AUTOMATIC UNSUPERVISED DATA CLASSIFICATION USING JAYA EVOLUTIONARY ALGORITHM

Ramachandra Rao Kurada[1] and Dr. Karteeka Pavan Kanadam[2]

[1]Asst. Prof., Department of Computer Science & Engineering, Shri Vishnu Engineering College for Women, Bhimavaram
[2]Professor, Department of Information Technology, RVR & JC College of Engineering, Guntur

ABSTRACT

In this paper we attempt to solve an automatic clustering problem by optimizing multiple objectives such as automatic k-determination and a set of cluster validity indices concurrently. The proposed automatic clustering technique uses the most recent optimization algorithm Jaya as an underlying optimization stratagem. This evolutionary technique always aims to attain global best solution rather than a local best solution in larger datasets. The explorations and exploitations imposed on the proposed work results to detect the number of automatic clusters, appropriate partitioning present in data sets and mere optimal values towards CVIs frontiers. Twelve datasets of different intricacy are used to endorse the performance of aimed algorithm. The experiments lay bare that the conjectural advantages of multi objective clustering optimized with evolutionary approaches decipher into realistic and scalable performance paybacks.

KEYWORDS

Multi objective optimization, evolutionary clustering, automatic clustering, cluster validity indexes, Jaya evolutionary algorithm.

1. INTRODUCTION

For the past three decades, majority of optimization problems demands improvement issues with multiple objectives and are attracted towards evolutionary computation methodologies due their simplicity of transformative calculation. The leverage of these evolutionary approaches are flexible, to add, remove, modify any prerequisite regarding problem conceptualization, generation of comparative Pareto set and has ability to tackle higher complexities than the mainstream methods. These robust and powerful search procedures generally portray a set of candidate solutions, selection procedure for mating, segmenting and re-assembling of set of several solutions to produce new solutions. This is reflected by the speedily increasing of interest in the field of evolutionary clustering with multi objective optimizations [1].

Data clustering is recognized as the most prominent unsupervised technique in machine learning. This technique apportions a given dataset into homogeneous groups in view of some likeness/disparity metric. Conventional clustering algorithms regularly make previous assumptions about grouping a cluster structure, and adoptable with a suitable objective function so that it can be optimized with classical or metaheuristic techniques. These estimations grade inadequately when clustering presumptions are not hold in data [2].

The natural paradigm to fit the data distribution in the entire feature space, discovering exact number of partitions is violated in single objective clustering algorithm if distinctive locales of

DOI:10.5121/acii.2016.3204

the component space contain clusters of diverged space. Estimating a combined solution which is stable, confident and lower sensitivity to noise is unattainable by any single objective clustering algorithm. Multi-objective clustering can be perceived as a distinct case of multi-objective optimization, targeting to concurrently optimize several trade-off with numerous objectives under specific limitations. The aim objective of multi-objective clustering is to disintegrate a dataset into comparable groups, by exploiting the multiple objectives analogously [3-4].

In this paper, we provide an clustering algorithms underplayed with Jaya evolutionary algorithm [15] to solve large set of objectives, for affricating factual automatic k determination, that are interesting, suitable detachment prompted in data sets, and optimizing a set of cluster validity indices (CVIs) simultaneously for encouraging most favourable convergence at final solutions. For conquering high intra-cluster likeness and low inter-cluster likeness, this algorithm uses CVIs as objective functions as mentioned in [5]. The set of internal and external validity indices used as fitness functions in this paper are Rand, Adjusted Rand, Siloutte, Chou Be, Davies–Bouldin and Xie–Beni indexs [6].

The remainder of this paper is organized as follows. Section II presents a review of recent automatic clustering algorithms. In Section III, describes the scalability of the proposed AutoJAYA algorithm and original Jaya evolutionary algorithm. The effectiveness of our scheme is discussed in Section IV. Finally, Section V concludes this paper.

2. LITERATURE REVIEW

The survey published by Mukhopadhyay, Maulik and Bandyopadhyay, S. in 2015 argue the importance of using multiobjective clustering in the domains of image segmentation, bioinformatics, web mining with real time applications. The survey urges the importance of Multiobjective clustering for optimizing multiple objective functions simultaneously. The authors highlights the techniques for encoding, selection of objective functions, evolutionary operators, schemes for maintaining non-dominated solutions and assorting an end solution [7].

