Applied Intelligence (2025) 55:67
https://doi.org/10.1007/s10489-024-06092-8

SDROF: outlier detection algorithm based on relative skewness density ratio outlier factor

Zhongping Zhang[1,2] · Kuo Wang[1] · Jinyu Dong[1] · Sen Li[1]

Accepted: 19 November 2024 / Published online: 2 December 2024

Abstract
Outlier detection is a crucial research problem in data mining, aiming to identify data objects that significantly deviate from the distribution of other data. To solve the issues of low-density patterns and low local density problems in nearest neighbor-based outlier detection methods, this paper proposes an outlier detection algorithm based on the relative skewness density ratio outlier factor. An adaptive determination of the number of neighbors (k value) and neighborhood is achieved using the natural neighbor search algorithm, effectively addressing parameter setting challenges. It introduces the concept of relative skewness to quantify how much data objects deviate from their neighbors, along with a local density ratio to capture variations in local density. This leads to a new outlier measure called the Relative Skewness Density Ratio Outlier Factor, which uses the ratio of relative skewness to local density as the outlier factor. The outlier degree of each data object is further assessed by evaluating the deviation of this factor from its neighbors. Experimental validation of the proposed algorithm is conducted on both artificial and real-world datasets, with comparisons against recent novel outlier detection algorithms, demonstrating the effectiveness of the proposed algorithm.

Keywords Data mining · Outlier detection · Skewness · K nearest neighbors · Outlier factor

1 Introduction

Outlier detection is an essential research problem in data mining, aiming to discover data objects that significantly deviate from the distribution of other data [1, 2]. The presence of outliers in a dataset typically affects the quality of data mining and analysis results. Outlier detection methods can identify abnormal data in the dataset, while also revealing valuable information. Outlier detection holds significant value and practical applications in real life. For instance, in the financial domain, outlier detection aids in timely identification of fraudulent activities, safeguarding investors' interests. Similarly, in the realm of cybersecurity, outlier detection helps in detecting network attacks and intrusion attempts, thereby ensuring network security. Currently, outlier detection has found widespread applications in these practical domains, including medical diagnosis [3], wireless sensor networks [4], fraud detection [5–7], fault detection [8], network intrusion detection [9], and urban abnormal traffic detection [10, 11].

The main research methods for outlier detection can be divided into supervised, semi-supervised, and unsupervised approaches. Supervised methods require a training set with labels for normal data and outliers, which can be challenging to obtain in many applications. Semi-supervised methods only require labels for normal data during training, making them more practical than supervised methods. Unsupervised methods, on the other hand, have a wider applicability as they do not require any labels or training data.

In supervised and semi-supervised methods, outlier detection is treated as a binary classification problem requiring labels [12]. With complete labels for normal and outlier data

✉ Kuo Wang
wangkuowork@163.com

Zhongping Zhang
zpzhang@ysu.edu.cn

Jinyu Dong
djy15863676733@163.com

Sen Li
lisenjc@163.com

[1] College of Information Science and Engineering, Yanshan University, Hebei Street, Qinhuangdao 066004, Hebei, China

[2] The Key Laboratory for Computer Virtual Technology and System Integration of Hebei Province, Yanshan University, Hebei Street, Qinhuangdao 066004, Hebei, China

points, supervised methods can identify known outliers but may miss unknown outliers. Currently, there are no specialized supervised outlier detection algorithms, and existing classifiers such as random forests [13] and neural networks [14] are often used. However, supervised methods suffer from inaccuracies in labeling and may fail to detect all types of outliers, limiting their effectiveness to known types of outliers. Semi-supervised methods aim to utilize partial labels for detection while retaining the ability to detect unknown types of outliers. In recent years, some research has focused on effectively utilizing partial label data to improve detection performance and promote representation learning. For example, some semi-supervised algorithms train only on normal samples and detect outliers deviating from the learned normal representation [15–17].

Unsupervised outlier detection methods can be classified into clustering-based and nearest neighbor-based approaches. Clustering-based methods partition data objects into different clusters based on their similarity and define outliers as objects not belonging to any cluster or lying far from the nearest cluster. Ester et al. proposed a density-based clustering algorithm called DBSCAN [18] (Density-Based Spatial Clustering of Applications with Noise), which can identify clusters of different shapes but requires setting thresholds and may not work well for clusters with varying densities. He et al. [19] proposed a new clustering-based local outlier factor algorithm called CBLOF (Cluster-based Local Outlier Factor), which defines the outlier factor based on the size of clusters and the distance of data objects from their nearest clusters. Rodriguez and Laio [20] proposed a method that identifies cluster centers as local density maxima far from high-density points, relying solely on relative density rather than absolute values. However, clustering methods primarily focus on optimizing clustering, and if the clustering is poor or if outliers are assigned to clusters, this approach may not be effective.

Nearest neighbor-based methods can be further divided into distance-based and density-based approaches. The core of distance-based methods for outlier detection lies in evaluating the distance relationship between a data point and its neighboring data points. In essence, outliers are identified due to their significant deviation from the majority of points in the dataset. This is typically achieved by computing the distance from each data point to its nearest neighbor and performing comparative analysis based on these distances. During the data traversal process, the emphasis is placed on measuring and comparing these distance values to identify points that are markedly isolated. Based on the construction of the reference set, outliers can be classified into global outliers and local outliers. Global outliers are identified with respect to the entire dataset as the reference set, while local outliers are identified using the nearest neighbors of the data point as the reference set. When the pattern of normal data is homogeneous, distance-based methods can effectively detect global outliers. However, when the data distribution is uneven and forms clusters with varying densities, such methods may incorrectly classify sparse normal points as outliers, leading to reduced detection accuracy. Knorr and Ng [21] first proposed distance-based outlier detection methods, followed by Ramaswamy et al. [22] proposing a partition-based method that uses the k-th nearest neighbor to determine outlier values. Zhang et al. proposed a method called LDOF [23] (Local Distance-based Outlier Factor) based on local distance outlier factors, where the relative positions of data objects and their neighbors determine the degree to which an object deviates from its neighbors; if a data object is far from its neighborhood space, it is considered an outlier. However, this method cannot detect outliers between two dense clusters. Yang et al. proposed MOD [24] (Mean-shift Outlier Detection), an outlier detection method using mean shift. It measures outliers based on the distance of data shifts: larger distances indicate higher likelihood of being an outlier. However, this method is inefficient due to iterative shifts and may not detect local outliers well. In summary, Distance-based methods are good at detecting global outliers but have difficulty effectively identifying local outliers. Additionally, selecting appropriate parameters is crucial, such as choosing the number of neighbors (k) in many distance-based methods.

However, density-based methods effectively address these issues by considering the density relationships of data objects within a neighborhood. These methods compare the density of a data object to the density of its surrounding neighborhood, where objects with lower density may be identified as outliers. Breunig et al. introduced the concept of local outlier factor (LOF) [25], which measures the outlier degree of data points by computing LOF scores. This method performs poorly on sparse datasets, and if the density of outliers is close to their neighborhood density, normal scoring may not be achieved. Subsequently, other methods were introduced to improve the efficiency of the LOF algorithm, such as the connectivity-based outlier factor (COF) [26], a robust kernel-based outlier factor (RKOF) [27], kernel density estimates outlier score (KDEOS) [28], influenced outlierness (INFLO) [29] and neighborhood weighted local outlier factor (NWLOF) [30]. However, density-based outlier detection methods mostly face the problem of low density, where the density of outliers cannot be well distinguished from the density of their neighborhoods in sparse datasets. Additionally, density-based methods face parameter selection issues, such as setting the parameter k for object neighbors.

In addition, numerous algorithms have been developed to enhance existing outlier detection methods by combining the advantages of density and distance. For instance, Zhang et al. proposed a two-parameter outlier detection algorithm called TPOD [31] (Two-Parameter Outlier Detection). This method

utilizes the ratio of local density to relative distance as an outlier factor and optimizes the algorithm through a combination of these two parameters. Similarly, Li et al. introduced an outlier detection algorithm based on a density-distance decision graph [32] (Outlier Detection Based on the Density-Distance Decision Graph). This approach measures the degree of local outlierness using local density ratios and assesses global outlierness using distance, subsequently integrating these two metrics to create a density-distance decision graph and compute the product of the two indicators as the final outlier score. Although many existing algorithms attempt to address the limitations of density and distance by combining them, this remains a challenging issue that requires further resolution.

In summary, the aforementioned methods do not effectively address issues such as excessive dependence on parameter selection, as well as the handling of data that is sensitive to low density and variations in local density. To overcome these challenges, this paper proposes an outlier detection algorithm based on the relative skewness density ratio, named SDROF (Outlier Detection Algorithm Based on Relative Skewness Density Ratio Outlier Factor). First, the algorithm introduces a method for adaptively selecting the value of k using the concept of natural neighbors, determining the natural neighborhood to avoid uncertainties caused by manually setting the value of k. Then, it defines relative skewness to characterize the distribution of data objects and their natural neighborhoods, where relative skewness reflects the distance of data objects from their neighborhoods; global outliers often have greater distances. Next, to address the problem of low-density patterns, the algorithm defines local density ratio to extract local information of data objects, thereby improving the algorithm's ability to identify local outliers. Finally, the ratio of relative skewness to local density ratio is used as the outlier factor, and the difference between the outlier factor of data objects and their neighborhoods is calculated as the final outlier value measure; the greater the difference, the more likely it is an outlier. This algorithm can adapt well to datasets with complex density distributions and can identify both global and local outliers. Additionally, for high-dimensional datasets, this paper uses Manhattan distance instead of Euclidean distance to enhance the algorithm's detection capabilities on high-dimensional datasets.

The main contributions of this paper are summarized as follows:

1. This paper applies the concept of natural neighbors to the SDROF algorithm, eliminating the need to manually define the number of neighbors k. This avoids issues of redundant or insufficient neighbor information caused by parameter selection, resulting in a more stable performance of the SDROF algorithm that does not depend on the k value.
2. A new outlier measure, the Relative Skewness Density Ratio Outlier Factor, is introduced. This method uses the ratio of relative skewness to local density ratio as an outlier factor. Relative skewness clearly describes the distribution relationship between data objects and their natural neighborhoods, considering outlierness from a global perspective, while the local density ratio extracts local information, characterizing the closeness between data objects and natural neighbors and evaluating outlierness from a local perspective.
3. This concept of factor difference is proposed to compute the deviation of the Relative Skewness Density Ratio Outlier Factor between data objects and their neighbors. This effectively captures local differences between data objects and neighbors, allowing for a deeper exploration of the distribution relationship based on the Relative Skewness Density Ratio Outlier Factor.
4. This paper validates the correctness and effectiveness of the SDROF algorithm through comparative experiments on synthetic and real datasets. Compared to existing algorithms, SDROF demonstrates significant improvements in detecting local outliers and achieves better performance.

The rest of this paper is organized as follows: Section 2 outlines the prerequisites, including concepts related to k-nearest neighbors, natural neighbors, and skewness. Section 3 introduces the proposed method. Section 4 demonstrates the effectiveness of our approach through experimental results on synthetic and real-world datasets. Section 5 presents the conclusions.

2 Prerequisite

Prior to exploring the subsequent content, it is imperative to establish a groundwork of fundamental concepts and definitions. These serve as the prerequisites and cornerstone for comprehending the methodologies presented in this paper. This section will elucidate the pertinent notions of K Nearest Neighbors, Natural Neighbors, and Skewness.

2.1 K nearest neighbors

Definition 1 (KNN distance of x_i). The k-nearest neighbor distance [33] of a data object x_i refers to the average Euclidean distance from x_i to its k nearest neighbors, denoted as d_i. The calculation formula is as follows:

$$d_i = \frac{1}{k} \sum_{x_j \in KNN_i} dist(x_i, x_j) \tag{1}$$

In (1), KNN represents the k nearest neighbors of the data object x_i, and dist(x_i,x_j) denotes the Euclidean distance between x_i and x_j. The KNN distance can reflect the compactness between data objects.

