Assessing Data Anomaly Detection Algorithms in Power Internet of Things

Zixiang Wang[1], Zhoubin Liu[1], Xiaolu Yuan[2], Yueshen Xu[3], and Rui Li[3](✉)

[1] State Grid Zhejiang Electric Power Research Institute, Hangzhou 310000, Zhejiang, China
{wangzixiang, liuzhoubin}@zj.sgcc.com.cn
[2] RUN Corporation, Wuxi 214000, Jiangsu, China
rubin0513@gmail.com
[3] Xidian University, Xi'an 710071, Shannxi, China
{ysxu, rli}@xidian.edu.cn

Abstract. At present, the data related to the Internet of Things has shown explosive growth, and the importance of data has been greatly improved. Data collection and analysis are becoming more and more valuable. However, a large number of abnormal data will bring great trouble to our research, and even lead people into misunderstandings. Therefore, anomaly detection is particularly necessary and important. The purpose of this paper is to find an efficient and accurate outlier detection algorithm. Our work also analyzes their advantages and disadvantages theoretically. At the same time, the effects of the data set size, number of proximity points, and data dimension on CPU time and precision are discussed. The performance, advantages and disadvantages of each algorithm in dealing with high-dimensional data are compared and analyzed. Finally, the algorithm is used to analyze the actual anomaly data collected from the Internet of Things and analyze the results. The results show that the LOF algorithm can find the abnormal data in the data set in a shorter time and with higher accuracy.

Keywords: Anomaly detection · Internet of Things · LOF

1 Introduction

The application of the Internet of Things technology in the power industry is the result of the development of information and communication technology to a certain stage. The Internet of Things integrates communication, information, sensing, automation and other technologies. It deploys a wide range of intelligence with certain perception, computing and execution capabilities in all aspects of power production, transmission, consumption and management. It can sense equipment, adopt standard protocol based on IP, realize reliable transmission of information security, cooperative processing, unified service and application integration through power information communication network, thus realize holographic perception, interconnection and seamless integration in the whole process of power grid operation and enterprise management. The construction of power Internet of Things can effectively integrate communication infrastructure resources and power system infrastructure resources, improve the level of

H. Gao et al. (Eds.): CollaborateCom 2018, LNICST 268, pp. 403–416, 2019.
https://doi.org/10.1007/978-3-030-12981-1_28

power system information, improve the efficiency of existing power system infrastructure, and provide important technical support for power grid generation, transmission, transformation, distribution and power consumption.

2 Angle-Based Anomaly Detection (ABOD)

At present, there are many kinds of abnormal data detection algorithms, and each method faces some problems which are suitable for different scenarios. For example, anomaly detection algorithms based on statistics generally need to know the model of a given data set, the distribution parameters and the expected number of anomaly data objects. However, these parameters are very difficult to obtain [1–3]. Distance-based abnormal data detection algorithms [4–6] have good effects on the detection of high-dimensional data, but the relevant parameters must be set in advance, and the setting of parameters is related to the detection results of the entire abnormal data object, and the detection of abnormal data based on distance is only to detect the global data. When detecting abnormal data objects, the whole data set must be scanned frequently, and it is a difficult problem for the rapid mining of data flow. Density-based anomaly data detection algorithms [7, 8] are for local anomaly data object detection, and they have great dependence on the nearest, index data structure and other methods, high computational complexity.

In view of the above situation, Kriege et al. [9] proposed an outlier detection algorithm based on angle to mine outlier data objects in high-dimensional data sets. At the same time, Pham and others proposed a new outlier detection algorithm based on angle analysis [10]. The basic principle is to compare the angle variance between each data object and other data objects in the hyperplane. The smaller the variance, the farther away from the center point. In the high-dimensional massive data space, the angle is more stable than the distance calculation, moreover, the method based on the angle distribution will not deteriorate substantially [11].

According to the method based on angle distribution proposed by Kriegel and others to calculate the anomalies of each data object, the distribution of a 2-dimensional cube on the plane is shown in Fig. 1 above. Point P is a normal point (see Fig. 1(a)). Since all the other points are distributed in all directions around it, the angles of point P and any point are not uniform, so these angles fluctuate more, that is, the angular variance of point P is larger. Therefore, for other points, the greater the angular variance of a point, the greater the likelihood that the point will be normal.

If point P is outside the cluster (see Fig. 1(b)), each angle is made up of point P and arbitrary points. Because all the other points are in a specific direction of P, the size of these angles is very close. Moreover, the angular fluctuation is also small, that is, the angle variance of point P is small. For other points, if the angular variance of a point is smaller, then this point is more likely to be an outlier.

As is known above, when the variance of the angle is between the two, it can be seen as a boundary point (see Fig. 1(c)). Therefore, we can use the variance of the angle to obtain the abnormality of each point, to distinguish normal points, outliers and boundary points.

