# Far Efficient K-Means Clustering Algorithm

Bikram Keshari Mishra
Department Of Computer Sc. & Engineering
Silicon Institute of Technology
Bhubaneswar, India
Tel. : +91 - 9937994665
Email: bikrammishra2012@gmail.com

Nihar Ranjan Nayak
Department Of Computer Sc. & Engineering
Silicon Institute of Technology
Bhubaneswar, India
Tel. : +91 - 9861686330
Email: nihar@silicon.ac.in

Amiya Rath
Department Of Computer Sc. & Engineering
DRIEMS
Cuttack, India
Tel. : +91 - 9437577560
Email: amiyaamiya@rediffmail.com

Sagarika Swain
Department Of Computer Sc. & Engineering
Koustav Institute of Self Domain
Bhubaneswar, India
Tel. : +91 - 9937345754
Email: sagarika_bkm@yahoo.com

## ABSTRACT

Clustering in data analysis means data with similar features are grouped together within a particular valid cluster. Each cluster consists of data that are more similar among themselves and dissimilar to data of other clusters. Clustering can be viewed as an unsupervised learning concept from machine learning perspective. In this paper, we have proposed an effective method to obtain better clustering with much reduced complexity. We have evaluated the performances of the classical K-Means approach of data clustering and the proposed Far Efficient K-Means method. The accuracy of both these algorithms were examined taking several data sets taken from UCI [13] repository of machine learning databases. Their clustering efficiency has been compared in conjunction with two typical cluster validity indices, namely the Davies-Bouldin Index and the Dunn's Index for different number of clusters, and our experimental results demonstrated that the quality of clustering by proposed method is much efficient than K-Means algorithm when larger data sets with more number of attributes are taken into consideration.

## Categories and Subject Descriptors

H.2.8 [**Database Management**]: Database Applications – *Data Mining.*

## General Terms

Algorithms, Performance.

## Keywords

Cluster analysis, Cluster validity indices, K-Means clustering, Efficient K-Means and Far Efficient K-Means**.**

*ICACCI'12*, August 3-5, 2012, Chennai, T Nadu, India.
Copyright 2012 ACM 978-1-4503-1196-0/12/08…$10.00.

## 1. INTRODUCTION

Retrieving information faster from a group has always been an important issue. Several approaches have been developed for this purpose, one of them is data clustering. Therefore much attention is now paid to invent new fast and improved clustering algorithms. The main goal of clustering is that, the objects present in a group will be much similar to one another and different from the objects present in other groups.

The definition of what constitutes a cluster is always not well defined, and in most applications clusters are not well separated from each other hence, most clustering techniques represent a result as a classification of the data into non-overlapping groups. Clustering is often confused with classification, but there are some differences between the two. In classification, the objects are assigned to some already pre-defined class, whereas in clustering the classes are to be defined.

Learning valuable information from huge volume of data makes the clustering techniques widely applicable in several domains including artificial intelligence, data compression, data mining and knowledge discovery, information retrieval, pattern recognition and pattern classification , and so on.

In this paper, we have implemented the conventional K-Means [12] based clustering algorithm on various benchmark data sets. The result strongly depends on the initial selection of centroids. It is very difficult to compare the quality of the clusters produced (e.g. different initial partitions of K produce different results), and also very far data from the centroid may pull the centroid away from the real one. In order to curtail such difficulties and improve the clustering quality and efficiency especially on larger data sets, we have proposed a simple model known as Far Efficient K-Means clustering. Our goal is to analyze and compare the K-Means clustering method with our proposed scheme and check the quality of clustering results by using Dunn's separation index (DI) [5] and Davies-Bouldin's index (DBI) [4] respectively.

This paper is organized as follows: In Section II we briefly present the basic idea of cluster validity measures and two widely used validity indices such as DI and DBI used for determining the quality of results obtained from clustering. Section III presents the efficient and productive works done by several researchers in this relevant area. The K-Means clustering method is briefly discussed in Section IV and our proposed far efficient data clustering method is mentioned in Section V. Simulation and experimental results are shown in Section VI. Finally, Section VII concludes the paper.

