

A New Outlier Detection Algorithm Based on Fast Density Peak Clustering Outlier Factor

ZhongPing Zhang, College of Information Science and Engineering, Yanshan University, China

Sen Li, College of Information Science and Engineering, Yanshan University, China*

 <https://orcid.org/0000-0002-2571-7580>

WeiXiong Liu, College of Information Science and Engineering, Yanshan University, China

Ying Wang, College of Information Science and Engineering, Yanshan University, China

Daisy Xin Li, Herbalife Nutrition, USA

ABSTRACT

Outlier detection is an important field in data mining, which can be used in fraud detection, fault detection, and other fields. This article focuses on the problem where the density peak clustering algorithm needs a manual parameter setting and time complexity is high; the first is to use the k nearest neighbors clustering algorithm to replace the density peak of the density estimate, which adopts the KD-Tree index data structure calculation of data objects k close neighbors. Then it adopts the method of the product of density and distance automatic selection of clustering centers. In addition, the central relative distance and fast density peak clustering outliers were defined to characterize the degree of outliers of data objects. Then, based on fast density peak clustering outliers, an outlier detection algorithm was devised. Experiments on artificial and real data sets are performed to validate the algorithm, and the validity and time efficiency of the proposed algorithm are validated when compared to several conventional and innovative algorithms.

KEYWORDS

Centripetal Relative Distance, Data Mining, K Nearest Neighbor, Outlier, Peak Density Clustering

INTRODUCTION

Outlier refers to the data points in the data set that deviate from the normal data distribution; their observations are significantly different from other observations. Outlier detection is the process of finding outliers in a data set and is an important area of data mining research. At present, outlier detection is used in a variety of fields, including industrial wireless sensor networks (Ramotsoela et al., 2018; Safaei et al., 2020), fraud detection (Avdiienko et al., 2017; Ngai et al., 2011), intrusion detection (Denning, 1987), fault detection (Bhatti et al., 2020) and electrocardiogram anomaly detection (Andrysiak, 2020). Research on outlier detection has also proposed many methods, mainly based on statistics (Kafadar et al., 1995; Seheult et al., 1989), distance-based (Knorr et al., 2000; Knorr & Ng, 1997), cluster-based (Ester et al., 1996; Jain et al., 1999; Karypis et al., 1999), density-based

DOI: 10.4018/IJDWM.316534

*Corresponding Author

This article published as an Open Access article distributed under the terms of the Creative Commons Attribution License (<http://creativecommons.org/licenses/by/4.0/>) which permits unrestricted use, distribution, and production in any medium, provided the author of the original work and original publication source are properly credited.

(Breunig et al., 2000; Schubert et al., 2014; Wahid & Annavarapu, 2021; Wang et al., 2019; Yang & Liu, 2020; Zhang et al., 2009), and other methods.

Since Breunig et al. (2000) published the local outlier factor method, the density-based outlier identification approach has been one of the most prominent directions in the area of outlier detection. The primary idea behind this categorization approach is to determine the density of each data object using a density estimate algorithm, and then judge outliers based on the region where the data objects are situated. In terms of data objects, there are considerable variances. With the development of clustering approaches in recent years, clustering-based outlier detection algorithms have emerged as an essential component of the outlier detection area. The core idea is to use clustering methods to cluster data sets; outliers are usually in those small-scale clusters that contain very few data objects. The core ideas of the two methods evidences that the two types of methods have certain similarities, and small-scale clusters are often sparse areas. These two types of methods have their advantages and disadvantages. In recent years, many scholars have begun to focus on how to combine the advantages of the two methods to propose a more robust outlier detection algorithm.

Rodriguez and Laio (2014) published an important new clustering algorithm, namely, the density peak clustering (DPC) algorithm. The DPC algorithm is a simple and efficient clustering algorithm. The data set of any dimension is mapped into a two-dimensional decision diagram by estimating the relative distance and density in the nearby area. The decision diagram can clearly reflect the hierarchical relationship of the data set, and then choose the data set with the highest density and the greatest separation from other data sets. The data objects with high density data objects are farther away; these data objects are called density peak points, as the cluster center of the data set. Finally, the remaining data objects are assigned to the cluster center closest to it. Compared with other traditional clustering methods, the DPC algorithm is less affected by outliers and noise. In addition, the DPC algorithm does not require multiple iterations in the clustering process. Therefore, regardless of the detection accuracy or time efficiency, the DPC algorithm is more suitable for combination with the outlier detection algorithm.

Although the DPC algorithm can complete clustering simply and efficiently, there are still some problems. Since the DPC algorithm needs to calculate the density of each data object, a distance matrix will be generated in the process of calculation, which will enhance the temporal complexity while increasing the space complexity. The time complexity of the DPC algorithm is $O(n^2)$. For some large-scale data sets, the time efficiency drops severely.

As to the problems of high time complexity in the DPC algorithm, as well as the setting of the cutoff distance d_c (variable of distance) and the selection of clustering centers, the authors enhanced the DPC method, in this study, and combined the improved DPC algorithm with outlier identification. Then, they proposed a new outlier detection algorithm based on the fast density peak clustering outlier factor (FDPC-OF). The FDPC-OF algorithm replaces the density estimation component of the DPC algorithm with the k-closest neighbor approach, which eliminates establishing the cutoff distance d_c , and reduces the time complexity of the algorithm by using the KD-Tree index data structure to assist in computing each data object's k closest neighbors. KD-Tree is a kind of tree data structure that stores the instance points in the k-dimensional space for quick search. KD-Tree can be used to construct the original data set into the binary tree structure data set to speed up the k-nearest neighbor (KNN) search. To avoid searching the entire data set, the authors considered only the k-closest neighbors of the data item for computing the relative distance. Second, after performing the local density and relative distance calculations, they utilized the product of local density and relative distance to automatically pick the cluster center, avoiding any issues that may arise from artificially selecting the cluster center. To increase the applicability of the algorithm on data sets with varying densities, the authors improved the relative distance in DPC, defined the average centripetal distance, and combined the relative distance and average centripetal distance of the data objects to give a centripetal relative distance. Finally, combined with the centripetal relative distance and local density, they devised an outlier factor based on rapid density peak clustering to describe the outlier degree of data objects.

