Hindawi
Advances in Fuzzy Systems
Volume 2018, Article ID 2634861, 8 pages
https://doi.org/10.1155/2018/2634861

Research Article

KC-Means: A Fast Fuzzy Clustering

Israa Abdzaid Atiyah,[1] Adel Mohammadpour ,[1] and S. Mahmoud Taheri[2]

[1] *Faculty of Mathematics and Computer Science, Amirkabir University of Technology, Tehran, Iran*
[2] *School of Engineering Science, College of Engineering, University of Tehran, Tehran, Iran*

Correspondence should be addressed to Adel Mohammadpour; adel@aut.ac.ir

Received 17 January 2018; Accepted 17 April 2018; Published 3 June 2018

Academic Editor: Ferdinando Di Martino

A novel hybrid clustering method, named *KC*-Means clustering, is proposed for improving upon the clustering time of the Fuzzy *C*-Means algorithm. The proposed method combines *K*-Means and Fuzzy *C*-Means algorithms into two stages. In the first stage, the *K*-Means algorithm is applied to the dataset to find the centers of a fixed number of groups. In the second stage, the Fuzzy *C*-Means algorithm is applied on the centers obtained in the first stage. Comparisons are then made between the proposed and other algorithms in terms of time processing and accuracy. In addition, the mentioned clustering algorithms are applied to a few benchmark datasets in order to verify their performances. Finally, a class of Minkowski distances is used to determine the influence of distance on the clustering performance.

1. Introduction

Clustering is a method of separating similar data from distinctly different ones into relevant categories or clusters. Being an unsupervised approach, it helps to recognize and extract hidden patterns within the data. The distance, such as Euclidean and Manhattan as a special case of Minkowski, plays an important role in clustering algorithms. Clustering techniques enjoy some advantages as no requirement for domain knowledge or labeled data while they are able to deal with a wide variety of data, including noise and outliers, as well.

Clustering methods may be categorized into two general types: hard and soft. Hard clusters possess well-defined boundaries; examples include *K*-Means (KM) and hierarchical methods [1]. To improve the time processes of fuzzy clustering, we propose a 2-step hybrid method of *K*-Means Fuzzy *C*-Means (KCM) clustering that combines the KM clustering algorithm with that of the Fuzzy *C*-Means (CM).

We begin with a review of the current literature on classical and fuzzy clustering methods. Huang [2] extended the KM algorithm to categorical domains. In order to decrease the computational complexity associated with the conventional CM clustering method, Chang et al. [3] proposed a CM using the cluster center displacement of successive iterative processes clustering method. Volmurgan [4] investigated the performance of two partitions-based clustering methods, i.e., KM and CM algorithms. He made the comparison through clustering randomly distributed data points. Havens et al. [5] compared the efficacy of three different techniques in order to extend the application of CM clustering to very large datasets. Panda et al. [6] implemented clustering techniques in such wide areas as medicine, business, engineering systems, and image processing. Grover [7] studied a wide variety of fuzzy clustering methods such as CM, Possibilistic CM, and Fuzzy Possibilistic CM algorithm and reported their advantages and drawbacks. Bora and Gupta [8] conducted a comparative study of the fuzzy and hard clustering methods. Finally, Fajardo et al. [9] investigated the fuzzy clustering of certain spectra for the objective recognition of soil morphological horizons in soil profiles.

The present study proposes a hybrid clustering algorithm by the name of KCM that combines KM and CM algorithms to achieve its objective by improving the time processing of the CM method. The performances of KM, CM, and KCM techniques are then compared in terms of their accuracy and time processing using simulated data from sub-Gaussian distributions. The methods are also applied to the three standard real datasets, to determine and compare the precision and accuracy of the investigated algorithms.

```
(1) Let $T$ be the maximum number of iterations allowed, $0 < \varepsilon < 1$, $v_i^{(t)}$, $t = 0$;
be the initial centers, $i = 1, \ldots, c$, and $V^{(t)}$ be the set of centers in the iteration $t$;
(2) Compute the value of $\|x_k - v_i\|_p$,
(3) Assign the elements $x_k$ to the clusters, according to
    $A_i(x_k) = \begin{cases} 1, & \|x_k - v_i\|_p = \min_{l=1,\ldots,c} \|x_k - v_l\|_p \\ 0, & \text{otherwise;} \end{cases}$
(4) Update the cluster centers, take $t = t + 1$, $v_i^{(t)} = \sum_{k=1}^{n} A_i(x_k) x_k / \sum_{k=1}^{n} A_i(x_k)$;
(5) If $\|V^{(t-1)} - V^{(t)}\|_p < \varepsilon$; or $T = t$, then stop. Otherwise, go to step (2);
(6) End.
```