Based on the above idea, Pham and others put forward the concept of anomaly factor based on angular distribution. The specific definitions are as follows:

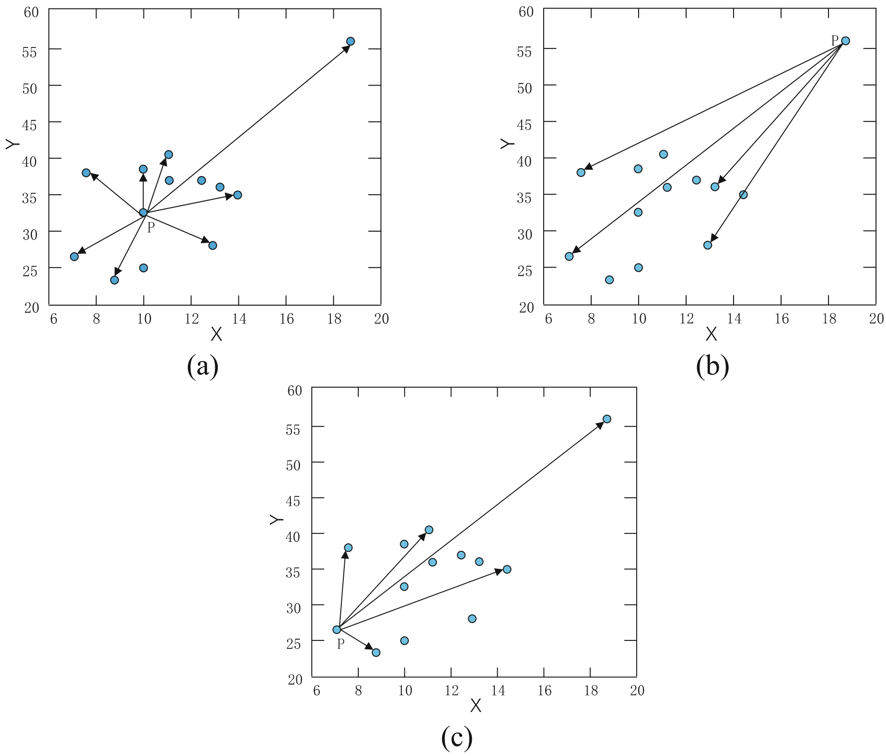


Fig. 1. Distribution of the data set

Given a data set, and a sample point, randomly select a sample point, and have different vectors and angles between them, then all variance angular distribution anomaly factor VOA (p), i.e.,

$$\begin{aligned} VOA(\mathrm{p}) &= VOA[\Theta_{apb}] \\ &= \mathrm{MOA}_2(\mathrm{P}) - (\mathrm{MOA}_1(\mathrm{P}))^2 \end{aligned} \tag{1}$$

$$MOA_1(\mathrm{p}) = 2\frac{\sum\limits_{a,b \in S \setminus \{\mathrm{p}\}, \mathrm{a} \neq \mathrm{b}} \Theta_{apb}}{(\mathrm{n}-1)(\mathrm{n}-2)} \tag{2}$$

$$MOA_2(\mathrm{p}) = 2\frac{\sum\limits_{a,b \in S \setminus \{\mathrm{p}\}, \mathrm{a} \neq \mathrm{b}} \Theta_{apb}^2}{(\mathrm{n}-1)(\mathrm{n}-2)} \tag{3}$$

In the upper form, they are the 1-order matrix and the 2-order matrix of point P respectively. Therefore, VOA has no parameters. Therefore, this method is suitable for unsupervised anomaly data monitoring algorithm [12, 13]. ABOD algorithm prototype algorithm calculates VOA of each data point, and returns the minimum *m* points in VOA as the anomaly data points to be mined. At the same time, the time complexity of the prototype algorithm is illustrated, where D is the dimension of the dataset, and N represents the number of data sets. Because its time complexity is cubic time complexity, it will be very difficult to mine high-dimensional massive abnormal data [14, 15].

3 An Approximate Algorithm Based on ABOD

Because the time complexity of ABOD is cubic, in order to avoid such a high time complexity, we propose a near-linear time complexity algorithm to estimate the angle variance of each data point.

(1) We first approximate estimate the first order matrix.

$$
\begin{aligned}
F_1(\mathrm{p}) &= \frac{2}{(\mathrm{n}-1)(\mathrm{n}-2)}\left(2\pi \sum_{\substack{a,b \in S\setminus\{\mathrm{p}\} \\ \mathrm{a}\neq\mathrm{b}}} E\left[X_{apb}^{(\mathrm{i})}\right]\right) \\
&= \frac{2\pi}{(\mathrm{n}-1)(\mathrm{n}-2)} \sum_{\substack{a,b \in S\setminus\{\mathrm{p}\} \\ \mathrm{a}\neq\mathrm{b}}} \left(E\left[X_{apb}^{(\mathrm{i})}\right] + E\left[X_{bpa}^{(\mathrm{i})}\right]\right) \\
&= \frac{2\pi}{(\mathrm{n}-1)(\mathrm{n}-2)} \left|L_p^{(\mathrm{i})}\right|\left|R_p^{(\mathrm{i})}\right| \qquad (4)
\end{aligned}
$$

The set $L_p^{(i)} = \{x \in S\setminus\{p\} | x \cdot r_i < p \cdot r_i\}$ and set $R_p^{(i)} = \{x \in S\setminus\{p\} | x \cdot r_i \rangle p \cdot r_i\}$ are composed of points on both sides of the P point.

