J. Basic. Appl. Sci. Res., 2(11)11795-11802, 2012

© 2012, TextRoad Publication

ISSN 2090-4304
Journal of Basic and Applied Scientific Research
www.textroad.com

Application of Teaching-Learning-Based Optimization Algorithm on Cluster Analysis

Babak Amiri

Islamic Azad University, Fars Science and Research Branch

ABSTRACT

Cluster analysis has received attention in many scientific fields. The purpose of clustering analysis is to detect group data points, which are close to one another. One of the most widely used techniques for clustering is the K-means algorithm. The performance of K-means algorithm which converges to numerous local minima depends highly on initial cluster centers. In order to overcome local optima problem lots of studies done in clustering. A population-based method called Teaching-Learning-Based-Optimization (TLBO) is proposed to solve the clustering problem. TLBO is a robust and effective search algorithm. The most salient advantage of this algorithm is that it does not require the tuning of any kind of controlling parameters. The efficiency of the proposed algorithm is studied by testing on several data sets. Numerical results show that the proposed evolutionary optimization algorithm is robust and suitable for data clustering.

KEYWORDS: Teaching-Learning-Based Optimization, Data clustering, K-means clustering, Evolutionary Algorithm.

INTRODUCTION

One of the most important techniques of unsupervised classification is clustering. In clustering objects with the same attributes will be grouped in a same cluster. Clustering techniques can be classified in to major classes: hierarchical and partitional. The hierarchical clustering can be divided into agglomerative and divisive. In hierarchical clustering n objects will be grouped into k clusters by minimizing some measure of dissimilarity in each group and maximizing the dissimilarity of different groups [1, 2, 3 and 4]
In this paper our focus is on partitional clustering, and in particular the K-means algorithm that is one of the most efficient clustering algorithms. However, the K-means algorithm suffers from several drawbacks [5]. The objective function of the K-means algorithm may contain several local optima because it is not convex. Therefore the outcome of K-means algorithm heavily depends on the initial solution [6]. To overcome these shortcomings recently many algorithms have been developed based on evolutionary algorithms like GA, TS, PSO and SA [7, 8, 9, 10, 11, 12 and 13]. But problem is that most of these evolutionary algorithms are very slow to find optimal solution.

In this paper Teaching-Learning-Based Optimization (TLBO), is applied to find global solutions for clustering problem with less computational effort and high reliability [17]. The TLBO algorithm is a new robust and effective evolutionary optimization algorithm. The principle idea behind TLBO is the simulation of teaching process in the traditional classroom. The performance of TLBO ends in two basic stages: (1) "teacher phase" or learning from the teacher, and (2) "learner phase" or trade off information between learners. The teacher is the one who promotes students' knowledge to his or her current level. The implementation of TLBO does not require the assertion of any kind of controlling parameters. This causes the algorithm to turn into a strong one.

In the following, the cluster analysis problem is discussed in section 2. Sections 3 introduce the TLBO algorithm. In section 4, the application of the proposed algorithm in clustering is presented. In section 5, the feasibility of the proposed algorithm is demonstrated and compared with the ACO, GA, SA, PSO and K-means for different data sets.

1. Cluster Analysis Problem

Clustering analysis that is an NP-complete problem to find groups in heterogeneous data by minimizing dissimilarity measures is one of the fundamental tools in data mining, machine learning and pattern classification solutions [9]. Clustering in N-dimensional Euclidean space R^N is the process of

***Corresponding Author:** Babak Amiri, Islamic Azad University, Fars Science and Research Branch.

partitioning a given set of n points into K groups (or, clusters) based on some similarity (distance) metric that is Euclidean distance, which derived from the Minkowski metric (equations 1 and 2).

