K-Means Clustering using Max-min Distance Measure

N. Karthikeyani Visalakshi
Lecturer in Computer Science
Vellalar College for Women
Erode, Tamil Nadu, India
karthichitru@yahoo.co.in

J. Suguna
Lecturer (SG) in Computer Science
Vellalar College for Women
Erode, Tamil Nadu, India
sugunajravi@yahoo.co.in

Abstract—**The cluster analysis deals with the problems of organization of a collection of data objects into clusters based on similarity. It is also known as the unsupervised classification of objects and has found many applications in different areas. An important component of a clustering algorithm is the distance measure which is used to find the similarity between data objects. K-means is one of the most popular and widespread partitioning clustering algorithms due to its superior scalability and efficiency. Typically, the K-means algorithm determines the distance between an object and its cluster centroid by Euclidean distance measure. This paper proposes a variant of K-means which uses an alternate distance measure namely, Max-min measure. The modified K-means algorithm is tested with six benchmark datasets taken from UCI machine learning data repository and found that the proposed algorithm takes less number of iterations to converge than the existing one with improved performance.**

Keywords - Clustering; Euclidean distance; K-Means algorithm; Max-min distance.

I. INTRODUCTION

Cluster analysis is an unsupervised learning method that constitutes a cornerstone of an intelligent data analysis process. It is the process of grouping objects into clusters such that the objects in the same cluster are similar where as objects in different clusters is different. Clustering has become an increasingly important task in modern application domains such as marketing and purchasing assistance, multimedia, and molecular biology as well as many others. There is no clustering algorithm performing best for all datasets. Each dataset requires both expertise and insight to choose a single best clustering algorithm, and it depends on the nature of application and patterns to be extracted. Different types of algorithms have been proposed in the literature [7, 13] to solve the clustering problem.

Each clustering algorithm is based on some kind of distance measures, which leads to grouping of related objects. As each distance measure follows different methods for determining the degree of similarity between two objects, the selection of an appropriate distance measure plays a vital role in any clustering algorithm. The distance measure will influence the shape of the clusters, as some elements may be close to one another according to one distance measure and farther away according to another.

K-Means [11] is a prototype-based, partitional clustering technique that attempts to find a user-specified number of clusters (K), which are represented by centroids. There are different variants of K-Means algorithm in the literature [1, 2, 3, 9, 14], each emphasizes various aspects like centroid initialization [14], number of clusters determination [2, 3], global optimization [1], etc.

The K-Means algorithm typically uses Euclidean or squared Euclidean distance to measure the distortion between a data object and its cluster centroid [17]. The Euclidean and squared Euclidean distances are usually computed from raw data and not from standardized data. While using Euclidean distances, the distance between any two objects is not affected by the addition of new objects to the analysis. However, the clustering results can be greatly affected by differences in scale among the dimension from which the distances are computed. Hence an effort is taken here to standardize the input objects before clustering and suggest an alternate distance measure. This paper proposes a modified K-Means algorithm, by applying Min-max normalization procedure [10] and Max-min distance measure [16] to reach better performance.

The paper is organized as follows: Section 2 presents an overview of clustering algorithms, K-Means clustering, distance measures and need for normalization. The modified K-Means algorithm is proposed in Section 3. Section 4 discusses the experimental analysis. Section 5 concludes the paper and outlines scope for future research work.

II. BACKGROUND

A. Clustering Algorithms

Clustering is a process of partitioning a set of data objects into a set of meaningful subclasses, called clusters. A cluster is a collection of data objects that are similar to one another based on their attribute values, and thus can be treated collectively as one group. The goal of the clustering technique is to decompose or partition a data set into groups such that both intra-group similarity and inter-group dissimilarity are maximized [17].

Clustering algorithms can be classified along different, independent dimensions. One well-known dimension categorizes clustering methods according to the result they produce. Here, we can distinguish between hierarchical and partitioning clustering algorithms [8]. Partitioning algorithms construct a flat (single level) partition of a database D of n objects into a set of K clusters such that the objects in a cluster are more similar to each other than to objects in different clusters. Hierarchical algorithms decompose the database into several levels of nested partitioning (clustering), represented for example by a dentogram, i.e. a tree that iteratively splits D into smaller subsets until each subset consists of only one object. In such a hierarchy, each node of the tree represents a cluster of D.