## 2. CLUSTERING VALIDATION

Cluster validity issue by and large concerned with determining the optimal number of clusters and checking the fineness of clustering results. Assessment of clustering results is commonly referred to as cluster validation. Many different indices of cluster validity have been already proposed. In this section, we discuss briefly the Dunn's separation Index and Davies-Bouldin's Index which we have used in our proposed clustering algorithm for examining the soundness of clusters.

### 2.1 Dunn's Index

The main goal of Dunn's index (DI) measure [5] is to maximize the inter-cluster distances and minimize the intra-cluster distances. Dunn's index is defined as:-

$$DI(c) = \min_{i \in c} \left\{ \min_{j \in c, j \neq i} \left\{ \frac{\delta(A_i, A_j)}{\max_{k \in c} \left\{ \Delta(A_k) \right\}} \right\} \right\}$$

where,

$$\delta(A_i, A_j) = \min \left\{ d(\underline{x}_i, \underline{x}_j) \middle| \underline{x}_i \in A_i, \underline{x}_j \in A_j \right\}$$

$$\Delta(A_k) = \max \left\{ d(\underline{x}_i, \underline{x}_j) \middle| \underline{x}_i, \underline{x}_j \in A_i \right\}$$

*d* is a distance function, and $A_j$ is the set whose elements are the data points assigned to the $i^{th}$ cluster. The number of cluster that maximizes DI is taken as the optimal number of the clusters.

### 2.2 Davies-Bouldin's Index

Another measure, the Davies-Bouldin's index (DBI) [4] is a function of the ratio of the sum of within-cluster distribution to between-cluster separation.

The within $i^{th}$ cluster distribution is defined as:-

$$S_{i,q} = \left( \frac{1}{|A_i|} \sum_{\underline{x} \in A_i} \left\| \underline{x} - \underline{v}_i \right\|_2^q \right)^{1/q}$$

The between $i^{th}$ and $j^{th}$ separation is given by:-

$$d_{ij,t} = \left\{ \sum_{s=1}^{p} \left| v_{si} - v_{sj} \right|^t \right\}^{1/t} = \left\| \underline{v}_i - \underline{v}_j \right\|_t$$

where, $\underline{v}_i$ is the $i^{th}$ cluster center, and $(q, t) \geq 1$, and both *q* & *t* are integers and can be selected independently of each other. $|A_i|$ is the number of elements in $A_i$.

Next, define $R_{i,qt}$ which is given by:-

$$R_{i,qt} = \max_{j \in c, j \neq i} \left\{ \frac{S_{i,q} + S_{j,q}}{d_{ij,t}} \right\}$$

Finally, Davies-Bouldin's index is given by:-

$$DB(c) = \frac{1}{c} \sum_{i=1}^{c} R_{i,qt}$$

The objective is to minimize the DBI for achieving proper clustering.

The appropriate clustering algorithm and parameter settings heavily depend on the input data set taken into consideration. An ideal cluster can be said to be a set of data points that is more isolated and compact from other data points.