$$d(x,y)=\left(\sum_{i=1}^{m}\left|x_i-y_j\right|^r\right)^{1/r} \tag{1}$$

$$d(x,y)=\sqrt{\sum_{i=1}^{m}\left(x_i-y_j\right)^2} \tag{2}$$

Let the set of n points $\{X_1, X_2,\dots, X_n\}$ be represented by the set S and the K clusters be represented by $C_1, C_2\dots,C_K$ Then:

$$C_i \neq \phi \qquad for\ i=1,\dots,K,$$

$$C_i \cap C_j = \phi \qquad for\ i=1,\dots,K,\ \ j=1,\dots,K,\ and\ i\neq j$$

$$and \qquad \bigcup_{i=1}^{K} C_i = S.$$

In this study the Euclidian metric has been used as a distance. All of clustering algorithms can be classified into two categories: hierarchical clustering and partitional clustering. Partitional clustering methods are the most popular class of center based clustering methods. The K-means algorithms, is one of the most widely used center based clustering algorithms. To find K centers, the problem is defined as an optimization performance function (minimization), *Perf(X, C)*, defined on both the data items and the center locations. A popular performance function for measuring goodness of the K clustering is the total within-cluster variance or the total mean-square quantization error (MSE), equation 3 [12].

$$Perf(X,C)=\sum_{i=1}^{N} Min\left\{\left\|X_i-C_l\right\|^2 \middle| l=1,\dots,K\right\} \tag{3}$$

The steps of the K-means algorithm are as follow [7]:

Step 1: Choose K cluster centers $C_1, C_2, \dots, C_k$ randomly from n points $\{X_1, X_2,\dots, X_n\}$.

Step 2: Assign point X_i, $i=1, 2, \dots, n$ to cluster C_j, $J\in\{1, 2, \dots, K\}$ if $\| X_i - C_j \| < \| X_i - C_p \|$, p=1, 2, …, K, and $j\neq p$.

Step 3: Compute new cluster centers $C_1^*, C_2^*, \dots, C_K^*$ as follows:

$$C_i^* = \frac{1}{n}\sum_{x_j\in C_i} X_j, \qquad i=1,2,\dots,K, \tag{4}$$

where n_i is the number of elements belonging to cluster C_i.

Step 4: If termination criteria satisfied, stop otherwise continues from step 2

Note that in case the process close not terminates at step 4 normally, then it executed for a mutation fixed number of iterations. Attempts to directly

2. TLBO Algorithm

The TLBO algorithm is a newly developed meta-heuristic optimization algorithm [17]. It is a population-based optimization algorithm that is modelled based on the transfer of knowledge to the classroom environment, where learners first gain knowledge from a teacher (Teacher Phase) and then from fellow-students (Learner Phase). The structure of the proposed algorithm can be explicated as follows:

Teacher phase: In this phase the solution nominations are randomly distributed throughout the search space. Thus, the best solution will be selected amongst all and will interact the knowledge with other candidates. Elaborately, since a teacher, who is the most skilled person about the objective in the population, influences the student's deed to take part some pre-planned aim. It is desired that the teacher augments the mean of his or her class information level depending on his or her experience. The teacher, thus, will put maximum effort into training his or her learners, but learners will acquire information according to the worthiness of training delivered by a teacher and the worthiness of learners in the class. The teacher phase is structured as follows:

$$ME_{jw}^{inter} = \frac{(x_{1,jw}^{iter} + x_{2,jw}^{iter} + \cdots + x_{learner,jw}^{iter})}{N_{learner}} \quad (5)$$

$$DM^{iter} = rand() \times (TR^{iter} - TF^{iter} - ME^{iter}) \quad (6)$$

TF is a teaching factor that is randomly determined as either 1 or 2.

$$TF^{iter} = round(1 + rand(\quad)) \quad (7)$$

$$x_{n,new1}^{iter} = X_n^{iter} + DM^{iter} \quad (8)$$

The fitness function $\hat{F}_i(x_{n,new1}^{iter})$ is compared with the fitness function of the target vector $\hat{F}_i(x_{n,learner}^{iter})$ as Eq. (9) in order to calculate each element $x_{n,jw,new1}^{iter}$of the n^{th} learner $x_{n,new1}^{iter}$.