Algorithm 1: KM algorithm.

Finally, KM, CM, and KCM are compared using Minkowski distances. The objective is to identify the best combinations of the clustering method and distance measure with higher precision, accuracy measures, and cluster quality in terms of compactness and distinctiveness.

2. Clustering Algorithms

By definition, clustering groups a sample of vectors to c clusters, using an appropriate similarity criterion such as distance from the center of the cluster.

2.1. K-Means Algorithm. KM is one of the most popular clustering algorithms [10, 11]. The clustering results of the KM algorithm are very sensitive to the positions of the initial cluster centers. Being efficient in clustering large data sets, it often terminates at a local optimum and applies only to numeric values [12]. Given a set of n elements $\{x_1, \ldots, x_n\}$ and a set of centers $V = \{v_1, \ldots, v_c\}$, where $x_k = (x_{k1}, \ldots, x_{kd})$, $v_i = (v_{i1}, \ldots, v_{id}) \in R^d$, $k = 1, \ldots, n$; $i = 1, \ldots, c$, We recall that Minkowski distance for two points x_k, v_i is defined as follows:

$$\mathrm{MD}\left(x_k, v_i\right) = \left\|x_k - v_i\right\|_p = \left(\sum_{j=1}^{d} \left(x_{kj} - v_{ij}\right)^p\right)^{1/p}. \quad (1)$$

Euclidean and Manhattan distances are two special cases of Minkowski distance with $p = 2$ and $p = 1$, respectively. In the rest of the paper, for Minkowski distance, we consider $p = 1.5$. The steps of the KM clustering algorithm are shown in Algorithm 1.

2.2. Fuzzy Clustering Algorithms. In KM clustering, data is divided into disjoint clusters, where each data element belongs to exactly one cluster. In fuzzy clustering, an object can belong to one or more clusters with probabilities [13]. One of the most widely used fuzzy clustering methods is the CM algorithm, originally due to Dunn [14] and later modified by Bezdek [15]. The CM method attempts to partition a finite collection of n elements $X = \{x_1, \ldots, x_n\}$ to a collection of c fuzzy clusters with respect to some given criterion, where $x_k \in R^d$ is an observation vector. A fuzzy c-partition of X is a family of fuzzy subsets of X denoted by $U = \{A_1, \ldots, A_c\}$, which satisfies $\sum_{i=1}^{c} A_i(x_k) = 1$, $k = 1, \ldots, n$, and $0 < \sum_{k=1}^{n} A_i(x_k) < n$, $i = 1, \ldots, c$, where $c < n$ is a positive integer. The problem of fuzzy clustering is to find a fuzzy c-partition and the associated cluster centers by which the structure of the data is represented as best as possible. To solve the problem of fuzzy clustering, this criterion needs to be formulated in terms of a performance index. The c cluster centers $v_1, v_2, \ldots, v_c$ associated with the partition are calculated as follows:

$$v_i = \frac{\sum_{k=1}^{n} \left[A_i\left(x_k\right)\right]^m x_k}{\sum_{k=1}^{n} \left[A_i\left(x_k\right)\right]^m}, \quad i = 1, \ldots, c, \quad (2)$$

where $m > 1$ is a real number that governs the influence of membership grades, v_i is viewed as the cluster center of the fuzzy class A_i, and the performance index of a fuzzy c-partition U, $J_m(U)$, is then defined in terms of the cluster centers using the formula

$$J_m\left(U\right) = \sum_{k=1}^{n} \sum_{i=1}^{c} \left[A_i\left(x_k\right)\right]^m \left\|x_k - v_i\right\|_p^2. \quad (3)$$