(2) We further estimate the two-order matrix.

$$
\begin{aligned}
MOA_2(\mathrm{p}) &= \frac{2}{(\mathrm{n}-1)(\mathrm{n}-2)} \sum_{\substack{a,b \in S\setminus\{\mathrm{p}\} \\ \mathrm{a}\neq\mathrm{b}}} \Theta_{apb}^2 \\
&= \frac{2}{(\mathrm{n}-1)(\mathrm{n}-2)} \sum_{\substack{a,b \in S\setminus\{\mathrm{p}\} \\ \mathrm{a}\neq\mathrm{b}}} \left(\Theta_{apb}^2 + \Theta_{bpa}^2\right) \\
&= \frac{4\pi^2}{t(\mathrm{t}-1)(\mathrm{n}-1)(\mathrm{n}-2)} \left(\sum_{i=1}^{n-1}\sum_{j=1}^{n-1} E\left[P_{ij}^2\right] - \frac{t}{\pi} \sum_{\substack{a,b \in S\setminus\{\mathrm{p}\} \\ \mathrm{a}\neq\mathrm{b}}} \Theta_{apb}\right) \\
&= \frac{4\pi^2}{t(\mathrm{t}-1)(\mathrm{n}-1)(\mathrm{n}-2)} \left(E\left[\|P\|_F^2\right] - \frac{t(\mathrm{n}-1)(\mathrm{n}-2)}{2\pi} MOA_1(\mathrm{P})\right) \\
&= \frac{4\pi^2}{t(\mathrm{t}-1)(\mathrm{n}-1)(\mathrm{n}-2)} E\left[\|P\|_F^2\right] - \frac{2\pi}{t-1} MOA_1(\mathrm{p}) \qquad (5)
\end{aligned}
$$

Based on the above formula, we can estimate *$MOA_2(p)$*:

$$F_2^{'}(\mathrm{P}) = \frac{4\pi^2}{t(\mathrm{t}-1)(\mathrm{n}-1)(\mathrm{n}-2)}\|p\|_F^2 - \frac{2\pi}{t-1}F_1(\mathrm{p}) \tag{6}$$

The time complexity of FAST-ABOD has been greatly improved for ABOD.

4 An Improved Algorithm Based on Filter-Refinement

From above section, we can see that FAST-ABOD is more sensitive to the dimension of data. What's more, we find that abnormal data from ABOD is always those data points with the highest ranking, and ABOD always has a lower bound [16]. Therefore, based on the above analysis, we can select the lower bound of the angle variance anomaly factor from L candidate outliers and correct it until no point in the candidate list has an angle variance anomaly factor smaller than the corrected one.

Therefore, we get a more accurate *$MOA_1(p)$* unbiased estimator:

$$\mathrm{F}_1(\mathrm{p}) = \frac{2\pi}{t(\mathrm{n}-1)(\mathrm{n}-2)}\sum_{i=1}^{t}\left|L_p^{(\mathrm{i})}\right|\left|R_p^{(\mathrm{i})}\right| \tag{7}$$

The last two order matrix $\mathrm{F}_2(\mathrm{p})$ is estimated to be:

$$F_2(\mathrm{p}) = \frac{4\pi^2\left(\sum_{i=1}^{t} AMS\left(\mathrm{L}_p^{(\mathrm{i})}\right)\mathrm{AMS}\left(R_\mathrm{p}^{(\mathrm{i})}\right)\right)}{t(\mathrm{t}-1)(\mathrm{n}-1)(\mathrm{n}-2)} - \frac{2\pi F_1(\mathrm{p})}{t-1} \tag{8}$$

The vectors $AMS(L_p^{(i)})$ and vectors $AMS(R_p^{(i)})$ are estimated by product domain *AMS* Sketch. And there are:

$$\|P\|_F^2 = \left(\sum_{i=1}^{t} AMS\left(L_p^{(\mathrm{i})}\right)AMS\left(R_p^{(\mathrm{i})}\right)\right)^2 \tag{9}$$

Therefore, for the first L outlier data points, we propose the following methods to find the first L outlier data objects:

1. (Filtering Process) For each point in the region D, find *k* points (e.g. *k* nearest points) that have the greatest impact on it.
2. Calculate the LB-ABOF value of each point.
3. The LB-ABOF of each point calculated by 2 will be arranged in ascending order and coexist in the candidate column.
4. (Correction Process) Calculate the true ABOF of the first L objects in the candidate column, delete them from the candidate column and insert them into the result column.

