Clustering Mixed Data Points Using Fuzzy C-Means Clustering Algorithm for Performance Analysis

T.Velmurugan
Associate Professor
PG and Research Department of Computer Science
D.G.Vaishnav College
Arumbakkam, Chennai-600106

T.Santhanam
Associate professor & Head
PG and Research Department of Computer Science
D.G.Vaishnav College
Arumbakkam, Chennai-600106

***Abstract*: Clustering plays an outstanding role in data mining research. Among the various algorithms for clustering, most of the researchers used the Fuzzy C-Means algorithm (FCM) in the areas like computational geometry, data compression and vector quantization, pattern recognition and pattern classification. In this research, a simple and efficient implementation of FCM clustering algorithm is presented. Three types of inputs are given to algorithm. The data points are first distributed manually, the statistical distributions Normal and Uniform are anther two methods by using the Box-Muller formula. The algorithm is analyzed based on their clustering quality. The behavior of the algorithm depends on the number of data points as well as on the number of cluster. The performance of the algorithm is investigated during different execution of the program on the input data points. The execution time for each cluster and total elapsed time to cluster all the data points is also analyzed and the results are compared with one another.**

Keywords: Fuzzy C-Means Algorithm, Fuzzy Clustering, Unsupervised Clustering, Data Clustering.

I. INTRODUCTION

Clustering problems arise in many different applications, such as data mining, knowledge discovery, data compression, vector quantization, and pattern recognition and pattern classification. Clustering is a field of research belonging to both data analysis and machine learning major domains. Because new challenges appear permanently, new approaches have to be developed to deal with large amount of data, heterogeneous in nature (numerical, symbolic, spatial, etc.) and to produce several types of clustering schemes (crisp, overlapping or fuzzy partitions and hierarchies). Many methodologies have been proposed in order to organize, to summarize or to simplify a dataset into a set of clusters such that data belonging to a same cluster are similar and data from different clusters are dissimilar [4][5][6]. The clustering process is usually based on a proximity measure or, in a more general way, on the properties that data share. This can easily mention three major types of clustering processes according to the way they organize data: hierarchical, partitioning and mixture model methods. Most of the clustering methods have been developed in these frameworks in the last decades and allow a large amount of application fields. Nevertheless, some fields which led to recent attentions are still inefficiently processed. This is all the more true when the natural classes of data are neither disjoint nor fuzzy but clearly overlap. This situation occurs in important fields of applications such that Information Retrieval (several thematic for a single document), biological data (several metabolic functions for one gene). The definition of what constitutes a cluster is not well defined, and, in many applications clusters are not well separated from one another. Nonetheless, most cluster analysis seeks as a result, a crisp classification of the data into non-overlapping groups [2][3][7]. In this research work, the Fuzzy C-Means (FCM) is examined based on the distance between data points and the clustering quality of the algorithm.

Determining the quality of a clustering algorithm involves evaluating and assessing the quality of the clusters produced and is an important task in data mining. There are three approaches to measuring cluster quality, based on external, relative and internal criteria. The term external validity criteria are used when the results of the clustering algorithm can be compared with some pre-specified clustering structures (Halkidi et al., 2002). Relative validity criteria measure the quality of clustering results by comparing them with others generated by other clustering algorithms, or by the same algorithm using different parameters (Al-Harbi, 2003). An internal validity criterion involve the development of functions that compute the distances between objects within each cluster, or the distance between the clusters themselves, and uses such distances to assess the clustering quality [2][8]. To achieve a good clustering, these criteria are in the form of measures to assess the quality of a clustering. This research work uses only the internal validity criteria in a random way.