Definition 2 (KNN local density of x_i). The k-nearest neighbor local density [33] of a data object is defined as the reciprocal of the average distance to its k nearest neighbors, denoted as den_i. The calculation formula is as follows:

$$den_i = \frac{1}{d_i} = \frac{k}{\sum_{x_j \in KNN_i} dist(x_i, x_j)} \tag{2}$$

The KNN local density is a classic density calculation method based on Euclidean distance. When a data object is located near a dense cluster, its KNN local density will be large, whereas when it is located near a sparse cluster, its KNN local density will be relatively small.

2.2 Natural neighbors

In recent years, natural neighbors [34] have been widely applied in data mining fields such as clustering analysis and outlier detection. Zhu et al. [35] proposed a novel concept of parameter-free natural neighbors, inspired by human social networks: when two individuals consider each other as friends, they are considered true friends. This concept can be extended to data objects, where if data object x is a neighbor of data object y and y is also a neighbor of x, then data objects x and y are natural neighbors. The natural neighbor method does not require any parameters to determine the neighborhood of data objects. Its core idea is to gradually expand the search scope to find neighbors until all objects are considered neighbors, or until the number of individuals not considered neighbors by other objects remains stable.

Definition 3 (Natural Neighborhood). The natural neighborhood of data object x_i, denoted as $NaN(x_i)$, refers to the closest neighbor set searched during the process of finding natural neighbors based on the parameter λ, which represents the adaptively selected number of neighbors.

In Algorithm 1, r represents the search rounds, $KNN(x)$ denotes the nearest neighbors of data object x, $NaN(x)$ represents the natural neighborhood of x, and $Nb(x)$ indicates the number of objects considering x as a neighbor. To shorten the search time, this paper utilizes the KD-Tree data structure to enhance the efficiency of neighbor search.

2.3 Skewness

Skewness [36] has been widely applied in the field of data mining. Skewness refers to the degree of deviation of a data object relative to its neighbors. If a data object is located at the center of a cluster, its neighbors will be evenly distributed around it. However, if a data object is located at the boundary of a cluster or far away from the cluster, its neighbors will significantly lean towards one side. Figure 1 illustrates the skewed distribution of two-dimensional data objects with the nearest neighbor parameter k = 4. In the figure, three points A, B, and C are taken as centers, and a Cartesian coordinate system is established. The dimensions of the neighbors of these three points are mapped onto the x-axis and y-axis, respectively. From the figure, it can be observed that point A is located in a dense area, and its neighbors are evenly distributed around the coordinate axes. Point B is located at the edge of the cluster, with its neighbors leaning towards one side of the coordinate axis. Point C is situated at a more distant position, with its neighbors all mapped within the same quadrant of the coordinate axis. Compared to point A and point B, point C obviously deviates more from their neighborhoods. Moreover, as the expansion continues from the central area outward, the degree of deviation of the data objects at the periphery gradually increases. This illustrates how the position of a data object can affect its skewness relative to its neighbors.

Algorithm 1 Natural neighbor search algorithm.

Input: D(the dataset);
Output: λ (the natural characteristic value), NaN (the set of natural neighbors)
Initial:$KNN(x) = \varnothing, NaN(x) = \varnothing, Nb(x) = 0, r = 1, num = 0$;
Create a KD-Tree T from the dataset D.
while true **do**
  for each $x_i \in$ T **do**
    Find the r-th neighbor x_j of x_i.
    $KNN_r(x_i) = KNN_{r-1}(x_i) \cup \{x_j\}$;
    $Nb(x_j) = Nb(x_j) + 1$;
  end for
  $num = count(Nb(x_i) == 0)$;
  if num remains unchanged or $num == 0$ **then**
    break;
  end if
  $r = r + 1$;
end while
for each $x_i \in D$ **do**
  $NaN(x_i) = KNN_r(x_i)$;
end for
$\lambda = r$;
return λ, NaN;

Definition 4 (Skewness). Skewness of data object x_p refers to the degree of skewness of x_p relative to its k-neighborhood, denoted as $S_c(x_p)$. The calculation formula is as follows:

$$Sc(x_p) = \frac{\sum_{j=1}^{m} \sum_{i=1}^{k} (x_{ij} - x_{pj})^2}{k}, x_{ij} \in KNN_p \tag{3}$$

In (3), m represents the dimensionality of the dataset, and k denotes the number of neighbors of a data object. From the formula, it can be observed that when a data object x_p

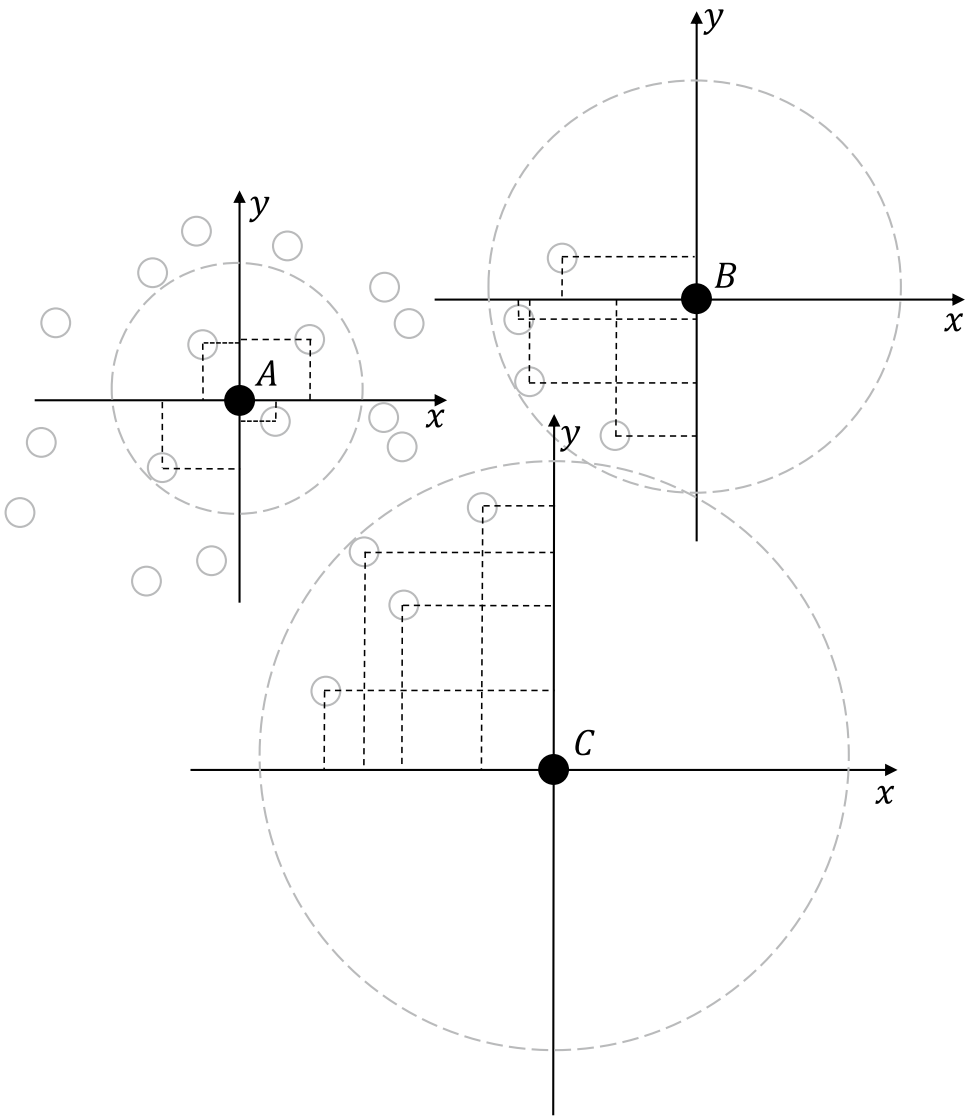


Fig. 1 Skewed distribution of data

is located in a dense region, the distances between x_p and its neighbors in each dimension are relatively small, resulting in a small skewness value. As the data object expands outward from the dense region, the outer data objects gradually deviate from their neighborhoods, and the distances between them and their neighbors in each dimension gradually increase. The skewness also increase, and the degree of outlier gradually increase.

3 SDROF algorithm

In this section, we present the full process and basic definitions of the SDROF algorithm. SDROF integrates global and local outlier detection methods, offering a detailed and precise assessment of data object abnormalities. The algorithm combines the strengths of relative skewness and local density ratio. Relative skewness, incorporating the concept of relative distance, portrays the distribution between data objects and their neighbors more clearly, enabling global anomaly assessment. The local density ratio captures local information and proximity between objects, facilitating local outlier detection. By combining these two indicators as a ratio, SDROF provides a comprehensive anomaly assessment from both perspectives. To delve deeper into data object distribution, we introduce the concept of factor difference, which measures deviations in the relative skewness density ratio outlier factor between objects and their neighbors, effectively capturing local differences.

3.1 Definition of SDROF algorithm

In this section, we introduce a novel outlier detection algorithm based on the Relative Skewness Density Ratio Outlier Factor (SDROF) and demonstrate its entire process. We incorporate the concept of relative distance into the frame-

work of skewness, leading to the development of a new metric, relative skewness. Relative skewness provides a clearer characterization of the distributional relationship between a data object and its natural neighbors, thereby assessing the outlierness of data objects from a global perspective. Additionally, we define the local density ratio, a metric that extracts local information about data objects and captures the closeness between a data object and its natural neighbors, thus evaluating outlierness from a local perspective. We combine relative skewness and local density ratio in a ratio form to form the Relative Skewness Density Ratio Outlier Factor, enabling a comprehensive assessment of outlierness from both global and local perspectives. To further explore the distributional relationship of data objects based on the Relative Skewness Density Ratio Outlier Factor, we propose the concept of factor difference. The factor difference calculates the deviation of the Relative Skewness Density Ratio Outlier Factor between a data object and its neighbors, thereby more effectively characterizing the local differences between a data object and its natural neighbors.

Definition 5 (KNN Relative Distance of x_i). The KNN Relative Distance ω_i refers to the shortest distance to the data object x_i among the objects whose density is greater than x_i among its k nearest neighbors. The calculation formula is as follows:

$$\omega_i = \left\{ min(dist(x_i, x_j)) \,\middle|\, x_j \in KNN_i, den_j > den_i \right\} \quad (4)$$

In (4), den_i represents the local density of the data object x_i. The KNN Relative Distance ω_i is a density-based distance metric, which varies with the density change around the data object itself, thereby identifying the shortest distance tending towards dense clusters.

Definition 6 (Relative Skewness of x_i). Relative skewnes ReS_i refers to the product of the skewness of data object x_i and its KNN relative distance. The calculation formula is as follows:

$$ReS_i = Sc(x_i) * \omega_i = \frac{\sum_{j=1}^{m} \sum_{i=1}^{\lambda} (x_{ij} - x_{pj})^2 * \omega_i}{\lambda}, \quad x_{ij} \in NaN_i \quad (5)$$

In (5), Sc_i represents the skewness of data object x_i, ω_i represents the relative distance of data object x_i with respect to its NaN neighbors, NaN_i represents the natural neighbors of the data object, and λ represents the number of data objects in NaN_i. Relative Skewness ReS_i characterizes the deviation degree of data object x_i relative to its k natural neighbors.

The concept of relative skewness introduces the notion of relative distance based on skewness. When a data object is located in a dense region, its skewness tends to be small because the distances between data objects in the dense region are relatively close. Consequently, in the k nearest neighbor region of this data object, the distances to the data objects with a higher density are also small, indicating a small relative distance for this data object. As shown in (5), the relative skewness value of data objects in dense regions tends to be small. Conversely, when a data object is situated in a sparse region, both the skewness and relative distance tend to be large, resulting in a large relative skewness value. The combination of skewness and relative distance further enhances the deviation degree of data objects from their neighborhoods, highlighting the outlier characteristics of data objects, thereby enabling the algorithm to effectively detect outliers.

