Available Online at www.ijcsmc.com

International Journal of Computer Science and Mobile Computing

A Monthly Journal of Computer Science and Information Technology

ISSN 2320–088X

IJCSMC, Vol. 2, Issue. 7, July 2013, pg.219 – 223

RESEARCH ARTICLE

Modified K-Means for Better Initial Cluster Centres

Kalpana D. Joshi[1], P.S. Nalwade[2]
[1]Department of Computer Science, SGGSIE&T, Nanded, India
[2]Department of Computer Science, SGGSIE&T, Nanded, India

[1] kalpanadjoshi22@gmail.com; [2] psnalwade@yahoo.com

Abstract— ***The k-means clustering algorithm is most popularly used in data mining for real world applications. The efficiency and performance of the k-means algorithm is greatly affected by initial cluster centers as different initial cluster centers often lead to different clustering. In this paper, we propose a modified k-means algorithm which has additional steps for selecting better cluster centers. We compute Min and Max distance for every cluster and find high density objects for selection of better k.***

Key Terms: - k-means; clustering; data mining; initial cluster centers; density objects

I. INTRODUCTION

The k-means algorithm is a well-known partition based clustering algorithm. It is used in real world applications such as marketing research, image processing, data mining etc. to cluster very large data sets due to its efficiency and ability to handle numeric and categorical variables that are ubiquitous in real databases. For the traditional K-means algorithm, initial cluster centers play a key role in the performance. Different initial cluster centers often lead to different clustering, and thus provide unstable clustering results. Several improved methods are proposed to avoid such sensitivity.

One of the most popular heuristic algorithms for k-means is Lloyd’s algorithm [1], which initially chooses k centers randomly. For each input point, the nearest center is identified and points that choose the same center belong to the same cluster. Now new centers are calculated for the clusters. Each input point identifies its nearest center and so on. This process is repeated until no changes occur. The process of identifying the nearest center for each input point and recomputing centers is referred to as iteration. The number of iterations taken by Lloyd’s algorithm is unknown. This algorithm may converge to a local minimum with an arbitrarily bad distortion with respect to optimal solution. Thus, the K-means algorithm suffers from the well-known problem of locally optimal solutions. Furthermore, the final partition is dependent upon the initial configuration, making the choice of starting partitions all the more important. For better selection of k initial points, density based approaches proposed in literature [6][8][9]. This paper presents an overview on several approaches of seeding k points as initial centers and provides an improved method for the k-means problem.

II. K-MEANS ALGORITHM

The traditional K-means algorithm is based on decomposition, most widely used in data mining field. The concept is use K as a parameter, Divide n object into K clusters, to create relatively high similarity in the cluster, relatively low similarity between clusters. And minimize the total distance between the values in each cluster to the cluster center. The cluster center of each cluster is the mean value of the cluster. The calculation of similarity is done by mean value of the cluster objects. The distance between the objects is calculated by using Euclidean distance. The closer the distance, bigger the similarity of two objects, and vice versa.

Algorithm: k-means. The k-means algorithm for partitioning, where each cluster's center is represented by the mean value of the objects in the cluster. [11]

```
Input:  k: the number of clusters,
        D: a data set containing n objects.
Output: A set of k clusters.
Method:
(1) arbitrarily choose k objects from D as the initial cluster centers;
(2) repeat
(3)     (re)assign 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) until no change.
```

K-means usually chooses sum squared error criterion function based on Euclid distance as its clustering criterion function. If the difference among all clusters is obvious, that is, the similarity among the clusters is very obscure, and then the sum squared error criterion function is relatively effective. On the contrary, it will lead to the problem that the large cluster should be further divided.

III. Performance Analysis

A. Advantages:

- K-means is a classical algorithm to resolve clustering problems simplify and quickly and it is easy to implement and understand.
- Better efficiency in clustering high dimensional data.
- Complexity of k-mean algorithm is *O(ntk*) where *n* is number of objects, *t* is number of iteration and *k* is number of cluster.

B. Disadvantages:

K-means only can be used under the situation that the average value has been defined. This may not suit some applications, such as mobile objects clustering, data concerned about classified attributes.

- In k-mean algorithm user need to specify the number of cluster that is k.
- It's sensitive to the initial centroids and change in initial centroids can lead to different clustering results with different initial value.
- k-means is not fit to non-convex cluster, or big difference on size. Besides, it's sensitive to noisy data and isolated points data, a little data like this can make huge effects on average values. In the other way we can say k-mean algorithm is unable to handle noisy data and outliers.

