Partition Based with Outlier Detection

Saswati Bhattacharyya[1],RakeshK. Das[2],Nilutpol Sonowal[3],Aloron Bezbaruah[4], Rabinder K. Prasad[5]
[#]Student[1] , Student[2],student[3],student[4],Assistant Professor[5]& Departmentof Computer Science & Engineering, Dibrugarh University Institute of Engineering and Technology,Dibrugarh University Dibrugarh, Assam, India

Abstract—*Partition based method is widely used in every field of science and technology. It can detect spherical shaped clusters, but cannot detect any noisy information that is present in a data set. In this paper, we have proposed a partition based method with an outlier detection feature which can detect good quality clusters as well as identify the outliers present in it in optimal time. We have figured the outlier detection issue for the most part and planned calculations which can precisely identify anomalies in a way that the time complexity ought to be least. We have calculated the degree of outlier of each data object and included in existing partition based clustering technique to get good quality clusters along with the required anomalies. Additionally, utilizing a real world data set, we will exhibit that our methodologies can abstain from distinguishing false anomalies as well as discover genuine outliers overlooked by existing techniques.*

Keywords—*Partition-based Clustering, Outlier detection, degree of outlier, k-Mean*

I. INTRODUCTION

In recent years, our society has experienced a data explosion. Automated data collection tools and mature database technology are now used to collect tremendous amounts of data, far outstripping our ability to analyse and interpret them. Data mining [1] is a process of extracting valid, previously unknown, and ultimately comprehensible information from large datasets and using it for organizational decision-making. This information can be used to identify the patterns that occur frequently, illustrate the interesting associations among different patterns, and classify data items based on their characteristics. Besides extracting general rules and patterns, an equally important task of data mining is to identify abnormal patterns (outliers), which were often ignored or discarded as noise.

An outlier [2] is a data object that digresses fundamentally from the typical objects as though it were created by an alternate mechanism. Outliers have various applications which incorporate fraud detection in case of credit cards, intrusion detection in case of computer systems and networks, ecosystem disturbances, public health, medicine.

The reasons for anomalies can be information from various classes, regular variations in datasets, or information estimation and collection errors.

A.APPLICATIONS OF OUTLIERS [3]

The recognition of such unusual characteristics provides useful application-specific insights. Some examples are as follows:

1. Intrusion detection systems: In many computer systems, different types of data are collected about the operating system calls, network traffic, or other user actions. This data may show unusual behaviour because of malicious activity. The recognition of such activity is referred to as intrusion detection.
2. Credit-card fraud [3]: Credit card fraud has increasingly prevalent because of greater ease with which sensitive information such as a credit-card number can be compromised. In many cases, unauthorized use of a credit card may show different patterns, such as buying sprees from particular location or very large transactions. Such patterns can be used to detect outliers in credit-card transaction data.
3. Interesting sensor events: Sensors are often used to track various environmental and location parameters in many real-world applications. Sudden changes in the underlying patterns may represent events of interest. Event detection is one of the primary motivating applications in the field of sensor networks.
4. Medical diagnosis: In many medical applications, the data is collected from a variety of devices such as magnetic resonance imaging (MRI) scans, positron emission tomography (PET) scans or electrocardiogram (ECG) time-series. Unusual patterns in such data typically reflect disease conditions.
5. Law enforcement: Outlier detection finds numerous applications in law enforcement, especially in cases where unusual patterns can only be discovered over time through multiple actions of an entity. Determining fraud in financial transactions, trading activity, or insurance claims typically requires the identification of unusual patterns in the data generated by the actions of the criminal entity.
6. Earth science: A significant amount of spatiotemporal data about weather patterns, climate changes, or land-cover patterns is collected through a variety of mechanisms suchassatellites or remote sensing. Anomalies in such data provide significant insights about human activities or environmental trends that may be the underlying causes.

B. TYPES OF OUTLIERS [4]

There are three types of outliers –

1. Global Outliers (also called point anomalies):

A data point is considered a global outlier if its value is far outside the entirety of the data set in which it is found (similar to how "global variables" in a computer program can be accessed by any function in the program).

2. Contextual (conditional) Outliers:

A data point is considered a contextual outlier if its value significantly deviates from the rest the data points in the same context. Contextual outliers are common in time series data.

