Novel Auxiliary Techniques in Clustering

Mohammad Taheri, Reza Boostani

Abstract— **Clustering is grouping of patterns according to similarity or distance in different perspectives. Various data representations, similarity measurements and organization manners are led to several classes of clustering methods. In this paper a new combinatorial method is proposed that iteratively uses another clustering method such as Rival Penalized Competitive Learning (RPCL) or K-Means as the core of clustering system. Moreover, some novel auxiliary techniques are suggested to increase the clustering performance. The proposed method has been compared with well known clustering methods such as K-Means, its improvement, ISODATA and DSRPCL2. The new combinatorial technique can detect the drawbacks of core clustering method and improve its efficiency. Our method is applied on some standard multi-class datasets. After clustering, labels of grouped samples in each cluster are compared with their real class labels to show the accuracy of clustering.**

Index Terms— **Clustering, Merging, Pattern Recognition, Splitting, Unsupervised learning**

I. Introduction

Clustering methods have been employed in different fields such as pattern analysis [4], decision making and various applications in machine learning include data mining, document retrieval [3], [6], and segmentation in image processing [7], provide enough motivation to look more precisely at this content. Some researches at the role of art [14] in clustering, use of clustering to find minimum spanning tree[8], vice versa or even optimization applications[9], [10] show the vast application of this field. Most important discrepancy of clustering methods is usually their approach to use of similarities to grouping data instances. Generally speaking, clustering methods are divided to two branches: Hierarchical and partitioning [2], [16]. Hierarchical methods use agglomerative or divisive policies which production of larger clusters from merging smaller ones or splitting a cluster to some parts, respectively determine their manners. There are some other divisions are introduced in [2]. But success of combinatorial approach must be considered [11] that often iteratively use merging and splitting techniques serially or in parallel even in nonhierarchical methods. These techniques usually propose a numerical criterion in comparison of a threshold to identify whether one or more clusters can be split or merged. ISODATA [15] uses distribution variance of instances belonged to specified cluster as the splitting condition and distance of center of clusters is considered as the merging criterion. In addition of mentioned conditions, ISODATA executes a classic method k-Means [12] iteratively and tries to improve it in each step by splitting or merging some clusters. K-Means is based on determining centers of k clusters to minimize squared distance of each instance from center of its container (nearest) cluster. Inasmuch as k is a constant in this method and usually desired or exact number of clusters is not identified, ISODATA can often improve it by changing k in each step toward its exact value. Criteria introduced in ISODATA algorithm are week from various perspectives. In this article, novel criteria are proposed which described and compared to ISODATA. The rest of this paper is organized as follows. In section II, RPCL [5] clustering method is described. In section III, a novel technique has been introduced to initialize the centers of clusters. Section IV, proposes two criterions for splitting and merging. A restorative technique based on neighborhood information is proposed in V. Section VI explains a sample system that uses all of techniques and in part VII, results of introduced methods are compared visually on two artificial datasets and numerically by classification rate criterion on two standard datasets *wine* and *iris* chosen from UCI database [20]. Finally a conclusion is presented.

Manuscript received March 22, 2007.

Mohammad Taheri and Reza Boostani are with the department of computer science and engineering of Shiraz University, Shiraz, Iran. (phone: +989171021361; e-mail: mamadtaheri@gmail.com , boostani@shirazu.ac.ir

II. Rival Penalized Competitive Learning

Another implementation of K-Means is based on rewarding to winner clusters. This process takes place by considering sample data, instance by instance. In each step, selected instance rewards to nearest cluster as the winner such that center of the cluster approaches to the instance. In RPCL method, in addition of rewarding to the winner, rivals will be penalized too in order of (1) and (2).

$$W_{new}^{\rho} = W_{old}^{\rho} + \Delta W^{\rho} \qquad \rho = 1,2,...,K \tag{1}$$

Where, K is the number of clusters. W_{new}^{ρ} and W_{old}^{ρ} are new and old centers of ρ^{th} cluster respectively. So ΔW^{ρ} is the location change of ρ^{th} cluster center and other variables are defined previously.

$$\Delta W^{\rho} = \begin{cases} \alpha_c (X^n - W_{old}^{\rho}) & if \quad \rho = c(n) \\ -\alpha_r (X^n - W_{old}^{\rho}) & if \quad \rho \neq c(n) \end{cases} \tag{2}$$

$$0 < \alpha_c, \alpha_r < 1$$

Where X^n is the instance that rewards and punishes; α_c and α_r are coefficients of reward and penalty respectively.

