International Journal of Computer Applications Technology and Research
Volume 2– Issue 5, 552 - 556, 2013

Detection of Outliers and Reduction of their Undesirable Effects for Improving the Accuracy of K-means Clustering Algorithm

Bahman Askari
Department of Computer Science and Research Branch, Islamic Azad University, Khouzestan, Iran

Sattar Hashemi
Department of Computer Science & Engineering, Shiraz University, Shiraz, Iran

Mohammad Hossein Yektaei
Department of Computer Abadan Branch, Islamic Azad University, Abadan, Iran

Abstract: Clustering is an unsupervised categorization technique and also a highly used operation in data mining, in which, the data sets are divided into certain clusters according to similarity or dissimilarity criterions so that the assigned objects to each cluster would be more similar to each other comparing to the objects of other clusters. The k-means algorithm is one of the most well-known algorithms in clustering that is used in various models of data mining. The k-means categorizes a set of objects into certain number of clusters. One of the most important problems of this algorithm occurs when encountering to outliers. The outliers in the data set lead to getting away from the real cluster centers and consequently a reduction in the clustering algorithm accuracy. In this paper, we separate outliers from normal objects using a mechanism based on dissimilarity of objects. Then, the normal objects are clustered using k-means algorithm process and finally, the outliers are assigned to the closest cluster. The experimental results show the accuracy and efficiency of the proposed method.

Keywords: clustering, k-means, outliers, outlier detection

1. INTRODUCTION

Clustering is one of the highly used methods in data mining [12], wireless sensor networks [8,13], pattern recognition [14] and machine learning [7], which is used to detect the groups that are different enough from each other and contain similar objects [4,5]. The importance of clustering in various fields and also the type of data being used, clustering speed, accuracy and lots of other parameters, leads to introduce various methods and algorithms in data clustering. Clustering is an unsupervised technique in which the data sets which are usually vectors in multi-dimension space are divided into a certain number of clusters based on a similarity or dissimilarity criterions. For example, if the number of clusters is K, and there exist n number of m-dimension data, the clustering algorithm will assign each one of these data to a cluster. This assignment takes place according to this rule that the assigned data to a certain cluster are more similar to each other rather than the other clusters. The k-means algorithm is one of the most well-known clustering algorithms and is being used in various types of data mining. The k-means categorizes data set objects in certain numbers of clusters [10,3]. This method is one of the most attractive and highly used operations in clustering techniques, because it is simple and understandable and its time complexity is linear. In general, this algorithm consists of two phases. In the first phase, k numbers of objects are selected from data set in a random manner and are considered as the initial centers of clusters. In the second phase, the distance between objects and the clusters centers is determined and each object is placed in the nearest cluster. To determine the distance between objects, the Euclidean distance criterion is used, generally. When all of the objects have been placed in the corresponding clusters, the clusters centers are calculated using repetitive averaging of objects of each cluster. The second phase continues until satisfying the algorithm ending condition. The Pseudo-code of k-means algorithm is shown in figure 1.

K-means algorithm

Input: Data set D = { d_1, d_2, … d_n }, where d_i =data points, n= number of data points
K = number of cluster centers
Output:
Clusters : K clusters with their centers

Step 1:
Randomly select k data object from dataset D as initial cluster centers.
Step 2:
Repeat step 3 to step 4 till no new cluster centers are found
Step 3:
Calculate the distance between each data object di (1<=i<=n) and all K cluster centers Cj (1<=j<=K) and assign data object di to the nearest cluster.
Step 4:
For each cluster j (1<=j<=K), recalculate the cluster center.

Figure 1. The pseudo-code of k-means algorithm

The distance to the cluster center is calculated in the following method:

$$Distance(d_i, C_j) = \sqrt{\sum_{p=1}^{m} d_{ip} - C_{ip}} \quad (1)$$

Where, d_i denotes the i-th vector of data, C_j is the center of j-th cluster and m denotes the number of attributes and cluster centers.

The cluster centers are updated according to the following equation:

$$C_j = \frac{1}{n_j}\left[\sum_{\forall d_i \in Cluster_j} d_i\right] \quad (2)$$

Where n_j denotes the number of vectors in the j-th cluster and $cluster_j$ is a subset of all vectors which form the j-th cluster.

