns, the highest course in which each student is enrolled in the academic course, the identification number of the student in the university,[10] or the gender of the student.

3. Binary, or Logical. Its values can be numerical or textual, but in both cases, the meaning of the value will be logical, whether the characteristic exists or not exists.

Logical characteristics. To introduce the concept of logical characteristic through an example, we are going to again use the group of students of the Data Science subject of one academic course, and in this case, we use if each one of them has a driving license or not or if the student has ever worked for someone else.

Data

Once we know the fundamentals of the characteristic concept, let us follow with the concept of *Data*. This section introduces the concept of Data, its definition, and its types[11] from the perspectives of their nature and storage.

Definition of Data

A data can be defined as: the value obtained for a characteristic of the object of study in an observation.[12]

Considering this definition, we denote each data point of the observation of the characteristic x as x_i, so that n observations of the characteristic x for n different individuals or objects would be the following set x:

[10] Although is given by a number, this characteristic and the previous one, are qualitative characteristics because the same identification could have come from the student's name. To know if a characteristic whose data are given by numbers is quantitative or qualitative, it can be reasoned if with said numbers it makes sense to perform simple arithmetic operations such as addition or subtraction. In this case, for example, it is clear that it does not make sense to add two identification numbers of two students.

[11] Every programming language defines their data types and structures.

[12] We think it is interesting to complement this definition with some of those that, for the concept of data, can be found in dictionaries: "1. Information about something specific that allows its exact knowledge or serves to deduce the consequences derived from an event.". In addition, it is curious and, in some manner, surprising and interesting for anyone that comes from the Computer Science field, the following definition: "3. Information arranged in a suitable way for processing by a computer." And the term *Observation* is related to the term *Experiment* that will be introduced in the next section.

$$x = \{x_1, x_2, \ldots, x_k, \ldots, x_n\}$$

We could also observe a set of characteristics x_i for the same individual or object, and this situation can also be written with a similar set as in the previous case.

$$x = \{x_1, x_2, \ldots, x_k, \ldots, x_n\}$$

However, in this case, x is the individual or object, and x_i is the data of every one of the characteristics observed in an observation, while in the previous case, x is the characteristic, and x_i are the values of the data observed for each individual or object.

We can also have a set of n individuals or objects for which m different characteristics have been observed. This situation can be represented with a matrix, in which each row (also it could be written transposed, with the values of the rows in the columns) represents a different individual or object, and each column represents the value, data, observed for each characteristic identified for that individual or object. The matrix would be:

$$x = \begin{pmatrix} x_{11} & \cdots & x_{1k} & \cdots & x_{1m} \\ \vdots & \ddots & \vdots & & \vdots \\ x_{j1} & \cdots & x_{jk} & \cdots & x_{jm} \\ \vdots & & \vdots & \ddots & \cdots \\ x_{n1} & \cdots & x_{nk} & \cdots & x_{nm} \end{pmatrix}$$

From that matrix, it is possible to introduce the concept of *Instance* as the set of values of each row; that is, in the previous case, we would have n instances of m values for each one of them. In this case, each instance would be composed of the values of the m observed characteristics of an individual or object. If only one characteristic is observed, an instance is each observation. Synonyms of instance are *register* or *case*.

If the data obtained for a characteristic does not vary in all the observations made of it, it is said that characteristic has a *constant*[13] value. If the data obtained for a characteristic have different values in different observations, the characteristic is said to have a *variable*[14] value. Once it has been established that a characteristic has a constant value, its study usually has ended because it has no interest, so normally the studied characteristics have a variable value, this is another reason complementary to the previously introduced that makes it very common to use the variable name to refer to the characteristics.

[13] Additionally, it can be defined as: "5 . F. Mat. Quantity that has a fixed value in a certain process, calculation, etc." See the example of ordinal qualitative data below.

[14] A complementary definition of variable to the previously introduced is: "Magnitude whose values are determined by the laws of probability, such as the points resulting from the roll of a die".

Types of Data from Their Nature

From the perspective of their nature, data can have one of three fundamental types,[15] and inside one type, they will have one specific subtype. The definition of the three types and their subtypes are the following:

1. Quantitative data. Numerical value observed for a certain quantitative characteristic of an object in a certain observation. Considering the nature of the numbers used to obtain the data, quantitative data can be expressed as follows:

   1.1. Discrete quantitative data. They can only have values within a countable set of possible values between two given numbers. Consequently, if x_i is the discrete data resulting from the measurement of the characteristic x, then x_i is obtained by means of a whole number and $x_i \in \mathbb{Z}$.

   Discrete quantitative data. To introduce the concept of discrete quantitative data through an example,[16] we use the number of different subjects in which each of the students described in the previous section is enrolled in the academic course in the first semester. The data are {6, 8, 7, 6, 4, 4, 4, 4, 3, 3, 3, 7, 5, 9, 6, 4, 4, 5, 4, 8, 4, 6, 3, 5, 5, 4, 3, 5, 5, 6, 4, 7, 7, 7, 7, 5, 9, 3, 5, 8, 5, 7, 9, 3, 3, 3, 3, 6, 3, 3, 1, 6, 6, 7, 7, 5, 4, 3, 7, 7, 4, 7, 5, 8, 3, 4, 4, 6, 5, 5, 4, 5, 6}.

   Instance. An example of an instance is the number of courses of the first student, which is 6. If we had more than one characteristic observed for the student, the set of the values of all the data for all the characteristics in each specific observation would be an instance.

   1.2. Continuous quantitative data. It can take any value between two given numbers.[17] Consequently, if x_i is the continuous data resulting from the measurement of the characteristic x, then x_i is obtained by means of a Real number and $x_i \in \mathbb{R}$.

   Continuous quantitative data. To introduce the concept of continuous quantitative data through an example, we are going to use the distances, measured in kilometres, km, between their homes and the University[18] for the group of students of the Data Science subject of one academic course. The data are

[15] Which will correspond to the three types of characteristics.

[16] All the data used for this example and for the rest of the lesson are real data obtained from the students enrolled in one academic year.

[17] Although subject to a resolution of measurement or quantification.

[18] When the book is used as the textbook of a Data Science course: It can be interesting to ask to the students at the beginning of the course to collect all these data in the reality to use and analyze them during the course. It is not difficult to do it. As instance: to collect these distances they can follow the simple way to do it as it is to enter google maps and in the text box search write *distance*. A new text box is opened and there they can write the address of their house and the address of the university. They get the distance, and the provided time to travel it.

{16.5, 34.8, 20.7, 6.2, 4.4, 3.4, 24, 24, 32, 30, 33, 27, 15, 9.4, 2.1, 34, 24, 12, 4.4, 28, 31.4, 21.6, 3.1, 4.5, 5.1, 4, 3.2, 25, 4.5, 20, 34, 12, 12, 12, 12, 12, 12, 5, 19, 30, 5.5, 38, 25, 3.7, 9, 30, 13, 30, 30, 26, 30, 30, 1, 26, 22, 10, 9.7, 11, 24.1, 33, 17.2, 27, 24, 27, 21, 28, 30, 4, 46, 29, 3.7, 2.7, 8.1, 19, 16}.

2. Qualitative data. Textual description observed for a certain qualitative characteristic of an object in a certain observation. Qualitative data can be distinguished between nominal, when the text only serves to describe the data, and ordinal, when the texts corresponding to each data also serve to establish an ordering thereof. Although this distinction can be made, qualitative data have not been separated into two different types, as has been done above with quantitative data, because in both cases, the data are of the same nature. It may be the case that the nominal values of a characteristic are numbers, but they will only have a qualitative value, that is, of distinction of the possible values that the characteristics may have, but arithmetic operations cannot be carried out with them.

Qualitative Nominal data. To introduce the concept of Qualitative nominal data through an example, we again use the group of students of the data science subject of one academic course, and in this case, we use the name of their residence towns as an example of nominal qualitative data. The data are[19]: {Villalbilla, Ensanche de Vallecas, Villalbilla, Alcalá de Henares, Alcalá de Henares, Alcalá de Henares, Cifuentes, Cifuentes, El Casar, Fuente el Saz del Jarama, ND, Coslada, Daganzo de Arriba, Alcalá de Henares, Alcalá de Henares, Arganda del Rey, Coslada, Alcalá de Henares, Alcalá de Henares, Madrid, Madrid, Mejorada del Campo, Alcalá de Henares, Alcalá de Henares, Alcalá de Henares, ND, ND, ND, Alcalá de Henares, Guadalajara, Guadalajara, Torrejón de Ardoz, Torrejón de Ardoz, Torrejón de Ardoz, Torrejón de Ardoz, Alcalá de Henares, Torrejón de Ardoz, Guadalajara, ND, ND, Chiloheches, Alcalá de Henares, ND, ND, ND, Guadalajara, Guadalajara, Coslada, Guadalajara, Cabanillas del Campo, Alcalá de Henares, Madrid, ND, ND, Daganzo, Alcalá de Henares, Torres de la Alameda, Velilla de San Antonio, Daganzo, Guadalajara, ND, Guadalajara, Cobeña, Galapagos, Madrid, Alcalá de Henares, Madrid, Coslada, ND, Alcalá de Henares, ND, Alovera, Torrejón de Ardoz}

Qualitative Ordinal data. To introduce the concept of Qualitative ordinal data through an example, we use the highest course in which each participant is enrolled in the academic course. The data are {4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4}. These data are ordinal because in the case in which they were variable, all the students could be ordered by the highest course in which they are enrolled, that is, we can take another group of five students with the following data: (Name of the student, Highest Course) {John, 3; Martha, 4; Mike, 3; Tom, 2; Caroline, 1}. We can order them from the students enrolled in the highest course to the lowest one in that manner: Martha, John, Mike, Tom, Caroline.

[19]The list are towns near to Madrid, Spain, or Madrid itself.

3. Logical data. Logical value, usually of existence or not, is observed for a certain logical characteristic of an object in a certain observation. Logical data can be treated directly as logical data but can also be treated as discrete quantitative data, assigning, for example, a 1 to the existence of the characteristic and a 0 to the nonexistence, or vice versa, or as qualitative data, assigning, for example, a true or T to existence and a false or F to nonexistence.

Logical data. To introduce the concept of logical data through an example, we are going to again use the group of students of the Data Science subject of one academic course, and in this case, we use whether each student has a driving license or not. The data are as follows: {No, Yes, Yes, No, No, No, No, Yes, No, No, No, No, No, No, No, No, No, No, No, No, No, No, No, No, No, No, No, No, No, No, No, No, No, No, Yes, No, Yes, No, No, No, No, No, No, No, No, Yes, No, No, No, No, No, No, No, Yes, No, No, No, No, Yes, No, No, No, No, No, Yes, No, No, No, No, Yes, No, No, No, No, Yes, No, No, Yes, Yes, Yes, No, No, No, Yes}. Other example of logical data is the data about if the group of students have ever worked for someone else, and the data are: {No, No, No, No, Yes, No, No, No, No, No, No, No, No, No, Yes, No, No, No, No, No, No, No, No, Yes, No, No, No, No, No, No, No, No, No, No, No, No, No, Yes, No, No, No, No, No, No, No, No, No, No, No, Yes, No, No, No, No, No, No, No, No, No, No, No, No, No, No, No, No, No, No, No, No, No, No, No, No, No, No, No, No, No, No, No, Yes, Yes, No, No, No, No, Yes, No, No}

Once the different data types from the perspective of their nature have been introduced and before introducing the types of data from the perspective of their storage, it is important to see another view of the data from the perspective of their values obtained in the different observations of an experiment.[20,21] From this perspective, there can be two types of values for the data:

- *Variable.* Data are variable when their values for a specific characteristic change in the different observations of an experiment.

Variable Data. As an example of variable characteristics, any of those collected in the examples of quantitative or qualitative variables can be taken, with the exception of the example of qualitative ordinal data, the highest course in which each one of the students is enrolled in the academic course, which is a constant.

- *Constant.* Data are constant when their values for a specific characteristic do not change in the different observations of an experiment. Another definition of a constant is "Quantity that has a fixed value in a certain process, calculation, etc."

Constant Data. The example of qualitative ordinal data, the highest course in which each one of the every student is enrolled in the academic course, is also an example of a constant because how Data Science is a subject of the fourth course and

[20] The definition of Experiment will be introduced in the next section.

[21] It is the data that is constant or variable, not the characteristic, because a characteristic can have constant data in an experiment and variable data in other. This explanation will be extended in the examples.

this course is the highest one of the studies none student can be enrolled in a higher course, for that reason all the data observed in all the observations are the same, and consequently the characteristic highest course in which the student is enrolled has a constant value in this analysis, and it is an example of a constant. It can be seen from this example that a characteristic can be a constant in an analysis and a variable in another because if we were applying this characteristic to, for example, all the students enrolled in the degree, it will be variable.

Types of Data from Their Storage

The observable data may have different dimensions, structures, and models, that is, they can be formed by the value of a single characteristic, of two, or of *n*. With this in mind, their registration structures and models can be different, and data types can be defined in the following ways:

- *Formal Data.* Those are data described via formal data. This definition of the data is used in the majority of structured data. Examples are data stored in databases, archives, etc.
- *Formalized Grammar.* Those are data described via a formalized grammar. Examples are machine-generated textual data or forms.
- *Standard Format.* Those are data described via a standard format. Examples are digital images, audio, or video files.
- *Arbitrary textual* or *Binary Data.*

Related to the Data Models, some of the more used are:

- *Structured* data. Data are defined by a model or relations between characteristics. Data can be observed and recorded individually in such a way that what is obtained is a continuous set of records. This is what happens when a single feature is observed. However, the most common is that data analysis is performed on more complex structures. Next, we will see that some of the most common are:
  - Records. The data consist of a collection of records of a set of characteristics or elementary events that are the same for all. Each record is a separate record or event.[22] When data are collected for the simultaneous study of more than one characteristic, the data for each one of them can be of different types; for example, for two characteristics, corresponding to the n-tuples c and d between which a binary relation exists, the following combinations can be given: c and d Qualitative; c Qualitative and d Measurable or vice versa; c and d Measurable.
  - Matrix. A set of records make up a matrix,[23] as we saw in the definition of data, which can be quantitative, qualitative, or mixed. If it is quantitative, the records can be represented in an r-dimensional space in a scatter diagram.

[22] As an example, this is an array in R, the programming language that we use in the book.

[23] This will be a matrix or a data frame in R, both of them and their differences will be described.

Structured data have a series of properties such as:

- Dimension. When dealing with data, one should try to avoid the curse of dimension (the word dimensionality appears in the texts, but this word does not exist in the dictionary) (it is necessary to introduce here how the dimension is treated in statistics, that there is an extensive treatment based on correlation), which consists in that when the dimension increases, the data becomes grey and the definitions of distance and density between points become less significant. For all this, it is necessary to reduce the dimension. This is done for several reasons: to eliminate irrelevant characteristics and noise, to visualize the data more easily, and to reduce the time and memory used by the data mining algorithms. The dimension analysis techniques are as follows:
- Aggregation. It is about combining two or more values into one. There are different aggregation techniques, among them is clustering, which we will see in a separate chapter.
- Extraction (which we could also call elimination). It is about eliminating redundant characteristics (e.g., price and taxes) or irrelevant (eye colour to predict salary). There are different extraction techniques, such as principal component analysis (which must be developed in depth) or decomposition into singular values.
- Sparseness, that is, the distance between data.
- Resolution. The patterns depend on the scale and gradation of the recorded/measured data.

- *Unstructured* data. In the case of unstructured data, the data can be observed in different sources, but they are not organized as in the previous cases, so preprocessing must be performed to give them structure before they can be analysed. The most common sources are as follows:
  - Documents. Each document is transformed into a record in which each term is an attribute or elementary event, and the value that each record has is the number of times it appears in the document. A special case of a document is a transaction record (shopping cart), where each record includes a set of elementary events or items.
  - Graphs. The information is extracted from graphs, such as the molecular structure or a map.[24]
- *Ordered or sorted* data. They are structured or unstructured data[25] that are connected to each other or follow an order that is relevant for the analysis to be made of them. Examples include sequences of transactions in an association analysis, genomic sequences, or space-time such as a temperature map.

[24] Very fashionable with Twitter and R

[25] This is a clear example where the term Data is used in plural.

- *Semistructured* data. An example is Tables.
- *Key-value pairs.*
- *XML: Hierarchical* data. For example, Document.
- *RDF: Semantic* data. For example, RDF or triple store.

Available Data

As we have just seen, to carry out studies on the characteristics, it is mandatory to collect data. From the previous subsection, we know that data are defined as information that is collected through observation, so in this subsection, we are going to see the concepts associated with the observation that must be done to obtain the available data for the study. These concepts are Experiment, Population, and Sample. The subsection also briefly discusses the quality of data.

Experiment

As mentioned above, the data or values of one or more characteristics are obtained through observations of the characteristic. These observations are made through what is defined as an experiment. An Experiment can be defined as the "action and effect of experimenting". The authors define experiencing as follows: "In the physicochemical and natural sciences, carry out operations destined to discover, verify or demonstrate certain phenomena or scientific principles."

This definition can be expanded and applied not only to scientific areas of knowledge but also to the rest of all areas of knowledge. According to the expected results of the experiments, these can be divided into two main types:

- *Deterministic*, which are those in which the result is completely determined by the initial conditions.
- *Random*, which are those in which the result is or is not determined by the initial conditions or defined by conditions and environment, that are not known or controlled, and, in consequence, it is unforeseen.

Data science studies are usually carried out on results obtained in random experiments.[26]

[26] Many times, the objective of statistical studies on random experiments is to serve as a basis for the search for equations or laws that link identified conditions, or influencing variable, with the experiment result, and make the phenomenon studied deterministic.

Data Population

Population can be defined as a "Set of individuals or things subjected to statistical evaluation by sampling". If the study applying data science is going to be carried out only on the data obtained in the experiment, then we have the entire population of data under study. This will allow you to obtain results with absolute certainty. The population can be described using analytic parameters, as will be described in the next sections.

Data Sample

Sample can be defined as: "Part or portion extracted from a set by methods that allow it to be considered representative of it." If the study applying data science is going to be carried out on more data than have been obtained in the experiment, then there is only a sample of the population data (which is a larger data set). This will lead us to the fact that the results obtained are totally true for the sample studied but only probable for the population. When statistics are applied to a sample of data from a population, Statistical Inference is being carried out. To perform statistical inference, it is necessary to apply Probability. The size of the sample is very important in statistical inference.

As in practically all experiments, it is either impossible or very expensive to obtain the population of interest, so it is necessary to resort to obtaining a sample or a set of samples from which the information can be inferred. As it is immediate to deduce, for the inferences obtained to be valid, the samples defined in the experiment must be as representative as possible. This need to define the quality of the samples gives rise to a whole theory of definition of samples called Sampling Theory. It is very important to establish that it is possible to work with samples if the sample is well obtained and it is statistically representative of the population.

There are different sampling types:

- *Sample Random*. The probability of selecting any event is the same.
- *Sample without replacement*. Selected objects are removed from the population, and they can only be selected once.
- *Sampling with replacement*. Selected objects are not removed from the population, and they can be selected multiple times.
- *Stratified sampling*. The data set is partitioned following some stratification criterion, and random samples are taken from each partition.

As mentioned above, to extend the results of data analysis obtained from samples to the populations to which they belong, it is necessary to apply concepts from probability theory. In Lesson 3, the fundamental concepts of probability will be studied. The sample can be described or used to perform inference for the population.

Data Quality

The data obtained from one or more samples, through sampling, or from the entire population, where available, may present quality problems. The most common data quality problems are as follows:

- *Missing values*. The reasons for missing data may be multiple; for example, it was not possible to collect the values of all the characteristics in all cases, or the sources did not want to provide them (age, salary, etc.). When this is the case, the actions that can be taken to solve it are fundamentally eliminating the instance, eliminating the value, ignoring all the missing values during the analysis, or replacing all the missing values with values generated artificially using probabilistic techniques. (In the cardata exercises we have missing data that must be dealt with before we can do the data analysis)
- *Duplicate values*. Data can be duplicated when data from heterogeneous sources (e.g., the same person with several postal addresses) are mixed. When this case occurs, the action to solve it is to clean the data by eliminating this duplication.
- *Noise*. There is noise in the data when the original data are modified due to the superposition of a signal that modifies the values of the original data (e.g., in an audio signal).
- *Outliers*. These are data whose values are very different from the rest of the values obtained. It is very important to note that outliers may not be a quality problem in the data but have a meaning that needs to be analysed. The treatment of outliers is studied in depth in lesson 4.

Another aspect related to the quality of the data is their *Statistical Reliability* or sometimes called *Veracity,* which measures the degree of repeatability of the observations.

Frequency

Once we have finished introducing the main concepts related to the concept of data, now we are going to start the introduction of the methods that will allow us to know more in depth the set or sets of data that we must study in the analysis that we must do over them. Those methods are called *data description methods,*[27] and their

[27]The description of data has traditionally been part of descriptive statistics, although with the development of new techniques, especially visualization, their current knowledge exceeds the limits of statistics. The description of the data has also been called as Statistical Summary or Exploratory Data Analysis (EDA), definition, the latter given by J. Tukey in his book of the same name, although to our understanding, the EDA should be treated in visualization, since its fundamental philosophy rests on a search for knowledge of the data through its visual analysis, that is, through graphics, in addition to presenting graphic techniques not only for the description of the data but also for other purposes such as the detection of clusters or outliers.

application will facilitate the subsequent analysis that is going to be carried out on them.

Definition of Frequency

The first method of the description of the data that is usually applied is *frequency analysis*, that is, the one that we are going to see in this subsection. We are going to see its definition, its types, its distribution, and the related concept of *Mode*. Another way to obtain a better understanding of the data you are working with is to obtain one or more values that represent all of them, that is, what you are going to get through the data summary measures, or to use the ordered measurements, which will be presented in the next sections.

Once you have the data available, an initial analysis to do over them is to determine how many times each different data value[28] appears, with the double objective of determining the importance of each[29] data value within the data set, since the data that appear more times will have greater weight in the study and to reduce the number of data processed. Associated with this process, the concept of *frequency* is defined.[30,31]

Types of Frequency

In the first level, two fundamental types of frequency are defined: *punctual*, or associated only with the data, and *accumulated,*[32] or associated with the data and all data with a value lower than it. In addition, each of them can be of two types: absolute or relative. The definitions of all of them will be seen in detail below. The frequency can be applied to data of both quantitative and qualitative characteristics or of logical.

---

[28] A certain observation can be made up of a single data or value, a set of values, which can be two, three, or more data.

[29] It is very important to highlight the fact that frequency is a measure that is going to be given for each different data, that is, you can have, for example, 100 data, but only three different data values, a, b, and c, but each of them is repeated a set of times, in such a way that in the end there are 100 data, since there will only be three frequencies, not 100.

[30] Frequency can be defined as: "Number of elements within an interval in a given distribution."

[31] It is very important to understand, know, and be familiar with the concept of frequency, especially with the concepts of relative frequency and accumulated relative frequency, in order to understand and more easily assimilate the concepts of probability and probability distributions that will be seen in the second part of the book dedicated to studying probability.

[32] This type can be only applied to quantitative data.

There are two types of *Punctual Frequency*[33] or frequency:

- *Absolute frequency*. It is defined as the number of times that a given data value appears in a set of observations, that is,

$$f_i = n_i$$

where i is each of the observations or data of different values that has been obtained. If the observations are made up of a single piece of data, the absolute frequency is given by the number of times that data appear in the set of observations that have been made. If the observations are made up of pairs of data, the absolute frequency is given by the number of times a certain pair of data appears, and the same applies for larger dimensions. In the case of using equivalence classes, the frequency of a class is given by the number of data belonging to that class.

We are going to see the concept of Absolute Frequency using the data of the number of different subjects in which each one of the students described in the previous section are enrolled in the first semester of the academic course. The data are: {6, 8, 7, 6, 4, 4, 4, 4, 3, 3, 3, 7, 5, 9, 6, 4, 4, 5, 4, 8, 4, 6, 3, 5, 5, 4, 3, 5, 5, 6, 4, 7, 7, 7, 7, 5, 9, 3, 5, 8, 5, 7, 9, 3, 3, 3, 3, 6, 3, 3, 1, 6, 6, 7, 7, 5, 4, 3, 7, 7, 4, 7, 5, 8, 3, 4, 4, 6, 5, 5, 4, 5, 6}. The first thing that we must do is to identify the set of different data that we have in the observed series of data, and that set is 3, 4, 5, 6, 7, 8, and 9. Once we have this set, we calculate the number of times that is observed or appears each one of them. That is:

If the first different data observed is $x_1 = 3$,

$$f_1 = n_1$$

and n_1 is equal to the number of times that value 3 has been observed in the whole data. If we analyse the set of data, the value 3 appears 14 times, and in consequence,

$$f_1 = 14$$

The rest of the frequencies, with $x_2 = 4, x_3 = 5, x_4 = 6, x_5 = 7, x_6 = 8, x_7 = 9$, are:

$$f_2 = 15,\ f_3 = 13,\ f_4 = 10,\ f_5 = 12,\ f_6 = 4,\ f_7 = 3$$

- *Relative frequency*. It is defined as the number of times a given observation appears in a set of observations divided by the total number of observations, that is,

[33] The punctual frequency of a given observation is usually called simply frequency and is distinguished from the cumulative frequency because it always carries the cumulative name.

$$fr_i = \frac{n_i}{n}$$

where

$$n = \sum_{i=1}^{j} n_i$$

and j is the total number of observations with different values. It is important to consider that the sum of the relative frequencies of the observed data is equal to 1,

$$\sum_{i=1}^{j} fr_i = 1$$

The same considerations as in the previous definition apply to the treatment of one or more data points. In the case of equivalence classes, the relative frequency of a class is given by the number of data belonging to that class divided by the total amount of data in all classes. The new perspective offered by the relative over the absolute frequency is that it gives us a vision of the *relative importance* of a certain piece of data in the data set, that is, a piece of data can have an apparently high absolute frequency, but when considering all the data, through the calculation of the relative frequency, it can be seen that the importance of these data is low because there are many observations.

We are going to see the concept of Relative Frequency using the data of the number of different subjects in which each one of the students described in the previous exercise. From the previous exercise, we know that the sets of different data that we have in the observed series of data are 3, 4, 5, 6, 7, 8, and 9. From the previous exercise, we also know the number of times that each one of them has been observed because there are their absolute frequencies, and they are:where $x_1 = 3$, $x_2 = 4$, $x_3 = 5$, $x_4 = 6$, $x_5 = 7$, $x_6 = 8$, $x_7 = 9$ are:

$$f_1 = 14, f_2 = 15, f_3 = 13, f_4 = 10, f_5 = 12, f_6 = 4, f_7 = 3$$

From its definition above, the relative frequency is:

$$fr_i = \frac{n_i}{n} = \frac{fa_i}{n}$$

where

$n = \sum_{i=1}^{j} n_i$ is the total number of data, which in this case is $n = 14 + 15 + 13 + 10 + 12 + 4 + 3 = 71$

Consequently, the relative frequency of $x_1 = 3$ is:

$$fr_1 = \frac{n_1}{n} = \frac{14}{71} = 0.20$$

The rest are:

$$fr_2 = \frac{15}{71}, fr_3 = \frac{13}{71}, fr_4 = \frac{10}{71}, fr_5 = \frac{12}{71}, fr_6 = \frac{4}{71}, fr_7 = \frac{3}{71}$$
$$\rightarrow \text{(rounded to two decimals)}$$
$$fr_2 = 0,21, fr_3 = 0.18, fr_4 = 0.14, fr_5 = 0.17, fr_6 = 0.06, fr_7 = 0.04$$

The relative frequency must be verified:

$$\sum_{i=1}^{j} fr_i = 1 \rightarrow 0.2 + 0.21 + 0.18 + 0.14 + 0.17 + 0.06 + 0.04 = 1$$

There are two types of *Cumulative frequency*. The accumulated frequency can only be applied to data with both quantitative characteristics since as a step prior to calculating this type of frequency, the data must be ordered by magnitude.

The types of cumulative frequency are:

- *Cumulative absolute frequency*. It is defined as, with the data ordered by numeric value, from the lowest to the highest, the sum of the absolute frequencies of the data lower than the data for which the accumulated absolute frequency is being calculated, plus that of the data itself, that is,

$$fc_k = \sum_{i=1}^{k} f_i$$

  It must be verified that the accumulated absolute frequency of the data, different, of greater value is equal to the total number of data, since it is the sum of the relative frequencies of all the other data plus it.

- *Cumulative relative frequency*. It is defined as, with the data ordered by numeric value, from the lowest to the highest, the sum of the relative frequencies of the data lower than the data for which the accumulated absolute frequency is being calculated, plus that of the data itself, that is,

$$fcr_k = \sum_{i=1}^{k} fr_i$$

It must be verified that the accumulated relative frequency of the data, different, of greater value is equal to 1, since it is the sum of the relative frequencies of all the other data plus it.

We are going to see the concept of cumulative frequency, absolute and relative, using the data of the number of different subjects in which each one of the students described in the previous exercises. From the previous exercises, we know that the set of different data that we have in the observed series of data are 3, 4, 5, 6, 7, 8, and 9. From the previous exercises, we also know that their absolute and relative frequencies are: where $x_1 = 3$, $x_2 = 4$, $x_3 = 5$, $x_4 = 6$, $x_5 = 7$, $x_6 = 8$, $x_7 = 9$

Absolute frequencies are:

$$f_1 = 14,\ f_2 = 15,\ f_3 = 13,\ f_4 = 10,\ f_5 = 12,\ f_6 = 4,\ f_7 = 3$$

The relative frequencies are:

$$fr_1 = 0,\ 20,\ fr_2 = 0,\ 21,\ fr_3 = 0.18,\ fr_4 = 0.14,\ fr_5 = 0.17,\ fr_6 = 0.06,\ fr_7 = 0.04$$

From its definition above, the cumulative absolute frequency is:

$$fc_k = \sum_{i=1}^{k} f_i$$

From the data above and this equation and how the data are ordered by their values, that is, 3 is before 4, 4 is before 5, and so on, the cumulative absolute frequencies for our problem are:

$$fc_1 = 14$$

$$fc_2 = \sum_{i=1}^{2} f_i = f_1 + f_2 = 14 + 15 = 29$$

$$\mathrm{fc}_3 = \sum_{i=1}^{3} f_i = f_1 + f_2 + f_3 = 14 + 15 + 13 = 42$$

and the rest are: $fc_4 = 52, fc_5 = 64, fc_6 = 68, fc_7 = 71$.

How can be seen the cumulative absolute frequency of the highest value is the same as the total number of data points.

The cumulative relative frequency is:

$$fcr_k = \sum_{i=1}^{k} fr_i$$

From the data above and this equation and how the data are ordered by their values, that is, 3 is before 4, 4 is before 5, and so on, the cumulative absolute frequencies for our problem are:

$$frc_1 = 0.20$$

$$frc_2 = \sum_{i=1}^{2} fr_i = fr_1 + fr_2 = 0.20 + 0.21 = 0.41$$

$$frc_5 = \sum_{i=1}^{5} f_i = f_1 + f_2 + f_3 + f_4 + f_5 = 0.20 + 0.21 + 0.18 + 0.14 + 0.17 = 0.9$$

and the rest are:$fc_3 = 0.59, fc_4 = 0.73, fc_6 = 0.96, fc_7 = 1$

The cumulative relative frequency of the highest value is the same as the total amount of data.

Frequency of Grouped Data

In certain cases, when the data analysis is being performed on continuous quantitative data, or in some cases on discrete quantitative data, and the number of data available is very large, it may be useful to group the data in intervals[34] called *equivalence classes*[35] to reduce the amount of data processed and facilitate its analysis. It will be seen later that it is also useful to know the concepts of data grouping when conducting supervised classification studies.

It is important to note that to carry out a grouping of data, it is necessary to perform arithmetic operations, so it is only possible to do so on quantitative characteristics. We are going to see below the definitions and techniques associated with this grouping. The data grouping process can be carried out by following the next four steps[36]:

- The first step in data grouping is to determine the number of groups into which the complete data set is to be divided. Each of these groups will constitute a different equivalence class,[37] so that certain data can only belong to a single equivalence class. The number of classes, n_c, into which the data set is to be divided is

[34] When some of the types of studies are carried out within the framework of the Data Science discipline, such as supervised classification studies, it is useful to apply the concepts of data grouping.

[35] In set theory, an equivalence class is one each of the disjoint subsets of elements into which an equivalence relation divides a complete set.

[36] The definitions of the concepts associated with the grouping will be entered in the step where they are needed.

[37] In set theory an equivalence class is each of the disjoint subsets of elements into which an equivalence relation divides a complete set.

arbitrary and depends on the analyst, but it is recommended that it be a maximum of 10% of the available data, that is, $n_c \leq 0.1\ n$ where n is the number of data available.

- The second step consists of establishing the amplitude of each class, or the difference between the highest and the lowest value of the data belonging to each class. To do this, the first thing to do is sort the data by magnitude, from smallest to largest. In general, classes of the same amplitude are usually defined, for which the procedure consists of calculating the range[38] of the data, or what is the same, the difference between the largest and the smallest value of the observed data for the characteristic under study, that is,

$$r = v_{\max} - v_{\min}$$

and divide this value by the number of classes decided in the first step, that is,

$$a_c = \frac{r}{n_c}$$

It is important to note that, as with the number of classes, the amplitude of the classes can also be arbitrarily decided by the analyst, and they can be different for each class.

- The third step consists of determining the limits and the borders of the classes, or what is the same, the minimum and maximum value of each one of them and which are the borders between all of them. To obtain them, we start from the ordered data and take the lowest value of the data, which will correspond to the minimum value or lower limit of the first class, and add the amplitude, or the first class, if the amplitudes are different for each class, or the one common to all classes, and the data whose value is closest to the result will be the upper limit of that class, and the next data in magnitude will be the lower limit of the subsequent class. This process is repeated until the highest value of the observed data is reached, which will be the upper limit of the last class. When arbitrarily defining the criteria, the limits can also be arbitrarily set.

  Class boundaries are obtained by adding the upper bound of one class to the lower bound of the next class and dividing by two. The boundary thus determined is the boundary of both classes. That is, taking the same argument as in the two previous sections, the boundary between classes a_i and a_{i+1} would be obtained by performing the following calculation: $\frac{b_i+b_{i+1}}{2}$. Class boundaries must not match data in the set being analyzed.

[38] Range can be defined as: "Amplitude of the variation of a phenomenon between a limit clearly specified minor and major. "That can be a good starting definition for the range concept, but that it must be expanded later in this lesson with the text that has been introduced in the paragraph to better understand it.

If borders are not used, but limits are used to establish the divisions between classes, it may be the case that a data in the set coincides with a limit, in which case it would belong to two classes, which it cannot be. To avoid this problem, the criterion is to take the classes as open intervals on the right.

- The fourth and last step in data grouping consists of obtaining a representative of the class, which is usually called the class mark, and which is the value that is substituted for that of all the data belonging to that class, that is, it is as if all the data that belong to that class happened to have that value. As a mark of each class, the midpoint is taken, that is, the lower and upper limits of the class are added, and the result is divided by two, that is,

$$m_c = \frac{l_s - l_s}{2}$$

After grouping, the size of the data set will be reduced according to the criteria taken in the first step, so if the indications given have been followed, the size reduction will have been at least 90%, and it will stop working with the observed data to start working with the class marks.

For grouped data, the frequency is given for each equivalence class, and calculations come from:

- Absolute frequency is given by the amount of data in the observed set that belongs to the class.
- Relative frequency is given by the amount of data in the observed set that belongs to the class divided by the total number of data in the set.
- The cumulative absolute frequency is given by the amount of data in the observed set that belongs to the class and the previous classes in the ordered observed set. Previous classes are those with a mark of the class lower than the mark of the class for which the frequency is being calculated.
- The cumulative relative frequency is given by the amount of data in the observed set that belongs to the class and the previous classes in the ordered observed set divided by the total amount of data in the whole data set. Previous classes are those with a mark of the class lower than the mark of the class for which the frequency is being calculated.

To see an example of grouping data and obtain the frequencies of grouped data, we are going to use the data of the distances of the students of the data science subject to the university, that are: {16.5, 34.8, 20.7, 6.2, 4.4, 3.4, 24, 24, 32, 30, 33, 27, 15, 9.4, 2.1, 34, 24, 12, 4.4, 28, 31.4, 21.6, 3.1, 4.5, 5.1, 4, 3.2, 25, 4.5, 20, 34, 12, 12, 12, 12, 12, 12, 5, 19, 30, 5.5, 38, 25, 3.7, 9, 30, 13, 30, 30, 26, 30, 30, 1, 26, 22, 10, 9.7, 11, 24.1, 33, 17.2, 27, 24, 27, 21, 28, 30, 4, 46, 29, 3.7, 2.7, 8.1, 19, 16}, For those data, we obtain the range and group them into five equivalence classes, including in each class the values in the same ten, the limits and amplitude of each class, and the class mark.

We will apply the steps for grouping:

1. First step: Determine the number of equivalence groups or classes. As there are $n = 73$ distances and the recommended criteria is $n_c \leq 0.1n$, the class number should be $n_c \leq 7$, but how the statement of the problem says that we must be established by tens, we are going to obtain the range of the data to determine the number of classes. The range is

$$\text{range} = v_{\max} - v_{\min} = 46 - 1 = 45$$

Since the highest value is 46 and the lowest is 1, we will have a class of tens for the first five tens. All the classes will have the same Amplitude 10, and the classes will be:

$$[0, 10), [10, 20), [20, 30), [30, 40), [40, 50)$$

2. Second step: Obtaining the amplitude of the classes. The recommended criterion in this case is to obtain classes of equal amplitude. For the statement of the problem, we know that, in this case, this is mandatory because each class will be a ten and, in consequence, the amplitude of all of them will be 10.
3. Third step: Obtaining the borders and the limits of the classes. To obtain the limits of the classes in this case, since the amplitude of the classes has been established by the statement of the problem, the first limit between the first and the second class is:

$$\frac{b_{(1u)} + b_{(2l)}}{2} = \frac{10 + 10}{2} = 10$$

Because the upper limit of the first class and the lowest limit of the second class are the same value, that value is the limit, and the decision of to which class each limit belongs is taken by the analyst, and in this case, it has been chosen that it belongs to the upper class; for that reason, the division of the data into classes remains as follows:

$$[0, 10), [10, 20), [20, 30), [30, 40), [40, 50)$$

4. Fourth step: Determination of the representative of the data group or class brand. To calculate the mark of the two classes, the limits of both are taken, and the equation $m_c = \frac{l_s + l_i}{2}$ is applied in both cases. For the first class, the mark is

$$m_c = \frac{l_s + l_i}{2} = \frac{10 + 0}{2} = 5,$$

for the second class, the mark is

$$m_c = \frac{l_s + l_i}{2} = \frac{20 + 10}{2} = 15.$$

For the third class, the mark is

$$m_c = \frac{l_s + l_i}{2} = \frac{30 + 20}{2} = 25.$$

Applying the same equations for the other two, the marks will be 35 and 45.

This step completes the grouping of the data into the five equivalence classes using the usual criteria.

Once grouped, the data have been reduced close to 90%, it has gone from 71 data to five. Once the grouping is done, only the data would be worked: 5, 15, 25, 35, and 45, as representative of the classes.

Let us now calculate the frequencies of each class. We start for the first [0, 10) with mark 5. Taking into account that the data are:

{16.5, 34.8, 20.7, 6.2, 4.4, 3.4, 24, 24, 32, 30, 33, 27, 15, 9.4, 2.1, 34, 24, 12, 4.4, 28, 31.4, 21.6, 3.1, 4.5, 5.1, 4, 3.2, 25, 4.5, 20, 34, 12, 12, 12, 12, 12, 12, 5, 19, 30, 5.5, 38, 25, 3.7, 9, 30, 13, 30, 30, 26, 30, 30, 1, 26, 22, 10, 9.7, 11, 24.1, 33, 17.2, 27, 24, 27, 21, 28, 30, 4, 46, 29, 3.7, 2.7, 8.1, 19, 16}

We select from them the data that are between 0 and 10, that are:

$$6.2, 4.4, 3.4, 9.4, 2.1, 4.4, 3.1, 4.5, 5.1, 4, 3.2, 4.5, 5, 5.5, 3.7, 9, 1, 9.7, 4, 3.7, 2.7, 8.1.$$

From this, the absolute frequency of the first class, that is, the number of data in the class, is:

$$f_5 = n_5 = 22$$

We write a 5 in the subindex of the class because it is the mark of the class and the number that defines it.

If we do the same for the rest of the classes, we have:

$$f_{15} = 14,\ f_{25} = 20,\ f_{35} = 16,\ f_{45} = 1$$

To calculate the relative frequencies, we must divide the value of the absolute frequency of each class by the total number of data, as is pointed out in the equation

$$fr_i = \frac{n_i}{n}$$

where

$$n = \sum_{i=1}^{j} n_i$$

That, in this case, is

$$n = \sum_{i=1}^{j} n_i = 22 + 14 + 20 + 16 + 1 = 73$$

Consequently, the relative frequency of the first class is

$$fr_1 = \frac{22}{73} = 0.3$$

If we do the same for the rest of the classes, we have:

$$fr_{15} = 0.19,\ fr_{25} = 0.27,\ fr_{35} = 0.22,\ fr_{45} = 0.01$$

and j is the total number of observations with different values. It is important to consider that the sum of the relative frequencies of the observed data is equal to 1,

$$\sum_{i=1}^{j} fr_i = 1$$

That, in our case, is

$$\sum_{i=1}^{j} fr_i = 0.3 + 0.19 + 0.27 + 0.22 + 0.01 \cong 1$$

It is not exactly one by the rounded but it is correct.
For the cumulative absolute frequency

$$fc_1 = 22$$

$$fc_2 = \sum_{i=1}^{2} f_i = f_1 + f_2 = 22 + 14 = 36$$

For the rest:

$$fc_3 = 56,\ fc_4 = 72,\ fc_5 = 73$$

$fc_5 = 73$ verify that it is equal to the total number of data.

For the cumulative relative frequency

$$frc_1 = 0.3$$

$$frc_2 = \sum_{i=1}^{2} fr_i = fr_1 + fr_2 = 0.3 + 0.19 = 0.49$$

For the rest:

$$fc_3 = 0.76,\ fc_4 = 0.98,\ fc_5 = 0.99$$

$fc_5 = 0.99$ verify that it is equal to one.

Frequency Distribution

A *frequency distribution* is the set formed by all the pairs made up of each different value of the observed data and its frequency, the frequency that forms the pair being any of the views in the previous definitions. It will be called with whatever type of frequency it is, for example, if it is the absolute frequency, it will be called absolute frequency distribution and the same for the rest of the types of frequencies seen, both for discrete data and for data groupings. Frequency distributions are usually given as the pair formed by each different value and its frequency for discrete qualitative and quantitative data and as the pair formed by each data interval, or equivalence class, defined in the set of observations, and their frequency for continuous quantitative data, although intervals with discrete quantitative data can also be used.

To see an example of a frequency distribution, we use the absolute and relative frequency of the number of different subjects in which each of the students is enrolled. Those relative and absolute frequencies are:

$$f_1 = 14,\ f_2 = 15,\ f_3 = 13,\ f_4 = 10,\ f_5 = 12,\ f_6 = 4,\ f_7 = 3$$

$$fr_1 = 0,20, fr_2 = 0,21, fr_3 = 0.18, fr_4 = 0.14, fr_5 = 0.17, fr_6 = 0.06, fr_7 = 0.04$$

The values are:

$$x_1 = 3, x_2 = 4, x_3 = 5, x_4 = 6, x_5 = 7, x_6 = 8, x_7 = 9$$

Consequently, the frequency distributions are:
For the absolute frequency:

$$(3, 14), (4, 15), (5, 13), (6, 10), (7, 12), (8, 4), (9, 3)$$

For the relative frequency:

$$(3,0.20), (4, 0.21), (5, 0.18), (6, 0.14), (7, 0.17), (8, 0.06), (9, 0.04)$$

Mode

Once the frequency analysis has been carried out, the data of the variable being analysed are better known, and from said first analysis, a first approximation can be made in the search for a value that can represent the entire data set. The reason is that when you have a significant amount of data, the enumeration of all the different values and their frequencies, although it allows us to better understand the variable, can also give us a very broad set of different values, so one of the fundamental objectives of data description is to explore the possibility of finding a value that can represent the set as a whole. The first concept defined in order to try to represent the entire set of observed data on which the statistical analysis is being carried out and therefore to be able to assign a single value to the variable on which said data have been observed is the *Mode*.

The Mode is the most frequently observed value, that is, the value whose absolute frequency, f_i, is greater than those of the other values. The mode is not always unique; if the sample has two modes, it is *bimodal*; if it has three modes, it is *trimodal*; if it has more, the concept begins to lose its meaning as a representative of the data. The mode is a value that can be obtained for both qualitative and quantitative, discrete, and continuous data; for logical data, this value has no application since there are only two different values.

In the case of non-grouped data, the calculation of the mode consists solely of calculating the absolute frequencies and identifying the data with a higher frequency and calling it the mode of the set of observations, but when the data are grouped in equivalence classes, the calculation is more complicated. When there are intervals, the mode will be a value that will be in the interval in which the absolute frequency divided by the amplitude is greater. To calculate it, we start from the basis that said value will be closer to, of the two intervals contiguous to the interval in which the mode is, the upper and the lower, the one that has a higher frequency–amplitude ratio for different amplitudes or a higher frequency for equal amplitudes. From this conclusion, the following hypothesis is established: the ratio of the distances of the mode to the intervals contiguous to the one in which it is found is inversely proportional to the ratio of absolute frequencies of said intervals if the intervals are of equal amplitude. If the intervals are of different amplitudes, the ratio is inversely proportional to the ratio of absolute frequencies of said intervals divided by their amplitudes.

Considering the hypothesis stated in the previous paragraph, if i is the interval with the highest frequency in which the mode is found, f_{i-1} is the frequency of the interval adjacent to i on its left and f_{i+1} is the frequency of the interval adjacent to i on its right. If we assume that $f_{i-1} > f_{i+1}$, that d_{i-1} is the distance from the mode

value *mo* to the interval $i-1$ and $a_i - d_{i-1}$ is the distance from *mo* to the interval $i+1$, where a_i is the width of the interval i in which the mean lies, then:

$$\frac{d_{i-1}}{a_i - d_{i-1}} = \frac{f_{i+1}}{f_{i-1}} \rightarrow d_{i-1} f_{i-1} = f_{i+1}(a_i - d_{i-1}) \rightarrow$$

$$d_{i-1} f_{i-1} = f_{i+1} a_i - f_{i+1} d_{i-1}) \rightarrow$$

$$d_{i-1} f_{i-1} + f_{i+1} d_{i-1} = f_{i+1} a_i \rightarrow$$

$$d_{i-1} = \frac{f_{i+1}}{f_{i-1} + f_{i+1}} a_i \rightarrow$$

Consequently, if $l_{s_{i-1}}$ is the upper limit of the interval $i-1$, the value of the mode is:

$$mo = l_{s_{i-1}} + \frac{f_{i+1}}{f_{i-1} + f_{i+1}} a_i$$

If the intervals have the same amplitude and if they do not, then the equation for calculating the mode is:

$$mo = l_{s_{i-1}} + \frac{\frac{f_{i+1}}{a_{i+1}}}{\frac{f_{i-1}}{a_{i-1}} + \frac{f_{i+1}}{a_{i+1}}} a_i$$

This equation of the mode is derived as the first one by changing f_{i+1} and f_{i-1} by $\frac{f_{i+1}}{a_{i+1}}$ and $\frac{f_{i-1}}{a_{i-1}}$ in the starting equation.

For the mode we are going to see three examples, one for the data of the number of different subjects in which each one of the students described in the previous sections are enrolled in the academic course first semester: {6, 8, 7, 6, 4, 4, 4, 4, 3, 3, 3, 7, 5, 9, 6, 4, 4, 5, 4, 8, 4, 6, 3, 5, 5, 4, 3, 5, 5, 6, 4, 7, 7, 7, 7, 5, 9, 3, 5, 8, 5, 7, 9, 3, 3, 3, 3, 6, 3, 3, 1, 6, 6, 7, 7, 5, 4, 3, 7, 7, 4, 7, 5, 8, 3, 4, 4, 6, 5, 5, 4, 5, 6}. The different values are:

$$x_1 = 3,\ x_2 = 4,\ x_3 = 5,\ x_4 = 6,\ x_5 = 7,\ x_6 = 8,\ x_7 = 9.$$

Their frequencies are:

$$f_1 = 14,\ f_2 = 15,\ f_3 = 13,\ f_4 = 10,\ f_5 = 12,\ f_6 = 4,\ f_7 = 3.$$

Analysing them, we observe that the value with more data is $x_2 = 4$, 15 data, and in consequence, this set of data has a mode of 4, but since the value $x_1 = 3$ has 14 data, and the value $x_3 = 5$ has 13 data, that is too close we could say that we have a trimodal set of data, with a mode of 3, 4, 5.

The second example is for the data of the distances of the students to the University 16.5, 34.8, 20.7, 6.2, 4.4, 3.4, 24, 24, 32, 30, 33, 27, 15, 9.4, 2.1, 34, 24, 12, 4.4, 28, 31.4, 21.6, 3.1, 4.5, 5.1, 4, 3.2, 25, 4.5, 20, 34, 12, 12, 12, 12, 5, 19, 30, 5.5, 38, 25, 3.7, 9, 30, 13, 30, 30, 26, 30, 30, 1, 26, 22, 10, 9.7, 11, 24.1, 33, 17.2, 27, 24, 27, 21, 28, 30, 4, 46, 29, 3.7, 2.7, 8.1, 19, 16)

The mode is the value 30 with an absolute frequency of 8.

And the third example is for the grouped data of the distances, if we take absolute frequencies of that data, we have:

$$f_5 = 22,\ f_{15} = 14,\ f_{25} = 20,\ f_{35} = 16,\ f_{45} = 1.$$

We write a 5 in the subindex of the class because it is the mark of the class and the number that defines it.

Analysing them, we observe that the class with more data is [0,10], 22 data, and in consequence this set of data has a mode of 5, which is the mark of the interval, but since the class [20,30] has 20 data, that is too close we could say that we have a bimodal set of data, with a mode of 5 and 25. The last number is the mark of the interval [20,30].

Mean

As explained when the concept of Mode has been defined, one of the main objectives to be achieved when performing a descriptive analysis of observed data of a variable is to try to find a value that summarizes the set. Obtaining the mode is the first approximation that is usually made in the search for said value, but not the only one, we are now going to see another of the values calculated in order to obtain a representative of the data that is being analysed, such as is the *Mean*. In this subsection, we are going to see its different definitions and the related concepts of variance and standard deviation.

Definition of Mean

When you have a data set of a quantitative statistical variable, you can define a single measure that can represent the whole set, in such a way that it is not necessary to list all the data to know what data are being treated but it is enough to give only the data that represent them. One of the more known of these measures is called the *Mean*.

The *Mean* can be defined as follows: "Number that results when carrying out a certain series of operations with a set of numbers and that, under certain conditions, can represent the entire set by itself. It receives different names according to the operations carried out to obtain it, such as arithmetic mean, geometric mean, etc. Arithmetic. Quotient of dividing the sum of several quantities by their number. Quadratic. Given the fluctuations of a magnitude, it is called the square root of the

quotient of dividing the sum of the squares of the fluctuations by their number. Geometric. Nth root of the product of numbers. Weighted. Result of multiplying each of the numbers in a set by a particular value called its weight, adding the quantities thus obtained, and dividing that sum by the sum of all the weights. Proportional. Geometric mean of two numbers."

From this definition, we can conclude the following:

- First, the objective sought when calculating the mean is to represent all the data by means of a single value, which will be something like the fact the mean and the mode are included within a group of measures called *central measures* of said data.
- Second, there are many ways to calculate the mean. We are going to see in this section of the lesson the most commonly used, the *arithmetic mean*, but in the last part of the lesson the rest of the means will be studied in depth, depending on the type of analysis we are carrying out, we will have to use a kind of mean or other.

To finish this introduction to the concept of mean, it is interesting to know that the invention of the *Arithmetic*, *Geometric*, and *Harmonic*[39] mean is usually attributed to the Greek philosopher Pythagoras (570–500 BC), but there are documents that prove the use of these three means by different civilizations many years before the birth of Pythagoras; for example, the first documents on the use of the arithmetic mean were written around the year 7290 BC. It belongs to the Babylonian civilization, and it is written in cuneiform language. They describe the calculation of the area of a trapezoid, with two opposite sides of equal length and two opposite sides of different lengths, as the product of the arithmetic mean of the length of the sides of different lengths by the length of one from the two other sides.

Arithmetic Mean

The Arithmetic Mean, $\overline{x}$, or $\overline{x}_a$, of a set of data obtained in n observations is obtained by adding the values of all those data, x_i, and dividing the result by n. Consequently, the equation to calculate the *arithmetic mean is.*[40] Of n data of a variable is:

$$\overline{x} = \frac{\sum_{i=1}^{n} x_i}{n}$$

If the definition of absolute frequency seen above is used, the arithmetic mean can be defined using not all the data that we have but all the different data that we have, which we call x'_j, and their absolute frequencies. As we have n observed values, we have used i as a subscript to differentiate them, and i will go from 1 to n, and of those

[39] The last two ones will be introduced at the end of the lesson.

[40] In statistics texts, it is also called *Average*.

n observed values, we will have m different values to differentiate them from all. For the values we have used x', which refers to the x whose value is different for each one of them, and the subscript j, which will go from 1 to m. If we use x' for each different data point and its frequency f_j, we can define the arithmetic mean as:

$$\overline{x} = \frac{\sum_{j=1}^{m} f_j . x_j}{\sum_{j=1}^{m} f_j}$$

As we know from the definition of the accumulated absolute frequency, the sum of the absolute frequencies of all the different observed elements is equal to the total number of observed elements, so in the definition equation of the mean we have substituted n times $\sum_{j=1}^{m} f_j$, but n could have been left, and from this conclusion, we can obtain a third equation to define the arithmetic mean, and that, from the definition of relative frequency that we know, if we divide each of the absolute frequencies of the m different values that we have by the total number of data n that we have, what we obtain is the relative frequency.[41] Of each of the m different values that we have, we can define the arithmetic mean as:

$$\overline{x} = \frac{\sum_{j=1}^{m} f_j . x'_j}{\sum_{j=1}^{m} f_j} = \frac{\sum_{j=1}^{m} f_j . x'_j}{n} = \sum_{j=1}^{m} \frac{f_j}{n} x'_j$$

Therefore, we can define the arithmetic mean as:

$$\overline{x} = fr_j . x'_j$$

When data are being analysed for which intervals or equivalence classes have been defined, the x'_j will be the marks of each class m_c and their frequencies, absolute or relative, those calculated for each of them.

If weights w_i are used, the equation of calculus for the arithmetic mean is:

[41] It could also be the weight, in the case that weights or values that reflect a different importance are used for different observations, even if the value is observed the same number of times. The weight or relative importance given to each piece of information is usually represented by a w. This mean in some texts is called *weighted mean*.

$$\overline{x} = \frac{\sum_{i=1}^{n} w_i x_i}{\sum_{i=1}^{n} w_i}$$

For the arithmetic mean, we use the example of the distances between the homes of the students and the University. We remember that the data are {16.5, 34.8, 20.7, 6.2, 4.4, 3.4, 24, 24, 32, 30, 33, 27, 15, 9.4, 2.1, 34, 24, 12, 4.4, 28, 31.4, 21.6, 3.1, 4.5, 5.1, 4, 3.2, 25, 4.5, 20, 34, 12, 12, 12, 12, 5, 19, 30, 5.5, 38, 25, 3.7, 9, 30, 13, 30, 30, 26, 30, 30, 1, 26, 22, 10, 9.7, 11, 24.1, 33, 17.2, 27, 24, 27, 21, 28, 30, 4, 46, 29, 3.7, 2.7, 8.1, 19, 16}

We calculate the mean for all the data without groups and grouped.

For all the data, the arithmetic mean is:

$$\overline{x}_a = \frac{\sum_{i=1}^{73} x_i}{n} = \frac{16.5 + 34.8 + \ldots + 19 + 16}{73} = \frac{1353}{73} = 18.53$$

Since some of the data are repeated, if we use their frequencies, we have:

$$\overline{x}_a = \frac{\sum_{j=1}^{47} f_j x_j}{\sum_{j=1}^{47} f_j} = \begin{cases} \dfrac{1.1 + \ldots + 2.3, 7 + \ldots + 8.30 + \ldots + 1.46}{\mathrm{F_n}} \\ \sum_{j=1}^{47} f_j = 1 + \ldots + 2 + \ldots + 8 + \ldots + 1 = 73 \end{cases}$$

$$\overline{x}_a = \frac{1353}{73} = 18.53$$

Now, we are going to calculate the arithmetic mean with the data grouped in the classes defined in the previous exercise.

$$\overline{x}_a = \frac{\sum_{j=1}^{5} f_j x_j}{\sum_{j=1}^{5} f_j} = \begin{cases} \dfrac{22.5 + 14.15 + 20.25 + 16.35 + 1.45}{\sum_{i=1}^{5} f_j} \\ \sum_{i=1}^{5} f_i = 22 + 14 + 20 + 16 + 1 = 73 \end{cases}$$

$$\overline{x}_a = \frac{1425}{73} = 19.52$$

Variance and Standard Deviation

Whenever we have a set of observed quantitative data on which we are performing a statistical analysis, it is possible to calculate its arithmetic mean by applying the equations seen above, and once calculated, there is a single value that, in theory,

represents everyone. However, it is possible that the mean is just a calculated value, but it does not serve its purpose of being a value that represents the data set. This can be due to different reasons, including for example, that the data values are widely separated from each other or that there is one or more values that are very far from the rest. Consequently, it is essential to have a value that allows us to know if the arithmetic mean can be considered a representative of the data or if it is simply a calculated number without any meaning.

The measures of dispersion[42] allow us to solve this problem because they allow us to know how the values of the sample are distributed around values such as the mean or the median, which are intended to represent the sample as a whole. If the dispersion values are high, the summary values of the sample will have little or no meaning, so every measure of the mean or median must always have its dispersion value associated with it.[43] Next, in each of the following, the different existing measures to obtain the absolute dispersion will be described.

In addition to the absolute spread, the relative spread is also obtained by calculating the quotient between the absolute spread and the value used to summarize the data. In the case in which the absolute dispersion has been calculated with the standard deviation and the summary measure of the data is the mean, the relative deviation is called the variation coefficient[44] and is calculated using the equation:

$$cv = \frac{s}{\overline{x}}$$

where $\overline{x}$ is the arithmetic mean and s is the absolute dispersion, usually measured using the standard deviation, a measure that we will see later in this lesson.

The first absolute dispersion measure that we are going to introduce is the *variance*, which will obtain the bonanza of the arithmetic mean as representative of the data set through the calculation of the arithmetic mean of the square of what is called *deviation* from the entire data set to its arithmetic mean, and if the value obtained is high, it means that the data are far from the mean and its value is not valid as a representative of the set. Consequently, to calculate the variance, the first thing we have to calculate is the deviation of each data point with respect to the arithmetic mean calculated for the data set, which is done simply by obtaining the distance of each data point with respect to the mean,

[42] The term dispersion means: Statistical distribution of a set of values.

[43] This is not something that always happens, especially in the media, such as TV or newspapers, where they usually use only the media to make value judgements that, in many cases, do not make any sense.

[44] The variation coefficient is also measured as a percentage, using the equation:

$$cv = 100\frac{s}{\overline{x}}$$

$$x_i - \overline{x}$$

then we square the result of said subtraction,

$$(x_i - \overline{x})^2$$

The reason for doing this is that in the data set, some of them will be greater than the mean $x_i > \overline{x}$ and other minor ones $x_i < \overline{x}$ for which they would be compensated and the variance would always be zero and it would not make sense to calculate it, when raising all the subtractions to the square, they all add up and the problem disappears. Later in the lesson, we will see the properties of the arithmetic mean, and we will see the demonstration that the sum of the deviations of the values of a set with respect to its mean is zero. Once all the squares of all deviations have been obtained, we obtain their arithmetic mean, which, as we know, will consist of adding all the values and dividing it by the number of data we have. According to all of the above, the equation for calculating the variance is:

$$s^2 = \frac{\sum_{i=1}^{n} (x_i - \overline{x})^2}{n}$$

This equation for calculating the variance would be valid when applied to all the observed values of the set, but following the same reasoning that we follow when we define the arithmetic mean, if we apply what we have learned about frequencies in the first part of the lesson, we can work not on all the observed values but only on the different observed values, x'_j, and their frequencies, absolute and relative, in which case we can obtain, as was the case for the mean, two new equations for calculating the variance, the first one using absolute frequencies, f_j, and that is:

$$s^2 = \frac{\sum_{j=1}^{m} f_j \left(x\prime_j - \overline{x}\right)^2}{\sum_{j=1}^{m} f_j}$$

The second one uses relative frequencies, fr_j, that is:

$$s^2 = \sum_{j=1}^{m} fr_j \left(x\prime_j - \overline{x}\right)^2$$

Once we know what the variance is and its meaning and use, we can see immediately that to avoid the problem with the sum of positive and negative distances between the points and the mean, we have used the square of the differences, but it is impossible to compare the variance with the mean because the units of the variance are the squared units of the mean. To solve this problem, a new magnitude is defined, the standard deviation that will be presented next.

For the variance, we use the example of the distances between the homes of the students and the University. We remember that the data are {16.5, 34.8, 20.7, 6.2, 4.4, 3.4, 24, 24, 32, 30, 33, 27, 15, 9.4, 2.1, 34, 24, 12, 4.4, 28, 31.4, 21.6, 3.1, 4.5,

5.1, 4, 3.2, 25, 4.5, 20, 34, 12, 12, 12, 12, 5, 19, 30, 5.5, 38, 25, 3.7, 9, 30, 13, 30, 30, 26, 30, 30, 1, 26, 22, 10, 9.7, 11, 24.1, 33, 17.2, 27, 24, 27, 21, 28, 30, 4, 46, 29, 3.7, 2.7, 8.1, 19, 16}.

According to the equation, the variance is:

$$s^2 = \frac{\sum_{i=1}^{n}(x_i - \overline{x})^2}{n} = \frac{(16.5 - 18.53)^2 + (34.8 - 18.53)^2 + \ldots + (16 - 18.53)^2}{73}$$
$$= \frac{9209.59}{73} = 126.16$$

Since some of the data are repeated, if we use their frequencies, we have:

$$s^2 = \frac{\sum_{j=1}^{47} f_j \left(x\prime_j - \overline{x}\right)^2}{\sum_{j=1}^{47} f_j}$$
$$= \begin{cases} \dfrac{4.12 + 264.71 + \ldots + 6.4}{\mathrm{F_n}} \\ \sum_{j=1}^{47} f_j = 1 + 1 + \ldots + 1 = 73 \end{cases}$$

$$s^2 = \frac{9209.59}{73} = 126.16$$

The standard deviation measures the same as the variance and uses the same principle, the distances between the values in the set that is being analysed and their mean, but as a difference from the variance, the measurement unit is the same as the mean, not its squared value, to obtain the equation used:

$$s = \sqrt{\frac{\sum_{i=1}^{n}(x_i - \overline{x})^2}{n}}$$

Using the same principle as in the mean and in the variance definitions, if we use the frequencies, absolute and relative, we have the following equations:

Using the absolute frequency:

$$s = \sqrt{\frac{\sum_{j=1}^{m} f_j \left(x\prime_j - \overline{x}\right)^2}{\sum_{j=1}^{m} f_j}}$$

The second one uses relative frequencies, fr_j, that is:

$$s = \sqrt{\sum_{j=1}^{m} fr_j \left(x\prime_j - \overline{x}\right)^2}$$

That, if it is compared with the variance, it can be seen that it is the squared root of the variance or, on the contrary, the variance is the square of the standard deviation.

$$s = \sqrt{s^2} \text{ or} s^2 = (s)^2$$

For notation, it is commonly accepted to call the standard deviation for the sample[45] as s.

If we use the definition of the standard deviation with the absolute frequency and the square inside the summatory is developed, we have the following equation:

$$s = \sqrt{\frac{\sum_{j=1}^{m} f_i\left(x_j^2 + \overline{x}^2 - 2x_j\overline{x}\right)}{\sum_{j=1}^{m} f_j}} = \sqrt{\frac{\sum_{j=1}^{m} f_i x_j^2 + \overline{x}^2 \sum_{j=1}^{m} f_j - 2\overline{x}\sum_{j=1}^{m} f_j x_j}{\sum_{j=1}^{m} f_j}}$$

$$= \sqrt{\frac{\sum_{j=1}^{m} f_j x_j^2}{\sum_{i=1}^{n} f_i} + \overline{x}^2 \frac{\sum_{j=1}^{m} f_j}{\sum_{j=1}^{m} f_j} - 2\overline{x}\frac{\sum_{j=1}^{m} f_j x_j}{\sum_{j=1}^{m} f_j}} = = \sqrt{\frac{\sum_{j=1}^{m} f_j x_j^2}{\sum_{i=1}^{n} f_i} + \overline{x}^2.1 - 2\overline{x}.\overline{x}}$$

$$= \sqrt{\frac{\sum_{i=1}^{n} f_i x_i^2}{\sum_{i=1}^{n} f_i} - \overline{x}^2} = = \sqrt{\frac{\sum_{i=1}^{n} f_i x_i^2}{\sum_{i=1}^{n} f_i} - \left(\frac{\sum_{i=1}^{n} f_i x_i}{\sum_{i=1}^{n} f_i}\right)^2}$$

As seen, this equation can be calculated directly from the measured data; however, in the previous equation, it was necessary to previously calculate the mean but in this last equation, it is not necessary because the standard deviation is equal to the mean of the squares minus the square of the mean.

For the standard deviation, we use the example of the distances between the homes of the students and the University. We remember that the data are {16.5, 34.8, 20.7, 6.2, 4.4, 3.4, 24, 24, 32, 30, 33, 27, 15, 9.4, 2.1, 34, 24, 12, 4.4, 28, 31.4, 21.6, 3.1, 4.5, 5.1, 4, 3.2, 25, 4.5, 20, 34, 12, 12, 12, 12, 5, 19, 30, 5.5, 38, 25, 3.7, 9, 30, 13, 30, 30, 26, 30, 30, 1, 26, 22, 10, 9.7, 11, 24.1, 33, 17.2, 27, 24, 27, 21, 28, 30, 4, 46, 29, 3.7, 2.7, 8.1, 19, 16}.

According to the equation, the standard deviation is:

[45] We will see the difference in notation between the sample and the population later in this lesson.

$$s = \sqrt{\frac{\sum_{i=1}^{47} f_i x_i^2}{\sum_{i=1}^{47} f_i} - \left(\frac{\sum_{i=1}^{47} f_i x_i}{\sum_{i=1}^{47} f_i}\right)^2}$$

$$= \sqrt{\frac{272.25 + 1211.04 + \ldots + 256}{73} - \left(\frac{16.5 + 34.8 + \ldots + 16}{73}\right)^2}$$

$$= 11.23$$

Median

In neither of the previous steps performed to have a better understanding of the set of data that has been analysed, the frequency, to know the degree of importance of each one of the observed values; and the analysis and the mean, to know if we can find a value that could represent all the values in the whole set, we have not needed to order in no manner the observed values. However, if we order the observed values from the lowest to the highest, we can use a new set of parameters to describe the set of data, which can give new, and in many cases more interesting, information about the set of data. For that reason, from now all the concepts introduced, that will be the range, the median, the quantiles, and the quantiles range, will be applied to an ordered, by the value of the data, from the lowest to the highest, set of a quantitative characteristic.

We are going to use the example of the distances between the homes of the students and the University. We remember that the data are {16.5, 34.8, 20.7, 6.2, 4.4, 3.4, 24, 24, 32, 30, 33, 27, 15, 9.4, 2.1, 34, 24, 12, 4.4, 28, 31.4, 21.6, 3.1, 4.5, 5.1, 4, 3.2, 25, 4.5, 20, 34, 12, 12, 12, 12, 5, 19, 30, 5.5, 38, 25, 3.7, 9, 30, 13, 30, 30, 26, 30, 30, 1, 26, 22, 10, 9.7, 11, 24.1, 33, 17.2, 27, 24, 27, 21, 28, 30, 4, 46, 29, 3.7, 2.7, 8.1, 19, 16}.

According to the above explanation, the values must be ordered: {1, 2.1, 2.7, 3.1, 3.2, 3.4, 3.7, 3.7, 4, 4, 4.4, 4.4, 4.5, 4.5, 5, 5.1, 5.5, 6.2, 8.1, 9, 9.4, 9.7, 10, 11, 12, 12, 12, 12, 12, 13, 15, 16, 16.5, 17.2, 19, 19, 20, 20.7, 21, 21.6, 22, 24, 24, 24, 24, 24.1, 25, 25, 26, 26, 27, 27, 27, 28, 28, 29, 30, 30, 30, 30, 30, 30, 30, 30, 31.4, 32, 33, 33, 34, 34, 34.8, 38, 46}

Once we have the set of data that we are analysing ordered by its values, the first magnitude that we can obtain to know them better is the range.

Range

The *Range*[46] provides a measure of the difference in value between two data points belonging to the sample located in the orders chosen for each one of them. If only called a range, the range measure provides the difference between the largest and smallest values in the sample. In another case, the data positions for which you want to obtain the range will be defined, thus, the interquartile range provides the difference between the values located in the third and first quartiles, or there is other specific range definition as interpercentile range or interdacile range, we will come back over them when the quantiles were explained.

For the range, we use the example of the distances between the homes of the students and the University. We remember that the ordered data are: {1, 2.1, 2.7, 3.1, 3.2, 3.4, 3.7, 3.7, 4, 4, 4.4, 4.4, 4.5, 4.5, 5, 5.1, 5.5, 6.2, 8.1, 9, 9.4, 9.7, 10, 11, 12, 12, 12, 12, 13, 15, 16, 16.5, 17.2, 19, 19, 20, 20.7, 21, 21.6, 22, 24, 24, 24, 24, 24.1, 25, 25, 26, 26, 27, 27, 27, 28, 28, 29, 30, 30, 30, 30, 30, 30, 30, 30, 31.4, 32, 33, 33, 34, 34, 34.8, 38, 46}.

Since the range is calculated as the difference between the highest and the lowest values, in this case, the range is equal to 45 km (46 minus 1).

Median

The most important magnitude applied over the ordered set of values observed for the characteristic that it has been analysed is the Median, that is because as happened with the Mean, the value of the Median is intended that was the representative of the whole set, and in although it is, in the non-specialized data analysis world, less known and used than the mean, in many cases the Median is a better representative of the whole set than the mean.

The median, which can be defined as: “Element of an ordered series of increasing values in such a way that it divides it into two equal parts, higher and lower than it.”. Allows us to obtain the central value of the data ordered according to its magnitude. Consequently, once the data have been sorted by magnitude, the median is calculated as follows:

- For an even number of data is the sum of the two central values divided by 2.

$$\tilde{x} = \frac{x_{n/2} + x_{(n/2)+1}}{2}$$

- For an odd number of data points, the median is the centre value.

[46]The term range means: Amplitude of the variation of a phenomenon between a clearly specified lower and a higher limit.

$$\tilde{x} = x_{(n+1)/2}$$

The reason to try to find the median value is because if it is the central value, it can make this value a good representative of the whole set, and the reason that it can be a better representative than the mean is because the calculation of its value is not affected, which can be shown in the calculation equations by the value of the more extreme, lower or higher data in the set, as happens with the calculation of the mean.

For the median, we use the example of the distances between the homes of the students and the University. We remember that the ordered data are: {1, 2.1, 2.7, 3.1, 3.2, 3.4, 3.7, 3.7, 4, 4, 4.4, 4.4, 4.5, 4.5, 5, 5.1, 5.5, 6.2, 8.1, 9, 9.4, 9.7, 10, 11, 12, 12, 12, 12, 13, 15, 16, 16.5, 17.2, 19, 19, 20, 20.7, 21, 21.6, 22, 24, 24, 24, 24, 24.1, 25, 25, 26, 26, 27, 27, 27, 28, 28, 29, 30, 30, 30, 30, 30, 30, 30, 30, 31.4, 32, 33, 33, 34, 34, 34.8, 38, 46}.

Since there are 73 values in the data set, which is an odd number, the median is obtained as the centre value of the data set. Therefore, in this case, the median is:

$$\tilde{x} = x_{(73+1)/2} = x_{37} = 20$$

To see an example about the calculation of the median for an even number of data, we suppose that one of the students, the first one, whose distance to the university is 1, changes the university and goes to another one. In this case, we have 72 data points, starting with 2.1.

Since now there are 72 values in the data set, and this is an even number, the median is obtained as the median of the two central values. Therefore, in this case, the median is:

$$\tilde{x} = \frac{x_{n/2} + x_{(n/2)+1}}{2} = \frac{x_{73/2} + x_{(73/2)+1}}{2} = \frac{x_{36} + x_{37}}{2} = \frac{20 + 20.7}{2} = 20.35$$

Quantiles

Once the Median has been introduced and is has been seen that the value that divides the whole set in two equal parts can have meaning and a use, the question that immediately arises is that perhaps to find other values, as the three values that divide the whole set in four equal parts, or other divisions of the ordered set values, can have meaning and use too. The answer is yes, that happens, and the consequence is the definition of the quantiles of the ordered set of observed values that is being analysed.

The quantiles allow us to obtain which of the data we have are in the final stages when we have the data ordered by magnitude, to those three main quantiles have been defined:

- *Quartiles* can be defined as a value that divides the ordered set of observed data into four parts with the same number of observations in each part. Consequently, there are three quartiles. The first quartile is the value that left 25% of the values below it, that is, they are lower than it, and 75% of the values are greater than it. With this definition, it is easy to conclude that the second quartile is the median. The third quartile is the value that leaves 75% of the values below it, that is, they are lower than it, and 25% of the values are greater than it.

  The equations to calculate the quartiles are:

$$\tilde{x}_c = x_{\left[n\frac{c}{4}\right]+1} \ \text{ sin } \ \frac{c}{4} \notin \mathbb{N}$$

$$\tilde{x}_c = \frac{x_{n\frac{c}{4}} + x_{n\frac{c}{4}+1}}{2} \ \text{ sin } \ \frac{c}{4} \in \mathbb{N}$$

  where n is the number of data in the set, and c is the number of quartiles, which can be 1, 2, or 3 for the first, second, or third quartiles, respectively.

For the quartiles, we use the example of the distances between the homes of the students and the University. We remember that the ordered data are: {1, 2.1, 2.7, 3.1, 3.2, 3.4, 3.7, 3.7, 4, 4, 4.4, 4.4, 4.5, 4.5, 5, 5.1, 5.5, 6.2, 8.1, 9, 9.4, 9.7, 10, 11, 12, 12, 12, 12, 12, 13, 15, 16, 16.5, 17.2, 19, 19, 20, 20.7, 21, 21.6, 22, 24, 24, 24, 24, 24.1, 25, 25, 26, 26, 27, 27, 27, 28, 28, 29, 30, 30, 30, 30, 30, 30, 30, 30, 31.4, 32, 33, 33, 34, 34, 34.8, 38, 46}.

Since there are 73 values in the data set, which is an odd number, the quartiles are obtained by applying the first equation as follows:

$$\tilde{x}_c = x_{\left[n\frac{c}{4}\right]+1}$$

For the first quartile, c is equal to 1:

$$\tilde{x}_1 = x_{\left[73\frac{1}{4}\right]+1} = x_{19} = 8.1$$

For the second quartile, c is equal to 2, and its value is the median:

$$\tilde{x}_2 = 20$$

For the third quartile, c is equal to 3:

$$\tilde{x}_3 = x_{\left[73\frac{3}{4}\right]+1} = x_{55} = 28$$

For grouped data, the calculation equation is:

$$\tilde{x}_c = l_{i-1} + \frac{n\frac{c}{4} - \sum_{i=1}^{i-1} n_i}{n_i}.a_i$$

where n_i is the number of data in the interval; c is the number of quartiles, which can be 1, 2, or 3 for the first, second, or third quartiles, respectively; $l_{i\ -\ 1}$ is the limit of the previous interval; and a_i is the amplitude of the interval.

Following the same example, for grouped data, we can consider that all the classes have the same amplitude, 10, so the classes will be:

$$[0, 10), [10, 20), [20, 30), [30, 40), [40, 50)$$

Therefore, we have the following set of data in each class:

$$\begin{aligned}
[010 &= \{1, 2.1, 2.7, 3.1, 3.2, 3.4, 3.7, 3.7, 4, 4, 4.4, 4.4, 4.5, 4.5, \\
&\quad 5, 5.1, 5.5, 6.2, 8.1, 9, 9.4, 9.7\} \\
[10, 20) &= \{10, 11, 12, 12, 12, 12, 12, 13, 15, 16, 16.5, 17.2, 19, 19\} \\
[20, 30) &= \{20, 20.7, 21, 21.6, 22, 24, 24, 24, 24, 24.1, 25, 25, 26 \\
&\quad 26, 27, 27, 27, 28, 28, 29\} \\
[30, 40) &= \{30, 30, 30, 30, 30, 30, 30, 30, 30, 31.4, 32, 33, 33, 34, 34, 34.8, 38\} \\
[40, 50) &= \{46\}
\end{aligned}$$

If we apply the previous equation to obtain the first quartile, c is equal to 1, n is equal to 73, and the amplitude of every class is 10:

$$\tilde{x}_1 = l_{i-1} + \frac{73\frac{1}{4} - \sum_{i=1}^{i-1} n_i}{n_i}.10$$

- *Deciles* can be defined as the values that divide the ordered set of observed data into ten parts with the same number of observations in each part. Consequently, there are nine deciles. The first decile is the value that leaves 10% of the values below it, that is, they are lower than it, and 90% of the values are greater than it.

  The equations to calculate the deciles are:

$$\tilde{x}_d = x_{\left[n\frac{d}{10}\right]+1} \ \sin \ \frac{d}{10} \notin \mathbb{N}$$

$$\tilde{x}_d = \frac{x_{n\frac{d}{10}} + x_{n\frac{d}{10}+1}}{2} \ \sin \ \frac{d}{10} \notin \mathbb{N}$$

  where n is the amount of data in the set, and d *i*s the number of deciles, which can be 1, . . ., 9, for the first to the nine.

For grouped data, it is not used to calculate the deciles, but in the case that is wished to do it, the same equation than with quartiles can be used, only changing the c for the d a*n*d the 4 for a 10, in the equation.

For the deciles, we use the example of the distances between the homes of the students and the University. We remember that the ordered data are: {1, 2.1, 2.7, 3.1, 3.2, 3.4, 3.7, 3.7, 4, 4, 4.4, 4.4, 4.5, 4.5, 5, 5.1, 5.5, 6.2, 8.1, 9, 9.4, 9.7, 10, 11, 12, 12, 12, 12, 13, 15, 16, 16.5, 17.2, 19, 19, 20, 20.7, 21, 21.6, 22, 24, 24, 24, 24, 24.1, 25, 25, 26, 26, 27, 27, 27, 28, 28, 29, 30, 30, 30, 30, 30, 30, 30, 30, 31.4, 32, 33, 33, 34, 34, 34.8, 38, 46}.

Since there are 73 values in the data set, which is an odd number, the deciles are obtained by applying the first equation as follows:

$$\tilde{x}_d = x_{\left[n\frac{d}{10}\right]+1}$$

For the first decile, d is equal to 1:

$$\tilde{x}_1 = x_{\left[73\frac{1}{10}\right]+1} = x_8 = 3.7$$

For the second decile, d is equal to 2:

$$\tilde{x}_2 = x_{\left[73\frac{2}{10}\right]+1} = x_{15} = 5$$

For the rest, we have the following results:

$$\tilde{x}_3 = x_{22} = 9.7$$

$$\tilde{x}_4 = x_{30} = 13$$

$$\tilde{x}_5 = x_{37} = 20$$

$$\tilde{x}_6 = x_{44} = 24$$

$$\tilde{x}_7 = x_{52} = 27$$

$$\tilde{x}_8 = x_{59} = 30$$

$$\tilde{x}_9 = x_{66} = 32$$

- *Percentiles* can be defined as the values that divide the ordered set of observed data into one hundred parts with the same number of observations in each part. Consequently, there are ninety percentiles; the first one is the value that leaves 1% of the values below it, that is, they are lower than it, and 99% of the values are greater than it.

The equations to calculate the percentiles are:

$$\tilde{x}_p = x_{\left[n\frac{p}{100}\right]+1} \text{ sin } \frac{p}{100} \notin \mathbb{N}$$

$$\tilde{x}_p = \frac{x_{n\frac{p}{100}} + x_{n\frac{p}{100}+1}}{2} \text{ sin } \frac{p}{100} \notin \mathbb{N}$$

where n is the number of data in the set, and p is the number of percentiles, which can be 1, . . ., 99, for the first to the ninety-nine data points.

For grouped data, the same equation as with quartiles can be used, only changing c for p and 4 for a 100 in the equation.

For the percentiles, we are going to use the example of the distances between the homes of the students to the University. We remember that the ordered data are: {1, 2.1, 2.7, 3.1, 3.2, 3.4, 3.7, 3.7, 4, 4, 4.4, 4.4, 4.5, 4.5, 5, 5.1, 5.5, 6.2, 8.1, 9, 9.4, 9.7, 10, 11, 12, 12, 12, 12, 12, 13, 15, 16, 16.5, 17.2, 19, 19, 20, 20.7, 21, 21.6, 22, 24, 24, 24, 24, 24.1, 25, 25, 26, 26, 27, 27, 27, 28, 28, 29, 30, 30, 30, 30, 30, 30, 30, 30, 31.4, 32, 33, 33, 34, 34, 34.8, 38, 46}.

Since there are 73 values in the data set, which is an odd number, the percentiles are obtained by applying the first equation as follows:

$$\tilde{x}_p = x_{\left[n\frac{p}{100}\right]+1}$$

For the first percentile, p is equal to 1:

$$\tilde{x}_1 = x_{\left[73\frac{1}{100}\right]+1} = x_1 = 1$$

For the percentile number 18, for example, p is equal to 18:

$$\tilde{x}_{18} = x_{\left[73\frac{18}{100}\right]+1} = x_{14} = 4.5$$

Other examples of percentiles are as follows:

$$\tilde{x}_{39} = x_{29} = 12$$

and

$$\tilde{x}_{52} = x_{38} = 20.7$$

Quantiles Range

Once the quantiles have been introduced, we can return to the concept of *range* applied to all of them, and the definitions are:

- Interquartile range: $Rc = \tilde{x}_{3/4} - \tilde{x}_{1/4.}$
- Intercentile range: $Rp = \tilde{x}_{90/100} - \tilde{x}_{10/100}$
- Interdecyl Range: $Rd = \tilde{x}_{9/10} - \tilde{x}_{1/10}$

For data grouped in equivalence classes, the concept of data is changed to class.

B. Computer-Based Solving

This section starts with an introduction about what computer-based solving means, that is, the application of a systematic process of designing, implementing, and using programming tools to solve the data science subject treated in the lesson.

In this section, in each lesson, the R environment will be used to solve the same cases that had been solved theoretically in the previous sections, and at the same time, the main features of the R environment will be introduced. In this lesson, R will be used for Computer *R*-based Data Description solving.

In this lesson, as this section appears in the book for the first time, the section will start with an introduction of the R Project and the RGUI so that the reader becomes familiar with both of them before starting to use them.[47] In future lessons, other basic knowledge of R, such as the R Packages or the R tool RStudio, will be introduced jointly with the first lesson that uses them.

R Project

The first page of the R Project reads "R is a language and environment for statistical computing and graphics."

The term "environment" is intended to characterize it as a fully planned and coherent system, rather than an incremental accumulation of very specific and inflexible tools, as is often the case with other data analysis software.

R was initially written in 1993 by Robert Gentleman and Ross Ihaka, also known as "R & R", from the Department of Statistics at the University of Auckland, New Zealand. R was inspired by language and environment S, which was developed based on Fortran in 1976 and first published in 1984 at Bell Labs (owned by AT&T, now Lucent Technologies) by John Chambers (and four contributors), who contributed in the early stages of R and later became a member of the core team. In 1988, the R core system was migrated to the C language, and the object-oriented development

[47] In this point, we made a decision about if the R environment and the RGUI must be introduced here in the lesson or in an appendix, but finally was decided to leave them here to avoid that the reader must come and go, many times for this point to the end of the book and in consequence, improve the readability of the book.

paradigm was introduced. Since 1997, the development of R has been managed by the R Development Core Team, belonging to the R Foundation (Fundación *R*).

The first instruction that we will introduce when we open the RGui, as a tribute to all those who developed and work on the R project, will be:

```
> contributors ()
```

The R Foundation is a nonprofit organization that works in the public interest. It was founded by members of the R Development Core Team to:

- Provide support for the R project and other innovations in statistical computing, based on the belief that R has become a mature and valuable tool and consequently ensures its continuous development, as well as the development of future innovations in software for statistical and computational research.
- Provide a point of reference for individuals, institutions, or commercial companies that want to support or interact with the R development community.
- Maintain and manage copyright of R software and documentation.

R is an official part of the GNU project of the Free Software Foundation, and the R Foundation has similar goals to other open source software foundations such as the Apache Foundation or the GNOME Foundation.

You can find definitions of R that say: "R is a functional language intended for data processing" or "intended for statistical studies." With regard to function, this definition refers to the fact that it is based on functions that implement the different functionalities that we need from the language. It is correct that it is fundamentally oriented to data analysis, although its scope is broader than that of statistics, since it also allows, for example, machine learning.

R can be installed on Windows, Mac OS and Unix (Linux).

Website of the R Project

https://www.r-project.org/

All the information about R managed by the R Foundation can be found on the R project website. We are going to navigate all the links in the left column, the first one is Download.

Download

Download takes us to a page where the international mirrors can be found from where we can enter the CRAN. The CRAN can be accessed from any of them, but we are going to go down to the Spanish links, which are under Spain and there are two: cixug, which is the link maintained by the Galician interuniversity consortium Cixug; and Rediris, which is the one that maintains the Spanish national

research network, we enter through either of the two and we are already in the CRAN.

CRAN stands for Comprehensive R Archive Network, and it is the repository where all the downloadable R archives are located and, as we are going to see, there is very useful information.

On the right-hand side of the page, you will find the links to download and install the R environment, which we will see in detail in the next lesson. We are now going to navigate through the CRAN, again using the navigation column on the left, below the R logo.

- Mirrors: Takes us back to the page where all international R mirrors are listed.
- What's new: It takes us to a page where the latest R news published by the R Development Core Team are, within the page there is a set of links to each year's news page, and within each year there is a set of news or announcements.
- Task Views: It gives us a list with links to everything that R can offer us to solve different types of data analysis; there is a link for each type.
- Search: This allows us to search by keywords within the R Project website using the Google search engine.
- R Homepage: Takes us to the main page of the R Project
- The R Journal: It takes us to the R Journal page. We will see the R journal in detail in one of the following sections.

At any time if we want to return to the CRAN main page, we can click on the logo.

The following links are grouped under the headings, Software, as downloadable software, and Documentation, as downloadable documentation.

Software

- *R* Sources: We can download the R source files.
- *R* Binaries: We can download the executable files of R. It takes us to the installation page of R.
- Packages: We can download packages that are extensions of R and are one of the characteristics that make the use of R so useful and interesting. We will see it in more detail in a topic of R packages itself. Until now, we have contributed from the University of Alcalá the authors of this book, they have been LearnClust and LearningRLab, both packages were developed as End-of-Degree Projects and were carried out by students Dennis Monheimius and Eduardo Benito, the first; and Roberto Alcantará, the second. More details on these will be given in the lesson on packages.
- Other: We can download other software related to R.

Documentation

- Manuals: We can download the fundamental R manuals edited by the R Development Core Team and published by the R Foundation. It is composed of seven (7) documents:

  1. An Introduction to R
  2. *R* Data Import/Export
  3. *R* Installation and Administration
  4. Writing R Extensions
  5. The R language definition
  6. *R* Internals
  7. The R Reference Index

There are translations of some of them into other languages in the Contributed link, which we will see below.

- FAQs: We can access the answers for frequently asked questions. They are divided into three types: general questions, questions about R in Windows, and questions about R in MacOS.
- Contributed: It allows us to download other documentation on R developed by users. There is documentation not only in English but also in many other languages. As mentioned above, you can also find some of the official manuals translated into other languages. Currently this page is frozen and is not being maintained.

R Project

Under the R Project heading, there are a series of links related to the R project. Let us navigate through the navigation column on the left of the page under the R Project heading:

- About *R*: It takes us to a page that introduces what R is and what the R environment is. This gives us a new definition: "*R* is a free software environment for statistical computing and graphics."
- Logo: It allows us to download the R logo and informs us about its copyright.
- Contributors: It presents us with the list of the main R developers in their historical evolution and the current core team.
- What's new: It is the same What's new from the CRAN page seen above.
- Reporting Bugs: This explains what to do if we find a bug in R or if we have a patch for a bug that we want to send.
- Conferences: This is a very important link for all those who want to start collaborating with the R community, so, although it is in the same heading as the R Project, we have separated it and put it in bold. This link provides all the information related to the two annual conferences supported by the R Foundation:

1. useR! – International R User Conference
2. DSC – Directions in Statistical Computing.

On two regional conferences, in languages other than English, also supported by the R Foundation:

1. R @ IIRSA. In English
2. Connector. In Spanish
3. LatinR. In Spanish, Portuguese and English
4. *R* Day. In Portuguese

- Search: It allows us to perform searches on R with other possibilities different from the CRAN search engine. It also allows us to enter a very active forum about R, Nabble R Forum, which allows us to consult our doubts with other members of the R community.
- Get involved: mailing list: Mailing lists of topics of interest about R.
- Developers page: It is an intermediate repository of ideas and plans more or less completed for R.
- Rblog: Blog with news about R.

R Foundation

Under the heading R Foundation, there are a series of links related to the R foundation. Let us navigate through the navigation column on the left of the page under the heading R Foundation:

- Foundation: Describes everything related to the R foundation, and its statutes can be downloaded.
- Board: We can see the members of the Board of Directors, the Steering Committee, and the official direction of the R Foundation, currently in the Institute for Statistics and Mathematics, of the Vienna University of Economics and Business.
- Members: We can see the ordinary members of the R Foundation. The election of these members is made on the merits of his work with R.
- Donors: We can see a list of support members, supporting members, and donors. The difference between one and the other is the economic amount with which they have contributed to the support of R.
- Donate: It allows us to make donations to the R foundation and thus obtain the status of Supporting Member or Donor.

Help with R

- This section presents a single link: Getting Help. This link takes us to a page that shows us all the ways to get help with R. In the lesson in which we will see the RGUI, we will study all the help functions of R, and when we see the package lesson, we will see all the help functions with packages.

Documentation

Under the heading Documentation, there are a series of links related to downloadable documentation on *R*. Let us navigate through the navigation column on the left of the page, under the heading Documentation:

- Manuals: It is the same link as the CRAN Manuals link, remember that we can download the fundamental R manuals edited by the R Development Core Team and published by the R Foundation.
- FAQs: It is the same link as the CRAN FAQ link, remembering that we can access the answers to frequently asked questions.
- The R Journal: This is, together with Conferences, a very important link for all those who want to start collaborating with the R community, so, although it is in the same Documentation section, we have separated it and put it in bold. This link provides all the information related to the R Journal published by the R Foundation. R Journal is published biannually, with editions in June and December of each year. On the page, you can find all the numbers and the way to publish articles.
- Books: This page provides an annotated list of some of the books published annually that are related to S or R and may be useful to the R user community. It only describes the books, but they cannot be downloaded.
- Certification: This page offers us two links in principle with little use: The first is on regulatory compliance and validation issues for the use of R in regulated clinical trial settings; and the second is about the development life cycle of R.
- Other: It takes us to links on other downloadable documentation related to R and maintained on other websites, some of it very interesting and some of it in languages other than English.

Links

Under the heading Links, there are three very interesting links related to R. Let us navigate through the navigation column that is on the left of the page, under the heading Links, to see each one of them:

- Bioconductor: Takes us to the Bioconductor website, which is an open-source, open-development software project that provides high-performance tools, downloadable from the page, for analysis and understanding of genomic data. It is primarily based on the R programming language.
- Related Projects: It offers us links to other software projects related to R or based on R, it is very interesting for advanced R users.
- GoSC: It offers us the link that connects R with the Google Summer of Code, Google's Summer of Code. The Google Summer of Code is a global program focused on attracting more than student code developers to open-source software development. Students work on a three-month programming project with an

open-source organization during their break from college. It would be very interesting to work on this program. The email to participate in an open-source development project with the R Foundation as hosting institution is gsoc-r@googlegroups.com.

R Graphical User Interface, RGUI

We start from the R project website: https://www.r-project.org/

To download the RGUI to work with R, click on: download *R.*

When we click, it takes us to the windows page so that we can choose from which window we want to download R. In principle, the R Foundation only admits two windows per country, but if it is believed that downloads to the community can be improved, an institution can request to host one more.

Let us go to those of Spain, there are currently two:

Free software office (CIXUG): https://ftp.cixug.es/CRAN/
Spanish National Research Network, Madrid: https://cran.rediris.es/

We enter any of them (they are mirrors, so they are therefore identical) and download R for the operating system we have. Let us see how the download is done for Windows.

Click on: Download R for Windows.

In the next topic, we will see how to install R.

Then, on the next screen, install R for the first time.

Then, on the next screen, at: Download R 3.4.3 for Windows (62 megabytes, 32/64 bit).

It automatically starts the download in the download folder. Download a single installer:

R-4.2.1-win.exe

R Installation

Click twice and it begins to install R, during the installation it will ask us where we want to install it and if the recommended folder does not exist, it will ask us if we want to create it, to which we will answer yes, and click next.

Then, it will ask us:

- Language: We can choose Spanish, English, or whatever we want, but this is only for the installation.
- Legal information: we must accept the agreement.
- Installation folder: it is convenient to leave the one offered by default to facilitate subsequent downloads and installation of packages.
- If we want to install the Core Files, we answer yes.

If we want to install the version of 32-bit files or 64-bit files, we will select the appropriate one for our processor (in the case of a 64-bit computer, we will be able to select both). We will see when we deal with the packages in depth that there are certain packages that present installation problems in R x64, so it is interesting when R is loaded to also load the R i386 version.

- and if we want to install the message translations, we also select yes. In addition, we click next.

Then, it asks us if we want to use the configuration options, we can say no, because if we say yes, it is better to leave the ones that come by default, which are:

- If you prefer a single MDI document interface, which will open a single window, or SDI, multiple document interface, which will open separate windows, by default we choose MDI.
- If we want help in plain text or HTML, by default, we choose HTML.
- Finally, if we want direct access to R at startup and if we want to register entries, we leave it as it is by default and click next.

Then, we install R on our system.

Starting to Work with the RGui

To start working with the RGui (R Graphical User Interface), we open the program by clicking on the R icon in the start menu and the R Graphical user interface or RGui opens. Two windows open to us:

- The RGui, which is the largest window, is the container.
- The R Console is the console window in which we enter the R code instructions.

In addition to the console, which will always be inside the RGui window, we will be able to find more windows inside the RGui, such as the graphics when we get them.

As we said in the previous section about R Project, the first thing we are going to do when opening the RGui is to introduce our first instruction, which will be:

$$> \text{contributors}\ ()$$

Having paid tribute to the people who created and developed R, and hoping to maybe see our name there sometime, we start working with RGui.

The RGui has seven menus:

1. File/File
2. Edit/Edit
3. View/Visualize
4. Misc
5. Packages/Packages

6. Windows/Windows
7. Help/Help

We will see each of them in detail:

1. File/File: You have twelve (12) options:
   1.1. Source R Code/Interpret source code in R: Loads to the console and executes a code in R or S written in a file with an .R or .S extension, that is, in script.
   1.2. New Script/New Script: Opens a window to write a script (a script is a program stored in a file, which is usually plain text). The file menu is modified and now refers to the active window in which we are writing the script, with five options, file, edit, packages, window and help, and allows us to save it with the save as option. We will choose the name and save directory and it will save it with the extension .R. In the window in which we can write the script, if we press the right button, we have the option run line, execute line, or selection to execute and test the script. The execution will take place on the console.
   1.3. Open Script/Open Script: This allows us to open, in a script window, a script with an .R extension, previously written, to modify it.
   1.4. Display Files//Show File: It opens any file with an .R, .RData, or Rhistory extension in a window within the RGui (one different each time) to see it, but we cannot modify it.
   1.5. Load Workspace/Load Work Area: Loads the variables saved in a file with the extension: .RData from the R session that we have previously saved in the file.
   1.6. Save Workspace/Save Workspace: Saves in a file with extension: .RData, in the working directory in which we are working all the variables (not the instructions) of the R session in which we are working.
   1.7. Load History/Load History: Loads the history saved in a file with the extension: .Rhistory, of all the instructions (only the input instructions, not the results) of the R session that we have previously saved in the file.
   1.8. Save History/Save History: Saves in a file with the extension:.Rhistory, in the working directory, the history of all the instructions (only the input instructions, not the results) of the R session in which we are working.
   1.9. Change dir/Change directory: It is a very useful option in the menu. It indicates the directory in which we are working with the RGui and allows us to change, through a window system, said working directory to the one we want. As a result of this change, the session will take place in said directory. In addition, it also allows us to change to a new working folder that we create when we are executing the directory change. In a new execution of the program, it would return to the initially defined directory.
   1.10. Print/Print: Prints on the printer we select, including a file, for example .pdf, the session in which we are working. Print the entire session, with all the messages, inputs and outputs.

1.11. Save to File/Save to File: Saves the session we are working on in a .txt file. Print the entire session, with all the messages, inputs and outputs.
1.12. Exit/Exit: We will leave the RGui. Before leaving, it asks us if we want to save a workspace image (save workspace image?)

2. Edit/Edit: Which has eight (8) options:

   2.1. Copy/Copy: Copy the selected text.
   2.2. Paste/Paste: Pastes what was copied.
   2.3. Paste commands only/Copy only commands: If a set of instructions is copied, the > symbols are also copied, and when these lines are pasted, execution is erroneous because the program cannot interpret those symbols. If the paste command only statement is used, only clean statements are pasted and the RGui interprets them correctly.
   2.4. Copy and Paste/Copy and Paste: Immediately copy and paste the pasted text on the last line of the console.
   2.5. Select All/Select all: Select everything written in the console.
   2.6. Clear Console/Clean Console: Completely clears the console.
   2.7. Data Editor/Data Editor: Allows you to edit any defined data of any type and fix the changes.
   2.8. Gui Preferences/Graphical interface preferences: Allows you to change multiple RGui features and save different RGui configurations, with the Save option, and load them with the Load option.

3. View/Visualize: Which has two (2) options:

   3.1. Toolbar/Toolbar: Activates or deactivates the R toolbar, which is made up of eight (8) buttons (all of them do what is described in their respective menus):

      1. Open Script/Open Script
      2. Load Workspace/Load workspace
      3. Save Workspace/Save workspace
      4. Copy/Copy
      5. Paste/Paste
      6. Copy and Paste/Copy and Paste
      7. Stop current computation/Stop current computation
      8. Print/Print

   3.2. Status bar/Status bar: Activates or deactivates the status bar that indicates the version of R with which we are working.

4. Misc: Which has eight (8) options:

   4.1. Stop current computation/For current computation: For the execution of the last order that R was executing.
   4.2. Stop all computations/For current computation: To execute all the orders that R was executing.
   4.3. Buffered output/Output with buffer: It stores the outputs as it calculates them.

4.4. Word completion: When it is checked, if we write the first letters of a known word, for example, the name of a function, and press Tab, the RGui automatically completes the word.
4.5. Filename completion/Finish filename: When it is checked, if we write the first letters of a known word, for example the name of a file, and press Tab, the RGui automatically completes the word.
4.6. List Object/List objects: It shows us all the variables that we have defined in the execution of R.
4.7. Remove all objects/Remove all objects: This removes all the variables that we have defined in the execution of R. It asks us for confirmation before carrying out the deletion.
4.8. List search path/List the search path: It lists the environment and the R packages that we have active in the execution of R.

5. Packages/Packages: which has six (6) options. We will see them in detail in the topic of Packages.
6. Windows/Windows: Which has (4) options plus a list of the windows open within the RGui:

6.1. Cascade/Cascade: Distribute as a waterfall.
6.2. Tile Horizontally/Divide horizontally: Distribute horizontally.
6.3. Tile Vertically/Divided Vertically: Distribute vertically.
6.4. Arrange Icons/Organize icons: Organize the minimized icons of the windows that we have opened in the RGui.

7. Help/Ayuda: Which has (12) options:

7.1. Console/Console: If we press this option, a pop-up window appears that indicates all the combinations of keys that can be used to operate the console. It is very important to note that the letters that are indicated in uppercase must be in uppercase to work.
7.2. FAQ on R/FAQ on R: It opens the same web page as the CRAN FAQs link, and the FAQs link of the RProject Documentation section, remember that we can access the answers to frequently asked questions.
7.3. FAQ on R for Windows/FAQ in R for Windows: It opens the same web page as the FAQs for Windows link of CRAN, remembering that we can access the answers to frequently asked questions.
7.4. Manuals (in PDF)/Manuals (in PDF): It provides direct links, that is, if we click, each one of the .pdfs opens to the seven manuals that appear on the Manuals website under the CRAN Documentation heading. In addition, there is another very interesting manual that is the Sweave User. Sweave provides a flexible framework for mixing text and code for automatic report generation. The basic idea is to replace the code with your output, so that the final document only contains the text and the result of the statistical analysis; however, the source code can also be included.

7.5. R Functions (text)/R Functions (text): It presents us with a window in which we can enter the name of the function for which we want help and opens a web page with all the information for that function.
7.6. Html Help/Html Help: Opens the official RProject help page. Among the links that are provided deserves special mention the link Writing R Extensions, which gives us all the information necessary to write new packages in R.
7.7. Search Help/Search help: It opens a window in which, if we enter the search term, for example, of a function, it opens a page with all the links in which we can find information about the search term. It is broader than R function because it not only opens the link to help about the function but also to all the pages where we can find help.
7.8. search.r-project.org: It opens a window in which, if we enter the search term, for example, of a function, it opens a page with all the links in which we can find information about the search term within the RProject website. It is broader than Search Help because it opens all kinds of links or documents that contain the term.
7.9. Apropos/About. . .: Finds all those functions whose name contains the word given as an argument for the packages loaded in memory.
7.10. R Project home page/R Project home page: Opens the R Project home page.
7.11. CRAN home page/CRAN home page: Opens the CRAN home page.
7.12. About/About. . .: It tells us the version and the credits of the RGui.

Data Exercises Solved with R

Once we have studied the R environment and the RGUI, we are ready to start to solve, using R, the cases that have been theoretically solved in this lesson about Data and its description, applying all the concepts introduced in the lesson. We are going to calculate all the magnitudes seen in the theory, that is, we are going to obtain the frequencies, the mean, which will be the arithmetic; the measures of dispersion, standard deviation, and variance, and the measures of ordering, median and quartiles, including percentile 54. To solve this problem, we are going to use a file of data, which will be of type .txt, that is, plain text, and it will be made up of the data of the distances between the homes of the students to the University. We remember that the data are: {16.5, 34.8, 20.7, 6.2, 4.4, 3.4, 24, 24, 32, 30, 33, 27, 15, 9.4, 2.1, 34, 24, 12, 4.4, 28, 31.4, 21.6, 3.1, 4.5, 5.1, 4, 3.2, 25, 4.5, 20, 34, 12, 12, 12, 12, 5, 19, 30, 5.5, 38, 25, 3.7, 9, 30, 13, 30, 30, 26, 30, 30, 1, 26, 22, 10, 9.7, 11, 24.1, 33, 17.2, 27, 24, 27, 21, 28, 30, 4, 46, 29, 3.7, 2.7, 8.1, 19, 16}.

To solve it, the first function that we are going to see in R is:

$$> \text{help}\ ()$$

The help function will open a window in the web browser that will give us all the information about the function that we enter between the parentheses, and if we do not enter any, it will give us information about the help function itself. To see how it works, we are going to find information about the next function that we are going to see from R, which is getwd () and which is a function that tells us the directory in which is working R. The complete instruction is:

$$> \text{help (getwd)}$$

If we introduce the instruction ? followed by the instruction about which we want information, we would obtain the same result as with the help function. Applied to the getwd instruction, it would be:

$$> ?\text{getwd}$$

If we introduce the help.start () function, we will open a window in the browser in which all the information about R is presented. To check it, we introduce the function:

$$> \text{help.start ()}$$

Next, we introduce the function getwd () to see in which directory we are:

$$> \text{getwd ()}$$

If we want to change the working directory, we must use the setwd () function and between parentheses and in quotes we have to put the full path of the directory where we want to work, for example, to work in the R directory located in the root directory of C, we would enter the instruction:

$$> \text{setwd ("C : /R")}$$

Or on a pendrive that the system would have assigned as drive G, in a directory called R the instruction would be:

$$> \text{setwd ("G : /R")}$$

To verify that we have changed the directory, we introduce the function again:

$$> \text{getwd ()}$$

To see what files we have in the working directory, we use the list.files () function. We introduce it to check it:

```
> list.files ()
```

Once we have seen these preliminary instructions, which are important to start working with R, we begin to solve what they ask us in the exercise, for which the first thing we have to enter is the data. Normally, we will not enter the data by keyboard, although in future exercises, we will indicate how to do it, but we will read them from files. We are going to start by reading the data from a .txt text file, for which the first thing we have to do is generate it. It is very important to strictly follow the rules for generating the file because otherwise R will not read it and the data analysis will not be possible.

Rules for Generating the txt File:

- There must be a tab between data and data (if there are more tabs, nothing happens).
- There must be a first column that numbers the rows, except the first, which will have a blank space; a first row with the name of the variables.
- An enter must be entered at the end of the last row.
- The use of semicolons in the separation of decimal numbers is not the same and it is very important to take it into account, hours can be lost because the system does not give us the results we are looking for because we have put a comma where you could only put a point. In the input files, such as the Uranus.txt satellite, the decimal numbers must be separated by periods, never by commas.
- Another problem that can occur when reading the data is, for example, that the name of a town as "Alcalá Henares" is not written in a single word but in two, that is, with a space between Alcalá and Henares. If we do not do any step correctly, it will present us with an error message when we read it. The correct manner would be "Alcala_Henares" or another way to write both words together.

Following these instructions, we introduce the data of the distances between the homes of the students to the University, in a .txt file with the name "distances.txt" as follows:

Distance	
1	16.5
2	34.8
3	20.7
...	
23	3.1
24	4.5
25	5.1
...	
46	30
47	30
48	26
...	

(continued)

Distance	
60	27
61	24
62	27
...	
71	8.1
72	19
73	16

We will assign the data from the file to an array that we will call d using the read.table () instruction. It is very important to keep in mind that the file can only be loaded in R if it is in the working directory of R. To assign a value to a variable, you can use two commands <- or =. We will start using <- because it is very common in R.

$$> \mathrm{d} < - \text{read.table}\left(^{''}\text{distances.txt}^{''}\right)$$

To check that the data have been loaded correctly, we use the print() function. The full instruction is:

$$> \text{print}\ (\mathrm{d})$$

which shows us the matrix d on the screen, or we simply introduce the name of the matrix:

$$> \mathrm{d}$$

And it shows it on the screen. Whenever we enter the name of a variable, it will show it to us on the screen.

If we would like to know the dimensions of the table, we would use the dim () function, the complete instruction would be:

$$> \text{dim}\ (\mathrm{d})$$

If we wanted to order the distances table according to distance, which we would call so, we could use the order () function. As an order argument, we would put the name of the variable by which we want to order the table, which in this case is the variable Distance (it should be noted that since R takes capital letters into account, the R in Distance must be capitalized because this is how we have entered it from the text file), but since said variable is inside a matrix, we have to indicate that we only want that component of the matrix using the $ symbol between the name of the matrix and the variable we want from it. The full instruction is:

$$> \text{do} = \mathrm{d}\ [\text{order}\ (\mathrm{d}\ \$\ \text{Distance}),\ ]$$

The brackets indicate every position inside the matrix, the first position, before the comma, is for the files, and the second, after the comma, is for the columns. For

that reason, the previous instruction means that the files are ordered by distance, and that is applied to all the columns because it is nothing written in the column site after the comma.

To see the result we would enter:

$$> \text{do}$$

If we wanted to order them in descending order, we would use the instruction:

$$> \text{do} = \text{d}\,[\text{rev}\,(\text{order}\,(\text{d \$ Distance})),\,]$$

To see the result, we introduce again:

$$> \text{SW}$$

If we wanted to know the length, that is, the number of data points of a variable, we would use the function length (), and in the argument, we would put the name of the variable; for the radius, the complete instruction would be:

$$> \text{length}\,(\text{d \$ Distance})$$

If we wanted to know the range of the distances, which we would call range, that is, the difference between the maximum and minimum values in the sample, we would use the max () and min () functions. The complete instruction would be:

$$> \text{range} = \text{max}\;\;(\text{d \$ Distance}) - \text{min}\;\;(\text{d \$ Distance})$$

To see the result, we enter:

$$> \text{rank}$$

As we have seen, we have to calculate the range because there is no function in R that gives it to us directly, as we have defined it in theory. To solve this problem, we will start to see how a function is programmed, which will be something of an essential utility in R, so it will be deepened throughout the text. Whenever we want to define a function, we must assign it a variable name, which in this case is range; and we will use the function instruction followed by a parenthesis, inside which we will include the variables that the function is going to use, which in this case is going to be Distance, so previously we are going to define that variable as:

$$> \text{Distance} = \text{d \$ Distance}$$

Then, between braces, we will introduce the instructions that make up the function and that will give us its result, which in this case is only a subtraction. The full instruction is:

$$> \text{range} = \text{function}\ (\text{Distance})\ \{\max\ (\text{Distance}) - \min\ (\text{Distance})\}$$

To check that it works, we introduce:

$$> \text{range}\ (\text{Distance})$$

The next essential aspect to consider is also essential in the use of functions and is the fact that the defined function will only be available for the execution of the program in which it has been defined. For it to be available, we have to save it. To do this, there are different ways, we can save it as a script, through an option that we open from the menu line or from the command line with the dump () function which saves the file, whose name we have put between the parentheses, and it will have the extension .R, in the working directory. In the example, solving the complete statement is:

$$> \text{dump}\ (''\text{range}'', \text{file} = ''\text{range.R}'')$$

Now we are going to use the function ls ( ) to know which variables we have defined in this execution of R, and we can see the function range. To see how we can load the function range stored, we will remove it from the current execution. To do this, we remove it from the current execution using the function rm ( ), including as its attribute the function that we want to remove. The full instructions are as follows:

$$> \text{rm}(\text{range})$$

To check that the function range has been removed, we can again use the function ls ( ) or introduce the instruction:

$$> \text{range}(\text{radio})$$

and see that we obtain nothing. Once this is done, if we want to load the function in another execution of the program, one of the possible, and most used, instructions that we can use is source (). In this case, the complete instruction will be:

$$> \text{source}\ (''\text{rank.R}'')$$

Keep in mind that for the range to work, the variables it uses must be loaded.

Once we have the data in R, we begin to carry out the analysis that the exercise asks of us.

