International Journal of Research Studies in Computer Science and Engineering (IJRSCSE)
Volume 7, Issue 1, 2020, PP 38-46
ISSN 2349-4840 (Print) & ISSN 2349-4859 (Online)
DOI: http://dx.doi.org/10.20431/2349-4859.0701005
www.arcjournals.org

Uncertain Data Outlier Detection Algorithm based on iForest and LOF

AMADY BA , Zhenfang ZHU*

School of Information Science and Electric Engineering, Shandong Jiaotong University, 250357, Jinan, China

Corresponding Author:** ***Amady BA, *School of Information Science and Electric Engineering, Shandong Jiaotong University, 250357, Jinan, China*

Abstract: *With the advancement of technology and the deepening of people's understanding of data acquisition and processing technologies, as a new type of data model, uncertainty data is widely used in finance, military, logistics, telecommunications and other fields. The traditional data management technology can not manage the uncertain data effectively. The management technology for uncertain data has become a research hotspot in the field of data mining. Many traditional data mining technologies have been extended and applied to uncertain data. Outlier detection is an important technology in the field of data mining. It is used to discover objects whose behavior is different from other data objects and has high application value in network intrusion and sensor network detection. Outlier detection for deterministic data has been studied in depth, but it is a new topic in the new area of uncertain data. We study a possible world model based on attribute-level uncertain data and proposes a new outlier detection method. Firstly, iForest's anomaly score calculation method was improved to make it applicable to uncertain data. Second, the local outliers of uncertain data are redefined. Then, we use iForest and K nearest neighbor query optimization to reduce the candidate set of data without expanding the possible world. Finally, experiments verify the performance of the algorithm. Experimental results show that the proposed algorithm can effectively improve the accuracy, reduce the time complexity, and improve the outlier detection performance of uncertain data.*

Keywords: *Uncertain Data; Outlier Detection; Isolation Forest; LOF*

1. Introduction

Outlier detection is one of important research fields of data mining, and its purpose is to eliminate noise or find potential and meaningful knowledge, and find data objects of which behavior is obviously different from other objects [1]. The existing outlier detection technology is mainly applied to the traditional database at present, and the data existence and accuracy are certain, and with the constant deepening of data collection and processing technology, people’s attention has been drawn by uncertain data. In lots of practical application, due to data noise, transmission delay, inaccurate measurement and other factors, uncertain data widely exists, such as sensor network and network intrusion detection information retrieval. Because of randomness and complexity of uncertain data, traditional data analysis technologies cannot effectively process uncertain data, so it is of important significance to study the outlier detection of uncertain data.

At present, among most application, uncertain data is mainly divided into two types: existence-level uncertainty and attribute-level uncertainty. Existence-level uncertainty refers to an uncertain value showing whether an object exists in the database, and it is usually expressed as probability value. The uncertainty is divided into two situations: mutual independence or correlative dependence; attribute-level uncertainty refers to the situation that an object certainly exists, but due to measurement errors and other reasons, an attribute of an object exists at a certain probability, and it is the uncertainty of attribute of object. This article researches attribute-level uncertain data.

Table1. *Uncertain Data*

ID	Attribute 1	Attribute 2	……	Probability
t_1	32	34		0.5
t_2	23	43		0.2
t_3	25	35		0.4
t_4	32	43		0.7

Table 1 is a typical table composed of uncertain data; ID represents uncertain data object ID; Attribute 1 and Attribute 2 represent two attributes on the dimension of object value, respectively; probability represents the existence probability of the object.

At present, the outlier detection of deterministic data is mainly divided into the following types [2,3]:

Distance-based outlier detection: The distance-based outlier detection judges whether an object is an outlier by detecting whether a data object in a data set has sufficient data objects within a certain range. Reference [4] is the first one to put forward the concept of distance-based outlier: with object X as the center, within the range of radius R, if the number of data objects is less than the threshold value K, then the object X is the outlier. In other words, the distance-based outlier has no sufficient neighbors. This method is simple and easy to realize, but it has a quite higher time complexity, and the effect of local outlier detection is not obvious.

