ISSN:2249-5789

Pallavi Purohit et al , International Journal of Computer Science & Communication Networks,Vol 4(3),125-129

An Efficient Approach towards K-Means Clustering Algorithm

Pallavi Purohit
Department of Information Technology, Medi-caps Institute of Technology, Indore
purohit.pallavi@gmail.com

Ritesh Joshi
Department of Master of Computer Application, Medi-caps Institute of Technology, Indore
riteshjoshi.indore@gmail.com

ABSTRACT

K-Means clustering algorithms are used in various practical applications countless times. Original K-Means algorithm select initial centroids randomly it generates unstable cluster as the value of object in cluster depend on the selection of initial cluster means which is done by random selection of objects. The number of times different selection of initial centroids will give number of different clusters with different accuracy.

The algorithm used in base paper's clustering method eliminates the deficiency of K-Means. It first computes the initial centroids k according to the requirements of users and then gives better, effective and good cluster. To improve accuracy it generates stable clusters. It also reduces the mean square error and improves the quality of clustering, but this algorithm has large execution time, which makes it expensive. As this algorithm requires storing lot of calculations it requires lot of space

The proposed algorithm that is the new efficient approach towards K-Means algorithm combines the method of both the algorithm it systematically chooses the initial centroids for the procedure in such a way that it reduces mean square error without a high increase execution time. This algorithm requires less space than the base paper's algorithm.

Keywords

Cluster analysis, Centroids, K-mean

1. INTRODUCTION

Data Mining involves the process of extracting interesting and hidden patterns or characteristics from very huge datasets and using it in decision making and prediction of future behavior. This improves the need for effective and efficient analysis methods to make use of this information. One of these tasks is clustering. Clustering is the process of grouping the data into clusters or classes so that objects within a cluster may have more similarity when compared to one another, but are very different to objects in other clusters. Clustering deals with collection of similar data objects within the similar cluster and are dissimilar to the objects in different clusters. Clustering can be known as physical object or set of abstract grouped into classes of similar objects.

Many clustering algorithms have been developed. Clustering is categorized into partition method, hierarchical method, density based method, grid based method, and model based methods.

In partitioning method first the number of cluster (k) to be created is specified. Initially k clusters are obtained by some method than an iterative approach is used to relocate the object into correct cluster.

A hierarchical method can be described as being either agglomerative or divisive, depend on formation of hierarchical decomposition. To suppress the rigidity of dividing or grouping, the quality of hierarchical agglomeration can be enhanced by analyzing object connections at each hierarchical partitioning.

A density based method clusters objects based on the density notation. It creates cluster according to the density of neighborhood objects or according to some density function.

A grid based method first quantizes the object space into number of cells which make a grid structure, and then perform clustering on the grid structure. A model-based method hypothesized a model for each of the clusters.

Partition clustering algorithm attempts to determine k partitions from a collection of nd-dimensional objects (vectors). The aim of these methods is to create large variance between the clusters and reduce the variance within the cluster. K-Means is the most common partition algorithm used for clustering. Variation in K-Means makes it more effective and efficient if the work towards better method is applied.

In Section-2 the process of original K-Means is explained. In section-3 the previous method of enhancing the K-Means is

explained. In Section-4 discussion on a new approach of variation of K-Means is done. In section Section-5 the performance study of K-Means and variations of K-Means is explained. Finally Section-6, Section-7 and Section-8 contain the conclusion, future work and references respectively.

2. K-MEANS ALGORITM

K-Means is the simple unsupervised learning algorithm that can solve the well-known problem of clustering. The process involves simple way of creating clusters assume the value be k, of provides data samples. The cluster contains the value having similarity. Find k centroids for cluster. As different location creates different results so the centroids selection should be done in proper way. So, the best choice is to make them as far as possible.

Assume k values for initial means. Until there is not any change in any mean Use the estimated means for creating clusters of similar type samples. For next mean calculation, calculate K-Means for the similar data in the clusters. The mean of k clusters are calculated and defined as new mean. Repeat this work till no change in mean.