The rest of this paper is structured as follows: The second section introduces the DPC algorithm and the definition of KNN; the third section introduces the fast DPC algorithm and the proposed FDPC-OF algorithm; the fourth section provides a comparison experiment the authors conducted to verify that the proposed algorithm has better performance; finally, the fifth section concludes the paper.

RELATED WORK

The classical clustering algorithm K-means (MacQueen, 1967) specifies the preliminary clustering center, and then finds the final clustering center through iterative updating. Since every point is assigned to the nearest clustering center, it cannot detect the data distribution of the aspheric surface category. Ester et al. (1996) proposed the density-based spatial clustering of applications with noise (known as DBSCAN) algorithm, which can cluster data with arbitrary shape distribution, but a density threshold must be specified. Noise points below this density threshold are removed. Comanicu and Meer (2002) proposed the mean shift algorithm, which adopts mean shift to make data points extend along the direction of higher density until they converge to the cluster center. However, mean migration is an iterative process with high time complexity. The DPC algorithm solves the above deficiencies. The DPC algorithm is simple, requires fewer parameters, does not need iteration, and can recognize clusters of different shapes and sizes.

Density Peak Clustering Algorithm

In recent years, clustering algorithms have been widely developed. Since the proposal of the DPC (Rodriguez & Laio, 2014), many experts have improved it. Xu et al. (2018) proposed a grid-based density peak clustering algorithm (known as DPCG). When determining the density of data items on a local scale, the grid idea is used to calculate the distance instead of the DPC algorithm; this method improves the calculation speed of the algorithm. Second, at the start of the DPC algorithm process, the cutoff distance d_c must be set, and the cluster center must be manually selected. Therefore, d_c has a large impact on the algorithm, and the manual setting will easily lead to a decrease in the accuracy of the algorithm. Huang et al. (2016) proposed a fast clustering algorithm ADPclust using adaptive density peak detection, which replaces the density estimation in the DPC algorithm with nonparametric Gaussian kernel density estimation, avoiding the d_c setting of the cutoff distance. Besides, Huang et al. proposed a cluster center selection method based on the contour optimization algorithm to automatically select the cluster center. However, maintaining normal accuracy with this strategy is difficult in the face of some sparse and small-scale cluster data sets.

Below, the authors will briefly introduce the DPC algorithm (Rodriguez & Laio, 2014). The DPC algorithm must calculate two indicators: The local density of the data object and the relative distance of the data object.

Density Peaks Cluster Local Density ρ_i

The DPC local density ρ_i refers to the number of other data objects contained in the data object x_i within the cutoff distance.

Equation 1 shows the calculation method of the DPC local density ρ_i (Rodriguez & Laio, 2014) of the data object x_i :

$$\rho_i = \sum_{j=1}^n \chi(d_{ij} - d_c) \quad (1)$$

In Equation 1, n is the number of data objects in the data collection, d_{ij} represents the distance between data objects x_i and x_j , which is usually calculated by Euclidean distance, d_c represents the cutoff distance, which needs to be initialized at the beginning of the algorithm, and $\chi(\cdot)$ is an indicator

function. When $x < 0$, $\chi(x) = 1$. If $x \geq 0$, then $\chi(x) = 0$. In simple terms, the cutoff distance d_c is similar to the radius, and DPC local density ρ_i calculates the number of other data objects within the radius through the indicator function.

Density Peak Cluster Relative Distance δ_i

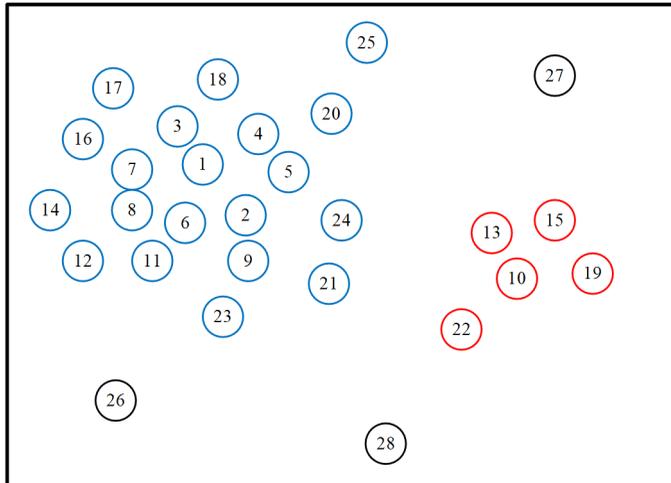
The DPC relative distance δ_i refers to the shortest distance between data object x_i and additional data objects with a higher local density than it.

Equation 2 shows the calculation method of the DPC relative distance δ_i (Rodriguez & Laio, 2014) of the data object x_i :

$$\delta_i = \min_{j: \rho_j > \rho_i} (d_{ij}) \tag{2}$$

In Equation 2, ρ_i represents the DPC local density of the data object and d_{ij} represents the Euclidean distance between the data object x_i and the data object x_j . At the same time, for the data object with the largest DPC local density ρ_i , its DPC relative distance δ_i is usually set as the distance between the objects farthest from the data object, $\delta_i = \max_j (d_{ij})$. It is worth noting that, when the data object x_i is the DPC local or global density maximum, the DPC relative distance will be greater than the traditional KNN distance, so the DPC relative distance of the cluster center will be a very large value.

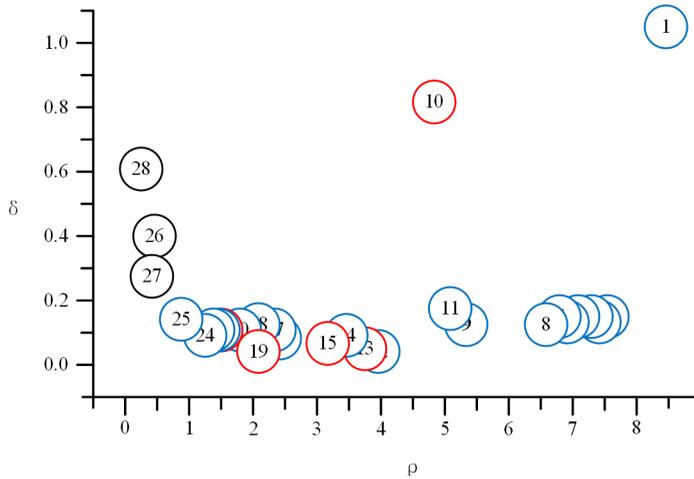
Figure 1. Density Peak Clustering Two-Dimensional Data Distribution Map