A cost functional approach to RPCL is introduced in [5] and tries to minimize the functions in (3) - (5).

$$E(W)=E_1(W)+E_2(W) \quad E_1(W)=E_{MSE} \tag{3}$$

$$E_{MSE}(W)=\frac{1}{2}\sum_{n=1}^{N} f\left(X^n,W^{c(n)}\right)^2 \; c(n)\in\{1,2,...,K\} \tag{4}$$

$$E_2(W)=\frac{2}{P}\sum_{n=1}^{N}\sum_{\substack{k=1\\k\neq c(n)}}^{K} f\left(X^n,W^k\right)^{-P} \tag{5}$$

Where N and K are the number of instances of dataset and desired clusters respectively. $W=[W^1,W^2,...,W^K]$ is the set of centers of clusters. Also, if X_n is n^{th} instance, $W^{c(n)}$ represents nearest cluster center to X_n. Indeed, $c(n)$ is index of the cluster with nearest center to X_n. Finally, $f(a, b)$ is a distance function between a and b. $E(W)$ is the total error where $E_1(W)$ is E_{MSE} introduced in (4) and $E_2(W)$ approximates similarities between clusters. If similarity of instances in each cluster increases, $E_1(W)$ decreases and whenever $E_2(W)$ is decreased, clusters are more different and distinguishable than before. According to this cost function, α_r is defined with respect to α_c as (6).

$$\alpha_r=\alpha_c f(X^n,W_{old}^{\rho})^{-p-2} \tag{6}$$

Where, p is a constant set to 0.2 in this article. α_c is set to 0.1 and decreases in each iteration to be sure of convergence. One of the most important features of RPCL is its ability to push centers of extra clusters out of sample data scope. Our experiments show that α_r must be so less than α_c to avoid of throwing exact clusters out of scope. To this, $f(X^n,W_{old}^{\rho})$ must be always bigger than one. Although this condition is not controllable but to increase its chance in this article, each data set is scaled such that minimum distance between instances would be one. In this way, repeated instances must be removed earlier.

DSRPCL is a Distance *Sensitive RPCL* such as what has been described here. In this article, Euclidean distance is used as $f(.,.)$. In each step, if all of rival clusters are penalized, the method is named *DSRPCL1* but if only the nearest one of them (next nearer after winner) is penalized, it is known as *DSRPCL2*. In our special experiments, DSRPCL never pushed a cluster center out of scope so desired number of clusters are equal to initial one.

III. Novel Technique #1: Initialization of Centers of Clusters

In this article a clustering method must be used as the center of system and then proposed techniques try to improve its performance. Applicability of each technique may be dependent of the main method. The technique proposed in this section is usable before the first use of main method if it requires a policy to choose some points as the initial centers of clusters. Some of them such as K-Means or RPCL may generate these points randomly and of course their results are so dependent to their initial points. For example in RPCL, if density of cluster centers in a scope is more than needed, some of cluster centers may be thrown out of data scope although in some other part of feature space, there is shortage of cluster center to completely grouping instances located there. Also unsuitable choosing initial points may guide the method toward a local optimum which is not globally the best. So far some methods have been proposed to initialize centers of clusters such as variable division and leader algorithm [1]. Our experiments show, the more distance between initial centers, the more chance to find the best final clusters. So in this article, a method is considered which is proposed by Mico [13] and is based on distance between the centers of clusters. From now, W^{ρ} represents the ρ^{th} initial point as the center of a cluster. This method chooses these points from instances of sample data. At the first, W^1 is chosen randomly from instances and then other points are selected by (7).

$$W^{\rho}=\underset{x\in(M-B_{\rho})}{\operatorname{argmax}}\left(\sum_{j=1}^{\rho-1} f\left(x,W^j\right)\right) \quad \rho=2,3,...,K \tag{7}$$

Where, M is the set of instances of sample data and $B_{\rho}=\{W^1,W^2,...,W^{\rho-1}\}$ is the set of chosen instances. K is the number of initial clusters and other variables are defined as same as before. Indeed, in each step, one unselected instance with the most summation of distances from selected ones is chosen as the next point. But this method seems to be inefficient if K is bigger than $2d$ where, the feature space is d-dimensional (d is the number of features). As the simplest example, in a 1-dimensional space depicted in Fig. (1), after selecting A and B as two instances with the most distance from each other, then each other point has a same distance summation from them. So what is the best one?

