A Novel Approach for Cluster Outlier Detection in High Dimensional Data

Ravi Kumar Gupta[1], **Sandeep Bhargava**[2]

[1]Suresh Gyan Vihar University, Jagatpura, Jaipur, Rajasthan
[2]Suresh Gyan Vihar University, Jagatpura, Jaipur, Rajasthan

Abstract: ***In modern era there are lots of data mining algorithms which focus on clustering methods. There are also several types of approaches designed for outlier detection. Outliers are those data objects that do not fulfill with the common behavior or model of the data. Many data mining algorithms try to reduce the effects of outliers or remove them all together. We investigated that in many different conditions clusters and outliers whose meanings are connected to each other, especially for those data sets which contains some noise. So it is important to deal clusters and outliers as concepts of the same significance in data analysis. So in this paper we introduce an algorithm which is based on k means [1] for the detection of clusters and outliers that aim to detect the clusters and the outliers in a different view for those data sets which contains some noise. In this algorithm clusters are detected and managed according to the intra-relationship within the clusters and inter-relationship between the clusters and the outliers. The whole management and modification of the clusters and outliers are done repeatedly just before a certain termination is reached.***

Keywords: K means, Clusters, Outliers, Data analysis**.**

1. Introduction

A large quantity of multi dimensional data wants to be clustered and analyzed. Several clustering, outlier detection and cluster evaluation methods have been presented in the last few years [2, 3, 4, 5]. Nowadays many real data sets contain lots of noises, which results in the poor performance of the designed algorithms and also degrade their efficiency and working. We investigated that in many different conditions clusters and outliers whose meanings are connected to each other, especially for those data sets which contains some noise. So it is important to deal clusters and outliers as concepts of the same significance in data analysis. Other major problem in data analysis field is that clusters and outliers are detected mainly on the information of the features of data sets and the results are compared to ground truth of the natural clusters and outliers. In the real world the ground truth and the information of features of the real datasets do not match each other and good results are hard to achieve even when using dimension reduction method. We try to detect the clusters and outliers in another context by not only depending on the characteristics of the data sets, but also using the relationship between the clusters and the outliers in a measurable way.

2. Problem descriptions

The basic conception of both clusters and outliers are linked to each other. Real world data do not consist of natural cluster at all and for those which have clusters there are rarely few cases in reality that the data objects or data points in the data all belong to the natural clusters. The cluster outlier detection problem is described as follows. We shall introduce a few notation and definitions. Let n be the total number of data points and d be the data space dimensionality. Let the input d dimensional dataset be **A**

$$\mathbf{A} = \{A_1, A_2, \ldots\ldots, A_n\}$$

Which is normalized to be in the hypercube $[0, 1]^d \subset R^d$. Now each data point is a dimensional vector.

$$\vec{A_i} = [A_{i1}, A_{i2}, \ldots\ldots, A_{id}].$$

According to the input parameter of the initial cluster outlier data set division, we perform this algorithm repeatedly. In a given step we imagine that the current number of clusters is F_c and the current numbers of outliers is F_o. The set of clusters are $C' = \{C', C', \ldots., C'_{Fc}\}$, and the set of outliers $O' = \{O', O', \ldots.., O'_{Fo}\}$. Here we use the term compactness for measuring the quality of a cluster according to the closeness of data points to the cluster centroid.

2.1 Cluster compactness

A cluster which is obtained from a data set is a subset in which the points have a closer relationship with each other than the points outside of the cluster. From the literature [6, 7], the intra cluster relationship is measured by compactness and the inter cluster relationship is measured by separation. Compactness is a relative term it says an object is compact in relation to a flaccid surrounding environment.
Given set of clusters $C' = \{C', C', \ldots., C'_{Fc}\}$ and the set of outliers $O' = \{O', O', \ldots.., O'_{Fo}\}$, the compactness (CP) can be the closest measurement of the data points in C_i to the centroid of C_i
.

$$CP\ (C_i) = \frac{\sum_{p \in C_i} d\,(p, k_{C_i})}{|C_i|}$$

Where k_{C_i} is the centroid of the Cluster $C_{i,}$ p is a data point in Cluster $C_{i,}$ $|C_i|$ number of data points in C_i, and $d\ (p, k_{C_i})$ is the distance between p and k_{C_i}. The centroid k_{C_i} of the cluster is the algebraic average of all the points in the cluster: $k_{C_i} = \frac{\sum_{p \in C_i} p}{|C_i|}$.