Figure 2 illustrates the clustering steps of a manual data set. This dataset is divided into two clusters using the k-means algorithm. At first, two clusters are formed by random selection of objects (figure 2-a), then the clusters centers are determined by averaging of objects of each cluster (figure 2-b) and the clustering process continues using the new cluster centers (figure 2-c). For this dataset, the clustering is finished after two iterations of the algorithm (figure 2-d).

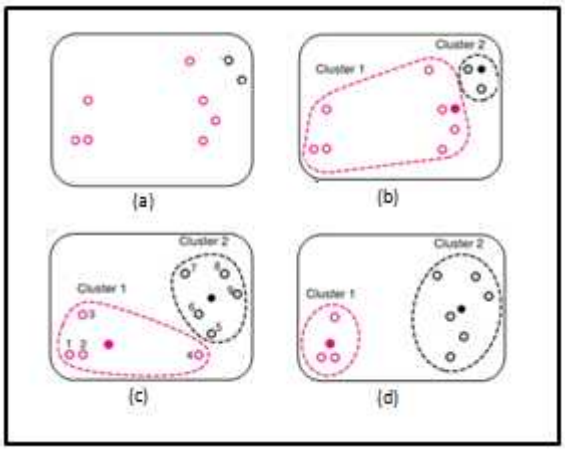

Figure 2. clustering of manual data using the k-means algorithm

The k-means algorithm has some pitfalls. For instance, it stops in local optimums and is sensitive to the initial values of clusters centers and outliers in dataset. In each dataset, an outlier is an object which its distance is not normal comparing to the other objects. In other words, an outlier is an object that has less similarity than the other objects. The existence of these objects in dataset has an undesirable effect on the efficiency and accuracy of the clustering algorithms such as the k-means.

2. RELATED WORK

Outlier detection has been a very interesting topic for research community [11,1,6,2,9]. Ramaswamy et al proposed a distance based outlier detection method in [11]. According to which, given parameters k and n, an object is an outlier if no more than n-1 other object in the dataset have higher value for D_k than object o, where $D_k(o)$ denotes the distance of k-th nearest neighbor of object o. This idea is further extended in [14], where each data point is ranked by the sum of distance from its k-th nearest neighbors. Breunig et al introduced the notion of the Local Outlier Factor (LOF) in [1] which captures the relative degree of outlierness of an object. It is local in sense that the degree of outlierness depends on how isolated an object is with respect to surrounding neighborhood. Above described methods are either distance based or nearest neighbors based that are not suitable for outlier detection in data streams due to their high time complexity. He et al in [6] presented new definition of outlier which they named as cluster-based local outlier, which provides importance to the local data behavior. They defined Cluster-Based Local Outlier Factor (CBLOF), a measure for identifying the physical significance of an outlier and an algorithm for discovering outliers is also proposed by them. After that Duan et al in [9] proposed a cluster based outlier detection algorithm which can detect both single point outliers and cluster-based outliers, and can assign each outlier a degree of being an outlier. Zhuo et al in [15] presented an outlier mining algorithm based on dissimilarity (OMABD), which detected outliers by comparing the dissimilarity degree with dissimilarity threshold. A dissimilarity based method is used for improving the efficiency of k-means algorithm in this study.

3. presented model

To improve the efficiency of the K-means algorithm, at first the dataset should be investigated for specifying and detecting the outliers. After that, the dataset should be divided into two subsets of normal object and outliers. Then, the clustering process should be performed separately for normal object and outliers. The normal object are clustered using the mentioned steps of the k-means algorithm shown in figure 1, and finally, the outliers are assigned to the nearest cluster according to the Euclidean distance criterion and calculated cluster centers from the previous steps.

3.1 Outlier detection in dataset

In this paper, the ODBD algorithm is presented for specifying and detecting the outliers which is based on the dissimilarity of objects. According to this algorithm, a value is calculated for all of the dataset objects which is called the dissimilarity degree. The objects that their degree is higher than the threshold value are considered as the outlier objects.