Figure (1): selecting initial points in 1-dimensional space

With the preceding discussion in the mind, formula in (7) has been changed to (8).

$$W^{\rho}=\underset{x\in(M-B_{\rho})}{\operatorname{argmax}}\left(\sum_{j}^{\rho-1}\sqrt{f(x,W^j)}\right)\rho=2,3,...,K \tag{8}$$

With this approach, in Fig. (1), the point with the most equality in distance from A and B will be chosen as the next point. Also the first point will be chosen by (9) and (10) instead of randomly.

$$W^1=\underset{X\in M}{\operatorname{argmax}}\left(f(X,S)\right) \quad S=(s_1,s_2,...,s_d) \tag{9}$$

$$s_i=\min_{n=1,2,...,N}\left(x_i^n\right) \qquad i=1,2,...,d \tag{10}$$

At the beginning, a point on the boundary of dataset or even a corner is a good candidate to be chosen as W^1. S is a point at the one of the corners of smallest hyper rectangle which surrounds all of instances of sample data and W^1 is the farthest instance from S. si is the value of i^{th} feature of S and N is the number of instances. The space is d-dimensional and x_i^n identifies the value of i^{th} feature of n^{th} instance. Relation (10) identifies a

corner which none of its feature values is greater than the same feature of other corners. So (10) can change to introduce another corner. To justify this method of selecting W^1, it seems that S and W^1 are approximately the heads of one of diameters of surrounding hyper rectangle. S is exactly located at the corner but there is often no instance at that point. Unlike S, W^1 is one instance of sample data which may be not exactly located at a corner.

IV. Novel technique #2: New conditions of splitting and merging

Some clustering methods are not based on merging or splitting but using this technique may improve their performance. Main motivation of ISODATA is also improving k-Means by this technique. ISODATA uses special criterions and suffers some lacks to detect correct clusters. Novel criterion and conditions to merging and splitting are proposed here to overcome these lacks.

A. Merging condition

Merging is a technique used in many clustering methods to prevent production of extra clusters. Therefore so many criterions are yet proposed to check the condition of merging. One of the most important of them is the distance between centers of two clusters which is the most important criterion used in ISODATA for merging [15]. In this case, if the centers of two clusters are nearer than specified threshold, the main condition is satisfied. This criterion doesn't consider other properties of clusters such as their largeness and distribution.

In this section, a novel criterion is proposed which is based on the expected internal distances of a cluster. This criterion introduced in (11) is the average of instance Solitude Radiuses D_{MSR} that is the *Mean of Solitude Radiuses* of instances belonged to the specified cluster. Solitude Radius of an instance is its distance from nearest cluster mate. Hence each instance in a cluster is expected to be at D_{MSR} from its nearest cluster mate. Also a threshold is needed here; therefore another criterion is proposed which is based on external distances of two clusters. D_{min} of two specified clusters A and B is the minimum distance between them, instance to instance and is presented in (12).

$$D_{MSR}^{A} = \underset{X^n \in A}{Mean}\left(f\left(X^n, SN^n\right)\right) \tag{11}$$

$$D_{\min}(A,B) = \min\left(\begin{array}{l} f\left(X^a, X^b\right) \\ \left| \begin{array}{l} W^{c(a)} = A\,, \\ W^{c(b)} = B \end{array} \right. \end{array}\right) \tag{12}$$

Where, D_{MSR}^{A} is the D_{MSR} of cluster A and SN^n is the nearest instance to X^n such that both of them are belonged to a same cluster. Other variables are declared previously.

$$D_{\min}(A,B) \le \beta \times \max(D_{MMI}^{A}, D_{MMI}^{B}) \tag{13}$$

With these definitions, the merging condition is proposed by (13). Indeed, if two instances from two clusters are nearer than a specified threshold, their clusters are suitable to merge. This threshold is β times greater than the maximum D_{MSR} of these clusters. β is a constant and considered 1.5 in this article.

B. Splitting condition

Splitting is a technique to break a large cluster to some smaller ones if needed. This may improve the power of categorization. ISODATA detects large clusters by checking variance of their instances. If the variance is more than a specified threshold, that cluster is considered such large that can break to two smaller ones.