Density-based outlier detection: When the data distribution density is large, Breunig et al. [5] put forward the concept of local outlier factor, and calculated the distance between objects through the dimension of data space, and obtained the reachable density of objects, and finally, sorted the LOF value of each object calculated in descending order. The first n data objects are density-based outliers, and the outlier degree of object depends on the isolation situation of surrounding neighbors. This method requires calculating the LOF value of each data object, but in reality, there are a very few outliers in data sets, so a lot of calculations are unnecessary.

Statistics-based outlier detection [6,7]: The statistics-based outlier detection is a model-based method, and it creates a distribution or probability model for the data set to be detected, and then, detects the outlier objects and the number by use of inconsistency. This method applies to numeric data only, and it is essential to know the distribution feature of data [8].

Deviation-based outlier detection: The deviation-based outlier detection identifies outliers through detection of the main features of a group of objects. At present, the deviation-based outlier detection is mainly divided into two types: sequence abnormity technology [9] and data cube technology [10]. In theory, this method can mine various types of data, but this shall be built under the condition of knowing the features of data in advance, so this detection technology is rarely used, compared with the said three methods.

Among uncertain data, the most basic method for evaluating whether an object is an outlier is the possible world model. As for how to effectively detect outliers, the naïve idea is to expand all possible worlds, but this method is not feasible because of the huge number of data objects and exponential growth of possible world space. Thus, the main problem confronted at present is to find a feasible and effective filtering method so as to reduce the detection times.

Reference [11] is the first one to put forward performing outlier detection of uncertain data, and express each object with the probability density function, and sample uncertain data, and determine whether it is an outlier or not based on the probability of uncertain data in a certain range.

Reference [12] put forward a distance-based outlier detection method. Firstly, outliers were defined by use of possible world model, and then, the fundamental detection algorithm and dynamic pruning method were put forward by use of the grid pruning. But this method achieves a good effect with respect to data sets with even density.

Reference [13] put forward a density-based outlier detection method. Firstly, local outlier factors based on uncertain data were designed, an LOF algorithm of uncertain data was deduced, and pruning was conducted by use of network. But this method is greatly affected by parameters.

Salman Ahmed Shaikh et al. [14-16] put forward the cell-based outlier detection algorithm for uncertain data, and aimed at the problem of parameter threshold setting and outlier degree ordering, a sequential cell list structure, and an approximate detection algorithm for uncertain data outlier is raised. This article studies uncertain data with discrete probability, and in combination with the features of Isolation Forest and LOF, the outlier detection method of uncertain data set is proposed. Firstly, this article uses the Isolation Forest algorithm for filtering and reducing of original data size; then, with them as the candidate set, on the basis of determining the local outlier factor of data set, the definition of local outlier factor of uncertain data is designed, and the algorithm is improved. Outliers can be obtained with the help of this algorithm at a high efficiency.

2. Relevant Work

2.1. Isolation Forest

The iForest (Isolation Forest) [17,18] was raised by Liu Fei based on the feature that the abnormal data size is small and the difference with the normal data is large. The core of this algorithm is the iForest composed of iTrees. iTree is a random binary tree, and to build an iTree, it is essential to obtain a data subset D1 from data set D by means of random sampling, and then, an attribute X and a separated value p are selected randomly from D1={d1, d2,……, dn,}; finally, each data di is divided according to the value of attribute X; if di(X)<p, the data shall be put at the left child node, and on the contrary, it shall be put at the right child node. An iTree shall be constituted iteratively in this mode, till either of the following conditions is satisfied: (1) The tree reaches the maximum height; (2) The data set has only one piece of data; (3) there are many pieces of the same data in the data set.

The iForest algorithm is constructed by t iTrees built and designed by users, and detection object x shall transverse each iTree so as to determine the child node of x and calculate the abnormal score of x and obtain the abnormal degree of object x.