IV. Selecting Cluster Centers

This paper mainly focuses on two important issues regarding initial cluster centers for k-means algorithm

- Specify the number of clusters that is the numeric value k=2,3,4 etc.
- Explicitly choose k objects from data as initial cluster centers.

Both issues are challenging and sensitive in K-means algorithm as they directly affect the algorithm performance and accuracy. Traditionally for k-means algorithm, the number of clusters k is usually selected by a lot of experiments and the initial clustering centers are usually selected randomly. This selection way is sensitive to noise points and isolated points, a little data like this can make huge effects on average values.

Specifying the right number k of clusters for k-means clustering algorithm is often not obvious and choosing k automatically is a hard algorithmic problem [10] presented an improved algorithm for learning k while clustering. The proposed G-means algorithm [10] repeatedly makes decisions based on a statistical test for the data assigned to each center. If the data currently assigned to a k-means center appears to be Gaussian then it represents that data with only one center. However if the same data do not appear to be Gaussian use multiple centers to model the data properly. This proposed method can be used for finding right number of clusters and the location of genuine cluster centers for moderately high dimensions.

Recently, there are several approaches proposed for selecting initial cluster centers which are based on traditional clustering methods like density based, random partitioning, graph based etc. Some of different methods are also introduced for better k center selection**.**

A. Density based Method

Density based method is the most commonly used in many initial cluster center defining algorithm where the high density object within the region is considered as initial cluster center. Xuhui CHEN, Yong XU. [6] Use the traditional data space density approach which identifies the high-density of objects within the regions as the initial cluster center that greatly improve the efficiency of clustering algorithm. Now the fundamental task is to accurately define high density region. Density parameters τ presents the unmarked statistics of spatial data-object x_i as center and *r* as the radius which contained the number of data-objects computed for building high-density areas set D. For K initial cluster centers firstly select the greatest density data- object from the set D, adding to the set of cluster center; secondly, successively find out k-1 data-objects, ensuring the largest distance between the k initial cluster centers. The above Density based method for initial cluster center eliminates the randomness of traditional k-means algorithm leads to clustering result instability.

Another density based method is proposed by Baolin Yi, Haiquan Qiao, Fan Yang [9] for the initial center point algorithm where Gaussian function is used to meet the global consistency of feature clustering. The basic idea of the algorithm is that we can select the greatest density point as the initial center point firstly from the sample transaction database, and then determine the second initial center using the same method from dataset that delete the first point and its neighborhood, this process continues until the initial set M contains k points. The proposed method for initial cluster center gives superior result over traditional K-means when the neighborhood radius, adjacent coefficient coefR, the experience-values are correctly defined for density calculation.

In the improved algorithm introduced by [6], the concept of distance along with the density-based clustering is used to select the initial cluster centroids, this selection is more in line with the actual distribution of data sets. For the K-means algorithm, it's more representative to choose *k* predetermined centroid which is farthest from each other than the random centroids. But in the real-world, dataset often exist outliers, what's the worse; it may lead to clusters of poor quality. Generally, the dataset where the high density object area divided by the low density object area, so the data object in the low density area must be considered. In order to eliminating this phenomenon, this algorithm uses the mutual farthest data object which lies in the high density area as the initial centroids.

In order to calculating the density area to which the object belongs. At first, the improved algorithm defines the traditional density parameter □: the neighborhood within a radius □ of a given object x_i is called the □-neighborhood of the objects x_i . If the □-neighborhood of the object x_i contains at least a minimum number, *Minpts*, of objects, then the object *xi* is the core object, and it means the object *xi* located in the high density area, nonetheless, in the low density area called outliers. At the same time, improved algorithm deletes the outliers from the dataset, and can get a dataset *D* located in the high density area. Mainly, algorithm selects two objects with farthest distance and eliminates recursively till the number of object in *center* reaches threshold *k*.

B. Random partition based Method

Bradley and Fayyad [5] present a technique based on partitioning approach for initializing the k-means algorithm. They begin by randomly breaking the data into 10, or so, subsets. They then perform a k-means clustering on each of the 10 subsets, all starting at the same set of initial seeds, which are chosen using Forgy's method. The result of the 10 runs is 10k centre points. These 10 k points are then used as inputs of k-means algorithm and the algorithm run 10 times, each of the 10 runs initialized using the k-final center locations (known as centroid) from one of the 10 subset runs. The resulting k-center locations are used to initialize the kmeans algorithm for the entire datasets.

