Multimed Tools Appl (2018) 77:11277–11305
https://doi.org/10.1007/s11042-017-5523-y

An effective information detection method for social big data

Jinrong He[1] · **Naixue Xiong**[2]

Received: 2 May 2017 / Revised: 7 December 2017 / Accepted: 8 December 2017 /
Published online: 19 December 2017

Abstract In data mining and knowledge discovery applications, outlier detection is a fundamental problem for robust machine learning and anomaly discovery. There are many successful outlier detection methods, including Local Outlier Factor (LOF), Angle-Based Outlier Factor (ABOF), Local Projection Score (LPS), etc. In this paper, we assume that outliers lie in lower density region and they are at relatively larger distance from any points with a higher local density. In order to identify such outliers quantitatively, the paper proposed a decision graph based outlier detection (DGOD) method. The DGOD method works by firstly calculating the decision graph score (DGS) for each sample, where the DGS is defined as ratio between discriminant distance and local density, next ranking samples according to their DGS values, and finally, returning samples with top-r largest DGS values as outliers. Experimental results on synthetic and real-world datasets have confirmed its effectiveness on outlier detection problems, and it is a general and effective information detection method, which is robust to data shape and dimensionality.

Keywords Outlier detection · Decision graph · Local density · Discriminant distance · Outlier score · Social big data

✉ Jinrong He
hejinrong@nwafu.edu.cn

✉ Naixue Xiong
xiongnaixue@gmail.com

[1] College of Information Engineering, Northwest A & F University, Yangling, Shaanxi Province 712100, China

[2] School of Computer Science, Colorado Technical University, Colorado Spring, CO, USA

1 Introduction

Due to vast amounts of multiple distributed sensors, social media has become the most representative and relevant data sources for big data. Information detection is one of the most popular topics in social big data research, especially outlier and detection is the critical issue [4, 36]. In reality, within the massive data, there are inevitable exceptional behaviors or inconsistent patterns that often exhibit as the representations of noises or interesting facts, such as cyber-intrusion and terrorist activities [37]. In data mining and knowledge discovery applications, outliers are also referred as anomalies, deviants, or discordants of data generating process, which will lead to model misspecification, biased parameter estimation and incorrect results [22]. The detection of such unusual characteristics provides useful application-specific insights [1], such as network intrusion detection [10, 12], social media security and trustworthiness evaluation [38, 41], credit card fraud detection, clinical trials, voting irregularity analysis, severe weather prediction, athlete performance analysis, terrorist activity investigation et al. To address these challenges, many researchers have proposed several outlier detection methods according to different definitions of outliers. Johnson [18] suggests that an outlier is an observation which appears to be inconsistent with the rest of the dataset. Barnett and Lewis [3] define outliers as objects that appears to deviate markedly from other samples in which it occurs. The most popular definition of outlier is proposed by Hawkins [13], and he defined outlier as a sample that appears to deviate so much from other samples as to arouse suspicion that it was generated by a different mechanism. Since outlier detection can reveal unusual behaviors, interesting patterns and exceptional events from datasets, it is of great interest to the communities of machine learning and data mining.

There are two perspectives on outlier detection problem formulation, one is learning to rank which output a score about the level of "outlierness" of a sample, and the other is learning to classification which output a binary label indicating whether a sample is an outlier or not. Due to its nature of unsupervised learning, how to detect outliers from normal data samples with noise is often a subjective process. In this work, we formulate the outlier detection problem as a ranking problem, and present a quantified outlierness measure of a sample.

Many efforts have been devoted to detect outliers. In statistical analysis, the simplest method for outlier detection is Z-value test which assumes that the data samples are modeled from a normal distribution. The samples with more than 3 standard deviations from the mean are recognized as outliers. But it is not always the case for real data. Generally speaking, existing outlier detection methods falls into three main categories: distance-based methods, density-based methods and clustering-based methods.

Distance-based methods consider the data points having large average distances to the k-th nearest neighbors as outliers. Distance-based outlier was originally proposed by Knorr and Ng [19] in 1998, in which a sample x_i in a dataset X is a DB(p,T)-outlier if at least fraction p of the samples in X lies greater than distance T from x_i. Although a number of efficient algorithms for detecting distance-based outliers are proposed, it doesn't provide a ranking for outliers. Based on the distance of each

sample from its k-th nearest neighbor, Sridhar et al. [26] proposed a partition-based algorithm to rank each point and declare the top r points in this ranking to be outliers. To detect outliers in scattered datasets, Zhang et al. proposed Local Distance-based Outlier Factor (LDOF) [39] which is defined as the ratio of the average of distances from a data point to its k-nearest neighbors over the average of pairwise distances among these k-1 data points, and then the degree to which a sample deviates from its neighborhood system is captured. To circumvent the distance concentration problem in high-dimensional space, Liu et al. [21] introduced Local Projection Score (LPS) to represent deviation degree of a sample to its neighbors, in which the LPS can be computed by the technique of low-rank approximation.

Though distance-based methods are simple and elegant, they didn't work well for datasets that have more complex structures and are sensitive to data locality [5]. Density-based methods are robust to data locality, which assume that the density around an outlier is significantly different from densities around its neighbors. Then outliers can be identified by comparing the density of a sample's neighborhood with that of its neighbor's neighborhood. The most popular density-based method is Local Outlier Factor(LOF) [5], in which outliers are detected by measuring the local deviation of a given data point with respect to its neighbors, and outliers are considered as data points that have a substantially lower density than their neighbors. However, selecting neighborhood parameter k in LOF is non-trivial. Following the idea of local density measure, several extensions to the basic LOF model have been proposed. Tang et al. proposed the Connectivity-based Outlier Factor (COF) [32] to deal with the case that a cluster and a neighboring outlier have similar neighborhood densities. Rather than examining an individual sample, the Local Correlation Integral (LOCI) [25] method looks for groups of outliers. LOCI method provides an "outlier plot" which gives the user an idea on how data is distributed in the vicinity of the analyzed sample. From the plot, one can assess whether the sample is inside a cluster, a part of a micro-cluster or if it is an outstanding outlier. Different with LOF, LOCI uses ε-neighborhoods rather than k-nearest neighbors, and can deal with multi-granularity problem in the dataset. Jin et al. proposed a measure of outlierness named INFLO [17] by considering the union of a point's k-nearest neighbors and its reverse nearest neighbors. In 2011, in order to detect anomalies from high-dimensional data streams, Zhang et al. [40] developed a method named Stream Projected Otlier Detector (SPOD), which constructs sparse subspace template and then anomalies are more likely to be detected in the set of subspaces. To achieve good detection performance, application-dependent feature selection and data partition based on different temporal contexts are conducted. S. Hido et al. [14] proposed density-ratio based outlier detection approach which find outliers in the test set based on the training set consisting only of inliers. The outliers are recognized by the ratio of training and testing data densities. This method can be viewed as supervised outlier detection. However, there are not training data in most cases, and the performance of this method heavily depends on the accuracy of density ratio estimation which is a challenging problem for high-dimensional data sets.

The clustering problem has a complementary relationship to the outlier detection problem, in which points either belong to clusters or outliers. Clustering-based

methods define outliers as clusters of small size, especially including the size of one data point. Density-Based Spatial Clustering of Applications with Noise (DBSCAN, [9]) detects outliers by checking the connections between data samples and clusters, and the samples that do not belong to any clusters or belong to small clusters are identified as outliers. In 2008, Jiang et al. [16] proposed a clustering-based outlier detection method, which consists of two stages, firstly, dataset is clustered by on-pass clustering algorithm, then the outlier factors of clusters are determined. The robust clustering with outliers is also called noise clustering, which define outliers in terms of noise distance. Rehm et al. [27] proposed a method to estimate the noise distance in noise clustering based on the preservation of the hypervolume of the feature space. However, how to estimate the hypervolume of the feature space is also a challenge. In 2011, Shi et al. [31] proposed Cluster-Outlier Iterative Detection (COID) method, in which clusters are obtained firstly, then the intra-relationship and inter-relationship are defined. After performing the alteration of clusters and outliers iteratively, COID method consistently outputs natural clusters and outliers. To circumvent sensitivity of parameter k in k-nearest neighbors based methods, Wang et al. [34] proposed a minimum spanning tree clustering based global outlier factor and local outlier factor. In order to handle large-scale datasets, a robust Novel Local Outlier Detection (NLOD, [7]) method is proposed, which finds density peaks of dataset by 3σ standard firstly, then all the samples are clustered by neareast neighbor criterion, finally the local outliers of each cluster are identified by Chebyshev's inequality and density peak reachability. However, the performance of clustering-based methods are highly dependent on the effectiveness of the clustering algorithm in capturing the cluster structure of normal samples.

Furthermore, there are some other methods are designed for special background. For high dimensional data, Kriegel et al. proposed Angle-Based Outlier Factor (ABOF) [20] to evaluate the variance in angles among the difference vectors from the analyzed sample to other samples in the dataset. However, ABOF only considers the relationships between each sample and its neighbors and does not consider the relationships among these neighbors, thus it may not detect outliers correctly. Scholkipf et al. [30] extended SVM to outlier detection which aims to separate data samples into outliers and inliers by a hyperplane in a Gaussian reproducing kernel Hilbert space. However, setting of tuning parameters in this method is difficult. Manifold is a useful tool to model the structure of data samples. Since manifold learning methods are sensitive to outliers, Onderwater [23] proposed an outlier detection method based on Local Reconstruction Weights (LRW). The samples with large local reconstruction weights can be considered as outliers. By utilizing the concept of the center of gravity, Ha et al. [11] introduced the instability factor of a sample to detect local and global outliers. A sample with a high instability factor is a promising candidate for an outlier. This approach eliminates the problem of density calculation in the neighborhood of a sample, but with a high computational cost. Recently, Huang et al. [15] proposed Rank-Based Detection Algorithm (RBDA) and the degree of isolation of a sample is measured with sum of ranks of a sample. Dufrenois et al. [8] proposed one class Fisher's linear discriminant criterion and its kernelized version to detect isolate outliers from normal samples.