To obtain the absolute frequency of the variable distance, which we call frecabsdist, we use the table ( ) function, and as an argument, we introduce the variable for which we want to obtain its absolute frequency, which is Distance. The full instruction is:

$$> \text{frecabsdist} < - \text{table (d \$ Distance)}$$

To see the result, we enter

$$> \text{frecabsdist}$$

To calculate the accumulated absolute frequency, which we call frecabsacumdist, we use the cumsum () function that gives the accumulated frequency of each previous value, and as an argument, we introduce the absolute frequency of the radius calculated in the previous step. The complete instruction is:

$$\text{frecabsacumdist} < - \text{cumsum (frecabsdist)}$$

To see the result, we enter

$$\text{Frecabsacumdist}$$

To calculate the relative frequency, there is no function in R, so we generate one that we call frecreldist. To define a function, we will again use the function function () to which we will give as an argument the input variable that it will have, which will be x, and then, between braces, we will put the definition of the function. The complete instruction will be:

$$\text{frecrel} < - \text{function (x) \{table (x)/length (x)\}}$$

As we know, the frecrel function would only work in this execution of the program; if we wanted to save it to use it in other executions, we would use the dump () function that would save it in the working directory, and the complete instruction would be:

$$\text{dump } (\text{"frecrel"}, \text{file} = \text{"frecrel.R"})$$

To load the function in other executions of the program, we would use the source () function, and the complete instruction would be:

$$\text{source } (\text{"frecrel.R"})$$

Next, for the variable x, we assign the value of the distances, and the complete instruction is:

$$\text{x} = \text{d \$ Distance}$$

Finally, we calculate the value of the relative frequency function to which we will assign the name frecrelradio, and the complete instruction is:

$$\text{frecreldist} < - \text{frecrel}\ (\text{x})$$

To see the result, we enter:

$$\text{frecreldist}$$

To calculate the accumulated relative frequency, which we call frecrelsacumdist, we do the same as in the case of the absolute frequency, and we use the cumsum () function. In this case, we introduce the relative frequency of the distance as an argument. The complete instruction is:

$$\text{frecrelacumdist} < - \text{cumsum}\ (\text{frecreldist})$$

To see the result, we enter

$$\text{frecrelacumdist}$$

To calculate the mean, which we call md, we use the mean () function, and as an argument, we introduce the variable in this case, when found within a matrix, we know that it is d $ Distance, the complete instruction is:

$$\text{md} < - \text{mean}\ (\text{d \$ Distance})$$

To see the result, we enter

$$\text{md}$$

The measures of dispersion that we will calculate are the standard deviation and the variance, which we will call sdd and vard, respectively, for which we will use the functions sd () and var (), respectively, so the instructions will be:

$$\text{sdd} < - \text{sd}\ (\text{d \$ Distance})$$

$$\text{vard} < - \text{var}\ (\text{d \$ Distance})$$

To see the result, we enter:

$$\text{sdd}$$

As seen, the result is not the same as the one obtained in theory because R uses different equations to calculate the standard deviation and the variance and divides by n–1 and not by n, so to obtain the same results that in theory you have to do the operations:

$$\text{sdd} = \text{sqrt}\left(\left(\text{sdd}^2\right)^* 72/73\right)$$

$$\text{vard} = \text{vard}^* 72/73$$

We check that we already obtain the same as in theory by introducing sdd and vard again.

The ordering measures that we are going to calculate will be the median, the quartiles, and the 54th percentile, which we will call mediand and quart1d, quart2d, quart3d, cuan54d, respectively, for which we will use the median (), and quantile () functions, respectively, so the instructions will be:

$$\text{mediand} < - \text{median}\left(\text{d \$ Distance}\right)$$

$$\text{cuar1d} < - \text{quantile}\left(\text{d \$ Distance}, 0.25\right)$$

$$\text{cuar2d} < - \text{quantile}\left(\text{d \$ Distance}, 0.5\right)$$

$$\text{cuar3d} < - \text{quantile}\left(\text{d \$ Distance}, 0.75\right)$$

$$\text{cuan54d} < - \text{quantile}\left(\text{d \$ Distance}, 0.54\right)$$

To see the result, we enter:

mediand

quar1d

quar2d

cuar3d

cuan54d

We see that in this case, the results are not the same as in theory because the calculation equations used have not been the same either; as part of the exercise, it can be solved as in theory.

C. Data Exercises Solved

This section has two parts. In the first part, a set of exercises solved in detail are presented to allow you to check if all the knowledge has been correctly acquired. The advice is to try to solve the exercises by yourself, and then to get the solution to check it with the proposed one by the book. This procedure will make this section truly useful for you. In the second part, the exercises will be solved in R.

Hand-Made Exercises

1. Give at least an example of each existing type of characteristics using the planets of the Solar system. All the characteristics selected must be real ones.

There are three types of characteristics: quantitative, qualitative, and logical; if we use the planets of the Solar system to have examples of each one of them, we can use the following:

- For Quantitative characteristics, examples can be the number of satellites that each planet has; or the equatorial (maximum) diameter of the planet, the density or the rotation period.
- For Qualitative characteristics, examples can be the name of each planet, which is the principal gaseous component in the atmosphere of each one of them; their positions considered the distance to the sun; or the name of the first planet exploration spaceship for each one of them.
- For Logical characteristics, examples can be if the planet has satellites or not or if the planet is solid or not; the latter can also be considered a qualitative characteristic.

2. Give at least an example of each existing type of data using the planets of the Solar system. All the data selected must be real data.

For each type of characteristic, there are different types of data: for quantitative characteristics, there are discrete and continuous data; for qualitative characteristics, there are nominal and ordinal data; and there are also logic data. If we use the planets of the Solar system to have examples of each one, we can use the following:

For Quantitative Characteristics:

- For Quantitative discrete data, an example can be the number of satellites that each planet has; if we collected them, the data are {Mercury, 0; Venus, 0; Earth, 1; Mars, 2; Jupiter, 67; Saturn, 61; Uranus, 27; and Neptune, 14}. As can be verified, all the values belong to $\mathbb{Z}$, and the defined arithmetic operations for quantitative discrete data can be performed with them; for example, the number of joint satellites of Earth and Mars is 3, which is the sum of the satellites of both planets $ns = ns(Mars) + ns(Earth) = 2 + 1 = 3$.

An example of Instance, record or case. From the previous data, as an example of an instance, we can take the attributes of Jupiter, which define Jupiter from the characteristics treated in Jupiter, which has 67 satellites.

- For Quantitative continuous data, an example can be the equatorial (maximum) diameter of the planet, measured, for example, in thousands of km, that the planet has can be taken. They are Mercury, 4879; Venus, 12,106; Earth, 12,756; Mars, 6794; Jupiter, 142,984; Saturn, 108,728; Uranus, 51,118; Neptune, 48,572. All the values belong to $\mathbb{R}$, and arithmetic operations can be performed with them; for example, the size of Mars is approximately half that of Earth since the

relationship between its diameters is $(Mars)/d(Earth) = 6.794/12.756 = 0.53$, and consequently, Mars has a diameter close to half the diameter of Earth.

Other quantitative continuous data examples are the data of the density of the planet, and they are, in g/cm^3: {Mercury, 5.4; Venus, 5.2; Earth, 5.5; Mars, 3.9; Jupiter, 1.4; Saturn, 0.7; Uranus, 1.3; Neptune, 1.8}.

Or the rotation period, and the data are in hours: {Mercury, 1407; Venus, 24.62; Earth, 23.93; Mars, 24.62; Jupiter, 9.84; Saturn, 10.24; Uranus, 15.6; Neptune, 18.5}.

For Qualitative Characteristics:

- For Qualitative nominal data, examples can be the name of each planet, that if we collected them, the data are {Mercury, Venus, Earth, Mars, Jupiter, Saturn, Uranus, and Neptune}; or which is the principal gaseous component in the atmosphere of each one of them; if we collected them, the data are {Mercury, potassium, K; Venus, carbon dioxide, CO_2; The Earth, nitrogen, N; Mars, carbon dioxide, CO_2; Jupiter, hydrogen, H; Saturn, hydrogen, H; Uranus, hydrogen, H; Neptune, hydrogen, H. As seen, all the data are descriptive textual}.
- For Qualitative ordinal data, an example is their positions considering the distance to the sun. Although the data for characteristics are numeric, they are qualitative because nonarithmetic operations can be performed with them, and they are ordinal because they allow us to obtain an ordination of the planets using them, from the closest to the furthest. The data are: {Mercury, 1: Venus, 2; Earth, 3; Mars, 4; Jupiter, 5; Saturn, 6; Uranus, 7; Neptune; 8}.

Another example of qualitative ordinal data is the name of the first planet exploration spaceship for each one. It is ordinal because the names of the spaceships include the year of exploration, which allows us to sort the spaceships by age or the order of exploration of the planets. The names of the first spaceship that explored the planets are Mercury, Mariner10-1974; Venus, Mariner2-1962; The Earth, not included; Mars, Mariner4-1965; Jupiter, Pioneer10-1972; Saturn, Pioneer11-1979; Uranus, Voyager2-1986; Neptune, Voyager2-1989. It is important to realize that although all these data contain numbers, for example, in the case of Mars 4-1965, they are actually qualitative data since the numbers are identifying and cannot be performed on the same arithmetic operations. What these qualitative data allow us is to order the data; thus, for example, the order of exploration of the planets is Venus, Mars, Jupiter, Mercury, Saturn, Uranus, and Neptune.

For Logical Characteristics:

- Logical data examples can be whether the planet has satellites or not. The data on whether or not the planets have satellites are Mercury, 0; Venus, 0; Earth, 1; Mars, 1; Jupiter, 1; Saturn, 1; Uranus, 1; Neptune, 1. As mentioned above, there are different options to treat logical data. In this case, discrete quantitative data were used, with values of 0 for the absence of satellites and 1 for the existence of satellites.

Another example is whether the planet is solid or not. The data on whether the planet is solid or not are Mercury, true; Venus, true; Earth, true; Mars, true; Jupiter, false; Saturn, false; Uranus, false; Neptune, false. In this case, qualitative data were used, with the values true for solid and 1 false for not solid.

3. Give an example of a characteristic with a variable value of its data in an experiment and an example of a variable. All the characteristics and data selected must be real.

As examples of variable characteristics, or variables, all those that will be described in the previous three exercises can be taken.

4. Give an example of a characteristic with a constant value of its data in an experiment and an example of a variable. All the characteristics and data selected must be real.

As an example of a characteristic with constant data in an experiment, the one corresponding to the number of suns around which the planets orbit can be taken, the result will always be 1 for all of them, so it is a constant characteristic.

5. Given an example of each one of the concepts experiment, population and sample using the existence or absence of satellites can be taken from a planet.

The random experiment consists of observing whether a planet has satellites or not. It is a random experiment because there is no law that determines it, and each planet must be observed to determine if it has satellites or not. As an example of a deterministic experiment, we could take the measurement of the duration t of the translation period of the planets around the sun. Knowing those of a set of input characteristics and the equation of angular motion, the values of the characteristic can be obtained, and they will always be the same for the same input values.

To see the difference between what a population and a sample is, suppose that the object of study is composed only of the Mercury and Venus planets; that is, the population is made up of only these two planets. In this case, the conclusion is, with complete certainty, that the planets do not have satellites. Now suppose that the study population is all the planets of the solar system and we have as a sample, that is, we are only going to observe the eight planets, Mercury and Venus, the conclusion for the sample is, as in the previous case, that the planets do not have satellites, but is that conclusion valid with absolute certainty for the population? Clearly, not, because we know, although we have not observed it, that the rest of the planets do have satellites, and that conclusion would be wrong, so we could only extend that conclusion to the population also indicating the probability, or the degree of certainty that we believe for that statement.

1. The rotation period data, in days, of the planets of the solar system are Mercury, 58; Venus, 0.4; Earth, 1; Mars, 1; Jupiter, 0.4; Saturn, 0.4; Uranus, 0.7; and Neptune, 0.7. From these new data, we give the absolute frequency of these data and which kind of data they are. Give also the absolute frequency of the qualitative data of the principal gaseous component in the atmosphere of the

planets of the solar system; and the absolute frequency of the logical data of the existence or not of satellites in each planet.

- The data of the rotation period are quantitative continuous data, and if they are analysed, it can be seen that the value 0.4 is repeated 3 times on Venus, Jupiter, and Saturn; data 0.7 appears 2 times on Uranus and Neptune; 1 appears 2 times on Earth and Mars; and 58 appears 1 time on Mercury. If we take the equation $f_i = n_i$, we can see that i takes 4 different values $i_1 = 0.4$; $i_1 = 0.7$; $i_3 = 1$; $i_4 = 58$, and the frequencies for each of them are $f_1 = 3$; $f_2 = 2$; $f_3 = 2$; $f_4 = 1$.
- If the main component of the atmosphere is observed, we are analysing qualitative nominal data, and it can be seen that the potassium value, K, is observed 1 time in Mercury; the nitrogen data, N, is observed 1 time on Earth; the carbon dioxide data, CO_2, is observed twice on Venus and Mars; and the hydrogen value, N, is observed 4 times in Jupiter, Saturn, Uranus, and Neptune. If we take the equation $f_i = n_i$, we can see that i takes 4 different values $i_1 = K$; $i_2 = N$; $i_3 = CO_2$; $i_4 = H$, and the frequencies for each of them are $f_1 = 1$; $f_2 = 1$; $f_3 = 3$; $f_4 = 4$.
- When we observe if the planet has satellites or not, we are analysing logical data, and it can be seen that the absence of satellites is observed 2 times, Mercury and Venus, and the existence of satellites is observed 6 times, on Earth, Mars, Saturn, Uranus, and Neptune. If we take the equation $f_i = n_i$, we can see that i takes 2 different values $i_1 = \textit{Not Satellites}$; $i_2 = \textit{Satellites}$, and the frequencies for each of them are $f_1 = 2$; $f_2 = 6$

2. Calculate the relative frequency of the three previous groups of data.

- For the quantitative data rotation period, to calculate the relative frequency, we start from the absolute frequency and divide it by the number of data, as the equation is $fr_i = n_i/n_T$, where n_T is the total number of data, and we know for the calculation of the absolute frequency that i takes 4 different values $i_1 = 0.4$; $i_1 = 0.7$; $i_3 = 1$; $i_4 = 58$, the relative frequencies for each one of them are $fr_1 = \frac{3}{8} = 0.375$; $fr_2 = \frac{2}{8} = 0.25$; $fr_3 = \frac{2}{8} = 0.25$; $fr_4 = \frac{1}{8} = 0.125$. It can be easily verified that $\sum_{j=1}^{m} fr_j = 1 \leftrightarrow 0.375 + 0.25 + 0.25 + 0.125 = 1$
- For the qualitative data, the principal component of the atmosphere, as in the previous case, to calculate the relative frequency, we start from the equation $f_i = n_i/n_T$, and we know for the calculation of the absolute frequency that i takes 4 different values $i_1 = K$; $i_2 = N$; $i_3 = CO_2$; $i_4 = H$ and, in consequence, the relative frequencies for each one of them are $fr_1 = \frac{1}{8} = 0.125$; $fr_2 = \frac{1}{8} = 0.125$; $fr_3 = \frac{2}{8} = 0.25$; $fr_4 = \frac{4}{8} = 0.5$.
- For the logical data, if they have satellites or not, i takes 2 different values $i_1 = \textit{Not Satellites}$; $i_2 = \textit{Satellites}$, and the relative frequencies for each of them are $fr_1 = \frac{2}{8} = 0.25$; $fr_2 = \frac{6}{8} = 0.75$.

3. Calculate the accumulated absolute and relative frequency of the three previous groups of data.

First, it must be noted that it is impossible to calculate the cumulative relative frequency for all three previous groups of data because it can only be calculated for quantitative characteristics, and for this reason, we calculate the cumulative absolute frequency and the cumulative relative frequency only for the rotation period.

- To calculate the accumulated absolute frequency of the rotation periods of the planets, we apply the equation $fa_k = \sum_{j=1}^{k} f_j$, from the smallest to the highest value of the data. We start with the smallest rotation period $i_1 = 0.4$, and the equation for this initial value remains as $fa_1 = \sum_{j=1}^{1} f_1$; consequently, the value is $fa_1 = f_1 = 3$. Next, we calculate the accumulated absolute frequency for the next value in order of magnitude, 0.7, in this case $fa_2 = \sum_{j=1}^{2} f_i = f_1 + f_2 = 3 + 2 = 5$, and finally, we calculate the remaining two: $fa_3 = \sum_{j=1}^{3} f_i = f_1 + f_2 + f_3 = 3 + 2 + 2 = 7$, and $fa_3 = \sum_{j=1}^{4} f_i = f_1 + f_2 + f_3 + f_4 = 3 + 2 + 2 + 1 = 8$, and as 8 is the total number of data, the calculation is correct.

- To calculate the accumulated relative frequency of the rotation periods of the planets, we apply the equation $fra_k = \sum_{j=1}^{k} fr_j$, from the smallest to the highest value of the data. We start with the smallest rotation period $i_1 = 0.4$, and the equation for this initial value remains as $fra_1 = \sum_{j=1}^{1} fr_1$; consequently, the value is $fra_1 = f_1 = 0.375$. Next, we calculate the accumulated absolute frequency for the next value in order of magnitude, 0.7, in this case $fra_2 = \sum_{j=1}^{2} fr_i = fr_1 + fr_2 = 0.375 + 0.25 = 0.625$, and finally, we calculate the remaining two: $fra_3 = \sum_{j=1}^{3} fr_i = fr_1 + fr_2 + fr_3 = 0.375 + 0.25 + 0.25 = 0.875$ and $fra_3 = \sum_{j=1}^{4} fr_i = fr_1 + fr_2 + fr_3 + fr_4 = 0.375 + 0.25 + 0.25 + 0.125 = 1$. As 1 is the amount that we must obtain when we calculate the accumulated relative frequency of the highest value, the calculation is correct.

4. Give the frequency distribution for any of the previously calculated frequencies.

The frequency distribution, as we saw in the theory of the lesson, is the set of pairs of data formed by the different data values observed and each one of their associated frequencies. Since the statement of the exercise asks us for any of them, we are going to select the last one calculated in the previous exercise, the frequencies of the rotation period of the planets, the values are {0.4, 0.7, 1, 58} and their corresponding accumulated relative frequencies are {0.375, 0.625, 0.875, 1}, and the asked frequency distribution is {(0.4, 0.375), (0.7, 0.625), (1, 0.875), (58, 1).

5. Perform a data grouping and give the equivalence classes, first using the grouping criteria described in the lesson, a second with arbitrary decisions, and a third for

tens, in only the smaller ones, those whose radius is lower than 50 km, using data from the satellites of Uranus.[48] For the third grouping criteria, calculate the absolute frequency of each class. The characteristic that we are going to analyse is the radius in kilometres of the satellites. They are Cordelia, 13; Ophelia, 16; Bianca, 22; Crésida, 33; Desdemona, 29; Juliet, 42; Portia, 55; Rosalinda, 27; Belinda, 34; Luna-1986 U10, 20; Puck, 77; Miranda, 235; Ariel, 578; Umbriel, 584; Titania, 788; Oberon, 761; Calíbano, 30; Luna-1999 U1, 20; Sycorax, 60; Luna-1999 U2, 15.

- To solve this exercise first, we use the recommended criteria:

1. First step: Determine the number of equivalence groups or classes. As there are $n = 20$ satellites and the recommended criterion is $n_c \leq 0.1n$, the class number should be $n_c \leq 2$; consequently, the number of classes is 2. For the classes to have the same or similar amplitude, the first class between the first value and the Miranda would be taken whose amplitude would be "Amplitude" = 235-13 = 222, and the second class would be between Ariel and Titania, with an amplitude of 788–578 = 220.

However, other classes could also be defined, such as a class per hundred, and there would be 8 classes, of which 4 would have data: 0–100, 200–300, 500–600 and 700–800.

2. Second step: Obtaining the amplitude of the classes. The recommended criterion in this case is to obtain classes of equal amplitude, for which the first thing to do is sort the data by magnitude. The result of this ordering for the satellites of Uranus is Cordelia, 13; Luna-1999 U2, 15; Ophelia, 16; Luna-1986 U10, 20; Luna-1999 U1, 20; Bianca, 22; Rosalinda, 27; Desdemona, 29; Calíbano, 30; Crésida, 33; Belinda, 34; Juliet, 42; Portia, 55; Sycorax, 60; Puck, 77; Miranda, 235; Ariel, 578; Umbriel, 584; Oberon, 761; Titania, 788. Next, the range of the observed data is calculated as the difference between the largest of the observed data, which in this case is Titania, with 788 km, and the smallest, which is Cordelia, with 13 km, so that the range is $r = v_{\max} - v_{\min} = 788 - 13 = 775$; and once the rank is obtained, we divide its value by the number of classes to obtain the amplitude, that is, $a_c = \frac{r}{n_c} = \frac{775}{2} = 387.5$.
3. Third step: Obtaining the borders and the limits of the classes. To obtain the limits of the two classes, we start from the ordered data and take as the lower limit of the first class the lowest observed value, so the lower limit of the first class is Cordelia, 13. At the lower limit of the first class, the amplitude is added to it to obtain the upper limit of the first class. The result is 13 + 387.5 = 400.5, so the upper limit of the first class will be Miranda, 235, since its distance to the value 400.5 is 165.5, which is lower than Ariel's distance, 578, to the value 400.5, which is 177.5. Consequently, Ariel, 578 will be the lower limit of the second

[48] In this case, we cannot use the data of the planets of the solar system because there are very few.

class. The upper limit of the second class is Titania 788, which is the highest value observed. Since there are no more classes, the third step is finished. The boundary between classes: In this case, there is only one:

$$\frac{b_{(i+1,1)} + b_{(i,k)}}{2} = \frac{578 + 235}{2} = 406.5$$

4. Fourth step: Determination of the representative of the data group or class brand. To calculate the mark of the two classes, the limits of both are taken, and the equation $m_c = \frac{l_s + l_i}{2}$ is applied in both cases. For the first class, the mark is

$$m_c = \frac{l_s + l_i}{2} = \frac{13 + 235}{2} = 124,$$

and for the second class, the mark is

$$m_c = \frac{l_s + l_i}{2} = \frac{578 + 788}{2}\ 683$$

This step completes the grouping of the data in the two equivalence classes using the usual criteria.

Once grouped, the data have been reduced by 90%, it has gone from 20 data to two. Once the grouping is done, only the data would be worked: 124 and 683.

- Second, we are going to group them with arbitrary criteria. The above grouping is correct and has been done using the usual general criteria, but it may not be the most correct or the most useful if the specificity of the data being analysed is considered. If the data are observed, it can be seen that there are 14 of them below 100 km in radius and 11 of them below 50, so although the class mark is 124 because the upper limit is 235 and the calculation has been moved toward higher values, it seems that this value is not sufficiently representative of the data contained in the class. Taking into account the above, so that the marks of the classes are more representative of the data included in the class, arbitrary criteria will be taken both in the selection of the number of classes, as in the amplitude of the same and in the limits of the classes. The grouping process is as follows:

  1. First step: Determine the number of equivalence groups or classes. It is arbitrarily decided to take four equivalence classes in the tens in which there are radius data.
  2. Second step: Obtaining the amplitude of the classes. The width of the classes is also arbitrarily determined to be 100 km.
  3. Third step: Obtaining the borders and the limits of the classes. The limits are established as the minimum and maximum values of the defined classes; consequently, they are 0 and 100 for the first class, 200 and 300 for the second, 500 and 600 for the third, and 700 and 800 for the fourth.

4. Fourth step: Determination of the representative of the data group or class brand. Once we have the limits, the marks of the classes are established using the equation $m_c = \frac{l_s+l_i}{2}$; therefore, they are.[49]

$$m_{c1} = \frac{l_s + l_i}{2} = \frac{0 + 100}{2} = 50$$

$$m_c = \frac{l_s + l_i}{2} = \frac{200 + 300}{2} = 250,$$

$$m_c = \frac{l_s + l_i}{2} = \frac{500 + 600}{2} = 550, \text{and}$$

$$m_c = \frac{l_s + l_i}{2} = \frac{700 + 800}{2} = 750.$$

In this case, once grouped, the data have been reduced by 80%, since it has gone from 20 to 4 data points, but although it is a lower percentage, the class marks are much more representative of the data that the class contains, so these grouping criteria, in this particular case, would be better than the usual ones.

- Third, for tens, in only the smaller ones, those whose radius is lower than 50 km. The obtained set is 13, 15, 16, 20, 20, 22, 27, 29, 30, 33, 34, 42.

1. First step: Determine the number of equivalence groups or classes. It has been decided in the statement of the case to take five equivalence classes in the tens in which there are radius data, from 0 to 50, but we will not consider the first class between 0 and 10 because we have no data in that class.
2. Second step: Obtaining the amplitude of the classes. The width of the classes has also been decided in the statement of the problem as 10 km.
3. Third step: Obtaining the borders and the limits of the classes. The limits are established as the minimum and maximum values of the defined classes; consequently, they are 10 and 20 for the first class, 20 and 30 for the second, 30 and 40 for the third, and 40 and 50 for the fourth. The values in the limits of each class can be arbitrarily joined to one of the two classes, closing the classes by the left or the right, but in all of them, we must have the same closure. In this case, we have decided to close by the left, which means that, for example, the two 20 belong to the second class. The classes are [10,20), [20,30), [30,40), [40,50).
4. Fourth step: Determination of the representative of the data group or class brand. Once we have the limits, the marks of the classes are established using the equation $m_c = \frac{l_s+l_i}{2}$; therefore, they are 15, 25, 35, and 45.

[49] This case is an academic example to better understand the concepts, in a real case, it would be impossible to have a class with only one value, nor with few values, so the problem that the only value of the class does not match the brand of the class is not going to occur in an environment real work.

In this last part of the exercise, we must calculate the absolute frequency of each class, and to do that, we must count the number of observations in each class:

- We start for the first [10, 20), and we observe three values between its limits of 10 and 20: 13, 15, and 16. For that reason, the absolute frequency of class 1 is: $f_{C1} = 3$.
- For the second [20, 30), and we observe five values between its limits of 20 and 30, they are: 20, 20, 22, 27, and 29. As can be observed, the value 20 has been counted in this class because it is close to the left. For that reason, the absolute frequency of class 2 is $f_{C2} = 5$.
- For the third [30, 40), we observe three values between its limits 30 and 40: 30, 33, 34. For that reason, the absolute frequency of class 3 is $f_{C3} = 3$.
- For the last [40, 50), we observe only one value between its limits 40 and 50, that is, 42. For that reason, the absolute frequency of class 4 is $f_{C4} = 1$.

1. Calculate the mode of the data of the radio in km of the satellites of Uranus.[50] Remember that the data are Cordelia, 13; Ophelia, 16; Bianca, 22; Crésida, 33; Desdemona, 29; Juliet, 42; Portia, 55; Rosalinda, 27; Belinda, 34; Luna-1986 U10, 20; Puck, 77; Miranda, 235; Ariel, 578; Umbriel, 584; Titania, 788; Oberon, 761; Calíbano, 30; Luna-1999 U1, 20; Sycorax, 60; Luna-1999 U2, 15.

   The mode is the most frequent value in the data, that is, the value with the highest absolute frequency, and if we analyse the data from the satellites of Uranus, we can observe that all of them have an absolute frequency of 1, which means that all of them appear only one time, with only one exception, the value 20, for which there are two observations, Luna-1986 U10 and Luna-1999 U1, which means that the mode for the set of data of the satellites of Uranus is 2.
2. For the smallest satellites of Uranus, with radius lower than 50 km, calculate the arithmetic mean for data without grouping and grouping by tens.

   If we extract the data of the smaller satellites of Uranus from the data of the satellites of Uranus, we obtain the set 13, 15, 16, 20, 20, 22, 27, 29, 30, 33, 34, 42. If we apply the calculus equations, the arithmetic mean is:

$$\overline{x}_a = \frac{\sum_{i=1}^{n} x_i}{n} = \frac{\sum_{i=1}^{12} x_i}{12}$$
$$= \frac{13 + 15 + 16 + 20 + 20 + 22 + 27 + 29 + 30 + 33 + 34 + 42}{12} =$$
$$= \frac{301}{12} = 25.08$$

   Alternatively, using the equation with the absolute frequencies, we use not the 12 values but the 11 different values because the value 20 is repeated twice:

[50] In this case, we cannot use the data of the planets of the solar system because there are very few.

$$\overline{x}_a = \frac{\sum_{i=1}^{11} f_i x_i}{F_n} = \begin{cases} \dfrac{1.13+15+16+2.20+22+27+29+30+33+34+42}{\mathrm{F_n}} \\ \mathrm{F_n} = \sum_{i=1}^{11} f_i = 1+1+1+2+1+1+1+1+1+1+1 = 12 \end{cases}$$

$$\overline{x}_a = \frac{301}{12} = 25.08$$

Once we have calculated the mean without groups, now we are going to calculate it using groups of ten. For this, we use the frequency of the classes calculated in the previous exercise. We have four classes, the first between 10 and 20 with a mark of the class in 15, and the rest in the following three tens, with marks 25, 35, and 45. For each one of those classes, we have the following frequencies: 3, 5, 3, 1. If we use the equation for the calculation of the arithmetic mean with classes and their frequencies, we have:

$$\overline{x}_a = \frac{\sum_{i=1}^{4} f_i x_i}{F_n} = \begin{cases} \dfrac{3 \cdot 15 + 5 \cdot 25 + 3 \cdot 35 + 1 \cdot 45}{\mathrm{F_n}} \\ \mathrm{F_n} = \sum_{i=1}^{4} f_i = 3+5+3+1 = 12 \end{cases}$$

$$\overline{x}_a = \frac{320}{12} = 26.67$$

If we compare the arithmetic means obtained with all the data and with data grouped in classes, we can see that they are different, the first is 25.08 and the second 26.67, which is logical because the procedures and the equations used to calculate both of them are different, the first is correct because it uses all the information and the second is only an approximation, but it is not very far from the correct value and reduces the cost of the calculus, or computation, because it reduces the number of operations, for that reason it is interesting to use it.

3. For the smallest satellites of Uranus, calculate their variance.

We remember that the equation to calculate the variance is:

$$s^2 = \frac{\sum_{i=1}^{n} (x_i - \overline{x})^2}{n} = \frac{\sum_{j=1}^{m} f_j (x_j - \overline{x})^2}{\sum_{j=1}^{m} f_j}$$

The first does not use the absolute frequencies, and the second uses them. We apply the first to solve the problem:

$$s^2 = \frac{(13-25.08)^2 + \ldots + 2(20-25.08)^2 + (42-25.08)^2}{12} = 71.91$$

4. For the smallest satellites of Uranus, calculate their standard or typical deviation for data without grouping and grouped by tens.

We remember that the equation to calculate the standard deviation is:

$$s = \sqrt{\frac{\sum_{i=1}^{n}(x_i - \overline{x})^2}{n}} = \sqrt{\frac{\sum_{j=1}^{m} f_j (x_j - \overline{x})^2}{\sum_{j=1}^{m} f_j}}$$

We use the first to calculate the standard deviation with the data without grouping:

$$s = \sqrt{\frac{(13-25.08)^2 + \ldots + (20-25.08)^2 + \ldots + (42-25.08)^2}{12}} = 8.48$$

And the second, with frequencies for the data grouped by tens:

$$s = = \sqrt{\frac{3.(15-26.67)^2 + 5(25-26.67)^2 + 3(35-26.67)^2 + (45-26.67)^2}{12}}$$
$$= 7.07$$

5. Analyse the representativeness of the mean for the following set of data of the rotation period of the Solar System planets: {Mercury, 58; Venus, 0.4; Earth, 1; Mars, 1; Jupiter, 0.4; Saturn, 0.4; Uranus, 0.7; Neptune, 0.7}.

In the previous exercises, the mean, the variance, and the standard deviation have been calculated, but the last ones have not been used to analyse how good the mean is as representative of the data set. To better understand the concepts of arithmetic mean and standard deviation and the use of the standard deviation to know how good the mean is as representative of the data and if it is valid to use it for that, we calculate both measurements first for all the data in the set and then for all the data except Mercury. For all the data applying the equations, we have:

$$\overline{x}_a = \frac{\sum_{i=1}^{n} x_i}{n} = \frac{\sum_{i=1}^{8} x_i}{8} = \frac{58 + 0.4 + 1 + 1 + 0.4 + 0.4 + 0.7 + 0.7}{8} = \frac{62.6}{8}$$

$$\overline{x}_a = 7.83$$

If we would use frequencies, the equation would be:

$$\overline{x}_a = \frac{\sum_{j=1}^{m} f_i x_i}{F_m} = \frac{\sum_{j=1}^{m} f_j x_j}{F_m} = \begin{cases} \dfrac{1.58 + 3.0, 4 + 2.1 + 2.07}{\mathrm{F_n}} \\ \mathrm{F_n} = \sum_{j=1}^{m} f_j = \sum_{j=1}^{4} f_j = 1 + 3 + 2 + 2 = 8 \end{cases}$$

$$\overline{x}_a = \frac{62.6}{8} = 7.83$$

Both results are the same, but the second has less calculus than the first, and it is important to note that i goes up to 8 because it takes into account all eight data points, but j goes up to 4 because it takes into account only the different four data points. For that reason, we use n for the limit of the first and m for the limit of the second.

Now, let us calculate the standard deviation:

$$s = \sqrt{\frac{\sum_{i=1}^{n} (x_i - \overline{x})^2}{n}} =$$

$$\sqrt{\frac{(58 - 7.83)^2 + (0.4 - 7.83)^2 + (0.4 - 7.83)^2 + (0.4 - 7.83)^2 + (1 - 7.83)^2 + (1 - 7.83)^2 + (0.7 - 7.83)^2 + (0.7 - 7.83)^2}{8}}$$

$$= \sqrt{\frac{2877.61}{8}} = 18.97$$

Let us now calculate it for all data except Mercury. Applying the equations, we have for the mean:

$$\overline{x}_a = \frac{\sum_{i=1}^{7} x_i}{7} = \frac{0.4 + 1 + 1 + 0.4 + 0.4 + 0.7 + 0.7}{7} = \frac{4.6}{7} = 0.66$$

For the standard deviation:

$$s = \sqrt{\frac{\sum_{i=1}^{n} (x_i - \overline{x})^2}{n}} =$$

$$\sqrt{\frac{\begin{array}{c}(0.4-0.66)^2+(0.4-0.66)^2+(0.4-0.66)^2+(1-0.66)^2\\ +(1-0.66)^2+(0.7-0.66)^2+(0.7-0.66)^2\end{array}}{8}}$$

$$=\sqrt{\frac{0.51}{7}}=0.07$$

As seen, in both cases, values for the mean and the standard deviation have been obtained because when applying the equations, results will always be obtained, but the meaning, that is, the representativeness of both results, is very different. If the values are observed carefully, it can be seen that the Mercury data are clearly very different from the rest of the values since all the others have a value equal to or less than 1, with a minimum of 0.4, and Mercury has a value of 58, that is, it is 58 times greater than the largest of the rest of the values and 145 times greater than the smallest of the rest of the values. Therefore, when the average is calculated, the weight of mercury is very large and causes an average of almost 8 days to be obtained, which is very different from all values, including mercury, and therefore is a meaningless value that does not represent the whole. To corroborate this conclusion, when the standard deviation is calculated, it is seen that it is a very large value, almost three times higher than the mean.

In the second calculation, when the Mercury value is eliminated, the mean obtained is 0.56, which is quite representative of the values of that subset, since there are two values that are 1, two that are 0.7, and three that are 0.4, so in this case the mean makes sense and all the values of the set could be substituted for the mean, since only by observing the mean, without seeing any other value, we would know that the rotation periods of the planets of the solar system are lower than those of the Earth and are approximately 60% of that period. In addition, when the standard deviation is calculated, a value that is eight times lower than the mean is obtained, so it can be seen that said mean is a very good representative of the data, since they differ very little around it.

6. For the smallest satellites of Uranus, order them and calculate their range.

   We remember that the data of the satellites of Uranus are Cordelia, 13; Ophelia, 16; Bianca, 22; Crésida, 33; Desdemona, 29; Juliet, 42; Portia, 55; Rosalinda, 27; Belinda, 34; Luna-1986 U10, 20; Puck, 77; Miranda, 235; Ariel, 578; Umbriel, 584; Titania, 788; Oberon, 761; Calíbano, 30; Luna-1999 U1, 20; Sycorax, 60; and Luna-1999 U2, 15. For these data, the statement of problem only asks us for the smallest ones, and we know that those ones are those with a radius less than 50 km, that is, the following twelve: Cordelia, 13; Ophelia, 16; Bianca, 22; Crésida, 33; Desdemona, 29; Juliet, 42; Rosalinda, 27; Belinda, 34; Luna-1986 U10, 20; Calíbano, 30; Luna-1999 U1, 20; Luna-1999 U2, 15.

   And we now must order them using the value of their radius, from the lowest to the highest that is: Cordelia, 13; Luna-1999 U2, 15; Ophelia, 16; Luna-1999 U1, 20; Luna-1986 U10, 20; Bianca, 22; Rosalinda, 27; Desdemona, 29; Calíbano, 30; Crésida, 33; Belinda, 34; Juliet, 42.

Once we have ordered them, we can calculate their range as the rest of the highest values less the lowest ones are Juliet, 42 and Cordelia, 13, and the range is

$$range = 42 - 13 = 29$$

7. For the smallest satellites of Uranus, calculate their median.

To calculate the median of the smallest satellites of Uranus, we must start from their ordination, done in the previous exercise, that is, Cordelia, 13; Luna-1999 U2, 15; Ophelia, 16; Luna-1999 U1, 20; Luna-1986 U10, 20; Bianca, 22; Rosalinda, 27; Desdemona, 29; Calíbano, 30; Crésida, 33; Belinda, 34; Juliet, 42, and apply the equations of calculus of the median, that are:

$$\tilde{x} = \frac{x_{n/2} + x_{(n/2)+1}}{2} \text{ if } n \text{ is even}$$

and

$$\tilde{x} = x_{(n+1)/2} \text{ if } n \text{ is odd}$$

As we have 12 satellites, $n = 12$, *is even* and the equation to be applied is the first one, the result is:

$$\tilde{x} = \frac{x_{n/2} + x_{(n/2)+1}}{2} = \frac{x_6 + x_7}{2} = \frac{22 + 27}{2} = 24.5$$

x_6 is 22 because the ordered set is (13, 15, 16, 20, 20, 22, 27, 29, 30, 33, 34, 42), and the sixth value is 22.

8. For the smallest satellites of Uranus, calculate their quartiles.

To solve this problem, we start again from the ordered set of values of the Uranus satellites (13, 15, 16, 20, 20, 22, 27, 29, 30, 33, 34, 42) and apply the equation to calculate the quartiles, which are:

$$\text{if } nc \notin \mathbb{N} : \tilde{x}_c = x_{[nc]+1} \; [nc] \text{ integer part of } nc$$

$$\text{if } nc \in \mathbb{N} : \tilde{x}_c = \frac{x_{nc} + x_{nc+1}}{2}$$

c means quartile, and it will take the values 1/4, 2/4, and 3/4; and n is the total number of data.

We start calculating the first quartile, in this case $n = 12$ and $c = 1/4$, which means that:

$$nc = 12.\frac{1}{4} = 3 \in \mathbb{N}$$

and we must apply the second equation

$$\tilde{x}_c = \frac{x_{nc} + x_{nc+1}}{2} \rightarrow \tilde{x}_{\frac{1}{4}} = \frac{x_3 + x_4}{2} = \frac{16 + 20}{2} = 18$$

We are not going to calculate the second quartile because it is the same as the median, and we have calculated it in the previous exercise.

For the third:

$$nc = 12.\frac{3}{4} = 9 \in \mathbb{N}$$

and we must apply the second equation

$$\tilde{x}_c = \frac{x_{nc} + x_{nc+1}}{2} \rightarrow \tilde{x}_{\frac{3}{4}} = \frac{x_9 + x_{10}}{2} = \frac{30 + 33}{2} = 31.5$$

9. For the smallest satellites of Uranus, calculate the percentile 54.

   We start again from the ordered set of data (13, 15, 16, 20, 20, 22, 27, 29, 30, 33, 34, 42) and apply the equation to calculate the percentile. As with the quartiles, we have two:

   if $np \notin \mathbb{N} : \tilde{x}_p = x_{[np]+1}$ $[np]$ integer part of np:

$$\text{if } np \in \mathbb{N} : \tilde{x}_c = \frac{x_{np} + x_{np+1}}{2}$$

For that reason, the first calculation that we must perform is np to know which equation applies

$$np = 12.\frac{54}{100} = 6.48 \notin \mathbb{N}$$

Consequently, the equation is:

$$\tilde{x}_p = x_{[np]+1} \rightarrow \tilde{x}_{54/100} = x_{[6.48]+1} = x_{6+1} = x_7 = 27$$

Exercises Solved in R

In this section, previous exercises 6–19 will be solved using the R software.

Once we have presented the R environment and the RGUI, we are ready to start to solve using R the cases that have been theoretically solved in this lesson about Data and its description, applying all the concepts introduced in the lesson, that is, for each of the files, we are going to calculate all the magnitudes seen in the subject. We are going to obtain the frequencies, the mean, which will be the arithmetic; the

measures of dispersion, standard deviation and variance, and the measures of ordering, median and quartiles, including percentile 54. To solve this exercise, we will use a file of data that will be of type .txt, that is, plain text, and it will be made up of the data from the minor satellites of Uranus that we have used in the theoretical description of the subject, which we will call satellites.txt.[51] Note that exercises 1–5 cannot be solved in R, so we begin with exercise 6.

10. The rotation period data, in days, of the planets of the solar system are Mercury, 58; Venus, 0.4; Earth, 1; Mars, 1; Jupiter, 0.4; Saturn, 0.4; Uranus, 0.7; and Neptune, 0.7. These new data give the absolute frequency.

Remember that the first step to work in R is to create the file that contains the data of the rotation periods and save it with a name, for example, periods.txt:

	Planet	Period
1	Mercury	58
2	Venus	0.4
3	Earth	1
4	Mars	1
5	Jupiter	0.4
6	Saturn	0.4
7	Uranus	0.7
8	Neptune	0.7

Once we have saved the above data in a file, the next step is to load the file with the command:

$$\mathrm{p} < - \text{read.table}\left(\text{"periods.txt"}\right)$$

To obtain the absolute frequency of the variable Period, which we call freqabsperiod, we use the table () function, and as an argument, we introduce the variable for which we want to obtain its absolute frequency, which is Period. The full instructions are as follows:

$$> \text{freqabsperiod} < - \text{table}\left(\mathrm{p}\ \$\ \text{Period}\right)$$

To see the result, we enter

$$> \text{freqabsperiod}$$

[51] (Satellite name, radius in km): Cordelia, 13; Ophelia, 16; Bianca, 22; Crésida, 33; Desdemona, 29; Juliet, 42; Rosalinda, 27; Belinda, 34; Luna-1986 U10, 20; Calíbano, 30; Luna-999 U1, 20; Moon 1999 U2, 15.

0.4	0.7	1	58
3	2	2	1

That result is the same as the result that we obtained in the handmade exercise: the value 0.4 is repeated 3 times, on Venus, Jupiter, and Saturn; data 0.7 appears 2 times, on Uranus and Neptune; 1 appears 2 times, on Earth and Mars; and 58 appears 1 time, on Mercury. Therefore, the frequencies for each one of the four possible values (0.4, 0.7, 1, 58) are $f_1 = 3; f_2 = 2; f_3 = 2; f_4 = 1$.

11. Calculate the relative frequency of the previous data.

We remember that there is no function in R to calculate the relative frequency, so we generate one that we call freqrelperiod. To define a function, we will again use the function function () to which we will give as an argument the input variable that it will have, which will be x, and then, between braces, we will put the definition of the function. The complete instruction will be

$$\text{freqrel} < - \text{function (x) } \{\text{table (x)}/\text{length (x)}\}$$

As we know the freqrel function would only work in this execution of the program, if we wanted to save it to use it in other executions, we would use the dump () function that would save it in the working directory, and the complete instruction would be:

$$\text{dump } \left(^{''}\text{freqrel}^{''}, \text{file} = {}^{''}\text{freqrel.R}^{''}\right)$$

As we know, to load the function in other executions of the program, we would use the source () function, and the complete instruction would be:

$$\text{source } \left(^{''}\text{freqrel.R}^{''}\right)$$

Next, for the variable x, we assign the value of the rotation period of the planets, and the complete instruction is:

$$x = \text{p \$ Period}$$

Finally, we calculate the value of the relative frequency function to which we will assign the name freqrelperiod, and the complete instruction is:

$$\text{freqrelperiod} < - \text{freqrel } (x)$$

To see the result, we enter

$$\text{Freqrelperiod}$$

0.4	0.7	1	58
0.375	0.250	0.250	0.125

That result is the same as the result that we obtained in the handmade exercise. It can be easily verified that the sum of the relative frequencies is equal to 1.

12. Calculate the accumulated absolute and relative frequency of the previous data.

To calculate the accumulated relative frequency, which we call freqrelacumperiod, we do the same as in the case of the absolute frequency, and we use the cumsum ( ) function. In this case, we introduce the relative frequency of the rotation period as an argument. The complete instruction is:

$$\text{freqrelacumperiod} < - \text{cumsum (freqrelperiod)}$$

To see the result, we enter

$$\text{freqrelacumperiod}$$

0.4	0.7	1	58
0.375	0.625	0.875	1.000

That result is the same as the result that we obtained in the handmade exercise. Remember that 1 is the amount that we ever must obtain when we calculate the accumulated relative frequency of the highest value.

13. Give the frequency distribution of the previously calculated frequencies.
The frequency distribution is directly obtained from the above exercises:

$$\{(0.4, 0.375), (0.7, 0.625), (1, 0.875), (58, 1)\}.$$

14. Perform data grouping and give the equivalence classes using data from the satellites of Uranus.[52] Calculate the absolute frequency of each class. The characteristic that we are going to analyse is the radius in kilometres of the satellites. They are Cordelia, 13; Ophelia, 16; Bianca, 22; Crésida, 33; Desdemona, 29; Juliet, 42; Portia, 55; Rosalinda, 27; Belinda, 34; Luna-1986 U10, 20; Puck, 77; Miranda, 235; Ariel, 578; Umbriel, 584; Titania, 788; Oberon, 761; Calíbano, 30; Luna-1999 U1, 20; Sycorax, 60; Luna-1999 U2, 15.

Remember that the first step to work in R is to create the file that contains the data of the radius of the satellites and save it with a name, for example, satellites.txt:

[52] In this case, we cannot use the data of the planets of the solar system because there are very few.

	Name	Radius
1	Cordelia	13
2	Ofelia	16
3	Bianca	22
4	Cresida	33
5	Desdemona	29
6	Julieta	42
7	Rosalinda	27
8	Belinda	34
9	Luna-1986 U10	20
10	Calibano	30
11	Luna-999 U1	20
12	Luna-1999 U2	15

Once we have saved the above data in a file, the next step is to load the file with the command:

$$\mathrm{s} < - \mathrm{read.table}\left(^{''}\mathrm{satellites.txt}^{''}\right)$$

We group the data into four intervals as follows:

$$[10, 20)$$

$$[20, 30)$$

$$[30, 40)$$

$$[40, 50)$$

To do that, the first thing that we have to do is to define a vector L that will contain the extremes of the intervals:

$$\mathrm{L} = 10 + 10^{*}(0 : 4)$$

Next, we use the function cut to generate the intervals:

$$\mathrm{s\_int} = \mathrm{cut}(\mathrm{s\ \$\ Radius}, \mathrm{breaks} = \mathrm{L}, \mathrm{right} = \mathrm{FALSE})$$

Finally, to obtain the absolute frequency of each class, we enter:

$$\mathrm{table}(\mathrm{s\_int})$$

[10,20)	[20,30)	[30,40)	[40,50)
3	5	3	1

15. Calculate the mode of the data of the radius in km of the satellites of Uranus.[53] Remember that the data are Cordelia, 13; Ophelia, 16; Bianca, 22; Crésida, 33; Desdemona, 29; Juliet, 42; Portia, 55; Rosalinda, 27; Belinda, 34; Luna-1986 U10, 20; Puck, 77; Miranda, 235; Ariel, 578; Umbriel, 584; Titania, 788; Oberon, 761; Calíbano, 30; Luna-1999 U1, 20; Sycorax, 60; Luna-1999 U2, 15.

    Since the mode is the most frequent value in the data, that is, the value with the highest absolute frequency, we calculate the absolute frequencies of the variable Radius. The instruction is:

$$\text{table}(\text{s \$ Radius})$$

13	15	16	20	22	27	29	30	33	34	42
1	1	1	2	1	1	1	1	1	1	1

As we can see, each satellite has an absolute frequency of 1, which means that all of them appear only one time, with only one exception, the value 20, for which there are two observations, Luna-1986 U10 and Luna-1999 U1, which means that the mode for the set of data of the satellites of Uranus is 2.

16. For the smallest satellites of Uranus, with radius lower than 50 km, calculate the arithmetic mean for data without grouping.

To calculate the mean, which we call mr, we use the mean ( ) function, and as an argument, we introduce the variable that in this case, when found within a matrix, we know that it is s $ Radius, the complete instruction is:

$$\text{mr} < - \text{mean}\ (\text{s \$ Radius})$$

To see the result, we enter:

$$\text{mr}$$

17. 25.08333, which is the same as the result that we obtained in the handmade exercise.

18. For the smallest satellites of Uranus, calculate their variance.

The variance is calculated with the function var (), so the instruction will be:

$$\text{varr} < - \text{var}\ (\text{s \$ Radius})$$

To see the result, we enter:

[53] In this case, we cannot use the data of the planets of the solar system because there are very few.

varr

78.44697

In this case, the result is not the same as the result obtained in the handmade exercise, which was 71.91, because the formula used in the R software divides by 11 (n–1) instead of dividing by 12 (n). Therefore, to obtain the same results, you have to perform the operation:

$$\text{varr} = \text{varr}^{*}\ 11/12$$

We check that we already obtain the same as in the handmade exercise by introducing varr again.

19. For the smallest satellites of Uranus, calculate their standard or typical deviation.

    The standard deviation is calculated with the function sd (), so the instruction will be:

$$\text{sdr} < - \text{sd}\ (\text{s \$ Radius})$$

    To see the result, we enter:

sdr

8.857029

In this case, the result is not the same as the result obtained in the handmade exercise, which was 8.48, because the formula used in the R software divides by 11 (n–1) instead of dividing by 12 (n). Therefore, to obtain the same results, you have to perform the operation

$$\text{sdr} = \text{sqrt}\ \left(\left(\text{sdr}^2\right)^{*}\ 11/12\right)$$

We check that we already obtain the same as in the handmade exercise by introducing sdr again.

20. Analyse the representativeness of the mean for the following set of data of the rotation period of the Solar System planets: {Mercury, 58; Venus, 0.4; Earth, 1; Mars, 1; Jupiter, 0.4; Saturn, 0.4; Uranus, 0.7; Neptune, 0.7}.

    To analyse the representativeness of the mean, the first step is to calculate the mean with the following instruction:

$$\text{mp} < - \text{mean}\ (\text{p \$ Period})$$

To see the result, we enter:

```
mp
```

```
7.825
```

The next step is to obtain the standard deviation as follows:

```
sdp <- sd (p $ Period)
```

```
sdp
```

```
20.2753
```

As we saw in the handmade exercise, if the values are observed carefully, it can be seen that the Mercury data are clearly very different from the rest of the values since all the others have a value equal to or less than 1, with a minimum of 0.4, and Mercury has a value of 58, that is, it is 58 times greater than the largest of the rest of the values and 145 times greater than the smallest of the rest of the values. Therefore, when the average is calculated, the weight of mercury is very large and causes an average of almost 8 days to be obtained, which is very different from all values, including mercury, and therefore is a meaningless value that does not represent the whole. To corroborate this conclusion, when the standard deviation is calculated, it is seen that it is a very large value, almost three times higher than the mean.

21. For the smallest satellites of Uranus, order them and calculate their range.

The instruction order allows us to order the data indicated as an argument. In this case, we want to order the data from the lowest to the highest, so the second argument must be set to false, as follows:

```
os <- order (s $ Radius, decreasing = FALSE)
```

Since the output of that instruction is the index of each element in the vector, we have to enter the following instruction to obtain the ordered data:

```
(s $ Radius)[os]
```

```
13  15  16  20  20  22  27  29  30  33  34  42
```

Finally, to calculate the range, we use the command:

```
max(s $ Radius) - min(s $ Radius)
```

```
29
```

That result is the same as the result that we obtained in the handmade exercise.

22. For the smallest satellites of Uranus, calculate their median.

To obtain the median of the radius, we must enter the following instruction:

$$\text{medianr} < - \text{median}\ (\text{s \$ Radius})$$

To see the result, we enter:

$$\text{medianr}$$

$$24.5$$

That result is the same as the result that we obtained in the handmade exercise.

23. For the smallest satellites of Uranus, calculate their quartiles.

To obtain the first quartile, we have to enter the following instruction:

$$\text{quar1r} < - \text{quantile}\ (\text{s \$ Radius}, 0.25)$$

To see the result, we enter:

$$\text{quar1r}$$

$$19$$

To obtain the second quartile, we have to enter the following instruction:

$$\text{quar2r} < - \text{quantile}\ (\text{s \$ Radius}, 0.5)$$

To see the result, we enter:

$$\text{quar2r}$$

$$24.5$$

We can see that the second quartile coincides with the median.

Finally, to obtain the third quartile we have to enter the following instruction:

$$\text{quar3r} < - \text{quantile}\ (\text{s \$ Radius}, 0.75)$$

To see the result, we enter

$$\text{quar3r}$$

30.75

We see that in this case, the results are not the same as in the handmade exercise because the equations used in the R software are different.

24. For the smallest satellites of Uranus, calculate the percentile 54.

To obtain the 54th percentile, we have to enter the following instruction:

$$\text{quar54r} < -\text{quantile}\,(\text{s \$ Radius}, 0.54)$$

To see the result, we enter

quar54r

26.7

We see that in this case, the results are not the same as in the handmade exercise because the equations used in the R software are different.

Annex. Data Extended Concepts

Frequency

Extended concepts about frequency

Absolute Frequency

Another, more mathematical formal definition of the absolute frequency is the following: Let be an m-tuple $\underline{b}$. For each element of $\underline{b}$, b_i, the absolute frequency of b_i is equal to the number of elements of its equivalence class, and each different value of $\underline{b}$ constitutes an equivalence class. The equivalence classes of the m-tuple $\underline{b}$ will be written as the n-tuple $\underline{a}$. Each of the elements of $\underline{a}$, a_i will correspond to the subset of elements of $\underline{b}$ belonging to the equivalence class whose representative is a_i. The cardinal of that subset corresponds to the absolute frequency of a_i. Therefore, a bijective correspondence can be established between the n-tuple $\underline{a}$ and the n-tuple $\underline{f}$ corresponding to the absolute frequencies of each of the elements of $\underline{a}$.

We will present a complete example of calculating the absolute frequency of an m-tuple of data $\underline{b}$. Suppose that $\underline{b} = \{2, 2, 3, 8, 9, 9, 9, 6, 7, 7, 7, 7\}$, with which $m = 12$ because we have 12 data points, that is, we have a 12-tuple. Starting from the tuple $\underline{b}$, we construct the tuple a as the equivalence classes of $\underline{b}$, that is, the different values of $\underline{b}$. Consequently, the tuple a is equal to $\underline{a} = \{2, 3, 8, 9, 6, 7\}$. Each element of a corresponds to a subset of elements of $\underline{b}$ belonging to the equivalence class whose representative is a_i. Consequently, for $a_1 = 2$, it corresponds to the

subset of 2 elements formed by b_1 and b_2, whose value is 2. For $a_2 = 3$ corresponds a subset of only element b_3 whose value is 3. For $a_3 = 8$ corresponds also a subset of only element b_4 whose value is 8. For $a_4 = 9$, corresponds to the subset of 3 elements formed b_5, b_6 and b_7 whose value is 9. For $a_5 = 6$ there also corresponds a subset of only element b_8 whose value is 6. And finally for $a_6 = 7$ there corresponds a subset of 4 elements formed by b_9, b_{10}, b_{11} and b_{12}. The cardinals of each subset are the absolute frequencies corresponding to each element of $\underline{a} = \{2, 3, 8, 9, 6, 7\}$, consequently the set of absolute frequencies is $\underline{f} = \{2, 1, 1, 3, 1, 4\}$. By using the values of the absolute frequencies, we went from having to work with 12 data to work with 6 data.

From the concept of absolute frequency, the concept of weight can be defined, as follows: Let $\underline{a}$ be an n-tuple, if, for its data analytics treatment, we want to give a different importance to each of the elements of $\underline{a}$, the concept of element weight is used, in such a way that a bijective correspondence is established between the n-tuple $\underline{a}$ and the n-tuple $\underline{w}$ corresponding to the weights of each of the elements of $\underline{a}$. The statistical treatment, as will be seen later, of the absolute frequency and the weight of the elements of $\underline{a}$ is analogous. For this reason, the mathematical expressions will refer only to the weights $\underline{w}$. To obtain them as a function of the absolute frequencies, we only have to change $\underline{w}$ by $\underline{f}$.

Let's see a complete example of defining the weight of an n-tuple of data a. Suppose that $\underline{a} = \{2, 3, 8, 9, 6, 7\}$, with which $n = 6$ because we have 6 data, that is, we have a 6-tuple. Now suppose that for our study the importance of all the values of $\underline{a}$ is not the same (for example in the case of test scores with different values) and that the first value is worth twice as much as the second and the third and the fifth, the fourth value is worth three times as much as the second, third, and fifth, and the sixth value is worth four times as much as the second, third, and fifth. This can be reflected by a bijective association of a new tuple of weights $\underline{w} = \{2, 3, 8, 9, 6, 7\}$ with the tuple $\underline{a}$.

Relative Frequency

Using again the concept of n-tuple as with the absolute frequency, the definition of relative frequency is the following one:

The relative frequency is the number of occurrences of a given data divided by the amount of data. From a more formal point of view, it is defined as: The n-tuple $\underline{f}$ corresponds to the absolute frequencies of each of the elements of a. If each of the elements of $\underline{f}$ is divided by the cardinal[54] of $\underline{f}$, that is, between m, the n-tuple $\underline{f_r}$ corresponding to the relative frequencies of each of the elements of $\underline{a}$ is their relative frequencies.

[54] See in the definition of n-tuple how the n-tuple $\underline{f}$ was defined.

The n-tuple $\underline{w}$ corresponds to the weights of each of the elements of $\underline{a}$. If each of the elements of $\underline{w}$ is divided by the cardinal of $\underline{w}$, we obtain the n-tuple $\underline{w_r}$ corresponding to the relative weights of each of the elements of $\underline{a}$.

The treatment of the relative frequency and relative weight of the elements of a is analogous. For this reason, the mathematical expressions will refer only to the relative weights $\underline{w_r}$. To obtain them as a function of the relative frequencies, we only have to change $\underline{w_r}$ by $\underline{f_r}$.

Cumulative Frequency

Using the concept of tuple all the previous definitions of frequency are complemented with the concept of frequency and its associated definitions. The accumulated frequency is: with the data ordered by magnitude, the sum of the absolute or relative frequencies of the data below the data plus that of the data. From a more formal point of view it is defined as: Given two n-tuples $\underline{a}$ and $\underline{f}$, the accumulated absolute frequency of $\underline{a}$ with frequency $\underline{f}$ up to the value a_k is:

$$F_k\left(\underline{a};\underline{f}\right) = \sum\nolimits_{i=1}^{k} f$$

Given two n-tuples $\underline{a}$ and $\underline{w}$ the cumulative weight of $\underline{a}$ a with weight $\underline{w}$ up to the value a_k is:

$$W_k(\underline{a};\underline{w}) = \sum\nolimits_{i=1}^{k} w$$

Given two n-tuples $\underline{a}$ and $\underline{f_r}$ the cumulative relative frequency of $\underline{a}$ a with a relative frequency $\underline{f_r}$ up to the value a_k is:

$$F_{r_k}\left(\underline{a};\underline{f_r}\right) = \sum\nolimits_{i=1}^{k} f_r$$

Frequency Distribution

From a more formal point of view, it is defined as: The pairs of tuples $\left(\underline{a};\underline{f}\right)$ and $\left(\underline{a};\underline{f_r}\right)$ are called frequency distributions. The first is the absolute frequency distribution and the second the relative frequency distribution

Mean

Extended concepts about Mean

Geometric Mean

Other[55] concepts related to the concept of mean that can be useful in the description of the set of data that is being analysed are introduced in this section. They are other definitions of the mean, needed in specific situations and other definitions of the data dispersion. For all the means for data grouped in equivalence classes, the concept of data is changed to that of class.

The Geometric[56] mean must be used in specific situations, for example, when the data in the set are related by a function in which each data in the ordered set is obtained from the previous one multiplied by an index, that are growing data. The calculation equation is:

$$\overline{x}_g = \left(\prod_{i=1}^{n} x_i \right)^{1/n}$$

And using absolute frequencies, of the j different values, the equation is:

$$\overline{x}_g = \left(\prod_{j=1}^{m} x_j^{f_j} \right)^{1/\sum_{j=1}^{m} f_j}$$

If dataset has only two data with different values, it is called as Proportional Mean, and its equation is:

$$\overline{x}_g = \left(\prod_{i=1}^{2} x_i \right)^{1/2}$$

If weights are used, the equation for the calculus of the geometric mean is:

[55] They have not been introduced in the previous section about the mean because they are significantly less used than the concepts present there and maintain a rhythm of the reading and study have prevalence. For this reason they are presented in this last section of the lesson.

[56] From a geometric point of view, the geometric mean of the sides of a rectangle gives the side of a square of equal area. The geometric mean of 3 numbers a, b, and c is the length of a face of a cube whose volume is the same as the cuboid whose sides are the length of the initial numbers.

$$\overline{x}_g = \left(\prod_{i=1}^{n} a_i^{w_i} \right)^{1/\sum_{i=1}^{m} w_i}$$

Moore established in 1965 that the number of transistors in an integrated circuit would double every year. In 1975, he amended his own law and stated that it would double every two years. Taking into account the following data on processors, creation dates, and number of transistors, reasonably indicate whether, in 1973, Moore had reason to think about modifying his law statement. (Processor name, creation date, number of transistors): 4004, 1971, 2300; 8008, 1972, 3500; 8080, 1973, 4500; 8086, 1978, 29,000; 286, 1982, 134,000; 386, 1985, 275,000

The solution is: In 1971, the 4004 processor had 2300 transistors. In 1972, the 8008 processor had 3500. Consequently, the growth rate during 1971 was: $3500 = 2300.x1 \rightarrow x1 = 1.52$. In 1973 the 8080 processor had 4500 transistors. Consequently, the growth rate during 1972 was: $4500 = 3500.x2 \rightarrow x2 = 1.29$. The mean to apply is geometric because the indices multiply the previous value:

$$\overline{x}_g = \left(\prod_{i=1}^{2} x_i^{f_i} \right)^{1/F_2} = \begin{cases} \sqrt[2]{1,52^1.1,29^1} \\ F_2 = \sum_{i=1}^{2} f_i = 1 + 1 = 2 \end{cases}$$

$$\overline{x}_a = \sqrt{1,96} = 1,4$$

$$1,4 < 2/1,4.1,4 = 1,96 \sim 2$$

He had reasons to change his law

Harmonic Mean

The harmonic mean[57] is used in same specific problems, for example in cinematics. The calculation equation is:

$$\overline{x}_h = \frac{n}{\sum_{i=1}^{n} \left(\frac{1}{x_i} \right)}$$

And using absolute frequencies, of the *j* different values, the equation is:

[57] The first documents on the use of the Harmonic mean belong to the Egyptian civilization. They described the decomposition of a fraction into an equivalent sum of unit fractions, the original fraction being the harmonic mean of the denominators of the unit fractions.

$$\overline{x}_h = \frac{\sum_{i=1}^{n} f_i}{\sum_{i=1}^{n} \left(\frac{f_i}{x_i}\right)}$$

If weights are used, the equation for the calculus of the geometric mean is:

$$\overline{x}_h = \frac{W_n}{\sum_{i=1}^{n} \left(\frac{w_i}{a_i}\right)}$$

The first flight of the Boeing 747 took place between New York and London on January 21, 1970, at a speed of 895 km/h. If one-tenth of Moore's law of 1975 had been applied to aviation and the speed of airplanes had doubled every 20 years since then, what would be the average speed of a round-trip flight between New York and London, if the first leg were made at the speed of 1970 and the return at the speed of 2010?

Solution: In 1970, the speed was 895 km/h. The speed doubles every 20 years. Consequently, from 1970 to 2010 it would have doubled twice: $895.2.2 = 3580$ km / h

The average to be applied at speeds is the harmonic.

$$\overline{x}_{ar} = \frac{F_2}{\sum_{i=1}^{2} \left(\frac{f_i}{x_i}\right)} = \begin{cases} \dfrac{2}{\dfrac{1}{895} + \dfrac{1}{3580}} & \overline{x}_a = 1432 \text{ km/h} \\ F_2 = \sum_{i=1}^{2} f_i = 1 + 1 = 2 \end{cases}$$

Potential Mean

Once the Arithmetic, Geometric, and Harmonic means are known is the moment to introduce the Potential Mean, because is their natural extension and include inside all of them, as we are going to see from its equation:

$$\overline{x}_p = \begin{cases} \left(\dfrac{\sum_{i=1}^{n} x_i^r}{n}\right)^{1/r}, & \text{si } r \in \mathbb{R}^* \\ \left(\prod_{i=1}^{n} x_i\right)^{1/n}, & \text{si } r = 0 \end{cases}$$

And using absolute frequencies, of the *j* different values, the equation is:

$$\overline{x}_p = \begin{cases} \left(\dfrac{\sum_{i=1}^{n} x_i^r}{n} \right)^{1/r}, & \text{si } r \in \\ \left(\prod_{j=1}^{m} x_j^{f_j} \right)^{1/\sum_{j=1}^{m} f_j}, & \text{si } r = 0 \end{cases}$$

We can see that depending on the values of R we have:

- If $r = 1$, we have the Arithmetic mean
- If $r = 1$, we have the Geometric mean
- If $r = -1$, we have the Harmonic mean
- And there is another particular case that is if $r = 2$ that is called Quadratic Mean and its equation is:

$$\overline{x}_c = \left(\frac{\sum_{i=1}^{n} x_i^2}{n} \right)^{1/2}, \quad \text{si } r = 2$$

Mean Deviation

As we saw, in the standard deviation we elevated to the square the differences between each value and the mean to the square to avoid that those values which the mean was higher than the data compensate those ones which the data were lower than the mean, and obtain a low value for the standard deviation being all the values very different from the mean and, in consequence, to avoid to say that taking into account the low value of the standard deviation to say that the mean is a good representative of the data when it is not.

Another possibility to avoid using the square of the difference and in consequence do not need to use the squared root is to use the absolute value of the differences between the mean and each value. If we do that, we have a new definition for the deviation, the mean deviation, and a new equation for its calculus, that is:

$$s = \frac{\sum_{i=1}^{n} |x_i - \overline{x}|}{n} = \frac{\sum_{j=1}^{m} f_j |x_j - \overline{x}|}{\sum_{j=1}^{m} f_j}$$

The mean deviation is less used than the standard deviation because reasons related to its use for inference problems, where the standard deviation is more suitable.

Probability

In this third chapter, we are going to see the essential aspects related to the concept of *Probability*. As in the previous chapters and the coming ones, it is structured in three sections.

Section A introduces, in a theoretical and, at the same time, practical way, all the basic theoretical knowledge related to the concept of Probability that a Data Analyst should know in depth, from its definition to the related concepts that will be applied in the analysis of the data set under study.

Section B presents the computer-based solving of the same examples used in section A to introduce the theoretical knowledge.

Section C will consist of a set of statements of exercises about Probability for which detailed solutions can also be found in this section of the chapter.[1]

In this chapter, as in the previous chapter, the reader can find an Annex with an extended version of some of the concepts treated in the chapter, such as the Kolmogorov or Axiomatic Probability.

A. Theory

This first section of the chapter is structured in six subsections: 1. Introduction, 2. Event, 3. Set Theory Axioms and Operations, 4. Laplace or Classic Probability, 5. Bayesian Probability, 6. Probability Distribution of Random Variables. The basic knowledge related to the concept of data and the initial description of the available data are presented in detail.

[1] As was said in the first two chapters, but it is very important in order to obtain the best results for the learning process throughout the use of the book, that the reader tries to solve the exercises by himself before seeing their solutions, and that only once solved check if the obtained solutions are correct.

J. J. Cuadrado-Gallego, Y. Demchenko, *Data Analytics*,
https://doi.org/10.1007/978-3-031-39129-3_3

Introduction

Mastering the concept of probability and all the knowledge associated with it is fundamental in the study of Data, since it allows us to study and analyse its concepts not only to the observed data, or certain data but also to carry out studies or inferences about the data not observed, or probable, that is to say, make an inference. That is, the conclusions obtained for a sample can be extended to a population. In addition, the knowledge of probability allows a better understanding of many Data Science concepts, such as the support and confidence parameters in association studies. Consequently, understanding probability is essential when studying data science.

In the previous chapter, it was studied that one or more characteristics define an object and that the data are the values that this or those characteristics have when observations of the same obtained in an experiment are made. However, as has already been advanced in the previous chapter, both concepts, characteristics, and data, can be expanded when they belong to a sample that does not contain the whole population. It introduces the concept of probability into the study, since from the concept of probability, the values of the data of the characteristics with which he works may not only be the values obtained in the observations, and therefore totally true, but they may also be unobserved or probable values, and therefore with a percentage of uncertainty. This will lead to the characteristics, although they continue to have exactly the same definition as in the previous chapter, to be called random variables.

Considering what was said in the previous paragraph, it is easy to deduce that the introduction of the concept of probability and all the knowledge associated with it is very important in the study of statistics since it allows us to apply its concepts not only to the observed data or truth but also to carry out studies or inferences on the unobserved or probable data. In other words, the conclusions obtained for a sample can be extended to a population. This extension is known as performing statistical inference. In the following subsections, the concept will be deepened, and all the basic knowledge about probability that is essential to know to apply it in the study of data science will be introduced.

In this chapter, the fundamentals of probability theory will be studied. Before studying the concept of the random variable and its use in data science through probability functions, which will be seen in the next chapter, it is essential to understand in depth the concepts of event and probability and all the mathematics associated with them. This chapter begins by introducing the concept of a random event, which is very important to understand well before approaching the concept of probability. Next, we will study classical or Laplace probability, in which we will study the definition of probability and the associated concepts and the properties of probability. The next subsection presents the conditional probability from which the Bayesian probability is introduced. For those readers interested in a more rigorous mathematical approach to the concept of probability, in the annex under the title of Kolmogorov's axiomatic probability, advanced concepts of probability will be

presented. In this subsection, the concepts related to the basic probability will be deepened, with a greater mathematical component, and new ones will be introduced. In addition, in the annex, the aspects of combinatorial mathematics necessary to solve the counting aspects implicit in the study of probability problems are introduced. In this chapter, we use games of chance and especially the throwing of dice as the domain of the examples.

Event

In this first part of the chapter, the event concept and the related concepts are introduced, which are necessary/important to understand the concept of probability.

We begin this topic by studying the concept of *random events* and the concepts associated with them. However, before introducing the concept of a random event, we have to remember the definition of a previous concept that was introduced in the previous chapter, namely, the concept of a *random experiment*, which we remember can be defined as conducting an experiment is, according to the dictionary, "test and examine practically the virtue and properties of something", and random, is "that it depends on chance"; thus, we can define a random experiment as a test or practical test of something that depends on chance.

From the previous definition of the experiment, we can understand the definition of a *random event.* An *event* can be defined as follows: "In a random experiment, a subset of the total possible results"; and an *event* can also be defined as "a set of results of interest".

From the definition of a random experiment, it can be concluded that the result to be produced cannot be known a priori, but what can be known a priori is the set of all elementary results that can be produced. Taking this into account, we can define an *elemental event* as each of the simplest results that can occur in conducting a random experiment. An elemental event is each of the simplest results that can occur in conducting a random experiment. Another interesting concept associated with the *elemental event* is provided, which is that of the Basic Cell and which is defined as "a term proposed by Mahalanobis to name the smallest area for which a random variable can be considered to have a sufficiently precise meaning".

To see an example of the concepts seen of event and elementary event as was said in the introduction to the chapter, the experiment of rolling a die will be used: In the experiment of rolling a die, an example of an event is to obtain an even number, the event will be made up of the numbers {2,4,6}. In the same experiment of rolling a die, an elemental event is rolling a 2.

Once the concept of the elementary event has been defined, the mathematics of set theory can be applied to the study of events associated with a random experiment. From here on, we introduce the definitions associated with the concept of the event using the usual structure and order of definition of the concepts of Set Theory. The first consequence of this application is the definition of a *set* as a group of objects that fulfil the property of *membership*. The membership property is the basic property of sets and establishes that an element belongs to a set if it either appears in a list that

specifies all the elements that belong to the set or meets the requirements expressed in one or more sentences. Membership is denoted as $x \in A$ when element x belongs to set A and $x \notin A$ when element x does not belong to set A. Sets are denoted with capital letters A, B, C,. . . .

Applied to a random experiment, the property of belonged allows defining the set *sample space*, E or Ω as the one whose elements are the elementary events of the random experiment. The sample space is denoted as E and each of the elementary events as e_i, in such a way that the sample space is $E = \{e_1 \ldots e_n\}$.

The example of the sample space in the experiment of rolling a die is the set {1, 2, 3, 4, 5, 6} of the possible elemental outcomes in the rolling.

The definition of the sample space allows defining the concepts of certain event, complementary event, and impossible event, whose definitions are:

- *Sure event* is the one that is always verified, that is, it is that event such that whenever the experiment is carried out, some of the elementary events that compose it are obtained. From its definition, it follows that the safe event is E.

An example of a sure event in the experiment of rolling a die is the set {1, 2, 3, 4, 5, 6} of all the possible elemental outcomes in the rolling. The roll of a die is {1, 2, 3, 4, 5, 6}, which is the sample space. In rolling a die, many possible outcomes can occur, such as: A: obtain a 2, the subset of E is {2}, which is itself an elementary event; B: obtain an even number, the subset of E is {2, 4, 6}, in this case, it is not an elementary event; C: get a number other than 1 or 2, the subset of E is {3,4,5,6}, in this case, it is not an elementary event either. In the three events of the example, it can be verified that the elementary events that form them coincide with one or more elements of the sample space. Following the previous example, A has 2, B has 2, 4, and 6, and C has 3, 4, 5, and 6. If this is extended to any possible result, it can be concluded that the random experiment cannot be carried out to roll a dice without obtaining some element of the sample space, so the sample space constitutes the sure event.

- *Complementary event* of a given event A is verified whenever A is not verified. The complementary event is denoted as $\overline{A}$.
  Following the previous example, the complementary of the event $A = \{2\}$ is $\overline{A} = \{1, 3, 4, 5, 6\}$, since whenever A is not verified, some of the $\overline{A}$ elements are verified.

- *Impossible event* is one that is never verified, that is, it is that event such that whenever the experiment is carried out, none of the elementary events that compose it are ever obtained. In the same way that to define a set, the property of membership of set theory was used, in order to define the impossible event, the first axiom of sets,[2, 3] the axiom of existence.

[2] The six (6) axioms of set theory will be discussed in detail in the chapter.

[3] We are going to indicate the axiom number in parentheses.

The *Axiom of Existence* (1) implies that there exists a set that has no elements. Such a set is called *empty* and is denoted as ∅. Applied to a random experiment, it allows defining the impossible event as one that is never verified, that is, it is an event such that whenever the experiment is carried out, none of the elementary events that compose it is ever obtained. From which it follows that the impossible event is the empty set ∅. From the sure event and complementary event definitions, it follows that $\overline{E} = \varnothing$ and $\overline{\varnothing} = E$

To see an example of the concept of an impossible event in rolling a dice, it would be not to obtain a number between 1 and 6, which cannot be given; therefore, it is a set without any element or empty ∅.

Sets Theory Axioms and Operations

In the previous subsection, when we introduced the concept of the impossible event, we had to introduce the first axiom of set theory, the Axiom of Existence, but to understand in depth most of the concepts related to probability, we needed to introduce more definitions from set theory, more axioms, such as the axioms of extensionality, pair, and union, and set operations, such as union and intersection of sets.

The *Axiom of Extensionality* (2) states the following: If every element of A is an element of B and every element of B is an element of A, then A = B; that is, if two sets have the same elements, then they are identical. This allows us to define the motto: there is only one empty set or impossible event. From this axiom, the *inclusion* relationship between sets can be defined as follows: A is a subset of B if all the elements of A belong to B, but not vice versa, that is, not all the elements of B belong to A, or what is the same A is a subset of B if for all $x \in A$ x implies that $x \in B$ x. The relation of inclusion of A in B is denoted as $A \subseteq B$. The relation of incluision verifies the following properties:

- Reflective. $A \subseteq B$
- Transitive. If $A \subseteq B$ and $B \subseteq C$ B, it is verified that $A \subseteq C$
- If $A \subseteq B$ and B ⊆ A, the extensionality axiom $A = B$ is verified.
- If $A \subseteq B \rightarrow \overline{B} \subseteq A$.

To see an example of the inclusion relation, we can continue using the previous example with A = {2} and B = {2, 4, 6}, which means A ⊂ B, because all elementary events, that is, 2, which are part of A, are also part of B, since 2 is an element of B, but not the inverse, since B also has events 4 and 6 that are not part of A. Or what is the same, as long as A is given, that is, a 2 is obtained, B will also be occurring, but not always if B is given, A will be given, because they may be giving a 4 or a 6, which are not part of A.

After we have introduced the second axiom, we introduce the third and the fourth axioms, pair and union, respectively, because we need to know two of them before introducing the operations that can be done with sets.

The *Axiom of Pair* (3) is the third axiom of set theory, and it states that for all A and B, there exists a set C such that $x \in C$ x if and only if $x \in A$ x or $x \in B$.

The *Axiom of Union* (4) is the fourth axiom of set theory, and it states that for any set S, there exists a set U such that $x \in U$ if and only if $x \in A$, for any $A \in S$.

The pair and union axioms are necessary to define the union operation between sets and the extensionality to guarantee that it is unique. In addition to the pair and union axioms, it is necessary to define the union operation between sets; extensionality is also necessary to guarantee that it is unique. Once we have seen the axioms of extensionality, pair and union, we are going to define the four operations union, intersection, difference, and symmetric difference, of sets.

1. *Union or sum of sets* is defined as follows: Let A and B be two sets, the union of A and B is defined, A ∪ B as the set whose elements belong to A or B, or otherwise, let A and B∈ P(E), ∪: P(E) × P(E) → P(E)/(A, B) → A∪B. The Union of sets is denoted[4] as ∪. Applied to a random experiment with sample space E and two sets of P(E), the set A∪B can be defined as that composed of all the elementary events that make up A or B or both. Consequently, event A∪B is verified if at least one of the two events, A or B.

To see an example of the Union of sets, we will follow the previous example about the rolling of a dice in which the sets or events A, B, and C were: A, get a 2, consequently the subset of E is {2}; B: get an even number, the subset of E is {2, 4, 6}; and C: get a number other than 1 or 2, the subset of E is {3, 4, 5, 6}. The union of events A and C, $A \cup C$, is $A \cup C = \{2, 3, 4, 5, 6\}$.

2. *Intersection of sets* is defined as follows: Let A and B be two sets, define the intersection of A and B, A∩B as the set whose elements belong to A and B, or otherwise, let A and B∈ P(E), ∩: P(E) × P(E) → P(E)/(A, B) → A∩B. The intersection of sets is denoted as ∩. Applied to a random experiment with sample space E and two sets of P(E), the set A∩B can be defined as that composed of all the elementary events that make up A and B, and it is necessary that they belong to A and B at the same time. The event A∩B is verified if the two events, A and B, are verified when performing the experiment. The definition of the intersection of sets allows us to define incompatible events as those whose intersection is the empty set. Events A and B are said to be incompatible if A∩B = Ø.

To see an example of the Intersection of sets, we will follow the previous example of the intersection of the events B = {2, 4, 6} and C = {3, 4, 5, 6}, B∩C, is B∩C = {4, 6}. The intersection of A and C is A∩C = Ø, so they are incompatible events, which is logical if we pay attention to the definitions of both events since event A is to obtain a 2 and event C is to obtain a number other than 1 and 2.

[4]The symbol ∪ can also be written as a plus with a period above it ⊍.

3. *Difference of sets* is defined as follows: Let A and B be two sets, define the difference of A and B, A-B as the set whose elements belong to A and not to B, or otherwise, they are A and B $\in$ P(E), A-B: {x $\in$ A and x $\notin$ B}. Using the intersection operation and the definition of a complementary set of a given set, we have that A-B = A $\cap$B complementary (the difference of sets is denoted as -). Applied to a random experiment with sample space E and two sets of P(E), the set AB can be defined as that composed of all the elementary events that compose A and do not compose B; it is necessary that they belong to A and not to B. Event AB is verified if, when performing the experiment, event A is verified and not event B.

To see an example of the Difference of sets following the previous example, the difference of events B and C, B-C, is B-C = {2}. The difference between A and B is A-B = Ø because all the elementary events in A are also in B.

4. *Symmetric difference* of sets is defined as follows: Let A and B be two sets, define the symmetric difference of A and B, $A \triangle B$ as the set whose elements belong to A and B, minus those that belong to both at the same time, or otherwise, using the union and intersection operations, let A and B $\in$ P(E), $A \triangle B$:{x $\in$ (A $\cup$ B) and x $\notin$ (A $\cap$ B)}. Using the difference operation, we have that "$A \triangle B$ = (A-B) $\cup$ (B-A). The difference of sets is denoted as $\triangle$. Applied to a random experiment with sample space E and two sets of P(E), the set $A \triangle B$ can be defined as that composed of all the elementary events that compose A and compose B, but do not compose A and B at the same time, it is necessary that they belong to A and B, but not to both at the same time. Event A-B is verified if, when performing the experiment, events A and B are verified, but not through common elementary events but through noncommon ones.

To see an example of the Symmetric Difference of sets following the previous example, the symmetric difference of events B and C, B-C, is $B \triangle C = \{2, 3, 5\}$. The symmetric difference between A and B is A-B = {4,6}.

The *Axiom Power Set* (5) is the fifth axiom of set theory, and it states that for any set S, there exists a set P such that $A \subset P$ if and only if $A \subseteq S$. As the set P is uniquely determined, the set of all subsets of S can be called the power set or parts of S, which is denoted P(S). Applied to a random experiment with sample space E, the set of parts of E can be defined as one whose elements are all possible subsets of E. In the set theory, for any set A, the set of parts of A, P(A), is the set formed by all the subsets that can be formed with the elements of A. The set P(E) with the inclusion relation (P(E), $\subset$) is a partially ordered set, where E is the maximal and $\varnothing$ is the minimal.

Each element of P(E), that is, each subset of E, is one of the different events or results, both elementary and non-elementary, that can occur when performing the random experiment; that is, it is the set of all possible results that can be given when conducting the experiment, so P(E) is called the event space of the random experiment. From this, it is logical to ask: how many are the elements of P(E)? The solution is calculated using combinatorics. If the cardinal of E is n, that is, the number of elementary events or elements of E is n, a set of n elements contains $\binom{n}{m}$

subsets of m elements each, since they are combinations of n elements taken from m to m. They are combinations and not variations because two subsets are different only if they have one or more different elements, not if the elements are listed in a different order. Since the number of subsets of E would be the sum of all the possible subsets that could be formed with the elements of E, subsets that could have from 0 to n elements, the sum would be $\binom{n}{0} + \binom{n}{1} + \ldots + \binom{n}{n-1} + \binom{n}{n} = 2^n$. Therefore, the number of elements of P(E) is 2^n.

To see an example of the set Parts of E, we take the same E as in the example of the E of the rolling of a die, that is {1,2,3,4,5,6}, starting for this, in roll of a die, the set of parts of E is:

P(E) = {∅, 1, 2, 3, 4, 5, 6, {1,2 }, {1,3}, {1,4}, {1,5}, {1,6}, {2,3}, {2,4}, {2,5}, {2,6}, {3.4}, {3.5}, {3.6}, {4.5}, {4.6}, {5.6}, {1,2,3}, {1,2,4}, {1,2,5}, {1,2,6}, {1,3,4}, {1,3,5}, {1,3,6}, {1,4,5}, {1,4,6}, {1,5,6}, {2,3,4}, {2,3,5}, {2,3,6}, {2,4,5}, {2,4,6}, {2,5,6}, {3,4,5}, {3,4,6}, {3,5,6}, {4,5,6}, {1,2,3,4}, {1,2,3,5}, {1,2,3,6}, {1,2,4,5}, {1,2,4,6}, {1,2,5,6}, {1,3,4,5}, {1,3,4,6}, {1,3,5,6}, {1,4,5,6}, {2,3,4,5}, {2,3,4,6}, {2,3,5,6}, {2,4,5,6}, {3,4,5,6}, {1,2,3,4,5}, {1,2,3,4,6}, {1,2,3,5,6}, {1,2,4,5,6}, {1,3,4,5,6}, {2,3,4,5,6}, {1,2,3,4,5,6}}.

The number of elements of parts of E on the roll of a die is

$$2^{\mathrm{n}} = 2^6 = 64$$

In addition to the axioms described in detail before, due to their clear application to the study of the probability of events, in the set theory, one more axiom is defined, other than the sixth, the Understanding axiom (6), which has not been dealt with in the text, as it is not of such obvious utility to its domain.

Laplace or Classic Probability

The formal axiomatic mathematical definition of the concept of probability from set theory was given by the Russian mathematician Andrei Nikolayevich Kolmogorov, and it can be found in the Annex about Advanced Concepts of Probability in this chapter for those readers interested in a deeper mathematical knowledge on probability. In this subsection, we introduce the definition of probability known as the classical definition of probability or Laplace probability, and this definition will be sufficient for the scope of this book. The properties of probability, the probability of joint occurrence of events, the Bayesian probability, and the probability distributions will also be introduced in this chapter.

The classical definition of probability says that the probability of the occurrence of an event A is equal to the number of cases in which A appears, n_A, divided by the total number of cases, n_T. The mathematical equation for its calculation is:

$$p(A) = \frac{n_A}{n_T}$$

To give some examples of the classic or the Laplace definition of probability, we are going to follow using the roll of a die and try to answer different questions using the definition of probability.

The first one is: What is the probability of rolling a 5 on a die?

If the classical definition of probability is taken: $p(A) = {}^{n_A}\!/_{n_T}$, the number of favourable cases, that is, in which A is obtained, which in this case is to obtain a 5, and which consequently is 1, and the possible cases, which in the case of rolling a die are 6, since it is possible to obtain:[5] 1, 2, 3, 4, 5, or 6. Therefore, the probability of obtaining a 5 in the roll of a die is:

$$p(A) = \frac{n_A}{n_T} = \frac{1}{6} = 0.16$$

Or what is the same[6] a 16%.

What is the probability of obtaining an even number when rolling a die?

In this case, the event obtaining an even number is formed by the set A = {2, 4, 6}, and consequently, the number of favourable cases is 3, and the possible cases, as in the previous case, are 6 since the sample space is the same. Therefore, the probability of obtaining an even number in the roll of a die is:

$$p(A) = \frac{n_A}{n_T} = \frac{3}{6} = 0.5$$

Or what is the same a 50%.

Once we have seen the concept of Classical probability, we are going to see the properties it fulfils. To do that, if A and B belong to a sample space on which a classical probability is applied, it is verified:[7]

1. The probability of obtaining an event, A, is always greater than or equal to 0 and less than or equal to 1.

$$0 \leq P(A) \leq 1$$

2. The probability of obtaining the complementary event $\overline{A}$ of a given event, A, is equal to 1 minus the probability of obtaining A.

[5] As the concept of sample space is already known, it is observed that what can be obtained is the sample space of the experiment.

[6] Sometimes, to see the result clearer, the same is given as a percentage. If this is the case, the result is a 1 chance of rolling a 5 on the roll of a die.

[7] The proof of each of them can be found in the later section of the chapter *Advanced Concepts of Probability*.

$$P(\overline{A}) = 1 - P(A)$$

3. If event A is contained in event B, the probability of obtaining event A is less than or equal to the probability of obtaining event B.

$$A \subseteq B \longrightarrow \mathrm{P(A)} \leq \mathrm{P(B)}$$

4. The probability of obtaining the union of two events A and B is equal to the sum of the probabilities of obtaining each of the two events minus the probability of obtaining the intersection of both events.

$$P(A \cup B) = P(A) + P(B) - P(A \cap B)$$

5. The probability of obtaining the empty set is zero.

$$P(\varnothing) = 0$$

Now we are going to present an example of each one of the properties of the probability, all the examples will continue to be about the throwing of a die:

1. $0 \leq P(A) \leq 1$: Continuing with the example that we are using of the roll of the dice, the lowest probability that we can obtain is 0, which is the probability of obtaining a value other than 1, 2, 3, 4, 5, or 6, which, applying the classical probability equation, is:

$$p(A) = \frac{n_A}{n_T} = \frac{0}{6} = 0 \equiv 0\%$$

but you cannot get a *negative* probability. The highest probability that can be obtained is the certainty that it is going to happen, which will be obtained when calculating the probability of the coincident event with the sample space since it includes everything that can occur, that is, the event A = {1, 2, 3, 4, 5, 6}. If the classical probability equation is applied, we have:

$$p(A) = \frac{n_A}{n_T} = \frac{6}{6} = 1 \equiv 100\%$$

It is immediate to see that you cannot have an n_A greater than 6, so the division can never give a value greater than one.

2. $P(\overline{A}) = 1 - P(A)$: To see an example of this property, we return to the second of the examples that have been exposed to see the concept of classical probability: what is the probability of getting an even number on the roll of a die? As seen in this case, the event to obtain an even number is formed by the set A = {2, 4, 6}, as the sample space is E = {1, 2, 3, 4, 5, 6}, so its complementary set is

$\overline{A} = \{1, 3, 5\}$. The probability of A that was calculated in the previous example is 50%. If the probability of $\overline{A}$ is now calculated, it is obtained that the number of favourable cases is 3, and the possible cases, as in the previous case, are 6 since the sample space is the same. Therefore, the probability of obtaining an odd number in the roll of a die is:

$$p(\overline{A}) = \frac{n_{\overline{A}}}{n_T} = \frac{3}{6} = 0.5 \equiv 50\%$$

That is,

$$P(\overline{A}) = 1 - P(A) = 0.5$$

3. $A \subseteq B \longrightarrow P(A) \leq P(B)$: To expose an example of this property, we need two sets, A and B, in such a way that A is contained in B. We take B as the event to obtain a number less than 6 in the roll of a die, which is formed by the elements B = {1, 2, 3, 4, 5}. We are going to take as a set B the event to obtain an odd number, which, as we know from the previous exercise, is formed by the events A = {1, 3, 5}. Therefore, it is immediate to check that $A \subseteq B$. If we calculate the probabilities of both events, it is also immediate to check that the probability of obtaining A is less than or equal to the probability of obtaining B since the probability of obtaining B is:

$$p(B) = \frac{n_B}{n_T} = \frac{5}{6} = 0.83 \equiv 83\%$$

and obtaining A will always have a lower number of favourable cases, since A is contained in B,

$$p(A) = \frac{n_A}{n_T} = \frac{3}{6} = 0.5 \equiv 50\%$$

4. $P(A \cup B) = P(A) + P(B) - P(A \cap B)$: As example sets for this property, we take the events to obtain a number less than four and an odd number. These sets are A = {1, 2, 3} and B = {1, 3, 5}. From them, the first thing we obtain is their union: $A \cup B = \{1, 2, 3, 5\}$. Once we have the three events, we apply the classical probability calculation equation to obtain their probabilities:

$$p(A) = \frac{n_A}{n_T} = \frac{3}{6} = 0.5 \equiv 50\%$$

$$p(B) = \frac{n_B}{n_T} = \frac{3}{6} = 0.5 \equiv 50\%$$

$$p(A \cup B) = \frac{n_{A \cup B}}{n_T} = \frac{4}{6} = 0.66 \equiv 66\%$$

As seen immediately, the probability of $A \cup B$ is not equal to P(A) + P(B), which would give 1 but is 0.66. The reason that the intersection has to be removed is that when we calculate the probability of obtaining A, we are including as favourable cases obtaining a 1 and a 3, and when we calculate the probability of obtaining B, we also include them, so if we simply add the probabilities, we are including them twice, while if we calculate the probability of the union, we only include them once. For this reason, we subtract the probability of the intersection to remove one of the two times we count them by adding the probabilities of the two individual sets.

$$p(A \cap B) = \frac{n_{A \cap B}}{n_T} = \frac{2}{6} = 0.33 \equiv 33\%$$

$$P(A \cup B) = P(A) + P(B) - P(A \cap B) = 0.5 + 0.5 - 0.33 = 0.66$$

5. $P(\varnothing) = 0$:In $\varnothing$, the toss of a die, as seen when studying the complementary set of a given die, is formed by the complementary event of its sample space. Since the sample space of the roll of the dice is E = {1, 2, 3, 4, 5, 6}, its complement is formed by the rest of the numbers plus zero. Consequently, if we apply the classical definition of probability seen above, in this case, the favourable cases, A, would be those in which an element of the sample space was not obtained, that is, those in which a different number of {1, 2, 3, 4, 5, 6}, which will be none because it is impossible, that is, $n_A = 0$. Consequently,

$$p(A) = \frac{n_A}{n_T} = \frac{0}{6} = 0$$

Thus far, we have seen concepts associated with the probability of a single event, but the probability of the occurrence of more than one event, from 2 to n, can also be studied. We study the concepts associated with the appearance of two events because this knowledge can be applied to any number of events. The probability of joint appearance of two events is defined as follows: Let A and B be two events that appear together, and the probability of appearance of A, having given B, $p(A|B)$, is defined as:

$$p(A|B) = \frac{n_{A \cap B}}{n_B}$$

From the previous definition, the concepts of independent and dependent events can be defined as follows:

- *Independent Events*. Two events are independent if the occurrence of one of them does not change the probability of occurrence of the other:

$$p(A|B) = p(A) \rightarrow p(A \cap B) = p(A)P(B)$$

- *Dependent Events*. Two events are dependent if the occurrence of one of them changes the probability of occurrence of the other. The conditional probability deals with obtaining the probability of events whose appearance is influenced by the appearance of other events. Thus, if A and B are two dependent events, the probability of appearance of A, having given B, is defined as:

$$p(A|B) \neq p(A) \rightarrow p(A \cap B) = p(A|B)P(B)$$

  Since then,

$$p(A|B) = p(B|A) \rightarrow p(A|B)p(B) = p(B|A)p(A)$$

To understand what means independent and dependent events and their differences, we solve the two following examples: There are two urns, A and B, with 10 balls. In urn A, there are 4 white balls and one back ball. In urn B, there are 2 white balls and 3 black balls. It must be calculated as follows:

(a) Probability of obtaining two white balls in two extractions made one in each urn.
  As each extraction is made in an urn, the two events are independent.
  The probability of obtaining a white ball in urn A is the number of favourable cases, which is 4 because there are four white balls, divided by the number of possible cases, which is 5 because there are 5 balls in urn A.

$$P(A) = \frac{n_{\text{white } UA}}{N} = \frac{4}{5} = 0.8$$

  If we apply the same reasoning to urn B, we have:

$$P(B) = \frac{n_{\text{white } UB}}{N} = \frac{2}{5} = 0.4$$

  As the extraction of the balls in each urn does not affect the other and the events are independent, the join probability of both events is:

$$P(A \cap B) = P(A)P(B) = 0,8.0,4 = 0,32$$

(b) Calculate the probability of obtaining two white balls in two extractions made in the same urn, A.
  As the extraction is made in the same urn, the events are dependent.

The probability of obtaining a white ball in urn A is the number of favourable cases, which is 4 because there are four white balls, divided by the number of possible cases, which is 5 because there are 5 balls in urn A:

$$P(A) = \frac{n_{\text{white } UA}}{N} = \frac{4}{5} = 0.8$$

As the extraction is made in the same urn, the first event alters the probability of the second, because as one ball has been extracted in the previous step, the number of favourable and possible cases has been both changed, now there 3 favourable cases in 4 possible cases, and the probability is:

$$P(B|A) = \frac{n_{\text{white } UA}}{N} = \frac{3}{4} = 0.75$$

As the extraction of the first ball in the same urn affects the extraction of the second ball in the same urn, the events are dependent, and the join probability of both events is:

$$P(A \cap B) = P(B|A)P(A) = 0,8.0,75 = 0,6$$

It is interesting to note that the conditional probability provides the probability of the occurrence of event A when event B has occurred, regardless of whether the probability of occurrence of B is high or low, that is, B may have a low probability of appearance, but if almost every time that B happens A also happens, then the probability of appearance of A given B will be very high.[8] This conclusion can also be reached by developing the equation of the conditional probability:

$$p(A|B) = \frac{p(A \cap B)}{P(B)} = \frac{\frac{n_{A \cap B}}{n_T}}{\frac{n_B}{n_T}} = \frac{n_{A \cap B}}{n_B}$$

From the dependence and independence of events, we can establish the concepts of *A posteriori* probability and *A priori* probability.

$P(A|B)$ is the probability of occurrence of A being happened B, and it is the *A posteriori* probability.

$P(B)$ is the probability of occurrence of B without anything happening before, and it is the *A priori* probability.

From these two definitions, if there is a set of previous exclusionary events *A priori* (A_i) that modify the occurrence probability of other event *A posteriori* (B), the *Total Probability Theorem* is verified.

[8]The previous conclusion will have an interesting application in the study of the confidence parameter in the association studies.

$$P(B) = \sum_{i=1}^{n} P(B|A_i)P(A_i)$$

As an example of the *total probability theorem,* we can take the same urns and balls as in the previous example, and we calculate the probability of obtaining a white ball when a ball of any a ball is drawn from either of the two urns.

If we apply the total probability theorem:

$$P(B) = \sum_{i=1}^{n} P(B|A_i)P(A_i)$$

And the event B is to obtain a white ball.

First, its probability is conditioned by the chosen urn, which can be written as $P(A_i)$, and second, the probability of obtaining a white ball in each urn can be written as $P(B|A_i)$. With these definitions, the problem can be solved in the following way:

Probability of selecting urn A: $P(A_1) = \frac{\text{favourable cases}}{\text{possible cases}} = \frac{\text{one urn}}{\text{two urns}} = \frac{1}{2}$

Conditioned probability of choosing a white ball if urn A has been selected:

$$P(B|A_1) = \frac{n_{\text{white } UA}}{N} = \frac{4}{5} = 0.8$$

Probability of selecting urn B: $P(A_2) = \frac{1}{2}$

Conditioned probability of choosing a white ball if urn B has been selected:

$$P(B|A_2) = \frac{n_{\text{blancas } UB}}{N} = \frac{2}{5} = 0.4$$

Consequently, the probability of selecting a white ball for any urn is:

$$P(B) = \sum_{i=1}^{2} P(B|A_i)P(A_i) = P(B|A_1)P(A_1) + P(B|A_2)P(A_2) = 0,8.0,5 + 0,4.0,5 = 0,4 + 0,2 = 0,6$$

Bayesian Probability

Thus far, we have the following equations:

$$P(A \cap B) = P(A|B)P(B) = P(B|A)P(A)$$

$$P(A|B) = \frac{P(A \cap B)}{P(B)}$$

$$P(B) = \sum_{i=1}^{n} P(B|A_i)P(A_i)$$

From where we can obtain the *Bayes Probability*:

$$P(A_i|B) = \frac{P(A_i \cap B)}{P(B)} = \frac{P(B|A_i)P(A_i)}{B} = \frac{P(B|A_i)P(A_i)}{\sum_{i=1}^{n} P(B|A_i)P(A_i)}$$

Bayes' Theorem allows us to know the *A priori* (A_i) probability in terms of what happened *A posteriori* (B), and it is, in consequence, a total change of paradigm because it is the opposite of what we have been doing until now.

To see an example of the Bayes Theorem, we calculate the probability that once a white ball is obtained, it comes from urn A.

We apply the Bayes Theorem to obtain the probability *A priori* that the white ball comes from urn A from the known *A posteriori* fact that we have a white ball. The Bayes probability equation is:

$$P(A_i|B) = \frac{P(B|A_i)P(A_i)}{\sum_{i=1}^{n} P(B|A_i)P(A_i)}$$

B event is to have obtained a white ball. To have obtained a white ball conditions the selection of the urn A_i. The probability of having chosen A_i when B has been obtained is $P(A_i|B)$. From these definitions, we have:

Probability of selecting urn A: $P(A_1) = \frac{\text{favourable cases}}{\text{possible cases}} = \frac{\text{one urn}}{\text{two urns}} = \frac{1}{2}$

Conditioned probability of choosing a white ball if urn A has been selected:

$$P(B|A_1) = \frac{n_{\text{white } UA}}{N} = \frac{4}{5} = 0.8$$

Probability of selecting urn B: $P(A_2) = \frac{1}{2}$

Conditioned probability of choosing a white ball if urn B has been selected:

$$P(B|A_2) = \frac{n_{\text{blancas } UB}}{N} = \frac{2}{5} = 0.4$$

Consequently, by applying the Bayes probability, we have the probability that the white ball comes from urn A:

$$P(A_1|B) = \frac{P(B|A_1)P(A_1)}{\sum_{i=1}^{2} P(B|A_i)P(A_i)} = \frac{P(B|A_1)P(A_1)}{P(B|A_1)P(A_1) + P(B|A_2)P(A_2)}$$
$$= \frac{0,8.0,5}{0,8.0,5 + 0,4.0,5} = \frac{0,4}{0,6} = 0.67$$

Probability Distribution of Random Variables

We will start this subsection by introducing the concept of Random Variable and their associated knowledge, and after that, we will introduce probability distributions, both discrete and continuous.

Random Variable

In the topic dedicated to Data, the concept of characteristic, or attribute, and that of data associated with a characteristic, and its frequency were studied in detail, and the concept of a statistical variable was presented. At this point, we remember that its definition is A real function defined on a finite population or a sample, which takes the values of each of the modalities of an attribute and to which it associates a frequency distribution. Once the concepts of probability have been seen, the concept of a statistical variable can be extended to the concept of a *random variable*, and a random variable can be defined as: "variable associated with a certain law or probability distribution, in which each of the values that it can take corresponds to a specific relative or probability frequency".

That is, when you have all the data of a characteristic, that is, when a sample is being studied, the frequency distributions of the values of said characteristic will be known, so, in that case, what you have is a statistical variable. On the opposite, when you do not have all the data of a characteristic, that is, when a population is being studied, the probability distributions of the appearance of the data of that characteristic will be known, so, in that case, what is has is a random variable.

Probability Distributions

From that definition of a random variable, we can conclude that before beginning to study in depth the concept of discrete or continuous probability distributions, it is very important to remember the concept of *Frequency Distribution*, and a frequency distribution is formed by the pairs composed of each data and its frequencies. If it is the absolute frequency, it will be called the absolute frequency distribution, and the same is true for the rest of the types of frequencies seen, both for discrete data and for data groupings. If we associate this definition with that of a statistical variable, it is immediate to conclude that each statistical variable will have its frequency distribution. Among the types of frequencies defined, in this subsection, we will stop at the relative frequency, which we remember is defined for the appearance of a certain value of a data as the number of times that value appears between the number of data.

Now we are going to present an example of a frequency distribution using the values of numbers of rainy days in May in Madrid, Spain, in the 1940–1960 period: {0,0,0,0,0,8,15,8,12,9,9,9,10,2,2,2,6,8,8,8,9,6} From these data, we are going to

answer the following questions: What are the relative frequencies of the values of rainy days in May? How many days will it rain in May of any year?

From the information contained in this data set, we obtain the relative frequency of each data point, for which we use the well-known equation:

$$fr_i = \frac{n_i}{n_T}$$

and we apply it to each value other than the ones we have in the set, containing the n_i:

$$n_1(0) = 5, n_2(8) = 5, n_3(15) = 1, n_4(12) = 1, n_5(9) = 4,$$
$$n_6(10) = 1, n_7(2) = 3, n_8(6) = 2$$

and

$$n_T = 22$$

Therefore, the relative frequencies of the values are:

$$fr_1 = \frac{n_1}{n_T} = \frac{5}{22} = 0.227, fr_2 = \frac{5}{22} = 0.227, fr_3 = \frac{1}{22} = 0.045,$$
$$fr_4 = \frac{1}{22} = 0.045, fr_5 = \frac{4}{22} = 0.182, fr_6 = \frac{1}{22} = 0.045,$$
$$fr_7 = \frac{3}{22} = 0.136, fr_8 = \frac{2}{22} = 0.091,$$

With these data, the first question is answered, but it is important to emphasize the fact that all these values are completely true, that is, the proportion of times that in the period 1940–1960 it did not rain in May in Madrid was approximately 23% of the years, and that is for sure. And the number of rainy days is a statistical variable. Regarding the second question, it cannot be answered from the information provided in the statement of this question since the conclusions obtained are only applicable to the period studied 1940–1960. It could be said that in this period, the most common number of rainy days in May was none or 8.

When what is studied are not statistical variables but random variables, we do not know the frequency of appearance of the data or values that the variable may have, but rather the probability of appearance of each data or value, so what we do is not a distribution of frequencies, which we do not know, but a distribution of probabilities, which is what we know. [RAE2014] defines distribution as follows: 4. f. Mat. Function that represents the probabilities that define a random variable or a random phenomenon.

As an example of the probability distribution, we can take the probabilities of rainy days in May in Madrid as p(0) = 22.7%, p(2) = 13.6%, p(6) = 9,1%, p(8) = 22.7%, p(9) = 18.7%, p(10) = 4.5%, p(12) = 4.5%, p(15) = 4.5%. From these data, how many days will it rain in May of any year?

In this example, unlike the example of relative frequencies, what we have are the probabilities of different numbers of days of rain in May and those probabilities are applicable to any month of May of any year, so with these data can be answered to the question they ask us, which is the same as the one they asked us in the absolute frequencies example and in that case we could not answer. The answer would be that it will most likely rain, since it is approximately 88%, compared to 22% that it will not rain, and it will most likely rain 8 or 9 days.

From the two previous examples on relative frequency and probability, the immediate reflection that arises is: from the data on the number of rainy days in May provided over 22 years, the relative frequencies of occurrence of a different value of rainy days, but how were the probabilities given in the probabilities example obtained? If the numbers of the relative frequencies obtained and those of the probabilities provided are compared, it is seen that they are similar, so the only thing that has been done is to assume that the relative probabilities of the different values obtained from the studied population, the data from the years 1940–1960 are not only valid for said population but are also valid if that period is taken only as a sample and all subsequent or previous years are taken as a population, in such a way that the relative frequencies become the probabilities of appearance of the different values.

The conclusion of what is stated in the previous paragraph is that a possible way to obtain the probabilities of the appearance of a set of values of a variable is to take a sample of them and obtain the probabilities from the relative frequencies that are calculated in it. This is a valid way to do so, but the correction of its results, that is, the degree of success that we will have when approximating the probabilities by the relative frequencies, will depend on some factors, which will be discussed in more detail in later chapters, and between those in which the size of the sample and its representativeness play a predominant role. However, when there are few values in discrete random variables, it is possible to give the values of the probability of each value. An example of this type of case is the one used in the examples above with rainy days in a month. In the examples, the probabilities of eight values were given, but the maximum that could be given would be 31 values, since the months do not have more than those days. However, if the discrete variable has many values or if it is a continuous variable, it is impossible to give the probabilities of the values in this way, and the only way is to use mathematical functions, which allow obtaining the probabilities of each value as an output variable, having used the value itself as the input variable of the function. These functions are called *Probability Distribution Functions*.

Depending on the type of characteristic, which in this case will be random variables that are being analysed, dictated or continuous, there will be either a discrete or continuous probability distribution. In this chapter, we will study the main distributions of each kind. The distributions to be studied are those that appear in a more generalized way, but in a specific study, a distribution that is not one of the general ones can be obtained. Discrete probability distributions and in the next the continuous

- Discrete probability distributions are usually called *Probability Functions*,

$$p(x_i) = p(x = x_i)$$

- And continuous probability functions are often called *Density functions*,

$$p(x_i) = f(x = x_i)$$

Associated with the concept of the probability distribution of a random variable, the concepts of Probability Function and Density Function have just been presented. In addition to these two concepts, there is another concept that is fundamental in working with the probability distributions of random variables: and that is *Distribution Function*. A distribution function is a discrete or continuous mathematical function that specifies for each value of the sample space of a discrete or continuous random variable the probability of the appearance of said value and of all those less than it:

$$F(x_i) = p(x \leq x_i)$$

From this definition, it is immediate to conclude that the value of the distribution function of the highest value in the population studied must be equal to 1 since it will provide the sum of the probability of obtaining all the values of the population. To better understand this conclusion, it is possible to return to what has been seen above and think of the probability of appearance of each value as the extension to a population with unknown data of the relative frequencies obtained for values of a population with all known data. The relative frequency of a certain value is obtained as the quotient of the times that this value appears between the total number of data available, so if we add all the relative frequencies, the result must be 1. Consequently, if the value of the distribution function were calculated at the highest value for the population and it did not turn out to be 1, the mathematical function that is being considered as a probability function or a density function is not actually one.

If it is a discrete random variable:

$$F(x_k) = p(x \leq x_k) = \sum_{i=1}^{k} p(x_i)$$

And if x_n is the largest value in the sample space,

$$F(x_n) = \sum_{i=1}^{n} p(x_n) = 1$$

If it is a continuous random variable:

$$F(x_k) = p(x \leq x_k) = \int_{-\infty}^{k} x_i dx$$

If x_n is the largest value in the sample space,

$$F(x_n) = \int_{-\infty}^{+\infty} x dx = 1$$

As an example of the probability distribution and the density function, we can take a die that is rolled three times, and the probability function and the distribution function must be calculated to obtain an even number.

The first thing to understand is why they ask us for a probability function and not a density function. The reason is that only 4 discrete values can be obtained, or 0 even numbers are obtained in the three throws, or 1 is obtained, or 2 are obtained, or 3. In other words, only the probabilities of obtaining four discrete values can be given, and consequently, it is a probability function. Next, we must obtain the function obtaining the probabilities of each option.