$$x_{r1}^{iter} = \begin{cases} x_{n,new1}^{iter} & if\ \hat{F}_i(x_{n,new1}^{iter}) \leq \hat{F}_i(x_{n,learner}^{iter}) \\ x_{n,learner}^{iter} & otherwise \end{cases} \quad (9)$$

Learning Phase: As stated above, by means of interaction between each student according to teaching–learning process students can also increase their knowledge. So, a solution is randomly interacted to learn something new with other solutions in the population. If the other solutions have more knowledge than him or her, a solution will learn new piece of information.21 The latter way is described as follows.

$$X_{n,new2}^{iter} = \left[x_{n,1,new2}^{iter}, x_{n,2,new2}^{iter}, \cdots, x_{n,NU,new2}^{iter}\right] \quad (10)$$

$$X_{n,new2}^{iter} = X_{n,new2}^{iter} + rand \times (X_{u1}^{iter} - X_{u2}^{iter}) \quad (11)$$

The replacement process can be implemented Similar to the teacher phase as below:

$$X_{r2}^{iter} = \begin{cases} x_{n,new2}^{iter} & if\ \hat{F}_i(x_{n,new2}^{iter}) \leq \hat{F}_i(x_{n,learner}^{iter}) \\ x_{n,learner}^{iter} & otherwise \end{cases} \quad (12)$$

3. Application of TLBO algorithm on Clustering

This section introduces application of the TLBO algorithm on cluster analysis. To apply the TLBO algorithm on clustering the following steps should be repeated:

Step 1: Initializing the problem and algorithm parameters

Step 2: Establishing the initial population learners.

Step 3: Compute the objective function.

Step 4: Compute the mean of the population.

Step 5: Determine the best solution (Teacher).

Step 6: Modify solutions based on the teacher knowledge according to teacher phase.

Step 7: Update solutions according to learner phase and Steps 3.

Step 8: Go to Step 4 until the iteration number arrives at the maximum iteration number.

4. Experimental results

The efficiency of the TLBO algorithm on clustering has been tested on several well known datasets such as: four artificial data sets and six real-life data sets and compared with the ACO, PSO, SA and K-means algorithms [14, 15 and 16]. In stochastic algorithms the effectiveness highly depends on the initial solutions. To overcome these drawback each algorithms performed 100 times individually with randomly generated initial solutions. The simulations are performed on a Core i7 2.7 GHz computer with 4 GB RAM memory. The software is developed using MATLAB 7.13.

Artificial data sets:

Dataset 1: This dataset has 10 data points with two non-overlapping clusters and two dimensions.

Dataset 2: This dataset has 75 data points with three non-overlapping clusters and two dimensions.

Dataset 3: This dataset has 95 data points with nine overlapping clusters with equal probability (1/19). The dataset has two dimensions and generated using a triangular distribution. The *X*-*Y* ranges for the nine classes are as follows:

Class 1: [-3.3,-0.7] × [0.7, 3.3],
Class 2: [-1.3, 1.3] × [0.7, 3.3],
Class 3: [0.7, 3.3] × [0.7, 3.3],
Class 4: [-3.3,-0.7] × [-1.3, 1.3],
Class 5: [-1.3, 1.3] × [-1.3, 1.3],
Class 6: [0.7, 3.3] × [-1.3, 1.3],
Class 7: [-3.3,-0.7] × [-3.3,-0.7],
Class 8: [-1.3, 1.3] × [-3.3,-0.7],
Class 9: [0.7, 3.3] × [-3.3,-0.7].

Data 4: This dataset has 1000 data points with two overlapping clusters. The dataset has ten dimensions and generated using a triangular distribution.