This is a normal original version of the K-Means procedure which is also act as a greedy algorithm for creating k clusters by partitioning the n samples so that the sum of the squared distances to the cluster centers is minimized.

2.1 Initialization

The K-Means algorithm creates k clusters of dataset based on similarity. To initialize K-means random selection of cluster means is done. The most common method is to select the k random objects from dataset as cluster means.

2.2 Algorithm

Input:

- k: Number of clusters to be created.
- D: A set of data to be clustered having n elements.

Output:

k clusters containing n elements.

1. Arbitrarily choose k objects from D as initial cluster center.
2. Repeat.
3. Reassign values to the cluster according to the data which is most accurate based on the mean value of the objects.
4. Update the cluster means by obtaining the mean of each cluster.
5. Until no changes.

3. REVIEW OF EXISTING ENHANCED CLUSTERING ALGORITHM:

In K. A. Abdul Nazeer, M. P. Sebastian's paper, "Improving the Accuracy and Efficiency of the K-Means Clustering Algorithm" [1] explains that the mean square error of clusters can be reduced by making changes in creation of initial centroids. If initial centroid is chosen systematically and properly than much better cluster are created. For better result two main tasks are done. Instead of initial centroids are selected randomly, the initial centroids are determined systematically to create stable cluster. When initial centroids are obtained it creates the clusters by adding the data points into the cluster according to the minimum distance between cluster centroid and the data point. By using a heuristic approach these clusters are subsequently fine-tuned, thereby improving the efficiency.

3.1 Initial Centroid Calculation

It first calculates the distance between each data point. Now find the pair of data points which has the minimum distance and add these point to a set and remove the points from the population. Now find the minimum distance between the data point in new set and the values in the population add that value to the set and remove it from the population. Repeat this process till a threshold value is achieved. Now create k sets in the same way. The initial centroids will be the mean of each set.

3.2 Algorithm

Input:
D = {d1, d2,.....,dn} // set of n items
Let k be the number of desired clusters
Output:
k set of clusters
Steps:
1. Set S = 1
2. Calculate the distance between each data point in D.
3.Find the pair of data points with minimum distance and then form data-point set Am (1<= S <= k) which had these two data-points; Remove these data points from the set D.
4. Find the minimum distance data point in D to the point in set Ap, Add it to Ap and remove it from D
5. Repeat step 4 up to the value in Am reaches 0.75*(n/k)
6.If S<k, then S = S+1, find a different pair of data points in between which the distance is the shortest, make data-point set Ap and delete from D, Go to step 4
7. For each data-point set Am (1<=p<=k) find the mean of data points Cp(1<=S<=k) in Ap, these means will be the initial centroids
8. Compute the distance of each data-point di (1<=i<=n) to all the centroids cj (1<=j<=k) as d(di, cj)
9. Find the closest centroid cj for every data-point di and add di to cluster j.
10. Set ClusterId[i]=j; // j: Id of the nearest cluster
11. Set Nearest_Dist[i]= d(di, cj)
12. For each cluster j (1<=j<=k), again calculate the centroid
13. Repeat
14. For each data-point di
    14.1 Distance from the nearest cluster centroid is calculated.
    14.2 The data point stays in the cluster if the distance is lesser than or equal to the nearest distance,
    14.3 else
    14.3.1For each cj (1<=j<=k) Compute the distance (di, cj);
    Endfor
    14.3.2 Put the data-point di to the cluster having the closest centroidCj

14.3.3 Set ClusterId[i] =j
14.3.4 Set NearestDist[i] = d (di, cj);
Endfor
15. Each cluster j (1<=j<=k), recalculate the centroids; until the stable cluster is achieved.