Definition 7 (Local Density Ratio of x_i). The Local Density Ratio ldr_i refers to the ratio of the density of data object x_i to the average density of data objects in its natural neighborhood. The calculation formula is as follows:

$$ldr_i = \frac{\lambda \times den_i}{\sum_{j \in NaN_i} den_j} \quad (6)$$

In (6), den_i and den_j represent the local densities of data objects x_i and x_j respectively, NaN_i denotes the natural neighbor set of data object x_i, and λ represents the number of data objects in the natural neighborhood NaN_i.

From the description of Definition 7, it can be seen that the local density ratio ldr is related to both the density of the data object itself and the densities of its neighbors. For normal data, their distribution is similar to that of their neighborhood, so the difference between the density of the data object itself and the average density of its neighbors is not significant, and the local density ratio tends to be close to 1. However, for outliers, their distribution differs from that of their neighborhood, resulting in a significant difference between the density of the data object itself and the average density of its neighbors, leading to a large variation in the local density ratio. If only the local density of the data object itself is considered, it may not be a good indicator for distinguishing outliers. For example, a data object located in a normal sparse cluster, i.e. , a low-density area, although it is a normal data object, may be mistakenly identified as an outlier because it has a smaller density compared to data objects in high-density areas. Because the local density ratio reflects the relative size of densities in the form of a ratio, it is more adaptable to situations with complex density distributions.

As shown in Fig. 2, point A is located near a high-density region, indicating a local outlier, while point B resides in a low-density area, representing a normal point within a sparse cluster. However, if only the local density of each point is considered, point A would likely have a higher local density compared to point B, as point A is closer to other data objects in its neighborhood. Consequently, point B might erroneously be considered as an outlier. By utilizing the local density ratio method, it's observed that the average density of

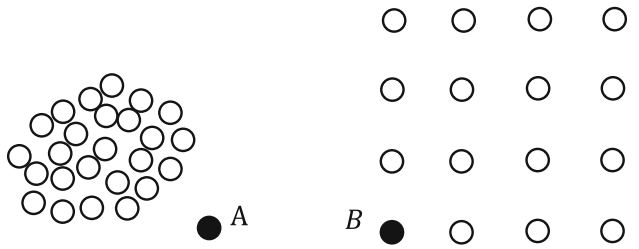


Fig. 2 Local density

point A's neighborhood significantly exceeds its own density, resulting in a local density ratio (ldr) much smaller than 1 for point A. Conversely, point B exhibits a similar distribution to its neighborhood, leading to an ldr close to 1. Thus, the local density ratio method correctly identifies point A as the more probable outlier.

Based on relative skewness and local density ratio, it can be observed that normal data objects exhibit smaller relative skewness and larger local density ratio values. Conversely, outliers tend to have larger relative skewness and smaller local density ratio values. Therefore, this section proposes a novel outlier measurement metric, termed as the relative skewness-density ratio outlier factor. This outlier factor effectively captures the relationship between the deviation degree of data objects and their local density, thereby highlighting the outlier degree for each data object.

Definition 8 (Relative Skewness-Density Ratio Outlier Factor of x_i). The relative skewness-density ratio outlier factor ($SDROF_i$) is defined as the ratio of the relative skewness to the local density ratio of a data object. The calculation formula is as follows:

$$SDROF_i = \frac{ReS_i}{ldr_i} \tag{7}$$

In (7), ReS_i represents the relative skewness of the data object x_i, and ldr_i represents the local density ratio of the data object x_i.

This outlier factor is composed of the ratio of relative skewness and local density ratio. From a distance perspective, this section combines relative distance with skewness, using relative skewness to characterize the degree to which data objects deviate from their neighbors. Typically, outliers have greater distances from their neighbors, indicating a higher degree of deviation. From a density standpoint, the local density ratio represents the relative difference in density between a data object and its neighbors through a ratio. This effectively captures the density differences between data objects and their neighbors, regardless of whether they are in high-density or low-density regions. In general, outliers tend to have lower local densities compared to their neighbors, resulting in smaller local density ratios. Therefore, the relative skewness density ratio outlier factor (SDROF) highlights the outlier characteristics, enabling better outlier detection. The larger the outlier value, the more likely it is to be an outlier.

Definition 9 (Factor Difference of x_i). Factor Difference (FD_i) refers to the average difference between the outlier values of data object x_i and its neighbors. The calculation formula is as follows:

$$FD_i = \frac{\sum_{j \in NaN_i} \left| SDROF_i - SDROF_j \right|}{\lambda} \tag{8}$$

In (8), λ represents the number of natural neighbors. To better characterize the local differences between data objects and their neighbors, this section introduces the factor difference to calculate the deviation of outlier values between data objects and their neighbors. Normal data objects exhibit small differences in outlier values with their neighbors, and their outlier values are close to each other. Conversely, outliers demonstrate distinct characteristics from their neighbors; the outlier values of outliers are significantly larger than those of surrounding normal data objects, resulting in larger factor differences. By incorporating factor differences on top of the relative skewness density ratio outlier factor, this method further extracts local differentiation information, enabling a better characterization of the outlier degree of data objects.

3.2 Description of SDROF algorithm

Based on the relevant definitions proposed in Section 3.1, this section presents an outlier detection algorithm based on relative skewness density ratio outlier factor. The algorithm first employs the natural neighbor search algorithm described in Algorithm 1 to select the natural neighbors for each data object and determine the adaptively selected number of neighbors, λ. Then, it executes the outlier detection algorithm based on relative skewness density ratio outlier factor as described in Algorithm 2. In Algorithm 2, it first computes the skewness and relative distances for each data object and calculates the relative skewness value based on the concept of relative skewness, which better characterizes the deviation degree of data objects under complex data distributions. Next, it calculates the density for each data object and the average density of its natural neighbors, using their ratio as the local density ratio for the object. The local density ratio reflects the relative density size and is more adaptable to complex density distributions. Subsequently, it calculates the ratio of relative skewness and local density ratio as the relative skewness-density ratio outlier factor (SDROF), which highlights the outlier characteristics of data points. Finally, it computes the deviation of outlier values between each data

object and its neighbors as the final outlier value measurement, denoted as FD_i. The factor difference, FD_i, further extracts local differentiation information on top of the relative skewness-density outlier factor, enabling a better characterization of the outlier degree of data objects. The SDROF algorithm ultimately selects the top o data objects with the largest factor differences as outliers, where the value of o is determined based on the number of labeled outliers in the dataset. Algorithm 2 describes the specific steps of the outlier detection algorithm based on relative skewness density ratio outlier factor.

Algorithm 2 Outlier detection algorithm based on relative skewness density ratio outlier factor (SDROF).

Input: D (the dataset), λ (the natural characteristic value), NaN (the set of natural neighbors).
Output: o outliers.
Initial:$KNN(x) = \varnothing, NaN(x) = \varnothing, Nb(x) = 0, r = 1, num = 0$;
for each $x \in D$ **do**
  Compute the local density of x using (2).
  Compute the skewness of x using (3).
end for
for each $x \in D$ **do**
  Compute the KNN relative distance of x using (4).
end for
for each $x \in D$ **do**
  Compute the relative skewness of x using (5).
  Compute the local density ratio ldr_i of x using (6).
end for
for each $x \in D$ **do**
  Compute the relative skewness density outlier factor ($SDROF_i$) of x using (7).
end for
for each $x \in D$ **do**
  Calculate the factor difference (FD_i) of x using (8).
end for
Sort the factor differences of each data object in descending order.
Output the top o data objects as outliers.

3.3 SDROF algorithm analysis

3.3.1 Rationality analysis

In the SDROF algorithm, first, the natural neighbors of each data object are identified using Algorithm 1. The natural neighbor search algorithm avoids the uncertainty introduced by manually setting the value of k and speeds up the search process by using the KD-Tree indexing structure. Then, based on the natural neighbors, the relative skewness and local density ratio of each data object are computed to characterize the degree to which a data object deviates from its neighbors and the relative density. Finally, the ratio of relative skewness to local density ratio is used as the relative skewness density ratio outlier factor (SDROF), and the factor difference is introduced to highlight the local differences between the data object and its neighbors, further characterizing the degree of outlier of the data object. A larger factor difference indicates a higher likelihood of being an outlier.

3.3.2 Time complexity analysis

The time complexity of the SDROF algorithm can be decomposed into three parts: (1) Natural Neighbor Search: When executing the natural neighbor search algorithm using the KD-Tree indexing structure, the time complexity is $O(n \cdot logn)$, where n is the number of data objects. (2) Computing Relative Skewness and Local Density Ratio: Calculating the relative skewness and local density ratio for each data object has a time complexity of $O(n \cdot m \cdot \lambda) + O(n \cdot \lambda)$, where m is the dimensionality of the dataset and λ is the natural neighbor feature, representing the adaptively selected number of neighbors. (3) Calculating Factor Difference for Each Data Object: The time complexity of computing the factor difference for each data object is $O(n \cdot \lambda)$. Therefore, the overall time complexity of the SDROF algorithm is $O(n \cdot logn) + O(n \cdot m \cdot \lambda) + O(n \cdot \lambda) + O(n \cdot \lambda)$, which can be simplified to $O(n(logn + m \cdot \lambda))$.

3.3.3 Space complexity analysis

The space complexity of SDROF consists of two main parts: (1) Algorithm 1 requires storing the natural neighbor information for all data objects, leading to a space complexity of $O(\lambda \cdot n)$, where λ represents the number of adaptively selected neighbors and n is the number of data objects. (2) Algorithm 2 needs to store results such as local density ratios, relative skewness, relative skewness density ratios, factor differences, and temporary data structures generated during computation, resulting in a space complexity of $O(n)$. Thus, the overall space complexity of the SDROF algorithm is $O(\lambda \cdot n)$.

4 Experimental evaluation and analysis

The experimental setup is illustrated in Table 1, primarily encompassing the software and hardware environment, along with the parameters corresponding to each configuration.

To evaluate the performance of the SDROF algorithm, experiments are conducted using seven comparison algorithms: LOF [25], COF [26], NANOD [37], ADD [38], RDOF [39], MOD [24], and DOD [24]. These seven algorithms are all based on nearest neighbors and are thus highly comparable to the SDROF algorithm presented in this paper, demonstrating its detection performance. Additionally, the SDROF algorithm does not require parameter selection, whereas the parameters for comparison algorithms are kept consistent with those in the original literature. The compared algorithms are detailed in Table 2.

Table 1 Experimental setup

Software and hardware environment	Parameters
CPU	Inter Core i9-12900H 2.5GHz 14-core
Memory	32.0 GB
Hard Disk	512 GB
Operating System	64-bit Windows 11
Develop Environment	PyCharm
Compilation Environment	Python 3.8
Virtualization Tools	Python 3.8

4.1 Experimental evaluation indicators

This paper will evaluate the performance of the proposed algorithm using three evaluation metrics: precision (Pr), area under the ROC curve (AUC), and Rank Power (RP) [39]. Precision is used to indicate the proportion of correctly detected outliers to the total number of outliers, as shown in (9).

$$Pr = \frac{o}{n} \tag{9}$$

In (9), n represents the total number of outliers in the dataset, and o represents the number of truly outliers among the top n outliers output by the algorithm. As can be observed from the equation, a higher value of Pr indicates that the algorithm has correctly detected more outliers.