Another dimension according to which we can classify clustering algorithms is from an algorithmic point of view. Here we can distinguish between optimization based or distance based algorithms and density based algorithms. Distance based methods use the distances between the objects directly in order to optimize a global cluster criterion. In contrast, density based algorithms apply a local cluster criterion. Clusters are regarded as regions in the data space in which the objects are dense, and which are separated by regions of low object density (noise).

Studies have shown that partitional clustering algorithms are more suitable for clustering large datasets due to their relatively low computational requirements [15]. The time complexity of the partitioning technique is almost linear, which makes it widely used. One of the best-known distance based partitioning clustering algorithms is the K-means algorithm [7, 13].

B. K-Means Clustering

K-Means is a typical clustering algorithm [17]. It is attractive in practice, because it is simple and it is generally very fast. A set of n objects $x_i, i = 1, 2, \ldots, n$, are to be partitioned into K groups, $C_j, j = 1, 2, \ldots, n$. The objective function, based on the Euclidean distance between an object x in group j and the corresponding cluster centroid C_j, can be defined by:

$$J = \sum_{j=1}^{K} \sum_{i=1}^{n} \left\| x_i - C_j \right\|^2 \qquad ((1)$$

In detail, it randomly selects *K* of the given objects to represent the cluster centroid. Based on the selected objects, all remaining objects are assigned to their closer centroid one by one. The Euclidean distance between the object and every centroid is computed, and then the object is moved to the one of the clusters which yields minimum distance. The value of the selected centroid is recalculated by taking the mean of all data points belonging to the same cluster. The operation is iterated for all the objects. The same procedure is repeated until the objective function converges. If *K* cannot be known ahead of time, various values of *K* can be evaluated until the most suitable one is found. The effectiveness of this method as well as of others relies heavily on the objective function used in measuring the distance between objects. Choosing the proper initial centroid is the key step of the basic K-Means procedure.

Generally, the K-Means algorithm has the following important properties: (i) it is efficient in processing large data sets, (ii) it often terminates at a local optimum, (iii) the clusters have spherical shapes, (iv) it is sensitive to noise. Distance Measures

The concept of distance is the essential component of any form of clustering that helps to navigate through the data space and form clusters [15]. By computing distance, it is sensed and articulated how close together two patterns are and, based on this closeness, allocate them to the same cluster. Formally, the distance $d(x, y)$ between x and y is considered to be a two-argument function satisfying the following conditions:

$$\begin{aligned} &d(x, y) \geq 0 \quad \text{for every } x \text{ and } y \\ &d(x, x) = 0 \quad \text{for every } x \\ &d(x, y) = d(y, x) \\ &d(x, y) + d(y, z) \geq d(x, z) \end{aligned} \qquad (2)$$

When the components of the data object vectors are all in the same physical units then it is possible that the simple Euclidean distance metric is sufficient to successfully group similar data instances. However, even in this case the Euclidean distance may sometimes be misleading. Despite different measurements being taken in the same physical units, an informed decision has to be made as to the relative scaling, because different scaling can lead to different types of clustering.

Actually a mathematical formula is used to combine the distances between the single components of the data feature vectors into a unique distance measure. When the clustering process is being done using this formula, different formulas may also lead to different kinds of clustering. Domain knowledge must be used to guide the formulation of a suitable distance measure for each particular application. However, there are no general theoretical guidelines for selecting a measure for any given application.

It is often the case that the components of the data feature vectors are not immediately comparable. It can be that the components are not continuous variables, like length, but nominal categories, such as the days of the week. In these cases again, domain knowledge must be used to formulate an appropriate measure.

There are several distance measures used in the literature [15, 17]. Three distance measures used in this work for comparative analysis are defined as follows:

- **Euclidean** distances are usually computed from raw data and not from standardized data.

$$d(t_i, t_j) = \sqrt{\sum_{k=1}^{d} (t_{ik} - t_{jk})^2} \qquad (3)$$

- **Cosine** coefficient relates the overlap to the geometric average of the two sets.

$$d(t_i, t_j) = \frac{\sum_{h=1}^{d} t_{ih} t_{jh}}{\sqrt{\sum_{h=1}^{d} t_{ih}^2 \sum_{h=1}^{d} t_{jh}^2}} \quad (4)$$

- **Max-min** distance measure is found through simple min and max operations on pairs of data objects.

$$d(t_i, t_j) = \frac{\sum_{k=1}^{d} \min(t_{ik}, t_{jk})}{\sum_{k=1}^{d} \max(t_{ik}, t_{jk})} \quad (5)$$

C. *Need for Normalization*

Preprocessing [10] is required before using any data mining algorithms to improve the results' performance. Data normalization is one of the preprocessing procedures in data mining, where the attribute data are scaled so as to fall within a small specified range such as -1.0 to 1.0 or 0.0 to 1.0. Normalization before clustering is often needed for distance metric, such as Euclidian distance, which are sensitive to differences in the magnitude or scales of the attributes. In real applications, because of the differences in range of attributes' value, one attribute might overpower the other one. Normalization prevents outweighing features with large range like 'salary' over features with smaller range like 'age'. The goal is to equalize the size or magnitude and the variability of these features.

