Pattern Recognition Letters 32 (2011) 1701–1705

Contents lists available at SciVerse ScienceDirect

Pattern Recognition Letters

journal homepage: www.elsevier.com/locate/patrec

A new algorithm for initial cluster centers in *k*-means algorithm

Murat Erisoglu *, Nazif Calis, Sadullah Sakallioglu

Department of Statistics, Faculty of Science and Letters, Cukurova University, 01300 Adana, Turkey

ARTICLE INFO

Article history:
Received 28 September 2010
Available online 6 August 2011
Communicated by F. Roli

Keywords:
k-Means algorithm
Initial cluster centers
Rand index
Error percentage
Wilks' lambda test statistic

ABSTRACT

Clustering is one of the widely used knowledge discovery techniques to reveal structures in a dataset that can be extremely useful to the analyst. In iterative clustering algorithms the procedure adopted for choosing initial cluster centers is extremely important as it has a direct impact on the formation of final clusters. Since clusters are separated groups in a feature space, it is desirable to select initial centers which are well separated. In this paper, we have proposed an algorithm to compute initial cluster centers for *k*-means algorithm. The algorithm is applied to several different datasets in different dimension for illustrative purposes. It is observed that the newly proposed algorithm has good performance to obtain the initial cluster centers for the *k*-means algorithm.

1. Introduction

Clustering is an important tool for a variety of applications in data mining, statistical data analysis, data compression, and vector quantization. The goal of clustering is to group data into clusters such that the similarities among data members within the same cluster are maximal while similarities among data members from different clusters are minimal. Clustering algorithms can be broadly classified into hierarchical and non-hierarchical clustering algorithms.

Hierarchical algorithms decompose a dataset X of n objects into several levels of nested partitioning (clustering), represented by a dendrogram (tree). Non-hierarchical clustering algorithms construct a single partition of a dataset X of n objects into a set of k clusters, such that the objects in a cluster are more similar to each other than to objects in different clusters.

k-Means algorithm (Mac Queen, 1967) is the most well known and the fast method in non-hierarchical cluster algorithms. Because of the simplicity of *k*-means algorithm, this algorithm is used in various fields. *k*-Means algorithm is a partitioning clustering method that separates data into k mutually excessive groups. Through such the iterative partitioning, *k*-means algorithm minimizes the sum of distance from each data to its clusters. *k*-Means algorithm is very popular because of its ability to cluster a kind of huge data, and also outliers, quickly and efficiently. However, *k*-means algorithm is very sensitive to the designated initial starting points as cluster centers. *k*-Means does not guarantee unique clustering because we get different results with randomly chosen initial clusters. The final cluster centroids may not be the optimal ones as the algorithm can converge into local optimal solutions. An empty cluster can be obtained if no points are allocated to the cluster during the assignment step. Therefore, it is quite important for *k*-means to have good initial cluster centers.

Several methods proposed to solve the cluster initialization for *k*-means algorithm. A recursive method for initializing the means by running k clustering problems is discussed by Duda and Hart (1973). A variation of this method consists of taking the entire data into account and then randomly perturbing it k times. For the initial cluster center, Jain and Dubes (1988) applied the *k*-means with several times by randomly selected initial values and selected the average of these final cluster centers.

Bradley and Fayyad (1998) proposed the refinement algorithm that builds a set of small random sub-samples of the data, then clusters data in each sub-samples by *k*-means. All centroids of all sub-samples are then clustered together by *k*-means using the *k*-centroids of each sub-sample as initial centers. The centers of the final clusters that give minimum clustering error are to be used as the initial centers for clustering the original set of data using *k*-means algorithm.

Likas et al. (2003) proposed the global *k*-means algorithm which is an incremental approach to clustering which dynamically adds one cluster center at a time through a deterministic global search procedure consisting of N (with N being the size of the dataset) executions of the *k*-means algorithm from suitable initial positions.

Khan and Ahmad (2004) proposed Cluster Center Initialization Algorithm (CCIA) to solve cluster initialization problem. CCIA is based on two observations, which some patterns are very similar to each other. It initiates with calculating mean and standard

* Corresponding author. Tel.: +90 3223386084; fax: +90 3223386070.
E-mail addresses: merisoglu@cu.edu.tr (M. Erisoglu), ncalis@cu.edu.tr (Nazif Calis), sadullah@cu.edu.tr (S. Sakallioglu).