The range for class 1 is $[0, 2]^{10}$, and for class 2 is $[1, 3] \times [0, 2]^9$, with peaks at (1, 1) and (2, 1). We may quantify the distribution on the first axis (*X*) for class 1 as follow:

$$f_1(x) = \begin{cases} 0 & for\ x \le 0, \\ x & for\ 0 < x \le 1, \\ 2 - x & for\ 1 < x \le 2, \\ 0 & for\ x > 2. \end{cases}$$

for class 1 similarly for class 2 is as follow:

$$f_1(x) = \begin{cases} 0 & for\ x \le 1, \\ x - 1 & for\ 1 < x \le 2, \\ 3 - x & for\ 2 < x \le 3, \\ 0 & for\ x > 3. \end{cases}$$

For both class 1 & 2 the distribution on the other axes (Y_i, i=1, 2, …, 9) is as follow:

$$f_1(x) = \begin{cases} 0 & for\ y_i \le 0, \\ y_i & for\ 0 < y_i \le 1, \\ 2 - y_i & for\ 1 < y_i \le 2, \\ 0 & for\ y_i > 2. \end{cases}$$

Real-life data sets are as follows:

Iris data (N=150, d=4, K=3): This dataset has 150 points which are random samples of three species of the iris flower such as: setosa, versicolor, and virginica. For each clusters we have 50 samples with 4 dimensions [21].

Wine data (N=178, d=13, K=3)*:* The wine dataset is result of chemical analysis of wines in the same areas in Italy. The dataset has 178 points with 13 continues attributes

Contraceptive Method Choice (N = 1473, d = 10, K = 3): This dataset is output of a survey in Indonesia in 1987. Points in this dataset are married women who were not pregnant or do not know that they are. The dataset has 178 points with 13 dimensions and 3 clusters.

Vowel data set (N = 871, d = 3, K= 6). This dataset has 871 three dimensional points and six overlapping clusters[22].

Wisconsin breast cancer (*N=683, d=9, K=2*). This dataset has 683 points with nine features such as: clump thickness, cell size uniformity, cell shape uniformity, marginal adhesion, single epithelial cell size, bare nuclei, bland chromatin, normal nucleoli, and mitoses. There are two clusters in this dataset: malignant (444 objects) and benign (239 objects).

Ripley's glass (*N=214, d=9, K=6*). This dataset has 214 points with nine features. The dataset has six different clusters which are: building windows float processed (70 objects), building windows non-float processed (76 objects), vehicle windows float processed (17 objects), containers (13 objects), tableware (9 objects), and headlamps (29 objects).

The efficiency of the proposed algorithm has been compared with other algorithms by applying them on above datasets. The best solution of 100 run of each algorithm, number of function evaluation and standard deviation of solutions obtained by applying algorithms on the datasets has been used for comparison. The quality of solution is considered based on the average and worst values of the clustering metric (F_{avg} and F_{worst}). F is the performance of clustering algorithms that has been shown in equation 3. Tables 1 to 10 present a comparison among the results of algorithms.

Table 1. Result obtained by the five algorithms for 100 different runs on dataset 1

Method	**Function Value**			**Standard deviation**	**Number of function evaluations**
	F_{best}	$F_{average}$	F_{worst}		
TLBO	3.114821	3.128761	3.168349	0.013940	823
SA	3.217832	3.382089	3.539115	0.164257	1751
PSO	3.189459	3.351617	3.438486	0.162158	1642
ACO	3.142375	3.163422	3.352843	0.021047	1639
K-means	3.206981	3.350941	3.486195	0.143960	43

Table 2. Result obtained by the five algorithms for 100 different runs on dataset 2

Method	**Function Value**			**Standard deviation**	**Number of function evaluations**
	F_{best}	$F_{average}$	F_{worst}		
TLBO	51.827619	51.915611	52.008617	0.087992	1,948
SA	53.562492	53.635943	53.929748	0.073451	3,846
PSO	52.034921	52.258617	52.516815	0.223696	3,246
ACO	52.082746	52.212071	52.729373	0.129325	3,183
K-means	53.872432	56.624922	59.496185	2.752490	97

Table 3. Result obtained by the five algorithms for 100 different runs on dataset 3