In order to improve searching skills, in 2015, Abadi, & Rezaei combined of continuous ant colony optimization and particle swarm optimization and proposed a strategy which is a combination of these two algorithms with genetic algorithm, the results demonstrated were of high capacity and resistance [8].

In 2015, Ozturk, Hancer and Karaboga used artificial bee colony algorithm in dynamic (automatic) clustering discrete artificial bee colony as a similarity measure between the binary vectors through Jaccard coefficient [9]. In 2014, Kumar and Chhabra, gravitational search algorithm in real life problems, where prior information about the number of clusters is not known in image segmentation domain to attain automatic segmentation of both gray scale and colour images [10].

In 2014, Kuo, Huang, Lin, Wu and Zulvia determined the appropriate number of clusters and assigns data points to correct clusters, with kernel function to increase clustering capability, in this study they have used with bee colony optimization for attaining stable and accurate results [11]. In 2014, Wikaisuksakul presented a multi-objective genetic algorithm for data clustering methods, to handle the overlapping clusters with multiple objectives, using the fuzzy c-means method. The real-coded values are encoded in a string to represent cluster centers and Pareto solutions corresponding to a trade-off between the two objectives are finally produced [12].

In 2014, Mukhopadhyay, Maulik,Bandyopadhyay, and CoelloCoello, published two survey's with Part I and Part II on multiobjective evolutionary algorithms for data mining with

Evolutionary Computation [13-14]. Part I survey holds literature for basic concepts related to multi-objective optimization in data mining and evolutionary approaches for feature selection and classification. In part II the authors present the rules for association, clustering and other data mining tasks related to different multi-objective evolutionary algorithms.

3. AUTOMATIC CLUSTERING ALGORITHM - AUTOJAYA

This paper attempts to constellate exact number of proper detachment in datasets automatically without any human intervention during the algorithm execution. The objective functions for assorting an end solution is postured as a multi-objective optimization problem, by optimizing a customary of cluster validity indices concurrently. The proposed multi-objective clustering technique uses a most recently developed evolutionary algorithm Jaya [15], based on multi-objective optimization method as the underlying optimization strategy. The points are assigned randomly to selected cluster centres based on Euclidean distance. The Rand, Adjusted Rand, Silhouette, Chou Be, Davies–Bouldin and Xie–Beni CVIs are optimized simultaneously to endorse the validity of aimed algorithm. Determinately, the aimed algorithm is able to perceive both the best possible number of clusters and proper apportioning in the dataset. The efficiency of the proposed algorithm is shown for twelve real-time data sets of varying complexities. The results of this multi objective clustering techniques presented in Table 1, Table 2.

3.1. INITIALIZATION

To initialize the candidate solutions, the cluster centres are encoded as chromosomes. The population α or number of candidate solutions are initialized randomly with n rows and m columns. The set of solutions are represented as $\alpha_{i,j}(0) = \alpha_j^{min} + \text{rand}(1) * \left(\alpha_j^{max} - \alpha_j^{min}\right)$ and each solution contains Max_knumber of selected cluster centers, where Max_k is randomly chosen activation thresholds in [0, 1].

3.2. OBJECTIVE / FITNESS FUNCTIONS

A straightforward way to pose clustering as an optimization problem is to optimize some CVIs that reflect the goodness of the clustering solutions. The correctness or accuracy of any optimization method depends on its objective or fitness function being used in the algorithm [2-3]. In this manner, it is regular to instantaneously advance with numerous of such measures for optimizing distinctive attributes of data. To compute the distance between the centroid and candidate solutions Euclidean distance measure is used, along with it the other objective functions optimized simultaneously are the RI, ARI, DB, CS, XI, SIL CVIs [6].

.

3.3. JAYA EVOLUTIONARY ALGORITHM

Jaya is a simple, powerful optimization algorithm proposed by R Venkata Rao in 2015 for solving the constrained and unconstrained optimization problems [15]. This algorithm is predicated on the idea that the outcome obtained for a given problem should move towards the best solution and evade the worst solution. This evolutionary approach does not require any particular algorithm-specific control parameter, rather mandates common control parameters. The working procedure of this evolutionary method is as follows:

Let $f(\alpha)$ is the objective function to be minimized or maximized. At any iteration i, assume that there are $'m'$ number of design variables i. e $(j = 1,2, \dots m)$, $'n'$ number of candidate solutions (i.e. population size, $k = 1,2, \dots n$). Let the best candidate best obtains the best value of