5. Calculate the ABOF value of the next object in the candidate column and delete it from the candidate column. If the ABOF value of the next object is smaller than the maximum ABOF value in the result column, interchange the object corresponding to the maximum ABOF value in the result column, remove the point from the result column, and insert the next object into the result column.
6. If the largest ABOF value in the result column is smaller than the smallest approximate ABOF value in the candidate column (that is, LB-ABOF), the algorithm terminates; otherwise, step 5 is executed.

The LB-ABOD algorithm combines the scalability of Fast ABOD on data scale and the robustness of ABOD in dimensionality. The time complexity of the filtering process is (same as FAST-ABOD), and the time complexity of the correction process is, where n is the number of corrected data points. Therefore, the acceleration effect of ABOD depends on the value of the lower bound and the number of final corrected object points n. In practice, the running time of LB-ABOF is very unstable, which is closely related to the number of neighboring points (kNN).

5 Performance Analysis

5.1 The Influence of the Number of Adjacent Points (KNN)

All the above algorithms are implemented in C# language on Visual Studio 2016 development platform, and all the experiments are implemented on a PC running Windows 10 64bit operating system.

Due to the unsupervised nature of the actual collected data sets of power sensor networks, we cannot determine whether the abnormal data detected by the above algorithms are abnormal data in the real sense. In order to compare the performance and accuracy of the above detection algorithm more comprehensively, we randomly generated a number of different dimensions of the data set. Generation rules are: using time as random number seed to generate dimension D data sets, by controlling the range of random numbers to generate outliers and normal points. There is one outlier per 50 points in the generated dataset (here we call each table item of the dataset a “point”).

This experiment will test the performance of the algorithm based on the above dataset. Therefore, we will use the precision and recall to evaluate the performance of the algorithm.

The precision ratio (precision) is an index to measure the signal-to-noise ratio (SNR) of a retrieval system, that is, the percentage of the relevant literature detected and the total literature detected. It is generally expressed as: precision ratio = (retrieves the total amount of information/information retrieved) $\times$ 100%. The retrieval language with strong generality (such as upper class and upper subject word) can improve the recall rate, but the precision rate decreases.

Recall rate (recall rate) is a measure of a retrieval system from the collection of relevant documents to detect the success of an indicator, that is, the number of relevant documents detected in the retrieval system and the total amount of relevant literature ratio. It is generally expressed as: recall ratio = (the total amount of relevant information retrieved/the relevant information in the system) $\times$ 100%. The retrieval language with strong generality (such as upper class and upper subject word) can improve the recall rate, but the precision rate decreases.

We hope that the higher the precision, the better the recall. But in fact, this is not the case. In some extreme case, we only detected an abnormal result, and it was accurate. Well, we can say that the precision rate is 100%, but the recall rate is very low. Conversely, when we return all the data, our recall rate is very high, but the precision rate is very low.

As a result, when we want both of them to be very high, we use F1 (F-score) to measure the equilibrium point, where $P(Precision) = \frac{A}{A+B}$, $R(Recall) = \frac{A}{A+C}$, and $F1 = \frac{2PR}{P+R}$.

In order to study the effect of the number of neighboring points on the performance of each algorithm, we tested 1000 data objects with 50 dimensions. According to the simulation test data set generation rules, there are 20 outliers.

For Fast ABOD and LB-ABOD, the theory holds that the former top K object is an outlier. For LOF algorithm, the theory holds that when the local reachable density of P is much smaller than that of its neighbors, P is an outlier. Because of the different data sets, it is difficult to have a unified definition of k, that is, it is difficult to determine a unified value, and the size of K has a greater impact on the detection results. For convenience of comparison, we calculate the LOF values of each point and sort them. The top K points with the largest LOF values are treated as outliers. Top K must be specified here. Because we know in advance which points are outliers, we set the number of outliers to be detected in the program (Top K) equal to the number of real outliers in the data set. Therefore, in this section, for FAST-ABOD, LB-ABOD, and LOF, the precision equals the recall equals its harmonic mean (precision = recall = F-Score).