## 3. RELATED WORKS

A non-metric distance measure for similarity estimation based on the characteristic of differences [1] is presented and implemented on K-Means clustering algorithm. The performance of this kind of distance and the Euclidean and Manhattan distances were then compared. A new line symmetry based classifier (LSC) [2] deals with pattern classification problems. LSC is well-suited for classifying data sets having symmetrical classes, irrespective of any convexity, overlap and size. The shortcomings of the standard K-Means clustering algorithm can be found in the literature [3] in which a simple and efficient way for assigning data points to clusters is proposed. Their improved algorithm reduces the execution time of K-Means algorithm to a great extend. A simple and efficient implementation of K-Means clustering algorithm called the filtering algorithm [6] shows that the algorithm runs faster as the separation between clusters increases. The various types of clustering algorithms along with their applications in some benchmark data sets were surveyed in [7]. Several proximity measures, cluster validation and various tightly related topics were discussed. A new generalized version of the conventional K-Means clustering algorithm which performs correct clustering without pre-assigning the exact cluster number can be found in [8]. Based on the definition of nearest neighbor pair C. S. Li et al. [9] proposed a new cluster center initialization method for K-Means algorithm. In iterative clustering algorithms, selection of initial cluster centers is extremely important as it has a direct impact on the formation of final clusters. An algorithm to compute the initial cluster centers for K-Means algorithm was given by M. Erisoglu et al. [10] and their newly proposed method has good performance to obtain the initial cluster centers converges to better clustering results and almost all clusters have some data in it. An Efficient K-Means Clustering Algorithm for Reducing Time Complexity using Uniform Distribution Data Points [11] was proposed. The accuracy of the algorithm was investigated during different execution of the program on the input data points. Finally, it was concluded that the elapsed time taken by proposed efficient K-Means is less than K-Means algorithm.

## 4. K-MEANS CLUSTERING ALGORITHM

The K-Means Clustering algorithm is one of the simplest unsupervised learning algorithms that solve the well known clustering problem. In 1967, Mac Queen [12] firstly proposed the K-Means algorithm.

During every pass of the algorithm, each data is assigned to the nearest partition based upon some similarity parameter (such as Euclidean distance measure). After the completion of every successive pass, a data may switch partitions, thereby altering the values of the original partitions.

Various steps of the standard K-Means clustering algorithm is as follows: -

(1) The number of clusters is first initialized and accordingly the initial cluster centers are randomly selected.

(2) A new partition is then generated by assigning each data to the cluster that has the closest centroid.

(3) When all objects have been assigned, the positions of the K centroids are recalculated.

(4) Steps 2 and 3 are repeated until the centroids no longer move any cluster.

The main objective of K-Means is the minimization of an objective function that determines the closeness between the data and the cluster centers, and is calculated as follows:

$$J = \sum_{j=1}^{K} \sum_{i=1}^{N} \left\| d(X_i , C_j) \right\|$$

where, $\left\| d(X_i , C_j) \right\|$ is the distance between the data $X_i$ and the cluster center $C_j$.

The downside of K-Means algorithm is that, the result of clustering mostly depends on the initially selected centroids. Spherical data sets cannot be efficiently clustered using K-Means. And only numerical values attributes can be ably clustered.

# 5. PROPOSED CLUSTERING METHOD

In this proposed method of clustering, the K-Means algorithm is slightly customized and is used more effectively than its normal mode of implementation The Far Efficient K-Means algorithm is outlined below.

## 5.1 Far Efficient K-Means Algorithm

(1) Initially the distance between each pair of data in the data set is computed and the farthest pair of data (say $d_1$ and $d_2$) are picked, which will be treated as initial cluster center.

(2) Determine the data which is nearest to the center $d_1$ and add it to $d_1$ cluster. Remove the data belonging to this group from the data set.

(3) This procedure is repeated until the number of elements present in the $d_1$ cluster reaches a calculated threshold value 50% of (N/K). This value is chosen so as to ideally accommodate a reasonable amount of data in its respective cluster, regardless of any input dataset. Then find the arithmetic mean of $d_1$ cluster to obtain its centroid $c_1$.

(4) Repeat steps (2) and (3) for cluster $d_2$ to obtain its centroid $c_2$.

(5) The next step is to select the third cluster center. For this, we pick a data (say $d_i$) such that:-

$$\max ( \min ( \text{distance} ( \{d_i , c_1 \} , \{ d_i , c_2 \} ) ) )$$

(6) After finding the center, assign data to that cluster till the number of elements in the cluster reaches the given threshold value. Remove the data belonging to this new cluster from the data set.

(7) Find the arithmetic mean of cluster $d_i$ to obtain its centroid $c_i$.

(8) Repeat steps (5) to (7) till the number of clusters is less than initially chosen K.

(9) Finally, the K-Means algorithm is applied for K centers until convergence criterion is met.