Due to the presence of class imbalance in most datasets, particularly in the field of outlier detection where outliers constitute only a small portion of the dataset, the Area Under the Receiver Operating Characteristic Curve (AUC) is insensitive to the balance of sample classes. In situations of sample imbalance, it can still provide a reasonable evaluation of the classifier. Therefore, using the AUC value as an evaluation metric is highly appropriate. The ROC graph can be determined by evaluation of all possible thresholds, suggesting that the number of samples correctly classified (abnormal scores) known to be true-positive changes with the number of false-positive samples (ordinary or inliers) [37]. From a probabilistic perspective, AUC can be understood as the probability that the predicted probability of positive samples exceeds that of negative samples. In this paper, positive samples refer to outliers, with values ranging from 0 to 1. A higher AUC value indicates a better performance in predicting outliers. The AUC can be expressed as:

$$AUC = \int_0^1 ROC(t)\, dt \tag{10}$$

Precision is used to assess the detection capability of the algorithm, but it overlooks the positions of the detected outliers. This paper employs Rank Power to determine the order of true outliers among the outlier scores generated by the algorithm. For instance, if the algorithm outputs 10 outliers, among which 5 are true outliers, these 5 outliers can be positioned either near the top or near the bottom. Although the precision calculated remains the same, the algorithm's performance differs. Sorting the outlier scores in descending order, the better the performance of the algorithm, the higher the position of the outliers. The formula for calculating the Rank Power (RP) is shown as follows:

$$RP = \frac{t_0(t_0+1)}{2(\sum_{i=1}^{t_0} R_i)} \tag{11}$$

In (11), t_0 represents the number of true outliers among the first t data objects, and R_i denotes the position of the i-th true outlier. For a fixed value of t, a higher RP value indicates better performance of the algorithm. If all the first t data objects are true outliers, the maximum value of RP is 1.

Table 2 The compared algorithms

Algorithm	Basis	Publication and year
LOF	Density	ACM SIGMOD, 2000
COF	Density	Advances in Knowledge Discovery and Data Mining, 2002
NANOD	Density	Neural Computing and Applications, 2021
ADD	Distance	Applied Intelligence, 2022
RDOF	Density	Expert Systems, 2022
MOD	Meanshift	Pattern Recognition, 2021
DOD	Meanshift	Pattern Recognition, 2021

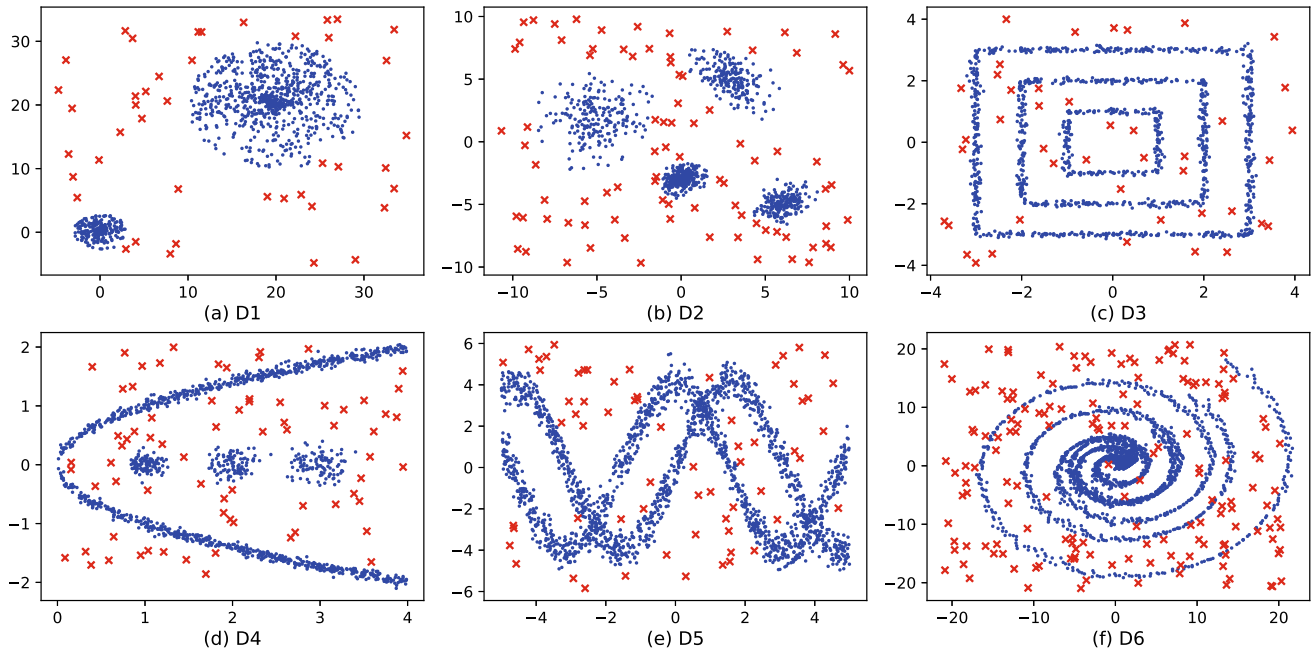


Fig. 3 Data distribution of synthetic Datasets from D1 to D6

4.2 Description of experimental data for algorithm SDROF

To examine the outlier detection capability of the proposed algorithm under different data distributions, the SDROF algorithm conducted comparative experiments using six synthetic two-dimensional datasets and four UCI real-world datasets. Figure 3 displays the data distributions of the six synthetic datasets, where the red "×" denotes outliers and the remaining blue "•" represents normally distributed points. Table 3 presents the characteristics of different synthetic datasets, including the number of samples in each dataset, the total number of outliers, and the proportion of outliers in the total samples.

Combining the data distribution and data characteristics from Fig. 3 and Table 3, it is evident that dataset D1 consists of two spherical clusters of different sizes, with the sparse cluster located in the top right corner and the dense cluster in the bottom left corner. This dataset contains a total of 1043 samples, including 43 outliers. Dataset D2 exhibits a more complex data distribution, composed of four spherical clusters of varying sizes, with the cluster in the top left corner being the sparsest, presenting a low-density pattern issue. There are 1000 samples in this dataset, with a total of 85 outliers, including both global and local outliers. Dataset D3 comprises three nested rectangular clusters, with outliers scattered around these clusters. It consists of 1256 samples, including 43 outliers. Dataset D4 is composed of clusters of different shapes, including one U-shaped cluster and three spherical clusters, totaling 1372 samples. Dataset D5 consists of two S-shaped clusters, with outliers dispersed around the clusters. There are 2024 samples in this dataset, with a total of 64 outliers. Dataset D6 consists of a spiral cluster, with the density of the cluster gradually becoming sparse from the center of the spiral outward. Outliers are distributed around the cluster. It contains 2259 samples, including 159 outliers. Overall, the six synthetic datasets exhibit complex data distributions, with variations in cluster density, sparsity, and shape across datasets. They encompass both global and local outliers, serving as effective benchmarks for evaluating

Table 3 The data characteristics of the synthetic datasets utilized in the experiments for the SDROF algorithm and other algorithms

Dataset	Number of instances	Number of outliers	Outlier ratio
D1	1043	43	4.1%
D2	1000	85	8.5%
D3	1256	43	3.4%
D4	1372	72	5.2%
D5	2024	64	3.2%
D6	2259	159	7.0%

Table 4 The data characteristics of real datasets used in the SDROF algorithm and other algorithm experiments

Dataset	Number of instances	Number of attribute	Number of outliers
Wdbc	390	30	33 (8.5%)
Ecoli	168	7	25 (14.9%)
Pendigits	1641	17	20 (1.2%)
Vowels	1456	12	50 (3.4%)
Annthyroid	7200	6	534 (7.42%)
Ionosphere	351	33	126 (36%)
Letter	1600	32	100 (6.25%)
Pima	768	8	268 (35%)
Toxicity	171	1203	56 (32.7%)
Period Changer	90	1177	27 (30%)

algorithms' outlier detection capabilities in diverse scenarios.

Additionally, this paper utilizes 10 real-world datasets from the UCI (University of California, Irvine) Machine Learning Repository to evaluate the SDROF algorithm. In the domain of outlier detection, these datasets require additional preprocessing. Specifically, for datasets with imbalanced classes, the majority class is labeled as normal, while the minority class is designated as anomalous. For datasets with reasonably balanced classes, a uniform downsampling of one of the majority classes is employed to generate the minority class.The Wdbc dataset consists of 390 samples of breast cancer cases, including 33 outliers, accounting for 8.5% of the dataset. Each sample is described by 30 attributes. The Ecoli dataset comprises 168 samples of protein localization sites, with 25 outliers, representing 14.9% of the dataset. Each sample contains 7 attributes. The Pendigits dataset contains 1641 handwritten digit samples, including 20 outliers, making up 1.2% of the dataset. Each sample is characterized by 17 attributes. The Vowels dataset includes 1456 Japanese vowel samples, with 50 outliers, accounting for 3.4% of the dataset. Each sample consists of 12 attributes. The attribute range across all real datasets falls between 7 and 30, while the outlier ratio ranges from 1.2% to 14.9%. The Annthyroid dataset comprises 7200 thyroid disease samples, including 534 outliers accounting for 7.42% of the total. Each sample in this dataset possesses 6 attributes. The Ionosphere dataset consists of 351 ionosphere data samples, with 126 outliers representing 36% of the total. Each sample in this dataset has 33 attributes. The Letter dataset encompasses 1600 letter samples, among which 100 are outliers, contributing to a 6.25% outlier ratio. Each sample in this dataset contains 32 attributes. The Pima dataset is made up of 768 diabetes samples from Native Americans, featuring 268 outliers, which translates to a 35% outlier ratio. Each sample in this dataset has 8 attributes. In addition, two high-dimensional datasets (Number of attribute > 1000) are selected to assess SDROF's performance. The Toxicity dataset comprises 171 molecules , including 56 outliers accounting for 32.7% of the total. Each sample in this dataset possesses 1203 attributes. Lastly, the Period Changer dataset consists of 90 non-toxic molecules , with 27 outliers representing 30% of the total. Each sample in this dataset has 1177 attributes. Table 4 presents the data characteristics of various real-world datasets, showcasing a range of attribute counts from 6 to 1203 and outlier proportions varying between 1.2% and 36%. On the whole, these 10 real-world datasets exhibit varying sample sizes, dimensionality, and outlier proportions, reflecting the diversity of data distributions within them. This diversity enables a robust evaluation of algorithms' outlier detection capabilities in realistic scenarios.

4.3 Synthetic dataset experiments and analysis

Table 5 illustrate the precision of the SDROF algorithm compared to seven other algorithms on various synthetic datasets.

Table 5 The Precision of SDROF algorithm and other algorithms on synthetic datasets

Dataset	LOF	COF	NANOD	ADD	RDOF	MOD	DOD	SDROF
D1	0.930	0.953	0.325	**1.000**	0.302	0.884	0.884	**1.000**
D2	0.847	0.764	0.717	0.864	0.8	0.788	0.776	**0.882**
D3	0.581	0.860	0.488	0.816	0.465	0.395	0.781	**1.000**
D4	0.944	0.930	0.180	**0.968**	0.847	0.722	0.861	0.958
D5	0.750	0.781	0.281	0.840	0.625	0.578	0.516	**0.843**
D6	0.465	0.610	0.477	**0.948**	0.252	0.566	0.377	0.867
AVG	0.752	0.816	0.411	0.906	0.549	0.656	0.699	**0.925**

The bold entries are used to emphasize the optimal values of the experimental results

Additionally, we selected three different types of datasets from the artificial dataset for outlier detection visualization: D1, D3, and D6. The detection results are shown in Figs. 4, 5 and 6.

