Transactions on Engineering and Computing Sciences - Vol. 12, No. 2
Publication Date: April 25, 2024
DOI:10.14738/tecs.122.14317.
Quddus, S., & Bagirov, A. (2024). A Comparative Study of Fast and Accurate Clustering Algorithms in Multi-Sized Data Sets. Transactions on Engineering and Computing Sciences, 12(2). 110-120.
SERVICES FOR SCIENCE AND EDUCATION

A Comparative Study of Fast and Accurate Clustering Algorithms in Multi-Sized Data Sets

Syed Quddus
University of the Bahamas
UB │Oakes Field P.O. Box N-4912 │Nassau, Bahamas

Adil Bagirov
Federation University, Ballarat Vic, Australia

ABSTRACT

Unsupervised learning or clustering in large data sets is a challenging problem. Most clustering algorithms are not efficient and accurate in such data sets. Therefore, development of clustering algorithms capable of solving clustering problems in large data sets is very important. In this paper, we present an overview of various algorithms and approaches which are recently being used for Clustering of large data and Edocument. We use the squared Euclidean norm to define the similarity measure. In this paper, a comparative study of the performance of various clustering algorithms: the global kmeans algorithm (GKM), the multi-start modified global kmeans algorithm (MSMGKM), the multi-start kmeans algorithm (MS-KM), the difference of convex clustering algorithm (DCA), the incremental clustering algorithm based on the difference of convex representation of the cluster function and non-smooth optimization (DC-L2), is carried out using Python. CCS Concepts: Information systems Data mining, Information systems, Data cleaning, Information systems Clustering.

Keywords: Cluster Analysis, Data Mining, Algorithms.

INTRODUCTION

Clustering is among most important tasks in data mining. It has many applications in business, biology, astronomy, to name just a few. Clustering is a process of dividing, grouping a dataset into meaningful partitions based on some criteria. In recent years there has been a rapid and massive increase in amount of data accumulated. This stimulates the development of new clustering algorithms applicable to large datasets. Clustering is now a key component of interactive- systems which gather information on millions of users on everyday basis [1-10, 20]. Existing clustering algorithms are not always efficient and accurate in solving clustering problems in large datasets. The accurate and real time clustering is essential and important for making informed policy, planning and management decisions. Recent developments in computer hardware allow to store in Random Access Memory of computers and repeatedly read data sets with hundreds of thousands and even millions of data points. However, most of existing clustering algorithms require much larger computational time and fail to produce an accurate solution in such data sets [16, 17, 18, and 19]. Therefore, it is important to develop accurate and fast (real-time) clustering algorithms. One such algorithm is studied in this paper using results numerical experiments.

More specifically, we consider an incremental clustering algorithm based on the nonsmooth optimization formulation of the clustering problems. We use the squared Euclidean norm to define the similarity measure. This type of clustering is also known as the minimum sum-of-squares clustering. Our algorithm also exploits the difference of convex representation of the clustering function. Details of this algorithm can be found in [9]. In this paper, we modified this algorithm by improving the procedure for finding starting cluster centres. In numerical experiments we use data sets containing from tens of thousands to hundreds of thousands of data points. Our results demonstrate that optimization based clustering algorithms can be extended to solve clustering problems.

Optimization Algorithms for Lustring Problems

Optimization methods, both deterministic and stochastic, have been applied to develop different algorithms for solving clustering problems and especially, for solving the minimum sum-of-squares clustering problems. These algorithms can be categorized into the following groups:

In this section we give a nonsmooth optimization formulation of clustering problems and their DC representations [3, 8, and 9]. In cluster analysis we assume that we are given a finite set of points A in the n−dimensional space R^n, that is $A = \{a^1, \ldots, a^m\}$, where $a^i \in R^n$,, $i = 1, \ldots, m$. The hard unconstrained clustering problem is the distribution of the points of the set A into a given number k of disjoint subsets Aj, j = 1, . . ., k such that [24]:

1. $A^j \neq \emptyset \; and \; A^j \cap A^l = \emptyset, j, l = 1, \ldots., k, j \neq 1$
2. $A = U_{j=1}^{k} A^j$