The example that 4 detected samples traverse one iTree is shown in Figure 1:

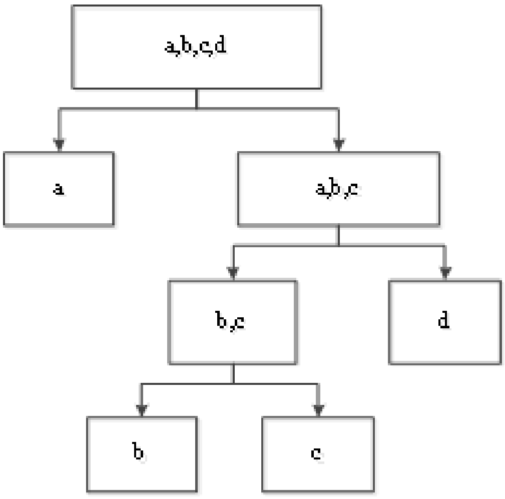

Figure1. *iTree Traversal of Detection Sample*

It can be seen from Figure 1 that the height of b and c is 3; the height of d is 2; the height of a is 1. Thus, a is the earliest to be isolated, so it is most likely to be abnormal.

Since the structure of iTree is random, the result of single figure is not reliable, but through a large number of iTrees, the robustness of this algorithm is greatly enhanced. For n sample data, the path length is h(x); since iTree has the same structure with binary search tree, h(x) is equal to the path length of unsuccessful query in the binary search tree. Reference [19] provides the path length of binary search tree:

$$c(n) = 2H(n-1) - (2(n-1)/n) \qquad (1)$$

Where, H(n)=ln(n)+γ, γ is Euler's constant, c(n) is the average value of h(x), when n training samples are given, and it is used for standardization of h(x). The abnormal score s of detection object x is shown in Formula (2):

$$s(x,n) = 2^{-\frac{E(h(x))}{c(n)}} \qquad (2)$$

Where, E(h(x)) is the average value of h(x) in the iForest. It can be seen from Formula (2) that

- When E(h(x)) →c(n), s→0.5, that is, when s of all points is around 0.5, there are no obvious abnormal values in the samples.
- When E(h(x)) →0, s→1, then the object is an abnormal value.
- When E(h(x)) →n-1, s→0, then the object is a normal value.

2.2. LOF Local Outlier Factor of Uncertain Data

The study in this article is based on attribute-level uncertain data, and it adopts the possible world model. Each data object consists of several examples, and each example has n-dimensional features; the distance between two examples adopts Euclidean distance, and the following is the basic definition:

Definition 1 Possible world Set of uncertain data D={u1, u2,… ,ui,… ,un},$\tilde{u}_i$ represents a possible example of uncertain data object ui, and the possible world consists of the set of any example of each data object. Thus, the total number of possible worlds in uncertain data is:

$$|N| = \prod_{u\in D} |u| \qquad (3)$$

Provided that uncertain objects are mutually independent, the probability of possible world is:

$$P(W) = \prod_{u\in D, |\tilde{u}\cap W|=1} P(\tilde{u}) \prod_{u\in D, |\tilde{u}\cap W|=\varnothing} (1-P(\tilde{u})) \qquad (4)$$

Definition 2 k neighbor distance of example For any natural number k, in the possible world W, the kth distance of example $\tilde{u}_i$ is defined as the distance between example $\tilde{u}_i$ and example $\tilde{u}_j$, and recorded as $k - \mathrm{dis}(\tilde{u}_i)$, where $\tilde{u}_j$ meets the following conditions:

- At least k examples $\tilde{u}_j{}' \in W\backslash\{p\}$ exist, which satisfy $\mathrm{dis}(\tilde{u}_i, \tilde{u}_j{}') \le \mathrm{dis}(\tilde{u}_i, \tilde{u}_j)$;
- At least k-1 examples $\tilde{u}_j{}' \in W\backslash\{p\}$ exist, which satisfy $\mathrm{dis}(\tilde{u}_i, \tilde{u}_j{}') < dis(\tilde{u}_i, \tilde{u}_j)$.

Definition 3 Reachable distance of example In the possible world W, the reachable distance of $\tilde{u}_j$ relative to example $\tilde{u}_i$ is

$$rdis(\tilde{u}_i) = \max(k - dis(\tilde{u}_i), dis(\tilde{u}_j, \tilde{u}_i)) \qquad (5)$$

Definition 3 k neighbor distance sum of example In the possible world W, $\mathrm{n}_\mathrm{k}^\mathrm{W}(\tilde{u}_i)$ represents the k neighbor set of example $\tilde{u}_i$ in the possible world W, and $\mathrm{dis}_{\mathrm{k(W)}}(\tilde{u}_i)$ represents the k neighbor distance sum of example $\tilde{u}_i$ in the possible world:

$$dis_{k(W)}(\tilde{u}_i) = \sum_{\tilde{u}_j \in n_k^W(\tilde{u}_i)} rdis(\tilde{u}_i) \qquad (6)$$