Assume that a data set DS is defined in the form of: DS= (D,A) in which $D=\{d_1,d_2,\dots d_n\}$ is the set of n objects and $A=\{a_1,a_2,\dots a_m\}$ is the set of attributes with the order of m. The dissimilarity degree of two objects d_i, $d_j \in D$ on the attribute of $f \in A$ is calculated as following:

$$ad_{ij}^{f} = \frac{\left(\left|d_{if} - d_f\right| - \left|d_{jf} - d_f\right|\right)^2}{d_{\max f} - d_{\min f}} \quad (3)$$

Where dif and d_{jf} are the values of attribute f in the objects i and j, respectively. d_f, d_{max} and d_{min} are the average, the maximum and the minimum value of attribute f on all objects of the dataset, respectively.

The dissimilarity degree of two objects can be obtained from the average of these objects dissimilarity on each of attributes, as following:

$$od(i,j) = \frac{\sum_{ak=1}^{m} ad_{ij}^{ak}}{m} \quad (4)$$

Where ad_{ij}^{ak} is the dissimilarity value of the objects i, j on the a_k-th attribute.

According to the relations (3) and (4), the dissimilarity matrix d_m, which is an order N square matrix, is created to calculate the dissimilarity of each object with respect to the other objects. By adding the row elements of this matrix, the objects synergic dissimilarity matrix is made which shows the dissimilarity degree of each object with respect to all other objects of the data set. For the sake of simplicity and simplification of comparisons, the synergic dissimilarity degree matrix is normalized and then, the average of elements of this matrix with an impact factor value in the range of [0,1] is considered as the threshold similarity value. The ODBD pseudo-code algorithm is shown in figure 3.

ODBD algorithm

input:
data set D = { $d_1, d_2, \dots d_n$ }, where d_i =data points, n= number of data points
impact factor value IFV ∈ [0,1]
output:
outlier and normal objects
step 1:
step 1-1:
for each data object d_i from D
  for each data object d_j from D
    calculate od(i,j) by using equation (3) and (4)
    dm(i,j)=od(i,j)
  end for
end for
step 1-2:
for each row r_i in dm
  calculate sum of elements and assign in sd_i
end for
calculate d_{max} = maximum value in sd
step 1-3:
for each data value sd_i in sd
  $dd_i = (d_{max} - sd_i) / d_{max}$
end for
td=mean(dd) *IFV
step 2:
for each data object d_i from D
  if dd_i < td
    assign d_i to outlier objects
  else
    assign d_i to normal objects
  end if
end for

Figure 3. The pseudo-code of ODBD algorithm

3.2 Improving the Clustering process with the ODBD-k-means algorithm

Selecting the initial values in the normal k-means algorithm is completely random. Thus, the existence of the outliers results in the cluster centers getting distance from real position and consequently decreasing the accuracy of this algorithm. In the presented algorithm it is attempted that the outliers do not affect the process of selecting the clusters center. For this purpose, after separating the data set objects using ODBD, the clustering is done during two phases. In the first phase, the normal process of k-means algorithm is used to cluster the normal object. In this phase, because of data set being pruned by the ODBD algorithm and the use of normal object, the centers and the objects are determined in a more accurate way. In the second phase, we use the centers obtained from the previous phase and with iteration, we calculate the distance between each of outliers and these centers. Then, each outlier is assigned to the nearest cluster. The Euclidean distance criterion is used for the calculation of this distance. The presented algorithm pseudo-code is illustrated in figure 4.

ODBD-K-Means algorithm

input:
data set D = { $d_1, d_2, \dots d_n$ }, where d_i =data points, n= number of data points
k = number of cluster centers
output:
k clusters with their centers
step 1:
find outliers and normal objects by using ODBD algorithm
step2:
step 2-1:
randomly select k data object from normal objects as initial cluster centers.
step 2-2:
repeat step 2-3 to step 2-4 till no new cluster centers are found
step 2-3:
calculate the distance between each data object d_i (1<=i<=size(normal object)) and all k cluster centers C_j (1<=j<=k) and assign data object d_i to the nearest cluster.
step 2-4:
for each cluster j (1<=j<=k), recalculate the cluster center.