$f(\alpha)(\text{i. e. } f(\alpha)_{best})$ in the entire candidate solutions and the worst candidate worst obtains the worst value of $f(\alpha)(\text{i. e. } f(\alpha)_{worst})$ in the entire candidate solutions. If $\alpha_{j,k,i}$ is the value of the j^{th} variable for the k^{th} candidate during the i^{th} iteration, then this value is modified as per the following equation

$$\alpha'_{j,k,i} = \alpha_{j,k,i} + r_{1,j,i}\left[(\alpha_{j,best,i}) - |\alpha_{j,k,i}|\right] - r_{2,j,i}\left[(\alpha_{j,worst,i}) - |\alpha_{j,k,i}|\right]. \qquad (1)$$

where$\alpha_{j,best,i}$ is the value of the variable j for the best candidate and $\alpha_{j,worst,i}$ is the value of the variable j for the worst candidate. $\alpha'_{j,k,i}$is updated value of $\alpha_{j,k,i}$ and $r_{1,j,i}$ and $r_{2,j,i}$ are the two random numbers for the j^{th} variable during the i^{th} iteration in the range [0,1]. The term $r_{1,j,i}\left[(\alpha_{j,best,i}) - |\alpha_{j,k,i}|\right]$ indicates the tendency of the solution to move closer to the best solution and the term $r_{2,j,i}\left[(\alpha_{j,worst,i}) - |\alpha_{j,k,i}|\right]$ indicates the tendency of the solution to avoid the worst solution. $\alpha'_{j,k,i}$is accepted if it gives better function value. All the accepted function values at the end of the termination are maintained and these values become the input to the next iteration. At the end of each iteration all the accepted function values are retained and are fed as inputs to the next iteration. This algorithm intends to reach best solution and tries to avoid worst solution.

The steps in Jaya algorithm are as follows:

1. Initialize population size, number of design variables and termination condition
2. Identify best and worst solution in the population
3. Modify the solutions based on best and worst solutions using (1)
4. Is the solution corresponding to $\alpha'_{j,k,i}$ better than the corresponding to $\alpha_{j,k,i}$
    a. accept and replace the previous solution
5. Else keep the previous solution
6. Is the termination criterion satisfied
    a. report as optimum solution
7. Else go to Step 2

3.4. PROPOSED AUTOJAYA ALGORITHM

The working procedure of the aimed algorithm AutoJAYA is as follows:

1. Initialize the number of candidate solutions randomly as α, in n rows and m columns.
2. The set of solutions are represented as $\alpha_{i,j}(0) = \alpha_j^{min} + \text{rand}(1) * \left(\alpha_j^{max} - \alpha_j^{min}\right)$ and each solution contain Max_k number of selected cluster centers, where Max_k is randomly chosen activation thresholds in [0, 1].
3. The fitness function $f(x)$to be maximized by default is Rand Index, and the i^{th} solution of α at current generation with design variables is represented as $\left[\alpha_{i,1}(t), \alpha_{i,2}(t), \ldots, \alpha_{i,d}(t)\right]$
4. Spot the active cluster centers with value greater than 0.5 as best candidates, α_{best} solutions and less than 0.5 as worst candidates, α_{worst} solutions
5. For $t = 1\ to\ t_{max}$do
    a. For each data vector α_j, calculate its distance from all active cluster centers using Euclidean distance
    b. Assign α_j to closest cluster
    c. Evaluate each candidate solution quality using the fitness functions and find α_{best}, α_{worst} solutions
    d. Modify the solutions based on best and worst solutions using (1)

6. If the solution corresponding to $\alpha'_{j,k,t}$ better than the corresponding to $\alpha_{j,k,t}$ accept and replace the previous solution else keep the previous solution

4. EXPERIMENTAL ANALYSIS AND DISCUSSION

In this section, we report on experiments that use multi-objective clustering to identify partitions in diverged set of datasets. The enactment of the aimed algorithm is pragmatic from the results conquered by the following criteria elected, i.e. automatic k-detection, minimal consumption of CPU time, low percentage of error rate and ideal values in CVIs.The number of iterations is restricted to 30 independent runs for all the datasets. Table 1, Table 2 demonstrates the results of AutoJAYA algorithm used over real-time datasets. These real-time datasets are extracted from UCI Machine Learning Repository [18].The best results are shown as bold face.