The sets A^j, $j = 1, \ldots, k$ are called clusters and each cluster A^j can be identified by its center $x_j \in R_n, j = 1, \ldots, k$. The problem of finding these centers is called the k-clustering (or k-partition) problem. In order to formulate the clustering problem, one needs to define the similarity (or dissimilarity) measure. In this paper, the similarity measure is defined using the L_2 norm:

$$d_2(x, a) = \sum_{i=1}^{n} (x_i - a_i)^2$$

The nonsmooth optimization formulation of the MSSC problem is [8, 9]: minimize $fk\ (x)$ subject to $x = (x^1 \ldots, x^k) \in R^{nk}$

$$fk(x^1, \ldots \ldots, x^k) = \frac{1}{m} \sum_{a \epsilon A} \min_{j=1,\ldots,k} d_2(x^j, a)$$

The above objective function $fk\ (x)$ can be expressed as a DC function:

$$fk\ (x) = fk1(x) - fk2(x), x = (x^1, \ldots, x^k) \in R^{nk},$$

where,

$$fk1(x) = \frac{1}{m}\sum_{a\epsilon A}\sum_{j=1}^{k} d_2(x^j, a)$$

and

$$fk2(x) = \frac{1}{m}\sum_{a\epsilon A}\max_{j=1,..,k}\sum_{s=1,s\neq j}^{k} d_2(x^8, a)$$

solution to the (k-1)-clustering problem is known and we show how each algorithm solves the k-clustering problem.

All these algorithms compute clusters incrementally that is they start with one cluster center which is the center of the whole data set A and gradually add a new cluster center. Main difference between these algorithms is in the way they compute the starting point for the next cluster center. For given k>1 assume that the solution to the (k-1)-clustering problem is known and we show how each algorithm solves the k-clustering problem.

Global k-Means Algorithm (GKM)

The GKM algorithm is introduced in. In this algorithm all data points are considered as a starting point for the kth cluster center. Each data point is added to (k-1) cluster centers and the k-means algorithm starts from these points to solve k-clustering problem [16,18]. This means that one gets new m solutions to the k-clustering problem. The solution with the smallest value of the objective function in clustering problem is chosen as a solution to the k-clustering problem. Although such an algorithm is accurate however it is not efficient even for data sets containing tens thousands of data points. Therefore, in its implementation not all data points but the data point providing largest decrease of the objective function is chosen as a starting point for the k^{th} cluster center [20, 23].

Modified Global kmeans Algorithm (MGKM)

The MGKM algorithm was introduced in [16]. In this algorithm data points providing the decrease of the objective function more than some thresholds are chosen as starting points to solve the auxiliary clustering problem. Then the solution to the auxiliary clustering problem with the smallest value of the auxiliary clustering function is chosen as the starting point for the kth cluster center. This point is added to $k-1$ cluster centers to form a starting point for the k-clustering problem and the kmeans algorithm is applied to solve the problem starting from this starting point [16.18].

Multi-start Modified Global Kmeans Algorithm (MS-MGKM)

The multi-start modified global kmeans algorithm (MSMGKM) was developed. This algorithm is an extension of the MGKM algorithm. In this algorithm a special procedure is introduced to generate a set of starting cluster centers. Data points and the auxiliary clustering problem are used to generate these centres [13, 21, and 22]. The kmeans algorithm is applied starting from each of these points and previous k-1 cluster centers to solve the k-clustering problem. The

best solution with lowest value of the clustering function is accepted as a solution to the kth clustering problem.

Difference of Convex Model Based Clustering Algorithm (DC-L2)

This algorithm was developed in [2,17,19]. It is based on the difference of convex model of the clustering problem. The nonsmooth optimization algorithm was introduced to solve this problem. A special procedure is applied to get good starting points for cluster centers.

Algorithm Based on The Difference of Convex Algorithm (DCA)

This algorithm is considered in [2]. It is also based on the difference of convex model of the clustering problem. The Difference of Convex Algorithm, introduced in [12,14], is applied to solve optimization problems both clustering and auxiliary clustering problems. A special procedure is used to generate starting points for cluster centers.