This performance index measures the weighted sum of distances between cluster centers and elements in the corresponding fuzzy clusters. The goal of the CM clustering method is to find a fuzzy c-partition that minimizes the performance index $J_m(U)$. In other words, the clustering problem is an optimization problem [16]. The convergence properties of CM algorithms are theoretically important. The optimal cluster centers are the fixed points of CM clustering algorithms. The algorithm is limited by long computational time and sensitivity to noise, outliers, and initial guess [17, 18]. The two steps of the CM clustering algorithm which should be modified in KM algorithm are shown in Algorithm 2.

3. A Hybrid Method: *KC*-Means Algorithm

A novel approach called KCM method is proposed herein that combines the KM and CM methods. The combination is meant to overcome the limitations of both but enjoys their advantages. One of the disadvantages of CM method is long computational time while quick running is one of the advantages of KM method. The goal of the hybrid method is to introduce a fuzzy method faster than CM while its

```
(3) Compute the membership and assign the elements $x_k$ to the clusters, according to
    $A_i(x_k) = \left( \sum_{j=1}^{c} \left( \frac{\|x_k - v_i\|_p}{\|x_k - v_j\|_p} \right)^{2/(m-1)} \right)^{-1}$;
(4) Compute the new cluster centers $v_i = \sum_{k=1}^{n} [A_i(x_k)]^m x_k / \sum_{k=1}^{n} [A_i(x_k)]^m$;
```

Algorithm 2: CM algorithm.

```
(1) Apply the KM clustering algorithm on data set $X$;
(2) Let $V$ be the set of final centers, which obtained from KM algorithm;
(3) Consider $V$ as a new data set;
(4) Apply the CM clustering algorithm on $V$;
(5) Recover corresponding data set clusters based on CM clustering output.
(6) End.
```

Algorithm 3: KCM algorithm.

accuracy is close to the CM. In the proposed technique, KM is initially applied to individual data objects to generate c clusters, designated as middle-level clusters. Each cluster is then represented by its centroid. The CM clustering is subsequently applied to those centroids in order to structure the final clustering. The distance between two middle-level clusters is measured as the distances between their centroids. The hybrid method considers the final centers produced by KM as the dataset for CM, so that the number of observations in the CM is equal to the number of centers produced by the KM method. Therefore, the KCM time is much less than the time of CM method.

The hybrid method is more suitable for the large dataset, where it has reduced clusters of observations by their centers, eventually computed from the KM. The performance of the proposed approach is evaluated by comparing it with KM and CM algorithms in terms of both accuracy and time processing. It is shown that the proposed technique outperforms CM in time processing; it yields results over shorter times when compared with the CM algorithm. Given a set of n elements $X = \{x_1, \ldots, x_n\}$, where $x_k \in R^d$, $k = 1, \ldots, n$, the steps of the KCM clustering algorithm are shown in Algorithm 3.

4. Evaluation

Simulated datasets are used to evaluate the KM, CM, and KCM clustering methods. We use an external clustering evaluation criterion for comparisons. The Rand index is a criterion used to compare an induced clustering structure (C_1) with a given clustering structure (C_2) defined as follows [13]:

$$\text{RAND} = \frac{a+d}{a+b+c+d}, \qquad (4)$$

where a, b, c, and d are the numbers defined as follows:

(i) a is the number of two points belonging to the same cluster, according to C_1 and C_2.

(ii) b is the number of points belonging to the same cluster according to C_1 but not C_2.

(iii) c is the number of points belonging to the same cluster according to C_2 but not C_1.

(iv) d is the number of points that do not belong to the same cluster, according to C_1 and C_2.

The quantities $\{a, d\}$ can be interpreted as agreements and $\{b, c\}$ as disagreements. The Rand index value lies within the range $[0, 1]$ and the clustering performance is considered to be good if the Rand index value converges to one [4, 13].