# 6. EXPERIMENTAL RESULTS

We examined the performance of the above described algorithms on a number of benchmark data sets taken from the UCI [13] repository of machine learning databases. To assess the efficiency of our method, we compare the results obtained by general K-means clustering method against the clustering results returned by the proposed Far Efficient K-Means algorithm on different data sets varying in their size and characteristics. The initial number of clusters is given by the user during the execution of the program. Table 1 shows some characteristics of the data sets used in this paper.

| Data set | Number of Attributes | Number of Classes | Number of Records |
|---|---|---|---|
| Iris | 4 | 3 | 150 |
| Wine | 13 | 3 | 178 |
| Abalone | 8 | 3 | 4177 |

**Table 1: Characteristics of some data sets from UCI repository**

The performance of traditional K-Means and our proposed Far Efficient K-Means algorithms are measured in terms of two standard validity measures namely Dunn's index (DI) [5] and Davies-Bouldin's index (DBI) [4] on various sized data sets. Table 2 gives a comparative analysis of the above said facts. The graphical representation of the performance analysis of K-Means and Far Efficient K-Means algorithm is shown in Figure 1(a) and 1(b) respectively. We have computed the running time of the two specified clustering algorithms. The clustering results for K-Means and Far Efficient K-Means algorithms are listed in Table 3 taking several data sets into consideration. Figure 2(a) shows the data distribution in Iris data set using the K-means algorithm by considering three numbers of clusters and Figure 2(b) shows the same using Far Efficient K-Means algorithm. Similarly, Figure 2(c) shows the data distribution in Wine data set using the K-Means algorithm by considering three numbers of clusters and Figure 2(d) shows the data distribution using Far Efficient K-Means algorithm. The algorithms were implemented in MATLAB 7.8.0 on Intel Core 2 Duo system.

| Data set<br>Algorithms | | Wine | Iris | Abalone |
|---|---|---|---|---|
| K-Means | *DI* | 0.0227 | 0.0233 | 0.0056 |
| | *DBI* | 0.6894 | 0.6791 | 0.8966 |
| Far Efficient K-Means | *DI* | **0.0477** | **0.0833** | **0.0122** |
| | *DBI* | **0.6880** | **0.6701** | **0.8647** |

**Table 2: Comparison of K-Means and Far Efficient K-Means clustering algorithms by considering Dunn's and Davies-Bouldin's index on different sized data sets (with K=3).**

| Algorithm | Wine | Iris | Abalone |
|---|---|---|---|
| K-Means | 0.2031 | 0.1250 | 256.6412 |
| Far Efficient K-Means | 0.6563 | 0.5156 | 743.6719 |

**Table 3: Running times of K-Means and Far Efficient K-Means**

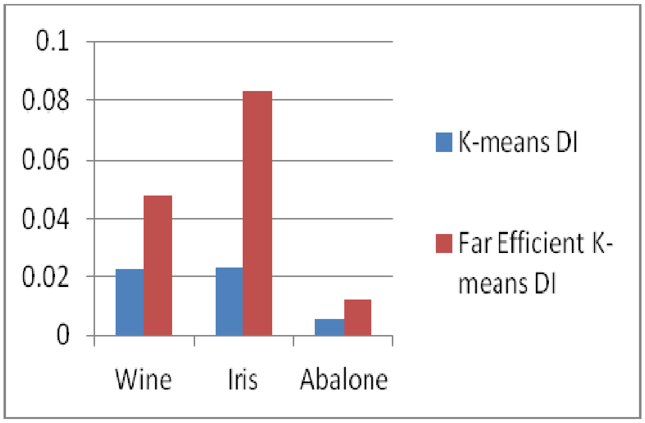

**Figure 1(a) : Performance analysis of K-Means and Far Efficient K-Means algorithm on several data sets based on Davies-Bouldin's index.**