DATA SETS AND IMPLEMENTATION

Five real-life data sets have been used in numerical experiments [1,21]. The brief description of these data sets is given in Table 1. All data sets contain only numeric features and they do not have missing values.

Table 1: The brief description of datasets [25,26].

Data Sets	Data Points	No. of Features	No. of Entrices
Waveform Generator	5000	40	200,000
Bank Marketing	45211	17	768587
Artificial-2stateequence	250000	14	3500000
Shuttle Landing	58000	10	58000
Ijcnn1	191681	23	4408663

Five data sets were used in numerical experiments. We include small, medium size and large data sets to demonstrate accuracy and efficiency of these algorithms in comparison. The detailed description of data sets can be found in [1,26, 27].

To get as more comprehensive picture about the performance of the algorithm as possible the datasets were chosen so that:(i) the number of attributes is ranging from small (10) to large (40); (ii) the number of data points is ranging from thousands (smallest1, 5000) to hundreds of thousands (largest 250000). We computed up to 25 clusters in all data sets. The CPU time used by algorithms is limited to 20h. we present results with the maximum number of clusters obtained by an algorithm during this time. The algorithm was implemented in python compiler. Computational results were obtained on a Laptop with the Intel(R) Core (TM) i3-3110M CPU @ 2.4GHz and RAM 4 GB (Toshiba). Five data sets were used in numerical experiments.

The algorithm was implemented in python. Computational results were obtained on a Laptop with the Intel(R) Core (TM) i3-3110M CPU @ 2.4GHz and RAM 4 GB (Toshiba). Data set Ijcnn1 is a dataset which contains information on protein sequence identication and it is developed by using winner's transformation theorem. Five categorical features are removed from data-input file [25, 27].

NUMERICAL RESULTS

We run experiments on these real-life data sets to compute the Cluster function values obtained by algorithms, CPU time and the total number of distance function evaluations for all these five datasets.

To present numerical results the following notations are used:

k - is the number of clusters; f - is the optimal value of the clustering function obtained by the algorithm; N - is the total number of distance function evaluations; t- is the CPU time.

The following algorithms are used for comparison: the global kmeans algorithm (GKM), the multi-start modified global kmeans algorithm (MS-MGKM), the multi-start kmeans algorithm (MS-KM), the difference of convex clustering algorithm (DCA), the clustering algorithm based on the difference of convex representation of the cluster function and nonsmooth optimization (DC-L2) [1]. The results of implementation of these algorithms are illustrated, respectively, in Figures-1-2,3-4,5-6,7-8 and in 9-10 for the five real time datasets. All these data sets are taken from three categories of datasets [1-5].

The first category is called small datasets and it contains data sets with small number of attributes, the number of instances in these data sets is less than ten thousand and the number of attributes is ranging from 10 to 40.

Clustering results for the Waveform Generator data set are illustrated in Figures-1-2. Results depicted in Figures-1-2, demonstrate that in these datasets the performance of algorithm is similar in the sense of accuracy. All algorithms can find at least near best known solutions in these datasets.

The second category is called medium size datasets and it contains data sets with relatively large number of attributes. The number of instances in these data sets is between ten thousand and one hundred thousand and the number of attributes is ranging from 2 to 128. Clustering results for the Bank Marketing data set are illustrated in Figures-3-4. Results depicted in Figures-3-4, show that the algorithm is very efficient to find (near) best known solutions. Clustering results for the Shuttle Landing data set are illustrated in Figures-5-6. Results depicted in Figures-5-6, show that the algorithm is very efficient to find (near) best known solutions. The dependence of the number of distance function evaluations on the number of clusters in group1 of datasets is similar and the dependence of the number of distance function evaluations on the number of clusters in group2 of datasets is also similar.

The third category of datasets is called large and very large to big datasets data sets. The number of instances in these data sets is from one hundred thousand to over one million and the number of attributes is ranging from 2 to 23.