We used R 3.3.3 software, on a PC with CPU Core i5-3210 with 4 GB RAM to run all experiments in the next sections. For a fair comparison, termination condition of the algorithms is set as default of R standard codes.

5. Simulation Study

A d dimensional random vector Y has a sub-Gaussian distribution with location vector μ and dispersion matrix Q if its characteristic function is of the form

$$\varphi(u) = \exp\left(i \cdot u^T \mu\right) \exp\left(-\left|u^T Q u\right|^{\alpha/2}\right), \qquad (5)$$

where $u^T = (u_1, \ldots, u_d)$, $\mu \in R^d$, $i = \sqrt{-1}$, $\alpha \in [0, 2]$, and Q is a positive definite matrix. In the case of α equal to 2, we get the multivariate normal distribution that its covariance matrix is $2Q$ [19]. If $\alpha > 1$ then $\mu = E(X)$. However, the expectation of X does not exist for $\alpha \leq 1$.

In this simulation study, a set of real and simulated data generated by the sub-Gaussian and multivariate normal distributions was used. For clustering data using the proposed KCM method, the three Euclidean, Manhattan, and Minkowski distances were used. In addition, the results obtained from the KM, CM, and KCM algorithms were compared in terms of their time processing (in milliseconds) and accuracy. A set of data of 15000 observations having 30

Table 1: The time processing of CM versus KCM with Euclidean, Manhattan, and Minkowski ($p = 1.5$) distances when the number of clusters is 40.

α	Euclidean	Manhattan	Minkowski
0.5	KCM = 6.25% CM	KCM = 1.82% CM	KCM = 0.83% CM
1	KCM = 9.09% CM	KCM = 1.15% CM	KCM = 0.79% CM
1.5	KCM = 11.11% CM	KCM = 1.06% CM	KCM = 2.56% CM
2	KCM = 11.76% CM	KCM = 0.59% CM	KCM = 1% CM

attributes and parameter of stability in the range of $\alpha = 0.5, 1, 1.5, 2$ was generated, where if $\alpha = 2$, there will be a multivariate normal distribution. Then, the data were partitioned into 5, 10, 15, 20, 25, 30, and 40 clusters. As previously mentioned, our CM method is based on KM where c-value, which is the cluster number, is to be defined. The simulation results of the test are shown in Figures 1 and 2 showing the accuracy and time of KM, CM, and KCM method with Euclidean, Manhattan, and Minkowski ($p = 1.5$) distances for $\alpha = 0.5, 1, 1.5, 2$.

We have implemented the algorithms 100 times, and the average values of accuracy and time processes were computed. We classify the results as follows.

Time. In general, the time processes of KM were less than the time processes of CM and KCM algorithms for all values of α and the number of clusters, c. CM recorded a long-time process compared with either of the KM or KCM algorithms. The type of distance did not significantly affect the time processing of KM, where the results obtained with the three distances were close to one another. While the time processing of CM and KCM with Minkowski distance is longer than with Euclidean and Manhattan distance.

The increase in values of α does not affect the time processing of KM and KCM, where the values of time are almost close for all α, while the time processing of CM is decreasing with increasing value of α if we used the Euclidean distance, but if we used the Manhattan distance, it is increasing when the value of α increased. The speed of KM and CM is decreased if the number of clusters increases, but it does not affect much the speed of the KCM algorithm with Euclidean and Manhattan distances.

Generally, the processing time of the KCM algorithm is less than the CM algorithm. For example, when the number of clusters is 40, the processing time of CM and KCM is shown in Table 1.

Accuracy. Distance type had no significant effect on the accuracy of the KM and CM algorithms as almost the same results obtained with either. However, accuracy increased with $\alpha > 0.5$. The accuracy of KCM algorithm is increasing with increasing the values of α. In general, the accuracy of KCM and CM with Euclidean and Minkowski distances is better than that with Manhattan.