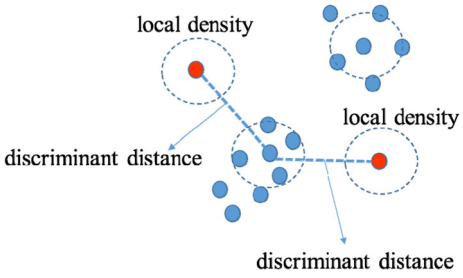


Fig. 1 Graphical illustration of DGOD method

All these methods are closely related, since they are based on the proximity of samples. However, these methods do not work well if there are various degrees of cluster density in dataset. Also, it is difficult to select appropriate values for the model parameters, such as the size of the neighborhood around a sample. What is more, they are often unsuitable for high-dimensional datasets and for arbitrary datasets without prior knowledge of the underlying data distribution. To overcome these weaknesses and detect all kinds of outliers simultaneously, motivated by the idea in [28], we proposed a Decision Graph based Outlier Detection (DGOD) method. DGOD detects outlier from clusters by incorporating the advantages of density-based and clustering-based methods. Each sample is ranked by the proposed decision graph score. The basic idea of DGOD is illustrated in Fig. 1, and the main contributions of this paper are listed as follows:

(1) Two metrics are proposed to analyze the distribution of data samples, which are called local density and discriminant distance.
(2) A simple and intuitive outlier detection criterion named decision graph score is defined to measure the outlierness of each sample, which is computational efficient and scalable for large- scale datasets.
(3) Comprehensive experiments on several synthetic and real-world datasets demonstrate the efficiency and effectiveness of DGOD method. It is not only computational efficient, but also robust to distribution and dimensionality of datasets.

The rest of the paper is organized as follows. In Section 2, we reviewed some related works on outlier detection. Then the proposed method is presented in Section 3. The experimental analysis is conducted in Section 4. Section 5 concludes the paper and Section 6 outlines the limitations of the proposed method as well as the scope for future work.

2 Related work

In this section, we discuss six related works in the area of detecting outliers, such as Local Outlier Factor (LOF), Local Reconstruction Weights (LRW) based outlier detection method, Rank-Based Detection Algorithm (RBDA), Angle-Based Outlier

Factor (ABOF), Density-Based Spatial Clustering of Applications with Noise (DBSCAN) and Local Projection Score (LPS). Table 1 presents the notations used in the remainder of the paper.

2.1 Local outlier factor (LOF)

Let $dist_k(x_i)$ be k-distance of a sample x_i which is defined as the distance between x_i and its kth nearest neighbor. Then k-distance neighbor of x_i is defined as

$$N_k(x_i) = \left\{x_j \middle| dist(x_i, x_j) \leq dist_k(x_i)\right\}$$

Noted that the cardinality of $N_k(x_i)$ is greater than k. Then reachability distance from x_i to x_j is defined as

$$reachdist_k(x_i \rightarrow x_j) = \max\left\{dist(x_i, x_j), dist_k(x_i)\right\}$$

Obviously, the reachability distance is not symmetric, i.e., $reachdist_k(x_i \rightarrow x_j) \neq reachdist_k(x_j \rightarrow x_i)$. It indicates that reachability distance measures the dissimilarity between x_i and x_j by considering their locality. The reachability distance from x_i to its k-distance neighbor is its k-distance, and the reachability distance from x_i to the samples that are not its k-distance neighbors is their actual distance. Therefore, statistical fluctuations of pairwise distances for all the nearby samples can be significantly reduced.

In order to compare the densities of different neighborhood sets of samples dynamically, local reachability density of x_i is defined as

$$lrd_k(x_i) = \frac{|N_k(x_i)|}{\sum_{x_j \in N_k(x_i)} reachdist_k(x_j \rightarrow x_i)}. \tag{1}$$

Here $|N_k(x_i)|$ is number of samples contained in $N_k(x_i)$. Then LOF of a sample x_i is defined as the average of the ratio of local reachability of x_i and those of x_i's k-nearest neighbors

$$\begin{aligned} LOF_k(x_i) &= \frac{1}{|N_k(x_i)|} \sum_{x_j \in N_k(x_i)} \frac{lrd_k(x_j)}{lrd_k(x_i)} \\ &= \sum_{x_j \in N_k(x_i)} lrd_k(x_j) \sum_{x_j \in N_k(x_i)} reachdist_k(x_j \rightarrow x_i) \end{aligned} \tag{2}$$

Table 1 Notations used in the paper

Symbol	Meaning
X	Data matrix whose columns represent samples
x_i	A sample point
d	Dimensionality of samples, i.e., number of rows of X
n	Total number of samples, i.e., number of columns of X
r	Total number of outliers
k	Neighborhood parameter
$dist(x_i,x_j)$	Distance between x_i and x_j

Note that the lower the local reachability density of x_i, and the higher the local reachability density of the k-nearest neighbors of x_j, then the higher LOF. Obviously, for most samples in a cluster, the LOF values of them are approximately equal to 1.

2.2 Local reconstruction weights (LRW) based method

LRW method derived from local linear embedding [29], and has three steps. Firstly, local reconstruction weights are computed, and then compute the reliability score of each sample point using local reconstruction weights. At last, the outliers are detected using the reliability scores. LRW method assumes that each data point can be linearly reconstructed from its neighborhoods, i.e.

$$\min \sum_i \left\| x_i - \sum_j w_{ij} x_j \right\|_2^2 \tag{3}$$

where w_{ij} is x_i's local reconstruction weight from x_j. The objective function (3) is minimized with following two constraints:

First, each data point x_i is reconstructed only from its neighbors, enforcing $w_{ij} = 0$ if x_j does not belong to the set of neighbors of x_i; Second, the rows of the weight matrix sum to one, i.e.,

$$\sum_{j=1}^{n} w_{ij} = 1 \tag{4}$$

After obtaining the optimal local reconstruction weights, the sample points with large reconstruction weights are suspected as outliers in the dataset. Therefore, the reliability score of each sample can be defined as

$$LRW(x_i) = \sum_{j=1}^{n} \left| w_{ij} \right| \tag{5}$$

2.3 Rank-based detection algorithm (RBDA)

RBDA uses mutual closeness of each data point and its neighbors to detect outliers. Different from other outlier detection methods, RBDA uses rank instead of distance. Let R be the rank matrix of dataset X. If x_j is not the k-distance neighbor of x_i, or x_i is not the k-distance neighbor of x_j, $R_{ij} = 0$; Otherwise, R_{ij} is the rank of distance between x_i and x_j among sorted distances between any other samples and x_j with ascending order.

In order to measure the outlierness of x_i, RBDA use the ranks based on neighborhood relationships between x_i and $N_k(x_i)$, which can be defined as:

$$Outlierness(x_i) = \frac{1}{|N_k(x_i)|} \sum_{x_j \in N_k(x_i)} R_{ij} \tag{6}$$

If outlierness of x_i is large, it will be suspected as an outlier.

2.4 Angle-based outlier factor (ABOF)

The angle-based outlier factor ABOF(x_i) is defined as the variance over the angles between the difference vectors of x_i to all pairs of samples in X weighted by the distance of the samples, which can be formulated as:

$$ABOF(x_i) = \underset{\forall x_j, x_t \in X}{VAR}\left(\frac{\left\langle (x_j - x_i), (x_t - x_i)\right\rangle}{\left\|x_j - x_i\right\|^2 \left\|x_t - x_i\right\|^2}\right) \tag{7}$$

where $\langle \cdot, \cdot \rangle$ denote inner product between two vectors. Since for each sample all pairs of samples must be considered, the computational cost is high. Thus in applications, the ABOF can be approximated as follows:

$$ABOF(x_i) = \underset{\forall x_j, x_t \in N_k(x_i)}{VAR}\left(\frac{\left\langle (x_j - x_i), (x_t - x_i)\right\rangle}{\left\|x_j - x_i\right\|^2 \left\|x_t - x_i\right\|^2}\right) \tag{8}$$

2.5 Density-based spatial clustering of applications with noise (DBSCAN)

In DBSCAN [9] method, outliers are identified as sample points that lie in low-density regions whose nearest neighbors are too far away. Since DBSCAN is a density-based clustering method, the points that do not belong to any of the clusters will be identified as outliers. The cluster of DBSCAN is defined as follows.

Definition 1 The **ε-neighborhood** of a sample is defined as

$$N_\varepsilon(x_i) = \left\{x_j \in X \middle| dist(x_i, x_j) < \varepsilon\right\}$$

Definition 2 A sample x_i is **directly density-reachable** from a sample x_j if $x_i \in N_\varepsilon(x_j)$ and $|N_\varepsilon(x_j)| \geq MinPts$, where $MinPts$ is a given integer.

Definition 3 A sample x_i is **density-reachable** from a sample x_j if there is a chain of samples $x_{p_1}, x_{p_2}, \cdots, x_{p_m}$, and $x_{p_1} = x_j, x_{p_m} = x_i$, such that x_{p_i+1} is directly density-reachable from x_{p_i}.

Definition 4 A sample x_i is **density-connected** to a sample x_j if there is a sample x_t sucht that both x_i and x_j are density-reachable from x_t .

Definition 5 A **cluster** C is non-empty subset of X satisfying the following conditions:

(1) For any x_i and x_j, if x_j is density-reachable from x_i that belong to C, then x_j will belong to C.
(2) For any x_i and x_j in C, xi is density-connected to x_j

Definition 6 Let C_1, …, C_s be the clusters of the dataset X, then

$$outlliers = \left\{x_i \in X \middle| \forall j : x_i \notin C_j\right\}$$

2.6 Local projection score (LPS)

LPS [21] assumes that the sparser the neighborhood of the sample, the higher probability of being outlier the sample. Since the nuclear norm of X can efficiently measure the divergence of X, the nuclear norm of neighborhood is adopted as the outlierness called LPS which is defined as

$$LPS(x_i) = \|N(x_i)\|_* \tag{9}$$

where $\|X\|_* = \sum_{i=1}^{r} \sigma_i$ and $N(x_i) = \left\{x_i^{(1)}, x_i^{(2)}, \cdots, x_i^{(k)}\right\}$. Note that the larger the LPS(xi) is, the sparser the neighborhood of xi is. The LPS(xi) can be estimated in low-dimensional embedding space, the projection procedure can be formulated as the following nuclear norm minimization problem:

$$\min \quad \frac{1}{2}\left\|X-\hat{X}\right\|_F^2 + \lambda\left\|\hat{X}\right\|_* \tag{10}$$

3 Proposed method

Compared with their neighbors, outliers can be characterized by a lower density and by a relatively large distance from points with higher densities. Based on such observation, the proposed DGOD method uses cluster structure to determine normal samples, from which the outliers are identified.

