BCD-Clustering algorithm for Breast Cancer Diagnosis

Dr Sulochana Wadhwani, Dr A.K. Wadhwani, Tripty Singh, Dr. Sarita Singh Bhadauoria

Abstract- Cluster analysis or clustering is the task of assigning a set of objects into groups (called clusters) so that the objects in the same cluster are more similar (in some sense or another) to each other than to those in other clusters. Cluster analysis can be used to differentiate between different types of calcification clusters in a mammogram image. In this application, actual position does not matter, but the cluster size shape and intensity is considered as a time based feature, for each image that was taken over time. This technique allows, for example, accurate measurement of the rate a radioactive tracer is delivered to the area of interest, without a separate sampling of clusters, an intrusive technique that is most common today. Choosing cluster centers is crucial to the clustering. In this paper we compared two fuzzy algorithms : Fuzzy c-means algorithm and the new fuzzy clustering and fuzzy merging algorithm. Fuzzy c-means algorithm uses the reciprocal of distances to decide the cluster centers. The representation reflects the distance of a feature vector from the cluster center but does not differentiate the distribution of the clusters. The Breast Cancer Diagnosis-Clustering algorithm uses Gaussian weights and the generated cluster centers are more representative. When a feature vector is of equal distance from two cluster centers, it weighs more on the widely distributed cluster than on the centrally located cluster. We implemented both algorithms MATLAB.

Index Terms- BCD-Clustering algorithm, empty clusters, hard clusters, soft clusters, and mammogram

I. Introduction

The notion of a cluster varies between algorithms and is one of the many decisions to take when choosing the appropriate algorithm for a particular problem. At first the terminology of a cluster seems obvious: a group of data objects. However, the clusters found by different algorithms vary significantly in their properties, and understanding these cluster models is the key to understanding the differences between the various algorithms.

Typical cluster models include:

1. *Connectivity models:* for example hierarchical clustering builds models based on distance connectivity.
2. *Centroid models:* for example the k-means algorithm represents each cluster by a single mean vector.
3. *Distribution models:* clusters are modeled using statistic distributions, such as multivariate normal distributions used by the Expectation-maximization algorithm.
4. *Density models:* for example DBSCAN and OPTICS defines clusters as connected dense regions in the data space.
5. *Subspace models:* in Biclustering (also known as Co-clustering or two-mode-clustering), clusters are modeled with both cluster members and relevant attributes.

Group models: Some algorithms (unfortunately) do not provide a refined model for their results and just provide the grouping information.

A clustering is essentially a set of such clusters, usually containing all objects in the data set. Additionally, it may specify the relationship of the clusters to each other, for example a hierarchy of clusters embedded in each other. Clustering can be roughly distinguished in:

1. *Hard Clustering:* Each object belongs to a cluster or not
2. *Soft Clustering (also: fuzzy clustering):* Each object belongs to each cluster to a certain degree (e.g. a likelihood of belonging to the cluster)

There are also finer distinctions possible, for example:

- Strict partitioning clustering: here each object belongs to exactly one cluster
- Strict partitioning clustering with outliers: object can also belong to no cluster, and are considered outliers.
- Overlapping clustering (also: alternative clustering, multi-view clustering): while usually a hard clustering, objects may belong to more than one cluster.
- Hierarchical clustering: objects that belong to a child cluster also belong to the parent cluster
- Subspace clustering: while an overlapping clustering, within a uniquely defined subspace, clusters are not expected to overlap.

Clustering is the process of grouping feature vectors into classes in the self-organizing mode. Let $\{x^{(q)}: q = 1,\ldots,Q\}$ be a set of Q feature vectors. Each feature vector $x^{(q)} = (x_1^{(q)}, \ldots, x_N^{(q)})$ has N components. The process of clustering is to assign the Q feature vectors into K clusters $\{c^{(k)}: k = 1, \ldots, K\}$ usually by the minimum distance assignment principle. Choosing the representation of cluster centers (or prototypes) is crucial to the clustering. Feature vectors that are farther away from the cluster center should not have as much weight as those that are close. These more distant feature vectors are outliers usually caused by errors in one or more measurements or a deviation in the processes that formed the object [2].