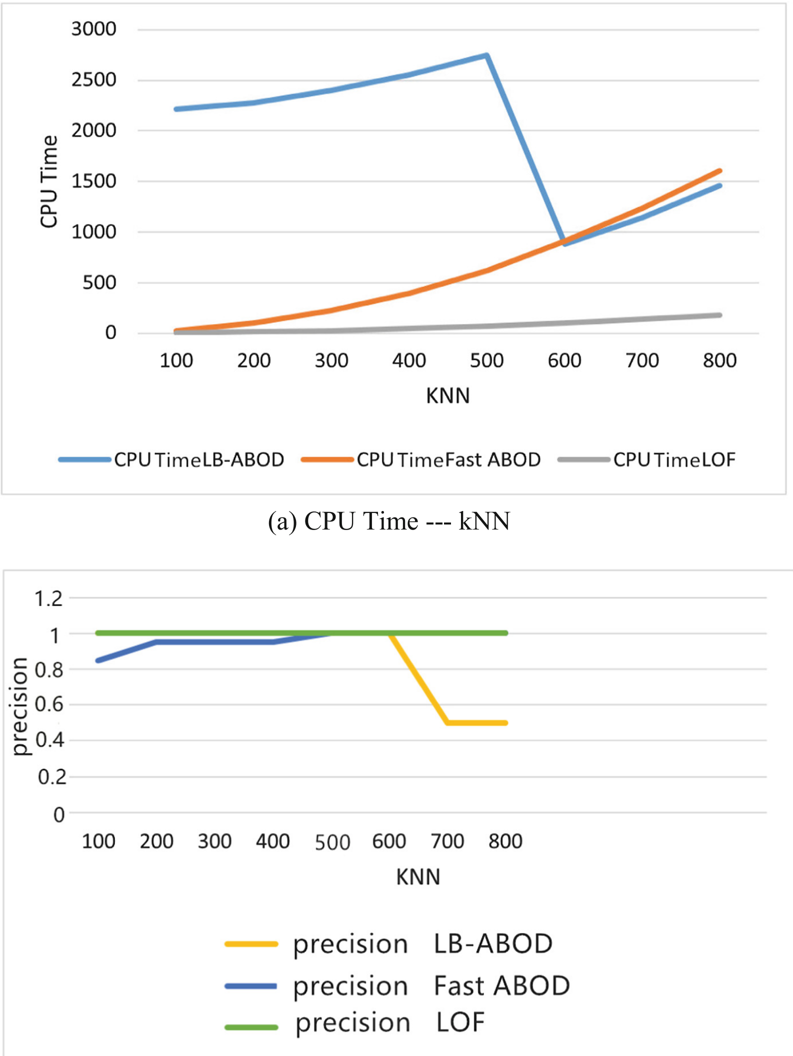
Table 1. Relationship between CPU time and kNN (*N = 1000, D = 50, Top K = 20*)

(kNN)	CPU time (s)			Precision		
	LB-ABOD	Fast ABOD	LOF	LB-ABOD	Fast ABOD	LOF
100	2217.6	28.45	5.53	1	0.85	1
200	2280.5	101.22	13.26	1	0.95	1
300	2396.3	224.63	26.51	1	0.95	1
400	2555.2	397.32	44.87	1	0.95	1
500	2751.1	618.87	68.7	1	1	1
600	882.5	910.93	103.19	1	1	1
700	1139.7	1236.38	137.79	0.5	1	1
800	1459.4	1609.21	177.8	0.5	1	1

From Table 1, we can see that the CPU time of LB-ABOD decreases with the increase of KNN, while the CPU time of Fast-ABOD and LOF both go to a larger direction with the increase of KNN. At the same time, for their precision LOF has always maintained an efficient precision of 100%, on the contrary, LB-ABOD, and FAST-ABOD are not so ideal precision.

Figure 2 shows the influence of the number of adjacent points kNN on the performance of the algorithm. As can be seen, for Fast-ABOD, the CPU time increases exponentially with the increase of kNN, while the growth of LOF is relatively flat (Fig. 2(a), (b)).

Compared with Fast ABOD, LB-ABOD maintained 100% precision at fewer proximity points, which corresponded to several times the CPU time of FAST-ABOD under the same conditions (Fig. 2(a)); at that time, LB-ABOD had a CPU time of



(a) CPU Time --- kNN

(b) Precision --- kNN

Fig. 2. Influence of kNN on algorithm performance ($N = 1000$, $D = 50$, $TopK = 20$)

2217.6 s, 78 times that of Fast ABOD (28.45 s). This is because when the kNN is relatively small, the LB-ABOF calculated by each object is negative, and there is no object in the candidate column whose LB-ABOF is greater than that of the object in the result column. At this point, the ABOF value of each object needs to be computed, and LB-ABOD degenerates into ABOD algorithm, so its running time is longer. When the kNN increases to 60% of the data set size, the run time of LB-ABOD drops sharply, and then the CPU time continues to increase with the increase of the kNN, and the CPU time of LB-ABOD is less than that of Fast ABOD under the same conditions (Fig. 2 (a)). This is because, when the LB-ABOF value of each object is positive, the correction process starts to work: the algorithm terminates when the largest ABOF in the result column is less than the smallest LB-ABOF in the candidate column. KNN continues to grow, and the LB-ABOF computing time of each object increases, so its CPU time continues to grow.