II. METHODOLOGY

Fuzzy C-Mean (*FCM*) is an unsupervised clustering algorithm that has been applied to wide range of problems involving feature analysis, clustering and classifier design. *FCM* has a wide domain of applications such as agricultural engineering, astronomy, chemistry, geology, image analysis, medical diagnosis, shape analysis, and target recognition. With the developing of the fuzzy theory, the fuzzy c-means

clustering algorithm based on Ruspini fuzzy clustering theory was proposed in 1980s. This algorithm Fuzzy C-Means is examined to analyze based on the distance between the various input data points. The clusters are formed according to the distance between data points and cluster centers are formed for each cluster. The basic structure of the FCM algorithm is discussed below. The Algorithm Fuzzy C-Means (FCM) is a method of clustering which allows one piece of data to belong to two or more clusters. This method is frequently used in pattern recognition. It is based on minimization of the following objective function:

$$J_m = \sum_{i=1}^{N}\sum_{j=1}^{C} u_{ij}^{m}\|x_i - c_j\|^2 \quad , \quad 1 \le m < \infty$$

where m is any real number greater than 1, u_{ij} is the degree of membership of x_i in the cluster j, x_i is the i^{th} of d-dimensional measured data, c_j is the d-dimension center of the cluster, and ||*|| is any norm expressing the similarity between any measured data and the center. Fuzzy partitioning is carried out through an iterative optimization of the objective function shown above, with the update of membership u_{ij} and the cluster centers c_j by:

$$u_{ij} = \frac{1}{\sum_{k=1}^{C}\left(\frac{\|x_i - c_j\|}{\|x_i - c_k\|}\right)^{\frac{2}{m-1}}} \quad , \quad c_j = \frac{\sum_{i=1}^{N} u_{ij}^{m} \cdot x_i}{\sum_{i=1}^{N} u_{ij}^{m}}$$

This iteration will stop when $\max_{ij}\left\{\left|u_{ij}^{(k+1)} - u_{ij}^{(k)}\right|\right\} < \varepsilon$, where ξ is a termination criterion between 0 and 1, whereas k is the iteration steps. This procedure converges to a local minimum or a saddle point of J_m. The algorithm is composed of the following steps:

1. Initialize $U=[u_{ij}]$ matrix, $U^{(0)}$
2. At k-step: calculate the centers vectors $C^{(k)}=[c_j]$ with $U^{(k)}$
3. Update $U^{(k)}$, $U^{(k+1)}$
4. If $\| U^{(k+1)} - U^{(k)}\| < \xi$ then STOP; otherwise return to step 2.

In this algorithm, data are bound to each cluster by means of a Membership Function, which represents the fuzzy behavior of the algorithm [9],[10][12]. To do that, the algorithm have to build an appropriate matrix named U whose factors are numbers between 0 and 1, and represent the degree of membership between data and centers of clusters[1][13][14]. FCM clustering techniques are based on fuzzy behavior and provide a natural technique for producing a clustering where membership weights have a natural (but not probabilistic) interpretation. This algorithm is similar in structure to the K-Means algorithm and also behaves in a similar way. Based on the distance between two data points, the clusters are formed in this research work. To find the distances between the data points, the symmetric distance and in which the 2-norm distance measure is used. In the Euclidean space R^n, the distance between two points is usually given by the Euclidean distance (2-norm distance). The formula for 2-norm distance is

$$\text{2-norm distance} = \left(\sum_{i=1}^{n} |x_i - y_i|^2\right)^{1/2}$$

The 2-norm distance is the Euclidean distance, a generalization of the Pythagorean Theorem to more than two coordinates. It is what would be obtained if the distance between two points were measured with a ruler: the "intuitive" idea of distance [12][16][18]. Based on this idea of finding the distance, the clustering qualities of the proposed algorithm is analyzed here.

III. RESULTS AND DISCUSSION

As stated above, the implementation plan will be in three parts, one is manual creation of data points (by pressing the mouse button in the Applet window), second is Normal distribution and third one is Uniform distribution. The Normal and Uniform distribution of input data points are created by using Box-Muller formula. The program is written by JAVA language. After the creation of data points, the user has to specify the number of clusters. In the output window, the data points in each cluster are displayed by different colors and the execution time is calculated in milliseconds. Presented here the manual creation of input data points first, second the normal distribution of data points followed by uniform distribution. The resulting clusters of the manual distribution of data points for FCM algorithm is given in Fig. 1. The number of clusters and the data points as input are given by the user during the execution of the program. The number of data points is 1000 and the number of clusters is 5 (k = 5). The algorithm is repeated 1000 times (one iteration for each data point) to get efficient output. The cluster centers (centroids) are calculated for each cluster by its mean value and clusters are formed depending upon the distance between data points.