The simplest weighting method is arithmetic averaging. It adds all feature vectors in a cluster and takes the average as prototype. Because of its simplicity, it is still widely used in the clustering initialization.

The arithmetic averaging gives the central located feature vectors the same weights as outliers. To lower the influence of the outliers, median vectors are used in some proposed algorithms.

To be more immune to outliers and more representative, the fuzzy weighted average is introduced to represent prototypes:

$Z_n^{(k)} = \Sigma_{\{q:\, q\in\, k\}}\, w_{qk}x^{(q)}{}_n;$ (1)

Rather than a Boolean value 1 (true, which means it belongs to the cluster) or 0 (false, does not belong), the weight w_{qk} in equation (1) represent partial membership to a cluster. It is called a *fuzzy weight*. There are different means to generate fuzzy weightsOne way of generating fuzzy weights is the reciprocal of distance

$w_{qk} = 1/\, D_{qk}$, (w_{qk}= 1 if $D_{qk} = 0$) (2)

It was used in earlier fuzzy clustering algorithms [2]. When the distance between the feature vector and the prototype is large, the weight is small. On the other hand, it is large when the distance is small. Using Gaussian functions to generate fuzzy weights is the most natural way for clustering. It is not only immune to outliers but also provides appropriate weighting for more centrally and densely located vectors. It is used in the *fuzzy clustering and fuzzy merging* (FCFM)

We implemented the *fuzzy c-means* (FCM) algorithm and the fuzzy clustering and merging algorithm in Java, applied the algorithms to several data sets and compared the weights of the two algorithms.

II. Clustering Algorithms

The clustering groups a sample set of feature vectors into K clusters via an appropriate similarity (or dissimilarity) criterion (such as distance from the center of the cluster).

The k-means Algorithm

The k-means algorithm assigns feature vectors to clusters by the minimum distance assignment principle [5], which assigns a new feature vector $x^{(q)}$ to the cluster $c^{(k)}$ such that the distance from $x^{(q)}$ to the center of $c^{(k)}$ is the minimum over all K clusters. The basic k-means algorithm is as follows:

- Put the first K feature vectors as initial centers
- Assign each sample vector to the cluster with minimum distance assignment principle.
- Compute new average as new center for each cluster
- If any center has changed, then go to step 2, else terminate.

The advantages of the method are its simplicity, efficiency, and self-organization. It is used as initial process in many other algorithms. The disadvantages are: 1) K must be provided; 2) it is a linearly separating algorithm.

III. BCD-Clustering algorithm for Breast Cancer Diagnosis

1. K Is Initial Number Of Clusters, Imax Is The Iteration Of Fuzzy
2. C-Means, P Is For The Weight
3. Input Initial Number Of Clusters K, Imax, P
4. Initialize Weights Of Prototype
5. Standardize The Initial Weight Over K
6. Starting Fuzzy C-Means Loop
7. Standardize Cluster Weights Over Q
8. Compute New Prototype Center
9. Compute New Weight
10. End Of Fuzzy C-Means Loop
11. Assign Feature Vector According The Max Weight
12. Remove Clusters With No Feature Vectors Eliminate(0),
13. Compute Arithmetic Center Of Clusters.
14. Calculate Sigma And Xie_Beni Value.
15. Calculate Fuzzy Weight
16. Get Variance (Mean-Square Error) Of Each Cluster
17. Compute Modified XB

Standardize the Weights over Q. During the FCM iteration, the computed cluster centers get closer and closer. To avoid the rapid convergence and always grouping into one cluster, we use

$$w[q,k] = (w[q,k] - w_{min})/(\, w_{max} - w_{min}) \quad (5)$$

before standardizing the weights over Q. Where w_{max}, w_{min} are maximum or minimum weights over the weights of all feature vectors for the particular class prototype.

Removing Empty Clusters- After the fuzzy clustering loop we add a step (Step 8) to eliminate the empty clusters. This step is put outside the fuzzy clustering loop and before calculation of modified XB validity [2]. Without the elimination, the minimum distance of prototype pair used in Equation (8) may be the distance of empty cluster pair. We call the method of eliminating small clusters by passing 0 to the process so it will only eliminate the empty clusters.