In ISODATA, conditions of merging and splitting are neither disjoint nor complement. Indeed, they are not even related to each other. Centers of two clusters may be such near to each other that the condition of merging is satisfied but variance of instances belonged to the produced cluster is greater than determined threshold such that it will be split. Even two clusters merged previously may be produced again. To be sure that conditions proposed in this method are complementary, splitting condition is tried to be inverse of merging condition. Although the proposed methods in here, are not completely complementary since of complexity in inverting action, but experiments show that this approach is so effective to remove cycles in merging and splitting that consequently increases the chance of convergence.

Suppose $V=\{V^1, V^2, ..., V^d\}$ is the set of vectors in d-dimensional feature space such that each V^i and V^j are perpendicular where i is not equal to j and size of each vector approximates distribution of sample data in associated direction. Principle Component Analysis (PCA) [17] method produces V by computing the eigenvectors of Covariance matrix of instances. To check the splitting condition of a cluster, V is computed for instances belonged to it. Then instances are projected on each vector separately. After projection, the cluster is considered in d independent 1-dimensional.space.

Splitting condition is satisfied for a specified cluster if one of its projections is qualified to be split. So the problem has been reduced to 1-dimensional problem. In this case, each instance has only one value associated to its unique feature. Then instances will be sort on this feature value. Each 1-dimensional cluster is qualified to be split if distance between two consequent instances is more than a threshold which is β times greater than D_{MSR} of the original cluster. β is the constant used in (13). Since boundary instances are usually far from cluster, only middle 1/3 of instances are considered for splitting condition. If a cluster must be split, its center is removed and two new centers on both sides of it in direction of splitter vector will be produced. Although this dimension reduction may disturb inverting action, this approach is so effective in merging-splitting cycle removal.

In comparison with ISODATA, proposed novel criterions and conditions are approximately complementary and consider the

clusters more comprehensive, instance by instance and overall. In addition, ISODATA parameters are set for the whole of clusters but our techniques act on each cluster or pair of clusters independently except that β is global. Also the number of parameters which should be set in proposed technique is considerably less than the number of ISODATA parameters.

V. Novel technique #3: k-Nearest Neighbor restoration

Regardless of which clustering method has been used as the main method, according to the complexity of dataset, some methods may make mistakes about some samples of data located at the boundary of clusters. In this article, a new technique is used to overcome this problem.

A well known classification method named *k-Nearest Neighbors (KNN)* has been introduced in [1] that classifies each instance of unseen data by investigation in *k* nearest instances of training data. In classification job, each training data has a label and the goal is labeling new unseen data, instance by instance. In this method, each one of *k* nearest neighbors of the new instance which are selected from the training data votes to its label. Finally, the label with the maximum votes will be determined as the label of new instance. This section goes on with a proposed clustering technique inspired from *KNN*.

After reaching the stop condition of main method, the whole of instances belonged to a unique cluster will be labeled uniquely but different with other clusters. Finally a post-processing based on *KNN* method will be done to overcome the lack of producing small extra clusters although this process has more benefits as an auxiliary technique in any clustering method. This technique takes place instance by instance. In each step, one instance of sample data will be chosen as a new unseen instance in *KNN* method and other ones assumed to be training data with their associated label. So *KNN* overwrites the label of the selected instance and this process goes on for each other ones of sample data. This post-processing is done on the sample data iteratively until no change in labels would be occurred.
This technique can be beneficial to collect small clusters surrounding a larger one and join them. But it can also be useful from other perspectives. For example in k-means method, each instance belongs to a cluster with the nearest center regardless of the largeness and distribution of them. So a small cluster may be known as the container of some boundary instances of a large cluster since the distance from centers is only regarded criterion. *KNN* technique can overcome to this lack to a certain extent. It also can recognize clusters with unusual shapes which are not describable by a few parameters in any clustering method. This technique is proposed previously [10], [16] where *k* is equal to 1. In the main *KNN* method, all of *k* nearest neighbors has a same weight in voting but in this article, this weight is inversely proportional to its distance from the new instance.

This technique runs iteratively until no change takes place. Sometimes the stop condition is never reached and *KNN* suffers a loop. Therefore a maximum number of iterations should role as another stop condition.

VI. Introduction of a combinatorial clustering system

Here, a clustering system that uses techniques described in this article has been explained. This system uses DSRPCL2 as the main method although a simpler method such as *k-Means* can also be used. Since of using DSRPCL2, sample data is scaled such that the minimum distance of instances would be 1. Then some points from instance will be chosen as the initial centers of clusters by the related technique proposed in section III. Afterwards, iteratively, DSRPCL2, KNN restoration and merging is run on sample data. If no merging is occurred, splitting condition is checked. In final iteration, no merging or splitting condition is satisfied and consequently, the system result converges unless the number of iterations exceeds a specified maximum. This plan is depicted in Fig. (2).