For different input data points, the algorithm gives different types of outputs. The input data points are generated in red color and the output of the algorithm is displayed in different colors as shown in Fig. 1. The center point of each cluster is displayed in white color. The total elapsed time and the execution time for each cluster to each run are calculated in milliseconds. The time taken for execution of the algorithm varies from one run to another run and also it differs from one computer to another computer. The number of data points is the size of the cluster. If the number of data points are 1000 then the algorithm is repeated the same one thousand times [15]. For each data point, the algorithm executes once. The algorithm takes 3937 milliseconds for 1000 data points and 5 clusters for the manual creation of data points as in Fig 1. The same algorithm is executed 5 times and results are tabulated in table 1. The total elapsed time for all clusters is given at end of the column 'Size' and the total of the execution time for each cluster is given at the end of the column 'Time'. The difference between these two is available in the last row.

Next, the same algorithm is executed by giving Normal distribution of 1000 data points as input and the result is shown in Fig.2. In this case, the algorithm takes 4157 msec for 5 clusters. Also the algorithm is executed 5 times and the results are listed in table 2. The algorithm is also executed by taking Uniform distribution of 1000 data points and the result is shown in Fig.3. The total execution time for clustering 5 clusters is 3719 msec. Table 3 lists the cluster result of 5 runs of the same distribution of input data points.

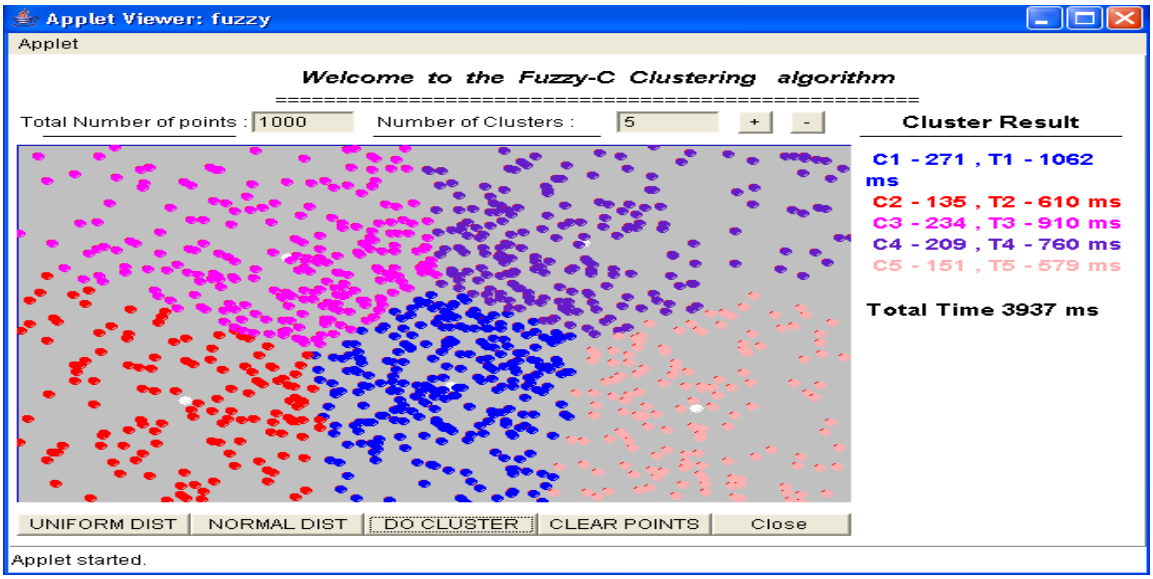

Fig. 1: Cluster Result of Manual Distribution

Table 1: Results for Manual Distribution

Cluster\Run	1		2		3		4		5	
	Size	Time	Size	Time	Size	Time	Size	Time	Size	Time
1	271	1062	180	783	163	671	214	939	266	983
2	135	610	151	671	256	980	203	812	168	660
3	234	910	244	986	228	955	192	875	227	859
4	209	760	282	1106	183	677	237	915	191	764
5	151	579	143	532	170	806	154	721	148	688
Time(ms)	3937	3921	4096	4078	4109	4089	4276	4262	3969	3954
Diff. Time	16		18		20		14		15	