For modified XB. After the fuzzy c-means iteration, for the purpose of comparison and to pick the optimal result, we add Step 9 to calculate the cluster centers and the modified Xie-Beni clustering validity κ [7]:[2]

The XB validity is a product of compactness and separation measures [10]. The compactness-to-separation ratio ν is defined by Equation (6). [2]

$$\nu = \{(1/K)\Sigma_{(k=1,K)}\, \sigma_k{}^2\}/D_{min}{}^2 \quad (6)$$

$$\sigma_k{}^2 = \Sigma_{(q=1,Q)}\, w_{qk} \,||\, x^{(q)} - c^{(k)} \,||^2 \quad (7)$$

D_{min} is the minimum distance between the cluster centers.

The Modified Xie-Beni validity κ is defined as

$$\kappa = D_{min}{}^2/\, \{\Sigma_{(k=1,K)}\, \sigma_k{}^2\, \} \quad (8)$$

The variance of each cluster is calculated by summing over only the members of each cluster rather than over all Q for each cluster, which contrasts with the original Xie-Beni validity measure.

$$\sigma_k^2 = \sum_{\{q:\ q \text{ is in cluster } k\}} w_{qk} \parallel x^{(q)} - c^{(k)} \parallel^2 \qquad (9)$$

IV. Data Set

We used the Wisconsin breast cancer data set (WCBD data sets to run both algorithms: The Wisconsin breast cancer data set (WCBD) [7] consists of 200 randomly selected from more than 500 breast cancer vectors of the University of Wisconsin Medical School. Each feature vector has 30 features in [0, 1]. The vectors are labeled for two classes. One label is attached to 121 vectors while the other is attached to 79 vectors.

The geological data set [7] is labeled for K = 2 classes. Each of the Q = 70 feature vectors has N = 4 features. The data labels are estimates by humans that give 35 to each class. These were assigned by humans providing their best guesses.

V. Results

The BCD-Clustering algorithm uses reciprocal distance to compute the fuzzy weights. When a feature vector is of equal distance from two cluster centers, it weights the same on the two clusters no matter what is the distribution of the clusters. It cannot differentiate the two clusters with different distributions of feature vectors. Therefore, the BCD-Clustering algorithm is more suited to data that is more or less evenly distributed around the cluster centers. The BCD-Clustering algorithm lumps the two clusters with natural shapes but close boundaries into a large cluster. For some difficult data such as WBCD data, it is hard to for the BCD-Clustering algorithm to cluster the very closed classes together without the help of other mechanisms such as elimination of small clusters. Results are cited at the end of this paper. Results are shown in Table 1and 2

VI. Conclusion and Future Work

The BCD-Clustering algorithm uses Gaussian weights, which are most representative and immune to outliers. Gaussian weights reflect the distribution of the feature vectors in the clusters. For a feature vector with equal distance from two prototypes, it weighs more on the widely distributed cluster than on the narrowly distributed cluster. The BCD-Clustering algorithm outperforms the BCD-Clustering algorithm on all the test data we used in this paper.