0167-8655/$ - see front matter © 2011 Elsevier B.V. All rights reserved.
doi:10.1016/j.patrec.2011.07.011

deviation for data attributes, and then separates the data with normal curve into certain partition. CCIA uses *k*-means and density-based multi scale data condensation to observe the similarity of data patterns before finding out the final initial clusters. The experiment results of the CCIA performed the effectiveness and robustness this method to solve the several clustering problems.

Deelers and Auwatanamongkol (2007) proposed an algorithm to compute initial cluster centers for *k*-means algorithm. They partitioned the data set in a cell using a cutting plane that divides cell in two smaller cells. The plane is perpendicular to the data axis with the highest variance and is designed to reduce the sum squared errors of the two cells as much as possible, while at the same time keep the two cells far apart as possible. Also they partitioned the cells one at a time until the number of cells equals to the predefined number of clusters, *k*. In their method the centers of the *k* cells become the initial cluster centers for *k*-means algorithm.

The rest of the paper organized as follows. In Section 2, we present our proposed algorithm to compute initial cluster centers for *k*-means algorithm. Section 3 describes the comparison criteria those are used in the experiments. In Section 4, the algorithm was applied to Iris, Wine, Letter and Ruspini datasets. Also the algorithm was compared with randomly initial cluster centers. Conclusion follows in Section 5.

2. Proposed algorithm

In this section, the proposed algorithm to compute initial cluster centers for optimizing *k*-means algorithm is explained. This algorithm based on the choosing the two of the p variables that best describes the change in the dataset according to two axes. Firstly absolute value of the variation coefficient in Eq. (1) is considered for the determination of the main axis,

$$cv_j = \left|\frac{s(x_j)}{\bar{x}_j}\right|, \quad j = 1, 2, \ldots, p \tag{1}$$

where $s(x_j)$ and $\bar{x}_j$ are the standard deviation and mean of the j variable respectively. The main axis is selected as the variable which has maximum value of the coefficient of variation. When coefficient of the variation is used to determine the main axis, this is eliminating the problems of the size of terms and differences in measurement units. After determining the main axis, the correlation coefficient is used to determine the second axis. The correlation coefficient between selected variable for main axis and the other variables are compute using Eq. (2),

$$r_{jj'} = \frac{\sum_{i=1}^{n}(x_{ij} - \bar{x}_j)(x_{ij'} - \bar{x}_{j'})}{\sum_{i=1}^{n}(x_{ij} - \bar{x}_j)^2 \sum_{i=1}^{n}(x_{ij'} - \bar{x}_{j'})^2} \tag{2}$$

Second axis is determined by the minimum absolute value of the correlations among the main axis variable and the other variables. The first axis belongs to variable has the greatest spread in the data and the second axis as possible should be perpendicular to the first axis. Here, the purpose of election these two axes, spread of the data should be provided the best explain in this bi-dimensional feature space. Selection of two axes is not a problem for the large data in terms of number of features and number of patterns. However, more than two explanatory variables may need to be selected when the number of clusters are increased.

After determining two axes for the proposed algorithm, the mean of data points is calculated as the center of the dataset according to selected axis

$$m = [\bar{x}_I \quad \bar{x}_{II}] \tag{3}$$

where $\bar{x}_I$ is mean according to variable of selected main axis, $\bar{x}_{II}$ is defined similarly. Euclidean distances

$$d_{im} = \left((x_{iI} - \bar{x}_I)^2 + (x_{iII} - \bar{x}_{II})^2\right)^{\frac{1}{2}}, \quad i = 1, 2, \ldots, n \tag{4}$$

are computed between each data point and the center. Then the data point with the highest distance in c_1 will be selected as the first candidate of the initial cluster center. Fig. 1 illustrates m as the mean of data points and c_1 which has the farthest distance to m is the candidate of the first initial cluster center.