step 3:
for each data object d_i from outliers
  step 3-1
  calculate the distance of d_i to all k final cluster centers C from step 2 by using euclidean distance
  step 3-2
  find the closest center c_j and assign d_i to the cluster with nearest center C_j
end for

Figure 4. The pseudo-code of ODBD-k-means algorithm

4. Results and discussion

To evaluate the algorithm presented in this study, the algorithm is implemented using MATLAB 2010 programming software and the results are compared with the k-means algorithm. The Iris, Bupa and Glass data sets from UCI are used in the experiments. The Iris data set is a categorization of iris flowers in which, there exists three different classes of iris and each class contains 50 objects. Each object has 4 attributes. In the Bupa data set, 345 objects exist each having 6 attributes. The attributes are gathered from blood tests concerning the diagnosis of liver hampering caused by irregular drink of alcohol. Each object of this data set is the record of a male person. In the glass data set, 214 objects exist and each object has 9 attributes and this set has 6 classes. The properties of these data sets are listed in table 1.

Table 1. The datasets and their properties

dataset	#objects	#features	#clusters
Iris	150	4	3
Bupa	345	6	2
Glass	214	9	6

Selecting the impact factor and consequently the similarity threshold value is very important to detect and determine the number of outliers. This factor value might be different for each data set. According to the ODBD algorithm, if the impact factor assumed to be zero, the number of outliers would be zero. This means that the ODBD-k-means algorithm changes to the normal k-means algorithm. For calculating the algorithm accuracy, we can use accuracy indicator which is obtained using to the following relation:

$$Accuracy = \frac{True\ Positive}{TruePositive + False\ Postive} \quad (5)$$

Because the number of normal object affects the selection of cluster centers and also the proposed algorithm like the k-means algorithm calculates the initial centers by random selection of objects, the results of each execution of this algorithm may not the same. For different values of impact factor in the range [0,1], the presented algorithm is executed for 100 times and according to the relevant number of outliers, the average value of results is considered as the algorithm accuracy. These results are presented in table 2, for Iris data set.

Table 2. Results of ODBD-k-means algorithm on the iris

Impact factor	#outlier	Accuracy
0.0	0	88.98
0.1	1	87.57
0.2	3	88.75
0.3	4	89.31
0.4	**5**	**90.36**
0.5	12	89.60
0.6	17	87.69
0.7	26	87.17
0.8	32	86.03
0.9	45	86.51
1.0	54	85.89

According to this table, it is obvious that the algorithm accuracy is dependent to the number of outliers. If we select zero as the value of the impact factor, the presented algorithm would be equivalent to the normal k-means algorithm. The real outliers of the Iris data set are the points that are separated with an impact factor of 0.4. By running the ODBD algorithm and selecting the most suitable impact factor for data set, the number of outliers in each data set is obtained. The results are shown in table 3.

Table 3. The number of outliers in data sets

dataset	Impact factor	#outlier objects
Iris	0.4	5
Bupa	0.8	12
Glass	0.6	8

Results of algorithm accuracy on the data sets are shown in table 4.

Table 4. The results of algorithms on datasets

Dataset	K-means	ODBD-K-means
Iris	88.98	**90.36**
Bupa	52.43	**53.72**
Glass	45.79	**47.09**

Figure 5 depicts the algorithm accuracy results on a diagram. According to this diagram, effect of the presented algorithm on the standard data sets can be observed.

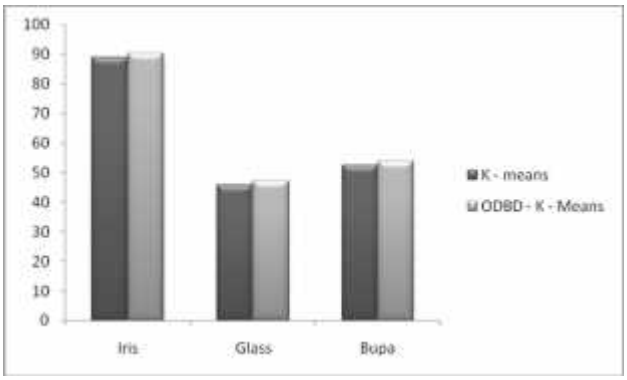

Figure 5. The results of algorithms on standard datasets

The k-means algorithm has a good accuracy on the Iris data set, because of a suitable distribution and structure of objects in data set. The existence of the outliers in the Bupa and Glass

data sets, has an undesirable effect on the selection of centers and objects and it consequently leads to reduction in the accuracy of the k-means algorithm. After temporarily removing this objects and using the presented algorithm, the retrieval accuracy on these three data sets have been improved.