3. Collective Outliers:

A subset of data points within a data set is considered anomalous if those values as a collection deviate significantly from the entire data set, but the values of the individual data points are not themselves anomalous in either a contextual or global sense. In time series data, one way this can manifest is as a normal peaks and valleys occurring outside of a time frame when that seasonal sequence is normal or as a combination of time series that is in an outlier state as a group.

C. OUTLIER DETECTION METHODS [5]

An outlier can be defined as an observation that deviates so much from other observation that it arise suspicion that it is generated by some other mechanism.

There are certain methods in order to detect the outlier. They are listed below-

1. Extreme value method.
2. Clustering method.
3. Distance-based method.
4. Density-based method.

The methods are briefly discussed below:

1. Extreme value method: A data point can be considered as an extreme value if it lies in at one end of the probabilistic distribution. This data point can be considered as an outlier.
2. Clustering method: The clustering method is the clustering of objects that are similar in their characteristics from the objects of different characteristics. It is commonly used in statistical analysis.
3. Distance-based method: In this method the k^{th} nearest neighbour is analyzed in order to detect outlier. The data point is an outlier if k^{th} nearest neighbour is larger than other data points.
4. Density-based method: In this method, the local density of a data point is taken into account for outlier detection. If the local density of a data point is low, then it is considered as an outlier.

D. LOCAL OUTLIER FACTOR (LOF) [6]

The local outlier factor is defined as the approach that is based on local density where the locality is given by local density of the k^{th} nearest neighbours. The outlier is detected by comparing the local density of the given data point with the local densities of the other data points.

II. RELATED WORK

The classic definition of an outlier is due to Hawkins [6] who defines "an outlier is an observation which deviates so much from other observations as to arouse suspicions that it was generated by a different mechanism". Most of the previous work on anomaly detection is based on the field of statistics which use a standard distribution to fit the dataset. Outliers are described based on the probability distribution.

Knorr et al. proposed a new definition based on the concept of distance which is the notion of distance based outliers [7]. In the notion of distance based outliers is extended by using the distance to the *k*-nearest neighbour to rank the outliers. A very efficient algorithm to compute the top *n* outliers in this ranking is given by Knorr and Ng. This is formalized in the following definition of outliers [8] :

Definition 1: (k-distance of an object p) [9]

For any positive integer k, the k-distance of object p, denoted as k-distance (p), is defined as the distance d (p,o) between p and an object o ЄD such that:

(i) For at least k objects o' Є D |{p} it holds that d (p,o') <= d (p,o), and

(ii) For at most k-1 objects o' Є D | {p} it holds that d(p,o') <d(p,o).

Where,P and o are the objects in the dataset D.

d(p,o) is the distance between the objects p and o.

Definition 2: (k-distance neighbourhood of an object p)-Given the k-distance of p, the k-distance neighbourhood of p contains every object whose distance from p is not greater than the k-distance,

I.e. $N_{k\text{-distance(p}}$) = { $q \in D|\{p\} \mid d(p, q) <=$ k-distance(p) }.

These objects q are called the *k*-nearest neighbours of p.

Where, $N_{k\text{-distance}}$ (p) is the k-distance neighbourhood of p.

p and q are the objects in the dataset D.

Definition 3: (reachability distance of an object p w.r.t. object o)

Let *k* be a natural number. The reachability distance of object p with respect to object ois defined as

reach-$dist_k$(p, o) = max {k-distance(o), d(p, o) }.

Where ,reach-$dist_k$(p,o) is the reachability distance of an object p with respect to object o.

p and o are the object of the dataset D.

k-distance (o) is k-distance of object o.

d(p,o) is the distance between the object p and o.

Definition 4: (local reachability density of an object *p*)

The local reachability density of p is defined as

$$lrd_{minpts}(p) = 1 \Big/ \left(\frac{\sum_{o \in N_{minpts}(p)} reach\text{-}dist_{minpts}(p,o)}{|N_{minpts}(p)|} \right)$$

where, lrd is the local rechability density of object p.
$N_{MinPts}(p)$is the k nearest neighbourhood of the object p.
reach - $dist_{MinPts}(p,o)$ is the reachability distance between objects p and o.

