Volume 3, Issue 10, October 2013 ISSN: 2277 128X

International Journal of Advanced Research in Computer Science and Software Engineering

Research Paper

Available online at: www.ijarcsse.com

An Efficient Data Clustering Algorithm over IRIS Dataset

A Bhaskara Srinivas[1]
M.Tech,(Student)CSE, PVP SIT
India

B Vishnu Vardhan[2]
Asst. Prof. CSE, PVP SIT.
India

L Ravi Kumar[3]
Asst. Prof. IT, PVP SIT.
India

Dr. J Rajendra Prasad [4]
Head of the Dept. IT, PVP SIT
India

***Abstract:* We are proposing an efficient data clustering mechanism with incremental clustering algorithm with genetic feature, Clustering is a mostly unsupervised procedure and the majority of the clustering algorithms depend on certain assumptions in order to define the subgroups present in a data set. In our approach we are performing incremental clustering on IRIS dataset for optimal clusters apart from the traditional approaches.**

***Keywords---* Clustering, Genetic, PAM, Iris, Setosa, virginica, Versicolor**

I. INTRODUCTION

One of the most challenging analysis problems in the data mining domains is organizing large amounts of information. One approach to this problem is to group information based on the content of a collection of documents. Clustering is a widely used technique in data mining application for discovering patterns in original data. Most customized clustering algorithms are limited in handling datasets that contain unconditional attributes. However, datasets with unconditional types of attributes are common in real life data mining problem. For each pair of documents, a comparison vector is constructed that contains binary features that measure the overlap for highly informative but sparse features between the two documents and numeric features. The aggregating the comparison vector into one value that belongs to time interval. The aggregation pace is attain by taking a subjective standard the information gain has a tendency to favor features with many potential values over feature with fewer possible values, we used a normalized version of information gain, called gain ration as weighting metric[1][2][3]. Clustering is of prime importance in information analysis, machine knowledge and statistics. It is defines as the process of grouping N item sets into distinct clusters based on similarity or distance function A good clustering technique may yield clusters thus have high inter cluster and low intra cluster distance[7]. The objective of clustering is to maximize the similarity of the data points within each cluster and maximize dissimilarity diagonally clusters. [5][6]. Broadly speaking clustering algorithms can be divided into two types partitioned and hierarchical. Partitioning algorithms construct a partition of a database D of n objects into a set of clusters where k is a input parameter. Hierarchical algorithms create decomposition of the database D they are Divisive and Agglomerative. Hierarchical clustering builds a ranking of clusters, also known as a dendrogram. Every group node contains child cluster. An agglomerative clustering initiate with one-point (singleton) Clusters and recursively merges two or more most appropriate clusters. A divisive clustering starts with one group of all data points and recursively split into the most suitable clusters. The process continue until a stopping condition is achieved. There are two main issues in clustering techniques. Initially, discovering the best possible number of clusters in a given dataset and secondly, given two sets of clusters, computing relative measure of goodness between them.

II. GENERIC ALGORITHM

Evolutionary Algorithms is the name for the algorithms in the field of Evolutionary Computation which is a subfield of Natural Computing and already exists more than forty years. It was instinctive from the idea to use ideology of natural evolution as a paradigm for solving search and optimization problem in high dimensional combinatorial or continuous search spaces. The most widely known instances are genetic A detailed introduction to all these algorithms can be found e.g. in the manual of Evolutionary Computation [3]. At present the Evolutionary Computation field is very active. It involves fundamental research as well as a variety of applications in areas ranging from data analysis and machine learning to production processes, logistics or contracting and development, technological engineering, and others. Across every one of these fields, evolutionary algorithms have convinced practitioners by the results obtained on hard problems that they are very powerful algorithms for such applications. The general working principle of all instances of evolutionary algorithms is based on a program loop that involves simplified implementations of the operators mutation, recombination, selection, and fitness evaluation on a set of candidate solutions population of individuals) for a given problem. In this general setting, mutation corresponds to a modification of a single candidate solution, typically with a preference for small variations over large variations. Recombination corresponds to an exchange of components between two or more candidate solutions. Selection drives the evolutionary process towards populations of increasing average fitness by preferring better candidate solutions to proliferate with higher probability to the next generation than worse candidate solutions. By fitness evaluation, the calculation of a measure of goodness associated with candidate solutions is meant, i.e., the fitness function corresponds to the objective function of the Optimization problem at hand.