2.2 Diversities of data groups

The term diversity is used to describe the difference between the two clusters, the difference between the two outliers and the one difference between the cluster and an outlier. The method which is used to designate the diversity between a cluster and an outlier is same to the problem of the distance measurement between a query point and a cluster in the field of data mining. We can use the compactness (CP) concept of the cluster instead of using the density for setting up the distance measurement weights.

Diversity between a cluster C and an outlier O is:

$$\mathcal{D}_1 (C, O) = w_1.d_{min} (O, C) + w_2.d_{avr} (O, C)$$

Where $w_1 = \frac{1}{CP\ (C)+1}$, $w_2 = \frac{CP\ (C)}{CP\ (C)+1}$, d_{avr} (O,C) = d (O, k_c) and d_{min} (O,C) = max (d(O, k_c) – r_{max}, 0) where r_{max} is the data points distance in C from its centroid. The rule for setting up the weights w_1 and w_2 are same in [8]. Here we employ a simple measure which integrates the concept of compactness into the diversity measurement of two clusters which is use to show how compact the data points inside the cluster. Diversity between two clusters C_1 and C_2 is:

$$\mathcal{D}_2 (C_1, C_2) = \frac{d(C1,C2)}{CP\ (C_1) + CP\ (C_2)}$$

Where d (C_1, C_2) be the average distance between the two clusters or the minimum distance between them. Though the value of the $\mathcal{D}_2$ (C_1, C_2) will be larger enough then the diversity between the clusters C_1 and C_2 will also be large.

Diversity between two outliers O_1 and O_2 is:

$$\mathcal{D}_3 (O_1, O_2) = d (O_1, O_2)$$

2.3 Qualities of data groups

The quality of cluster C is not only reflected by the diversity between it and other clusters but also between it and outliers. This means that how far they are from each other. Suppose if C is located near some outliers its quality will be affected because outliers are meant to be far away from any type of clusters. We take consideration for checking of the both diversity between the clusters and diversity between a cluster and an outlier which will define the clusters quality.

Quality of the cluster C is:

$$Q (C) = \frac{\frac{\sum_{C' \in C, C' \neq C} D_2(C,C')}{F_C - 1} + \frac{\sum_{O \in O} D_1(C,O)}{F_O}}{CP\ (C)}$$

Whenever the larger Q (C) is, the quality of the cluster C will be better.

Similarly the quality of an outlier O is:

$$Q (O) = \frac{\sum_{O' \in O, O' \neq O} D_3(O,O')}{F_O - 1} + \frac{\sum_{C \in C} D_1(C,O)}{F_C}$$

The larger Q (O) is, the outlier O will attain the better quality.

3. Cluster outlier detection

The fundamental objective of this algorithm is to extract the optimal set of clusters and outliers from the given set of input data. Clusters and outliers are closely linked to each other and they also influence each other in certain way. In this algorithm the clusters are detected and managed according to the intra-relationship within the clusters and inter-relationship between the clusters and the outliers, and vice versa. In this we will apply an approach which is similar as greedy method proposed in [9]. The whole management and modification of the clusters and outliers are done repeatedly just before a certain termination is reached.

The algorithm is partitioned into two stages: In the first stage, we have to find the cluster's centres and the outliers locations. In the second stage we have to refine the set of clusters and outliers regularly by optimally transferring some outliers and some boundary data points of the clusters.

3.1 First stage

In this stage we have to find the initial set of medoids. In the next step after finding the medoids, we have to dispatch the data points to the nearest medoids and forms data subsets associated with medoids. Then we have to attain some kind of methods to conclude whether a data subset is a cluster or a group of outliers. We explain each step in detail as follows.