3.1 Decision graph

Definition 7 The local density ρ_i of a sample x_i is defined as

$$\rho_i = \sum_{j=1}^{m} \theta\left(d_c - dist\left(x_i, x_j\right)\right) \tag{11}$$

where $d_c > 0$ is a given threshold which can be called cutoff distance, and $\theta(\cdot)$ is an indicator function which is defined as

$$\theta(t) = \begin{cases} 1 & t \geq 0 \\ 0 & t < 0 \end{cases} \tag{12}$$

From Definition 7, we can see that the local density of each sample point is the number of samples contained in a hypersphere with the radius d_c which centered in that sample point. The lower local density, the sparser samples distributed. Therefore, local density in Definition 7 is a useful quantity measure to describe distribution of samples. In the application, we can define other form local density similarly, such as

$$\rho_i = \sum_{j=1}^{n} e^{\left(\frac{dist\left(x_i, x_j\right)}{d_c}\right)^2} \tag{13}$$

By using the potential entropy of data field, Wang et al. [35] proposed an automatic selection of the threshold value of d_c. In data field, the potential of each sample point can be calculated as follows:

$$\varphi(x_i) = \sum_{j=1}^{n} e^{\left(-\frac{dist(x_i,x_j)}{\sigma}\right)^2} \tag{14}$$

which is very similar to the equation that is used to calculate local density. Since the sample points with larger potentials located in the dense region, by using the potential entropy of data field, the optimal threshold value d_c can be calculated as $\frac{3}{\sqrt{2}}\sigma$, where impact factor σ can be chosen with smallest entropy. The entropy H of data field can be calculated as follows:

$$H = -\sum_{i=1}^{n} \frac{\varphi(x_i)}{\sum_{i=1}^{n} \varphi(x_i)} \log\left(\frac{\varphi(x_i)}{\sum_{i=1}^{n} \varphi(x_i)}\right) \tag{15}$$

Definition 8 The discriminant distance δ_i of a sample x_i is defined as

$$\delta_i = \begin{cases} \min\limits_{j:\rho_j>\rho_i} \left(dist(x_i, x_j)\right) & \rho_i \neq \max\limits_j \left(\rho_j\right) \\ \min\limits_j \left(dist(x_i, x_j)\right) & \rho_i = \max\limits_j \left(\rho_j\right) \end{cases} \tag{16}$$

Similar to reachability distance in LOF, the larger discriminant distance of x_i, the more likely it is an outlier. It represents the difference between x_i and its neighborhood samples. Particularly, for those samples located in two different cluster centers, their discriminant distance may be large, but they cannot be outliers, since their local densities are large. In order to visualize the identification of outliers intuitively, we construct the following decision graph.

Definition 9 Decision Graph. Decision graph of a dataset X is a scatter-plot (ρ_i, δ_i) which of the plot of discriminant distance δ_i along local density ρ_i for each sample.

Here is a toy example for illustrating the idea of decision graph. In Fig. 2, 40 sample points with two clusters are randomly generated. For statement conveniently, each sample point is marked with a number. Intuitively, we can see that from Fig. 2, Point 20, Point 17, Point 13 may be suspected as outliers. This suspicious can be confirmed obviously in Fig. 3, which shows the decision graph of the toy data. Samples with lower density and larger discriminant distance are located in the left corner of decision graph. It indicates that the outlier score of each sample can be calculated based on the decision graph, which is defined by local density and discriminant distance.

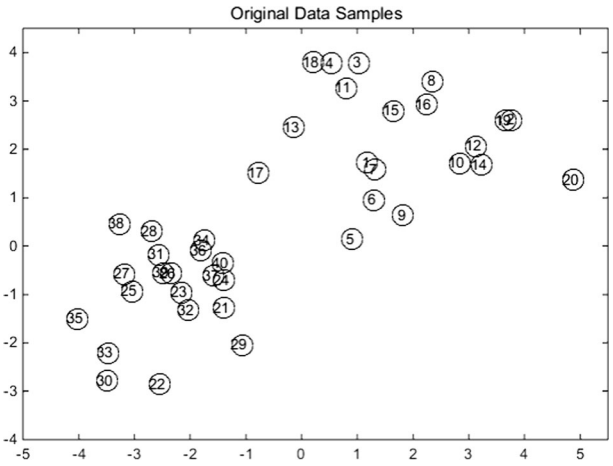


Fig. 2 Two-dimensional toy data

3.2 Outlier detection criteria

Finally, we define following outlier detection criteria based on decision graph of dataset.

Definition 10 Decision Graph Score. The decision graph score γ_i of a sample x_i is defined as

$$\gamma_i = \frac{\delta_i}{\rho_i} \tag{17}$$

Essentially, discriminant distance is determined by local density. Therefore, the decision graph score is sensitive only to the relative magnitude of local density ρ_i in different samples, and it is robust with respect to the choice of d_c. Compared with related outlier detection methods, our DGOD method is simple and intuitive.

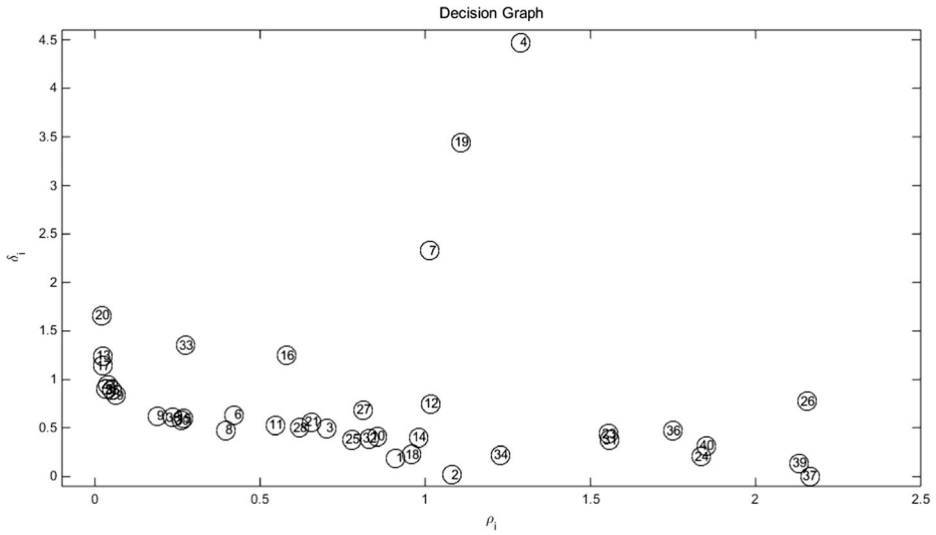


Fig. 3 Decision graph of toy data

Figure 4 shows the plot of decision graph score γ_i with descending order. We find that the samples with largest decision graph score are Point 20, Point 17 and Point 13. Figure 5 shows the original sample points marked with circles, whose radius represent the value of decision graph score. Obviously, the most suspicious outliers are Point 20, Point 17 and Point 13.

The implementation details of decision graph based outlier detection (DGOD) are summarized in Algorithm 1. It comprises two major procedures: estimating local densities and computing discriminant distances. After sorting decision graph scores in a descending order, the top r samples will be ranked as desired outliers. The dominating steps are pairwise distances computing and sorting, and the computational complexity of pairwise distance matrix computing is $O(n^2d)$ and the same as Step 2, the computational complexity of DGOD is $O(n^2d)$.

Algorithm 1: DGOD

Input: data matrix X, number of outliers r, cutoff distance d_c .

Output: projection matrix V, low-dimensional representation matrix Y.

Procedure:

Step 1. Compute pairwise distances between any two samples.

Step 2. Compute local density and discriminant distance of each sample x_i according to Equation (13) and (16).

Step 3. Compute decision graph score of each sample x_i according to Equation (17).

Step 4. Sorting decision graph scores with descending order, and selecting samples with first r largest scores as detected outliers.

4 Experimental results

In this section, we show the effectiveness of the proposed method on Synthetic and real-world datasets with known ground-truth outliers. We compared the performance of our proposed method (DGOD) with seven existing approaches, Z-value, LOF, LRW, RBDA, ABOF, DBSCAN and LPS. All methods are implemented using MATLAB 2014b running on Intel core i7 processor with an 8 GB RAM. Although d_c can be automatic selected by potential entropy of data field, it is computational expensive. In the experiments, the cutoff distance d_c is set as

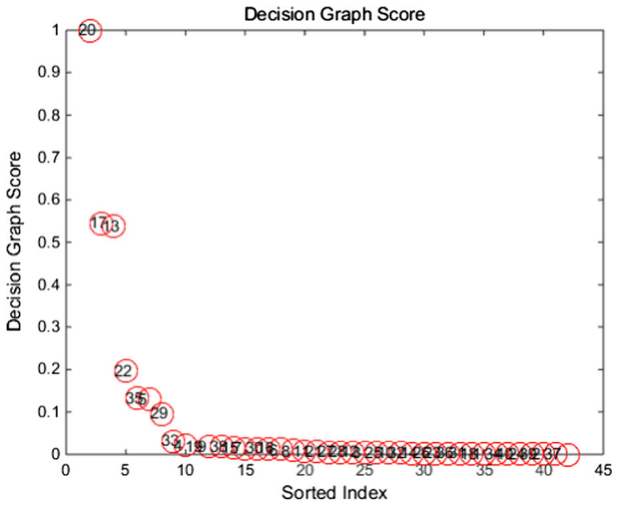


Fig. 4 Decision graph score of toy data

follows: firstly, all pairwise distances between samples are sorted with ascending order, then the distance value in the position of 2 % of total number of samples is set as cutoff distance d_c. Given a fixed number of desired outliers, the detected outliers will be obtained by different methods, and then the performances are compared with true outliers.