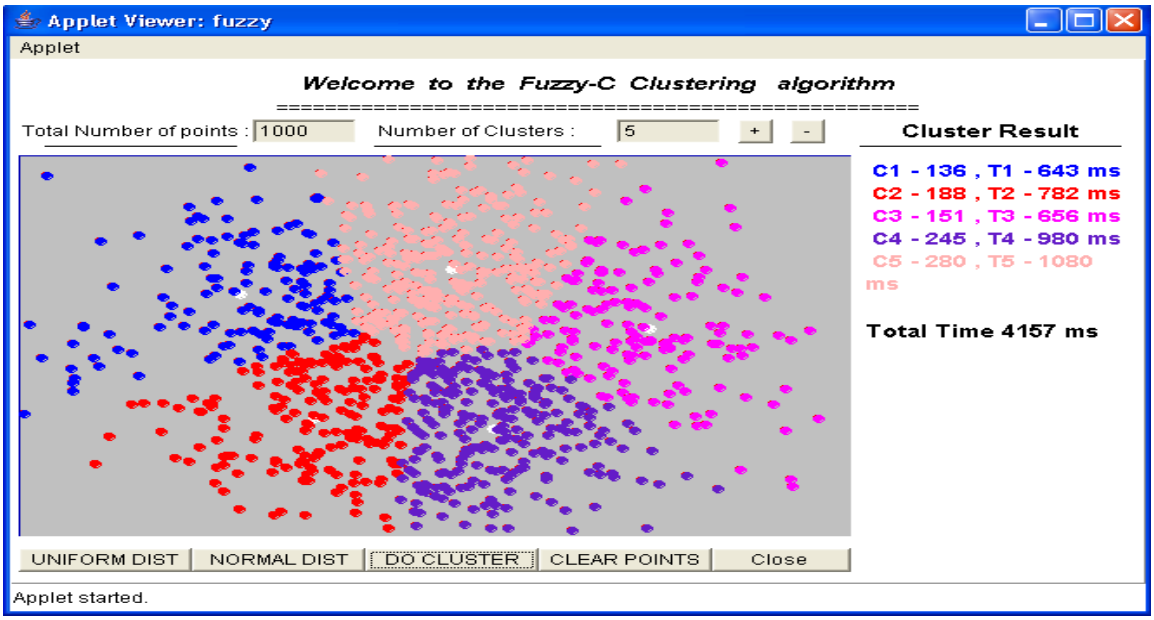

Fig. 2: Cluster Result of Normal Distribution

Table 2: Results for Normal Distribution

Cluster\Run	1		2		3		4		5	
	Size	Time	Size	Time	Size	Time	Size	Time	Size	Time
1	136	643	152	854	173	815	185	756	220	880
2	188	782	181	827	196	756	247	961	193	885
3	151	656	196	835	191	827	242	938	240	913
4	245	980	261	1013	190	827	184	738	167	713
5	280	1080	210	781	250	985	142	754	180	835
Time(ms)	4157	4141	4325	4310	4240	4210	4162	4147	4247	4226
Diff. Time	16		15		30		15		21	