4. PROBLEM DOMAIN

Clustering is right now one of the core topics of research related to Data Mining. Though lots of research has been done, a lot remains to be still explored. Clustering is still in the primary stage of research. This is basically because clustering is unsupervised learning and so the work starts without any assumptions and so the scope for research increases. The main criteria of research are related to the followings:

- It reduces the mean square error of creating clusters
- Work can be done on to improve accuracy and generate better and stable clusters.
- It makes the algorithm simple and efficient.

To achieve the above mentioned criteria, it is necessary to explore a number of existing clustering algorithms and try to implement them on different data set. This study and research is must to design or improvise a new algorithm.

4.1 Need for Research

In the above clustering algorithm, the approach used for calculating initial centroids is very time consuming. It reduces the mean square error but increases the execution time of the algorithm. The processing time is very large. As it calculates the distance between each and every data point it require lot of space and lot of calculations.

Considering all this drawbacks and the benefits of the above algorithm the work is done to improve it. The improvement is applied in the direction of initial centroid selection in which calculation is reduced.

5. PROPOSED ALGORITHM

In proposed Algorithm, New and efficient approach is applied, for better result the drawback of the previous algorithms is reduced. The concept of change in initial centroid selection is used but in a way that execution time will not sacrificed. The cluster formed will be better and stable with reduced mean squared error. After initial centroids are selected, heuristic approach is used to fine-tune the clusters, hence improving the accuracy.

5.1 Concept Used

Let U is a data-point set. To create k clusters, the initial centroids are calculated systematically.
Instead of initial centroids are selected randomly, for creating stable cluster the initial centroids are determined systematically.
It first calculates the mean of all data, and then finds the distance between the mean and the data points.
Sort the data point according to distance calculated. Now divide them into k sets.
Take mean of each set which will be the initial centroids. Assign each data point to the cluster whose centroid is closest.
Now subsequently fine tuned the cluster by using some heuristic approach.

5.2 Algorithm

Our Proposed Algorithm is as follow:
Input:
D = {d_1, d_2,.....,d_n} // set of n items
Let k be the number of desired clusters
Output:
Let k be set of clusters
Steps:

1. Calculate the mean for each attribute in data set D.
2. Let O be the origin point. Assign the mean calculated in step 1 as attributes to point O.
3. Calculate distance between each data point and origin.
4. Sort the data points in ascending order of the value obtained in step 3.
5. Partition the sorted data points into k equal sets.
6. Calculate the mean of each set; take the mean as the initial centroids.
7. Compute the distance of each data-point d_i (1<=i<=n) to all the centroids c_j (1<=j<=k) as d(d_i, c_j)
8. Repeat
9. Find the closest centroid $c_j$for data points in di and assign d_i to cluster j.
10. Set ClusterId[i]=j. // j:Id of the closest cluster.
11. Set NearestDist[i]= d(di, cj).
12. For each cluster j (1 <= j <= k), again calculate centroids.
13. For each di,
    13.1 calculate the distance with current nearest centroid.
    13.2 The data point stays in the same cluster if the current nearest distance is less or equal.
    Else
    13.2.1 For every centroid cj (1<=j<=k) compute the distance d(di, cj).
    End for;
    Until the convergence criteria is met.

6. PERFORMANCE STUDY

6.1 Distance Calculation

Calculate Euclidean distance for similarity between two data points or the data point and the centroid.
X= (x_1, x_2…..,x_n),
Y= (y_1, y_2……,y_n)
d(X,Y) = sqrt ((x_1 –y_1)^2+ (x_2-y_2)^2….+(x_n-y_n)^2). (1)
Where,
X and Y are data points or centroid points.
x_1,x_2,….,x_n & y_1, y_2,…,y_n are attributes of data points.

d(X,Y) is distance between X and Y.

6.2 Mean Square Error

Mean Squared Error is calculated to know the accuracy of

algorithm by calculating the distance between the cluster centers and its data points.