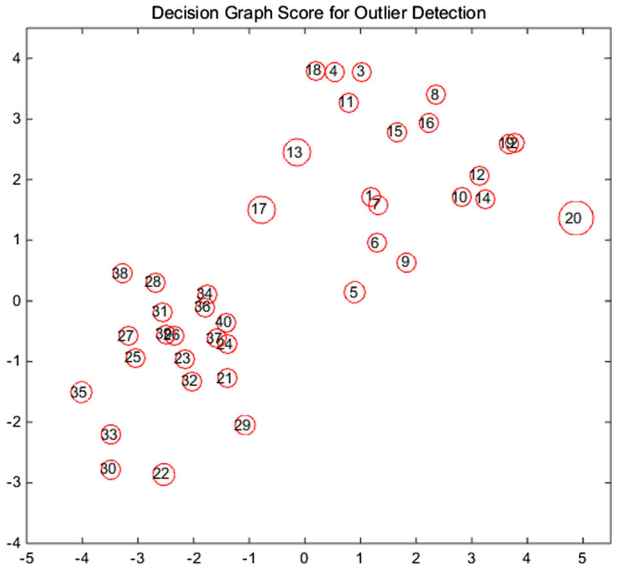


Fig. 5 Outlier detection of toy data

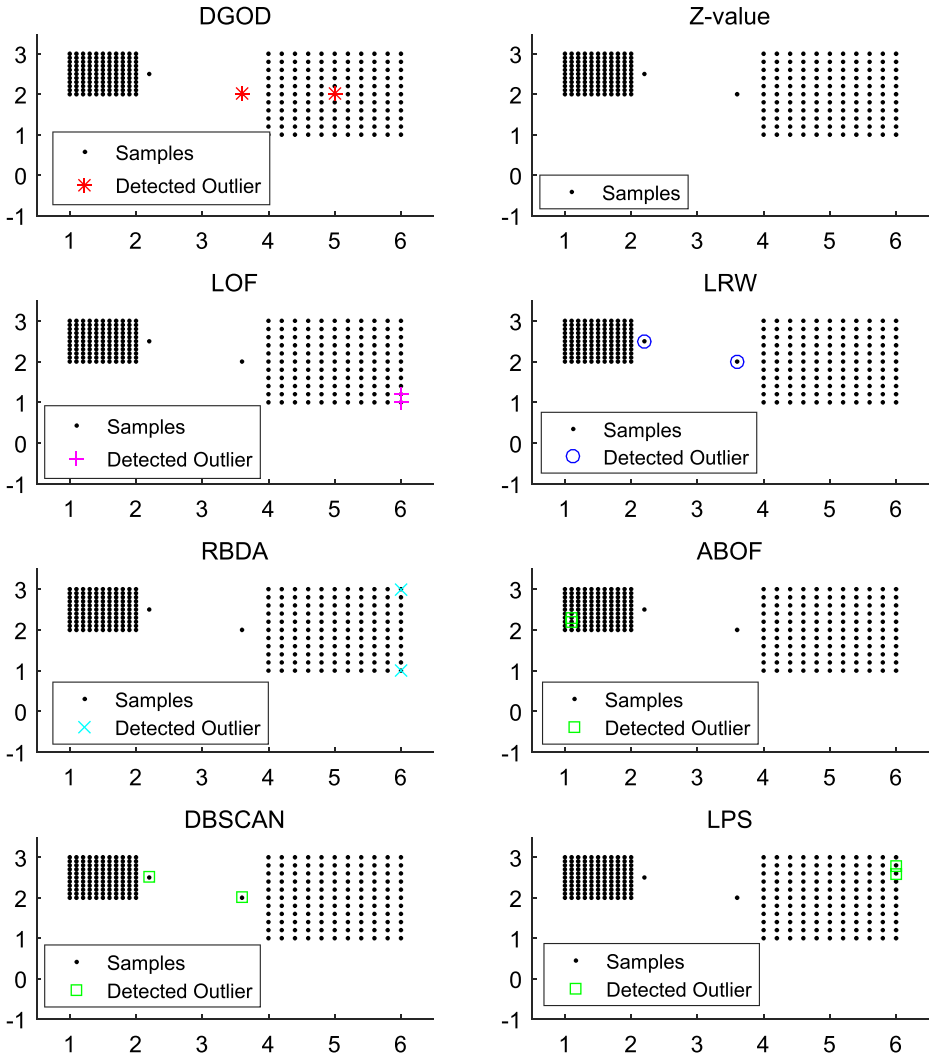


Fig. 6 Results comparison on local density synthetic dataset

4.1 Synthetic data

In order to illustrate how DGOD behaves in outlier detection, four synthetic datasets, shown in Figs. 6, 7, 8, and 9, are designed to consider the various situations of datasets structure, including outliers planted in datasets with different local density and multi-granularity, outliers in linear regression and clustering. These four synthetic datasets are named as Local Density Synthetic Dataset, Multi-Granularity Synthetic Dataset, Linear Regression Synthetic Dataset and Two Cluster Synthetic Dataset, respectively. Figsures 6, 7, 8, and 9 displays the detected outliers by different methods on the four datasets. For each method, the detected outliers are marked with different colors and symbols.

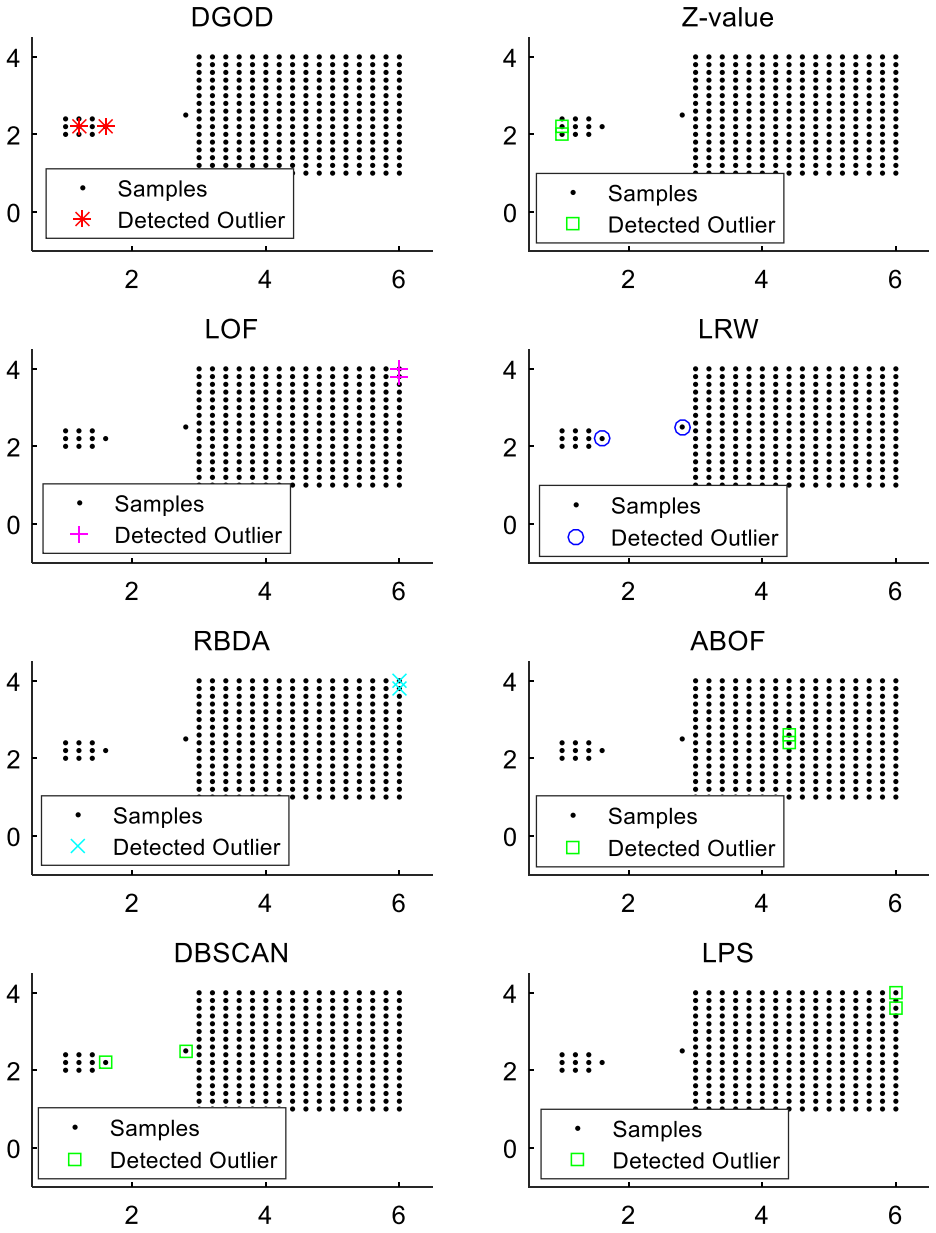


Fig. 7 Results comparison on multi-granularity synthetic dataset

Density and multi-granularity of dataset are challenges for outlier detection task, since the normal samples or clusters are generated by placing all points uniformly with varying degrees of densities. From Figs. 6 and 7, we can see that LRW and DBSCAN detected the two ground-truth outliers and DGOD detected only one ground-truth outlier, while other methods failed. It indicates that LRW and DBSCAN may be suitable for Local Density Synthetic Dataset and Multi-granularity Synthetic Dataset.