Method	**Function Value**			**Standard deviation**	**Number of function evaluations**
	F_{best}	$F_{average}$	F_{worst}		
TLBO	963.836486	963.912484	964.016483	0.075998	3,351
SA	966.418263	966.614089	967.397392	0.195826	8,984
PSO	964.264831	965.016542	966.168253	0.751711	8,879
ACO	964.739472	965.048327	966.283745	0.308855	7,836
K-means	967.584392	968.245317	969.186242	0.660925	176

Table 4. Result obtained by the five algorithms for 100 different runs on dataset 4

Method	**Function Value**			**Standar d deviation**	**Number of function evaluations**
	F_{best}	$F_{average}$	F_{worst}		
TLBO	1247.958617	1248.012407	1248.426183	0.053790	3,631
SA	1251.736287	1253.115831	1254.895375	1.379544	9,794
PSO	1248.876294	1249.027819	1249.219345	0.151525	9,682
ACO	1248.958685	1249.034036	1249.335442	0.075351	8,846
K-means	1251.563183	1253.151574	1255.068316	1.588391	203

Table 5 Result obtained by the five algorithms for 100 different runs on Iris data

Method	**Function Value**			**Standard deviation**	**Number of function evaluations**
	F_{best}	$F_{average}$	F_{worst}		
TLBO	96.6500	96.6500	96.6500	0	2,468
SA	97.4573	99.957	102.01	2.018	5314
PSO	96.8942	97.2328	97.8973	0.347168	4,953
ACO	96.753	97.453	98.023	0.567	4,931
K-means	97.333	106.05	120.45	14.6311	120

Table 6 Result obtained by the five algorithms for 100 different runs on Wine data

Method	Function Value			Standard deviation	Number of function evaluations
	F_{best}	$F_{average}$	F_{worst}		
TLBO	16,295.31	16,323.17	16,345.26	26.824	6,316
SA	16,473.4825	17,521.094	18,083.251	753.084	17,264
PSO	16,345.9670	16,417.4725	16,562.3180	85.4974	16,532
ACO	16,346.7820	16,417.127	16,502.943	80.3731	15,473
K-means	16,555.68	18,061	18,563.12	793.213	390

Table 7 Result obtained by the five algorithms for 100 different runs on CMC data

Method	Function Value			Standard deviation	Number of function evaluations
	F_{best}	$F_{average}$	F_{worst}		
TLBO	5,694.2816	5,699.9281	5,700.1053	5.9382	6,865
SA	5,849.038	5,893.4823	5,966.947	501.8672	26,829
PSO	5,700.98530	5,820.96475	5,923.24900	46.95969737	21,456
ACO	5,701.923	5,819.1347	5,912.4300	45.6347	20,436
K-means	5,842.20	5,893.60	5,934.43	473.16	270

Table 8 Result obtained by the five algorithms for 100different runs on Vowel data

Method	Function Value			Standard deviation	Number of function evaluations
	F_{best}	$F_{average}$	F_{worst}		
TLBO	148,976.0005	149,007.0010	149,200.0100	25.9637	3,484
SA	149,468.268	162,108.5381	165,996.428	2846.23516	9,528
PSO	149,370.4700	161,566.2810	165,986.4200	2847.08594	9,423
ACO	149,201.632	161,431.0431	165,804.671	2746.0416	8,436
K-means	149,422.26	15,9242.89	161,236.81	916	180

Table 9 Result obtained by the five algorithms for 100different runs on Wisconsin breast cancer

Method	Function Value			Standard deviation	Number of function evaluations
	F_{best}	$F_{average}$	F_{worst}		
TLBO	2,964.25	2,966.72	2,970.02	2.263	3,495
SA	2,992.53	3,086.06	3,451.38	111.0102	17,853
PSO	2,973.50	3,050.04	3,318.88	110.8013	16,290
ACO	2,970.49	3,046.06	3,242.01	90.50028	15,983
K-means	2,987.19	2,987.78	2,988.24	0.38	180

Table 10 Result obtained by the five algorithms for 100different runs on Ripley's glass