Next, we calculate the Euclidean distance

$$d_{ic_1} = \left((x_{iI} - x_{c_1 I})^2 + (x_{iII} - x_{c_1 II})^2\right)^{\frac{1}{2}}, \quad i = 1, 2, \ldots, n \tag{5}$$

between each data points and c_1 by Eq. (5). To select a candidate for the second initial cluster center, the same mechanism is applied using d_{ic_1} instead of d_{im}. The data point with the highest distance of d_{ic_1} will be selected as the second initial cluster center candidate c_2, as shown in Fig. 2.

To select a next c_r for the candidate of the rest initial cluster centers, d_{ic_r} (where r is the current iteration step) is calculated between each data points and c_{r-1}. The Sd_{ir} is then added to the sum of distances as $(r-1)$ in rth iteration. For example, Sd_{i3} is calculated with Eq. (6) in third iteration.

$$Sd_{i3} = d_{ic_1} + d_{ic_2}, \quad i = 1, 2, \ldots, n \tag{6}$$

This accumulation scheme can avoid the nearest data points to c_{r-1} being chosen as the candidate of the next initial cluster center. It consequently can spread out the next initial cluster centers far away from the previous ones. The data point with the highest distance of Sd_{i3} will be selected as the second initial cluster center candidate c_3, as shown in Fig. 3.

The process is repeated until the number of initial cluster centers equals to the predefined number of clusters. Then, cluster membership of each points are determined according to candidate initial cluster centers and selected two axis. For p variables, the initial cluster centers are created using the determined cluster memberships.

3. Comparison criteria

To compare the clustering results, we will use the criteria which are the error percentage, the Rand index and Wilks' lambda test

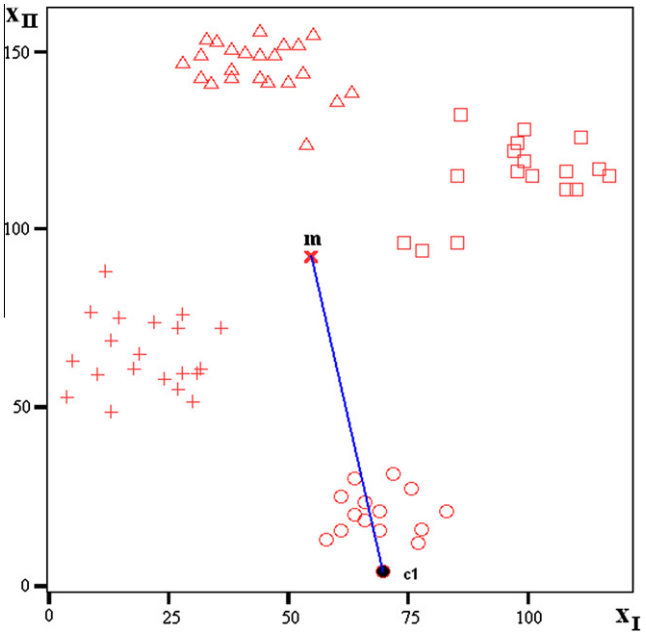

Fig. 1. Selection for first candidates of the initial cluster center.

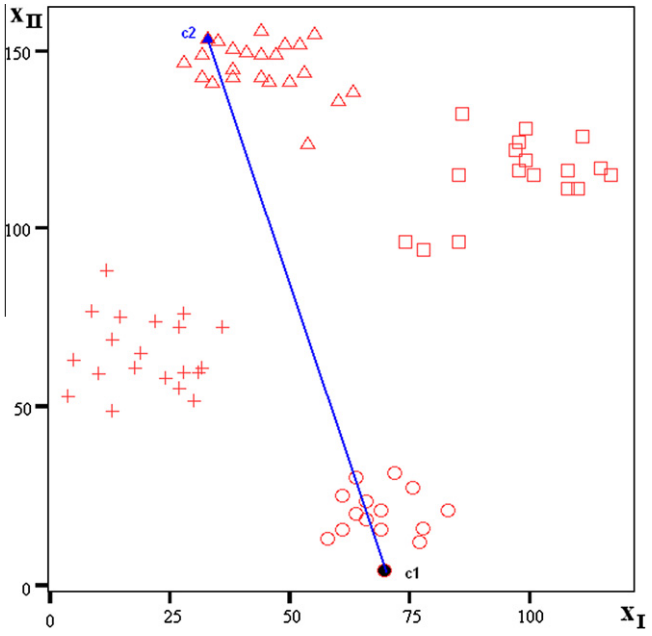

Fig. 2. Selection for second candidates of the initial cluster center.