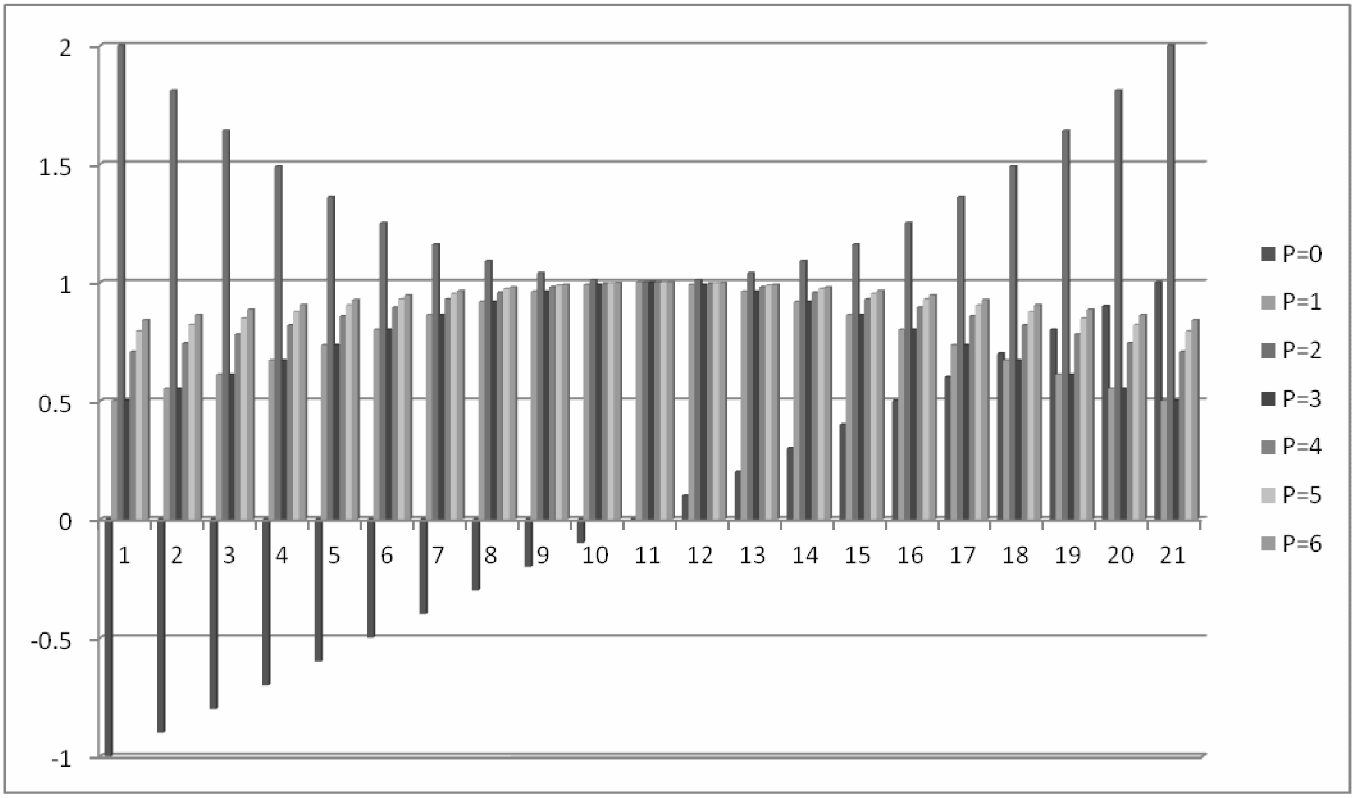
VII. ACKNOWLEDGMENTS

Authors are thankful to **UGC New Delhi** for providing financial assistance to this research paper for IJSAT journal. We would like to express our deep thanks for assisting us in this research work to **Madhav Institute of Technology and Science Gwalior** and **East Point College of Engineering for Women Bangalore** and **Cancer Hospital of Research Institute Hospital Gwalior**, M.P India .We are thankful to **Dr B R Shrivastava Director & Dr Neeraj Sharma** Head of Radiology ,CHRI Gwalior for generous cooperation for analyzing and validating our research work.