After calculating the two indicators of local density and relative distance, the DPC algorithm will create a cluster center decision diagram based on the DPC local density and DPC relative distance to pick the cluster center. The core of the DPC method is the decision graph observation process. Figure 1 is a two-dimensional data distribution map generated by 28 data objects. The distribution map highlights that data objects 1 and 10 are located in the density peak areas in their respective clusters, so the authors regard them as cluster centers. Figure 2 shows a decision graph based on the DPC relative distance and DPC local density. This graph shows that, although the DPC local density of data objects 9 and 10 are very close, the positions of the two data objects in the decision

graph are very different, because the DPC local density of the data object 10 in its cluster is the maximum, and the data object with greater local density than that at this point is located in another cluster, while data object 9 is not the maximum DPC local density in its cluster. As a result, the relative distance of the DPC of data object 10 is much larger than that of the DPC of the data object 9, which is more consistent with the DPC algorithm clustering center hypothesis. Second, as data object 1 is the maximum value of the local density of DPC, two clustering centers, 1 and 10, can be easily selected through this decision graph. Finally, it can be found that data objects 26, 27, and 28 have high relative distances and small DPC local densities. Data objects of this type usually regard these as outliers and cluster them separately.

Figure 2. Density Peak Clustering Decision Diagram



Following selection of the clustering center in the decision graph, the DPC method will assign the remaining data items in the data set to the cluster with the closest clustering center to complete clustering. The density peak clustering DPC algorithm is described as Algorithm 1 (table 1). Importantly, the cluster number c and cluster center must be manually determined using the decision graph.

Table 1. Algorithm 1: Density Peak Clustering Algorithm

Input: Initial data set D and cutoff distance d_c	
Output: c clusters	
(1)	for each $x \in D$ do
(2)	calculate the Euclidean distance between x and other data objects;
(3)	end for
(4)	for each $x \in D$ do
(5)	Equation 1 and the cutoff distance d_c are used to calculate the DPC local density of x ;

Table 1 continued on next page

Table 1 continued

(6)	end for
(7)	for each $x \in D$ do
(8)	Traverse data set D ;
(9)	Equation 2 is used to calculate the DPC relative distance of x ;
(10)	end for
(11)	According to the calculated DPC local density and DPC relative distance, a two-dimensional decision graph is constructed.
(12)	According to the decision graph, the clustering center and the number of clustering c are selected manually.
(13)	Allocate the remaining data objects in the data set to the nearest cluster center.
(14)	Complete clustering and output c clusters.

K-Nearest Neighbor

K-Nearest Neighbor Distance

The KNN distance d_i of data object x_i refers to the average distance between x_i and all data objects in its KNN neighbors.

Equation 3 shows the calculation method of the KNN distance d_i (Dang et al., 2015) of data object x_i :

$$d_i = \frac{1}{k} \sum_{x_j \in KNN_i} dist(x_i, x_j) \quad (3)$$

In Equation 3, KNN_i is the set of KNNs of the exponential data object x_i , and $dist(x_i, x_j)$ is the distance measurement between exponential data objects x_i and x_j , usually using Euclidean distance.

Local Density of K-Nearest Neighbor den_i

The KNN local density den_i of data object x_i is a local density estimation metric constructed from the KNN distance of x_i .

Equation 4 shows the calculation method of the KNN local density den_i of data object x_i :

$$den_i = \frac{1}{d_i} = \frac{k}{\sum_{x_j \in KNN_i} dist(x_i, x_j)} \quad (4)$$

Equation 4 indicates the KNN distance of the data object x_i . Simply speaking, the KNN local density den_i of data object x_i is equal to the reciprocal of the KNN distance d_i of x_i .

In the following, the authors will analyze the calculation process of the KNN distance. When the algorithm calculates KNN distance of data objects, the Euclidian distance between data items must first be determined, with a time complexity of $O(n^2)$. When dealing with large data sets, the

algorithm's time efficiency decreases dramatically as the size of the data set grows. As a result, to increase the algorithm's applicability on large data sets, this work uses the KD-Tree index structure to optimize the KNN distance computation process. First, the data set is modeled and the KD-Tree is constructed. Then, the neighbor sample data are obtained according to the KD-Tree model. The time complexity of the algorithm may be successfully reduced to $O(n \cdot \log n)$ by utilizing the KD-Tree index structure to determine the KNN distance of the data object.

FDPC-OF ALGORITHM

Fast Density Peak Clustering Algorithm

The DPC algorithm requires the initialization of the setting off distance and the artificial selection problem of the clustering center. In this paper, the authors propose using the KNN local density to replace the traditional local density estimation in the DPC algorithm, avoiding manually setting the cutoff distance, and solves the DPC algorithm for the cutoff distance parameter sensitivity issue. To avoid the potential difficulties of picking the cluster center artificially, the cluster center is determined using the product of local density and relative distance. In addition, to lower the temporal complexity of the DPC algorithm while computing the KNN local density, the authors employed the KD-Tree index data structure to aid in calculating the K closest neighbor of each data object and reducing the algorithm's time complexity. In addition, when calculating relative distance, only the data object's K closest neighbors are evaluated, avoiding global search when calculating relative distance.

K-Nearest Neighbor Relative Distance ω_i

The KNN relative distance ω_i is the shortest distance between the exponential data object x_i and other data objects whose KNN local density is higher in KNN.

Equation 5 shows the calculation method of the KNN relative distance ω_i of the data object x_i :

$$\omega_i = \left\{ \min(\text{dist}(x_i, x_j)) \mid x_j \in KNN_i, \text{den}_j > \text{den}_i \right\} \quad (5)$$

In Equation 5, $\text{dist}(x_i, x_j)$ is the distance measurement between data objects x_i and x_j which usually adopts Euclidean distance. KNN_i is the set of KNNs of the data object x_i ; den_i is the KNN local density of the data object x_i . In addition, when the data object x_i is the local maximum density in its KNN_i , it will be put into *maxden*, the neighbor maximum density set. Then, the minimum distance between the data object x_i and other data objects in *maxden* set is regarded as the KNN relative distance ω_i of x_i . Equation 6 shows the calculation method6:

$$\omega_i = \left\{ \min(\text{dist}(x_i, x_j)) \mid x_j \in \max \text{den} \right\} \quad (6)$$

Clustering Center Decision Indicators cen_i

The product of the KNN local density and KNN relative distance of data object x_i is called clustering center decision indicator cen_i .