3.1.1 Acquiring medoids

It is crucial to find the medoids which can be the almost centres of different clusters for our method. A similar method which is proposed in [10] we have to select a random set of data points T_1 from the original data set which having the size of RS1 which is equivalent to the required K cluster number. Now we have to apply the greedy technique [9] for finding the other random set T_2 from T_1 having the size of RS2 which is also equivalent to the K and RS1>RS2. After imposing the greedy technique on RS2 the efficiency is highly improved of the algorithm, and there is a high reduction in the growth of the number of outliers produced by the algorithm.

3.1.2 Dispatch the data points

After getting the smaller random set T_2 of medoids, we have to find a technique to conclude which medoids are in some clusters and which ones are outliers. First we have to assign each data point *dp* to a specific medoids $\in T_2$ which is the closest one to *dp*. After completion of this step each $\text{medoid}_i \in T_2$ is associated with the set of data points.

3.1.3 Data set division

Now as we get the set H of the initial division of the input dataset **A**, now we have to check the size of each medoid associated with the subset of the data $\in$ H. Now we exploit some kind of method which adjusts the criterion for determining that whether a medoid is an outlier or it belongs to the cluster. After the completion of the process of cluster or outlier, it should be contrary that the size of the cluster set Ƈ < K if the initial sizes RS1 and RS2 are large enough. If it happens again we just run the initial setup to make sure that the size of the cluster set Ƈ is at least K.

3.2 Second stage

In this second stage, first we have to merge the initial set of clusters into K clusters. And in the second step grouping of clusters and outliers according to their qualities and selection of the worst cluster and outliers is done. For checking the quality of each cluster it has to be calculated according to the intra relationship within the clusters and the inter relationship between the clusters and the outliers. In the third step we will attain some methods for selecting the boundary data points for the worst qualities of clusters. And in the fourth step we will refine the set of clusters and outliers continuously by optimally exchanging the selected boundary data points and the worst qualities of outliers. Hence steps two, three and four are done repeatedly just before a certain termination condition is reached. We explain each step in details.

3.2.1 Merging of the clusters

Before performing the outlier detection process we have to first merge the current set of cluster Ƈ to K cluster. This process is an iterative one in each iteration phase, whichever two nearest or closest clusters are found in Ƈ they are merged together. According to the diversity measurement $\mathcal{D}_2$ (C_1, C_2) of two clusters (defined in the part 2.2) the distance between the clusters C_1 and C_2 is calculated. The iteration step is performed continuously until the total number of clusters in Ƈ is K. Now we have to compute the centroid of each cluster C_i ∈ Ƈ (denoted as c_i).

3.2.2 Grouping of clusters and outliers

Now for each outlier ∈ Ơ we have to find its nearest cluster ∈ Ƈ. According to the diversity measurement $\mathcal{D}_1$ (C, O) which is (defined in part 2.2) the distance between the cluster C and outlier O is calculated. And according to the information of the outliers in Ơ and the information of the clusters in Ƈ the quality Q (O) (which is defined in the part 2.3) is calculated. The worst qualities of outliers are put into the set Ơ'. Thus, for each cluster C ∈ Ƈ. According to not only the information of the clusters in Ƈ but also the information of outliers in Ơ the quality Q (C) (defined in the part 2.3) is calculated. The worst qualities of clusters are put into set Ƈ'.

3.2.3 Find boundary data points

We need those data points into the clusters that are not only more distant from the centroid of the clusters but also contain the smallest number of neighboring data points as the data points of the clusters. The latest circumstances assure that this method not only promote clusters of standard geometries such as hyper spherical.

3.2.4 Exchange of data points

In this step it is use to exchange the outliers and the boundary data points characteristics. Now for each outlier O in |Ơ'| we add it into its closest cluster. For each boundary data point bp in BP we change it into a new outlier. Thus, we do not exchange the data points of the boundary between clusters are that whole data division quality will be degraded if it is carried.