Definition 5: (Local Outlier Factor)

$$LOF_{minpts}(p) = \frac{\sum_{o \in N_{minpts}(p)} \frac{lrd_{minpts}(o)}{lrd_{minpts}(p)}}{|N_{minpts}(p)|}$$

Where,LOF is the Local Outlier Probability.
$N_{MinPts}(p)$ is the k nearest neighbourhood of the object p.
is the ratio between local reachability density of p and local reachability of o.

The computation time of LOF [10] grows dimensionally with the number of dataset dimensions n, called the "curse of dimensionality". For high dimensional data, the complexity of the algorithm frequently becomes O (n^2). Various efforts have been made to reduce the computational complexity related to high dimensional data.

Clustering is all about finding "crowds" of data points, whereas outlier analysis is all about finding data points that are far away from these crowds. Clustering [11] and outlier detection, therefore, share a well-known complementary relationship. A simplistic view is that every data point is either a member of a cluster or an outlier. The detection of outliers as a side-product of clustering methods is, however, not an appropriate approach because clustering algorithms are not optimized for outlier detection.We shall concentrate our discussion on *Partition-based Clustering* [12] techniques. A partitioned clustering a simply a division of the set of data objects into non-overlapping subsets (clusters) such that each data object is in exactly one subset. Suppose we are given a database D of 'n' objects and the partitioning method constructs 'k' partition of data. Each partition will represent a cluster and k ≤ n. There are two partition methods- K-means and K-medoids [13].

K-Means [14]
The K-means algorithm is a clustering algorithm designed in 1967 by MacQueen which allows the dividing of groups of objects into ***K*** partitions (clusters) based on their attributes. The algorithm [12] for K-Means is given below:
1. Arbitrarily choose k objects from D as the initial cluster centre.
2. Repeat
3. Reassign each object to the cluster to which the object is the most similar, based on the mean value of the objects in the cluster.
4. Update the cluster means i.e., calculate the mean value of the objects for each cluster.
5. Stop, if no data point was reassigned.

Usually, the complexity of K-means algorithm is O (n*K*I*d), where n is the number of data points, K is the number of clusters, I is the number of iterations and d is the number of attributes. Since I<<n, so, complexity is O (1). Hence, it is quite efficient clustering algorithm. However, since the main objective of a clustering algorithm is to find clusters, they are developed to optimize clustering, and not to optimize outlier detection. The exceptions (called "noise"[15] in the context of clustering) are typically just tolerated or ignored when producing the clustering result. Therefore, we propose an efficient method which is discussed in detail in the next section of the paper.

III. PROPOSED METHOD

By comparative study of the clustering methods and outlier detection methods which is *k-means and local outlier factor* we have come to the conclusion that both these methods have some demerits in handling data with outliers and detecting outliers of large data objects where there can be local as well as global outliers [16] and sometimes it is very difficult to detect and eliminate these outliers if noisy information is present in the data object. So, to overcome this problem we have introduced a method which is '*Partition Based with Outlier Detection'*. This helps us to detect outliers and eliminate them in various data objects that are implemented.

The k-means method and the local outlier method has certain limitations that is the k-means method is very simple and efficient algorithm and has very less complexity of O(n) but it faces a lot of problems in handling data with outliers. On the other hand local outlier factor algorithm is very good in detecting outliers but it has a high complexity of O (n^2) [17].

To overcome these limitations of k-means and local outlier factormethod we have proposed this method. This method is basically works in two steps. In first step, it computes the degree of outlier factor of each data object with respect to k and assigned local outlier factor to each object.

In second step, it partitions the data objects those degree of outlier factor less than some threshold value into k-numbers of clusters.

This method then merge the result of these two algorithms and eliminates the detected outliers except the formed clusters in much lesser time and gives us an optimal result with less complexity as compared to the result that is obtained by executing the local outlier factor method[18].

Our proposed method can be illustrated with the following algorithm:
Input- Dataset D= {D_1, D2 …. D_n},
K number of initial centroids
Output- K clusters
Step 1: Compute degree of outlier of each data point from D.

Step 2: Assign each data point whose degree of outlier is greater than threshold value to outlier.
Step 3: Select K number of points as initial centroids.
Step 4: Assign those data points which are not outliers to its nearest centroid.
Step5: Recompute the centroids and repeat steps 4 and 5 until the centroids are same.