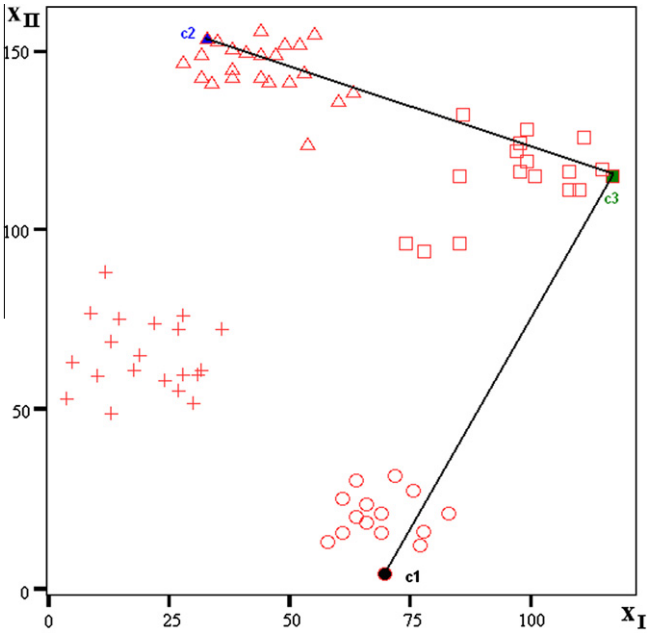

Fig. 3. Selection for third candidates of the initial cluster center.

statistic. The error percentage is calculated from number of misclassified observations and the total number of observations in the datasets. The error percentage is defined as follows

$$Error = \frac{\varepsilon}{n} \times 100 \tag{7}$$

where ε is the number of misclassified observations and n is the total number of observations.

The Rand index (Rand, 1971) has been traditionally used to measure the degree of agreement between two data partitions. Given a set of n objects $X = \{x_1 \cdots x_n\}$, suppose $P = p_1 \cdots p_k\}$ and $Q = q_1 \cdots q_k\}$ represent partitions of the objects with k-means algorithm and real cluster memberships respectively. For each object pair $\{x_i, x_j\}$ there are four possible outcomes:

a: x_i and x_j are in the same partition in P and in the same partition in Q.
b: x_i and x_j are in different partition in P but in the same partition in Q.
c: x_i and x_j are in the same partition in P but in different partition in Q.
d: x_i and x_j are in different partition in P and in different partition in Q.

Rand index is given by,

$$Rand = \frac{a+d}{a+b+c+d} \tag{8}$$

Rand index ranges from 0 to 1, where 0 means that the two partitions are entirely different, and 1 means that the two partitions are identical.

The Wilks' lambda test statistic is given by

$$\lambda = \frac{|W|}{|W+B|} \tag{9}$$

where W is within the sum of squares and products matrix and $W + B$ is the total sum of squares and products matrix. Differences between clusters are significant for small values of Wilks' lambda test statistics λ.

4. Experiments and results

To establish practical applicability of the proposed algorithm, we implemented it and tested its performance on a number of other real world datasets, the iris data, the wine recognition data, the letter image recognition data, the Ruspini data and the Spambase data. We evaluated the proposed algorithm on five datasets from UCI (http://archive.ics.uci.edu/ml/datasets.html) Machine Learning Repository.

The Iris dataset (Fisher, 1936) has often been used as the standard for testing clustering algorithms. This dataset has three classes that represents three different varieties of Iris flowers namely Iris setosa (I), Iris versicolor (II) and Iris virginica (III). Fifty samples were obtained from each of the three classes, thus a total of 150 samples is available. Every sample is described by a set of four attributes viz sepal length, sepal width, petal length and petal width.

The Wine 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 were overall 178 instances. There are 59, 71 and 48 instances in class I, class II and class III respectively. The classes are separable.

For the letter image recognition data, the objective is to identify each of the large number of black-and-white rectangular pixel displays as one of the 26 capital letters in the English alphabet. The character images 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 which were then scaled to fit into a range of integer values from 0 through 15. For experimental purpose we

Table 1
Descriptive statistics for the Iris dataset.