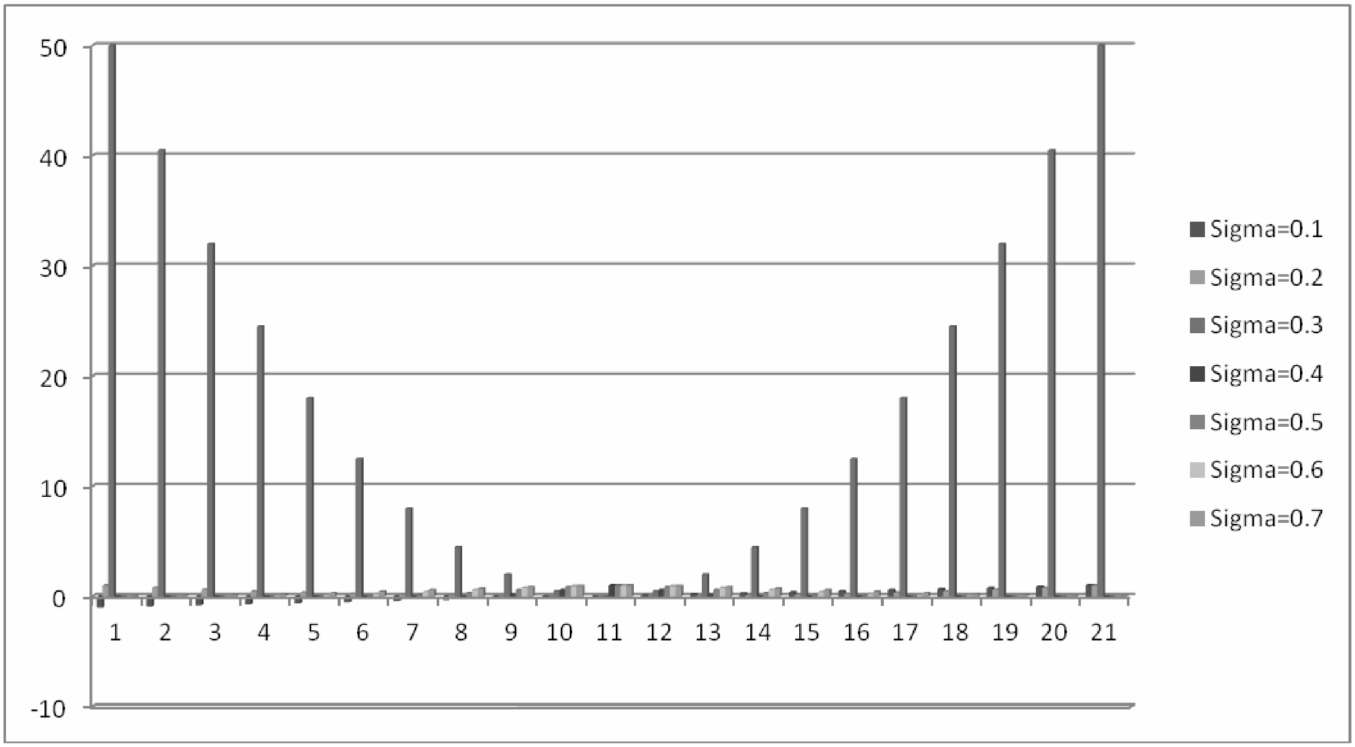
P=0	P=1	P=2	P=3	P=4	P=5	P=6
-1	0.5	2	0.5	0.707107	0.793701	0.840896
-0.9	0.552486	1.81	0.552486	0.743294	0.820554	0.862145
-0.8	0.609756	1.64	0.609756	0.780869	0.84798	0.883668
-0.7	0.671141	1.49	0.671141	0.819232	0.87553	0.905114
-0.6	0.735294	1.36	0.735294	0.857493	0.902583	0.926009
-0.5	0.8	1.25	0.8	0.894427	0.928318	0.945742
-0.4	0.862069	1.16	0.862069	0.928477	0.951731	0.963575
-0.3	0.917431	1.09	0.917431	0.957826	0.971683	0.978686
-0.2	0.961538	1.04	0.961538	0.980581	0.987012	0.990243
-0.1	0.990099	1.01	0.990099	0.995037	0.996689	0.997516
0	1	1	1	1	1	1
0.1	0.990099	1.01	0.990099	0.995037	0.996689	0.997516
0.2	0.961538	1.04	0.961538	0.980581	0.987012	0.990243
0.3	0.917431	1.09	0.917431	0.957826	0.971683	0.978686
0.4	0.862069	1.16	0.862069	0.928477	0.951731	0.963575
0.5	0.8	1.25	0.8	0.894427	0.928318	0.945742
0.6	0.735294	1.36	0.735294	0.857493	0.902583	0.926009
0.7	0.671141	1.49	0.671141	0.819232	0.87553	0.905114
0.8	0.609756	1.64	0.609756	0.780869	0.84798	0.883668
0.9	0.552486	1.81	0.552486	0.743294	0.820554	0.862145