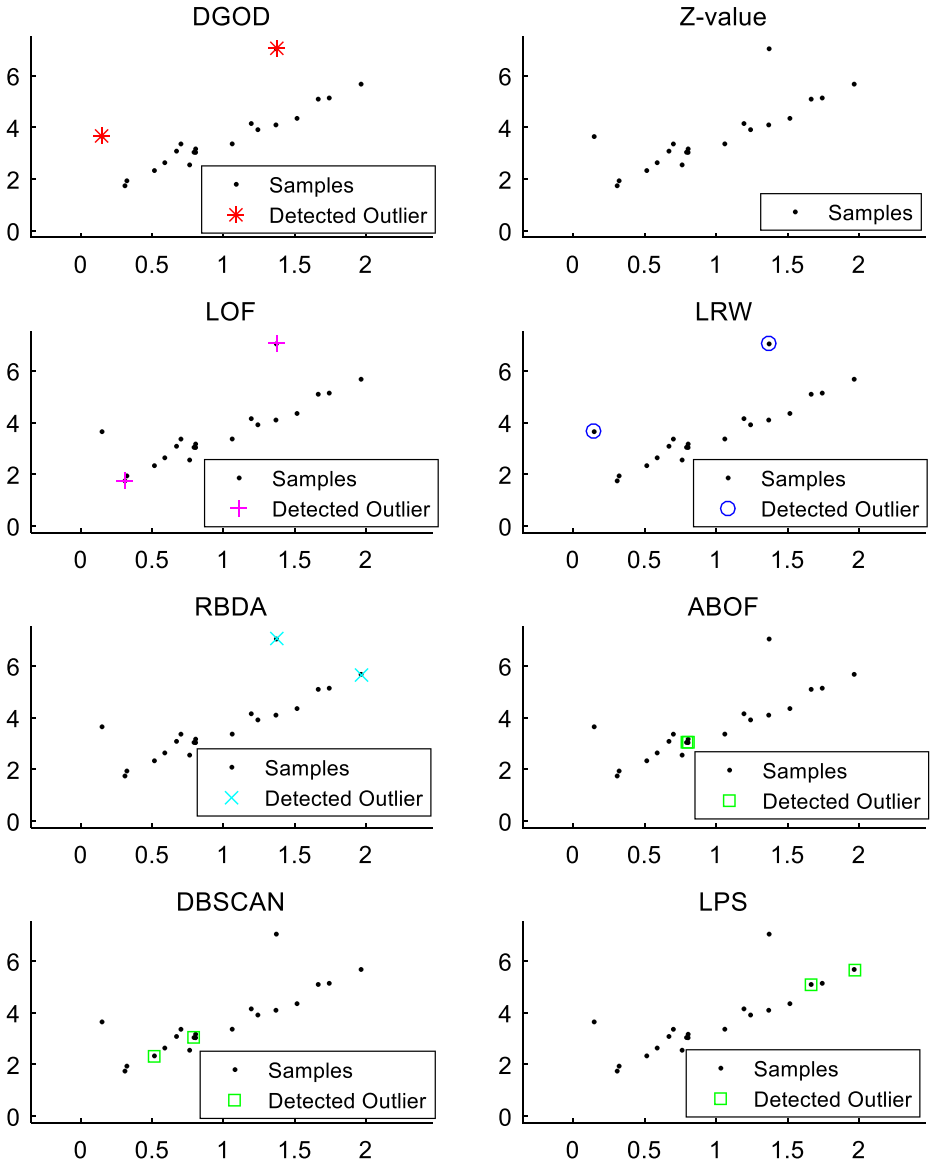


Fig. 8 Results comparison on linear regression synthetic dataset

Outliers in regression and clustering tasks are main issue for robust data mining. In Fig. 8, both DGOD and LRW detect two true outliers successfully, and LOF and RBDA detect only one true outlier, while other methods failed. In Fig. 9, DGOD detected 5 true outliers successfully, and Z-value detected 4 true outliers, LOF and LPS detected 3 true outliers, RBDA detected 2 outliers, while other methods failed. It indicates that DGOD method performs well for robust regression and clustering tasks.

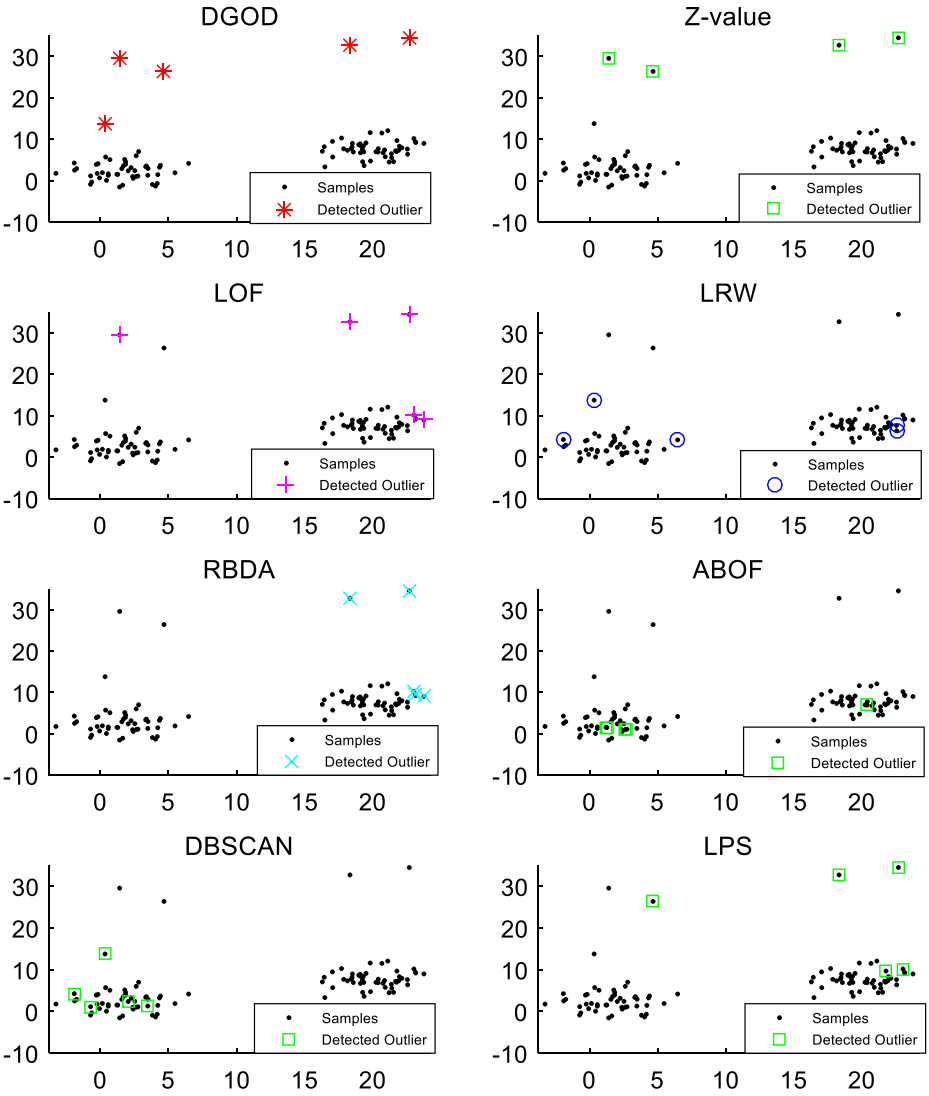


Fig. 9 Results comparison on two cluster synthetic dataset

Table 2 UCI datasets description

Dataset	Dimensionality of samples	Total number of inliers	Total number of outliers
sonar	60	111	10/20/30
wdbc	30	357	22/43/64
soybean2	35	120	2/4/5
vowel	10	480	5/10/15
ionosphere	34	225	13/26/38

Table 3 Confusion matrix

	#Positive	#Negative	#Total
#True	r_0	r - r_0	r
#False	r - r_0	n - 2r + r_0	n - r
#Total	r	n - r	n

4.2 Real datasets with rare classes

We also applied the DGOD method to six real-world UCI datasets. For each UCI dataset, following the data preprocessing strategy used in [33], the class with minimum number of samples is made `rare' by removing most of its samples, and the remaining samples are used in the final dataset. Specifically, only 10, 20 and 30% of the samples in the class with minimal size is contained as outlier. The datasets used in experiments are described in Table 2.

The number of detected outliers is presented as the number of true outliers, except for Z-value method. To make a comprehensive comparison, three metrics called Precision, Recall and F_1 measure are adopted in the experiments to evaluate the performances of different methods. We defined Precision (P) as follows

$$\mathrm{P} = r_0/r \tag{18}$$

where r_0 is the number of true outliers among r detected outlier by an algorithm. It measures the percentage of true outliers among top r ranked. Note that precision defined above in fact measures the proportion of detected outliers that are correctly identified.