So, by applying the above algorithm we can detect both local and global outliers while forming the k clusters. We have taken 2-dimensional data sets so that we can properly visualize it on a graph. We have considered K number of data points as initial centroids. Then we have assigned a degree of outlier to each and every data point to calculate the degree of outlierness of the objects. Next, we assigned those data points whose degree of outlier is greater than a threshold value (that we shall assume) as outliers. In the next step, we have assigned the data points that are not outliers to its nearest centroids. Finally, we recomputed the centroids until the centroids are same and formed the k clusters with distinguished outliers which can be later eliminated. We have overlapped the combination of all three of red, green and blue to distinguish the clusters from each other and detect the clusters in case of a large data set.

IV. Experimental Results

Our proposed method is designed with the help of *Java 1.8, Intel(R) Core(TM) i3-4030U CPU@ 1.90GHz* which has *64 bit operating* system with *4GB RAM*.

We have taken three datasets, dataset 1, dataset 2 and dataset 3. We have assumed four clusters i.e. k=4 for dataset 1 and 2 and three clusters for dataset 3. We have calculated the degree of outlier [19] of each data point in all the datasets.
First of all we have considered a case i.e. case 1 where we have used the dataset 1 with nine hundred and eighty data points where there are no outliers. In fig 1(a) we have executed the classical k means algorithm. Since this dataset is void of outliers, the algorithm will output us a set of k clusters C1, C2,C3 and C4. In fig 1 (b) we have executed our proposed method i.e. "*Partition based with outlier detection"* which also forms normal k clusters C1, C2, C3 and C4.

In case 2, we have considered the dataset 2 which contains twelve hundred data points with outliers P1, P2 and P3. In fig 2 (a) we have executed the classical k means algorithm which outputs us the k clusters C1, C2,C3 and C4. However, it fails to detect the outliers P1, P2, P3. In fig 2 (b) we have executed our proposed method which forms the k clusters C1,C2,C3 and C4 while simultaneously detecting and eliminating the outliers P1, P2 and P3.

Similarly, we have taken case 3 where we have taken the dataset 3 which contains eighteen hundred and six data points with outliers P1 and P2. In fig 3 (a) we have executed the classical k means algorithm which outputs us the three clusters C1, C2 and C3 and in fig 3(b), we have executed our proposed method which forms the k clusters C1, C2 and C3 while simultaneously detecting and eliminating the outliers P1 and P2.

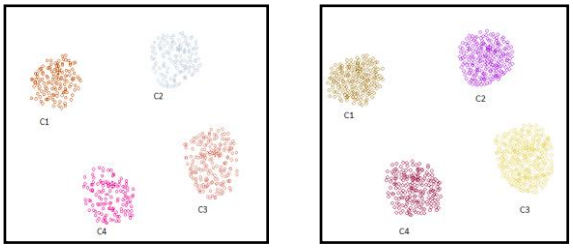

Fig1 (a):k-Mean(Dataset 1,k=4) 1(b)Proposed Method(k=4)

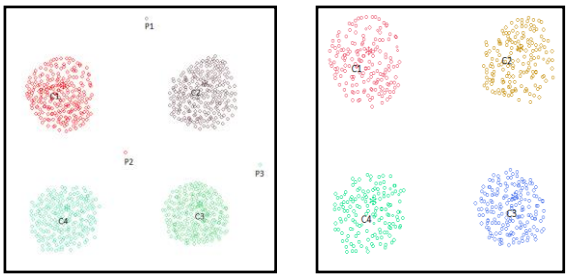

Fig2 (a): k-Mean (Dataset 2,k=4) 2(b)Proposed Method(k=4)

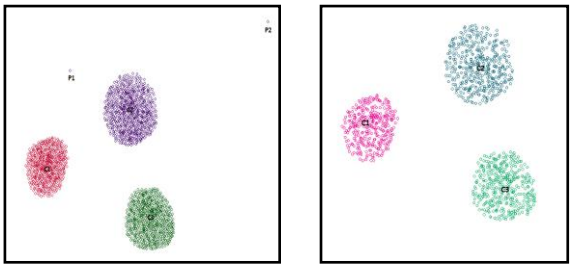

Fig3 (a): k-Mean (Dataset 3,k=3) 3(b)Proposed Method(k=3)

V.CONCLUSION

Finding outliers is an important task for many data mining applications. In this paper, we have proposed a partition based method, which can detect clusters and identify outliers simultaneously in optimal time from given datasets.