The sample space is: E = {eee, eeo, eoo, ooo, ooe, oee, oeo, eoe}

and since these are three independent events since a subsequent roll of the die does not influence any previous roll at all, the probability of intersection of the three is the product of the probabilities of each one.

$$p(eee) = p(A \cap B \cap C) = p(A)p(B)p(C)$$

Consequently, the first thing we have to calculate is the probability of obtaining an even and odd number. In each die, there are three even numbers and three odd numbers, the number of favourable cases in each case is 3, and the number of possible cases is 6, so applying the classic definition of probability, we obtain:

$$p(e) = p(o) = \frac{n_e}{n_T} = \frac{n_o}{n_T} = \frac{3}{6} = 0.5 \equiv 50\%$$

Therefore,

$$p(eee) = p(e)p(e)p(e) = 0.5 \cdot 0.5 \cdot 0.5 = 0.125$$

It is the same for the rest of the possibilities:

$$\begin{aligned} p(eee) &= p(eeo) = p(eoo) = p(ooo) = p(ooe) = p(oee) = p(oeo) \\ &= p(eoe) = 0.125 \end{aligned}$$

Once these values have been obtained, the probability function can be defined as follows: The probabilities of the four possible outcomes have to be obtained: 0, 1,

2, and 3. The probability of obtaining 0 even numbers is very easy to calculate, and only p(ooo) is given, which is 0.125. The probability of obtaining an even number is given by the events p(eoo), p(ooe), and p(oeo). Since they are disjoint events, the probability of the union of the three events is obtained by adding their probabilities:

$$\begin{aligned} p(eoo) \cap p(ooe) \cap p(oeo) &= p(eoo) + p(ooe) + p(oeo) \\ &= 0.125 + 0.125 + 0.125 = 0.375 \end{aligned}$$

and the same happens for the probability of obtaining two even numbers:

$$\begin{aligned} p(eeo) \cap p(oee) \cap p(eoe) &= p(eeo) + p(oee) + p(eoe) \\ &= 0.125 + 0.125 + 0.125 = 0.375 \end{aligned}$$

Finally, to obtain three even numbers, the same probability is obtained as to obtain none.

$$p(ooo) = p(o)p(o)p(o) = 0.5 \cdot 0.5 \cdot 0.5 = 0.125$$

Once this last result has been calculated, the probability function can be given as:

$$p(x) = f(x), x \in \{0,1,2,3\} = \begin{cases} 0.125 & \text{if } x = \{0,3\} \\ 0.375 & \text{if } x = \{1,2\} \\ 0 & \text{if } x = \text{other value} \end{cases}$$

Then, the distribution function is obtained by applying its definition:

$$F(x_k) = p(x \leq x_k) = \sum_{i=1}^{k} p(x_i)$$

$$F(x) = f(x), x \in \{0,1,2,3\} = \begin{cases} 0.125 & \text{if } x \leq 0 \\ 0.5 & \text{if } 0 < x \leq 1 \\ 0.875 & \text{if } 1 < x \leq 2 \\ 1 & \text{if } x \geq 3 \end{cases}$$

as it can be verified that $x_4 = 3$ is the largest value of the sample space:

$$F(x_4) = \sum_{i=1}^{4} p(x_i) = 1$$

Therefore, it can be concluded that $p(x)$ is a function probability

Next, we present an example of a density function and its associated distribution function. Obtaining a density function from data observed in an experiment, as we

have done in the previous example, is a very complicated process that is beyond the scope of this text, but what we can determine from the concepts seen is, on the one hand, if a function proposed as a density function is truly such; and on the other hand, we can obtain the distribution function associated with a density function and the density function associated with a distribution function. We will present an example of each of these tasks.

If you have the following mathematical function:

$$f(x) = \begin{cases} mx^2 & \text{if } 0 < x \leq 2 \\ 3 - mx & \text{if } 2 < x \leq 4 \\ 0 & \textit{if } x = \text{other value} \end{cases}$$

We are going to see what value m must have to be a probability density function, and we are going to calculate the probability distribution function.

As explained above, for the sum, in this case the integral, to be a probability density function, of the probabilities of all the values of the function, that is, all R, must be 1.

$$F(x_n) = \int_{-\infty}^{+\infty} x dx = 1$$

to obtain m, we apply it to the function we are studying:

$$F(x_n) = \int_{-\infty}^{0} 0 dx + \int_{0}^{2} mx^2 dx + \int_{2}^{4} 3 - mx dx + \int_{4}^{+\infty} 0 dx = 1$$

The result of these integrals is:

$$0 + m\frac{x^3}{3}\Big|_0^2 + 3x\Big|_2^4 - m\frac{x^2}{2}\Big|_2^4 + 0 = 1$$

$$m\frac{8}{3} + 3(4-2) - m\left(\frac{16}{2} - \frac{4}{2}\right) = 1$$

$$m\frac{8}{3} + 6 - 6m = 1$$

$$m\frac{8}{3} - m\frac{18}{3} = -5$$

$$-m\frac{10}{3} = -5 \rightarrow m = 1.5$$

That is, for $m = 1.5$, the studied function can be a probability density function; for any other m, it is not.

Next, we obtain its distribution function, with $m = 1.5$ determined in the previous step.

$$f(x) = \begin{cases} \frac{3}{2}x^2 & \text{if } 0 < x \leq 2 \\ 3 - \frac{3}{2}x & \text{if } 2 < x \leq 4 \end{cases}$$

We have to define it in all the intervals of existence of the function, that is, $(-\infty, 0)$, $(0, 2)$, $(2, 4)$, and $(4, +\infty)$. In each interval, we will have to define it for the variable used, in this case x, so in order to integrate over x we will make a change of variable to t. It must also be remembered that the distribution function will include, in addition to the probability of the appearance of the value in which it is being calculated, that of all values lower than it, and consequently the integral in an interval must include that of all intervals less than the same. We are going to obtain the function in each interval:

- $(-\infty, 0)$

$$F(x \leq 0) = \int_{-\infty}^{0} 0dx = 0$$

- $(0, 2)$

$$\begin{aligned} F(0 < x \leq 2) &= \int_{-\infty}^{0} 0dx + \int_{0}^{x} \frac{3}{2}t^2 dt \\ &= 0 + \frac{3}{2}\left(\frac{t^3}{3}\right)\Big|_0^x = \\ &= \frac{x^3}{2} \end{aligned}$$

- $(2, 4)$

$$\begin{aligned} F(2 < x \leq 4) &= \int_{-\infty}^{0} 0dx + \int_{0}^{2} \frac{3}{2}t^2 dt + \int_{2}^{x} 3 - \frac{3}{2}t dt \\ &= 0 + \frac{3}{2}\left(\frac{t^3}{3}\right)\Big|_0^2 + 3t\Big|_2^x + \frac{3}{2}\left(\frac{t^2}{2}\right)\Big|_2^x = \\ &= 4 + 3x - 6 - \frac{3}{4}x^2 + 3 = \\ &= 1 + 3x + \frac{3}{4}x^2 \end{aligned}$$

- $(4, \infty)$

$$
\begin{aligned}
F(x>4) &= \int_{-\infty}^{0} 0dx + \int_{0}^{2} \frac{3}{2}t^2 dt + \int_{2}^{4} 3 - \frac{3}{2}t dt + \int_{4}^{x} 0dx \\
&= 0 + \frac{3}{2}\left(\frac{t^3}{3}\right)\Big|_0^2 + 3t\Big|_2^4 + \frac{3}{2}\left(\frac{t^2}{2}\right)\Big|_2^4 + 0 = \\
&= 4 + 12 - 6 - \frac{3}{4}16 + 3 = \\
&= 1
\end{aligned}
$$

which is the result that it must give because the probabilities of all the values have already been added.

Consequently, the definitive distribution function is:

$$
F(x) = \begin{cases} 0 & \text{if } x \leq 0 \\ \dfrac{x^3}{2} & \text{if } 0 < x \leq 2 \\ 1 + 3x + \dfrac{3}{4}x^2 & \text{if } 2 < x \leq 4 \\ 1 & \text{if } x \geq 4 \end{cases}
$$

Each experiment that is carried out to determine the probability of the appearance of the different values of a random variable, both discrete and continuous, may provide a different probability or density function specific to the study being carried out, which must be established in said study. However, there is a set of distributions, both discrete and continuous, of great importance since they appear a large number of times in different types of studies and must be specifically studied.

Discrete Probability Distributions

These distributions, as we will see below, are related, and we are going to study them in an approximate order of complexity, starting with the simplest. In this subsection, we will study the probability functions, that is, the distribution functions associated with discrete random variables, and in the next subsection, we will study the density functions, that is, the distribution functions associated with continuous random variables. Before we begin, we recall the general equation of a probability and distribution function for discrete variables:

Probability function:

$$p(x_i) = p(x = x_i)$$

Distribution function:

$$F(x_k) = p(x \leq x_k) = \sum_{i=1}^{k} p(x_i)$$

If x_n is the largest value in the sample space,

$$F(x_n) = \sum_{i=1}^{n} p(x_n) = 1$$

The discrete probability distributions that we are going to study are the Bernoulli, Binomial, Geometric, and Poisson distributions.

Bernoulli Probability Distribution

The Bernoulli probability distribution, or function, is one of the simplest that can be defined since its sample space is composed of only two values, 0 and 1, E = {0, 1}, and the probabilities of appearance of the two values are complementary and constant, that is, the probability of appearance of value 0 is p, and the probability of appearance of value 1 is q; and $q = 1–p$. In addition, said probabilities of appearance must be constant whenever the same experiment is carried out and the experiments must be independent, that is, the result of one does not influence the result of the next.

The definition of the probability function is:

$$p(x_i) = p(x = x_i; x_1 = 0, x_2 = 1) = p^{1-x}.q^x$$

The Bernoulli probability function allows statistical studies to be carried out on those populations in which the variable studied in all its elements can only be classified into two categories, for example, correct elements and failed elements.

Next, we present an example of a density function and its associated. To see an example of Bernoulli's distribution function, we are going to use again the roll of a dice and different events that can be obtained. We start with a simple one: we are going to obtain the probability function of obtaining an even number when rolling a die, as seen the variables are 0, corresponding to obtaining an even number and 1, corresponding to obtaining an odd number or not obtaining an even number. Once 0 and 1 have been defined, then to fully define the function, p and q must be obtained, where p is the probability of obtaining an even number or 0, and q is its complementary, that is, the probability of not obtaining it. If the equation for obtaining the classical probability is applied to its calculation, the following is obtained:

$$p(\text{even}) = \frac{n_{\text{even}}}{n_{\text{Total}}} = \frac{3}{6} = 0.5$$

and since the probability of odd is complementary, it is also 0.5. Consequently, the function is:

$$p(x_i) = p(x = x_i; x_1 = 0, x_2 = 1) = p^{1-x}.q^x = 0.5^{1-x}.0.5^x$$

As you can see, it is a very simple function.

We are now going to see another example, slightly more complicated and that will serve as a basis for seeing in the next subsection an example in which the famous De Mere problem will be solved. We calculate the Bernoulli function of obtaining a six in the roll of a die. In this case, taking into account that the case we are looking for, that is, the one corresponding to the variable 0, is obtaining a six, p and q will be:

$$p(\text{even}) = \frac{n_{\text{six}}}{n_{\text{Total}}} = \frac{1}{6} = 0.17$$

and since the probability of not obtaining a six is complementary, it is $q = 1 - 0.17 = 0.83$. Consequently, the function is:

$$p(x_i) = p(x = x_i; x_1 = 0, x_2 = 1) = p^{1-x}.q^x = 0.16^{1-x}.0.87^x$$

From the Bernoulli probability distribution, a set of associated probability distributions can be defined, among which two of the most used are those defined in the two subsequent subsections: the Binomial and Geometric distribution functions.

Binomial Probability Distribution

The Binomial probability distribution, or function, extends the sample space used by the Bernoulli distribution to the set formed by the n elements that make up the population, $\{0,1,2, \ldots, n\}$, but as in Bernoulli's case, the observed characteristic can only be classified into two categories and the probabilities of appearance of the two categories are complementary and constant. That is, the probability of appearance of category A is p, and the probability of appearance of category B is q; and $q = 1 - p$. In addition, the probability of appearance of the categories must be constant and the events of each experiment must be independent. Hence, it is considered a distribution derived from that of Bernoulli.

The definition of the probability function is:

$$p(x_i) = p(x = x_i = r; x_1 = 0, 1, 2, \ldots, r, \ldots n) = \binom{n}{r} p^r.q^{n-r}$$

The probability function is obtained from Bernoulli's function by applying the property of the probability of independence of events and combinatorial calculus. Let us see how it is obtained:

The Bernoulli function allows us to obtain the probability of the appearance of the desired event in a single experiment by means of the function:

$$p(x_i) = p(x = x_i; x_1 = 0, x_2 = 1) = p^{1-x}.q^x$$

If two experiments are carried out, as they must be independent, the probabilities of the appearance of the desired events are multiplied, and the sample space is enlarged because the elementary event sought may no longer only appear or not but may also appear twice. We calculate the probability that it appears:

- Once: We will have for the first experiment:

  $$p(x_0) = p^{1-0}.q^0 = p$$

  but as it has already appeared in the first experiment in the second it cannot appear so both will have:

  $$p(x_1) = p^{1-1}.q^1 = q$$

  Then, the probability of occurrence once in two experiments will be the multiplication of both probabilities, which is:

  $$p(1) = p.q$$

- Twice: We will have for the first experiment:

  $$p(x_0) = p^{1-0}.q^0 = p$$

  and in the second the same, therefore, we will have:

  $$p(x_0) = p^{1-0}.q^0 = p$$

  Then, the probability of occurrence once in two experiments will be the multiplication of both probabilities, which is:

  $$p(2) = p.p = p^2$$

- No time: We will have for the first experiment:

  $$p(x_1) = p^{1-1}.q^1 = q$$

and in the second the same, therefore, we will have:

$$p(x_1) = p^{1-1}.q^1 = q$$

Then, the probability of occurrence once in two experiments will be the multiplication of both probabilities, which is:

$$p(0) = q.q = q^2$$

Let us now see what happens if we calculate the probability of obtaining two desired events in three observations. Following the previous construction rule, we would have for two experiments:

$$p(x_0) = p^{1-0}.q^0 = p$$

and for the other one:

$$p(x_1) = p^{1-1}.q^1 = q$$

Thus, the total probability is:

$$p(2) = p.p.q$$

Following this construction law, one more example is the probability for 17 desired events in 21 observations:

$$p(17) = p^{17}.q^4$$

From these reflections, it can be deduced that the general expression of the probability function that gives the probability of obtaining k desired observations in n observations is:

$$p(k) = p^k.q^{n-k}$$

And with this result, it seems that we already have the binomial function, but we still have to do one more reflection and that is to include all the possible placements of probabilities that allow us to obtain the same result, that is, if we return to the example seen above in which we were looking for the probability of obtaining two desired observations in three experiments and we saw that it was:

$$p(2) = p.p.q$$

but in this case, it had been assumed that the first two observations were the desired ones and the third not, but the permutation of observations that give us the desired could have been another, for example, having obtained first the unwanted one and then the two desired ones:

$$p(2) = q.p.p$$

or first the desired one, then the unwanted one and then the desired one again.

$$p(2) = p.q.p$$

In other words, there would be three placements that give us the desired joint observation.

That is, we have permutations with repetition. To calculate the permutations with repetition, the following equation is used:

$$P_{\alpha,\beta,\ldots,\kappa} = \frac{n!}{\alpha!\beta!\ldots\kappa!}$$

As in this case, we only have two different sets of elements, p that is repeated k times and q that is repeated (n–k) times (it would be ppppqqqq ... ppqqqq, or another permutation), the equation remains:

$$P_{pq} = \frac{n!}{k!(n-k)!}$$

Equation that matches the definition of the combinatorial number:

$$\binom{n}{k}$$

In the case that we have used to introduce the concept, we had two desired observations in three possible ones, so the placements were as follows:

$$\binom{n}{k} = \binom{3}{2} = \frac{3!}{2!1!} = 3!$$

Thus, the final version of the binomial function is:

$$p(k) = \binom{n}{k} p^k q^{n-k}$$

The Binomial probability function allows statistical studies such as how many elements of the desired category will be observed when performing *n* experiments.

We are now going to see some examples of binomial distributions, and we are going to start with the binomial distribution linked to the famous De Mere problem: What is more likely: to get at least a six in 4 tosses given or get a double six in 24 rolls of a die? The first thing we do is establish the probability of observing a six, which is 1/6 = 0.17, and its complement, which is 5/6 = 0.83, and then we construct the binomial probability function:

$$p(k) = \binom{n}{k} p^k q^{n-k} = \binom{n}{k} 0.17^k 0.83^{n-k}$$

From this function, we answer De Mere's first question: What is the probability of getting at least one six in four tosses of a die?

The probability of obtaining a six in four rolls is obtained from the binomial probability function:

$$p(k) = \binom{4}{1} 0.17^1 0.83^{4-1} = 0.388$$

Since it is at least one six, we also have to calculate the probability of obtaining 2, 3, and 4 sixes, and we do it in the same way:

$$p(2) = \binom{4}{2} 0.17^2 0.83^{4-2} = 0.116$$

$$p(3) = \binom{4}{3} 0.17^3 0.83^{4-3} = 0.015$$

$$p(4) = \binom{4}{4} 0.17^4 0.83^{4-4} = 0.0007$$

As there is no intersection between the events, the probability of obtaining any of them, that is, their union, is the sum of their probabilities:

$$p(1 \cup 2 \cup 3 \cup 4) = 0.388 + 0.116 + 0.015 + 0.0007 = 0.519$$

Once the first question has been answered, we are going to see the solution to the second: What is the probability of obtaining at least one double six in 24 rolls of two dice?

As it would be very long to solve it by the direct probability that they ask us, we are going to solve it by calculating the probability of the complementary event, which is what is the probability of not getting a double six in 24 rolls of two dice?

The first thing we do is define the binomial function for this experiment. The probability that we have to define in the function is that of obtaining a double six on a roll of two dice and its complementary. Applying the equation for obtaining the probability is:

$$p(\text{DoubleSix}) = \frac{n_{\text{DoubleSix}}}{n_{\text{Total}}} = \frac{1}{36} = 0.028$$

We remember because it was already seen in a previous exercise that the total number of possible results in the roll of two dice is 36.

Consequently, the binomial function in this case is:

$$p(k) = \binom{n}{k} p^k q^{n-k} = \binom{n}{k} 0.028^k 0.0972^{n-k}$$

Therefore, the probability of not rolling any double six in 24 rolls is:

$$p(0) = \binom{24}{0} 0.028^0 0.0972^{24} = 0.505$$

And consequently the probability of obtaining at least a double six is:

$$1 - p(0) = 1 - 0.505 = 0.495$$

and with this solution, we can already answer De Mere's problem and say that it is more likely to get a six in 4 tosses of a die than a double six in 24 tosses of a die.

Geometric Probability Distribution

The distribution, or Geometric probability function, is another distribution based on Bernoulli's, which, like the binomial distribution, extends the sample space used to the set formed by all natural numbers plus zero, $\mathbb{N}$, but as in the case of Bernoulli and Binomial, the observed characteristic can only be classified into two categories, and the probabilities of appearance of the two categories are complementary and constant, that is, the probability of appearance of category A is p, and the probability of appearance of category B is q; and $q = 1 - p$. The Geometric probability function allows statistical studies of the type, which is the probability of observing r correct elements until observing a failed element.

The definition of the probability function is:

$$p(x_i) = p(x = x_i = r; \{x_i = 0, 1, 2, \ldots, r, \ldots\}) = p.q^{r-1}$$

As an example of applying the Geometric distribution function, we are going to continue using the roll of a die, and we are going to solve the question, what is the probability of rolling a die ten times without obtaining a six double to the eleventh?

As we know from the previous examples, the probability of obtaining a double six is $p = \frac{1}{6} = 0.16$. We have called p the probability of obtaining a double six because it would be the "failed" element, since we are going to calculate how many times, we would get other "correct" results before obtaining the double six, and

consequently the probability of not obtaining it is $q = 1 - 0.16 = 0.83 = 0.16$ so the geometric probability function will have the form:

$$p(r) = p \cdot q^{r-1} = 0.16 \cdot 0.83^{10} = 0.024 \equiv 2.4\%$$

As you can see, it is very difficult to make ten rolls of a die without first having rolled a double six, and there is a 98% probability that it will not pass.

Poisson Probability Distribution

The sample space of the Poisson probability distribution, or function, is made up of all natural numbers plus zero, $\mathbb{N}$, and the observed characteristic can also have any value pertaining to the numbers natural.

The definition of the probability function is:

$$p(x_i) = p(x = x_i = r; \{x_i = 0, 1, 2, \ldots, r, \ldots\}) = \frac{\lambda^r}{r! e^{-\lambda}}$$

The Poisson probability function allows us to carry out statistical studies of the type that is the probability of the appearance of r elements of the population in an interval of duration, length or quantity or another fixed magnitude. The process is stable, that is, the average number of events per unit of time, length, quantity, etc., is constant.

As an example of applying the Poisson probability distribution, we will solve the following example.

An average of 3 cars arrive at a gas station per minute. Determine:

(a) What is the probability function?
(b) Probability that 2 cars will arrive in one minute.
(c) Probability that 12 cars will arrive in five minutes.

It is a process that quantifies the number n of elements of the population that are observed in an interval of fixed duration.

Since we observe 1 minute and there is an average of 3 arrivals in 1 minute, we are looking for average numbers of arrivals in different intervals.

(a) Consequently, the probability function is a Poisson function with $\lambda = 3$:

$$p(r) = P(x = r) = \frac{3^r}{r!} e^{-3}$$

(b) The probability that 2 cars will arrive in one minute is calculated as follows:

$$p(2) = P(x = 2) = \frac{3^2}{2!} e^{-3} = 0{,}224 \rightarrow 22{,}4\%$$

(c) The probability that 12 cars will arrive in five minutes is calculated as follows:

$$\lambda' = \lambda \cdot 5 = 3 \cdot 5 = 15$$

$$p(12) = \frac{15^{12}}{12!} e^{-15} = 0,0828 \rightarrow 8,28\%$$

Continuous Probability Distributions

In the previous subsection, we studied the probability functions, that is, the distribution functions associated with discrete random variables, and in this subsection, we studied the density functions, that is, the distribution functions associated with continuous random variables. Before starting, let us remember its general equation and that of its associated distribution function:

Density function:

$$p(x_i) = f(x = x_i)$$

Distribution function:

$$F(x_k) = p(x \leq x_k) = \int_{-\infty}^{k} x_i dx$$

If x_n is the largest value in the sample space,

$$F(x_n) = \int_{-\infty}^{+\infty} x dx = 1$$

We will begin by studying the most widely used distribution, which is the Normal distribution, and then we will discuss three other basic functions for use in statistics, t of Students, Pearson's chi-square, and F of Fisher.

Normal Distribution

The sample space of the Normal density distribution, or function, is made up of all real numbers plus zero, $\mathbb{R}$, and the observed characteristic can also have any value pertaining to real numbers.

The definition of the density function is:

$$p(x_i) = f(x = x_i) = \frac{1}{\sigma_x \sqrt{2\pi}} \exp\left(-\frac{1}{2\sigma_x^2} (x_i - \mu_x)^2 \right)$$

but to be able to treat it more easily you can change the variable to z:

$$z = \frac{x - \mu_x}{\sigma_x}$$

with the following density function:

$$p(z_i) = f(z = z_i) = \frac{1}{\sigma_x \sqrt{2\pi}} \exp\left(-\frac{z_i^2}{2}\right)$$

Furthermore, the distribution function of z is tabulated.

The normal density function allows statistical studies to be carried out when the results of an experiment are due to a very large set of independent causes, which act by adding their effects, each individual effect being of little importance with respect to the set, since in those cases it is expected that the probability of appearance of the values of the characteristic, or random variable, studied in the experiment will follow a normal distribution.

The normal density function has at least three derived density functions: Pearson's Chi-square, Student's t, and Fisher's F. They are considered functions derived from the normal distribution because the construction of the corresponding variables requires at least one normal distribution.

As an example of applying the Normal probability distribution, we will solve the following example.

It is known that, due to the filling processes, the content of a 33 cl is not exactly 33 cl in all cans but is normally distributed with a mean of 33 cl and a standard deviation of 2 cl. Determine:

(a) The density function.
(b) What is the probability that the content of a can is greater than 35 cl?
(c) If a pack consists of 6 cans, what is the probability that the content is less than 192 cl?

The statement indicates that the values of the variable content of the can are normally distributed with a deviation of 2 cl about an average of 33 cl.

(a) Consequently, the density function is a Normal (33, 2):

$$p(x_i) = f(x = x_i) = \frac{1}{2\sqrt{2\pi}} \exp\left(-\frac{1}{8}(x_i - 33)^2\right)$$

(b) The content of a can be greater than 35 cl:

$$p(x \geq 35) = p\left(z \geq \frac{35 - 33}{2}\right) = 1 - p(z \leq 1) = 1 - 0,8413 = 0,1587 \rightarrow 15,87\%$$

(c) Content of 6 cans less than 192 cl:

Central Limit Theorem: The sum of n variables that are normally distributed μ, σ, also has a normal distribution with mean $n \cdot \mu$ and standard deviation $\sqrt{n} \cdot \sigma$

$$p\left(z < \frac{192 - 6.33}{\sqrt{6}.2}\right) = p(z < -1,22) = p(z > 1,22) = 1 - p(z < 1,22)$$
$$= 1 - 0,8888 = 11,12\%$$

Pearson Chi-Squared Distribution

If we have n independent random variables $x_1 \ldots x_n$ and each one of them has a normal density function with mean 0 and standard deviation 1, N (0, 1) and is defined as the random variable $x = x_1^2 + \ldots + x_n^2$, x has a Pearson Chi-square density function. The definition of Pearson's Chi-square density function is:

$$f(x) = p(x) = f(x; x \in \mathbb{R})$$
$$f(x) = \begin{cases} \frac{e^{-\frac{x}{2}} x^{\frac{n}{2}-1}}{2^{\frac{n}{2}} \Gamma\left(\frac{n}{2}\right)} si\ x \geq 0 \\ 0\ si\ x < 0 \end{cases}$$

with n degrees of freedom. The sample space of the Chi-square distribution is made up of all the real numbers plus zero, $\mathbb{R}$, and the observed characteristic can also have any value belonging to the real numbers.

In addition, its distribution function is tabulated.

As an example of applying the Pearson Chi-Squared probability distribution, we will solve the following example.

If a variable is distributed in χ^2 with 7 degrees of freedom, obtain:

(a) $p\left(u > \chi_7^2\right) = 0.025$
(b) $p\left(u \leq \chi_7^2\right) = 0.5$
(c) $p\left(\chi_7^2 \leq u \leq \chi_7^2\right) = 0.9$

(a) $p\left(u > \chi_7^2\right) = 0.025 \rightarrow p\left(u \leq \chi_7^2\right) = 0.975 \rightarrow u = 16.01$
(b) $p\left(u \leq \chi_7^2\right) = 0.5 \rightarrow u = 6.35$
(c) $p\left(\chi_7^2 \leq u \leq \chi_7^2\right) = 0.9 \rightarrow p\left(u \leq \chi_7^2\right) = 0.05 \rightarrow u = 2.17$
$p\left(u \leq \chi_7^2\right) = 0.95 \rightarrow u = 14.07$

t-Student Distribution

If you have 2 independent random variables x_1, x_2, x_1 with a normal density function with mean 0 and standard deviation 1, N (0, 1), and x_2 distributed according to a Chi-square function with n Pearson degrees of freedom, χ_n^2, and defined from them the random variable:

$$t = \frac{x}{\sqrt{\frac{\Sigma}{n}}},$$

t has a density function t of student. The definition of the density function t of student is:

$$f(x) = \frac{\Gamma\left(\frac{n+1}{2}\right)}{\Gamma\left(\frac{n}{2}\right)} \cdot \frac{\left(1 + \frac{t^2}{n}\right)^{\frac{n+1}{2}}}{\sqrt{n\pi}} \forall t \in \mathbb{R}$$

with n degrees of freedom. The sample space of the Chi-square distribution is made up of all the real numbers plus zero, $\mathbb{R}$, and the observed characteristic can also have any value belonging to the real numbers.

In addition, its distribution function is tabulated.

As an example of applying the *t*-Student probability distribution, we are going to solve the following example.

If a variable is distributed in *t*-Student with 9 degrees of freedom, obtain:

(a) $p(t \leq t_1) = 0.95$
(b) $p(t > t_1) = 0.025$
(c) $p(t \leq t_1) = 0.995$
(d) $p(t \leq t_1) = 0.9$

(a) $p(t \leq t_1) = 0.95 \rightarrow t_1 = t_9(0.95) = 1.833$
(b) $p(t > t_1) = 0.025 \rightarrow p(t > t_1) = 0.975 \rightarrow t_1 = t_9(0.975) = 2.262$
(c) $p(t \leq t_1) = 0.995 \rightarrow t_1 = t_9(0.995) = 3.250$
(d) $p(t \leq t_1) = 0.9 \rightarrow t_1 = t_9(0.995) = 1.383$

F of Fisher Distribution

If we have 2 independent random variables x_1, x_2, distributed according to a Chi-square function with n Pearson, x_1 with m degrees of freedom, χ_m^2, and x_2 with n degrees of freedom, χ_n^2. It is defined the random variable:

$$F = \frac{\frac{x}{m}}{\frac{y}{n}} = \frac{xn}{ym}$$

It has a function of Fisher's density with m degrees of freedom in the numerator and n degrees of freedom in the denominator, $F_{m,\,n}$ defined as:

$$f(x) = \begin{cases} \dfrac{\Gamma\left(\dfrac{m+n}{2}\right)}{\Gamma\left(\dfrac{m}{2}\right)\Gamma\left(\dfrac{n}{2}\right)} \left(\dfrac{m}{n}\right)^{\frac{m}{2}} \dfrac{F^{\frac{m}{2}-1}}{\left(1+\frac{m}{n}F\right)^{\frac{m+n}{2}}} si\ F \geq 0 \\ 0\ si\ F < 0 \end{cases}$$

The sample space of the Chi-square distribution is made up of all the real numbers plus zero, $\mathbb{R}$, and the observed characteristic can also have any value belonging to the real numbers.

In addition, its distribution function is tabulated.

As an example of applying the Fisher probability distribution, we will solve the following example.

If a variable is distributed in F of Fisher with 3 degrees of freedom in the numerator and 4 degrees of freedom in the denominator, obtain:

(a) $p(F \leq F_{3,\,4}) = 0.975$
(b) $p(F \leq F_{3,\,4}) = 0.25$

(a) $p(F \leq F_{3,\,4}) = 0.975 \rightarrow F_{3,\,4}(0.975) = 9.9792$
(b) $p(F \leq F_{3,4}) = 0.25 \rightarrow F_{3,4}(0.250) = \frac{1}{F_{4,3}(0.750)} = \frac{1}{2.3901} = 0.4184$

B. Computer-Based Solving

Probability Exercises solved in R

In this subsection, the R environment will be used to solve the same cases that had been solved theoretically in the previous subsection. As many of the theoretical contents are only calculated by arithmetic calculations, we start with the examples in discrete probability distributions.

1. What is more likely: to get at least a six in 4 tosses given or get at least a double six in 24 rolls of a die?

Case 1: get at least a six in 4 tosses.

The probability that a number 6 will come up is 1/6, so p=1/6, the number of tests to be carried out or attempts, that is, size is 4, and the number of successes or values for which we want to know the probability is 1. We set them into the dbinom() function and obtain:

dbinom(x, size, prob) = dbinom(1,4,1/6) = 0.388

Since it is at least one six, we also have to calculate the probability of obtaining 2, 3, and 4 sixes, and we do it in the same way:

dbinom(x, size, prob) = dbinom(2,4,1/6) = 0.116
dbinom(x, size, prob) = dbinom(3,4,1/6) = 0.015
dbinom(x, size, prob) = dbinom(4,4,1/6) = 0.0007

As there is no intersection between the events, the probability of obtaining any of them, that is, their union, is the sum of their probabilities: 0.519

Case 2: get at least a double six in 24 tosses.

We are going to solve it by calculating the probability of the complementary event, which is what is the probability of not getting a double six in 24 rolls of two dice?

The probability of obtaining a double six on a roll of two dice is 1/36, so p = 1/36. The number of tests to be carried out or attempts is 24, and the number of successes is 0. Therefore, the probability of not rolling any double six in 24 rolls is:

dbinom(x, size, prob) = dbinom(0, 24,1/36) = 0.505

Consequently, the probability of obtaining at least a double six is 0.495.

With this solution, we can already say that it is more likely to get a six in 4 tosses of a die than a double six in 24 tosses of a die.

2. An average of 3 cars arrive at a gas station per minute. Determine:

   (a) Probability that 2 cars will arrive in one minute.
   (b) Probability that 12 cars will arrive in five minutes.

   (a) The probability that 2 cars will arrive in one minute.

   ```
   > dpois(2,3)
   [1] 0.224
   ```

   (b) The probability that 12 cars will arrive in five minutes.

   ```
   > dpois(12,15)
   [1] 0.0828
   ```

3. It is known that, due to the filling processes, the content of a 33 cl it is not exactly 33 cl in all cans, but is normally distributed with a mean of 33cl and a standard deviation of 2cl. Determine:

   (a) What is the probability that the content of a can is greater than 35 cl?
   (b) If a pack consists of 6 cans, what is the probability that the content is less than 192 cl?

   The statement indicates that the values of the variable content of the can are normally distributed with a deviation of 2 cl about an average of 33 cl.

(a) The content of a can be greater than 35 cl:

> 1-pnorm(35, mean = 33, sd = 2)
> 1-pnorm((35-33)/2)
[1] 0.1587

(b) Content of 6 cans less than 192 cl:

Central Limit Theorem: The sum of n variables that are normally distributed μ, σ, also has a normal distribution with mean $n \cdot \mu$ and standard deviation $\sqrt{n} \cdot \sigma$

> 1-pnorm(192, mean = 33·6, sd = $\sqrt{6}$·2)
> 1-pnorm((192–198) / ($\sqrt{6}$·2))
[1] 0.1112

4. What is the probability of rolling a die ten times without obtaining a six double to the eleventh?

The probability of obtaining a double six is $p = \frac{1}{6} = 0.16$. We have called p the probability of obtaining a double six because it would be the "failed" element, since we are going to calculate how many times, we would get other "correct" results before obtaining the double six, and consequently the probability of not obtaining it is $q = 1 - 0.16 = 0.83 = 0.16$.

>dgeom(10,0.16)
[1] 0.024

5. If a variable is distributed in χ^2 with 7 degrees of freedom, obtain:

(a) $p\left(u > \chi_7^2\right) = 0.025$
(b) $p\left(u \leq \chi_7^2\right) = 0.5$

(a) $p\left(u > \chi_7^2\right) = 0.025$
qchisq(1-0.025,7)
[1] 16.01276
(b) $p\left(u \leq \chi_7^2\right) = 0.5$
qchisq(0.5,7)
[1] 6.345811

6. If a variable is distributed in t-Student with 9 degrees of freedom, obtain:

(a) $p(t \leq t_1) = 0.95$
(b) $p(t > t_1) = 0.025$
(c) $p(t \leq t_1) = 0.995$
(d) $p(t \leq t_1) = 0.9$

(a) $p(t \leq t_1) = 0.95$
qt(0.95,9)
[1] 1.833113

(b) $p(t > t_1) = 0.025$
qt(1-0.025,9)
[1] 2.262157
(c) $p(t \leq t_1) = 0.995$
qt(0.995,9)
[1] 3.249836
(d) $p(t \leq t_1) = 0.9$
qt(0.9,9)
[1] 1.383029

7. If a variable is distributed in F of Fisher with 3 degrees of freedom in the numerator and 4 degrees of freedom in the denominator, obtain:

(a) $p(F \leq F_{3,\,4}) = 0.975$
(b) $p(F \leq F_{3,\,4}) = 0.25$

(a) $p(F \leq F_{3,\,4}) = 0.975$
qf(0.975,3,4)
[1] 9.979199
(b) $p(F \leq F_{3,\,4}) = 0.25$
qf(0.25,3,4)
[1] 0.418391

C. Probability Exercises Solved

This subsection has two parts. In the first part, a set of exercises solved in detail are presented to allow you to check if all the knowledge has been correctly acquired. The advice is to try to solve the exercises by yourself, and then to get the solution to check it with the proposed one by the book. This procedure will make this subsection truly useful for you. In the second part, the same exercises will be solved in R.

Hand-Made Exercises

1. There are two urns with 10 balls. In urn A, there are 4 white balls and 1 black ball. In urn B, there are 2 white balls and 3 black balls. Calculate the probability of choosing randomly:

(a) A ball from urn A
(b) A ball form urn B
(c) A white ball
(d) A black ball

Applying the definition of probability, we have the following solutions:

(a) $1.P(A) = \frac{n_A}{N} = \frac{5}{10} = 0.5$
(b) $2.P(B) = \frac{n_B}{N} = \frac{5}{10} = 0.5$
(c) $3.P(\text{white}) = \frac{n_{\text{white}}}{N} = \frac{6}{10} = 0.6$
(d) $4.P(\text{black}) = \frac{n_{\text{black}}}{N} = \frac{4}{10} = 0.4$

2. With the same urns as in the previous exercise, verify that all the probabilities calculated in exercise 1 accomplish the probability property. $0 \leq P(A) \leq 1$

$$P(A) = 0.5; P(B) = 0.5; P(\text{white}) = 0.6; P(\text{black}) = 0.4$$

3. With the same urns as in the previous exercises, verify that the probabilities to choose ball from both urns are complementary, and that choose white or black ball are also complementary ¿Are complementary the possibilities to be a ball form urn A and black?

   From the property of probability $P(\overline{A}) = 1 - P(A)$

   That means that $P(A) = 1 - P(B) = 0.5$ and $P(\text{white}) = 1 - P(\text{black}) = 0.6$

   In the case of the last question $(A) \neq 1 - P(\text{negra}) \rightarrow 0.5 \neq 1 - 0.4$. They are not complementary because don't belong to urn A doesn't mean that the balls are black, because in urn B there are black and white balls.

4. Calculate the probability of choosing randomly a. A white ball from urn A and b. A ball that was white or from urn A.

   First, we calculate question a and apply the definition of probability only to urn A.

$$\text{P}(\text{white } A) = \frac{n_{\text{white } A}}{N} = \frac{4}{10} = 0.4$$

   Next, we can apply the property of probability

$$P(A \cup B) = P(A) + P(B) - P(A \cap B)$$

   for solving the question b. that it is the probability of the union of two events, white ball and urn A, both of them are valid, because the or means union. The probability of the intersection of both events has been calculated in question a.

$$P(\text{white} \cup A) = P(\text{white}) + P(A) - P(\text{white} \cap A) \rightarrow 0.6 + 0.5 - 0.4 = 0.7$$

5. Verify that the probability of choosing a white ball from urn A is lower than choosing a ball from urn A.

   From the property of probability, if $\text{A} \subset B = P(A) \leq P(B)$ the statement of the exercise must be accomplished.

$$P(\text{white } A) = 0.4 \leq P(A) = 0.5$$

6. Calculate the probability that any of the balls was a. white or black; and b. that is not from any urn.

   From the properties of probability, as black and white are or the sample space: $P(E) = 1$. How all the balls belong to one of the two urns: $P(\varnothing)$

7. Now, we will present an exercise of conditioned probability with the roll of a dice. In this case, since there must be two events, although, as indicated at the beginning of the topic, the roll of dice will be kept as the domain in which the examples will be carried out, the roll of a dice cannot be used but at least you have to roll two dice. Consequently, the exercise will consist of calculating the probability of obtaining in two consecutive throws of a die the same result in both dice in the second toss, event A, and an odd number in the first toss, event B.

   To solve the exercise, we first define the possible cases or sample space of the roll of two dice: E = {{1,1}, {1,2}, {1, 3}, {1.4}, {1.5}, {1.6}, {2.1}, {2.2}, {2.3}, {2.4}, {2.5}, {2.6}, {3.1}, {3.2}, {3.3}, {3.4}, {3.5}, {3.6}, {4.1}, {4.2}, {4.3}, {4.4}, {4.5}, {4.6}, {5.1}, {5.2}, {5.3}, {5.4}, {5.5}, {5.6}, {6.1}, {6.2}, {6.3}, {6.4}, {6.5}, {6.6}}. That is, it is composed of 36 possible cases.

   To solve the exercise, we apply the equation of the probability of two events

$$p(A|B) = \frac{p(A \cap B)}{P(B)}$$

   We start by obtaining the probability of B, for which we use the classical probability equation:

$$p(B) = \frac{n_B}{n_T}$$

   where $n_T = 36$. The number of favourable cases will be those in which an odd number was obtained in the first roll, which if we take the sample space will be {{1,1}, {1,2}, {1,3}, {1.4}, {1.5}, {1.6}, {3.1}, {3.2}, {3.3}, {3.4}, {3.5}, {3.6}, {5.1}, {5.2}, {5.3}, {5.4}, {5.5}, {5.6}}. That is, there are 18 favourable cases, $n_B = 18$.

   Therefore, the probability of obtaining event B is:

$$p(B) = \frac{n_B}{n_T} = \frac{18}{36} = 0.5$$

   Next, we calculate $p(A \cap B)$, for which, to obtain the intersection of both events, in the previous set, we select only those in elements in which the values coincide, and we obtain the set {{1.1}, {3.3}, {5.5}}. That is, there are 3 favourable cases, $n_{A \cap B} = 3$. Thus, the probability is:

$$p(A \cap B) = \frac{n_{A \cap B}}{n_T} = \frac{3}{36} = 0.083$$

Consequently, the requested probability of obtaining two equal numbers on the second roll of a die when an odd number has been obtained on the first roll is:

$$p(A|B) = \frac{p(A \cap B)}{P(B)} = \frac{0.083}{0.5} = 0.166 \equiv 16.6\%$$

8. We have the following data:

- The probability that someone was ill with a specific illness is 0,001, that is, 1 person in 1000.
- The probability that someone who was ill of that illness has a positive test of the illness is 99%, that is, 0,99.
- The probability that someone who was not ill of that illness has a positive test is 2%, that is, 0,02.

With these data, the probability that someone who had obtained a positive test of the illness was truly ill of the illness must be obtained.

How it is a problem in which we know that someone has obtained a positive test of the illness we are in *A Posteriori* probability problem and we apply the Bayes theorem to solve it.

Probability of being ill of the illness: $P(A_1)$= 0,001
The probability of not being ill with the illness is complementary to the previous probability: $P(A_2)$= 0,999
Conditional probability of being ill if a positive test of the illness is obtained: $(B|A_1)$= 0,99
Conditional probability to not be ill if a positive test of the illness is obtained: $P(B|A_2)$= 0,02

Consequently, if we apply the equation of calculation of the Bayes probability, we obtain:

$$\begin{aligned} P(A_1|B) &= \frac{P(B|A_1)P(A_1)}{\sum_{i=1}^{2} P(B|A_i)P(A_i)} \\ &= \frac{P(B|A_1)P(A_1)}{P(B|A_1)P(A_1) + P(B|A_2)P(A_2)} \\ &= \frac{0,99.0,001}{0,99.0,001 + 0,02.0,999} = \frac{0,00099}{0,02097} = 0.047 \cong 5\% \end{aligned}$$

which is very surprising from the statement of problem.

9. We have the following probability density function:

$$f(x)=\begin{cases} m\cdot\left(1-x^4\right) & \text{if } 0<x<1 \\ 0 & \text{otherwise} \end{cases}$$

(a) Calculate m.
(b) Calculate the mean and the variance.

(a)

$$\int_{-\infty}^{\infty} f(x)dx=1\rightarrow \int_{0}^{1} m\left(1-x^4\right)dx=1\rightarrow mx\Big|_0^1-\frac{mx^5}{5}\Big|_0^1=1$$

$$(m\cdot 1-m\cdot 0)-\left(m\cdot\frac{1^5}{5}-m\cdot\frac{0^5}{5}\right)=1$$

$$m-\frac{m}{5}=1\rightarrow m=\frac{5}{4}=1.25$$

(b) $\mu_x=E(x)=\int_{-\infty}^{\infty} x\cdot f(x)dx=\int_0^1 x\cdot 1.25\cdot\left(1-x^4\right)dx=1.25\cdot\left(\frac{x^2}{2}-\frac{x^6}{6}\right)\Big|_0^1$

$=0.4167$

$$\sigma_x^2=\int_{-\infty}^{\infty} x^2\cdot f(x)dx-\mu_x^2=\int_0^1 x^2\left(1.25\cdot\left(1-x^4\right)\right)dx-(0.4167)^2$$

$$=1.25\cdot\left(\frac{x^3}{3}-\frac{x^7}{7}\right)\Big|_0^1-(0.4167)^2=0.0645$$

10. The number of hours of normal operation without failure of software is 750 hours with a standard deviation of 8 hours. Calculate:

(a) The density function.
(b) The probability of operating at least 760 hours without failure.
(c) The probability to operate at most 748 hours without failure.
(d) The probability to run exactly 755 hours without failure (taking only one decimal place).

(a) The density function is a Normal (750, 8):

$$p(x_i)=f(x=x_i)=\frac{1}{8\sqrt{2\pi}}\exp\left(-\frac{1}{128}(x_i-750)^2\right)$$

(b) 760h or more without failures:

$$p(x\geq 760)=p\left(z\geq\frac{760-750}{8}\right)=1-p(z\leq 1,25)=1-0{,}894=0{,}105\rightarrow 10,5\%$$

(c) At most 748 hours without failure:

$$p(x \leq 748) = p\left(z \leq \frac{748-750}{8}\right) = p(z \leq -0,25) = 1 - p(z \leq 0,25)$$
$$= 1 - 0,598 \rightarrow 40,1\%$$

(d) Exactly 755 hours without failure (1 decimal):

$$p(x = 755) \cong p(754,5 \leq x \leq 755,5) = p(x \leq 755,5) - p(x \leq 754,5)$$
$$= p(z \leq 0,6875) - p(z \leq 0,5625) = 0,751 - 0,712 = 0,039 \rightarrow 3,9\%$$

11. The average number of visits to a website is 5 every minute. Determine:

(a) The probability function.
(b) The mean and the standard deviation.
(c) The probability that there are 17 visits in 3 minutes.
(d) The probability that there are no visitors in 1 second.

It is a process that quantifies the number n of elements of the population that are observed in an interval of fixed duration.

Since we observe 1 minute and there is an average of 5 visits, we are looking for the average number of visits in different intervals.

(a) Consequently, the probability function is a Poisson function with $\lambda = 5$:

$$p(r) = P(x = r) = \frac{5^r}{r!} e^{-5}$$

(b) The mean and the standard deviation:

$$\mu_x = \lambda = 5$$
$$\sigma_x = \sqrt{\lambda} = 2,24$$

(c) Probability of 17 visits in 3 minutes:

$$\lambda' = \lambda.3 = 5.3 = 15$$
$$p(17) = P(x = 17) = \frac{15^{17}}{17!} e^{-15} = 0,084 \rightarrow 8,43\%$$

(d) Probability of no visitors in 1 second:

$$\lambda'' = \lambda/60 = 5/60 = 0,083$$

$$p(0) = P(x = 0) = \frac{0{,}083^0}{0!} e^{-0{,}083} = 0.92 \rightarrow 92\%$$

12. In a quality control, 20 components manufactured by a machine are observed, assigning each one a 0 if it is correct and a 1 if it is defective. The results obtained are: 0, 0, 0, 1, 0; 0, 0, 0, 0, 0; 0, 0, 0, 0, 0; 0, 0, 0, 1, 0. If this is extended to the entire population:

    (a) What is the Geometric probability function?
    (b) What is the probability of getting 9 correct components before the first defective?
    (c) What is the probability of getting 19 correct components before the first defective?

    (a) The Geometric probability function is:

$$p(x) = p(1-p)^{x-1} = pq^{x-1} = 0,1(0,9)^{x-1} \; x = 0, 1, \ldots, n$$

    (b) We replace x with 10, because there are 9 observations:

$$p(10) = 0,1(0,9)^9 = 0,0387 \rightarrow 3,9\%$$

    (c) We replace x with 20, because there are 19 observations:

$$p(20) = 0,1(0,9)^{19} = 0,0135 \rightarrow 1,35\%$$

13. A company applies a discount on any invoice that is paid within 30 days of its issuance. Of all invoices, 10% received the discount. In an audit of the company, 12 invoices were randomly selected. What is the probability that, of the 12 invoices, less than 4 have a discount?

    In this case, we have to use the binomial probability function:

$$p(n/m) = \binom{m}{n} p^n (1-p)^{m-n} = \frac{m!}{n!(m-n)!} p^n (1-p)^{m-n}$$

$$p(0/12) = \binom{12}{0} \left(\frac{1}{10}\right)^0 \left(\frac{9}{10}\right)^{12} = 0,2824$$

$$p(1/12) = \binom{12}{1} \left(\frac{1}{10}\right)^1 \left(\frac{9}{10}\right)^{11} = 0,3765$$

$$p(2/12) = \binom{12}{2} \left(\frac{1}{10}\right)^2 \left(\frac{9}{10}\right)^{10} = 0,2301$$

$$p(3/12) = \binom{12}{3} \left(\frac{1}{10}\right)^3 \left(\frac{9}{10}\right)^9 = 0{,}085$$

    If we sum the values, we obtain 0,974 = 97,4%.

14. A die is tossed eight times. What is the probability that 2 number 6 s come up?
In this case, we have to use the binomial probability function:

$$p(n/m) = \binom{m}{n} p^n (1-p)^{m-n} = \frac{m!}{n!(m-n)!} p^n (1-p)^{m-n}$$

$$p(2/8) = \binom{8}{2} \left(\frac{1}{6}\right)^2 \left(\frac{5}{6}\right)^{8-2} = \frac{8!}{2!6!} \left(\frac{1}{6}\right)^2 \left(\frac{5}{6}\right)^6 = 0,2625 \rightarrow 26,25\%$$

15. What is the Bernoulli probability function of success for a die roll if only a 6 is considered successful?

$$p(x) = \left(\frac{1}{6}\right)^x \left(\frac{5}{6}\right)^{1-x} \text{ for } x = 0, 1 \rightarrow p(1 \equiv \text{Success}) = \frac{1}{6}; p(0 \equiv \text{Failure}) = \frac{5}{6}$$

Exercises Solved in R

In this subsection, the previous exercises will be solved using the R software. In addition, as we did in the section B, we are going to solve only problems after probability distributions, that means, since exercise 9 in the hand-made exercises.

9. We have the following probability density function:

$$f(x) = \begin{cases} m \cdot \left(1 - x^4\right) & \text{if } 0 < x < 1 \\ 0 & \text{otherwise} \end{cases}$$

(a) Calculate m.

We have to check that $\int_{-\infty}^{\infty} f(x)dx = 1$

To solve this type of integral with R, we have to perform a two-step process:

First, we define the function that we want to integrate. In this case, we are going to give it the name "f", so the instruction is:

```
f <- function(x) {1-x^4}
```

Second, we use the integrate() function, for which there is no additional package to load because it is in the stat package. The integrate() function has the arguments (function, lower bound, upper bound), so in this case, it becomes:

```
integrate(f,0,1)
```

The solution would be, clearing m in the equation, that m is equal to 1 divided by the result of the integral:

```
1/0.8
[1] 1.25
```

(b) Calculate the mean and the variance.

The function to calculate the mean is:

$$\mu_x = E(x) = \int_{-\infty}^{\infty} x.f(x)dx$$

First, we have to define the function:

```
mf <- function(x) {1.25*(x-x^5)}
```

Then, we apply the integrate function:

```
integrate(mf,0,1)
[1] 0.4166667
```

The function to calculate the variance is:

$$\sigma_x^2 = \int_{-\infty}^{\infty} x^2.f(x)dx - \mu_x^2$$

Following the same steps, we first have to define the function:

```
vf <- function(x) {1.25*(x^2-x^6)}
```

Then, we apply the integrate function:

```
integrate(vf,0,1)
[1] 0.2380952
```

Finally, the variance is:

```
0.2380952-0.416667^2
[1] 0.06448381
```

10. The number of hours of normal operation without failure of software is 750 hours with a standard deviation of 8 hours. Calculate:

(a) The probability of operating at least 760 hours without failure.

```
1-pnorm(760, mean = 750, sd = 8)
1-pnorm((760-750)/8)
[1] 0.1056498
```

(b) The probability to operate at most 748 hours without failure.

```
> pnorm(748, mean = 750, sd = 8)
> pnorm((748-750)/8)
[1] 0.4012937
```

(c) The probability to run exactly 755 hours without failure.

> dnorm(755, mean = 750, sd = 8)
[1] 0.04102012

11. The average number of visits to a website is 5 every minute. Determine:

(a) The probability that there are 17 visits in 3 minutes.

> dpois(17,15)
[1] 0.08473555

(b) The probability that there are no visitors in 1 second.

>dpois(0,5/60)
[1] 0.9200444

12. In a quality control, 20 components manufactured by a machine are observed, assigning each one a 0 if it is correct and a 1 if it is defective. The results obtained are as follows: 0, 0, 0, 1, 0; 0, 0, 0, 0, 0; 0, 0, 0, 0, 0; 0, 0, 0, 1, 0. If this is extended to the entire population:

(a) What is the Geometric probability function?

The Geometric function is covered in the R stats package so there is no additional package to load. We can work with the Geometric function through different functions, but the main one is:

dgeom (x, prob)

(b) What is the probability of getting 9 correct components before the first defective?

>dgeom(10,0.1)
[1] 0.0348

(c) What is the probability of getting 19 correct components before the first defective?

>dgeom(20,0.1)
[1] 0.0121

As it can be seen, the results are not the same as in the handmade exercises subsection. The reason is that the function implemented in R is:

$$p(n) = P(x = n) = p(1-p)^n; con\ n = 1, \ldots$$

where n is the number of correct elements observed without including the defective one. Therefore, the correct solutions are:

> dgeom(9,0.1)
[1] 0.0387

```
> dgeom(19,0.1)
[1] 0.0135
```

13. A company applies a discount on any invoice that is paid within 30 days of its issuance. Of all invoices, 10% received the discount. In an audit of the company, 12 invoices were randomly selected. What is the probability that, of the 12 invoices, less than 4 have a discount?

    The Binomial function is covered in the R stats package so there is no additional package to load. We can work with the Binomial function through different functions, of which we will use two:

    – The function dbinom (x, size, prob), whose arguments are: x, value of which we want to know the probability; size, number of tests to perform; and prob, value of the probability of success.
    – And the function pbinom(x, size, prob). It gives us the distribution function, that is, the sum of the probabilities of the values less than or equal to x, with size, number of tests to be carried out; and prob, value of the probability of success.

    In this case, we are going to use both functions to check that we obtain the same result:

```
> dbinom(0,12,0.1)+dbinom(1,12,0.1)+dbinom(2,12,0.1)+dbinom(3,12,0.1)
[1] 0.9743625
> pbinom(3,12,0.1)
[1] 0.9743625
```

14. A die is tossed eight times. What is the probability that 2 number 6s come up?

    The probability that a number 6 will come up is 1/6, so p = 1/6, the number of tests to be carried out or attempts, that is, size is 8, and the number of successes or value of which we want to know the probability is 2. We set them into the dbinom() function and get:

    dbinom(x, size, prob)= dbinom(2,8,1/6) = 0.26

15. What is the Bernoulli probability function of success for a die roll if only a 6 is considered successful?

    dbern(x, 1/6)

Next, four additional exercises are solved using R.

1. Represent the Binomial function for the following cases B1 (0.3, 30); B2 (0.6, 30), and B3 (0.8, 100).

   To represent the binomial function, we will use the well-known function plot () and as an argument, either we do it through a variable B1 or we introduce the function that allows us to calculate the binomial function, which is: dbinom (), whose arguments are: x, the value of which we want to know the probability, size, and number of tests to perform; and prob, the value of the probability of success. For B1, we would have

>B1 = dbinom (x, 30, 0.3).

However, before calculating it, we will have to determine the value of x. To do this, we use the sequence function between 0 and 30, with a step equal to 1.

> x = seq (0, 30, by = 1).

Therefore, we can enter any of the following settings for the plot function:

>plot (B1)
>plot (dbinom (x, 30, 0.3))
> plot (dbinom (seq (0, 30, by = 1), 30, 0.3))

obtaining the same result. To obtain the result of the following two cases, we introduce the following:

>plot (dbinom (seq (0, 30, by = 1), 30, 0.6));
>plot (dbinom (seq (0, 100, by = 1), 100, 0.8)).

2. For each of the previous probability distribution functions:

(a) Calculate the probability that the variable takes the value 20.
(b) Calculate the probability that the variable takes a value equal to or less than 10.
(c) Calculate the probability that the variable takes a value greater than 25.

To perform the calculations in subsection (a), we use the dbinom () function again, putting the appropriate values for this calculation, which are:

>ab1 = dbinom (20, 30, 0.3)
> ab2 = dbinom (20, 30, 0.6)
> ab3 = dbinom (20, 100, 0.8).

To perform the calculations in subsection (b), we use a variant that is the pbinom () function that gives us the distribution function, and for each case, they are:

>bb1 = pbinom (10, 30, 0.3);
>bb2 = pbinom (10, 30, 0.6)
> bb3 = pbinom (10, 100, 0.8).

Finally, for subsection (c), we use the distribution function again and calculate the probability of the complementary event q = 1-p. In this case, it is:

>cb1 = 1-pbinom (25, 30, 0.3)
>cb2 = 1-pbinom (25, 30, 0.6)
>cb3 = 1-pbinom (25, 100, 0.8)

3. Obtain the normal function, with 50 randomly generated numbers with the following mean and standard deviations: (0, 0.5); (0, 1) and (0, 2).

To represent the normal function, we use the normal function dnorm (). The arguments of this function are x, value of which we want to know the probability,

mean, mean of the normal distribution; and sd, standard deviation. They ask us to obtain the value of x randomly, for which we use the rnorm () function. Its arguments are n, number of observations; mean and sd. Consequently, to solve the problem for the first pair (0, 0.5), we introduce the following instructions:

>x = rnorm (50, 0, 0.5)
>d1 = dnorm (x, 0, 0.5)
>plot (d1). It could also be written synthesized in a single expression:
>plot (x, dnorm (x, 0, 0.5))

For the other two pairs, we have:

>plot (x, dnorm (x, 0, 1))
>plot (x, dnorm (x, 0, 2))

We put them all in a single graph to compare them with the instruction:

>op = even (mfrow = c (2, 2))

4. For each of these functions, calculate:

(a) The value of the probability function at point 0.2.
(b) The probability that the variable is greater than 1
To calculate the probability at point 0.2, we use the commands:

a1 = dnorm (0.2, 0, 0.5)
a2 = dnorm (0.2, 0, 1)
a3 = dnorm (0.2, 0, 2)

To calculate the probability that the variable is greater than 1, we use pnorm () and the property of the probability of the complementary event:

>b1 = 1-pnorm (1, 0, 0.5);
>b2 = 1-pnorm (1, 0, 1);
>b3 = 1-pnorm (1, 0, 2).

Annex: Probability Extended Concepts

In this section, of the chapter, more mathematically advanced concepts are introduced that will lead to the axiomatic definition of the Kolmogorov probability. When dealing with topics of greater mathematical complexity, it will also be in this section where concepts that have already been introduced in the previous sections are delved into, such as those related to set theory or demonstrations of the properties of probability.

Axiomatic Probability of Kolmogorov

There are two definitions that we need to go deeper into the probability concepts, the fifth axiom of the set theory and the *axiom of the power set*, which states that for any set S, there exists a set P such that $A \subset P$ if and only if $A \subseteq S$. From the fifth axiom of the set theory, since the set P is uniquely determined, the set of all subsets of S can be called the power set or sections of S, denoted by P(S), and this is the second definition that we must remember now. In the previous subsection, we saw that applied to a random experiment with sample space E, the set of parts of E can be defined as one whose elements are all possible subsets of E. In set theory, for any set A, the set of parts of A, P(A), is the set formed by all the subsets that can be formed with the elements of A. The set P(E) with the inclusion relation (P(E), $\subset$), is a partially ordered set, where E is the maximal and $\varnothing$ is the minimal. In the footnote,[9] we can read, to remember more easily, the rest of the contents introduced for the definition of parts of E and the example[10] introduced to understand the concept.

Once the set of parts of E has been defined, if set theory is still applied, it can be verified that the set powers or parts of a set S, or in our case E, with the operations union and intersection of sets and the definition of complementary set (P(S),$\cup$, $\cap$, —) has a *Boolean Algebra Structure*. The fact that P(E) has a Boolean Algebra structure leads to the verification of the duality principle, which says that a set, an event, can be expressed from others through the operations associated with a Boolean Algebra, that is, substituting the union for the intersection and vice versa

[9] Each element of P(E), that is, each subset of E, is one of the different events or results, both elementary and nonelementary, that can occur when performing the random experiment, that is, it is the set of all possible results that can be given when conducting the experiment, so P(E) is called the event space of the random experiment. From this it is logical to ask: how many are the elements of P(E)? The solution is calculated using combinatorics. If the cardinal of E is n, that is, the number of elementary events or elements of E is n, a set of n elements contains $\binom{n}{m}$ subsets of m elements each, since they are combinations of n elements taken from m to m. They are combinations and not variations because two subsets are different only if they have one or more different elements, not if the elements are listed in different order. Since the number of subsets of E would be the sum of all the possible subsets that could be formed with the elements of E, subsets that could have from 0 to n elements, the sum would be: $\binom{n}{0} + \binom{n}{1} + \ldots + \binom{n}{n-1} + \binom{n}{n} = 2^n$. So the number of elements of P(E) is 2^n.

[10] To see an example of the set Parts of E, we take the E of seen example of the E of the rolling of a die, that is {1,2,3,4,5,6}, starting for this, in roll of a die the set of parts of E is: P(E) = {$\varnothing$, 1, 2, 3, 4, 5, 6, {1,2 }, {1,3}, {1,4}, {1,5}, {1,6}, {2,3}, {2,4}, {2,5}, {2, 6}, {3.4}, {3.5}, {3.6}, {4.5}, {4.6}, {5.6}, {1,2,3}, {1,2,4}, {1,2,5}, {1,2,6}, {1,3,4}, {1,3,5}, {1,3,6}, {1,4,5}, {1,4,6}, {1,5,6}, {2,3,4}, {2,3,5}, {2,3,6}, {2,4,5}, {2,4,6}, {2,5,6}, {3,4,5}, {3,4,6}, {3,5,6}, {4,5,6}, {1,2,3,4}, {1,2,3,5}, {1,2,3,6}, {1,2,4,5}, {1,2,4,6}, {1,2,5,6}, {1,3,4,5}, {1,3,4,6}, {1,3,5,6}, {1,4,5,6}, {2,3,4,5}, {2,3,4,6}, {2,3,5,6}, {2,4,5,6}, {3,4,5,6}, {1,2,3,4,5}, {1,2,3,4,6}, {1,2,3,5,6}, {1,2,4,5,6}, {1,3,4,5,6}, {2,3,4,5,6}, {1,2,3,4,5,6}}. The Number of elements of parts of E on roll of a die is: 2 ^ n = 2 ^ 6 = 64.

and the certain event for the impossible, since for every event A, B, C ∈ P(S), the following five 5 properties are fulfilled:

- Commutative: $A \cup B = B \cup A$; $A \cap B = B \cap A$
- Associative: $A \cup (B \cup C) = (A \cup B) \cup C$; $A \cap (B \cap C) = (A \cap B) \cap C$
- Distributive: $A \cup (B \cap C) = (A \cup B) \cap (A \cup C)$; $A \cap (B \cup C) = (A \cap B) \cup (A \cap C)$
- Neutral element: $A \cup \emptyset = A$; $A \cap \emptyset = A$
- Complementary element: $A \cup \overline{A} = E$; $A \cap \overline{A} = \emptyset$

  From property 5, it follows that $\overline{E} = \emptyset$. Proof: $\overline{E} = \emptyset$. $E \cup \overline{E} = E$, The only possibility of this happening is $\overline{E} = \emptyset$

As the *event space* set, P(E), is the power set or parts of the sample space set, E, and it verifies the operations union and intersection of events and the definition of the opposite event (P(E),∪, ∩ , ‾) can be said to have a *Boolean Algebra structure*. From here on, we are going to go from talking about sets in general to sets of events in particular, since the events are the objects the domain of the text, and we understand that the bases of set theory on which the concepts used rest are already properly established. In formal theory, the events considered must all be sufficiently regular sets (Borel-measurable) in some space of possible results of a random system".

In addition to the five properties that allow P(E) to be given the Boolean Algebra structure, the operations union and intersection of events and the definition of the opposite event also allow A, B, C ∈ P(E) to fulfil the following five additional properties:

- Idempotency: $A \cup A = A$; $A \cap A = A$
- Identity: $A \cup E = E$; $A \cap E = A$
- Involution: A double overline = A
- Simplification: $A \cup (A \cap B) = A$; $A \cap (A \cup B) = A$
- Morgan's Laws:

  (a) $\overline{A \cup B} = \overline{A} \cap \overline{B}$. Proof:

  $$\text{If } x \in \overline{A \cup B} \rightarrow x \notin A \cup B \rightarrow x \notin A \text{ y } x \notin B \rightarrow x \in \overline{A} \text{ y } x \in \overline{B} \rightarrow x \in \overline{A} \cap \overline{B}$$
  $$\text{If } x \in \overline{A} \cap \overline{B} \rightarrow x \in \overline{A} \text{ y } x \in \overline{B} \rightarrow x \notin A \text{ y } x \notin B \rightarrow x \notin A \cup B \rightarrow x \in \overline{A \cup B}$$

  (b) $\overline{A \cap B} = \overline{A} \cup \overline{B}$. Proof:

  $$\text{If } x \in \overline{A \cap B} \rightarrow x \notin A \cap B \rightarrow x \notin A \text{ y } x \notin B \rightarrow x \in \overline{A} \text{ y } x \in \overline{B} \rightarrow x \in \overline{A} \cup \overline{B}$$
  $$\text{If } x \in \overline{A} \cup \overline{B} \rightarrow x \in \overline{A} \text{ y } x \in \overline{B} \rightarrow x \notin A \text{ y } x \notin B \rightarrow x \notin A \cap B \rightarrow x \in \overline{A \cap B}$$

Once we have seen what refers to the concept of event and before entering the concepts of random variable and probability, we are going to see the concept *σ-algebra* on A, where A is any subset of the space of events P(E), $A \subseteq P(E)$.[11]

[11] This definition is valid if E is finite, if E was an infinite set and non-numerable of $\mathbb{R}$ the σ-algebra was not be defined over $A \subseteq P(E)$ but it would be of Boel.

S is said to be a σ-algebra defined on E if it verifies the following two properties:

1. $A \in S \rightarrow \overline{A} \in S$.

   If this property is verified by S, two associated conclusions are immediately deduced:

   (a) E∈ S. Proof:

   By property 5, the complementary element of the event space, $A \cup \overline{A} = E$. Since $A \in S \rightarrow \overline{A} \in S$, both A and $\overline{A}$ as their union, E, belong to S.

   (b) $\overline{\varnothing} \in S$. Proof:

   E ∈ S y de 1. $A \in S \rightarrow \overline{A} \in S$, and by the definition of a complementary element, we know that $\overline{E} = \varnothing$; therefore, $\overline{\varnothing} \in A$.

2. $\{A_i\}_{i=1}^{\infty} \subset S \rightarrow \cup_{i=1}^{\infty} A_i \in S$

As an example continuing with the example of the throwing of a die, we are going to define two examples of two σ-algebra of the possible ones and we are going to verify that they fulfil the properties to be so:

The first is S = {∅, {1, 2, 3}, {4, 5, 6}, E}. We verify the first property: We verify the properties: $A \in S \rightarrow \overline{A} \in S$. {1, 2, 3} ∈ *S*, the complement of {1, 2, 3} is {4, 5, 6}, which also belongs to S; therefore, the first property is true. We verify the second property. The second property says that the union of all sets, events, contained in S, also belongs to S; consequently, {1, 2, 3} ∪ {4, 5, 6} = {1, 2, 3, 4, 5, 6} ∈ S, and since {1, 2, 3, 4, 5, 6} = E and E ∈ S, the second property also holds. Since E belongs to S, ∅ must also belong, but we see in the definition of S that it does, so S fulfils both properties and is a σ-algebra.

The second is S = {∅, {1, 2}, {3,4}, {5,6}, {1, 2, 3, 4}, {1, 2, 5, 6}, {3, 4, 5, 6}, E}. The first property is verified since there are the sets ∅ and E, and each of the sets of S has its complement, {1, 2} ↔ {3, 4, 5, 6}, {3, 4} ↔ {1, 2, 5, 6}, {5,6} ↔ {1, 2, 3, 4}. The second property is also verified because all unions of sets of S are also found in S, as there are many we put only two examples {1, 2} ∪ {5, 6} → {1, 2, 5, 6}; {3, 4} ∪ {1, 2, 5, 6} → E.

The definition of a σ-algebra A on an E allows defining a c*omplete system of events* or *partition* of E to all set of events $\{A_i\}_{i=1}^{\infty} \subset S$ that verifies the following two properties:

1. $A_i \cap A_j = \varnothing \ \forall \ i \neq j$
2. $\cup_{i=1}^{\infty} A_i = E$

As an example of a complete system of events or partition of E, following the example of σ-algebra, we will see that two complete systems of events or partitions of both σ-algebras can be defined. For the first one is S = {∅, {1, 2, 3}, {4, 5, 6}, E}, the partition is {∅, {1, 2, 3}, {4, 5, 6}}, since it is verified that $A_i \cap A_j = \varnothing \ \forall \ i \neq j$, that is, {1, 2, 3} ∩ {4, 5, 6} = ∅, and the intersections with the empty set are the empty set; it also fulfils the second property ∅ ∪ {1, 2, 3} ∪ {4, 5, 6} = E.

For the second σ-algebra, the partition is {∅, {1, 2}, {3,4}, {5,6}}, since their intersections are ∅ and its union E. As seen, the events with four elementary events have been eliminated because the intersections with any other set of events would not be empty.

Once we know the basics of the event concept and σ-algebra, we can get into the concepts of random variable and probability.

Let the pair formed by a sample space E and a σ-algebra S defined on it, (E, S), consider a random variable any application v: S → R// A → v (A) such that ∀ x ∈ R ∃ a set Ax = {A ∈ E / v (A) ≤ x} is an event belonging to the σ-algebra S, or what is the same, v^{-1} ((−∞, x)) ∈ σ -algebra S. If the map v is S → N, v is called a *discrete random variable.*

Random variable. The first thing to be very clear is that a random variable is an application or function and as an example we will take the application or function that gives us the number of sixes obtained in 2 rolls of a die, the well-known Chevalier De Mere problem.[12] Following the definition, to be a valid random variable ∀ x ∈ R ∃ a set Ax = {A ∈ E/v (A) ≤ x}. Being a count of the number of times the smallest x that can give is 0, and if it is a random variable, there must be a set of the sample space such that the function applied to said set, that is, the number of sixes in the set, is less or just like a hill. To check this, the first thing we do is determine the sample space of the random experiment. For the rolls of a die, the sample space is the known E = {1, 2, 3, 4, 5, 6}. For the rolls of two dice, the elemental events will be all possible variations with repetition of 6 elements, which are the six faces of the dice, taken two by two, which is the number of dice that are rolled each time. They are variations and not combinations because, for example, the result of the first die 1 and the second die 4 is different from the result of the first die 4 and the second die 1. They are repeated because the same result can be given on both dice. Consequently, the number of elementary events that we have is Vm. n = mn → V6,2 = 62 = 36. The elements and elementary events of the sample space are E = {(1,1), (1, 2), (1,3), (1,4), (1,5), (1,6), (2,1), (2,2), (2,3), (2,4), (2,5), (2,6), (3,1), (3,2), (3,3), (3,4), (3,5), (3,6), (4.1), (4.2), (4.3), (4.4), (4.5), (4.6), (5.1), (5.2), (5.3), (5.4), (5.5), (5.6), (6.1), (6.2), (6.3), (6.4), (6.5), and (6,6)}. Returning to the original problem that was to find in E a set such that the number of sixes is less than or equal to zero, as can be easily observed, said set exists and is: A0 = {A ∈ E / v (A) ≤ 1} = v-1 ((−∞, 0)) = {(1,1), (1,2), (1,3), (1,4), (1,5), (2,1), (2,2), (2,3), (2,4), (2,5), (3,1), (3,2), (3,3), (3,4), (3, 5), (4.1), (4.2), (4.3), (4.4), (4.5), (5.1), (5.2), (5.3), (5.4), (5.5)}.

The next x that we must check is x = 1, there must be an A1 = {A ∈ E / v (A) ≤ 1} = v^{-1} ((−∞, 1))., That is, the number of sixes is less than or equal to 1, said set exists and is A1 = {(1,1), (1,2), (1,3), (1,4), (1,5), (1,6), (2,1), (2,2), (2,3), (2,4), (2,5), (2,6), (3,1), (3,2), (3,3), (3,4), (3,5), (3,6), (4,1), (4,2), (4,3), (4,4), (4,5), (4.6),

[12] The Chevalier (Knight, in French) De Mere, was a Renaissance libertine who posed a famous mathematical puzzle: Which is more likely: to get at least a 6 in 4 rolls of a die or to obtain at least two 6 in 24 rolls of two dice?

(5.1), (5.2), (5.3), (5.4), (5.5), (5.6), (6.1), (6.2), (6.3), (6.4), (6.5)}. Finally, the last x to check is x = 2, the set also exists, and it can be easily observed that it is E. Any x greater than 2 would not make sense by the logic of the experiment. Therefore, it can be concluded that the number of sixes obtained in two rolls of a die is a random variable, and furthermore, as the values of said variable belong to the natural numbers, that is, it is a function S → N, it is a discrete random variable.

Let the pair formed by a sample space E and a σ-algebra defined on it, (E, S), called the *probabilizable space*, define a *probability* on it for every application p: S → $S \rightarrow \mathbb{R}^+$, which verifies the following two axioms:

1. p(E)=1
2. $\forall\{A_i\}_{i=1}^{\infty} \subset S \text{ con } A_i \cap A_j = \varnothing \forall i \neq j \rightarrow p\left(\cup_{i=1}^{\infty} A_i\right) = \sum_{i=1}^{\infty} p(A_i)$

In this example, continuing with the example of rolling a die, whose sample space we know to be E = {1, 2, 3, 4, 5, 6}, and taking a σ-algebra S defined on it, for example, one of those that we verified that they were in the example on σ-algebras S = {∅, {1, 2, 3}, {4, 5, 6}, E}, let us see if some applications of $S \rightarrow \mathbb{R}^+$ can be considered probability functions on said probabilizable space.

The first is:

$$p_1 : S \rightarrow \mathbb{R}^+ : [\, p(\varnothing) = 0, p(E) = 1, p(\{1,2,3\}) = 7/6, p(\{4,5,6\}) = 1/6]$$

To see if p_1 is a probability function on the probabilizable space (E, S), we check if it verifies the axioms. The verification of the first axiom is immediate since it is part of the definition of the application p_1, p(E) = 1. The verification of the second axiom applies only between the sets ∅, {1, 2, 3}, {4, 5, 6}, which are those that verify $A_i \cap A_j = \varnothing \ \forall\ i \neq j$, and it must be verified that $p\left(\cup_{i=1}^{\infty} A_i\right) = \sum_{i=1}^{\infty} p(A_i)$, which in this case is $\left(\cup_{i=1}^{\infty} A_i\right) = \varnothing \cup \{1, 2, 3\} \cup \{4, 5, 6\} = \{1, 2, 3, 4, 5, 6\} = E$, from where p(∅ ∪ {1, 2, 3} ∪{4, 5, 6}) = p(E) = 1; on the other hand, p(∅) + p({1, 2, 3}) + p({4, 5, 6}) = 0 + 6/7 + 1/7 = 1, so it is verified that p1 is a probability map over the probabilizable space (E, *S*).

The second application that we are going to see is p_2: $S \rightarrow \mathbb{R}^+$: [p(∅) = 0, p(E) = 1, p({1, 2, 3}) = 1/2, p({4, 5, 6}) = 2/3]. It is easy to see that the second axiom does not hold, since p(∅ ∪ {1, 2, 3} ∪{4, 5, 6}) = p(E) = 1; and on the other hand p(∅) + p({1, 2, 3}) + p({4, 5, 6}) = 0 + 1/2 + 2/3 = 1.16, so what is not verified that p_2 is a probability map on the probabilizable space (E, *S*)

In the definition of the probability function and in the examples about it, it is easy to see, on the one hand, that different probability functions can be defined on a probabilizable space, all of them equally valid; and on the other hand, that for different probabilizable spaces, different equally valid probability functions can be defined. For this reason, a definition of probability that is universally valid for any probabilizable space is highly valid, and this is verified by the *classical definition of*

probability or *Laplace probability*.[13] As was seen in the previous subsection, the classical definition of probability says that the probability of the occurrence of an event A is equal to the number of cases in which A appears, n_A, divided by the total number of cases, n_T, or what is the same:

$$p(A) = \frac{n_A}{n_T}$$

In this example, we are going to prove that the classical Laplace probability verifies the axioms of the Kolmogorov probability, that is, if we apply the classical definition of probability to the previous example of the rolling of a die whose sample space we know is E = {1, 2, 3, 4, 5, 6}; and we take the same σ-algebra *S* on it, S = {∅, {1, 2, 3}, {4, 5, 6}, E}, the only mapping of $S \rightarrow \mathbb{R}^+$ that can be considered classical probability functions on said probabilizable space is p_c: $S \rightarrow \mathbb{R}^+$: [p (∅) = 0, p(E) = 1, p({1, 2, 3}) = 1/2, p({4, 5, 6}) = 1/2].

It is easy to check, following the same reasoning as the probability examples, that it fulfils the Kolmogorov probability axioms. However, now the probability values for the events {1, 2, 3} and {4, 5, 6} can no longer be any pair that allows verifying the second Kolmogorov axiom, as they were for example 6/7 and 1/7 in the example above, but they have to verify the classical definition of probability and can only be 1/2 and 1/2. To determine this value, the equation $p(A) = \frac{n_A}{n_T}$ is applied. In the case of the set A = {1, 2, 3}, n_A is the number of cases in which, when rolling a die, we obtain a result, an elementary event, that belongs to said set, and there are 3 cases, get a 1, a 2, or a 3; n_T is the number of total cases, and these are as we have already seen in several example 6 cases, therefore we have p(A) = 3/6 = 1/2. For the set {4, 5, 6}, the reasoning is analogous.

Once the concepts of sample space, σ-algebra and probability are known, we can define a probabilistic space: A triple formed by a sample space, a σ-algebra A defined on it, and a probability defined on A, (E, A, p) is called *probability* or *probabilistic space*. From here, we refer to the probability function using the classical definition of probability to define the spaces probabilistic.

Once we have introduced the concept of probability space, we will show that the property application verifies the properties that we listed for classical probability in the previous section of the chapter. Consequently, if A and B ∈ at (E, A, p), the probability application verifies the following five properties:

1. $A \subset B \rightarrow p(A) \leq p(B)$.

   Proof: If $A \subset B \rightarrow B = A \cup (\overline{A} \cap B)$ From the definition of a complementary element, we know that $A \cap (\overline{A} \cap B) = \emptyset$, so they verify the first part of axiom 2, which implies that they verify the second part, $p(A \cup (\overline{A} \cap B)) = p(A) + p(\overline{A} \cap B)$,

[13] It is not a universal definition because, on the one hand, it is not applicable to an infinite set, that is, with infinite events; and on the other, it requires that the events be symmetrical, that is, that the elementary events that compose them be equiprobable

which implies that p(B)= p(A)+p($\overline{A}\cap$B). Since A$\subset$B$\rightarrow$B$\geq$A$\rightarrow$ $\overline{A}\cap$B$\neq\emptyset\rightarrow$ p($\overline{A}\cap$B)$\geq$0$\rightarrow$p(B)$\geq$p(A).

2. $p(\overline{A}) = 1 - p(A)$

   Proof: By the definition of a complementary element, we know that A$\cap\overline{A} = \emptyset$, so A and $\overline{A}$ verify the first part of axiom 2, which implies that they verify the second part p(A$\cup\overline{A}$) = p(A)+p($\overline{A}$). By the definition of the complementary element, we know that A$\cup\overline{A}$ = E, which implies that p(E) = p(A)+p($\overline{A}$). Therefore, by the first axiom, we have 1= p(A)+p($\overline{A}$).

3. *0$\leq$p(A)$\leq$1*

   Proof: There cannot be any A/A$\subset\emptyset$ since $\emptyset$ is the smallest subset in A, which implies that p($\emptyset$) is the smallest probability that we can find in A, and by property 1, we know that p($\varnothing$) = 0. There cannot be any A/ / E $\subset$A, since E is the largest subset in A, which implies that, by property 1, p(E) = 1 is the highest probability that we can find in A. Any other set A$\in$A will satisfy that A$\subset$E, therefore, by property 4, p(A) $\leq$ p(E).

4. *p(A$\cup$B)=p(A)+p(B)-p(A$\cap$B)*

   Proof: A$\cup$B=($\overline{A}\cap$B)$\cup$($\overline{B}\cap$A) $\cup$ (A$\cap$B), where ($\overline{A}\cap$B), ($\overline{B}\cap$A), (A$\cap$B) disjoint subsets, that is, their two-by-two intersections give $\emptyset$, so they verify the first part of axiom 2, which implies that they verify the second part p(A$\cup$B) = p(($\overline{A}\cap$B)$\cup$ ($\overline{B}\cap$A)$\cup$ (A$\cap$B))=p($\overline{A}\cap$B)+p($\overline{B}\cap$A)+p(A$\cap$B). On the other hand A=($\overline{B}\cap$A)$\cup$ (A$\cap$B) which are also disjoint sets and the second axiom can be applied to them, p (A)=p($\overline{B}\cap$A)+p(A$\cap$B). The same happens for B, p(B)=p($\overline{A}\cap$B)+p(A$\cap$B). Solving in these last two equations p($\overline{B}\cap$A) and p($\overline{A}\cap$B) and substituting in the first one, we have p(A$\cup$B)= (p(A)-p(A$\cap$B))+ (p(B)-p(A$\cap$B))+ p(A$\cap$B).

5. *p($\varnothing$)=0*

   Proof 1:

   From the definition of the complementary element, we know that E$\cap\varnothing = \varnothing$. Therefore, E and $\varnothing$ verify the first part of axiom 2, which implies that they verify the second part p(E$\cup\varnothing$)=p(E)+p($\varnothing$). By the neutral element property E$\cup\emptyset$=E, which implies that p(E) = p(E)+p($\varnothing$)$\rightarrow$p($\varnothing$)=0.

   Proof 2:

   Let $\{A_i\}_{i=1}^{\infty} = \emptyset$. From the definition of the neutral element, we know that $\emptyset\cap\emptyset = \emptyset$, so they verify the first part of axiom 2, which implies that they verify the second part $p\left(\cup_{i=1}^{\infty}\varnothing\right) = \sum_{i=1}^{\infty} p(\varnothing)$. By the property of neutral element $\emptyset\cup\emptyset = \emptyset$, for as many $\emptyset$ as there are, which implies that $p(\emptyset) = \sum_{i=1}^{\infty} p(\varnothing)\rightarrow$p ($\varnothing$)=k. p($\varnothing$), where k is the number of $\emptyset$ there is, which implies that for the equality to be fulfilled, the only possibility is that p($\varnothing$) = 0.

On the basis of a probability space, the probability of joint occurrence of more than one event is defined as:

Let a (E, A, p), let A and B $\in$ A, and let p(B) $>$ 0, the probability of appearance of A having given B, p(A | B),[14] is defined as the application, p:$A \rightarrow \mathbb{R}^+$,which fulfills the following two axioms of probability:

p: $A \rightarrow \mathbb{R}^+$
p(A|B)=p(A∩B)/p(B)

Conditional probability[15] fulfils the two axioms of probability:

1. p(E) = 1
2. $\forall \{A_i\}_{i=1}^{\infty} \subset A$ con $A_i \cap A_j = \varnothing \forall i \neq j \rightarrow p\left(\cup_{i=1}^{\infty} A_i\right) = \sum_{i=1}^{\infty} p(A_i)$

From the previous definition, it can be defined that two events are *independent* if the appearance of one of them does not change the probability of the appearance of the other:[16]

$$p\left(A_i \cap A_j\right) = p(A_i)p\left(A_j\right) \forall i \neq j$$

From the definition of the probability of joint occurrence of events, can be stated the *Total Probability Theorem* as:

Let (E, A, p) and let $\{B_i\}_{i=1}^{\infty} \subset A$ be a *partition* of A, such that the probabilities of all its elements are known, and let A∈A / let p(A| B_i)$\forall i \in \mathbb{N}$,$, the probability of A is:

$$p(A) = \sum_{i=1}^{\infty} p(A|B_i)p(B_i)$$

[14] The probability of A is marked P(A), the probability of A given the event B, which can be an elementary event or a set of events, is marked P(A | B)

[15] For an event A there is a probability, called a priori, which is the one it has when more events are not taken into account; for said event A there is another possibility, called a posteriori, which is what it has when the occurrence of other events is taken into account. If the occurrence of these events does not change the prior probability of A and its posterior probability is the same, these events are independent of event A.

[16] If two events are independent, the probability of joint occurrence of both is equal to the multiplication of their prior probabilities; if there are more events, the joint probability is the multiplication of the occurrence of all of them.

Anomaly Detection

In this fourth chapter, we are going to see the foundations of the identification of anomalous data, *outliers*, and the main techniques used to carry it out, is also known as *anomaly detection*. It is structured as the rest of the chapters in three sections.

Section A introduces, in a theoretical and, at the same time, practical way, all the basic theoretical knowledge related to the concept of anomaly detection or Outliers identification that a data analyst must know in depth, from its definition of what means the concept to the introduction of some of the most commonly used techniques to identify them.

Section B presents the computer-based solving of the same examples used in section A to introduce theoretical knowledge. Section B presents the computer-based solving. The reader continues going into a deep knowledge of R and is presented how anomaly detection problems are solved with the use of R.

Section C will consist of a set of statements of exercises about anomaly detections in which detailed solutions can also be found in this section of the chapter.[1]

A. Theory

The first section of the chapter is structured into four subsections: 1. Introduction, 2. Anomaly Detection Based on Statistics, 3. Anomaly Detection Based on Proximity, 4. Anomaly Detection Based on Density. The basic knowledge related to the concept of anomalies or outliers and some of the main techniques used to identify them are presented in detail.

[1] We repeat again here that in order to obtain the best results for the learning process throughout the use of the book, it is very important that the reader tries to solve the exercises by him/herself before seeing their solutions and that only once solved they check if the obtained solutions are correct.

J. J. Cuadrado-Gallego, Y. Demchenko, *Data Analytics*,
https://doi.org/10.1007/978-3-031-39129-3_4

Introduction

We are going to study the concept of *detecting anomalous events*, which can also be called *anomalies* or *outliers*.[2] The studies of anomaly detection or anomalous event identification, also called outliers, seek to find and classify as outliers those events that are very different[3] from the rest of those that make up the studied sample of the set P(Ω). To give a measure of how anomalous an event is, the variable score *degree of outlier*, *outlier score*, or *degree of anomaly* have been defined. The way to measure the degree of outliers will depend on the technique used.

Identifying anomalous data is essential for good data analysis.

The outlier degree is arbitrarily set by the data analyst taking into account the study they are conducting.

Anomalous data can be:

- Erroneous data, coming from measurement errors, must be eliminated because they will lead to an analysis of data with erroneous conclusions. Outlier detection is often part of data preprocessing, specifically data cleansing.

  Correct data, with a lot of significance, that deviate from the normal must be analyzed very carefully because they can lead to important findings.

To introduce the concept of identifying outliers through an example, we use marks in a subject from a group of students. The elementary events are each of the marks individually E = {Theory, Laboratory}. The elementary event ratings will have values from 0 to 5, where 5 will be the highest possible rating and 0 the lowest. The event sample in which the outliers will be searched is made up of the following five events, or student grades: 1. {4, 4}; 2. {4, 3}; 3. {5, 5}; 4. {1, 1}; 5. {5, 4}. The problem to be solved is to identify if one or more of these qualifications are very different from the rest and, therefore, can be considered outliers.

Different identification techniques can be used to obtain the outliers. This differentiation is based on the fundamentals of the analysis technique used. Some of the best known and most commonly used techniques[4] can be classified as follows:

- Based on *statistics*. Based on statistical knowledge, they use statistical parameters to look for anomalous events.
- Based on *proximity*.[5] They look for events that are widely separated from other events, so they are based on some definition of distance.

[2] From here and in the rest of the text, we will refer to anomalous data as anomalies or outliers since both terms are used interchangeably.

[3] An event consisting of multiple individual characteristics can be abnormal even though the individual values of each characteristic are not.

[4] Other classifications can be found in the literature. We like this simplified one for the entry level.

[5] In the case of one-, two-, or three-dimensional events, which can be represented graphically, their graphical representation is very useful to make a first visual identification of the outliers.

- Based on *density*.[6] They look for events that are located in a spatial zone in which there is less density of events than the observed mean for that sample.

In the next sections of this chapter, we are going to see how each one works in a specific way.

Anomaly Detection Based on Statistics

These techniques use statistical parameters to look for anomalous events. Two of these techniques will be explained: The first one is based on the use of the mean and standard deviation, and the second one is based on the use of quartiles.

Anomaly Detection Based on the Mean and Standard Deviation

Tchebychev's inequality establishes that, in any normal distribution, between the mean and two standard deviations, there are at least 75% of the data and between the arithmetic mean and three standard deviations, 89%. Therefore, it can be used to identify anomalous data, establishing as anomalous data those data that are at a distance greater than three, or even more, depending on the analyst decision, standard deviations from the mean value.

The identification of anomalous events using the standard deviation technique follows a five-step process:

1. Determination of the degree of outlier or distance at which an event (point) is considered an outlier. It is chosen arbitrarily.
2. As we saw in Chapter 2, "Data", the arithmetic mean is obtained using the equation:

$$\overline{x} = \frac{\sum_{i=1}^{n} x_i}{n}$$

And using frequencies:

$$\overline{x}_a = \frac{\sum_{i=1}^{n} f_i x_i}{\sum_{i=1}^{n} f_i}$$

[6] Although the graphical representation of the data can help a lot in the identification of the events that can be outliers, using only visual identification is not possible because it cannot be known if the identified values exceed the threshold of the chosen outlier degree.

3. The standard deviation is obtained using the equation:

$$s = \sqrt{\frac{\sum_{i=1}^{n}(x_i - \overline{x})^2}{n}}$$

And using frequencies:

$$s_a = \sqrt{\frac{\sum_{i=1}^{n} f_i (x_i - \overline{x}_a)^2}{\sum_{i=1}^{n} f_i}}$$

4. The limits of the interval for the outliers are calculated using the equation:

$$(\overline{x}_a - ds_a, \overline{x}_a + ds_a)$$

where d is the distance established in step 1.

5. Outliers are identified as the values that fall outside the range calculated in step 4.

To introduce the mean and standard deviation technique for identifying outliers through an example, we use the example described in the introduction in which the outliers will be searched in the following five events, or student grades: 1. {4, 4}; 2. {4, 3}; 3. {5, 5}; 4. {1, 1}; 5. {5, 4}. As this technique can be applied to only one variable, it is going to be applied over the theory marks, which are: {4, 4, 5, 1, 5} (Fig. 1).

In the first step, we establish the outlier degree as 1.5, d = 1.5.

We calculate the arithmetic mean:

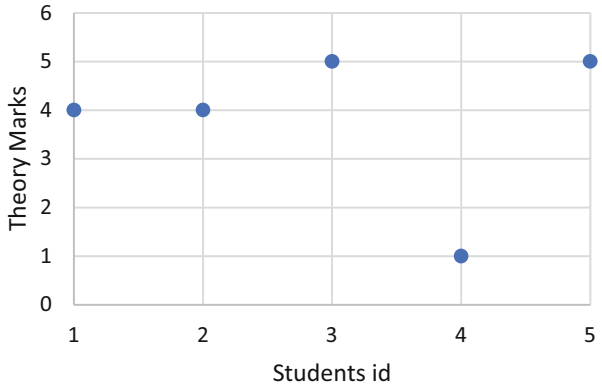


Fig. 1 Graphical representation of the theory marks data

$$\bar{x}_T = \frac{\sum_{i=1}^{5} x_i}{n} = \frac{4+4+5+1+5}{5} = 3.8$$

And from the mean, we calculate the standard deviation:

$$s_T = \sqrt{\frac{\sum_{i=1}^{5} (x_i - \bar{x}_T)^2}{n}}$$

$$= \sqrt{\frac{(4-3.8)^2 + (4-3.8)^2 + (5-3.8)^2 + (1-3.8)^2 + (5-3.8)^2}{5}} = 1.469$$

From these results, and taking into account the degree of outlier taken of 1.5, the interval for normal data is:

$$(\bar{x} - 1,5.s, \bar{x} + 1,5.s) = (3.8 - 1{,}5.1{,}47, 3.8 + 1,5, 1.47) = (1.59, 6.00)$$

Consequently, the mark value 1 is outside the interval, in the lower limit, and in consequence, it is an anomaly detected and can be considered an outlier (Fig. 2).

The mean and standard deviation technique for outlier identification is well known and used, but it has the problem that the outliers are used to calculate the mean and standard deviation and, therefore, impact its own calculation, which distorts the result. To solve this problem, other techniques are used, and one of the most well-known techniques is based on quartiles.

Anomaly Detection Based on the Quartiles

The outlier detection model based on quartiles is based on the previous calculation of the data ordering measures and, in particular, in the identification of quartiles. This

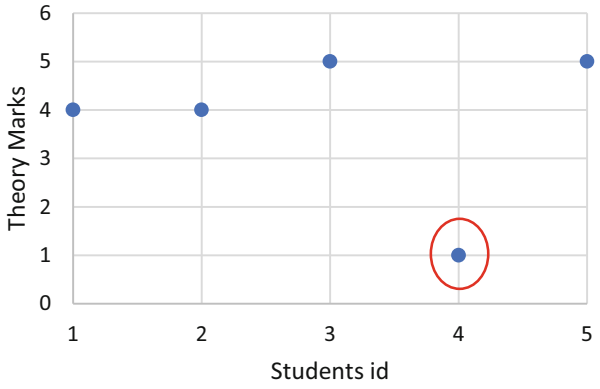


Fig. 2 Graphical representation of the theory marks data with the outlier

data identification technique is related to and complementary to the data visualization technique called Box and Whiskers.

The identification of anomalous events from the Box and Whiskers technique follows a four-step process:

1. Determination of the degree of outlier or distance at which an event (point) is considered an outlier. It is chosen arbitrarily.
2. The data are ordered, and the quartiles are obtained using the equation:

$$\tilde{x}_c = x_{[nc]+1} \text{ if nc} \notin \mathbb{N}\ [nc] \text{ integer part of } nc$$

$$\tilde{x}_c = \frac{x_{[nc]} + x_{[nc]+1}}{2} \text{ fi nc} \in \mathbb{N}$$

3. The interval limits for outliers are calculated using the equation:

$$(Q_1 - 1.5(Q_3 - Q_1), Q_3 + 1.5(Q_3 - Q_1))$$

4. Outliers are identified as the values that fall outside the range calculated in Step 3.

To introduce the quartiles technique for identifying outliers through an example, we use the example described in the introduction and the example of the previous method, in which the outliers will be searched in the following five events, or student grades: 1. {4, 4}; 2. {4, 3}; 3. {5, 5}; 4. {1, 1}; 5. {5, 4}. As this technique can be applied to only one variable, it will be applied over the laboratory marks, which are {4, 3, 5, 1, 4} (Fig. 3).

We calculate the quartiles:

To do that, we must first organize the values of the observed event by magnitude, which is, $\{1, 3, 4, 4, 5\} .\equiv \{x_1, x_2, x_3, x_4, x_5\}$

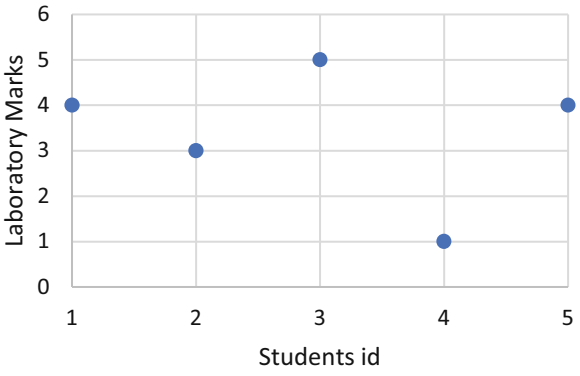


Fig. 3 Graphical representation of the laboratory marks data

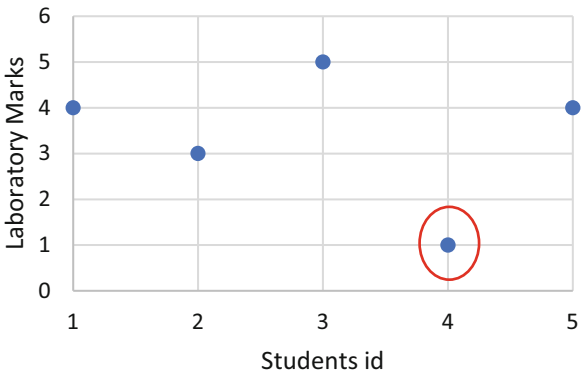


Fig. 4 Graphical representation of the laboratory marks data with the outlier

First quartile: n = 5, c = 1/4 in consequence n.c = 5/4 = 1.25 $\notin \mathbb{N}$ and the equation to calculate the first quartile, where $[nc]$ nondecimal part of nc, is:

$$Q_1 = \tilde{x}_{\frac{1}{4}} = x_{[nc]+1} = x_{[1.25]+1} = x_{1+1} = x_2 = 3$$

And the third quartile: n = 5, c = 3/4 in consequence n.c = 15/4 = 3.75 $\notin \mathbb{N}$ and the equation to calculate the third quartile is:

$$Q_3 = \tilde{x}_{\frac{3}{4}} = x_{[nc]+1} = x_{[3.75]+1} = x_{3+1} = x_4 = 4$$

From these results, and taking into account the degree of outlier taken of 1.5, the interval for normal data is:

$$\begin{aligned}(Q_1 - 1.5(Q_3 - Q_1), Q_3 + 1.5(Q_3 - Q_1)) &= (3 - 1.5(4-3), 4 + 1.5(4-3)) \\ &= (1.5, 5.5)\end{aligned}$$

Consequently, if the mark value 1 is outside the interval, it is an anomaly detected and can be considered an outlier (Fig. 4).

Anomaly Detection Based on the Standard Error of the Residuals

This technique is applied to pairs of data for which a previous statistical regression analysis has been performed. The identification of anomalous events using the regression technique follows a five-step process:

1. Determination of the degree of outlier or distance at which an event (point) is considered an outlier. It is chosen arbitrarily (usually 3 or 4).
2. Linear regression is obtained using the equations:

$$b = \frac{s_{xy}}{s_x^2}$$

$$\mathrm{a} = \overline{y} - b\overline{x}$$

where s_{xy} is the covariance of x and y, which can be calculated using the equation:

$$s_{xy} = \frac{\sum_{i=1}^{n} x_i y_i}{n} - \left(\frac{\sum_{i=1}^{n} x_i}{n}\right).\left(\frac{\sum_{i=1}^{n} y_i}{n}\right) = \frac{\sum_{i=1}^{n} x_i y_i}{n} - \overline{x}.\overline{y}$$

which in its extended version is:

$$s_{xy} = \frac{\sum_{i=1}^{n} \sum_{j=1}^{m} f_{ij} x_i y_j}{\sum_{i=1}^{n} f_i} - \left(\frac{\sum_{i=1}^{n} f_i x_i}{\sum_{i=1}^{n} f_i}\right).\left(\frac{\sum_{j=1}^{m} f_j y_j}{\sum_{j=1}^{m} f_j}\right);$$

s_x^2 is the variance of x, and its equation of calculus is:

$$s_x^2 = \frac{\sum_{i=1}^{n} (x_i - \overline{x})^2}{n}$$

3. The standard error of the residuals is obtained using the equation:

$$s_r = \sqrt{\frac{\sum_{i=1}^{n} (y_i - y_{ci})^2}{n}}$$

where $y_i - y_{ci}$ are the residuals of each value, with y_{ci} being the value of each y corresponding to each x for each i calculated using the regression equation calculated in step 2, that is:

$$y = a + bx$$

$$y_{ci} = a + bx_i$$

4. The limits of the interval for the outliers are calculated using the equation:

$$d.s_r$$

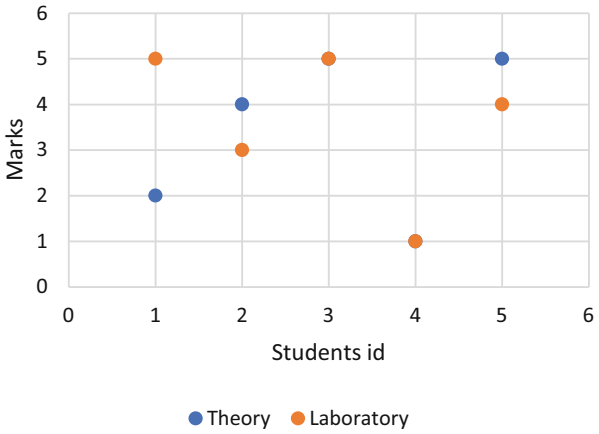


Fig. 5 Graphical representation of the theory and laboratory marks data

5. Outliers are identified as those such that $|y_i - y_{ci}| > ds_r$.

To introduce the standard error of the residuals technique for identifying outliers through an example, we use the example described in the introduction and the example of the previous methods, in which the outliers will be searched in the following five events, or student grades, but with a small change to allow us to obtain a function[7] and to see the outlier more clearly, the first value has changed to {2, 5}, and as a result, we have 1. {2, 5}; 2. {4, 3}; 3. {5, 5}; 4. {1, 1}; 5. {5, 4}. As this technique can be applied to two variables, it will be applied over the Theory and Laboratory marks (Fig. 5).

To apply the method of the standard error of the residuals, the first thing that we must do is to calculate the regression function that gives us the marks of laboratory as a function of the marks of theory. To do that, we must calculate the means of theory data and laboratory data, the covariance of both, and the variance of theory.

To calculate the mean, we use the known equations:

$$\overline{x} = \frac{\sum_{i=1}^{n} x_i}{n} = \frac{\sum_{i=1}^{5} x_i}{5} = \frac{2+4+5+1+5}{5} = 3.4$$

$$\overline{y} = \frac{\sum_{i=1}^{n} y_i}{n} = \frac{\sum_{i=1}^{5} y_i}{5} = \frac{5+3+5+1+4}{5} = 3.6$$

Now, we calculate the covariance:

[7] To be possible to obtain a function each x must have a different y, it is not possible that the same x can have two different y.

$$s_{xy} = \frac{\sum_{i=1}^{n} x_i y_i}{n} - (\overline{x}.\overline{y}) = \frac{2.5 + 4.3 + 5.5 + 1.1 + 5.4}{5} - 3, 4.3, 6 =$$
$$= \frac{68}{5} - 12, 24 = 13.6 - 12, 24 = 1.36$$

Now, we calculate the variance of x:

$$s_x^2 = \frac{\sum_{i=1}^{n} (x_i - \overline{x})^2}{n}$$
$$= \frac{(2 - 3.4)^2 + (4 - 3.4)^2 + (5 - 3.4)^2 + (1 - 3.4)^2 + (5 - 3.4)^2}{5}$$
$$= \frac{13.2}{5} = 2.64$$

With the values calculated in the previous steps, we can calculate the parameters a and b in the equation

$$b = \frac{s_{xy}}{s_x^2} = \frac{1.36}{2.74} = 0.5$$
$$\mathrm{a} = \overline{y} - b\overline{x} = 3.6 - 0{,}5.3{,}4 = 1.9$$
$$y = 1.9 + 0.5x$$

Once we have the equation of the regression, we can calculate the residuals. To do that, we must first calculate the values of y using the equation

$$y_{ci} = 1.9 + 0.5x_i$$
$$y_{c1} = 1.9 + 0.5(2) = 2.9$$
$$y_{c2} = 1.9 + 0.5(4) = 3.9$$
$$y_{c3} = 1.9 + 0.5(5) = 4.4$$
$$y_{c4} = 1.9 + 0.5(1) = 2.4$$
$$y_{c5} = 1.9 + 0.5(5) = 4.4$$

From the y_{ci} residuals are calculated as follows:

$$r_1 = y_1 - yc_1 = 5 - 2.9 = 2.1$$

In the same way are calculated $r_2 \ldots r_5$, obtaining the following values:

$$r_2 = y_2 - yc_2 = 3 - 3.9 = 0.9$$

$$r_3 = y_3 - yc_3 = 5 - 4.4 = 0.6$$
$$r_4 = y_4 - yc_4 = 1 - 2.4 = -1.4$$
$$r_5 = y_5 - yc_5 = 4 - 4.4 = -0.4$$

Once we have the residuals, we calculate the standard deviation of the residuals with the following equation:

$$s_r = \sqrt{\frac{\sum_{i=1}^{n}(y_i - y_{ci})^2}{n}}$$
$$= \sqrt{\frac{(5-2.9)^2 + (3-3.9)^2 + (5-4.4)^2 + (1-2.4)^2 + (4-4.4)^2}{7}}$$
$$= \sqrt{\frac{(2.1)^2 + (0.9)^2 + (0.6)^2 + (-1.4)^2 + (-0.4)^2}{5}} = 1.24$$

Step 4. Calculate the limits of the interval for the outliers: Since the outlier degree is d = 1.5, the limits are:

$$d.s_r = 1{,}5.1.24 = 1.86$$

Outlier identification, that is, if for any value:

$$|y_i - yc_i| > d.s_r = |y_i - yc_i| > 1.86$$

The point (1, 1) is identified as an outlier since:

$$|5 - 2.9| = 2.1 > 1.86$$

The analysis of outliers is based on the fact that the normal data will all have a similar influence on the definition of the regression line, and if any of them are missing, the shape of the line will not vary substantially. However, the inclusion or not of atypical data does significantly vary the parameters and the shape of the line. Since a line has two parameters: The slope and the ordinate at the origin, analogue data can vary either the slope or the ordinate at the origin or both at the same time.

We are going to start studying how to identify bad data that impact the slope that the line will have if we eliminate said data and, therefore, if we obtain a line more representative of the data that we have. The way to identify atypical data in the analysis of the relationship of two variables that may distort the calculation of the correct slope of the line is based on obtaining a measure called the degree of influence. The *degree of influence*, *id*, on the two-variable regression function of a given data point is measured by the equation:

$$id_i = \frac{1}{n} + \frac{x_i - \overline{x}^2}{\sum_{i=1}^{n} x_i - \overline{x}^2}$$

Using the following data [3, 2, 7, 5, 4, 4, 6, 9], obtain the degree of influence of the first point of the distribution.

If we apply the previous equation, the degree of influence is obtained as follows. First, the mean has to be calculated:

$$\overline{x} = \frac{3+2+7+5+4+4+6+9}{8} = 5$$

Then the equation of the degree of influence is applied for the first point:

$$id_1 = \frac{1}{8} + \frac{3-5^2}{40-5^2} = 0.125 - 1.467 = -1.342$$

As can be deduced from the calculation equation, the minimum degree of influence of any point of the distribution is 1/n, which corresponds to a point coinciding with the mean of x, while the maximum degree of influence of any point of the distribution is 1, which corresponds to a point that forces the regression line to pass through that point. It can also be deduced from the equation that the sum of the degrees of influence of all the data in the analysis is equal to 2, so the mean degree of influence is 2/n.

Using the concept of degree of influence, those with a degree of influence of id $>$ 6/n, or what is equal to or greater than three, can be catalogued as anomaly candidate observations to be equal to or superior to three times the average degree of influence.

The second parameter that can vary the atypical data with respect to the one obtained if we only had normal data in the sample is the ordinate at the origin. That is, we can have one or more data in the sample whose inclusion in the calculations does not vary the slope of the line of fit but does raise or lower it, that is, to increase or decrease the cut-off point with the y-axis. To identify this type of atypical data, the residuals of each data point can be used. If the residue of the data studied is much higher than the rest of the data, the observation can be classified as a candidate for abnormal observation.

The residual, id, in the two-variable regression function of a given piece of data is measured by the equation:

$$r_i = \overline{y}_i - y_i$$

However, atypical data can vary the two parameters. In fact, although the influence on either of the two is very slight, the presence of atypical data will always vary at both values. One way to identify atypical data in the analysis of the

relationship of two variables that may distort both parameters is based on obtaining a measure called Cook's influence. Cook's influence measures the change that occurs in the regression line whether or not it considers the data under study.

The influence $c_i$on the two-variable regression function of a given piece of data is measured by the equation:

$$c_i = \frac{(\overline{y}_i - \overline{y}_{-i})^2}{2s_r - 2id_i}$$

From the equation for calculating Cook's influence, it is established that a piece of data is influential if its degree of Cook's influence is greater than 1.

In addition to Cook's degree of influence, there are two other measures that can be used to determine if a piece of data is an anomaly or wrong. The first is the *standardized residual*, which, in the two-variable regression function of a given piece of data, is measured by the equation:

$$rs_i = \frac{e_i}{s_r\sqrt{1-h_i}}$$

The standardized residual follows a normal distribution with a mean of 0 and a standard deviation of 1.

The second is the *studentized residual,* which, in the two-variable regression function of a given piece of data, is measured by the equation:

$$t_i = \frac{e_i}{s_r\sqrt{1-h_i}}$$

The studentized residual follows a Student's t distribution with n-3 degrees of freedom. If t_i in absolute value is greater than the value of the student's t distribution with the desired probability, the value is considered atypical.

Anomaly Detection Based on Proximity

These techniques base the identification of outliers on the concept of distance, which is usually the Euclidean,[8] but can be others, and identify an event that, if it has two or three dimensions, can be represented by a point, as anomalous or outlier if it is very distant from most other events, or, in two or three dimensions, points.

[8] As have been seen, the Euclidean distance between two points P and Q in an n-dimensional space is defined as $d_{PQ} = \sqrt{\sum_{i=1}^{n}(p_i - q_i)^2}$, where $p_i$and $q_i$are each of the elements of P and Q.

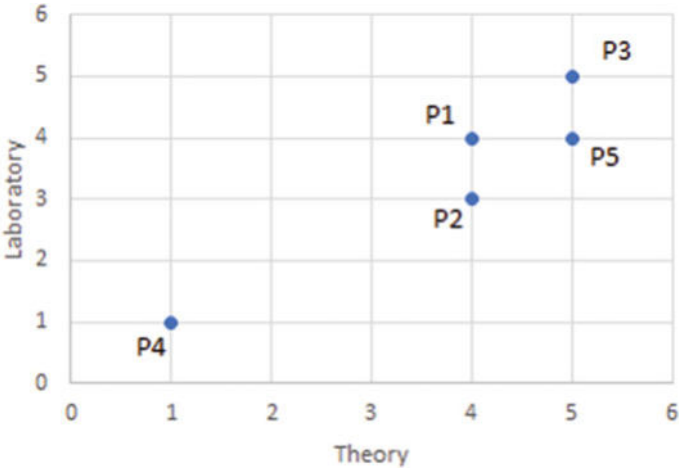


Fig. 6 Graphical representation of the theory and laboratory marks

K-Nearest Neighbor Algorithm

The outlier degree of an event is to be measured through the distance between said event and its nearest neighbor K event, which means that the distances of all the events between them must be calculated. The identification of outliers through the K-Nearest Neighbor technique follows a 2-step[9] process. Let us see how each step is treated.

As in the rest of the techniques, to apply the nearest K-neighbors technique, it is essential to always have a sample of events for which we have all the values.

To introduce the K-nearest neighbor technique for identifying outliers through an example, we use the example described in the introduction and the example of the previous methods, in which the outliers will be searched in the following five events, or student grades: 1. {4, 4}; 2. {4, 3}; 3. {5, 5}; 4. {1, 1}; 5. {5, 4} (Fig. 6).

A. Step A consists of two substeps in which the parameters with which we want to solve the problem of identifying outliers are arbitrarily set.