In the synthetic datasets, D1 and D2 consist of spherical clusters with varying degrees of sparsity, surrounded by both local and global outliers. The distribution in D1 is simpler, with both the SDROF and ADD algorithms achieving a precision rate of 1, indicating excellent performance. Apart from NANOD and RDOF, which show weaker results, the other algorithms also perform well. From Fig. 4, it can be observed that both the LOF and COF algorithms detect almost all of the global outliers. However, due to their heavy reliance on local information of data objects, some intra-cluster points are misclassified as outliers, resulting in poorer performance compared to the SDROF and ADD algorithms. The NANOD and RDOF algorithms exhibit similar detection patterns; they fail to detect most of the global outliers and had a high misclassification rate, leading to lower detection accuracy. The MOD and DOD algorithms produce similar results, detecting nearly all of the global outliers but missing some of the local outliers. The D2 dataset, however, is more complex than D1, consisting of four clusters with densely distributed local outliers, making it more challenging for algorithms to distinguish. Nonetheless, the SDROF algorithm still demonstrate strong detection capabilities, achieving a precision rate of 0.882. This is attributed to the use of relative skewness in the SDROF algorithm to characterize the degree of deviation

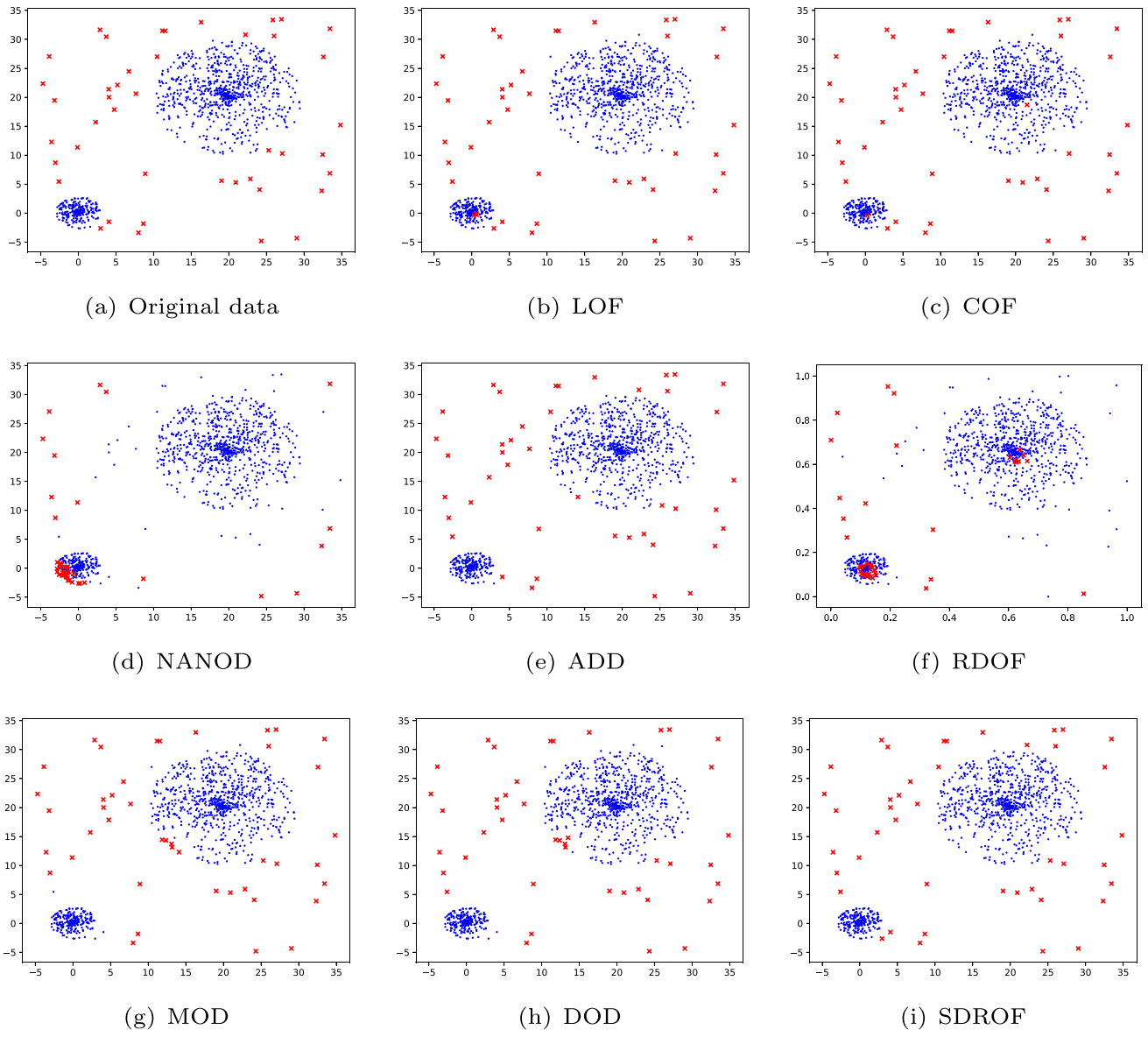


Fig. 4 Outliers detected by LOF, COF, NANOD, ADD, RDOF, MOD, DOD, SDROF on D1

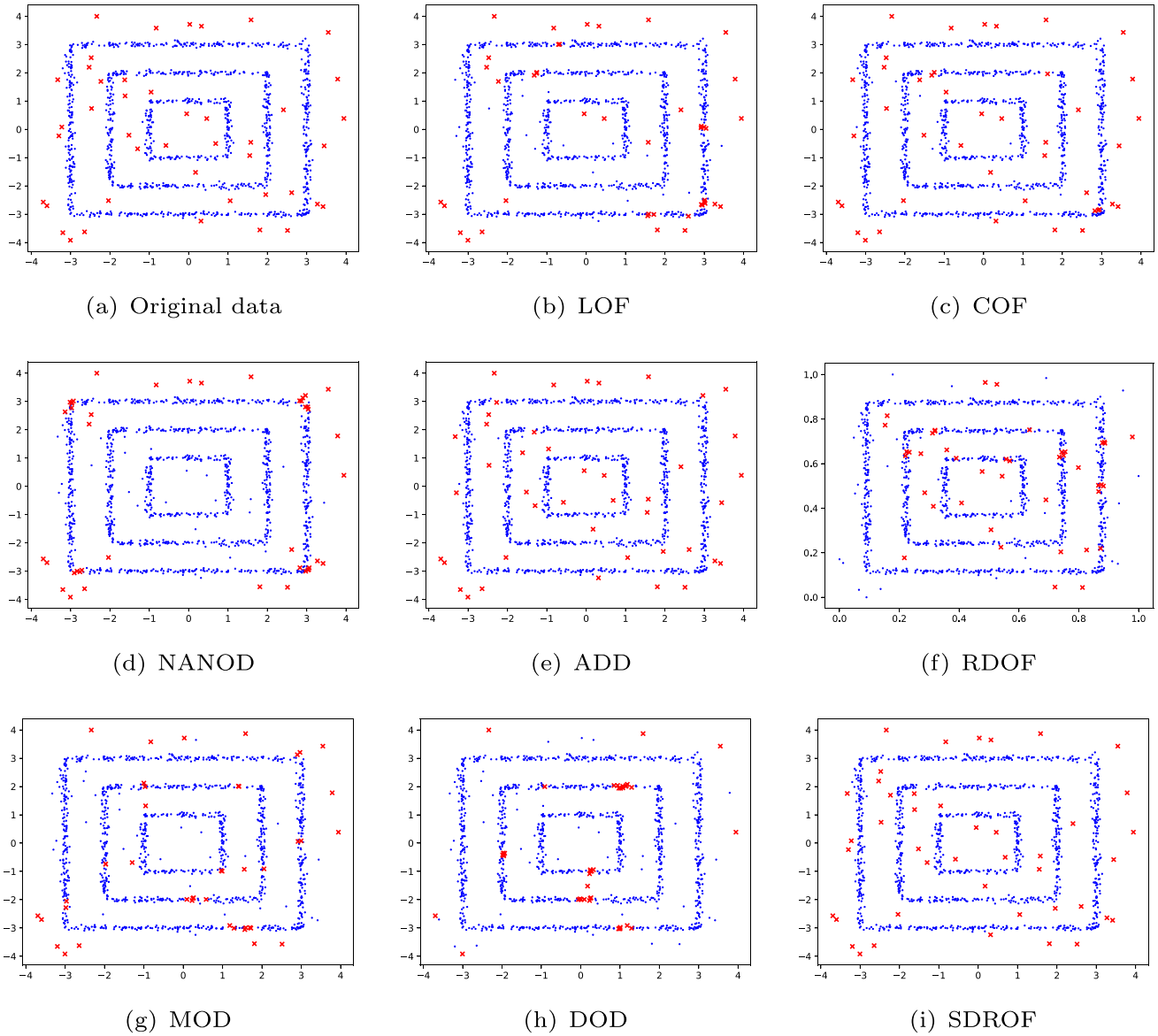


Fig. 5 Outliers detected by LOF, COF, NANOD, ADD, RDOF, MOD, DOD, SDROF on D3

of data objects, which effectively identifies global outliers, while the local density ratio enhances the detection of local outliers. Figure 5 shows that in the line-shaped cluster of the D3 dataset, only SDROF and ADD effectively detected all local outliers. Other algorithms either miss some outliers or exhibit high misclassification rates, highlighting SDROF's effectiveness in non-spherical clusters. D3 and D4 are composed of linear clusters with varying shapes. According to Table 5, the SDROF algorithm achieves precision rates of 1 and 0.958, respectively, on D3 and D4, demonstrating its effectiveness in identifying outliers in non-spherical clusters. The ADD algorithm slightly outperforms SDROF on the D4 dataset in terms of precision. D5 and D6 aim to address spiral-shaped datasets with outliers. D5 consists of two spiral clusters with numerous local outliers, while D6 has a single spiral cluster with uniformly distributed outliers. As seen in Table 5, the ADD algorithm performs best on the D5 dataset, followed closely by SDROF, while the other algorithms struggled. On the D6 dataset, SDROF achieves the best detection performance, followed by ADD, with NANOD and RDOF showing weaker results. Figure 6 shows that LOF, COF, and RDOF have similar outlier detection patterns with high misclassification rates at cluster centers, resulting in lower accuracy. NANOD, MOD, and DOD also have similar patterns, with high misclassification rates at the edges or certain parts of the spiral shape, reducing their accuracy. In contrast, SDROF and ADD detect most outliers with lower misclassification rates. In summary, the SDROF algo-