5. Conclusion

In this paper, a new algorithm is presented based on the k-means algorithm and dissimilarity of the objects. In the presented model, the data sets were analyzed to specify the outliers. By detecting these objects, the data set pruned and the outliers and normal objects were separated from each other. Then, the normal objects were grouped using the k-means algorithm. Finally, using of the final cluster centers of previous stage and by calculating the distances between the outliers and the final cluster centers, these objects were assigned to the nearest cluster center, separately. The experimental results on the standard data sets showed that the presented algorithm probes the data sets with a better accuracy for finding better results. Despite the good results of the presented algorithm, different results were observed at different executions of the algorithm and that is due to the random selection of the initial points. In the future studies, it is possible to present better solutions for this challenge.

6. REFERENCES

[1] Breuing, M., Kriegel, H., Ng, R., and Sander J. 2000. LOF: identifying density-based local outlier factor. In Proceedings of ACM SIGMOD International Conference on management of Data., 29(2), pp. 93-104.

[2] Fabrizio, A., Stefano B., and Clara P. 2006. Distance-based detection and prediction of outliers. IEEE Transaction on Knowledge. And data Engineering., 18(2), pp. 145-160.

[3] Frogy, E. 1965. Cluster analysis of multivariate data: efficiency vs interpretability of classification. Biometrics., 21(3), pp. 768-779.

[4] Guha, S., Rastogi, R., and Shim, K. 1998. An efficient clustring algorithm for large database. In Proceedings of the ACM SIGMOD Conference., 27(2), pp. 73-84.

[5] Guha, S., Rastogi R., and Shim K. 2000. A Robust clustring algorithm for categorical attributes. In Proceedings of the third IEEE International Conference on Data Mining., 25(5), pp. 345-366.

[6] He, Z., Xu X., and Deng, S. 2003. Discovering cluster-based local outlier. Pattern Recognization Letters., 24(9), pp. 1641-1650.

[7] Kao, Y., and Lee S.Y. 2009. Combining k-means and particle swarm optimization for dynemic data clustring problems. IEEE, International Conference on Intelligent Computing and Intelligent Systems., Shanghai, pp.757-761 .

[8] Kumar, M., and Verma, S. 2008. Data clustring in sensor network using art. 4th International Conference on Wireless Communication and Sensor Network, Allahabad, pp. 51-56.

[9] Lian D., Lida, X., Ying, L., and Jun, L. 2009. Cluster-based otlier detection. Annals of Operation Research, 68(11), pp. 151-168.

[10] Pelleg, D., and Moore, A. 2000. X-means: extending k-means with efficient estimation of the number of clusters. In Proceedings of ICML the Seventeenth International Conference on Machine Learning., San Francisco, pp. 727-734.

[11] Romaswany, S., Rastogi R., and Shim K. 2000. Efficient algorithm for mining outliers from large data set. In Proceedings of the ACM SIGMOD International Conference on Management of Data., Texas, ACM press, pp. 473-478.

[12] Tsai, C. F., Tsai, C. W., Wu, H. C. and Yang, T. 2006. A new data clustering approach for data mining in large databases. The 6th IEEE International Symposium on Parallel Architectures, Algorithms, and Networks., pp. 278-283.

[13] Wang, T., and Yang, Z. A location-aware-based data clustring algorithm in wireless sensor networks. 2008. IEEE, 11th International Conference on Communication Systems., pp. 1-5.

[14] Wong, A. K. C., and Li, G. C. L. 2008. simultaneous pattern and data clustring for pattern cluster analysis. IEEE Transaction on Knowledge and Data Engineering., 20(8), pp. 911-923.

[15] Zhou, M., and Chen, X. 2011. an outlier mining algorithm based on dissimilarity. International Conference on Environmental Science and Engineering., pp. 810-814.