Definition 4 Local reachable density of examples in the possible world In the possible world W, the local reachable density of example $\tilde{u}_i$ is the reciprocal of k neighbor distance sum of example $\tilde{u}_i$ and its $\mathrm{n}_\mathrm{k}^\mathrm{W}(\tilde{u}_i)$, i.e.

$$lrd_{k(W)}(\tilde{u}_i) = \frac{|n_k^w(\tilde{u}_i)|}{dis_{k(W)}(\tilde{u}_i)} \qquad (7)$$

Definition 5 Local outlier factor of example Provided that example $\tilde{u}_j$ is k of example $\tilde{u}_i$, and $\mathrm{lrd}_{\mathrm{k(W)}}(\tilde{u}_i)$ and $\mathrm{lrd}_{\mathrm{k(W)}}(\tilde{u}_j)$ respectively represent the local reachable density of example $\tilde{u}_i$ and example $\tilde{u}_j$ in the possible world W, the local outlier factor of example is shown as follows:

$$LOF(\tilde{u}_i) = \frac{\sum_{\tilde{u}_j \in n_k^W(\tilde{u}_i)} lrd_{k(W)}(\tilde{u}_j) P(W)}{|n_k^W(\tilde{u}_i)| \times lrd_{k(W)}(\tilde{u}_i)} \qquad (8)$$

Definition 6 Local outlier factor of uncertain object ui The local outlier factor of uncertain object ui represents the abnormal degree of object, and the larger the outlier factor is, the higher the possibility that the object is abnormal is, and on the contrary, the lower it is, i.e.

$$LOF(u_i) = \sum_{\tilde{u}_i \in u_i} LOF(\tilde{u}_i) P(\tilde{u}_i) \qquad (9)$$

Definition 7 Outlier of uncertain data Arrange the objects of uncertain data according to LOF values in the descending order, the first n objects are outliers of uncertain data.

3. Outlier Detection Algorithm of Uncertain Data

The complexity of iForest algorithm is low, but this algorithm is sensitive to global sparsity only, and it is not good at treatment of local relative sparse points. Though the LOF algorithm has a high discernibility quality, its time complexity is high. In view of the features of the said algorithms, the two algorithms are improved in this article, and they are applied to uncertain data, and the improved iForest is taken as a filter, and the filtered data set is regarded as the input of the next algorithm; finally, the improved LOF algorithm is used to obtain more accurate outliers. The flow chart of this detection algorithm is roughly shown in Figure 2.

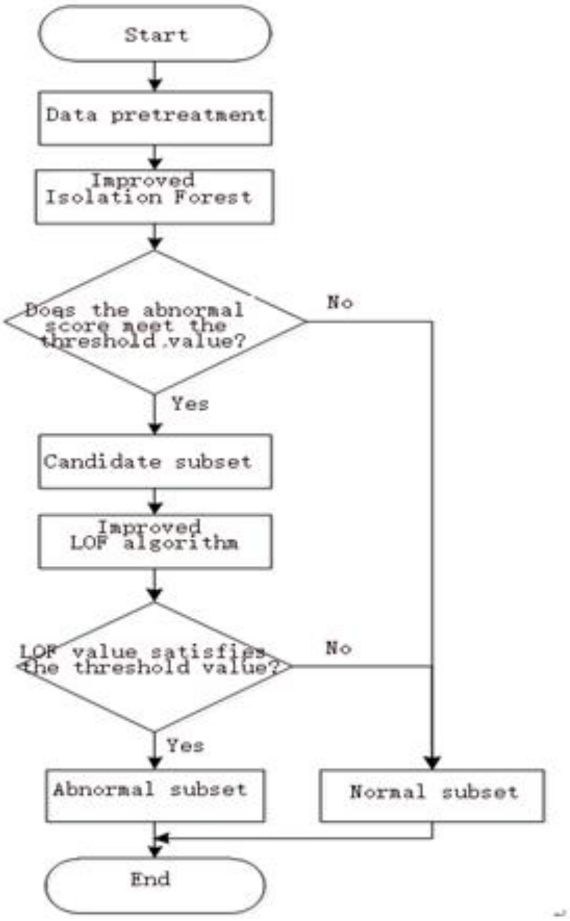
Start
Data pretreatment
Improved Isolation Forest
Does the abnormal score meet the threshold value?
No
Yes
Candidate subset
Improved LOF algorithm
LOF value satisfies the threshold value?
No
Yes
Abnormal subset
Normal subset
End