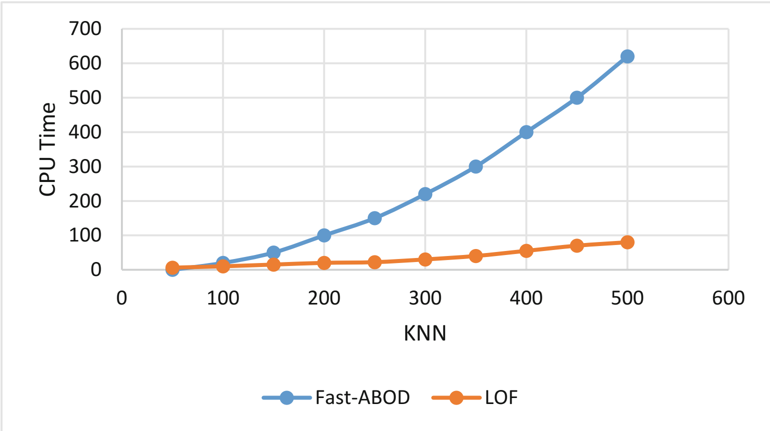
From the above discussion, the value of kNN is very important for LB-ABOD: kNN is too small, the operation efficiency of the algorithm is too low (ABOD); kNN is too large, while increasing the operation time will reduce the accuracy (Fig. 2(b)). Therefore, in practical applications, it is necessary to find a suitable balance between CPU time and accuracy, that is, the "balance point" of the minimum CPU time of LB-ABOD in Fig. 2(a). This needs to be analyzed according to the characteristics of different data sets. In this section, it is more appropriate for kNN to take about 60% of the dataset scale.

Figure 3 shows the relationship between algorithm accuracy and kNN, where Fig. 3(c) is a partial amplification of Fig. 3(b). Obviously, when the number of adjacent points is too small, it will seriously affect the accuracy of LOF; with the increase of the number of adjacent points, the precision of LOF increases gradually; when the number of adjacent points increases to a certain degree (in this case, kNN = N/40 = 25), the precision of LOF reaches 100%, and then it remains unchanged at this level.

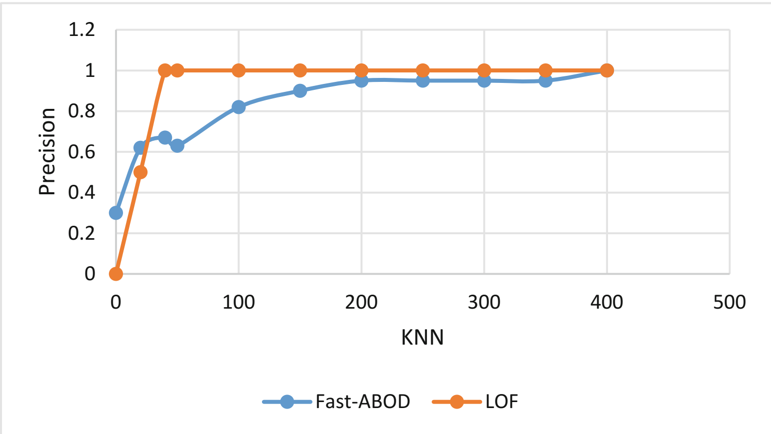
Compared with LOF, the precision of Fast ABOD fluctuates greatly. When the number of adjacent points is small, the precision increases with the increase of kNN. When kNN reaches a certain size (kNN = 50), the precision decreases; then, with the increase of kNN, the precision increases gradually (Fig. 3(b), (c)). The above situation is more evident in the higher dimensional data set (Fig. 3(d)).

There are many reasons for the instability of the above-mentioned Fast ABOD precision: when the number of adjacent points is small (kNN = N/200 = 5), the precision is low because of the small number of samples compared; when the number of adjacent points is large (kNN = N/2 = 500), the Fast ABOD approximates to the ABOD algorithm, so the precision is high.

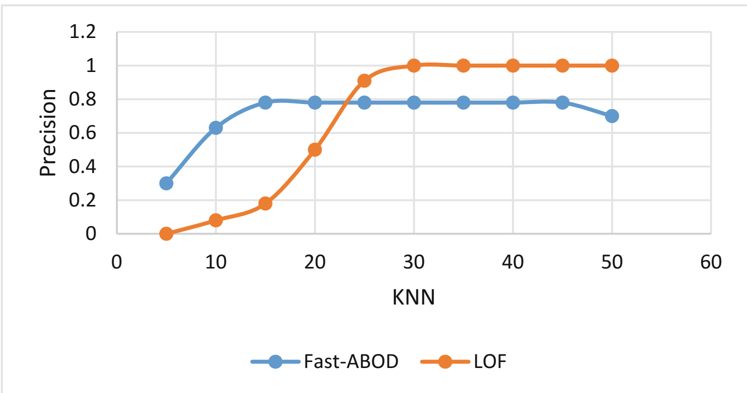
Considering the case shown in Fig. 4, it shows the effect of different proximity values on the results of Fast ABOD detection in a two-dimensional plane. When the number of kNN is small (kNN = 6), point P is an outlier relative to its proximity (Fig. 4(a)); however, as the number of proximities increases, the proximity of point P appears in all directions around it, and then P is no longer an outlier relative to its proximity (Fig. 4(b)). This also explains why the precision of Fast ABOD in Figs. 3, 4 and 5 increases first and then decreases as the number of near points (kNN) increases. It can be seen that another disadvantage of Fast ABOD is that it is sensitive to the number of adjacent points, and the algorithm is unstable.



