Comparative Performance of Some Center-adjustment Clustering Algorithms

Shri Kant*
S.S. Khan*
P.K. Saxena*

1. Introduction

Clustering is the most natural phenomenon of human beings. Whenever one encounters with unknown collection of objects to be put in a different fashion, he will start putting these objects into most natural groups. The formation of natural groups depends on individual perception and his understanding of the problem. Cluster analysis is the process of automating this natural activity of human being. To automate the process of object clustering through machine, one has to represent these objects in the form of a multidimensional pattern vector. Mathematically a pattern vector is a set of measurements, that describes an object $X = [x_1, x_2, \ldots, x_d]$. Each component of a pattern is called feature or attribute. A cluster may be defined by defining a set of centers $M_1, M_2, \ldots, M_k$ and measure of proximity $d_i (X, M_i)$. A cluster is a set of points, which are nearer to particular center $C_i = \{X / d_i (X, M_i) < d_j (X, M_j) \} i \neq j$. The objects within a cluster show high degree of natural association and objects between the clusters show low degree of natural association. The degree of natural association is decided by proximity measures [4] which are either of similarity or dissimilarity type. Numerous algorithms are available in literature for creating natural clusters in the given body of complex data sets. These clustering algorithms are broadly covered under the two headings [5].

(i) Hierarchical clustering and

(ii) Partitional clustering

In hierarchical clustering the objects are ordered in such a way that the individual objects in the same cluster at any level remain together at all higher levels. Whereas, in Partitional clustering objects are taken serially and dynamic allocation of the object into their respective cluster are carried out during the execution of algorithm till the convergence is achieved. In hierarchical clustering the process of construction becomes computationally infeasible as the data size increases, whereas in partitional clustering this limitation is not there because the objects are taken serially. The rationale behind this work is to seek optimal number of clusters with fast converging algorithm and the algorithm should be able to sustain some level of noise in the data set.

The paper is organized as follows. In section 2 ASCA is described briefly along with the method to check its

ABSTRACT

In the present paper the robustness of Automatic and Stable Clustering Algorithm (ASCA) [1] has been studied against Gaussian noise. It has been observed that ASCA is well robust up to 25% of noise. In addition to it performance of four center adjustment clustering algorithms viz. K-means [2], Fuzzy C-means [3] and K-Harmonic means [7] and ASCA has been studied on well-known data sets. It has been observed that ASCA's performance is better in comparison to other center-adjustment algorithms.

Keywords : Clustering Algorithms, Gaussian Noise, K-Harmonics, Objective Function and Percentage Misclassification Factor

* **Corresponding Author** : Dr. Shri Kant, Scientific Analysis Group, Defense R & D Organization, Metcalfe House Complex, Delhi-110054. {ojha_sk@msn.com, saxenapk@hotmail.com}

robustness against Gaussian noise. Section 3 deals with brief introduction of K-Means, Fuzzy c-means and K-Harmonic Means algorithms. The performance of these algorithms is discussed in Section 4.

2. Robustness of ASCA

ASCA [1] has been developed to remove two major drawbacks of K-means algorithm viz. (i) dependence of clustering performance on initial choice of clusters centers and (ii) influence of the order of presentation of data on final cluster arrived. The steps of the algorithm are described as under: -

1. Set the initial parameters; ND : Number of data points, NV-number of variable describing each data, NC-number of initial cluster center chosen, NI-number of iteration and NCR-Number of final clusters arrived at
2. Generate ND non-repeated random integer sequence and get class string IR(i),
   i = 1,2...ND
3. Compute initial NC centers
$$C_{k,l} = \sum_{j=1}^{n(k)} \frac{P_k(j,l)}{n(k)} \quad \text{.....(2.1)}$$
   where $P_i(j, l)$ are patterns of i^{th} class and n(k) is the number of pattern in that class.
4. Create NC partitions using MCA: moving center algorithm [1], update the class string IR(i)
5. Repeat Step 2 to Step 5 NI-times for (NI * ND) matrix.
6. Calculate the frequencies of the NC^{NI} clusters and arrange them in descending order
   $NF(m), m = 1,2,...,NC^{NI}$
7. Choose NCR-stable clusters from
   NF(m), m = 1,2, ... , NCR.
8. Compute the stable seed points and again apply MCA and perform the statistical analysis to see the validity of the classification

The above algorithm always provides stable clusters. The only external input to the algorithm is the seed for random number generator. Some time recording of data could be erroneous or noisy due to human error or problems with the recording medium. The robust clustering algorithm should be resistant to these noises. And hence we have tested the robustness of ASCA against Gaussian noise as described below:

1 : Generate Gaussian Noise

2 : Add noise to data objects, say % (initially a small value)

3 : Apply ASCA for cluster formation

4 : Compare the results, if there is no major change in cluster formation, go to step 5, otherwise stop.

5 : Increase the level of noise by such that = + (for next iteration) and repeat Step 2 to Step 4

(Refer to Table 2 for the effect of noise of clustering performance)

2.1 Performance of ASCA against Noise

After establishing the consistency and clustering accuracy of ASCA we checked the clustering performance by introducing Gaussian noise in the data objects in the following ways.

a) **Pattern Level :** Patterns were randomly exposed to the Gaussian noise

b) **Attribute Level :** Different attributes of patterns were exposed to Gaussian noise in a random fashion.