VII. Results

At the beginning, proposed method has been compared with ISODATA in clustering two artificial 2-dimentional datasets. The first dataset 'dades.Mat' and some ISODATA parameters specially set for this dataset are reachable in [18] and Fig.(3) shows the result. Tables (1) and (2) introduce ISODATA parameters and their values used in this article respectively. Some instances in fig.(3) which are surrounded by circles are cases belonged to an incorrect cluster. Fig.(4) depicts result of the run of the method proposed in section VI on this dataset. Comparison of figs.(3) and (4) clears the effect of KNN restoration technique more than before. In this article k is set to 5 and initial number of clusters is usually not important and effective on final result. This matter shows the stability of the system.

Table (1): ISODATA parameters description

σS2	Maximum variance in a cluster
DMERGE	minimum distance between clusters
ND	Desired number of clusters
NMIN_EX	Minimum number of instances of a cluster
NMERGE	Maximum number of clusters which can be merged at the same time

Table (2): ISODATA parameters values used in this article

Name	Value	Name	Value
σ_S^2	7	D_{MERGE}	10
N_D	4	N_{MIN_EX}	15
N_{MERGE}	2	Initial number of clusters	1

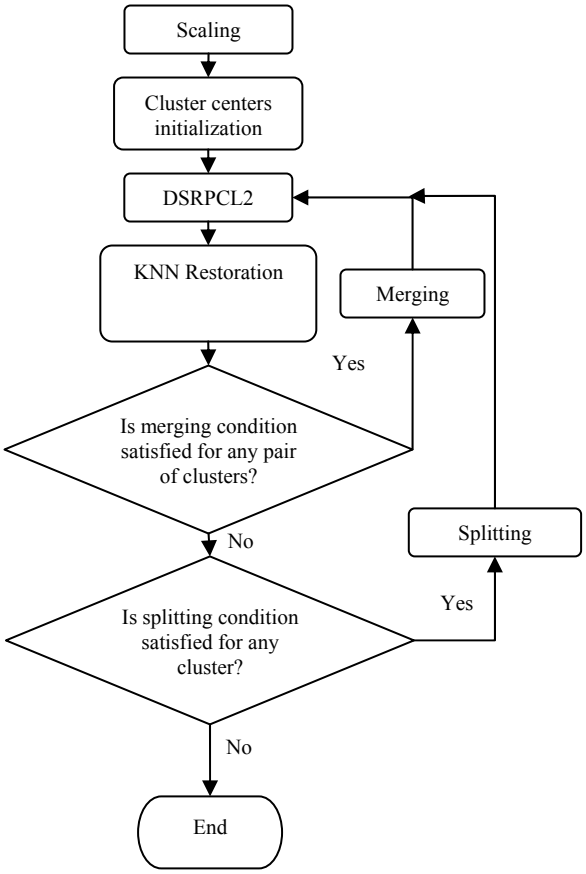

Figure (2): Plan of the combinatorial clustering system used in this article

Another artificial dataset with 360 sample data in 4 or 12 clusters is provided from [19]. Fig.(5) shows the result of ISODATA on this dataset when 4 clusters are desired but with no parameters ISODATA can split sample data to 12 clusters unless it acts completely like k-Means when 12 clusters are desired. With this condition, final result is so dependent to initial centers of clusters. Fig.(6) shows the best result of ISODATA on this dataset when it's capable to merge or split clusters. Fig.(7) depicts the result of our method on this dataset and correctly splits it to exactly 12 clusters. 10 initial clusters have been considered in this article.

Finally, k-Means, DSRPCL2 and our method are executed on two datasets of UCI database [20] (wine & iris) which are described in Table (3). Samples of these datasets also have a class label for classification goal. Although clustering is not a tool for classification but classification rate has been introduced as a goodness measurement of clustering [1]. In this way, each produced cluster is labeled same as the labels of its sample data belonged to it. If there is more than one label, one of them with majority is chosen. Then each instance in a cluster will be classified as the label of its cluster and if it comes into opposition with its original label, it will be classified incorrectly. Classification rate is the number of correct instance classification with respect to the number of instances in dataset. *Iris* has 3 repeated instances which have been removed here.