Table 1. Results of Automatic Clustering algorithms Real-time datasets

Datasets (size*dim, k)	No. of auto clusters	CPU time (sec)	% of error rate	Mean value of CVIs					
				ARI	RI	HIM	SIL	CS	DB
Iris (150*4, 3)	3.01	19.45	10.11	0.9815	**0.9987**	0.0922	0.9214	0.8416	0.7152
Wine (178*13, 3)	3.00	105.32	40.64	0.8414	0.7912	0.4910	0.6048	0.6417	**0.8915**
Glass (214*9, 6)	6.00	114.23	30.78	0.8000	0.9000	0.6018	0.6980	0.5297	**1.0050**
Ionosphere (351*3, 4)	2.00	30.12	8.01	0.9580	0.9877	0.9632	1.2580	**1.0470**	0.9784
Ecoil (336*7, 8)	8.00	45.12	11.48	0.9587	**1.0145**	0.9964	1.0258	0.9478	0.7859
Rocks (208*60, 2)	2.01	74.20	12.45	0.9971	0.9999	**1.0000**	1.0001	0.9478	0.9481
Parkinson (195*22, 2)	2.10	42.14	12.08	0.9240	0.9920	0.9974	0.9608	0.9814	**1.0040**
Diabetic (768*9, 2)	2.00	74.25	10.45	0.9871	**0.9997**	1.0024	0.9999	0.8478	0.8481
Segment (1500*20, 2)	3.01	1041.02	14.12	0.9631	**0.9941**	0.0786	0.9740	0.8994	0.4448
Weighting (500*8, 2)	5.99	110.29	15.23	**1.0019**	0.9663	0.0222	1.0025	0.9648	0.2560
Sonar (208*60, 2)	1.98	67.20	24.23	0.9988	**1.0205**	0.9635	0.9845	0.8458	0.8932
Rippleset (250*3, 2)	2.00	10.23	5.48	1.2800	1.0200	0.9874	**1.0000**	0.9990	0.9011

The AutoJAYA renders exact number of automatic clusters in Wine, Glass, Ionosphere, Ecoil, Diabetic, Sonar, Rippleset, when compared to actual number of clusters (k) shown in column 1 of Table 1. The Rippleset dataset is the only dataset where AutoJAYA consumes minimum amount of CPU time among all the datasets of varying size and complexity. In general, the CPU time consumed by all the comparing datasets is between 5.48 sec to 1041.12 sec, which is purely dependent on the volume and complexity of the dataset. Likewise, minimal percentage of error rate is logged for the aimed algorithm in Iris and Wine datasets and the other likening datasets registers the error rate between 5.48 % and 40.64%.

The CVIs in RI in Iris, DB in Wine, DB in Glass, CS in Ionosphere, RI in Ecoil and HIM in Rocks and mines dataset registers optimal mean value, by endorsing the validity of the algorithm. The CVIs DB in Parkinson, RI in Diabetic, RI in Segment, ARI in Weighting, RI in Sonar and SIL Rippleset also follow the same tendency by submitting optimal mean value towards the frontiers of CVIs. All these implications elevate the supremacy of proposed algorithm in obtaining favourable results. The automatic clusters generated by AutoJAYA algorithm are shown in Figure 1.

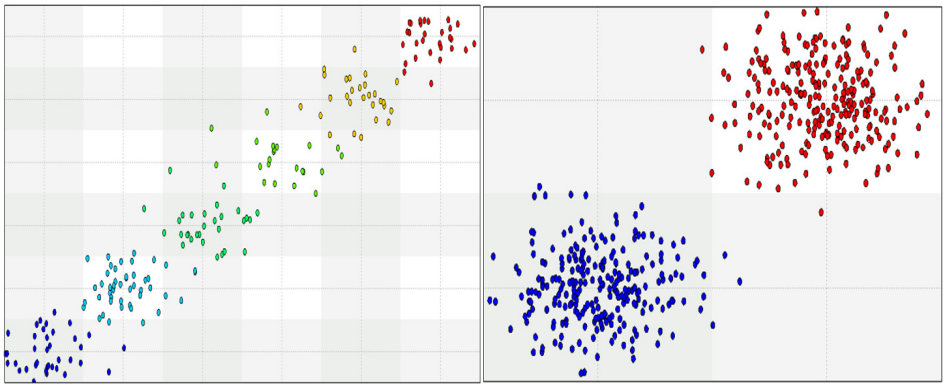

Figure 1. Automatic clusters produced by AutoJAYA in Glass and Weighting datasets