Generally, we cannot avoid the possibility of points from the tails appearing in the subsample. Therefore, the estimate is unstable due to elements of the tails appearing in the sample. In order to overcome this problem Xiaoping Qing, Shijue Zheng [8] draw multiple subsamples(say *h*), and all subsamples are clustered, so as to produce *h* estimates of the true cluster centers. Initially the proposed algorithm chooses *h* small random sub-samples of the data, , *i=1,…,h*. If there are empty clusters finally, we will re-assign initial centers and re-cluster the sub-sample. The sets , *i=1,…,h,* are these clustering solutions over the sub-samples which come from the data set *DM*. *DM* is then clustered via KMeans initialized with *DM i.* which produce a solution *PMi*. Then we choose the *PMi* as initial point having minimal distortion over the set *DM*. Proposed method can avoid the empty clusters problem that plagues traditional K-Means which is likely to lead to a "bad" solution.

C. Graph based approaches

A. M. Fahim, A. M. Salem, F. A. Torkey, G. Saake and M. A. Ramadan [2] presents a novel approach for initial cluster center selection based on BIRCH algorithm. The main idea of this algorithm is to compress the dataset into finite number of representative. Each representative is the mean value of some data points form a small cluster. The algorithm compress the data set of size N into smaller data set of size k*m; where k is the required number of partition for each block, m is the number of blocks. This process has been done at the first phase. In the second phase, apply the k-means on the compressed dataset to get the k representative points that will be the initial starting points for the k-means on the full dataset. The idea of compression of dataset comes from the BIRCH algorithm.

Lan Huang, Shixian Du, Yu Zhang, Yaolong Ju, Zhuo Li [7] Proposed an approach based on kruskal's algorithm used to build Minimum spanning tree (MST), then obtains the initial clustering centers with the help of the nodes of this MST. For any given data set X = $(x_1, x_2, . . . , x_n)$ and k. The goal is to divide the data objects of X to k clusters. In this approach, Euclidean distance among the data objects as the edge weights between any two objects. Therefore an undirected weighted connected graph G(X) is generated, which represents the data set X. By using the famous Kruskal algorithm, the Minimum Spanning Tree of the undirected weighted graph can be generated. According to the weights value ranging from large to small, k=1 edges of the Minimum Spanning Tree should be deleted. Then k connected subgraphs are obtained, and the averages of each connected subgraph define the initial clustering center. After that, the original kmeans algorithm to cluster the data set X can be used to get the final clustering results, which include the ultimate iterating times of algorithm, k clustering centers, and the objective function value.

Shou Qiang Wang and Data Ming Zhu [4] present an algorithm with expected approximate factor at most 2 and restrict the center point to be in the original set of points. The expected 2-approximation algorithm takes Point set P,α as input parameters and then perform sampling. Using each k-subset of sample points as centers, calculate the cost of clustering with respect to all the original input points and retrieve the k-subset that results in the minimum cost. The above algorithm obtains an expected approximate ratio at most 2 to the optimal solution with probability at least 1/2. However, the result is calculated by means of enumerating all k-subset of samples. It is obvious that the algorithm will take much time and be unpractical if the k is enough large. For good approximation, the initial k centers for k-means can be selected from S instead of P with a set followed by the standard k-means algorithm (Lloyd's algorithm). An expected 2-aprroximation can be got, if we use the centers belonging to different optimal sub cluster.

V. PROPOSED WORK

Limitations of k-means algorithm may overcome by properly selecting initial k. Firstly user has to specify numeric value of k=2,3,4 etc.. and then randomly select k objects from data as initial centers. Now we can take k=2 as default value and randomly select two objects from data as first initial centers. Depending on data distribution we can increment value of k by splitting previously selected centers and for splitting we apply some conditions.

Let $X=\{X_1,X_2,...,X_i\}$ be a data objects with k cluster centers $C=\{C_1,C_2,...,C_k\}$. *NPT* and *MPT* are two sets having densely connected objects.