Equation 7 shows the calculation method of decision index cen_i of the clustering center of data object x_i :

$$\text{cen}_i = \text{den}_i \times \omega_i \quad (7)$$

In Equation 7, den_i is the KNN local density of the data object x_i ; ω_i is the KNN relative distance of the data object x_i . After calculating the KNN local density den_i , KNN relative distance ω_i and clustering center decision index cen , the first c data objects with the largest cen are selected as clustering centers according to the clustering parameter c initialized by users. Finally, the remaining data objects in the data set are successively allocated to the nearest cluster center to complete clustering. The fast density peak clustering algorithm (Fast-DPC) is described as Algorithm 2 (table 2).

Table 2. Algorithm 2: Fast-DPC Algorithm

Input: Initial data set D , parameter k , and clustering number c	
Output: c clusters	
(1)	KD-Tree is constructed according to data set D ;
(2)	for each $x \in D$ do
(3)	according to KD-Tree and Equation 3, the KNN distance of data object x is calculated;
(4)	end for
(5)	for each $x \in D$ do
(6)	Equation 4 is used to calculate the KNN local density of x ;
(7)	end for
(8)	for each $x \in D$ do
(9)	Equation 5 and Equation 6 are used to calculate the KNN relative distance of x .
(10)	end for
(11)	for each $x \in D$ do
(12)	Equation 7 is used to calculate the clustering center decision indicators of x .
(13)	end for
(14)	the data object with a large c before the decision indicators cen of the cluster center is taken as the cluster center;
(15)	allocate the remaining data objects in the data set to the nearest cluster center;
(16)	complete clustering and output c clusters.

FDPC-OF Algorithm Description

In this research, the authors proposed a new outlier detection algorithm based on the FDPC-OF by combining the measurement indexes of Fast-DPC. Following the completion of the clustering algorithm, to better describe the degree of outliers in some data sets with different densities, the KNN relative distance in fast density peak clustering will be further improved.

Mean Centripetal Distance $davg_i$

The mean distance between all data objects in each cluster and the cluster center is regarded as the mean centripetal distance $davg_i$.

Equation 8 shows the calculation method of the mean centripetal distance $davg_i$ of clustering C_i :

$$davg_i = \frac{1}{n} \sum_{x_j \in C_i} dist(x_c, x_j) \quad (8)$$

In Equation 8, n represents the number of data objects clustering C_i ; $dist(x_c, x_j)$ is the distance between the data object x_j and the cluster center x_c of the cluster to which it belongs.

Centripetal Relative Distance $discen_i$

The ratio of the KNN relative distance ω_i of data object x_i to the average centripetal distance $davg_i$ of its cluster is regarded as the centripetal relative distance $discen_i$.

Equation 9 shows the calculation method of the centripetal relative distance of data object x_i :

$$discen_i = \frac{\omega_i}{davg_i} \quad (9)$$

In Equation 9, ω_i represents the KNN relative distance of data object x_i ; $davg_i$ represents the mean centripetal distance of the data object x_i .

Figure 3. Two-Dimensional Data Distribution of Fast-DPC



Centripetal relative distance describes the degree of outliers in some data sets with different densities in the form of the ratio of KNN relative distance to average centripetal distance. As Figure 3 shows, data object 24 is the global outlier, and the relative distance of this data object is the maximum. However, data object 25 is a local outlier, but the relative distance of the data object is smaller than that of the normal points in the sparse cluster. The local outlier is easily hidden by the normal points in the sparse region if the relative distance of KNN is used as an outlier indicator. Therefore, the

authors propose the average centripetal distance based on the KNN relative distance, and form the centripetal relative distance in the form of a ratio. Since data object 25 belongs to the left cluster, when the KNN relative distance is higher than the average centripetal distance of the upper left cluster, the outlier characteristics of the local outlier are amplified and the outlier degree is better described.

Fast Density Peak Clustering Outliers FOF_i

The ratio of the centripetal relative distance $discen_i$ and KNN local density den_i of data object x_i is regarded as the fast density peak clustering outlier FOF .

Equation 10 shows the calculation method:

$$FOF_i = \frac{discen_i}{den_i} \tag{10}$$

This outlier is composed of the ratio of the centriole relative distance and KNN local density. From the perspective of density, the authors first constructed a KD-Tree according to the data set, and then calculated the KNN local density of the data object according to the index structure. The local density can accurately describe the density of the region where the data object is located. Outliers are usually located in regions with low local density. From the perspective of distance, centripetal relative distance makes use of the relationship between clustering center and data object, and can have good adaptability in some data sets with different densities. Usually, the centripetal relative distance of outliers is much larger than that of normal points. Therefore, the FDPC-OF can be a very good description of the degree of outlier data objects, usually outlier factor larger points are more likely to stray, so after the sort algorithm is selected from the group of factors first output, o points as outliers in the actual application of the algorithm, and the o value is based on the actual situation of the artificial data sets. The value of o is selected according to the number of outlier labels in the data set in the experiment in this study to validate the efficacy of the algorithm on each data set. Algorithm 3 (table 3) is an outlier detection algorithm based on the FDPC-OF.

Table 3. Algorithm 3: FDPC-OF Algorithm

Input: Initial data set D , parameter k , and clustering number c	
Output: o outliers	
(1)	Algorithm 2 is called to cluster data set D and c clusters are outputted;
(2)	for each $c \in C$ do
(3)	Equation 8 is used to calculate the average centripetal distance of c ;
(4)	end for
(5)	for each $x \in D$ do
(6)	Equation 9 is used to calculate the centripetal relative distance of x ;

Table 1 continued on next page

Table 1 continued

(7)	end for
(8)	for each $x \in D$ do
(9)	Equation 10 is used to calculate the outlier factor of the fast density peak clustering of x ;
(10)	end for
(11)	the outliers of fast density peak clustering in descending order;
(12)	output the first o points as outliers.

FDPC-OF Algorithmic Correctness

In the FDPC-OF algorithm, the KD-Tree index structure can effectively reduce the time complexity; thus, the density estimate of each data object based on KNN can well represent the sparse density of the region where the data object is located, improving computing performance. Then, according to Algorithm 2 after clustering, taking the ratio of the KNN relative distance to the average centripetal distance as the centripetal relative distance can better magnify outlier characteristics in some data sets with different density distributions. Finally, the outlier degree of data items is defined using the fast density peak clustering outlier factor, and the data objects are sorted in decreasing order, with the first o outliers being output.