Table 2. Values of F-measure, ROC and SSE in real-time datasets using AutoJAYA algorithm

Datasets	F-Measure	ROC	SSE
Iris	0.940	0.955	7.81
Wine	1.115	1.132	9.25
Glass	0.186	0.472	52.18
Ionosphere	0.501	0.485	726.10
Ecoil	0.479	0.464	695.10
Rocks	0.171	0.420	49.812
Parkinson	1.180	0.459	50.22
Diabetic	0.513	0.497	149.51
Segment	0.043	0.496	532.78
Weighting	0.893	0.941	520.9
Sonar	0.153	0.464	21.71
Rippleset	0.441	0.421	44.81

The results of F-measure, ROC area and Sum of Squared Error (SSE) of the proposed algorithm on each real-time dataset are included in Table 2. The value deviations of F-measure, ROC area and SSE amongst all the datasets is shown in Fig. 2. It is observed from both Table 1 and Figure 2 that the aimed algorithm has obtained better result in most of the cases for all the real-time datasets.

Table 2 shows the corresponding values of F-Measure, ROC area and SSE for all comparing real-time datasets. A significant remark on Table 2 is all the datasets tender better values for F-measure and ROC area. The SSE value is very nominal for Iris and Wine datasets and relatively mere optimal values for remaining datasets.

The culminating remarks after examining the applicability of AutoJAYA algorithm over real-time is the aimed algorithm lodges better in most of the datasets in identifying the exact number of automatic partitions, with minimum consumption of CPU and relatively low percentage of error rate.. Hence these experiments speculate fact that AutoJAYA algorithm lay bare the advantages of multiobjective clustering optimized with evolutionary approaches decipher into realistic and scalable performance paybacks.

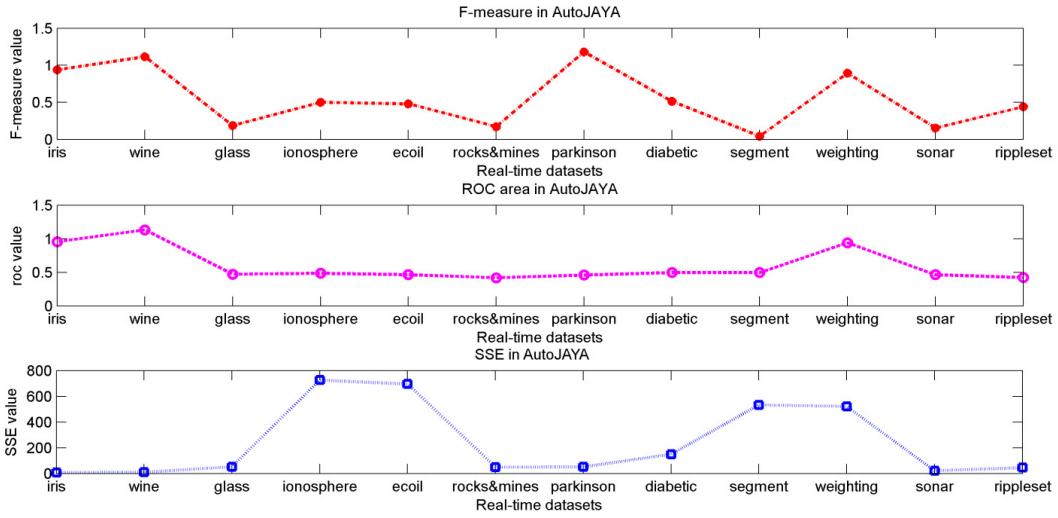

Figure 2. Values of F-Measure, ROC and SSE rendered by AutoJAYA in real-time datasets

5. CONCLUSIONS

In this article, a novel multi-objective clustering technique AutoJAYA based on the newly developed Jaya Evolutionary algorithm is proposed. The explorations and exploitations enforced on the technique, automatically determine the proper number of clusters, proper partitioning from a given dataset and mere optimal values towards CVIs frontiers, by optimizing fitness functions simultaneously.

Furthermore, it is observed that the aimed algorithm exhibits better performance in most of the considered real-time datasets and is able to cluster appropriate partitions. Much further work is needed to investigate the profound algorithm using different and more objectives, compare with well established automatic clustering algorithm and to test the approach still more extensively over diversified domains of engineering.