Figure2. *Flow Chart of Algorithm*

3.1. Improved iForest Algorithm

The iForest judges whether an object is abnormal through the abnormal score of each data object among deterministic data. Among attribute-level uncertain data, each data object is composed of different examples, and the probability sum of examples of each object is 1. In this article, the example of object is regarded as input, and the abnormal score of examples is obtained through Formula (2), and the definition of abnormal score of object is as follows:

$$s(x,n)=\sum 2^{-\frac{E(h_{\tilde{u}_i}(x))}{c_{\tilde{u}_i}(n)}} P(\tilde{u}_i) \qquad (10)$$

Where, $P(\tilde{u}_i)$ represents the probability of occurrence of this example. To calculate the abnormal score of uncertain data object, it is essential to construct an iTree first, and the algorithm for construction of iTree is described as follows:

Input: Uncertain data set X, current tree height e, height limit 1

Output: iTree

- Set Q as attribute set of data set X, randomly select an attribute q there from, and select a separated value p from the maximum value and the minimum value of the attribute.

- Classify all examples in the data set, and put the examples with the attribute value q<p at the left child node, and the examples with q≥p at the right child node.
- Repeat Steps 1) and 2), till e≥l or |X|≤1.

Then, the same method is employed, and a sub-data set is randomly sampled, and the said algorithm is repeated to obtain an iForest.

The random sampling conducted every time when an iTree is constructed is for the purpose of guaranteeing the difference of different trees.

After the construction of iForest, work out the abnormal score s of object by use of Formula (10), and when s is larger than a value, put the corresponding object in the candidate subset, and when s is less than a value, put the corresponding object in the normal subset.

3.2. Improved LOF Algorithm

The candidate subset obtained in the previous section is used as the input data of this algorithm, and the LOF algorithm is used for working out the LOF value of object of candidate subset, and the outlier degree is determined through judging whether LOF is approximate to 1. If LOF is far larger than 1, it is considered as outlier; on the contrary, if it is close to 1, it is considered as normal point.

The calculation of LOF value can be obtained according to definition, but the number of possible worlds may show exponential growth with the increase of the number of examples, and it is hard to accept the calculation cost. Thus, each data example has k neighbor query in this article, and k neighbor sets and corresponding probabilities are obtained, making it unnecessary to expand all possible worlds and greatly increasing the efficiency. The probability of k neighbor is defined as follows:

It is known that two examples $\widetilde{u}_i$ and $\widetilde{u}_j$ are different data objects, and example $\widetilde{u}_j$ is the k neighbor of $\widetilde{u}_i$; $N_{\widetilde{u}_j \to \widetilde{u}_i}$ represents that example $\widetilde{u}_j$ is the set of possible worlds where k neighbor of example $\widetilde{u}_i$; $P_k(\widetilde{u}_j)$ represents the probability that $\widetilde{u}_j$ is k neighbor of $\widetilde{u}_i$, then

$$P_k(\tilde{u}_j) = \sum\nolimits_{W \in N_{\tilde{u}_j \to \tilde{u}_i}} P(W) \qquad （11）$$

Only when there are k-1 examples before $\widetilde{u}_j$ can $\widetilde{u}_j$ be the kth neighbor of $\widetilde{u}_i$, and $P(\widetilde{u}_j, k)$ represents the probability that $\widetilde{u}_j$ is the kth neighbor:

$$P(\tilde{u}_j, k) = P(\tilde{u}_j) P(S_{t_{i-1}}, k-1) \qquad （12）$$

Then

$$P_k(\tilde{u}_j) = \sum_{k=1}^{k} P(t_i, k) \qquad （13）$$

Then, local outlier factor of example of uncertain data object:

$$LOF(\tilde{u}_i) = \frac{\sum\nolimits_{\tilde{u}_j \in N_k(\tilde{u}_i)} lrd_k(\tilde{u}_j) P_k(\tilde{u}_j)}{\left| n_k^W(\tilde{u}_i) \right| \times lrd_k(\tilde{u}_i)} \qquad （14）$$