III. CHALLENGES AND LIMITATIONS

Currently used many clustering methods have the following limitations and shortcomings for data clustering in enterprise computing. unsuitable for large data sets, since these algorithms are basically sequential or small-scale parallel, with the run-time increasing rapidly with the data set size and the number of clusters

- Needs of some time-consuming pair wise computation or pre-processing on all the data objects. For example, the partitioning method, such as the PAM algorithm, needs sequential pair wise swap operations on all the data objects. The iterative spectral methods need to compute the pair wise similarities between all the data objects and to previously generate a n _ n similarity matrix for n data objects. Other hierarchical clustering methods also need to previously construct some structured hierarchy data like a feature graph or tree. The locality-based algorithms need to search the whole data space, and accordingly compute the distribution density of every data object;

- No guarantee for the clustering optimality, since the evaluation function for clustering algorithm usually stagnates in a local minima, rather than a global minima corresponding to the optimal clustering

- Sensitivity of the clustering performance and clustering quality to the cluster shape and cluster distribution. Particularly, non-symmetric shape and non-uniform distribution of clusters may give rise to significant deterioration of clustering performance and quality

- Unable to well suppress the noise affect. Most partitioning methods are usually unable to completely eliminate adjoint outliers from clustering outcome. The shrinking strategy of the CURE hierarchical algorithm also does not work very well for eliminating outliers

- Poor clustering performance for high-dimensional data. For example, Clustering Using Representatives has a time density of O(n^2) on low-dimensional data, but O(n^2) log n on high-dimensional data; the DBSCAN's time density is claimed to be O(n log n), but actually DBSCAN is unsuitable for high-dimensional data since all the cluster structures may not be characterized by global density parameters specified by users;

- no learning ability to use previous clustering results or prior probability distributions for current clustering, with every data set being clustered from scratch no openness in the sense that the dynamic change of clustered data objects are usually not allowed during the algorithm execution

IV. PROPOSED WORK

We have also tested the one-dimensional clustering algorithm with a real dataset as the Iris dataset. The Iris dataset [15] was first used and even created by Fisher [16] in his pioneering research work on linear discriminant analysis, and today it is still an up-to-date, standard pattern recognition problem for testing discriminant techniques and algorithms. In this well-known and classical multi class pattern recognition problem, three classes of Iris flavours (setosa, versicolor and virginica) have to be classified according to four continuous discriminant variables measured in centimeters:SL length, SL width, PL length and PL width. Fig.1 shows the three classes. We have represented all variables of this four-dimensional dataset: the SL length, the SL width, PL length and the PL width. The triangle marks represent the Iris setosa, the circles are the Iris versicolor items and the squares correspond to the Iris virginica. It is well known that this dataset only contains two clusters with an obvious separation. The Iris set oasis in one of those clusters, while the other two species, Irisversi color and Iris virginica, are in the other cluster. As the Iris dataset contains three classes we have only employed the black, grey and white colors to represent the Iris setosa, Irisversicolor and Irisvirginica data items, respectively. The final tape's state is achieved in the iteration 1291 although perfect clustering is not obtained in this case. Only the black zone which represents the Iris set is as shows a compact state. Some individuals corresponding to the Irisversicolor(grey cluster) are included in the middle of the Iris virginica zone(white cluster) and viceversa. Fig. 2 shows the chainmap of the Iris dataset. In this case three localmaxima can be clearly distinguished , each of one corresponding to an individual cluster.

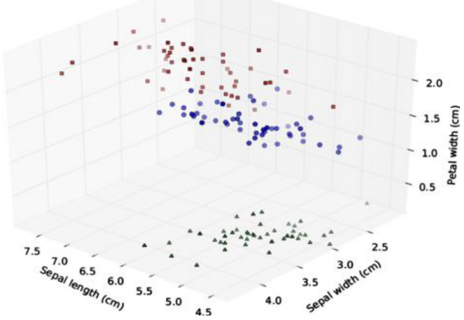


Figure 1 : The Iris dataset. The triangle marks represent the Iris setosa, the circles correspond to the Iris versicolor and the squares are used for the Iris virginica.