Statistics	Sepal length	Sepal width	Petal length	Petal width
$\bar{x}_j$	5.8433	3.0573	3.7580	1.1993
$s(x_j)$	0.8281	0.4359	1.7653	0.7622
cv_j	0.1417	0.1426	0.4697	**0.6356**

Table 2
The correlations among petal width and other variables.

	Petal width
Sepal length	0.818
Sepal width	**−0.366**
Petal length	0.963

have taken 789 patterns of letter A and 805 patterns of letter D from the dataset.

The Ruspini dataset (Ruspine, 1970) is popular to illustrate clustering techniques. It consists of 75 observations on two variables making up four natural groups including 23, 20, 17 and 15 entities in classes I, II, III and IV respectively.

The Spambase dataset (Arthur and Vassilvitskii, 2006) consists of 4601 points in 58 variables and it represent features available to an e-mail spam detection system. There are 1813 and 2788 points in spam and non-spam classes respectively.

The mean, standard deviation and variation coefficient, computed for the Iris dataset are given in Table 1.

The main axis is selected as the petal width which has maximum value of the coefficient of variation according to Table 1 by proposed algorithm. Second axis is determined by the minimum absolute value of the correlations among the petal width and the other variables. The correlations among the petal width and the other variables are given in Table 2.

The second axis is selected as the sepal width which has minimum absolute value of the correlation according to Table 2 by proposed algorithm. The data center was determined as $m = [1.1993,\ 3.0573]$ according to selected two axis. The scatter plot for Iris dataset according to petal width and sepal width is given in Fig. 4(a). Candidate initial cluster centers are determined according to petal width and sepal width as shown in Fig. 4(b)–(d) respectively. Cluster memberships of each observation in two axes are created by using distances between each observation and candidate initial cluster centers. The initial cluster centers are created in $n \times p$ dimensional dataset according to the determined cluster memberships. Initial cluster centers of the Iris data for k-means algorithm are obtained as $m_1 = [3.6516\ 0.2677\ 5.1774\ 1.4903]$, $m_2 = [2.9506\ 1.7916\ 6.4024\ 5.1193]$, $m_3 = [2.7917\ 0.6361\ 5.1278\ 2.5722]$ with the proposed algorithm.

We compared clustering results achieved by the k-means algorithm using random initial centers and initial centers derived by the proposed algorithm. The clustering results of k-means using random initial centers are the mean results over 10 runs since each run gives different results. The comparison of initial cluster centers computed using proposed algorithm and random initial cluster centers, for the data sets, is shown in Table 3. We also compare the results in terms of the classification error (%), given in Fig. 5, from proposed algorithm with the results from CCIA algorithm and the results from Deelers and Auwatanamongkol (2007).