Based on the confusion matrix in Table 3, the Recall (R) and F_1 measure can be computed as follows:

$$R = \frac{TP}{TP + FN} = \frac{r_0}{n{-}2r + 2r_0} \tag{19}$$

$$F_1 = \frac{2PR}{P + R} = \frac{2 \times \frac{r_0}{r} \times \frac{r_0}{n{-}2r + 2r_0}}{\frac{r_0}{r} + \frac{r_0}{n{-}2r + 2r_0}} \tag{20}$$

Table 4 Outlier detection results on sonar dataset

Method	10 Outliers			20 Outliers			30 Outliers		
	P	R	F_1	P	R	F_1	P	R	F_1
DGOD	**25.0**	**2.3**	**4.3**	**30.5**	**5.9**	**9.9**	**29.7**	**9.0**	**13.8**
Z-value	1.0	0.1	0.2	5.5	1.2	1.9	11.4	3.5	5.3
LOF	2.0	0.2	0.3	9.5	2.0	3.3	16.0	5.3	7.9
LRW	11.0	1.1	1.9	18.0	3.6	6.0	20.7	6.5	9.9
RBDA	10.0	0.9	1.7	14.0	2.9	4.8	21.0	6.7	10.1
ABOF	0.0	0.0	0.0	2.0	0.4	0.7	8.3	2.9	4.3
DBSCAN	13.0	1.2	2.2	19.5	3.6	6.1	22.7	7.0	10.7
LPS	8.0	0.8	1.4	13.5	2.8	4.6	19.3	6.2	9.4

Table 5 Outlier detection results on WDBC dataset

Method	22 Outliers			43 Outliers			64 Outliers		
	P	*R*	F_1	*P*	*R*	F_1	*P*	*R*	F_1
DGOD	75.0	4.5	8.5	77.2	8.7	15.7	82.7	13.3	22.9
Z-value	40.9	2.5	4.8	48.8	5.9	10.5	50.3	7.9	13.6
LOF	1.4	0.1	0.2	8.4	1.1	2.0	6.3	1.3	2.2
LRW	15.5	1.0	1.9	20.7	2.7	4.7	23.6	4.7	7.8
RBDA	0.5	0.0	0.1	3.0	0.4	0.7	3.1	0.7	1.1
ABOF	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
DBSCAN	**85.0**	**5.0**	**9.5**	**86.0**	**9.5**	**17.1**	**96.7**	**14.8**	**25.7**
LPS	74.5	4.5	8.4	80.0	9.0	16.2	83.1	13.3	23.0

Since the number of detected outliers (#Positive) is fixed as *r*, the False Alarm Rate (False Positive Rate) in our experiment is 50%.

For each dataset, we select different proportion of outliers for experimental comparable analysis. For each fixed number of outliers, the outliers used in experiment are selected randomly and random selection is repeated 10 times. Finally, the average detection precision, recall and F_1 measure are reported in Tables 4, 5, 6, 7 and 8, in which the optimal result of each measure is presented in bold.

From the Tables 4, 5, 6, 7, and 8, we can clearly observe that, on Sonar and Ionosphere datasets, DGOD performs better than other methods under the three evaluation measures. However, on WDBC and Vowel datasets, DBSCAN performs better than other methods under the three evaluation measures, and the DGOD method is the second best result. For the most cases, DGOD method consistently yields a better performance. Most existing methods first find the neighbors for each sample based on *k* nearest neighbor, and then compute local density of a sample using the neighbors, where the significant differences between local densities give us more confidence to declare an outlier. However, these methods may not be able to obtain reliable outliers in real world application, due to existing highly heterogeneous neighborhoods of some samples. To handle heterogeneous problems, DGOD uses local density and discriminant distance of each sample to detect more reliable outliers. Since the DGOD method is based on pairwise distances and local densities, when the samples are densely distributed, the performance will be better.

Table 6 Outlier detection results on soybean2 dataset

Method	2 Outliers			4 Outliers			5 Outliers		
	P	*R*	F_1	*P*	*R*	F_1	*P*	*R*	F_1
DGOD	**100.0**	**1.6**	**3.2**	72.5	2.4	4.6	74.0	3.0	5.8
Z-value	50.0	0.8	1.6	10.0	0.3	0.7	20.0	0.9	1.6
LOF	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
LRW	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
RBDA	5.0	0.1	0.2	0.0	0.0	0.0	0.0	0.0	0.0
ABOF	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
DBSCAN	**100.0**	**1.6**	**3.2**	20.0	0.7	1.3	18.0	0.7	1.4
LPS	40.0	0.7	1.3	**100.0**	**3.2**	**6.3**	**100.0**	**4.0**	**7.7**

Table 7 Outlier detection results on vowel dataset

Method	5 Outliers			10 Outliers			15 Outliers		
	P	R	F_1	P	R	F_1	P	R	F_1
DGOD	72.0	0.7	1.5	61.0	1.3	2.5	63.3	2.0	3.8
Z-value	13.5	0.1	0.2	23.0	0.2	0.5	21.0	0.2	0.4
LOF	0.0	0.0	0.0	0.0	0.0	0.0	1.3	0.0	0.1
LRW	24.0	0.3	0.5	17.0	0.4	0.7	13.3	0.4	0.8
RBDA	2.0	0.0	0.0	3.0	0.1	0.1	0.7	0.0	0.0
ABOF	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
DBSCAN	**78.0**	**0.8**	**1.6**	**86.0**	**1.8**	**3.4**	**78.0**	**2.4**	**4.6**
LPS	0.0	0.0	0.0	0.0	0.0	0.0	2.0	0.1	0.1

4.3 Real datasets with planted outliers

To validate the effectiveness of DGOD method on high-dimensional data, we conduct an experiment on image data. Firstly, we select Yale face dataset[1] as normal image set, which contains 165 grayscale images of 15 individuals. Then, we add six cat face images as outliers. Each image is resized to 64×64 and can be viewed as a point in 4096 dimensional space.

The experiment aims to recognize cat faces in Yale human face dataset. Figure 10 shows the first six outlier images detected by different methods. Among them, DGOD recognizes five cat faces successfully, and Z-value recognizes only two cat faces, while other methods failed. Moreover, from Fig. 10, we can find that illumination, with/without glasses and expressions are main interfere factors for the detection task. The possible reasons could be as follows: (1) The original Yale face dataset has a clear cluster structure, and the DGOD method considered local density and discriminant distance simultaneously, which make the outliers easier to be detected. (2) The intra-class variations in Yale face dataset distorted the distance neighborhood relationships between images, which further influence the detection rates of baseline methods.

For the purpose of visualization, we employed multi-dimensional scaling [6] to embedding the original face images into two-dimensional space. The comparative results are shown in Fig. 11. The embedding six outliers are marked with red circles. For each method, the marked points with different colors in the plots are the samples with highest outlier score. DGOD and LPS successfully detected the true outliers. LOF and RBDA seem to behave in the similar way. As can be seen in Fig. 11, the detected outliers located in the region of top-left corner. The images located in this region are the faces with darkest illumination.

Table 9 shows the time costs of all methods in face image datasets with planted cat face images. As one can see, DGOD is more efficient than LRW, ABOF, DBSCAN and LPS, and computational comparable with Z-value, LOF and RBDA. Although DGOD and DBSCAN perform comparably under detection rate on some case, this observation experimentally demonstrates that the proposed DGOD method is more efficient than DBSCAN in terms of computational time.

4.4 Parameter sensitivity analysis

In this section, we analyze the performance of the proposed method by varying the cutoff distance d_c and the percentage of outliers. Similar to experiments in section 4.3,

[1] http://vision.ucsd.edu/content/yale-face-database

Table 8 Outlier detection results on ionosphere dataset

Method	13 Outliers			26 Outliers			38 Outliers		
	P	R	F_1	P	R	F_1	P	R	F_1
DGOD	**77.7**	**4.3**	**8.2**	**81.2**	**8.7**	**15.8**	83.9	**12.7**	**22.1**
Z-value	70.8	4.0	7.5	84.6	7.0	13.0	94.5	8.8	16.0
LOF	20.0	1.2	2.3	33.8	4.0	7.2	37.6	6.6	11.3
LRW	13.1	0.8	1.5	23.5	2.9	5.1	29.7	5.4	9.1
RBDA	34.6	2.0	3.8	42.7	5.0	8.9	42.9	7.4	12.6
ABOF	0.0	0.0	0.0	0.8	0.1	0.2	0.3	0.1	0.1
DBSCAN	67.7	3.7	7.1	52.3	5.5	9.9	**86.6**	12.6	22.0
LPS	15.4	0.9	1.7	19.2	2.4	4.2	23.9	4.4	7.5

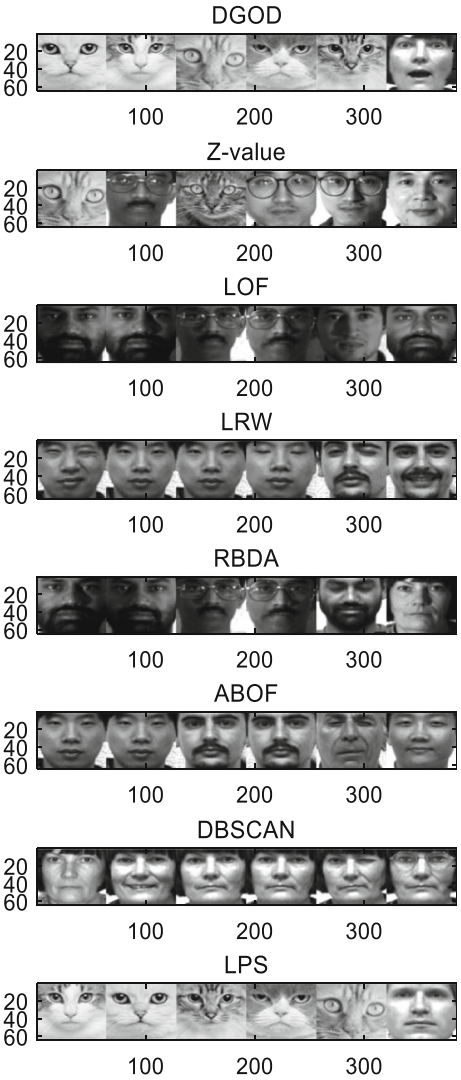
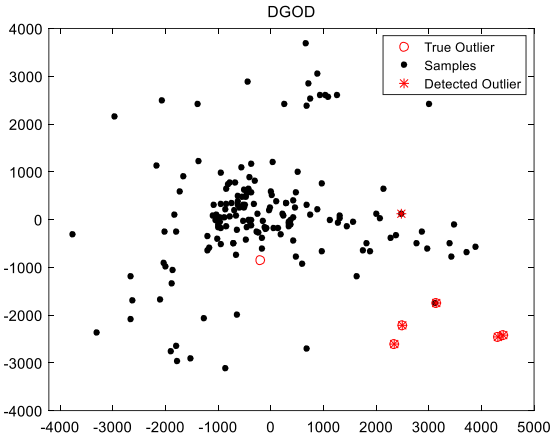
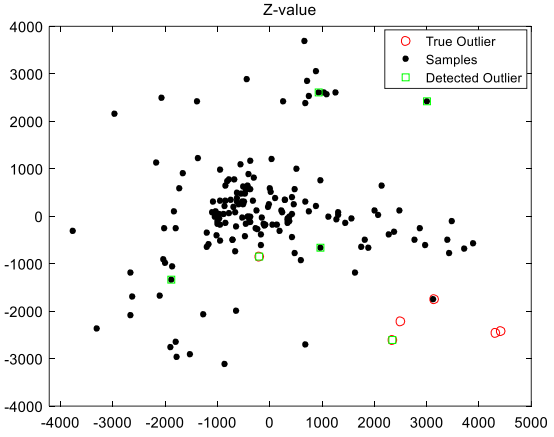