FDPC-OF Algorithm Complexity

The following two pieces make up the temporal complexity of the FDPC-OF algorithm:

- Calculate the KNN local density of the data object using the index structure KD-Tree, and the time complexity is $O(n \cdot \log n)$, where, n is the number of data objects in the data set.
- Calculate the FDPC-OF of the data object using the index structure KD-Tree, and the time complexity is $O(n)$.

Thus, the total time complexity of the FDPC-OF algorithm is:

$$O(n \cdot \log n) + o(n) \approx o(n \cdot \log n)$$

EXPERIMENT AND ANALYSIS

The authors carried out studies on actual and artificial data sets to assess the FDPC-OF algorithm in this study, and chose seven other outlier detection algorithms for comparison experiments. These algorithms include the COF(connectivity-based outlier factor) algorithm (Tang et al., 2002), NOF(natural outlier factor) algorithm (Huang et al., 2016), RDOS(relative density-based outlier score) algorithm (Tang & He, 2017), LDF(local density factor) algorithm (Latecki et al., 2007), IForest algorithm (Liu et al., 2012), NANOD(natural neighbour-based outlier detection) algorithm(Wahid & Annavarapu, 2021), and MOD(mean-shift outlier detector) algorithm (Yang et al., 2021). The authors

adopted the above seven comparison algorithms because the FDPC-OF algorithm in this study is an outlier detection algorithm based on density as a whole. The NOF algorithm, RDOS algorithm, LDF algorithm, and COF algorithm are all classical outlier detection algorithms based on density, in recent years, so they have strong comparison. The IForest algorithm is a novel outlier detection algorithm with better accuracy and time efficiency, in recent years. The NANOD algorithm and MOD algorithm are relatively advanced density-based and distance-based outlier detection algorithms, in the last two years. Table 4 shows the configuration of the experimental environment.

Table 4. Configuration Table of the Experimental Environment

Hardware and software environment	Parameter
CPU	2.60 Hz Inter i7-4720HQ
The hard disk	512.0 GB
Memory	16.0 GB
The development environment	PyCharm
Compile environment	Python 3.8
Visualization tool	PyCharm

Evaluation Measures

The authors evaluated the effectiveness of the algorithm experiment using two performance metrics, in this study, including precision and running time.

The precision evaluation index, also called the precision rate, is the ratio of the number of outliers successfully recognized by the algorithm to the number of outliers incorrectly detected by the method. The accuracy ranges from 0 to 1. The large the index value is, the better the outlier identification algorithm’s detection impact. Equation 11 shows the calculation methods of precision:

$$Pr = \frac{TP}{TP + FP} \tag{11}$$

In Equation 11, TP denotes the number of outliers successfully detected by the algorithm, and FP denotes the number of outliers mistakenly detected by the algorithm.

In terms of parameter selection, the authors set the related parameters of the FDPC-OF algorithm as follows: *k* is set to 5 in the nearest neighbor calculation, and *c* is set to 2 in the automatic selection of the cluster center.

In addition, they set the parameters of the comparison algorithms as follows: The default value of parameter *k* of the RDOS algorithm and COF algorithm is 5; the default values of θ and *k* of the LDF algorithm are 0.5 and 5, respectively. The NOF algorithm is the no-parameter algorithm without setting, while IForest and MOD are the default setting in the original literature.

Artificial Data Set

The authors employed four types of two-dimensional artificial data sets (Figure 4) for comparison studies to validate the detection capacity of outliers of the algorithm under varied data distributions. Outliers are the points represented by “o,” and Table 5 shows the data characteristics of the four kinds of artificial data sets.

Figure 4. Data Distribution of Manual Data Sets D1~D4

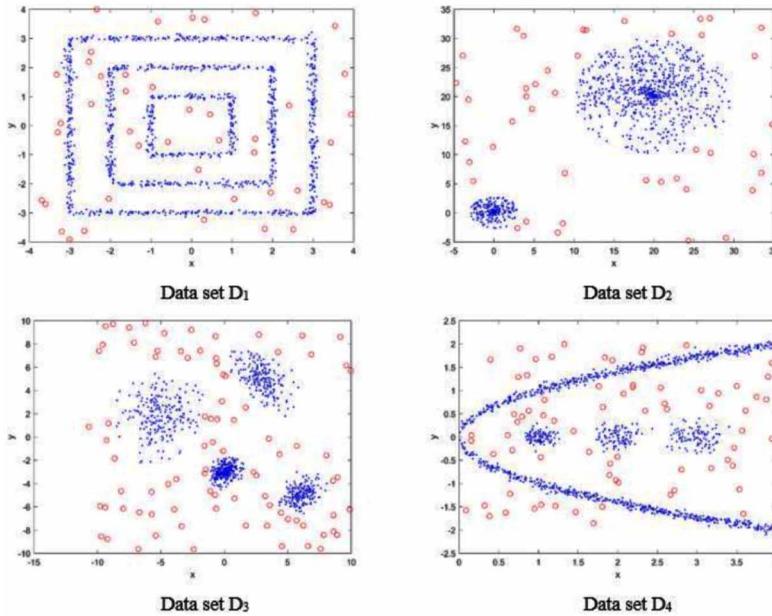


Table 5. Data Characteristics of Artificial Data Sets

Data set	Number of samples	Number of outliers	Outlier ratio
D1	1256	43	3.4%
D2	1043	43	4.1%
D3	1000	85	8.5%
D4	1372	72	5.2%

Table 6 shows the experimental results of the precision of the FDPC-OF algorithm and the other seven comparison algorithms on different artificial data sets. As the experimental data in table 6 show, the average precision of the FDPC-OF algorithm in this study under the four kinds of artificial data sets reaches more than 95%, which is higher than the other seven comparison algorithms, and the precision is as high as 100% on the D1 data set.