   1. In step 1, the outlier degree is set, that is, the distance at which an event, or point, will be considered an outlier. To choose this distance, it is useful, whenever possible, to carry out a previous observation of the sample so that the chosen value is logical.
      Selection of the outlier degree. The Euclidean distance 2.5 is chosen as the outlier degree. As mentioned above, the value is chosen arbitrarily from the observation of the sample.

[9] To continue with the same notation system used in the previous topics, we are going to call the highest level steps A, B, C, etc.

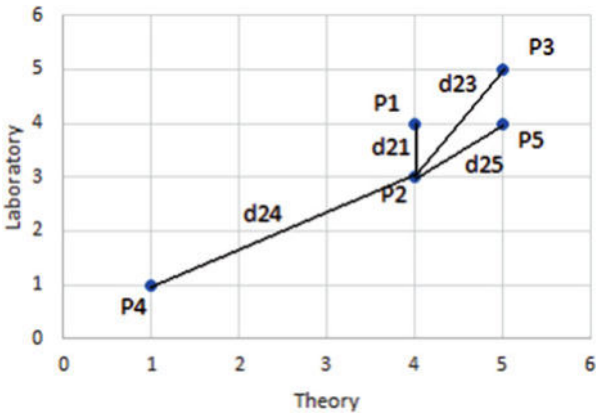


Fig. 7 Distances from P2 to the other points

2. In Step 2, the K is set, that is, the order number of the closest neighbor for which an event must have the degree of outlier for the event to be considered an outlier.

   Selection of K. As K, that is, the order number of the nearest neighbor for which the distance to the event will be measured, 3 is taken, or what is the same, the third closest event.

B. Step B also consists of two substeps, in which distances are calculated and outliers are identified.

1. In Step 1, the distances, Euclidean or others, between all the points are calculated, the neighbors of each point are ordered by distance until reaching the defined K, and those events whose K neighbor had a distance greater than the defined outlier degree are defined as outliers.

   Calculation of distances. We calculate the distance, which in this example will be the Euclidean, of each point with the rest of the points in the sample (Fig. 7).
   Points 1-2, {{4, 4}, {4, 3}}:

$$d_{12} = \sqrt{\sum_{i=1}^{2} (p_i - q_i)^2} = \sqrt{(4-4)^2 + (4-3)^2} = 1$$

   Points 1-3, {{4, 4}, {5, 5}}:

$$d_{13} = \sqrt{(4-5)^2 + (4-5)^2} = 1.41$$

   Points 1-4, {{4, 4}, {1, 1}}:

$$d_{14} = \sqrt{(4-1)^2 + (4-1)^2} = 4.24$$

Points 1-5, {{4, 4}, {5, 4}}:

$$d_{15} = \sqrt{(4-5)^2 + (4-4)^2} = 1$$

Points 2-3, {{4, 3}, {5, 5}}:

$$d_{23} = \sqrt{(4-5)^2 + (3-5)^2} = 2.24$$

Points 2-4, {{4, 3}, {1, 1}}:

$$d_{24} = \sqrt{(4-1)^2 + (3-1)^2} = 3.61$$

Points 2-5, {{4, 3}, {5, 4}}:

$$d_{25} = \sqrt{(4-5)^2 + (3-4)^2} = 1.41$$

Points 3-4, {{5, 5}, {1, 1}}:

$$d_{34} = \sqrt{(5-1)^2 + (5-1)^2} = 5.66$$

Points 3-5, {{5, 5}, {5, 4}}:

$$d_{35} = \sqrt{(5-5)^2 + (5-4)^2} = 1$$

Points 4-5, {{1, 1}, {5, 4}}:

$$d_{45} = \sqrt{(1-5)^2 + (1-4)^2} = 5$$

2. In step 2, the neighbors of each point are ordered by distances until reaching the point whose order value is equal to K, and those points whose value of the distance to the K point is greater than the chosen outlier degree are identified. Ordering of values and identification of outliers. In the case of the example, it is necessary to reach the third closest neighbor. In this example, as there are only four neighbors for each point, the ordering of the first three events or points implies the ordering of the fourth. The following arrangements are obtained:

   - Point 1: The minimum distance, and therefore the closest point is to point 5: 1. The second distance, and therefore the second closest point, is to point 5: 1; and finally, the distance to the third closest point, which is the

chosen K, is to point 3 and is 1.41. Since this distance is less than the chosen outlier degree, which is 2.5, the consequence is that point 1 is not an outlier.

- Point 2: The minimum distance, and therefore the closest point is to point 2: 1. The second distance, and, therefore, the second closest point, is to point 5: 1.41; and finally, the distance to the third closest point, which is the chosen K, is to point 3 and is 2.24. Since this distance is less than the chosen outlier degree, which is 2.5, the consequence is that point 2 is not an outlier.
- Point 3: The minimum distance, and, therefore, the closest point, is to point 2: 1. The second distance, and, therefore, the second closest point, is to point 1: 1.41; and finally, the distance to the third closest point, which is the chosen K, is to point 2 and is 2.24. Since this distance is less than the chosen outlier degree, which is 2.5, the consequence is that point 3 is not an outlier.
- Let us look at Point 5 before moving on to 4. Point 5: The minimum distance, and therefore the closest point is to point 1: 1. The second distance, and therefore the second closest point, is to point 3: 1; and finally, the distance to the third closest point, which is the chosen K, is to point 2 and is 1.41. Since this distance is less than the chosen outlier degree, which is 2.5, the consequence is that point 5 is not an outlier.
- Finally, we are going to see Point 4: The minimum distance, and therefore the closest point is to point 2: 3.61. The second distance, and, therefore, the second closest point, is to point 1: 4.24; and finally, the distance to the third closest point, which is the chosen K, is to point 5 and is 5. As the distance is greater (twice) than the chosen outlier degree, which is 2.5, point 4 (1, 1) is an outlier (Fig. 8).

As a consequence, in the observed sample, a single outlier can be identified, which is point 4 {1,1}. Although visually we could have a previous impression that

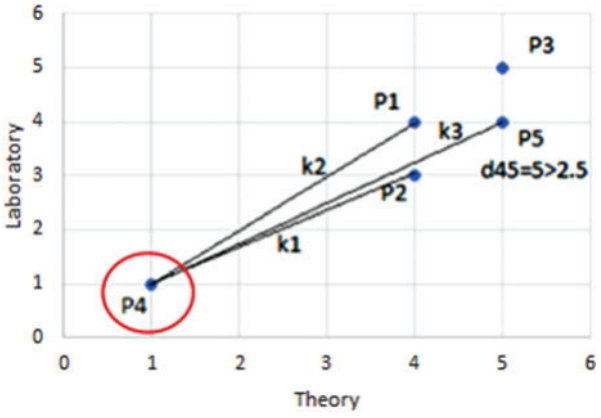


Fig. 8 Graphical identification of P4 as an outlier

this point could be outlier, it has been the application of the K-neighbors technique that has allowed us to demonstrate it mathematically.

At this point, it is very important to remember the importance of choosing the K values and the outlier degree in an analysis of proximity outliers using the closest K-neighbor technique.

Anomaly Detection Based on Density

These techniques look for events that are located in a spatial zone in which there is less density of events than the observed mean for that sample.

Simplified Local Outlier Factor Algorithm

The identification of anomalous events using relative density and the Simplified Local Outlier Factor technique follows a three-step process:

A. Determination of the outlier degree of each point by calculating the density, d, of each point. There are different definitions of density. One of the most commonly used is

$$\text{density}\ (x_i, K) = \left(\frac{\sum_{x_j \in N(x_i,K)} \text{distance}\left(x_i, x_j\right)}{\text{cardinal}\ N(x_i, K)} \right)^{-1}$$

Step A involves 4 substeps:

1. Determination of the order number, or K, of the nearest neighbor to use to calculate the density of each point. It is chosen arbitrarily.
2. Calculation of the distances between each point and the rest of the points. Manhattan distance, for two dimensions:

$$\text{distance}\ \left(x_i, x_j\right) = \left|x_{i1} - x_{j1}\right| + \left|x_{i2} - x_{j2}\right|$$

3. Calculation of the cardinal or size of the set N for each point. N is the set that contains the neighbors whose distance to *x_i* is equal to or less than that of the nearest K neighbor.
4. Calculation of the density, d, of each point. By calculating the density at each point as the inverse of the mean of the distance of the nearest K neighbors, this technique is closely related to that of proximity.

To introduce the density algorithm for identifying outliers through an example, we use the example described in the introduction and used in the previous method, in

which the outliers will be searched in the following five events, or student grades: 1. {4, 4}; 2. {4, 3}; 3. {5, 5}; 4. {1, 1}; 5. {5, 4}.

Carry out step A.1 of the density algorithm to search for anomalous data.

Determination of the order number, or K, of the nearest neighbor used to calculate the density of each point. It is chosen arbitrarily.

As it is chosen arbitrarily, taking into account the characteristics of the problem and of the sample, we take the third neighbor or closest event. K = 3.

Carry out step A.2 of the density algorithm for the identification of outliers from the previous exercise.

Calculation of Manhattan distances between all points. We calculate the distance of each point with the rest of the points in the sample (Fig. 9).

Points 1-2, {{4, 4}, {4, 3}}:$d_{12} = |x_{11} - x_{21}| + |x_{12} - x_{22}| = |4 - 4| + |4 - 3| = 1$

Points 1-3, {{4, 4}, {5, 5}}: $d_{13} = |4 - 5| + |4 - 5| = 2$

Points 1-4, {{4, 4}, {1, 1}}: $d_{14} = |4 - 1| + |4 - 1| = 6$

Points 1-5, {{4, 4}, {5, 4}}: $d_{15} = |4 - 5| + |4 - 4| = 1$

Points 2-3, {{4, 3}, {5, 5}}: $d_{23} = |4 - 5| + |3 - 5| = 3$

Points 2-4, {{4, 3}, {1, 1}}: $d_{24} = |4 - 1| + |3 - 1| = 5$

Points 2-5, {{4, 3}, {5, 4}}: $d_{25} = |4 - 5| + |3 - 4| = 2$

Points 3-4, {{5, 5}, {1, 1}}: $d_{34} = |5 - 1| + |5 - 1| = 8$

Points 3-5, {{5, 5}, {5, 4}}: $d_{35} = |5 - 5| + |5 - 4| = 1$

Points 4-5, {{1, 1}, {5, 4}}: $d_{45} = |1 - 5| + |1 - 4| = 7$

Carry out step A.3 of the simplified density algorithm for the identification of outliers from the previous exercise.

Sorting by distances of the neighbors of each point until reaching the defined K, 3, to calculate the N of each point.

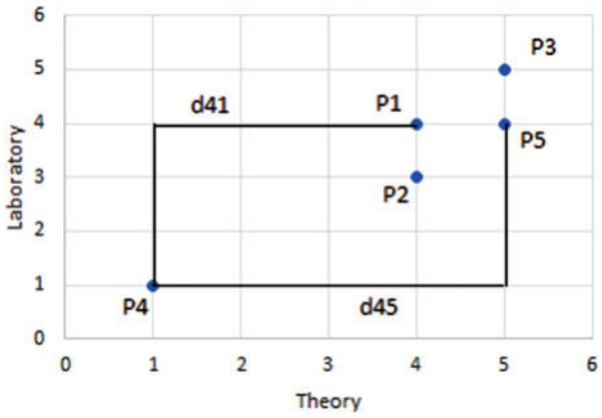


Fig. 9 Manhattan distances from P4 to points P1 and P5

- Point 1: Minimum distance, closest point, point 2: 1. Second distance, point 5: 1. Finally, the distance to the third closest point, which is the chosen K, is to point 3: 2. Therefore, N = 3.
- Point 2: Minimum distance, point 1: 1. Second distance, point 5: 2. Distance to the third closest point, which is the K chosen, is to point 3: 3. Therefore, N = 3.
- Point 3: Minimum distance, point 5: 1. Second distance, point 1: 2. Distance to the third closest point, which is the K chosen, is to point 2: 3. Therefore, N = 3.
- Point 4: Minimum distance, point 2: 5. Second distance, point 1: 6. Distance to the third closest point, which is the K chosen, is to point 5: 7. Therefore, N = 3.
- Point 5: Minimum distance, point 1: 1. Second distance, point 3: 1. Distance to the third closest point, which is the K chosen, is to point 2: 2. Therefore, N = 3.

N does not always coincide with K, and we are going to answer a question with an example: Which was N if we would have another point, P6, in (6,3)?

If we had P6 in (6,3), we would have the following N:

- Point 1: Minimum distance, closest point, point 2: 1. Second distance, point 5: 1. Finally, the distance to the third closest point, which is the chosen K, is to point 3: 2. Therefore, N = 3.
- Point 2: Minimum distance, point 1: 1. Second distance, point 5: 2. The distance to the third closest point, which is the K chosen, is to point 6: 3 (P6 could be the second one and P6 the third, but the result is the same). Therefore, N = 3.
- Point 3: Minimum distance, point 5: 1. Second distance, point 1: 2. The distance to the third closest point, which is the K chosen, is to point 2: 3 and point 6, which is also 3. Consequently, in this case, with K=3, N is not 3 but 4, N=4.
- Point 4: Minimum distance, point 2: 5. Second distance, point 1: 6. The distance to the third closest point, which is the K chosen, is to point 5: 7, and P6 is also 7; therefore, in this case, K and N are also different because N = 4.
- Point 5: Minimum distance, point 1: 1. Second distance, point 3: 1. Distance to the third closest point, which is the K chosen, is to point 2: 2 and point 6, therefore, again K and N are different N=4.

Carry out step A.4 of the density algorithm for the identification of outliers from the previous exercise.

Calculation of the density, d, of each point (Figs. 10 and 11).

$$\text{density}\ (x_i, K) = \left(\frac{\sum_{x_j \in N(x_i,K)} \text{distance}\left(x_i, x_j\right)}{\text{cardinal}\ N(x_i, K)} \right)^{-1}$$

$$\text{P1} : d(x_1, 3) = \left(\frac{\text{distance}\ (x_1, x_2) + \text{distance}\ (x_1, x_5) + \text{distance}(x_1, x_3)}{\text{cardinal}\ N(x_1, 3)} \right)^{-1}$$

$$= \left(\frac{1 + 1 + 2}{3} \right)^{-1} = 0.75$$

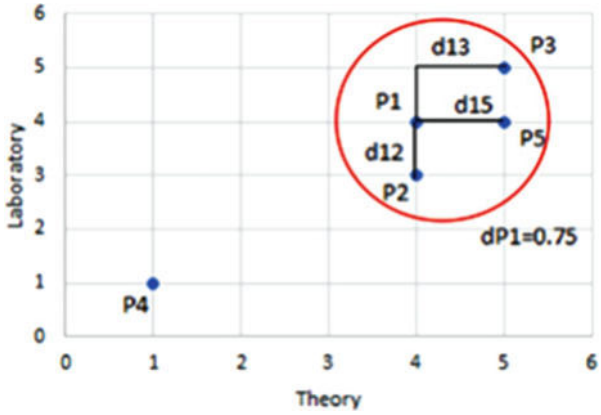


Fig. 10 Graphical representation of the density of P1

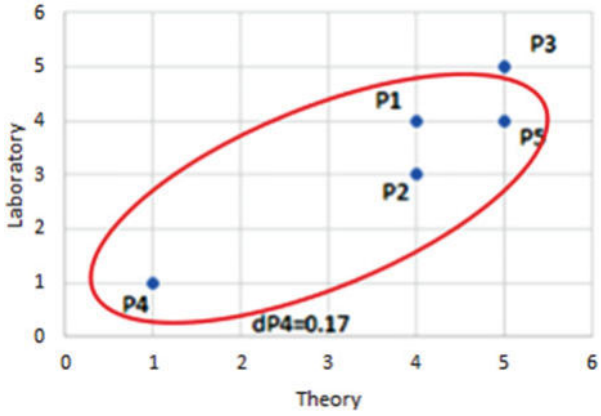


Fig. 11 Graphical representation of the density of P4

$$\text{P2} : d(x_2, 3) = \left(\frac{1+2+3}{3}\right)^{-1} = 0.5$$

$$\text{P3} : d(x_3, 3) = \left(\frac{1+2+3}{3}\right)^{-1} = 0.5$$

$$\text{P4} : d(x_4, 3) = \left(\frac{5+6+7}{3}\right)^{-1} = 0.17$$

$$\text{P5} : d\,(x_5, 3) = \left(\frac{1+1+2}{3}\right)^{-1} = 0.75$$

B. Calculation of the mean relative density, drm, of each point. There are different definitions of mean relative density. One of the most commonly used is

$$\text{mean relative density } (x_i, K) = \frac{\text{density } (x_i, K)}{\frac{\sum_{x_j \in N(x_i,K)} \text{density}(x_j, K)}{\text{cardinal } N(x_i, K)}}$$

This calculates the proportion between the density at a point and the mean of the densities of the set N that defines said point from the order number K. The mean relative density will tend to zero in the outliers.

The relative density, which takes into account the neighbourhood of the point, the set N, is used because if only the absolute density is used, outliers may not be correctly identified in data samples with regions of different densities.

Carry out step B of the density algorithm to identify outliers from the previous exercise.

The *drm* we have for the points are:

P1: $drm\ (x_1, 3) = \frac{\text{densidad } (x_1, 3)}{\frac{\text{densidad } (x_2, 3) + \text{densidad } (x_5, 3) + \text{distancia } (x_3, 3)}{\text{cardinal } N(x_1, 3)}} = \frac{0.75}{\frac{0.5+0.75+0.5}{3}} = 1.29$

P2: $drm\ (x_2, 3) = \frac{0.5}{\frac{0.75+0.75+0.5}{3}} = 0.75$

P3: $drm\ (x_3, 3) = \frac{0.5}{\frac{0.75+0.75+0.5}{3}} = 0.75$

P4: $drm\ (x_4, 3) = \frac{0.17}{\frac{0.5+0.75+0.75}{3}} = 0.26$

P5: $drm\ (x_5, 3) = \frac{0.75}{\frac{0.75+0.5+0.5}{3}} = 1.29$

C. Obtaining outliers, such as those points whose average relative density is significantly lower than that of the rest of the elements in the sample. Different methods can be set to establish when the drm is significantly lower.

Apply step C of the density algorithm to the previous exercise:

The *drm* we have for the points are P1: 1.29, P2: 0.75, P3: 0.75, P4: 0.26, P5: 1.29

Therefore, simply comparing them, it is observed that the drm of point 4, 0.26, is significantly lower than the rest. The mean would be 0.86, and the median would be 0.75.

Consequently, the only point with a density significantly lower than the rest of the points in the sample is Point 4 (1, 1), so it must be considered an outlier.

B. Computer-Based Solving

As in the other chapters, this section will address the use of software to solve the problem of anomaly detection analysis, but before applying R for solving this kind of problem, we introduce in this chapter an important issue related to the use of R as

is the use of packages in R, which is its most important feature and the thing that makes R as important as it is the use of packages, which will be introduced next.

The reason for introducing R Packages is that we have been using in the previous chapters the default installed packages to perform all the process that we need to solve the problems, but from this chapter, in the rest of the chapters, we will need additional packages to solve the problems that we will face, and this is the right moment to learn how we can use them.

R Packages

R works by using packages that contain the different functionalities that it can offer us. All published packages are documented.

R packages are installed in libraries, which are directories on the file system that contain a subdirectory for each package installed there.

R comes with a single library, R-4.0.3/library, which is the value of the R object '. Library' that contains the default and standard packages, but users can create other libraries and make use of the packages installed in them (or not) in an R session or in all R sessions.

R Default Packet Loading

When we start a session in R, a set of packages are loaded by default that give us the basic functionalities of R.

Although in the next chapter we are going to see the functions in detail, we have to anticipate what they are here, even briefly, because we are going to use some of them. The functions in R are preprogrammed instructions, that is, we will not have to program ourselves; they give us different results that we want to obtain and, in addition, they allow us to handle the operation of R.

To know the packages that R loads by default, we use the getOption () function. We remember that after the function, parentheses are always written to indicate the function's arguments, that is, what we want the function to do exactly. The getOption() function allows the user to set and examine a variety of global options that affect the way R calculates and displays its results. With the argument getOption ("defaultPackages"), we obtain the list of packages that are connected by default when R is started.

Therefore, we introduce the function

```
> getOption ("defaultPackages")
```

In addition, we obtain the list. There six (6) packages but there is one package more that it is the core package that R needs to run. It is the package *base*. To see that the base package can also be loaded, we can use the function search ( ). With the package base, the set of seven (7) packages are:

1. *base*: Contains the basic functions that allow R to function as a language: arithmetic, input/output, basic programming support, and so on. To obtain a complete list of the functions of a package, you can use the library () function, and as an argument, enter help = to the name of the package in quotes. The full instructions are as follows:

   > library (help = "base")

2. *datasets*: Contains a variety of datasets. To get a complete list you can use the instruction:

   > library (help = "datasets")

   If we write in the command line, the name of the dataset R displays the data. For instance, the Nile data

   > AirPassengers

3. *utils*: Contains a collection of utility functions. For example, the help () function opens the help on the word that we introduce as an argument.

   > library (help = "utils")

4. *grDevices*: Contains functions that support base and grid graphs.

   > library (help = "grDevices")

5. *graphics*: Contains functions for making basic graphics. To obtain a complete list of the graphics and functions in the graphics package, we use the library function again.

   > library (help = "graphics")

6. *stats*: Contains functions for statistical calculations and random number generation.

   > library (help = "stats")

7. methods: Contains automatically defined methods and classes for R objects and other programming tools. It allows programming under the object-oriented paradigm.

   > library (help = "methods")

Loading Packages from the R Standard Library

In addition to the seven packages that are loaded by default, R proposes another set of 23 recommended packages, which are found in R's standard library. This set consists of the following packages:

1. boot: Contains resampling-based data analysis functions known as bootstrap methods or computationally intensive methods.

   > library (help = "boot")

2. class: Contains unsupervised classification functions.
3. cluster: Contains cluster-based unsupervised classification functions.
4. codetools: Contains code analysis functions for R.
5. compiler: Contains the R compiler package.
6. foreign: Contains functions to read data stored by applications such as 'Minitab', 'S', 'SAS', 'SPSS', 'Stata', 'Systat', 'Weka', 'dBase', . . .
7. grid: Contains functions for making grid graphs.
8. KernSmooth: Contains Kernel-based nonparametric estimator functions.
9. Lattice: Contains functions to make more complex graphics. In particular, the package supports the creation of trellis charts: charts that show a variable or the relationship between variables, conditional on one or more variables.
10. MASS: Contains functions and datasets for the contents of Venables and Ripley's book *Modern Applied Statistics with S* (4th edition, 2002).
11. Matrix: Contains functions to handle matrixes.
12. mgcv: Contains functions to handle modeling performed with generalized additive modeling and the use of automatic smoothness selection.
13. nlme: Contains functions to treat models with linear and non-linear mixed effects
14. nnet: Contains functions to handle supervised classification with Feed-Forward Neural Networks and Multinomial Log-Linear Models.
15. parallel: Contains functions to support parallel computing in R.
16. rpart: Contains functions to perform supervised classification with Recursive Partitioning and Regression Trees
17. spatial: Contains functions to perform analysis based on Kriging and Point Pattern
18. splines: Contains functions to work with regression splines using the B-spline basis, the natural cubic spline basis.
19. stats4: Statistical functions using S4 classes
20. survival: Contains functions and databases to treat statistics applied to health.
21. tcltk: Contains language interface links to Tcl and the Tk GUI elements. Tcl (Tool Command Language) is a programming language suitable for web and desktop applications, networking, administration or testing. Tk is a set of graphical user interface tools. Tk is the standard GUI not only for Tcl but also for many other languages and can produce applications that run unchanged on Windows, Mac OS X, Linux, and more.
22. tools: Contains all the functions and tools necessary for the development of new packages.
23. translations: Contains the translation of the messages.

To load a package installed in the standard R library but not loaded by default, the following procedures are used:

- Using the Packages menu of the RGui. We select Load Package / Load Package, a window opens with the packages available in R's standard library, we select it, press OK, and we have it loaded. We can check it using the search () function, which outputs the list of packages we have loaded. To load a package using this method, we can load the Matrix package.
- Using the library () function and introducing in the argument the name of the package that we want to load. Again, we can check it by introducing the search () function. If we only introduce the library () function without an argument, it gives us a list of the packages available in the standard library. To load a package using this method, we can load the rpart package that we will use in the supervised classification chapter. The full instructions are as follows:

  >library(rpart)

  If we wanted to unload a package, we could use the function detach ( ). If we would like to unload rpart, the full instruction would be:

  >detach("package=rpart", unload=TRUE)

  We can introduce a search ( ) to see that the package rpart has been unloaded.

Install and Load Other R Packages

In addition to the packages included in the standard R library, there are a multitude of other packages that allow us to solve almost all types of defined data analysis. This external set of packages is constantly being developed and updated, with new packages being continuously published by the R community that address new data analysis problems and existing ones being updated and improved. These packages are found in different repositories, the main being the one in the CRAN.

The main R package repositories are as follows:

- CRAN
- R-Forge
- Omegahat
- BioC. They are Bioconductor-related packages for bioinformatics.

There are also package environments such as Tidyverse for Data Science that we will see in a later section.

Packages that are not found in the standard R library must first be installed in a library of the machine in which we are working with R, and once installed in said library, they can be loaded in R, that is, for these packages, it will be a two-step process: Install and load; we must not forget to load them once installed. However, before doing both, we must check that the package has been loaded by default with the function getOption ("defaultPackages"); if it is not, we must also check that it is

not in the standard R library with the function library (). If it is not, we proceed to install and load it following the instructions below.

To load an install and load a package that is not in the standard R library, the following procedures are used:[10]

- Using the Packages menu of the RGui. We select Select repositories to select which repository we are going to use to install the package, and the package we want to install must be in said repository. A window opens with the available repositories, which are the ones listed above; we select it, click OK, and we already have the repository selected. Next, we click on Install packages, which gives us a list of the packages available to install from the repository that we have selected in the previous step. We choose the package we want and click OK. We are going to test the loading of a package by loading LearningRlab. It gives us a message saying that it can install the package in the standard R library and that it is going to create a new library to install it; we say OK, and it gives us a new message to tell us in which directory will create the library, which will be in a subdirectory of the directory in which we are working that will call R, and within it will create another that will call win-library, and within this another that will call according to the version of R with which we are working, for example 3.4, and within this it will create a directory with the name of the package in which all the files of the package will be.

  In the process described in the previous paragraph, there is an intermediate step: When we have the CRAN selected as the repository or when we have not selected any, since the one that is selected by default is the CRAN, and we press Install packages, a previous window opens. to the list of installable packages, in which it asks us to select the CRAN mirror from which we want to install the package. It is the same window that opens when we click on the Set CRAN mirror option from the Packages menu.

  Once installed, we can load the package as we have done in the previous section with the standard library packages using the library () function and the name of the package as an argument, or with the Load Package option from the Packages menu. When we click Packages will see that in the window that opens, in addition to the standard library packages that appeared before, the new package that we have installed appears. We can check that it has been loaded correctly by using the search () function again.

  Before looking at other options for loading packages, let us finish looking at the RGui Packages/Packages menu options: The Load Packages option only loads the packages from the standard library or the packages that we have previously installed in the library itself. The option to select CRAN mirror

[10] Before starting to see how the packages are installed, it is important to note that there are certain packages that present installation problems in R x64, so it is interesting when R is loaded to also load the R i386 version.

previously selects the mirror we want to download the CRAN, and so when we want to install a CRAN package, it will no longer ask us from which mirror we want to install it. If we first select the CRAN repository before choosing the mirror, it will ask from which mirror we want to download it before opening the window with the list of packages. The option to select repositories allows us to select the repository from which we want to install the package, and we already know that depending on the repository we choose, we will be able to load some packages or others because the different repositories have different packages. The next option in the packages menu is to update package, if we press it will look for if any of the packages that we have installed has published a new version, it will show us the list and will ask us for confirming installation. If the confirmation is positive, it will install them.

- Another option to install packages is by using the install.packages ( ) function, which we can also find as the last option, the one we needed to see, from the Packages menu, and inside the parentheses, we put a single argument, that is, in quotes, the name of the package we want to install. When we press enter, a window will open for us to choose the mirror from which we want to install it, and when we press OK, it will install it. If the package has dependent packages, which are needed for its operation, it will install them automatically.
- Another option to install packages is to do it from a local directory of the machine with which we are working. To do this, we first have to save the package files we want to load on our machine. This has advantages as the total control of all referred with the package. Let's see how it is done from the CRAN repository but from any other repository it would be the same.

  We go to the CRAN page https://cran.rediris.es/s/ and in the left navigation menu click on the Packages link. We enter the Packages page, and as we know, we have two links: Table of available packages, sorted by date of publication, and Table of available packages, sorted by name. As we will know the name of the package, click on the second one, and the list of packages by name appears. At the top of the page, we have the alphabet to be able to access the package more quickly by its initial. Once we have the package, click on its link and enter its own web page. For example, we will enter the LearnClust page, which is a Learning Hierarchical Clustering Algorithms package, which is discussed in a later chapter. We go to L and click, and then click on LearnClust. We enter the LearnClust page titled LearnClust: Learning Hierarchical Clustering Algorithms. Below the title, we have a brief description of the package, and on the page, we have the following menu of options:

  First, there are 16 options that give us information about the package:

  1. Version: It gives us the most recent version of the package. 1.1
  2. Depends: Dependencies. Latest version of R that is required for the package to work. magick
  3. Suggest: Suggested packages for the package to work. knitr, markdown
  4. Published: Date of the last publication. 2020-11-29

5. Author: Authors. Roberto Alcantara [aut, cre], Juan Jose Cuadrado [aut], Universidad de Alcala de Henares [aut]
6. Maintainer: Maintainer of the package. Roberto Alcantara <roberto. alcantara at edu.uah.es>
7. License: Package license type. Unlimited.
8. NeedsCompilation: Indicates if the package needs compilation.
9. CRAN Checks: This gives us proof of the package. LearnClust results

The following are the links to the package downloads:

10. Reference manual: This gives us the operating reference manual of the package. LearnClust.pdf
11. Vignettes: They are vignettes or small documents that can be illustrated to explain the operation of the package. Learning Clusterization
12. Package Source: These are the files that have the source code of the package. LearnClust_1.1.tar.gz
13. Windows binaries: These are the installation files of the package in Windows. r-devel: LearnClust_1.1.zip, r-release: LearnClust_1.1.zip, r-oldrel: LearnClust_1.1.zip
14. OS X El Capitan binaries: These are the installation files of the package on Mac.
15. OS X Mavericks binaries: These are the installation files of the package on Mac. r-release (arm64): LearnClust_1.1.tgz, r-oldrel (arm64): LearnClust_1.1.tgz, r-release (x86_64): LearnClust_1.1.tgz, r-oldrel (x86_64): LearnClust_1.1.tgz
16. Old sources: These are the old versions of the package. LearnClust archive

Following are how to link the page from others:

> Please use the canonical form https://CRAN.R-project.org/package=LearnClust to link to this page.

Once we know what the package page looks like, we are going to download it to our machine, assuming we have Windows, if we click on the latest version of the package: r-devel: learnclust_1.1.zip. The.zip file is automatically downloaded to our download directory. It is interesting to also download the.pdf manual to be able to consult it, as well as to consult the vignettes.

We download both things in the downloads folder and return to R. To install the package we are going to use the install.packages () function again in which this time we are going to put two arguments: the first is which.zip file we want to install. It is important to bear in mind that for R to install it, the.zip file must be found in a temporary directory called tmp in the root directory of the hard disk, so we create it and place the file LearnClust_1.1.zip there. The second argument consists of giving the variable repos the value NULL so that it does not look for it in any repository,

since the files are on our machine. Consequently, the function is install.packages ("c:/tmp/LearnClust_1.1.zip", repos = NULL).

When I run the R function, as happened when loading it using the menu, it gives us a message saying that it can install the package in the standard R library and that it will create a new library to install it. We say yes, and it will do the same thing we saw when we installed it from the menu, that is, it will create a library in the same directory.

Next, we load the LearnClust package in R using the well-known library ( ) function. It is important to note that with this loading method, R will not automatically install the packages with which the LearnClust package has a dependency but will point them out to us, and we will have to load them in the same way manually, and it will not load it until the others are installed. To determine if it is installed, we execute the search () function.

We can also do so with the package's menu and the Install Packages from files option.

There may also be groups of associated packages because they facilitate work on certain types of studies, such as Tidyverse.

Tidyverse is an annotated collection of R packages designed for data science. All packages share an underlying philosophy of design, grammar, and data structures. All Tidyverse packages can be installed with a single installation instruction: install. packages ("tidyverse")

Modifying the Default Packet Load of R

As we have learned in a previous subsection, when we start R, there is a set of packages that are loaded by default, and to know what packages they are, we can use the function getOption ("defaultPackages"), but this set of packages loaded initially is not invariable and can be modified by reprogramming the start code.

The file that controls this startup code is Rprofile and is in the folder:

Program Files / R / R-3.3.2 / library / base / R /

In this file (We open it with the wordpad program, with the notepad it does not look good. If it does not let us open it in the R folder, we copy it to the desktop, we modify it and we paste it replacing the existing one in the R folder), there an instruction, which is:

dp <- c ("datasets", "utils", "grDevices", "graphics", "stats", "methods")

In this variable dp, we can include or remove the packages we want. If we remove all of them, only the "base" package would remain, which is not listed because it cannot stop loading for the system to work.

To see an example, we are going to include the package *foreign* within the packages by default because it will be used a lot normally when we work with

R. For which we introduce the word foreign after "methods" in the instruction that defines the variable dp:

dp <- c ("datasets", "utils", "grDevices", "graphics", "stats", "methods", "foreign").

To see if everything has gone well and the foreign package is loaded by default, we re-execute the getOption ("defaultPackages") or the search() instructions.

> getOption ("defaultPackages")

> search()

Anomaly Detection Exercises Solved in R

In this subsection, an anomaly detection analysis is be carried out by applying all the concepts seen in the topic and using the computer programming environment R.

The example used will be the same that used in the previous theoretical sections: The sample of the qualifications {Theory, Laboratory} of five students: 1. {4, 4}; 2. {4, 3}; 3. {5, 5}; 4. {1, 1}; 5. {5, 4}

In the previous chapters, all the data have been introduced in R by keyboard or simple text files, but that manner to introduce the data can be used only when few data are analyzed, how is the case of academic exercises for learning, in the real world all the data will be introduced in R for other procedures, with different levels of complexity. All facts related to the load of data to be analyzed in R and how to manage those data are under the scope of the Data Engineering Knowledge Area Group, which will be discussed in the third book of this series, but here we are going to see how to manage the charge load of data coming for some kind of data files, as for example, two of the more commonly used in a medium level of complexity, as Microsoft Excel; other extensions coming for other data analysis software solutions, like, for example, Minitab, SAS, SPSS, Stata, Systat, Weka, and dBase; or csv files. In this chapter, we introduce the reading of Excel files and Minitab, SAS, SPSS, Stata, Systat, Weka, and dBase files.

To be able to load an Excel file, with.xlxs format, we need to have loaded in R some additional package to the ones that R loads at the beginning; that allows loading Excel files; there are several, but none of them are in the standard library. Let us see the XLConnect package. To check if the package is loaded, we use the well-known search () function that we remember that it gives us the list of packages that we have loaded and we see that it is not loaded. To continue reviewing what we have learned, let us remember that to check if it is in the standard R library, we use the library () function and we see that it is not there, so we only have to install and load it in the program. To install it, we use the install.packages() function with the "XLConnect" attribute. The full instructions are as follows:

> install.packages ("XLConnect")

To work, the XLConnect package needs the rJava and XLConnectJars packages, which are downloaded simultaneously with XLConnect[11] when you run the install function.

To load it into the R session, we use the library () function. The full instructions are as follows:

> library (XLConnect)

Once we have XLConnect loaded and XLConnectJars has also been loaded, we can check that they are loaded using the search () function, and we can load the data with which we are working. For this, we must carry out two steps:

1. In the first one, we load all the data sheets that make up the Excel file through the loadWorkbook () instruction. As an attribute, we introduce the name of the file to load.

   To do this, we must generate an Excel file with the marks of the students. We will do it with two sheets, the first one will contain the data of the marks of theory that we have seen in the previous examples, and we rename that sheet in Excel as Theory; to do it we put the cursor over the name of the name Sheet click on the right button in the mouse and select change the name. It is important to bear in mind that the name of the variables must be entered in the first row; in this case Theory, we write the capital letter T. Data are:

   T
   4.00
   4.00
   5.00
   1.00
   5.00

   The second sheet, which we rename as Laboratory, will contain a list of the marks of laboratory, and we name the Data in the first row with the capital letter L. Data are:

[11] We may encounter problems when loading it related to the version of java that we are using on our machine, since, to work, the XLConnect package needs the rJava and XLConnectJars packages, which are downloaded simultaneously with XLConnect when executing the install function. packages. However, when we run the library function to load XLConnect, it will not work, as it will say that it cannot load XLConnect or XLConnectJars if we do not have the latest version of java loaded on our machine. If you have a 64-bit operating system, it is not enough to download the latest version of java that is downloaded by default; you have to look for the latest version for 64 bits; to do that you must go to the java webpage: https://www.java.com/en/and, click on the top of the page in *download*, and in the download page, you must click on the button below, which says *Agree and Start Free Download*.

Once you have the last version of Java installed in your computer, you must restart the computer and restart R and introduce again the instruction library(XLConnect); you must only introduce use library because the package is already installed in your personal library, and the package will be loaded in the R environment, and you can use it to load the Excel file.

L
4.00
3.00
5.00
1.00
4.00

Once the file is defined, we continue. We must assign a variable name to the load so that we can then select the different sheets of said data file, which we call marksxlsx. The full instructions are as follows:

> marksxlsx <- loadWorkbook ("marks.xlsx")

2. Once the Excel file is loaded, we read the sheet, to which we will assign a variable name, which, in this case, will be theorymarks, with which we want to work through the readWorksheet () instruction, which will take at least two arguments, the name of the variable of the Excel file preloaded and the name of the sheet, which we will select using the attribute sheet = "".

   To follow with the example, we are going to load both sheets, the first is the sheet with the marks of theory as the sheet in the Excel file has the name Sheet1, and the name of the Excel file preloaded is marksxlsx. We name the variable with the theory marks data theorymarksxlsx. Consequently, the complete instruction is:

   > theorymarksxlsx <- readWorksheet (marksxlsx, sheet = "Theory")

   and we obtain:

   T
   1 4
   2 4
   3 5
   4 1
   5 5

   If we introduce the variable theorymarksxlsx, we see all the data of the theory marks.

   > theorymarksxlsx

   If we change sheet to "Sheet2", we will have the laboratory marks, and we change the variable name to laboratorymarksxlsx. The full instructions are as follows:

   > (laboratorymarksxlsx = readWorksheet (marksxlsx, sheet = "Laboratory"))

   In this case, we have used the two parentheses at the beginning at the end of the instruction to immediately obtain the value of the variable without having to reintroduce its name. The result is:

```
L
1  4
2  3
3  5
4  1
5  4
```

We have already seen how files from Excel are loaded. Let us now see how to work with files from Minitab, SAS, SPSS, Stata, Systat, Weka, and dBase files. To be able to load all those types of files, we need to have loaded in R some additional package to the ones that R loads at the beginning and that allows loading them, but there is one package in the standard library, the package *foreign*, that allows us to load them.

To check if the package is loaded, we use the well-known search () function, which we remember that will give us the list of packages that we have loaded

```
>search()
```

and we see that it is not loaded. To continue reviewing what we have learned, let us remember that to check if it is in the standard R library, we use the library () function, and we see that it is, so we only have to load it in the program. To load it, we know that there are different methods, and we use the library () function. The full instructions are as follows:

```
> library (foreign)
```

Once we have foreign loaded, we use the different functions that the package has to read each different extension to load it in R. For example, if we would have the marks of theory and laboratory in an SPSS file, the function would be read.spss (), and introducing the name of the file that we want to load as an argument, we would have the data loaded in R. The instruction would be:

```
> (markssspss = read.spss ("marks.sav"))
```

Once we have loaded the data in R, we are going to perform with R the same different analysis that we have done in the theoretical section of the chapter, with the same data, and in consequence, the results must be the same.

Anomaly Detection Based on Statistics: Mean and Standard Deviation

To perform an outlier analysis with the mean and standard deviation method, it is not necessary to load any additional package in R, and since the data are already in R, because they are the same as in the previous case, we directly perform the analysis, which will only consist of programming the instructions used manually with functions already seen. The instructions are as follows:

Before starting, we must prepare the data. As we have loaded them with the instruction readWorksheet ( ) they have been loaded as a list, we can check this introducing the instruction type of ( ):

Following with the example that we are seeing and applying to the variable theorymarksxlsx, the instruction is:

> typeof (theorymarksxlsx)

To be able to perform calculations as the mean, we must have the data as a vector, with only numbers, without the name of the variable, T, because in other cases, it is impossible to perform the arithmetic calculations. To do that, we will use the function unlist ( ), which converts a list into a vector.

Applying the previous to the example, and, at the same time, to have easier variables to write, we rename the variable theorymarksxlsx, that is, the variable that we have used in the theoretical section to perform this kind of outlier analysis, as t. The instruction is:

>(t = unlist(theorymarksxlsx))

As we know, from theory, to obtain the interval of the normal values, we need to calculate the mean and standard deviation of the analyzed variable, but we can calculate them in the same manner in which we calculate the interval. We call the resulting vector with the limits of the interval int. To obtain the interval, the instruction is:

>(int <- c (mean ( ) -2 * sd ( ), mean ( ) + 2 * sd ( )))

If we apply the previous instruction to the variable Theory, t, of the example, we have:

>(int <- c (mean (t) -1.5 * sd (t), mean (t) + 1.5 * sd (t)))

The result is:

$$(1.34, 6.26)$$

How in the theoretical section of the chapter we have obtained the result:

$$(0.59, 6.00)$$

We can see that the same interval is not obtained as in hand calculations because the sd() function divides the data by n-1, not by n. If we want to obtain the same as in the calculations by hand, we have to define and obtain another standard deviation, as follows:

>sdd = sqrt (var ( ) * ((length ( ) -1) / length ( )))
and then perform the first statement using sdd

If we apply the previous correction to the calculation of the normal interval for theory we have:

```
>sdd = sqrt (var (t) * ((length (t) -1) / length (t)))
>(intdes = c (mean (t) -1.5 * sdd, mean (t) + 1.5 * sdd)))
```

The result is the same as in the theoretical section:

$$(0.59, 6.00)$$

Once we have obtained the interval of the normal values, we must apply it to all the values in the sample set to know if any one of them is outside the interval, because in that case, it must be identified as an outlier. To do that for all the values, we need two important control flows of programming, the loop and the conditional statements.

The loop will be solved with a *for* whose syntax in R is:

```
for (i in 1: length ())
```

it will allow us to apply the condition to belong to the calculated interval for all the data in the sample set.

The conditional will be solved with an *if*, whose syntax in R is:

```
If (logical condition)
{ instructions}
```

and it will be applied to each data to be pointed out if the data are outside the interval or will do nothing with it if it is inside the interval.

If we apply both statements to the example that we are solving, the solution is:

```
> for (i in 1: length (t))
```

This instruction makes that the following statements to be applied to all the data in the sample set between 1 and all the data in the sample, which comes with a length of t.

```
{if (t[i] <int [1] || t[i]> int [2])
```

t[i] is each one of the values of t, and int [1] is the lower value of the vector int, which, as we know, has two values, the lower limit of the interval of normal values and the upper limit of the interval, which comes with int [2].

```
{print ("the event"); print (i); print (t[i]); print ("it is an anomalous event or
    outlier")}}}
```

We can put several instructions in the same line separated by a, and they will be executed sequentially.

The result is:

```
"the event" 4 1 "it is an anomalous even or outlier"
```

Anomaly Detection Based on Statistics: Quartiles

To perform an outlier analysis with Quartiles method, it is not necessary to load any additional package in R. We can use the function quantile ( ) to obtain the values of the first and third quartiles and then obtain the interval of the normal values in transcribing to R the equation that we have shown in the theoretical section of the chapter in the following manner:

>int = c (quantile (, 0.25)-d * (quantile(, 0.75)-quantile(, 0.25)), quantile (,0.75 + d * quantile(, 0.75)-quantile(, 0.25))

where d is the degree of outlier and inside the quantile function; before the comma, we must indicate over which variable it is applied.

We apply the quartile method to identify the outliers of the laboratory marks, and we use the same degree of outlier that we used in the theoretical section of the chapter, that is, d = 1.5

The variable that we have for the laboratory marks was introduced in the R environment at the beginning of this section and is laboratorymarksxlsx. As happened with theorymarksxlsx, this a list variable that must be converted as a vector using the function unlist (), and, at the same time, we change its name to l. The instruction is:

>(l = unlist(laboratorymarksxlsx))

and to obtain the interval, we apply the equation above to the variable l, with the selected degree of d=1.5, in the following manner:

>int = c (quantile (l, 0.25)-1.5 * (quantile(l, 0.75)-quantile(l, 0.25)), quantile (l,0.75) + 1.5 * quantile(l, 0.75)-quantile(l, 0.25))

and the obtained result is:

$$(1.5, 5.5)$$

The result is the same as in the theoretical section:

$$(1.5, 5.5)$$

Once we have the interval, we use the same procedure with the loop and the conditional statements to identify the outliers than we have used in the previous case:

>for (i in 1: length (l))

this instruction makes that the following statements to be applied to all the data in the sample set between 1 and the all number of data in the sample, that comes with length of l.

{if (l[i] <int [1] || l[i]> int [2])

l[i] is each one of the values of l, and int [1] is the lower value of the vector int that, as we know, has two values, the lower limit of the interval of normal values and the upper limit of the interval, which comes with int [2].

{print ("the event"); print (i); print (l[i]); print ("it is an anomalous event or outlier")}}}

The result is:

"the event" 4 1 "it is an anomalous even or outlier"

Anomaly Detection Based on the Standard Error of the Residuals

To perform an outlier analysis for a regression, it is not necessary to load any additional package in R, and since the data are already in R because they are the same as in the previous cases, we directly perform the analysis. The first instruction will be the one necessary to obtain the regression, that means that is used the lm ( ) function. For a linear regression as we are trying to find the arguments of the function are, first, the dependent variable, and second, the dependent variable, separated by the [12]symbol ~, that is$y = f(x) \rightarrow y \sim x$,

As in the example of the theoretical section, we calculate the mark of laboratory as a function of the marks of theory, and we call the variable associated with the regression lft (laboratory as a function of theory). We do not need to load the data of the laboratory because we have introduced it in the previous exercises, but we must introduce the data of theory because we must remember that we change the first data from (2, 5) to allow us to have a function and see the outlier more clearly.[13] To do that, we only change the data using the assignation function. The first data in t are t[1] and l[1], and we assign the new value as follows:

>t[1]=2
>l[1]=5

Once we have changed the new vector t, the instruction of the regression is:

>(lft = lm (l ~ t))

The result is the same as that obtained in the theoretical section.

$$a = 1.9 \text{ and } b = 0.5$$

[12]If there was a problem to find this symbol in the keyboard its ASCII code is 126, which can be introduced writing ALT+126.

[13]The new dataset is: {Theory, Laboratory}: 1. {2, 5}; 2. {4, 3}; 3. {5, 5}; 4. {1, 1}; 5. {5, 4}.

Now, once the regression equation has been calculated, to obtain the standard error of the residuals, the residuals must be calculated so that the summary() function can be used. The instruction is as follows:

We apply the function summary to the lft regression to obtain the residuals:

>(summary (lft))

As a part of the result has been obtained the residuals:

Residuals:

2.12 −0.90 0.57 −1.36 −0.42

That is, taking into account the differences coming by the decimals, the same that we have obtained in the theoretical section

We can extract a vector that we will call res; with the values of the residuals for this output of the summary function, we use the instruction

>(res = summary( )$residuals)

We apply that function to the exercise

>(res = summary(lft)$residuals)

And from this vector, we calculate the value of the residual error by applying the equation observed in the theoretical section:

>sr = sqrt (sum (res^ 2) / length()))

We apply that instruction to the exercise

>(sr = sqrt (sum (res^2) / length(t)))

With a result of sr = 1.24
That it is exactly the same as in the theoretical section

To obtain the anomalous values, we program something analogous to the previous cases but with the standard error of the residuals system to identify the outliers: res > d*sr, where d is the degree of outlier that in the theoretical section we establish in d = 1.5.

If we apply it to the example, we have

>{for (i in 1: length (res))

Once we have the interval, we use the same procedure with the loop and the conditional statements to identify the outliers than we have used in the previous case:

>for (i in 1: length (l))

this instruction allows that the following statements were applied to all the data in the sample set between 1 and the all number of data in the sample, that comes with length of l.

{if (res[i]>1.5*sr)

l[i] is each one of the values of l, and int [1] is the lower value of the vector int, which, as we know, has two values, the lower limit of the interval of normal values and the upper limit of the interval, which comes with int [2].

{print ("the event"); print (i); print (res[i]); print ("it is an anomalous event or outlier")}}}

The result is:

"the event" 1 2.12 "it is an anomalous even or outlier"

Anomaly Detection Based on Proximity: K-Nearest Neighbor Algorithm

In this subsection, the K-nearest neighbor outlier detection technique will be used for detecting the anomaly if any of the students present abnormal marks, and we use the initial data without the correction in the first data used for the standard error of the residuals method.

As in the theoretical section, K will be 3 and the degree of outlier is 2.5, which means that those that move away from their third closest neighbor by a distance of 2.5 will be considered outliers. We are going to programme the algorithm for reason, and it is not necessary to load any additional package because the[14] list of R instructions or programme that solve this problem can be written using functions included in the packages loaded by the default; this list of instructions is:

First, to solve the problem, the first thing we need is to have the data in R, for which we use the matrix (( ) function, with dimension 2, 5, because in this case, we need both data in the pair, with two rows and five columns, and we assign the value m, from the word marks, so that the complete function is:

>m = matrix (c (4,4,4,3,5,5,1,1,5,4), 2,5)

Another possibility to obtain the matrix m is to use the cbind( ) function, which that allows us to join vectors, to join the current[15] t and l vectors, in one matrix. The instruction is:

>m2 = cbind(t, l)

Returning to the first instruction, we transpose the matrix with the function t (), so that the distances can be calculated correctly. If we do not transpose it, they will not

[14] As ever, the reader is encouraged to develop his/her own solution, list of instructions, programme, or script.

[15] We must remember to change the first data of t and l to his previous values (4,4), t[1] = 4, l[1] = 4.

be calculated well, and we have not entered it as a transpose because it is more convenient to enter it that way, and furthermore, errors are avoided. We follow the transposed matrix by calling x, so the complete function is:

>m = t(m)

Once the vectors have been introduced in R by means of the x matrix, the next step is to calculate the distances between them, which we do with the dist() function that calculates the distances between the vectors that make up the matrix that has been introduced as an argument, but as a result, it only gives the lower matrix and not a complete matrix, since, as it is symmetric, it does not give us the upper part. To avoid this and obtain a complete matrix, we must introduce the dist() function inside the as.matrix() function. The complete statement is:

>dm = as.matrix (dist (m))

However, we still do not have the definitive matrix of distances since, and although dm is presented in the form of a matrix, it is not interpreted by R as a matrix, so we convert it into a matrix interpretable by R with the instruction

>dmd= matrix (dm,5,5)

We have given it the name dmd to the definitive distance matrix.[16]

Once we have the distance matrix, we will identify those events or points whose 3rd closest neighbor is at a distance greater than 2.5. The first thing we have to do is realize that the distance matrix includes, on the main diagonal, the distance of each point with itself, which is 0 and will be the shortest distance, so we will always have to look for the K+1 nearest neighbor, since the first nearest neighbor will always be the point itself. Therefore, in this case, we will find the 4th closest neighbor. To do this, the first thing we will have to do is reorder the distance vectors that make up the matrix by magnitudes of the distances. To do this, we use the sort() function that orders the values of a vector from smallest to largest, resulting in a new vector. Thus, the complete statement will be:

>dmd [, i]=sort (dmd [, i])

which will result in a new md matrix in which all column vectors are arranged from top to bottom in increasing order.

However, as you can see, the value i has been left because we are going to use the two control statements to apply it at once to all the vectors that make up the matrix, since if the matrix were large, it would be impossible to apply it one by one to all vectors. To solve this problem, the repetition statement that we will use to solve is for, which is written as follows:

>for (i in 1: 5) {dmd [, i] = sort (dmd [, i])}

[16] If we do the same with the distance matrix coming from m2, we can see that finally both matrixes have the same appearance.

this statement would sort the five column vectors that we have, if we had n it would be in 1: n.

We will also use the same loop to find if any point has a distance to its "fourth" nearest neighbor greater than the outlier degree, which is 2.5, to identify said point as such. For this, we use the if statement. The complete statement will be:

```
> for (i in 1: 5) {if (dmd [4, i]> 2.5) {print (i); print ("it is an anomalous event or
  outlier")}}
```

Introducing the line break is essential for the statement to work, a statement that we include within the for, so that we will search for it at all points.

If we write together all the instructions in the loop, the final statement is:

```
>for (i in 1: 5) {dmd [, i] = sort (dmd [, i])
                              if (md [4, i]> 2.5) {print (i)
                              print ("is an anomalous event or outlier")}}
```

Entering the line breaks is essential for the statement to work,

With this code, R identifies the outliers using the closest K-neighbor technique.

Anomaly Detection Based on Density: Simplified Local Outlier Factor, LOF

In this subsection, the Local Outlier Factor detection technique will be used for detecting the anomaly if any of the students present abnormal marks.

In this case, the results are not equal, numerically, to those obtained in the theoretical section of the chapter because the R functions that will be used to identify the outliers implement the Local Outlier Factor method and not the Simplified one,[17] with the relative density used in theory, but the outliers data identified must be the same.

Nevertheless, as in the theoretical section, K will be 3, and the degree of outlier will be 2.5, which means that those that move away from their thirdclosest neighbor by a distance of 2.5 will be considered outliers. As in this case we haven't the results of the problem solved manually we are going to use two different packages that allows to calculate the LOF in other to be able to compare the results.

The first package will be *dbscan* and the second will be *DescTools*. We will use both of them in parallel. As we know, we must install and load both before to use them to perform the analysis. To do that, we use the instructions *install.packages* and *library*. First, we install both packages:

```
>install.packages("dbscan")
```

[17] The programming skills needed to implement the Simplified Local Outlier Factor are over the level introduced in the book, but we would like to encourage the readers to try to implement the algorithm by themselves.

>install.packages("DescTools")

Next, we load them:

>library(dbscan)

>library(DescTools)

Once we have the packages loaded we need to have the data in R, for which we create using the notepad the csv file marks.csv, with the data of the marks. The file is:

```
      T,L
Event1,4,4
Event2,4,3
Event3,5,5
Event4,1,1
Event5,5,4
```

We read the *marks.csv* file using the function read.csv in the variable *sample*. The instruction is as follows:

>sample<-read.csv("marks.csv")

The result is:

```
       T L
Event1 4 4
Event2 4 3
Event3 5 5
Event4 1 1
Event5 5 4
```

Now, we apply the function *lof( )* of dbscan to obtain the outliers of the sample. The attributes of the function are the data in the sample and the selected *minPts* = with the number of points depending on the K selected plus the point studied, that for this exercise, in the theoretical section as K was 3, *minPts*=4. Consequently, the instruction is:

>lof (sample, minPts=4)

The result is:

[1] 1.1081851 0.9069197 0.9069197 2.3007780 1.1081851

This establishes a very different local outlier factor for point 4 than for the others.

Let's apply now the function LOF ( ) of the DesTools package, which has as attributes the sample of data, the value of K as we saw in the theoretical section of the chapter. Consequently, the instruction is:

>LOF(sample, 3)

The result is:

[1] 1.1081851 0.9069197 0.9069197 2.3007780 1.1081851

Exactly the same result as in the previous solution.

C. Anomaly Detection Exercises Solved

This section has two parts. In the first part, a set of exercises solved in detail are presented to allow you to check if all the knowledge has been correctly acquired. The advice is to try to solve the exercises by yourself, and then to get the solution to check it with the proposed one by the book. This procedure will make this section truly useful for you. In the second part, the same exercises will be solved in R.

Hand Made Exercises

1. An analysis with the standard deviation method must be carried out to detect anomalous data using the density in a dataset that consists of the following seven values of resistance and density for different types of concrete: {resistance, density}: {3, 2; 3.5, 12; 4.7, 4.1; 5.2, 4.9; 7.1, 6.1; 6.2, 5.2; 14, 5.3}.

   To solve the exercise applying the mean and the standard deviation technique, the following steps are performed:

   Step 1. Calculate the arithmetic mean of the density values:

$$\overline{x}_T = \frac{\sum_{i=1}^{7} x_i}{7} = \frac{2 + 12 + 4.1 + 4.9 + 6.1 + 5.2 + 5.3}{7} = 5.66$$

   Step 2. From the mean, we calculate the standard deviation:

$$\begin{aligned} s_T &= \sqrt{\frac{\sum_{i=1}^{7} (x_i - \overline{x}_a)^2}{7}} \\ &= \sqrt{\frac{(2-5.66)^2 + 40.2 + 2.43 + 0.58 + 0.19 + 0.21 + 0.13}{7}} \\ &= \sqrt{8.16} = 2.86 \end{aligned}$$

   Step 3. From these results, and taking into account the degree of outlier taken of 1.5, the interval for normal data is:

$$(\overline{x}_a - ds_a, \overline{x}_a + ds_a) = (5.66 - 1.5 \cdot 2.86, 5.66 + 1.5 \cdot 2.86) = (1.37, 9.95)$$

Consequently, since the highest value is 12.74, it is outside the limits of the interval and, therefore, is an outlier.

2. An analysis with the quartile's method must be carried out to detect anomalous data using the data of the previous exercise.

To solve the exercise by applying the quartiles method, the following steps are applied:

Step 1. Determine the outlier degree, which in this case will be d = 1.5.
Step 2. Order the resistances by increasing value: {3, 3.5, 4.7, 5.2, 6.2, 7.1, 14}.
Step 3. Obtain the quartiles:

First quartile: n=7, c=1/4 in consequence n·c=7/4=1.75 $\notin \mathbb{N}$ and the equation to calculate the first quartile, where $[nc]$ non-decimal part of nc, is:

$$Q_1 = \tilde{x}_{\frac{1}{4}} = x_{[nc]+1} = x_{[1.75]+1} = x_{1+1} = x_2 = 3.5$$

And the third quartile: n=7, c=3/4 in consequence n·c =21/4=7.25 $\notin \mathbb{N}$ and the equation to calculate the third quartile is:

$$Q_3 = \tilde{x}_{\frac{3}{4}} = x_{[nc]+1} = x_{[7.25]+1} = x_{7+1} = x_8 = 7.1$$

From these results, and taking into account the degree of outlier taken of 1.5, the interval for normal data is:

$$\begin{aligned}&(Q_1 - 1.5(Q_3 - Q_1), Q_3 + 1.5(Q_3 - Q_1))\\ &\quad = (3.5 - 1.5(7.1 - 3.5), 7.1 + 1.5(7.1 - 3.6))\\ &\quad = (-1.9, 12.35)\end{aligned}$$

Consequently, since the largest value is 14, point $x_7 = 14$ is an outlier because it is out of the interval.

3. An analysis will be carried out to detect anomalous data on the regression of the variables, density as a function of resistance, using the standard error of the residuals in a dataset that consists of the following 7 values of resistance and density for different types of concrete: {resistance, density}: {3, 2; 3.5, 12; 4.7, 4.1; 5.2, 4.9; 7.1, 6.1; 6.2, 5.2; 14, 5.3}.

To solve the exercise applying the measures of analysis of outliers of a regression, the following steps must be applied:

Step 1. Calculate the regression line and obtain:

$$\frac{\sum_{i=1}^{n} f_i x_i}{\sum_{i=1}^{n} f_i} = 6.24 \quad \frac{\sum_{j=1}^{m} f_j y_j}{\sum_{j=1}^{m} f_j} = 5.66$$

$$s_{xy} = \frac{\sum_{i=1}^{n}\sum_{j=1}^{m} f_{ij} x_i y_j}{\sum_{i=1}^{n} f_i} - \left(\frac{\sum_{i=1}^{n} f_i x_i}{\sum_{i=1}^{n} f_i}\right) \cdot \left(\frac{\sum_{j=1}^{m} f_j y_j}{\sum_{j=1}^{m} f_j}\right)$$

$$= \frac{3.2+3,5.12+4,7.4,1+5,2.4,9+7,1.6,1+6,2.5,2+14.5,3}{7} - 6.24.5.66$$

$$= \frac{242,5}{7} - 35,32 = 34,64 - 35,32 = -0,68$$

$$s_x^2 = \frac{\sum_{i=1}^{n}(x_i - \overline{x})^2}{n} = \frac{(3-6.24)^2 + (3.5-6.24)^2 + \ldots + (14-6.24)^2}{7}$$

$$= \frac{84.42}{7} = 11.77$$

$$s_x = \sqrt{s_x^2} = 3.43$$

$$s_y^2 = \frac{\sum_{i=1}^{n}(y_i - \overline{y})^2}{n} = \frac{(2-5.66)^2 + (12-5.66)^2 + \ldots + (5.3-5.66)^2}{7}$$

$$= \frac{57.14}{7} = 8.16$$

$$s_y = \sqrt{s_y^2} = 2.85$$

$$r_{xy} = \frac{s_{xy}}{s_x s_y} = \frac{-0.68}{3.43 \cdot 8.16} = -0.024$$

$$b = \frac{s_{xy}}{s_x^2} = \frac{-0.68}{11.77} = r_{xy}\frac{s_y}{s_x} = -0.024\frac{2.85}{3.43} = -0.057$$

$$\mathrm{a} = \overline{y} - b\overline{x} = 5.66 + 0.057.6,24 = 6.015$$

$$v = 6.015 - 0.057t$$

Step 2. Calculate the residuals:

$$y_{ci} = 6.015 - 0.057x_i$$

$$y_{c1} = 6.015 - 0.057(3) = 5.84$$

$$y_{c2} = 6.015 - 0.057(3.5) = 5.82$$

$$y_{c3} = 6.015 - 0.057(4.7) = 5.75$$

$$y_{c4} = 6.015 - 0.057(3.2) = 5.72$$

$$y_{c5} = 6.015 - 0.057(7.1) = 5.61$$
$$y_{c6} = 6.015 - 0.057(6.2) = 5.66$$
$$y_{c7} = 6.015 - 0.057(14) = 5.22$$

From the vc_i residuals, the following are calculated:

$$r_1 = v_1 - vc_1 = 2 - 5.84 = -3.84$$

In the same way are calculated $r_2 \ldots r_7$, obtaining the following values:

$$r_2 = -0.02$$
$$r_3 = -2.98$$
$$r_4 = 0.52$$
$$r_5 = 1.59$$
$$r_6 = 3.21$$
$$r_7 = -1.11$$

Step 3. Calculate the standard deviation of the residuals:

$$s_r = \sqrt{\frac{\sum_{i=1}^{n}(y_i - y_{ci})^2}{n}} = \sqrt{\frac{(2-5.84)^2 + \ldots + (5.3-5.22)^2}{7}} = 2.85$$

Step 4. Calculate the limits of the interval for the outliers: Since the outlier degree is d = 1.5, the limits are:

$$d.s_r = 1{,}5.2.85 = 4.25$$

Step 5. Outlier identification, that is, if for any value:

$$|y_i - yc_i| > d.s_r = |y_i - yc_i| > 4.25$$

The point (3.5, 12) is identified as an outlier since

$$|12 - 5.82| = 6.18 > 5.7$$

4. For the data in a.txt file generated from the sample data of the following 11 values of response speeds and normalized temperatures of a microprocessor {speed, temperature}: {10, 7.46; 8, 6.77; 13, 12.74; 9, 7.11; 11, 7.81; 14, 8.84; 6, 6.08;

4, 5.39; 12, 8.15; 7, 6.42; 5, 5.73}, obtain the temperature outliers with the mean and the standard deviation technique.

To solve the exercise applying the mean and the standard deviation technique, the following steps are performed:

Step 1. Calculate the arithmetic mean:

$$\begin{aligned}\overline{x}_T &= \frac{\sum_{i=1}^{11} x_i}{11} \\ &= \frac{7.46+6.77+12.74+7.11+7.81+8.84+6.08+5.39+8.15+6.42+5.73}{11} \\ &= 7.5\end{aligned}$$

Step 2. From the mean, we calculate the standard deviation:

$$\begin{aligned}s_T &= \sqrt{\frac{\sum_{i=1}^{11}(x_i-\overline{x}_a)^2}{11}} = \sqrt{\frac{(7.46-7.5)^2+(6.77-7.5)^2+\ldots+(5.73-7.5)^2}{11}} \\ &= \sqrt{\frac{41.23}{11}} = \sqrt{3.75} = 1.93\end{aligned}$$

Step 3. From these results, and taking into account the degree of outlier taken of 1.5, the interval for normal data is:

$$(\overline{x}_a - ds_a, \overline{x}_a + ds_a) = (7.5 - 1.5 \cdot 1.93, 7.5 + 1.5 \cdot 1.93) = (4.61, 10.40)$$

Consequently, since the highest value is 12.74, it is outside the limits of the interval and, therefore, is an outlier.

5. For the data in a.txt file generated from the sample data of the following 11 values of response speeds and normalized temperatures of a microprocessor {speed, temperature}: {10, 7.46; 8, 6.77; 13, 12.74; 9, 7.11; 11, 7.81; 14, 8.84; 6, 6.08; 4, 5.39; 12, 8.15; 7, 6.42; 5, 5.73}, the outliers for the variable speed must be obtained using the quartiles algorithm.

   To solve the exercise by applying the quartiles method, the following steps are applied:

   Step 1. Determine the outlier degree, which in this case will be d = 1.5.
   Step 2. Order the speeds by increasing value: {4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14}.
   Step 3. Obtain the quartiles:

   First quartile: n=11, c=1/4 in consequence n·c=11/4=2.75 $\notin \mathbb{N}$ and the equation to calculate the first quartile, where [*nc*] nondecimal part of nc, is:

$$Q_1 = \tilde{x}_{\frac{1}{4}} = x_{[nc]+1} = x_{[2.75]+1} = x_{2+1} = x_3 = 6$$

And the third quartile: n=11, c=3/4 in consequence n·c=33/4=8.25 $\notin \mathbb{N}$ and the equation to calculate the third quartile is:

$$Q_3 = \tilde{x}_{\frac{3}{4}} = x_{[nc]+1} = x_{[8.25]+1} = x_{8+1} = x_9 = 12$$

From these results, and taking into account the degree of outlier taken of 1.5, the interval for normal data is:

$$(Q_1 - 1.5(Q_3 - Q_1), Q_3 + 1.5(Q_3 - Q_1)) = (6 - 1.5(12-6), 12 + 1{,}5(12-6)) = (-3, 21)$$

Consequently, since the largest value is 14, there are no outliers in the sample because all the values are inside the interval.

6. For the data of a.txt file generated from the sample data of the following 11 values of response speeds and normalized temperatures of a microprocessor {speed, temperature}: {10, 7.46; 8, 6.77; 13, 12.74; 9, 7.11; 11, 7.81; 14, 8.84; 6, 6.08; 4, 5.39; 12, 8.15; 7, 6.42; 5, 5.73}, obtain the outliers of the regression of speed as a function of temperature, using the standard error of the residuals.

   To solve the exercise applying the measures of analysis of outliers of a regression, the following steps must be applied:

   Step 1. Calculate the regression line and obtain:

$$\frac{\sum_{i=1}^{n} f_i x_i}{\sum_{i=1}^{n} f_i} = 9 \frac{\sum_{j=1}^{m} f_j y_j}{\sum_{j=1}^{m} f_j} = 7.5$$

$$\begin{aligned} s_{xy} &= \frac{\sum_{i=1}^{n}\sum_{j=1}^{m} f_{ij} x_i y_j}{\sum_{i=1}^{n} f_i} - \left(\frac{\sum_{i=1}^{n} f_i x_i}{\sum_{i=1}^{n} f_i}\right) \cdot \left(\frac{\sum_{j=1}^{m} f_j y_j}{\sum_{j=1}^{m} f_j}\right) \\ &= \frac{10 \cdot 7.46 + 8 \cdot 6.77 + 13 \cdot 12.74 + \cdots + 5 \cdot 5.73}{11} - 9 \cdot 7.5 \\ &= \frac{797.47}{11} - 67.5 = 72.5 - 67.5 = 5 \end{aligned}$$

$$s_x^2 = \frac{\sum_{i=1}^{n} (x_i - \overline{x})^2}{n} = \frac{(7.46-9)^2 + (6.77-9)^2 + \ldots + (5.73-9)^2}{11} = \frac{65.97}{11}$$

$$= 5.99$$

$$s_x = \sqrt{s_x^2} = 2.45$$

$$s_y^2 = \frac{\sum_{i=1}^{n}(y_i - \overline{y})^2}{n} = \frac{(10-7.5)^2 + (8-7.5)^2 + \ldots + (5-7.5)^2}{11} = \frac{134.75}{11} = 12.25$$

$$s_y = \sqrt{s_y^2} = 3.5$$

$$r_{xy} = \frac{s_{xy}}{s_x s_y} = \frac{5}{2.45 \cdot 3.5} = 0.583$$

$$b = \frac{s_{xy}}{s_x^2} = \frac{5}{5.99} = r_{xy}\frac{s_y}{s_x} = 0.583\frac{3.5}{2.45} = 0.835$$

$$\mathrm{a} = \overline{y} - b\overline{x} = 7.5 + 0.835 \cdot 9 = 15.015$$

$$v = 15.015 + 0.835t$$

Step 2. Calculate the residuals:

$$y_{ci} = 15.015 + 0.835x_i$$

$$y_{c1} = 15.015 + 0.835(7.46) = 21.24$$

$$y_{c2} = 15.015 + 0.835(6.77) = 20.67$$

$$y_{c3} = 15.015 + 0.835(12.74) = 25.65$$

$$y_{c4} = 15.015 + 0.835(7.11) = 20.95$$

$$y_{c5} = 15.015 + 0.835(7.81) = 21.54$$

$$y_{c6} = 15.015 + 0.835(8.84) = 22.39$$

$$y_{c7} = 15.015 + 0.835(6.08) = 20.09$$

$$y_{c8} = 15.015 + 0.835(5.39) = 19.52$$

$$y_{c9} = 15.015 + 0.835(8.15) = 21.82$$

$$y_{c10} = 15.015 + 0.835(6.42) = 20.38$$

$$y_{c11} = 15.015 + 0.835(5.73) = 19.79$$

From the vc_i residuals, the following are calculated:

$$r_1 = v_1 - vc_1 = 2 - 5.84 = -3.84$$

In the same way are calculated $r_2 \ldots r_{11}$, obtaining the following values:

$$r_2 = -0.02$$

$$r_3 = -2.98$$

$$r_4 = 0.52$$

$$r_5 = 1.59$$

$$r_6 = 3.21$$

$$r_7 = -1.11$$

Step 3. Calculate the standard deviation of the residuals:

$$s_r = \sqrt{\frac{\sum_{i=1}^{n}(y_i - y_{ci})^2}{n}} = \sqrt{\frac{(2-5.84)^2 + \ldots + (5.3-5.22)^2}{7}} = 2.85$$

Step 4. Calculate the limits of the interval for the outliers: Since the outlier degree is d = 1.5, the limits are:

$$d.s_r = 1{,}5.2.85 = 4.25$$

Step 5. Outlier identification, that is, if for any value:

$$|y_i - yc_i| > d.s_r = |y_i - yc_i| > 4.25$$

The point (3.5, 12) is identified as an outlier since

$$|12 - 5.82| = 6.18 > 5.7$$

7. For the data of a.txt file generated from the sample data of the following 5 seminars: {girls, boys}: 1. {9, 9}; 2. {9, 7}; 3. {11, 11}; 4. {2, 1}; 5. {11, 9}, the outliers must be obtained using the K-neighbors algorithm.

   The exercise must be solved by applying the K-neighbors algorithm to obtain the outliers. For which the steps of the algorithm will be applied:

   Step A.1. Select the outlier degree: d = 10
   Step A.2. Select the closest neighbor K: K = 3
   Step B.1. Calculate the Euclidean distances:

   Points 1-2, {{9,9}, {9,7}}:

$$d_{12} = \sqrt{\sum_{i=1}^{2}(p_i - q_i)^2} = \sqrt{(9-9)^2 + (9-7)^2} = 2$$

   Points 1-3, {{9.9}, {11.11}}:

$$d_{13} = \sqrt{(9-11)^2 + (9-11)^2} = 2.83$$

Points 1-4, {{9.9}, {2.1}}:

$$d_{14} = \sqrt{(9-2)^2 + (9-1)^2} = 10.63$$

Points 1-5, {{9.9}, {11.9}}:

$$d_{15} = \sqrt{(9-11)^2 + (9-9)^2} = 2$$

Points 2-3, {{9.7}, {11.11}}:

$$d_{23} = \sqrt{(9-11)^2 + (7-11)^2} = 4.47$$

Points 2-4, {{9.7}, {2.1}}:

$$d_{24} = \sqrt{(9-2)^2 + (7-1)^2} = 9.22$$

Points 2-5, {{9.7}, {11.9}}:

$$d_{25} = \sqrt{(9-11)^2 + (7-9)^2} = 2.83$$

Points 3-4, {{11,11}, {2,1}}:

$$d_{34} = \sqrt{(11-2)^2 + (11-1)^2} = 13.45$$

Points 3-5, {{11,11}, {11,9}}:

$$d_{35} = \sqrt{(11-11)^2 + (11-9)^2} = 2$$

Points 4-5, {{2,1}, {11,9}}:

$$d_{45} = \sqrt{(2-11)^2 + (1-9)^2} = 12.04$$

Step B.2. Order the distances of each point:

- Point 1. Minimum distance to points 2 and 5, distance equal to 2. The third is 3 and the distance is 2.83. Since the degree of outlier is $ d = 10 $, it is not an outlier.

- Point 2. Minimum distance to points 1 and 5, distance equal to 2 and 2.83, respectively. The third is the 3 and the distance is 4.47. Since the degree of outlier is $ d = 10 $, it is not an outlier.
- Point 3. Minimum distance to points 1 and 5, distance equal to 2.83 and 4.47, respectively. The third is 2 and the distance is 4.47. Since the degree of outlier is $ d = 10 $, it is not an outlier.
- Point 5. Minimum distance to points 3 and 1, distance equal to 2. The third is 2, and the distance is 2.83. Since the degree of outlier is $ K = 3 $ it is not an outlier.
- Point 4. Minimum distance to points 2 and 1, distance equal to 9.22 and 10.63, respectively. The third is 5, and the distance is 12.04. Since the outlier degree is $ d = $ 10, point 4 is an outlier.

8. For the data of a .txt file generated from the sample data of the following 6 seminars: {girls, boys}: 1. {9, 9}; 2. {9, 7}; 3. {11, 11}; 4. {2, 1}; 5. {11, 9}; 6. {11,7}, the outliers must be obtained, using the Simplified Local Outlier Factor the outliers with K=3.

   Carry out step A.1 of the density algorithm to search for anomalous data.
   Determination of the order number, or K, of the nearest neighbor used to calculate the density of each point. It is chosen arbitrarily.
   As it is chosen arbitrarily, taking into account the characteristics of the problem and of the sample, we take the third neighbor or closest event. K = 3.

   Carry out step A.2 of the density algorithm for the identification of outliers from the previous exercise.
   Calculation of Manhattan distances between all points. We calculate the distance of each point with the rest of the points in the sample.

1.{9, 9}; 2. {9, 7}; 3. {11, 11}; 4. {2, 1}; 5. {11, 9}

Points 1-2, {{9, 9}, {9, 7}}:$d_{12} = |x_{11} - x_{21}| + |x_{12} - x_{22}| = |9 - 9| + |9 - 7| = 2$

Points 1-3, {{9, 9}, {11, 11}}:$d_{13} = |9 - 11| + |9 - 11| = 4$

Points 1-4, {{9, 9}, {2, 1}}: $d_{14} = |9 - 2| + |9 - 1| = 15$

Points 1-5, {{9, 9}, {11, 9}}: $d_{15} = |9 - 11| + |9 - 9| = 2$

Points 2-3, {{9, 7}, {11, 11}}: $d_{23} = |9 - 11| + |7 - 11| = 6$

Points 2-4, {{9, 7}, {2, 1}}: $d_{24} = |9 - 2| + |7 - 1| = 13$

Points 2-5, {{9, 7}, {11, 9}}: $d_{25} = |9 - 11| + |7 - 9| = 4$

Points 3-4, {{11, 11}, {2, 1}}:$d_{34} = |11 - 2| + |11 - 1| = 19$

Points 3-5, {{11, 11}, {11, 9}}: $d_{35} = |11 - 11| + |11 - 9| = 2$

Points 4-5, {{2, 1}, {11, 9}}: $d_{45} = |2 - 11| + |1 - 9| = 17$

Carry out step A.3 of the simplified density algorithm for the identification of outliers from the previous exercise.

Sorting by distances of the neighbors of each point until reaching the defined K, 3, to calculate the N of each point.

- Point 1: Minimum distance, closest point, point 2: 2. Second distance, point 5: 2. Finally, the distance to the third closest point, which is the chosen K, is to point 3: 4. Therefore, N = 3.
- Point 2: Minimum distance, point 1: 2. Second distance, point 5: 4. Distance to the third closest point, which is the K chosen, is to point 3: 6. Therefore, N = 3.
- Point 3: Minimum distance, point 5: 2. Second distance, point 1: 4. Distance to the third closest point, which is the K chosen, is to point 2: 6. Therefore, N = 3.
- Point 4: Minimum distance, point 2: 13. Second distance, point 1: 15. Distance to the third closest point, which is the K chosen, is to point 5: 17. Therefore, N = 3.
- Point 5: Minimum distance, point 3: 2. Second distance, point 2: 4. The distance to the third closest point, which is the K chosen, is 1:9. Therefore, N = 3.

Carry out step A.4 of the density algorithm for the identification of outliers from the previous exercise.

Calculation of the density, d, of each point.

$$\text{density}\ (x_i, K) = \left(\frac{\sum_{x_j \in N(x_i,K)} \text{distance}\left(x_i, x_j\right)}{\text{cardinal}\ N(x_i, K)} \right)^{-1}$$

$$\text{P1: } d\ (x_1, 3) = \left(\frac{\text{distance}\ (x_1, x_2) + \text{distance}\ (x_1, x_5) + \text{distance}\ (x_1, x_3)}{\text{cardinal}\ N(x_1, 3)} \right)^{-1} = \left(\frac{2+2+4}{3} \right)^{-1} = 0.375$$

$$\text{P2: } d\ (x_2, 3) = \left(\frac{1+2+3}{3} \right)^{-1} = 0.25$$

$$\text{P3: } d\ (x_3, 3) = \left(\frac{1+2+3}{3} \right)^{-1} = 0.25$$

$$\text{P4: } d\ (x_4, 3) = \left(\frac{5+6+7}{3} \right)^{-1} = 0.067$$

$$\text{P5: } d\ (x_5, 3) = \left(\frac{1+1+2}{3} \right)^{-1} = 0.375$$

D. Calculation of the mean relative density, drm, of each point. There are different definitions of mean relative density. One of the most commonly used is:

$$\text{mean relative density}\ (x_i, K) = \frac{\text{density}\ (x_i, K)}{\frac{\sum_{x_j \in N\left(x_i,K\right)} \text{density}\left(x_j, K\right)}{\text{cardinal}\ N(x_i, K)}}$$

This calculates the proportion between the density at a point and the mean of the densities of the set N that defines said point from the order number K. The mean relative density will tend to zero in the outliers.

The relative density, which takes into account the neighborhood of the point, the set N, is used because if only the absolute density is used, outliers may not be correctly identified in data samples with regions of different densities.

The *drm* we have for the points are:

$$\text{P1: } drm\ (x_1, 3) = \frac{\text{densidad } (x_1, 3)}{\frac{\text{densidad } (x_2, 3) + \text{densidad } (x_5, 3) + \text{distancia } (x_3, 3)}{\text{cardinal } N(x_1, 3)}} = \frac{0.375}{\frac{0.25+0.375+0.25}{3}} = 1.284$$

$$\text{P2: } drm\ (x_2, 3) = \frac{0.25}{\frac{0.375+0.375+0.25}{3}} = 0.75$$

$$\text{P3: } drm\ (x_3, 3) = \frac{0.25}{\frac{0.375+0.375+0.25}{3}} = 0.75$$

$$\text{P4: } drm\ (x_4, 3) = \frac{0.067}{\frac{0.25+0.375+0.375}{3}} = 0.201$$

$$\text{P5: } drm\ (x_5, 3) = \frac{0.375}{\frac{0.375+0.25+0.25}{3}} = 1.284$$

E. Obtaining outliers, such as those points whose average relative density is significantly lower than that of the rest of the elements in the sample. Different methods can be set to establish when the drm is significantly lower.

Apply step C of the density algorithm to the previous exercise:

The *drm* we have for the points are P1: 1.284, P2: 0.75, P3: 0.75, P4: 0.201, P5: 1.284.

Therefore, simply comparing them, it is observed that the drm of point 4, 0.201, is significantly lower than the rest.

Consequently, the only point with a density significantly lower than the rest of the points in the sample is Point 4 (2, 1), so it must be considered an outlier.

Exercises Solved in R

In this section, the previous handmade exercises will be solved using the R software.

1. An analysis with the standard deviation method must be carried out to detect anomalous data using the density in a dataset that consists of the following seven values of resistance and density for different types of concrete: {resistance, density}: {3, 2; 3.5, 12; 4.7, 4.1; 5.2, 4.9; 7.1, 6.1; 6.2, 5.2; 14, 5.3}.

   To perform an outlier analysis with the mean and standard deviation method, it is not necessary to load any additional packages in R, and since the data are already in R, because they are the same as in the previous case, we will directly perform the analysis, which will only consist of programming the instructions used manually with functions already seen. The instructions are as follows:

   To obtain the interval:

   ```
   >(intdes = c (mean (sample $ d) -2 * sd (sample $ d), mean (sample $ d) +
     2 * sd (sample $ d)))
   ```

   The same interval is not obtained as in hand calculations because the sd () function divides the data by n-1, not by n. If we want to obtain the same as in the calculations by hand, we have to define and obtain another standard deviation, as follows.

```
sdd = sqrt (var (show \ $ d) * ((length (show \ $ d) -1)/length (show \ $ d)))
```

and then perform the first statement using sdd

```
(intdes = c (mean (shows \ $ d) -2 * sdd, mean (shows \ $ d) + 2 * sdd))
```

To obtain the outliers, we do the same as in the previous cases.

```
{for (i in 1: length (sample \ $ d))
        {if (shows \ $ d [i] <intdes [1] || shows \ $ d [i]> intdes [2])
        {print ("the event"); print (i); print (show \ $ d [i]); print ("it is an
    anomalous event or outlier")}}}
```

2. An analysis with the quartile's method must be carried out to detect anomalous data using the data of the previous exercise.

To perform an analysis of outliers with the box and whiskers method, it is not necessary to load any additional package in R, so we start by entering the data through a matrix that we are going to call sample, the function that we are going to use is \ textbf {matrix ()}. In this exercise, unlike the previous ones, we introduce the matrix in the way that interests us most so as not to make mistakes, but we transpose it with the same sentence, so the complete instruction is:

```
>(sample = t (matrix (c (3,2,3.5,12,4.7,4.1,5.2,4.9,7.1,6.1,6.2,5.2,14,5.3), 2,7,
  dimnames = list (c ("r", "d"))))))))
```

To perform the analysis, the matrix must be converted into a data frame using the *data.frame()* function, and the instruction is:

```
>(sample = data.frame (sample))
```

To perform the analysis of outliers with the box and whisker method, the function boxplot () is used.

```
>(boxplot (shows \ $ r, range = 1.5, plot = FALSE))
```

This problem can also be solved by calculating the quartiles, the first and third quartiles:

```
>(quar1r <-quantile (sample \ $ r, 0.25))

>(quar3r <-quantile (sample \ $ r, 0.75))
```

Then calculate the interval:

```
int = c (quart1r-1.5 * (quart3r-quart1r), quart3r + 1.5 * (quart3r-quart1r))
```

And to know if a piece of data is an anomaly or not, a for and if structure can be used where the *length ()* function has been used to leave the number of iterations open to the length of the vector, which would change with each dataset and the command || which means or.

```
for (i in 1: length (shows \ $ r))
        {if (show \ $ r [i] <int [1] || show \ $ r [i]> int [2])
        {print ("the event"); print (i); print (show \ $ r [i]); print ("it is an anomalous
event or outlier")}}}
```

3. An analysis will be carried out to detect anomalous data on the regression of the variables, density as a function of resistance, using the standard error of the residuals on the same sample from exercise 1.

To perform an outlier analysis for a regression, it is not necessary to load any additional package in R, and since the data are already in R, because they are the same as in the previous case, we will directly perform the analysis, which will consist only of programming the instructions used manually with functions already seen. The first instruction will be the one necessary to obtain the regression of density as a function of resistance. The full instructions are as follows:

```
> (dfr = lm (show \ $ d ~ show \ $ r))}
```

Then using the *summary ()* function, we obtain the residuals. The instruction is

```
>(summary (dfr))
```

To extract the vector from the residuals, we use the instruction:

```
>(res = summary (dfr) \ $ residuals)
```

From this vector, we calculate the value of the residual error:

```
>paragraph {} \ textbf {(sr = sqrt (sum (res \ ^ 2)/7))
```

To obtain the anomalous values, we program something analogous to the previous cases.

```
textbf {for (i in 1: length (res))
{if (res [i]> 2 * sr)
{print ("the event"); print (res [i]); print ("is an anomalous event or outlier")}}
```

4. For the data in a.txt file generated from the sample data of the following 11 values of response speeds and normalized temperatures of a microprocessor {speed, temperature}: {10, 7.46; 8, 6.77; 13, 12.74; 9, 7.11; 11, 7.81; 14, 8.84; 6, 6.08; 4, 5.39; 12, 8.15; 7, 6.42; 5, 5.73}, obtain the temperature outliers with the mean and the standard deviation technique.

To perform an outlier analysis with the mean and standard deviation method, it is not necessary to load any additional packages in R, and since the data are already in R, because they are the same as in the previous case, we will directly perform the analysis, which will only consist of programming the instructions used manually with functions already seen. The instructions are as follows:

To obtain the interval:

>(intdes = c (mean (sample $ t) -2 * sd (sample $ t), mean (sample $ t) + 2 * sd (sample $ t)))

The same interval is not obtained as in hand calculations because the sd () function divides the data by n-1, not by n. If we want to obtain the same as in the calculations by hand, we have to define and obtain another standard deviation, as follows.

sdd = sqrt (var (show \ $ t) * ((length (show \ $ t) -1)/length (show \ $ t)))

and then perform the first statement using sdd:

(intdes = c (mean (shows \ $ t) -2 * sdd, mean (shows \ $ t) + 2 * sdd))

To obtain the outliers, we do the same as in the previous cases.

{for (i in 1: length (sample \ $ t))
{if (shows \ $ t [i] <intdes [1] || shows \ $ t [i]> intdes [2])
{print ("the event"); print (i); print (show \ $ t [i]); print ("it is an anomalous event or outlier")}}}

5. For the data in a.txt file generated from the data of the sample used to carry out exercise 5, the outliers for the variable speed must be obtained using the quartiles algorithm.

As mentioned before, to perform an analysis of outliers with the box and whiskers method, it is not necessary to load any additional package in R, so we start by entering the data through a matrix that we are going to call sample, and the function that we are going to use is \ textbf {matrix ()}. We introduce the matrix in the way that interests us most so as not to make mistakes, but we transpose it with the same sentence, so the complete instruction is:

>(sample = t (matrix (c (10, 7.46, 8, 6.77, 13, 12.74, 9, 7.11, 11, 7.81, 14, 8.84, 6, 6.08, 4, 5.39, 12, 8.15, 7, 6.42, 5, 5.73), 2,11, dimnames = list (c ("s", "t")))))))))))

At this point, we can see that the function matrix runs perfectly without installing and loading the package Matrix, and we can ask ourselves why this is possible. This is because there are different matrix functions. We must remember that R is case sensitive to the function matrix, with an initial capital letter that is different from the function matrix. The first one is in the package Matrix, which must be installed and uploaded, and the second one, which is running perfectly without load any package, must belong to some package loaded by defect when R starts. To know which package is being used, we can use the following instruction:

>help(matrix) y obtenemos que pertenece al paquete base.

In the previous instruction entering the sample, we have also observed that we can obtain the output of any instruction putting it between brackets without the need to introduce again the name of the variable calculated.

To perform the analysis, the matrix must be converted into a data frame using the *data.frame()* function, and the instruction is:

>(sample = data.frame (sample))

We can see that the type of output changed.

In the data frame, the columns can have different types of data; in the matrix, all the data must have the same type.

To perform the analysis of outliers with the box and whisker method, the function boxplot () is used.

>(boxplot (sample$s, range = 1.5, plot = FALSE))

This problem can also be solved by calculating the quartiles, the first and third quartiles:

>(quar1r <-quantile (sample$s, 0.25))

>(quar3r <-quantile (sample$s, 0.75))

Then calculate the interval:

int = c (quart1r-1.5 * (quart3r-quart1r), quart3r + 1.5 * (quart3r-quart1r))

And to know if a piece of data is an anomaly or not, a for and if structure can be used where the *length ()* function has been used to leave the number of iterations open to the length of the vector, which would change with each dataset and the command ||, which means or.

```
for (i in 1: length (shows \ $ s))
        {if (show \ $ s [i] <int [1] || show \ $ s [i]> int [2])
        {print ("the event"); print (i); print (show \ $ s [i]); print ("it is an
    anomalous event or outlier")}}}
```

6. For the data of a.txt file generated from the data of the sample used to carry out exercise 4, obtain the outliers of the regression of speed as a function of temperature, using the standard error of the residuals.

   To perform an outlier analysis for a regression, it is not necessary to load any additional package in R, and since the data are already in R because they are the same as in the previous case, we will directly perform the analysis, which will consist only of programming the instructions used manually with functions already seen. The first instruction will be the one necessary to obtain the regression of speed as a function of temperature. The full instruction is:[18]

   > (sft = lm (sample$s ~ sample$t))}

[18]The symbol ~ can be obtained using the ASCII code with Bloq Num activado y ALT + 126

This is the speed as a function of temperature.

Then using the summary () function, we obtain the residuals. The instruction is

>(summary (sft))

To extract the vector from the residuals, we use the instruction:

>(res = summary (sft) \ $ residuals)

From this vector, we calculate the value of the residual error:

>paragraph {} \ textbf {(sr = sqrt (sum (res \ ^ 2)/11))

To obtain the anomalous values, we programme something analogous to the previous cases.

textbf {for (i in 1: length (res))
{if (res [i]> 2 * sr)
{print ("the event"); print (res [i]); print ("is an anomalous event or outlier")}}

7. For the data of a.txt file generated from the sample data of the following 5 seminars: {girls, boys}: 1. {9, 9}; 2. {9, 7}; 3. {11, 11}; 4. {2, 1}; 5. {11, 9}, the outliers must be obtained using the K-neighbors algorithm with K and an outlier degree of 5.

A degree of outlier of 5 means that those that move away from their third closest neighbor by a distance of 5 will be considered outliers. The degree of outlier has changed from the example observed previously because the size of the numbers that we are dealing with now has increased.

First, to solve the problem, the first thing we need is to have the data in R, for which we use the matrix (( ) function, with dimension 2, 5, because in this case, we need both data in the pair, with two rows and five columns, and we assign the value m, from the word marks, so that the function complete is:

>m = matrix (c (9,9,11,2,11, 9,7,11,1,9), 2,5, byrow=T)

Now, we transpose the matrix with the function t () so that the distances can be calculated correctly:

>m = t(m)

Once the vectors have been introduced in R by means of the x matrix, the next step is to calculate the distances between them, which we do with the dist() function that calculates the distances between the vectors that make up the matrix that has been introduced as an argument, but as a result, it only gives the lower matrix and not a complete matrix, since, as it is symmetric, it does not give us the upper part. To avoid this and obtain a complete matrix, we must introduce the dist() function inside the as.matrix() function. The complete statement is:

>dm = as.matrix (dist (m))

However, we still do not have the definitive matrix of distances since, although dm is presented in the form of a matrix, it is not interpreted by R as a matrix, so we convert it into a matrix interpretable by R with the instruction:

>dmd= matrix (dm,5,5)

Once we have the distance matrix, we will identify those events or points whose third closest neighbor is at a distance greater than 5. The first thing we have to do is realize that the distance matrix includes, on the main diagonal, the distance of each point with itself, which is 0 and will be the shortest distance, so we will always have to look for the K+1 nearest neighbor, since the first nearest neighbor will always be the point itself. Therefore, in this case, we will find the 4th closest neighbor. To do this, the first thing we will have to do is reorder the distance vectors that make up the matrix by magnitudes of the distances. To do this, we use the sort() function that orders the values of a vector from smallest to largest, resulting in a new vector. Thus, the complete statement will be:

>dmd [, i]=sort (dmd [, i])

which will result in a new md matrix in which all column vectors are arranged from top to bottom in increasing order.

However, as you can see, the value i has been left because we are going to use the two control statements to apply it at once to all the vectors that make up the matrix, since if the matrix were large, it would be impossible to apply it one by one to all vectors. To solve this problem, the repetition statement that we will use to solve is for, which is written as follows:

>for (i in 1: 5) {dmd [, i] = sort (dmd [, i])}

This statement would sort the five column vectors that we have; if we had n, it would be in 1: n.

We will also use the same loop to find if any point has a distance to its "fourth" nearest neighbor greater than the outlier degree, which is 2.5, to identify said point as such. For this, we use the if statement. The complete statement would be:

> for (i in 1: 5) {if (dmd [4, i]> 5) {print (i); print ("it is an anomalous event or outlier")}}

Introducing the line break is essential for the statement to work, a statement that we include within the for, so that we will search for it at all points.

If we write together all the instructions in the loop, the final statement is:

```
>for (i in 1: 5) {dmd [, i] = sort (dmd [, i])
                              if (md [4, i]> 5) {print (i)
                              print ("is an anomalous event or outlier")}}
```

8. For the data of a.txt file generated from the sample data of the following 5 seminars: {girls, boys}: 1. {9, 9}; 2. {9, 7}; 3. {11, 11}; 4. {2, 1}; 5. {11, 9}, the outliers must be obtained using the LOF algorithm with K and an outlier degree of 5.

As in the example, we use the dbscan and DesTools packages to obtain the outliers.

The data are already in the R environment because we have loaded them in the previous exercise. For that reason, we go directly to the lof () and LOF () functions.

We remember that both of them have the same first attribute, the name of the sample and the second for lof ( ) is the minPts= and for LOF K. The difference between them is that minPts include the treated point, and for that reason, we must increase the number of K in one. The instructions and their results are as follows:

>lof(m, minPts=4)

Result: 1.1081851 0.9069197 0.9069197 2.8554511 1.1081851

>LOF(m, 3)

Result: 1.1081851 0.9069197 0.9069197 2.8554511 1.1081851

Both results are the same and identify a clear LOF in point 4, which is the same that we obtained by applying the K-neighbors algorithm.

Unsupervised Classification

In this fifth chapter, we are going to see the theoretical foundations of *Unsupervised Classification* of events and the main techniques used to carry it out. As in all the previous chapters, it is structured into three sections.

Section A introduces, in a theoretical and, at the same time, practical way, all the basic theoretical knowledge related to unsupervised classification, that is, the concepts and techniques that allow us to perform the analysis, from the basic distance base clustering of events to hierarchical clustering.

Section B presents the computer-based solving of the same examples used in section A to introduce theoretical knowledge. Section B presents the computer-based solving. The packages needed to carry out these computational solutions are also introduced, and in this chapter, the RStudio tool that allows us to improve the work with R is introduced.[1]

Section C will consist of a set of statements of exercises about unsupervised classification in which detailed solutions can also be found.[2]

A. Theory

The first section of the chapter is structured into three subsections: 1. Introduction, 2. Unsupervised Classification based on distance, 3. Agglomerative Hierarchical Clustering.

[1] Is in this chapter, when the RStudio tool is introduced and not in the first one because the authors think that in first chapters the reader, or the student, must be totally focused in R itself more than in the tool used to work with R and this can be obtained better with the use of the RGui only. Once it has been obtained, it can be get a tool like RStudio that it is the adequate to work professionally with R.

[2] We repeat again here that in order to obtain the best results for the learning process throughout the use of the book, it is very important that the reader tries to solve the exercises by themself before seeing the solutions and that only once solved check if the obtained solutions are correct.

J. J. Cuadrado-Gallego, Y. Demchenko, *Data Analytics*,
https://doi.org/10.1007/978-3-031-39129-3_5

Introduction

Unsupervised classification studies seek to define, for a certain characteristic, elementary event, a set of groups of observations, events, with close values. These groups, called clusters, will allow that, based on the values of the characteristics, elementary events, that make up an event, it can be assigned as belonging to one of them. Each cluster is defined through its parameters, so it is essential to have a sample of events to be able to determine them. This sample will include the values of the elementary event or characteristic for which the classification is being sought.

The unsupervised classification of events is called Unsupervised or Clustering[3,4] because the values that define the different clusters are defined during the same classification process, while in the case of supervised classification, the events to be classified will be classified into classes whose values have been previously defined. In the case that the term classification is used without saying whether it is supervised or not, it usually refers to supervised classification.

Clustering of events. To introduce the concept of event clustering through an example, we use the grades of a subject from two groups[5] or sets of students. Each of the two groups was taught the same subject with two different pedagogical techniques and, without knowing which group each student belongs to, but only through their qualifications, it will be seen whether through a clustering analysis the group to which each student belongs. In this way, it will be possible to verify that teaching techniques have had an impact on student learning. The qualifications will be made up of two marks, corresponding to the theory and laboratory tests. The elementary events are each one of the notes individually E = {Theory, Laboratory}, which will have values from 0 to 5, where 5 will be the highest possible score and 0 the lowest. The groups sought in this case will be those that, depending on the values of the two elementary events, allow classifying an event in a given cluster, or what is the same, it is sought to know if based on the qualifications in theory In the laboratory, the student can be grouped into a specific teaching group that has not been indicated beforehand (it will be acted as if it was not known to which group the student belongs).

[3] From here and in the rest of the text, we will refer to anomalous data as anomalies or outliers since both terms are used interchangeably.

[4] In addition to the term clustering, the terms segmentation and partitioning are sometimes used to refer to unsupervised classification, but you have to be careful when using them as synonyms for clustering because they actually refer to different topics. Segmentation refers to the division of data into groups using a simple method, such as the selection of a certain value of an elementary event; and partitioning to graphics division techniques.

[5] Unlike what would happen in a supervised classification, in this case, there is no elementary variable or event, which could be, for example, "Group" that allows students to classify in advance the group to which they belonged. Although, at the end of the clustering study, that new variable could be defined.

Clustering can also be defined as a measure of the similarity or disparity between two events, applied to all pairs of events, and grouping similar events in the same cluster; similarity and disparity can be defined as follows:

- Similarity: Numerical measure of how similar two events (objects) are, ranging from 0 to 1.
- Disparity: Numerical measure of how much two objects differ; the lower bound is 0, and the upper bound is variable.

Both measures are usually calculated through:

Distances. Among the most commonly used are *Euclidean distance*. It is calculated through the equation:

$$d_{pq} = \sqrt{\sum_{i=1}^{n} (p_i - q_i)^2}$$

Or Minkowski. It is calculated through the equation:

$$d_{pq} = \sqrt[r]{\sum_{i=1}^{n} (p_i - q_i)^r}$$

where if $r = 1$, it is called Norm. If $r = 2$, it coincides with the Euclidean.

Densities. Defined as the number of points per unit volume, or *probability density*.

Different clustering techniques can be used to obtain the different groups or clusters. This differentiation is based on the fact that each of them uses different algorithms. All of them use a sample or set of events, for which the values of all elementary events are known to define the clusters. Once the groups are defined, they will be used to cluster new events.

Some of the best known and most used are as follows:

- Based on distance: K-means algorithm
- Agglomerative Hierarchical clustering

In the next subsections of this chapter, we are going to see how each one works in a specific way.

Unsupervised Classification Based on Distances

Different clustering techniques can be used to obtain the different groups or clusters. This differentiation is based on the fact that each of them uses different algorithms. All of them use a sample or set of events for which the values of all elementary events are known to define the clusters. Each data point will be assigned to the group whose centroid was at a lower distance.

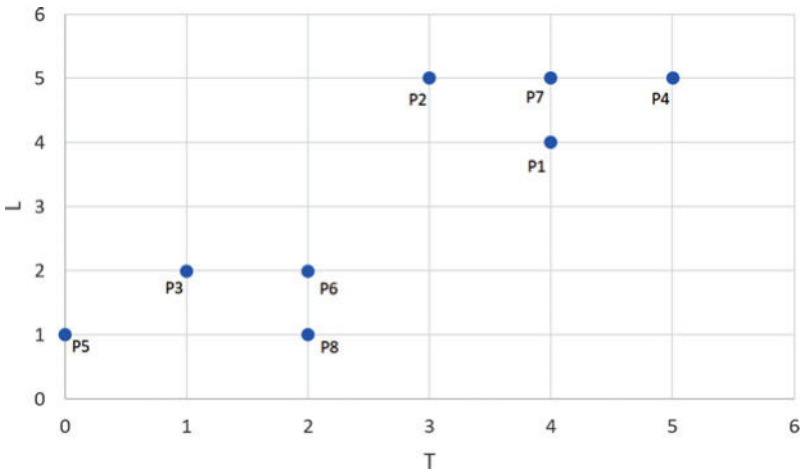


Fig. 5.1 Data representation

K-Means Algorithm

The definition of a set of clusters from the K-means technique follows a process of 2 to *n* steps that will be applied to a sample for which the values of all the elementary events that configure them are known; therefore, as mentioned above, it is essential to have such a sample to carry out the study.

To introduce the K-means technique for clustering through an example, we use the following sample of eight events, which the students mark: 1. {4, 4}; 2. {3, 5}; 3. {1, 2}; 4. {5, 5}; 5. {0, 1}; 6. {2, 2}; 7. {4, 5}; 8. {2, 1} (Fig. 5.1)

Let us see how each step is treated.

A. Step A[6] can, in turn, be separated into three substeps:

   1. Selection of the number K of clusters, in which the data will be grouped and the centroids will represent them. They are chosen arbitrarily by the user. The centroids will be the midpoints of the group of points (events) that make up the cluster.
      Selection of K and its centroids.[7] As we are dealing with only eight points, we are going to make the assumption that the number of clusters is not going to be high and we are going to start assuming that there are only two clusters.

[6]The same structure as in previous chapters will be used here for the steps and for the highest level it will be A, B, C, etc.

[7]To make the initial approximation to the number of clusters and their centroids, it is very useful to analyze the graphical representation of the data that is being analyzed.

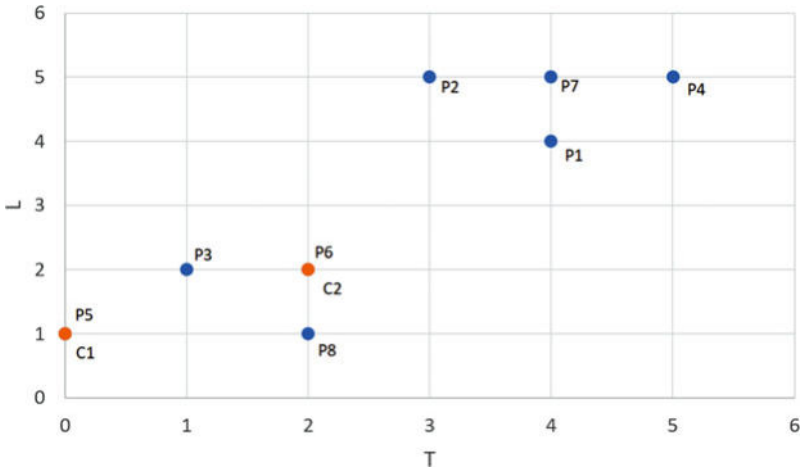


Fig. 5.2 Initial centroids in orange color

We take, arbitrarily, as centroids of both clusters the values: c1 = {0, 1} and c2 = {2, 2} (Fig. 5.2)[8]

2. Calculation of the Euclidean[9] distance from each point to each of the defined centers.
   Euclidean distances. We calculate the Euclidean distance from each point to the two defined centroids.

Point 1, {4, 4} (Fig. 5.3):

$$d_{1c_1} = \sqrt{\sum_{i=1}^{2} (p_i - q_i)^2} = \sqrt{(0-4)^2 + (1-4)^2} = 5$$

$$d_{1c_2} = \sqrt{(2-4)^2 + (2-4)^2} = 2.83$$

Point 2, {3, 5}:

[8] Although from the graphical representation of the data it can be intuited quite easily that these points are going to be far from what are ultimately the centers, we choose them to see how the algorithm corrects them to obtain the correct ones.

[9] As have been seen, the Euclidean distance between two points P and Q in an n-dimensional space is defined as $d_{PQ} = \sqrt{\sum_{i=1}^{n} (p_i - q_i)^2}$, where p_i and q_i are each of the elements of P and Q

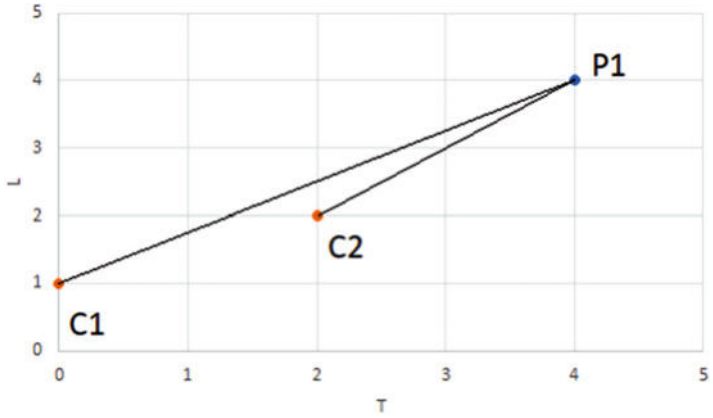


Fig. 5.3 Graphic with distance from point P1 (4,4) to C1 and C2

$$d_{2c_1} = \sqrt{(0-3)^2 + (1-5)^2} = 5$$
$$d_{2c_2} = \sqrt{(2-3)^2 + (2-5)^2} = 3.16$$

Point 3, {1, 2}:

$$d_{3c_1} = \sqrt{(0-1)^2 + (1-2)^2} = 1.41$$
$$d_{3c_2} = \sqrt{(2-1)^2 + (2-2)^2} = 1$$

Point 4, {5, 5}:

$$d_{2c_1} = \sqrt{(0-5)^2 + (1-5)^2} = 5$$
$$d_{2c_2} = \sqrt{(2-5)^2 + (2-5)^2} = 4.24$$

Point 5, {0, 1}:

$$d_{2c_1} = \sqrt{(0-0)^2 + (1-1)^2} = 0$$
$$d_{2c_2} = \sqrt{(2-0)^2 + (2-1)^2} = 2.23$$

Point 6, {2, 2}:

$$d_{2c_1} = \sqrt{(0-2)^2 + (1-2)^2} = 0$$
$$d_{2c_2} = \sqrt{(2-2)^2 + (2-2)^2} = 2.23$$

Point 7, {4, 5}:

$$d_{2c_1} = \sqrt{(0-4)^2 + (1-5)^2} = 5.66$$
$$d_{2c_2} = \sqrt{(2-4)^2 + (2-5)^2} = 3.61$$

Point 8, {2, 1}:

$$d_{2c_1} = \sqrt{(0-2)^2 + (1-1)^2} = 2$$
$$d_{2c_2} = \sqrt{(2-2)^2 + (2-1)^2} = 1$$

3. Assignment of points or events to clusters. With the results obtained in Step 2, a matrix of distances to the two centroids can be constructed in such a way that the distances to the centroids will go in the rows, the distance to the first centroid in the first row and the distance to the second centroid in the second row, and in each column one of the points; in the first column the first point and in the eighth column the eighth point. Once the distance matrix has been constructed, the cluster assignment matrix is constructed by assigning each point to each cluster, taking into account the distances to each centroid, in such a way that the point is assigned to the closest centroid. The columns will continue to have the same points as in the distance matrix and in each row a one if the point has been assigned to that centroid or a zero if it has not been assigned.
   Assignment to the cluster
   Taking into account the results of Step 2, the distance matrix is:

$$\begin{pmatrix} & P1 & P2 & P3 & P4 & P5 & P6 & P7 & P8 \\ C1 & 5 & 5 & 1.44 & 5 & 0 & 2.23 & 5.66 & 2 \\ C2 & 2.83 & 3.16 & 1 & 4.24 & 2.23 & 0 & 3.61 & 1 \end{pmatrix}$$

Starting from this matrix of distances, we construct the matrix of assignments. As seen in the theoretical description of how the assignment matrix is constructed, the eight points will continue to be in the columns in the same way as before and in each row a one if the point has been assigned to that centroid or a one if not assigned (Fig. 5.4):

$$\begin{pmatrix} & P1 & P2 & P3 & P4 & P5 & P6 & P7 & P8 \\ C1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ C2 & 1 & 1 & 1 & 1 & 0 & 1 & 1 & 1 \end{pmatrix}$$

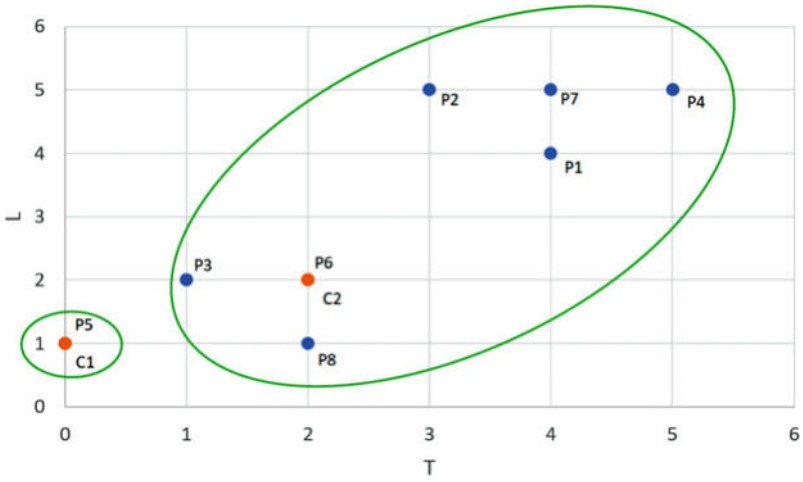


Fig. 5.4 First assignment

B. Step B and the rest of the m steps that are necessary to finish the analysis will have the same process that can be separated in turn into three substeps, in which the last two are equal to the last two substeps of Step A, but the first one is different:

   1. Recalculation of centroids. Now it is no longer done arbitrarily, but based on what was obtained in the first iteration (for the second iteration) or what happened in the previous iteration (for any other), the centroids are recalculated, giving them the mean[10] value of the points assigned to said centroid.
   Recalculation of centroids. Taking into account the assignment matrix, we see that only one point, the fifth (0, 1), has been assigned to the first cluster, whose centroid coincides with the point c1 = (0,1), so the first cluster, formed only by this point, does not change its centroid that coincides with the point. In the second cluster are the rest of the points, so it is necessary to recalculate the centroid of this cluster, making the average of the points that are in it, which is equal to (Fig. 5.5):[11]

$$c'_2 = \left(\frac{4+3+1+5+2+4+2}{7}, \frac{4+5+2+5+2+5+1}{7}\right) = (3, 3.43)$$

   2. Calculation of the Euclidean distance from each point to each of the defined centers.

[10] The mean value is calculated using the arithmetic mean.