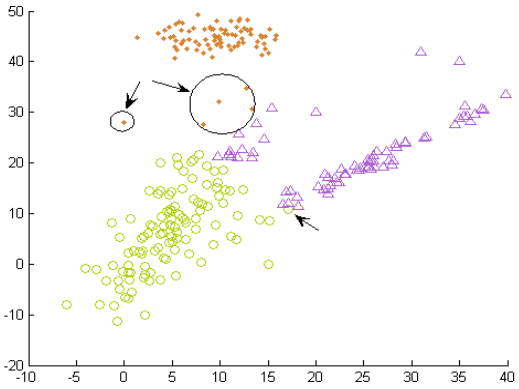

Figure (3): Result of ISODATA run on 'dades.Mat'

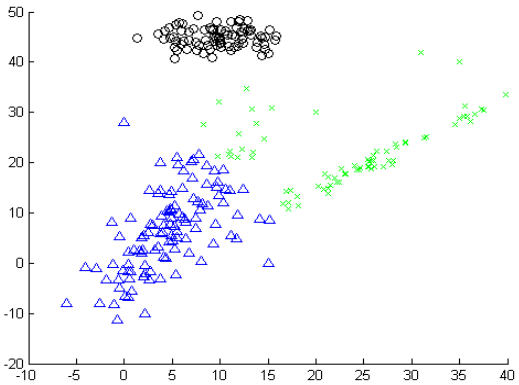

Figure (4): Result of proposed system run on 'dades.Mat'

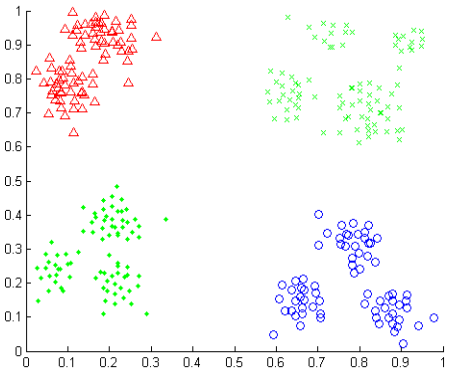

Figure (5): ISODATA executed on 360 artificial instances with 4 desired clusters

Table (3): properties of UCI datasets used

Dataset	Number of instances	Number of features	Number of classes
IRIS	150-3=147	4	3
WINE	178	13	3

In this way, our method with 10 initial clusters converges to 5 final clusters in wine and 13 clusters in iris. β has been set to 0.5 in classification. Of course, a normalization preprocessing part has been done on each dataset before classification. In normalization, each feature is scaled such that minimum value of that feature in the whole of dataset would be 0 and the maximum one would be 1. Indeed, it maps all features to a same

interval between 0 and 1 inclusively. Table(4) shows the classification rate of k-Means, DSRPCL2 and our method on normalized wine and iris datasets with 5 and 13 desired clusters respectively.

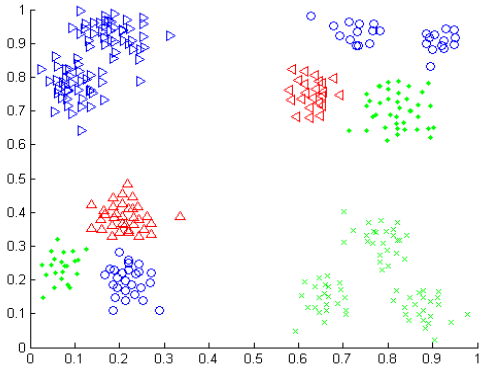

Figure (6): ISODATA executed on 360 artificial instances with 12 desired but 8 produced clusters

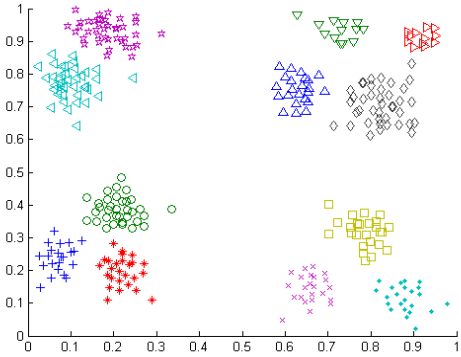

Figure (7): Proposed method executed on 360 artificial instances

Table (4): Classification rate of introduced methods on normalized datasets ***wine*** and ***iris***

Clustering method	Wine	Iris
	desired clusters=5	desired clusters=13
k-Means	94.38	95.24
DSRPCL2	**94.94**	87.07
proposed	93.26	**95.92**

Conclusion

In this article, some novel clustering techniques have been proposed and a combinatorial system has been compared with a well known classic method k-Means, its improvement ISODATA and a new method called DSRPCL2. This comparison takes place visually, on two artificial 2-dimensional datasets, and shows that precision of our method in categorization is more than ISODATA which uses special conditions to split or merge clusters. Also classification rate of our method is better than others on dataset iris from UCI database although this result is not good on wine which is a simple dataset. Of course it must be considered that classification rate is not a complete criterion to determine how a method is good.