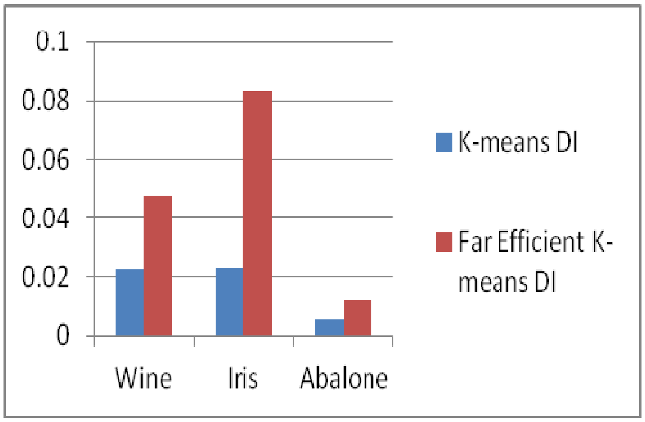

**Figure 1(b): Performance analysis of K-Means and Far Efficient K-Means algorithm on several data sets based on Dunn's index.**

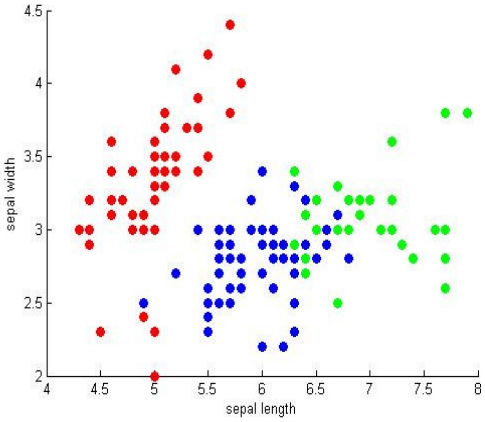

**Figure 2(a): K-Means Clustering on Iris data set with K=3**

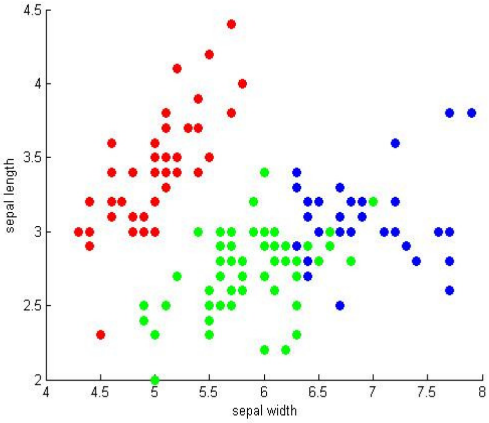

**Figure 2(b): Far Efficient K-Means Clustering on Iris data set with K=3**

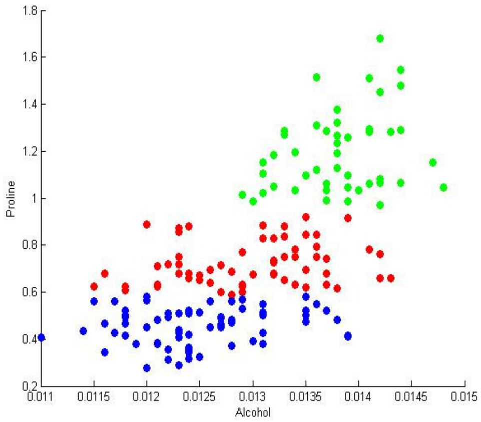

**Figure 2(c): K-Means Clustering on Wine data set with K=3**

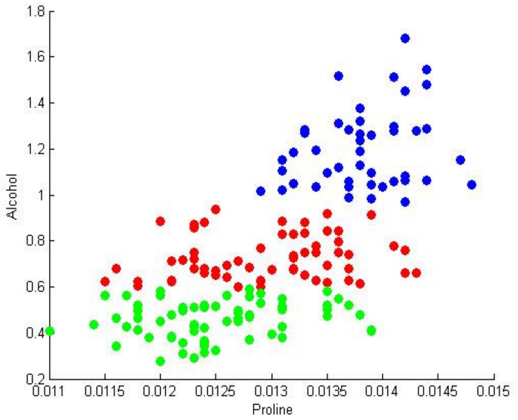

**Fig. 2(d): Far Efficient K-Means Clustering on Wine data set with K=3**

## 7. CONCLUSION

In this paper, we have examined two varieties of clustering algorithms – the customary K-Means algorithm and our proposed Far Efficient K-Means algorithm. It can be seen from the experimental result that, generally speaking, K-Means algorithm can do a pretty good job in clustering data sets in any K numbers of clusters. However, the algorithm is significantly responsive to the preliminary selection of cluster centroids. Hence, in order to minimize such complexities we have proposed a simple and efficient method for finding the initial cluster centers. Further, considering both the DI and DBI parameters for cluster validation on various sized data sets, the results obtained by our proposed algorithm produces better quality of clustering as compared to K-Means. In addition to this, the proposed algorithm also works much efficiently on larger data sets with versatile properties. But, the only negative aspect of this proposed method is its slight difference in computational time as compared to K-Means.

## 8. REFERENCES

[1] Z. Li, J. Yuan, H. Yang and Ke Zhang, “K-Mean Algorithm with a Distance Based on the Characteristic of Differences”, ”IEEE International conference on Wireless communications, Networking and mobile computing”, pp. 1-4, Oct.2008.

[2] S. Saha S. Bandyopadhyay and C. Singh, “A New Line Symmetry Distance Based Pattern Classifier”, ”International joint conference on Neural networks as part of 2008 IEEE WCCI”, pp.1426-1433, 2008.