Method	Function Value			Standard deviation	Number of function evaluations
	F_{best}	$F_{average}$	F_{worst}		
TLBO	199.53	199.66	201.31	0.125	6,501
SA	273.27	276.21	285.93	4.976188	199,468
PSO	270.57	275.71	283.52	4.557134	198,765
ACO	269.72	273.46	280.08	3.584829	196,581
K-means	215.74	12.47107	255.38	235.50	630

The simulation results given in Tables 1 to 10 show that TLBO is very precise. In the other word, it provides the optimum value and small standard deviation in compare to those of other methods. The results obtained on the artificial datasets (tables 1-4) show that the proposed algorithm reaches to the best optimum values of in compare to other methods. Table 5 shows results on iris dataset shows that TLBO converges to the global optimum of 96.6500 in all of runs while the best solutions of SA, PSO, ACO and K-means are 97.4573, 96.8942, 96.853 and 97.333, respectively. The standard deviation of the fitness function for this algorithm is 0, which it significantly is smaller than other methods. Table 6 shows the result of algorithms on the wine dataset. The optimum value is 16,295.31, which is obtained in 90% runs of TLBO algorithm. Noticeably other algorithms fail to attain this value even once within 100 runs. Table 7 provides the results of algorithms on the CMC dataset. As seen from the results of the TLBO algorithm are far superior that of others. For the vowel data set, the best global solution, the worst global solution, the average and the standard deviation of the TLBO are 148,976.0005, 149,200.0100, 149,007.0010 and 25.9637 respectively. For the PSO algorithm they are 149,370.4700, 165,986.4200, 161,566.2810 and 2847.08594, respectively. The result of the TLBO algorithm is much better than those of other algorithms. On Wisconsin breast cancer dataset results given in Table 9, show that the TLBO provide the optimum value of 2,964.25 while the SA, PSO, ACO and K-means algorithms obtain 2,992.53, 2,973.50, 2,970.49 and 2,987.19, respectively. The TLBO was able to find the optimum in 95% of runs. Finally, Table 10 shows the best, average, worst and standard deviation values obtained by algorithms for Ripley's glass dataset. It is found that the TLBO clustering algorithm is able to provide the same partition of the data points in most of runs. As earlier, the results of the other algorithms are in inferior to that of ours.

In terms of the number of function evaluations, K-means needs the least number of function evaluations, but the results are less than satisfactory. For artificial datasets the number of function evaluation of TLBO algorithm is less than others except the K-means algorithm. For the iris dataset, the number of function evaluations of TLBO, SA, PSO, ACO and K-means are 2468, 2523, 4953, 4931 and 120, respectively. The number of function evaluations of TLBO for artificial datasets, Wine, CMC, Vowel, Wisconsin breast cancer and Ripley's glass are 6316, 6865, 3484, 3495 and 6501, respectively. These results show that the number of function evaluations of TLBO is less than those of other evolutionary algorithms. Based on the obtained simulation results, we can conclude that the changes of the number of fitness function evaluations of the proposed algorithm are less than other evolutionary algorithms for all cases.

The simulation results in the tables demonstrate that the proposed evolutionary algorithm converges to global optimum with a smaller standard deviation and less function evaluations and leads naturally to the conclusion that the TLBO algorithm is a viable and robust technique for data clustering.

5. Conclusion

The clustering analysis is a very important technique and has attracted much attention of many researchers in different areas. The K-means algorithm one of the most efficient clustering method and is very simple that has been applied to many engineering problems. This paper has applied a newly developed TLBO algorithm for solving the clustering problem. The proposed algorithm has been implemented and tested on several artificial and well known real, the result illustrate that the proposed TLBO optimization algorithm can be considered as a viable and an efficient heuristic to find optimal or near optimal solutions for clustering problems of allocating N objects to k clusters. The experimental results indicate that the proposed optimization algorithm is at least comparable to the other algorithms in terms of function evaluations and standard deviations.

REFERENCES

[1] Yi-Tung Kao, Erwie Zahara, I-Wei Kao, A hybridized approach to data clustering, Expert Systems with Applications, 34, pp: 1754–1762, 2008.