The changed LOF algorithm is described as follows:

Input: Candidate subset D1, the number of neighbors k, the number of abnormal points n

Output: Abnormal point O

For each data example, carry out k neighbor query and obtain k neighbor

- The local reachable density of examples can be worked out through Formula (7);
- The local outlier factor of examples can be worked out through Formula (8);
- The local outlier factor of each data object can be worked out through Formula (14);
- Conduct descending ordering of local outlier factors obtained through calculation, and the first n are abnormal points.

4. Experiment and Results

4.1. Experiment Setting

The computer processor for experiment and detection is Intel(R) Core(TM) i3-3200 3.3Ghz; memory: 4GB; operating system: Windows7. The algorithm is realized under the Matlab R2016a environment. The test data adopts two real data sets and one synthesized data set. In order to verify the effectiveness of algorithm in this article, this article takes the PUDOL of Reference [20] and the ULOF algorithm of Reference [21] to make a comparison. The real data set adopts the LDPA data set and the MAGIC data set used in the two references. The synthesized data set adopts Gaussian distribution, consisting of 100000 objects, forming three different types, where the density of each type is different; then 100 objects not belonging to those three types are generated as abnormal points. In order to simulate uncertain data, each data object is expressed by several examples, and each object contains the example range of [1,5], and the sum of probabilities of examples belonging to the same data base is 1.

4.2. Analysis of Experimental Results

Efficiency Test

Figure 3 shows the relationship between the running time of algorithm and the number of uncertain data objects, and the range of data size is [0, 100]. It can be observed from the figure that data size has an obvious impact on the time of algorithm. With the increase of data size, the running becomes longer and longer, but no matter how the data size changes, the running time of the algorithm mentioned in this article is shorter than that of other algorithms. The reason is that the more the data is, the more data to be processed by the algorithms becomes, causing the running time of algorithms to be longer. However, this article employs the Isolation Forest algorithm to quickly filter out some obvious normal points in the initial stage, reducing the data size of subsequent LOF calculation, shortening the calculation time and improving the efficiency.

Figure 4 represents the relationship between the number of abnormal points and running time, and n represents the number of abnormal points in data set. It can be observed from the figure that with the increase of abnormal objects, the running time of algorithms increases, but the running time of the algorithms in this article is shorter than that of other algorithms. The reason is that in the algorithm PUDLO, the larger n value is, the more the examples of abnormal objects become, causing the pruning rate of PUDLO algorithm to reduce, making the size of data for which it is necessary to accurately work out the probability of outliers be large, and consequently making the running time of algorithm become longer. In the algorithm of this article, if n value increases, it indicates that there are more data objects selected in the earlier stage, and the pruning rate of the later stage will decline, which increases the calculation amount to some extent.