(a) *CPU Time—kNN* (*N=1000, D=20*)

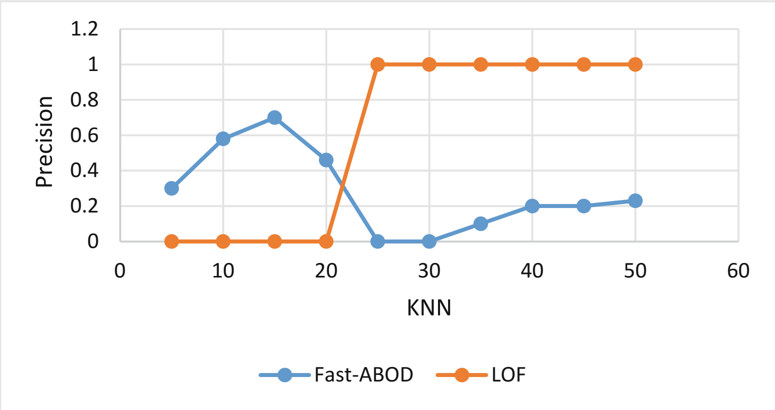


(b) *Precision—kNN (N=1000, D=20)*



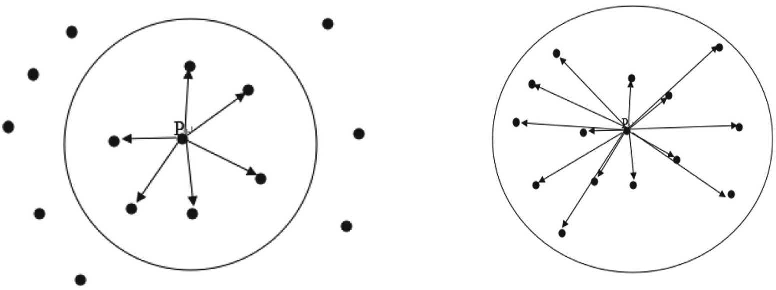
(c) *Precision—kNN (N=1000, D=20)*

Fig. 3. The relationship between kNN and Fast ABOD or LOF performance



(d) *Precision—kNN (N=1000, D=50)*

Fig. 3. (*continued*)



(a) kNN=6 (b) kNN=14

Fig. 4. The influence of the proximity points on the two-dimensional plane to the determination of outliers

5.2 Influence of Top K on Algorithm Accuracy

Let N = 1000, in which there are 20 outliers; data dimension D = 20; for LB-ABOD, take the number of near points kNN = 650; for Fast ABOD, LOF, kNN = 100, get the relationship between topK and algorithm precision, recall, F-Score as shown in Table 2.

Figure 5 is an intuitive display of Table 2. When topK is less than the number of outliers in the data set (topK < Outlier = 20), the precision of FAT ABOD and LOF is 100%, and the recall and F-Score are increased with the increase of topK. For LB-ABOD, the precision of F-Score is decreased with the increase of topK. When topK is greater than Outlier, for LOF, because it has found all the outliers, the increase of topK

Table 2. Influence of topK on algorithm accuracy (*N = 1000, D = 20, Outlier = 20*)

topK	Accuracy								
	LB-ABOD			Fast ABOD			LOF/ABOD		
	P	*R*	*F*	*P*	*R*	*F*	*P*	*R*	*F*
10	1	0.5	0.67	1	0.5	0.67	1	0.5	0.67
15	0.67	0.5	0.57	1	0.75	0.86	1	0.75	0.86
20	0.5	0.5	0.5	0.85	0.85	0.85	1	1	1
25	0.44	0.55	0.49	0.72	0.9	0.8	0.8	1	0.89
30	0.37	0.55	0.44	0.6	0.9	0.72	0.67	1	0.8

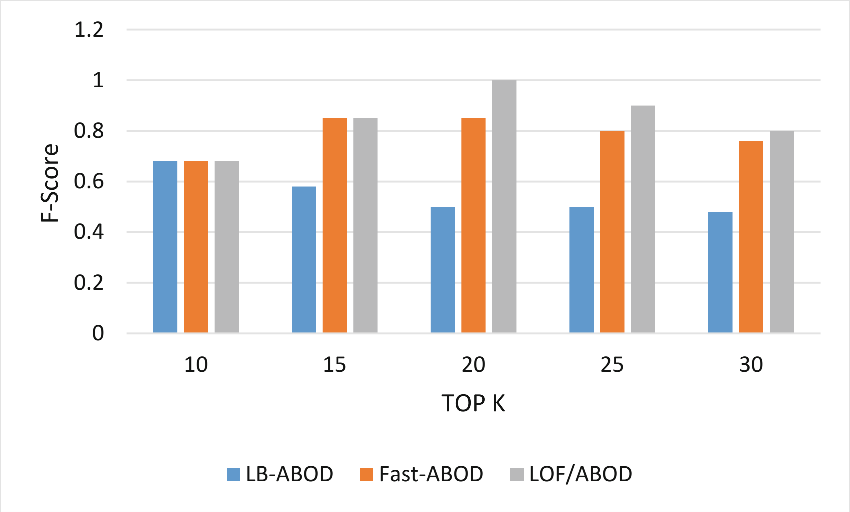


Fig. 5. The influence of topK on F-Score

is meaningless (recall = 1 and remains unchanged), but leads to the decrease of precision; for Fast ABOD, for LB-ABOD, the precision is gradually increased, but the increase of precision is far less than the decrease of recall. In general, its F-Score is decreasing. In summary, to detect outliers in the most efficient and accurate way (i.e. when F-Score is maximum), the size of topK must be equal to the number of real outliers in the data set.