6. Comparison of Algorithms Using Real Data

In this section, the KM, CM, and KCM algorithms are tested for their performance using Iris (150 × 4), Wine (178 × 13), and Lens (24 × 4) datasets. The three Euclidean (Euc), Manhattan (Man), and Minkowski (Min) distance measures are used to see how they influence the overall clustering performance. The performance of these three techniques has been compared based on the following parameters:

(1) Precision = $T_P/(T_P + F_P)$.

(2) Accuracy = $(T_P + T_N)/(T_P + T_N + F_P + F_N)$.

A true positive (T_P) decision assigns two similar documents in the same cluster; a true negative (T_N) decision assigns two dissimilar documents to different clusters. A (F_P) decision assigns two dissimilar documents to the same cluster. A (F_N) decision assigns two similar documents to different clusters. The experimental results indicating the performance of each technique on the three datasets are reported in Table 2.

Using the Iris dataset led to a greater average precision of clusters formed by KCM-Euc and KCM-Man than those by KM and CM with the three distances. CM-Man recorded a greater accuracy than any of those formed by KM or KCM. Distance and algorithm type had no significant effect on the accuracy. As for the Lens dataset, average precision was generally low with all the algorithms examined. It, however, yielded acceptable accuracy values with the KM, CM, and KCM algorithms, but it does not exceed 0.50. With the Wine dataset, distance and algorithm type had a significant effect on the accuracy and average precision. The average of precision does not exceed 0.70 and the highest average recorded by KCM-Man. The CM-Man recorded a greater accuracy than any of those formed by KM or KCM.

7. Conclusions

In this paper, the two most famous clustering techniques, namely, K-Means and Fuzzy C-Means, were investigated for their performance. To improve the time processes of the fuzzy clustering technique, a hybrid algorithm, named KCM, combining the KM and CM algorithms, was proposed.

It was found that the KM algorithm had shorter time processes than CM and KCM algorithms for all values of α and c. In addition, the speed of CM was observed to be less than those of KM and KCM. However, the time processes of CM with the Euclidean and Manhattan distances were observed to be shorter than that with Minkowski distance. The value of α did not affect time processes under KM and KCM; however, that of CM decreased with increasing values of α with Euclidean but increased with Manhattan distance.

The accuracy of KM, CM, and KCM algorithms was increasing for $\alpha > 0.5$. Distance type had a significant