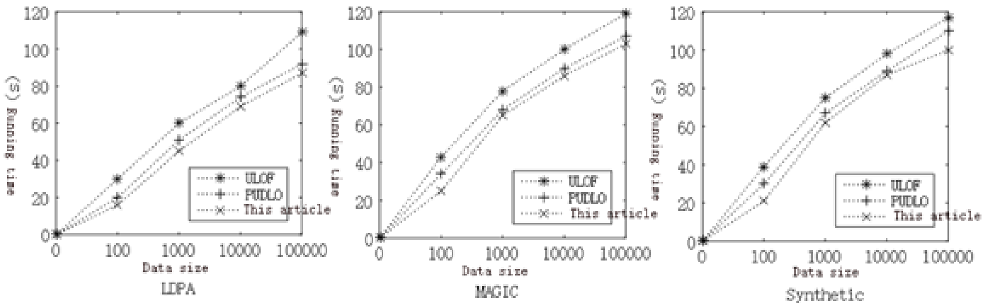

Figure3. *Running Time vs Data Size*

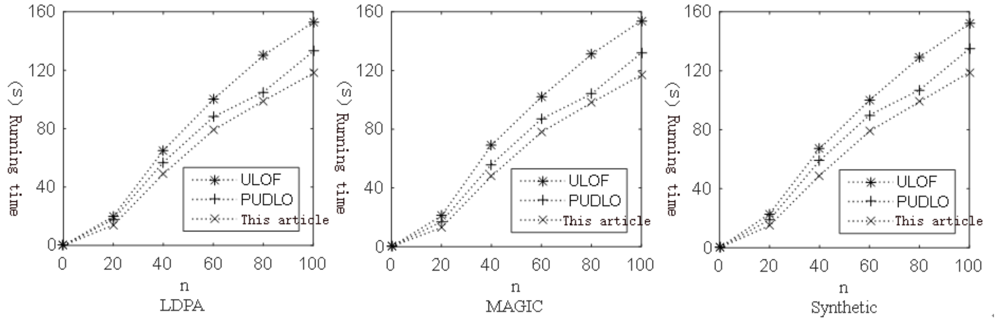

Figure4. *Running Time vs n*

Accuracy Test

Figure 5 shows the relationship between algorithm accuracy and parameter k. Figure 6 shows the relationship between algorithm accuracy and the number of abnormal objects n. It can be observed from the two figures that the accuracy declines with the increase of k and n, but no matter how k and n change, the accuracy rate of algorithm of this article is higher than that of the other two algorithms. In the algorithm ULOF, since it is hard to find abnormal points on the margin with this algorithm, the accuracy of algorithm is seriously affected. In the algorithm PUDLO, it carries out query and pruning in strict accordance with the definition of local abnormal points, and the algorithm accuracy is less affected. The algorithm raised in this article also conducts calculation and pruning in strict accordance with the definition, so the algorithm accuracy declines a little.

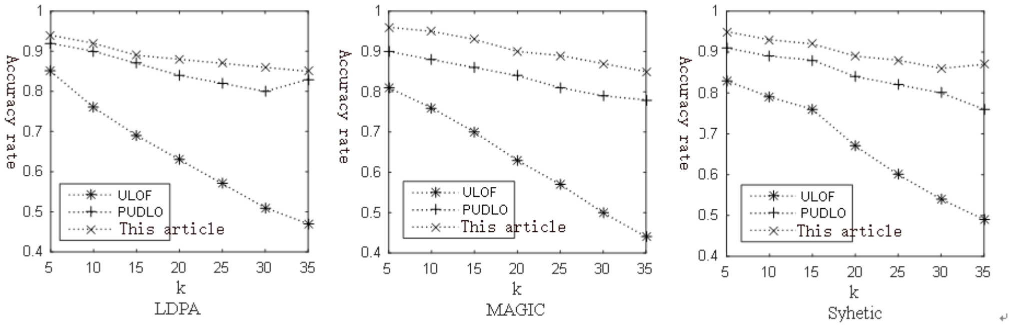

Figure5. *Accuracy Rate vs k*

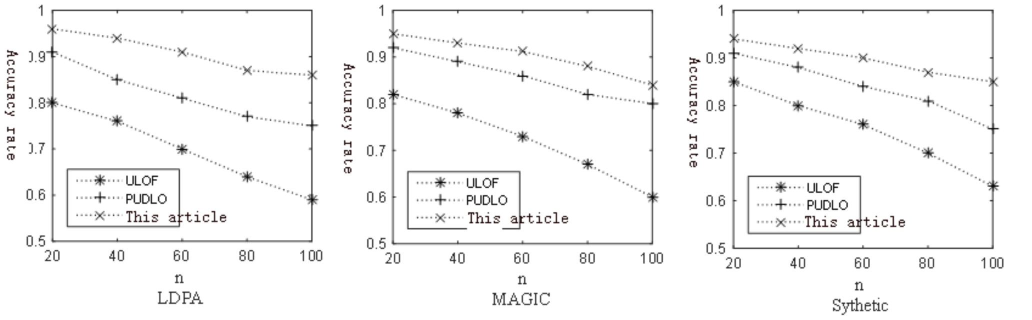

Figure6. *Accuracy Rate vs n*

5. Conclusion

This article puts forward an outlier detection method based on iForest and LOF. It re-defines local outlier factor 1 of uncertain data, and then, optimizes and reduces the candidate set of data through iForest and K neighbor query, and verifies the feasibility and efficiency of algorithm through experiment.