Algorithm (K: Number of clusters)

```
Start
1. First stage
Repeat
A and B are the two equivalent constant to K, where A>B;
RS1 = A. K;
RS2 = B. K;
$T_1$ = it is the random set having the size of RS1;
$T_2$ = Finding K medoids from $T_1$ having size of RS2;
H ← Dispatch the data points ($T_2$);
Ƈ and Ơ ← it is the cluster or the outlier ();
Until |Ƈ| ≥ K
2. Second stage
Ƈ ← Merging of the Cluster (Ƈ);
Repeat
For each outlier o ∈ Ơ do
Start
Find the nearest cluster ∈ Ƈ
Stop
Group the current set of clusters and current set of outliers in ascending order according to their qualities;
Now exchange the cluster and the outlier ();
Ơ' it is the set of outliers according to the worst qualities;
BP it is the set of boundary data points according to the worst qualities;
U = Ơ' ∪ BP;
Until (U gives a constant value or the iteration ≥ Ω)
Stop.
```

3.3 Analysis of time and space

Let us assume size of data set is n. Now during the whole process we need to keep track of information on all points which collectively holds O (n) space. For the second stage or we can say the iteration stage we need space for the information of current set of clusters Ƈ and the current set of outliers Ơ, the boundary data points for each cluster, the worst qualities of outliers and clusters in each iteration. The total amount of space needed is O (n). The time required for each iteration is O (n + $|Ƈ| \, log|Ƈ| + |Ơ| \, log|Ơ|$) particularly for the computation of the various types of qualities and sorting process. Ƈ and Ơ.

So the total amount of time required for the algorithm is O (Ω *(n + $|Ƈ| \, log|Ƈ| + |Ơ| \, log|Ơ|$)) in which Ω is known as the threshold for the number of iterations.

3.4 Workflow of the algorithm

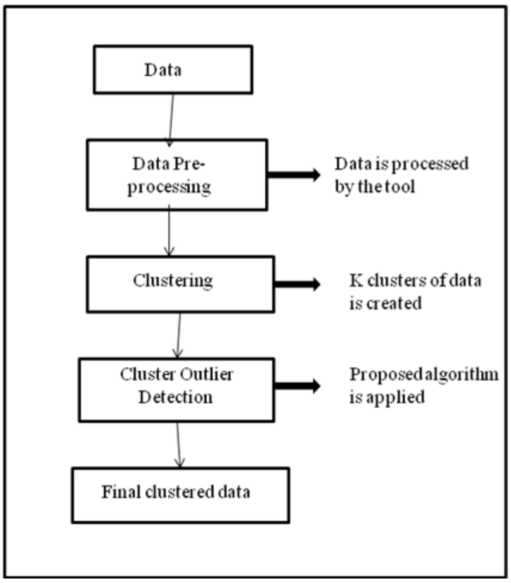
Data
Data Pre-processing
Data is processed by the tool
Clustering
K clusters of data is created
Cluster Outlier Detection
Proposed algorithm is applied
Final clustered data

Figure 1: Workflow of the algorithm

Above is the figure depicting the workflow chart of our proposed algorithm? Firstly the data set is taken and feed into the tool for pre-processing of the data and after the completion of this step K clusters of Data is being created. Then our proposed algorithm is applied on the data for cluster outlier detection and in the last we get the final clustered data.

4. Experiments and Results

In this section we have done the experiment on Ecoli data set which is taken from [11]. We conducted our experiment on Cluster 3.0 tool [12]. We use an Ecoli data set which is used to find Localization site of proteins. The dataset contains a total of 336 instances (objects) each having attributes (1 name and 7 input features). It contains eight clusters having sizes of (143, 77, 52, 35, 20, 5, 2 and 2). Now we perform our algorithm on the Ecoli data and after the experiment we observe that our proposed algorithm has the most information about the first three largest clusters.