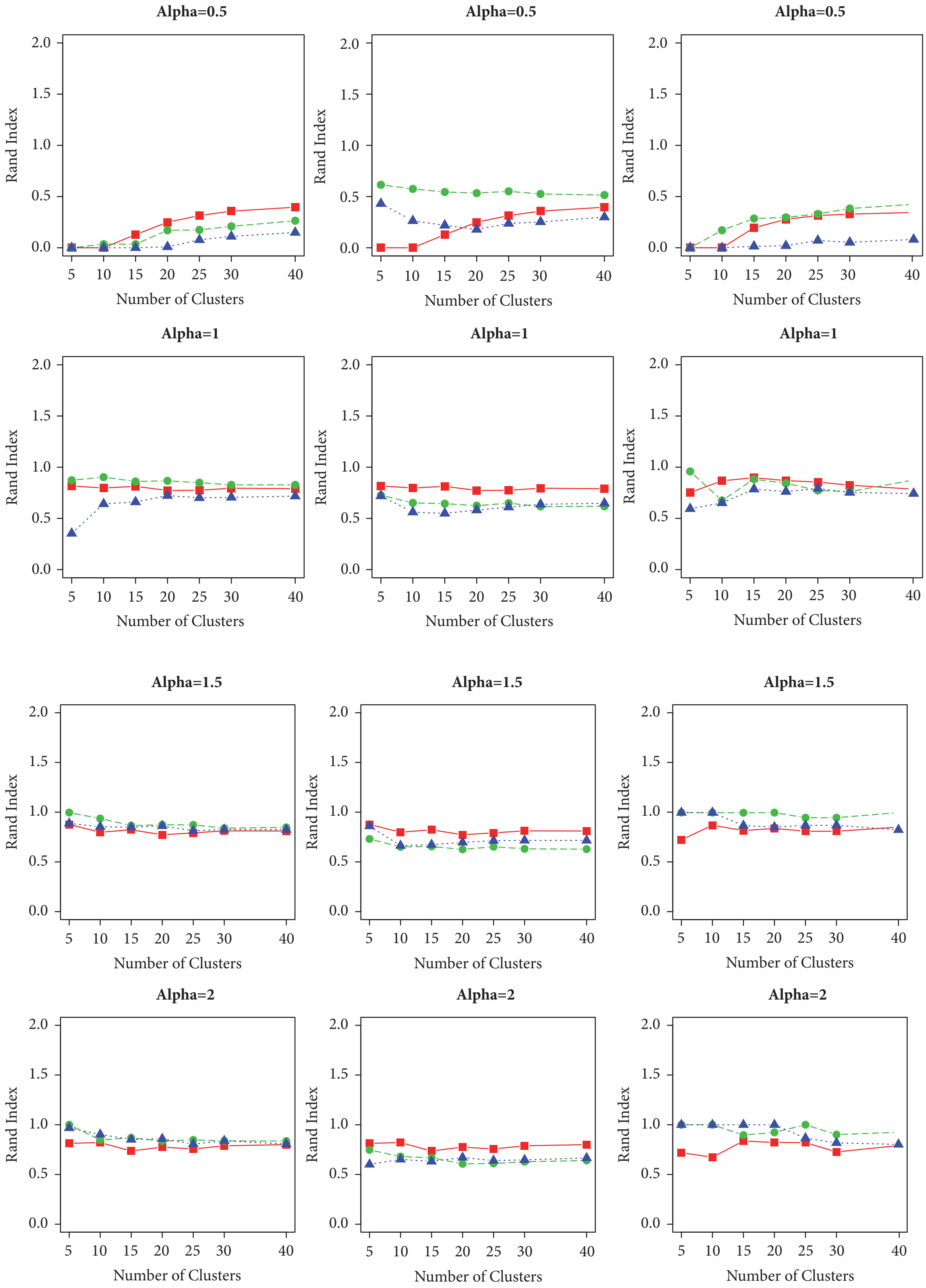


Figure 1: Comparison of KM, CM, and KCM algorithms in terms of accuracy based on Euclidean, Manhattan, and Minkowski ($p = 1.5$) distances for $\alpha = 0.5, 1, 1.5, 2$.