Data is clustered in such a way that the squared-error distortion is minimized. The effectiveness of the algorithms analyzed and measured against this criterion. The mean squared-error distortion is defined as

$$d(V, X) = (d(v_1, X)^2 + d(v_2, X)^2 + ... + d(v_n, X)^2 \quad (2)$$

Where $X = \{x_1, x_2... x_k\}$ is the closest cluster center to a point in $V = \{v_1, v_2... v_n\}$ and n is the total number of points.

6.3 Comparison between Algorithms

The algorithms are compared on the Iris dataset. This dataset contains 150 instances. It has 4 attributes. The algorithm is compared on the basis of mean square error and execution time.

Table 1 Mean Squared Error performance

Dataset: Iris

Clusters	K-Means(MSE)	Existing enhanced K-Means(MSE)	Proposed algorithm(MSE)
5	69.542	53.48	52.99
10	50.52	40.07	33.93
15	29.95	26.45	25.66
20	25.91	20.15	20.63
25	21.34	17.57	20.9

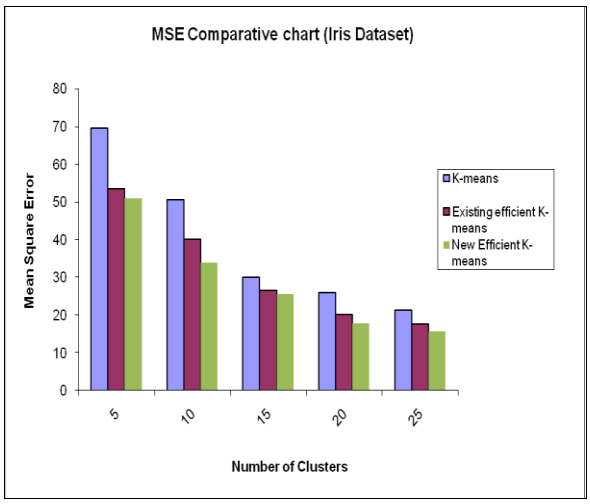

Figure 1. Mean Squared Error performance

The table 1 and figure 1 clearly shows the comparison preformed between the three algorithms based on mean square error. The data clearly shows that proposed algorithm gives much better results than the other two algorithms.

Table 2 Comparison between execution time of algorithms

Dataset: Iris

Clusters	K-Means(Time in ms)	Existing enhanced K-Means(Time in ms)	Proposed algorithm(Time in ms)
5	65.5	116.23	48.21
10	70.78	132.01	62.7
15	79.08	153.34	65.82
20	87.21	164.56	79.53
25	94.43	180.52	85.83

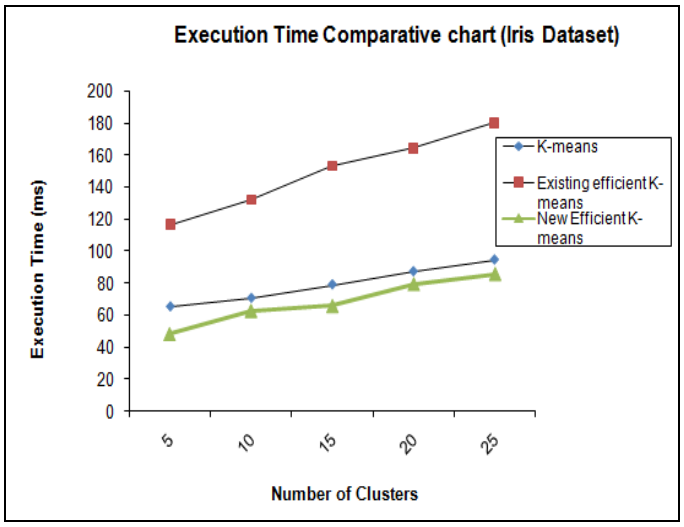

Figure 2 Comparison between execution time of algorithms

Figure 2 and Table 2 shows that the time required for the execution of algorithm or creation of cluster is maximum in the existing enhanced algorithm and minimum time required by the proposed algorithm.