Results for the data set Artificial-2state-equence are illustrated in Figures-7-8. Figures-7-8, illustrate that the algorithms: DC L2 and DCA outperform better than other algorithms in sense of clustering accuracy. These figures significantly depict that the algorithms: DC-L2 and DCA

use computational time less than other algorithms. Results for the Ijcnn1 data set are illustrated in Figures-9-10. Figures-9-10 illustrate dependence of the number of distance functions evaluations on the number of clusters. This figure clearly shows that the algorithms: GKM and MS-MGKM and DC-L2 require significantly less distance function evaluations than any other clustering algorithms used in comparison. Moreover, they also use less computational time than the other algorithms [1,2,3,4].

The dependence of the CPU-time on the number of clusters for all datasets in group1 is similar. As the number of clusters increase, the dependence of CPU time monotonically increases. It is obvious that as the size (the number of data points) of a data set increase this algorithm requires more CPU time. But the algorithm takes almost similar time pattern in clustering datasets: Shuttle Control data and Bank Marketing data sets [1].

CONCLUSION

In this paper, we developed these algorithms for solving the minimum sum-of-squares clustering problems. In order to design these algorithms, we use non-smooth non-convex formulation of the clustering problem and apply the hyperbolic smoothing technique to approximate this problem with the sequence of smooth optimization problems [1,3]. We implemented the algorithm in Python and presented the results of the numerical experiments. Results demonstrate that these algorithms are able to find global solutions to clustering problems with up to a level of accuracy. We also compared the performance of five clustering algorithms: (the global kmeans algorithm (GKM), the multi-start modified global kmeans algorithm (MS-MGKM), the multi-start kmeans algorithm (MS-KM), the difference of convex clustering algorithm (DCA), the clustering algorithm based on the difference of convex representation of the cluster function and nonsmooth optimization (DCClust)). This comparison demonstrate that these algorithms differ with each other in terms of accuracy and efficiency and their computational efforts varies in small, medium and large size data sets. In conclusion, we can say that this research demonstrates how the existing clustering algorithms can be scaled up to solve clustering problems in very large data sets [1,3,4,5].

FUTURE WORK

As we know very large data set clustering is an emerging field. In this research, different evolutionary methods, their features and their applications have been discussed. We have implemented the techniques and we may implement more in future. The expectation is to implement the best technique which can efficiently solve the minimum sum-of-squares clustering problems and find the best solution in real time. In this paper we did not consider the problem of clustering in data sets which cannot be stored in the random-access memory of a computer. These problems can be considered as directions of future research. The main direction which can be identified: Most clustering algorithms have good potential for parallelization. The use of many processors in supercomputers will significantly accelerate the convergence of such algorithms. Here we can mention two possibilities for parallelization. One possibility is that to divide the data set into many pieces and pass each piece to one processor and to solve the clustering separately for each piece of data. Then special techniques should be developed to merge clustering results from each processor [1,5,6].

ACKNOWLEDGMENTS

This research by Syed Abdul Quddus and Dr. Adil Bagirov was supported under Australian Research Council's training and research funding scheme.