1	0.5	2	0.5	0.707107	0.793701	0.840896

Sigma=0.1	Sigma=0.2	Sigma=0.3	Sigma=0.4	Sigma=0.5	Sigma=0.6	Sigma=0.7
-1	1	50	1.92875E-22	3.72665E-06	0.003866	0.043937
-0.9	0.81	40.5	2.57676E-18	4.00653E-05	0.011109	0.07956
-0.8	0.64	32	1.26642E-14	0.000335463	0.028566	0.135335
-0.7	0.49	24.5	2.28973E-11	0.002187491	0.065729	0.216265
-0.6	0.36	18	1.523E-08	0.011108997	0.135335	0.324652
-0.5	0.25	12.5	3.72665E-06	0.043936934	0.249352	0.457833
-0.4	0.16	8	0.000335463	0.135335283	0.411112	0.606531
-0.3	0.09	4.5	0.011108997	0.324652467	0.606531	0.75484
-0.2	0.04	2	0.135335283	0.60653066	0.800737	0.882497
-0.1	0.01	0.5	0.60653066	0.882496903	0.945959	0.969233
0	0	0	1	1	1	1
0.1	0.01	0.5	0.60653066	0.882496903	0.945959	0.969233
0.2	0.04	2	0.135335283	0.60653066	0.800737	0.882497
0.3	0.09	4.5	0.011108997	0.324652467	0.606531	0.75484
0.4	0.16	8	0.000335463	0.135335283	0.411112	0.606531
0.5	0.25	12.5	3.72665E-06	0.043936934	0.249352	0.457833
0.6	0.36	18	1.523E-08	0.011108997	0.135335	0.324652
0.7	0.49	24.5	2.28973E-11	0.002187491	0.065729	0.216265
0.8	0.64	32	1.26642E-14	0.000335463	0.028566	0.135335
0.9	0.81	40.5	2.57676E-18	4.00653E-05	0.011109	0.07956
1	1	50	1.92875E-22	3.72665E-06	0.003866	0.043937



Graph 1



Graph 2

References

[1] E. Anderson, "The iris of the Gaspe peninsula" Bulletin American Iris Society, Vol. 59, 2-5, 1935.

[2] Liyan Zhang "Comparison of Fuzzy c-means Algorithm and New Fuzzy Clustering and Fuzzy Merging Algorithm" , www.cse.unr.edu/~lzhang/fuzzyCluster

[3] J. C. Bezdek, etc. Convergence Theory for Fuzzy c-Means: Counterexamples and Repairs, IEEE Trans. Syst., September/October 1987.

[4] Maria Colmenares & Olaf WolkenHauer, "An Introduction into Fuzzy Clustering", http://www.csc.umist.ac.uk/computing/clustering.htm, July 1998, last update 03 July,2000

[5] Marti Hearst, K-Means Clustering, UCB SIMS, Fall 1998, http://www.sims.berkeley.edu/courses/is296a-3/f98/lectures/ui-bakground/sld025.htm.

[6] Uri Kroszynski and Jianjun Zhou, Fuzzy Clustering Principles, Methods and Examples, IKS, December 1998

[7] Carl G. Looney A Fuzzy Clustering and Fuzzy Merging Algorithm, CS791q Class Notes, http://www.cs.unr.edu/~looney/.

[8] Carl G. Looney Pattern Recognition Using Neural Networks, Oxford University Press, N.Y., 1997.

[9] Carl G. Looney "Chapter 5. Fuzzy Clustering and Merging", CS791q Class Notes, http://www.cs.unr.edu/~looney/.

[10] Ramze Rezaee M, Lelieveldt B P F and Reiber J H C, A New Cluster Validity Index for the Fuzzy c-mean, Pattern Recognition Letters, (Netherlands) Mar 1998.

[11] M.J.Sabin, Convergence and Consistency of Fuzzy c-means /ISODATA Algorithms, IEEE Trans. Pattern Anal. Machine Intell., September 1987.

First Author –Tripty Singh, Department of Information Science and Engg, EPCEW, Bangalore, India, email: triptysmart@gmail.com

Second Author – Dr. Sarita Singh Bhadauoria, Department of Electronics Engineering, MITS Gwalior, M.P India, email: saritamits61@yahoo.co.in

Third Author – Dr Sulochana .Wadhwani, Department of Electrical Engineering, MITS Gwalior , M.P India, email: wadhwani_arun@rediffmail.com

Forth Author Dr AK Wadhwani, Department of Electrical Engineering, MITS Gwalior , M.P India, email: sulochana_wadhwani1@rediffmail.com