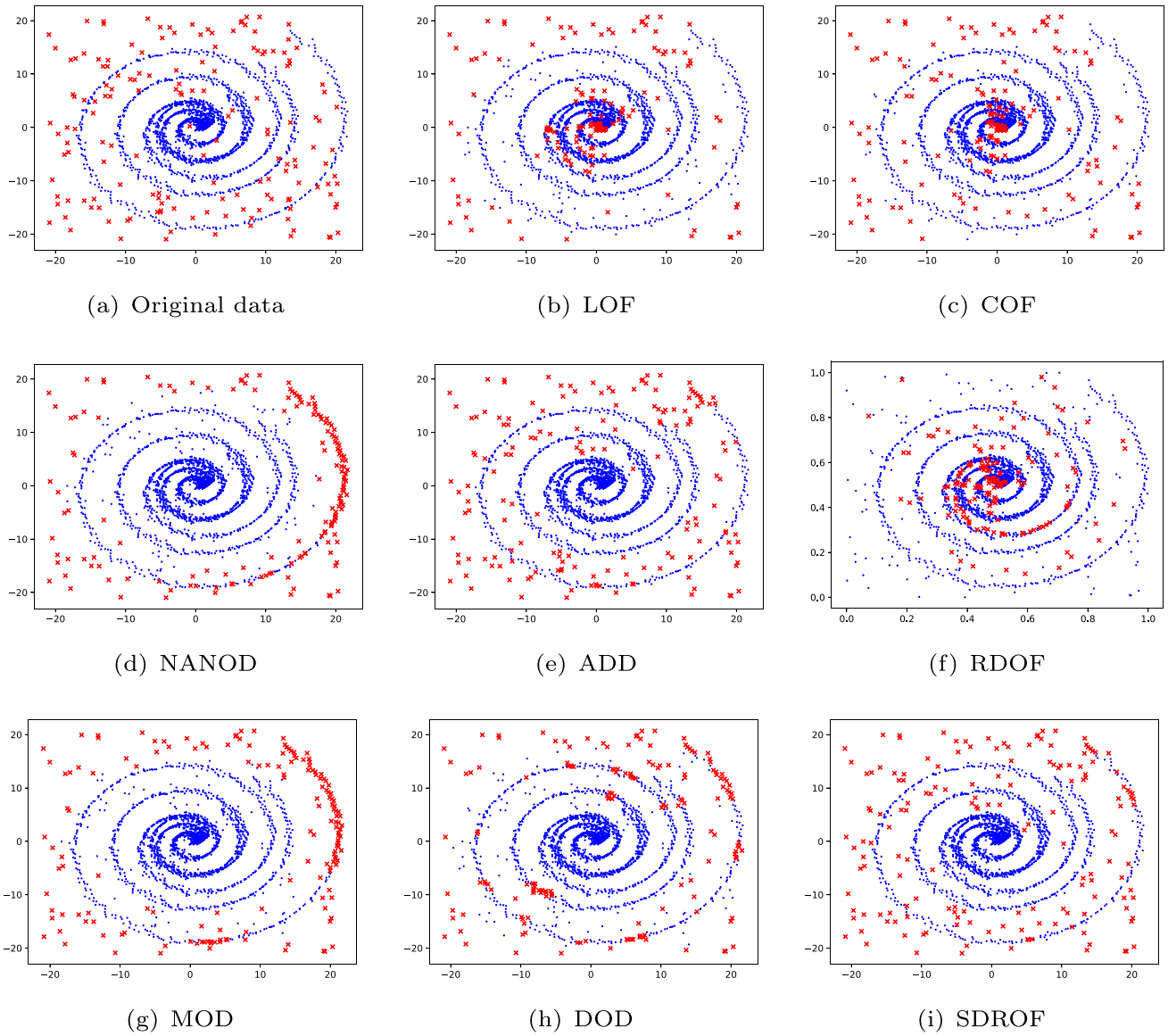


Fig. 6 Outliers detected by LOF, COF, NANOD, ADD, RDOF, MOD, DOD, SDROF on D6

rithm proposed in this paper consistently outperforms the other algorithms across the six synthetic datasets, demonstrating high detection accuracy on datasets with various cluster shapes and complex density distributions.

The AUC values of the SDROF algorithm and the other seven comparative algorithms on different synthetic datasets are shown in Table 6.

From the experimental data in Table 6, it can be observed that the AUC values of the SDROF algorithm on six synthetic datasets are superior to those of other comparative algorithms. Specifically, on datasets D1 and D2, the AUC values are the highest, reaching 1, indicating that the SDROF algorithm can correctly identify each data object as an outlier. Although the performance of the SDROF algorithm is slightly inferior on dataset D6, with an AUC value of 0.988, it is still effective. Among the comparative algorithms, the ADD algorithm closely follows the SDROF algorithm, exhibiting excellent performance on all datasets, with the highest AUC value of 0.999 achieves on dataset D1. The COF algorithm demonstrates stable performance, with an average AUC value around 0.97 across all datasets. The LOF algorithm has a low AUC value of only 0.78 on the D6 dataset, but performs well on other datasets, with the best performance on the D1 dataset, with an AUC value of 0.999. The NANOD and RDOF algorithms exhibit the least satisfactory performance among all the compared algorithms, with low AUC values across all synthetic datasets and insufficient stability. The MOD and DOD algorithms also underperform, outperforming only the NANOD and RDOF algorithms when compared to the other seven comparative algorithms. Over-

Table 6 The AUC value of SDROF algorithm and other algorithms on synthetic datasets

Dataset	LOF	COF	NANOD	ADD	RDOF	MOD	DOD	SDROF
D1	0.999	0.998	0.854	0.999	0.899	0.997	0.998	**1.000**
D2	0.993	0.977	0.894	0.992	0.965	0.986	0.984	**0.995**
D3	0.935	0.998	0.862	0.998	0.870	0.780	0.825	**1.000**
D4	0.998	0.998	0.796	0.998	0.958	0.987	0.995	**0.999**
D5	0.988	0.990	0.752	0.995	0.973	0.958	0.952	**0.996**
D6	0.780	0.907	0.913	0.979	0.684	0.911	0.875	**0.988**
AVG	0.948	0.978	0.845	0.993	0.892	0.937	0.938	**0.996**

The bold entries are used to emphasize the optimal values of the experimental results

all, the SDROF algorithm demonstrates exceptional outlier detection capabilities, maintaining a stable performance in identifying outliers even when confronted with various complex data distributions.

The RP values of the SDROF algorithm and the other seven comparative algorithms on different synthetic datasets are presented in Table 7, where the value of t is chosen as the actual number of outliers in the dataset.

From the experimental data in Table 7, it can be observed that the SDROF algorithm achieves the maximum RP value of 1 on datasets D1 and D3. This implies that the SDROF algorithm can identify all true outliers among the top t outliers when sorted in descending order. As the distribution of datasets becomes more complex, such as in datasets D5 and D6, although the RP values of the SDROF algorithm gradually decrease, its performance remains satisfactory. Among the comparative algorithms, the ADD algorithm demonstrates stability across all six datasets, with its outlier ranking relatively high among the top t positions. However, on dataset D6, the RP value of the ADD algorithm is only 0.649, indicating poor performance. The LOF, COF, MOD, and DOD algorithms show strong RP values on the simpler D1, D2, and D4 datasets but perform poorly on more complex datasets. The NANOD and RDOF algorithms exhibit suboptimal performance on synthetic datasets, with relatively low RP values. Overall, the SDROF algorithm demonstrates superior RP values across six synthetic datasets compared to other methods, effectively identifying outliers and positioning them higher in the ranking. This underscores the effectiveness of the SDROF algorithm on synthetic datasets as proposed in this study.

4.4 Experimental analysis on real datasets

According to the experimental data presented in Table 8, the SDROF algorithm achieves the highest average precision of 0.595 across 10 real-world datasets. Specifically, The SDROF algorithm achieves the highest precision on the Wdbc, Pendigits, Vowels, Letter and Period Changer datasets, and it also attains the second-best results on the Annthyroid, Ionosphere, and Pima datasets. Its performance is average on the Ecoli dataset. In the Ecoli dataset, the ADD algorithm demonstrates the highest precision, reaching a value of 1. This result is due to ADD's use of a threshold-based method for outlier detection. There are 25 outliers in this dataset, and ADD correctly identifies 13 of them, resulting in very high precision. However, despite ADD's precision of 1, it does not detect all outliers, identifying only about half. Moreover, in the Period Changer dataset, the precision of the ADD algorithm is a mere 0. This unsatisfactory result can be attributed to its threshold-based approach, which only enables the ADD algorithm to detect three outliers, all of which are identified incorrectly. The SDROF algorithm performs slightly better than the COF and MOD algorithms on the Ecoli dataset. In addition, compared to other comparative algorithms, SDROF demonstrates good performance and strong stability on the high-dimensional Toxicity and Period Changer datasets.

Table 7 The RP value of SDROF algorithm and other algorithms on synthetic datasets

Dataset	LOF	COF	NANOD	ADD	RDOF	MOD	DOD	SDROF
D1	0.958	0.940	0.131	0.993	0.179	0.898	0.910	**1.000**
D2	0.880	0.676	0.307	0.869	0.572	0.774	0.743	**0.920**
D3	0.218	0.926	0.116	0.924	0.123	0.076	0.094	**1.000**
D4	0.944	0.959	0.121	0.954	0.402	0.684	0.852	**0.985**
D5	0.583	0.624	0.062	0.778	0.377	0.279	0.256	**0.844**
D6	0.147	0.291	0.306	0.649	0.108	0.301	0.234	**0.772**
AVG	0.621	0.736	0.173	0.861	0.294	0.502	0.515	**0.920**

The bold entries are used to emphasize the optimal values of the experimental results

Table 8 The Precision of SDROF Algorithm and Other Algorithms on Real Datasets

Dataset	LOF	COF	NANOD	ADD	RDOF	MOD	DOD	SDROF
Wdbc	0.606	0.363	0.787	0.435	0.58	0.823	0.839	**0.848**
Ecoli	0.800	0.600	0.840	**1**	0.8	0.37	0.88	0.760
Pendigits	**0.900**	0.600	0.85	0.096	0.1	0.9	0.9	**0.900**
Vowels	0.340	0.500	0.52	0.230	0.28	0.44	0.38	**0.660**
Annthyroid	0.29	0.24	0.13	**0.41**	0.16	0.26	0.25	0.31
Ionosphere	0.76	0.77	0.74	**1**	0.58	0.68	0.71	0.84
Letter	0.46	0.43	0.18	0.46	0.22	0.33	0.25	**0.58**
Pima	0.37	0.37	0.45	**0.5**	0.38	0.47	0.45	0.47
Toxicity	**0.321**	0.303	0.267	0.25	0.285	0.267	0.25	0.285
Period Changer	0.222	0.259	0.185	0	0.259	0.259	0.222	**0.296**
AVG	0.51	0.44	0.493	0.441	0.336	0.48	0.513	**0.595**

The bold entries are used to emphasize the optimal values of the experimental results

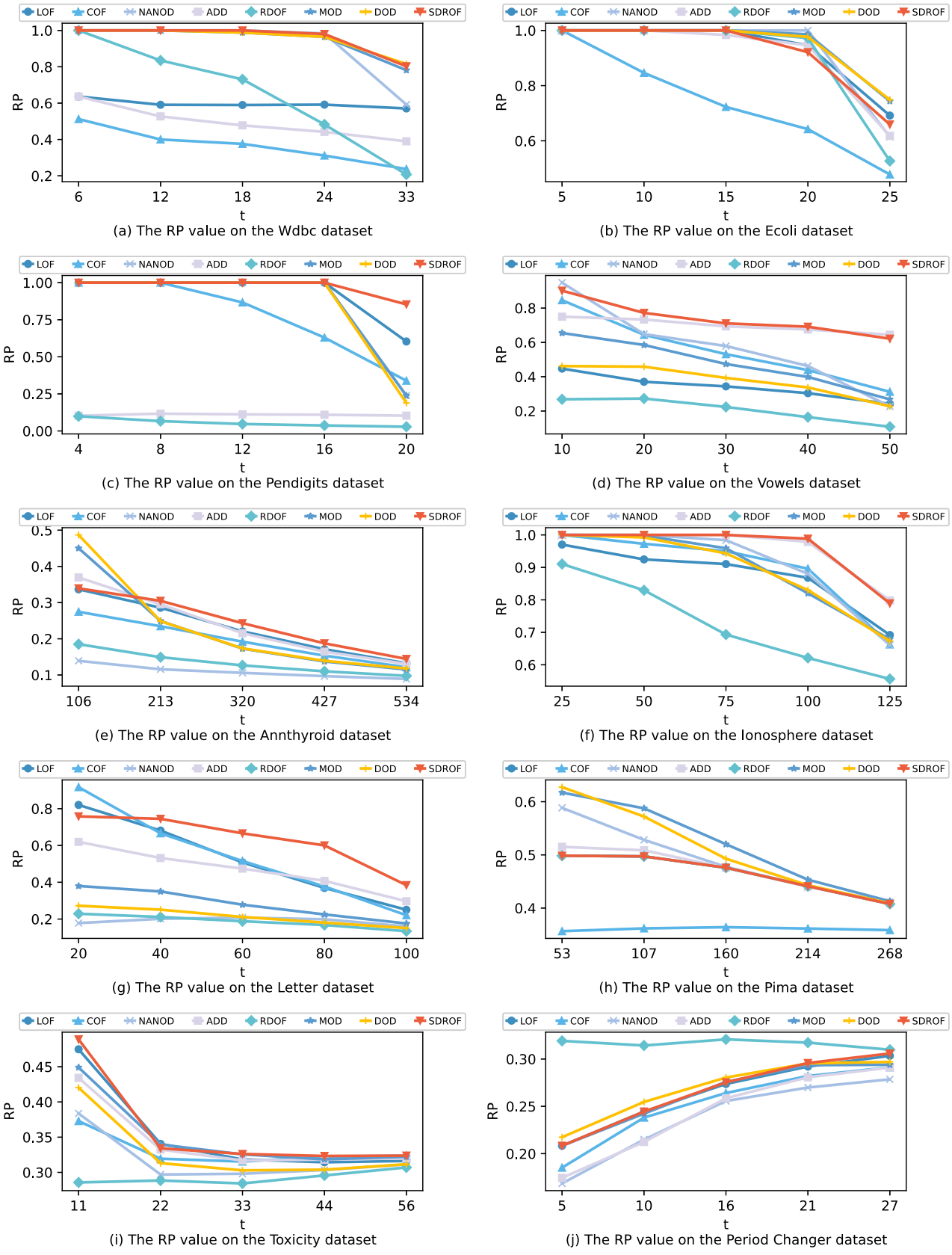
From the experimental data presented in Table 9, it can be observed that the SDROF algorithm achieves the highest average AUC value of 0.815 across 10 real datasets. Specifically, The SDROF algorithm achieves the highest AUC values on the Wdbc, Pendigits, Vowels, Annthyroid, Letter, Toxicity and Period Changer datasets, and it attains the second-best result on the Ionosphere dataset. The SDROF algorithm also performs well on the Ecoli and Pima datasets, with AUC values of 0.953 and 0.609, respectively. Although SDROF's precision on the Ecoli dataset is only better than that of the COF and MOD algorithms, its AUC value ranks third among all comparison algorithms and is only marginally lower than that of the best-performing algorithm. This indicates that the SDROF algorithm's overall performance is stable and shows improvement. Additionally, SDROF achieves the highest AUC values among the competing algorithms on the high-dimensional Toxicity and Period Changer datasets, further confirming its superiority in high-dimensional contexts. Overall, the SDROF algorithm outperforms other comparison algorithms in terms of AUC, demonstrating excellent outlier detection capability on real datasets, and maintains effective outlier detection even with multidimensional and multi-feature datasets.