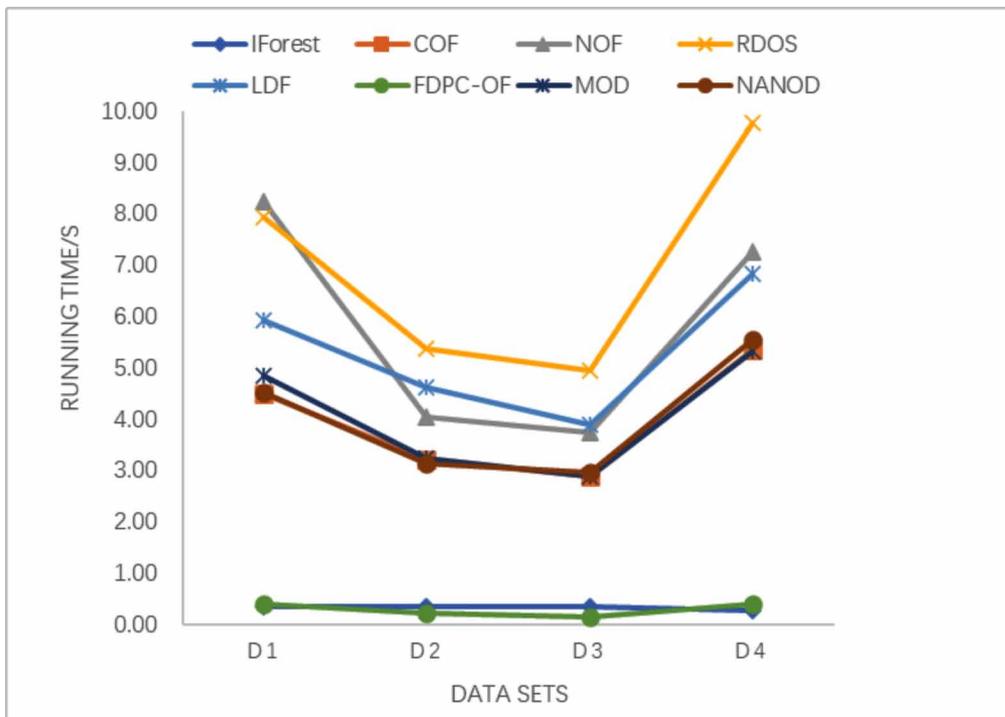
Simultaneously, the experimental results show that the FDPC-OF algorithm has higher outlier identification stability than other algorithms on four distinct types of artificial data sets with varying data distributions. This is because the FDPC-OF, on the basis of the original index of density peak clustering, combined with the concept of clustering center, proposed the centripetal relative distance, and improved the applicability of the algorithm in some data sets with different densities.

Table 6. Accuracy of Each Algorithm on the Artificial Data Set

Algorithm	D1	D2	D3	D4
FDPC-OF	1	0.9767	0.8705	0.9583
COF	0.8604	0.9534	0.7647	0.9305
NOF	0.9534	0.4186	0.5764	0.75
LDF	0.9302	0.9767	0.847	0.8194
RDOS	0.9767	0.3255	0.5882	0.8472
IForest	0.5348	0.9302	0.7647	0.2638
NANOD	0.4883	0.3255	0.7176	0.1805
MOD	0.8372	0.9302	0.7647	0.8889

Figure 5 shows the experimental results of the FDPC-OF algorithm and seven other comparison algorithms on different manual data sets. As the experimental data in Figure 5 show, the average running time of the FDPC-OF algorithm on four kinds of artificial data sets is only 0.29 seconds, which is faster than other comparison algorithms. This is because the FDPC-OF algorithm uses the index structure when calculating the KNN, thus reducing the time complexity of the algorithm. Simultaneously, the algorithm in this paper has an excellent running rate due to the simple calculation process.

Figure 5. Running Time of Each Algorithm



Real Data Set

In this study, the authors used the following four real data sets, all from UCI data sets: Ionosphere, Iris, Wdbc, and Vowels. The ratio of outliers selected in the real data set ranges from 3.4% to 35.9% and the number of attributes ranges from 4 to 34. This ensures the diversity and complexity of the ratio of outliers and the number of attributes in the data set, and can better test the outlier detection ability of the proposed algorithm and the comparison algorithm on the real data set. Table 7 shows the data characteristics of its real data set.

Table 7. Data Characteristics of Real Data Sets

Data set	Number of samples	Number of attributes	Number of outliers	Outlier ratio
Ionosphere	351	34	126	35.9%
Iris	110	4	10	9.1%
Wdbc	390	30	33	8.4%
Vowels	1456	12	50	3.4%

Table 8 shows the experimental results of the precision of the FDPC-OF algorithm and the other seven comparison algorithms on four real data sets. As the experimental data in table 8 show, on four real data sets, the precision of the FDPC-OF method approaches 80%, which is greater than the precision of the other seven comparison algorithms. In four real data sets of varying sizes, different numbers of attributes and different outlier ratios, the FDPC-OF algorithm has stable outlier detection efficiency.

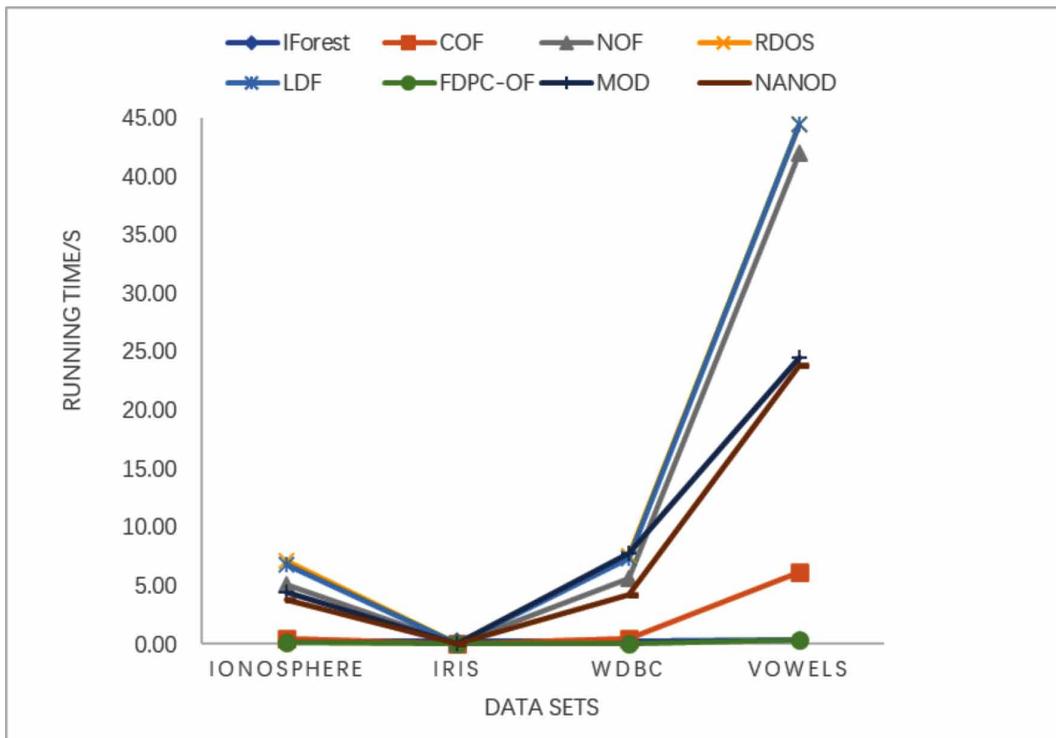
Table 8. Accuracy of Each Algorithm on the Real Data Set