There are many methods for data normalization which include Min-max normalization, z-score normalization and normalization by decimal scaling [5, 10]. Min-max normalization performs a linear transformation on the original data. Min-max normalization maps a value of each attribute to the range [0, 1]. In z-score normalization, the values of attributes are normalized based on the mean and standard deviation of corresponding attributes. This method of normalization is useful when the actual minimum and maximum of every attribute is unknown. Normalization by decimal scaling normalizes by moving the decimal point of values of attribute. The number of decimal points moved depends on the maximum absolute value of the attribute.

III. PROPOSED ALGORITHM

Normally, K-Means clustering algorithm uses Euclidean distance as the distance measure to compute the similarity between the object and its centroid. Alternately, the Cosine distance measure [15] is often used for document clustering. In this paper, Max-min distance measure [16] is suggested in place of Euclidean distance measure. The Max-min distance measure requires the values to be in the range [0, 1]. In order to scale the given objects to fall within a small specified range [0, 1], Min-max normalization procedure [10] is followed as a pre-processing step for the proposed K-Means algorithm. The step by step procedure of the proposed K-Means clustering algorithm based on Max-min distance measure is given here under.

Algorithm 1

K-Means Algorithm based on Max-min Distance Measure

Input : Dataset of n objects with d features and the value of *K*

Output: Partition of the input data into *K* clusters

Procedure:

Step 1: Normalize the input data objects to fall into the range [0, 1] using Min-max normalization procedure

Step 2: Declare a membership matrix U of size *n* x *K*

Step 3: Generate *K* cluster centroids randomly within the range of the data or select *K* objects randomly as initial cluster centroids. Let the centroids be $C_1, C_2, \ldots, C_K$

Step 4: Calculate the distance measure d_{ij} using Max-min similarity measure

$$d_{ij} = \frac{\sum_{k=1}^{d} \min(x_{ik}, C_{jk})}{\sum_{k=1}^{d} \max(x_{ik}, C_{jk})}$$

for all cluster centroids C_j, $j = 1, 2, \ldots, K$ and data objects $x_i, i = 1, 2, \ldots, n$

Step 5: Compute the *U* membership matrix

$$U_{ij} = \begin{cases} 1; & d_{ij} \le d_{il},\ j \ne l \\ 0; & otherwise \end{cases} \quad \begin{matrix} i = 1, 2, \ldots, n \\ j = 1, 2, \ldots, K \end{matrix}$$

Step 6: Compute new cluster centroids C_j

$$C_j = \frac{\sum_{i=1}^{n} (U_{ij})\, x_i}{\sum_{i=1}^{n} (U_{ij})} \quad for\ j = 1, 2, \ldots, K$$

Step 7: Repeat steps 4 to 6 until convergence

IV. EXPERIMENTAL ANALYSIS

The main purpose of this work is to explore the impact of Max-min distance measure in the K-Means clustering algorithm with normalization. The experiment analysis is performed with six benchmark datasets available in the UCI machine learning data repository [12]. The information about the datasets is shown in Table 1. The values of all datasets are normalized by Min-max normalization method before performing clustering process using K-Means algorithm.

The performance of K-Means algorithm is measured in terms of four external validity measures [4, 6] namely Rand index, Jaccard index, F-Measure and Entropy along with number of iterations required to reach the desired clusters. The external validity measures test the quality of clusters by comparing the results of clustering with the 'ground truth' (true class labels). All these four measures have a value between 0 and 1. In case of Rand index, Jaccard index and F-Measure, the value 1 indicates that the data clusters are exactly same and so the increase in the values of these measures proves the better performance. But, the value 1 signifies that the data clusters are entirely different for Entropy measure and so the value of this measure is to be decreased to reach better quality clusters.