The robustness check of ASCA has been carried out as per the algorithm described in section 2. Initially Gaussian noise was kept at a low level at $\alpha = 1\%$, then subsequently increased with an equal step size of $\beta = 2\%$. It has been observed that ASCA is able to sustain the Gaussian noise up to 25% in terms of clustering performance. Beyond that *PMF* (refer to section 4.2) becomes very high. The effect on the performance of ASCA after introducing the Gaussian noise in Iris Data [8] is depicted in Table 1.

Gaussian Noise in (%age)	5	10	15	18	22	25
PMF	0	0	2-6	7-10	11-16	High

Table 1

3. Some K-Center based clustering algorithms

Several K-center based clustering algorithms are available in literature. We have studied some of the most common and efficient algorithms viz. K-Means, Fuzzy c-means and K-Harmonics clustering algorithms along with ASCA, for comparing their performance on some well known data sets.

K-means Clustering algorithm is a crisp unsupervised clustering algorithm that considers non-overlapping partitions meaning that a data point either belongs to a

cluster or not. K-means clusters data objects iteratively by minimizing the objective function of the form

$$J = \sum_{i=1}^{N}\sum_{j=1}^{K} d_{ji} \quad \text{..... (3.1)}$$

where d_{ji} is the squared Euclidean distance from pattern i to cluster j, K is the number of clusters desired, N is the total number of data objects.

K-means does not guarantee unique clustering result because of its dependence on choice of initial cluster centers. It gives better results only when the initial partitions are close to the final solution [5].

Fuzzy c-means clustering algorithm is the fuzzy equivalent of nearest mean "hard" clustering algorithm [6]. In this approach, the boundaries between sub-partitions generated by the algorithm are vague. This means that each pattern of object data of a fuzzy partition belongs to different classes with different membership values. Bezdek used the concept of fuzzy logic, where decisions are made through analog weighting, and applied to the objective function J defined as

$$J = \sum_{i=1}^{N}\sum_{j=1}^{K} (u_{ji})^{q} d_{ji} \quad \text{..... (3.2)}$$

where u_{ji} is the degree of membership of i^{th} pattern in the j^{th} cluster and q is the fuzzification parameter

The exponent q controls the sharpness of the decision boundaries, so that when $q=1$, hard clusters are constructed, and when $q=\infty$, all patterns share the same membership to each cluster. According to Bezdek [3] the Fuzzy c-means algorithm always converges to strict local minima of the objective function starting from an initial guess of class membership values, but different values of fuzzy class membership might lead to land up in different local minima. An important factor in the use of Fuzzy c-means clustering is the optimal selection of parameter q. To this point, there is no automated way of selecting the best value of q for any one cluster, but most applications seem to find reasonable values lying somewhere between 1.2 and 4.0.

K-Harmonics developed by Bin Zhang [7] is a center based clustering algorithm and is insensitive to the initialization of centers. This insensitivity to initialization is attributed to a dynamic weighting function, which increases importance of the data points that are far from any centers in the next iteration. K-Harmonic Means algorithm also addresses the intrinsic problem by replacing the minimum distance from a data point to the centers, used by K-Means, by the Harmonic averages of the distances from the data point to all centers. The most general form of K-Harmonic Means objective function is defined as

$$J = \sum_{i=1}^{N} \frac{K}{\sum_{j=1}^{K} \left(\frac{1}{d_{ji}}\right)^{p}} \quad \text{..... (3.3)}$$

where the quantity inside the outer summation is the harmonic average of K squared distances and p is any real value > 2.

Bin Zhang [7] showed that the desired weighting function can be derived theoretically by using the p^{th} power of the Euclidean distance d_{ik}. For $p>2$, it boosts, in the next iteration, the participation of the data objects that are not close to any centers. The more centers are near a data object the smaller the weight for that data object. This has the effect of flattening out a local density that trapped more than one centers and reduces the chance of multiple centers being trapped by a single local cluster of data.

4. Experimental Comparison and Discussion

4.1 Data sets worked upon

The performance of the above discussed algorithms has been compared with following real world data sets:-

(i) Iris Data [8] : This data has three classes that represent three different varieties of Iris flowers, namely Iris setosa, Iris versicolor, Iris virginica. Fifty samples were obtained from each of three classes, thus a total of 150 samples are available. Every sample is described by a set of four attributes, viz. sepal length, sepal width, petal length, petal width. Two of the classes (virginica, versicolor) have a large overlap while setosa is well separated from the other two.

(ii) Ionosphere Data [9] : This radar data was collected by a system in Goose Bay, Lab rador. This system consists of a phased array of 16 high-frequency antennas with a total transmitted power on the order of 6.4 kilowatts. The targets were free electrons in the ionosphere. "Good" radar returns are those showing evidence of some type of structure in the ionosphere. "Bad" returns are those that do not; their signals pass through the ionosphere. Received signals were processed using an autocorrelation function whose

arguments are the time of a pulse and the pulse number. There were 17 pulse numbers for the Goose Bay system. Instances in this database are described by 2 attributes per pulse number, corresponding to the complex values returned by the function resulting from the complex electromagnetic signal. This data set is described by 351 instances having 34 continuous numeric attributes.