[11] We repeat the sample values here to make the text easier to read: 1. {4, 4}; 2. {3, 5}; 3. {1, 2}; 4. {5, 5}; 5. {0, 1}; 6. {2, 2}; 7. {4, 5}; 8. {2, 1}

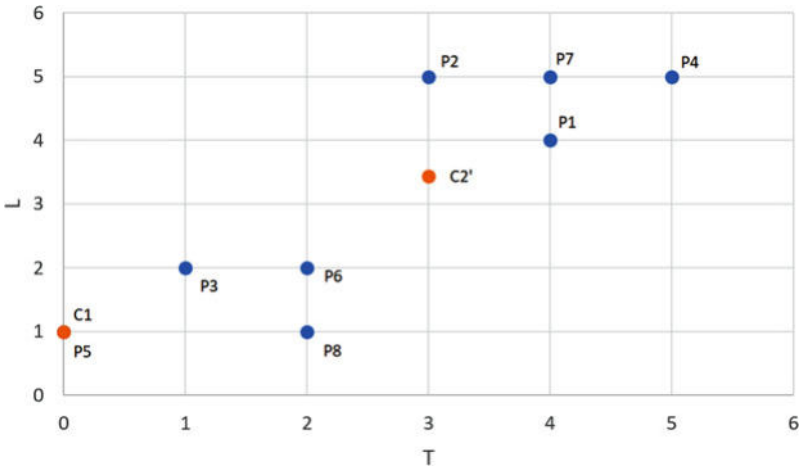


Fig. 5.5 Second centroids in orange color

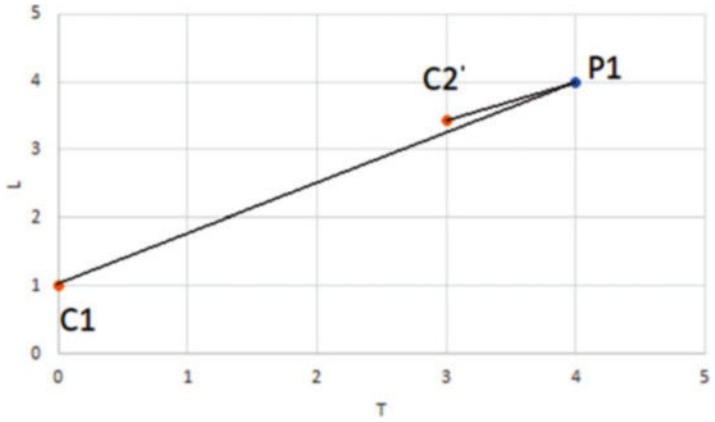


Fig. 5.6 Distance from point 1 to the second centroids

Euclidean distances. We calculate the Euclidean distance from each point to the new calculated centroids, which in this case is only $c_2 = (3, 3.43)$ because c_1 is the same.

Point 1, {4, 4} (Fig. 5.6):

$$d_{1c_1} = \sqrt{\sum_{i=1}^{2} (p_i - q_i)^2} = \sqrt{(0-4)^2 + (1-4)^2} = 5$$

$$d_{1c_2} = \sqrt{(3-4)^2 + (3.43-4)^2} = 1.15$$

Point 2, {3, 5}:

$$d_{2c_1} = \sqrt{(0-3)^2 + (1-5)^2} = 5$$
$$d_{2c_2} = \sqrt{(3-3)^2 + (3.43-5)^2} = 1.57$$

Point 3, {1, 2}:

$$d_{2c_1} = \sqrt{(0-1)^2 + (1-2)^2} = 1.41$$
$$d_{2c_2} = \sqrt{(3-1)^2 + (3.43-2)^2} = 2.46$$

Point 4, {5, 5}:

$$d_{2c_1} = \sqrt{(0-5)^2 + (1-5)^2} = 5$$
$$d_{2c_2} = \sqrt{(3-5)^2 + (3.43-5)^2} = 2.54$$

Point 5, {0, 1}:

$$d_{2c_1} = \sqrt{(0-0)^2 + (1-1)^2} = 0$$
$$d_{2c_2} = \sqrt{(3-0)^2 + (3.43-1)^2} = 2.54$$

Point 6, {2, 2}:

$$d_{2c_1} = \sqrt{(0-2)^2 + (1-2)^2} = 0$$
$$d_{2c_2} = \sqrt{(3-2)^2 + (3.43-2)^2} = 1.74$$

Point 7, {4, 5}:

$$d_{2c_1} = \sqrt{(0-4)^2 + (1-5)^2} = 5.66$$
$$d_{2c_2} = \sqrt{(3-4)^2 + (3.43-5)^2} = 1.86$$

Point 8, {2, 1}:

$$d_{2c_1} = \sqrt{(0-2)^2 + (1-1)^2} = 2$$
$$d_{2c_2} = \sqrt{(3-2)^2 + (3.43-1)^2} = 2.86$$

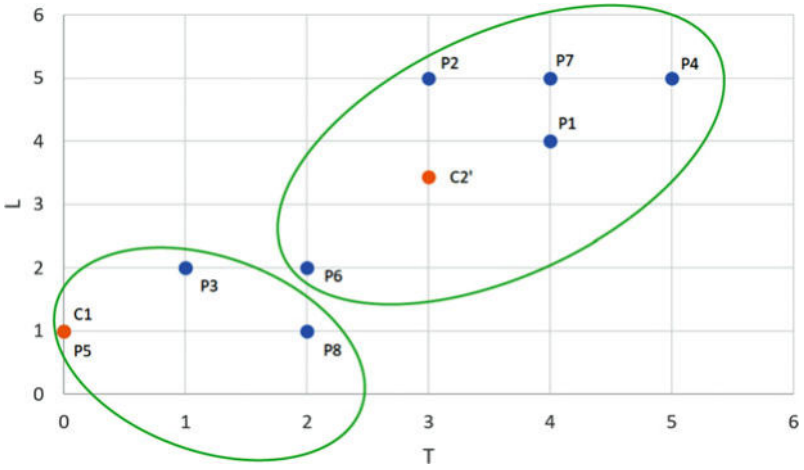


Fig. 5.7 Second assignment

Assignment to the cluster
Taking into account the results of Step 2, the distance matrix is:

$$\begin{pmatrix} & P1 & P2 & P3 & P4 & P5 & P6 & P7 & P8 \\ C1 & 5 & 5 & 1.41 & 5 & 0 & 2.23 & 5.66 & 2 \\ C2' & 1.15 & 1.57 & 2.46 & 2.54 & 2.54 & 1.74 & 1.86 & 2.63 \end{pmatrix}$$

Starting from this matrix of distances, we construct the matrix of assignments (Fig. 5.7).

$$\begin{pmatrix} & P1 & P2 & P3 & P4 & P5 & P6 & P7 & P8 \\ C1 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 1 \\ C2' & 1 & 1 & 0 & 1 & 0 & 1 & 1 & 0 \end{pmatrix}$$

C. As the assignment of the points to the clusters has changed in Step B, it is necessary to carry out at least one new Step C, the process of which will be absolutely the same as in Step B. We will repeat this until in one step the assignment of the points does not change from the previous step. The substeps in Step C will, therefore, be:

1. Recalculation of centroids. Taking into account the matrix of assignments, we see that now there is no longer a single point assigned to the first cluster, but there are three, points 3, 5, and 8. In the second cluster, they ranged from seven to five. Therefore, we recalculate the centroids with the new data (Fig. 5.8).

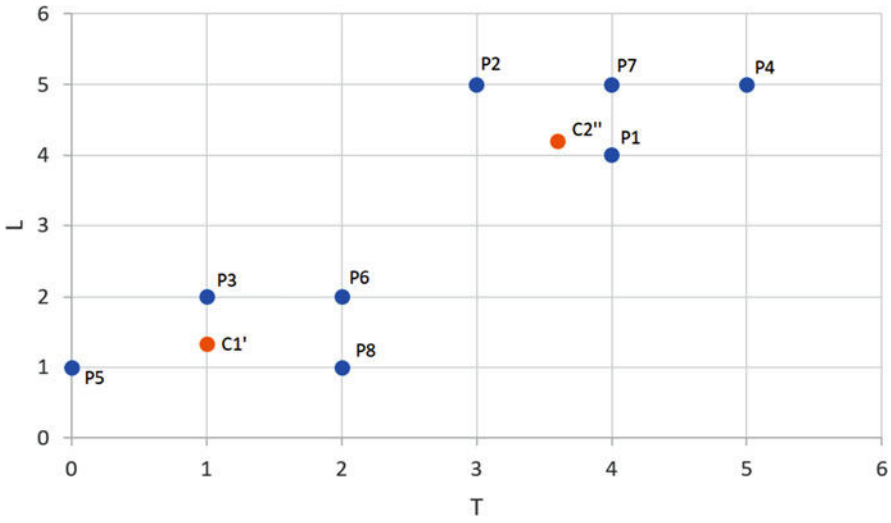


Fig. 5.8 Third centroids in orange color

$$c_1 = \left(\frac{1+0+2}{3}, \frac{2+1+1}{3}\right) = (1, 1.33)$$
$$c_2'' = \left(\frac{4+3+5+2+4}{5}, \frac{4+5+5+2+5}{5}\right) = (3.6, 4.2)$$

2. Calculation of the Euclidean distance from each point to each of the defined centers.

   Euclidean distances. We calculate the Euclidean distance from each point to the new calculated centroids, $c_1 = (1, 1.33)$ and $c_2 = (3.6, 4.2)$.
   Point 1, {4, 4} (Fig. 5.9):

$$d_{1c_1} = \sqrt{(1-4)^2 + (1.33-4)^2} = 4.02$$
$$d_{1c_2} = \sqrt{(3.6-4)^2 + (4.2-4)^2} = 0.45$$

   Point 2, {3, 5}:

$$d_{2c_1} = \sqrt{(1-3)^2 + (1.33-5)^2} = 4.18$$
$$d_{2c_2} = \sqrt{(3.6-3)^2 + (4.2-5)^2} = 1$$

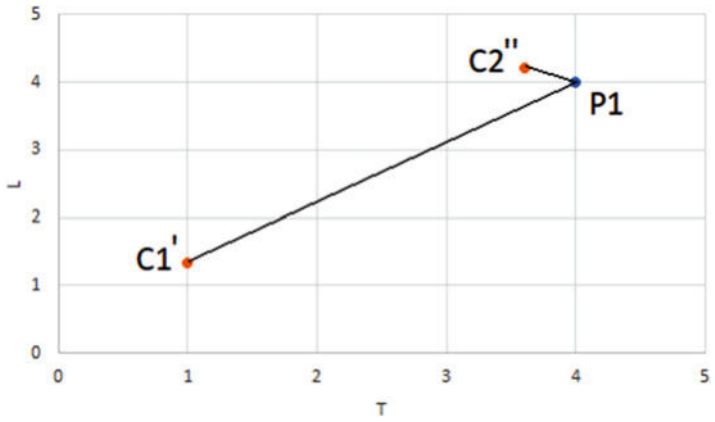


Fig. 5.9 Distance from P1 to the third centroids

Point 3, {1, 2}:

$$d_{2c_1} = \sqrt{(1-1)^2 + (1.33-2)^2} = 0.67$$
$$d_{2c_2} = \sqrt{(3.6-1)^2 + (4.2-2)^2} = 3.41$$

Point 4, {5, 5}:

$$d_{2c_1} = \sqrt{(1-5)^2 + (1.33-5)^2} = 5.43$$
$$d_{2c_2} = \sqrt{(3.6-5)^2 + (4.2-5)^2} = 1.61$$

Point 5, {0, 1}:

$$d_{2c_1} = \sqrt{(1-0)^2 + (1.33-1)^2} = 1.05$$
$$d_{2c_2} = \sqrt{(3.6-0)^2 + (4.2-1)^2} = 4.82$$

Point 6, {2, 2}:

$$d_{2c_1} = \sqrt{(1-2)^2 + (1.33-2)^2} = 1.2$$
$$d_{2c_2} = \sqrt{(3.6-2)^2 + (4.2-2)^2} = 2.72$$

Point 7, {4, 5}:

$$d_{2c_1} = \sqrt{(1-4)^2 + (1.33-5)^2} = 4.74$$
$$d_{2c_2} = \sqrt{(3.6-4)^2 + (4.2-5)^2} = 0.89$$

Point 8, {2, 1}:

$$d_{2c_1} = \sqrt{(1-2)^2 + (1.33-1)^2} = 1.05$$
$$d_{2c_2} = \sqrt{(3.6-2)^2 + (4.2-1)^2} = 3.58$$

3. Assignment of the points to the clusters
   Taking into account the results of Step 2, the distance matrix is:

$$\begin{pmatrix} & P1 & P2 & P3 & P4 & P5 & P6 & P7 & P8 \\ C1' & 4.02 & 4.18 & 0.67 & 5.43 & 1.05 & 1.2 & 4.74 & 1.05 \\ C2'' & 0.45 & 1 & 3.41 & 1.61 & 4.82 & 2.72 & 0.89 & 3.58 \end{pmatrix}$$

Starting from this matrix of distances, we construct the matrix of assignments (Fig. 5.10).

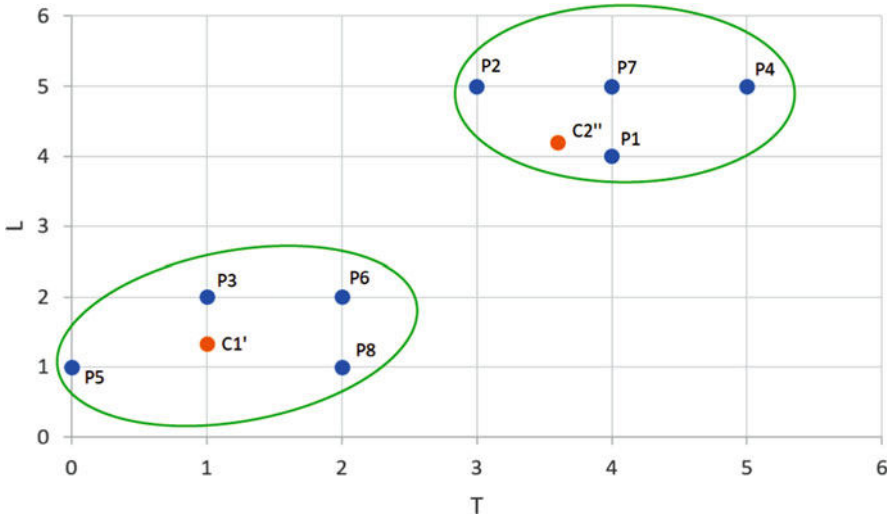


Fig. 5.10 Third assignment of the points

$$\begin{pmatrix} & P1 & P2 & P3 & P4 & P5 & P6 & P7 & P8 \\ C1' & 0 & 0 & 1 & 0 & 1 & 1 & 0 & 1 \\ C2'' & 1 & 1 & 0 & 1 & 0 & 0 & 1 & 0 \end{pmatrix}$$

C. As the assignment of the points to the clusters has changed in Step B, it is necessary to carry out at least one new Step C, the process of which will be absolutely the same as in Step B. We will repeat this until in one step the assignment of the points does not change from the previous step. The substeps in Step C will, therefore, be:

1. Recalculation of centroids. Taking into account the matrix of assignments, we see that now there is no longer a single point assigned to the first cluster, but there are four, points 3, 5, 6, and 8. In the second cluster, they ranged from seven to four. Therefore, we recalculate the centroids with the new data (Fig. 5.11).

$$c_1'' = \left(\frac{1+0+2+2}{4}, \frac{2+1+2+1}{4}\right) = (1.25, 1.5)$$
$$c_2''' = \left(\frac{4+3+5+4}{4}, \frac{4+5+5+5}{4}\right) = (4, 4.75)$$

2. Calculation of the Euclidean distance from each point to each of the defined centers.

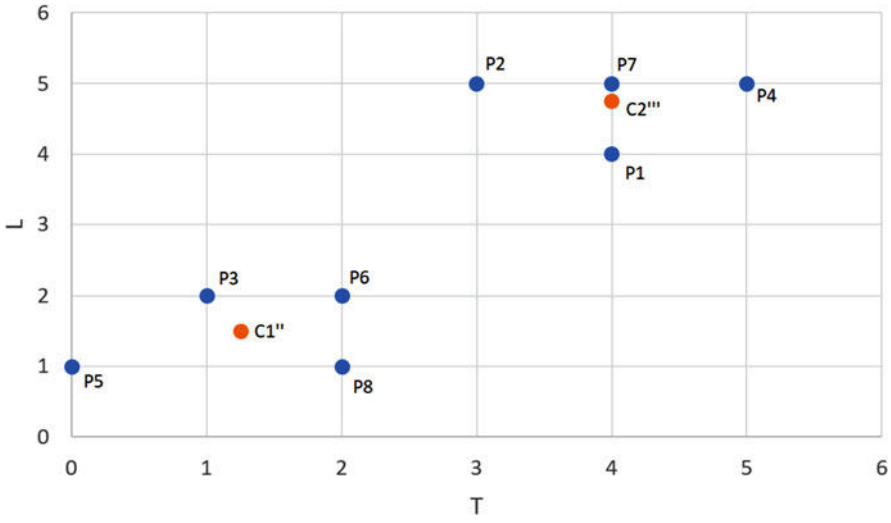


Fig. 5.11 Fourth centroids in orange color

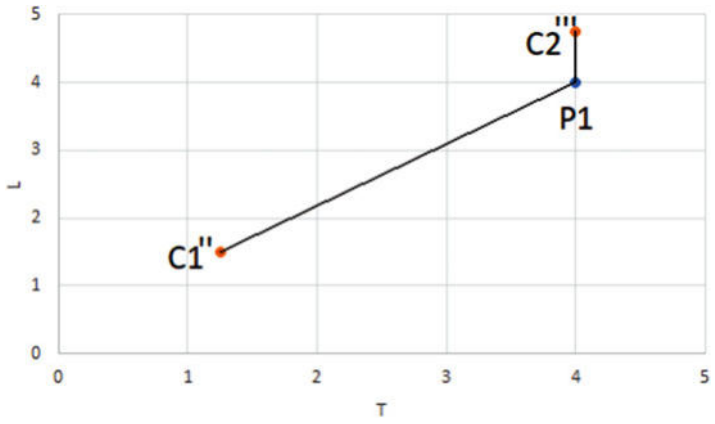


Fig. 5.12 Distance from P1 to the centroids

Euclidean distances. We calculate the Euclidean distance from each point to the new calculated centroids, $c''_1 = (1.25, 1.5)$ and $c'''_2 = (4, 4.75)$.

Point 1, {4, 4} (Fig. 5.12):

$$d_{1c_1} = \sqrt{(1.25-4)^2 + (1.5-4)^2} = 3.72$$
$$d_{1c_2} = \sqrt{(4-4)^2 + (4.75-4)^2} = 0.75$$

Point 2, {3, 5}:

$$d_{2c_1} = \sqrt{(1.25-3)^2 + (1.5-5)^2} = 3.91$$
$$d_{2c_2} = \sqrt{(4-3)^2 + (4.75-5)^2} = 1.03$$

Point 3, {1, 2}:

$$d_{2c_1} = \sqrt{(1.25-1)^2 + (1.5-2)^2} = 0.56$$
$$d_{2c_2} = \sqrt{(4-1)^2 + (4.75-2)^2} = 4.07$$

Point 4, {5, 5}:

$$d_{2c_1} = \sqrt{(1.25-5)^2 + (1.5-5)^2} = 5.32$$
$$d_{2c_2} = \sqrt{(4-5)^2 + (4.75-5)^2} = 1.03$$

Point 5, {0, 1}:

$$d_{2c_1} = \sqrt{(1.25-0)^2 + (1.5-1)^2} = 1.19$$
$$d_{2c_2} = \sqrt{(4-0)^2 + (4.75-1)^2} = 5.48$$

Point 6, {2, 2}:

$$d_{2c_1} = \sqrt{(1.25-2)^2 + (1.5-2)^2} = 1.19$$
$$d_{2c_2} = \sqrt{(4-2)^2 + (4.75-2)^2} = 3.4$$

Point 7, {4, 5}:

$$d_{2c_1} = \sqrt{(1.25-4)^2 + (1.5-5)^2} = 4.6$$
$$d_{2c_2} = \sqrt{(4-4)^2 + (4.75-5)^2} = 0.25$$

Point 8, {2, 1}:

$$d_{2c_1} = \sqrt{(1.25-2)^2 + (1.5-1)^2} = 1.19$$
$$d_{2c_2} = \sqrt{(4-2)^2 + (4.75-1)^2} = 4.25$$

3. Assignment of the points to the clusters
   Taking into account the results of Step 2, the distance matrix is:

$$\begin{pmatrix} & P1 & P2 & P3 & P4 & P5 & P6 & P7 & P8 \\ C1'' & 3.72 & 3.91 & 0.56 & 5.32 & 1.19 & 1.19 & 4.6 & 1.19 \\ C2''' & 0.75 & 1.03 & 4.07 & 1.03 & 5.48 & 3.4 & 0.25 & 4.25 \end{pmatrix}$$

Starting from this matrix of distances, we construct the matrix of assignments (Fig. 5.13).

$$\begin{pmatrix} & P1 & P2 & P3 & P4 & P5 & P6 & P7 & P8 \\ C1'' & 0 & 0 & 1 & 0 & 1 & 1 & 0 & 1 \\ C2''' & 1 & 1 & 0 & 1 & 0 & 0 & 1 & 0 \end{pmatrix}$$

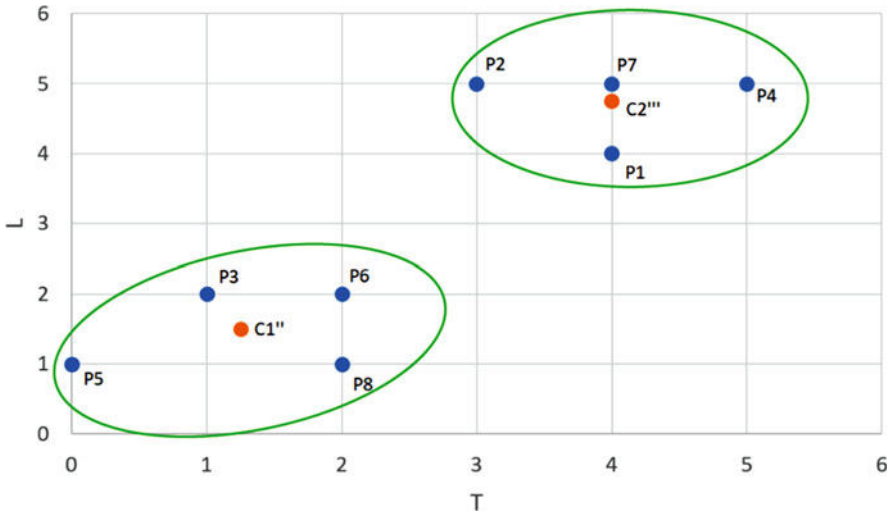


Fig. 5.13 Fourth assignment

As seen, no point has changed clusters, so the assignment of the points to the clusters has ended with this fourth step or iteration. The final clustering is:

- The events (points) belonging to the first cluster are 3 (1, 2), 5 (0, 1), 6 (2, 2), and 8 (2, 1), with the centroid at point $c_1 = (1.25, 1.5)$
- The points belonging to the second cluster are 1 (4, 4), 2 (3, 5), 4 (5, 5), and 7 (4, 5), with the centroid at point and $c_2 = (4, 4.75)$

To know if the new pedagogical techniques have had any effect on teaching, it would be necessary to see if students 3, 5, 6, and 8 belong to one of the groups and the rest to the other, since students who have similar grades are the ones that belong to each cluster.

Agglomerative Hierarchical Clustering

The definition of a set of clusters from the Agglomerative Hierarchical Clusterization technique follows a process of 2 to n steps, which will be repeated until there is only one cluster:

A. Step A: Obtain the matrix of Euclidean distances between clusters. In this step, the distance matrix will be calculated, whose values will be the distances from each cluster to the rest of the clusters.

   1. From the following sample: 1. {0.89, 2.94}; 2. {4.36, 5.21}; 3. {3.75, 1.12}; 4. {6.25, 3.14}; 5. {4.1, 1.8}; 6. {3.9, 4.27} carry out step A of the agglomerative hierarchical clustering algorithm (Fig. 5.14):

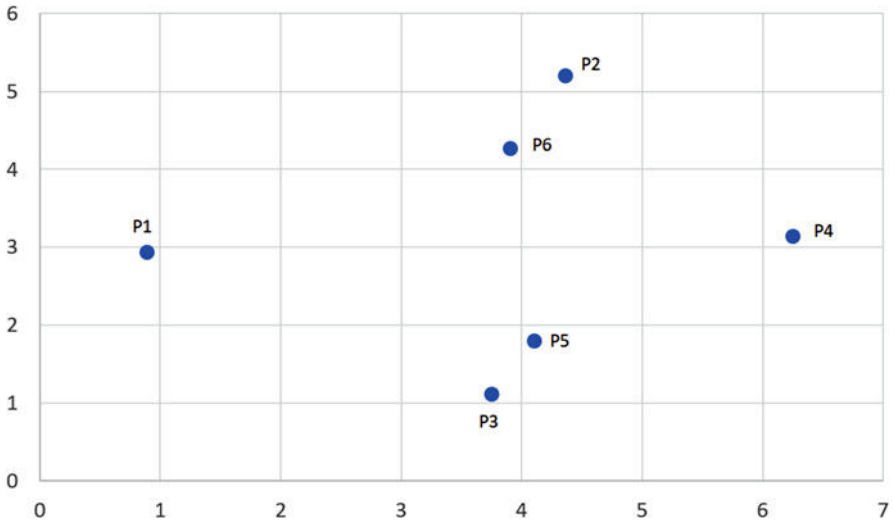


Fig. 5.14 Graphic of the points

The first step of each iteration is the calculation of the matrix of Euclidean distances between all the clusters, which in the case of the first iteration is the calculation of the distances between all the points because each point is a cluster.

You have to realize that the distance from 1 to 2 is the same as the distance from 2 to 1, so what we have are combinations of 6 elements, points, taken 2 by 2.

$$C_6^2 = \frac{6!}{2!(6-2)!} = \frac{6.5}{2} = 15$$

The Euclidean distances are (Fig. 5.15):

$$d_{12} = \sqrt{(0.89 - 4.36)^2 + (2.94 - 5.21)^2} = 4.15$$

$$d_{23} = \sqrt{(4.36 - 3.75)^2 + (5.21 - 1.12)^2} = 4.13$$

$$d_{13} = \sqrt{(0.89 - 3.75)^2 + (2.94 - 1.12)^2} = 3.39$$

$$d_{24} = \sqrt{(4.36 - 6.25)^2 + (5.21 - 3.14)^2} = 2.8$$

$$d_{14} = \sqrt{(0.89 - 6.25)^2 + (2.94 - 3.14)^2} = 5.36$$

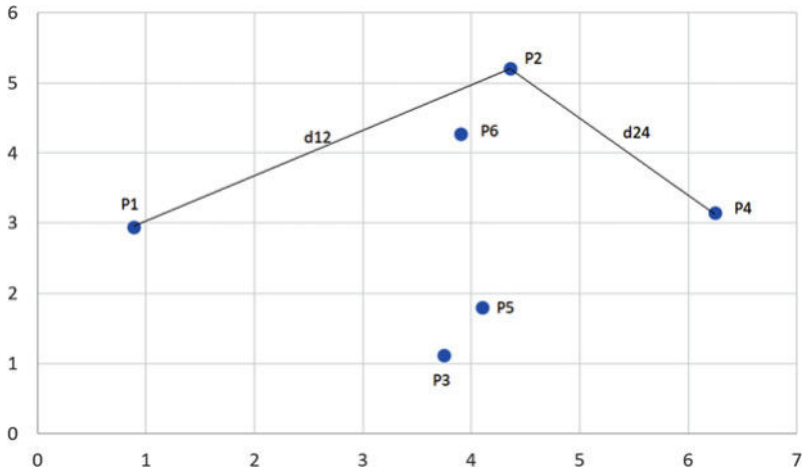


Fig. 5.15 Example of distances

$$d_{25} = \sqrt{(4.36 - 4.1)^2 + (5.21 - 1.8)^2} = 3.42$$

$$d_{15} = \sqrt{(0.89 - 4.1)^2 + (2.94 - 1.8)^2} = 3.41$$

$$d_{26} = \sqrt{(4.36 - 3.9)^2 + (5.21 - 4.27)^2} = 1.05$$

$$d_{16} = \sqrt{(0.89 - 3.9)^2 + (2.94 - 4.27)^2} = 3.29$$

$$d_{45} = \sqrt{(6.25 - 4.1)^2 + (3.14 - 1.8)^2} = 2.53$$

$$d_{34} = \sqrt{(3.75 - 6.25)^2 + (1.12 - 3.14)^2} = 3.21$$

$$d_{46} = \sqrt{(6.25 - 3.9)^2 + (3.14 - 4.27)^2} = 2.61$$

$$d_{35} = \sqrt{(3.75 - 4.1)^2 + (1.12 - 1.8)^2} = 0.76$$

$$d_{36} = \sqrt{(3.75 - 3.9)^2 + (1.12 - 4.27)^2} = 3.15$$

$$d_{56} = \sqrt{(4.1 - 3.9)^2 + (1.8 - 4.27)^2} = 2.48$$

From these results, the distance matrix is:

$$\begin{pmatrix} & p_1 & p_2 & p_3 & p_4 & p_5 & p_6 \\ p_1 & 0 & & & & & \\ p_2 & 4.15 & 0 & & & & \\ p_3 & 3.39 & 4.13 & 0 & & & \\ p_4 & 5.36 & 2.80 & 3.21 & 0 & & \\ p_5 & 3.41 & 3.42 & 0.76 & 2.53 & 0 & \\ p_6 & 3.29 & 1.05 & 3.15 & 2.61 & 2.48 & 0 \end{pmatrix}$$

Since it is a symmetric matrix, for clarity, only the lower part is shown.

B. Step B: Join the two closest clusters. In this step, the distances obtained will be ordered, and a new cluster will be generated joining the two closest clusters. In the first iteration, each individual point will be considered as a cluster.

What do we understand by proximity between clusters?

According to what we understand by proximity between clusters, we will have a different agglomerative hierarchical clustering algorithm. The different types of distance definitions for agglomerative hierarchical classification give different types of algorithms, and the types of agglomerative hierarchical classification algorithms according to the definition of proximity between clusters are:

- *MIN.* The proximity between two clusters is defined as the distance between the two closest points of the two clusters. It produces contiguous clusters, in which each point is closer to at least one point in its cluster than to any other point in another cluster. It is also called Single Link.

Over the sample that has been solved, carry out Step B of the agglomerative hierarchical clustering algorithm using the algorithm with the MIN proximity definition:

First Iteration: If we take the distance matrix between the clusters, we consider that in the first iteration, each point constitutes a cluster.

$$\begin{pmatrix} & p_1 & p_2 & p_3 & p_4 & p_5 & p_6 \\ p_1 & 0 & & & & & \\ p_2 & 4.15 & 0 & & & & \\ p_3 & 3.39 & 4.13 & 0 & & & \\ p_4 & 5.36 & 2.80 & 3.21 & 0 & & \\ p_5 & 3.41 & 3.42 & \mathbf{0.76} & 2.53 & 0 & \\ p_6 & 3.29 & 1.05 & 3.15 & 2.61 & 2.48 & 0 \end{pmatrix}$$

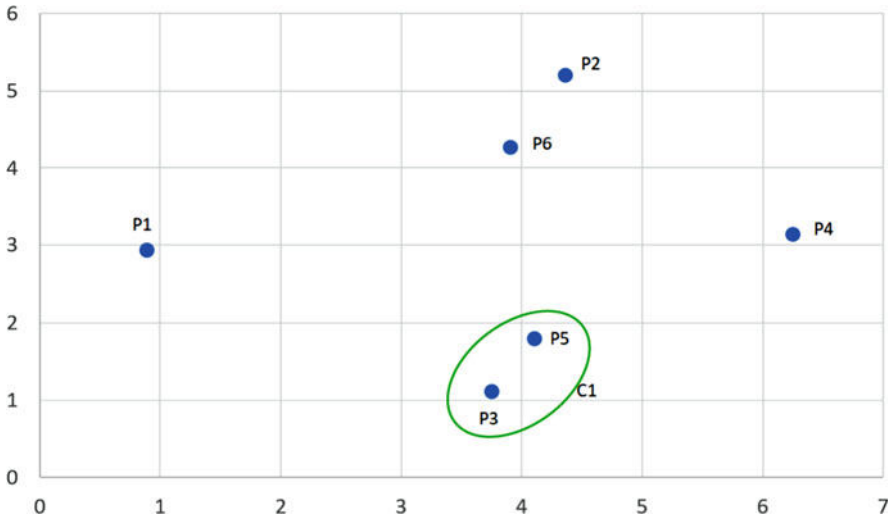


Fig. 5.16 First cluster MIN distances

The two closest clusters are 3 and 5. Therefore, the first cluster, C1, is the one formed by these two points (Fig. 5.16).

As we do not have a single cluster, we go to the second iteration.

Step A is carried out. Calculation of the matrix of distances between clusters.

The data are now as follows: 1. {0.89, 2.94}; 2. {4.36, 5.21}; 4. {6.25, 3.14}; 6. {3.9, 4.27} and C1 {3. {3.75, 1.12}; 5. {4.1, 1.8}}

The distance matrix is:

$$\begin{pmatrix} & p_1 & p_2 & p_4 & p_6 & C1_{p3} & C1_{p5} \\ p_1 & 0 & & & & & \\ p_2 & 4.15 & 0 & & & & \\ p_3 & 3.39 & 4.13 & & & 0 & \\ p_4 & 5.36 & 2.80 & 0 & & 3.21 & \\ p_5 & 3.41 & 3.42 & 2.53 & & 0 & 0 \\ p_6 & 3.29 & 1.05 & 2.61 & 0 & 3.15 & 2.48 \end{pmatrix}$$

The distance between clusters is now between the four points 1, 2, 4, and 6 and cluster 1, formed by points 3 and 5, in the previous iteration, and consequently, the distance between points 3 and 5 is now 0 because they are in the same cluster.

Step B is carried out. To identify the minimum distance between clusters. Data are now: 1. {0.89, 2.94}; 2. {4.36, 5.21}; 4. {6.25, 3.14}; 6. {3.9, 4.27} and C1 {3. {3.75, 1.12}; 5. {4.1, 1.8}}

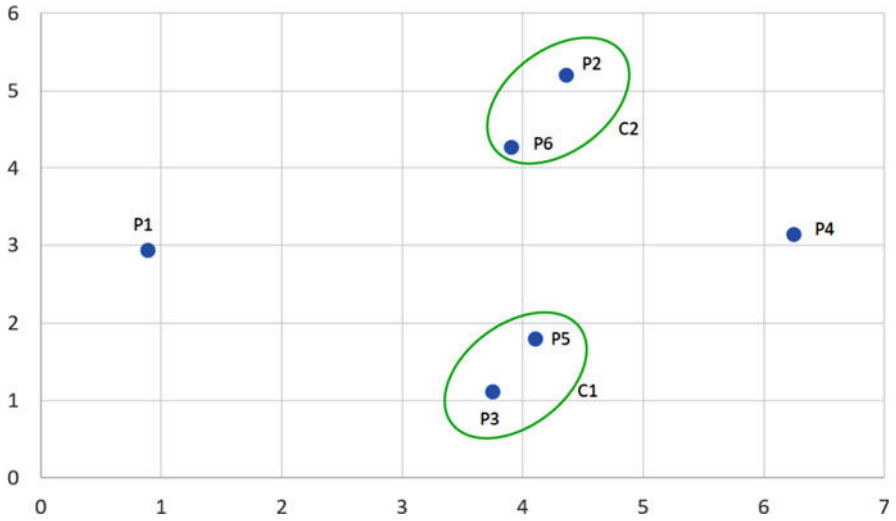


Fig. 5.17 Second cluster MIN distances

$$\begin{pmatrix} & p_1 & p_2 & p_4 & p_6 & C1_{p3} & C1_{p5} \\ p_1 & 0 & & & & & \\ p_2 & 4.15 & 0 & & & & \\ p_3 & 3.39 & 4.13 & & & 0 & \\ p_4 & 5.36 & 2.80 & 0 & & 3.21 & \\ p_5 & 3.41 & 3.42 & 2.53 & & 0 & 0 \\ p_6 & 3.29 & \mathbf{1.05} & 2.61 & 0 & 3.15 & 2.48 \end{pmatrix}$$

The two closest clusters are 2 and 6. Therefore, the second cluster, C2, is the one formed by these two points.

As we do not have a single cluster, we go to the third iteration (Fig. 5.17).

A new step A. Calculation of the matrix of distances between clusters is performed, and the data are now: 1. {0.89, 2.94}; 4. {6.25, 3.14}, C1 {3. {3.75, 1.12}; 5. {4.1, 1.8}}, and C2 {2. {4.36, 5.21}; 6. {3.9, 4.27}}. The new matrix of distances is:

$$\begin{pmatrix} & p_1 & p_4 & C1_{p3} & C1_{p5} & C2_{p2} & C2_{p6} \\ p_1 & 0 & & & & & \\ p_2 & 4.15 & & & & 0 & \\ p_3 & 3.39 & & 0 & & 4.13 & \\ p_4 & 5.36 & 0 & 3.21 & & 2.8 & \\ p_5 & 3.41 & 2.53 & 0 & 0 & 3.41 & \\ p_6 & 3.29 & 2.61 & 3.15 & 2.48 & 0 & 0 \end{pmatrix}$$

The distance between clusters is now between the two points 1, 4 and the clusters C1, formed by points 3 and 5, and C2, identified in the previous iteration, and consequently, the distance between points 2 and 6 is now 0 because they are in the same cluster.

A new Step B is performed to obtain the two closest clusters.

$$\begin{pmatrix} & p_1 & p_4 & C1_{p3} & C1_{p5} & C2_{p2} & C2_{p6} \\ p_1 & 0 & & & & & \\ p_2 & 4.15 & & & & 0 & \\ p_3 & 3.39 & & 0 & & 4.13 & \\ p_4 & 5.36 & 0 & 3.21 & & 2.8 & \\ p_5 & 3.41 & 2.53 & 0 & 0 & 3.41 & \\ p_6 & 3.29 & 2.61 & 3.15 & \mathbf{2.48} & 0 & 0 \end{pmatrix}$$

The two closest clusters are C1 and C2, and consequently, the third cluster C3 is formed by these two clusters. How there is not yet only one cluster a fourth iteration must be done (Fig. 5.18).

A new step A. Calculation of the matrix of distances between clusters is performed, and the data are now: 1. {0.89, 2.94}; 4. {6.25, 3.14} and C3 {C1 {3. {3.75, 1.12}; 5. {4.1, 1.8}}, C2 {2. {4.36, 5.21}; 6. {3.9, 4.27}}} The new matrix of distances is:

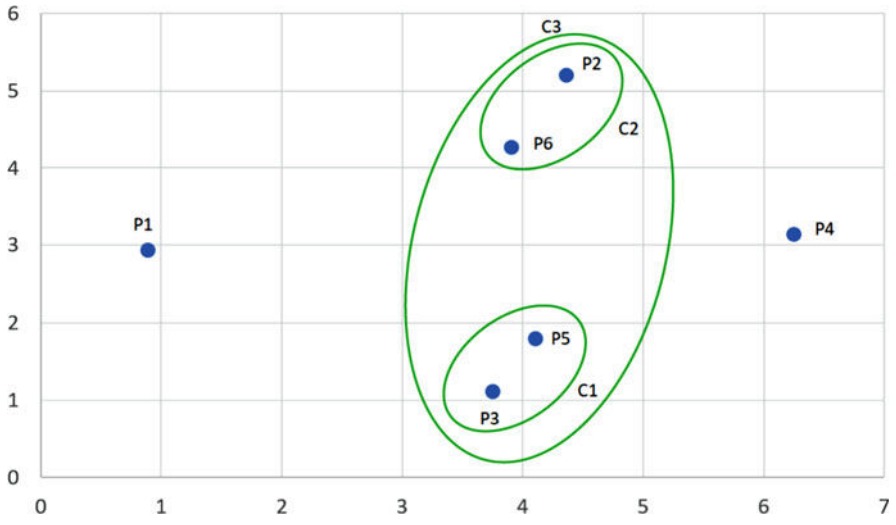


Fig. 5.18 Third cluster MIN distances

$$\begin{pmatrix} & p_1 & p_4 & C3-C1_{p3} & C3-C1_{p5} & C3-C2_{p2} & C3-C2_{p6} \\ p_1 & 0 & & & & & \\ p_2 & 4.15 & & & & 0 & \\ p_3 & 3.39 & & 0 & & 0 & \\ p_4 & 5.36 & 0 & 3.21 & & 2.8 & \\ p_5 & 3.41 & 2.53 & 0 & 0 & 0 & \\ p_6 & 3.29 & 2.61 & 3.15 & 0 & 0 & 0 \end{pmatrix}$$

The distance between clusters is now between the two points 1, 4 and cluster 3, formed by clusters 1 and 2, and consequently, the distance between clusters 1 and 2 is now 0 because they are in the same cluster.

In the new Step B, the closest clusters are found. The new data are now as follows: 1. {0.89, 2.94}; 4. {6.25, 3.14} and C3 {C1 {3. {3.75, 1.12}; 5. {4.1, 1.8}}, C2 {2. {4.36, 5.21}; 6. {3.9, 4.27}}}

$$\begin{pmatrix} & p_1 & p_4 & C3-C1_{p3} & C3-C1_{p5} & C3-C2_{p2} & C3-C2_{p6} \\ p_1 & 0 & & & & & \\ p_2 & 4.15 & & & & 0 & \\ p_3 & 3.39 & & 0 & & 0 & \\ p_4 & 5.36 & 0 & 3.21 & & 2.8 & \\ p_5 & 3.41 & \mathbf{2.53} & 0 & 0 & 0 & \\ p_6 & 3.29 & 2.61 & 3.15 & 0 & 0 & 0 \end{pmatrix}$$

The two closest clusters are points 4 and C3. Therefore, the third cluster, C4, is the one formed by these two clusters.

And now we do have a single C5 cluster that will be the one formed by C4 and point 1, and the algorithm has finished (Fig. 5.19).

- *MAX*. Define the proximity between two clusters as the distance between the two furthest points of the two clusters. It is also called Complete Link.

Over the sample that has been solved, carry out step B of the agglomerative hierarchical clustering algorithm using the algorithm with the MAX proximity definition:

First iteration: If we take the distance matrix between the clusters, we consider that in the first iteration, each point constitutes a cluster.

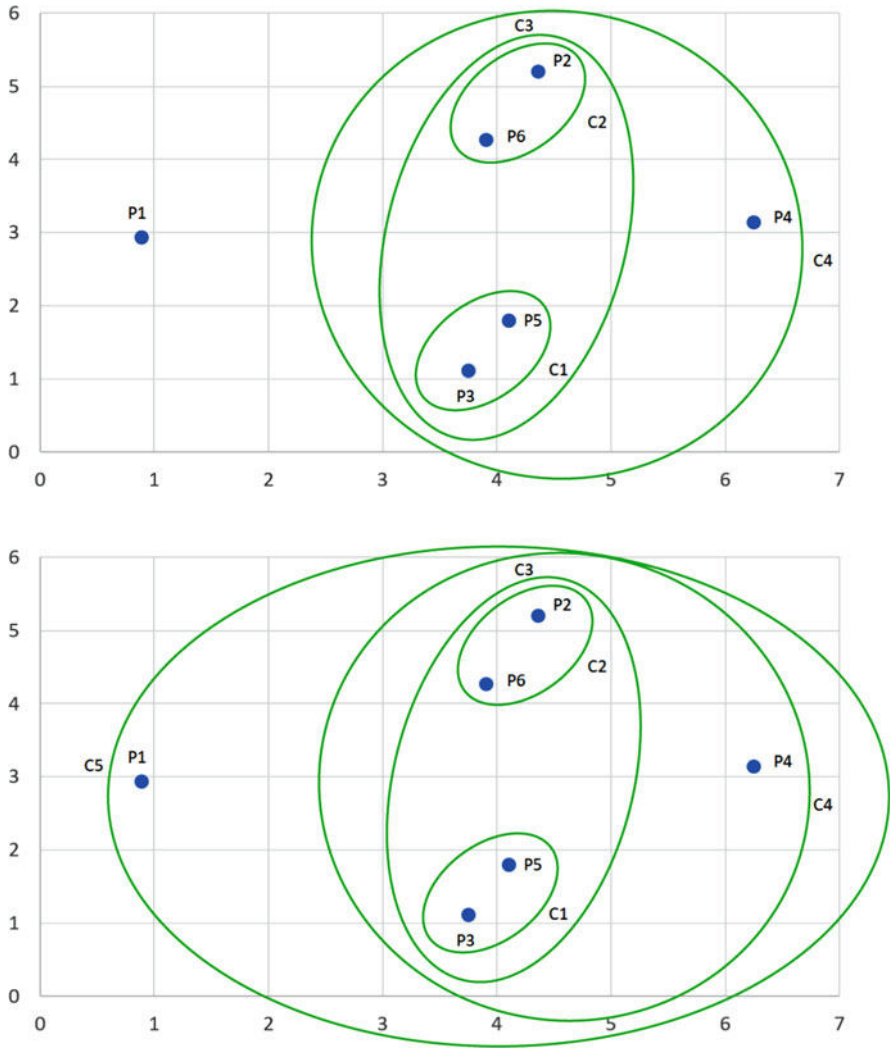


Fig. 5.19 Fourth and five clusters MIN distances

$$\begin{pmatrix} & p_1 & p_2 & p_3 & p_4 & p_5 & p_6 \\ p_1 & 0 & & & & & \\ p_2 & 4.15 & 0 & & & & \\ p_3 & 3.39 & 4.13 & 0 & & & \\ p_4 & 5.36 & 2.80 & 3.21 & 0 & & \\ p_5 & 3.41 & 3.42 & \mathbf{0.76} & 2.53 & 0 & \\ p_6 & 3.29 & 1.05 & 3.15 & 2.61 & 2.48 & 0 \end{pmatrix}$$

The two closest clusters are 3 and 5. Therefore, the first cluster, C1, is the one formed by these two points.

As we do not have a single cluster, we go to the second iteration.

Step A is carried out. Calculation of the matrix of distances between clusters.

The data are now {0.89, 2.94}; 2. {4.36, 5.21}; 4. {6.25, 3.14}; 6. {3.9, 4.27} and C1 {3. {3.75, 1.12}; 5. {4.1, 1.8}}

The distance matrix is:

$$\begin{pmatrix} & p_1 & p_2 & p_4 & p_6 & C1_{p3} & C1_{p5} \\ p_1 & 0 & & & & & \\ p_2 & 4.15 & 0 & & & & \\ p_3 & 3.39 & 4.13 & & & 0 & \\ p_4 & 5.36 & 2.80 & 0 & & 3.21 & \\ p_5 & 3.41 & 3.42 & 2.53 & & 0 & 0 \\ p_6 & 3.29 & 1.05 & 2.61 & 0 & 3.15 & 2.48 \end{pmatrix}$$

The distance between clusters is now between the four points 1, 2, 4, and 6 and cluster 1, formed by points 3 and 5, in the previous iteration, and consequently, the distance between points 3 and 5 is now 0 because they are in the same cluster.

Step B is now performed using the definition of proximity MAX. Data are now 1. {0.89, 2.94}; 2. {4.36, 5.21}; 4. {6.25, 3.14}; 6. {3.9, 4.27} and C1 {3. {3.75, 1.12}; 5. {4.1, 1.8}}. This first cluster is depicted in Fig. 5.16.

$$\begin{pmatrix} & p_1 & p_2 & p_4 & p_6 & C1_{p3} & C1_{p5} \\ p_1 & 0 & & & & & \\ p_2 & 4.15 & 0 & & & & \\ p_3 & 3.39 & 4.13 & & & 0 & \\ p_4 & 5.36 & 2.80 & 0 & & 3.21 & \\ p_5 & 3.41 & 3.42 & 2.53 & & 0 & 0 \\ p_6 & 3.29 & \mathbf{1.05} & 2.61 & 0 & 3.15 & 2.48 \end{pmatrix}$$

The two closest clusters are 2 and 6. Therefore, the second cluster, C2, is the one formed by these two points. This second cluster is depicted in Fig. 5.17.

As we do not have a single cluster, we go to the third iteration.

In Step A, the matrix of distances between points is calculated:

$$\begin{pmatrix} & p_1 & p_4 & C1_{p3} & C1_{p5} & C2_{p2} & C2_{p6} \\ p_1 & 0 & & & & & \\ p_2 & 4.15 & & & & 0 & \\ p_3 & 3.39 & & 0 & & 4.13 & \\ p_4 & 5.36 & 0 & 3.21 & & 2.8 & \\ p_5 & 3.41 & 2.53 & 0 & 0 & 3.41 & \\ p_6 & 3.29 & 2.61 & 3.15 & 2.48 & 0 & 0 \end{pmatrix}$$

The distance between clusters is now between the two points 1, 4 and the clusters C1, formed by points 3 and 5, and C2, identified in the previous iteration, and consequently, the distance between points 2 and 6 is now 0 because they are in the same cluster.

In Step B, the two closest clusters are merged. The data are now: 1. {0.89, 2.94}; 4. {6.25, 3.14}, C1 {3. {3.75, 1.12}; 5. {4.1, 1.8}}, and C2 {2. {4.36, 5.21}; 6. {3.9, 4.27}}

$$\begin{pmatrix} & p_1 & p_4 & C1_{p3} & C1_{p5} & C2_{p2} & C2_{p6} \\ p_1 & 0 & & & & & \\ p_2 & 4.15 & & & & 0 & \\ p_3 & 3.39 & & 0 & & 4.13 & \\ p_4 & 5.36 & 0 & 3.21 & & \mathbf{2.8} & \\ p_5 & 3.41 & 2.53 & 0 & 0 & 3.41 & \\ p_6 & 3.29 & 2.61 & 3.15 & 2.48 & 0 & 0 \end{pmatrix}$$

The two closest clusters are point 4 and C2. Therefore, the third cluster, C3, is the one formed by these two clusters (Fig. 5.20).

As we do not have a single cluster, we proceed to the fourth iteration, but before, it is important at this point to provide a deep explanation about why 2.8 is the minimum distance between clusters with the MAX definition distance algorithm.

The proximity between clusters with MAX is defined as the distance between the two farthest points of the two clusters:

- The first two clusters considered are points 1 and 4, for which the distances, as seen in the previous matrix, are: {4.15, 3.39, 5.36, 3.41, 3,29, 2,53, 2,61}, and from all of them, the maximum, MAX, is 5.36.

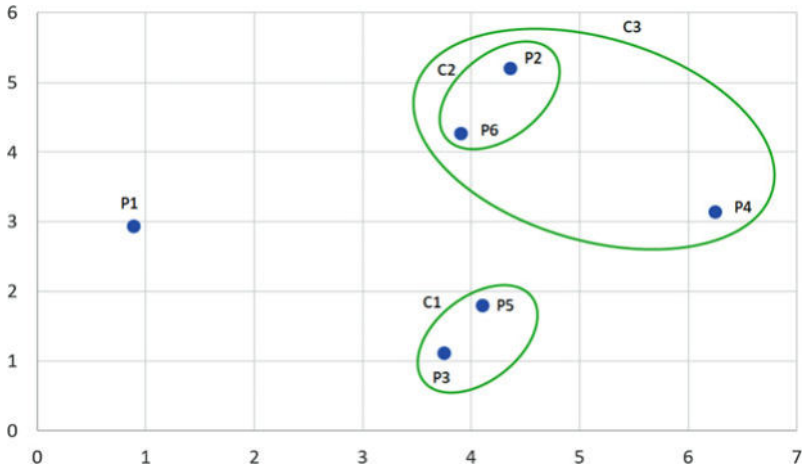


Fig. 5.20 Third cluster MAX distances

- The second two clusters considered are C1 and C2; all the distances between all the points in both clusters must be calculated, that is distance (3.2), distance (3.6), distance (5.2), and distance (5.6), which are: {4.13, 3.15, 3.42, 2.48} and from all of them to take the maximum, which is 4.13.
- The third two clusters considered are C1 and Point 1; the distances to be analyzed are (3,1) and (5,1), which are: {3.39, 3.41}, and the maximum is 3.41.
- The fourth two clusters to be considered are C1 and Point 4; the distances to be compared are (3,4) and (5,4), which are: {3.21, 2.53}, and the maximum is 3.21.
- And fifth, two clusters to be compared are C2 and Point 1; the distances to be compared are (2,1) and (6,1), which are: {4.15, 3.29}, and the maximum is 4.15
- And finally, the last two clusters to be compared are C2 and Point 4; the distances to be compared are (2,4) and (6,4), which are {2.8, 2,61}, and the maximum is 2.8.

Once you have all the distances between the clusters with the algorithm MAX, the minimum of all of them will be the selected one to merge both clusters in one, the distances obtained are: {5.6, 4.13, 3.41, 3.21, 4.15 and 2,8}, the minimum of all of them is 2.8, and in consequence the clusters merged in this iteration are C2 and Point 4.

In Step A of the fourth iteration of the agglomerative hierarchical classification algorithm with the distance definition MAX, the new matrix of distances between clusters is obtained.

$$\begin{pmatrix} & p_1 & C1_{p3} & C1_{p5} & C3-C2_{p2} & C3-C2_{p6} & C3_{p4} \\ p_1 & 0 & & & & & \\ p_2 & 4.15 & & & & 0 & \\ p_3 & 3.39 & 0 & & 4.13 & & \\ p_4 & 5.36 & 3.21 & & 0 & & 0 \\ p_5 & 3.41 & 0 & 0 & 3.42 & & 2.53 \\ p_6 & 3.29 & 3.15 & 2.48 & 0 & 0 & 0 \end{pmatrix}$$

The distance between clusters is now between Point 1 and the clusters, C1 formed by points 3 and 5, and C3 identified in the previous iteration, and consequently, the distance between Point 4 and cluster C2 is 0 because they are in the same cluster.

In Step B, the two closest clusters are merged. The data are now: 1. {0.89, 2.94}; C1 {3. {3.75, 1.12}; 5. {4.1, 1.8}} and C3 {C2 {2. {4.36, 5.21}; 6. {3.9, 4.27}}, 4. {6.25, 3.14}}

$$\begin{pmatrix} & p_1 & C1_{p3} & C1_{p5} & C3-C2_{p2} & C3-C2_{p6} & C3_{p4} \\ p_1 & 0 & & & & & \\ p_2 & 4.15 & & & & 0 & \\ p_3 & 3.39 & 0 & & 4.13 & & \\ p_4 & 5.36 & 3.21 & & 0 & & 0 \\ p_5 & \mathbf{3.41} & 0 & 0 & 3.42 & & 2.53 \\ p_6 & 3.29 & 3.15 & 2.48 & 0 & 0 & 0 \end{pmatrix}$$

The distances applying MAX are:

- The first two clusters considered are Points 1 and C1, for which the distances, as seen in the previous matrix, are 3.39 between points 1 and 3 and 3.41 between points 1 and 5, and from both of them, the maximum, MAX, is 3.41.
- The second and final two clusters considered are Points 1 and C3, for which the distances, as seen in the previous matrix, are 4.15 between points 1 and 2, 3.29 between points 1 and 6, and 5.36 between points 1 and 4, and from all of them, the maximum, MAX, is 5.36.

Once you have both the distances between the clusters with the MAX algorithm, the minimum of both of them will be the selected one to merge both clusters into one. Since the distances obtained are 3.41 and 5.36, the minimum of both of them is 3.41, and consequently, the clusters merged in this iteration are C1 and Point 1, that is, C4.

The previous result means that the algorithm has finished because a final unique cluster is obtained, that one constituted by the previous cluster C3 and the new cluster C4, that will be C5 (Fig. 5.21).

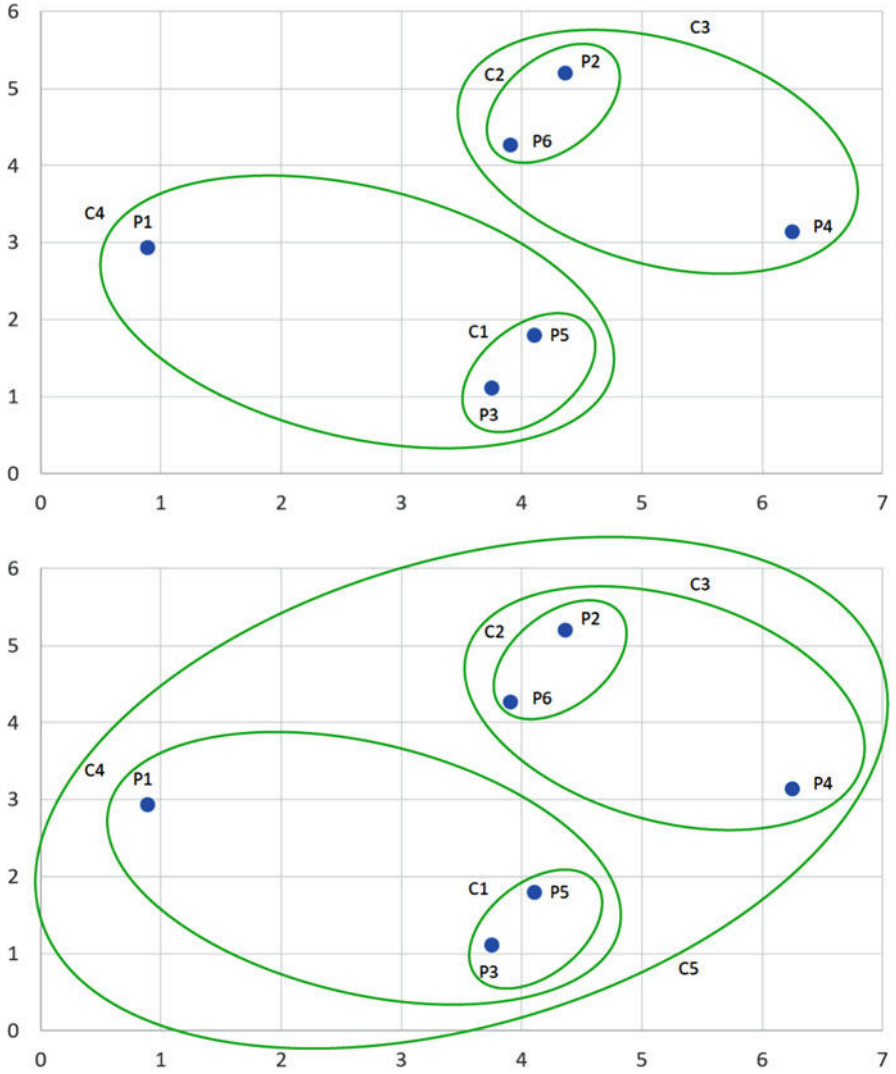


Fig. 5.21 Fourth and Five Clusters MAX distances

- *Group Average.* Define the proximity between two clusters as the average of the distances between all the pairs that can be formed with points from the two clusters.

$$\text{proximity}\left(C_i, C_j\right) = \frac{\sum_{i=1}^{m}\sum_{j=1}^{n} \text{proximity}\left(x_i, y_j\right)}{m * n}$$

Over the sample that has been solved, carry out step B of the agglomerative hierarchical clustering algorithm using the algorithm with the Group Average proximity definition (see Fig. 5.14):

First Iteration: If we take the distance matrix between the clusters, we consider that in the first iteration, each point constitutes a cluster.

	p_1	p_2	p_3	p_4	p_5	p_6
p_1	0					
p_2	4.15	0				
p_3	3.39	4.13	0			
p_4	5.36	2.80	3.21	0		
p_5	3.41	3.42	**0.76**	2.53	0	
p_6	3.29	1.05	3.15	2.61	2.48	0

The two closest clusters are 3 and 5. Therefore, the first cluster, C1, is the one formed by these two points (see Fig. 5.16).

As we do not have a single cluster, we go to the second iteration.

Step A of the second iteration of the agglomerative hierarchical classification algorithm with the distance definition Group Average is performed, and the new data are as follows: 1. {0.89, 2.94}; 2. {4.36, 5.21}; 4. {6.25, 3.14}; 6. {3.9, 4.27} and C1 {3. {3.75, 1.12}; 5. {4.1, 1.8}}

	p_1	p_2	p_4	p_6	$C1_{p3}$	$C1_{p5}$	p_1	p_2	p_4	p_6	$C1_{p3}$	$C1_{p5}$
p_1	0						0					
p_2	4.15	0					4.15	0				
$C1_{p3}$	3.39	4.13			0		**3.40**	**3.78**			0	
p_4	5.36	2.80	0		3.21		5.36	2.80	0		**2.87**	
$C1_{p5}$	3.41	3.42	2.53		0	0	**3.40**	**3.78**	**2.87**		0	0
p_6	3.29	1.05	2.61	0	3.15	2.48	3.29	1.05	2.61	0	**2.82**	**2.82**

The distance between clusters is now between the four points 1, 2, 4 and 6 and cluster 1, formed by points 3 and 5, in the previous iteration, but now the distances change because it is done with the mean.

$$\text{proximity}\,(p_1, C_1) = \frac{\sum_{i=1}^{2}\sum_{j=1}^{1}\text{proximity}\,((p1,p3),(p1,p5))}{2*1} = \frac{3.39+3.41}{2} = 3.40$$

$$\text{proximity}\,(p_2, C_1) = \frac{\sum_{i=1}^{2}\sum_{j=1}^{1}\text{proximity}\,((p2,p3),(p2,p5))}{2*1} = \frac{4.13+3.42}{2} = 3.78$$

$$\text{proximity}\ (p_4, C_1) = \frac{\sum_{i=1}^{2}\sum_{j=1}^{1} \text{proximity}\ ((p4, p3), (p4, p5))}{2 * 1} = \frac{3.21 + 2.53}{2} = 2.87$$

$$\text{proximity}\ (p_6, C_1) = \frac{\sum_{i=1}^{2}\sum_{j=1}^{1} \text{proximity}\ ((p6, p3), (p6, p5))}{2 * 1} = \frac{3.15 + 2.48}{2} = 2.82$$

1. In Step B, the minimum distance between clusters using the Group Average algorithm is used to select which two closest clusters must be merged. The data are now: 1. {0.89, 2.94}; 2. {4.36, 5.21}; 4. {6.25, 3.14}; 6. {3.9, 4.27} and C1 {3. {3.75, 1.12}; 5. {4.1, 1.8}}

	p_1	p_2	p_4	p_6	$C1_{p3}$	$C1_{p5}$
p_1	0					
p_2	4.15	0				
$C1_{p3}$	3.40	3.78			0	
p_4	5.36	2.80	0		2.87	
$C1_{p5}$	3.40	3.78	2.87		0	0
p_6	3.29	**1.05**	2.61	0	2.82	2.82

The two closest clusters are points 2 and 6. Therefore, the second cluster, C2, is the one formed by these two points (see Fig. 5.17).

As we do not have a single cluster, we go to the third iteration.

Step A of the third iteration of the agglomerative hierarchical classification algorithm with the distance definition group average is performed, and the new data are as follows: 1. {0.89, 2.94}; 4. {6.25, 3.14}; C1{3. {3.75, 1.12}; 5. {4.1, 1.8}, C2{2. {4.36, 5.21}; 6. {3.9, 4.27}}}

	p_1	p_4	$C1p_3$	$C1p_5$	$C2_{p2}$	$C2_{p6}$	p_1	p_4	$C1p_3$	$C1p_5$	$C2_{p2}$	$C2_{p6}$
p_1	0						0					
$C2p_2$	4.15				0		**3.72**				0	
$C1_{p3}$	3.39		0		4.13		3.40		0		**3.30**	
p_4	5.36	0	3.21		2.80		5.36	0	2.87		**2.70**	
$C1_{p5}$	3.41	2.53	0	0	3.42		3.40	2.87	0	0	**3.30**	
$C2p_6$	3.29	2.61	3.15	2.48	1.05	0	**3.72**	**2.70**	**3.30**	**3.30**	**3.30**	**0**

The distance between clusters is now between the two points 1 and 4 and the two clusters 1 and 2, but now almost all the distances change because it is done with the mean.

$$\text{proximity}\,(C_1, C_2) = \frac{\sum_{i=1}^{2}\sum_{j=1}^{2} \text{proximity}\,((p3, p2), (p3, p6), (p5, p2), (p5, p6))}{2 * 2}$$
$$= \frac{4.13 + 3.15 + 3.42 + 2.48}{4} = 3.30$$

$$\text{proximity}\,(p_1, C_2) = \frac{\sum_{i=1}^{2}\sum_{j=1}^{1} \text{proximity}\,((p1, p2), (p1, p6))}{2 * 1} = \frac{4.15 + 3.29}{2} = 3.72$$

$$\text{proximity}\,(p_4, C_2) = \frac{\sum_{i=1}^{2}\sum_{j=1}^{1} \text{proximity}\,((p4, p2), (p4, p6))}{2 * 1} = \frac{2.80 + 2.61}{2} = 2.70$$

2. In Step B, the minimum distance between clusters using the Group Average algorithm is used to select which two closest clusters must be merged. The data are now: 1. {0.89, 2.94}; 4. {6.25, 3.14}; C1{3. {3.75, 1.12}; 5. {4.1, 1.8}, C2 {2. {4.36, 5.21}; 6. {3.9, 4.27}}}

	p_1	p_4	$C1p_3$	$C1p_5$	$C2_{p2}$	$C2_{p6}$
p_1	0					
$C2p_2$	3.72				0	
$C1_{p3}$	3.40		0		3.30	
p_4	5.36	0	2.87		**2.70**	
$C1_{p5}$	3.40	2.87	0	0	3.30	
$C2p_6$	3.72	**2.70**	3.30	3.30	3.30	0

The two closest clusters are Points 4 and C2. Therefore, the third cluster, C3, is the one formed by these two clusters (see Fig. 5.18).

As we do not have a single cluster, we go to the fourth iteration.

1. Step A of the fourth iteration of the agglomerative hierarchical classification algorithm with the distance definition group average is performed, and the new data are as follows: 1. {0.89, 2.94}; 4. {6.25, 3.14}; C1{3. {3.75, 1.12}; 5. {4.1, 1.8}, C2{2. {4.36, 5.21}; 6. {3.9, 4.27}}}

	p_1	$C1p_3$	$C1p_5$	$C3.C2p_2$	$C3.C2_{p6}$	$C3_{p4}$	p_1	$C1p_3$	$C1p_5$	$C3.C2p_2$	$C3.C2_{p6}$	$C3_{p4}$
p_1	0						0					
$C3.C2p_2$	4.15			0			**4.27**			0		
$C1_{p3}$	3.39	0		4.13			3.40	0		**3.15**		
$C3p_4$	5.36	3.21		2.80		0	**4.27**	**3.15**		**0**		**0**
$C1_{p5}$	3.41	0	0	3.42		2.53	3.40	0	0	**3.15**		**3.15**
$C3.C2p_6$	3.29	2.61	2.48	0	0	2.61	**4.27**	**3.15**	**3.15**	**0**	0	**0**

The distance between clusters is now between point 1 and the two clusters C1 and C3, but now almost all the distances change because it is done with the mean.

$$\text{proximity}\,(C_1, C_3)$$

$$= \frac{\sum_{\substack{i=1\\j=1}}^{\substack{3\\2}} \text{proximity}\,((p3,p2),(p3,p6),(p3,p4)(p5,p2),(p5,p6),(p5,p4))}{2*3}$$

$$= \frac{4.13+3.15+3.21+3.42+2.48+2.53}{6} = 3.15$$

$$\text{proximity}\,(p_1, C_3) = \frac{\sum_{\substack{i=1\\j=1}}^{\substack{3\\1}} \text{proximity}\,((p1,p2),(p1,p6)(p1,p4))}{3*1}$$

$$= \frac{4.15+3.29+5.36}{3} = 4.27$$

3. In Step B, the minimum distance between clusters using the Group Average algorithm is used to select which two closest clusters must be merged. The data are now P1, C1, and C3.

	p_1	$C1p_3$	$C1p_5$	$C3.C2p_2$	$C3.C2_{p6}$	$C3_{p4}$
p_1	0					
$C3.C2p_2$	4.27			0		
$C1_{p3}$	3.40	0		**3.15**		
$C3p_4$	4.27	**3.15**		0		0
$C1_{p5}$	3.40	0	0	**3.15**		**3.15**
$C3.C2p_6$	4.27	**3.15**	**3.15**	0	0	0

The distances applying Group Average are:

- The first two clusters considered are Points 1 and C1, for which the mean distance applying the group average is 3.40. With this algorithm, as it is a mean, only one distance will be used.

- The second and final two clusters considered are Points 1 and C3, for which the mean distance applying the group average is 4.27.
- Finally, the third two clusters considered are C1 and C3, for which the mean distance applying the group average is 3.15.

Once you have all the distances between the clusters with the algorithm Group Average, the minimum of all of them will be the selected one to merge both clusters into one. Since the distances obtained are 3.40, 4.27 and 3.15, the minimum is 3.15, and consequently, the clusters merged in this iteration are C1 and C3, that is, C4.

The previous result means that the algorithm has finished because a final unique cluster is obtained, that one constituted by the new cluster C4, and the point, or cluster, 1, that will be C5 (Fig. 5.22).

B. Computer-Based Solving

As in the other chapters, this subsection will address the use of software to solve the problem of unsupervised classification, but before applying R for solving this kind of problem, we introduce in this chapter another important issue related to the use of R, as it is the main tool that is currently used to carry out R projects, that is, RStudio, which will be introduced next.

The reasons for introducing RStudio are that the complexity of the problems is increasing and the RGui has already given us everything we need to start working with R, but its functionalities are very limited as well as its interface, so this is a good time to start working with the tool that we will use when we work with R in a professional way.

RStudio

Download of RStudio

The first step that we have to do is to visit the website of the company that developed RStudio, that until 2022 was RStudio, but that since the is Posit. (https://posit.co/) to download it. If we click on *Download RStudio,* the following web page is opened: https://www.rstudio.com/products/rstudio/download/

Two versions with different features are offered in Desktop or Server. The only one that is free is RStudio Desktop Open Source License. We click on the button that says Download.

A process of two steps appears; the first says us that having R installed is mandatory, but as we have R installed since the first chapter of the book, this does not apply to us. The second is the installation of RStudio, which automatically takes care of the operating system that we are using.

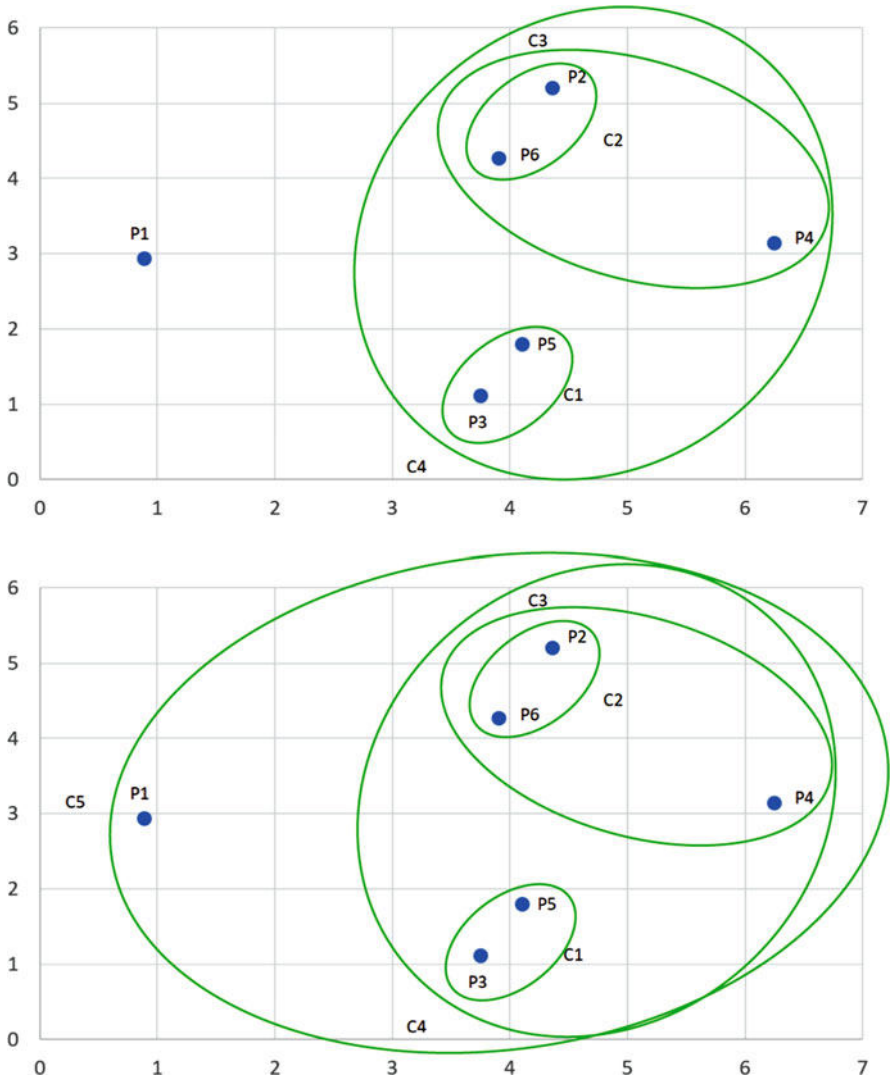


Fig. 5.22 Fourth and fifth cluster group average distances

Installation of RStudio

Before starting to install RStudio, it is important to bear in mind that it is essential to have previously installed R with a version equal to or greater than 3.3.0+.

Click twice on *RStudio-2022.07.02-576.exe* to install RStudio. During the installation, it will ask us where we want to install it and we will press Next.

Then it will ask us for the start menu folder, in which we want to load the program's shortcuts. Click Install.

We click on install, and the installer install RStudio in our computer.

Getting Started with RStudio

To start working with RStudio, we open the program by clicking on the RStudio icon in the start menu and the program opens.

RStudio is made up of four windows, one in each quadrant of the screen. Different information is displayed in each of them, which is fixed and immovable for each window. We are going to see what information is based on the configuration of quadrants that RStudio presents by default. However, as we will see later, we can change this because each window can be seen in the quadrant we want. By default, when we open a session in RStudio, window 1 does not appear. We have to go to the File menu, then New File option, and then RScript option.

1. Window 1. When we open it with the instructions explained in the previous paragraph, it appears in the upper left quadrant of the screen. By default, it has a single Source tab, which, as the name suggests, refers to the sources that we are developing. In this screen, we will see the new code, or text, or code and text that we are developing in the forms that RStudio allows. For each new source, a new tab is opened. It has different code types, between then:

    2. RScript. It allows us to develop Scripts.
    3. TextFile. It allows us to write text files.
    4. C/C++File. It allows us to write and compile programs in C/C++.
    5. R Sweave. It allows us to do literate programming using Sweave.
    6. RHTML. It allows us to develop HTML code with embedded R code.
    7. R Documentation. It allows us to document a function or a database in R.

2. Window 2. By default, it appears in the lower left quadrant of the screen. It has three tabs:

    2.1. Console. It allows us to execute the R code. It is similar to the RGui console window.
    2.2. Terminal. It allows us to manage the operating system through command lines. In the case of Windows MS-DOS.
    2.3. Background jobs: Allows to execute programs in background.

3. Windows 3 and 4. By default, it appears in the upper right quadrant of the screen. Windows 3 and 4 allow to show the same 10 tabs. We can show them in one or the other of our choice. Those that are not shown in window 3 will be shown in window 4 and vice versa. Neither of the two windows can have less than one tab.

    1. Environment. It allows us to see what environment we are working with, what packages and data sets we have loaded, and perform operations on them.
    2. History. It allows us to open and view a .Rhistory file and perform operations on it.
    3. Files. It allows us to see and operate with files and files in the working directory.

4. Plots. It allows us to see and carry out operations with the obtained graphs.
5. Connections. It allows us to see and make connections with database management systems.
6. Packages. It allows us to see and operate with the packages of the standard and user libraries.
7. Help. It allows us to see queries made on the help of R.
8. Tutorial.
9. Viewer.
10. Presentation.

To change and customize in which quadrant each window is displayed and which tabs we want to see in each of windows 3 and 4, we have two options:

– View menu; Panes option; Pane Layout option. . .
– Tools menu; Global Options option. . .; Pane Layout option

The first thing we are going to do when we open RStudio is enter our first instruction in window 2.1. Write the following:

>contributors()

As we can see, the result appears in the 1.2 Information window, which had not been opened by default. Having made our tribute to the people who have created and developed R, and hoping perhaps to see our name someday there, we start working with RStudio.

RStudio has eleven menus:

1. File
2. Edit
3. Code
4. View
5. Plot
6. Session
7. Build
8. Debug
9. Profile
10. Tools
11. Help

Let us see each of them in detail:

1. File: It has the following options:
   - New File: It allows us to start creating a new file. It offers different new file options, between them:
     – RScript: It opens the R script window in the first quadrant so we can write a new script.
     – RNotebook: It needs to install and load a set of packages to work.
     – R Markdown: It needs to install and load a set of packages to work.

  - Shiny web App: It need to install and load the Shiny package.
  - TextFile: Opens the Text File window in the first quadrant so we can write a text file.
  - C++File: It opens the C/C++ File window in the first quadrant to allow us to write and compile C/C++ programs. You need to install additional tools.
  - R Sweave: It opens the R Sweave window to allow us to perform literary programming, commenting on the R code.
  - RHTML: It opens the R HTML window to allow us to develop HTML code with embedded R code.
  - R Presentation.
  - R Documentation: It opens an R Documentation window to allow us to document a function or database in R, which will save with the extension .Rd.

- New Project: It allows us to create a new R project. To create a new project, we will follow these steps:

  - Select the working directory.
    Select the type of project:

    New project
    New package
    New Shiny web app
    Are also very interesting the Quarto projects

- Open File: It allows us to open a file with the extensions that can be handled by RStudio.
- Reopen with Encoding: This allows us to open a file with encoding.
- Recent Files: It allows us to open any of the files with which we have worked recently.
- Open Project: It allows us to open a project.
- Open Project in New Session: It allows us to open a project in a new session.
- Recent Projects: It allows us to open projects with which we have worked recently.
- Import Data set: It allows us to import data sets from:

  1.9.1. Text
  1.9.2. csv-text
  1.9.3. Excel
  1.9.4. SPSS
  1.9.5. SAS
  1.9.6. Stata

- Save: It allows us to save to a file.
- Save as: It allows us to save with the features and name that we decide.
- Save with Encoding. It allows us to save with encoding.

- Save All: It allows us to save everything.
- Print: It allows us to print. It gives us a preview.
- Close: It allows us to close the current work document but not to exit the program.
- Close All: It allows us to close all open work documents but not to exit the program.
- Close All Except Current: It allows us to close all open documents except the one we are working on.
- Close Project: It allows us to close the open project.
- Quit: It allows us to close the program

2. Edit: It has the following options:

   2.1. Back: Allows to go to previous decisions.
   2.2. Forward: Allows to go to following decisions.
   2.3. Undo: It deletes instructions backward every time we press, not in the console but in the editor.
   2.4. Redo: It retrieves forward instructions in the editor each time we press.
   2.5. Cut: It cuts the selected text in the editor.
   2.6. Copy: It copies the selected text in the editor and in the console.
   2.7. Paste: It pastes the selected text in the editor and in the console.
   2.8. Paste with Indent: It paste the selected text.
   2.9. Folding: You have four options:

      2.9.1. Collapse: It collapses the selected text in the editor.
      2.9.2. Expand: It expands the collapsed text.
      2.9.3. Collapse All: It collapses everything.
      2.9.4. Expand All: It expands all.

   2.10. Find: It searches the editor for the string of characters that we introduce.
   2.11. Find in Files: It searches for the string that we introduce in the files of the working directory.
   2.12. Clear Console: It clears the console.

3. Code: It has the options for codification in R.
4. View. It has the following options:

   4.1 Show/Hide Toolbar: It shows or hides the toolbar.
   4.2 Panes: It allows you to customize the quadrants and windows. You have six basic options:

      4.2.1 Show All Panes: It shows all quadrants.
      4.2.2 Console on Left/Right: It shows the console on the left or right.
      4.2.3 Pane Layout: It allows us to choose which windows we want to see in each of the four quadrants.
      4.2.4 Zoom: It has a zoom option for each of the windows, and if we press it, it shows it in the complete window of the program.

4.3 Zoom In/Out: It allows you to increase and decrease how big everything inside the program window is.
4.4 Switch to Tab/Next/Previous/First/Last: This allows us to move between the open editing windows.
4.5 Move Focus to Source/Console/Terminal/Help: This allows us to place the cursor in the window that interests us.
4.6 Show History/Files/Plots/Packages/Environment/Viewer/Connections: This allows us to choose which window we want to see from the options.

5. Plots: It has the following options:

5.1 Next/Previous Plot. It allows us to move up and down sequentially through the plots.
5.2 Zoom Plot. It allows us to zoom the selected plot.
5.3 Save as Image: It allows us to save the plot with image format.
5.4 Save as pdf: It allows us to save the image in pdf format.
5.5 Copy to Clipboard. It allows us to copy the image to the Clipboard.
5.6 Remove Plot. It allows us to delete the current plot.
5.7 Clear All. It allows us to delete all open plots.

6. Session: It has the following options:

6.1 New Session. It allows us to open a new RStudio session. They are different executions of the program, and we can have more than two open.
6.2 Interrupt R. It allows us to interrupt the execution of the program.
6.3 Terminate R. It allows us to terminate the current execution of R. It terminates the execution, but reopens the program, and leaves it in the situation prior to the execution of the command.
6.4 Rstart R. Restarts the session but keeps the parameters of the previous session.
6.5 Set Working Directory. You have three options:

6.5.1 To Source File Location
6.5.2 To Files Pane Location
6.5.3 Choose Directory: It allows us to choose the working directory.

6.6 Load Workspace: It allows us to load a work session.
6.7 Save Workspace: It allows us to save a work session.
6.8 Clear Workspace: It cleans up the work session.
6.9 Quit Workspace: It closes the work session and the program.

7. Build: It has options for programing. We don´t see them because they are out of the scope of the book.
8. Debug: It has different options. We don´t see them because they are out of the scope of the book.
9. Profile: It has five options.
10. Tools: It has the options:

10.1. Install Packages: It allows us to install R packages by selecting the repository. It is a functionality similar to what the RGui's Packages menu gave us.
10.2. Check for Package Updates: It allows us to check if any of the packages that we have installed (the package must be installed, if it is not, it does not verify it) has updates.
10.3. Version Control: It allows us to control the versions of the project.
10.4. Shell: It allows us to open a command window.
10.5. Terminal: It provides the following options:

10.5.1. New Terminal: It allows us to open the terminal window inside R.
10.5.2. Rename Terminal: It allows us to rename the terminal.
10.5.3. Copy Terminal to Editor: It allows us to copy the terminal in the editor.
10.5.4. Diagnostic terminal: It gives us a diagnosis of the terminal.
10.5.5. Move Focus to Terminal: Select terminal as work window.
10.5.6. Previous/Next Terminal: Select the previous or next terminal.
10.5.7. Clear Terminal Buffer: Clears the previous instructions from the terminal.
10.5.8. Close Terminal: Closes the terminal.

10.6. Addins: It allows to manage the Addins.
10.7. Keyboard Shortcuts Help: It opens a pop-up window that tells us which are the main RStudio shortcuts to handle it.
10.8. Modify Keyboard Shortcuts: It opens a pop-up window that allows us to modify the shortcuts and customize them.
10.9. Project Options: It allows us to change the parameters of a project.
10.10. Global Options: It allows us to change the global parameters of RStudio. It allows us to change the parameters of:

10.10.1. General
10.10.2. Code
10.10.3. Appearance
10.10.4. Quadrant distribution
10.10.5. Packages
10.10.6. R Markdown
10.10.7. Smooth
10.10.8. Git/SVN
10.10.9. Publishing
10.10.10. Terminal

11. Help: It has the following options:

11.1. R Help: It opens a window with R and RStudio help resources
11.2. About RStudio: It gives us the version of RStudio that is being used and information about the program.

11.3. Check for Updates: Check if there is a newer version of RStudio than the one you are using.
11.4. RStudio Docs: It takes us to a web page with documentation about RStudio.
11.5. RStudio Community Forum: It takes us to the RStudio User Forum web page.
11.6. RStudio Support: It takes us to a web page with RStudio help articles and frequently asked questions.
11.7. Cheatsheets: It allows us to download "cheat sheets," summary sheets, related to the use of RStudio and R.
11.8. Keyboard Shortcuts Help: It opens the same pop-up window on shortcuts as the Tools menu.
11.9. Markdown Quick Reference: It opens, in the Help window of RStudio, a quick reference of Markdown.
11.10. Roxygen Quick Reference: It opens, in the Help window of RStudio, a quick reference of Roxygen.
11.11. Diagnostics.

Unsupervised Classification Exercises Solved in R

In this subsection, an unsupervised classification analysis will be carried out by applying all the concepts seen in the topic and using the computer programming environment R.

The example used will be the same as it was used in the previous, theoretical subsections: The sample of the qualifications {Theory, Laboratory} of eight students: 1. {4, 4}; 2. {3, 5}; 3. {1, 2}; 4. {5, 5}; 5. {0, 1}; 6. {2, 2}; 7. {4, 5}; 8. {2, 1} using the K-means clustering technique, two groups or clusters for the events of said sample. It will be solved, as in the theoretical section, with two clusters and with the initial centroids, which will be (0, 1) and (2, 2).

Unsupervised Classification with the K-Means Algorithm

In this subsection, the K-means unsupervised classification technique will be used for finding prospective in the marks of theory and laboratory of eight students. As in the theoretical section, K will be 2, and the initial centroids will be (0, 1) and (2, 2). A possible[12] list of R instructions or programs that solve this problem is:

When we start R, there is a set of packages loaded by default; among them is the package "stats," and we check it by means of the *search()* instruction, which shows

[12] As ever, the readers are encouraged to develop their own solution, list of instructions, program, or script.

us the set of packages that we have installed. The stats package contains functions that allow clustering using the K-means technique, and it is the one that we are going to use to solve the exercise.

The stats package contains the *kmeans (x, centers, iter.max)* function that allows a clustering analysis using the K-means technique. The first argument that this function must have is x, a numeric matrix with the data to be analyzed, in this case:

$$x = \begin{pmatrix} 4 & 4 \\ 3 & 5 \\ 1 & 2 \\ 5 & 5 \\ 0 & 1 \\ 2 & 2 \\ 4 & 5 \\ 2 & 1 \end{pmatrix}$$

To introduce them, we use the function *matrix()*:

>x <-matrix (c (4.4, 3.5, 1.2, 5.5, 0.1, 2.2, 4.5, 2.1), 2, 8), and we then assign x to the value of its transpose with the function *t()*.

>x <-t (x). We do so this way because it is easier to enter the points in their pair to avoid confusion.

The second argument is the centers in which either the desired number of clusters or an initial set of centroids are indicated. If the initial set of centroids is not specified and only the number of clusters is specified, the function will randomly choose an initial set of centroids. In the exercise, we can solve it in both ways. In the first way, it is only to introduce the value 2. The second way is to introduce the matrix c:

$$c = \begin{pmatrix} 0 & 1 \\ 2 & 2 \end{pmatrix}$$

which we have previously entered into the system as follows:

>c <-matrix (c (0,1, 2,2), 2, 2), and as above, we assign to c the value of its transpose:

c <-t (c)

Finally, in the third argument iter.max you must indicate how many maximum iterations are allowed. In this case, we indicate that there are 4.

We introduce the values of the three arguments in the function *kmeans()*,

> (clasificationns=(kmeans(m, c, 4)))

and obtain the result of the classification as follows:

K-means clustering with 2 clusters of sizes 4, 4
Cluster means:
[, 1] [, 2]
1 1.25 1.50
2 4.00 4.75
Vector clustering: [1] 2 2 1 2 1 1 2 1

Next, we obtain a matrix for each cluster. To do this, we first add a column to the matrix m to put each data point in its corresponding cluster:

>(m = cbind (classifications$cluster, m))

Then, with the subset instruction, we obtain the two arrays.
the complete instruction is:

>mc1 = subset (m, m [, 1] == 1)

>mc2 = subset (m, m [, 1] == 2)}

And we end up eliminating the column that indicates the cluster:

(mc1 = mc1 [, - 1])

(mc2 = mc2 [, - 1])

This is the correct result that coincides with what was obtained in the theoretical calculation.

Finally, we draw the points (events) and the centroids to have a graphic view of what was obtained. To do this, we use the plot () function

The even () function is used to keep the same graph.

Agglomerative Hierarchical Clustering

To solve the Agglomerative Hierarchical clustering problem with R, we will use the package developed by a student from the University of Alcalá as his final project, LearnClust. We are going to install the package from local because we want to know it in depth and consequently we are going to go to its web page within the CRAN repository.

Consequently, the first thing we do is go to the CRAN website. Click on the link:

http://CRAN.R-project.org/

within the third heading 3. Files, and we go to a new page where all the downloadable files of R are located. Click on the link:

Packages

And we arrive to a new page where are all the R packages. We click on:
Table of available packages, sorted by name

And look for the package:

LearnClust

We arrive at the package website where it is absolutely all the information about. The title of the package *LearnClust: Learning Hierarchical Clustering Algorithms* and the downloadable:

The package page contains the following:

First, it is a description about the functionality that the package pretends to give to the user:

Classical hierarchical clustering algorithms, agglomerative and divisive clustering. Algorithms are implemented in a theoretical way, step by step. It includes some detailed functions that explain each step. Every function allows options to obtain different results using different techniques. The package explains to nonexpert users how hierarchical clustering algorithms work.

The following are the characteristics of the package:

- Version: 1.1
- Depends: magick
- Suggests: knitr, rmarkdown
- Published: 2020-11-29
- Author: Roberto Alcantara [aut, cre], Juan Jose Cuadrado [aut], Universidad de Alcala de Henares [aut]
- Maintainer: Roberto Alcantara <roberto.alcantara at edu.uah.es>
- License: Unlimited
- NeedsCompilation: no
- CRAN checks: LearnClust results

The following is the documentation that will allow us to understand and analyze the package:

- Reference manual: LearnClust.pdf
- Vignettes: Learning Clusterization

The downloads of the package that can be done are as follows:

- Package source: LearnClust_1.1.tar.gz
- Windows binaries: r-devel: LearnClust_1.1.zip, r-release: LearnClust_1.1.zip, r-oldrel: LearnClust_1.1.zip
- macOS binaries: r-release (arm64): LearnClust_1.1.tgz, r-release (x86_64): LearnClust_1.1.tgz, r-oldrel: LearnClust_1.1.tgz
- Old sources: LearnClust archive

And, finally, how the link to the package should be written:

Please use the canonical form:
https://CRAN.R-project.org/package=LearnClust to link to this page.

We install the package using one of the ways learned in the previous chapters.

Once the package is installed, we load it using the following instruction:
A continuación, cargamos el paquete LearnClust en R mediante la función

```
library(LearnClust)
```

We introduce the function:

search()

and check it.

If we use RStudio, the way to install and load LearnClust is, of course, significantly easier; first, to install the package, we only need to go to the menuTools, to the option Install Packages, and, there, select LearnClust, in the box of the window that appears. Only by typing the first letters of the package can we see it filled automatically.

Once we have the package installed, we can go to the card Packages in the corresponding window and click on LearnClust and the function library() to automatically load the package. If we click again, we unload the package, and a detach() instruction is executed automatically.

Once we have the LeanClust package loaded, we pass to use it to solve the same problem that we have solved in theory, that is, the hierarchical agglomerative clustering over the sample: 1. {0.89, 2.94}; 2. {4.36, 5.21}; 3. {3.75, 1.12}; 4. {6.25, 3.14}; 5. {4.1, 1.8}; 6. {3.9, 4.27}.

To do that, we first introduce the sample data in R, and we do it, as in the previous case, with a matrix

```
m<-matrix(c(0.89,2.94, 4.36,5.21, 3.75,1.12, 6.25,3.14, 4.1,1.8, 3.9,4.27), 2, 6)
```

and following, we assign the value m to its transpose:

```
(m<-t(m))
```

Once we have introduced the data we obtain, we calculate them hierarchical agglomerative clusterization using the *Euclidean* distance and the proximity definition *MIN*. To do that, we use the function included in the package agglomerativeHC, with the parameters m, which indicates the pair of data that we are going to cluster; 'EUC', which indicates the type of distance that we are going to use, which, in this case, is the Euclidean; and 'MIN', which indicates the type of proximity that we are going to use, that are me closets points. With all of this, the instruction is:

```
agglomerativeHC(m, 'EUC', 'MIN')
```

The resulting result is interpreted as follows:
We have 11 clusters:

- The first 6 correspond to the 6 points introduced and that, as we know, are considered, everyone of them, as clusters in the first iteration.
- The 7 is the cluster formed by the points, clusters, 3. {3.75, 1.12} y 5. {4.1, 1.8}, which corresponds with the first cluster, formed by points 3 and 5, which we obtained when we solved the exercise theoretically, C1.

- 8 is the cluster formed by the points, clusters, 2. {4.36, 5.21} y 6. {3.9, 4.27}, which corresponds to the second cluster, formed by points 2 and 6, which we obtained when we solved the exercise theoretically, C2.
- 9 is the cluster formed by point 3. {3.75, 1.12}, 5. {4.1, 1.8}, 2. {4.36, 5.21} y 6. {3.9, 4.27}, which belongs to the clusters C1, the first two, and C2, the second two, that is, is the cluster formed by the union of the two previous clusters, just as we obtained in theory, C3=C1+C2.
- 10 is the cluster formed by the clusters formed by point 4. {6.25, 3.14}; and cluster 3, obtained in the previous iteration, in the same manner that we obtained in theory, that is, C4=4+C3.
- Finally, 11 is obtained by joining point 1. {0.89, 2.94} to cluster C4, obtaining the same cluster 5 that we obtained in theory.

Finally, the last part of the solution tells us how the clusters have been obtained in each iteration:

1. Clusters (individuals) 3 and 5 are joined, resulting in cluster 7, since there are 6 individuals.
2. The (individual) clusters 2 and 6 are joined, resulting in cluster 8.
3. Clusters 7 and 8 are joined, giving rise to 9.
4. Cluster (individual) 4 joins cluster 9, giving rise to 10.
5. Finally, cluster (individual) 1 is joined to cluster 10, giving rise to the last cluster 11.

However, the package is called LeaningClust, and the reason for the name is that the package is not only intended to provide the hierarchical agglomerative clustering solution but also to teach how it works. To achieve this second objective, detailed functions are included, which explain how the algorithms that implement the functions that do not have that extension work. To see how they work and how they explain the algorithms, we are going to see the same function that we have used to obtain the previous agglomerative hierarchical clustering but with the details extension. The full instructions are as follows:

```
agglomerativeHC.details(m, 'EUC', 'MIN')
```

The solution that we obtain is the following:

- First, we obtain the 6 initial clusters corresponding to every point of the remaining 6 points.
- Next, the function tells us what the package is going to do.
- Next, start doing it, step by step, starting with the first one:
  - Calculate the distances using the distance measure selected, the Euclidean one.
  - Find the two closest points of the two clusters using the distance definition selected, the MIN.
  - Join the two closest points, in this case, points 3 and 5, into the next cluster, in this case, 7.
  - Show the cluster, which points constitute it.

- Follow with the rest of the steps, explaining, using the same way as the previous one, every one of the substeps.
- Perform the steps in an iterative way until all the clusters have been grouped into only one cluster.

Next, we apply the function agglomerativeHC to the same data sample, with the same distance definition 'MAX', and we see that the result is the same that we have observed in the theoretical solution on the example. The full instructions are as follows:

```
agglomerativeHC(m, 'EUC', 'MAX')
```

Next, we learn how the algorithm operates with the instruction:

```
agglomerativeHC.details(m, 'EUC', 'MAX')
```

Next, the package can also be applied to learn more details about the technique. For example, the graphics of the clusters:

```
cmax<- agglomerativeHC(m, 'EUC', 'MAX')
plot(cmax$dendrogram)
```

C. Unsupervised Classification Exercises Solved

This section has two parts. In the first part, a set of exercises solved in detail are presented to allow you to check if all the knowledge has been correctly acquired. The advice is to try to solve the exercises by yourself and then to get the solution to check it with the proposed one by the book. This procedure will make this section truly useful for you. In the second part, the same exercises will be solved in R.

Handmade Exercises

1. It is known that the calculation speed of a microprocessor model depends on the temperature in a linear way, but it is also known that for different temperature intervals, the dependency functions (supervised classification) have different parameters. From the sample below, made up of the observations of temperatures and normalized speeds of 15 microprocessors, perform an analysis of unsupervised classification or clustering to establish which are the clusters for which the different functions should be defined (from the visual analysis of the data, it has been concluded that there is a high probability that there are three clusters) {speed, temperature}: 1. {3.5, 4.5 }; 2. {0.75, 3.25}; 3. {0, 3}; 4. {1.75, 0.75}; 5. {3, 3.75}; 6. {3.75, 4.5}; 7. {1.25, 0.75}; 8. {0.25, 3}; 9. {3.5, 4.25}; 10. {1.5, 0.5}; 11. {1, 1}; 12. {3, 4}; 13. {0.5, 3}; 14. {2, 0.25}; 15. {0, 2.5}. It must be done without and with computer.

To solve the exercise applying the K-means method, the following steps are applied:

A. Step A is known to have 3 substeps:

1. Selection of the number K of clusters, in which the data will be grouped and the centroids that will represent them. They are chosen arbitrarily by the user. The centroids will be the midpoints of the group of points (events) that make up the cluster. In this case, following the indications of the statement of the exercise, they will be three, and it must be taken arbitrarily, the centroids of the three clusters, that for the analysis of the sample, the values will be $c_1 = \{1, 1\}$, $c_3 = \{2, 2\}$ and$c_3 = \{3, 3\}$

   The Euclidean distance from each point to the three defined centroids.

   Point 1, {3.5, 4.5}:

$$d_{1c_1} = \sqrt{\sum_{i=1}^{2} (p_i - q_i)^2} = \sqrt{(1-3.5)^2 + (1-4.5)^2} = 4.3$$

$$d_{1c_2} = \sqrt{(2-3.5)^2 + (2-4.5)^2} = 2.92$$

$$d_{1c_3} = \sqrt{(3-3.5)^2 + (3-4.5)^2} = 1.58$$

   Point 2, {0.75, 3.25}:

$$d_{2c_1} = \sqrt{(1-0.75)^2 + (1-3.25)^2} = 2.26$$

$$d_{2c_2} = \sqrt{(2-0.75)^2 + (2-3.25)^2} = 1.77$$

$$d_{2c_3} = \sqrt{(3-0.75)^2 + (3-3.25)^2} = 2.26$$

   Point 3, {0, 3}:

$$d_{3c_1} = \sqrt{(1-0)^2 + (1-3)^2} = 2.24$$

$$d_{3c_2} = \sqrt{(2-0)^2 + (2-3)^2} = 2.24$$

$$d_{3c_3} = \sqrt{(3-0)^2 + (3-3)^2} = 3$$

   Point 4, {1.75, 0.75}:

$$d_{4c_1} = \sqrt{(1-1.75)^2 + (1-0.75)^2} = 0.79$$

$$d_{4c_2} = \sqrt{(2-1.75)^2 + (2-0.75)^2} = 1.27$$

$$d_{4c_3} = \sqrt{(3-1.75)^2 + (3-0.75)^2} = 2.57$$

Point 5, {3, 3.75}:

$$d_{5c_1} = \sqrt{(1-3)^2 + (1-3.75)^2} = 0.79$$
$$d_{5c_2} = \sqrt{(2-3)^2 + (2-3.75)^2} = 1.27$$
$$d_{5c_3} = \sqrt{(3-3)^2 + (3-3.75)^2} = 2.57$$

Point 6, {3.75, 4.5}:

$$d_{6c_1} = \sqrt{(1-3.75)^2 + (1-4.5)^2} = 4.45$$
$$d_{6c_2} = \sqrt{(2-3.75)^2 + (2-4.5)^2} = 3.05$$
$$d_{6c_3} = \sqrt{(3-3.75)^2 + (3-4.5)^2} = 1.68$$

Point 7, {1.25, 0.75}:

$$d_{7c_1} = \sqrt{(1-1.25)^2 + (1-0.75)^2} = 0.35$$
$$d_{7c_2} = \sqrt{(2-1.25)^2 + (2-0.75)^2} = 1.46$$
$$d_{7c_3} = \sqrt{(3-1.25)^2 + (3-0.75)^2} = 2.85$$

Point 8, {0.25 3}:

$$d_{8c_1} = \sqrt{(1-0.25)^2 + (1-3)^2} = 2.14$$
$$d_{8c_2} = \sqrt{(2-0.25)^2 + (2-3)^2} = 2.02$$
$$d_{8c_3} = \sqrt{(3-0.25)^2 + (3-3)^2} = 2.75$$

Point 9, {3.5, 4.25}:

$$d_{2c_1} = \sqrt{(1-3.5)^2 + (1-4.25)^2} = 4.1$$
$$d_{2c_2} = \sqrt{(2-3.5)^2 + (2-4.25)^2} = 2.7$$
$$d_{2c_3} = \sqrt{(3-3.5)^2 + (3-4.25)^2} = 1.35$$

Point 10, {1.5, 0.5}:

$$d_{3c_1} = \sqrt{(1-1.5)^2 + (1-0.5)^2} = 0.71$$
$$d_{3c_2} = \sqrt{(2-1.5)^2 + (2-0.5)^2} = 1.58$$
$$d_{3c_3} = \sqrt{(3-1.5)^2 + (3-0.5)^2} = 2.92$$

Point 11, {1, 1}:

$$d_{4c_1} = \sqrt{(1-1.75)^2 + (1-0.75)^2} = 0.79$$
$$d_{4c_2} = \sqrt{(2-1.75)^2 + (2-0.75)^2} = 1.27$$
$$d_{4c_3} = \sqrt{(3-1.75)^2 + (3-0.75)^2} = 2.57$$

Point 12, {3, 4}:

$$d_{5c_1} = \sqrt{(1-3)^2 + (1-4)^2} = 3.61$$
$$d_{5c_2} = \sqrt{(2-3)^2 + (2-4)^2} = 2.24$$
$$d_{5c_3} = \sqrt{(3-3)^2 + (3-4)^2} = 1$$

Point 13, {0.5, 3}:

$$d_{6c_1} = \sqrt{(1-0.5)^2 + (1-3)^2} = 2.06$$
$$d_{6c_2} = \sqrt{(2-0.5)^2 + (2-3)^2} = 1.8$$
$$d_{6c_3} = \sqrt{(3-0.5)^2 + (3-3)^2} = 2.5$$

Point 14, {2, 0.25}:

$$d_{7c_1} = \sqrt{(1-2)^2 + (1-0.25)^2} = 1.25$$
$$d_{7c_2} = \sqrt{(2-2)^2 + (2-0.25)^2} = 1.75$$
$$d_{7c_3} = \sqrt{(3-2)^2 + (3-0.25)^2} = 2.93$$

Point 15, {0, 0.25}:

$$d_{8c_1} = \sqrt{(1-0)^2 + (1-0.25)^2} = 1.8$$
$$d_{8c_2} = \sqrt{(2-0)^2 + (2-0.25)^2} = 2.06$$
$$d_{8c_3} = \sqrt{(3-0)^2 + (3-0.25)^2} = 3.04$$

2. Assignment of points or events to clusters. With the results obtained in step 2, a matrix of distances to the two centroids can be constructed. Taking into account the results of Step 2, the distance matrix is, in column centroids and in file points:

$$\begin{pmatrix} c_1 & c_2 & c_3 \\ 4.30 & 2.26 & 1.58 \\ 2.26 & 1.77 & 2.26 \\ 2.24 & 2.24 & 3.00 \\ 0.79 & 1.27 & 2.57 \\ 3.40 & 2.02 & 0.75 \\ 4.45 & 3.05 & 1.68 \\ 0.35 & 1.46 & 2.85 \\ 2.14 & 2.02 & 2.75 \\ 4.10 & 2.70 & 1.35 \\ 0.71 & 1.58 & 2.92 \\ 0.00 & 1.41 & 2.83 \\ 3.61 & 2.24 & 1.00 \\ 2.06 & 1.80 & 2.50 \\ 1.25 & 1.75 & 2.93 \\ 1.80 & 2.06 & 3.04 \end{pmatrix}$$

In consequence, starting from this matrix of distances, the matrix of assignments matrix is:

$$\begin{pmatrix} c_1 & c_2 & c_3 \\ 0 & 0 & 1 \\ 0 & 1 & 0 \\ 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & 0 \\ 1 & 0 & 0 \\ 1 & 0 & 0 \end{pmatrix}$$

3. Step B. Recalculation of centroids. Taking into account the assignment matrix:

$$c_1' = \left(\frac{1.75+1.25+1.5+1+2+0}{6}, \frac{0.75+0.75+0.5+1+0.25+2.5}{6}\right)$$
$$= (1.25, 0.95)$$

$$c_2' = \left(\frac{0.75+0+0.25+0.5}{4}, \frac{3.25+3+3+3}{4}\right) = (0.375, 3.0625)$$

$$c_3' = \left(\frac{3.5+3+3.75+3.5+3}{5}, \frac{4.5+3.75+4.5+4.25+4}{5}\right) = (3.35, 4.2)$$

4. Once we have the new centroids, we calculate the distances from the 15 points to the three centroids, and from these calculations, we obtain the new distance matrix:

$$\begin{pmatrix} c_1 & c_2 & c_3 \\ 4.20 & 3.44 & 0.34 \\ 2.35 & 0.42 & 2.77 \\ 2.40 & 0.38 & 3.56 \\ 0.54 & 2.69 & 3.80 \\ 3.30 & 2.71 & 0.57 \\ 4.34 & 3.67 & 0.50 \\ 0.20 & 2.47 & 4.04 \\ 2.28 & 0.14 & 3.32 \\ 3.99 & 3.34 & 0.16 \\ 0.51 & 2.80 & 4.14 \\ 0.25 & 2.16 & 3.97 \\ 3.52 & 2.79 & 0.40 \\ 2.18 & 0.14 & 3.09 \\ 1.03 & 3.25 & 4.17 \\ 1.99 & 0.68 & 3.76 \end{pmatrix}$$

Starting from this matrix of distances, we construct the matrix of assignments:

$$\begin{pmatrix} c_1 & c_2 & c_3 \\ 0 & 0 & 1 \\ 0 & 1 & 0 \\ 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{pmatrix}$$

As seen, the allocation is practically the same; only the fifteenth value has passed from the first cluster to the second, but even if there is only this small difference, we must calculate the centroids of these two clusters again, and the third one remains the same:

$$c_1'' = \left(\frac{1.75 + 1.25 + 1.5 + 1 + 2}{5}, \frac{0.75 + 0.75 + 0.5 + 1 + 0.25}{5} \right) = (1.5, 0.65)$$

$$c_2'' = \left(\frac{0.75 + 0 + 0.25 + 0.5 + 0}{5}, \frac{3.25 + 3 + 3 + 3 + 2.5}{5} \right) = (0.3, 2.95)$$

When changing the centroids, it is necessary to recalculate the distance from the points to them and make a new matrix of distances and assignments. The new distance matrix is:

$$\begin{pmatrix} c_1 & c_2 & c_3 \\ 4.34 & 3.56 & 0.34 \\ 2.71 & 0.54 & 2.77 \\ 2.79 & 0.30 & 3.56 \\ 0.27 & 2.63 & 3.80 \\ 3.44 & 2.82 & 0.57 \\ 4.46 & 3.78 & 0.50 \\ 0.27 & 2.40 & 4.04 \\ 2.66 & 0.07 & 3.32 \\ 4.12 & 3.45 & 0.16 \\ 0.15 & 2.73 & 4.14 \\ 0.61 & 2.07 & 3.97 \\ 3.67 & 2.90 & 0.40 \\ 2.55 & 0.21 & 3.09 \\ 0.64 & 3.19 & 4.17 \\ 2.38 & 0.54 & 3.76 \end{pmatrix}$$

Starting from this matrix of distances, we construct the matrix of assignments:

$$\begin{pmatrix} c_1 & c_2 & c_3 \\ 0 & 0 & 1 \\ 0 & 1 & 0 \\ 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{pmatrix}$$

It can be seen that it is the same as the last one obtained; therefore, the unsupervised classification or clustering has already ended, and we have three clusters with centroids:

$$c_1'' = (1.5, 0.65\ ), c_2'' = (0.3, 2.95) \text{ and } c_3'' = (3.35, 4.2)$$

5. Solve the Agglomerative Hierarchical Clustering, including the Cophenetic Distance Matrix, with the proximity definitions a.) MAX y b.) Group Average of the sample: {P1(2,1), P2(2,2), P3(3,5), P4(5,1), P5(5,3)}. Use the Euclidean distance.

The definition of a set of clusters from the Agglomerative Hierarchical Clusterization technique follows a process of 2 to n steps, which will be repeated until there is only one cluster:

Step A: Obtain the matrix of Euclidean distances between clusters. In this step, the distance matrix will be calculated, whose values will be the distances from each cluster to the rest of the clusters.

The first step of each iteration is the calculation of the matrix of Euclidean distances between all the clusters, which in the case of the first iteration is the calculation of the distances between all the points because each point is a cluster. We have combinations of 5 elements, points, taken 2 by 2.

$$C_5^2 = \frac{5!}{2!(5-2)!} = 10$$

The Euclidean distances are:

$$d_{12} = \sqrt{(2-2)^2 + (1-2)^2} = 1$$

$$d_{13} = \sqrt{(2-3)^2 + (1-5)^2} = 4.12$$

$$d_{14} = \sqrt{(2-5)^2 + (1-1)^2} = 3$$

$$d_{15} = \sqrt{(2-5)^2 + (1-3)^2} = 3.6$$

$$d_{23} = \sqrt{(2-3)^2 + (2-5)^2} = 3.16$$

$$d_{24} = \sqrt{(2-5)^2 + (2-1)^2} = 3.16$$

$$d_{25} = \sqrt{(2-5)^2 + (2-3)^2} = 3.16$$

$$d_{34} = \sqrt{(3-5)^2 + (5-1)^2} = 4.47$$

$$d_{35} = \sqrt{(3-5)^2 + (5-3)^2} = 2.83$$

$$d_{45} = \sqrt{(5-5)^2 + (1-3)^2} = 2$$

From these results, the distance matrix is:

$$\begin{pmatrix} & p_1 & p_2 & p_3 & p_4 & p_5 \\ p_1 & 0 & & & & \\ p_2 & 1 & 0 & & & \\ p_3 & 4.12 & 3.16 & 0 & & \\ p_4 & 3 & 3.16 & 4.47 & 0 & \\ p_5 & 3.6 & 3.16 & 2.83 & 2 & 0 \end{pmatrix}$$

Step B: Join the two closest clusters. In this step, the distances obtained will be ordered and a new cluster will be generated joining the two closest clusters. In the first iteration, each individual point will be considered as a cluster. We use the definition of proximity MAX, which we know defines as the proximity between two clusters as the distance between the two furthest points of the two clusters. It is also called Complete Link.