[2] J.F. Lu, J.B. Tang, Z.M. Tang and J.Y. Yang, "Hierarchical initialization approach for K-Means clustering", Pattern Recognition Letters, January 2008.

[3] Chung-Chian Hsu, Chin-Long Chen and Yu-Wei Su , "Hierarchical clustering of mixed data based on distance hierarchy", Information Sciences, Volume 177, Issue 20, 15, pp. 4474-4492, October 2007.

[4] Georgios P. Papamichail and Dimitrios P. Papamichail , "The k-means range algorithm for personalized data clustering in e-commerce", European Journal of Operational Research, Volume 177, Issue 3, pp. 1400-1408, March 2007

[5] R.J. Kuo, H.S. Wang, Tung-Lai Hu and S.H. Chou , "Application of ant K-means on clustering analysis", Computers & Mathematics with Applications, Volume 50, Issues 10-12, pp. 1709-1724, November-December 2005.

[6] Stephen J. Redmond and Conor Heneghan , "A method for initialising the K-means clustering algorithm using kd-trees", Pattern Recognition Letters, Volume 28, Issue 8, pp. 965-973, June 2007.

[7] Ujjwal Mualik, Sanghamitra Bandyopadhyay, genetic algorithm-based clustering technique, pattern Recognition, 33 (2000) 1455-1465.

[8] K. Krishna, Murty, genetic k-means Algorithm, IEEE Transaction on Systems, man, and Cybernetics-Part B: Cybernetics 29 (1999), 433-439

[9] Shokri Z. Selim and K. Al-Sultan, A Simulated Annealing Algorithm for the Clustering problem, Pattern Recognition, Volume 24, Issue 10, 1991, Pages 1003-1008.

[10] C.S. Sung, H.W. Jin, A tabu-search-based heuristic for clustering, Pattern Recognition, Volume 33, 2000, Pages 849-858.

[11] R. I. kuo, H. S. Wang, Tung-Lai Hu, S. H. Chou, Application of Ant K-Means on Clustering Analysis, Computers and Mathematics with Applications 50 (2005) 1709-1724.

[12] P.S. Shelokar, V.K. Jayaraman, B.D. Kulkarni, An ant colony approach for clustering, Analytica Chimica Acta 509 (2004) 187–195.

[13] Zulal Gungor, Alper Unler, K-harmonic means data clustering with simulated annealing heuristic, Applied Mathematics and Computation, 2006.

[14] S.K. Pal, D.D. Majumder, Fuzzy sets and decision making approaches in vowel and speaker recognition, IEEE Trans. Systems, Man Cybernet. SMC-7 (1977) 625}629.

[15] C.L. Blake, C.J. Merz, UCI repository of machine learning databases. Available from: <http://www.ics.uci.edu/_mlearn/MLRepository.html>.

[16] R.A. Johnson, D.W. Wichern, Applied Multivariate Statistical Analysis, Prentice-Hall, Englewood Cli!s, NJ, 1982.

[17] R. V. Rao, V. J. Savsani, D. P. Vakharia: Teaching-learning-based optimization: A novel method for constrained mechanical design optimization problems. Computer-Aided Design 43(3): 303-315 (2011)

[18] Zulal Gungor, Alper Unler, K-harmonic means data clustering with simulated annealing heuristic, Applied Mathematics and Computation, 2006.

[19] R.A. Johnson, D.W. Wichern, Applied Multivariate Statistical Analysis, Prentice-Hall, Englewood Cli!s, NJ, 1982.

[20] R.O. Duda and P.E. Hart, Pattern Classification and Scene Analysis. New York: John Wiley & Sons, 1973.

[21] C.L. Blake, C.J. Merz, UCI repository of machine learning databases. Available from: <http://www.ics.uci.edu/_mlearn/MLRepository.html>.

[22] S.K. Pal, D.D. Majumder, Fuzzy sets and decision making approaches in vowel and speaker recognition, IEEE Trans. Systems, Man Cybernet. SMC-7 (1977) 625}629.