REFERENCES

[1] Zitzler, Eckart, Marco Laumanns, and Stefan Bleuler. "A tutorial on evolutionary multiobjective optimization." Metaheuristics for multiobjectiveoptimisation. Springer Berlin Heidelberg, 2004. 3-37.

[2] SriparnaSaha, SanghamitraBandyopadhyay,"A new point symmetry based fuzzy genetic clustering technique for automatic evolution of clusters",Information Sciences 179, 2009, pp. 3230–3246, doi:10.1016/j.ins.2009.06.013

[3] SriparnaSaha, SanghamitraBandyopadhyay,"A symmetry based multiobjective clustering technique for automatic evolution of clusters", Pattern Recognitions 43, 2010, pp. 738-751, doi:10.1016/j.patcog.2009.07.004

[4] Eduardo Raul Hruschka, Ricardo J. G. B. Campello, Alex A. Freitas, and Andr´e C. Ponce Leon F. de Carvalho, "A Survey of Evolutionary Algorithms for Clustering", IEEE transactions on systems, man, and cybernetics—part c: applications and reviews, Vol. 39-2, 2009, pp. 133-155.

[5] NobukazuMatake, Tomoyuki Hiroyasu, Mitsunori Miki, TomoharuSenda, "Multiobjective Clustering with Automatic k-determination for Large-scale Data", GECCO'07, July 7–11, 2007, London, England, United Kingdom, ACM 978-1-59593-697-4/07/0007

[6] EréndiraRendón, Itzel Abundez, Alejandra Arizmendi and Elvia M. Quiroz., "Internal versus External cluster validation indexes", International journal of computers and communications, 1(5), 2011.

[7] Mukhopadhyay, A., Maulik, U., &Bandyopadhyay, S. (2015). A Survey of Multiobjective Evolutionary Clustering. ACM Computing Surveys (CSUR),47(4), 61.

[8] Abadi, M. F. H., &Rezaei, H. (2015). Data Clustering Using Hybridization Strategies of Continuous Ant Colony Optimization, Particle Swarm Optimization and Genetic Algorithm. British Journal of Mathematics & Computer Science, 6(4), 336.

[9] Ozturk, C., Hancer, E., &Karaboga, D. (2015). Dynamic clustering with improved binary artificial bee colony algorithm. Applied Soft Computing, 28, 69-80.

[10] Kumar, V., Chhabra, J. K., & Kumar, D. (2014). Automatic cluster evolution using gravitational search algorithm and its application on image segmentation. Engineering Applications of Artificial Intelligence, 29, 93-103.

[11] Kuo, R. J., Huang, Y. D., Lin, C. C., Wu, Y. H., &Zulvia, F. E. (2014). Automatic kernel clustering with bee colony optimization algorithm.Information Sciences, 283, 107-122.

[12] Wikaisuksakul, S. (2014). A multi-objective genetic algorithm with fuzzy c-means for automatic data clustering. Applied Soft Computing, 24, 679-691.

[13] Mukhopadhyay, A., Maulik, U., Bandyopadhyay, S., &CoelloCoello, C. (2014). A survey of multiobjective evolutionary algorithms for data mining: Part I. Evolutionary Computation, IEEE Transactions on, 18(1), 4-19.

[14] Mukhopadhyay, A., Maulik, U., Bandyopadhyay, S., &Coello, C. (2014). Survey of multiobjective evolutionary algorithms for data mining: Part II.Evolutionary Computation, IEEE Transactions on, 18(1), 20-35.

[15] R. Venkata Rao, "Jaya: A simple and new optimization algorithm for solving constrained and unconstrained optimization problems",International Journal of Industrial Engineering Computations, 7, 2016, doi: 10.5267/j.ijiec.2015.8.004

[16] Ramachandra Rao Kurada, KanadamKarteekaPavan, AllamAppaRao,"Automatic Teaching–Learning-Based Optimization-A Novel Clustering Method for Gene Functional Enrichments",Computational Intelligence Techniques for Comparative Genomics, SpringerBriefs in Applied Sciences and Technology.2015. 10.1007/978-981-287-338-5.

[17] Ramachandra Rao Kurada, KarteekaPavanKanadam, "A generalized automatic clustering algorithm using improved TLBO framework", Int. Journal of Applied Sciences and Engineering Research, Vol. 4, Issue 4, 2015, ISSN 2277 – 9442.

[18] Lichman, M. (2013). UCI Machine Learning Repository [http://archive.ics.uci.edu/ml]. Irvine, CA: University of California, School of Information and Computer Science.