First Iteration: If we take the distance matrix between the clusters, considering that in the first iteration, each point constitutes a cluster.

$$\begin{pmatrix} & p_1 & p_2 & p_3 & p_4 & p_5 \\ p_1 & 0 & & & & \\ p_2 & \mathbf{1} & 0 & & & \\ p_3 & 4.12 & 3.16 & 0 & & \\ p_4 & 3 & 3.16 & 4.47 & 0 & \\ p_5 & 3.6 & 3.16 & 2.83 & 2 & 0 \end{pmatrix}$$

The two closest clusters are 1 and 2. Therefore, the first cluster, C1, is the one formed by these two points.

As we do not have a single cluster, we go to the second iteration.

Step A is carried out. Calculation of the matrix of distances between clusters.

The data are now {0.89, 2.94}; 2. {4.36, 5.21}; 4. {6.25, 3.14}; 6. {3.9, 4.27} and C1 {3. {3.75, 1.12}; 5. {4.1, 1.8}}

The distance matrix is:

$$\begin{pmatrix} & p_1 & p_2 & p_4 & p_6 & C1_{p3} & C1_{p5} \\ p_1 & 0 & & & & & \\ p_2 & 4.15 & 0 & & & & \\ p_3 & 3.39 & 4.13 & & & 0 & \\ p_4 & 5.36 & 2.80 & 0 & & 3.21 & \\ p_5 & 3.41 & 3.42 & 2.53 & & 0 & 0 \\ p_6 & 3.29 & 1.05 & 2.61 & 0 & 3.15 & 2.48 \end{pmatrix}$$

The distance between clusters is now between the four points 1, 2, 4 and 6, and cluster 1 is formed by points 3 and 5 in the previous iteration; consequently, the distance between points 3 and 5 is now 0 because they are in the same cluster.

Step B is now performed using the definition of proximity MAX. Data are now 1. {0.89, 2.94}; 2. {4.36, 5.21}; 4. {6.25, 3.14}; 6. {3.9, 4.27} and C1 {3. {3.75, 1.12}; 5. {4.1, 1.8}}. This first cluster is depicted in Fig. 5.16.

$$\begin{pmatrix} & p_1 & p_2 & p_4 & p_6 & C1_{p3} & C1_{p5} \\ p_1 & 0 & & & & & \\ p_2 & 4.15 & 0 & & & & \\ p_3 & 3.39 & 4.13 & & & 0 & \\ p_4 & 5.36 & 2.80 & 0 & & 3.21 & \\ p_5 & 3.41 & 3.42 & 2.53 & & 0 & 0 \\ p_6 & 3.29 & \mathbf{1.05} & 2.61 & 0 & 3.15 & 2.48 \end{pmatrix}$$

The two closest clusters are 2 and 6. Therefore, the second cluster, C2, is the one formed by these two points. This second cluster is depicted in Fig. 5.17.

As we do not have a single cluster, we go to the third iteration.

In Step A, the matrix of distances between points is calculated:

$$\begin{pmatrix} & p_1 & p_4 & C1_{p3} & C1_{p5} & C2_{p2} & C2_{p6} \\ p_1 & 0 & & & & & \\ p_2 & 4.15 & & & & 0 & \\ p_3 & 3.39 & & 0 & & 4.13 & \\ p_4 & 5.36 & 0 & 3.21 & & 2.8 & \\ p_5 & 3.41 & 2.53 & 0 & 0 & 3.41 & \\ p_6 & 3.29 & 2.61 & 3.15 & 2.48 & 0 & 0 \end{pmatrix}$$

The distance between clusters is now between the two points 1 and 4 and the clusters C1, formed by points 3 and 5, and C2 identified in the previous iteration; consequently, the distance between points 2 and 6 is now 0 because they are in the same cluster.

In Step B, the two closest clusters are merged. The data are now: 1. {0.89, 2.94}; 4. {6.25, 3.14}, C1 {3. {3.75, 1.12}; 5. {4.1, 1.8}}, and C2 {2. {4.36, 5.21}; 6. {3.9, 4.27}}

	p_1	p_4	$C1_{p3}$	$C1_{p5}$	$C2_{p2}$	$C2_{p6}$
p_1	0					
p_2	4.15				0	
p_3	3.39		0		4.13	
p_4	5.36	0	3.21		**2.8**	
p_5	3.41	2.53	0	0	3.41	
p_6	3.29	2.61	3.15	2.48	0	0

The two closest clusters are point 4 and C2. Therefore, the third cluster, C3, is the one formed by these two clusters (Fig. 5.23).

As we do not have a single cluster, we proceed to the fourth iteration, but before, it is important at this point to provide a deep explanation about why 2.8 is the minimum distance between clusters with the MAX definition distance algorithm.

The proximity between clusters with MAX is defined as the distance between the two farthest points of the two clusters:

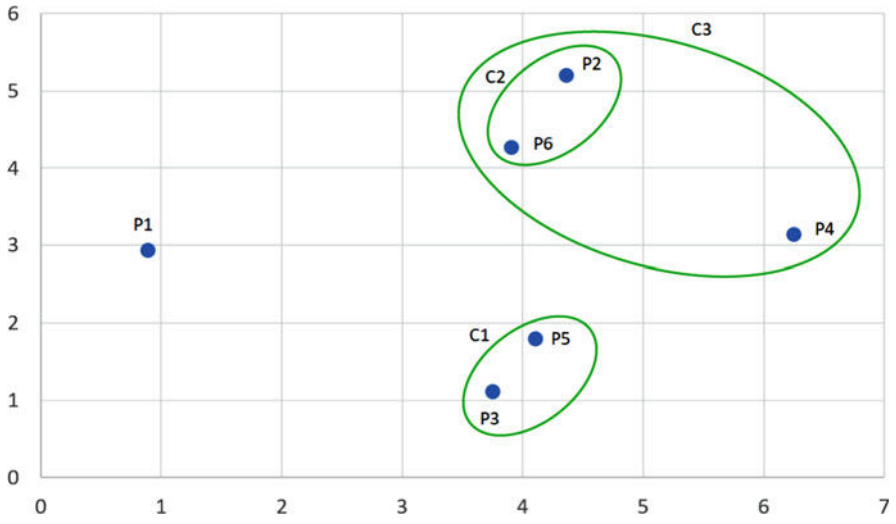


Fig. 5.23 Third cluster MAX distances

- The first two clusters considered are points 1 and 4, for which the distances, as seen in the previous matrix, are: {4.15, 3.39, 5.36, 3.41, 3,29, 2,53, 2,61}, and from all of them, the maximum, MAX, is 5.36.
- The second two clusters considered are C1 and C2; all the distances between all the points in both clusters must be calculated, that is, distance (3.2), distance (3.6), distance (5.2), and distance (5.6), which are: {4.13, 3.15, 3.42, 2.48}, and from all of them to take the maximum, that is, 4.13.
- The third two clusters considered are C1 and Point 1; the distances to be analyzed are (3,1) and (5,1), which are: {3.39, 3.41}, and the maximum is 3.41.
- The fourth two clusters to be considered are C1 and Point 4, the distances to be compared are (3,4) and (5,4), that are: {3.21, 2.53} and the maximum is 3.21.
- And fifth, two clusters to be compared are C2 and Point 1; the distances to be compared are (2,1) and (6,1), which are: {4.15, 3.29}, and the maximum is 4.15
- And finally, the last two clusters to be compared are C2 and Point 4; the distances to be compared are (2,4) and (6,4), which are {2.8, 2,61}, and the maximum is 2.8.

Once you have all the distances between the clusters with the algorithm MAX, the minimum of all of them will be the selected one to merge both clusters in one, the distances obtained are: {5.6, 4.13, 3.41, 3.21, 4.15 and 2,8}, the minimum of all of them is 2.8, and in consequence the clusters merged in this iteration are C2 and Point 4.

In Step A of the fourth iteration of the agglomerative hierarchical classification algorithm with the distance definition MAX, the new matrix of distances between clusters is obtained.

$$\begin{pmatrix} & p_1 & C1_{p3} & C1_{p5} & C3-C2_{p2} & C3-C2_{p6} & C3_{p4} \\ p_1 & 0 & & & & & \\ p_2 & 4.15 & & & 0 & & \\ p_3 & 3.39 & 0 & & 4.13 & & \\ p_4 & 5.36 & 3.21 & & 0 & & 0 \\ p_5 & 3.41 & 0 & 0 & 3.42 & & 2.53 \\ p_6 & 3.29 & 3.15 & 2.48 & 0 & 0 & 0 \end{pmatrix}$$

The distance between clusters is now between Point 1 and the clusters, C1 formed by points 3 and 5, and C3 identified in the previous iteration, and consequently, the distance between Point 4 and cluster C2 is 0 because they are in the same cluster.

In Step B, the two closest clusters are merged. The data are now: 1. {0.89, 2.94}; C1 {3. {3.75, 1.12}; 5. {4.1, 1.8}} and C3 {C2 {2. {4.36, 5.21}; 6. {3.9, 4.27}}, 4. {6.25, 3.14}}

$$\begin{pmatrix} & p_1 & C1_{p3} & C1_{p5} & C3-C2_{p2} & C3-C2_{p6} & C3_{p4} \\ p_1 & 0 & & & & & \\ p_2 & 4.15 & & & 0 & & \\ p_3 & 3.39 & 0 & & 4.13 & & \\ p_4 & 5.36 & 3.21 & & 0 & & 0 \\ p_5 & \mathbf{3.41} & 0 & 0 & 3.42 & & 2.53 \\ p_6 & 3.29 & 3.15 & 2.48 & 0 & 0 & 0 \end{pmatrix}$$

The distances applying MAX are:

- The first two clusters considered are Points 1 and C1, for which the distances, as seen in the previous matrix, are 3.39 between points 1 and 3 and 3.41 between points 1 and 5, and from both of them, the maximum, MAX, is 3.41.
- The second and final two clusters considered are Points 1 and C3, for which the distances, as seen in the previous matrix, are 4.15 between points 1 and 2, 3.29 between points 1 and 6, and 5.36 between points 1 and 4, and from all of them, the maximum, MAX, is 5.36.

Once you have both the distances between the clusters with the MAX algorithm, the minimum of both of them will be the selected one to merge both clusters into one. Since the distances obtained are 3.41 and 5.36, the minimum of both of them is 3.41, and consequently, the clusters merged in this iteration are C1 and Point 1, that is, C4.

The previous result means that the algorithm has finished because a final unique cluster is obtained, that one constituted by the previous cluster C3 and the new cluster C4, that will be C5 (Fig. 5.24).

- *Group Average.* Define the proximity between two clusters as the average of the distances between all the pairs that can be formed with points from the two clusters:

$$\text{proximity}\left(C_i, C_j\right) = \frac{\sum_{i=1}^{m}\sum_{j=1}^{n} \text{proximity}\left(x_i, y_j\right)}{m * n}$$

Over the sample that has been solved, carry out step B of the agglomerative hierarchical clustering algorithm using the algorithm with the group average proximity definition (see Fig. 5.14):

First Iteration: If we take the distance matrix between the clusters, we consider that in the first iteration, each point constitutes a cluster.

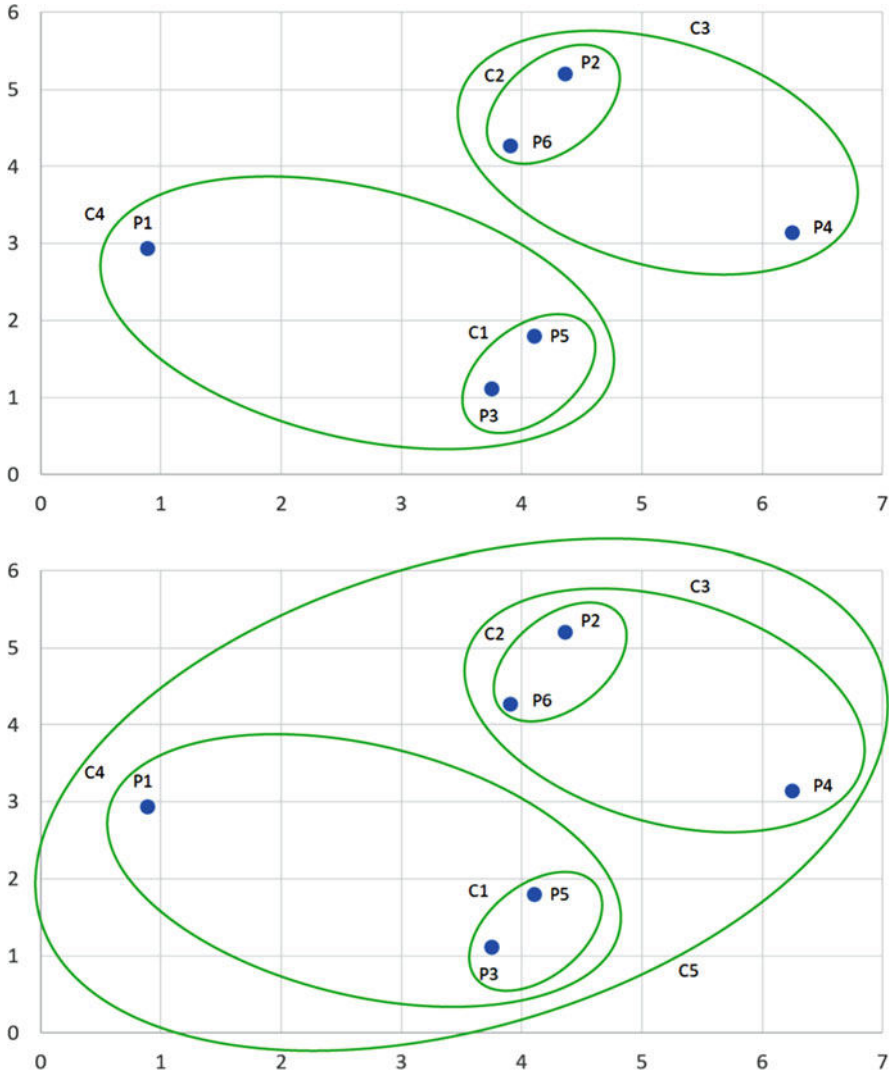


Fig. 5.24 Fourth and fifth clusters MAX distances

$$
\begin{pmatrix}
 & p_1 & p_2 & p_3 & p_4 & p_5 & p_6 \\
p_1 & 0 & & & & & \\
p_2 & 4.15 & 0 & & & & \\
p_3 & 3.39 & 4.13 & 0 & & & \\
p_4 & 5.36 & 2.80 & 3.21 & 0 & & \\
p_5 & 3.41 & 3.42 & \mathbf{0.76} & 2.53 & 0 & \\
p_6 & 3.29 & 1.05 & 3.15 & 2.61 & 2.48 & 0
\end{pmatrix}
$$

The two closest clusters are 3 and 5. Therefore, the first cluster, C1, is the one formed by these two points (see Fig. 5.16).

As we do not have a single cluster, we go to the second iteration.

Step A of the second iteration of the agglomerative hierarchical classification algorithm with the distance definition group average is performed, and the new data are as follows: 1. {0.89, 2.94}; 2. {4.36, 5.21}; 4. {6.25, 3.14}; 6. {3.9, 4.27} and C1 {3. {3.75, 1.12}; 5. {4.1, 1.8}}

	p_1	p_2	p_4	p_6	$C1_{p3}$	$C1_{p5}$	p_1	p_2	p_4	p_6	$C1_{p3}$	$C1_{p5}$
p_1	0						0					
p_2	4.15	0					4.15	0				
$C1_{p3}$	3.39	4.13			0		**3.40**	**3.78**			0	
p_4	5.36	2.80	0		3.21		5.36	2.80	0		**2.87**	
$C1_{p5}$	3.41	3.42	2.53		0	0	**3.40**	**3.78**	**2.87**		0	0
p_6	3.29	1.05	2.61	0	3.15	2.48	3.29	1.05	2.61	0	**2.82**	**2.82**

The distance between clusters is now between points 1, 2, 4 and 6, and cluster 1 is formed by points 3 and 5 in the previous iteration, but now the distances change because it is done with the mean.

$$\text{proximity}\,(p_1, C_1) = \frac{\sum_{i=1}^{2}\sum_{j=1}^{1}\text{proximity}\,((p1,p3),(p1,p5))}{2*1} = \frac{3.39+3.41}{2} = 3.40$$

$$\text{proximity}\,(p_2, C_1) = \frac{\sum_{i=1}^{2}\sum_{j=1}^{1}\text{proximity}\,((p2,p3),(p2,p5))}{2*1} = \frac{4.13+3.42}{2} = 3.78$$

$$\text{proximity}\,(p_4, C_1) = \frac{\sum_{i=1}^{2}\sum_{j=1}^{1}\text{proximity}\,((p4,p3),(p4,p5))}{2*1} = \frac{3.21+2.53}{2} = 2.87$$

$$\text{proximity}\,(p_6, C_1) = \frac{\sum_{i=1}^{2}\sum_{j=1}^{1}\text{proximity}\,((p6,p3),(p6,p5))}{2*1} = \frac{3.15+2.48}{2} = 2.82$$

6. In Step B, the minimum distance between clusters using the Group Average algorithm is used to select which two closest clusters must be merged. The data are now: 1. {0.89, 2.94}; 2. {4.36, 5.21}; 4. {6.25, 3.14}; 6. {3.9, 4.27} and C1 {3. {3.75, 1.12}; 5. {4.1, 1.8}}

	p_1	p_2	p_4	p_6	$C1_{p3}$	$C1_{p5}$
p_1	0					
p_2	4.15	0				
$C1_{p3}$	3.40	3.78			0	
p_4	5.36	2.80	0		2.87	
$C1_{p5}$	3.40	3.78	2.87		0	0
p_6	3.29	**1.05**	2.61	0	2.82	2.82

The two closest clusters are points 2 and 6. Therefore, the second cluster, C2, is the one formed by these two points (see Fig. 5.17).

As we do not have a single cluster, we go to the third iteration.

Step A of the third iteration of the agglomerative hierarchical classification algorithm with the distance definition group average is performed, and the new data are as follows: 1. {0.89, 2.94}; 4. {6.25, 3.14}; C1{3. {3.75, 1.12}; 5. {4.1, 1.8}, C2{2. {4.36, 5.21}; 6. {3.9, 4.27}}}

	p_1	p_4	$C1p_3$	$C1p_5$	$C2_{p2}$	$C2_{p6}$	p_1	p_4	$C1p_3$	$C1p_5$	$C2_{p2}$	$C2_{p6}$
p_1	0						0					
$C2p_2$	4.15				0		**3.72**				0	
$C1_{p3}$	3.39		0		4.13		3.40		0		**3.30**	
p_4	5.36	0	3.21		2.80		5.36	0	2.87		**2.70**	
$C1_{p5}$	3.41	2.53	0	0	3.42		3.40	2.87	0	0	**3.30**	
$C2p_6$	3.29	2.61	3.15	2.48	1.05	0	**3.72**	**2.70**	**3.30**	**3.30**	**3.30**	**0**

The distance between clusters is now between the two points 1 and 4, and the two clusters 1 and 2, but now almost all the distances change because it is done with the mean:

$$\text{proximity}\,(C_1, C_2) = \frac{\sum_{i=1}^{2}\sum_{j=1}^{2} \text{proximity}\,((p3,p2),(p3,p6),(p5,p2),(p5,p6))}{2*2}$$
$$= \frac{4.13 + 3.15 + 3.42 + 2.48}{4} = 3.30$$

$$\text{proximity}\ (p_1, C_2) = \frac{\sum_{i=1}^{2}\sum_{j=1}^{1} \text{proximity}\ ((p1, p2), (p1, p6))}{2 * 1} = \frac{4.15 + 3.29}{2} = 3.72$$

$$\text{proximity}\ (p_4, C_2) = \frac{\sum_{i=1}^{2}\sum_{j=1}^{1} \text{proximity}\ ((p4, p2), (p4, p6))}{2 * 1} = \frac{2.80 + 2.61}{2} = 2.70$$

7. In Step B, the minimum distance between clusters using the Group Average algorithm is used to select which two closest clusters must be merged. The data are now: 1. {0.89, 2.94}; 4. {6.25, 3.14}; C1{3. {3.75, 1.12}; 5. {4.1, 1.8}, C2 {2. {4.36, 5.21}; 6. {3.9, 4.27}}}

	p_1	p_4	$C1p_3$	$C1p_5$	$C2_{p2}$	$C2_{p6}$
p_1	0					
$C2p_2$	3.72				0	
$C1_{p3}$	3.40		0		3.30	
p_4	5.36	0	2.87		**2.70**	
$C1_{p5}$	3.40	2.87	0	0	3.30	
$C2p_6$	3.72	**2.70**	3.30	3.30	3.30	0

The two closest clusters are Points 4 and C2. Therefore, the third cluster, C3, is the one formed by these two clusters (see Fig. 5.18).

As we do not have a single cluster, we go to the fourth iteration.

8. Step A of the fourth iteration of the agglomerative hierarchical classification algorithm with the distance definition group average is performed, and the new data are as follows: 1. {0.89, 2.94}; 4. {6.25, 3.14}; C1{3. {3.75, 1.12}; 5. {4.1, 1.8}, C2{2. {4.36, 5.21}; 6. {3.9, 4.27}}}

	p_1	$C1p_3$	$C1p_5$	$C3.C2p_2$	$C3.C2_{p6}$	$C3_{p4}$	p_1	$C1p_3$	$C1p_5$	$C3.C2p_2$	$C3.C2_{p6}$	$C3_{p4}$
p_1	0						0					
$C3.C2p_2$	4.15			0			**4.27**			0		
$C1_{p3}$	3.39	0		4.13			3.40	0		**3.15**		
$C3p_4$	5.36	3.21		2.80		0	**4.27**	**3.15**		**0**		**0**
$C1_{p5}$	3.41	0	0	3.42		2.53	3.40	0	0	**3.15**		**3.15**
$C3.C2p_6$	3.29	2.61	2.48	0	0	2.61	**4.27**	**3.15**	**3.15**	**0**	0	**0**

The distance between clusters is now between point 1 and the two clusters C1 and C3, but now almost all the distances change because it is done with the mean.

$$\text{proximity}\ (C_1, C_3)$$

$$= \frac{\sum_{i=1}^{3}\sum_{j=1}^{2} \text{proximity}\ ((p3,p2),(p3,p6),(p3,p4)(p5,p2),(p5,p6),(p5,p4))}{2*3}$$

$$= \frac{4.13+3.15+3.21+3.42+2.48+2.53}{6} = 3.15$$

$$\text{proximity}\ (p_1, C_3) = \frac{\sum_{i=1}^{3}\sum_{j=1}^{1} \text{proximity}\ ((p1,p2),(p1,p6)(p1,p4))}{3*1}$$

$$= \frac{4.15+3.29+5.36}{3} = 4.27$$

9. In Step B, the minimum distance between clusters using the Group Average algorithm is used to select which two closest clusters must be merged. The data are now P1, C1, and C3.

	p_1	$C1p_3$	$C1p_5$	$C3.C2p_2$	$C3.C2_{p6}$	$C3_{p4}$
p_1	0					
$C3.C2p_2$	4.27			0		
$C1_{p3}$	3.40	0		**3.15**		
$C3p_4$	4.27	**3.15**		0		0
$C1_{p5}$	3.40	0	0	**3.15**		**3.15**
$C3.C2p_6$	4.27	**3.15**	**3.15**	0	0	0

The distances applying Group Average are:

- The first two clusters considered are Points 1 and C1, for which the mean distance applying the group average is 3.40. With this algorithm, as it is a mean, only one distance will be used.
- The second and final two clusters considered are Points 1 and C3, for which the mean distance applying the group average is 4.27.
- Finally, the third two clusters considered are C1 and C3, for which the mean distance applying the group average is 3.15.

Once you have all the distances between the clusters with the algorithm Group Average, the minimum of all of them will be the selected one to merge both clusters into one. Since the distances obtained are 3.40, 4.27 and 3.15, the minimum is 3.15, and consequently, the clusters merged in this iteration are C1 and C3, that is, C4.

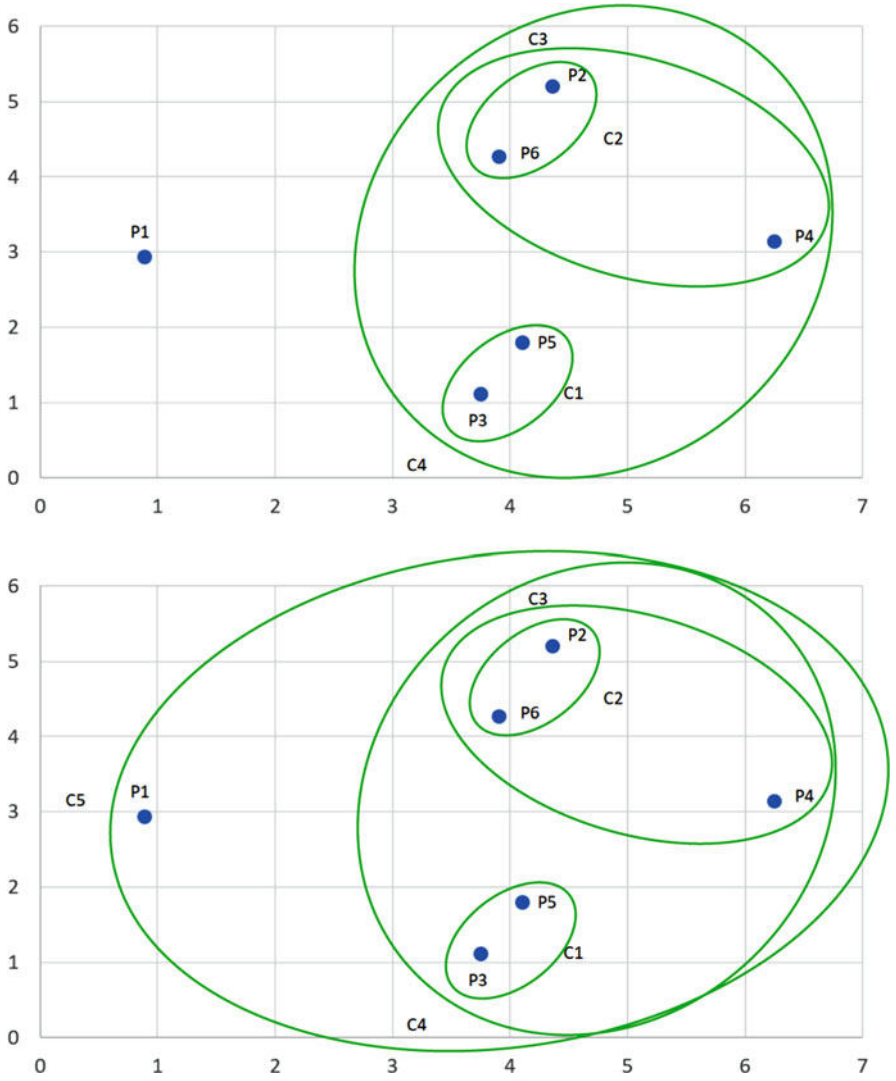


Fig. 5.25 Fourth and fifth cluster group average distances

The previous result means that the algorithm has finished because a final unique cluster is obtained, that one constituted by the new cluster C4, and the point, or cluster, 1, that will be C5 (Fig. 5.25).

Exercises Solved in R

In this section, the previous exercises will be solved using R software.

1. It is known that the calculation speed of a microprocessor model depends on the temperature in a linear way, but it is also known that for different temperature intervals, the dependency functions (supervised classification) have different parameters. From the sample below, made up of the observations of temperatures and normalized speeds of 15 microprocessors, perform an analysis of unsupervised classification or clustering to establish which are the clusters for which the different functions should be defined (from the visual analysis of the data, it has been concluded that there is a high probability that there are three clusters) {speed, temperature}: 1. {3.5, 4.5 }; 2. {0.75, 3.25}; 3. {0, 3}; 4. {1.75, 0.75}; 5. {3, 3.75}; 6. {3.75, 4.5}; 7. {1.25, 0.75}; 8. {0.25, 3}; 9. {3.5, 4.25}; 10. {1.5, 0.5}; 11. {1, 1}; 12. {3, 4}; 13. {0.5, 3}; 14. {2, 0.25}; 15. {0, 2.5}. Solve it now with computer and R, using the k-means algorithm.

Solution: Similar to the previous sections with the function k-means (m, cen-ters, iter.max). We start by introducing the matrix m, with the data to analyze in our case:

$$x = \begin{pmatrix} 3.5 & 4.5 \\ 0.75 & 3.25 \\ 0 & 3 \\ 1.75 & 0.75 \\ 3 & 3.75 \\ 3.75 & 4.5 \\ 1.25 & 0.75 \\ 0.25 & 3 \\ 3.5 & 4.25 \\ 1.5 & 0.5 \\ 1 & 1 \\ 3 & 4 \\ 0.5 & 3 \\ 2 & 0.25 \\ 0 & 2.5 \end{pmatrix}$$

To introduce them, we use the matrix function, transpose it and display it.

> (m <-t (matrix (c (3.5,4.5, 0.75,3.25, 0, 3, 1.75,0.75, 3,3.75, 3.75,4.5, 1.25,0.75, 0.25,3, 3.5, 4.25,1.5,0.5, 1,1,3,4,0.5,3,2,0.25,0,2.5), 2, 15)))

The second argument is the centers, which in the exercise asks us to be three and which we will introduce using the instruction:

(c <-t (matrix (c (1,1, 2,2, 3,3), 2, 3)))

which introduces the matrix with the same centroids that we used in the theoretical resolution of the problem:

$$c = \begin{pmatrix} 1 & 1 \\ 2 & 2 \\ 3 & 3 \end{pmatrix}$$

Finally, in the third argument {iter.max}, we are going to indicate, as in the previous case, that there are 4.

If we introduce the values of the three arguments in the k-means function, we have the instruction:

>(classifications = (kmeans (m, c, 4)))

and we obtain the following result of the unsupervised classification or clustering:
K-means clustering with 3 clusters of sizes 5, 5, 5
Cluster means:

[, 1] [, 2]
1 1.50 0.65
2 0.30 2.95
3 3.35 4.20
Clustering vector:
[1] 3 2 2 1 3 3 1 2 3 1 1 3 2 1 2

Next, we obtain the matrix for each cluster, similar to the first exercise. First, we add a column to the matrix m to put each data point in its corresponding cluster

>(m = cbind (classifications \ $ cluster, m))

Then, with the subset instruction, we obtain the three arrays. The complete instruction is

>mc1 = subset (m, m [, 1] == 1)

>mc2 = subset (m, m [, 1] == 2)

>mc3 = subset (m, m [, 1] == 3)

And we end up eliminating the column that indicates the cluster

>(mc1 = mc1 [, - 1])

>(mc2 = mc2 [, - 1])

>(mc3 = mc3 [, - 1])

2. It is known that the calculation speed of a microprocessor model depends on the temperature in a linear way, but it is also known that for different temperature intervals, the dependency functions (supervised classification) have different parameters. From the sample below, made up of the observations of temperatures and normalized speeds of 15 microprocessors, perform an analysis of

unsupervised classification or clustering to establish which are the clusters for which the different functions should be defined (from the visual analysis of the data, it has been concluded that there is a high probability that there are three clusters) {speed, temperature}: 1. {3.5, 4.5 }; 2. {0.75, 3.25}; 3. {0, 3}; 4. {1.75, 0.75}; 5. {3, 3.75}; 6. {3.75, 4.5}; 7. {1.25, 0.75}; 8. {0.25, 3}; 9. {3.5, 4.25}; 10. {1.5, 0.5}; 11. {1, 1}; 12. {3, 4}; 13. {0.5, 3}; 14. {2, 0.25}; 15. {0, 2.5}. Solve it now with computer and R, using Hierarchical Clustering.

To solve the Agglomerative Hierarchical clustering problem with R, we again use the package LearnClust. To do that, we first introduce the sample data in R, and we do it, as in the previous case, with the matrix:

```
(m <-t (matrix (c (3.5,4.5, 0.75,3.25, 0, 3, 1.75,0.75, 3,3.75, 3.75,4.5, 1.25,0.75,
    0.25,3, 3.5, 4.25,1.5,0.5, 1,1,3,4,0.5,3,2,0.25,0,2.5), 2, 15)))
```

and following, we assign the value m to its transpose:

```
(m<-t(m))
```

Once we have introduced the data we obtain, we calculate them hierarchical agglomerative clusterization using the *Euclidean* distance and the proximity definition *MIN*. To do that, we use the function included in the package agglomerativeHC, with the parameters m, which indicates the pair of data that we are going to cluster, 'EUC', which indicates the type of distance that we are going to use, that in this case is the Euclidean, and 'MIN', which indicates the type of proximity that we are going to use, that are me closets points. With all of this, the instruction is:

```
agglomerativeHC(m, 'EUC', 'MIN')
```

However, we now that the package provides the hierarchical agglomerative clustering solution but also teaches how it works. To achieve this second objective, we know thT the.detail functions are included, which explain how the algorithms that implement the functions that do not have that extension work.

```
agglomerativeHC.details(m, 'EUC', 'MIN')
```

Next, we apply the function agglomerativeHC to the same data sample, with the same distance definition ‘MAX’, and we see that the result is the same that we have observed in the theoretical solution on the example. The full instructions are as follows:

```
agglomerativeHC(m, 'EUC', 'MAX')
```

Next, we learn how the algorithm operates with the instruction:

```
agglomerativeHC.details(m, 'EUC', 'MAX')
```

Next, the package can also be applied to learn more details about the technique. For example, the graphics of the clusters:

```
cmax<- agglomerativeHC(m, 'EUC', 'MAX')

plot(cmax$dendrogram)
```

Supervised Classification

In this sixth chapter, we present the theoretical foundations of the supervised classification[1] and the main techniques used to carry it out.

Section A introduces, in a theoretical and, at the same time, practical ways all the basic theoretical knowledge related to supervised classification, that is, the concepts and techniques that allow us to perform the analysis from the decision trees to the regression functions.

Section B presents the computer-based solving of the same examples used in section A to introduce theoretical knowledge. Section B presents the computer-based solving. The packages needed to carry out these computational solutions are also introduced.

Section C consists of a set of statements of exercises about supervised classification in which detailed solutions can also be found.[2]

A. Theory

This first section of the chapter is structured in 5 subsections: 1. introduction, 2. decision trees, 3. neural networks, 4. naïve Bayes, and 5. regression functions.

[1] The supervised classification of events is called *supervised* because the values of the characteristic to be classified will be classified into classes whose values have been previously defined, while that in the case of unsupervised classification, the values that define the different classes, centroids, are determined during the same classification process. In the event that the term classification is used without saying whether it is supervised or not, it usually refers to supervised classification.

[2] We repeat again here that it is very important in order to obtain the best results for the learning process throughout the use of the book, that the reader tries to solve the exercises by himself/herself before seeing their solutions, and that only once solved check whether the obtained solutions are correct.

J. J. Cuadrado-Gallego, Y. Demchenko, *Data Analytics*,
https://doi.org/10.1007/978-3-031-39129-3_6

Introduction

Supervised classification studies seek to obtain a function, called a *classification model*, that allows obtaining the value of a certain characteristic of an object from the data that the rest of the characteristics of the same object. The parameters that define said function or classification model are determined using a definition or training sample in which, for each set of objects, there are the values of all the characteristics applicable to them, including those of the characteristic for which the classification model is being sought.

To introduce the concept of classifying events with qualitative values through an example, we use the grades of a subject from a group of students. The qualifications will be made up of four marks, corresponding to the Theory, Laboratory, Practices and Global Qualification tests. The elementary events that make up each event student grades are each individual grade: E = {Theory, Laboratory, Practice, Overall Grade}.

The elementary event that will be used as a classifier will be the global qualification, and following what is indicated in the theory of the subject, the supervised classification begins by establishing the possible values of each elementary classifying event. The two possible values for the global grade are Approved, Ap, and Fail, Ss, which allow establishing two complementary and disjoint equivalence classes, which will encompass all events (students) with a global pass grade and which will include events with a global rating of failure.

The rest of the events will have the following four values: A, B, C, and D, where A will be the highest possible rating and D the lowest. The classification function sought in this case will be the one that, based on the values of the first three elementary events, allows classifying an event, or what is the same, based on the qualifications in theory, laboratory, and practical, to obtain the overall student grade.

As mentioned above, different classification techniques can be used to obtain a classification function for events. This differentiation is based on the fact that each of them uses different algorithms. All of them use a sample or set of events for which the values of all elementary events are known, including the one that marks the classes, to define the classification function. Once defined, the function will be used to classify new events. The sample of events that will allow you to find the definition of the classification function is made up of eight events – 1. {A, A, B, Ap}; 2. {A, B, D, Ss}; 3. {D, C, C, Ss}; 4. {D, B, A, Ss}; 5. {B, C, D, Ss}; 6. {C, B, B, Ap}; 7. {B, B, A, Ap}; 8. {C, D, C, Ss} – which are a set of academic qualifications made up of four grades: Theory, Laboratory, Practices and Global Qualification. Starting from this, the elementary events are as follows:

E = <Theory, Laboratory, Practices, Global Qualification>

The equivalence classes of a classifier event are as follows:

Classifier = Global Qualification, which has two complementary and disjoint equivalence classes: Pass and Fail

The classifier function based on the values A, B, C, and D of the first three events will define the value of the fourth, Pass or Fail.

Different classification techniques can be used to obtain a classification function for events. This differentiation is based on the fact that each of them uses different algorithms. All of them use a sample or set of events for which the values of all elementary events are known, including the one that marks the classes, to define the classification function. Once defined, the function will be used to classify new events.

Some of the best known and most used are as follows:

- Decision trees
- Neural networks
- Naïve Bayes
- Regression

We are going to see how each of them works in a specific way.

Decision Trees

Decision trees. Hunt's algorithm for supervised classification.

The definition of the *decision tree* follows a process[3] from 1 to k steps, where k is the number of elementary events that allow the analysed event to be fully classified. The maximum number of steps will be $n-1$ since if n is the dimension of the set P (E) that is being studied or the number of elementary events that make up the analysed events, one of them will be used as a classifier. Let us see how each step[4] is treated.

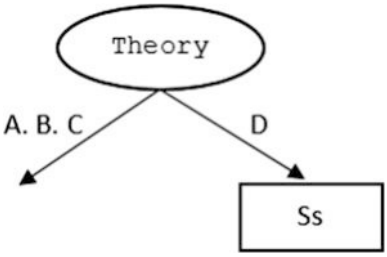
A. Step A. Select an elementary event from the members of the event that is being classified and analyse the possible results it may have. Any of the events to be analysed can be selected to carry out this first step, except for the elementary event used as a classifier. This first event is called the root node because it has no other previous event analysed, and after it, from zero to k more elementary events will be analysed. Once the analysis has been carried out, it is determined

[3] Another way of defining how Hunt's algorithm works that it is complementary to the one explained above and can help improve understanding of the concept is: Hunt's algorithm follows a process based on the 1 to k steps explained in the overview of defining a decision tree. To carry out each step, that is, to define each node, a procedure is applied based on the observation of the class to which all the events of the sample belong, used to find the classification function not classified in one step (node) previous. Depending on the result of said observation, one of the following two actions will be carried out:

1. If they all belong to the same class, the observed node is a terminal node or leaf.
2. If they all do not belong to the same class, an elementary event is selected, a component of the analyzed events, whose value will allow us to divide the set of events into subsets. For each of the defined subsets, an intermediate node will be created in which the process will be repeated.

[4] To follow the same structure as in topic 2, we are going to call them steps A and B.

Fig. 1 First classification



whether it allows you to fully classify any event to which it belongs or, if not, go to the next step and analyse another elementary event.

Choosing the root node. Of the four elementary events {Theory, Laboratory, Practices, Global Rating}, the last, Global Rating is the one that marks the class, so, following what has been seen in the theory, we cannot take it as the initial node. We take any of the other three as the initial node. In principle, we can choose it arbitrarily; later, we will see how to optimize the tree, but now we will only see how it is built, so we choose, for example, Theory as the initial node. Once chosen, we will analyse the relationship of its values with the Global Rating value. We analyse the events in the sample one by one; the first value is that of Theory and the second is that of Global Rating: 1. A, Ap; 2. A, Ss; 3. D, Ss; 4. D, Ss; 5. B, Ss; 6. C, Ap; 7. B, Ap; 8. C, Ss.

As you can check if you have a theory grade of A, B, or C, you can have a final grade of both Ap, as in cases 1, 6, and 7, and of Ss, as in cases 2, 5, and 8, whereas if you have a theory grade of D, you always have a grade of Ss. Consequently, Theory cannot be considered as a final node because it allows classifying the events whose classification is D, but not those that are A, B, or C. Consequently, the classification, after passing through the initial node Theory, would be (Fig. 1):

And events 3 and 4 have been classified.

B. In step B, and following, if there are any, intermediate nodes are arbitrarily chosen, which are new characteristics on which an analysis similar to that of step 1 is going to be carried out, until the object is fully classified based on the values of the classifier character.

Step B will be performed k times, with a maximum k equal to $n-1$.[5] We may find the classification model before using the $n-1$ events, that is, using k events, in which case we would have k steps. It may also be that we will not find it even using the $n-1$ events or steps, in which case there would be no classification model. Let us now see how step 2 would be carried out[6] and $n-1$, and all the others would be carried out following the same process:

[5] Since we have n events and one of them is the classifying event.

[6] Since step 1 is step A.

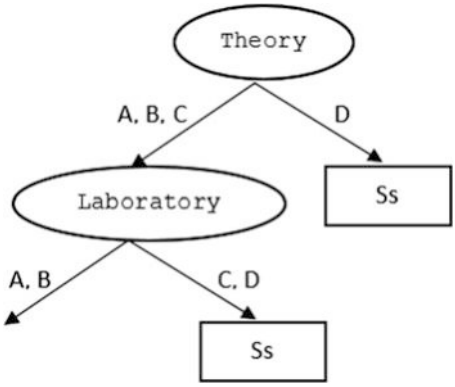


Fig. 2 Second classification

Step 2.

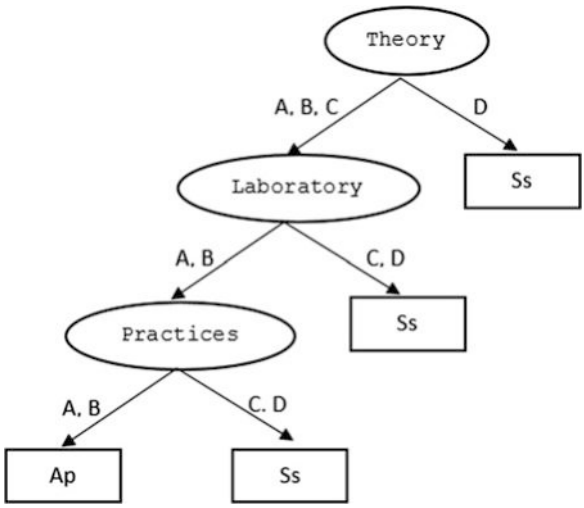
If the result of the elementary event analysed in the node allows classifying the event that it belongs to, it is passed to a set of leaf nodes or terminal nodes in which it is specified, depending on the values that the analysed elementary event may have, the values of the classes to which the event can belong. If, on the other hand, the result of the elementary event does not allow classifying the event to which it belongs, then an internal node applied to another elementary event is passed, and the same analysis is carried out as in step A.

Analysis of the internal node. Once the elementary event Theory has been analysed, we have seen that it does not allow us to fully classify the events in the sample and, consequently, any event that we analyse later, so we move on to an internal node. In the same way that when we analyse the initial node, we arbitrarily choose the elementary event for the first internal node and, as has also been said in step A, we will see later how this choice is optimized. We choose Laboratory and analyse the events in the sample that still remain unclassified, which are 1, 2, 5, 6, 7, and 8. We analyse the value of Laboratory and the Global Qualification[7]: 1. A, Ap; 2. B, Ss; 3. Classified in Step A. 4. Classified in Step A. 5. C, Ss; 6. B, Ap; 7. B, Ap; 8. D, Ss.

As you can check, if you have a laboratory grade of A or B, you can have a final grade of both Ap, as in cases 1, 6, and 7, and Ss, as in case 2. However, if you have a grade of C or D, you always have a grade of Ss. Therefore, Laboratory cannot be considered as a final node because it allows classifying events whose qualification is either C or D but not those that are A or B. Consequently, the classification, after passing through the intermediate Laboratory node, would be (Fig. 2):

[7] We repeat the Sample here to be able to see the values more clearly: 1. {A, A, B, Ap}; 2. {A, B, D, Ss}; 3. {D, C, C, Ss}; 4. {D, B, A, Ss}; 5. {B, C, D, Ss}; 6. {C, B, B, Ap}; 7. {B, B, A, Ap}; 8. {C, D, C, Ss}

Fig. 3 Final classification



And events 5 and 8 have been classified.

Step $n-1$. It would be done as step 2.

Terminal node. In the example we are working on, having an E with four elementary events and having taken a P (E) of dimension four, that is, with all the subsets of E formed by four elementary events, the maximum number of steps and therefore, the number of nodes is 3, so step k and $n-1$ are going to be seen together at this point, but it does not vary at all what you would do if they were different steps, the only thing that would happen is that it would be done more times. Once the elementary event Laboratory has been analysed, we have seen that it does not allow us to fully classify the events in the sample and, consequently, any event that we analyse later, so we go to the last possible node, which will be the Practices node. In addition, that it will have to be a final node. We analyse the events in the sample that still remain unclassified, which are 1, 2, 6, and 7. We analyse the value of Practices and Global Qualification: 1. B, Ap; 2. D, Ss; 3. Classified in Step A. 4. Classified in Step A. 5. Classified in Step 2. 6. C, Ap; 7. A, Ap. 8 Classified in Step 2.

If you have a laboratory grade of A or B, you always have a final grade of Ap, whereas if you have a grade of C or D, you always have a rating of Ss. Therefore, Practices can be considered as a final node because it allows the classification of events. Consequently, the classification, after passing through the terminal node Practices, would be (Fig. 3):

Optimizing the Construction of a Decision Tree: ID3 Algorithm

Once you have seen how a decision tree is defined or built, it is immediate to realize that for the same analysis, a set of classification functions based on decision trees can

be defined whose number is equal to all permutations or variations without repetition[8] of the n elementary events taken from n to n, which compose the events of the subset of P (E) of dimension n analysed. And it is logical to think that some of those functions are more efficient than others, as it is. Consequently, the immediate question is if whether there is a way to obtain a classification model based on optimal decision trees. To determine the best way to sequentially decompose the sample to build the tree decision, and consequently what will be the best structure of the tree, each algorithm uses a solution.

Decision tree. Optimization of the definition of the classification function. In the example we are seeing, we have defined a classification function through a classification tree with three nodes, but, as mentioned above, no criteria were used in ordering which elementary events should be analysed before, so the possible permutations, or variations without repetition of three elements taken three at a time, is $3! = 3.2.1 = 6$, that is, we have defined a tree of the six possible ones, that is, the nodes have been Theory-Laboratory-Practices, but they could have been Laboratory-Theory-Practices or any other of the possible permutations.

In algorithm ID3, the optimization in the construction of the decision tree, that is, the determination of which is the best way to sequentially decompose the sample into subsets that will allow the construction of the tree, and consequently which will be the best structure of the same, is done by obtaining the magnitude called *Information Gain* of each division, which is denoted as Δ_I. The information gain is obtained by measuring the impurity difference between the parent node and the child nodes resulting from the division performed. The greater the gain of information, the better the division made and, consequently, the better the structure of the tree. Information Gain is not the way to construct the best decision tree, which is used not only by the ID3 algorithm but also by others, as we will see later. The Δ_I is calculated as:

Information Gain

$$\Delta_I = I_{\text{father}} - \sum_{j=1}^{k} \frac{N(n_j)}{N} I(n_j)$$

where I_{father} is the impurity of the father node, $I(n_j)$ is the impurity of node son j, $N(n_j)$ is the number of events associated with node son j, and N is the total number of events associated with node father.

The number of events associated with the child nodes and the total number of events are both direct measures that can be obtained from observing the division performed, but impurity is an indirect measure for which a function must be defined of measurement, and in that sense, to measure the Impurity of a node and to do that

[8] The equation to calculate the number of permutations or variations without repetition of n elements taken from n in n is pn = vn, n = n!

measurements, a set of different measures, based on the absolute frequencies of the existing classes in the node, has been defined. They will be introduced in the following.

Entropy

The ID3 algorithm use the *Entropy* that, for each node *k* with c classes, is calculated through the equation:[9]

$$\text{Ent}(\text{node}) = -\sum_{i=0}^{c} f_{i\ \text{node}\ k}\, log_{\,2}(f_{i\ \text{node}\ k})$$

where $f_{i\ \text{node}\ k}$ is the relative frequency of class *i* in node k[10] and *c* is the number of classes.

Decision tree. Optimization of the definition of the classification function with the ID3 Algorithm: Calculation of the Impurity nodes to obtain the information gain. Once we have seen how the ID3 algorithm works, what we are going to see is how to build the decision tree of the example we are working on, not arbitrarily choosing the elementary events that will constitute each node as in the previous case but choosing in each case the elementary event that provides the highest gain of information. Since we are working with the ID3 algorithm, we calculate the impurity of each node using the Entropy measurement.

We start with the initial node and calculate the impurity of the node if we take Theory as an elementary event, as seen in the solution of the previous example. If we take Theory in the first node, we obtain a division of the events such that in the child nodes, we get the following: at node 1, N1, 3 events for each class, 1, 6, and 7 for the passed class and 2, 5, and 8 for the failed class. And at Node 2, N2, we have 0 events for the passed class and 2 events for the failed class, 3 and 4.

Consequently, when theory is taken at the initial node, the tree is (Fig. 4):

Therefore, the Entropy of Theory is as follows:

For the parent node, before establishing the division, there are 3 events belonging to class Ap, which will be class 0; therefore, the relative frequency of class 0 is 3/8, which is the number of approved divided by the total number of qualifications available, and since we are in node 1 it will be *f*01; and 5 events belonging to the class Ss, and consequently, therefore, the entropy of the parent node is $f11 = 5/8$

[9] It is interesting to remember here, because it will be necessary to calculate entropy, how to calculate the logarithm of a number in any base. The calculation equation is:

$$\text{Log}_a b = \frac{\text{Log}_x b}{\text{Log}_x a}$$

where *x* is whatever base we want, for example Ln.

[10] The first class is denoted with a 0, the second with a 1, etc.

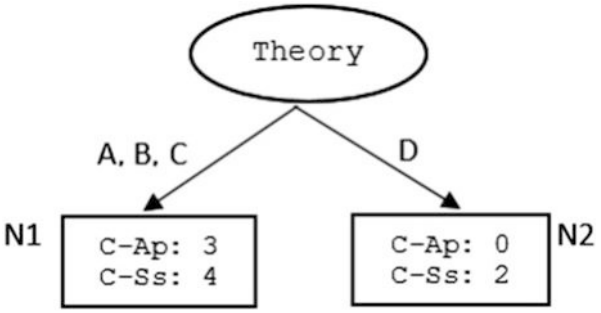


Fig. 4 First division with theory

$$\text{Ent}(f) = -\sum_{i=0}^{1} f_{ip} \log_2 f_{ip} = -f_{01} \log_2 f_{01} - f_{02} \log_2 f_{02} =$$
$$= -\left(\frac{3}{8}\right) \log_2 \left(\frac{3}{8}\right) - \left(\frac{5}{8}\right) log_2 \left(\frac{5}{8}\right) =$$
$$-0.375(-1.415) - (0.625(-0.678)) = 0.531 + 0.424 = 0,955$$

This calculation will be valid and equal for the analysis of the other two elementary events.

And for the child nodes obtained when the Theory event is used to partition the sample, the entropy is as follows:

For Node 1, N1, for class 0, passed, 3/6 is the relative frequency, 3 passed in the 6 grades in the node, and for class 1, fail, the relative frequency is 4/7. Consequently, the entropy of node 1 is:

$$\text{Ent}(1) = -\sum_{i=0}^{1} f_{ip} \log_2 f_{i1} = -\left(\frac{3}{6}\right) \log_2 \left(\frac{3}{6}\right) - \left(\frac{3}{6}\right) log_2 \left(\frac{3}{6}\right)$$
$$= -0.5(-1) - 0.5(-1) = 1$$

For Node 2, N2, for class 0, passed, 0/2 is the relative frequency, 0 passed for the 2 grades in the node, and for class 1, failed, the relative frequency is 2/2. Consequently, the entropy of node 2 is:[11]

$$\text{Ent}(2) = -\sum_{i=0}^{1} f_{ip} \log_2 f_{i1} = -\left(\frac{0}{2}\right) log_2 \left(\frac{0}{2}\right) - \left(\frac{2}{2}\right) \log_2 \left(\frac{2}{2}\right)$$
$$= -0(\log_2(0)) - 0) = 0$$

Once the impurities of the parent and child nodes have been obtained, as seen above, the information gain is equal to the impurity of the parent node minus the

[11] Whenever a complete classification of events occurs in that node in a node, that is, there are no events in one class and all are classified in the other, its entropy will be 0.

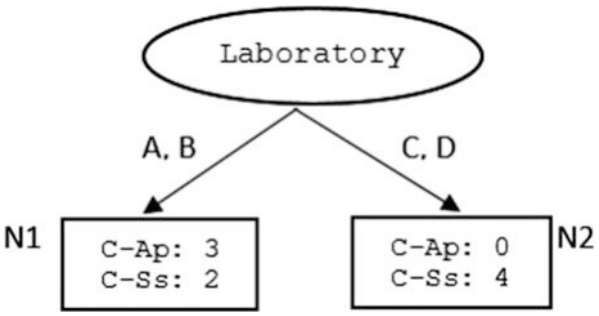


Fig. 5 First division with laboratory

weighted average of the impurities of the child nodes. We first calculate the weighted average of the impurities of the child nodes: As we have seen above, $N(n_1)$ is the number of events associated with child node 1, which in this case is 7, and $N(n_2)$ is the number of events associated with child node 2, which in this case is 2, and N is the total number of events in the parent node, which in this case is 8, so the weighted mean of impurity of the child nodes is:

$$\sum_{j=1}^{k} \frac{N(n_j)}{N} I(n_j) = \frac{N(n_1)}{N}.1 + \frac{N(n_2)}{N}.0 =$$

$$= \frac{6}{8}.1 + \frac{2}{8}.0 = 0.75$$

Since the impurity of the parent node is 0.955, the information gain performing the first division with the elementary event Theory is:

$$\Delta_I = I_{\text{father}} - \sum\nolimits_{j=1}^{k} \frac{N(n_j)}{N} I(n_j) = 0,955 - 0.75 = 0.205$$

We then perform the same calculations for the elemental event Laboratory:

When Laboratory is taken at the initial node, the tree is (Fig. 5):

For Node 1, N1, for class 0, passed, 3/5 is the relative frequency, 3 passed in the 5 grades in the node, and for class 1, failed, the relative frequency is 2/5. Consequently, the entropy of node 1 is:

$$\text{Ent}(1) = -\sum_{i=0}^{1} f_{ip} \log_2 f_{i1} = -\left(\frac{3}{5}\right) \log_2 \left(\frac{3}{5}\right) - \left(\frac{2}{5}\right) \log_2 \left(\frac{2}{5}\right) =$$
$$= -0.6(-0.73) - 0.4(-1.32) = 0.96$$

For Node 2, N2, for class 0, passed, 0/2 is the relative frequency, 0 passed for the 2 grades in the node, and for class 1, failed, the relative frequency is 4/4. Consequently, the entropy of node 2 is:

$$\begin{aligned}\text{Ent}(2) &= -\sum_{i=0}^{1} f_{ip}\log_2 f_{i1} = -\left(\frac{0}{2}\right)\log_2\left(\frac{0}{2}\right) - \left(\frac{4}{4}\right)\log_2\left(\frac{4}{4}\right) \\ &= -0(\log_2(0)) - 1(0) = 0\end{aligned}$$

Once the impurities of the parent and child nodes have been obtained, as seen above, the information gain is equal to the impurity of the parent node minus the weighted average of the impurities of the child nodes. We first calculate the weighted average of the impurities of the child nodes: As we have seen above, $N(n_1)$ is the number of events associated with child node 1, which in this case is 5, $N(n_2)$ is the number of events associated with child node 2, which in this case is 4, and N is the total number of events in the parent node, which in this case is 8, so the weighted mean of impurity of the child nodes is:

$$\begin{aligned}\sum_{j=1}^{k}\frac{N(n_j)}{N}I(n_j) &= \frac{N(n_1)}{N}.1 + \frac{N(n_2)}{N}.0 = \\ &= \frac{5}{8}.0.96 + \frac{4}{8}.0 = 0.6\end{aligned}$$

And since the impurity of the parent node is 0.955, the information gain performing the first division with the elementary event Theory is:

$$\Delta_I = I_{\text{father}} - \sum_{j=1}^{k}\frac{N(n_j)}{N}I(n_j) = 0,955 - 0.6 = 0.355$$

We then perform the same calculations for the last candidate elementary event, Practices:

When practices at the initial node, the tree is (Fig. 6):

We know that the entropy of the parent node, as in the two previous cases, is 0.955

And for the child nodes obtained when the Practices event is used to partition the sample, the entropy is:

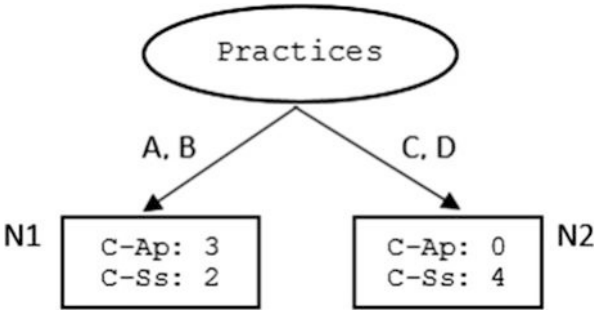


Fig. 6 First division with practices

For Node 1, N1, for class 0, passed, 3/5 is the relative frequency, 3 passed in the 5 grades in the node, and for class 1, failed, the relative frequency is 2/5. Consequently, the entropy of node 1 is:

$$\begin{aligned}\text{Ent}(1) &= -\sum_{i=0}^{1} f_{ip}\log_2 f_{i1} = -\left(\frac{3}{5}\right)\log_2\left(\frac{3}{5}\right) - \left(\frac{2}{5}\right)\log_2\left(\frac{2}{5}\right) = \\ &= -0.6(-0.73) - 0.4(-1.32) = 0.96\end{aligned}.$$

For Node 2, N2, for class 0, passed, 0/2 is the relative frequency, 0 passed for the 2 grades in the node, and for class 1, failed, the relative frequency is 4/4. Consequently, the entropy of node 2 is:

$$\begin{aligned}\text{Ent}(2) &= -\sum_{i=0}^{1} f_{ip}\log_2 f_{i1} = -\left(\frac{0}{2}\right)\log_2\left(\frac{0}{2}\right) - \left(\frac{4}{4}\right)\log_2\left(\frac{4}{4}\right) \\ &= -0(\log_2(0)) - 0) = 0\end{aligned}$$

Once the impurities of the parent and child nodes have been obtained, as seen above, the information gain is equal to the impurity of the parent node minus the weighted average of the impurities of the child nodes. We first calculate the weighted average of the impurities of the child nodes: As we have seen above, $N(n_1)$ is the number of events associated with child node 1, which in this case is 5, $N(n_2)$ is the number of events associated with child node 2, which in this case is 4, and N is the total number of events in the parent node, which in this case is 8, so the weighted mean of impurity of the child nodes is:

$$\begin{aligned}\sum_{j=1}^{k}\frac{N(n_j)}{N}I(n_j) &= \frac{N(n_1)}{N}.1 + \frac{N(n_2)}{N}.0 = \\ &= \frac{5}{8}.0.96 + \frac{4}{8}.0 = 0.6\end{aligned}$$

And since the impurity of the parent node is 0.955, the information gain performing the first division with the elementary event Practices is:

$$\Delta_I = I_{\text{father}} - \sum\nolimits_{j=1}^{k}\frac{N(n_j)}{N}I(n_j) = 0,955 - 0.6 = 0.355$$

From the Entropy calculations for Theory, Laboratory, and Practices, 0.205, 0.355, and 0.355, respectively, we see that the greatest information gain is obtained with Laboratory or Practices, so for the initial node, we select any of the two; we are going to select Laboratory.

Once Laboratory is selected, the first level of the classification tree is (Fig. 7):

And events 3, 4, 5, and 8 are already classified, so we will not use them for the next level analysis, which starts with the following sample:

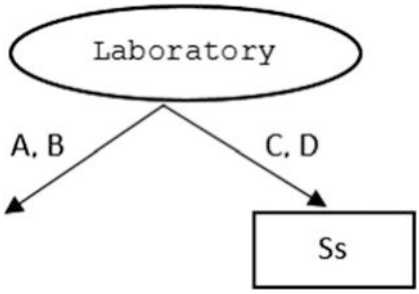


Fig. 7 First level of classification tree

- {A, A, B, Ap}
- {A, B, D, Ss}
- {D, C, C, Ss} Classified
- {D, B, A, Ss} Classified
- {B, C, D, Ss} Classified
- {C, B, B, Ap}
- {B, B, A, Ap}
- {C, D, C, Ss} Classified

Once the initial Laboratory node has been established, since two terminal nodes or leaves cannot be obtained from it, we proceed to analyse the first intermediate node, for which we have Theory and Practices as candidate elementary events.

The parent node is now made up of only five events, since four events, 3, 4, 5, and 8 have already been classified in the terminal node Fail, Ss. Therefore, for the parent node, before establishing the division, there are 3 events belonging to the class Approved, Ap, and 2 belonging to the class Fail, Ss. Taking this into account, we calculate its impurity.

$$\begin{aligned}\text{Ent}(f) &= -\sum_{i=0}^{1} f_{ip} \log_2 f_{ip} = -f_{01} \log_2 f_{01} - f_{02} \log_2 f_{02} = \\ &= -\left(\frac{3}{5}\right) \log_2 \left(\frac{3}{5}\right) - \left(\frac{2}{5}\right) \log_2 \left(\frac{2}{5}\right) = \\ &\quad -0.6(-0.73) - (0.4(-1.32)) = 0,97\end{aligned}$$

We begin by analysing Practices and we obtain a division of the events such that in the child nodes, we have, in node 1, 3 events, 1, 6, and 7, all in the approved class, grades A and B. In Node 2, we have 0 events for the approved class and 1 event, 2, for the failed class, with grade C. The tree remains (Fig. 8):

And for the child nodes obtained when the Practices event is used to partition the sample, the entropy is:

For Node 1, N1, for class 0, passed, 3/3 is the relative frequency, 3 passed in the 5 grades in the node, and for class 1, failed, the relative frequency is 0/3. Consequently, the entropy of node 1 is:

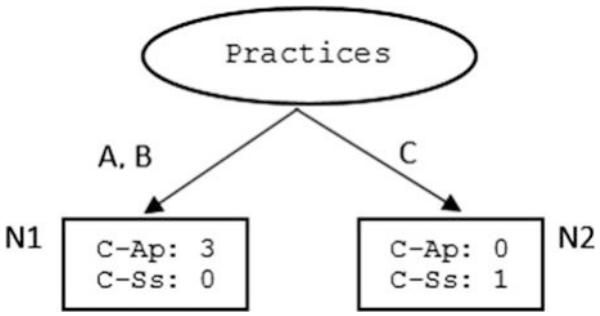


Fig. 8 Practices in the second level

$$\text{Ent}(1) = -\sum_{i=0}^{1} f_{ip} \log_2 f_{i1} = -\left(\frac{3}{3}\right) \log_2 \left(\frac{3}{3}\right) - \left(\frac{0}{3}\right) \log_2 \left(\frac{0}{3}\right) = 0$$

For Node 2, N2, for class 0, passed, 0/2 is the relative frequency, 0 passed for the 2 grades in the node, and for class 1, failed, the relative frequency is 4/4. Consequently, the entropy of node 2 is:

$$\text{Ent}(2) = -\sum_{i=0}^{1} f_{ip} \log_2 f_{i1} = -\left(\frac{0}{2}\right) \log_2 \left(\frac{0}{2}\right) - \left(\frac{4}{4}\right) \log_2 \left(\frac{4}{4}\right) = 0$$

Once the impurities of the parent and child nodes have been obtained, as seen above, the information gain is equal to the impurity of the parent node minus the weighted average of the impurities of the child nodes. We first calculate the weighted average of the impurities of the child nodes: As we have seen above, $N(n_1)$ is the number of events associated with child node 3, which in this case is 5, $N(n_2)$ is the number of events associated with child node 2, which in this case is 2, and N is the total number of events in the parent node, which in this case is 5, so the weighted mean of impurity of the child nodes is:

$$\sum_{j=1}^{k} \frac{N(n_j)}{N} I(n_j) = \frac{N(n_1)}{N}.1 + \frac{N(n_2)}{N}.0 =$$
$$= \frac{3}{5}.0 + \frac{2}{5}.0 = 0$$

And since the impurity of the parent node is 0.97, the information gain performing the first division with the elementary event Theory is:

$$\Delta_I = I_{\text{father}} - \sum_{j=1}^{k} \frac{N(n_j)}{N} I(n_j) = 0,97 - 0 = 0.97$$

From the calculations of the Entropy for Practices, we have seen that the impurity of the children does not subtract anything from that of the father, so if we calculate the information gain with the classification made by Theory it could only be the

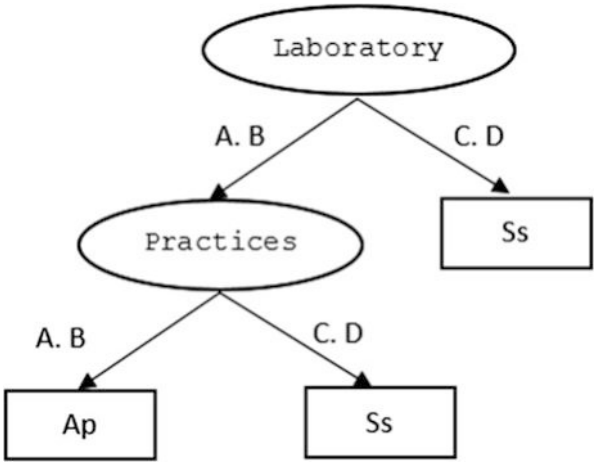


Fig. 9 Final classification

same. In consequence, with Practices we already have a complete classification of the events, so it would not be necessary to calculate the information gain by Theory and the classifier model will be (Fig. 9):

Although the classification model is already finished, we are going to analyse what would have happened if we had analysed Theory at this second level. If we only analyse the Theory rating and the Global rating, we have:

- {A, Ap}
- {A, Ss}
- Classified
- Classified
- Classified
- {C, Ap}
- {B, Ap}
- Classified

And we realize that if we take A or B, we have Approved and Failed, so we will not be able to have the two nodes with only one class because even if in one node we put only C that only has approved, in the other there would be A and B. Therefore, since there are not two nodes with zero impurity, the information gain will be less than for practices. Furthermore, if we used Theory, we would not have considered the D rating.

Optimizing the Construction of a Decision Tree: CART Algorithm

Once we have seen how the ID3 algorithm works, which is based on the information gain obtained through the impurity measurement that entropy gives us, we will see the *CART algorithm*, which uses the impurity measurement measure called the *Gini*

Gini

CART algorithm uses the *Gini*, which, for each node k with c classes, is calculated through the equation:

$$\text{Gin}(\text{node}) = 1 - \sum_{i=0}^{c-1} (f_{i\ \text{node}})^2$$

where $f_{i\ \text{node}\ k}$ is the relative frequency of class i in node k[12] and c is the number of classes.

And we remember how Gain of Information is measured.

$$\Delta_I = I_{\text{father}} - \sum_{j=1}^{k} \frac{N(n_j)}{N} I(n_j)$$

where I_{father} is the impurity of the father node, $I(n_j)$ is the impurity of node son j, $N(n_j)$ is the number of events associated with node son j, and N is the total number of events associated with node father.

Decision tree. Optimization of the definition of the classification function with the CART Algorithm: Calculation of the Impurity nodes to obtain the information Gain. Once we have seen how the CART algorithm works, what we are going to see now is how to build the decision tree of the example we are working on, not arbitrarily choosing the elementary events that will constitute each node as in the previous case but choosing in each case the elementary event that provides the highest gain of information. Since we are working with the CART algorithm, we calculate the impurity of each node using the Gini measurement.

We start with the initial node and calculate the impurity of the node if we take Theory as an elementary event, as seen in the solution of the previous example.[13] If we take Theory in the first node, we know that we obtain a division of the events such that in the child nodes, we have the following: in node 1, N1, 3 events for the approved class, 1, 6, and 7; and three events, 2, 5, and 8 for the fail class. And at Node 2, N2, we have 0 events for the passed class and 2 events for the failed class, 3 and 4.

Therefore, the Gini of Theory is: For the parent node, before establishing the division, there are 3 events belonging to class Ap, which will be class 0; therefore, the relative frequency of class 0 is 3/8, which is the number of approved divided by the total number of qualifications available, and since we are in node 1, it will be f_{01}; and 5 events belonging to the class Ss, $f_{11} = 5/8$; consequently, the entropy of the parent node is:

[12] The first class is denoted with a 0, the second with a 1, etc.

[13] The images are the same as in the ID3 example, we are not going to repeat it here so as not to be reiterative.

$$\text{Gin}(f) = 1 - \sum_{i=0}^{c-1} {f_{ip}}^2 = 1 - \left(\left(\tfrac{3}{8}\right)^2 + \left(\tfrac{5}{8}\right)^2\right) = \\ = 1 - (0.14 + 0.39) = 0,47$$

This calculation will be valid and the same for the analysis of the other two elementary events.

And for the child nodes obtained when the Theory event is used to partition the sample, the Gini is:

For Node 1, N1, for class 0, passed, 3/6 is the relative frequency, 3 passed in the 6 grades in the node, and for class 1, failed, the relative frequency is 3/6. Consequently, the entropy of node 1 is:

$$\text{Gin}(1) = 1 - \sum_{i=0}^{c-1} {f_{i1}}^2 = 1 - \left(\left(\tfrac{3}{6}\right)^2 + \left(\tfrac{3}{6}\right)^2\right) = \\ = 1 - (0.25 + 0.25) = 0,5$$

For Node 2, N2, for class 0, passed, 0/2 is the relative frequency, 0 passed for the 2 grades in the node, and for class 1, failed, the relative frequency is 2/2. Consequently, the entropy of node 2 is:[14]

$$\text{Gin}(2) = 1 - \sum_{i=0}^{c-1} {f_{i1}}^2 = 1 - \left(\left(\tfrac{0}{2}\right)^2 + \left(\tfrac{2}{2}\right)^2\right) = \\ = 1 - (0 + 1) = 0$$

Consequently, once the impurities of the parent and child nodes have been obtained, as seen above, the information gain is equal to the impurity of the parent node minus the weighted average of the impurities of the child nodes. We first calculate the weighted average of the impurities of the child nodes: As we have seen above, it is the number of events associated with child node 1, which in this case is 6, the number of events associated with child node 2, which in this case are 2, and N is the total number of events in the parent node, which in this case is 8, so the weighted mean of impurity of the child nodes is:

$$\sum_{j=1}^{k} \frac{N(n_j)}{N} I(n_j) = \frac{6}{8}.0.5 + \frac{2}{8}.0 = 0.375$$

And since the impurity of the parent node is 0.47, the information gain performing the first division with the elementary event Theory is:

[14] As we know whenever a complete classification of events occurs in that node in a node, that is to say that in one class there are no events and all are classified in the other, as happened with entropy, its Gini will be 0.

$$\Delta_I = I_{\text{father}} - \sum_{j=1}^{k} \frac{N(n_j)}{N} I(n_j) = 0,47 - 0.375 = 0.095$$

Once the information gain obtained by choosing Theory as the elementary event for the first node has been calculated, the calculations are made to see if any other elementary event provides more information gain, in which case it would be selected instead of Theory for the initial node. To be able to compare it, you have to choose the same unit of measurement for impurity. As it would be very long to do it with the three units of measurement, and as each of the three works has already been explained, for this first node we are going to do the calculations with the Gini index, that is, you must use in the same node the same algorithm.

We now select Laboratory[15] as the initial event and obtain a division of the events such that in the children nodes we have, in node 1, 3 events for the Ap class, 1, 6, and 7 for the approved class; and 2 events 2 and 4 for the class Fail. And in Node 2, we have 0 events for the passed class and 3 events for the failed class, 3 and 5.

If we calculate the Gini when the Laboratory event is used to classify the sample, we obtain the following:

The parent node calculation is the same as for Theory, since nothing has changed, since as we said above, this calculation will be valid and the same for the analysis of the other two elementary events, Laboratory and Practices. Therefore, the parent node has 3 events belonging to class Ap and 5 belonging to class Ss; therefore, the Gini of the parent node is 0.47.

And for the child nodes, it is as follows:

For Node 1, N1, for class 0, passed, 3/5 is the relative frequency, 3 passed in the 5 grades in the node, and for class 1, failed, the relative frequency is 2/5. Consequently, the Gini of node 1 is:

$$\begin{aligned} \text{Gin}(1) &= 1 - \sum_{i=0}^{c-1} {f_{i1}}^2 = 1 - \left(\left(\tfrac{3}{5}\right)^2 + \left(\tfrac{2}{5}\right)^2 \right) = \\ &= 1 - (0.36 + 0.16) = 0,48 \end{aligned}$$

For Node 2, N2, for class 0, passed, 0/2 is the relative frequency, 0 passed in the 2 grades in the node, and for class 1, fail, the relative frequency is 3/3. Consequently, the Gini of node 2 is:

$$\begin{aligned} \text{Gin}(2) &= 1 - \sum_{i=0}^{c-1} {f_{i1}}^2 = 1 - \left(\left(\tfrac{0}{2}\right)^2 + \left(\tfrac{3}{3}\right)^2 \right) = \\ &= 1 - (0 + 1) = 0 \end{aligned}$$

[15] We repeat the sample here to be able to see the values more clearly: 1. {A, A, B, Ap}; 2. {A, B, D, Ss}; 3. {D, C, C, Ss}; 4. {D, B, A, Ss}; 5. {B, C, D, Ss}; 6. {C, B, B, Ap}; 7. {B, B, A, Ap}; 8. {C, D, C, Ss}.

Consequently, once the impurities of the parent and child nodes have been obtained, as seen above, the information gain is equal to the impurity of the parent node minus the weighted average of the impurities of the child nodes. We first calculate the weighted average of the impurities of the child nodes: As we have seen above, it is the number of events associated with child node 1, which in this case is 5, and it is the number of events associated with child node 2, which in this case is 3, and N is the total number of events in the parent node, which in this case is 8, so the weighted mean of impurity of the child nodes is:

$$\sum_{j=1}^{k} \frac{N(n_j)}{N} I(n_j) = \frac{5}{8}.0.48 + \frac{3}{8}.0 = 0.3$$

And since the impurity of the parent node is 0.47, the information gain performing the first division with the elementary event Laboratory is:

$$\Delta_I = I_{\text{father}} - \sum_{j=1}^{k} \frac{N(n_j)}{N} I(n_j) = 0,47 - 0.3 = 0.17$$

From this result, it can be concluded that the information gain is greater than Theory if the elementary event Laboratory is used, since the laboratory event is 0.17 and the theory event is 0.095, which is why Laboratory would be chosen. We are now going to analyse Practices.

If we take Practices,[16] we obtain a division of the events such that in the children nodes we have, in node 1, 3 events for the class Ap, 1, 6, and 7 for the approved class; and 1 event, 4, for the fail class. In node 2, we have 0 events for the passed class and 4 events for the failed class, 2, 4, 5, and 8.

As mentioned above, the impurity measure to be used will be the Gini index. The Laboratory Gini is as follows:

The parent node calculation is the same as for Theory and Laboratory, but we are going to remember it here; therefore, the parent node has 3 events belonging to class Ap and 5 belonging to class Ss, and consequently, the Gini of the parent node is:

$$\begin{aligned} \text{Gin}(f) &= 1 - \sum_{i=0}^{c-1} {f_{ip}}^2 = 1 - \left(\left(\tfrac{3}{8}\right)^2 + \left(\tfrac{5}{8}\right)^2 \right) = \\ &= 1 - (0.14 + 0.39) = 0,47 \end{aligned}$$

And for the child nodes the Gini is as follows:

For Node 1, N1, for class 0, passed, 3/4 is the relative frequency, 3 passed in the 4 grades in the node, and for class 1, failed, the relative frequency is 1/4. Consequently, the Gini of node 1 is as follows:

[16] We repeat the sample here to be able to see the values more clearly: 1. {A, A, B, Ap}; 2. {A, B, D, Ss}; 3. {D, C, C, Ss}; 4. {D, B, A, Ss}; 5. {B, C, D, Ss}; 6. {C, B, B, Ap}; 7. {B, B, A, Ap}; 8. {C, D, C, Ss}.

$$\text{Gin}(1) = 1 - \sum_{i=0}^{c-1} {f_{i1}}^2 = 1 - \left(\left(\tfrac{3}{4}\right)^2 + \left(\tfrac{1}{4}\right)^2\right) = \\ = 1 - (0.5625 + 0.0625) = 0,625$$

For Node 2, N2, for class 0, passed, 0/4 is the relative frequency, 0 passed in the 4 grades in the node, and for class 1, failed, the relative frequency is 4/4. Consequently, the Gini of node 2 are:

$$\text{Gin}(2) = 1 - \sum_{i=0}^{c-1} {f_{i1}}^2 = 1 - \left(\left(\tfrac{0}{4}\right)^2 + \left(\tfrac{4}{4}\right)^2\right) = \\ = 1 - (0 + 1) = 0$$

Consequently, once the impurities of the parent and child nodes have been obtained, as seen above, the information gain is equal to the impurity of the parent node minus the weighted average of the impurities of the child nodes. We first calculate the weighted average of the impurities of the child nodes: As we have seen above, it is the number of events associated with child node 1, which in this case is 4, and it is the number of events associated with child node 2, which in this case is 4, and N is the total number of events in the parent node, which in this case is 8, so the weighted mean of impurity of the child nodes is:

$$\sum_{j=1}^{k} \frac{N(n_j)}{N} I(n_j) = \frac{4}{8}.0.625 + \frac{4}{8}.0 = 0.3125$$

And since the impurity of the parent node is 0.47, the information gain performing the first division with the elementary event theory is:

$$\Delta_I = I_{\text{father}} - \sum_{j=1}^{k} \frac{N(n_j)}{N} I(n_j) = 0,47 - 0.3125 = 0.1575$$

From this result of the Gini calculations for Theory, Laboratory, and Practices, 0.095, 0.17, and 0.1575, respectively, it can be concluded that the information gain is greater than that of Theory if the elementary event Practices is used in the initial node, but it is lower than the use of Laboratory, so Laboratory would be chosen.

And events 3, 4, 5, and 8 are already classified, so we will not use them for the next level analysis.

{A, A, B, Ap}
{A, B, D, Ss}
{D, C, C, Ss} Classified
{D, D, A, Ss} Classified
{B, C, B, Ss} Classified
{C, B, B, Ap}
{B, B, A, Ap}
{C, D, C, Ss} Classified

Once the initial Laboratory node has been established, since two terminal nodes or leaves cannot be obtained from it, we proceed to analyse the first intermediate node, for which we have Theory and Practices as candidate elementary events.

The parent node is now made up of only five events, since four, events 3, 4, 5, and 8 have already been classified in the terminal node Suspense, Ss. Therefore, for the parent node, before establishing the division, there are 3 events belonging to the class Approved, Ap, and 2 belonging to the class Suspense, Ss. Taking this into account, we calculate its impurity with the Gini.

$$\begin{aligned} \text{Gin}(f) &= 1 - \sum_{i=0}^{1} {f_{ip}}^2 = 1 - \left(\left(\tfrac{3}{5}\right)^2 + \left(\tfrac{2}{5}\right)^2 \right) = \\ &= 1 - (0.36 + 0.16) = 0.48 \end{aligned}$$

We begin by analysing Practices and we obtain a division of the events such that in the child nodes we have, in node 1, 3 events, 1, 6, and 7, all in the approved class, grades A and B. And in Node 2, we have 0 events for the passed class and 1 event, 2, for the failed class, with grade C.

And for the child nodes obtained when the Practices event is used to partition the sample, the Gini is as follows:

For Node 1, N1, for class 0, passed, 3/3 is the relative frequency, 3 passed in the 5 grades in the node, and for class 1, failed, the relative frequency is 0/3. Consequently, the Gini of node 1 is:

$$\begin{aligned} \text{Gin}(1) &= 1 - \sum_{i=0}^{c-1} {f_{i1}}^2 = 1 - \left(\left(\tfrac{3}{3}\right)^2 + \left(\tfrac{0}{3}\right)^2 \right) = \\ &= 1 - (1 + 0) = 0 \end{aligned}$$

For Node 2, N2, for class 0, passed, 0/2 is the relative frequency, 0 passed in 4 grades in the node, and for class 1, failed, the relative frequency is 2/2. Consequently, the Gini of node 2 is:

$$\begin{aligned} \text{Gin}(2) &= 1 - \sum_{i=0}^{c-1} {f_{i1}}^2 = 1 - \left(\left(\tfrac{0}{2}\right)^2 + \left(\tfrac{2}{2}\right)^2 \right) = \\ &= 1 - (0 + 1) = 0 \end{aligned}$$

Consequently, once the impurities of the parent and child nodes have been obtained, as seen above, the information gain is equal to the impurity of the parent node minus the weighted average of the impurities of the child nodes. We first calculate the weighted average of the impurities of the child nodes: As we have seen above, it is the number of events associated with child node 1, which in this case is 3, and it is the number of events associated with child node 2, which in this case is 2, and N is the total number of events in the parent node, which in this case is 8, so the weighted mean of impurity of the child nodes is:

$$\sum_{j=1}^{k} \frac{N(n_j)}{N} I(n_j) = \frac{3}{8}.0 + \frac{2}{8}.0 = 0$$

And since the impurity of the parent node is 0.48, the information gain performing the first division with the elementary event Theory is:

$$\Delta_I = I_{\text{father}} - \sum_{j=1}^{k} \frac{N(n_j)}{N} I(n_j) = 0,48 - 0 = 0.48$$

From the Gini calculations for Practices, we have seen that the impurity of the children does not subtract anything from that of the father, so if we calculate the information gain with the classification made by Theory, it could only be the same. In consequence, with Practices we already have a complete classification of the events so it would not be necessary to calculate the information gain by Theory and the classifier model will be (Fig. 10):

We realize that, as is logical, the classifier model that we obtain using the CART algorithm and the Gini impurity measure is the same as we obtained using the ID3 algorithm and the Entropy impurity measure.

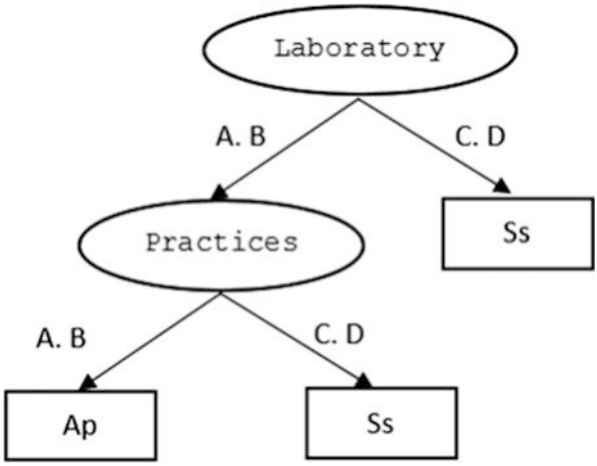
Optimizing the Construction of a Decision Tree: Error Algorithm

Error is defined as:

$$\text{Err}(\text{node}) = 1 - \acute{\text{m}}\text{ax}_i(f_{i\ \text{node}})$$

We have carried out all the calculations to obtain the information gain of the first node using the elementary event Theory to the end to give more clarity and consistency to the example, but as said above, the impurities and the information

Fig. 10 Final classification



gain were to be calculated using the three steps and you will not stop doing it. We now calculate the impurities using the Error.

When the Theory event is used to partition the sample, the impurity measure Error is:

We know that the parent node has 3 events belonging to class Ap and 5 belonging to class Ss; therefore, the error of the parent node is:

$$\text{Err}(f) = 1 - \max_i(f_{ip}) = 1 - \max_i\left(\frac{3}{8}, \frac{5}{8}\right) = 1 - 0,625 = 0,375$$

And for the child nodes, the error is:

$$\text{Err}(1) = 1 - \max_i(f_{i1}) = 1 - \max(f_{01}, f_{11}) = 1 - \max\left(\frac{3}{6}, \frac{3}{6}\right) = \\ = 1 - 0.5 = 0.5$$

$$\text{Err}(2) = 1 - \max_i(f_{i1}) = 1 - \max(f_{02}, f_{12}) = 1 - \max\left(\frac{0}{2}, \frac{2}{2}\right) \\ = 1 - 1 = 0$$

Consequently, the weighted mean impurity of the child nodes is:

$$\sum_{j=1}^{k} \frac{N(n_j)}{N} I(n_j) = \frac{6}{8}.0.5 + \frac{2}{8}.0 = 0.375$$

And since the impurity of the parent node is 0.375, the information gain performing the first division with the elementary event Theory is:

$$\Delta_I = I_{\text{father}} - \sum_{j=1}^{k} \frac{N(n_j)}{N} I(n_j) = 0,375 - 0.375 = 0$$

Once the initial node has been established, and since two terminal nodes or leaves cannot be obtained from it, we proceed to analyse the first intermediate node, for which we have theory and practices as candidate elementary events.

The parent node is now made up of only five events, since three, events 3, 5, and 8, have already been classified at the terminal node Ss. Therefore, for the parent node, before establishing the division, there are 3 events belonging to class Ap and 2 belonging to class Ss. For this second node, we are going to make the calculations with the unit of measurement Error (this calculation will be valid and the same for the analysis of the other elementary event).

If we take Theory, we obtain a division of the events such that in the child nodes we have, in node 1, 4 events: 3 events, 1, 6, and 7, for the approved class, and 1 event, 2, for the failed class. And in Node 2, we have 0 events for the passed class and 1 event for the failed class, elementary event 2.

Consequently, the error when using elementary event Theory to partition the sample in the analysed parent node, which is the second node or first intermediate node, is:

$$\text{Err}(f) = 1 - \max_i(f_{ip}) = 1 - \max_i\left(\frac{3}{5}, \frac{2}{5}\right) = 1 - 0,6 = 0,4$$

And for the child nodes obtained, the error is:

$$\text{Err}(1) = 1 - \max_i(f_{i1}) = 1 - \max(f_{01}, f_{11}) = 1 - \max_i\left(\frac{3}{4}, \frac{1}{4}\right) = \\ = 1 - 0,75 = 0,25$$

$$\text{Err}(2) = 1 - \max_i(f_{i2}) = 1 - \max(f_{01}, f_{11}) = 1 - \max_i\left(\frac{0}{1}, \frac{1}{1}\right) = \\ = 1 - 1 = 0$$

Consequently, the weighted mean impurity of the child nodes is:

$$\sum_{j=1}^{k} \frac{N(n_j)}{N} I(n_j) = \frac{4}{5}.0.25 + \frac{1}{5}.0 = 0.2$$

Since the impurity of the parent node is 0.375, the information gain performing the first division with the elementary event Theory is:

$$\Delta_I = I_{\text{father}} - \sum_{j=1}^{k} \frac{N(n_j)}{N} I(n_j) = 0,4 - 0.2 = 0.2$$

Once Theory has been analysed, we will see what information we gain if we choose Practices. If we take Practices, we obtain a division of the events such that in the child nodes we have, in node 1, 3 events, 1, 6, and 7, all in the approved class. In Node 2, we have 0 events for the passed class and 2 events, 2 and 4, for the failed class.

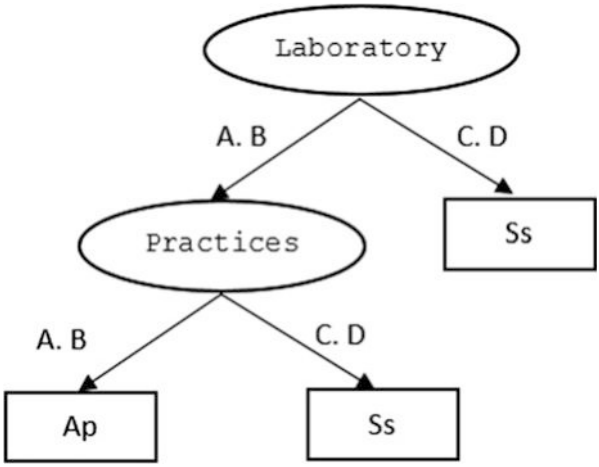
Consequently, the Theory Error in the analysed parent node, which is the second node or first intermediate node, is:

$$\text{Err}(f) = 1 - \max_i(f_{ip}) = 1 - \max_i\left(\frac{3}{5}, \frac{2}{5}\right) = 1 - 0,6 = 0,4$$

And for the child nodes obtained, the error is:

$$\text{Err}(1) = 1 - \max_i(f_{i1}) = 1 - \max(f_{01}, f_{11}) = 1 - \max_i\left(\frac{3}{3}, \frac{0}{3}\right) = = 1 - 1 = 0$$

Fig. 11 Final classification



$$\text{Err}(2) = 1 - \max_i(f_{i2}) = 1 - \max(f_{01}, f_{11}) = 1 - \max_i\left(\frac{0}{2}, \frac{2}{2}\right) = \\ = 1 - 1 = 0$$

Consequently, the weighted mean impurity of the child nodes is:

$$\sum_{j=1}^{k} \frac{N(n_j)}{N} I(n_j) = \frac{3}{5}.0 + \frac{2}{5}.0 = 0$$

And since the impurity of the parent node is 0, the information gain performing the first division with the elementary event Theory is:

$$\Delta_I = I_{\text{father}} - \sum_{j=1}^{k} \frac{N(n_j)}{N} I(n_j) = 0,4 - 0 = 0.4$$

From this result, it can be concluded that the information gain is greater if the elementary event Practices than Theory is used in the intermediate node, but it is also that a complete classification of the events in two leaf or terminal nodes has already been achieved and consequently the final tree looks like (Fig. 11):

Optimizing the Construction of a Decision Tree: Other Approaches

In some cases, the use of the information gain measure is not enough to solve the problem of looking for the best division because, for example, there may be two or more elementary events that provide the same information gain. To solve this, the measure known as the *Gain Ratio* is used, which is the existing proportion between the information gain and the *Division Information*. More formally, it is defined as:

$$\text{Gain Ratio} = \frac{\Delta_I}{\text{Division Information}}$$

And the division information is defined as:

$$\text{Division Information} = -\sum_{i=1}^{k} p(d_i) \log_2 p(d_i)$$

Where k is the number of divisions produced by the elementary event, and $p(d_i)$ is the relative frequency of division i, that is, the number of events that fall in that division divided by the total number of event values.

Decision tree. Profit Ratio. Division Information. We are going to calculate the division information and the profit ratio of the intermediate node Practices of the exercise we are doing.

$$\begin{aligned}\text{Division Information} &= -\sum_{i=1}^{k} p(d_i) \log_2 p(d_i) \\ &= p(d_1) \log_2 p(d_1) + p(d_2) \log_2 p(d_2) \\ &= \frac{3}{5}.\log_2\left(\frac{3}{5}\right) + \frac{2}{5}.\log_2\left(\frac{2}{5}\right) = -0.97\end{aligned}$$

There are only two divisions $k = 2$ and $p(d_1)$ is the relative frequency of the first division, which are the number of events that fall in that division divided by the total number of events.

The profit ratio is:

$$\text{Gain Ratio} = \frac{\Delta_I}{\text{Division Information}} = \frac{0,4}{-0.97} = -0.41$$

Once the classification has been completed from a decision tree (or any other classification technique) to establish the degree of precision with which it classifies a set of events from a random experiment, within the set of equivalence classes previously defined for these events, the calculation of the classical probability of classification of the events is used, or in its correct equivalence class, a measure called accuracy, which is more formally defined as:

$$\forall \{\mathrm{A}_i\}_{i=1}^{\infty} \subset \mathrm{P}\ (\mathrm{E}).f(\mathrm{A}_i) : \mathrm{P}\ (\mathrm{E}) \rightarrow C(\mathrm{P}\ (\mathrm{E}))/\mathrm{e}\ (f) = \frac{n_{\text{correct classifications}}}{n_{\text{classifications}}},$$

C("P (E))" are the equivalence classes defined on P(E).

or in an incorrect equivalence class, a measure called the error ratio, which is more formally defined as:

$$\forall \{A_i\}_{i=1}^{\infty} \subset P(E), f(A_i) : P(E) \rightarrow C(P(E))/re(f) = \frac{n_{\text{incorrect classifications}}}{n_{\text{classifications}}},$$

As seen from its definition, the accuracy and the error ratio are complementary probabilities, e (f) = 1 - re (f). The threshold of acceptance of an accuracy, or error, is not fixed but will be set arbitrarily a priori and will depend on the objectives of the study.

Neural Networks

As we know from what has been studied in Topic 6, every random experiment has an associated sample space E, as the set of all the elementary events of the random experiment, P (E)

From this knowledge, we are going to study the concept of the neural network. Classification studies based on artificial neural networks seek to define a function, called the neural network model, that allows, from the qualitative values of $n-1$ elementary events, of the n elementary events that make up an event to determine the value of the remaining elemental event. Neural networks work with numerical values, so it is necessary to convert qualitative values into numerical values to be able to operate with them.

Artificial neural networks are composed of two or more layers, each of which is composed of one or more nodes, which can be input if values are introduced in the network layer or output if values are obtained. These nodes are also known as neurons or units. Each input node is connected by means of a weighted connection with an output node in such a way that the values of those nodes whose connections have a greater weight will have a greater relevance in the calculation of the final result. Therefore, it is known as training a neural network, which is actually defining or specifying the neural network to optimize it for the resolution of a specific classification problem consisting of determining the weights of the connections between nodes, together with another value called polarization, which will be subtracted from the value resulting from the aggregation of the neuronal connections.

Neural network of events. To introduce the concept of a neural network of events, we will use an example very similar to the one we saw in the supervised classification when we were studying the decision tree technique, although we will introduce some modifications that allow us to see more clearly how a neural network works. The events that we are going to classify are the grades of a subject from a group of students. The qualifications will be made up of three marks, the qualifications of the Theory, Laboratory, and Global Qualification tests. The elementary events are each of the marks individually E = {Theory, Laboratory, Global Score}. All elementary events will have two possible values: Approved, which will become a 1 to be able to apply the network, and Fail, for which we will take the value −1. The classification function sought in this case will be that which, based on the values of the first three

elementary events, allows classifying an event, or what is the same, depending on the qualifications in theory and laboratory, obtaining the overall student grade.

To obtain the neural network that best relates the events, different types of neural networks can be used. This differentiation is based on the fact that each of them has a different structure. All of them use a sample or set of events for which the values of all elementary events are known, including the one to be inferred, to define the neural network. Once the function is defined, it will be used to infer the value of new events.

Sample. The sample of events that will allow finding the definition of the regression function is made up of the following four events: 1. {Ap, Ap, Ap}; 2. {Ap, Ss, Ss}; 3. {Ss, Ap, Ss}; 4. {Ss, Ss, Ss}. Or what, in its transformed values, to be able to work with the network are: 1. {1, 1, 1}; 2. {1, −1, −1}; 3. {−1, 1, −1}; 4. {−1, −1, −1}.

Among the best known and most commonly used types of neural networks are the following:

- Two-layer artificial neural network or perceptron
- Multilayer artificial neural network

We are going to see how each of them works in a specific way.

Two-Layer Artificial Neural Network or Perceptron: Rosenblatt Algorithm

A perceptron is the simplest artificial neural network model. It was developed by Frank Rosenblatt in 1957. The perceptron consists of two layers, an input layer with $n-1$ input nodes, where n is the number of elementary events that are had, and an output layer with a single node. Through the input nodes, the values of the $n-1$ elementary events of which their value is known will be introduced into the network (the elementary event n is the dependent event), and through the output node, the perceptron will provide the value of the unknown elemental event.

Following the generic operation of a neural network, the output node of the perceptron will obtain the output value through a weighted sum of the values obtained from the input nodes, that is, adding the weighted connections between the output node and those connected with it, which in this case are all the inputs. The mathematical equation that defines a perceptron is:

$$y' = \begin{cases} 1, \text{if } \sum_{i=1}^{n-1} w_i x_i \geq \theta \\ -1, \text{if } \sum_{i=1}^{n-1} w_i x_i < \theta \end{cases}$$

or what is the same, if we clear θ we have:

$$y' = \begin{cases} 1, \text{if } \sum_{i=0}^{n-1} w_i x_i \geq 0 \\ -1, \text{if } \sum_{i=0}^{n-1} w_i x_i < 0 \end{cases}$$

where $w_0 = -\theta$ and $x_0 = 1$

where w_i is the weight of the connection of node x_i with the final node, and θ is the value of the polarization. y' is the value that, for the elementary event whose value is unknown, gives the two-layer neural network or perceptron, and y is the real value of said elementary event, which we only know in the values of the training sample.

Therefore, to train or define a perceptron for the resolution of a specific classification problem, it is necessary to obtain the weights of each of the connections between the input and output nodes and add them by means of a sum, from which the polarization value is subtracted. To do this, the sample of complete events will be used, that is, the values of all its elementary events, which are available, are known.

To determine the weights that define the perceptron, a 3-step process is followed:

Step A. Step A consists of two substeps:

Step 1. The perceptron is initialized by arbitrarily assigning weights, comprised in the interval [−1, 1], to the connections of each node with the final node that is found.

Step 2. The value of y' is calculated for the first event in the training sample being used. The order of analysis of the events is chosen arbitrarily.

Step A of definition of a perceptron.[17] Step A consists of two substeps:

Step 1. Choice of initial values of the weights. The first thing to do is to arbitrarily choose the values for the weights in the interval [−1, 1]. As we are working with events composed of three elementary events, following the equation above, we must define three weights w_1, w_2, w_3, and the values chosen are:

$$w_1 = 0.6, w_2 = 0.3, w_3 = -0.2$$

Step 2. Calculation of y' for the first event.

Next, taking the sample of events that we are using, the events are calculated, which for event 1 are:

$$y' = \sum_{i=0}^{n-1} w_i x_i = -0.6(1) + 0.3(1) - 0.2(1) = -0.5$$

Since $-0.5 < 0$, the value of $y' = -1$ does not match with y

Step 2. If the value of y' is equal to that of y, y' is calculated for the second event using the same weights, and this process is repeated in the same way as long as the

[17] We rewrite the sample to make the text easier to read: 1. {1, 1, 1}; 2. {1, -1, -1}; 3. {-1, 1, -1}; 4. {-1, -1, -1}.

value of y' coincides with y for the different events of the training sample. In the event that y' and y do not match, for the first event analyzed or for any subsequent event, the weights must be recalculated. For which the equation is used:

$$w_i(k+1) = w_i(k) + \lambda(y - y'(k))x_i$$

where k refers to the iteration in the calculation of weights and λ is a factor called the learning ratio, with a value in the interval [0,1], which is arbitrarily chosen by the network designer. It must be taken into account that when observing the equation, if λ is close to 0, the new value of the weights will be very close to the old value, and if it is close to 1, the new value will be heavily influenced by the amount of adjustment made in the iteration. Once the weights have been recalculated, step 1.2 is carried out with the next event in the sample.

Step 2 of the definition of a perceptron. Recalculation of weights. As the value of $y' = -1$ does not coincide with y, the weights are recalculated, for which the following equation is used:

$$w_i(k+1) = w_i(k) + \lambda(y - y'(k))x_i$$

where $k = 1$ because it is the first iteration, and for λ, the most neutral value possible $\lambda = 0.5$ is taken, with which the calculations remain:

$$\begin{aligned} w_0(2) &= w_0(1) + 0.5(y - y'(1))x_0 = \\ &= -0.6 + 0.5(1 - (-1))(1) = -0.6 + 1 = 0.4 \\ w_1(2) &= w_1(1) + 0.5(y - y'(1))x_1 = \\ &= 0.3 + 0.5(1 - (-1))(1) = 0.3 + 1 = 1,3 \\ w_2(2) &= w_2(1) + 0.5(y - y'(1))x_2 = \\ &= -0.2 + 0.5(1 - (-1))(1) = -0.2 + 1 = 0.8 \end{aligned}$$

Substep 1.2 Calculation of y' for the second event.

Next, taking the sample events that we are using, the events are calculated as follows:

Event 2:

$$y' = \sum_{i=0}^{n-1} w_i x_i = 0.4(1) + 1.3(1) + 0.8(-1) = 0,9$$

Since 0.9 is > 0, the value of $y' = 1$ is different from y.

Step 2. Recalculation of weights. Consequently, we have to redo step 2 and recalculate the weights:

$$w_i(k+1) = w_i(k) + \lambda(y - y'(k))x_i$$

where $k = 2$ because it is the second iteration, and for λ, the most neutral value possible $\lambda = 0.5$ is taken, with which the calculations remain:

$$\begin{aligned} w_0(3) &= w_0(2) + 0.5(y - y'(2))x_0 = \\ &= 0.4 + 0.5(1 - (-1))(1) = -0.4 - 1 = -0.6 \\ w_1(3) &= w_1(2) + 0.5(y - y'(2))x_1 = \\ &= 1.3 + 0.5(1 - (-1))(1) = 1.3 - 1 = 0,3 \\ w_2(3) &= w_2(2) + 0.5(y - y'(2))x_2 = \\ &= 0.8 + 0.5(1 - (-1))(-1) = 0.8 + 1 = 1.8 \end{aligned}$$

We return to substep 1.2

Substep 1.2 Calculation of y' for the third event.

Event 3:

$$y' = \sum_{i=0}^{n-1} w_i x_i = 0.4(1) + 1.3(1) + 0.8(-1) = 0,9$$

Since 0.9 is > 0, the value of $y' = 0.9$, which does not match y.

Step 2. Recalculation of the weights.[18] Consequently, we have to redo step 2 and recalculate the weights:

where $k = 3$ because it is the third iteration and $\lambda = 0.5$, with which the calculations remain:

$$\begin{aligned} w_0(4) &= w_0(3) + 0.5(y - y'(3))x_0 = \\ &= -0.6 + 0.5(1 - (-1))(1) = -0.6 - 1 = -1.6 \\ w_1(4) &= w_1(3) + 0.5(y - y'(3))x_1 = \\ &= 0.3 + 0.5(1 - (-1))(-1) = 0.3 + 1 = 0.8 \\ w_2(4) &= w_2(3) + 0.5(y - y'(3))x_2 = \\ &= 1.8 + 0.5(1 - (-1))(1) = 1.8 - 1 = 0.8 \end{aligned}$$

We return to substep 1.2

Substep 1.2 Calculation of the fourth event.

[18] We rewrite the sample to make the text easier to read: 1. {1, 1, 1}; 2. {1, -1, -1}; 3. {-1, 1, -1}; 4. {-1, -1, -1}.

Event 4:

$$y' = \sum_{i=0}^{n-1} w_i x_i = -1.6(1) + 1.3(-1) + 0.8(-1) = -3.7$$

Since $-3.7 < 0$, the value of $y' = -1$ matches y. Therefore, there is no need to recalculate the weights.

Step 3. Once we have some weights that give us a correct classification, we apply them to the rest of the events that had an incorrect classification to see if they either classify them correctly or they serve as a basis to continue with the correction of the weights.

Event 1:

$$y' = \sum_{i=0}^{n-1} w_i x_i = -1.6(1) + 1.3(1) + 0.8(1) = 0.5$$

Since 0.5 is > 0, the value of $y' = -1$ matches y

Event 2:

$$y' = \sum_{i=0}^{n-1} w_i x_i = -1.6(1) + 1.3(1) + 0.8(-1) = -0.6$$

since -0.6 is < 0, the value of $y' = -1$, which matches y

Event 3:

$$y' = \sum_{i=0}^{n-1} w_i x_i = -1.6(1) + 1.3(-1) + 0.8(1) = -2.1$$

since -2.1 is < 0, the value of $y' = -1$, which matches y

Since event 4 was already well classified, we already have a well-defined neural network that classifies the grades following the same pattern as the sample of grades used. The perceptron is:

$$y' = \begin{cases} 1, & \text{if } \sum_{i=0}^{2} w_i x_i \geq 0, \\ -1, & \text{if } \sum_{i=0}^{2} w_i x_i < 0 \end{cases} \quad \text{with } w_0 = -1.6, w_1 = 1.3, w_2 = 0$$

The perceptron can only find an optimal solution if the problem is linearly separable.

In the example seen above, it has been chosen for its simplicity to teach the fundamentals of how a perceptron works, but it uses all the possible events that can be had to train the network, so it is not possible to verify how it would classify the network.

Naïve Bayes

In this subsection, the use of probability concepts to solve the problem of supervised classification is introduced. This use of probability to solve problems of supervised classification arises from the fact that not always all the factors and characteristics that can affect the performance of a classifier to correctly classify an event are known, and it is very possible that the set of attributes used is not enough to obtain a deterministic classification. That is, the same set of values of the used attributes introduced in the classifier could be classified into different classes. This is because the unknown values of the characteristics not considered are not being introduced in the classifier. Even if all the attributes were considered, the classification could not be deterministic if the scales used to measure the values of those characteristics present any ambiguities.

As an example of when a probabilistic approximation to the problem of supervised classification can be used is the classification of the expectation of life for a person. Even if many characteristics of the genetical information of the person, the health, and the way of life, the food that eat, the exercise that does, is introduced, other factors that for sure impact in the classification can be forgotten. In addition, even if all of them were introduced, the way to measure most of them is going to be ambiguous, for example, which amount of water drinks or how much exercise practices every day.

The problem of supervised classification is stated as the calculation of the probability for a specific value, corresponding to the class, for the variable B. If the probability of B were unconditional, that is, without the impact of the occurrence of any other variable, it would be calculated as P (B), but if the occurrence of B was conditioned for the occurrence of other single event, as A, we saw in the chapter of probability that the probability of B, is P (B | A), because B and A are related. This is the problem that must be solved to have the probability of B when we have the values of not only one A but multiple As.

To develop the classifier for each pair of values of A and B in the training set, it would be necessary to know a posterior P (B | A), and every register in the test set should be classified in class B, which gives the higher value for P (B | A).

As an example of the probabilistic approach we follow using the previous one, we are going to classify persons that reach the eighth ten years. As training set we take: Male or Female; Pathology: Cardiac, Respiratory, None; Monthly days of Exercise; Reach the Age. Persons will be classified by Reach the Age or not. For example, we

can have a register in the test set with the following set of values for the variables: Male, Respiratory Pathology, 3 days of exercise. To classify this person as reaching the age of eighty or not reaching the age of eighty, using the probability approach, we must calculate the probability *a posteriori* to reach it, given the values of the rest of the characteristics observed in the training set. If the probability of Yes is higher than the probability of No, the person is classified as having a long life expectancy.

We know, from chapter "Probability", that Bayes Probability allows us to calculate the probability *a posteriori* from the probability a priori using the equation

$$P(B|A) = \frac{P(A|B)P(B)}{P(A)}$$

In addition, as we are going to compare the values of $P(B|A)$ for different values of B, but ever with the same set of As, the denominator, $P(A)$ is ever the same and can be ignored, and the probability a priori of B, P(B), can be calculated from the test set calculating the amount of data that belongs to class B. Consequently, the remaining value in the equation that we need to calculate is $P(A|B)$. In the following, two different techniques to solve it are introduced, both of which are included in the set known as Bayesian Classifiers and are Naïve Bayes and Bayesian Belief Network.

Naïve Bayes Classifier

Naïve Bayes classifier estimates the need value of $P(A|B)$ when there is more than one characteristic, or As, as is usual, assuming that they are independent. Consequently, as we saw in chapter "Probability", their probability of co-appearance is calculated by multiplying their individual probability of appearance, that is, using the equation:

$$P(A|B) = \prod_{i=1}^{n} P(A_i|B)$$

where each set of characteristics that constitutes an instance is configured by the n values of n As and B.

$$\{A_1, .., A_n, B\}$$

Consequently, although the estimation of the probabilities *a posteriori* of each combination of kinds of classes and attribute values is a difficult problem because it requires a very large training set, even for a few attributes, the Naïve Bayes classifier, assuming the independence of attributes, allows us to solve it in an easier way because instead of having to calculate the probability of each combination of As and B, only is needed to calculate the probability of a given A_i, which produces a very large reduction in the training set needed.

To classify a test instance, the probability *a posteriori* of all the possible values of the classifier, that is, the characteristic used to classify, must be calculated for all the sets of values of the rest of the characteristics, $\{A_1, .., A_n, B\}$, and that one with more probability will allow us to classify the instance. As we saw above, the equation used is:

$$P(B|A) = \frac{P(A|B)P(B)}{P(A)}$$

where

$$P(A|B) = \prod_{i=1}^{n} P(A_i|B)$$

Consequently, we have

$$P(B|A) = \frac{\prod_{i=1}^{n} P(A_i|B)P(B)}{P(A)}$$

As we saw that $P(A)$ is the same for all of them, the needed equation is

$$P(B|A) = \prod_{i=1}^{n} P(A_i|B)P(B)$$

Consequently, we will need to calculate $\prod_{i=1}^{n} P(A_i|B)$ for all the qualitative and quantitative characteristics in the problem. Let's see now how we calculate them.

Qualitative Characteristics

For qualitative attributes, the conditional probability $P(A_i|B)$ is calculated as the relative frequencies of the values of the characteristic, such as A_i, in the training set, when and that is very important, the characteristic B takes each one of the classification values or classes.

To see an example of the calculation of the conditional probability of Qualitative Characteristics, we come back to the problem of life expectation. The complete training set is:

Data identifier	Gender	Pathology	Monthly exercise days	More than eighty
1	Male	Cardiac	7	No
2	Female	Respiratory	24	Yes
3	Female	Cardiac	13	No

(continued)

Data identifier	Gender	Pathology	Monthly exercise days	More than eighty
4	Male	Respiratory	8	No
5	Female	None	21	Yes
6	Male	Respiratory	4	No
7	Male	None	8	No
8	Female	Cardiac	22	Yes
9	Female	Respiratory	23	Yes
10	Female	Cardiac	14	No

With these data, we can calculate the probability for each value of each characteristic A_i, that is, Male and Female for A_1 Gender; and Cardiac, Respiratory, or None, for A_2, Pathology; for each class in B, that are more than eighty, yes or not.