[3] Shi Na, L. Xumin, G. Yong, “Research on K-Means clustering algorithm-An Improved K-Means Clustering Algorithm”, “IEEE Third International Symposium on Intelligent Information Technology and Security Informatics”, pp.63-67, Apr.2010.

[4] D.L. Davies and D.W. Bouldin, “A Cluster Separation Measure”, “IEEE Trans. Pattern Analysis and Machine Intelligence”,vol.1, pp.224-227, 1979.

[5] J.C. Dunn,“A Fuzzy Relative of the ISODATA Process and Its Use in Detecting Compact Well-Separated Clusters”, J. Cybernetics, vol. 3, pp. 32- 57, 1973.

[6] T. Kanungo, D. Mount, N. Netanyahu, C. Piatko and A. Wu, “An Efficient K-Means Clustering Algorithm: Analysis and Implementation”, “IEEE Transactions on Pattern analysis and Machine intelligence”, vol. 24, no.7, 2002

[7] R. Xu and D. Wunsch, “Survey of Clustering Algorithms”, “IEEE Transactions on Neural networks”, vol. 16, no. 3, May 2005.

[8] Y.M. Cheung, “A New Generalized K-Means Clustering Algorithm”, “Pattern Recognition Letters, Elsevier”,vol.24,issue15, 2883–2893, Nov.2003.

[9] C. S. Li, “Cluster Center Initialization Method for K-means Algorithm Over Data Sets with Two Clusters”, “2011 International Conference on Advances in Engineering, Elsevier”, pp. 324-328, vol.24, 2011.

[10] M. Erisoglu, N. Calis and S. Sakallioglu, “A new algorithm for initial cluster centers in K-Means algorithm”, “Published in Pattern Recognition Letters”, vol. 32, issue 14, Oct.2011.

[11] D. Napoleon and P. G. Laxmi, “An Efficient K-Means Clustering Algorithm for Reducing Time Complexity using Uniform Distribution Data Points”, “IEEE Trendz in Information science and computing”, pp.42-45, Feb.2011.

[12] J. Mac Queen, “Some methods for classification and analysis of multivariate observations”, ”Fifth Berkeley Symposium on Mathematics, Statistics and Probability”, pp.281-297, University of California Press, 1967.

[13] C. Merz and P. Murphy, UCI Repository of Machine Learning Databases,
Available: fttp://ftp.ics.uci.edu/pub/machine-learning-databases.