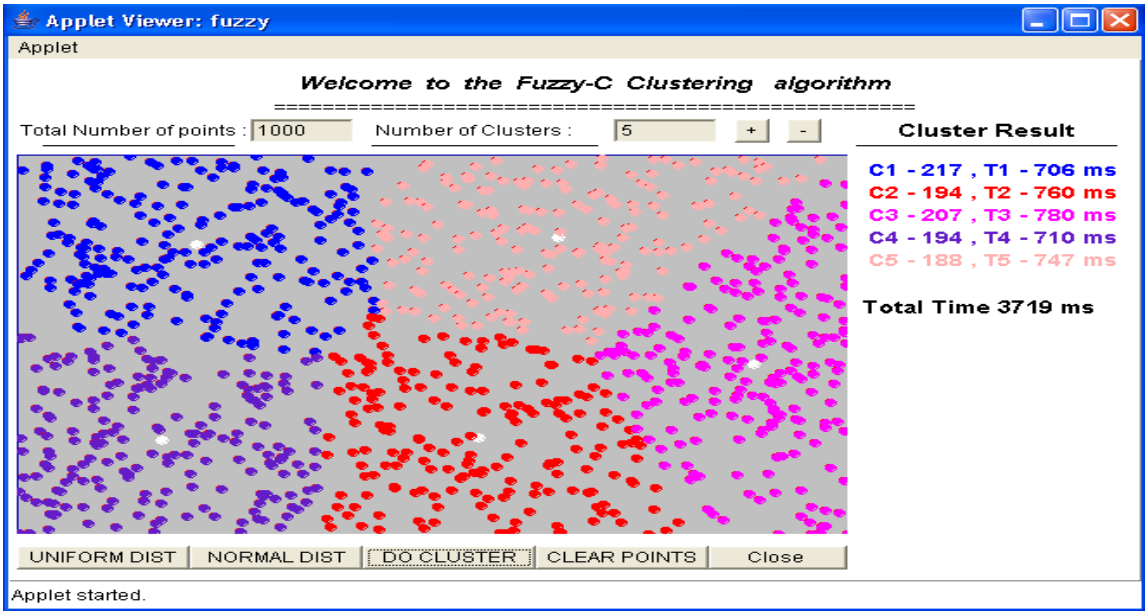

Fig. 3: Cluster Result of Uniform Distribution

Table 3: Results for Uniform Distribution

Cluster\Run	1		2		3		4		5	
	Size	Time	Size	Time	Size	Time	Size	Time	Size	Time
1	217	706	201	747	225	947	219	796	177	717
2	194	760	174	758	160	702	180	779	229	874
3	207	780	166	670	201	810	212	895	199	688
4	194	710	227	798	196	789	205	704	202	862
5	188	747	232	850	218	844	184	748	193	812
Time(ms)	3719	3703	3844	3823	4103	4092	3940	3922	3969	3953
Diff. Time	16		21		11		18		16	

Table 4: Performance Results Comparison

Manual Distribution		Normal Distribution		Uniform Distribution	
TET	ICT	TET	ICT	TET	ICT
4077.4	4060.8	4226.2	4206.8	3915	3898.6
4069.1		4216.5		3906.8	
16.6		19.4		16.4	

Table 4 shows that the average time for all the three types of distributions. Here TET stands for Total Elapsed Time and ICT means Individual Cluster Time. The last row contains the difference between TET and ICT times in each category. The average time of TET and ICT is given before the last row. It is easy to find the difference in times between the distributions. From table 4, it can easy to identify the experimental results. Like the algorithm is executed many times and the results are analyzed based on the number of data points and the number of clusters. The behavior of the algorithm is analyzed based on observations. The performance of the algorithm have also been analyzed for several executions by considering different data points (for which the results are not shown) as input (500 data points, 1500 data points etc.) and the number of clusters are from 5 to 10 (for which also the results are not shown), the outcomes are compared with one another.

IV. CONCLUSION

The execution time varies from one processor to another processor, which depends on the speed and type of the processor. It is not possible to get the exact results for clustering algorithms for any kind of data. Some nearest results only are found by experiments. The experimental results shows that the time taken for Uniform distribution of input data points is less than the time taken for manual and Normal distribution of data points. Among the other two types of distributions, the average time for manual distribution is less than the Normal distribution. The difference in time between total elapsed time and execution time for each cluster for Normal distribution is also higher than the other two kinds of input data points. Thus the FCM algorithm shows its efficiency for Uniform distribution of input data points.

REFERENCES

[1] Al-Zoubi, M.B., A. Hudaib and B. Al-Shboul, 2007, A fast fuzzy clustering algorithm, Proceedings of the 6th WSEAS Int. Conf. on Artificial Intelligence, Knowledge Engineering and Data Bases, February 2007, Corfu Island, Greece, pp. 28-32, 2007

[2] Berkhin P, "Survey of Clustering Data Mining Techniques", Technical Report, Accrue Software, Inc, 2002. www.ee.ucr.edu/~barth/EE242/clustering_survey.pdf

[3] Davies.D.L and D.W. Bouldin, A cluster separation measure, IEEE Trans. Pattern Anal. Machine Intell. Vol. 1, 2009. ISSN: 0162-8828, pp. 224-227, DOI: 10.1109/TPAMI.1979.4766909

[4] Halkidi, M., Batistakis, Y. and Vazirgiannis, M. (2002) 'Cluster validity methods: Part I', SIGMOD Record (ACM Special Interest Group on Management of Data), Vol. 31, No. 2, June-2002, ISSN: 0163-5808, pp.40–45.