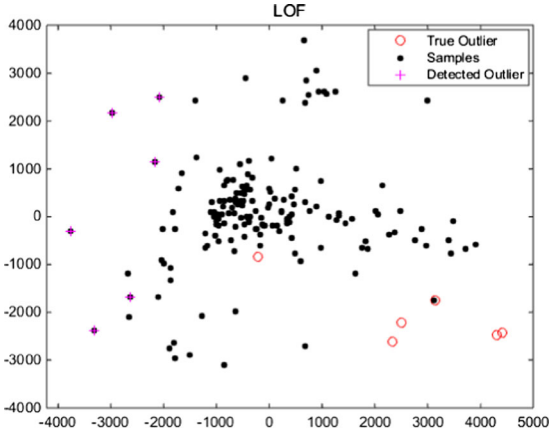
Fig. 10 Outliers detected by different methods on yale face dataset with planted cat faces



(a) DGOD Detection Result

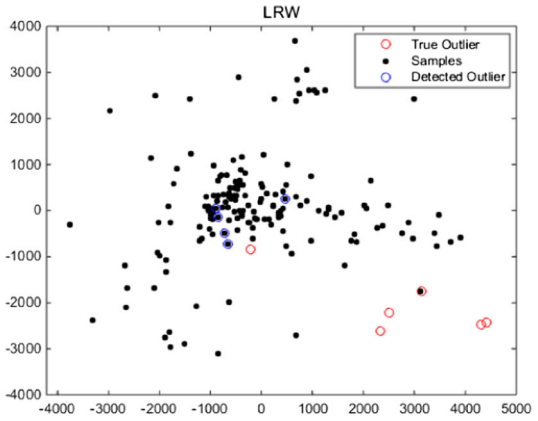


(b) Z-value Detection Results

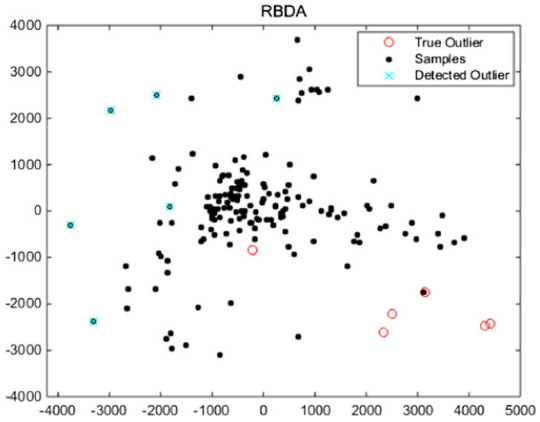


(c) LOF Detection Results

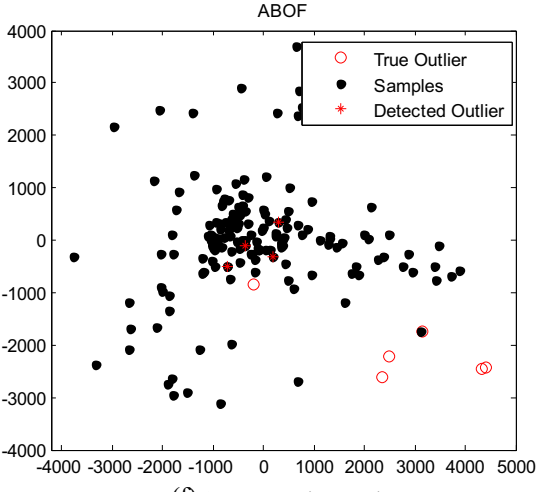
Fig. 11 2-dimensional embedding on yale face dataset with planted cat faces



(d) LRW Detection Results

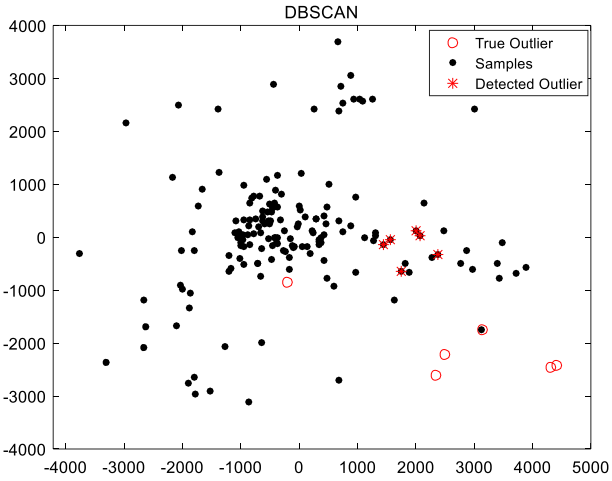


(e) RBDA Detection Results

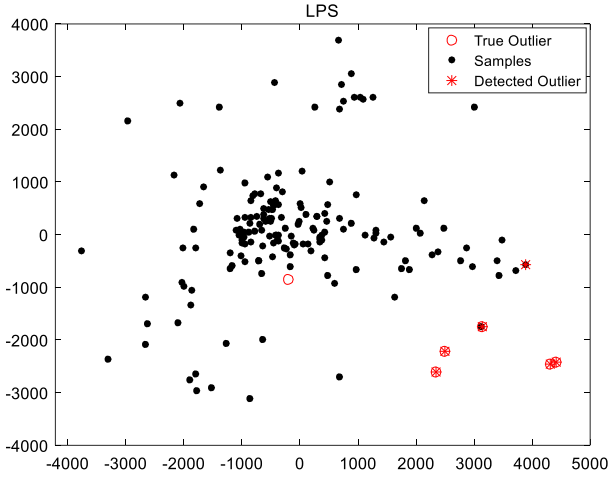


(f) ABOF Detection Results

Fig. 11 (continued)



(g) DBSCAN Detection Results



(h) LPS Detection Results

Fig. 11 (continued)

10 cat faces are added to Yale face data set successively. To analyze how DGOD method affected by distance metrics between samples, five distance metrics are used in experiment, i.e., Euclidean distance, Minkovski distance, Chebychev distance, cosine distance and spearman distance. Figure 12 shows the effect of types of distance

Table 9 Time costs comparison (millisecond)

Method	Time costs	Method	Time costs
DGOD	86.1	RBDA	81.2
Z-value	39.0	ABOF	221.6
LOF	77.2	DBSCAN	168.5
LRW	442.5	LPS	179.5

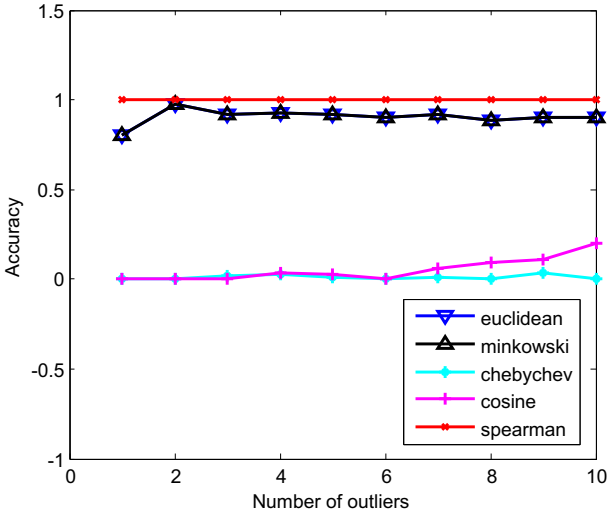


Fig. 12 Effect of types of distance metrics between samples

metrics between samples. As can be seen, Euclidean, Minkovski and spearman distance metric in DGOD have better performance than others. Then we set different values of cutoff distance d_c in a wide range, the performance of DGOD with different cutoff distance are shown in Fig. 13. According to rule of thumb, we set d_c to 2406. It is observed that as the cutoff distance increases, the detection rate of DGOD method increases correspondingly, and when the cutoff distance is greater than 2000, the detection rate tends to stable. There is a wide range for cutoff distance setting. Additionally, from Fig. 14, we can see that, DGOD method is robust to number of outliers.

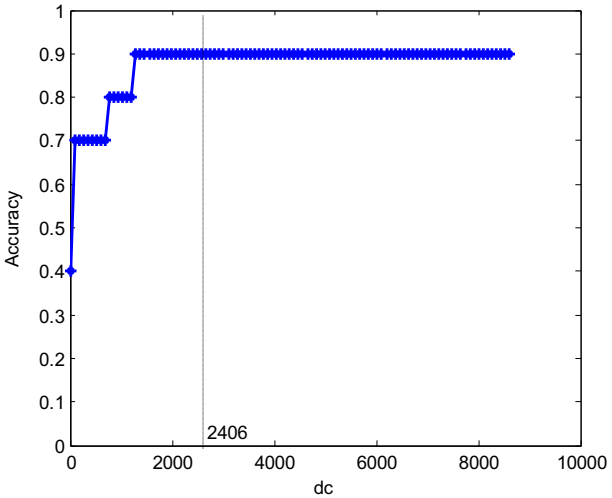


Fig. 13 Effect of cutoff distance d_c

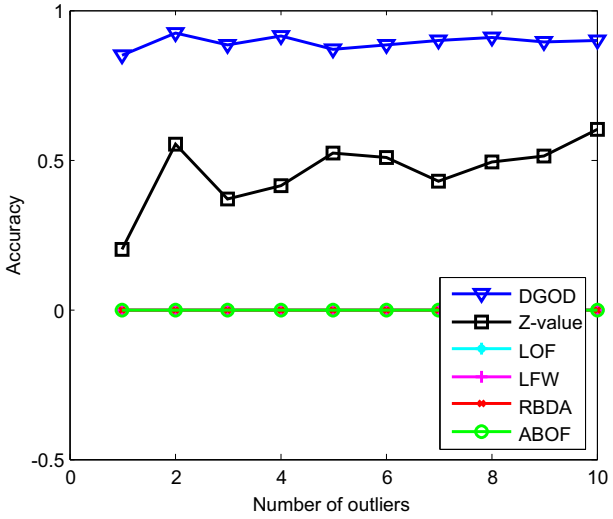


Fig. 14 Effect of number of outliers

5 Conclusion

SBD are about daily large data, produced from social communication and news dissemination, and it has been an effective platform for security and privacy issues on both theoretical and applied techniques, especially information control and detection is the critical issue. In order to detected outlier in various complex datasets, we present an outlier detection method by incorporating the idea of density-based and clustering-based methods. The proposed DGOD method starts by computing distance matrix of samples, then the local density and discriminant distance of each sample is computed. Finally, outliers are identified relatively with low local density and high discriminant distance. Empirical results on synthetic and real world dataset have demonstrated the effectiveness of our method in terms of data shape and dimensionality.