$P(A_i|B)$ = P (Male | > 80) = 0/4
P (Male | < 80) = 4/6
P (Female | > 80) = 4/4
P (Female | < 80) = 2/6

$P(A_i|B)$ = P (Cardiac | > 80) = 1/4
P (Cardiac | < 80) = 3/6

P (Respiratory | > 80) = 2/4
P (Respiratory | < 80) = 2/6

P (None | > 80) = 1/4
P (None | < 80) = 1/6

Quantitative Characteristics

For quantitative characteristics, when they have quantitative continuous data, the conditional probability $P(A_i|B)$ cannot be applied in the same way as for qualitative characteristics, and there are two main ways to calculate the required probabilities:

- The first way is to convert the continuous value into a discrete value and apply the same manner as for qualitative characteristics. To do that, we can use the grouping of data into classes, as we saw in chapter two, through the use of intervals; that is, we divide the continuous data into intervals and use the mark if the interval is the discrete value that represents the continuous values of that class. The problem of this solution is the correct selection of the number of intervals.
- The second way is to fit a probability distribution for the observed data of the characteristic and to use the training set to obtain the parameters of that distribution. The most commonly used distribution to make a first approximation is the Normal or Gaussian probability distribution that was introduced in chapter three. Normal distribution is characterized by the parameters, mean, and standard

deviation, that, as we know, for populations are called by the Greek letters μ and σ. The equation is:

$$P(A_i|B) = \frac{1}{\sqrt{2\pi}\cdot\sigma}\, exp\left(-\frac{(a-\mu)^2}{2\sigma^2}\right)$$

To see an example of the calculation of the conditional probability of quantitative continuous characteristics, we return to the problem of life expectations. The continuous value in this case is the time dedicated to exercise each month. We use the second way to calculate $P(A_i|B)$, which in this case is P (monthly time of exercise | not reaching the age expectation of 80). We suppose that the time of exercise has a Normal probability distribution, and consequently, we must find its mean and standard deviation. To calculate both of them, we use the training set of life expectation:

$$\mu = \overline{x} = \frac{7+13+8+4+8+14}{6} = \frac{54}{6} = 9$$

$$\begin{aligned}\sigma &= s \\ &= \sqrt{\frac{(7-14.4)^2+(13-14.4)^2+(8-14.4)^2+(4-14.4)^2+(8-14.4)^2+(14-14.4)^2}{6}} \\ &= 6.42\end{aligned}$$

From these two data, we have characterized the Normal distribution as

$$P(Time\ of\ exercice| < 80) = \frac{1}{\sqrt{2\pi}\cdot 6,42}\, exp\left(-\frac{(a-9)^2}{2.41,16}\right)$$

With this equation, we can calculate the probability of the number of days of exercise for someone who has not reached the eight decades. For example, if someone who does not reach eighty years of age has dedicated 15 days monthly to exercise, that is, $a = 15$:

$$P(\ 15| < 80) = \frac{1}{\sqrt{2\pi}\cdot 6,42}\, exp\left(-\frac{(15-9)^2}{2.41,16}\right)$$

Here, we must take into account that we are obtaining the probability of a point through a density function of probability and, consequently, should be zero, and the obtained probability is for an interval of approximately a.

Once we have seen how to obtain the conditional probability of the qualitative and quantitative characteristics, we are going to solve the complete problem of the supervised classification of the expectation of life. We classify one person who is male, has a respiratory disease, and practices exercise 15 days per month. The

problem is to classify him, that is, if he belongs to the group that lives more than 80 years or not.

Consequently, from a probabilistic point of view, we must establish the probability *a posteriori*

$$P(>80|A) \text{ and } P(<80|A)$$

and the highest probability will establish to which class the person belongs.

From the above, we know that the equations that must be used to calculate both probabilities are:

$$P(>80|A) = \prod_{i=1}^{3} P(A_i|>80)P(>80)$$

$$P(<80|A) = \prod_{i=1}^{3} P(A_i|<80)P(<80)$$

where the three A characteristics are Gender, Health, and Days of Exercise. We have calculated their probabilities *a posteriori* for this case in the previous exercises.

$P(A_i|B)$ = P (Male | > 80) = 0/4
P (Male | < 80) = 4/6

P (Respiratory | > 80) = 2/4
P (Respiratory | < 80) = 2/6

Consequently, to solve the exercise, we only need to calculate P(15 | >80)

We need to calculate the mean and the standard deviation for the data >80, that is:

$$\mu = \overline{x} = \frac{7+13+8+4+8+14}{6} = \frac{54}{6} = 9$$

$$\begin{aligned} \sigma &= s \\ &= \sqrt{\frac{(7-14.4)^2+(13-14.4)^2+(8-14.4)^2+(4-14.4)^2+(8-14.4)^2+(14-14.4)^2}{6}} \\ &= 6.42 \end{aligned}$$

From these two data, we have characterized the Normal distribution as

$$P(\ 15|>80) = \frac{1}{\sqrt{2\pi}\cdot 6,42}\ exp\left(-\frac{(15-9)^2}{2.41,16}\right) =$$

We need to calculate $P(>80)$ and $P(<80)$, and from the training set, both amounts can be calculated.

$$P(>80) = 4/10$$
$$P(>80) = 6/10$$

Consequently,

$$P(>80|A) = \prod_{i=1}^{3} P(A_i|>80)P(>80) = \text{P (Male I} > 80\text{). P (Respiratory I} > 80\text{). P(15 I} > 80\text{).}P(>80) = \tfrac{0}{4}.\tfrac{2}{4}.x.\tfrac{4}{10} = 0$$

$$P(<80|A) = \prod_{i=1}^{3} P(A_i|<80)P(<80) = \text{P (Male I} < 80\text{). P (Respiratory I} < 80\text{). P(15 I} < 80\text{).}P(<80) = \tfrac{4}{6}.\tfrac{2}{6}.x.\tfrac{6}{10} =$$

Consequently, as $P(<80|A) > P(>80|A)$, this person is classified in the class that does not reach the eighth decade, that is, No.

Regression Functions

Regression studies[19] seek to define a function, called a regression model, that allows, from the quantitative values of $n-1$ elementary events of the n elementary events that make up an event, to determine the value of the remaining elementary event, with the objective of obtaining the value of one of them, the dependent one, as a function of the rest of them, the independent ones.

The term regression has its origin in the Latin term regressio, -ōnis and means retrocession or action of going back. It was first introduced in the mid-nineteenth century by the English geneticist Francis Galton, based on a discovery he made in his studies of genetic inheritance, which he called "regression to the mean", and which consisted of what heights of the children tended to "return" to the average height of the population, that is, the average height of the children of tall parents was lower than the average height of their parents and that the average height of the children of short parents was higher than the average height of their parents. Galton developed regression analysis to study this effect, which he referred to as the "regression to mediocrity". Although the RAE dictionary does not yet include this mathematical meaning of the term regression, important dictionaries in the English language, such as Webster, do include it (Webster): "//d: a functional relationship between two or more correlated variables that is often empirically determined from data and is used especially to predict values of one variable when given values of the others <the regression of y on x is linear>; specifically: a function that yields the mean value of a

[19] In most data analysis books, regression studies are included within the supervised classification chapter since it can be understood as the search for a classification model to determine the value of an elementary event. In this text we have opted for this organization that seems better to understand the concepts.

random variable under the condition that one or more independent variables have specified values ".

Example of Regression. To introduce the concept of event regression, we use the grades of a subject from a group of students. The qualifications will be formed by two grades, the qualifications of the Theory and Laboratory tests. The elementary events are each of the grades individually E = {Theory, Laboratory}. Each grade can have any real value between 0 and 10. The regression function sought in this case will be the one that, based on the values of the first elementary event, the theory grade, allows obtaining the student's practice grade.

To obtain the regression function that best relates the events, different regression techniques can be used. This differentiation is based on the fact that each of them uses different mathematical functions. All of them use a sample or set of events for which the values of all elementary events are known, including the one to be inferred, to define the regression function. Once the function is defined, it will be used to infer the value of new events.

Example of Regression. Sample. The sample of events that will allow finding the definition of the regression function is made up of the following ten events: 1. {2.5, 3.7}; 2. {2.2, 1.7}; 3. {4.2, 4.5}; 4. {6.3, 5.1}; 5. {5.3, 7.2}; 6. {7.4, 8.5}; 7. {8.2, 7.2}; 8. {9.5, 10}; 9. {5, 5.3 }; 10. {9, 7.3}

In any case, to know which function can best fit the observed data, it is advisable to first make the scatter diagram of the same. If the typical shape of different linear and nonlinear functions is previously known, it can be seen which of them best suits the analysed diagram. It is also very useful not only to perform the representation with ordinary numerical scales but also to use logarithmic scales on one axis, called semilogarithmic representations, or in the two axes, called logarithmic representations, since, as will be seen later, logarithmic transformations allow a better analysis of the data.

The best known and most commonly used regression techniques are as follows:

- Linear regression of polynomials (or linear fit)
- Nonlinear regression of polynomials
- Regression or nonlinear fit with other types of equations other than polynomials

We are going to see how each of them works in a specific way.

Linear Regression of Polynomials (or Linear Fit) for Two Events

We begin by studying the case of two-dimensional events, that is, those formed by only two elementary events. The linear fit with a linear equation can be defined as "equation whose variables are of the first degree" and a linear function can be defined as "The one whose variable or variables are of the first degree ". And a polynomial can be defined as: "An expression composed of two or more algebraic terms joined by the plus or minus signs. Those with two or three terms are given the special names of binomial and trinomial, respectively" and linear can be defined as "That it has effects proportional to the cause". From the data of observed events

formed by two elementary events tries, following the definitions given above in the text and in the footnote references, to obtain an equation or linear function that relates both elementary events.

Mathematically the linear regression function is obtained as follows: if we call the regression function f; to the elementary input events x_i; and to the elementary output events y_i; and $f(x_i)$ to the elementary output events calculated through the linear regression function, the sum of the difference between the real value of each elementary event in the sample considered the output value and the value obtained for said elementary event, if the function defined with input value is used, the elementary event to which each output event is related is given by the function:

$$y = a + bx$$

where a and b are the regression coefficients.

The most widely used adjustment method is that of *Least Squares*, which is based on finding the function that minimizes the sum of the square of the difference between the real value of each elementary event in the sample, considering the output value, and the value that is obtained for said elementary event if the function defined with the input value is used to calaculate de output obtained when the same elementary event is introduced. That difference is called the remainder. The sum of all those differences in the sample is called the quadratic error, or the sum of the squares of the residuals. If the difference between the observed value of every elementary event in the sample and the value obtained for said elementary event if the function is used, is calcaluted and it is not squared, and the sum of all the differences is obtained, the result is called the absolute error. Taking into account its definitions, the mathematical expressions of both errors are:

$$ea = \sum_{i=1}^{n} (y_i - y_{ci})$$

$$ec = \sum_{i=1}^{n} (y_i - y_{ci})^2$$

Example of Regression. Least squares method. Taking the sample defined in the previous step, the linear least squares regression function will be the one that minimizes the difference of the sum described. In this case, the values to calcualte said difference would be: for example, for the first event, value 3.7, and the value that we would obtain from introducing it as an input value in linear function or regression line is value 2.5. For the second event, the value is 1.7, and the value that we would obtain from introducing it as an input value in linear function or regression line is value 2.2. These values would be calculated, in the same way, for the rest of the events until the entire sample is completed.

Consequently, the least squares adjustment method seeks that the following summation is minimum:

$$\sum_{i=1}^{n}(y_i - y_{ci})^2$$

where y_i is the value observed for elementary event y, corresponding to elementary event x_i, in the event formed by the pair of elementary events (x_i, y_i), and y_{ci} is the value obtained for elementary event y when x_i is introduced into the linear regression equation. From this premise, we will find the values of the parameters a and b.

To obtain the minimum value of the equation, the following calculations are performed:

First, the squared binomial is developed:[20]

$$\sum_{i=1}^{n}(y_i - y_{ci})^2 = \sum y^2 + y_c^2 - 2yy_c$$

Second, y_{ci} is replaced by its value based on the values of the final equation that will be obtained,

$$y_{ci} = a + bx_i$$

That is, we substitute this equation in the previous equation, and we obtain:

$$\sum \left(y^2 + (a + bx_i)^2 - 2y(a + bx_i) \right)$$

Third, to compute the values of a and b that minimize the equation, we derive with respect to a and b and set the equations equal to 0.

$$\begin{cases} \dfrac{\partial\left(\sum_{i=1}^{n}\left(\left(y_i^2 + a^2 + b^2x_i^2 + 2abx_i - 2y_ia - 2by_ix_i\right)\right)\right)}{\partial a} = \\ \quad \sum_{i=1}^{n}(a + bx_i - y_i) = na + b\sum x_i - \sum y_i = 0 \\ \dfrac{\partial\left(\sum_{i=1}^{n}\left(\left(y_i^2 + a^2 + b^2x_i^2 + 2abx_i - 2y_ia - 2by_ix_i\right)\right)\right)}{\partial b} = \\ \quad \sum_{i=1}^{n}\left(2bx_i^2 + 2ax_i - 2y_ix_i\right) = b\sum x_i^2 + a\sum x_i - \sum y_ix_i = 0 \end{cases}.$$

If we multiply the first by $\sum x_i$ and the second by n, we have:

[20] We do not write in all the sums the limits of the summation to improve the readability of the text, but in all of them it is $\sum_{i=1}^{n}$.

$$\begin{cases} \left(na + b\sum x_i - \sum y_i\right)\sum x_i = 0 \\ \left(b\sum x_i^2 + a\sum x_i - \sum y_i x_i\right)n = 0 \end{cases}$$

and we obtain the system of equations known as the *normal equation*. We are going to solve it in two ways:

The first is similar to a system of equations; for this, we subtract them and obtain:

$$na\sum x + b\sum x\sum x - \sum y\sum x - nb\sum x^2 - na\sum x + n\sum yx = 0$$

Solving for the value of b, we have

$$b = \frac{\sum x\sum y - n\sum xy}{(\sum x)^2 - n\sum x^2}$$

and dividing the numbered and denominator by n, we have:

$$b = \frac{n\overline{xy} - \sum xy}{n\bar{x}^2 - \sum x^2}$$

b is often called the *adjustment coefficient*.[21] If the equations seen in the data topic of covariance,[22] correlation,[23] and standard deviation are taken, the value of b can be obtained as:

[21] A coefficient can be defined as: (Co and efficient) .//4. m. Mat. Constant factor that multiplies an expression, generally located to its left.

[22] To obtain s_{xy}, the covariance of x and y, that is a measure of the dependence between x and y, the following equation is used

$$s_{xy} = \frac{\sum_{i=1}^{n} x_i y_i}{n} - \left(\frac{\sum_{i=1}^{n} x_i}{n}\right)\cdot\left(\frac{\sum_{j=1}^{n} y_j}{n}\right) = \frac{\sum_{i=1}^{n}\sum_{j=1}^{n}(x_i - \bar{x})(y_j - \bar{y})}{n}$$

[23] As the covariance is not a normalized value, other measure, correlation, is defined. Correlation is obtained with the equation

$$r_{xy} = \frac{s_{xy}}{s_x s_y}$$

The values of *r_xy* are in the interval [−1 and 1].

If the linear correlation is perfect, that is, if the values of x and y lie on a line, the value of r will be -1 if it has a negative slope and 1 if it has a positive slope.

If *r* is equal to 0, there is no linear dependence between the variables, which implies either that the variables are independent, or that there is a nonlinear dependence between them.

$$b = \frac{s_{xy}}{s_x^2} = r_{xy} = \frac{s_y}{s_x}$$

a is solved as a function of b in the first equation:

$$a = \frac{\sum y - b\sum x}{n} \rightarrow a = \overline{y} - b\overline{x}$$

Linear regression by the least squares method.[24] Following the equations above, to calculate the regression of the sample we are working with, we need to calculate $\overline{x}$, $\overline{y}$, s_x and/or s_{xy} or r_{xy}. However, if we take the calculation equation for r_{xy}, we see that it is equal to $r_{xy} = \frac{s_{xy}}{s_x s_y}$, so what we have to calculate is $\overline{x}$, $\overline{y}$, s_x and/or s_{xy}.

We calculate each of these magnitudes:

$$\overline{x} = \frac{\sum_{i=1}^{n} f_i x_i}{\sum_{i=1}^{n} f_i} = \frac{2.5 + 2.2 + 4.2 + 6.3 + 5.3 + 7.4 + 8.2 + 9.5 + 5 + 9}{10} = 5.96$$

$$\overline{y} = \frac{\sum_{j=1}^{m} f_j y_j}{\sum_{j=1}^{m} f_j} = \frac{3.7 + 1.7 + 4.5 + 5.1 + 2(7.2) + 8.5 + 10 + 5.3 + 7.3}{10} = 6.05$$

$$s_x = \sqrt{\frac{\sum_{i=1}^{n} f_i (x_i - \overline{x})^2}{\sum_{i=1}^{n} f_i}} = \sqrt{\frac{(2.5 - 5.96)^2 + \ldots + (9 - 5.96)^2}{10}} = 2.44$$

$$s_y = \sqrt{\frac{\sum_{j=1}^{m} f_j \left(y_j - \overline{y}\right)^2}{\sum_{j=1}^{m} f_j}} = \sqrt{\frac{(3.7 - 6.05)^2 + \ldots + (7.3 - 6.05)^2}{10}} = 2.32$$

$$s_{xy} = \frac{\sum_{i=1}^{n}\sum_{j=1}^{m} f_{ij} x_i y_j}{\sum_{i=1}^{n} f_i} - \overline{x}.\overline{y} = \frac{2,5.3.7 + \ldots + 9.7,3}{10} - (5.96)(6.05)$$
$$= 41.23 - 36.05 = 5.17$$

[24] To make the text easier to read, we remember here that the sample we are working with is 1. {2.5, 3.7}; 2. {2.2, 1.7}; 3. {4.2, 4.5}; 4. {6.3, 5.1}; 5. {5.3, 7.2}; 6. {7.4, 8.5}; 7. {8.2, 7.2}; 8. {9.5, 10}; 9. {5, 5.3}; 10. {9, 7.3}.

$$r_{xy} = \frac{s_{xy}}{s_x s_y} = \frac{5.17}{(2.44)(2.32)} = 0.91$$

Consequently, the equation of the linear function is:

$$b = \frac{s_{xy}}{s_x^2} = \frac{5.17}{2.44^2} = 0.92$$

$$a = \overline{y} - b\overline{x} = 6.05 - 0,92.5,6 = 0.88$$

$$y = a + bx = 0.88 + 0.92x$$

The second way is through matrix algebra. We remember that the normal equation is:

$$\begin{cases} \left(na + b\sum x_i - \sum y_i\right)\sum x_i = 0 \\ \left(b\sum x_i^2 + a\sum x_i - \sum y_i x_i\right)n = 0 \end{cases}$$

If we eliminate in the first equation $\sum x_i$, we have:

$$\begin{cases} \left(na + b\sum x_i - \sum y_i\right) = 0 \\ \left(b\sum x_i^2 + a\sum x_i - \sum y_i x_i\right)n = 0 \end{cases}$$

If we clear a and b, we have:

$$\begin{pmatrix} n & \sum x_i \\ \sum x_i & \sum x_i^2 \end{pmatrix}\begin{pmatrix} a \\ b \end{pmatrix} = \begin{pmatrix} \sum y_i \\ \sum y_i x_i \end{pmatrix}$$

Therefore, solving for the matrix $\begin{pmatrix} a \\ b \end{pmatrix}$, we have the solution for a and b:

$$\begin{pmatrix} a \\ b \end{pmatrix} = \begin{pmatrix} n & \sum x_i \\ \sum x_i & \sum x_i^2 \end{pmatrix}^{-1}\begin{pmatrix} \sum y_i \\ \sum y_i x_i \end{pmatrix}$$

Linear Regression of Polynomials (or Linear Fit) for Three Events

As in the previous case of two dimensions, the linear fit of the data of observed events formed by three elementary events tries to obtain a linear equation or function that relates both elementary events. In this case, a line will not be obtained in a plane but a plane in space. This is the last dimension in which we can obtain a graphical representation to be able to visually observe the behaviour of the data.

Mathematically, the linear regression function is obtained as follows: if we call the regression function f; to the elementary input events x_i e y_i; and to the elementary output events z_i; and f_i to the elementary output events calculated through the linear regression function, the sum of the difference between the real value of each elementary event in the sample considered the output value and the value obtained for said elementary event, if the function defined with input value is used, the elementary event to which each output event is related is given by the function:

$$z = a + bx + cy$$

where a, b, and c are the regression coefficients. This function is called the fit plane.[25,26] To perform the adjustment and obtain the parameters a, b, and c, the least squares method will be used, as in the case of two dimensions; therefore, it must be minimized:

$$\sum_{i=1}^{n} \left(z_{i_o} - z_{i_c}\right)^2$$

To obtain the minimum value in this equation, we carry out a process similar to the one we did in dimension two, consisting of the following steps: first, substitute z_{i_c} by its value as a function of the values of a, b, and c, using the equation; second, following the calculation rules, the equation resulting from the previous step is derived with respect to a, b, and c, which are the two variables whose values make the value of the equation a minimum, we want to calculate; third, the three resulting equations are set equal to zero; fourth, z_{i_o} is solved in both equations, obtaining a system of three equations with three unknowns called the system of normal equations:

[25] Quadratic can be defined as: "//2. adj. Mat. Which has squares as the highest power" and defines parable as: "(From lat. parabŏla, and east from Gr.) .//2. F. Geom. Locus of the points of the plane equidistant from a line and a fixed point, which results from cutting a right circular cone by a plane parallel to a generatrix".

[26] t is called an adjustment plane because it is geometrically a plane. It is proved from the following proposition: if x is taken constant, the equation is that of a line in three-dimensional space; and if y is taken constant, the equation is another line in three-dimensional space. Consequently, the equation is the one that contains both lines and therefore is the equation of the plane of fit.

$$\sum_{i=1}^{n}\left(z_{i_0}-(a+bx+cy)\right)^2$$

$$\frac{d\left(\sum_{i=1}^{n}\left(z_{i_0}-(a+bx+cy)\right)^2\right)}{da}=$$

$$-2\sum_{i=1}^{n}\left(z_{i_0}-(a+bx+cy)\right).\sum_{i=1}^{n}1=0$$

$$\frac{d\left(\sum_{i=1}^{n}\left(z_{i_0}-(a+bx+cy)\right)^2\right)}{db}=$$

$$-2\sum_{i=1}^{n}\left(z_{i_0}-(a+bx+cy)\right).\sum_{i=1}^{n}x=0$$

$$\frac{d\left(\sum_{i=1}^{n}\left(z_{i_0}-(a+bx+cy)\right)^2\right)}{dc}=$$

$$-2\sum_{i=1}^{n}\left(z_{i_0}-(a+bx+cy)\right).\sum_{i=1}^{n}y=0$$

and you get

$$\begin{cases}\sum_{i=1}^{n}z_{i_o}=a.n+b\sum_{i=1}^{n}x_i+c\sum_{i=1}^{n}y_i\\ \sum_{i=1}^{n}z_{i_o}.x_i=a\sum_{i=1}^{n}x_i+\sum_{i=1}^{n}x_i^2+c\sum_{i=1}^{n}x_i\sum_{i=1}^{n}y_i\\ \sum_{i=1}^{n}z_{i_o}.y_i=a\sum_{i=1}^{n}y_i+b\sum_{i=1}^{n}x_i\sum_{i=1}^{n}y_i+\sum_{i=1}^{n}y_i^2\end{cases}$$

This system is solved, as in the previous case, by using matrix calculation.

Linear Regression of Polynomials (or Linear Fit) for K Events

It is not possible to make graphical representations, so the adjustments must be made directly through mathematical calculations and observation of the results obtained.

The linear fit of the data of k observed characteristics tries to obtain an equation or linear function that relates the k characteristics. That is, if we have k characteristics $x, y, z, \ldots k$ the linear fit will obtain an equation such as:

$$k=a+bx+\text{cy}\ldots+tj$$

To carry out the adjustment and obtain the parameters $a, b, \ldots, t$, the least squares method is used; therefore, it must be minimized:

$$\sum_{i=1}^{n}(k_{i_o}-k_{i_c})^2$$

To obtain the minimum value in the equation, proceed as in the previous subsection.

No Linear Regression of Polynomials (or Linear Fit) for 2 Events

We start with the simplest nonlinear fit, which is the quadratic.[27] The quadratic or parabolic nonlinear fit of the data of two observed characteristics tries, following the definitions given above, to obtain a quadratic or parabola equation or function that relates both characteristics. That is, for the characteristics x and y between which there is a binary relationship, the quadratic nonlinear fit will obtain an equation such as:

$$y=a+b\mathrm{x}+\mathrm{c}x^2$$

To carry out the adjustment and obtain the parameters a, b, and e, the least squares method is used, which, as in the previous subsection, consists of minimizing the sum of the squares of the differences between the values of the dependent variable, obtained through the measured or observed data and those obtained through the above equation. That is, in minimizing:

$$\sum_{i=1}^{n}\left(y_{i_o}-y_{i_c}\right)^2$$

To obtain the minimum value in the equation, the following process is followed: first, y_{i_c} is replaced by its value based on the values of a, b, and c, using the initial equation; second, following the calculation rules (say more), the equation resulting from the previous step is derived with respect to a, b, and e, which are the two variables whose values make the value of the equation a minimum, we want to calculate; third, the three resulting equations are set equal to zero; fourth, we solve for y_{i_o} in both equations, obtaining a system of three equations with three unknowns called the system of normal equations:

$$\sum_{i=1}^{n}\left(y_{i_0}-\left(a+bx_i+cx_i^2\right)\right)^2$$

[27] Quadratic can be defined as: "//2. adj. Mat. Which has squares as the highest power" and defines parable as: "(From lat. parabŏla, and east from Gr.) .//2. F. Geom. Geometric location of the points of the plane equidistant from a line and a fixed point, which results from cutting a right circular cone by a plane parallel to a generatrix".

$$\begin{cases} \dfrac{d\left(\sum_{i=1}^{n}\left(y_{i_0}-(a+bx_i+cx_i^2)\right)^2\right)}{da}= \\ -2\sum_{i=1}^{n}\left(y_{i_0}-(a+bx_i)\right)=0 \\ \dfrac{d\left(\sum_{i=1}^{n}\left(y_{i_0}-(a+bx_i+cx_i^2)\right)^2\right)}{db}= \\ -2\sum_{i=1}^{n}\left(y_{i_0}-\left(a+bx_i+cx_i^2\right)\right)x_i=0 \\ \dfrac{d\left(\sum_{i=1}^{n}\left(y_{i_0}-(a+bx_i+cx_i^2)\right)^2\right)}{de}= \\ -2\sum_{i=1}^{n}\left(y_{i_0}-\left(a+bx_i+cx_i^2\right)\right)x_i^2=0 \end{cases}$$

Operating, we obtain:

$$\begin{cases} \sum_{i=1}^{n} y_{i_o}=n.a+b\sum_{i=1}^{n} x_i+c\sum_{i=1}^{n} x_i^2 \\ \sum_{i=1}^{n} y_{i_o}x_i=a\sum_{i=1}^{n} x_i+b\sum_{i=1}^{n} x_i^2+c\sum_{i=1}^{n} x_i^3 \\ \sum_{i=1}^{n} y_{i_o}{}^2x_i=a\sum_{i=1}^{n} x_i^2+b\sum_{i=1}^{n} x_i^3+c\sum_{i=1}^{n} x_i^4 \end{cases}$$

Solving for a in the first, substituting its value in the second and solving the value of b in the first and substituting its value in the third, the value of c is obtained based on the observed data. Once the value of c is obtained, the values of a and b are obtained through the system of equations.

Events of Dimension *k*

The adjustment of the data of two characteristics by means of curves of degree greater than two or *k*-th, whose general equation is given by the one presented below, is carried out following the same procedure described in the two previous subsections.

$$\mathrm{y}=a+bx+\ldots+kx^k$$

No Linear Regression of No Polynomials (or Linear Fit) for 2 Events

It is possible that the two-feature data could have a better fit, that is, with less error, using equations other than polynomials. These equations are described in the following subsections.

Exponential Nonlinear Fit

The exponential nonlinear fit from the data of two observed characteristics tries, following the definitions given above, to obtain an equation or exponential function that relates both characteristics. That is, regardless of the characteristics x and y, between which there is a binary relationship, the exponential adjustment will obtain an equation such as:

$$y = ab^x$$

To carry out the adjustment and obtain the parameters a and b, one method used is to use the logarithmic function to convert the above equation into the linear equation[28]

$$\log(y) = \log(a) + \log(b)x$$

and use the least squares method on the above equation to obtain the values of $\log(a)$ and $\log(b)$. Once these are obtained, the exponential function on them is used to obtain the values of a and b.

Geometric Nonlinear Fit

The geometric nonlinear fit of the data of two observed characteristics tries to obtain an exponential equation or function called geometric that relates both characteristics. That is, whether the characteristics x and y exist between which there is a binary relationship, the geometric fit will obtain an equation[29] such as:

$$y = ax^b$$

To perform the adjustment and obtain the parameters a and b, one method used is to use the logarithmic function to convert the above equation into the linear equation.

$$\log(y) = \log(a) + \text{blog}(x)$$

and on the above equation the least squares method is used to obtain the values of log (a) and b. Once these are obtained, the exponential function on them is used to obtain the value of a.

[28] This equation is called semilogarithmic (it must be by dealing with logarithms a of the variables and see how to write semilogarithmic).

[29] This equation is called a double logarithmic (it must be by dealing with logarithms of the two variables).

Linear Regression: Validity Analysis

Once the regression has been carried out, it is essential to know to what extent the results obtained through it will be good because that is what makes sense of it. To carry out this, numerous methods have been developed, which are based on the analysis of residues.[30] Let us see some of the most used.

To measure how good the regression fit we have performed is, the *r squared* analysis obtains the *coefficient of determination or square correlation a*, for which the following two parameters must be previously calculated:

Calculated deviation or dispersion, *SS*R, of the y values is calculated through the regression function, $\widehat{y}_i$. This value is calculated as the sum of the squares of the difference between the value calculated for each x through the equation of the line, $\widehat{y}_i$, and the mean of the value of y, $\overline{y}$. The calculation equation is:

$$SSR = \sum_{i=1}^{n} (\widehat{y}_i - \overline{y})^2$$

Observed deviation or dispersion, *SSy*, of the observed y values, y_i $. This value is calculated as the sum of the squares of the difference between the observed value for each y and the mean value of y, $\overline{y}$. The calculation equation is:

$$SSy = \sum_{i=1}^{n} (y_i - \overline{y})^2$$

Once the two previous values have been obtained, the *coefficient of determination or square correlation* r^2 is calculated as the ratio between SSR and SSy. The calculation equation is:

$$r^2 = \frac{SSR}{SSy}$$

Once we have them, the square correlation allows us to determine how good the adjustment is since r^2 will have a value that will vary between 0 and 1. In such a way that 0 indicates that there is no adjustment between both magnitudes, while 1 indicates that the fit is perfect.

Standard Error of the Residuals

Once the linear (straight) regression has been obtained, we must analyse to what extent it correctly establishes the relationship between the variables. The second

[30] Residue can be defined as: "(From lat. Residŭum) .//4. m. Mat. Rest of subtraction and division".

analysis is the standard error of the estimate or residual standard deviation. To do this, we calculate the following:

The standard error, s_r, of the residuals or difference between the observed values of y, y_i, and those calculated through the regression function, $\widehat{y}_i$

$$s_r = \sqrt{\frac{\sum_{i=1}^{n} (y_i - \widehat{y}_i)^2}{n}}$$

A value of s_r close to 0 indicates a good fit of the line.

B. Computer-Based Solving

As in the other chapters, this subsection starts with a reminder of what computer-based supervised classification solving means, that is, the application of a systematic process of designing, implementing, and using programming tools to solve the association problem. In this chapter, for reasons of space, we apply computer-based solving only to the first of the techniques introduced in section A, the Decision Trees.

From the sample: 1. {A, A, B, Ap}; 2. {A, B, D, Ss}; 3. {D, C, C, Ss}; 4. {D, B, A, Ss}; 5. {B, C, D, Ss}; 6. {C, B, B, Ap}; 7. {B, B, A, Ap}; 8. {C, D, C, Ss}, using Hunt's algorithm and a Decision Tree, the classification function for the qualifications {Theory, Laboratory, Practices, Global Qualification}, in which the elementary classifying event is the Global Rating, which can have the values Pass, Ap, and Fail, Ss; and the rest of the elementary events can have the values A, B, C and D.

Supervised Classification Exercises Solved in R

In this subsection, the Hunt- and Decision-Tree-supervised classification technique will be used for the classification of the elementary classifying event Global Rating, which can have the values Pass, Ap, and Fail, Ss, using the rest of the elementary events, or students marks, which can have the values A, B, C, and D.

When we start R, there is a set of packages that are loaded by default. To determine which packages are loaded, we use the function *getOption ()*. The instruction is as follows:

```
>getOption ("defaultPackages").
```

This set of initially loaded packages can be modified by reprogramming the start code. In windows, the file that controls this startup code is

Rprofile
and is in the folder
Program Files/R/R-3.1.2/library/base/
In this file, there is a piece of code that is

```
dp <- c ("datasets "," utils "," grDevices "," graphics "," stats "," methods ")
```

In this variable, we can include or remove the packages we want. If we removed them all, only the *base* package would remain, which is not listed because it cannot stop loading for the system to work. We are going to include the *foreing* package within the default packages because we use it a lot, as we have seen in previous practices. For which we introduce it after *methods*. The instruction is as follows:

```
dp <- c ("datasets", "utils", "grDevices", "graphics", "stats", "methods", "foreign")
```

In R, with the packages loaded by default, association analysis cannot be performed, so we will have to load a package that does allow it. Among the various existing packages at http://CRAN.R-Project.org, we will use the rpart package in this exercise.

The first thing we do is check if we have it among the standard library packages, for which we use the *library ()* function that gives us the size of the packages we have and we see that it is not among them, so we have to install it. There are many different alternatives to install a package, and we are going to see one of them, which, although it is not the shortest, allows us to see more additional options. We introduce in R the function: *help.start (),* which opens the R help page in a browser window. It is a page that is essential to know because within it is all the information necessary to work with R. In the case at hand, we will click on the resources link, which takes us to a new page that structures the resources available for R. Click on the link: http://CRAN.R-project.org/within the third section 3. *Archives*, and we go to a new page where all the downloadable files for R. We click on the *Packages* link and we get to a new page where all the packages are available for R. We click on *Table of available packages*, *sorted by name,* and search the *rpart* package. It takes us to a new page where there is absolutely all the information about the package and the downloadable archives.

It is very important to know that each package has a page of this type because it is important not to load them blindly without knowing anything about them. We download the file: *Windows binaries: rpart_4.1-9.zip* because we are working on Windows. It is important to also download the manual of the package to be able to consult it Reference manual: *rpart.pdf*. We download both things in the downloads folder and return to R.

To install the package, we use the Packages menu. Within the *Packages menu,* we click the option to *install package(s) from local zip files* and a window opens that

allows us to select the *rpart_4.1-9.zip* file from the download directory. As it was already created when we did practice 4, the library

```
c:/users/jjcg/documents/R/win-library/3.1 d
```

does not ask us anything about whether we want to create it, as it did in practice 4, but within folder 3.1 create a new folder called rpart and install the library in it. Next, we loaded the rpart package in R using the

```
>library (rpart)
```

function. We execute the search () function to verify that it is installed correctly.

Once we have the rpart package installed, we begin to solve exercise 1.

The first thing we have to do is enter the values of the events of the sample with which we work in R. It is very important to take into account that the events must be represented as a data frame, and a matrix or array will not be valid for the analysis with the functions of the rpart library. However, to enter the data in R, it is easier if we enter them as a matrix and then convert the matrix into a data frame. We introduce the matrix as follows: In each column, we will represent the values of the different elementary events with which we work {Theory, Laboratory, Practices, Global Qualification}, although to facilitate the treatment of the data, we will call them {T, L, P, GC}, and in each row, we will call the values of a determinated event (observation). According to this, the matrix corresponding to the observed sample 1. {A, A, B, P}; 2. {A, B, D, F}; 3. {D, C, C, F}; 4. {D, B, C, F}; 5. {B, C, D, F}; 6. {C, B, B, P}; 7. {B, B, A, P}; 8. {C, D, C, F}, P means Passed, F means Fail, is:

$$\begin{pmatrix} & T & L & P & GC \\ e_1 & A & A & B & P \\ e_2 & A & B & D & F \\ e_3 & D & C & C & F \\ e_4 & D & B & C & F \\ e_5 & B & C & D & F \\ e_6 & C & B & B & P \\ e_7 & B & B & A & P \\ e_8 & C & D & C & F \end{pmatrix}$$

We write the matrix in a txt document, named "sample.txt".

Once entered, we convert it into a data frame with the instruction

```
>m = data.frame (sample)
```

Then, we perform the classification analysis using the rpart functions.

The function that we will use to obtain the classification tree is *rpart ()*, which will provide us with the classification tree defined by the arguments:

- The first argument is the function we want, which is for which variable we will have the classification and depending on which other variables; in this case it is CG ~ T + L + P, as all variables are, it can also be written as CG ~.
- The second argument specifies from which data frame the data are obtained; it is written as data = m or simply m.
- The third argument specifies what type of classification we want (regression, classification, etc.). In this case, since it is qualitative data, it is a classification and it is written as method = "class", although if it is not specified as all data is character takes it by default. We assign the tree the variable name t (from tree).

Therefore, the final instruction is

```
>t = rpart (C. G ~., Da-ta = m, method = "class")
```

Once the instruction is executed, we enter t to see what we have obtained, and we see that the result is

```
n = 8
node), split, n, loss, yval, (yprob)
* denotes terminal node
1) root 8 3 Ss (0.3750000 0.6250000) *
```

This is not what we expected because it only obtains a terminal node for the classification tree. This is because the sample has very little data and the program does not do a larger division, as we have been able to do by hand in theory. To see the partition, we have to enter more data, for which we generate a new sample txt file repeating the values 5 times in a row. If we do the analysis again, we see that we make a partition in the variable L,

```
>t = rpart (CG ~ L, data = mm, method = "class")
```

and in the variable P,

```
>fit = rpart (CG ~ L, data = mm, method = "class")
```

and if we take all

```
>t = rpart (CG ~., data = m, method = "class")
```

it divides the variable P.

The result is:

```
n= 9
```

```
node), split, n, loss, yval, (yprob)
    * denotes terminal node

1) root 9 3 Ss (0.3333333 0.6666667)
2) L=A,B 5 2 Ap (0.6000000 0.4000000)
4) P=A,B 3 0 Ap (1.0000000 0.0000000) *
5) P=C,D 2 0 Ss (0.0000000 1.0000000) *
3) L=C,D 4 0 Ss (0.0000000 1.0000000) *
```

We can perform the same analysis using the *tree* package.

The first thing we do is check if we have them among the standard library packages, for which we use the library () function and we see that they are not between them, so we have to install them. As seen, the packages can be downloaded from

http://CRAN.R-project.org/- 3. Archives - Packages - Table of available packages - sorted by name

and look for the package tree. We downloaded the .zip and the manual. Windows binaries: tree \1.0-37.zip and Reference manual: tree.pdf.

To install the package, we click the menu *Packages,* and within that menu, we press the option *install package(s) from local zip files.*

Window opens that allows us to select the file tree \ _4.1-9.zip

Once we have the package installed in the 3.3 library, we load them in R using the function

```
>library (rpart)
```

We introduce the instruction

```
>library (tree)
```

To check that they are installed correctly, we execute the function

```
search ()
```

When we perform the analysis with the tree package, and the function that we use to obtain the classification tree is *tree()*, which provides us with the classification tree defined by the arguments.

The first argument is the function we want, and it is the same as in rpart, that is, for which variable we will obtain the classification and depending on which other variables; in this case it will be a global rating based on the rest of qualifications, which is written as CG ~ T + L + P, since the global qualification is going to be obtained based on all the rest of the variables that we have, and it can also be written as CG ~.

The second argument, like rpart, specifies from which data frame the data are to be obtained; it is written data = sample or simply shows.

The third argument specifies the minimum cut we want, which is written as mincut = 1.

The fourth argument is minsize = 2, which we must put because we have very little data, and if we do not put it, it will not perform the calculations.

We assign the tree the variable name sorttree, so the instruction remains

```
>sorttree = tree (C. G ~., data = sample, mincut = 1, minsize = 2)
```

Once the instruction is executed, we enter the instruction

```
> sorttree
```

to see what we have obtained and see that the result is

```
node), split, n, deviance, yval, (yprob)
    * denotes terminal node


1) root 9 11.46 Ss (0.3333 0.6667)
  2) L: A,B 5 6.73 Ap (0.6000 0.4000)
    4) P: A,B 3 0.00 Ap (1.0000 0.0000) *
    5) P: C,D 2 0.00 Ss (0.0000 1.0000) *
  3) L: C,D 4 0.00 Ss (0.0000 1.0000) *
```

C. Supervised Classification Analysis Exercises Solved

This subsection has two parts. In the first part, a set of exercises solved in detail are presented to allow you to check if all the knowledge has been correctly acquired. The advice is to try to solve the exercises by yourself, and then to get the solution to check it with the proposed one by the book. This procedure will make this subsection truly useful for you. In the second part, the same exercises will be solved in R. As in the previous section, in this chapter, for reasons of space, section C will be applied only to the first of the techniques introduced in section A, the Decision Trees.

Hand-Made Exercises

1. For the data in the sample with the characteristics of 10 vehicles of four different types, 1. {B, 4, 5, Car}; 2. {A, 2, 2, Motorbike}; 3. {N, 2, 1, Bicycle}; 4. {B, 6, 4, Truck}; 5. {B, 4, 6, Car}; 6. {B, 4, 4, Car}; 7. {N, 2, 2, Bicycle}; 8. {B, 2, 1, Motorbike}; 9. {B, 6, 2, Truck}; 10.{N, 2, 1, Bicycle}, the elements of each event are: {LicenseType, NumberWheels, NumberPass, VehicleType }. It must be done first manually, using Hunt's algorithm and the CART algorithm, that is, it

must be obtained using the Gain of information measure and through the measure Gini impurity, sorting function, and the classification exercise of the type of vehicle taking into account the other characteristics.

To solve the exercise applying the Hunts method, the following steps are applied:

First, the exercise is going to be solved manually because the Hunt algorithm and the CART algorithm are going to be applied, and consequently, the measure of impurity will be Gini. The classifier event is Type of Vehicle.

How the classifier is Type of Vehicle, we have:

Classifier = Type of vehicle, which has four classes, c= 4, and the Type of Vehicle classes are {Car=3, Motorbike=2, Bicycle=3, Truck=2}

In Step A, we calculate the impurity for the initial father node:

$$\text{Gin}(f) = 1 - \sum_{i=0}^{c-1} {f_{ip}}^2; c = 4 \rightarrow = 1 - \sum_{i=0}^{c-3} {f_{ip}}^2 =$$
$$= 1 - \left(\left(\frac{3}{10}\right)^2 + \left(\frac{2}{10}\right)^2 + \left(\frac{3}{10}\right)^2 + \left(\frac{2}{10}\right)^2 \right) = 0,74$$

Then, we calculate the impurities of the nonclassifying elementary events to see which of them provides the greatest information gain; all that characteristics, that cannot be type of vehicle, because that is the classifier, are analysed. Taking this into account, Type of License is selected for the first node.

If the Type of License is taken, the following binary division tree of the sample events can be established (Fig. 12).

Once the division is established in the two nodes, we calculate the impurity of each one of them using the Gini:

Node 1. The types of cards A and B have been grouped in this node, and there are 7 events in the sample, 1, 2, 4, 5, 6, 8, and 9. For each of the classes, there are 3 cars, corresponding to events 1, 5, and 6; 2 motorcycles, corresponding to events 2 and 8; and 2 trucks, corresponding to events 4 and 9. In addition, there is no bicycle. Consequently, the Gini of this node is:

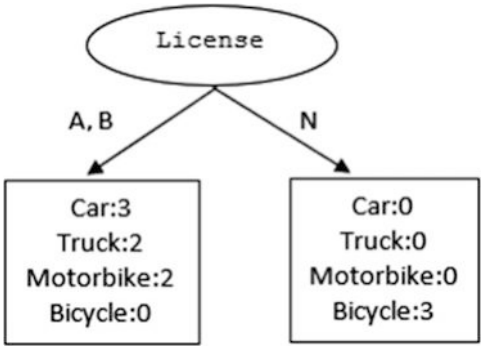


Fig. 12 First division with license

$$\text{Gin}(1) = 1 - \sum_{i=0}^{c-3} {f_{i1}}^2 = 1 - \left(\left(\frac{3}{7}\right)^2 + \left(\frac{2}{7}\right)^2 + \left(\frac{2}{7}\right)^2 + \left(\frac{0}{7}\right)^2 \right) = 0,65$$

Node 2. For the second node, not having a license has been taken, and a classification of bicycles is obtained. You have events 3, 7, and 10, all of which are bicycles, and you do not have any other type of vehicle. Consequently, before calculating its impurity, it will be zero:

$$\text{Gin}(2) = 1 - \sum_{i=0}^{c-3} {f_{i1}}^2 = 1 - \left(\left(\frac{0}{3}\right)^2 + \left(\frac{0}{3}\right)^2 + \left(\frac{0}{3}\right)^2 + \left(\frac{3}{3}\right)^2 \right) = 0$$

Once the impurities of the parent and child nodes have been obtained, as seen above, the information gain is equal to the impurity of the parent node minus the weighted average of the impurities of the child nodes. We first calculate the weighted average of the impurities of the child nodes: As we have seen above, it is the number of events associated with the child node 1, which in this case is 7, and it is the number of events associated with the child node 2, which in this case are 3, and N is the total number of events in the parent node, which in this case is 10, so the weighted mean of impurity of the child nodes is:

$$\sum_{j=1}^{k} \frac{N(n_j)}{N} I(n_j) = \frac{7}{10}.0.65 + \frac{3}{10}.0 = 0.46$$

Since the impurity of the parent node is 0.375, the information gain performing the first division with the elementary event Theory is:

$$\Delta_I = I_{\text{father}} - \sum_{j=1}^{2} \frac{N(n_j)}{N} I(n_j) = 0,74 - 0.46 = 0.28$$

There are other ways to make a binary division of the events over the possible values of the characteristic Type of License: A-BN, B-AN, NA-B,... In all of them information gain is lower. In most of them, because the Gini of none nodes is 0, and in those in which the value is 0, the other node has an impurity higher than the calculated one.

Before we analyse another characteristic, let us see what happens if a nonbinary analysis is performed (Fig. 13).

How can we see in the figure, it cannot be done because Motorbike would be classified in two different terminal nodes for this step.

Now Wheels are going to be analysed.

Before starting to calculate the impurities, we analyse which binary division gives us the smallest impurity in the sum of the child nodes and therefore the greatest information gain, and it will be the one that gives us a classification in a node and is

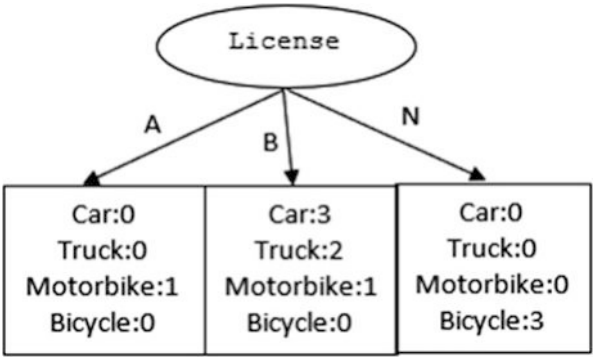


Fig. 13 First nonbinary division with license

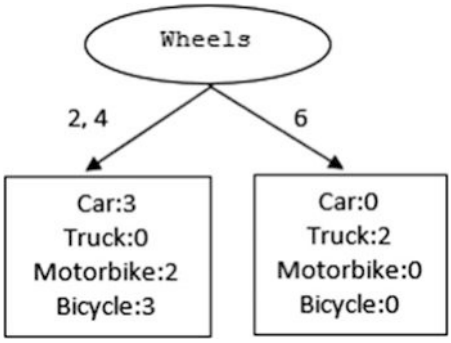


Fig. 14 First division with wheels

2,4 - 6[31] with which we have the following binary division tree of the sample events (Fig. 14).

Once the division is established in the two nodes, we calculate the impurity of each one of them using the Gini:

Node 1. The number of wheels 2 and 4 have been grouped in this node, and there are 8 events in the sample, 1, 2, 3, 5, 6, 7, 8, and 10. For each of the classes, there are 3 cars, corresponding to events 1, 5, and 6; 2 motorcycles, corresponding to events 2 and 8; and 3 bicycles, corresponding to events 3, 7, and 10. In addition, there is no truck. Consequently, the Gini of this node is:

$$\text{Gin}(1) = 1 - \sum_{i=0}^{c-3} {f_{i1}}^2 = 1 - \left(\left(\frac{3}{8}\right)^2 + \left(\frac{0}{8}\right)^2 + \left(\frac{2}{8}\right)^2 + \left(\frac{3}{8}\right)^2 \right) = 0,66$$

Node 2. For the second node, having 6 wheels has been taken, and a classification of the trucks is obtained. You have events 4 and 9, all of them trucks, and you do not

[31] It is left to the reader to verify that other binary divisions would provide less information gain.

have any other type of vehicle. Consequently, before calculating its impurity, it will be zero:

$$\text{Gin}(2) = 1 - \sum_{i=0}^{c-3} {f_{i1}}^2 = 1 - \left(\left(\frac{0}{2}\right)^2 + \left(\frac{2}{2}\right)^2 + \left(\frac{0}{2}\right)^2 + \left(\frac{0}{2}\right)^2 \right) = 0$$

Once the impurities of the parent and child nodes have been obtained, as seen above, the information gain is equal to the impurity of the parent node minus the weighted average of the impurities of the child nodes. We first calculate the weighted average of the impurities of the child nodes: As we have seen above, it is the number of events associated with child node 1, which in this case is 8, and it is the number of events associated with child node 2, which in this case is 2, and N is the total number of events in the parent node, which in this case is 10, so the weighted mean of impurity of the child nodes is:

$$\sum_{j=1}^{k} \frac{N(n_j)}{N} I(n_j) = \frac{8}{10}.0.65 + \frac{2}{10}.0 = 0.53$$

Since the impurity of the parent node is 0.48, the information gain performing the first division with the elementary event Theory is:

$$\Delta_I = I_{\text{father}} - \sum_{j=1}^{2} \frac{N(n_j)}{N} I(n_j) = 0,74 - 0.53 = 0.21$$

This information gain is lower than the previous information gain that we have for Type of License, that is, $\Delta_I = 0.28$ because there are more zeros in the child nodes in this Type License division and the impurity of the child nodes is lower. For that reason, Type License is selected for the first node against wheels.

As was done for License, a nonbinary division for wheels is analysed, for which we first perform a graphic partition.

As seen in the figure, if we make a nonbinary partition, two elementary events, cars and trucks, can be classified, so this partition would present a greater information gain. We would only have to calculate the Gini of Node 1 because the other two would be 0. We do it:

$$\text{Gin}(1) = 1 - \sum_{i=0}^{c-3} {f_{i1}}^2 = 1 - \left(\left(\frac{0}{5}\right)^2 + \left(\frac{0}{5}\right)^2 + \left(\frac{2}{5}\right)^2 + \left(\frac{3}{5}\right)^2 \right) = 0,52$$

Taking this into account, the weighted mean impurity of the child nodes is

$$\sum_{j=1}^{k} \frac{N(n_j)}{N} I(n_j) = \frac{5}{10}.0.52 + \frac{3}{10}.0 + \frac{2}{10}.0 = 0.26$$

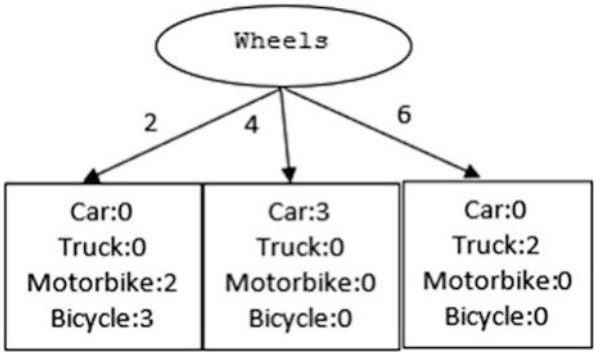


Fig. 15 First nonbinary division with wheels

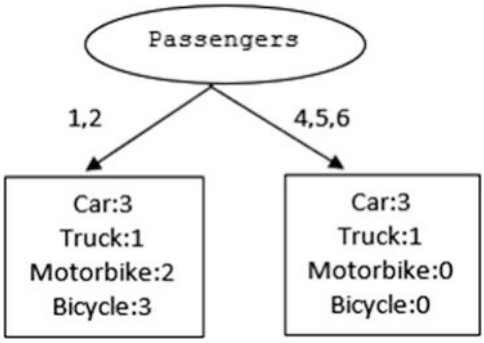


Fig. 16 Division with passengers

Since the impurity of the parent node is 0.48, the information gain performing the first division with the elementary event Theory is:

$$\Delta_I = I_{\text{father}} - \sum_{j=1}^{2} \frac{N(n_j)}{N} I(n_j) = 0,74 - 0.26 = 0.48$$

That is, a much higher information gain than those calculated thus far, so if we allowed nonbinary partitions, this would be the optimal one of those calculated thus far (Fig. 15).

Now, passengers will be analysed.

As in the previous case, before starting to calculate the impurities, we analyse which binary division gives us the least impurity in the sum of the child nodes and therefore the greatest information gain, and we see that there is no[32] binary nor nonbinary division that allows classification because elements of the same class always remain in different child nodes. As an example, one of the possible division trees is presented (Fig. 16).

Therefore, the elementary event or characteristic Number of Places could never be taken as a first-level classifier. Consequently, for the two possible options for the

[32] The verification of this statement is left to the reader.

initial node, the characteristic license is selected, although if multiple division would be considered, wheels, in its multiple division could have been chosen, which will be done later. If License is taken, the *N* brand is already classified.

Now, the second node is selected, which is the first intermediate node. We calculate the impurity of the parent node, which has changed because events 3, 7, and 10 are already classified, and now we only have 7 events:

{B, 4, 5, Car}
{A, 2, 2, Moto}
{N, 2, 1, Bicycle \} Classified
{B, 6, 4, Truck}
{B, 4, 6, Car}
{B, 4, 4, Car}
{N, 2, 2, Bicycle} Classified
{B, 2, 1, Moto}
{B, 6, 2, Truck}
{N, 2, 1, Bicycle} Classified

Consequently, we have 3 cars, 2 motorbikes, and 2 trucks, so the Gini of the impurity of the father node is calculated again because it has changed, now with only the no classified events, that are those ones in branch A, B, because the three in the *N* branch has already been classified. For that reason, there are seven events now. Consequently, the Gini of the father node is:

$$\text{Gin}(f) = 1 - \sum_{i=0}^{c-1} {f_{ip}}^2; c = 4 \rightarrow = 1 - \sum_{i=0}^{c-3} {f_{ip}}^2 =$$
$$= 1 - \left(\left(\frac{3}{7}\right)^2 + \left(\frac{2}{7}\right)^2 + \left(\frac{2}{7}\right)^2 \right) = 0,65$$

The first characteristic analysed is wheels. The division tree is (Fig. 17):

Once the division is established in the two nodes, we calculate the impurity of each one of them using the Gini:

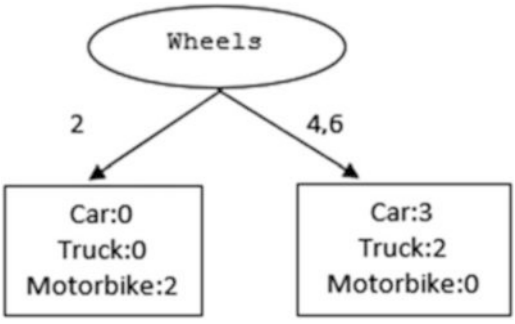


Fig. 17 Division with wheels

Node 1. The number of wheels 2 has been grouped in this node, and there are 2 events in the sample corresponding to events 2 and 8, motorcycles. In addition, you do not have a car or truck. Consequently, the Gini of this node is 0:

$$\text{Gin}(1) = 0$$

Node 2. For the second node, having 4 or 6 wheels has been taken, and a classification of cars and trucks is obtained. There are events 4 and 9, all of trucks, and events 1, 5, and 6, corresponding to cars, and 3 bicycles, corresponding to events 3, 7, and 10. Consequently, the Gini of this node is:

$$\text{Gin}(1) = 1 - \sum_{i=0}^{c-2} {f_{i1}}^2 = 1 - \left(\left(\frac{3}{5}\right)^2 + \left(\frac{2}{5}\right)^2 + \left(\frac{0}{5}\right)^2 \right) = 0,48$$

Once the impurities of the parent and child nodes have been obtained, as seen above, the information gain is equal to the impurity of the parent node minus the weighted average of the impurities of the child nodes. We first calculate the weighted average of the impurities of the child nodes: As we have seen above, it is the number of events associated with child node 1, which in this case is 2, and it is the number of events associated with child node 2, which in this case is 5, and N is the total number of events in the parent node, which in this case is 7, so the weighted mean of impurity of the child nodes is:

$$\sum_{j=1}^{k} \frac{N(n_j)}{N} I(n_j) = \frac{2}{7}.0 + \frac{5}{7}.0.48 = 0.34$$

Since the impurity of the parent node is 0.65, the information gain performing the first division with the elementary event Theory is:

$$\Delta_I = I_{father} - \sum_{j=1}^{2} \frac{N(n_j)}{N} I(n_j) = 0,65 - 0.34 = 0.31$$

The information gain for this division is the same as that for 2,4 - 6; for that reason, 2 - 4,6 is chosen.

Now Passengers is analysed. One of the possible binary divisions is analysed, 2 - 4,6. The division tree for this case is (Fig. 18):

Once the division is established in the two nodes, we calculate the impurity of each one of them using the Gini:

Node 1. With 1 or 2 seats, we have a truck, event 9, and two motorcycles, events 2 and 8, and no car\footnote {We put 0 directly in the sum of the car in the summation}. Consequently, the Gini of this node is:

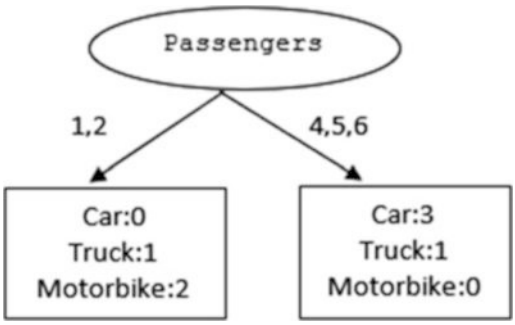


Fig. 18 Division with passengers

$$\text{Gin}(1) = 1 - \sum_{i=0}^{c-1} {f_{i1}}^2 =$$
$$= 1 - \left(\left(\frac{1}{3}\right)^2 + \left(\frac{2}{3}\right)^2 \right) = 1 - (0.11 + 0.44) = 0,45$$

Node 2. For the second node, having 4, 5 or 6 seats have been taken. You have events 1, 5, and 6, all cars, and event 4, a truck, and you have no motorcycles. Consequently, the impurity will be:

$$Gin(2) = 1 - \sum_{i=0}^{c-1} {f_{i1}}^2 =$$
$$= 1 - \left(\left(\frac{3}{4}\right)^2 + \left(\frac{1}{4}\right)^2 \right) = 1 - (0.56 + 0.06) = 0,38$$

Once the impurities of the parent and child nodes have been obtained, as seen above, the information gain is equal to the impurity of the parent node minus the weighted average of the impurities of the child nodes. We first calculate the weighted average of the impurities of the child nodes: As we have seen above, it is the number of events associated with the child node 1, which in this case is 3, and it is the number of events associated with the child node 2, which in this case is 4, and N is the total number of events in the parent node, which in this case is 7, so the weighted mean of impurity of the child nodes is:

$$\sum_{j=1}^{k} \frac{N(n_j)}{N} I(n_j) = \frac{3}{7}.0.45 + \frac{4}{7}.0.38 = 0.43$$

And since the impurity of the parent node is 0.65, the information gain performing the first division with the elementary event Theory is:

$$\Delta_I = I_{\text{father}} - \sum_{j=1}^{2} \frac{N(n_j)}{N} I(n_j) = 0,65 - 0.43 = 0.22$$

The information gain $\Delta_I = 0.22$ is less than what we have for Wheels, which was $\Delta_I = 0.31$, but it is not only in this division, it is in all the binary divisions that we can do, since in none of them would get a classifier node, and the same would happen for the multiple.[33] Therefore, in the second level, we choose Wheels.

Before going on to analyse the second level, we could reanalyse the Type of Licensed as a candidate for this level because the same classifying elementary events can be used at different levels. However, the same thing would happen as with squares; in this case, it would be a motorcycle that would be in two different nodes, and we could not obtain a classifier node. Therefore, the second intermediate node is wheels, and the classified events are now:

{B, 4, 5, Car }
{A, 2, 2, Moto} Classified
{N, 2, 1, Bicycle} Classified
{B, 6, 4, Truck \}
{B, 4, 6, Car \}
{B, 4, 4, Car \}
{N, 2, 2, Bicycle \} Classified
{B, 2, 1, Moto \} Classified
{B, 6, 2, Truck \}
{N, 2, 1, Bicycle \} Classified

Now, the second intermediate node will be selected.

They must be analysed only for the unclassified events, those that are in the branch with values 4 and 6 because the branch with value 2 is already classified and they are motorbikes. Therefore, in the parent node of the second level, we only have events 1, 4, 5, 6, and 9, three cars, and two trucks. Consequently, we recalculate the Gini of the parent node:

$$\text{Gin}(f) = 1 - \sum_{i=0}^{c-1} {f_{ip}}^2 = 1 - \left(\left(\frac{3}{5}\right)^2 + \left(\frac{2}{5}\right)^2 \right) = 0,48$$

V: Classifier; {Car=3, Truck=2}

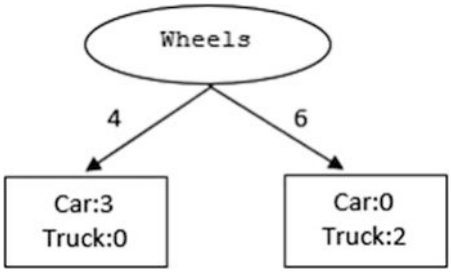
To look for the possible classifying event in this second level, as we already know elementary events well, we will start with Wheels. If we take Wheels, the division tree remains (Fig. 19).

Once the division has been established in the two nodes, we calculate the impurity of each one of them using the Gini:

Node 1. The number of wheels 4 has been grouped in this node, and there are 3 events in the sample corresponding to events 1, 5, and 6, cars. In addition, you do not have a truck. Consequently, the Gini of this node is 0:

[33] The reader can check it.

Fig. 19 Wheels divison



$$\text{Gin}(1) = 1 - \sum_{i=0}^{c-2} {f_{i1}}^2 = 1 - \left(\left(\frac{3}{3} \right)^2 \right) = 1 - 1 = 0$$

Node 2. The number of wheels 6 has been grouped in this node, and there are 2 events in the sample corresponding to events 4 and 9, truck. In addition, you do not have a car. Consequently, the Gini of this node is 0:

$$\text{Gin}(2) = 1 - \sum_{i=0}^{c-2} {f_{i1}}^2 = 1 - \left(\left(\frac{2}{2} \right)^2 \right) = 1 - 1 = 0$$

Once the impurities of the parent and child nodes have been obtained, as seen above, the information gain is equal to the impurity of the parent node minus the weighted average of the impurities of the child nodes. In this case, the weighted average of the impurities of the child nodes will be zero since they are both zero:

$$\sum_{j=1}^{k} \frac{N(n_j)}{N} I(n_j) = \frac{3}{5}.0 + \frac{2}{5}.0 = 0$$

Since the impurity of the parent node is 0, the information gain performing the first division with the elementary event Theory is:

$$\Delta_I = I_{\text{father}} - \sum_{j=1}^{2} \frac{N(n_j)}{N} I(n_j) = 0,48 - 0 = 0$$

If Passengers or License is analysed, their information gain is worse and does not allow the finalization of the classification; for that reason, this second intermediate node can be only a terminal node for the whole classification model only if wheels are selected, and in this case, the classification model is finished. Consequently, the qualifier model using binary divisions is (Fig. 20):

If the multiple division is used in the first level, with wheels in that level, cars and trucks would be classified directly and using a license in the second level, for which

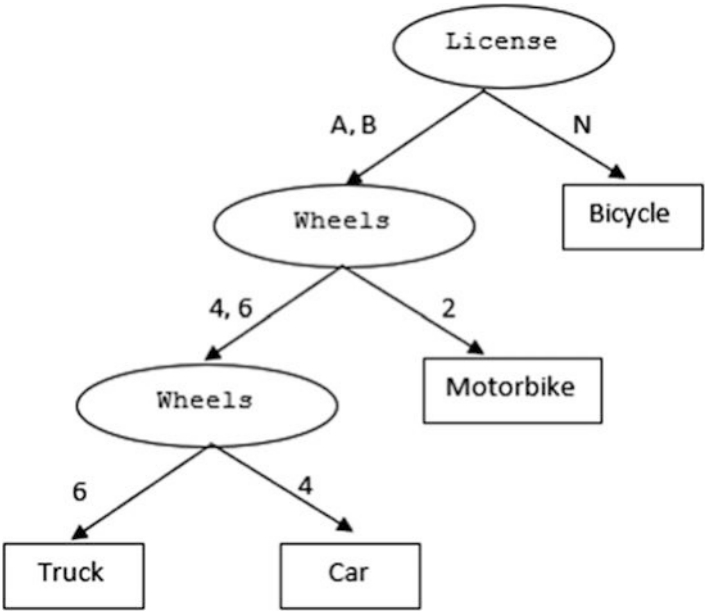


Fig. 20 Final binary classification

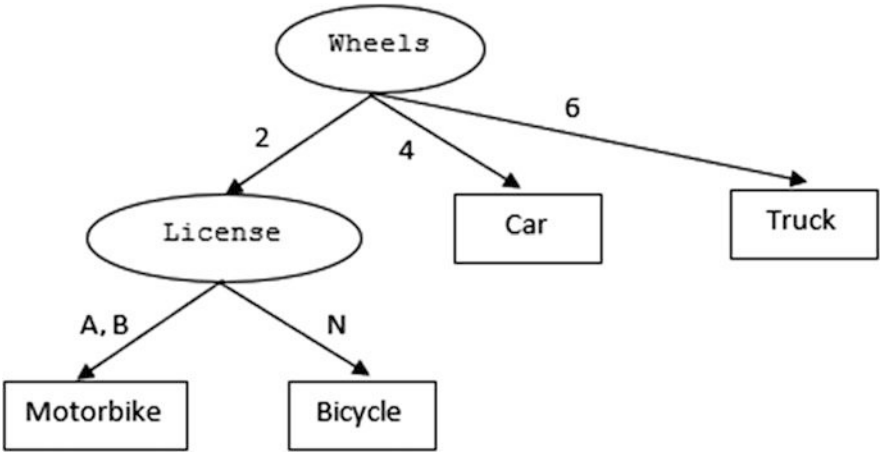


Fig. 21 Final non-binary classification

it would be the only possible qualifier[34] all the elementary events would be classified, as seen in the picture (Fig. 21).

The information gain for this division is the same as that for 2,4 - 6; for that reason, 2 - 4,6 is chosen.

[34] The reader can check it.

Exercises Solved in R

In this subsection, the previous exercises will be solved using the R software.

Once we have the *rpart* package loaded by default, we begin to solve the supervised classification analysis problem with R.

1. For the data in the sample with the characteristics of 10 vehicles of four different types, 1. {B, 4, 5, Car}; 2. {A, 2, 2, Moto}; 3. {N, 2, 1, Bicycle}; 4. {B, 6, 4, Truck}; 5. {B, 4, 6, Car}; 6. {B, 4, 4, Car}; 7. {N, 2, 2, Bicycle}; 8. {B, 2, 1, Moto}; 9. {B, 6, 2, Truck}; 10. {N, 2, 1, Bicycle}, the elements of each event are {LicenseType, NumberWheels, NumberPass, VehicleType}. It must be done first manually using Hunt's algorithm and the CART algorithm, that is, it must be obtained using the Gain of information measure and through the measure Gini impurity, sorting function, and the classification exercise of the type of vehicle taking into account the other characteristics. Solve the problem using R.

For solving the exercise with R, the following procedure is applied: As we already have the *rpart* and *tree* packages loaded, we begin to solve exercise directly with their use. The first thing we have to do is enter the values of the events of the sample we are working with in R. We remember that the events must be represented as a data frame, but as in exercise 1, to introduce the data in R, we introduce it as a matrix and then we convert the matrix into a data frame. We introduce the matrix as follows: In each column, we will represent the values of the different elementary events with which we work {LisenseType, NumberWheels, PassengerNumber, VehicleType}, although to facilitate the treatment of the data, we will call them {L, R, P, V }, and in each row, we will call the values of a given event (observation). According to this, the matrices corresponding to the observed sample are as follows: 1. {B, 4, 5, Car}; 2. {A, 2, 2, Moto}; 3. {N, 2, 1, Bicycle}; 4. {B, 6, 4, Truck}; 5. {B, 4, 6, Car}; 6. {B, 4, 4, Car}; 7. {N, 2, 2, Bicycle}; 8. {B, 2, 1, Moto}; 9. {B, 6, 2, Truck}; 10. {N, 2, 1, Bicycle}.

$$\begin{pmatrix} & L & W & P & V \\ e_1 & B & 4 & 5 & C \\ e_2 & A & 2 & 2 & M \\ e_3 & N & 2 & 1 & B \\ e_4 & B & 6 & 4 & T \\ e_5 & B & 4 & 6 & C \\ e_6 & B & 4 & 4 & C \\ e_7 & N & 2 & 2 & B \\ e_8 & B & 2 & 1 & M \\ e_9 & B & 6 & 2 & T \\ e_{10} & N & 2 & 1 & B \end{pmatrix}$$

As in the previous cases, we introduce the data to be analysed through a.txt file; for this, we write the matrix in a txt document with the name "vehicles.txt". We will assign the data from the file to an array that we will call qualifications through the instruction

```
>vehicles = read.table ("vehicles.txt")
```

If we are using RStudio, we only need to go to the card files in the corresponding window and navigate to the directory in which the file vehicles are and select that directory as the working directory. Set at working directory.

Next, we convert the matrix into a data frame that we call the sample with the instruction

```
>sample = data.frame (cvehicles)
```

Once we have the data frame, we perform the classification analysis first using the *rpart()* functions as in the previous exercise. For the *rpart ()* function, we know that the instruction is

```
>classification = rpart (V ~., data = sample, method = "class", minsplit
= 1)
```

Once the instruction is executed, we enter the instruction

```
>classification
```

to see what we have obtained and see that the result is

```
n= 10


node), split, n, loss, yval, (yprob)
    * denotes terminal node


1) root 10 7 Bike (0.3000000 0.2000000 0.3000000 0.2000000)
2) C=N 3 0 Bike (1.0000000 0.0000000 0.0000000 0.0000000) *
3) C=A,B 7 4 Car (0.0000000 0.2857143 0.4285714 0.2857143)
6) R>=3 5 2 Car (0.0000000 0.4000000 0.6000000 0.0000000)
12) R>=5 2 0 Truck (0.0000000 1.0000000 0.0000000 0.0000000) *
13) R< 5 3 0 Car (0.0000000 0.0000000 1.0000000 0.0000000) *
7) R< 3 2 0 Moto (0.0000000 0.0000000 0.0000000 1.0000000) *
```

Association

In this final chapter, we are going to see the theoretical foundations of events *Association*[1] analysis and the main techniques used to carry it out. As in all the previous chapters, it is structured in three subsections.

Subsection A introduces, in a theoretical and, at the same time, practical way, all the basic theoretical knowledge related to the association analysis, that is, the concepts and techniques that allow us to perform the analysis, from the association of events composed of a single ementary event to the association of events formed by more than one elementary event.

Subsection B presents the computer-based solution. The same association analysis solved as examples in subsection A is solved with the use of the R computational environment. The packages needed to carry out these computational solutions are also introduced.

Section C will consist of a set of statements of exercises about association in which detailed solutions can also be found in this section of the chapter.[2]

A. Theory

This first subsection of this chapter is structured in 4 subsubsections: 1. Introduction. 2. Analysis of the Association of Events Composed of a Single Elementary Event. 3. Analysis of the Association of Events Composed of More Than a Single Elementary Event. 4. Apriori Algorithm.

[1] Association analysis is also called in certain texts, especially statistics, as dependency analysis; and in data science texts, such as pattern search. The Dictionary of the Spanish Language [1] defines dependence as: "(//3. F. Relation of origin or connection."

[2] We repeat again here that it is very important in order to obtain the best results for the learning process throughout the use of this book, that the reader tries to solve the exercises by himself before seeing their solutions, and that only once solved check if the obtained solutions are correct.

J. J. Cuadrado-Gallego, Y. Demchenko, *Data Analytics*,
https://doi.org/10.1007/978-3-031-39129-3_7

Introduction

As we know, from what was studied in chapter "Probability", Probability, every random experiment has an associated sample space $E = \{e_i. . .e_n\}$, which is the set of all elementary events that can occur in the experiment. We also know that, from it, an event space, or set of parts of E, P (E), can be defined as one whose elements are all possible subsets of E. Based on this knowledge, we study what the association analysis of disjoint events consists of.

The studies of analysis of association of disjoint events try to discover if the joint appearance of some events exceeds a threshold value for, at least,[3] two measures that are complementary to each other, the *support* and the *confidence*. In addition, other measures are also used to complement the information on the degree of association between these events, such as the *contingency* and the *correlation* measures, and others that have also been defined, some of which will be introduced in this chapter.

First, in this chapter, we will define all these measures and their application in the case of the analysis of the association of two events composed of a single elementary event, and then, we will define the measures and their application to the analysis of the association of events formed by more than one elementary event. Although this way of presenting the subject entails the repetition of the exposition of some concepts, we prefer to do it this way because we continue to apply the principle that guides the entire conception of this book, which is based on the fact that we think that the best way to understand the concepts in depth is to present a practical example immediately after presenting the theoretical bases of the concept, although it may happen, as in this case, that this implies repeating the theoretical definitions of some concepts.

The association analysis can give us a result that there is a relationship between both disjoint events, in which case it will be said that there is an association or dependency between both values of the characteristics. The other possible result is that there is no relationship between the two of the observed values of both characteristics, in which case it will be said that there is independence between them. In the event that there is an association or dependence between the two disjoint events, it can be attractive when the frequency of occurrence of one of the events increases as the frequency of occurrence of the dependent event increases or repulsion when the frequency of one of the events increases as the frequency of the dependent event decreases.

In this introductory subsubsection on the analysis of association of disjoint events, it is also very important to indicate the fact that in association studies it is essential to bear in mind the fact that elementary events are not symmetric or equiprobable and consequently to be able to establish the probability of appearance of the elements of P (E), it is essential to have a sample of events.

Once the concepts of support, confidence, contingency, and correlation have been seen, we can establish what the purpose of an association analysis is going to be. For

[3] There are others as the *Lift*.

instance, if support and confidence are used, given a sample space E, its set P (E), and some set support and confidence threshold values arbitrarily based on the interests of the study, an association study will try to establish, from a sample of available events, which sets of P (E) can be considered associated, that is, whenever one of the sets is observed, the other will also be observed with a probability greater than or equal to that established by the threshold values. To consider the two associated sets, minimum thresholds for the support and confidence values will be arbitrarily established in each case.

Analysis of the Association of Events Composed by a Single Elementary Event

We will begin to see in detail the definition and how the measures indicated above are calculated and used to perform association analysis. For this, we start with the simplest type of study, which is the one constituted by the association analysis of two event disjoints, each formed by a single elementary event. We will see the association coefficients: Support, Confidence, Contingence, and Correlation.

To obtain these events, we will start from the simplest case, which is that of a sample space whose elementary events can be grouped into two subgroups, each consisting of k mutually exclusive elementary events. Once we have seen this case, we will increase the complexity of the problem starting by taking a sample space made up of more than two subsets with mutually exclusive elementary events.

Starting from said sample space, with subsets formed by exclusive elements, the set P (E) will be reduced to sets of one and two dimensions, regardless of the number of elementary events that constitute E.

In this first example, we use the following sample space: E = {Job, No Job, Pass, No Pass}, which refers to the students of Data Science Fundamentals, and Job means that the student combined studies and work; No Job means that the student dedicated his/her full time to study; Pass, that the student passed the course; and No Pass, that the student did not pass the course. Since Data Science Fundamentals is a four-course subject, there are many students that combine Job and Studies, and the final goal of the analysis is to know if the fact that they combine both things is associated with Pass, or Not Pass, the course. As has been said in the theoretical description, we will begin to see the association using sample spaces composed of subsets formed by exclusive elementary events, and in this case, there are two subsets with exclusive elements that they are: {Job, No Job} and {Pass, No Pass}.

Starting from the sample space E = {Job, No Job, Pass, No Pass}, established in the previous exercise, and taking into account that we have the two subsets with exclusive elements {Job, No Job} and {Pass, No Pass}, the set P (E) will be formed by the following sets: P (E) = {∅, {Job}, {No Job}, {Pass}, {No Pass}, {Job, Pass}, {Job, Not Pass}, {No Job, Pass}, {No Job, Pass}}. From here on, the association analysis that we are going to perform will be done only for those events formed by a

single elementary event, which are not exclusive, that is, they have a union event within the set P(E); that is, we can study the association of Job with Pass but not of Job with No Job.

Association of two events composed of a single elementary event. As mentioned in the introduction, to carry out an association analysis, it is essential to have a sample that allows us to calculate the values of the measures used to determine the degree of association. In this case, the sample we are going to have is the ten students of the subject Data Science Fundamentals: {1 {No Job, No Pass}, 2 {No Job, Pass}, 3 {Job, No Pass}, 4 {No Job, No Pass}, 5 {Job, Pass}, 6 {Job, Pass}, 7 {Job, Pass}, 8 {No Job, No Pass}, 9 {Job, Pass}, 10 {Job, Pass}}.

Once we have E, P (E), and a sample, we are going to begin the association analysis by calculating the first two measures, which, as we have seen in the introduction, are *support* and *confidence*. We are going to see the definition of both measures, and we are going to continue with the example to see, in a practical way, how they are applied.

Support

To be able to compare with the threshold value referred to in the previous paragraph, the degree of association[4] between various events is established as the calculation of the classical probability[5] of appearance of the elements of the set P (E) that encompass such events. This measure is called *Support*, s, which is more formally defined as:

$$\forall \{\mathrm{A}_i\}_{i=1}^{\infty} \subset \mathrm{P}\ (\mathrm{E})\ \mathrm{con}\ \mathrm{A}_i \cap \mathrm{A}_j = \varnothing \forall i \neq j, s : \mathrm{P}\ \text{€} \rightarrow \mathbb{R}^+ / s\ \left(\mathrm{A}_i \cup \mathrm{A}_j\right) = \frac{n_{\mathrm{A}_i \cup \mathrm{A}_j}}{n_T}$$

The acceptance threshold of support is not fixed but will be arbitrarily set a priori and will depend on the objectives of the study.

The support is interesting since it gives us a measure of how frequent the association we are looking for is because although the association of the elements was very strong, if this association were observed very seldom, it might not be of interest to study it.

Once we have defined the sample that allows us to establish the probability of appearance of the different events, we calculate the support of the association of the disjoint events $\mathrm{A}_1 = \{\mathrm{Job}\}$ and $\mathrm{A}_2 = \{\mathrm{Pass}\}$. What we have to calculate is the classical probability of appearance of the set $A_1 \cup A_2 = \{\mathrm{Job}, \mathrm{Pass}\}$, as a set formed

[4] In order not to repeat throughout the text the terminology: "search for patterns or association of disjoint events," from here on we will only use "association of events" as it is the most widely used terminology.

[5] It is very important to bear in mind that the definition of probability that will be used in association studies will be the classic one and from here on, whenever we refer to probability, it will be the classic one.

only by these elements or within as a subset of any set of P(E) that contains, among others, said elements, but in this case, that possibility will not be given. As seen above, this probability is given by:

$$p\,(\mathrm{A}_1 \cup \mathrm{A}_2) = s\,(\mathrm{A}_1 \cup \mathrm{A}_2 + = \frac{n_{\mathrm{A}_1 \cup \mathrm{A}_2}}{n_T}$$

Examining the sample, {1 {No Job, No Pass}, 2 {No Job, Pass}, 3 {Job, No Pass}, 4 {No Job, No Pass}, 5 {Job, Pass}, 6 {Job, Pass}, 7 {Job, Pass}, 8 {No Job, No Pass}, 9 {Job, Pass}, 10 {Job, Pass}}, the number of elements that the set contains $A_1 \cup A_2 = \{\text{Job}, \text{Pass}\}$, are 5: 5 {Job, Pass}, 6 {Job, Pass}, 7 {Job, Pass}, 9 {Job, Pass}, and 10 {Job, Pass}, and the total number of events in the sample is equal to 10.

Consequently, the support of the association of events $\mathrm{A}_1 = \{\text{Job}\}$ and $\mathrm{A}_2 = \{\text{Pass}\}$ is:

$$s\,(\mathrm{A}_1 \cup \mathrm{A}_2) = \frac{n_{\mathrm{A}_1 \cup \mathrm{A}_2}}{n_T} = \frac{5}{10} = 0.5$$

Therefore, this association has 50% support. As it is a probability, the support moves in a range from 0 to 1. Once we have seen how it is calculated and what is the value of the support of the association {Job, Pass}, we are going to calculate the support of the rest of possible associations: {Job, No Pass}, with a support of:

$$s\,(\mathrm{A}_1 \cup \mathrm{A}_2) = \frac{n_{\mathrm{A}_1 \cup \mathrm{A}_2}}{n_T} = \frac{1}{10} = 0.1$$

and therefore of 10%; {No Job, Pass}, with a support of 1/10, and therefore a 10%; and {No Job, No Pass} with a support of 2/10 and therefore 20%.

If we had established 25% support to determine which associations we would continue to analyse, we would continue only with the association {Job, Pass} because it would be the only one that appears enough time to make its analysis interesting. However, 25% is a very low value to take it as valid to select which associations are of interest to continue analysing. The logical thing would be to select those that have a value above 75%.

Confidence

As mentioned in the previous paragraph, measuring only the probability, or frequency, of the appearance of the association studied in the available sample is not enough to establish the degree to which the events studied are associated. For this reason, the second measure referred to at the beginning of this chapter called *Confidence*, c, is defined, which will measure the classic probability of appearance of the association for a subset of the original sample composed only of the events in

which one of the two sets whose association is being analysed. The one that is the origin of the association, that is, the confidence that a second set appears when the first has appeared, will be measured. More formally, it is defined as:

$$\forall \{\mathrm{A}_i\}_{i=1}^{\infty} \subset \mathrm{P}\,(\mathrm{E}) \text{ con } \mathrm{A}_i \cap \mathrm{A}_j = \varnothing \forall i \neq j, c : \mathrm{P}\,(\mathrm{E}) \rightarrow \mathbb{R}^+ / c\,\left(\mathrm{A}_i \cup \mathrm{A}_j\right) = \frac{n_{\mathrm{A}_i \cup \mathrm{A}_j}}{n_{\mathrm{A}_i}}$$

However, in this case, the sense of the association means for the confidence calculations because the denominator, n_{A_i}, can change. For that reason, for confidence, we must calculate both senses of the association.

As was also said at the beginning of the subsubsection, as with the support, for trust, there will be an acceptance threshold, which will not be fixed but will also be arbitrarily set a priori and will depend on the objectives of the study. It is also important to realize that the confidence measure of an association, contrary to what happened with the support, is not the same, neither reversing the sense of the association, much less for all combinations of associations of disjoint sets. This results in the same union set.

Example of calculation of confidence of two events composed of a single elementary event. We calculate the confidence of the association of the selected disjoint events of the previous example. These are {Job, Pass}, {Job, Not Pass}, {No Job, Pass}, and {No Job, Pass}. What we have to calculate is the probability of appearance, for example, of the set {Job, Pass}, but unlike the previous example, we do not calculate it on the complete sample, {1 {No Job, No Pass}, 2 {No Job, Pass}, 3 {Job, No Pass}, 4 {No Job, No Pass}, 5 {Job, Pass}, 6 {Job, Pass}, 7 {Job, Pass}, 8 {No Job, No Pass}, 9 {Job, Pass}, 10 {Job, Pass}}, but taking a subset of it formed only by those events that contain or are equal to the set: $\mathrm{A}_1 = \{\text{Job}\}$. The reduced sample is composed of the six events: {3 {Job, No Pass}, 5 {Job, Pass}, 6 {Job, Pass}, 7 {Job, Pass}, 9 {Job, Pass}, 10 {Job, Pass}}. Therefore, to calculate the confidence we use the equation:

$$p\,(\mathrm{A}_1 \cup \mathrm{A}_2) = c\,(\mathrm{A}_1 \cup \mathrm{A}_2) = \frac{n_{\mathrm{A}_1 \cup \mathrm{A}_2}}{n_{\mathrm{A}_1}}$$

As we know from the previous example, the number of elements that the set contains $A_1 \cup A_2 = \{\text{Job}, \text{Pass}\}$, are 5: 5 {Job, Pass}, 6 {Job, Pass}, 7 {Job, Pass}, 9 {Job, Pass}, and 10 {Job, Pass}; and the total number of events in the reduced sample, from the previous paragraph, is $n_{\mathrm{A}_1} = 6$, consequently the confidence of the association of the events $\mathrm{A}_1 = \{\text{Job}\} \rightarrow \mathrm{A}_2 = \{\text{Pass}\}$

$$c\,(\mathrm{A}_1 \cup \mathrm{A}_2) = \frac{n_{\mathrm{A}_1 \cup \mathrm{A}_2}}{n_{\mathrm{A}_1}} = \frac{5}{6} = 0.83$$

Again, since it is a probability, the support moves in a range that goes from 0 to 1; therefore, this association would have an 83.3% confidence level.

Once we have seen how it is calculated and what is the value of the confidence of the association {Job, Pass}, we calculate the confidence of the rest of the possible associations: $A_1 = \{\text{Job}\} \rightarrow A_2 = \{\text{No Pass}\}$, with a confidence of

$$c\,(A_1 \cup A_2) = \frac{n_{A_1 \cup A_2}}{n_{A_1}} = \frac{1}{6} = 0.16$$

and therefore of 16.6%; $A_1 = \{\text{No Job}\} \rightarrow A_2 = \{\text{Pass}\}$, with a confidence of 1/4, and therefore a 25%; and $A_1 = \{\text{No Job}\} \rightarrow A_2 = \{\text{No Pass}\}$ with a confidence of 3/4 and therefore 75%.

If we had established a 75% confidence to determine which associations have enough confidence to be considered associated, we would considered associated only the associations $A_1 = \{\text{Job}\} \rightarrow A_2 = \{\text{Pass}\}\{\text{Job, Pass}\}$ and $A_1 = \{\text{No Job}\} \rightarrow A_2 = \{\text{No Pass}\}$ because it would be the only ones with a confidence equal or over 75%.

However, as we have seen in the theory, the sense of the association means for the confidence calculations because the denominator can change. For that reason, for confidence, we must calculate both senses of the association.

Now, we calculate the confidence of the reverse associations, and we start with $A_1 = \{\text{Pass}\} \rightarrow A_2 = \{\text{Job}\}$, that is,

$$c\,(A_1 \cup A_2) = \frac{n_{A_1 \cup A_2}}{n_{A_1}} = \frac{5}{6} = 0.83$$

In this case, the result does not change because, although in this case the reduced sample change is 2 {No Job, Pass}, 5 {Job, Pass}, 6 {Job, Pass}, 7 {Job, Pass}, 9 {Job, Pass}, 10 {Job, Pass}}, the n_{A_1} is the same as in the other sense, but we are going to see that this is not the same in all the cases. We calculate the support of the rest of the possible associations:

$A_1 = \{\text{No Pass}\} \rightarrow A_2 = \{\text{Job}\}$ has a confidence of

$$c(A_1 \cup A_2) = \frac{n_{A_1 \cup A_2}}{n_{A_1}} = \frac{1}{4} = 0.25$$

that, as we can see, it is very different from the obtained value for $A_1 = \{\text{Job}\} \rightarrow A_2 = \{\text{No Pass}\}$, which was 0.16.

For the rest $A_1 = \{\text{Pass}\} \rightarrow A_2 = \{\text{No Job}\}$, the confidence is $\frac{1}{6} = 0.16$; and for $A_1 = \{\text{No Pass}\} \rightarrow A_2 = \{\text{No Job}\}$, the confidence is $\frac{3}{4} = 0.75$

Those are also different from the previous ones.

Contingency

In addition, as mentioned at the beginning of this subsubsection, other measures are used to measure the degree of association between these events. Now we are

going to introduce a new one, the *contingency*. We are going to see its theoretical definition and how it is used through the practical application example that we are using.

In the definitions of support and confidence, the values of the acceptance threshold have been discussed, and in both cases, it has been said that they were arbitrarily set by whoever was performing the association analysis. Although this is correct, there is a value, called *Contingency*, that allows setting the value of the confidence threshold from the values of the events in the available sample, allowing elimination, if desired. The name "contingency" comes from the fact that the use of this value is closely linked to the use of contingency tables for the description of the data. In addition to serving to set the trust acceptance threshold, contingency can be used as a measure of association in itself.

The theoretical basis on which the contingency calculation rests is based on the fact that if two events A_1 and A_2 are formed by a single elemental event from a sample space such that both elemental events belong to two distinct subsets of the sample space and, consequently, they are not mutually exclusive; if they are independent, then the association confidence, or what is the same, of the appearance of A_1 when A_2 is given must be proportional to the relative frequency of appearance of A1 in the complete sample. Therefore, if $n_{A_1 \cup A_2}$ is the number of times that A_1 and A_2 appear together in the sample and n_{A_2} is the number of times that A_2 appears, the confidence of the association will be:

$$c\left(A_1 \cup A_2\right) = \frac{n_{A_1 \cup A_2}}{n_{A_2}}$$

The relative frequency of appearance of A_1 in the whole sample is:

$$fr\left(A_1\right) = \frac{n_{A_1}}{n_T}$$

and if A_1 and A_2 are independent, it must be verified that:

$$c\left(A_1 \cup A_2\right) = \frac{n_{A_1 \cup A_2}}{n_{A_2}} = \frac{n_{A_1}}{n_T} = fr\left(A_1\right)$$

Therefore, the threshold must be exceeded in the confidence of the association so that it can be said that it exists between the two events.

If we solve $n_{A_1 \cup A_2}$ for the equality above, we obtain the absolute frequency, also called the theoretical frequency or contingency, which the union set of the two events must have to be considered to establish that both events are associated. This frequency is:

$$n_{A_1 \cup A_2} = \frac{n_{A_2} . n_{A_1}}{n_T}$$

If to denote the absolute frequency, we substitute n for f; we call event A1 p and event A1 q; and we write the above equation with addends, we have that the contingency of independence for the two events f′ is:[6]

$$f'_{pq} = \frac{\sum_{j=1}^{n} f_{pj} \sum_{i=1}^{m} f_{iq}}{\sum_{\substack{i=1 \\ j=1}}^{\substack{i=m \\ j=n}} f_{ij}}$$

and, consequently, p and q will be dependent when it is verified:

$$f'_{pq} \gg \frac{\sum_{j=1}^{n} f_{pj} \sum_{i=1}^{m} f_{iq}}{\sum_{\substack{i=1 \\ j=1}}^{\substack{i=m \\ j=n}} f_{ij}}$$

There will be a dependence of repulsion in the case that

$$f_{pq} < \frac{\sum_{j=1}^{n} f_{pj} \sum_{i=1}^{m} f_{iq}}{\sum_{\substack{i=1 \\ j=1}}^{\substack{i=m \\ j=n}} f_{ij}} \rightarrow f_{pq} < f'_{pq}$$

and attraction in the case that

$$f_{pq} > \frac{\sum_{j=1}^{n} f_{pj} \sum_{i=1}^{m} f_{iq}}{\sum_{\substack{i=1 \\ j=1}}^{\substack{i=m \\ j=n}} f_{ij}} \rightarrow f_{pq} > f'_{pq}$$

Starting from the theoretical basis of the definition of contingency, which, as we have seen, is based on the definition of two subsets, formed by exclusive elementary events, within the sample space, we can define characteristics that encompass all the values of the exclusive events of each subset. In such a way that the first subset would define the first characteristic, whose possible values would be each of the exclusive events belonging to said subset, and the second subset would define the second characteristic, whose possible values would be each of the exclusive events belonging to said subset.

[6]The equation is written for mxn tables, that is, when the two subsets of the sample space are formed by n exclusive events, the first, and m exclusive events, the second.

We are going now to see an example of the use of the contingency to measure the association of two events composed of a single elementary event. In the previous examples, the support and confidence thresholds have been chosen arbitrarily, and we are now going to perform the analysis using contingency. The first thing we will do is to generate the contingency table of the elementary events treated with the sample {1 {No Job, No Pass}, 2 {No Job, Pass}, 3 {Job, No Pass}, 4 {No Job, No Pass}, 5 {Job, Pass}, 6 {Job, Pass}, 7 {Job, Pass}, 8 {No Job, No Pass}, 9 {Job, Pass}, 10 {Job, Pass}}. In each column, we write one of the two exclusive elementary events belonging to the first subset of the sample space, that is, A1 {Job} and A2 {No Job}. In each row, we write the other two exclusive elementary events, that is, B1 {Pass} and B2 {No Pass}, in each cell, we write the absolute frequency of appearance of the union set of both elementary events. The table is:

2 × 2 crosstab or contingency table

	A_1	A_2	Total
B_1	f_{11}	f_{12}	$f_{11}+f_{12}$
B_2	f_{21}	f_{22}	$f_{21}+f_{22}$
Total	$f_{11}+f_{21}$	$f_{12}+f_{22}$	$f_{11}+f_{12}+f_{21}+f_{22}$

Contingency table with the problem values

	Job	No Job	Total
Pass	5	1	6
No Pass	1	3	4
Total	6	4	10

If we recall what was seen in the theoretical description of the contingency, from observing the values in the table, it could be concluded that if the event composed of the only elementary event {Pass} was independent of the value of the event composed of the only event elemental {Job}, the relative frequency of appearance of {Pass}, that is, students that pass the course, should be kept constant when calculating the relative frequency of {Pass} when the event {Job} also occurs, that is, when the relative frequency of appearance of students that pass the course between students with jobs is calculated. This is:

$$\frac{f_{11}}{f_{11}+f_{21}} = \frac{f_{11}+f_{12}}{f_{11}+f_{12}+f_{21}+f_{22}}$$

Substituting the values, we have:

$$\frac{5}{5+1} = 0,83 \neq \frac{5+1}{5+1+1+3} = 0,6$$

Therefore, there is a dependency relationship between passing the course and having a job. Later, it will be seen that type.

Once these calculations have been carried out, it is very important to note that, when performing them, we have also obtained the value that, f*rom the contingency calculation, the confidence threshold should have* for the possible associations of the event {Pass} with the rest of events, {Job} or {No Job}, which should be 60%. That is, if trust had a higher value, there would be an association between the events.

If the above equation is taken and the value f_{11} is cleared, the theoretical frequency f'_{11} is obtained, which should have the union sequence {Pass, Job} for these values to be independent. It is given by the equation:

$$f'_{11} = \frac{(f_{11} + f_{21})(f_{11} + f_{12})}{f_{11} + f_{12} + f_{21} + f_{22}}$$

If the result obtained for the pair of values {Pass, Job} $f'_{11} = 1$ and $f_{11} = 2$ is observed, it can be concluded, as seen in the theoretical description of the contingency, that there is an association relationship between {Pass, Job}, and taking into account that $2>1$, that is, $f'_{11} < f_{11}$, it is concluded that it is attraction. This means that students with jobs are associated with or tend to Pass the course.

To reinforce the understanding of the concept of contingency, we now analyse the association of the events {No Pass} and {No Job}. In this case, we have:

$$\frac{f_{22}}{f_{12} + f_{22}} = \frac{f_{21} + f_{22}}{f_{11} + f_{12} + f_{21} + f_{22}}$$

Substituting the values, we have:

$$\frac{3}{1+3} = 0.75 \neq \frac{1+3}{5+1+1+3} = 0,4$$

Therefore, there is a strong attraction dependency relationship between No Pass and No Have a Job. This means that students without jobs often do not pass the course.

Having seen the case of a 2 × 2 contingency table, we are now going to see an example, based on the way to reach the university for a set on 25 students of a course of advanced data since, of a m × n contingency table. m will by 2 since we are going to take the students that have passed the course and those ones that have not passed the course, those are two exclusive elementary events; n will be 5, since we are going to take 5 different transportation ways to reach the university, Car, Bus, Train, Bike and Walking. The contingency table is:

Crosstabulation or contingency table

	Car	Bus	Train	Bike	Walking	Total
Pass	1	5	8	3	2	19
No Pass	5	0	0	1	0	6
Total	6	5	8	4	2	25

In this case, we are going to analyse the association between the events formed by the elementary events {Car} and {Pass}, that is, we are going to observe if the fact that a student passes the course is associated with reaching the university by car. To do this to the values a1 (Pass the course) and b1 (transport way by car) of said table, the theoretical descriptions seen in for the concept of contingency in mxn tables are applied, in this case 2 × 5, and the value is obtained for contingency:

$$f'_{11} = \frac{\sum_{j=1}^{5} f_{1j} \sum_{i=1}^{2} f_{i1}}{\sum_{\substack{i=1 \\ j=1}}^{5} f_{ij}} = \frac{(1+5+8+3+2)(1+5)}{6+5+8+4+2} = 4.56$$

As $f_{11} = 1$, this implies that $f_{11} < f'_{11}$, and therefore, there is a repulsion association between passing the course and reaching the university by car since, as happened for the 2 × 2 tables, for the mxn tables, it is verified that if $f_{pq} < f'_{pq}$, you have a dependence on repulsion, and if $f_{pq} > f'_{pq}$, you have a dependence on attraction. Consequently, the analysis carried out means that students who arrive by car tend to not pass the course.

If we analyse the rest of pairs of variables, we have:

To see other examples, we will analyse whether some of the transportation methods are associated with passing or not passing the data science course.

If the contingency calculation equation is applied to the values {Pass} → {Bus}, the following is obtained:

$$f'_{12} = \frac{\sum_{j=1}^{5} f_{1j} \sum_{i=1}^{2} f_{i2}}{\sum_{\substack{i=1 \\ j=1}}^{5} f_{ij}} = \frac{(1+5+8+3+2)(5+0)}{6+5+8+4+2} = 3.8$$

As $f_{12} = 5$, this implies that $f_{12} > f'_{12}$, and therefore, there is dependence between passing the course and going to the university by bus. It is a dependency of attraction. This means that students who pass the course are associated with going to the university by bus.

We now analyse if there is an association between the elemental event {No Pass} and the elemental event {Car}.

If the concept of characteristic[7] is introduced, the concept of contingency of characteristics can also be introduced, that is, contingency between the complete subsets defined in the sample space and not only between the exclusive elementary events that form said subsets. To calculate this contingency between characteristics,

[7] Traditionally in textbooks, contingency has been described on the concept of characteristics and their possible values. Defining on the basis of elementary events is new to this book.

the Yule and Pearson contingency coefficients are used, among others.[8] Let us now see how both are calculated:

$$Q = \frac{f_{11}f_{22} - f_{12}f_{21}}{f_{11}f_{22} + f_{12}f_{21}}$$

As an example of the use of the Yule contingency to measure the association of two characteristics or subsets of events, it is necessary to have previously defined some characteristics that define the subsets of the sample space. If we take the sample space used in the first example of Contingency: {Pass, No Pass, Job, No Job}, we can define a first characteristic that is if the students pass the data science course, which groups together the exclusive elemental events {Pass, No Pass}, and a second, which is if they have Job or not that groups together the exclusive elementary events {Job, No Job}. The data that we have on these two characteristics in the table of the students.

Contingency table with the values of the problem

		Course		
		Pass	**No Pass**	**Total**
Occupation	**Job**	5	1	6
	No Job	1	3	4
	Total	6	4	10

and we apply the equation for calculating the Yule contingency, we have:

$$Q = \frac{f_{11}f_{22} - f_{12}f_{21}}{f_{11}f_{22} + f_{12}f_{21}} = \frac{5 \cdot 3 - 1 \cdot 1}{5 \cdot 3 + 1 \cdot 1} = 0.875$$

This indicates that there is a strong dependence of attraction between to have Job or No have Job and Pass or No Pass the course.

- Pearson C contingency: It is given by the equation:

$$C = \sqrt{\frac{\chi^2}{\chi^2 + \sum_{\substack{i=1 \\ j=1}}^{\substack{m \\ n}} f_{ij}}}$$

[8] For example, the PHI contingency coefficient.

The Chi-square value, χ^2, is called the contingency coefficient[9] and is given by the equation:

$$\chi^2 = \sum_{i=1}^{m} \sum_{j=1}^{n} \frac{\left(n_{ij} - n\prime_{ij}\right)^2}{n'_{ij}}$$

For each j

$$n'_{ij} = \frac{\sum_{i=1}^{m} f_{ij} \cdot \sum_{j=1}^{n} f_{ij}}{m.n}$$

The value of C is in the range (0,1). Values close to 0 indicate independence, and values close to 1 indicate dependency.

To see an example of calculating the contingency of Pearson C, we are going to study again the association between the means of transport used to go to the university and to pass or not pass the data science course, when both variables are analysed jointly. If we take the table of manner of transport and pass or not the course for the 25 students and the equations for calculating the contingency of Pearson C, we have:

Crosstabulation or contingency table

	Car	Bus	Train	Bike	Walking	Total
Pass	1	5	8	3	2	19
No Pass	5	0	0	1	0	6
Total	6	5	8	4	2	25

$$\chi^2 = \sum_{\substack{i=1 \\ j=1}}^{\substack{2 \\ 5}} \frac{\left(n_{ij} - n'_{ij}\right)^2}{n'_{ij}} = \frac{\left(1 - \frac{19 \cdot 6}{25}\right)^2}{\frac{19 \cdot 6}{25}} + \frac{\left(5 - \frac{19 \cdot 5}{25}\right)^2}{\frac{19 \cdot 5}{25}} + \frac{\left(8 - \frac{19 \cdot 8}{25}\right)^2}{\frac{19 \cdot 8}{25}}$$

$$+ \frac{\left(3 - \frac{19 \cdot 4}{25}\right)^2}{\frac{19 \cdot 4}{25}} + \frac{\left(2 - \frac{19 \cdot 2}{25}\right)^2}{\frac{19 \cdot 2}{25}} + \frac{\left(5 - \frac{6 \cdot 6}{25}\right)^2}{\frac{6 \cdot 6}{25}}$$

$$+ \frac{\left(0 - \frac{6 \cdot 5}{25}\right)^2}{\frac{6 \cdot 5}{25}} + \frac{\left(0 - \frac{6 \cdot 8}{25}\right)^2}{\frac{6 \cdot 8}{25}} + \frac{\left(1 - \frac{6 \cdot 4}{25}\right)^2}{\frac{6 \cdot 4}{25}} + \frac{\left(0 - \frac{6 \cdot 2}{25}\right)^2}{\frac{6 \cdot 2}{25}}$$

$$= 16.32$$

[9]The contingency coefficient can also be used to determine the probability of dependence of the two characteristics through Pearson's χ^2 distribution function, such that for a value of χ^2 with ν degrees of freedom and for a significance level α the characteristics will be considered independent when $\chi^2 < \chi_{\alpha,\,\upsilon}{}^2$

Once the value of $\chi^2 = 6.5$ is calculated, the value of C is calculated as follows:

$$C = \sqrt{\frac{\chi^2}{\chi^2 + \sum_{\substack{i=1 \\ j=1}}^{n} f_{ij}}} = \sqrt{\frac{16.32^2}{16.32^2 + 25}} = 0.956$$

Therefore, a value so close to one indicates a high degree of association between the two characteristics. They present a great dependence.

Correlation

The fourth coefficient that we are going to see is the *correlation coefficient*, which we know can be defined as "Measure of the existing dependency between random variants." We are going to apply it to qualitative ordinal data because in the chapter about regression, we saw how it can be applied to quantitative data.

To calculate the existing correlation between two events formed by elementary events whose values are ordinal qualitative, it is essential to be able to convert these qualitative values to numerical values (it is important to know that each of these events could be considered an ordering of a given characteristic). As they are ordinal qualitative values, they can be ordered according to some criteria in such a way that the numerical value of the order of each of them can be taken as their numerical value. In this way, each characteristic corresponds to a natural number. Once this is done, different coefficients can be used to calculate the correlation between both variables. Let us look at the Spearman, Kendall, and Goodman-Kruskal correlation coefficients.

- Spearman's correlation coefficient: It is given by the equation

$$r_s = 1 - \frac{6\sum_{i=1}^{n} d_i^2}{\left(\sum_i^{\mathrm{n}} f_i\right)^3 - \sum_i^{\mathrm{n}} f_i}$$

where $d_i = a_i - b_i$ and $-1 \leq r_s \leq 1$. When there is total dependence $a_i = b_i$ and therefore $d_i = 0$ and $r_s = 1$. When there is independence $r_s = -1$.

Now, we are going to see an example of the application of Spearman's correlation coefficient to the analysis of the association of two characteristics with ordinal qualitative values. To study the correlation between two ordinal qualitative characteristics we need, as has been said in the previous paragraph, they can be ordered in some way, so the characteristics used thus far in this topic cannot be used. Two new characteristics will be used, related to those previously seen, and which also correspond to data science course characteristics, for which they are easily ordered. They are given in the next Table and are the position in which the chapter is taught. The

first chapter studied in the course will be the first, and the relative success of the students in that chapter, that is, the first chapter, will be the chapter for which the students have obtained the best academic results.

Ordering of the chapters according to the relative success of the students in that chapter

Chapter	Schedule	Success
Introduction	1	2
Data	2	1
Anomalies	3	3
Probability	4	4
Clustering	5	5
Classification	6	7
Association	7	6

The following table presents the values of the coefficients used in the Spearman equation.

Spearman's coefficients

Chapter	a_i	b_i	$d_i = a_i - b_i$	d_i^2	f_{ij}
Introduction	1	2	−1	1	1
Data	2	1	1	1	1
Anomalies	3	3	0	0	1
Probability	4	4	0	0	1
Clustering	5	5	0	0	1
Classification	6	7	−1	1	1
Association	7	6	1	1	1
				$\sum_{i=1}^{n} d_i^2 = 4$	$\sum_{i}^{n} f_i = 8$

If the Spearman coefficient is applied, we have:

$$r_s = 1 - \frac{6\sum_{i=1}^{7} d_i^2}{\left(\sum_{i}^{7} f_i\right)^3 - \sum_{i}^{7} f_i} = 1 - \frac{6.4}{(7)^3 - 7} = 1 - 0,07 = 0,92$$

Therefore, when r_s is close to 1, there is dependence between the order in which the chapter has been taught and the academic results of the students.

- Kendall's correlation coefficient: It is given by the equation:

$$\tau = \frac{2\left(\sum_{i=1}^{n} P_i - \sum_{i=1}^{n} Q_i\right)}{\sum_{i}^{n} f_i \left(\sum_{i}^{n} f_i - 1\right)}$$

where Q is the total number of inversions of bi with respect to ai and P is the number of permanence.

To calculate Kendall's correlation coefficient, the first thing to do is to order the table in which the data of the two characteristics are displayed in such a way that in one of them, the values are ordered from lowest to highest, and the other characteristic will then appear in the order that corresponds to the values. In the event that any value has the same order number for both characteristics, that value is eliminated from the table and from the analysis.

Once the table has been ordered, the investments and permanence for each value are quantified in the variable that is not ordered from lowest to highest. By investment Qi, we mean the number of values less than Qi that are found in rows greater than Qi. By permanence Pi is understood the number of values greater than Pi that are found in rows greater than Pi.

Once all these previous results have been calculated, the equation is introduced, and the coefficient is obtained. The τ coefficient verifies $1 \leq \tau \leq 1$. When the value is close to 0, it indicates low correlation, and when it is close to 1 or -1, it indicates high correlation. The Kendall coefficient presents the problem that when there are ties, it cannot reach the limit values 1 and -1 because the denominator is always greater than Q and P.

To see an example of Kendall's correlation coefficient for the study of the association of two characteristics with ordinal qualitative values, we return from the table with the chapters and the relative success of the students in that chapter.

If the investments and permanence are quantified, the values of the rows greater than 3 have to be analysed, which is the one we are studying, that is, rows 4 to 7. In these rows, the values 1, 2, 3, and 4 are observed in rows 5, 6, and 7, respectively. These 4 values are less than 5, which is the value of row 3, and are found in rows greater than 3 in rows 5, 6, and 7. Therefore, $Q3 = 4$. Regarding permanence, there is only a value greater than 5 in the rows that follow row 3, which is found in row 4, and the value is 7. Therefore, $P3 = 1$. The table gives all the values of Q_i and P_i for all the values in the table.

Q_i and P_i values for all values in the table

Chapter	Schedule	Success	Q_i	P_i	f_{ij}
Introduction	1	2	1	5	1
Data	2	1	0	5	1
Anomalies	3	3	0	4	1
Probability	4	4	0	3	1
Clustering	5	5	0	2	1
Classification	6	7	1	0	1
Association	7	6	0	0	1
			$\sum_{i=1}^{7} Q_i = 2$	$\sum_{i=1}^{7} P_i = 19$	$\sum_{i}^{n} f_i =$

If the values from the table are introduced into the equation, we have:

$$\tau = \frac{2\left(\sum_{i=1}^{7} P_i - \sum_{i=1}^{7} Q_i\right)}{\sum_{i}^{7} f_i \left(\sum_{i}^{7} f_i - 1\right)} = \frac{2(19-2)}{7(7-1)} = 0.81$$

Being a value closer to 1 than 0 indicates a high correlation between the values, which agrees with the conclusion obtained using the Spearman coefficient.

- Goodman-Kruskal correlation coefficient. It is given by the equation:

$$\gamma = \frac{\left(\sum_{i=1}^{n} P_i - \sum_{i=1}^{n} Q_i\right)}{\left(\sum_{i=1}^{n} P_i + \sum_{i=1}^{n} Q_i\right)}$$

When the value is close to 0, it indicates low correlation, and when it is close to 1 or -1, it indicates high correlation. The advantage of Goodman-Kruskal's coefficient over Kendall's is that it is not influenced by the number of ties since it eliminates them from the numerator and the denominator. If there are no ties, both coefficients coincide.

We use the table for Kendall's example of the association of two characteristics with ordinal qualitative values. If the Goodman-Krustal equation is applied to these values, we obtain:

$$\gamma = \frac{\left(\sum_{i=1}^{7} P_i - \sum_{i=1}^{7} Q_i\right)}{\left(\sum_{i=1}^{7} P_i + \sum_{i=1}^{7} Q_i\right)} = \frac{(19-2)}{(19+2)} = 0.89$$

Therefore, the conclusion is the same as that obtained in the two previous cases and is highly correlated.

Analysis of the Association of Events Composed by More Than One Elementary Event

As we know, from the previous subsubsection, association studies seek to find patterns of joint occurrence of disjoint[10] events, that is, to see if the probability of joint occurrence of several events exceeds certain thresholds. In the analysis of the association of events composed of more than one elementary event we try to discover if the joint appearance of events the exceeds a threshold value for the two measures of the measures introduced in the previous subsubsection and that are complementary to each other, support and confidence. We are going to give again their definitions because we need to have them very clear and fresh to understand in depth all the contents of this subsubsection.