Table 1: Clustering result of the algorithm for Ecoli data

	K=0	K=1	K=2	K=3	K=4	K=5	K=6	K=7
$C^{*}(R)$	143	77	52	35	20	5	2	2
$C^{*}(D)$	138	80	73	N	N	4	N	N
$C^{*}(R)\cap C^{*}(D)$	132	54	50	N	N	4	N	N
Accuracy%	95.65	67.50	68.49	N	N	100	N	N
Recall value %	92.3	70.12	96.15	N	N	80	N	N

Now from the result table we see that the data set contains 8 clusters $C^{*}(R)$ for K = (0 to 7). There are three clusters which are too small so we set the clusters parameter number K to 5. And the accuracy of the detected cluster measured according to the accuracy % and the recall value %. Now for $C^{*}(D)$ detected cluster and $C^{*}(R)$ for the real cluster we calculate the accuracy of $C^{*}(D)$ with respect to $C^{*}(R)$ as $\frac{C^{*}(D)\cap C^{*}(R)}{C^{*}(D)}$ and the recall value is $\frac{C^{*}(D)\cap C^{*}(R)}{C^{*}(R)}$. Hence $C^{*}(D)$ is called as comparable cluster of $C^{*}(R)$ if the accuracy and the recall value of $C^{*}(D)$ and $C^{*}(R)$ are high.

5. Conclusions

In this paper we introduce a novel approach for cluster outlier detection in high dimensional data. This approach can be able to improve the clusters and outliers qualities for those high dimensional data which contains noise. The clusters are detected and managed according to the intra-relationship within the clusters and inter-relationship between the clusters and the outliers. The whole management and modification of the clusters and outliers are done repeatedly just before a certain termination is reached. Now further dealing with the clusters and outliers as a concept of the same significance in data analysis. We also concern about the flushing the difficulties of the deficiency of match between the ground truth of the real data and their obtainable characteristics.

References

[1] J. MacQueen. Some methods for classification and analysis of multivariate observations. Proceedings of the fifth Berkeley Symposium on Mathematical Statistics and Probability. Volume I, Statistics. 1967.

[2] D. Yu, G. Sheikholeslami and A. Zhang. Findout: Finding outliers in very large datasets. The Knowledge and Information System, (4), October 2000.

[3] Charu C. Aggarwal et al. Outlier Detection for high dimensional data. In SIGMOID Conference, 2001

[4] M.F. Jiang et al. Two phase clustering process for outlier detection. Pattern Recognition Letters 22 pages 691-700. 2001

[5] Angiuli et al. Fast outlier detection in high dimensional spaces. In the proceedings of KDD. 2002.

[6] Chi-Farn Chen, Jyh-Ming Lee. The Validity Measurement of Fuzzy C-Means Classifier for Remotely Sensed Images. In Proc. ACRS 2001 - 22nd Asian Conference on Remote Sensing, 2001.

[7] Maria Halkidi et al. A data set oriented Approach for clustering Algorithm Selection. In PKDD, 2001.

[8] Dantong Yu and Aidong Zhang. ClusterTree: Integration of Cluster Representation and Nearest Neighbor Search for Large Datasets with High Dimensionality. IEEE Transactions on Knowledge and Data Engineering (TKDE), 14(3), May/June 2003.

[9] T. Gonzalez. Clustering to minimize the maximum intercluster distance. Theoretical Computer Science, 38:311-322, 1985.

[10] Charu C. Aggarwal et al. Fast algorithms for projected clustering. In the proceedings of the ACM SIGMOID Conference on management of Data, pages 61-72, 1999

[11] The UCI KDD Archive [http://Kdd.ics.uci.edu]. University of California, Irvine, Department of Information and Computer Science.

[12] Michiel de Hoon. [http://bonsai.ims.u-tokyo.ac.jp], Open source clustering software. Institute of medical Science University of Tokyo.

Author Profile

Ravi Kumar Gupta was born in India in 1990. He completed his B. Tech degree with first class division in Information Technology and pursuing M. Tech in Software Engineering from Suresh Gyan Vihar University, Jaipur, India. His research interest includes A novel approach for cluster outlier detection in high dimensional data. He also submitted dissertation on the same topic. He is at Gyan Vihar School of Engineering and Technology, Jaipur (Rajasthan), India.

Sandeep Bhargava has completed his M. Tech degree from Suresh Gyan Vihar University, Jaipur, India. Presently he is an Asst. Professor at Suresh Gyan Vihar University, Jaipur (Rajasthan), India.