7. CONCLUSION

The previous algorithm gives the better clusters than the original K-Means algorithm but it is very lengthy and time consuming process. The New proposed algorithm generate better cluster by reducing the mean squared error of the final cluster without large increment in execution time. It follows an organized way to generate initial centroids and apply an effective way for allocation of data points into the clusters. It reduces the mean square error without sacrificing the execution time as compared to the previous algorithm. Proposed algorithm gives more accuracy for dense dataset rather than sparse dataset it is clearly observed from the experiment.

8. FUTURE ENHANCEMENT

The work toward advancement of the algorithm every will be achieved, algorithm can be reproduced with some changes in good direction. Some of the suggestions are :

- It can start with categorical attributes.
- Work towards empty cluster can be done.
- Implement algorithm by first dividing the dataset into initial centroids using distance measures.
- Use these initial centroids and number of clusters K as input to create the final cluster using Euclidean distance.
- Final clusters that generated are accurate and more efficient.

9. REFRENCES

[1] K. A. Abdul Nazeer, M. P. Sebastian's paper, "Improving the Accuracy and Efficiency of the k-means Clustering Algorithm" Proceedings of the World Congress on Engineering, Vol I,2009.

[2] Fahim A.M., Salem A.M., "Efficient enhanced k-means clustering algorithm", Journal of Zhejiang University Science, 1626 – 1633, 2006.

[3] Dechang Pi, Xiaolin Qin and Qiang Wang, "Fuzzy Clustering Algorithm Based on Tree for Association Rules", International Journal of Information Technology, vol.12, No. 3, 2006.

[4] Fang Yuag, Zeng Hui Meng, "A New Algorithm to get initial centroid", Third International Conference on Machine Learning and cybernetics, Shanghai, 26-29 August,1191 – 1193, 2004.

[5] Friedrich Leisch1 and Bettina Gr un2, "Extending Standard Cluster Algorithms to Allow for Group Constraints", Compstat 2006, Proceeding in Computational Statistics, Physica verlag, Heidelberg, Germany,2006

[6] J. MacQueen, "Some method for classification and analysis of multi varite observation", University of California, Los Angeles, 281 – 297.

[7] Maria Camila N. Barioni, Humberto L. Razente, Agma J. M. Traina, "An efficient approach to scale up k-medoid based algorithms in large databases", 265 – 279.

[8] Michel Steinbach, Levent Ertoz and Vipin Kumar, "Challenges in high dimensional data set", International Conference of Data management, Vol. 2,No. 3, 2005.

[9] Parsons L., Haque E., and Liu H., "Subspace clustering for high dimensional data: A review", SIGKDD, Explor, Newsletter 6, 90 -105, 2004.

[10] Rui Xu, Donlad Wunsch, "Survey of Clustering Algorithm", IEEE Transactions on Neural Networks, Vol. 16, No. 3, may 2005.

[11] Sanjay garg, Ramesh Chandra Jain, "Variation of k-mean Algorithm: A study for High Dimensional Large data sets", Information Technology Journal5 (6), 1132 – 1135, 2006.

[12] Vance Febre, "Clustering and Continues k-mean algorithm", Los Alamos Science, Georgain Electonics Scientific Journal: Computer Science and Telecommunication, vol. 4,No.3, 1994.

[13] Zhexue Huang, "A Fast Clustering Algorithm to Cluster Very Large Categorical Data Sets in Data Mining".

[14] Jiawei Han, Micheline Kamber,"Data Mining – Concepts and Techniques", Morgan Kaufmann Publishers

[15] G. K. Gupta, "Introduction to Data Mining with Case Studies", PHI, 2006

[16] Introduction to data mining and knowledge discovery – Twocrow Corporation (www.twocrows.com/intro-m.pdf).

[17] http://wapedia.mobi/en/K-Mediods

[18] http://archive.ics.uci.eddu/ml/