Algorithm	Ionosphere	Iris	Wdbc	Vowels
FDPC-OF	0.9126	0.7	0.7878	0.78
COF	0.7777	0.6	0.3636	0.5
RDOS	0.5317	0.6	0.1515	0.04
LDF	0.873	0.7	0.3333	0.48
NOF	0.6904	0.5	0.1212	0.26
IForest	0.6349	0.5	0.5757	0.14
NANOD	0.7381	0.8	0.7878	0.52
MOD	0.8492	0.6	0.7878	0.42

Figure 6 shows the experimental findings of running time of the FDPC-OF algorithm as well as the other seven comparative algorithms using actual data sets. As the experimental data in Figure 6 show, the average running time of the FDPC-OF algorithm on four kinds of artificial data sets is only 0.16 seconds, which is faster than other comparison algorithms. In Ionosphere and Wdbc data sets, the FDPC-OF algorithm, COF algorithm, and IForest algorithm have excellent running speed, and the running time is less than 1 second. In the Iris data set, the running rate of each data set has a good running rate, and the running time is less than 1 second. However, on Vowels data sets,

except for the FDPC-OF algorithm and IForest algorithm, other comparable algorithms have poor performance, with the NOF, RDOS, and LDF algorithms taking more than 40 seconds to complete. Thus, when the authors' algorithm processes large-scale data sets, it still has a very good running rate. Based on the above experimental analysis, the running rate of the proposed algorithm on real data sets is generally faster than that of other comparison algorithms, so the computational efficiency of the proposed algorithm is quite excellent.

Figure 6. Running Time of Each Algorithm



CONCLUSION

In this paper, the authors first analyzed the DPC algorithm and its algorithm idea, in view of the DPC algorithm with high time complexity and the problem of artificial setting parameters carried out on the thorough research, and they presented a Fast-DPC algorithm, the algorithm used in the KNNs clustering algorithm to replace the density peak of density estimate. They employed the index structure to improve the distance calculation, resulting in a significant reduction in clustering time complexity and avoiding setting truncated distance d_c . Then, the authors defined the centripetal-relative distance, proposed a new outlier detection technique based on the FDPC-OF, and defined the fast density peak clustering outlier factor to characterize the data object's outlier degree. Finally, they demonstrated that the FDPC-OF algorithm can detect outliers effectively and thoroughly by analyzing the algorithm's accuracy and complexity, as well as testing the algorithm in artificial and actual data sets. In the future, the authors will explore how to apply the proposed algorithm to various practical

applications. In addition, how to improve the effectiveness of the algorithm in high-dimensional data sets will also be the focus of their research.

COMPETING INTERESTS

The authors have declared that there is no conflict of interest regarding the publication of this paper.

FUNDING AGENCY

This research was supported by the National Natural Science Foundation of China [No.61972334]; the National Social Science Foundation of China General Project [No.20BJ122]; Hebei Province Innovation Capability Improvement Plan Project [No.20557640D]; and the Intelligent image workpiece recognition of Sida Railway (Enterprise Cooperative Project) [No. x2021134].

REFERENCES

- Andrysiak, T. (2020). Sparse representation and overcomplete dictionary learning for anomaly detection in electrocardiograms. *Neural Computing & Applications*, 32(5), 1269–1285. doi:10.1007/s00521-018-3814-5
- Avdiienko, V., Kuznetsov, K., Rommelfanger, I., Rau, A., Gorla, A., & Zeller, A. (2017). Detecting behavior anomalies in graphical user interfaces. In *Proceedings of the 2017 IEEE/ACM 39th International Conference on Software Engineering Companion (ICSE-C)* (pp. 201-203). IEEE. doi:10.1109/ICSE-C.2017.130
- Bhatti, M. A., Riaz, R., Rizvi, S. S., Shokat, S., Riaz, F., & Kwon, S. J. (2020). Outlier detection in indoor localization and Internet of Things (IoT) using machine learning. *Journal of Communications and Networks (Seoul)*, 22(3), 236–243. doi:10.1109/JCN.2020.000018
- Breunig, M. M., Kriegel, H. P., Ng, R. T., & Sander, J. (2000). LOF: Identifying density-based local outliers. In *Proceedings of the 2000 ACM SIGMOD international conference on Management of data* (pp. 93-104). ACM. doi:10.1145/342009.335388
- Comaniciu, D., & Meer, P. (2002). Mean shift: A robust approach toward feature space analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(5), 603–619. doi:10.1109/34.1000236
- Dang, T. T., Ngan, H. Y., & Liu, W. (2015). Distance-based k-nearest neighbors outlier detection method in large-scale traffic data. In *Proceedings of the 2015 IEEE International Conference on Digital Signal Processing (DSP)* (pp. 507-510). IEEE. doi:10.1109/ICDSP.2015.7251924
- Denning, D. E. (1987). An intrusion-detection model. *IEEE Transactions on Software Engineering*, 13(2), 222–232. doi:10.1109/TSE.1987.232894
- Ester, M., Kriegel, H. P., Sander, J., & Xu, X. (1996). A density-based algorithm for discovering clusters in large spatial databases with noise. *Data Mining and Knowledge Discovery*, 96(34), 226–231.
- Huang, J., Zhu, Q., Yang, L., & Feng, J. (2016). A non-parameter outlier detection algorithm based on natural neighbor. *Knowledge-Based Systems*, 92(15), 71–77. doi:10.1016/j.knsys.2015.10.014
- Jain, A. K., Murty, M. N., & Flynn, P. J. (1999). Data clustering: A review. *ACM Computing Surveys*, 31(3), 264–323. doi:10.1145/331499.331504
- Kafadar, K., Barnett, V., & Lewis, T. (1995). Outliers in statistical data, 3rd ed. Journal of the American Statistical Association, 90(429), 395. doi:10.2307/2291180
- Karypis, G., Han, E. H., & Kumar, V. (1999). Chameleon: Hierarchical clustering using dynamic modeling. *Computer*, 32(8), 68–75. doi:10.1109/2.781637
- Knorr, E. M., & Ng, R. T. (1997). A unified approach for mining outliers. In *Proceedings of the 1997 Conference of the Centre for Advanced Studies on Collaborative Research* (p. 11). IBM Press.
- Knorr, E. M., Ng, R. T., & Tucakov, V. (2000). Distance-based outliers: Algorithms and applications. *The VLDB Journal*, 8(3), 237–253. doi:10.1007/s007780050006
- Latecki, L. J., Lazarevic, A., & Pokrajac, D. (2007). Outlier detection with kernel density functions. In *International Workshop on Machine Learning and Data Mining in Pattern Recognition* (pp. 61-75). Springer. doi:10.1007/978-3-540-73499-4_6
- Liu, F. T., Ting, K. M., & Zhou, Z. H. (2012). Isolation-based anomaly detection. *ACM Transactions on Knowledge Discovery from Data*, 6(1), 1–39. doi:10.1145/2133360.2133363
- MacQueen, J. (1967). Classification and analysis of multivariate observations. *Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability* (vol. 1, pp. 281-297). University of California Press.
- Ngai, E., Hu, Y., Wong, Y. H., Chen, Y., & Sun, X. (2011). The application of data mining techniques in financial fraud detection: A classification framework and an academic review of literature. *Decision Support Systems*, 50(3), 559–569. doi:10.1016/j.dss.2010.08.006
- Ramotsoela, D., Abu-Mahfouz, A., & Hancke, G. (2018). A survey of anomaly detection in industrial wireless sensor networks with critical water system infrastructure as a case study. *Sensors (Basel)*, 18(8), 2491. doi:10.3390/s18082491 PMID:30071595