6 Limitations and future works

Since DGOD exploits pairwise distances between all samples to get density information, its performance will be affected by the distance computation to some extent. In our future work, we will extend DGOD to kernel version by using kernel similarity function as distance measure between samples. In addition, we would apply the decision graph based outlier detection method to detect abnormal apple samples with diseases by hyperspectral imaging in agricultural product inspection. Moreover, how to embed the proposed information detection method into distributed database system to deal with big social data incrementally is still a challenging problem [2, 24].

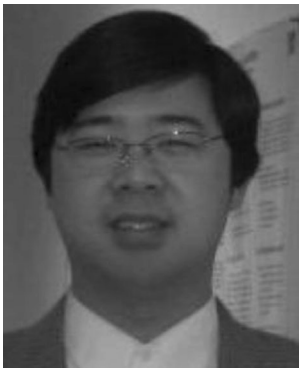
Acknowledgements The authors would like to thank all the anonymous reviewers and editors for their valuable comments and. Additionally, we would like to thank Bin Liu and Yaguang Jia who critically reviewed the study proposal. This work was supported in part by Yangling Demonstration Zone Science and Technology Planning Project under Grant 2016NY-31, Doctoral Starting up Foundation of Northwest A&F University under Grant 2452015302 and National Natural Science Foundation of China under Grants 61602388.

References

1. Aggarwal CC (2013) Outlier analysis[M]. Springer Science & Business Media, New York
2. Bagui S, Nguyen LT (2015) Database Sharding: to provide fault tolerance and scalability of big data on the cloud. Int J Cloud Appl Comput 5(2):36–52
3. Barnett V, Lewis T (1994) Outliers in statistical data[M]. Wiley, New York
4. Bello-Orgaz G, Jung JJ, Camacho D (2016) Social big data: recent achievements and new challenges[J]. Inf Fusion 28(3):45–59
5. Breunig MM, Kriegel HP, Ng RT et al (2000) LOF: identifying density-based local outliers[C]. Proceedings of ACM sigmod record. ACM 29(2):93–104
6. Cox T, Cox M (1994) Multidimensional scaling. Chapman & Hall, London
7. Du H, Zhao S, Zhang D et al (2016) Novel clustering-based approach for local outlier detection[C]. Int Conf Comput Commun 2016:802–811
8. Dufrenois F (2015) A one-class kernel fisher criterion for outlier detection[J]. IEEE Transactions on Neural Networks and Learning Systems 26(5):982–994
9. Ester M, Kriegel HP, Sander J et al (1996) A density-based algorithm for discovering clusters in large spatial databases with noise[C]. Kdd 96(34):226–231
10. Gupta S, Gupta BB (2017) XSS-secure as a service for the platforms of online social network-based multimedia web applications in cloud [J]. Multimed Tools Appl. https://doi.org/10.1007/s11042-016-3735-1
11. Ha J, Seok S, Lee JS (2014) Robust outlier detection using the instability factor[J]. Knowl-Based Syst 63:15–23
12. Hamedani K, Liu L, Rachad A et al (2017) Reservoir computing meets smart grids: attack detection using delayed feedback networks[J]. IEEE Trans Ind Inf. https://doi.org/10.1109/TII.2017.2769106
13. Hawkins D (1980) Identification of outliers. Chapman and Hall, New York
14. Hido S, Tsuboi Y, Kashima H et al (2011) Statistical outlier detection using direct density ratio estimation[J]. Knowl Inf Syst 26(2):309–336
15. Huang H, Mehrotra K, Mohan CK (2013) Rank-based outlier detection[J]. J Stat Comput Simul 83(3):518–531
16. Jiang S, An Q (2008) Clustering-based outlier detection method[C]. In Proceedings of IEEE fifth international conference on fuzzy systems and knowledge discovery 2:429–433
17. Jin W, Tung AKH, Han J et al (2006) Ranking outliers using symmetric neighborhood relationship[M]. Advances in knowledge discovery and data mining. Springer, Berlin, pp 577–593
18. Johnson RA, Wichern DW (1992) Applied multivariate statistical analysis[M]. Prentice hall, Englewood Cliffs
19. Knox EM, Ng RT (1998) Algorithms for mining distancebased outliers in large datasets[C]. In Proceedings of the international conference on very large data bases pp 392–403
20. Kriegel H P, Zimek A (2008) Angle-based outlier detection in high-dimensional data[C]. In Proceedings of the 14th ACM SIGKDD international conference on knowledge discovery and data mining. ACM 444–452
21. Liu H, Li X, Li J et al (2017) Efficient outlier detection for high-dimensional data[J]. IEEE Trans Syst Man Cybern Syst. https://doi.org/10.1109/TSMC.2017.2718220
22. Maimon O, Rockach L (2005) Data mining and knowledge discovery handbook: a complete guide for practitioners and researchers. Springer, New York
23. Onderwater M (2010) Detecting unusual user profiles with outlier detection techniques. Master Thesis, http://tinyurl.com/vu-thesis-onderwater
24. Ouf S, Nasr M (2015) Cloud computing: the future of big data management. Int J Cloud Appl Comput 5(2):53–61
25. Papadimitriou S, Kitagawa H, Gibbons PB et al (2003) Loci: fast outlier detection using the local correlation integral[C]. In Proceedings of IEEE 19th international conference on data engineering pp 315–326
26. Ramaswamy S, Rastogi R, Shim K (2000) Efficient algorithms for mining outliers from large data sets. ACM SIGMOD Rec 29(2):427–438
27. Rehm F, Klawonn F, Kruse R (2007) A novel approach to noise clustering for outlier detection[J]. Soft Comput 11(5):489–494
28. Rodriguez A, Laio A (2014) Clustering by fast search and find of density peaks[J]. Science 344(6191): 1492–1496
29. Roweis ST, Saul LK (2000) Nonlinear dimensionality reduction by locally linear embedding[J]. Science 290(5500):2323–2326
30. Schölkopf B, Platt JC, Shawe-Taylor J et al (2001) Estimating the support of a high-dimensional distribution[J]. Neural Comput 13(7):1443–1471

31. Shi Y, Zhang L (2011) COID: a cluster–outlier iterative detection approach to multi-dimensional data analysis[J]. Knowl Inf Syst 28(3):709–733
32. Tang J, Chen Z, Fu AWC et al (2002) Enhancing effectiveness of outlier detections for low density patterns[M]. Advances in knowledge discovery and data mining. Springer, Berlin, pp 535–548
33. Tax DMJ, Duin RPW (2004) Support vector data description[J]. Mach Learn 54(1):45–66
34. Wang X, Wang XL, Ma Y et al (2015) A fast MST-inspired kNN-based outlier detection method[J]. Inf Syst 48:89–112
35. Wang S, Wang D, Li C et al (2015) Comment on" Clustering by fast search and find of density peaks"[J]. arXiv preprint arXiv:1501.04267
36. Wu J, Guo S, Li J et al (2016) Big data meet green challenges: big data toward green applications[J]. IEEE Syst J 10(3):888–900
37. Wu J, Guo S, Li J et al (2016) Big Data Meet Green Challenges: Greening Big Data[J]. IEEE Syst J 10(3): 873–887
38. Zhang Z, Gupta BB (2017) Social media security and trustworthiness: overview and new direction[J]. Futur Gener Comput Syst. https://doi.org/10.1016/j.future.2016.10.007
39. Zhang K, Hutter M, Jin H (2009) A new local distance-based outlier detection approach for scattered real-world data[M]. Advances in Knowledge Discovery and Data Mining. Springer, Berlin, pp 813–822
40. Zhang J, Gao Q, Wang H et al (2011) Detecting anomalies from high-dimensional wireless network data streams: a case study[J]. Soft Comput 15(6):1195–1215
41. Zhang Z, Sun R, Zhao C et al (2017) CyVOD: a novel trinity multimedia social network scheme[J]. Multimed Tools Appl 76(18):18513–18529

Jinrong He received the BS degree in science of information and computation, and the MS degree in computational mathematics from Wuhan University of Technology, P.R. China, in 2007 and 2010, respectively. In 2014, he received the Ph.D. degree in computer software and theory from State Key Laboratory of Software Engineering, Wuhan University, P.R. China. Currently, he is a lecturer in Northwest A & F University. His research interests include computer vision, dimensionality reduction and feature extraction.



Naixue Xiong received his both PhD degrees in Wuhan University (about software engineering), and Japan Advanced Institute of Science and Technology, respectively (about dependable networks). Now, he is a professor of School of Computer Science, Colorado Technical University. His research interests include Security and Dependability, Cloud Computing, Network Architecture, and Optimization Theory. He is serving as an associate editor or editor member for over 10 international journals (including Information Science), and a guest editor for over 10 international journals, including Sensor Journal, WINET and MONET.