[5] Han J. and M. Kamber, Data Mining: Concepts and Techniques, Morgan Kaufmann Publishers, Second Edition, New Delhi, 2006. ISBN : 978-81-312-0535-8

[6] Jain A.K. and R.C. Dubes, Algorithms for Clustering Data, Prentice Hall Inc., Englewood Cliffs, New Jerssy, 1988. ISBN: 0-13-022278-X

[7] Jain, A.K., M.N. Murty and P.J. Flynn, 1999, "Data Clustering: A review", ACM Computing Surveys, Vol. 31, No. 3, September 1999, DOI:10.1.1.18.2720&rep=rep1&type=pdf

[8] Kaufman, L. and P.J. Rousseeuw, Finding Groups in Data: an Introduction to Cluster Analysis, John Wiley and Sons, 1990.

[9] Moh'd Belal Al- Zoubi, Amiad Hudaib, Basharal-Shboul, A Fast Fuzzy Clustering Algorithm, Proceedings of the 6th WSEAS Int. Conf. on Artificial Intelligence, Knowledge Engineering and Data Bases, 2007, Corfu Island, Greece, pp. 16-19.

[10] Nikhil R. Pal, Kuhu Pal, James M. Keller, and James C. Bezdek, A Possibilistic Fuzzy C-Means Clustering Algorithm, IEEE Transactions on fuzzy Systems, Vol. 13, NO. 4, Aug. 2005, ISSN: 1063-6706, PP. 517-530.

[11] Nikos Pelekis, Dimitris K. Iakovidis , Evangelos E. Kotsifakos , Ioannis Kopanakis, Fuzzy clustering of intuitionistic fuzzy data, Int. J. Business Intelligence and Data Mining, Vol. 3, No. 1, 2008, pp. 45-65, Inderscience Publishers, ISSN:1743-8195,DOI: 10.1504/IJBIDM.2008.017975

[12] Pal. N.R, Bezdek. J.C, On cluster validity for the Fuzzy C-Means model, IEEE Trans On Fuzzy Systems, vol. 3, 1995, ISSN: 1063-6706, pp. 370-379, DOI: 10.1109/91.413225

[13] Romesburg, H. Clarles, Cluster Analysis for Researchers, 2004, ISBN 1-4116-0617-5. www.lulu.com/items/volume_1/46000/46479/.../CLUSPreview.pdf

[14] Valente de Oliveira. J, W. Pedrycz, Advances in Fuzzy Clustering and its Applications, British Library Cataloguing in Publication Data, ISBN 978-0-470-02760-8 (HB).

[15] Velmurugan.T and Santhanam,T, Clustering of random data points using K-Means and Fuzzy-C Means Clustering Algorithms, Proceedings of the IEEE International Conference on Emerging Trends in Computing, Virudhunagar, India, Jan – 2009, pp.177-180, 8-10.

[17] Velmurugan.T and T.Santhanam, A Survey of Partition Based Clustering Algorithms in Data Mining: An Experimental Approach, Information Technology Journal, ISSN: 1812-5638, DOI: 10.3923/itj-2011.

[18] Velmurugan.T and T.Santhanam, Performance Evaluation of K-Means and Fuzzy C-Means Clustering Algorithms for Statistical Distributions of Input Data Points, European Journal of Scientific Research, Vol. 46, No. 3, 2010, pp. 320-330 ISSN: 1450-216X.
http://www.eurojournals.com/ejsr.htm

[19] Xie. X.L and G. Beni, "A validity measure for fuzzy clustering," IEEE Transactions on Pattern Analysis and Machine Intelligence, v. 13, n. 8, 1991, ISSN: 0162-8828, pp. 841–847.

[20] Yong.Y, Z. Chongxun, L. Pan, A Novel Fuzzy C-Means Clustering Algorithm for Image Thresholding, Measurement Science Review, Volume 4(1), 2004.

[21] YuChen Song, M.J. O'Grady, G.M.P. O'Hare, Research and Application of Clustering Algorithm for Arbitrary Data Set, 2008, International Conference on Computer Science and Software Engineering, ISBN: 978-0-7695-3336-0, PP. 251 - 254, DOI: 10.1109/CSSE.2008.415