The results of K-Means algorithm with Max-min distance, in comparison with the results of K-Means algorithm with traditional distances Euclidean and Cosine, in terms of Rand index, Jaccard index, F-Measure and Entropy are shown in Table 2, Table 3, Table 4 and Table 5 respectively. From the Tables, it is observed that Max-min distance yields better results than other two distances for almost all datasets. It is noted that both Euclidean and Max-min distances produce exactly same performance, for iris dataset, where as they yield approximately same performance for mammography dataset, in terms of all four validity measures. When dermatology dataset is considered, it is evident that the results of Max-min distance are highly appreciable than Euclidean distance, in terms of all four validity measures. However, the results of Max-min distance are approximately same as Cosine distance, in terms of Rand index, Jaccard index and F-Measure. The domino effect of Max-min distance in K-Means algorithm based on four validity measures Rand index, Jaccard index, F-Measure and Entropy is explored in Figure 1, Figure 2, Figure 3 and Figure 4 respectively.

In order to measure the computational efficiency of proposed K-Means clustering algorithm, number of iterations required for convergence is compared as shown in Figure 5. The figure shows that the Max-min distance measure has considerable advantage over Euclidean and Cosine distance measures, because it requires less number of iterations to reach convergence. From the comparative analysis, it is concluded that the Max-min distance is more suitable than the other two distances, namely Euclidean and Cosine for all experimented numerical datasets.

TABLE 1. DETAILS OF DATASETS

S. No.	Dataset	No. of Attributes	No. of Classes	No. of Instances
1	Australian	14	2	690
2	Breast Cancer	10	2	699
3	Dermatology	34	6	366
4	Hepatitis	19	2	155
5	Iris	4	3	150
6	Mammography	5	2	961

TABLE 2. COMPARATIVE ANALYSIS BASED ON RAND INDEX

Dataset	*Euclidean*	*Cosine*	*Max-min*
Australian	0.5071	0.6228	0.6697
Breast Cancer	0.9049	0.9417	0.8647
Dermatology	0.7018	0.9104	0.9134
Hepatitis	0.6434	0.5934	0.6723
Iris	0.9499	0.9272	0.9499
Mammography	0.6573	0.6305	0.6550

TABLE 3. COMPARATIVE ANALYSIS BASED ON JACCARD INDEX

Dataset	*Euclidean*	*Cosine*	*Max-min*
Australian	0.5047	0.4585	0.5096
Breast Cancer	0.8419	0.7758	0.8984
Dermatology	0.3065	0.6359	0.6476
Hepatitis	0.5700	0.5106	0.6012
Iris	0.8602	0.8033	0.8602
Mammography	0.4952	0.4705	0.4954

TABLE 4. COMPARATIVE ANALYSIS BASED ON F-MEASURE

Dataset	*Euclidean*	*Cosine*	*Max-min*
Australian	0.6071	0.4561	0.4538
Breast Cancer	0.6589	0.6179	0.6426
Dermatology	0.5631	0.8084	0.8191
Hepatitis	0.7602	0.6992	0.7892
Iris	0.9600	0.9401	0.9600
Mammography	0.7815	0.7579	0.7802

TABLE 5. COMPARATIVE ANALYSIS BASED ON ENTROPY INDEX

Dataset	*Euclidean*	*Cosine*	*Max-min*
Australian	0.3592	0.2943	0.2668
Breast Cancer	0.0811	0.1254	0.0689
Dermatology	0.9264	0.2831	0.2289
Hepatitis	0.4576	0.4772	0.4391
Iris	0.1494	0.1992	0.1494
Mammography	0.5057	0.5313	0.5013