[10] Disjoint events are those that do not have any elemental events in common.

Let us remember the first one, the *support*. To be able to compare with the threshold value referred to in the previous paragraph, the degree of association[11] between various events is established as the calculation of the classical probability[12] of appearance of the elements of the set P (E) that encompass said events. This measure is called support, s, which is more formally defined as:

$$\forall \{A_i\}_{i=1}^{\infty} \subset P(E) \; con \; A_i \cap A_j = \varnothing \forall i \neq j, s : P(E) \rightarrow \mathbb{R}^+ / s\left(A_i \cup A_j\right) = \frac{n_{A_i \cup A_j}}{n_T}$$

The acceptance threshold of support is not fixed but will be arbitrarily set a priori and will depend on the objectives of the study.

In association studies, it is essential to bear in mind the fact that elementary events are not symmetric or equiprobable, and consequently, to establish the probability of appearance of the elements of P (E), it is essential to have a sample of events.

To introduce the concept of event association through an example, we will use a shopping basket, which will be made up of five products: Bread, Water, Coffee, Milk, and Oranges. For this experiment, we must establish the sample space, the parts of the E set, and the number of elements of the parts of the E set.

We start the solution of this problem by defining the sample space, and the elemental events that constituted the sample space are going to be each of the products of the basket considered individually. For that reason, as we know from the probability chapter, the sample space is:

E = {Bread, Water, Coffee, Milk, Oranges}

From this sample space, the set P(E), as we know from the probability chapter, is the one formed by all the subsets that can be formed with the set of elements of the sample space, that is:

P (E) = { ∅,{Bread}, {Water}, {Coffee}, {Milk}, {Oranges}, {Bread, Water}, {Bread, Coffee}, {Bread, Milk}, {Bread, Oranges}, {Water, Coffee}, {Water, Milk}, {Water, Oranges}, {Coffee, Milk}, {Coffee, Oranges}, {Milk, Oranges}, {Bread, Water, Coffee}, {Bread, Water, Milk}, {Bread, Water, Oranges}, {Bread, Coffee, Milk}, {Bread, Coffee, Oranges}, {Bread, Milk, Oranges}, {Water, Coffee, Milk}, {Water, Coffee, Oranges}, {Coffee, Milk, Oranges}, {Bread, Water, Coffee, Milk}, {Bread, Water, Coffee, Oranges}, {Bread, Water, Milk, Oranges}, {Bread, Coffee, Milk, Oranges}, {Water, Coffee, Milk, Oranges}, {Bread, Water, Coffee, Milk, Oranges}}.

[11] In order not to repeat throughout the text the terminology: "search for patterns or association of disjoint events," from here on we will only use "association of events" as it is the most widely used terminology.

[12] It is very important to bear in mind that the definition of probability that will be used in association studies will be the classic one and from here on, whenever we refer to probability, it will be the classic one.

And finally, to solve all the items the statement of this case, we must calculate the number of elements in the set parts of E, and as we know from the probability problem, this number is calculated by:

Cardinal (P(E)) $= 2^n$, where *n is* the number of elements in the sample space, that, in this case, is 5; for that reason, the cardinal of the P(E) set is

$$(\mathrm{P}(\mathrm{E})) = 2^n = 2^5 = 32$$

To give an example of the support of an association, the support of the association between the sets A1 = {Bread, Water} and A2 = {Milk} must be calculated.

When we start to solve this problem, we can see immediately that, as mentioned above, to solve the problem of the association of this shopping basket,[13] it is essential to have a sample of events that allow us to establish the probabilities of appearance of the elements of P (E).

The sample that we have will consist of the following six shopping baskets: {Bread, Water, Milk, Oranges}, {Bread, Water, Coffee, Milk}, {Bread, Water, Milk}, {Bread, Coffee, Milk}, {Bread, Water}, {Milk}.

Once we have defined the sample that allows us to establish the probability of the appearance of the different events, we calculate the support of the association of the disjoint events A_1 = {Bread, Water} and A_2 = {Milk}. What we have to calculate is the classical probability of appearance of the set $A_1 \cup A_2$ = {Bread, Water, Milk}, as a set formed only by these elements or within as a subset of any set of P(E) that contains, among others, these elements. As seen above, this probability is given by:

$$s(A_1 \cup A_2) = \frac{n_{A_1 \cup A_2}}{n_T} = p(A_1 \cup A_2)$$

Examining the sample, the number of elements of P (E) that the set contains ($A_1 \cup A_2$) are 3: {Bread, Water, Milk, Oranges}, {Bread, Water, Coffee, Milk}, {Bread, Water, Milk}, in consequence $n_{A_1 \cup A_2} = 3$; and the total number of events in the sample is $n_T = 6$, consequently the support of the association of the events A_1 = {Bread, Water} and A_2 = {Milk}, is:

$$s(A_1 \cup A_2) = \frac{n_{A_1 \cup A_2}}{n_T} = \frac{3}{6} = 0.5$$

Therefore, this association would have 50% support. Being, as we know, the support a probability, it moves in a range that goes from 0 to 1.

What other associations would have the same support as the one studied?

[13] We use the usual example of a shopping basket because it seems very pedagogical, although we will only coincide with other texts in the domain, since the rest of the discussion will have a quite different approach.

The answer to this question is very interesting because it is important to realize that the support measure of an association is the same for all combinations of associations of disjoint sets that result in the same union set. If we consider the definition of support, it is very clear this conclusion because $n_{A_1 \cup A_2}$, and n_T will be the same for all of them, and, in consequence, $s(A_1 \cup A_2)$ will be too. For this reason, the associations $A_1 = \{\text{Bread, Water}\}$ and $A_2 = \{\text{Milk}\}$ would be the same for the associations $A_1 = \{\text{Bread, Milk}\}$ and $A_2 = \{\text{Water}\}$; $A_1 = \{\text{Milk, Water}\}$ and $A_2 = \{\text{Bread}\}$.

The support is an interesting measure because it gives us a measure of how frequent the association we are looking for is because although the association of the elements that make up the association was very strong and that we are going to measure with the measure that we are going to see then, if this association were observed very rarely, it would not make sense to study it.

However, measuring only the probability, or frequency, of the appearance of the association studied in the available sample is not enough to establish the degree to which the events studied are associated. For this reason, we must remember that we use the second measure referred to at the beginning of this chapter called *confidence*, c, is defined, which will measure the classic probability of the appearance of the association for a subset of the original sample composed only of the events in which occurs one of the two sets whose association is being analysed. The one that is the origin of the association, that is, the confidence that a second set appears when the first has appeared, will be measured. More formally, it is defined as:

$$\forall \{A_i\}_{i=1}^{\infty} \subset P(E) \; con \; A_i \cap A_j = \varnothing \forall i \neq j, c : P(E) \rightarrow \mathbb{R}^+ / c\left(A_i \cup A_j\right) = \frac{n_{A_i \cup A_j}}{n_{A_i}}$$

As was also said at the beginning of this chapter, as with support, for trust, there will be an acceptance threshold, which will not be fixed but will also be arbitrarily set a priori and will depend on the objectives of the study.

As an example of the confidence calculation, we calculate the confidence of the association of the disjoint events whose support we calculated in the previous example. These are $A_1 = \{\text{Bread, Water}\}$ and $A_2 = \{\text{Milk}\}$.

What we have to calculate is the probability of appearance of the set $A_1 \cup A_2 = \{\text{Bread, Water, Milk}\}$, as a set formed only by these elements or within as a subset of any set of P (E) that contains, among others, said elements, but unlike the previous example, let us not calculate the probability of occurrence over the complete sample but rather take a subset of it formed only by those events that contain or are equal to the set: $A_1 = \{\text{Bread, Water}\}$. The reduced sample is composed of the four events: {Bread, Water, Milk, Oranges}, {Bread, Water, Coffee, Milk}, {Bread, Water, Milk}, {Bread, Water}.

Examining the sample, the number of elements of P (E) that the set contains $(A_1 \cup A_2)$ are 3: {Bread, Water, Milk, Oranges}, {Bread, Water, Coffee, Milk}, {Bread, Water, Milk}, in consequence $n_{A_1 \cup A_2} = 3$; and the total number of events in

the reduced sample is four, which means that $n_{A_1} = 4$; consequently, the confidence of the association of the events A_1 = {Bread, Water} → [14]A_2 = {Milk}, is:

$$c(A_1 \cup A_2) = \frac{n_{A_1 \cup A_2}}{n_{A_1}} = \frac{3}{4} = 0.75$$

Therefore, this association would have 50% support. Again, as it is a probability, the support moves in a range that goes from 0 to 1; therefore, this association would have 75% confidence.

What other associations would have ever the same confidence as the one studied?

It is also important to realize that the confidence measure of an association, contrary to what happened with the support, is not the same, nor does it invert the sense of the association, much less for all combinations of associations of disjoint sets that result in the same union set.

A1 = {Bread, Water} → A2 = {Milk} ≠ A2 = {Milk} → A1 = {Bread, Water}

We are going to prove this in the next example.

To prove, with an example, the fact that the sense of the association means for the calculus of the confidence, we are going to answer the question: What is the confidence of A2 = {Milk} → A1 = {Bread, Water}?

The confidence obtained for the association A1 = {Bread, Water} → A2 = {Milk}, 75%, is not the same as the one obtained would be the same for the association: A1 = {Milk} → A2 = {Bread, Water}, since the latter would keep the numerator, but the denominator would be 5, since the events of the reduced sample, that is, the events that contain milk, are 5, and not 4 as in the previous case. Those events are {Bread, Water, Milk, Oranges}, {Bread, Water, Coffee, Milk}, {Bread, Water, Milk}, {Bread, Coffee, Milk}, {Milk}, whereby confidence is:

$$c(A_1 \cup A_2) = \frac{n_{A_1 \cup A_2}}{n_{A_1}} = \frac{3}{5} = 0.6 \equiv 60\%$$

The same happens, that is, the confidence also changes if we study any other combination of the elements of the set $c(A_1 \cup A_2)$, for example, A_1 = {Bread, Milk} → [15]A_2 = {Water}

[14] We write here a sense because, as explained above, for confidence it is essential to indicate the sense of the association and in this case what is being analysed is the trust of the association between the events Bread, Water, and Milk, but from the perspective of knowing what association there is the appearance of Bread and Water with Milk, this is to what degree when you have Bread and Water you have or not Milk.

[15] We write here a footnote because, as explained above, for confidence it is essential to indicate the sense of the association and in this case what is being analysed is the confidence of the association between the events Bread, Water, and Milk, but from the perspective of knowing what association there is the appearance of.

Once the concepts of support and trust have been seen, we can establish what the purpose of an association analysis is going to be: Given a sample space E, its set P (E), and support and trust threshold values arbitrarily set depending on the interests of the study, an association study will try to establish, from a sample of available events, which sets of P (E) can be considered associated, that is, whenever one of the sets is observed, it will also observe the other with a probability greater than or equal to the set by threshold values. To consider the two associated sets, minimum thresholds for the support and confidence values will be arbitrarily established in each case.

Once we have seen what an association analysis consists of, the difficulty it is going to have is immediately identified due to the high volume of calculations that it requires, even for small sample spaces. This volume is due to the high number of associations between sets that can be studied, since if there are n elementary events, the number of associations is:

$$3^n - 2^{n+1} + 1$$

To see an example about what means the equation of calculus of the number of the possible associations that we can have we are going to apply it to the example we are dealing with, with a sample space of 5 elements, E = {Bread, Water, Coffee, Milk, Oranges}, and try to answer the question: how many associations we must study? And after that how many we would have to study if we would have ten elements in the sample space? And with twenty?

For the first question, in which we have 5 elements in the sample space, we have:

$$3^n - 2^{n+1} + 1 = 3^5 - 2^6 + 1 = 180$$

That is not too much.

For the second question, we have:

$$3^n - 2^{n+1} + 1 = 3^{10} - 2^{11} + 1 = 57001$$

That can also be possible to analyse, but for the third question, we would have:

$$3^n - 2^{n+1} + 1 = 3^{20} - 2^{21} + 1 = 3500000000$$

We can see for the first time the real difficulty of the analysis, and a sample space of 20 elements is very small for real problems.

Therefore, once we can see the previous definitions and especially the previous example, it is essential to use algorithms that reduce the set of associations to be analysed and thus make the problem manageable.

The rest of this chapter will describe the operation of one of these algorithms.[16] However, before going into the specific details of how it is designed, we will describe the two[17] steps that, in a generic way, all tend to use all the algorithms for association analysis and that are based on the separation of the calculation of support and confidence:

A. Identification of frequent events. A minimum support threshold is set, and all the elements of P (E) that meet or exceed it are identified. Once identified, step 2 will only apply to those events.
B. Identification of trusted associations. A minimum confidence threshold is set, and associations that meet or exceed it are identified. It is important to remember here that the identified associations, in addition to identifying which elementary events are related, must also impose a meaning on the association, as seen above in the description of the concept of trust.

As an example, regarding how the association algorithms are designed, we are going to follow with the shopping basket.

In the first step, if the support acceptance threshold was 40%, the association between the events $A_1 = \{\text{Bread, Water}\}$ and $A_2 = \{\text{Milk}\}$ was studied because it had a support of 50%, but the association between the events $A_1 = \{\text{Bread, Coffee}\}$ and $A_2 = \{\text{Milk}\}$ was not studied because it did not reach it.

And once all the associations without enough support were removed from the sample, we only would pass to analyse the confidence of the possible associations with enough support, but in this second step, if the acceptance threshold of the confidence obtained for the association were 70%, the association $A_1 = \{\text{Bread, Water}\} \rightarrow A_2 = \{\text{Milk}\}$, would be accepted because it has a confidence level of 75%, but the association $A_1 = \{\text{Milk}\} \rightarrow A_2 = \{\text{Bread, Water}\}$, it would not be since its level is 60%.

Once we have seen how the association algorithms work in a generic way, we are going to see how they work in a specific way.

Apriori Algorithm

The definition of the Apriori algorithm follows the generic two-step process described above. Let us see how it treats each step.[18]

A. Step A. Identification of frequent associations. It is based on the calculus of the *support* and the identification of events with a support, s, greater than or equal to the threshold or frequent events. To optimize the identification of frequent events

[16] Although not all the existing ones will be seen, since only such a length could be covered in a monographic text on association, the most widely used and internationally disseminated algorithm will be seen.

[17] We will call them A and B for clarity.

[18] That, following what has been seen above, we will also call here, for reasons of clarity, A and B

and reduce the search space, the a priori algorithm is based on the fact that the support measure is *antimonotone*,[19] so if an event is frequent, all the subsets of said event are also frequent, since their support is greater than or equal to that of the event that contains them.

B. Step B. Identification of trusted associations. It is based on the calculation of the *confidence*, c, and the identification of events with a confidence support greater than or equal to the threshold or frequent events. The algorithm uses the ap-genrules function, which is based on the theorem: Let A and B be two sets. If the association $A \rightarrow B - A$ does not exceed the confidence threshold, then any association $A' \rightarrow B - A'$, where A' is any subset of A ($A' \subseteq A$), will also not reach it.

We are going now to see in detail how both steps are performed and which substeps they have.

Step A

We are going to start the description of step A with the definition and an example of the antimonotone properties that, as has been said in the previous paragraph, if an event is frequent, all the subsets of said event are also frequent, since their support is greater than or equal to that of the event that contains them. For a better understanding, we can see the following example:

We take the same sample basked with the six events as in the previous examples: {Bread, Water, Milk, Oranges}, {Bread, Water, Coffee, Milk}, {Bread, Water, Milk}, {Bread, Coffee, Milk}, {Bread, Water}, {Milk}.

If we apply the antimonotone property to any of the events, for example {Bread, Water, Milk}, and we calculate its support and form it, we determine that it is frequent, the consequence is that the events {Bread}, {Water}, {Milk}, {Bread, Water}, {Bread, Milk}, and {Water, Milk} must be frequent.

This is quite logical since when observing the number of times that the set {Bread, Water, Milk} appears, to calculate its probability or support, we have seen that it is 0.5 because it appears 3 times, {Bread, Water, Milk, Oranges}, {Bread, Water, Coffee, Milk}, {Bread, Water, Milk}, in 6 observations. If we look at the set {Bread, Water}, it would appear at least 3 times, which are the ones that appear next to Milk, and it could also appear more times, together with another product or alone. If we look at the sample,[20] which is the one we are working with throughout this

[19] A function – here we will take the support function s, because they are the one we are working on, but it could be any other – is antimonotonic on a set P (E) when it verifies that

$$\forall A, B \in P(E) / A \subseteq B \rightarrow s(B) \leq s(A)$$

[20] We repeat the sample here so as not to have to go back and make reading easier: {Bread, Water, Milk, Oranges}, {Bread, Water, Coffee, Milk}, {Bread, Water, Milk}, {Bread, Coffee, Milk}, {Bread, Water}, {Milk}.

chapter, we can see that it appears once more alone, in {Bread, Water}, that means that it appears 4 times, and consequently its support is equal to or greater than that of {Bread, Water, Milk}, in this case greater. In this case, 4/6=0.67, and consequently, its support is greater than the set {Bread, Water, Milk} that contains it.

Therefore, if it is turned around at the beginning, it can be concluded that no set of P(E) can have greater support than the subsets it contains. This allows the Apriori algorithm, once the support threshold has been set, to gradually reduce the set of P (E) events that are candidates to overcome it, starting from the elementary events. That is, if a set consisting only of an elemental event does not reach the support threshold, no set consisting of more elements, in which one must be that elemental event, will not exceed said support threshold either. Once we know this, we return to the description of step A and its substeps.

To carry out step A, the a priori algorithm will have two substeps:

Step A.1

In Step 1 of the a priori algorithm, the algorithm makes a single pass through all the elementary events to calculate their support and eliminate those that do not reach the set threshold support.

To see an example of Step 1 of the a priori algorithm, we set the support threshold at a support or probability of occurrence of the event of 50% applied to the basket sample.

From there, we begin to apply the a priori algorithm with the first iteration in which we analyse all the elementary events, that are {{Bread}, {Water}, {Coffee}, {Milk}, {Oranges}}

Since the number of events in the sample is six to be selected, they will have to appear 3 or more times, so that their support is 0.5 or more. If we take the support equation, we have:

$$s(A_1 \cup A_2) = \frac{n_{A_1 \cup A_2}}{n_T} = \frac{n_{A_1 \cup A_2}}{6} \geq 0.5 \rightarrow n_{A_1 \cup A_2} \geq 3$$

We start with {Bread} and observe that it appears 5 times in {Bread, Water, Milk, Oranges}, {Bread, Water, Coffee, Milk}, {Bread, Water, Milk}, {Bread, Coffee, Milk}, and {Bread, Water}; therefore, we select it. Next, we observe {Water}, which appears 4 times; therefore, we also select it.

If we look at {Coffee} and see that it appears only twice, in {Bread, Water, Coffee, Milk} and {Bread, Coffee, Milk}, we do not select it since its support is:

$$s(A_1 \cup A_2) = \frac{n_{A_1 \cup A_2}}{n_T} = \frac{2}{6} = 0.33 \equiv 30\%$$

We look at {Milk}, which appears 5 times, and we select it.

Finally, we observe {Oranges}, which appears once, in {Bread, Water, Milk, Oranges}, and we do not select it.

Therefore, after step 1, we have reduced the set of elementary events of interest to the events

$$\{\{\text{Bread}\},\{\text{Water}\},\{\text{Milk}\}\}$$

with which we have reduced the set of events to be analysed in the second step to those that contain only any combination of any dimension of the elementary events {{Bread}, {Water}, {Milk}}, since those containing {{Coffee}, {Oranges}} cannot have a support greater than 0.5 because 3 or more times, since the elements that compose them do not appear alone 3 or more times, it will be impossible for them to appear 3 or more times as part of any set of more elements. Once we have seen this, we move on to the second step.

Step A.2.

In Step 2, the a priori algorithm will carry out successive steps, two for each dimension, which will be called 2.1 and 2.2, starting with the sets of two elements and ending when it is not possible to identify a dimension in which there is equal or greater support than the threshold. It consists in turn of two substeps:

Step A.2.1

The first step that the a priori algorithm will perform in each dimension will be to apply the function called Apriori-gen to identify the candidate sets of a dimension, using as a basis the sets selected for the previous dimension, that is, for example, for the sets dimension, two will use the selected elementary events. There are different methods to carry out this identification, and the a priori-gen function uses the Fk−1 x Fk−1 method.

The Fk−1 x Fk−1 method generates the candidate sets in a given dimension k by joining pairs of candidate sets of the previous dimension k−1 but only joins those pairs in which their first k−2 elements, or what is the same, are formally expressed:

Let $A = \{a_1, a_2, \ldots, a_{k-1}\}$ and B $= \{b_1, b_2, \ldots, b_{k-1}\}$ be two frequent events identified when dimension k−1 has been analysed. A and B will only join to form a candidate event in dimension k if they satisfy the following two conditions:

1. $a_i = b_i$ for i = 1,2,. . ., k−2
2. $a_{k-1} \neq b_{k-1}$

In this example, we apply the *apriori-gen* function to identify candidate events in each dimension for the elemental events accepted, which are: {{Bread}, {Water}, {Milk}}.

k=2. We start applying the $F_{k-1} \times F_{k-1}$ method to identify the two-element candidate events, and in consequence, k is, in this case, 2, since we are in dimension 2. Consequently, k−1 is equal to 1 and is able to join two sets of dimension 1, that is, with a single element, the first k−2 elements must coincide, which in this case are $2-2 = 0$ elements and the elements $a_{k-1} \neq b_{k-1}$ do not coincide, which in this case

are $a_1 \neq b_1$. Therefore, in this case, it has been seen that the method can be applied perfectly, since the only element that the dimension 1 sets consist of does not have to match and there does not have to be any element that matches.

Considering all the previous, for this very simple dimension, the construction of new sets will consist of joining the frequent elementary events: {{Bread}, {Water}, {Milk}}, so these union sets are: {Bread, Water}, {Bread, Milk}, {Water, Milk}.

k=3. In step 2.2, we will see how the support is calculated for each of these two-dimensional sets, but we will assume that all three are selectable because the supports of all three are above the threshold, to see how the method $F_{k-1} \times F_{k-1}$ would identify the candidate events of dimension 3 starting from these three sets of dimension 2. According to the $F_{k-1} \times F_{k-1}$ 1 method, to identify the candidate events of three elements, the first thing we do is identify the value of k, which in this case is 3, since we are in dimension 3, consequently k−1 is equal to 2 and to be able to join two sets of dimension 2, that is, with two elements, the first $k-2$ elements must coincide, which in this case is: 3-2 = 1 element and the elements $a_{k-1} \neq b_{k-1}$, must not coincide, which in this case are $a2 \neq b2$.

Considering all the previous, for this dimension {Bread, Water} and {Bread, Milk} can be joined because it is true that $a_1 = b_1$ and $a_2 \neq b_2$, but it cannot be joined with either of the two sets, the set {Water, Milk}, because a_1 would be different from b_1 if you try to join {Water, Milk} with any of the other two sets. Consequently, the union set would be {Bread, Water, Milk}.

Step A.2.2

The second step that the a priori algorithm will perform in each dimension will be to calculate the support of the candidate events identified in step 2.1 to select the events with a value equal to or greater than the threshold. To do this, each candidate event must be compared with all the events that make up the analysed sample to identify those events in the sample that contain or are equal to said candidate event. To do this, the a priori algorithm will need to perform three new substeps within the second substep of step 2 of step A. They will be substeps:

- A.2.2.1. Partition the candidate events with a hash tree.
- A.2.2.2. Partition of the sample events with the same hash tree.
- A.2.2.3. Comparison of both partitions. All matching sheets increment the numerator by one unit in the candidate event support calculation.

Now, we will describe in detail how it works for each one.

Step A.2.2.1

In the first step, the candidate events are partitioned using a hash tree[21] and summary tree.

[21] The explanation of how a hash tree is built will be seen through the example because it carries greater clarity.

To see an example of Step 2.2.1. of the Apriori algorithm, the partition using a hash tree, it is very useful to enumerate the elementary events in the sample space and from here to carry out their subsequent treatment using their corresponding number.

We will now carry out this previous step and enumerate the events in the sample:

$$\{\{\text{Bread} = 1\}, \{\text{Water} = 2\}, \{\text{Coffee} = 3\}, \{\text{Milk} = 4\ \}, \{\text{Oranges} = 5\}\}$$

And from here, we are going to treat them by their numbers, retaking the candidate events for the dimension accordingly. Consequently, the candidate events for dimension 2, that is, sets with two elements that we had identified in the example of the previous step, were:

$$\{\text{Bread, Water}\}, \{\text{Bread, Milk}\}, \{\text{Water, Milk}\}$$

and changing to the numerical notation, we have the following sets:

$$\{12\}, \{14\}, \{24\}.$$

In the case of the possible candidate event of dimension 3 that we saw in the previous example

{Bread, Water, Milk}

That numerically it would correspond to

$$\{124\}$$

Once we have the candidate events named with numbers, in this Step A.2.2.1. To build the hash tree and structure each node, a partition (or hash) function must be used. To do that, we use the partition function h (p) = p mod 3.

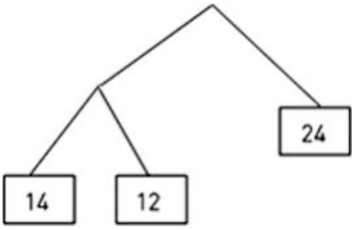
If this function is taken, the indices will be grouped according to the remainder of their division by 3; for example, the division of 1, 4, and 7 by 3 will give 1 because the divisions that give rise to these numbers are 1/3 = 0.3 with remainder 1; 4/3 = 1 with remainder 1; and 7/3 = 2 with remainder 1. Therefore, they will be in the same node.

If we apply this to the rest of the numbers, we have that the three nodes will be {1, 4, 7}, with rest 1; {2, 5, 8}, with rest 2, 2/3 = 0.6 with remainder 2; 5/3 = 1 with remainder 2; and 8/3 = 2 with remainder 2; and {3, 6, 9}, with rest 3, 3/3 = 1 with remainder 0; 6/3 = 2 with remainder 0; and 9/3 = 3 with remainder 0, and they will be arranged from left to right. Therefore, for example, events starting with 1, 4, or 7 will be structured at the first node.

Once we are in the first node, suppose, for example, for those that start with 1, those with a second digit, for example, a 4, will be in the node corresponding to the first set, that is, to the left, and those with, for example, a 2, will be in the central node.

Taking this into account, the candidate events of dimension 2 of the exercise and the description of how to obtain the partition tree, we must obtain the partition tree for the candidate events of all the possible dimensions of the exercise of the basket numbered in the previous step.

For dimension 2, the candidate events will be structured as follows:



The candidate events of dimension 3 will be structured as follows:

being a single element, it would only have one node.

Step A.2.2.2

In the second step, the events of the available sample for the analysis that is carried out must be partitioned using the same procedure as for the candidate events. First, we will numerate the sets of each event in the sample, and next, we will obtain a hash tree.

To see an example about how to perform step A.2.2.2 of the Apriori algorithm, we follow solving the problem of the basket. Following the theoretical description, the first thing we do is number the sets in the sample. We remember that the six events in the set of the sample were:

{Bread, Water, Milk, Oranges}, {Bread, Water, Coffee, Milk}, {Bread, Water, Milk}, {Bread, Coffee, Milk}, {Bread, Water}, {Milk},

so they are numbered:

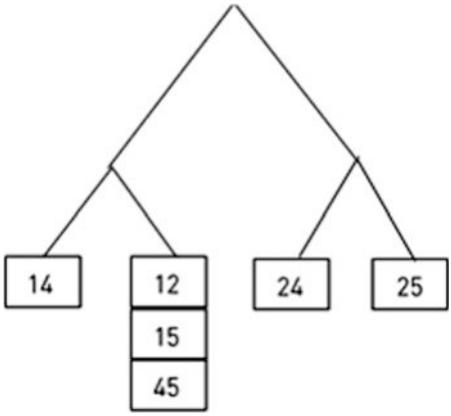
$$\{1245\}, \{1234\}, \{124\}, \{134\}, \{12\}, \{4\}.$$

Next, we partition them using the same hash tree function h(p) = p mod 3. We are going to do it sequentially one by one and for each dimension.

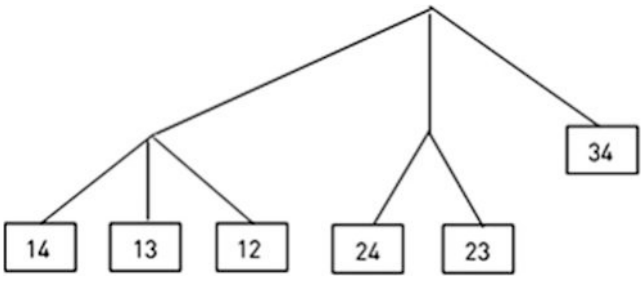
For Dimension 2, the sample is structured as follows:

We start with those of dimension 2, that is, sets of two elements. To do so, from each event in the sample, we will obtain all the possible subsets of 2 elements that can be obtained from it, that is, all the combinations without repetition of order two that can be formed with the elementary events that make up the event. Then, we will perform a hash tree to structure the sets obtained. The partitions obtained are:

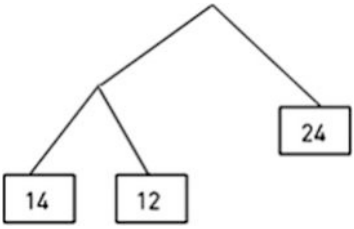
- 1245. From event 1245, the following events of dimension 2 can be generated, and the following partition:



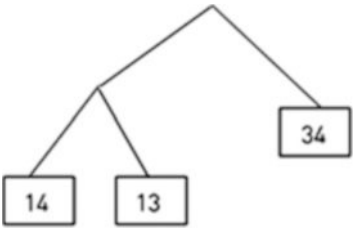
- 1234. From event 1234, the following events of dimension 2 can be generated, and the following partition:



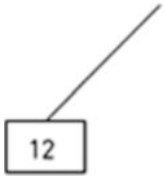
- 124. From event 124, the following events of dimension 2 can be generated, and the following partition:



- 134. From event 134, the following events of dimension 2 can be generated, and the following partition:



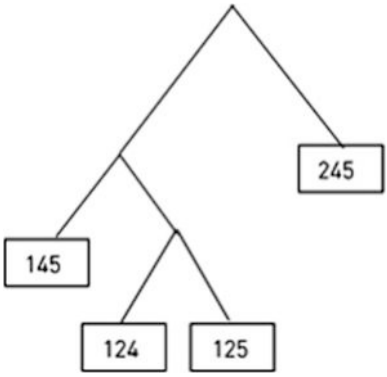
- 12. From event 12, the following event of dimension 2 can be generated, and the following partition:



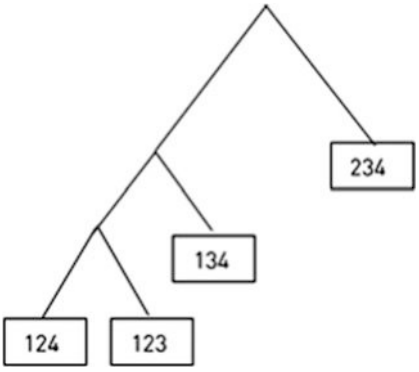
- 4. From event 4, no event of dimension 2 can be generated.

We are now going to do the same with the events of dimension 3. For dimension 3, the sample is structured as follows:

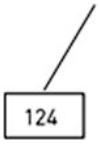
- 1245. From event 1245, the following events of dimension 3 can be generated, and the following partition:



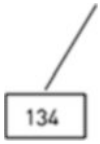
- 1234. From event 1234, the following events of dimension 3 can be generated, and the following partition:



- 124. From event 124, the following event of dimension 3 can be generated, and the following partition:



- 134. From event 134, the following event of dimension 3 can be generated, and the following partition:



- 12. From event 12, no event of dimension 3 can be generated.
- 4. As of event 4, no event of dimension 3 can be generated.

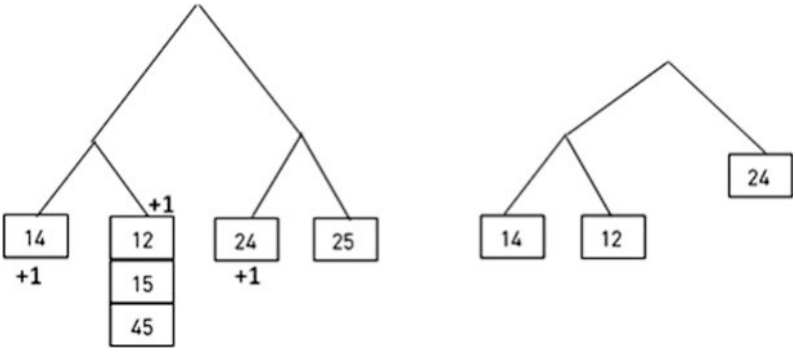
Step A.2.2.3

In the third step, both results are compared, and all the matching sheets will increase the numerator by one unit in the calculation of the support of the candidate event.

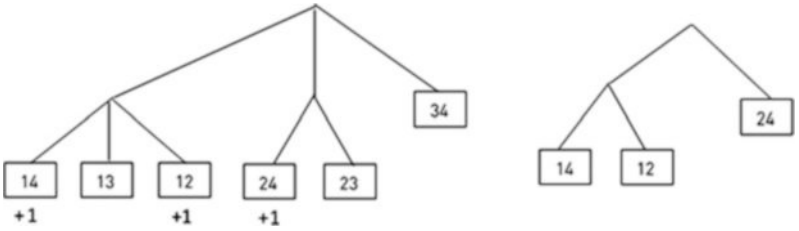
To see an example about how to perform step A.2.2.3 of the Apriori algorithm, we follow solving the problem of the basket. Following the theoretical description in this step, the results obtained in steps 2.2.1 and 2.2.2 are compared, and the numbering of the support of the candidate events is calculated. It is interesting to remember here that for the candidate event to exceed the 50% threshold, the numerator must be equal to or greater than 3. We are going to do it one by one with the events in the sample:

For Dimension 2, the comparison is as follows:

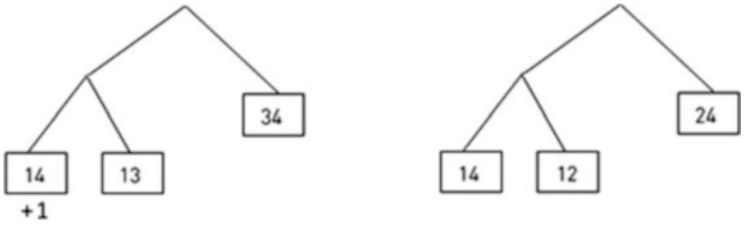
- 1245. Compared with the candidate events, sets 14, 12, and 24 add 1 to the numerator of the support calculation. Event 14 = 1; Event 12 = 1; Event 24 = 1.



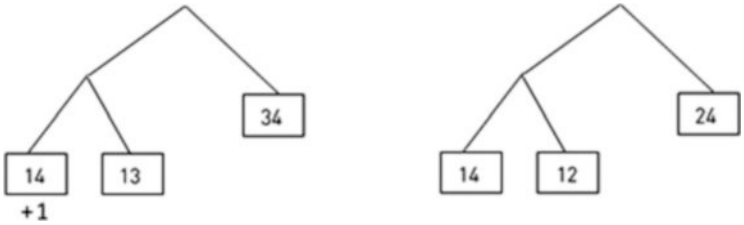
- 1234. Compared with the candidate events, sets 14, 12, and 24 add 1 to the numerator of the support calculation. Event 14 = 2; Event 12 = 2; Event 24 = 2.



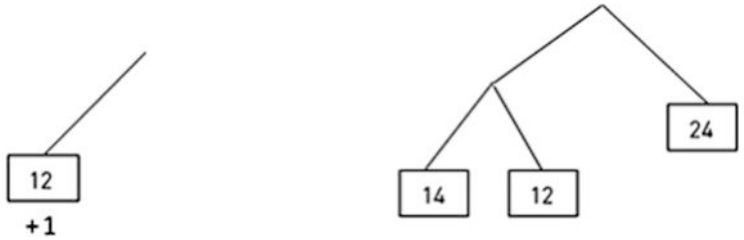
- 134. Compared with the candidate events, sets 14, 13, and 34 add 1 to the numerator of the support calculation. Event 14 = 3; Event 12 = 3; Event 24 = 3.



- 134. Compared with the candidate events, we have that set 14 adds 1 to the numerator of the support calculation. Event 14 = 4.



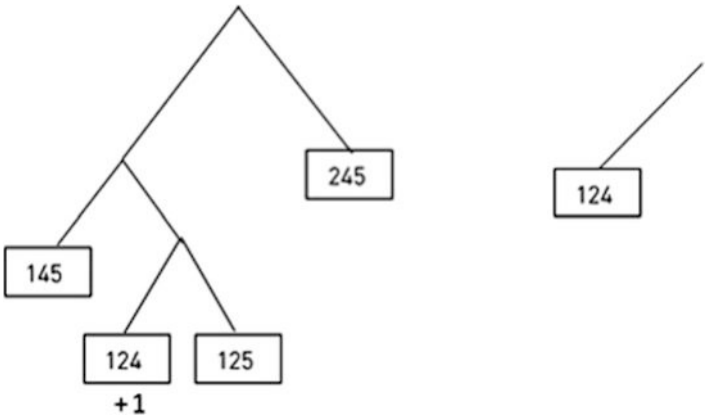
- 12. Compared with the candidate events, set 12 adds 1 to the numerator of the support calculation. Event 12 = 4.



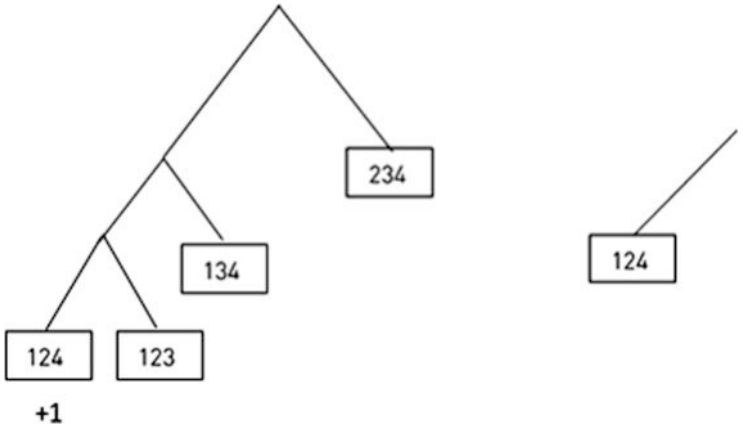
Consequently, all three candidate events exceed the support threshold and are valid.

We are now going to do the same with the events of dimension 3. For dimension 3, the comparison is as follows:

- 1245. Comparing with the candidate event, the set 124 adds 1 to the numerator of the support calculation. Event 124 = 1.



- 1234. Comparing with the candidate event, the set 124 adds 1 to the numerator of the support calculation. Event 124 = 2.



- 124. Comparing with the candidate event, the set 124 adds 1 to the numerator of the support calculation. Event 124 = 3.

- 134. Comparing with the candidate event, the set 123 does not add anything to the numerator of the support calculation. Event 124 = 3.

Consequently, the candidate event exceeds the support threshold and is valid.

With this, step 2.2.3 is completed, and consequently step 2.2, step 2 and step A, although before going on to see how step B is carried out, we are going to analyse to what extent the a priori algorithm has reduced the search space. We can see that for the first step, it does not reduce it at all, whether or not we use the algorithm we would have to observe $\binom{5}{1} = 5$ elementary events. However, in the second step, instead of having to study the support of $\binom{5}{2} = 10$: two dimensional sets, we proceed to study events, which are {14}, {12}, and {24}, since we have eliminated two elemental events and the initial set has gone from 5 $\binom{5}{3} = 10$ to 3 and possible three $\binom{3}{3} = 1$ dimensional events and we also avoid $\binom{5}{4} = 5$ four-dimensional events as they are not possible if the algorithm is used, so we go from $\binom{5}{4} = 5$ to 0 analysis, and the same for five dimensions, and we do not analyse the empty set because there are no associations. Even if there are five elements, it is not possible in any case, with or without the use of the algorithm, a set of five elements because there is none of that dimension in the sample and consequently its probability or support is zero.

With all this, it can be observed that, with the use of the a priori algorithm, we have gone from having to observe and analyse 5 + 10 + 10 + 5 + 1 = 30 events to analyse 5 + 3 + 1 + 0 + 0 = 9 events. The optimization provided by the algorithm is very evident. If this is so for this simple case, it will be much more significant in the analysis of complex cases.

Step B

In Step B, we will identify the associated events, that is, frequent events with a confidence greater than or equal to the threshold.

Once step A has been carried out, in which the candidate events that did not reach the chosen support threshold have been eliminated, work continues with those that have exceeded it to establish the existing associations between them, which will be, as already explained above, those that meet or exceed the confidence threshold set. It is important to remember here that associations, in addition to identifying which elementary events are related, also impose a meaning on the association, as we have seen in the example of trust.

For each selected event of dimension k,

$$2^k - 2$$

possible associations, since all these possible associations will exceed the support threshold because all of them are defined on selected sets because they exceeded the support threshold.

In this example that we are carrying out, we state which for possible associations we must calculate the confidence, and for the theory, we know that we must study the events selected for exceeding the support threshold, that are:

$$\{\{1,2\},\{1,4\},\{2,4\},\{124\}\}$$

or whatever same

{{Bread, Water}, {Bread, Milk}, {Water, Milk}, {Bread, Water, Milk}}.

Therefore, all the associations established between these sets or subsets of them will have a support threshold equal to or greater than 50%, as they come from sets that do.

To calculate the confidence of each selected event, the a priori algorithm uses the function called *Ap-genrules,* and from the results, it eliminates all the associations that do not reach the confidence threshold.

The *Ap-genrules* function is based on the following theorem: Let A and B be two sets; if the association $A \rightarrow B - A$ does not exceed the confidence threshold, then any association $A' \rightarrow B - A'$, where A' is any subset of A, ($A' \subseteq A$), will also not reach the confidence threshold. As a starting set B, the a priori algorithm uses the one formed by the selected events of greater dimension.

In this example, we calculate the confidence of the selected events by applying the Ap-genrules function. From the theory, we know that the confidence threshold must be fixed arbitrarily, and in this case, the confidence threshold for associations will be set at 80%; that is, only those that reach this threshold or higher will be accepted.

The associations that can be established from the selected sets are for those of dimension 2. 2^2-2 = 2, for each set. The possible rules are as follows:

{Bread} → {Water}, {Water} → {Pan}, {Pan} → {Milk}, {Milk} → {Bread}, {Water} → {Milk}, {Milk} → {Water}.

For dimension 3. 2^3-2 = 6, for each set. The possible rules are as follows:

{Water, Milk} → {Bread}, {Bread, Milk} → {Water}, {Bread, Water} → {Milk}, {Milk} → {Bread, Water}, {Water} → {Bread, Milk}, {Bread} → {Water, Milk}.

We apply the a priori algorithm and the ap-genrules function to determine which association rules exceed the confidence threshold for the possible associations

identified. We take as starting associations {Bread, Water} → {Milk}, where A = {Bread, Water}; {Bread, Milk} → {Water}, where A = {Bread, Milk}; {Water, Milk} → {Bread}, where A = {Water, Milk}, and we calculate the confidences of each of these associations.

- {Bread, Water} → {Milk}

We will analyse the confidence of this possible association with this sense of the association and all the related associations with each own sense using the function ap-genrules.

The B set is {Bread, Water, Milk}

From the possible subsets of B, we take as A set {Bread, Water} and obtain $B - A$, that, as we know for the theory of set seen in the Probability chapter, is $B - A =$ {Milk}.

Now, we calculate the confidence of the association {Bread, Water} → {Milk} that comes from the equation. To do it easily, we remember that the six events in the set of samples were {Bread, Water, Milk, Oranges}, {Bread, Water, Coffee, Milk}, {Bread, Water, Milk}, {Bread, Coffee, Milk}, {Bread, Water}, {Milk},

The confidence of this association is:

$$c(\{\text{Bread}, \text{Water}\} \cup \{\text{Milk}\}) = \frac{n_{\{\text{Bread,Water}\}\cup\{\text{Milk}\}}}{n_{\{\text{Bread,Water}\}}} = \frac{3}{4} = 0.75$$

Since it is not greater than 0.8, which is the threshold, this association is not selected, and applying the ap-genrules function, the associations $A' \rightarrow B - A'$, where A' is any subset of A, ($A' \subseteq A$), will not reach it either.

A' are subsets of A, they are:

$$A'_1 = \{\text{Bread}\}, A'_2 = \{\text{Water}\}$$

Consequently, the associations

$A'_1 \rightarrow B - A'_1$ that it is {Bread} → {Water, Milk}
$A'_2 \rightarrow B - A'_2$ that it is {Water} → {Bread, Milk}

Will not reach the confidence threshold. We calculate both confidences to prove this.

$$cA'_1 = c(\{\text{Bread}\} \cup \{\text{Water}, \text{Milk}\}) = \frac{n_B}{n_{A'_1}} = \frac{n_{\{\text{Bread}\}\cup\{\text{Water,Milk}\}}}{n_{\{\text{Bread}\}}} = \frac{3}{5} = 0.6$$

And

$$cA'_2 = c(\{\text{Water}\} \cup \{\text{Bread}, \text{Milk}\}) = \frac{n_B}{n_{A'_1}} = \frac{n_{\{\text{Water}\}\cup\{\text{Bread,Milk}\}}}{n_{\{\text{Water}\}}} = \frac{3}{4} = 0.75$$

Both of them are under 0.8, so they do not reach the threshold and are not selected.

- {Bread, Milk} → {Water}

We will analyse the confidence of this possible association with this sense of the association and all the related associations with each own sense using the function ap-genrules.

The B set is, as in the previous case, {Bread, Water, Milk}

From the possible subsets of B we take as A set {Bread, Milk} and obtain $B - A$, that, as we know for the theory of set seen in the Probability chapter, is $B - A =$ {Water}.

Now, we calculate the confidence of the association {Bread, Milk} → {Water} that comes from the equation. To do it easily, we remember that the six events in the set of samples were {Bread, Water, Milk, Oranges}, {Bread, Water, Coffee, Milk}, {Bread, Water, Milk}, {Bread, Coffee, Milk}, {Bread, Water}, {Milk}.

The confidence of this association is:

$$c(\{\text{Bread, Milk}\} \cup \{\text{Water}\}) = \frac{n_{\{\text{Bread,Milk}\}\cup\{\text{Water}\}}}{n_{\{\text{Bread,Milk}\}}} = \frac{3}{4} = 0.75$$

Since it is not greater than 0.8, which is the threshold, this association is not selected, and applying the ap-genrules function, the associations $A' \rightarrow B - A'$, where A' is any subset of A, ($A' \subseteq A$), will not reach it either.

A' are the subsets of A, they are:

$$A'_1 = \{\text{Bread}\}, A'_2 = \{\text{Milk}\}$$

Consequently, the associations

$A'_1 \rightarrow B - A'_1$ that it is {Bread} → {Water, Milk}

$A'_2 \rightarrow B - A'_2$ that it is {Milk} → {Water, Bread}

will not reach the confidence threshold. We do not need to calculate the first one because it is exactly the same as in the previous case, and we know that it is:

$$cA'_1 = 0.6$$

We calculate the second confidence to prove this.

$$cA'_2 = c(\{\text{Milk}\} \cup \{\text{Water, Bread}\}) = \frac{n_B}{n_{A'_2}} = \frac{n_{\{\text{Water}\}\cup\{\text{Bread,Milk}\}}}{n_{\{\text{Water}\}}} = \frac{3}{5} = 0.6$$

Both of them are under 0.8, so they do not reach the threshold and are not selected.

- {Water, Milk} → {Bread}

This is the last possible set A that we can select, and we are going to analyse the confidence of this last possible association with this sense of the association and all the related associations with each own sense using the function ap-genrules.

The B set is, as in the previous cases, {Bread, Water, Milk}

From the possible subsets of B, we take as A set {Water, Milk} and obtain $B - A$, that, as we know for the theory of set seen in the Probability chapter, is $B - A =$ {Bread}.

Now, we calculate the confidence of the association {Water, Milk} → {Bread} that comes from the equation. To do it easily, we remember that the six events in the set of samples were {Bread, Water, Milk, Oranges}, {Bread, Water, Coffee, Milk}, {Bread, Water, Milk}, {Bread, Coffee, Milk}, {Bread, Water}, {Milk}.

The confidence of this association is:

$$c(\{\text{Water}, \text{Milk}\} \cup \{\text{Bread}\}) = \frac{n_{\{\text{Water,Milk}\}\cup\{\text{Bread}\}}}{n_{\{\text{Water,Milk}\}}} = \frac{3}{3} = 1$$

Since this case is different from the previous case and is greater than 0.8, which is the threshold, this association can be selected, and consequently, in this case, we cannot apply the ap-genrules function and the associations $A' \rightarrow B - A'$, where A' is any subset of A, ($A' \subseteq A$) that must be analyzed one by one.

A' are the subsets of A, they are:

$$A'_1 = \{\text{Water}\}, A'_2 = \{\text{Milk}\}$$

Consequently, the associations

$A'_1 \rightarrow B - A'_1$ that it is {Water} → {Bread, Milk}
$A'_2 \rightarrow B - A'_2$ that it is {Milk} → {Water, Bread}

We do not need to calculate the first one because it is exactly the same as the cA'_2 of the first case, and we know that it is:

$$cA'_1 = 0.75$$

We do not need to calculate the first one because it is exactly the same as the cA'_2 of the previous case, and we know that it is:

$$cA'_2 = 0.6$$

Both of them are under 0.8, so they do not reach the threshold and are not selected.

We have finished the analysis of all the possible associations of dimension 3, for which we have selected only the association

$$\{\text{Water}, \text{Milk}\} \rightarrow \{\text{Bread}\}$$

Because it is the only one with a confidence over the threshold fixed of 0.8. It is therefore the only association selected because it has 50% support and 80% confidence. In other words, whenever you buy Water and Milk, you also buy Bread.

We now take the sets of dimension 2.

We start with B = {Bread, Water}

We take A = {Bread}

If we apply the first premise: $A \rightarrow B - A$, we obtain that $B - A = \{\text{Water}\}$. Therefore, we have the association:

{Bread} → {Water}

To calculate its confidence, we remember the sample to do it easily: {Bread, Water, Milk, Oranges}, {Bread, Water, Coffee, Milk}, {Bread, Water, Milk}, {Bread, Coffee, Milk}, {Bread, Water}, {Milk}, and we have:

$$c\text{A} = c(\{\text{Bread}\} \cup \{\text{Water}\}) = \frac{n_B}{n_A} = \frac{n_{\{\text{Bread}\}\cup\{\text{Water}\}}}{n_{\{\text{Bread}\}}} = \frac{4}{5} = 0.8$$

In this case, the confidence of the association exceeds the confidence threshold, so in addition to accepting the association, we cannot apply the second premise, and we must analyse $A' \rightarrow B - A'$ one by one. However, we realize that in dimension 2, there is no A' because there are no subsets of A, so all cases have to be analysed, and we have:

{Water} → {Bread}: $c = \frac{n_B}{n_A} = \frac{4}{4} = 1 > 0.8$. It is accepted
{Bread} → {Milk}: $c = \frac{n_B}{n_A} = \frac{4}{5} = 0.8 = 0.8$. It is accepted
{Milk} → {Bread}: $c = \frac{n_B}{n_A} = \frac{4}{5} = 0.8 = 0.8$. It is accepted
{Water} → {Milk}: $c = \frac{n_B}{n_A} = \frac{3}{4} = 0.75 < 0.8$. It is not accepted
{Milk} → {water}: $c = \frac{n_B}{n_A} = \frac{3}{5} = 0.6 < 0.8$. It is not accepted

Consequently, as a result of the association analysis performed, the associations identified because all of them present support equal to or over 50% and confidence equal to or over 80% are as follows:

- {Water, Milk} → {Bread}
- {Bread} → {Water}
- {Water} → {Bread}
- {Bread} → {Milk}
- {Milk} → {Bread}

In other words, whenever you buy Water and Milk, you also buy Bread. Whenever you buy Bread, you buy Water, and vice versa; and whenever Bread is bought, Milk is bought and vice versa.

B. Computer-Based Solving

As in the other chapters, this subsubsection starts with a reminder of what means computer-based association solving, that is, the application of a systematic process of designing, implementing, and using programming tools to solve the association problem.

Exercises of Association Analysis Solved in R

In this exercise, an analysis with R of event association will be carried out for the same problem of the shopping basket applying all the concepts seen in the topic, that is, we will obtain, using the Apriori algorithm, events whose support is equal to or greater than 50%. For the associations with support above the threshold, the confidence is higher than 80%. To do that and with the goal that the problem to be solved was exactly the same as we have used in the theoretical solution of the problem, that sample is {Bread, Water, Milk, Oranges}, {Bread, Water, Coffee, Milk}, {Bread, Water, Milk}, {Bread, Coffee, Milk}, {Bread, Water}, {Milk}.

As we know from the previous chapters, in R, with the packages loaded by default, association analysis cannot be performed, so we will have to load a package that does allow it. Among the various existing packages in the CRAN, we will use the *arules* package in this practice.

The first thing we do is check if we have it among the standard library packages, for which we know that we must use the function library() that gives us the list of the packages that we have in the standard library, and we see that it is not among them, so we have to install it.

We know that there are many different alternatives to install a package; we are going to use one of them, that although it is not the shortest, it will allow us to see more additional options. We go to the arules package page on the CRAN website. For this, we click on the link: http://CRAN.R-project.org/within the third subsubsection 3. Archives, and we go to a new page where we can find all the downloadable files for R. We click on the Packages link, and we get to a new page where all the packages available for R are included. We click on Table of available packages, sorted by name and search the arules package.

As we know, in this new page, all the information about the package and the downloadable is gathered. It is very important to know that each package has a page of this type because it is important not to load them blindly without knowing anything about them. We download the file:

Windows binaries: arules_1.1-6.zip

if we are working on windows. In addition, it is important to also download the package manual so that you can consult it. Reference manual:

arules.pdf.

We download both things in the downloads folder and return to R. To install the package we are going to use the known install.packages () function in which we are going to put two arguments: the first is which .zip file we want to install. For R to install it, the.zip file must be in a temporary directory called tmp in the root directory of the hard disk, so we create it and place the file arules_1.1-6.zip there. The second argument consists of giving the variable repos the value NULL. Consequently, the function is:

>install.packages ("c:/tmp/arules_1.1-6.zip", repos = NULL).

When we run the R function, it tells us that it cannot store arules in the standard R library, and in the case that we had not installed other packages outside the standard library and we would have a personal library, it gives us the option to use a personal library instead. We say yes and a new window appears asking if we want to create the library: c:/users/jjcg/documents/R/win-library/3.1. We say yes, and it creates all the folders that do not exist and installs in one folder inside 3.1, which names arules, the arules files (we remove the tmp because we will not need it anymore and delete download arules). We move the manual to the arules folder to have everything concentrated. Next, we load the arules package in R using the instruction

>library(arules)

R loads, in addition to the arules package, the matrix package (which is in the standard library and does not have to be installed) because the arules package needs it to work. We execute the function

>search ()

to verify that it is installed correctly.

Since we are going to use the package arules many times, we do not want to load it any time that we use R. For that reason, we want that the package was loaded by default by R when it starts.

When we start R, there is a set of packages that are loaded by default. To determine which packages are, we use the function

>getOption ("defaultPackages")

This set of initially loaded packages can be modified by reprogramming the start code. The file that controls this startup code is Rprofile, and it is, in a computer with Windows operating system, in the folder:

Program Files/R/R-3.1.2/library/base/
In this file, there is a piece of code that is

dp <- c ("datasets "," utils "," grDevices "," graphics "," stats "," methods ")

In this variable, we can include or remove the packages we want. If we removed them all, only the “base” package would remain, which is not listed because it cannot stop loading for the system to work. We are going to include the *arules* package

within the default packages because we use it a lot, as we have seen in previous chapters. For which we introduce it after "methods":

```
dp <- c ("datasets", "utils", "grDevices", "graphics", "stats", "methods", "arules")
```

We close the file, maintain the new name and now, each time that the R program starts, will load the arules package by default.

Once we have the arules package installed and loaded by default, we begin to solve the association analysis problem with R.

The first thing we have to do is to introduce the values of the events of the sample with which we are working in R. To do this, we can realize that the events can be represented as an array of binary values in the following way: in each column, the values of the different elementary events with which we work are represented, and in each row the values of a given event (observation), being a 1, if the elementary event occurs, or a 0, if it does not occur.

According to this, the matrix corresponding to the observed sample {Bread, Water, Milk, Oranges}, {Bread, Water, Coffee, Milk}, {Bread, Water, Milk}, {Bread, Coffee, Milk}, {Bread, Water}, {Milk} is:

	Bread	Water	Coffee	Milk	Oranges
Event 1	1	1	0	1	1
Event 2	1	1	1	1	0
Event 3	1	1	0	1	0
Event 4	1	0	1	1	0
Event 5	1	1	0	0	0
Event 6	0	0	0	1	0

We introduce this matrix in R with the Matrix () function. Let us call it a sample matrix, with what the instruction would be:

```
sample <- Matrix (c (1,1,0,1,1, 1,1,1,1,0, 1,1,0,1, 0, 1,0,1,1,0, 1,1,0,0,0, 0,0,0,1,0),
    6, 5, byrow = TRUE, dimnames = list (c ("event1", "event2", "event3", "event4",
    "event5", "event6"), c ("Bread", "Water", "Coffee", "Milk", "Oranges")), sparse
    = TRUE)
```

We have introduced the attribute sparse = TRUE because rules need to work a ngCMatrix or sparse matrix of nonzero positions, so we will have to convert the sample matrix into that type of matrix, and if we had not put the sparse attribute, we could not because the function as, which is the one that will convert the sample matrix into a ngCMatrix, can only do so if the sample matrix is sparse (what we have is a 6 x 5 sparse Matrix of class "dgCMatrix"), if not (what we would have without that attribute would be 6 x 5 Matrix of class "dgeMatrix") could not. The complete statement for converting to a ngCMatrix will be:

```
sample-trangCMatrix <-as (sample, "nsparseMatrix")
```

Then, before applying the arules functions, we must transpose the samplengCMatrix matrix to correctly analyse the sets. We do it through the t function.

Trans-pmsampleCMatrix <t (samplengCMatrix)

Then, we performed association analysis using the functions of arules. The first thing is to determine the possible transactions, which we do using the as function and the transactions attribute. We will assign the result to a new variable that we will call tr. The complete statement is:

tr <-as (Transpm SampleCMatrix, "transactions")

We observe the result by introducing tr. Next, we ask for a summary of tr using the summary (tr) command.

After these previous steps, to properly carry out the association analysis, we use the a priori algorithm through the function of arules apriori (as seen in theory). We define a variable as the output of the associations obtained, which we call variable rules. The complete instructions are as follows:

rules <-apriori (tr, parameter = list (support = 0.5, confidence = 0.8))

We have chosen, as in the manual exercise, a support of 50% and a confidence of 80%. Next, if we introduce rules to see the result, we find a set of 2 rules. To see what they are, we introduce the inspect function. The complete statement is:

inspect (rules)

and we obtain:

lhs, rhs, support, confidence, lift;

[1] {} => {Milk} 0.8333333 0.8333333 1.00
[2] {} => {Bread} 0.8333333 0.8333333 1.00
[3] {Water} => {Bread} 0.6666667 1.0000000 1.20
[4] {Bread} => {Water} 0.6666667 0.8000000 1.20
[5] {Milk} => {Bread} 0.6666667 0.8000000 0.96
[6] {Bread} => {Milk} 0.6666667 0.8000000 0.96
[7] {Water, Milk} => {Bread} 0.5000000 1.0000000 1.20

That is the same result that we have obtained in the theoretical subsubsection.

C. Association Analysis Exercises Solved

This subsection has two parts. In the first part, a set of exercises solved in detail are presented to allow you to check if all the knowledge has been correctly acquired. The advice is to try to solve the exercises by yourself, and then to get the solution to check it with the proposed one by this book. This procedure will make this subsection truly useful for you. In the second part, the same exercises will be solved in R.

Handmade Exercises

1. A sample space is established based on the real characteristics of planets of the solar system that can serve as the basis for an association analysis for two events composed of a single elementary event.

   In this first exercise, we use the following sample space: E = {Solid, Gaseous, With satellites, Without satellites}, which refers to the composition of a planet and the fact that said planet has satellites or not. As has been said in the theoretical description, we will begin to see the association using sample spaces composed of subsets formed by exclusive elementary events, and in this case, there are two subsets with exclusive elements: {Solid, Gaseous} and {With satellites, Without satellites}.
2. Establish the set parts of E, P(E), of the sample space stated in the previous exercise for the association of two events composed of a single elementary event.

   Starting from the sample space E = {Solid, Gaseous, With satellites, Without satellites}, and taking into account that we have the two subsets with exclusive elements {Solid, Gaseous} and {With satellites, Without satellites}, the set P (E) will be formed by the following sets: P (E) = {∅, {Solid}, {Gaseous}, {With satellites}, {Without satellites}, {Solid, With satellites}, {Solid, Without satellites}, {Gaseous, With satellites}, {Gaseous, Without satellites}}. And from here on, the association analysis that we are going to be able to do will be able to be only for those events formed by a single elementary event, which are not exclusive, that is, they have a union event within the set P (E), that is, we can study the association of Solid with satellites, but not of Solid with Gas.
3. Stablish a possible sample of observations, based on the set P(E) stated in the previous exercise that can serve to carry out an association analysis of two events composed of a single elementary event.

   As mentioned in the introduction, to carry out an association analysis it is essential to have a sample that allows us to calculate the values of the measures used to determine the degree of association. In this case, the sample we will have is the eight planets of the solar system:[22] {Mercury[23] {Solid, Without[24]}, Venus {Solid, Without}, Earth {Solid, With}, Mars {Solid, With}, Jupiter {Gaseous, With}, Saturn {Gaseous, With}, Uranus {Gaseous, With}, Neptune {Gaseous, With}}.
4. Using the sample of the previous exercise, calculate the support of the association of the possible disjoint events in the problem of the composition of the planets and the existence of satellites for them.

   Once we have defined the sample that allows us to establish the probability of appearance of the different events, we calculate the support of the association of

[22] All the data are real data.

[23] The name of the planet will only be used to identify the event, but will not be considered as an additional variable.

[24] Only Without or With will be put, without the word satellites to make the text easier to read.

the disjoint events A1 = {Solid} and A2 = {Without}. What we have to calculate is the classical probability of appearance of the set = {Solid, Without}, as a set formed only by these elements or within as a subset of any set of P (E) that contains, among others, said elements, but in this case, that possibility will not be given. As seen above, this probability is given by:

Examining the sample, the number of elements of P (E) that the set contains are 2: Mercury {Solid, Without}, Venus {Solid, Without}, and the total number of events in the sample is = 8; consequently, the support of the association of the events A1 = {Solid} and A2 = {Without} is 2/8 = 0.25. Therefore, this association would have 25% support. As it is a probability, the support moves in a range from 0 to 1. Once we have seen how it is calculated and what is the value of the support of the association {Solid, Without}, we are going to calculate the support of the rest of possible associations: {Solid, With}, with a support of 2/8, and therefore again of 25%; {Gaseous, Without}, with a support of 0/8, and therefore 0%; {Gaseous, With} with a support of 4/8 and therefore 50%.

If we had established 25% support to determine which associations we would continue to analyse, we would continue to analyse the associations.

5. Calculate the confidence of the association of the selected disjoint events from the value of their support calculated in the previous example, which are {Solid, Without}; {Solid, With}; {Gaseous, With}.

   What we have to calculate is the probability of appearance, for example, of the set {Solid, Without}, but unlike the previous example, we do not calculate it on the complete sample but taking a subset of it formed only by those events that contain or are equal to the set: A1 = {Solid}. The reduced sample is composed of four events: Mercury {Solid, Without}, Venus {Solid, Without}, Earth {Solid, With}, and Mars {Solid, With}. Therefore, to calculate the confidence, we use the equation:

   As we know from the previous example, the number of elements of P (E) that contain the set {Solid, Without} are 2: Mercury {Solid, Without}, Venus {Solid, Without}, and the total number of events in the reduced sample is = 4; consequently, the confidence of the association of the events A1 = {Solid} → A2 = {Without} is 2/4 = 0.5. Again, since it is a probability, the support moves in a range that goes from 0 to 1; therefore, this association would have a 50% confidence level.

   We are now going to calculate the confidence of the reverse associations. The confidence obtained for the association A1 = {Solid} → A2 = {Without}, 50%, is not the same as the one obtained would be the same for the association: A1 = {Without} → A2 = {Solid}, since the latter would keep the numerator, but the denominator would be 2, since the events of the reduced sample, that is, the events that contain milk are 5, and not 4 as in the previous case, these events are: Mercury {Solid, Without}, Venus {Solid, Without}, with which the confidence is c = 2/2 = 1 = 100%.

6. Generate the contingency table of the elementary events treated in the previous exercises and calculate the contingency of Yule and Pearson.

In each column, we write one of the two exclusive elementary events belonging to the first subset of the sample space, that is, A1 {Solid} and A2[25] {Gaseous}. In each row, we write the other two exclusive elementary events, that is, B1 {Without} and B2 {With}, in each cell, we write the absolute frequency of appearance of the union set of both elementary events. The table is:

2 × 2 crosstab or contingency table

	A_1	A_2	Total
B_1	f_{11}	f_{12}	$f_{11}+f_{12}$
B_2	f_{21}	f_{22}	$f_{21}+f_{22}$
Total	$f_{11}+f_{21}$	$f_{12}+f_{22}$	$f_{11}+f_{12}+f_{21}+f_{22}$

Contingency table with the problem values

	Solid	Gaseous	Total
Without	2	0	2
With	2	4	6
Total	4	4	8

If we recall what was seen in the theoretical description of the contingency, from observing the values in the table, it could be concluded that if the event composed of the only elementary event {Without} was independent of the value of the event composed of only the element {Solid}, the relative frequency of appearance of {Without}, that is, of planets without satellites, should be kept constant when calculating the relative frequency of {Without} when the event {Solid} also occurs, that is, when the relative frequency of appearance of planets without satellites between the solid planets is calculated. This is:

$$\frac{f_{11}}{f_{11}+f_{21}} = \frac{f_{11}+f_{12}}{f_{11}+f_{12}+f_{21}+f_{22}}$$

Substituting the values, we have:

$$\frac{2}{2+2} = 0,5 \neq \frac{2+0}{2+0+2+4} = 0,25$$

Therefore, there is a dependency relationship between being a solid planet and having satellites. Later, it will be seen that type.

Once these calculations have been carried out, it is very important to note that, when performing them, we have also obtained the value that, f*rom the contingency calculation, the confidence threshold should have* for the possible associations of the event {Without} with the rest of events, {Solid} or {Gaseous}, which should be

[25] We changed the numbering when naming the events for clarity. Event numbering is arbitrary and can be changed at any time.

25%. That is, if trust had a higher value, there would be an association between the events.

If the above equation is taken and the value f_{11} is cleared, the theoretical frequency f'_{11} is obtained, which should have the union sequence {Without, Solid} for these values to be independent. It is given by the equation:

$$f'_{11} = \frac{(f_{11} + f_{21})(f_{11} + f_{12})}{f_{11} + f_{12} + f_{21} + f_{22}}$$

If the result obtained for the pair of values {Without, Solid} $f'_{11} = 1$ and $f_{11} = 2$ is observed, it can be concluded, as seen in the theoretical description of the contingency, that there is an association relationship between {Without, Solid}, and taking into account that $2 > 1$, that is, $f'_{11} > f_{11}$, it is concluded that it is attraction. This means that planets without satellites are associated or tend to be solid.

To reinforce the understanding of the concept of contingency, we now analyse the association of the events {With} and {Gaseous}. In this case, we have

$$\frac{f_{22}}{f_{12} + f_{22}} = \frac{f_{21} + f_{22}}{f_{11} + f_{12} + f_{21} + f_{22}}$$

Substituting the values, we have:

$$\frac{4}{0+4} = 1 \neq \frac{2+4}{2+0+2+4} = 0,75$$

Therefore, there is a strong attraction dependency relationship between having satellites and being a gaseous planet. This means that gaseous planets often have satellites.

Having seen the case of a 2 × 2 contingency table, we are now going to see an example, based again on real characteristics of the planets of the solar system, of a mxn contingency table. m will remain 2 since we are going to take solid and gaseous again as the two exclusive elementary events of the first subset of the sample space; n will be 4 since we are going to take 4 main chemical compositions of atmospheric gas, K, CO_2, N, and H, that is, Potassium, Carbon Dioxide, Nitrogen, and Hydrogen. The contingency table is:

Crosstabulation or contingency table

	K	CO_2	N	H	Total
Solid	1	2	1	0	4
Gaseous	0	0	0	4	4
Total	1	2	1	4	8

In this case, we are going to analyse the association between the events formed by the elementary events {H} and {Solid}, that is, we are going to try to observe whether a planet has independent hydrogen as the main component of its atmosphere

or not that the planet is solid. To do this to the values a1 (composition of the solid planet) and b4 (main component of the hydrogen atmosphere) of the said table, the theoretical descriptions seen in for the concept of contingency in mxn tables are applied, in this case 2 × 4, and the value is obtained for contingency:

$$f'_{14} = \frac{\sum_{j=1}^{4} f_{1j} \sum_{i=1}^{2} f_{i4}}{\sum_{\substack{i=1 \\ j=1}}^{2,4} f_{ij}} = \frac{(1+2+1+0)(0+4)}{1+2+1+4} = 2$$

As $f_{14} = 0$, this implies that $f_{14} < f'_{14}$, and therefore, there is a repulsion association between the solid composition of the planet and that its main component is hydrogen; since as it happened for the 2 × 2 tables, for the mxn tables it is verified that, if $f_{pq} < f'_{\mathrm{pq}}$ you have a dependence on repulsion and if you have a dependence on attraction. Consequently, the analysis carried out means that solid planets tend not to have atmospheres whose main component is hydrogen.

We will analyse if the main component of its atmosphere carbon dioxide is independent or not of the planet being solid. If the contingency calculation equation is applied to the values (Solid} → {CO_2), the following is obtained:

$$f'_{12} = \frac{\sum_{j=1}^{4} f_{1j} \sum_{i=1}^{2} f_{i2}}{\sum_{\substack{i=1 \\ j=1}}^{2,4} f_{ij}} = \frac{(1+2+1+0)(2+0)}{1+2+1+4} = 1$$

As $f_{12} = 2$, this implies that $f_{12} < f'_{12}$, and therefore there is dependence between the solid composition of the planet and that its main component is hydrogen. It is a dependency of attraction. This means that solid planets tend to have atmospheres whose main component is carbon dioxide.

We are now going to analyse if there is an association between the elemental event {Gaseous} and the elemental event {CO_2}. We apply the equation:

$$f'_{22} = \frac{\sum_{j=1}^{4} f_{2j} \sum_{i=1}^{2} f_{i2}}{\sum_{\substack{i=1 \\ j=1}}^{2,4} f_{ij}} = \frac{(0+0+0+4)(2+0)}{1+2+1+4} = 1$$

As $f_{22} = 0$, this implies that $f_{22} < f'_{22}$, and therefore, there is a dependency between the gaseous composition of the planet and that its main component is carbon dioxide. It is a dependency of repulsion. This means that gaseous planets tend not to have atmospheres whose main component is carbon dioxide. The same results are obtained if the calculations are carried out for potassium and nitrogen.

Finally, it will be analysed whether the main component of its hydrogen atmosphere is independent or not of the planet being gaseous. If the contingency equation is applied to the values (a2, b4) and the following is obtained:

$$f'_{24} = \frac{\frac{\sum_{j=1}^{4} f_{2j} \sum_{i=1}^{2} f_{i4}}{\sum_{\substack{i=1 \\ j=1}}^{4} f_{ij}}}{2} = \frac{(0+0+0+4)(0+4)}{1+2+1+4} = 2$$

As $f_{24} = 4$, this implies that $f_{24} < f'_{24}$, and therefore, there is a dependency between the gaseous composition of the planet and that its main component is hydrogen, which is a dependency of attraction. This means that gaseous planets tend to have atmospheres whose main component is hydrogen.

Calculation of the contingency of Yule.

To calculate the Yule contingency, it is necessary to have previously defined some characteristics that define the subsets of the sample space. If we take the sample space used in the first example of Contingency: {Solid, Gaseous, Without, With}, we can define a first characteristic, that is, Composition of the planet, which groups together the exclusive elemental events {Solid, Gaseous}, and a second, which is the possession of Satellites that groups together the exclusive elementary events {Without, With}. If we have on these two characteristics in the table of the planets of the solar system.

Contingency table with the values of the problem

		Composition		
		Solid	**Gaseous**	**Total**
Satellites	**Without**	2	0	2
	With	2	4	6
	Total	4	4	8

and we apply the equation for calculating the Yule contingency, we have:

$$Q = \frac{f_{11} f_{22} - f_{12} f_{21}}{f_{11} f_{22} + f_{12} f_{21}} = \frac{2.4 - 0.2}{2.4 + 0.2} = 1$$

This indicates that there is a strong dependence of attraction between the composition of the planets and whether or not they have satellites.

Calculation of the contingency of Pearson.

To calculate the contingency of Pearson C, we study the association between the composition of a planet and the main component of its atmosphere when both variables are analysed jointly to verify whether the main component of a planet's atmosphere is dependent on its composition, solid or gaseous. If we take the table of composition of the atmosphere and the planet and the equations for calculating the contingency of Pearson C, we have:

Crosstabulation or contingency table

		Atmosphere composition				
		K	**CO_2**	**N**	**H**	**Total**
Composition	**Solid**	1	2	1	0	4
	Gaseous	0	0	0	4	4
	Total	1	2	1	4	8

$$\chi^2 = \sum_{\substack{i=1 \\ j=1}}^{\substack{2 \\ 4}} \frac{\left(n_{ij} - n'_{ij}\right)^2}{n'_{ij}} = \frac{(1-0.5)^2}{0.5} + \frac{(2-1)^2}{1} + \frac{(1-0.5)^2}{0.5} + \frac{(0-1)^2}{1}$$
$$+ \frac{(0-0.5)^2}{0.5} + \frac{(0-0.5)^2}{0.5} + \frac{(0-0.5)^2}{0.5} + \frac{(4-2)^2}{2} = 6.5$$

Once the value of $\chi^2 = 6.5$ is calculated, the value of is calculated:

$$C = \sqrt{\frac{\chi^2}{\chi^2 + \sum_{\substack{i=1 \\ j=1}}^{\substack{m \\ n}} f_{ij}}} = \sqrt{\frac{6.5^2}{6.5^2 + 8}} = 0.84$$

Therefore, a value close to one indicates a high degree of association between the two characteristics, and they present a great dependence.

7. Calculate the Spearman and Kendall correlation coefficients of the following data about the sizes of the planets of the solar system and their distances to the sun.

Ordering of the planets according to their distance from the sun and their size

Planet	Distance to the sun	Size
Mercury	1	8
Venus	2	6
Earth	3	5
Mars	4	7
Jupiter	5	1
Saturn	6	2
Uranus	7	3
Neptune	8	4

The Spearman correlation coefficient was calculated to analyse the association of two characteristics with ordinal qualitative values. To study the correlation between two ordinal qualitative characteristics we need, they must be ordered in some way, so

the characteristics used thus far in this topic cannot be used. Two new characteristics will be used, related to those previously seen, and which also correspond to planetary characteristics, for which they are easily ordered. These new charasteristics are given in the table and they are the position with respect to the sun, the closest will be the first in order; and its relative size, the largest will be the first in order (although both characteristics are based on numerical values, such as distance, when sun, and the diameter of the planet, both numbers are unknown and it is only known which planet is closer to the sun than another and which planet is larger than another, without knowing the distance to the sun or the exact size of each of them, so according to this it is qualitative and nonmeasurable characteristics).

The following table presents the values of the coefficients used in the Spearman equation.

Spearman's coefficients

Planet	a_i	b_i	$d_i = a_i - b_i$	d_i^2	f_{ij}
Mercury	1	8	−7	49	1
Venus	2	6	−4	16	1
Earth	3	5	−2	4	1
Mars	4	7	−3	9	1
Jupiter	5	1	4	16	1
Saturn	6	2	4	16	1
Uranus	7	3	4	16	1
Neptune	8	4	4	16	1
				$\sum_{i=1}^{n} d_i^2 = 142$	$\sum_{i}^{n} f_i = 8$

If the Spearman coefficient is applied, we have:

$$r_s = 1 - \frac{6\sum_{i=1}^{7} d_i^2}{\left(\sum_{i}^{7} f_i\right)^3 - \sum_{i}^{7} f_i} = 1 - \frac{6.142}{(8)^3 - 8} = 1 - 1,7 = -0,7$$

Therefore, when r_s is close to −1, there is independence between the size of the planets and their position with respect to the sun.

To calculate Kendall's correlation coefficient for the study of the association of two characteristics with ordinal qualitative values, we return from the table with the planets ordered according to their distance from the sun and their size.

If you look at the table in which the values have been ordered, from lowest to highest according to the characteristic Distance to the sun, for the value i = 3, Land, the value of the variable Size is 5. If the investments and permanence are quantified, the values of the rows greater than 3 have to be analysed, which is the one we are studying, that is, rows 4 to 8. In these rows, the values 1, 2, 3, and 4 are observed in rows 5, 6, 7, and 8, respectively. These 4 values are less than 5, which is the value of row 3, and are found in rows greater than 3, in rows 5, 6, 7, and 8. Therefore, Q3 = 4. Regarding permanence, there is only a value greater than 5 in the rows that follow

row 3, which is found in row 4, and the value is 7. Therefore, P3 = 1. The table gives all the values of Qi and Pi for all the values in the table.

Q_i and P_i values for all values in the table

Planet	Distance to the sun	Size	Qi	Pi	f_{ij}
Mercury	1	8	7	0	1
Venus	2	6	5	1	1
Earth	3	5	4	1	1
Mars	4	7	4	0	1
Jupiter	5	1	0	3	1
Saturn	6	2	0	2	1
Uranus	7	3	0	1	1
Neptune	8	4	0	0	1
			$\sum_{i=1}^{8} Q_i = 20$	$\sum_{i=1}^{8} P_i = 8$	$\sum_{i}^{n} f_i = 8$

If the values from the table are introduced into the equation, we have:

$$\tau = \frac{2\left(\sum_{i=1}^{8} P_i - \sum_{i=1}^{8} Q_i\right)}{\sum_{i}^{8} f_i \left(\sum_{i}^{8} f_i - 1\right)} = \frac{2(8-20)}{8(8-1)} = -0.43$$

Being a value closer to 0 than −1 indicates a low correlation between the values, which agrees with the conclusion obtained using the Spearman coefficient.

We use the table for Kendall's example of the association of two characteristics with ordinal qualitative values. If the Goodman-Krustal equation is applied to these values, we obtain:

$$\gamma = \frac{\left(\sum_{i=1}^{8} P_i - \sum_{i=1}^{8} Q_i\right)}{\left(\sum_{i=1}^{8} P_i + \sum_{i=1}^{8} Q_i\right)} = \frac{(8-20)}{(8+20)} = 0.43$$

Therefore, the conclusion is the same as that obtained in the two previous cases and has a low correlation.

8. Using the a priori algorithm, the associations whose support is equal to or greater than 40% and whose confidence is equal to or greater than 90% for the data the extent that customers request when buying a certain car model should be obtained. To carry out the study, the following sample is available, consisting of the extras included in 8 car sales:
   {X, C, N, B}
   {X, T, B, C}
   {N, C, X}
   {N, T, X, B}
   {X, C, B}

{N}
{X, B, C}
{T, A}
where {X: Xenon headlights, A: Alarm, T: Sunroof, N: Navigator, B: Bluetooth, C: Cruise control} are the extras that can be included in each car.

Step A.1

We begin to apply the algorithm a priori. The sample space of this analysis is {X, C, N, B, T, A}, so we apply the property that the support measure is antimonotone on said elementary events. Since the number of events in the sample is 8 and the support is 40%, the calculation equation for the support is:

$$s\left(A_i \cup A_j\right) = \frac{nA_i}{nA_j} n_T$$

We have that

$$0.4 = \frac{n_{A_i} \cup n_{A_j}}{8} \rightarrow 3.2 = n_{A_i} \cup n_{A_j}$$

Which means applied to elementary events that

$$3.2 = A_i$$

or what is the same, only those elementary events that appear in sample 4 or more times will have enough support. Therefore, they would be X, which appears 6 times; C, which appears 5 times; N, which appears 4 times; and B, which appears 5 times.

However, to reinforce the concept of calculating the support and to check that the reasoning in the previous paragraph is correct, we perform the calculation of the support of all elementary events:

$$S_X = \frac{6}{8} = 0.75 > 0.4$$
$$S_C = \frac{5}{8} = 0.625 > 0.4$$
$$S_N = \frac{4}{8} = 0.5 > 0.4$$
$$S_B = \frac{5}{8} = 0.625 > 0.4$$
$$S_T = \frac{3}{8} = 0.375 < 0.4$$
$$S_A = \frac{1}{8} = 0.125 < 0.4$$

Therefore, the selected elementary events are X, C, N, and B since they all have support equal to or greater than 40%.

Step A.2.

Applying the method $F_{k-1} \times F_{k-1}$ we determine the possible candidate sets of 2, 3, and 4 dimensions.

For 2 dimensions

K = 2 →K−1 = 1 and K−2 = 0. All those that have a single element and are different are valid, so consequently, we form all the possible two-dimensional sets that can be formed with the elementary events selected in the previous step A.1. These sets are the following six:

{X, C}
{X, N}
{X, B}
{C, N}
{C, B}
{N, B}

For 3 dimensions

K = 3 →K−1 = 2 and K−2 = 1. All those that have two elements are valid, and the first element is the same and the second is different. These sets are the following four:

{X, C} and {X, N} →{X, C, N}
{X, C} and {X, B} →{X, C, B}
{X, N} and {X, B} →{X, N, B}
{C, N} and {C, B} →{C, N, B}

For 4 dimensions

K = 4 →K−3 = 1 and K−1 = 0. Those that have three elements are valid; the first two elements are the same, and the last one is different. There is only one set that meets these conditions:

{X, C, N} and {X, C, B} →{X, C, N, B}

Step A.2.2.1

We hash the candidate events with the partition function pmod3, so the partitions that we are going to have will be {1, 4, 7}, {2, 5, 8}, {3, 6, 9}, and we change the notation of the sample space events to numerical notation as follows:

{X = 1}, {C = 2}, {N = 3}, {B = 4}, {T = 5}, {A = 6}

According to the new numerical notation, the candidate events would be:

{X, C} = {1 2}
{X, N} = {1 3}
{X, B} = {1 4}

{C, N} = {2 3}
{C, B} = {2 4}
{N, B} = {3 4}

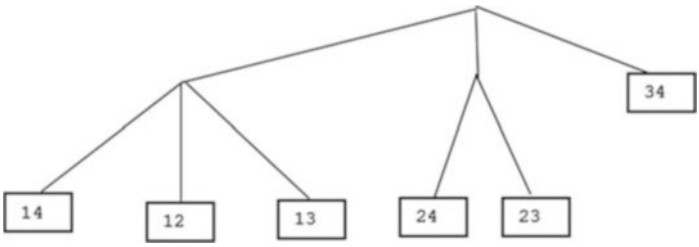
For 3 dimensions

{X, C, N} = {1 2 3}
{X, C, B} = {1 2 4}
{X, N, B} = {1 3 4}
{C, N, B} = {2 3 4}

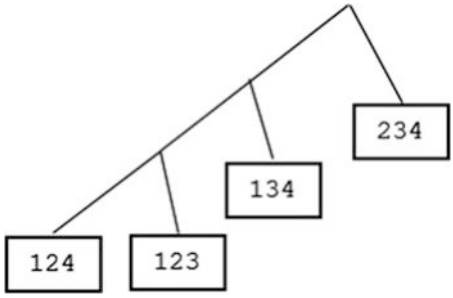
For 4 dimensions

{X, C, N, B} = {1 2 3 4}

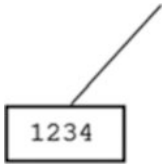
Therefore, the hash tree of the candidate events of dimension 2, {1 2}, {1 3}, {1 4}, {2 3}, {2 4}, {3 4}, would be structured as follows:



Candidate events of dimension 3, {1 2 3}, {1 2 4}, {1 3 4}, {2 3 4}, would remain.



And the dimension 4 candidate event, {1 2 3 4}, would remain.



Step A.2.2.1

In the second step, the sample events are partitioned using the same hash tree with the partition function pmod3.

According to the new numerical notation, the events in the sample would be:

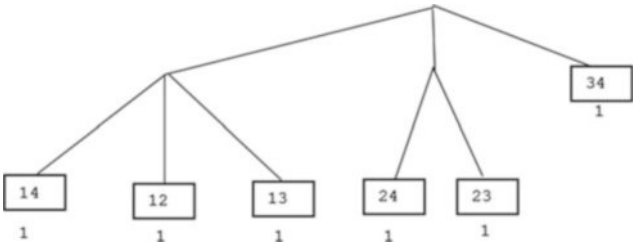
{X, C, N, B} = {1 2 3 4}
{X, T, B, C} = {1 2 5 4}
{N, C, X} = {1 2 3}
{N, T, X, B} = {1 5 3 4}
{X, C, B} = {1 2 4}
{N} = {3}
{X, B, C} = {1 2 4}
{T, A} = {5 6}

As in the example seen in the theoretical description of the topic, we are going to do it sequentially one by one and for each dimension.

Dimension 2

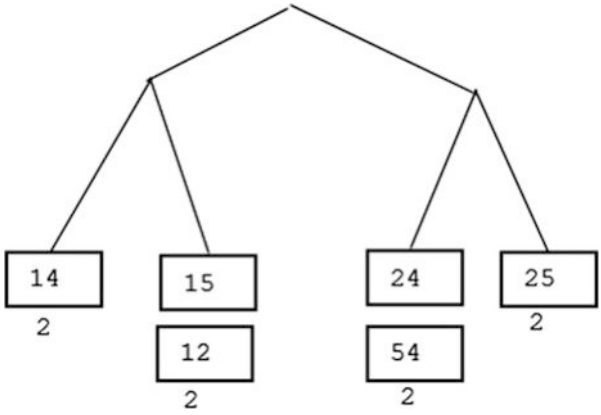
1234

From event 1234, the following events of dimension 2 can be generated, and the following partition:



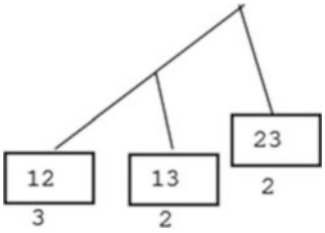
1254

From event 1254, the following events of dimension 2 can be generated, and the following partition:



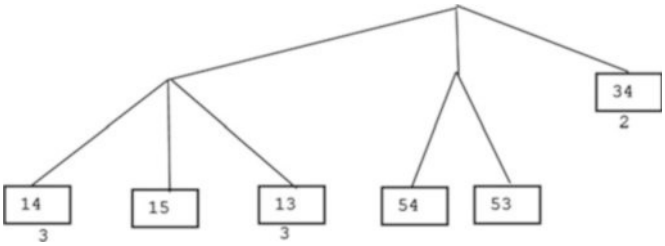
123

From event 123, the following events of dimension 2 can be generated, and the following partition:



1534

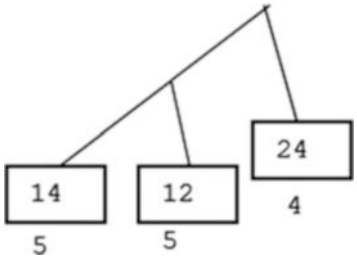
From event 1534, the following events of dimension 2 can be generated, and the following partition:



124[26]

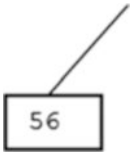
From event 124, the following event of dimension 2 can be generated, and the following partition:

[26] Do not forget that this event appears twice.



56

From event 56, the following event of dimension 2 can be generated, and the following partition:



3

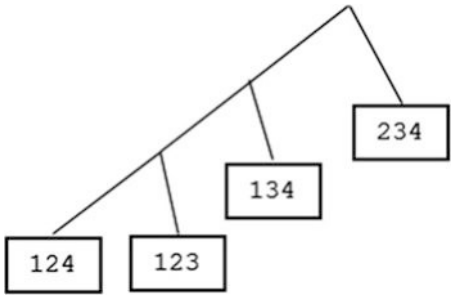
From event 3, no event of dimension 2 can be generated.

Dimension 3

We are now going to do the same with events of dimension 3. The partitions obtained are:

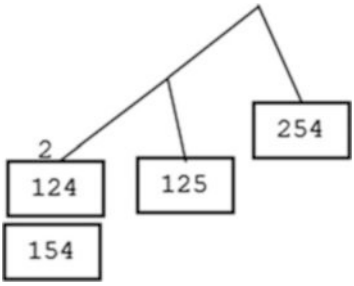
1234

From event 1234, the following events of dimension 3 can be generated, and the following partition:



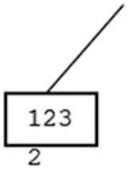
1254

From event 1254, the following events of dimension 3 can be generated, and the following partition:



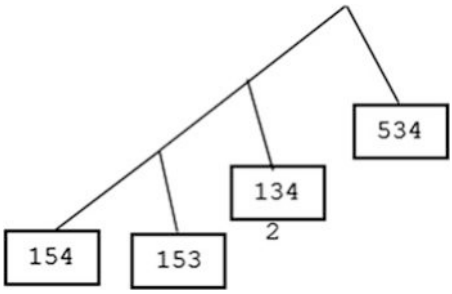
123

From event 123, the following event of dimension 3 can be generated, and the following partition:



1534

From event 1534, the following event of dimension 3 can be generated, and the following partition:



124

From event 124, the following event of dimension 3 can be generated, and the following partition:

56

As of event 56, no event of dimension 3 can be generated.

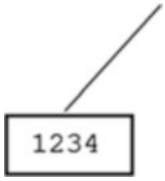
3

From event 4, no event of dimension 3 can be generated.

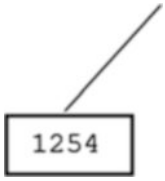
Dimension 4

We are now going to do the same with events of dimension 4. The partitions obtained are:

1234



From event 1254, only one event of dimension 4 can be generated, and the following partition:

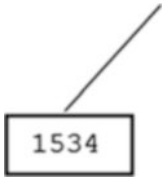


123

No event of dimension 4 can be generated from event 123.

1534

From event 1534, only one event of dimension 4 can be generated, and the following partition:



124

From event 124, no event of dimension 4 can be generated.

56

As of event 56, it cannot be generated

3

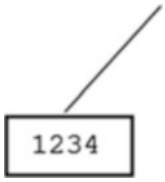
From event 4, no event of dimension 3 can be generated.

Dimension 4

We are now going to do the same with events of dimension 4. The partitions obtained are:

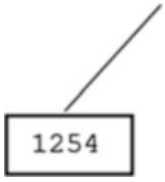
1234

From event 1234, only one event of dimension 4 can be generated, and the following partition:



1254

From event 1254, only one event of dimension 4 can be generated, and the following partition:

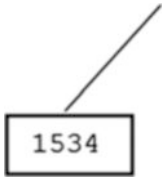


123

No event of dimension 4 can be generated from event 123.

1534

From event 1534, only one event of dimension 4 can be generated, and the following partition:



124

From event 124, no event of dimension 4 can be generated.

56

As of event 56, no event of dimension 4 can be generated.

3

For event 3, no event of dimension 4 can be generated.

Step A.2.2.3

In step A.2.2.3, we compared the hash trees of the candidate events and the sample. We start by comparing those of dimension 2:

If we analyse the previous figures, we can compare each cell of each event in the sample with the cells of the candidate events. Each matching box adds 1 to the support of the sample event. It can be seen with 1 next to the corresponding box. In the case of event 124, each box adds up to 2 because although that event has been represented once, it actually appears twice in the sample, and event 3 does not appear because it does not have dimension 2. All these sum numbers can be observed under the box of the event. According to this, the events that have sufficient support, that is, they appear more than 3.2 times, are the events:

{12}, {14}, and {24}

Let us now analyse those of dimension 3:

If we analyse the previous figures, we can compare each cell of each event in the sample with the cells of the candidate events. As in dimension 2 in the case of event 124, each box adds 2 because although that event has been represented once, it actually appears twice in the sample. Events 56 and 3 do not appear because they do not have dimension 3. Accordingly, the only event that has sufficient support is:

{124}

Let us now analyse those of dimension 4:

If we analyse previous figures, we can compare each cell of each event in the sample with the cells of the candidate events. Only events that have dimension 4 appear, and none of them appear 4 times; therefore, none have sufficient support.

Consequently, the events that we have to analyse in step B because they have sufficient support, which exceeds the established threshold of 40%, are:

{12}, {14}, {24}, and {124}

Step B

We carry out step B on the events with $s \geq 40\%$. As we know, we take the largest candidate set and call it B* (we call it B* to distinguish it from the elementary event B}, which in this case is B* = {X, B, C}.

Once we have defined B*, we take the first set A, which will be A = {X, C}, which implies that B* - A = {B}. We study A →B* - A, or what is the same {X, C} →{B}. The trust of the association is:

$$c = \frac{nB^*}{nA} = \frac{4}{5} = 0.8$$

As the confidence threshold has been set at 90 \% this association does not reach it and is not accepted, and due to the property of the apgenrules function, the associations

{X} →{c, B} and {C} →{X, B} are not sufficient either.

We now take A = {X, B}, which implies that B* - A = {C}. We study A →B* - A, or what is the same {X, B} →{C}. The trust of the association is:

$$c = \frac{nB^*}{nA} = \frac{4}{5} = 0.8$$

Therefore, it does not reach the trust threshold and neither is it accepted, and due to the property of the apgenrules function, the associations

{X} →{c, B} (which we already knew from the previous calculation) and {B} →{X, C} are not sufficient either.

Finally, we take A = {C, B}, which implies that B* - A = {X}. We study →B* - A, or what is the same {C, B} →{X}. The trust of the association is:

$$c = \frac{nB^*}{nA} = \frac{4}{4} = 1$$

In this case, the confidence threshold is reached, so apart from validating this association, we cannot apply the property. However, we already knew of the two previous cases that

{C} →{X, B} and {B} →{X, C} did not reach the threshold and were not accepted.

For dimension 2, we cannot apply the property, and we have to parse all of them:

{X} →{C}, $c = \frac{5}{6} = 0.83$ Not accepted
{X} →{B}, $c = \frac{5}{6} = 0.83$ Not accepted
{C} →{B}, $c = \frac{4}{5} = 0.8$ Not accepted
{C} →{X}, $c = \frac{5}{5} = 1$ Yes, it is accepted
{B} →{X}, $c = \frac{5}{5} = 1$ Yes, it is accepted
{B} →{C}, $c = \frac{4}{5} = 0.8$ Not accepted

Therefore, after applying the a priori algorithm, the associations selected by present 40% support and 90% confidence are:

{Cruise control, Bluetooth} →{Xenon headlights}
{Cruise control} →{Xenon headlights}
{Bluetooth} →{Xenon headlights}

Exercises Solved in R

In this subsection, the previous exercise 8 of the application of the Apriori algorithm will be solved using the R software. Once we have the arules package installed and loaded by default, we begin to solve association analysis problem with R.

8. Using the a priori algorithm, the associations whose support is equal to or greater than 40% and whose confidence is equal to or greater than 90% for the data the extent that customers request when buying a certain car model should be obtained. To carry out the study, the following sample is available, consisting of the extras included in 8 car sales:

   {X, C, N, B}
   {X, T, B, C}
   {N, C, X}
   {N, T, X, B}
   {X, C, B}
   {N}
   {X, B, C}
   {T, A}
   where {X: Xenon headlights, A: Alarm, T: Sunroof, N: Navigator, B: Bluetooth, C: Cruise control} are the extras that can be included in each car.

The first thing we have to do is introduce the values of the events of the sample with which we are working in R. To do this, we can realize that the events can be represented as an array of binary values in the following way. Each column represents the values of the different elementary events with which we work, and each row represents the values of a given event (observation), being a 1 if the elementary event occurs or a 0 if it does not occur.

According to this, the matrix corresponding to the observed sample: {X, C, N, B}, {X, T, B, C}, {N, C, X}, {N, T, X, B}, {X, C, B}, {N}, {X, B, C}, {T, A} is:

	X	A	T	N	B	C
Event 1	1	0	0	1	1	1
Event 2	1	0	1	0	1	1
Event 3	1	0	0	1	0	1
Event 4	1	0	1	1	1	0
Event 5	1	0	0	0	1	1
Event 6	0	0	0	1	0	0
Event 7	1	0	0	0	1	1
Event 8	0	1	1	0	0	0

We introduce this matrix in R and use a csv file. A csv file with the data are easy to write, and it is only necessary to open a notepad and introduce the data separated by commas. The first file will start with a tabulation and will have the name of the extras, and the first column will have the name of the events.[27] We are going to call it *extras*. The file will be loaded with the .csv extension and will have the following aspect:

[27] Without separation between the name "Event" and the number of the event.

```
   X,A,T,N,B,C
Event1,1,0,0,1,1,1
Event2,1,0,1,0,1,1
Event3,1,0,0,1,0,1
Event4,1,0,1,1,1,0
Event5,1,0,0,0,1,1
Event6,0,0,0,1,0,0
Event7,1,0,0,0,1,1
Event8,0,1,1,0,0,0
```

We use the function *read.csv* to introduce the data of the *extras.csv* file into R, and we call the entry variable extras:

```
extras<-read.csv("extras.csv")
```

Now we have the data in R but we have in a data frame format that we need to convert into matrix format, that we will call extrasM. To do this, we use the function *data.matrix* in the following way:

```
extrasM<-data.matrix(extras)
```

Now, we have the data as a type matrix. To be sure of this, we use the function *class* to get the type of extrasM

```
class(extrasM)
```

Once we have checked that we have now the data as a matrix, we know that we need a sparse matrix. To get this, we need to convert the matrix into a sparse matrix, but to do that, we need the function *as* of the package Matrix, but as we know that package Matrix is in the standard library of R, we do not need to install and it and only to load it with the function library

```
library(Matrix)
```

and once we have the package Matrix loaded, we convert extrasM into a sparse matrix that we call extrasMS

```
extrasMS<-as(extrasM, "sparseMatrix")
```

We need a sparse matrix because arules need to work a ngCMatrix or sparse matrix of nonzero positions, so we will have to convert the sample matrix into that type of matrix, and if we had not put the sparse attribute, we could not because the function as, which is the one that will convert the sample matrix into a ngCMatrix, can only do so if the sample matrix is sparse. The complete statement for converting to a ngCMatrix will be:

```
extrasMngC<-as(extrasMS, "nsparseMatrix")
```

Then, before applying the arules functions, we must transpose the extrasMngC matrix to correctly analyse the sets. We do so through the t function.

```
extrasMngCt<-t(extrasMngC)
```

Then, we performed association analysis using the functions of arules. If we have not installed the arules package, the following instructions will not run. The first thing is to determine the possible transactions, which we do using the as function and the transactions attribute. We will assign the result to a new variable that we will call tr. The complete statement is:

transactions<-as (extrasMngCt, "transactions")

We observe the result by introducing transactions. Next, we ask for a summary of tr using the function *summary*.

summary (transactions)

After these previous steps, to properly carry out the association analysis, we use the a priori algorithm through the function of arules apriori (as seen in theory). We define a variable as the output of the associations obtained, and we call that variable *associations*. The complete instructions are as follows:

associations <-apriori (transactions, parameter = list (support = 0.4, confidence = 0.9))

We have chosen, as in the manual exercise, a support of 40% and a confidence of 90%. Next, if we introduce associations to see the result,

associations

We find a set of 3 rules. To see what they are, we introduce the inspect function. The complete statement is:

inspect (associations)

And you get:

lhs, rhs, support, confidence, lift;

	lhs	rhs	support	confidence coverage	lift	count
[1]	{C}	=> {X}	0.625	1	1.333333	5
[2]	{B}	=> {X}	0.625	1	1.333333	5
[3]	{B, C}	=> {X}	0.500	1	1.333333	4

This is the same result that we have obtained in the previous solved by hand subsection.

Bibliography

1. Ahmad, I., 40 *Algorithms Every Programmer Should Know*, Packt, 2020, ISBN: 9781789801217
2. Adler, J., *R in a Nutshell*, O'Reilly, 2012, ISBN: 97833897216495
3. Baesens, B., *Fraud Analytics Using Descriptive, Predictive, and Social Network Techniques: A Guide to Data Science for Fraud Detection*, Wiley, 2015, ISBN: 9781119133124
4. Berry, M., Azlinah, M., and Yap, B. *Supervised and Unsupervised Learning for Data Science*, Springer, 2019, ISBN: 978-3-030-22474-5
5. Bishop, C., *Pattern Recognition and Machine Learning*, Springer, 2010, ISBN: 0387310738
6. Bonaccorso G., *Machine Learning Algorithms: Popular algorithms for data science and machine learning*, Packt, 2018, ISBN: 1789347998
7. Bramer, M., *Principles of Data Mining*, Springer, 2020, ISBN:
8. Breunig, M., Kriegel, H., Raymond, T., Sander, J., LOF: Identifying Density-Based Local Outliers, ACM SIGMOD, Volume 29, Issue 2, 2000 pp 93–104
9. Bruce, P., Bruce, A., Gedeck, P., *Practical Statistics for Data Scientists: 50+ Essential Concepts Using R and Python*, O'Reilly, 2020, ISBN: 149207294X
10. Celebi, M., Aydin, K. *Unsupervised Learning Algorithms*, Springer, 2018, ISBN: 3319795902
11. Chakraborty, S. Islam, S., Samanta, D., Data Classification and Incremental Clustering in Data Mining and Machine Learning, Springer, 2022, ISBN: 978-3-030-93087-5
12. Chatterjee, S., Hadi, S., *Regression Analysis by Example*, Wiley, 2006, ISBN: 9780471746966
13. Chengqi, Z., Zhang, S. *Association rule mining : models and algorithms*, Springer, 2002, ISBN 3-540-43533-6
14. Christian R., *The Bayesian Choice: From Decision-Theoretic Foundations to Computational Implementation*. Springer, 2007, ISBN:
15. Ciaburro, G., Venkateswaran, B., *Neural Networks with R*, Packt, 2017, ISBN: 9781788397872
16. Ciaburro, G., *Regression Analysis with R*, Packt, 2018 ISBN: 9781788627306
17. Cohen, J., Cohen, P., West, S., Aiken, L., *Applied Multiple Regression/Correlation Analysis for the Behavioral Sciences*, Routledge, 2013, ISBN: 9781134801015
18. Cormen, H., LeisersoN, C., Rivest, L., Stein, C., *Introduction to Algorithms*, MIT Press, 2009, ISBN: 9780262033848
19. Crawley, M. *The R Book*, Wiley,2007, ISBN: 9780470510247
20. Cuadrado-Gallego, J.J., Demcheko, Y., *The Data Science Framework: A view from the EDISON Project*, Springer, 2020, ISBN:
21. Dahan, H., Cohen, S., Rokach, L., Maimon, O., *Neural Networks: Methodology and Applications*, Springer, 2005, ISBN: 978-3-540-22980-3

J. J. Cuadrado-Gallego, Y. Demchenko, *Data Analytics*,
https://doi.org/10.1007/978-3-031-39129-3

22. Dreyfus, G., *Proactive Data Mining with Decision Trees*, Springer, 2014, ISBN: 978-1-4939-0538-6
23. Dunning, T., Friedman, E., *Practical Machine Learning: A New Look at Anomaly Detection*, O'Reilly, 2014, ISBN: 9781491911600
24. ElAtia, S., Ipperciel, D., Zaïane, O., *Data Mining and Learning Analytics*, Wiley, 2016, ISBN: 9781118998236
25. Feasel, K. *Finding Ghosts in Your Data: Anomaly Detection Techniques with Examples in Python*, Apress, 2022, ISBN: 9781484288702
26. Fischetti, T. et al., *R: Data Analysis and Visualization*, SIAM, 2007, ISBN:978-0-89871-623-8
27. Foreman, J., *Data Smart: Using Data Science to Transform Information into Insight*, Wiley, 2013, ISBN: 9781118661468
28. Forsyth, D., *Applied Machine Learning*, Springer, 2019, ISBN: 978-3-030-18113-0
29. Gan, G., Ma, C., Wu, J., Data Clustering: Theory, Algorithms, and Applications, SIAM, Published:2007, ISBN:978-0-89871-623-8, ASA-SIAM Series on Statistics and Applied Mathematics
30. Gkoulalas-Divanis, A., Verykios, V., Association Rule Hiding for Data Mining, Aris, Springer, 2010, ISBN: 1441965688
31. Goodfellow, I., Bengio, Y., Courville, A., Deep Learning, MIT Press, 2016, ISBN:
32. Govindan, G., Hora, S., Palagachev, K. *The Data Analysis Workshop: Solve business problems with state-of-the-art data analysis models, developing expert data analysis skills along the way*, Packt, 2020, ISBN: 1839211385
33. Grus, J., *Data Science from Scratch: First Principles with Python*, O'Reilly, 2015 ISBN: 149190142X
34. Grolemund, G., *R for Data Science*, O'Reilly, 2016, ISBN: 9781491910344
35. Han, J., Kamber, M., Pei, J., *Data mining : concepts and techniques*, Morgan Kaufman, 2012, ISBN 978-0-12-381479-1
36. Härdle, W., Horng-Shing, H., Shen, X., *Handbook of Big Data Analytics*, Springer, 2018, ISBN: 9783319182834
37. Hrbacek, K., Jech, T., *Introduction to set theory*, Marcel Dekker, AÑO, ISBN: 0-8247-7915-0
38. Jarman, K., *Beyond Basic Statistics: Tips, Tricks, and Techniques*, Wiley, 2015, ISBN: 9781118856116
39. Juretig, F., *R Statistics Cookbook*, Packt, 2019, ISBN: 9781789802566
40. James, G., Witten, D., Hastie, T., *An Introduction to Statistical Learning: with Applications in R*, Springer, 2022, ISBN: 1071614207
41. Kabacoff, R., *R in Action*, Manning Publications, 2011, ISBN: 9781935182399
42. Kotu, V., *Data Science*, Morgan Kaufmann, 2018, ISBN: 978012814762
43. Kotu, V., *Predictive Analytics and Data Mining*, Morgan Kaufmann, 2014, ISBN: 9780128016503
44. Kumar, R., *Machine Learning Quick Reference*, Packt, 2019, ISBN: 9781788830577
45. Lesmeister, C., Kumar, S., *Advanced Machine Learning with R*, Packt, 2019, ISBN: 9781838641771
46. Liu, Y. Maldonado, P., *R Deep Learning Projects*, Packt, 2018, ISBN: 9781788478403
47. Livshin, I., Artificial Neural Networks with Java: Tools for Building Neural Network Applications, Apress, 2021, ISBN: 9781484273685
48. López. V., Problemas Resueltos de Electromagnetismo, Editorial Centro de Estudios Ramón Areces, 1990, ISBN: 8487191622
49. Malik, A., Tuckfield, B., Applied Unsupervised Learning with R: Uncover hidden relationships and patterns with k-means clustering, hierarchical clustering, Packt, 2019, ISBN:
50. Montgomery, C., Peck, E., Vining, G., *Introduction to Linear Regression Analysis*, Wiley, 2012, ISBN: 9780470542811
51. Mailund, T., *Domain-Specific Languages in R: Advanced Statistical Programming*, Apress, 2018, ISBN: 9781484235881

52. Michelucci, U., *Applied Deep Learning: A Case-Based Approach to Understanding Deep Neural Networks*, Apress, 2018, ISBN: 9781484237908
53. Mathar, R., Alirezaei, R., Balda, E., Behboodi, A., *Fundamentals of Data Analytics, With a View to Machine Learning*, Springer, 2020, ISBN:
54. Nwanganga, F., *Practical Machine Learning in R*, Wiley, 2020, ISBN:
55. Nasraoui, O., N'Cir, C., *Clustering Methods for Big Data Analytics: Techniques, Toolboxes and Applications*, Springer, 2019, ISBN: 978-3-319-97863-5
56. Perros, H., An Introduction to IoT Analytics, Chapman and Hall, 2021, ISBN: 0367686317
57. Pardoe, I., *Applied Regression Modeling*, Wiley, 2020, ISBN: 9781119615866
58. Peña, D., Tsay, R., Statistical Learning for Big Dependent Data by Released, Wiley, 2021, ISBN: 9781119417385
59. Rosenblatt, F., The perceptron: A probabilistic model for information storage and organization in the brain. *Psychological Review*, 65, 386–408, 1958
60. Riazoshams, H., Midi, H., Ghilagaber, G., *Robust Nonlinear Regression*, Wiley, 2018, ISBN: 9781118738061
61. Rokach, L., Maimon, O., *Data Mining with Decision Trees*, Theory and Applications, EDIT, 2014, ISBN:
62. Saleem. T., Chishti, M., *Big Data Analytics for Internet of Things*, Wiley, 2021, ISBN: 978-1-119-74075-9
63. Sakr, S. Zomaya, A., Encyclopedia of Big Data Technologies, Springer 2019, ISBN: 978-3-319-77524-1
64. Sammut, C., Webb, G., *Encyclopedia of Machine Learning and Data Mining,* Springer, 2017, ISBN: 978-1-4899-7685-7
65. Sedgewick, R., Wayne, K., *Algorithms*, Addison-Wesley, 2011, ISBN: 32157351X
66. Sheppard, C., *Tree-based Machine Learning Algorithms: Decision Trees, Random Forests*, and Boosting CreateSpace, 2017, ISBN: 1975860977
67. Shikhman, V., Müller, D., *Mathematical Foundations of Big Data Analytics*, Springer, AÑO, ISBN 978-3-662-62520-0
68. Spiesser, M. Histoire de Moyennes, DEA mémoire, IREM de Toulouse, T 159, 1997
69. Sugomori, Y., Kaluža, B., Soares, F., Souza, A., *Deep Learning: Practical Neural Networks with Java*, Packt, 2017, ISBN: 9781788470315
70. Thanaki, J., *Machine Learning Solutions*, Packt, 2018, ISBN: 9781788390040
71. Warr, K., Strengthening Deep Neural Networks, O'Reilly, 2019, ISBN: 9781492044956
72. Wiley, M., Wiley, J., *Advanced R Statistical Programming and Data Models: Analysis, Machine Learning, and Visualization*, Apress, 2019, ISBN: 9781484228722
73. Wilson, H., Keating, B., Beal, M., *Regression Analysis*, Business Expert Press, 2015, ISBN: 9781631573866
74. Wilcox, R. *Understanding and Applying Basic Statistical Methods Using R*, Wiley, 2016, ISBN: 9781119061397
75. Winters, R., *Practical Predictive Analytics*, Packt, 2017, ISBN: 9781785886188
76. Ren, K., *Learning R Programming*, Packt, 2016, ISBN: 9781785889776
77. Zhao, Y., Zhang, C., Cao, L., *Post-Mining of Association Rules: Techniques for Effective Knowledge Extraction*, EDITORIAL, AÑO ISBN: 9781605664040