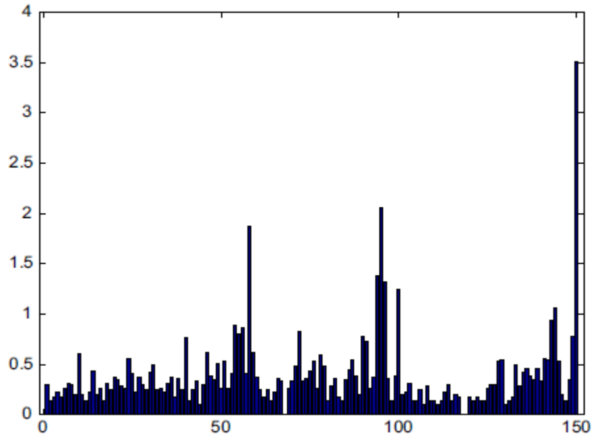


Fig2: Chain map of iris dataset

V. ALGORITHM: OPTIMAL BASSED CLUSTERING

```
 Step 1: Initialize number of clusters based on intra cluster variances

Step 2: Select random selection of the data points and centroid $C(c_1,c_2….c_n)$

Step 3: for i:=0 to Initial population_size

Step 4: if i==1

      a.   Calculate the fitness between      $C(c_1,c_2….c_n)$ and data cell
      b.   Calculate the optimal fitness with data cell and neighborhood.
      c.   Mutate chromosome and find fitness.
Else
      a.   Compute_centroid (clusters)
      b.   Calculate the distance between $C(c_1,c_2….c_n)$ and data cell.
      c.   Calculate the optimal distance with data cell and neighborhood.
      d.   Mutate chromosome and find fitness.

Step 6:  place the data cell in to respective cell
Step 7: next
Step 8: terminate
```

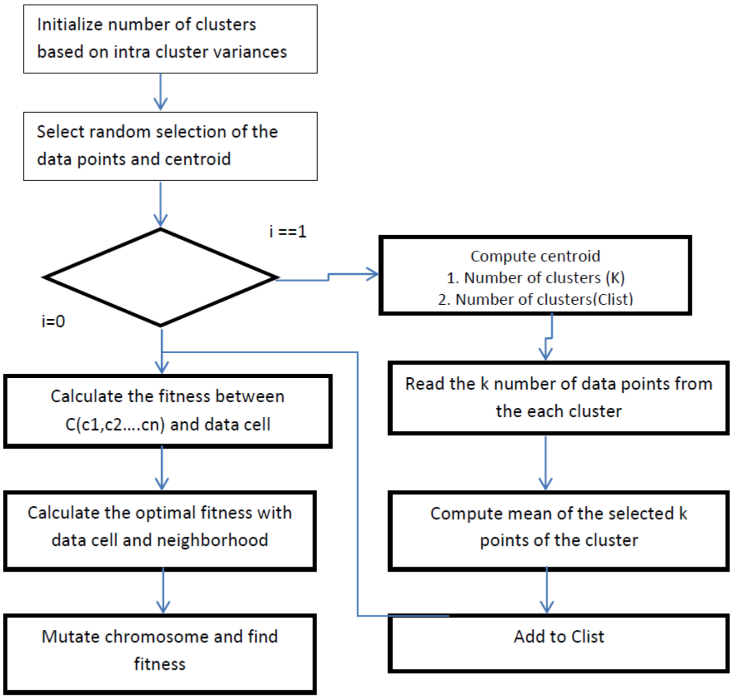

Initialize number of clusters based on intra cluster variances
Select random selection of the data points and centroid
i ==1
Compute centroid
1. Number of clusters (K)
2. Number of clusters(Clist)
i=0
Calculate the fitness between C(c1,c2….cn) and data cell
Read the k number of data points from the each cluster
Calculate the optimal fitness with data cell and neighborhood
Compute mean of the selected k points of the cluster
Mutate chromosome and find fitness
Add to Clist

Fig.3 Flow chart Representation of Optimal based clustering

Initially we need to consider the number of data points as chromosomes, count of the number of data points can be considered as initial population We find the appropriate number of clusters by computing an index based on the intra-cluster variances. The number of clusters in the dataset corresponds to the minimum of this index for all the possible number of clusters. The index J is defined as follows:

$$J = \left\| \sum_{i=1}^{c} \sigma_i^2 \right\|$$

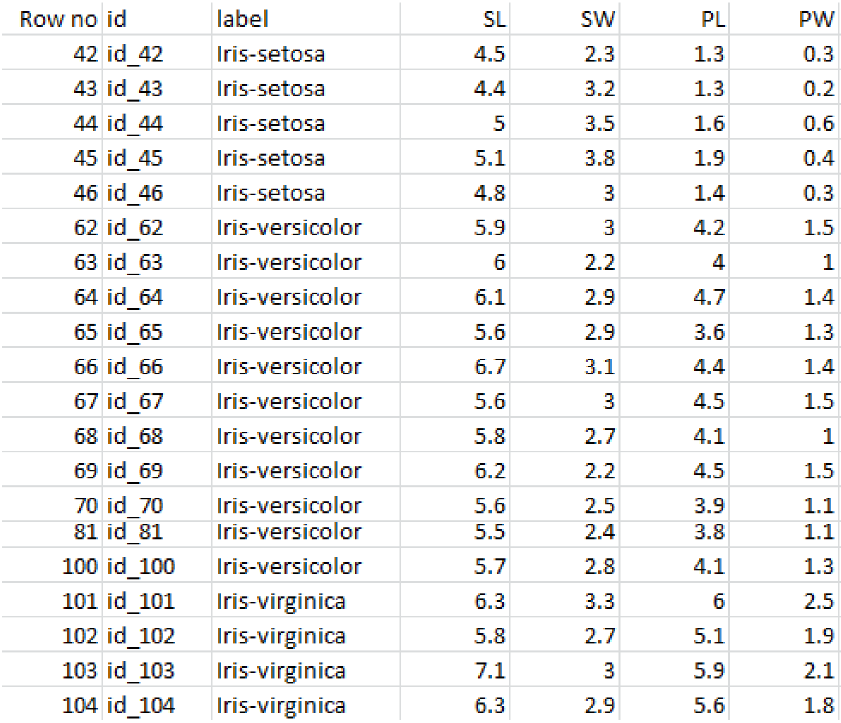
where c in the number of clusters and σ^2 is the intra-cluster variance of the cluster i. As we are considering multivariate datasets, we use the variance of each variable separately and we compute the norm of the result vector in order to get the index J. On some occasions some clusters could be composed just for one data item which can be easily detected. Usually these spare data items are very near to the centroid of a cluster. When it occurs, we introduce are finement to the solution by associating the spare items to the nearest clusters and reducing the number of classes.

```
Compute_centroid(clusters)
{
K----number of clusters
Clist---Cluster centroid
    1.  Read the k number of data points from the each cluster
    2.  Compute mean of the selected k points of the cluster
    3.  Add to Clist

}
```

VI. EXPERIMENTAL ANALYSIS

We have also tested the one-dimensional cellular automata- based clustering algorithm with are dataset as the Iris dataset. The Iris dataset [15] was first used and even created by Fisher [16] in his pioneering research work on linear discriminant analysis, and today it is still an up-to-date, standard pattern recognition problem for testing discriminant techniques and algorithms. In this well-known and classical multi class pattern recognition problem, three classes of Iris flowers (setosa, versicolor and virginica) have to be classified according to four continuous discriminant variables measure d in centimeters : SL length, SL width, PL length and PL width. Fig. 1 shows the three classes. We have represented all variables of this four-dimensional dataset: these PL,PW,SL and SW.



Row no	id	label	SL	SW	PL	PW
42	id_42	Iris-setosa	4.5	2.3	1.3	0.3
43	id_43	Iris-setosa	4.4	3.2	1.3	0.2
44	id_44	Iris-setosa	5	3.5	1.6	0.6
45	id_45	Iris-setosa	5.1	3.8	1.9	0.4
46	id_46	Iris-setosa	4.8	3	1.4	0.3
62	id_62	Iris-versicolor	5.9	3	4.2	1.5
63	id_63	Iris-versicolor	6	2.2	4	1
64	id_64	Iris-versicolor	6.1	2.9	4.7	1.4
65	id_65	Iris-versicolor	5.6	2.9	3.6	1.3
66	id_66	Iris-versicolor	6.7	3.1	4.4	1.4
67	id_67	Iris-versicolor	5.6	3	4.5	1.5
68	id_68	Iris-versicolor	5.8	2.7	4.1	1
69	id_69	Iris-versicolor	6.2	2.2	4.5	1.5
70	id_70	Iris-versicolor	5.6	2.5	3.9	1.1
81	id_81	Iris-versicolor	5.5	2.4	3.8	1.1
100	id_100	Iris-versicolor	5.7	2.8	4.1	1.3
101	id_101	Iris-virginica	6.3	3.3	6	2.5
102	id_102	Iris-virginica	5.8	2.7	5.1	1.9
103	id_103	Iris-virginica	7.1	3	5.9	2.1
104	id_104	Iris-virginica	6.3	2.9	5.6	1.8

Fig. 1

The triangle marks represent the Iris setosa, the circles are the Irisversicolor items and the squares correspond to the Irisvirginica. It is well known that this dataset only contains two clusters with an obvious separation. The Iris set oasis in one of those clusters, while the other two species, Irisversicolor and Iris virginica, are in the other cluster.