Existing outlier detection methods are very time-consuming, while our method "Partition based with outlier detection" detects outliers with adequate number of clusters in optimal time.

However we have observed that the cluster quality deteriorates when we increase the value of k (number of clusters) which is one of the demerits of this method and the complexity of this method is comparatively higher than the classical *partition-based method*.

References

[1] BharatiKamble ,KanchanDoke , "Outlier Detection Approaches in Data Mining"Computer Engineering, Mumbai University, April 2010.

[2] Pang Ning Tan, Vipin Kumar, Michael Steinbarch, *Introduction to Data Mining*, Sixth Edition,2011, Pearson Education

[3] Yufeng Kou, "Abnormal Pattern Recognition in Spatial Data", Virginia Polytechnic Institute and State University, Doctor of Philosophy in Computer Science and Applications, November, 2006.

[4] Jiwai Hen, MichelineKamber, Jianpei, *Data Mining Concepts and Techniques*, Third Edition, 2012, Morgan-Kaufmann.

[5] (2018) The Wikipedia [Online] Available: https://en.wikipedia.org/wiki/Local_outlier_factor

[6] Hawkins, D, "Identification of Outliers", Chapman and Hall, London, 1980.

[7] Edwin M. Knorr and Raymond T. Ng , "Algorithms for Mining Distance-Based Outliers in Large Datasets", Department of Computer Science, University of British Columbia, Vancouver, BC V6T 124 Canada, December 1998.

[8] Markus M. Breunig, Hans-Peter Kriegel, Raymond T. Ng, JörgSander, "LOF: Identifying Density-Based Local Outliers", London 1983.

[9] Jihwan Lee and Nam-Wook Cho, "Fast Outlier Detection Using a Grid-Based Algorithm", Department of Industrial and Management Engineering, Hankuk University of Foreign Studies, Republic of Korea , November 2016.

[10] Charu C. Aggarwal,*Outlier Analysis*, Second Edition, New York, November 25, 2016.

[11] Jaeshin Lee, Bokyoung Kang*, Suk-Ho Kang, "Independent component analysis and local outlier factor for plant-wide process monitoring", Department of Industrial Engineering, Seoul National University, Seoul, 151-742, Republic of Korea

[12] Barnett V., Lewis T.: "Outliers in statistical data", John Wiley, 1994.

[13] Ester M., Kriegel H.-P., Sander J., Xu X.: "A Density-Based Algorithm for Discovering Clusters in Large Spatial Databases with Noise", Proc. 2nd Int. Conf. on Knowledge Discovery and Data Mining, Portland, OR, AAAI Press, 1996.

[14] Knorr E. M., Ng R. T.: "Finding Intentional Knowledge of Distance-based Outliers", Proc. 25th Int. Conf. on Very Large Data Bases, Edinburgh, Scotland, 1999.

[15] Ramaswamy S., Rastogi R., Kyuseok S.: "Efficient Algorithms for Mining Outliers from Large Data Sets", Proc. ACMSIDMOD Int. Conf. on Management of Data, 2000.

[16] V. Barnett and T. Lewis. "Outliers in Statistical Data", John Wiley & Sons, 1994.

[17] Ng R.T., and Han J. 1994, "Efficient and Effective Clustering Methods for Spatial Data Mining", Proc. 20th Int. Conf. on Very Large Data Bases, 144-155. Santiago, Chile.

[18] K. Zhang, M. Hutter, and H. Jin, "A new local distance-based outlier detection approach for scattered real-world data", in Proc 13th Pacific-Asia Conf on Knowledge Discovery and Data Mining (PAKDD), 2009.

[19] M. E. Houle, H.-P. Kriegel, P. Kr ¨oger, E.Schubert, and A. Zimek, "Can shared-neighbour distances defeat the curse of dimensionality?" in Proc 22nd Int Conf on Scientific and Statistical Database Management (SSDBM), 2010.