6 Conclusion

The main work of this paper is to compare the application of different anomaly detection algorithms in anomaly data detection of power Internet of Things. Its data is mainly the data of electric power sensor network. Our goal is to improve the running time and efficiency of the algorithm on the basis of obtaining a suitable algorithm. It can adapt to the high-dimensional data acquired by the system. After our algorithm implementation, the CPU time of ABOD increases exponentially with the size of data set N, and the dimension increases linearly. The algorithm has high accuracy, but its time complexity is too high to be suitable for practical application. FAST ABOD and LB-ABOD are the improvements of ABOD algorithm, showing good results. The CPU time of LOF is approximately linear to data size N, and has a linear relationship with data dimension D. It can achieve very high accuracy only by taking fewer proximity points, and the ratio of CPU time to dimensionality remains unchanged, and it can still maintain very high accuracy in high-dimensional data.

Acknowledgement. This work is partially supported by Project No. 5211DS16001R of State Grid Zhejiang Electric Power Co., Ltd. This work is also supported by the National Natural Science Foundation of China (NSFC) under Grant No. 61502374.

References

1. Lyu, L., Jin, J., Rajasegarar, S., et al.: Fog-empowered anomaly detection in IoT using hyperellipsoidal clustering. IEEE Internet of Things J. **4**(5), 1174–1184 (2017)
2. Kriegel, H.-P., Schubert, M., Zimek, A.: Angle-based outlier detection in high-dimensional data. In: KDD, Las Vegas, Nevada, USA (2008)
3. Breunig, M.M., Kriegel, H.-P., Ng, R.T., et al.: LOF: identifying density-based local outliers. In: ACM MOD, Dallas, TX USA (2000)
4. Taghizadeh, M.J., Parhizkar, R., Garner, P.N., Bourlard, H., Asaei, A.: Ad hoc microphone array calibration: euclidean distance matrix completion algorithm and theoretical guarantees. Sig. Process. **170**, 123–140 (2014)
5. Zarpelão, B.B., Miani, R.S., Kawakani, C.T., et al.: A survey of intrusion detection in Internet of Things. J. Netw. Comput. Appl. **84**, 25–37 (2017)
6. Trihinas, D., Pallis, G., Dikaiakos, M.D.: ADMin: adaptive monitoring dissemination for the internet of things. In: IEEE Conference on Computer Communications, INFOCOM 2017, pp. 1–9. IEEE (2017)
7. Gregg, D., David, U.: Parameterizing dose-response models to estimate relative potency functions directly. Toxicol. Sci. **129**, 447–455 (2012)
8. Liu, C.-S.: Reconcile the perfectly elastoplastic model to simulate the cyclic behavior and ratcheting. Int. J. Solids Struct. **43**(2), 222–253 (2005)
9. Kriegel, H.P., Zimek, A.: Angle-based outlier detection in high-dimensional data. In: Proceedings of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 444–452. ACM (2008)
10. Lu, Y., Qin, X.S.: Multisite rainfall downscaling and disaggregation in a tropical urban area. J. Hydrol. **509**, 55–65 (2014)
11. Le Grand, S., Götz, A.W., Walker, R.C.: SPFP: speed without compromise—a mixed precision model for GPU accelerated molecular dynamics simulations. Comput. Phys. Commun. **184**(2), 374–380 (2013)
12. Polišenská, K., Chiat, S., Comer, A., McKenzie, K.: Semantic effects in sentence recall: the contribution of immediate vs delayed recall in language assessment. J. Commun. Disord. **52**, 65–77 (2014)
13. Feng, D.-C., Chen, F., Xu, W.-L.: Detecting local manifold structure for unsupervised feature selection. Acta Automatica Sinica **40**, 2253–2261 (2014)

14. Zhou, M., Wang, Y., Srivastava, A.K., et al.: Ensemble based algorithm for synchrophasor data anomaly detection. IEEE Trans. Smart Grid (2018)
15. Li, W., Mahadevan, V., Vasconcelos, N.: Anomaly detection and localization in crowded scenes. IEEE Trans. Pattern Anal. Mach. Intell. **36**(1), 18–32 (2014)
16. Erfani, S.M., Rajasegarar, S., Karunasekera, S., et al.: High-dimensional and large-scale anomaly detection using a linear one-class SVM with deep learning. Pattern Recogn. **58**, 121–134 (2016)