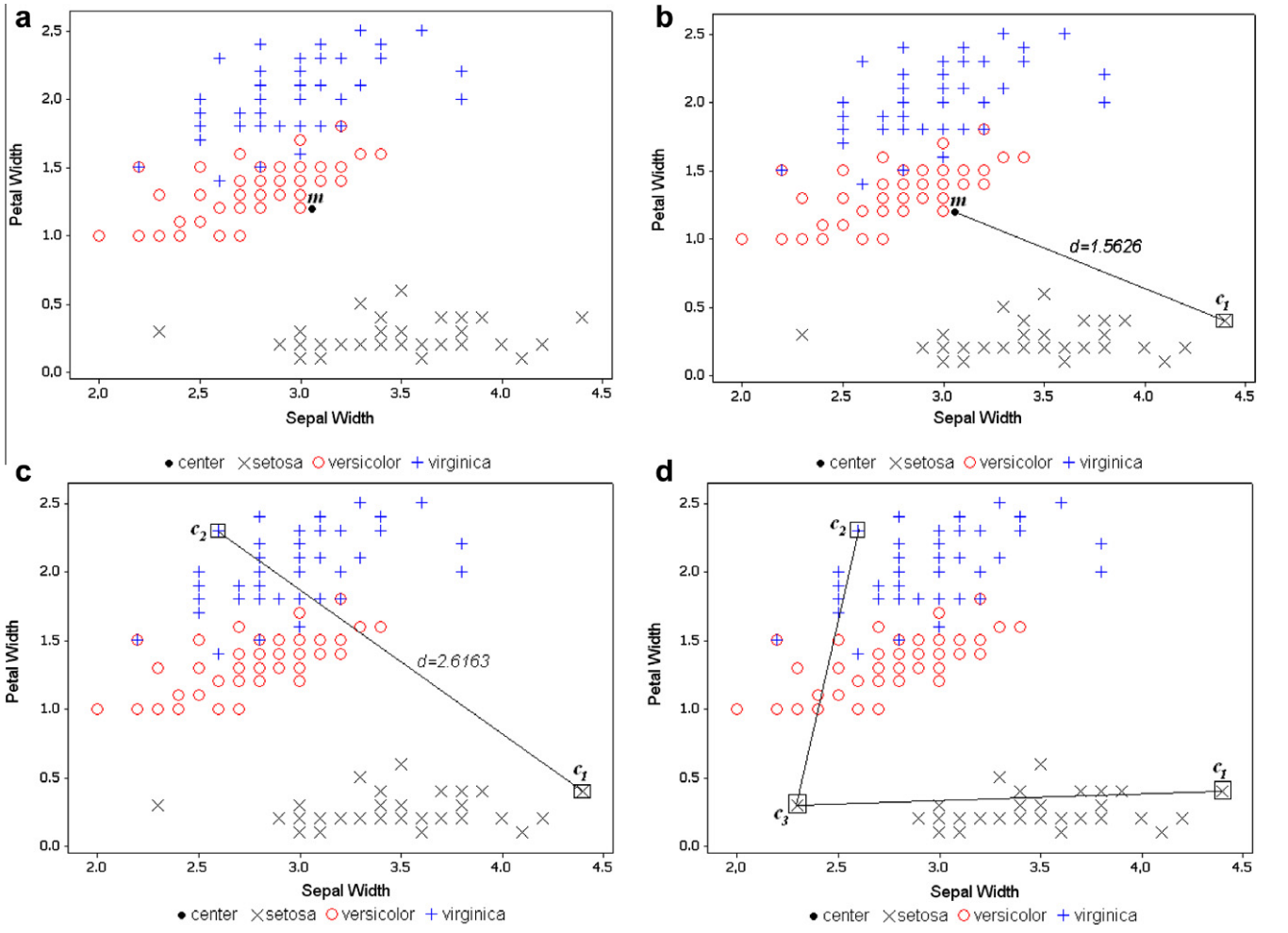

Fig. 4. Selection for candidates of the initial cluster center with proposed algorithm in Iris dataset. (a) The scatter plot for Iris data set according to petal width and sepal width. (b) Selection for first candidates of the initial cluster center in Iris dataset. (c) Selection for second candidates of the initial cluster center in Iris dataset. (d) Selection for third candidates of the initial cluster center in Iris dataset.

Table 3
Comparison results between proposed algorithm and random initial centers according to error percentage, Rand index and Wilks' lambda test statistic.

Dataset	Method	Error percentage	Rand index	Wilks' lambda
Iris	Proposed algorithm	**10.7000**	**0.8797**	**0.0322**
	Random	13.8300	0.8639	0.0376
Wine	Proposed algorithm	**3.4000**	**0.9543**	**0.0196**
	Random	10.5800	0.9018	0.0329
Letter	Proposed algorithm	**7.9046**	**0.8543**	**0.0877**
	Random	9.7380	0.6364	0.1071
Ruspini	Proposed algorithm	**0**	**1.0000**	**0.0034**
	Random	21.8667	0.8887	0.0160
Spambase	Proposed algorithm	**36.4051**	**0.5369**	**0.4171**
	Random	39.3393	0.5226	0.5912

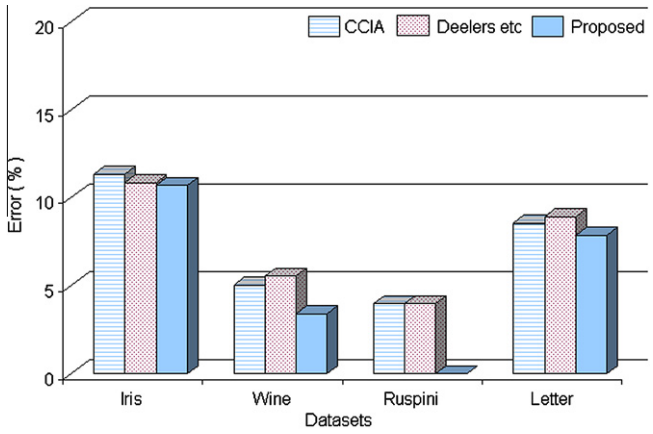

Fig. 5. Classification error comparisons among the three methods, CCIA, Deelers, etc. and proposed method.