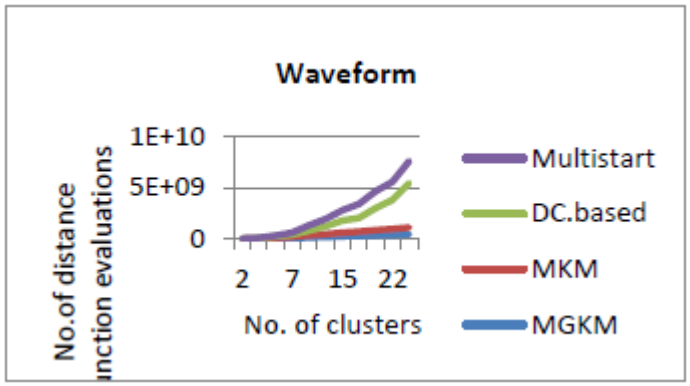


Figure 1: Waveform Generator data set: Cluster function

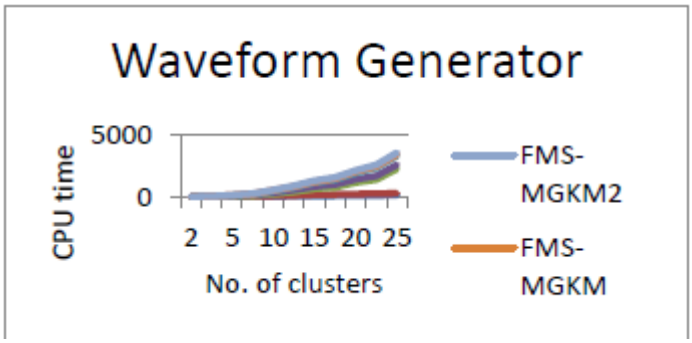


Figure 2: Waveform Generator data set: CPU time in Seconds

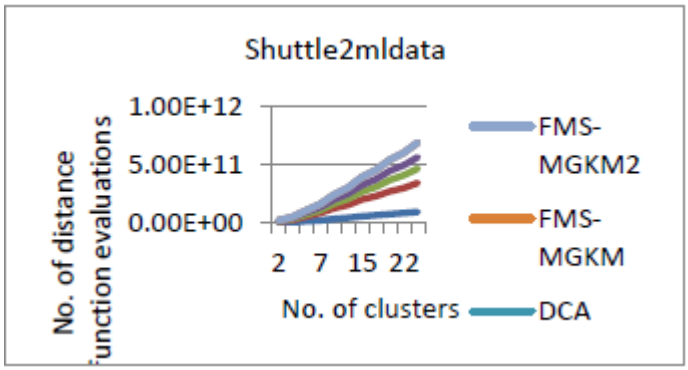


Figure 3: Shuttle data set: Cluster function

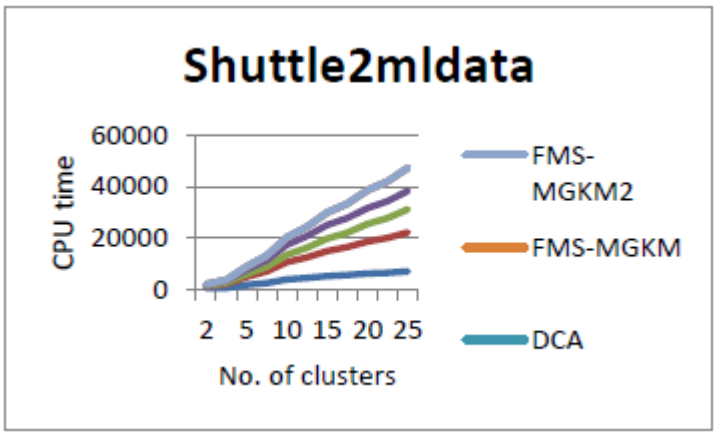


Figure 4: Shuttle data set: CPU time

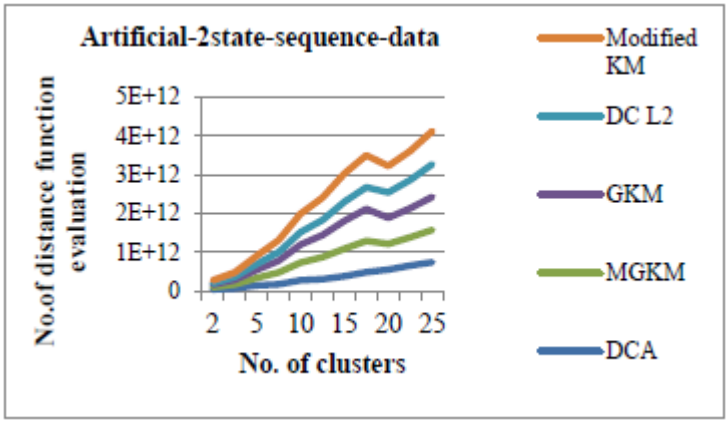


Figure 5: Artificial-2state-data cluster function

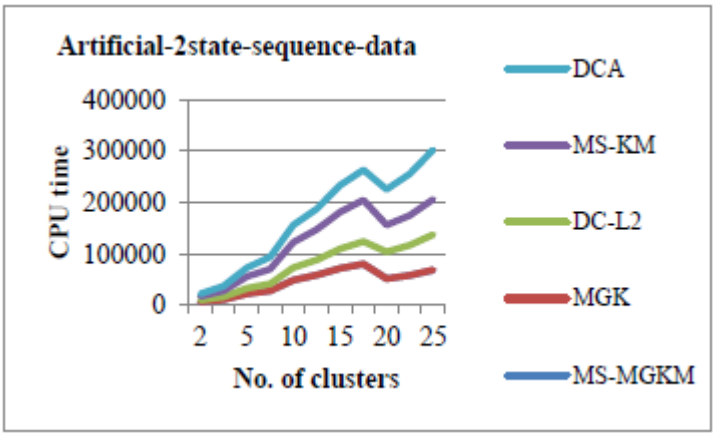


Figure 6: Artificial-2state -data: CPU-time.

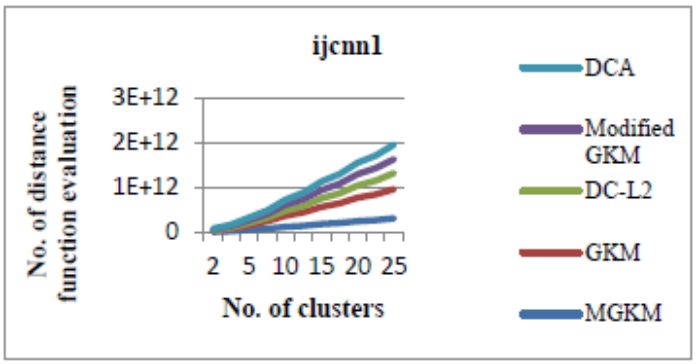
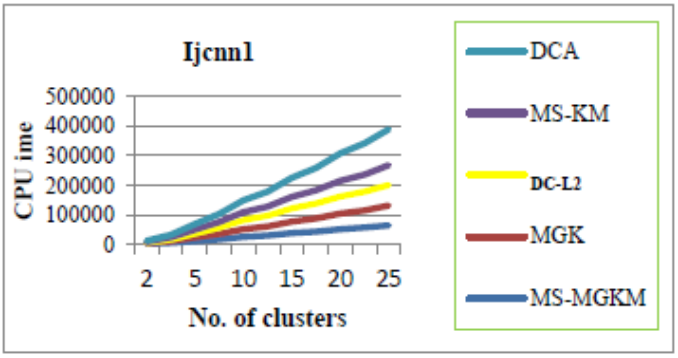


Figure 9: Ijcnn1 data set: Cluster functions.



References

[1] A.M. Bagirov and S. Quddus, (2019), Acomparative study of unsupervised classification algorithms in multi-sized data sets, International Conference Proceedings Series by ACM, ISBN 978-1-4503-7263-3; https://doi.org/10.1145/3375959.3375979

[2] Quddus S, Bahirov. A, "Fast algorithms for unsupervised learning in large data sets" in the Computer Science Conference Proceedings in Computer Science & Information Technology (CS & IT) series, (2017), 15-17.

[3] A.M. Bagirov, S. Taheri and J. Ugon, Nonsmooth DC programming approach to the minimum sum-of-squares clustering problems, Pattern Recognition ,53 (2016), 53, 12-24.

[4] Petar Ristoski, Heiko Paulheim, Semantic Web in data mining and knowledge discovery: A comprehensive survey, Science, Services and Agents on the World Wide Web, 36 (2016), 1-22.

[5] M. Vahdat, A. Ghio, L. Oneto, D. Anguita, M. Funk, M. Rauterberg, Advances in learning analytics and educational data mining, in: European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning (2015).

[6] A.M. Bagirov, S. Taheri and J. Ugon, Nonsmooth DC programming approach to the minimum sum-of-squares clustering problems, Pattern Recognition, 53 (2016), 53, 12-24.