References

[1] Cao L, Wei M, Yang D, et al. Online Outlier Exploration Over Large Datasets[C]// ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. ACM, 2015:89-98.

[2] Zhang J. Advancements of Outlier Detection: A Survey[J].ICST Transactions on Scalable Information Systems, 2013,13(1):1-26.

[3] Chandola V, Kumar V. Outlier Detection : A Survey. Minnesota, USA: Department of Computer Science and Engineering, University of Minnesota, Technical Report: TR .07-17, 2007

[4] Knorr E M, Ng R T. Algorithms for Mining Distance-Based Outliers in Large Datasets[C]// International Conference on Very Large Data Bases. Morgan Kaufmann Publishers Inc. 1998:392-403.

[5] Horn P S, Feng L, Li Y, et al. Effect of Outliers and Nonhealthy Individuals on Reference Interval Estimation[J]. Clinical Chemistry, 2001, 47(12):2137-2145.

[6] Solberg H E, Lahti A. Detection of outliers in reference distributions: performance of Horn's algorithm.[J]. Clinical Chemistry, 2005, 51(12):2326-2332.

[7] Breunig M M. LOF: identifying density-based local outliers[C]// ACM SIGMOD International Conference on Management of Data. ACM, 2000:93-104.

[8] Yang Jinwei. Research of detection of uncertain abnormal point based on distance and information entropy [D]. Yunnan University, 2011.

[9] Hido S, Kashima H, Sugiyama M, et al. Statistical outlier detection using direct density ratio estimation[J]. Knowledge & Information Systems, 2011, 6(2):309-336.

[10] Zhang Yu, Zhang Yansong, Chen Hong, Wang Shan. A mixed OLAP query processing model adapting to GPU [J]. Journal of Software, 2016,27(05):1246-1265.

[11] Aggarwal C C, Yu P S. Outlier Detection with Uncertain Data[C]// Siam International Conference on Data Mining, SDM 2008, April 24-26, 2008, Atlanta, Georgia, Usa. 2008:483-493.

[12] Wang B, Xiao G, Yu H, et al. Distance-Based Outlier Detection on Uncertain Data[C]// IEEE International Conference on Computer & Information Technology. IEEE, 2009:293-298.

[13] Hong Sha, Lin Jiali, Zhang Yueliang. Research of detection of density-based uncertain data outliers [J]. Computer Science, 2015,42(05):230-233.

[14] Shaikh S A, Kitagawa H. Distance-Based Outlier Detection on Uncertain Data of Gaussian Distribution[J]. World Wide Web-internet & Web Information Systems, 2012, 17(4):511-538.

[15] Shaikh S A, Kitagawa H. Fast Top-k Distance-Based Outlier Detection on Uncertain Data[C]// International Conference on Web-Age Information Management. Springer, Berlin, Heidelberg, 2013:301-313.

[16] Shaikh S A, Kitagawa H. Top-k Outlier Detection from Uncertain Data[J]. International Journal of Automation and Computing, 2014, 11(2):128-142.

[17] Liu F T, Kai M T, Zhou Z H. Isolation Forest[C]// Eighth IEEE International Conference on Data Mining. IEEE, 2009:413-422.

[18] Liu F T, Ting K M, Zhou Z H. Isolation-Based Anomaly Detection[J]. Acm Transactions on Knowledge Discovery from Data, 2012, 6(1):1-39.

[19] B. R. Preiss. Data Structures and Algorithms with Object-Oriented Design Patterns in Java. Wiley, 1999.

[20] Liu J, Deng H F. Outlier detection on uncertain data based on local information[J]. Knowledge-Based Systems, 2013, 51(1):60-71

[21] Cao K, Shi L, Wang G, et al. Density-Based Local Outlier Detection on Uncertain Data[C]// International Conference on Web-Age Information Management. 2014:67-71.

Citation: *AMADY BA , Zhenfang ZHU (2020). "Uncertain Data Outlier Detection Algorithm based on iForest and LOF", International Journal of Research Studies in Computer Science and Engineering (IJRSCSE), 7(1), pp.38-46. DOI:http://dx.doi.org/10.20431/2349-4859.0701005*

Copyright: *© 2020 Authors, This is an open-access article distributed under the terms of the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.*