In Fig. 7, the x-axis represents the number of top t outliers, while the y-axis denotes the RP values. The total number of outliers for each dataset is divided into five segments, and the RP value for each segment is computed. Considering the RP values across the five sections, the SDROF algorithm achieves optimal performance on the Wdbc, Pendigits, Vowels, Annthyroid, Ionosphere, Letter and Toxicity datasets. On the Ecoli dataset, when $t \leq 15$, the RP value of SDROF consistently remains at 1, indicating that the top t outliers identified by SDROF are all correct. When $t > 20$, the RP values for SDROF and the other seven comparison algorithms rapidly decline, up to $t = 25$. At $t = 25$, the RP value of SDROF is second only to LOF, MOD, and DOD algorithms. On this dataset, the COF algorithm performs the worst overall, while the RDOF algorithm shows the greatest fluctuation in RP values, indicating the least stability. The performance of other algorithms is relatively similar.

Table 9 The AUC value of SDROF Algorithm and Other Algorithms on Real Datasets

Dataset	LOF	COF	NANOD	ADD	RDOF	MOD	DOD	SDROF
Wdbc	0.964	0.846	0.967	0.925	0.817	0.817	0.980	**0.988**
Ecoli	0.959	0.900	0.942	0.943	0.918	0.918	**0.969**	0.953
Pendigits	0.995	0.987	0.979	0.943	0.779	0.779	0.900	**0.998**
Vowels	0.942	0.960	0.938	0.984	0.850	0.850	0.380	**0.988**
Annthyroid	0.737	0.709	0.589	0.732	0.629	0.691	0.699	**0.762**
Ionosphere	0.874	0.856	0.855	**0.928**	0.774	0.866	0.863	0.924
Letter	0.899	0.881	0.823	0.920	0.782	0.842	0.810	**0.946**
Pima	0.542	0.518	0.614	0.613	0.537	**0.617**	0.611	0.609
Toxicity	0.464	**0.482**	0.451	0.473	0.440	0.479	0.452	**0.482**
Period Changer	0.489	0.458	0.424	0.458	0.484	0.466	0.474	**0.495**
AVG	0.787	0.76	0.758	0.792	0.7	0.732	0.714	**0.815**

The bold entries are used to emphasize the optimal values of the experimental results



(a) The RP value on the Wdbc dataset

(b) The RP value on the Ecoli dataset

(c) The RP value on the Pendigits dataset

(d) The RP value on the Vowels dataset

(e) The RP value on the Annthyroid dataset

(f) The RP value on the Ionosphere dataset

(g) The RP value on the Letter dataset

(h) The RP value on the Pima dataset

(i) The RP value on the Toxicity dataset

(j) The RP value on the Period Changer dataset

Fig. 7 Comparison of the RP the value between SDROF algorithm and other algorithms on real datasets

In the Pendigits dataset, the RP values for the ADD and RDOF algorithms consistently decline and differ significantly from those of other algorithms. This is because both algorithms determine outliers based on thresholds. Specifically, the ADD algorithm identifies a significantly higher number of outliers compared to the actual number of outliers in the original dataset, while the RDOF algorithm detects far fewer outliers than are actually present. These discrepancies lead to notably poor precision and RP values for these two algorithms. On the Pima dataset, when $t \leq 160$, the PR values of DOD, MOD, NANOD, and ADD algorithms are higher than that of SDROF. However, as t increases, the gap between SDROF and these algorithms gradually narrows. From $t > 160$ onwards, the RP value of SDROF begins to converge with or even surpass other algorithms, demonstrating its stable outlier detection capability. In the Period Changer dataset, although the RDOF algorithm consistently achieves the highest RP value, it performs poorly on the other nine datasets. In contrast, the SDROF algorithm maintains a high RP value across all datasets. Overall, the SDROF algorithm shows superior outlier detection performance on real datasets compared to other algorithms, with minimal fluctuation in RP values as t changes. This experiment confirms that the SDROF algorithm offers stable and efficient outlier detection performance on multi-dimensional datasets.

4.5 Execution time analysis

In this study, we not only focus on the performance of the algorithms in outlier detection but also on the execution time efficiency of the SDROF algorithm. Although the computational complexity of the SDROF algorithm has been discussed in detail, we also provide a comprehensive comparison of the execution times of the SDROF algorithm against other comparative algorithms, as shown in Table 10.

SDROF's execution time is generally lower than that of NANOD, ADD, MOD and DOD algorithms, which exhibit higher computational costs across various datasets. Although SDROF's execution time is somewhat less favorable compared to LOF and COF, it maintains stable performance across different dataset sizes. Specifically, SDROF has an average execution time of 4.14 seconds, which is more efficient compared to NANOD (54.22 seconds) and ADD (53.54 seconds). This indicates that SDROF performs efficiently in handling datasets of varying scales and complexities. In practical applications, where dataset sizes are continually growing, the algorithm's computational efficiency is crucial. SDROF not only performs well but also reduces computation time, enhancing processing capability and user experience.

4.6 Ablation study

To evaluate the contribution of each component of the SDROF algorithm, we conducted an ablation study on 10 real-world datasets. The study involved comparing the performance of the full model (baseline) with several variants where specific components were removed (Table 11).

We performed ablation experiments by systematically removing individual components of the SDROF algorithm. The following variants are tested:

- **Baseline Model**: The complete SDROF algorithm with all components included.
- **Relative Skewness**: Model with component of Relative Skewness removed.
- **Local Density Ratio**: Model with component of Local Density Ratio removed.
- **Factor Difference**: Model with component of Factor Difference removed.

From the ablation study results, we observe that removing the relative skewness component from the baseline reduces global outlier detection ability and accuracy, while increasing misclassification rates due to a greater emphasis on local

Table 10 Execution time of SDROF compared to other algorithms on different datasets

Dataset	LOF	COF	NANOD	ADD	RDOF	MOD	DOD	SDROF
Wdbc	0.61	0.10	4.03	5.89	0.72	6.14	5.54	0.45
Ecoli	0.01	0.03	0.22	0.24	0.10	0.36	0.40	0.12
Pendigits	0.15	1.30	49.80	44.90	1.61	7.90	7.18	3.30
Vowels	0.06	0.68	27.81	24.60	1.32	5.95	4.66	2.53
annthyroid	0.15	11.63	331.04	329.92	8.98	22.71	21.97	19.47
Ionosphere	0.01	0.09	3.62	3.93	0.64	5.56	4.76	1.21
Letter	0.06	0.74	71.32	72.07	2.35	8.96	6.74	8.72
Pima	0.01	0.24	4.86	5.06	0.72	2.27	1.72	1.08
Toxicity	0.47	1.09	40.95	39.09	5.11	2.96	2.93	3.17
Period Changer	0.01	0.07	8.5	9.65	1.86	1.31	1.4	1.3
AVG	0.152	1.60	54.22	53.54	2.35	6.41	5.73	4.14

Table 11 Ablation Study Results on Real-word Datasets

Dataset	Model Variant	Pr	AUC	RP
Wdbc	Baseline Model	**0.848**	**0.988**	**0.803**
	Relative Skewness	0.212	0.665	0.125
	Local Density Ratio	**0.848**	**0.988**	0.800
	Factor Difference	0.818	0.960	0.546
Ecoli	Baseline Model	**0.76**	**0.953**	0.659
	Relative Skewness	0.24	0.722	0.246
	Local Density Ratio	**0.76**	0.949	**0.692**
	Factor Difference	**0.76**	0.935	0.584
Pendigits	Baseline Model	**0.9**	**0.999**	**0.854**
	Relative Skewness	0.45	0.969	0.175
	Local Density Ratio	**0.9**	0.998	0.820
	Factor Difference	**0.9**	0.998	0.820
Vowels	Baseline Model	0.66	**0.988**	0.622
	Relative Skewness	0.32	0.877	0.129
	Local Density Ratio	**0.7**	0.980	**0.651**
	Factor Difference	**0.7**	**0.988**	0.626
Annthyroid	Baseline Model	**0.311**	**0.762**	**0.144**
	Relative Skewness	0.219	0.743	0.135
	Local Density Ratio	**0.311**	0.743	0.135
	Factor Difference	0.297	**0.762**	**0.144**
Ionosphere	Baseline Model	**0.849**	**0.925**	**0.789**
	Relative Skewness	0.769	0.894	0.728
	Local Density Ratio	**0.849**	0.924	0.786
	Factor Difference	**0.849**	0.923	**0.789**
Letter	Baseline Model	**0.58**	**0.946**	**0.384**
	Relative Skewness	0.5	0.897	0.247
	Local Density Ratio	0.55	0.932	0.331
	Factor Difference	0.54	0.940	0.362
Pima	Baseline Model	**0.47**	**0.609**	**0.408**
	Relative Skewness	0.332	0.5	0.349
	Local Density Ratio	0.467	0.603	0.406
	Factor Difference	**0.47**	0.603	**0.408**
Toxicity	Baseline Model	**0.285**	**0.482**	0.324
	Relative Skewness	**0.285**	0.480	**0.325**
	Local Density Ratio	**0.285**	0.480	0.324
	Factor Difference	0.214	0.395	0.290
Period Changer	Baseline Model	**0.296**	**0.495**	0.306
	Relative Skewness	0.185	0.431	0.281
	Local Density Ratio	**0.296**	0.484	**0.318**
	Factor Difference	0.333	0.473	0.306

The bold entries are used to emphasize the optimal values of the experimental results

outliers. This combined effect leads to significantly lower accuracy compared to the baseline. Removing the local density ratio component weakens the algorithm's ability to detect local outliers, causing a slight decrease in accuracy, but it has minimal impact on global outlier detection, resulting in a smaller performance decline compared to the baseline. Removing the factor difference component has little effect on performance since it is intended to further refine local differentiation on top of the relative skewness density ratio, contributing to more stable outlier detection. Overall, the relative skewness component is crucial for algorithm performance, with its removal leading to significant declines in precision, AUC value, and RP value. In contrast, the local density ratio and factor difference components have a smaller impact on model performance. While these components contribute to model effectiveness, their influence is not as pronounced as that of the relative skewness component. Nonetheless, they play a significant role in maintaining the algorithm's effectiveness and stability. These findings provide valuable insights for future algorithm optimization and component prioritization.