TABLE I
COMPARISON OF GENETIC WITH DENSITY BASED ALGORITHM

Cluster|Alg	**Genetic Alg**	**Density Based Alg**
Cluster1	132 items	97 items
Cluster2	18 items	53 items

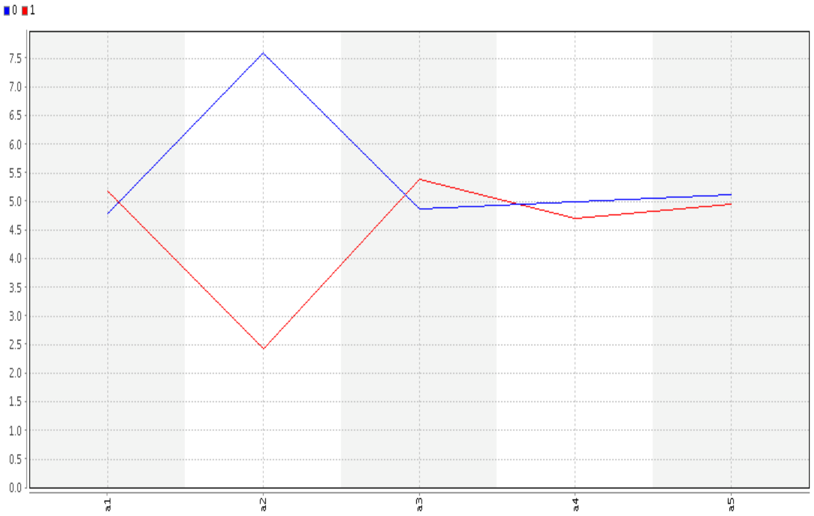


Fig 4.Experiment Results for Cluster 0 &1

VII. CONCLUSION

Optimal clustering of data items can be achieved through this mutation based incremental clustering algorithm than the traditional approaches, here we are considering the intra cluster variances for specifying the number of clusters, instead of random selection of centroids from the data sets we are computing the centroids with means of the intra clusters, for optimal clusters we are measuring the fitness through the mutation based distance.

REFERENCES

[1] Jain AK, Dubes RC, Algorithms for clustering. Prentice- Hall, Englewood C, 1988

[2] Kaufman L, Rousseeuw PJ, Findings Clusters in data. An introduction to cluster analysis. Canada, 1990.

[3] Jain AK, Murty M N, Flynn P J, Data clustering: a review. ACM Computer Survey 31(3), 1999, pp 265-323

[4] B Kamgar-parsi, AK Jain, automatic aircraft recognition: toward using human similarity measure in a recognition system. In: IEEE Computer Society Conference On computer vision and pattern recognition, 1999, pp 268-273

[5] S Santini, R Jain, Similarity Measures. IEEE Trans Pattern Anal Mach Intell 21(9), 1999, 871-883.

[6] Latecki LJ, Lakamper R, IEEE Trans Pattern Anal Mach Intell 22(10), 1185-1190, Shape similarity measure based on correspondence of visual parts, 2000.

[7] BS Everitt, S Landau, M Leese, A Cluster Analysis. Arnold London, 2001.

[8] J. Valente de Oliveira and W Pedrycz, Advances in Fuzzy Clustering and its Applications, 2007.

[9] plant Data, 2006. Ardian Krisanto Poernomo, Data Understanding for Iris

[10] Rui Xu, Donald Wunsch, Survey of Clustering Algorithms. IEEE Trans of Neural Networks. Volume.16 No.3

[11] Irszula Boryczka, Fining Groups in Data: Cluster Analysis with Ants. IEEE: Proceedings of the sixth International Conference on Intelligent System Design and applications, 2006

[12] Donghai Guan, Weiwei Yuan, Young-Koo Lee, Andrey Gavrilov and Sungyoung Lee, " Combining Multi-layer perception and Kmeans for Data Clustering with background Knowledge", The 2007 International Conference on Intelligent Computing(ICIC 2007, Springer), August 21-24, Qingdao, China.

[13] Donghai Guan, Andrey V.Gavrilov, Weiwei Yuan, Young-Koo Lee and Sungyoung Lee, "A Novel Hybrid Neural Network for Data Clustering", The 2007 International Conference on Machine Learning, Models, Technologies and Applications, WorldComp 2007, June 25-28, Las vegas, United States Of America.

[14] Erik Cuevas, Daniel Zaldivar and Raul Rojas, Fuzzy Segmentation applied to face segmentation, Freie University Berlin, Institute of Information Technology.9, D-14195 Berlin, Germany. Techical Report B-04-09, June 2004.

[15] Ana L.N. Fred and Anil K. Jain, Data Clustering using Evidence Accumulation, Supported by the Portuguese Foundation for Science and Technology (FCT), Portugese Ministry of Science and Technology, and Feder, under grant POSI/33143/SRI/2000, and ONR grant no. N00014-01-1-0266.