Modified k-means

```
Step 1. Select $k=2$ initial cluster centers $C_i$ randomly from data $X_i$ .
        Repeat following steps for every cluster center.
Step 2. Find Euclidean distance of each data objects $X_i$ from cluster centers and assign objects to cluster with minimum distance.
Step 3. Find Min_dist and Max_dist distance along with corresponding nearest object min_obj and farthest object max_obj.
Step 4. Calculate two sets of objects NPT and MPT contain densely connected objects to min_obj and max_obj
        within distance: avg_dist= (Min_dist+Max_dist)/3
Step 5. Selecting K
   i)$NPT_i \cap MPT_{\mathrm{i}}=\Phi$
     ii) $NPTi \cap NPT_{\mathrm{j}}=\Phi$ and $MPT_i \cap MPT_{\mathrm{j}}=\Phi$
     If (i) valid then split Ci and if both (i) and (ii) valid split both center and assign new center as min_obj and max_obj of corresponding cluster.
     If either condition is valid then goto step 2.
Step 6. Find mean for every cluster.
Step 7. If no change in cluster centers then exit.
```

The above Modified k-means algorithm has additional steps in traditional k-means algorithm for better cluster center selection. We use Euclidean distance for assigning object to proper cluster by using these calculated distances and we find nearest *min_obj* and farthest *max_obj* objects from cluster center and record its minimum *Min_dist* and maximum *Max_dist* distance values. For selecting better cluster centers we use two sets of densely connected objects. The *NPT* set contain objects within avg_dist from min_obj and *MPT* set contain objects within *avg_dist* distnace from *max_obj*.

For splitting previous cluster centers we use two conditions given in step 5.If cluster center satisfy both conditions means both cluster centers has high density *min_obj* and *max_obj* and hence we split both centers and assign *min_obj* and *max_obj* as new centers. If cluster center satisfy either condition then split that cluster center only into two cluster centers.

The proposed method can give effective results for k-means algorithm when data is distributed in well separated cluster format as it can decide value of *k* properly.

VI. CONCLUSIONS

The performance of k-means algorithm greatly depends on initial cluster centers. Selection of appropriate value of k and cluster center objects is a challenging issue. The proposed method can choose better value of k by splitting and select high dense object as cluster centers. So they can provide efficient clustering results for k-means algorithm.

REFERENCES

[1] Rafail Ostrovsky, Yuval Rabani, Leonard Cschulman and Chaitanya Swamy, "The effectiveness of Lioyd-Type methods for the k-means," In 47th IEEE Symposium on Foundation of Computer Science, 2006.

[2] A. M. Fahim, A. M. Salem, F. A. Torkey, G. Saake and M. A. Ramadan, " An efficient k-means with good initial starting points," Georgian Electronic Scientific Journal: Computer Science and Telecommunications, 2009,Vol. 2, No. 19, pp. 47-57

[3] Chen Zhang, Shixiong Xia et al, "K-means clustering Algorithm with Improved Initial Center," Second International Workshop on Knowledge Discovery and Data Mining, wkdd, pp.790-792,2009

[4] Shou Qiang Wang and Data Ming Zhu (2008), "Research on selecting initial points for k-means clustering", IEEE research paper, page-267

[5] P. S. Bradley, U. M. Fayyad, "Refining initial points for k-means clustering," In: Proc. 15th Internat. Conf. on Machine Learning, 1998

[6] Xuhui CHEN, Yong XU. "K-means Clustering Algorithm with Refined Initial Center," 2nd International Conference on Biomedical Engineering and Informatics, Tianjin, 2009, 1-4.

[7] Lan Huang, Shixian Du, Yu Zhang, Yaolong Ju, Zhuo Li, "K-means Initial Clustering Center Optimal Algorithm Based on Kruskal," Journal of Information & Computational Science 9: 9 (2012) 2387-2392

[8] Xiaoping Qing, Shijue Zheng "A new method for initializing the K-means clustering algorithm," Second International Symposium on Knowledge Acquisition and Modeling, Wuhan, 2009, 41-44

[9] Baolin Yi, Haiquan Qiao, Fan Yang, Chenwei Xu, "An Improved Initialization Center Algorithm for K-means Clustering," IEEE 2010.

[10] G. Hamerly and C. Elkan, "Learning the k in k-means," in Proc. 17th NIPS, 2003.

[11] J. Han, M. Kamber, Data Mining Concepts and Techniques, Morgan Kaufman, San Francisco, 2001.