Improving complementariness of merging and splitting conditions is one of following works in future. Also we try to make disjoint conditions. It seems that if complementary conditions of merging and splitting are disjoint too, in some special conditions which are under research, the convergence of system is proven.

References

[1] Wiley, John, Ltd., Sons, Statistical Pattern Recognition, second edition, Webb, Andrew R., Ltd, QuinetiQ, Malvern, UK, Copyright 2002

[2] Jain, A.K., Dubes, R., *Algorithms for Clustering Data*, *Prentice* Hall, 1988.

[3] ANDERBERG, M.R., *Cluster Analysis for Applications*. Academic Press, Inc., New York, NY., 1973.

[4] Jinwen, Ma., Taijun, Wang, A cost-function approach to rival penalized competitive learning (RPCL), IEEE Trans Image Process, 15:2755-61, . 2006.

[5] Rasmussen, E., *Clustering Algorithms.*, editors, Frakes, W.B., Yates, R. Baeza., Information Retrieval Data Structures and Algorithms., pp. 419-42., Prentice Hall. Englewood Cliffs, 1992.

[6] Salton, G., *Development in Automatic Text Retrieval*. Science, Vol. 253, pp. 974-980, 1991.

[7] Jain, A.K., Flynn, P.J., *Image Segmentation Using Clustering*, editors, Ahuja, N., Bowyer, K., Advances in Image Understanding: A Festschrift for Azriel Rosenfeld., IEEE Computer Society Press., pp. 65-83, 1996.

[8] Lee., R.C.T., *Cluster analysis and its applications*., editor, Tou, J.T., Advances in information systems science, Vol. 8, pp. 169-292., Plenum, New York, NY, USA, 1981.

[9] Al-Sultan, Khaled S., Khan, M. Maroof, *Computational experience on four algorithms for the hard clustering problem*, Pattern Recognition Letters, Vol. 17,Issue 3, pp. 295-308, Elsevier Science Inc. New York, NY, USA, 1996.

[10] Mishra, S.K., Raghavan, V.V., *An empirical study of the performance of heuristic methods for clustering*, editors, Gelsema, E.S., Kanal, L.N., Pattern Recognition in Practice, pp. 425-436, North Holland, 1994.

[11] Murty, M.N., Krishna, G.A., *computationally efficient technique for data clustering*, Pattern Recognition, Vol. 12, pp. 153, 1980.

[12] McQueen, J., *Some methods for classification and analysis of multivariate observations.*, 5th Berkeley Symposium on mathematics, Statistics and Probability, Vol. 1, pp. 281-298, 1967.

[13] Mic´o, M.L., Oncina, J., Vidal, E., *A new version of the nearest-neighbour approximating and eliminating search algorithm (AESA) with linear preprocessing time and memory requirements.* Pattern Recognition, Vol. 15, pp. 9–17, 1994.

[14] Jain, A. K., Murty, M.N.,Flynn, P.J., *Data Clustering: A review*, ACM Computing Surveys (CSUR), Vol. 31, Issue 3, pp. 264–323, ACM Press New York, NY, USA, 1999.

[15] Ball, G.H., Hall, D.J., *ISODATA, a Novel Method of Data Analysis and Classification*, Technical report, Stanford University, Stanford, USA, 1965.

[16] Jain, A.K., Murty M.N., Flynn, P.J., *Data Clustering: A review*, Dataclustering.cse.msu.edu

[17] Lindsay, I. Smith, A Tutorial on Principal Components Analysis, 2002 http://csnet.otago.ac.nz

[18] http://www.mathworks.com/matlabcentral

[19] http://nnmi.kaist.ac.kr/lectures/2001ee538/ Cluster_Data.Zip

[20] UCI Machine Learning Repository http://www.ics.uci.edu/~mlearn/databases/