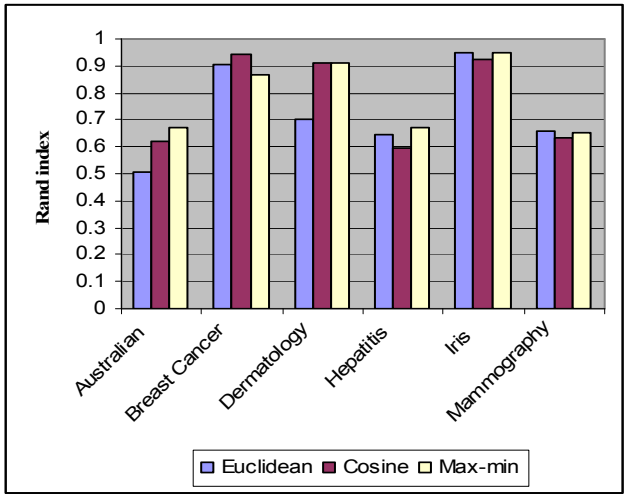

Figure 1. Performance Analysis based on Rand Idex

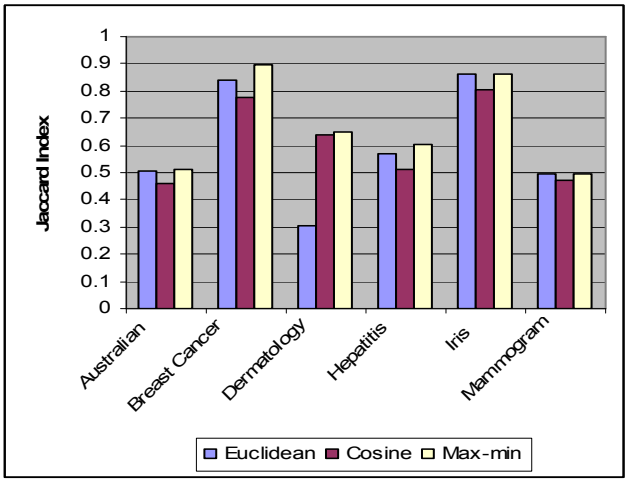

Figure 2. Performance Analysis based on Jaccard Coefficient

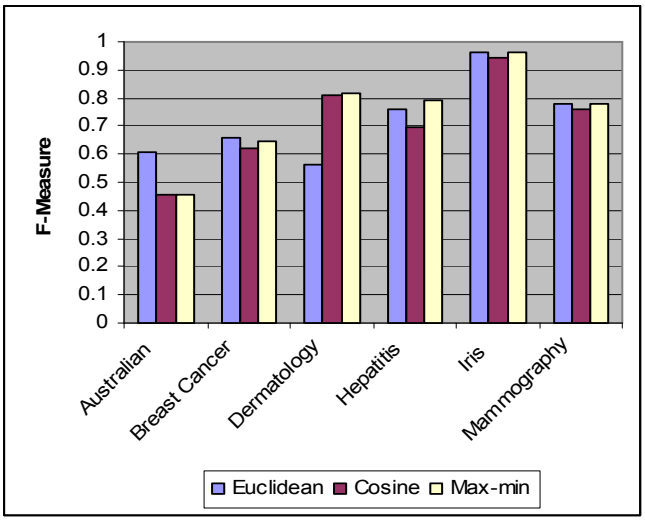

Figure 3. Performance Analysis based on F-Measure

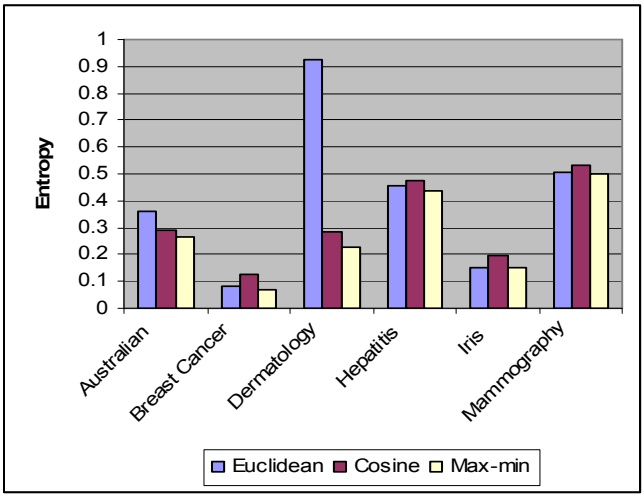

Figure 4. Performance Analysis based on Entropy

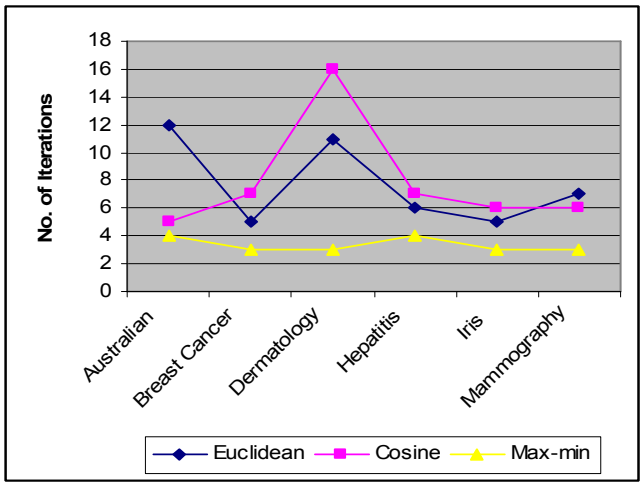

Figure 5. Performance Analysis based on No. of Iterations

V. CONCLUSION

The clustering problem has been widely studied since it arises in much knowledge management oriented applications. It aims at identifying the distribution of patterns and intrinsic correlations in datasets by partitioning the data objects into similarity clusters. In this paper, the modified K-Means clustering algorithm is proposed by applying Max-min distance measure in the place of Euclidean distance measure. The proposed algorithm performs normalization process using Min-max method, as the initial step of clustering process. The results of numerical experiments on six benchmark datasets demonstrate the superiority of the proposed algorithm, produces high quality clusters with minimum computational complexity. In future, the Max-min distance measure can be applied in fuzzy C-Means algorithm and its variants, to improve the performance.

REFERENCES

[1] Adil M. Bagirov, "Modified Global K-Means Algorithm for Minimum Sum-of-squares Clustering Problems", Pattern Recognition, Vol. 41, 2008, 3192-3199.

[2] Chieh-Yuan Tsai, Chuang-Cheng Chiu, "Developing a Feature Weight self-adjustment Mechanism for a K-Means Clustering Algorithm", Computational Statistics and Data Analysis, Vol. 52, 2008, pp. 4658-4672.

[3] Daxin Jiang, Chun Tang and Aidong Zhang, "Cluster Analysis for Gene Expression Data: A Survery", IEEE Transactions on Knowledge and Data Engineering, Vol. 16, No. 11, November 2004, pp. 1370-1386

[4] M. Halkidi, Y. Batistakis, M. Vazirgiannis, "Cluster Validity Methods", ACM SIGMOD Record, Vol. 31, No. 3, 2002, pp. 19-27.