Clustering results with the *k*-means algorithm using the initial centers computed by proposed algorithm suggest that we get improved and consistent clusters for all dataset in comparison to random initialization. We are getting better clustering results with *k*-means algorithm using proposed algorithm.This results are better than CCIA algorithm and proposed algorithm by Deelers and Auwatanamongkol (2007) according to error percentages.

In the CCIA, determining the initial cluster centers is quite complex because of the observations are separately classified according to each of the variable. For example, when $k = 3$ clusters and $p = 6$ variables, $3^6 = 729$ cluster may occur initially with the CCIA. Then these 729 clusters are reduced to $k = 3$ clusters by Merge-DBMSDC (Density-Based Multi Scale Data Condensation).

5. Conclusion

We have presented an algorithm for computing initial cluster centers in *k*-means algorithm. In this algorithm, two principal variables are selected according to maximum coefficient of the variation and minimum absolute value of the correlation. The reduced dataset is partitioned one at a time until the number of cluster equals to the predefined number of clusters. Then, cluster membership of each points are determined according to candidate initial cluster centers and selected two axis. For p variables, the initial cluster centers are created using the determined cluster memberships.

Also, after determined initial cluster centers and cluster membership of each of the data points according to selected two axes, the proposed algorithm can be applied by normalizing the dataset. Application of the proposed algorithm for the Wine dataset in this way increases the true classification rate.

The proposed algorithm is very effective, converges to better clustering results and almost all clusters have some data in it. Experimental results show improved and consistent cluster structures as compared to the random initial cluster centers. Also, the proposed algorithm is much simpler and easier to implement according to the previously proposed algorithm in the literature.

Acknowledgements

The authors thank editors and anonymous referee for his or her careful reading and valuable comments on improving the original manuscript.

References

Arthur, D., Vassilvitskii, S., 2006. *k*-Means++: The Advantages of Careful Seeding. Technical Report, Stanford.
Bradley, P.S., Fayyad, U.M., 1998. Refinning initial points for *k*-means algorithm. In: Proceeding of the 15th Internat. Conf. on Machine Learning (ICML'98).
Deelers, S., Auwatanamongkol, S., 2007. Enhancing *k*-means algorithm with initial cluster centers derived from data partitioning along the data axis with the highest variance. Internat. J. Comput. Sci. 2, 247–252.
Duda, R.O., Hart, P.E., 1973. Pattern Classification and Scene Analysis. John Wiley and Sons, NY.
Fisher, R.A., 1936. The use of multiple measurements in taxonomic problems. Ann. Eugenic. 7 (part 2), 179–188.
Jain, A.K., Dubes, R.C., 1988. Algorithms for Clustering Data. Prentice Hall, Englewood Cliffs, NJ.
Khan, S.S., Ahmad, A., 2004. Cluster center initialization algorithm for *k*-means algorithm. Pattern Recognition Lett. 25, 1293–1302.
Likas, A., Vlassis, N., Jakob, J.V., 2003. The global *k*-means algorithm algorithm. Pattern Recognition 36, 451–461.
Mac Queen, J., 1967. Some methods for classification and analysis of multivariate observations (pp. 281297). In: Le Cam, L.M., Neyman, J. (Eds.), Proc. 5th Berkley Symp. on Mathematical Statistics and Probability, vol. 1. University of California Press, p. 666, xvii.
Rand, W.M., 1971. Objective criteria for the evaluation of clustering methods. J. Amer. Statist. Assoc. 66, 846–850.
Ruspine, E.H., 1970. Numerical methods for fuzzy clustering. Inform. Sci. 2, 319–350.