5 Conclusion and future work

This paper addresses the problem of detecting outliers with low-density patterns and low local density in proximity-based outlier detection methods. We propose an outlier detection algorithm based on the relative skewness density ratio outlier factor (SDROF). Firstly, the algorithm employs an adaptive natural neighbor search algorithm to select neighbors for each data object, avoiding the uncertainty caused by manually selecting the value of k. Then, relative skewness is introduced along with relative distance to better characterize the deviation of data objects in complex data distributions. Secondly, the ratio of a data object's density to the average density of its natural neighbors is used as its local density ratio to highlight variations in local density. Thirdly, the ratio of relative skewness to local density ratio is defined as the relative skewness density ratio outlier factor, which highlights the outlier characteristics of outlier points. Finally, the deviation of outlier values between each data object and its neighbors is computed to further describe the local differences between data objects and their neighbors. The SDROF algorithm was compared with LOF, COF, NANOD, ADD, RDOF, MOD, and DOD algorithms in terms of Precision (Pr), AUC values, and Rank Power (RP). The results demonstrate that the proposed SDROF algorithm achieves high detection accuracy and exhibits excellent performance across various datasets with complex distributions.

While the proposed algorithm demonstrates promising performance across three evaluation metrics, showing robust outlier detection capabilities, there are still some issues to

be addressed, such as the efficiency problem when dealing with large-scale datasets. Therefore, enhancing detection efficiency on large-scale datasets will be a crucial direction for future research.

Acknowledgements This work was supported by a grant from The National Natural Science Foundation of China (No.61972334), the National Social Science Foundation of China General Project (No.20BJ 122), the Innovation Capability Improvement Plan Project of Hebei Province (No.22567626H), the Local Science and Technology Development Fund Project guided by the Central Government (No.226Z1707G), and the Intelligent image workpiece recognition of Sida Railway (No.x2021134).

Author Contributions Zhongping Zhang presents the core idea of the model and the experimental method. Kuo Wang implements the model, verifies its validity, and writes the paper. Jinyu Dong and Sen Li provides guidance and revised the paper.

Data Availability The datasets generated during and/or analysed during the current study are available from the corresponding author on reasonable request.

Declarations

Competing Interest The author(s) declare no potential conflicts of interest with respect to the research, authorship and/or publication of this paper.

Compliance with Ethical Standards No ethical data in this paper.

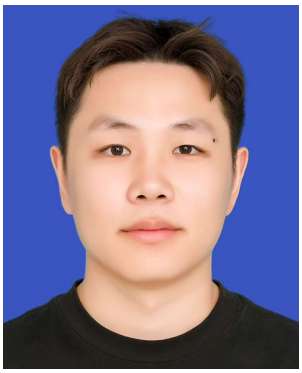
References

1. Rebjock Q, Kurt B, Januschowski T et al (2021) Online false discovery rate control for anomaly detection in time series. Adv Neural Inf Process Syst 34:26487–26498
2. Shen L, Li Z, Kwok J (2020) Timeseries anomaly detection using temporal hierarchical one-class network. Adv Neural Inf Process Syst 33:13016–13026
3. Aggarwal CC (2017) An Introduction to outlier analysis. Springer, Berlin
4. Safaei M, Asadi S, Driss M et al (2020) A systematic literature review on outlier detection in wireless sensor networks. Symmetry 12(3):328. https://doi.org/10.3390/sym12030328
5. Chakraborty D, Narayanan V, Ghosh A (2019) Integration of deep feature extraction and ensemble learning for outlier detection. Patt Recogn 89:161–171. https://doi.org/10.1016/j.patcog.2019.01.002
6. Andrysiak T (2020) Sparse representation and overcomplete dictionary learning for anomaly detection in electrocardiograms. Neural Comput Appl 32(5):1269–1285. https://doi.org/10.1007/s00521-018-3814-5
7. Domingues R, Filippone M, Michiardi P et al (2018) A comparative evaluation of outlier detection algorithms: Experiments and analyses. Pattern Recogn 74:406–421. https://doi.org/10.1016/j.patcog.2017.09.037
8. Bhatti MA, Riaz R, Rizvi SS et al (2020) Outlier detection in indoor localization and internet of things (iot) using machine learning. J Commu Netw 22(3):236–243. https://doi.org/10.1109/JCN.2020.000018
9. Alghushairy O, Alsini R, Ma X, et al (2021) Improving the efficiency of genetic-based incremental local outlier factor algorithm for network intrusion detection. In: Advances in Artificial Intelligence and Applied Cognitive Computing: Proceedings from ICAI'20 and ACC'20, Springer, pp 1011–1027
10. Djenouri Y, Belhadi A, Lin JCW et al (2019) A survey on urban traffic anomalies detection algorithms. IEEE Access 7:12192–12205. https://doi.org/10.1109/ACCESS.2019.2893124
11. Maskey SR, Badsha S, Sengupta S, et al (2020) Bits: Blockchain based intelligent transportation system with outlier detection for smart city. In: 2020 IEEE International Conference on Pervasive Computing and Communications Workshops (PerCom Workshops), IEEE, pp 1–6
12. Ruff L, Vandermeulen RA, Görnitz N, et al (2019) Deep semi-supervised anomaly detection. CoRR arXiv:1906.02694 https://doi.org/10.48550/arXiv.1906.02694
13. Breiman L (2001) Random forests. Mach Learn 45:5–32. https://doi.org/10.1023/A:1010933404324
14. LeCun Y, Bengio Y, Hinton G (2015) Deep Learn Nat 521(7553):436–444. https://doi.org/10.1038/nature14539
15. Akcay S, Atapour-Abarghouei A, Breckon TP (2019) Ganomaly: Semi-supervised anomaly detection via adversarial training. In: Computer Vision–ACCV 2018: 14th Asian Conference on Computer Vision, Perth, Australia, 2–6 December, 2018, Revised Selected Papers, Part III 14, Springer, pp 622–637
16. Akçay S, Atapour-Abarghouei A, Breckon TP (2019) Skip-ganomaly: Skip connected and adversarially trained encoder-decoder anomaly detection. In: 2019 International Joint Conference on Neural Networks (IJCNN), IEEE, pp 1–8
17. Zenati H, Romain M, Foo CS, et al (2018) Adversarially learned anomaly detection. In: 2018 IEEE International conference on data mining (ICDM), IEEE, pp 727–736
18. Ester M, Kriegel HP, Sander J, et al (1996) A density-based algorithm for discovering clusters in large spatial databases with noise. In: kdd, pp 226–231
19. He Z, Xu X, Deng S (2003) Discovering cluster-based local outliers. Pattern Recogn Lett 24(9–10):1641–1650. https://doi.org/10.1016/S0167-8655(03)00003-5
20. Rodriguez A, Laio A (2014) Clustering by fast search and find of density peaks. Sci 344(6191):1492–1496. https://doi.org/10.1126/science.1242072
21. Knorr EM, Ng RT (1998) Algorithms for mining distancebased outliers in large datasets. In: Proceedings of the international conference on very large data bases, Citeseer, pp 392–403
22. Ramaswamy S, Rastogi R, Shim K (2000) Efficient algorithms for mining outliers from large data sets. In: Proceedings of the 2000 ACM SIGMOD international conference on Management of data, pp 427–438
23. Zhang K, Hutter M, Jin H (2009) A new local distance-based outlier detection approach for scattered real-world data. In: Advances in Knowledge Discovery and Data Mining: 13th Pacific-Asia Conference, PAKDD 2009 Bangkok, Thailand, 27-30 April, 2009 Proceedings 13, Springer, pp 813–822
24. Yang J, Rahardja S, Fränti P (2021) Mean-shift outlier detection and filtering. Pattern Recogn 115:107874. https://doi.org/10.1016/j.patcog.2021.107874
25. Breunig MM, Kriegel HP, Ng RT, et al (2000) Lof: identifying density-based local outliers. In: Proceedings of the 2000 ACM SIGMOD international conference on Management of data, pp 93–104
26. Tang J, Chen Z, Fu AWC, et al (2002) Enhancing effectiveness of outlier detections for low density patterns. In: Advances in Knowledge Discovery and Data Mining: 6th Pacific-Asia Conference, PAKDD 2002 Taipei, Taiwan, 6–8 May, 2002 Proceedings 6, Springer, pp 535–548

27. Gao J, Hu W, Zhang Z, et al (2011) RKOF: robust kernel-based local outlier detection. In: Pacific-Asia conference on knowledge discovery and data mining, Springer, pp 270–283
28. Schubert E, Zimek A, Kriegel HP (2014) Generalized outlier detection with flexible kernel density estimates. In: Proceedings of the 2014 SIAM international conference on data mining, SIAM, pp 542–550
29. Jin W, Tung AK, Han J, et al (2006) Ranking outliers using symmetric neighborhood relationship. In: Advances in Knowledge Discovery and Data Mining: 10th Pacific-Asia Conference, PAKDD 2006, Singapore, 9-12 April, 2006. Proceedings 10, Springer, pp 577–593
30. Xiong ZY, Long H, Zhang YF et al (2023) A neighborhood weighted-based method for the detection of outliers. Applied Intell 53(9):9897–9915. https://doi.org/10.1007/s10489-022-03258-0
31. Zhang J, Yang Y (2023) Density-distance outlier detection algorithm based on natural neighborhood. Axioms 12(5):425. https://doi.org/10.3390/axioms12050425
32. Li K, Gao X, Jia X et al (2022) Detection of local and clustered outliers based on the density-distance decision graph. Eng Appl Art Intell 110:104719. https://doi.org/10.1016/j.engappai.2022.104719
33. Guha S, Rastogi R, Shim K (1998) Cure: An efficient clustering algorithm for large databases. ACM Sigmod record 27(2):73–84. https://doi.org/10.1145/276305.276312
34. Yang L, Zhu Q, Huang J et al (2017) Adaptive edited natural neighbor algorithm. Neurocomput 230:427–433. https://doi.org/10.1016/j.neucom.2016.12.040
35. Zhu Q, Feng J, Huang J (2016) Natural neighbor: A self-adaptive neighborhood method without parameter k. Pattern Recogn Lett 80:30–36. https://doi.org/10.1016/j.patrec.2016.05.007
36. Li X, Han Q, Qiu B (2018) A clustering algorithm using skewness-based boundary detection. Neurocomput 275:618–626. https://doi.org/10.1016/j.neucom.2017.09.023
37. Wahid A, Annavarapu CSR (2021) Nanod: A natural neighbour-based outlier detection algorithm. Neural Comput Appl 33(6):2107–2123. https://doi.org/10.1007/s00521-020-05068-2
38. Xiong ZY, Gao QQ, Gao Q, et al (2022) Add: a new average divergence difference-based outlier detection method with skewed distribution of data objects. Applied Intell pp 1–25. https://doi.org/10.1007/s10489-021-02399-y
39. Wahid A, Rao ACS (2022) Rdof: An outlier detection algorithm based on relative density. Exp Syst 39(2):e12859 https://doi.org/10.1111/exsy.12859

Zhongping Zhang is a professor at the School of Information Science and Engineering, Yanshan University. He obtained his Ph.D. degree from Fudan University, and his research directions include big data, data mining, semi-structured data, etc.

Kuo Wang is a graduate student pursuing a master's degree at the School of Information Science and Engineering, Yanshan University. His major is Computer Science and Technology. His research areas focus on big data and data mining.



Jinyu Dong is a graduate student pursuing a master's degree at the School of Information Science and Engineering, Yanshan University. His major is Computer Science and Technology. His research areas likewise concentrate on big data and data mining.

Sen Li received master's degree at the School of Information Science and Engineering, Yanshan University. His major is Computer Technology. His research interests lie in big data and data mining.