(iii) Letter Image Recognition Data [9] : The character images of 26 capital letters in the English alphabet were based on 20 different fonts and each letter within these 20 fonts was randomly distorted to produce a file of 20,000 unique stimuli. Each stimulus was converted into 16 primitive numerical attributes (statistical moments and edge counts), which were then scaled to fit into a range of integer values from 0 through 15. For the purpose of simplicity in experimentation we have taken 565 items corresponding to letter F and 595 items corresponding to letter A.

(iv) Wisconsin Breast Cancer Data [9] : This breast cancer databases was obtained from Dr. William H. Wolberg [10]. Samples arrive periodically as his clinical cases. The database therefore reflects this chronological grouping of the data. This data is nine dimensional having 699 samples in all, belonging to two classes i.e. Benign or Malignant.

(v) Wine Recognition Data [9] : This dataset is the result of a chemical analysis of wines grown in the same region in Italy but derived from three different cultivars. The analysis determined the quantities of 13 constituents found in each of the three types of wines. There are overall 178 instances.

4.2 Results and Discussion

To compute the error rate we define a Percentage Misclassification Factor (*PMF*) as

$$PMF = \frac{\sum_{i=1}^{K} mis_i}{N} \times 100 \qquad \text{..... (4.1)}$$

where mis_i is the number of misclassification in the i^{th} cluster corresponding to i^{th} class, K is the number of clusters formed, N is the total number of patterns in the data set.

Performance comparison of K-means, Fuzzy c-means, K-Harmonic means, and ASCA has been done by applying these algorithms to the above-mentioned data set. Each data set was presented to these algorithms randomly 50 times. The performances in terms of *PMF* values (in %age) for all the four algorithms on five data sets are summarized in Table 2.

It can be inferred from Table 2 that ASCA performs better on Ionosphere data, Letter Recognition data and Wine data in comparison to other center-adjustment clustering algorithms. Since ASCA addresses the cluster initialization problem by removing the outliers while deciding the cluster membership, probably that's why there is a consistency in the results obtained.

K-Means and Fuzzy c-means clustering algorithms are typically dependent on the choice of initial cluster centers. If improper centers are chosen then the algorithm may converge in one of the numerous local minima and that is why on some of the data set it lacks consistency. In Fuzzy c-means by varying the fuzziness parameter between 1.2 and 4 in equal small intervals, different cluster structures were formed which were not alike some of the times and hence creates little confusion occasionally.

K-Harmonic Means was found relatively insensitive to the choice of initialization of centers for $p > 2$. As the value of p was increased from 2 in equal intervals the consistency of results can be observed. But consistency in K-harmonics does not attribute to the clustering accuracy. Hence the *PMF* values using K-Harmonics are not the proper choice to arrive at appropriate conclusion.

	PMF (%age)			
Data Sets	**K-Means**	**Fuzzy c-means**	**K-Harmonic**	**ASCA**
Iris Data	11.33	12.66	11.13	**11.73**
Ionosphere	29.05	30.42	29.05	**28.83**
Leter F&A	5.75	7.02	6.21	**5.66**
Breat Cancer	4.14	4.06	4.46	**4.39**
Wine Data	5.61	9.21	7.47	**5.60**

Table 2.

Acknowledgements:

The authors are grateful to Dr. Laxmi Narain for his continuous patronage. The authors would also like to thank the UCI Repository for providing us the relevant data, without which this piece of work would be impossible.

References:

1. Kant, S., Rao, T.L., Sundaram, P.L., An Automatic and Stable Clustering Algorithm, Pattern Recognition Letters 15, 1994, p543-549
2. Fukunaga, K., Introduction to Statistical Pattern Recognition, San Diego, Academic Press, CA, 1990
3. Bezdek, J.C, Pattern Recognition with Fuzzy Objective Function Algorithms, Plenum, New York, 1981
4. Anderberg, M.R., Cluster Analysis for Applications, Academic Press, New York, 1973
5. Jain, A.K., Dubes, R.C., Algorithms for Clustering Data, Englewood Cliffs, Prentice Hall, 1988
6. Duda, R.O., and Hart, P.E., Pattern Classification and Scene Analysis, New York, Wiley, 1973
7. Zhang, B., Generalized K-Harmonic Means Boosting in Unsupervised Learning, Hewlett Packard Laboratory, 2000-137, October 12th 2000
8. Kendall, M.G. and Stuart, A., The Advance Theory of Statistics, vol 3, Griffin, London, 1961
9. UCI Machine Learning data repository, http://www.sgi.com/tech/mlc/db/
10. Mangasarian O. L. and Wolberg W. H: "Cancer diagnosis via Linear programming", SIAM News, Volume 23, Number 5, September 1990, pp 1 & 18

*Scientific Analysis Group, Defence R & D Organisation, Metcalfe House Complex, Delhi-110054