[7] A.M. Bagirov, S. Taheri and J. Ugon, Nonsmooth DC programming approach to the minimum sum-of-squares clustering problems, Pattern Recognition, 53 (2016), 53, 12-24.

[8] Petar Ristoski, Heiko Paulheim, Semantic Web in data mining and knowledge discovery: A comprehensive survey, Science, Services and Agents on the World Wide Web, 36 (2016), 1-22.

[9] M. Vahdat, A. Ghio, L. Oneto, D. Anguita, M. Funk, M. Rauterberg, Advances in learning analytics and educational data mining, in: European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning (2015).

[10] A.M. Bagirov, B. Ordin, G. Ozturk, A.E. Xavier, an incremental clustering algorithm based on hyperbolic smoothing, Comput. Optim. Appl, 61(2015), 219-241.

[11] B. Ordin and A.M. Bagirov, A heuristic algorithm for solving the minimumsum of squares clustering problems, J. Global Optim, 61(2015), 341-361.

[12] Neha Khan, Mohd Shahid, Mohd Rizwan, Big data classification using evolutionary techniques: A survey, (2015), IEEE International Conference (ICETECH), India.

[13] Md A. Rahman and Md Z. Islam, A hybrid clustering technique combining a novel genetic algorithm with kmeans, Knowledge-Based Systems, 71 (2014), 345-365.

[14] L.T.H. An, H.V. Ngai and P.D. Tao, Exact penalty and error bounds in DC programming, J. Global Optim, 52(3) (2012), 509-535.

[15] A.M. Bagirov, J. Ugon and D. Webb, Fast modified global kmeans algorithm for sum-ofsquares clustering problems, Pattern Recognition, 44 (2011), 866-876.

[16] A.E. Xavier and V.L. Xavier, Solving the minimum sum-of-squares clustering problem by hyperbolic smoothing and partition into boundary and gravitational regions, Pattern Recognition, 44(1), 2011, 70--77.

[17] A.E. Xavier, The hyperbolic smoothing clustering method, Pattern Recognition, 43(3), 2010, 731-737.

[18] A.M. Bagirov, modified global kmeans algorithm for sum-of-squares clustering problems, Pattern Recognition, 41(10), 2008, 3192--3199.

[19] Meng Piao Tan, James R. Broach, Christodoulos A. Floudas, A novel clustering approach and prediction of optimal number of clusters: global optimum search with enhanced positioning, Journal of Global Optimization, 2007, 39(3), 323--346.

[20] J.Z.C. Lai, T.-J. Huang, Fast global k-means clustering using cluster membership and inequality, Pattern Recognition, 2010, 43(3), 731--737.

[21] H.D. Meng, Y.C. Song, F.Y. Song and S.L. Wang, Clustering for Complex and Massive Data, International Conference on Information Engineering and Computer Science, 2009, 1--4.

[22] N. Alex, A. Hasenfuss and B. Hammer, Patch clustering for massive data sets. Neurocomputing, 72(2009), 1455-1469.

[23] A.M. Bagirov, J. Yearwood, A new nonsmooth optimization algorithm for minimum sum-ofsquares clustering problems, European Journal of Operational Research, 170(2006) 578-596.

[24] H.D. Sherali and J. Desai, A global optimization RLT-based approach for solving the hard clustering problem, Journal of Global Optimization, 2005, 32, 281--306.

[25] A. Likas, M. Vlassis, J. Verbeek, The global kmeans clustering algorithm, Pattern Recognition, 36(2003) 451-461.

[26] A.M. Bagirov, A.M. Rubinov, J. Yearwood, A global optimisation approach to classification, Optimization and Engineering, 3(2) (2002) 129-155.

[27] Chih-Chung Chang and Chih-Jen Lin, IJCNN 2001 challenge: General-ization ability and text decoding. In Proceedings of IJCNN. IEEE, 2001.

[28] Machine learning repository mldata.ord, http://mldata.org/repository/data/

[29] Blake, C., Keogh, E., Merz, C. J: UCI repository of machine learning databases, http://www.ics.uci.edu/mlearn/MLRepository.html.Irvine, CA: University of California, Department of Information and Computer Science, 1998.