- Rodriguez, A., & Laio, A. (2014). Clustering by fast search and find of density peaks. *Science*, 344(6191), 1492–1496. doi:10.1126/science.1242072 PMID:24970081
- Safaei, M., Asadi, S., Driss, M., Boulila, W., & Safaei, M. (2020). A systematic literature review on outlier detection in wireless sensor networks. *Symmetry*, 12(3), 328. doi:10.3390/sym12030328
- Schubert, E., Zimek, A., & Kriegel, H. P. (2014). Generalized outlier detection with flexible kernel density estimates. In *Proceedings of the 2014 SIAM International Conference on Data Mining* (pp. 542-550). Society for Industrial and Applied Mathematics. doi:10.1137/1.9781611973440.63
- Seheult, A. H., Green, P. J., Chevalier, A., & Sharples, L. (1989). Robust regression and outlier detection. *Journal of the Royal Statistical Society. Series A, (Statistics in Society)*, 152(1), 133. doi:10.2307/2982847
- Tang, B., & He, H. (2017). A local density-based approach for outlier detection. *Neurocomputing*, 241(7), 171–180. doi:10.1016/j.neucom.2017.02.039
- Tang, J., Chen, Z., Fu, A. W. C., & Cheung, D. W. (2002). Enhancing effectiveness of outlier detections for low density patterns. In *Pacific-Asia conference on knowledge discovery and data mining* (pp. 535–548). Springer. doi:10.1007/3-540-47887-6_53
- Wahid, A., & Annavarapu, C. S. R. (2021). NaNOD: A natural neighbour-based outlier detection algorithm. *Neural Computing & Applications*, 33(6), 2107–2123. doi:10.1007/s00521-020-05068-2
- Wang, L., Feng, C., Ren, Y., & Xia, J. (2019). Local outlier detection based on information entropy weighting. *International Journal of Sensor Networks*, 30(4), 207–217. doi:10.1504/IJSNET.2019.101239
- Xu, X., Ding, S., Du, M., & Xue, Y. (2018). DPCG: An efficient density peaks clustering algorithm based on grid. *International Journal of Machine Learning and Cybernetics*, 9(5), 743–754. doi:10.1007/s13042-016-0603-2
- Yang, J., Rahardja, S., & Fränti, P. (2021). Mean-shift outlier detection and filtering. *Pattern Recognition*, 115(161-171), 107874.
- Yang, X., & Liu, X. (2020). Local outlier factor algorithm based on correction of bidirectional neighbor. *Journal of Communication*, 41(8), 131–140.
- Zhang, K., Hutter, M., & Jin, H. (2009). A new local distance-based outlier detection approach for scattered real-world data. In *Proceedings of the Pacific-Asia Conference on Knowledge Discovery & Data Mining* (pp. 813-822). Springer. doi:10.1007/978-3-642-01307-2_84

ZhongPing Zhang (1972–), male, born in Songyuan, Jilin province, Ph. D., professor at Yanshan Uni-versity. He has been working on big data and data mining for many years, and his research interests include big data, data mining, and semi-structured data.

Li Sen (1997 –), male, born in Zhoukou, Henan province, his undergraduate major was computer Sci-ence and Technology. In 2021, he entered Yanshan University for a master's degree in computer tech-nology. His current research direction is data mining.

WeiXiong Liu (1997–), male, born in Guangzhou, Guangdong province. He is currently pursuing a mas-ter's degree in computer technology at Yanshan University in the field of data mining.

Ying Wang (1980-), female, born in Cixian, Hebei province, Ph.D., associate professor at Yanshan Uni-versity. She has a wide range of research areas, among which her main research interests are Business Process Management, semi-structured data and Petri net.

Daisy Xin Li (1972-), female, born in Tianjin, China, MBA, majors in Computer Science, Finance, Ac-counting. A former Senior Technical Product Manager/Product Owner at Wolters Kluwer Tax & Ac-counting, and the current Senior Business Analyst at Herbalife Nutrition. Her research interests include leveraging Artificial Intelligence with big data, and data mining in business solutions and implementations in Finance, Accounting, and Digital Marketing fields.