[5] J. Han, M. Kamber, Data Mining Concepts and Techniques, Morgan Kaufmarm Publishers, San Francisco, 2006.

[6] Hui Xiong, Junjie Wu, Jian Chen, "K-means clustering versus validation measures: a data distribution perspective", Proceedings of the 12th ACM SIGKDD international conference on Knowledge Discovery and Data Mining, August 2006, pp. 779-784.

[7] Jain A K, Murthy M N, Flynn P J., "Data Clustering: A Review", *ACM Computing Surveys*, Vol. 31 No.3, Septermber 1999, pp. 265-323.

[8] Januzaj E, Kriegel Hans P, Pfeifle M., "DBDC: Density Based Distributed Clustering", Advances in Databases Technology – EDBT 2004, Springer Berlin / Heidelberg, Vol. 2992, February 2004, pp. 529-530.

[9] Krista Rizman Zalik, "An Efficient K'means Clustering Algorithm", Pattern Recognition Letters, Vol. 29, 2008, pp. 1385-1391.

[10] Luai A. Shalabi, Zyad Shaaban and Basel Kassabeh, "Data Mining A Preprocessing Engine", Journal of Computer Science, Vol. 2, No. 9, 2006, pp. 735-739.

[11] J. B. MacQueen, "Some Methods for classification and Analysis of Multivariate Observations", Proceedings of 5th Berkeley Symposium on Mathematical Statistics and Probability. University of California Press, 1967, Vol 1, pp. 281–297

[12] Merz C J, Murphy P M., "UCI Repository of Machine Learning Databases", Irvine, University of California, 1998, **http://www.ics.uci.eedu/~mlearn/**.

[13] Pang-Ning Tan, Steinbach M, Kumar V., "Cluster Analysis: Basic Concepts and Algorithms", *Introduction to Data Mining,* Pearson Addison Wesley, Boston, 2006.

[14] Shehroz S. Khan, Amir Ahmad, "Cluster Center Initialization Algorithm for K-Means Clustering", Pattern Recognition Letters, Vol. 25, 2004, pp. 1293-1302.

[15] K. P. Soman, Shyam Diwakar and V. Ajay, Insight into Data Mining Theory and Practice, PHI, India, 2006.

[16] Timothy J Ross, Fuzzy Logic with Engineering Applications, Mcgraw Hill, Newyork.

[17] Xu R, Wunsch D II, "Survey of clustering algorithms", *IEEE Transaction on Neural Networks,* Vol. 16, Issue 3, May 2005, pp. 645-678.

.