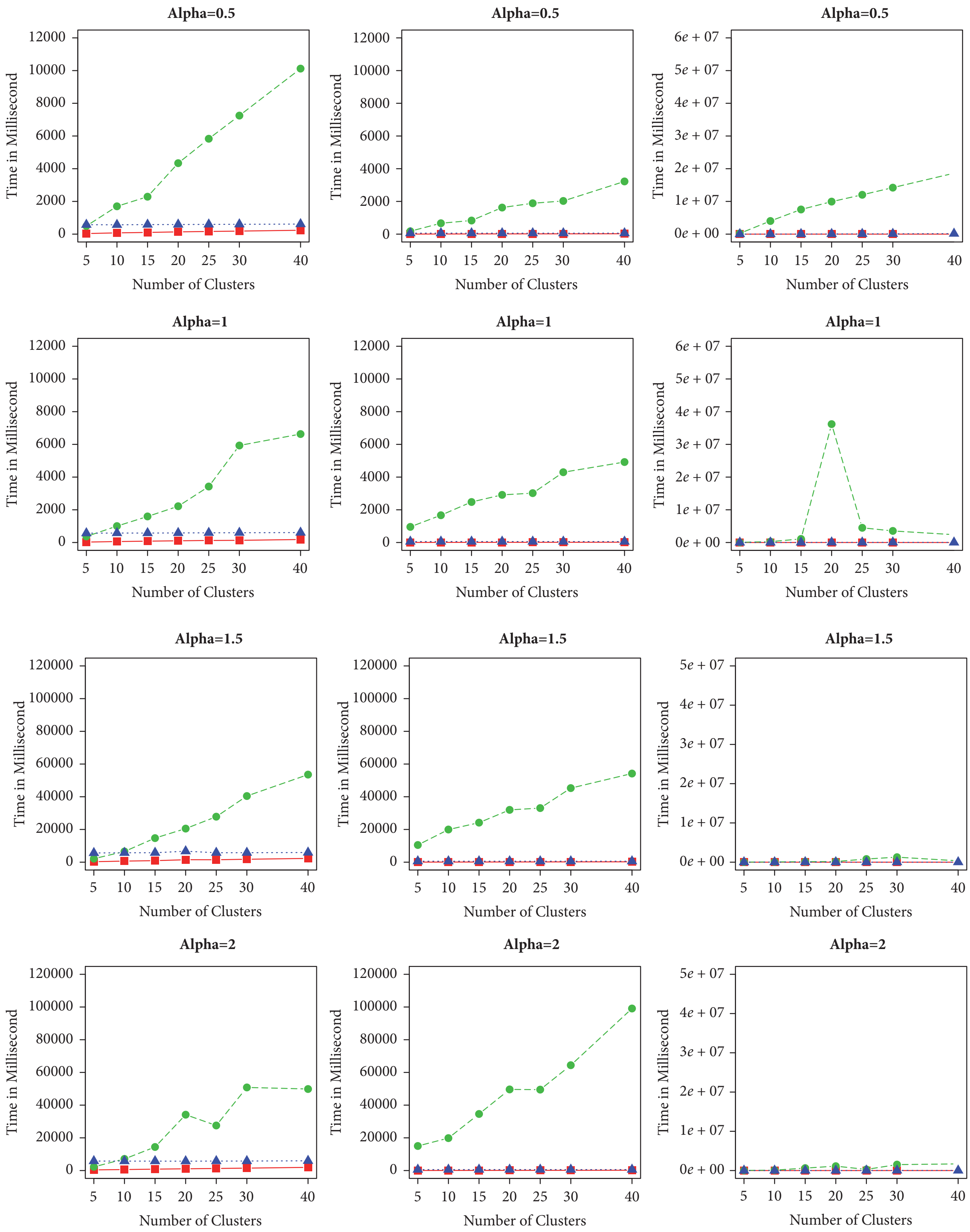


FIGURE 2: Comparison of KM, CM, and KCM algorithms in terms of time processing based on Euclidean, Manhattan, and Minkowski ($p = 1.5$) distances for $\alpha = 0.5, 1, 1.5, 2$.

Table 2: Performance analysis of KM, CM, and KCM clustering techniques on data and benchmarks Iris, Lens, and Wine datasets. Note that each data has 3 clusters; the best precision and accuracy for each data set is in bold font.

Dataset	Performance parameters	Clusters	*K*-Means			Fuzzy *C*-Means			*KC*-Means		
			Euc	Man	Min	Euc	Man	Min	Euc	Man	Min
Iris	Precision	1	0.898	0.898	0.898	0.858	0.905	0.861	0.627	0.627	1
		2	1	1	0.645	0.655	0.691	1	1	1	0.895
		3	0.644	0.645	1	1	1	0.670	1	1	0.631
		Average	0.847	0.847	0.848	0.837	0.865	0.844	**0.876**	**0.876**	0.842
		Accuracy	0.880	0.880	0.880	0.880	**0.899**	0.796	0.796	0.880	0.874
Lens	Precision	1	0.167	0.167	0.25	0.286	0.267	0.278	0.200	0.238	0.286
		2	0.273	0.200	0.238	0.200	0.250	0.25	0.167	0.286	0
		3	0.250	0.286	0.278	0.167	0.200	0.238	0.295	0	0.238
		Average	0.230	0.217	**0.255**	0.217	0.239	**0.255**	0.221	0.175	0.175
		Accuracy	0.522	0.507	**0.547**	0.547	0.533	**0.547**	0.504	0.504	0.504
Wine	Precision	1	0.356	0.517	0.957	0.345	0.858	0.957	0.360	0.504	1
		2	0.595	0.430	0.356	0.577	0.605	0.577	0.605	1	0.595
		3	0.957	1	0.595	0.956	0.386	0.345	0.957	0.556	0.338
		Average	0.636	0.649	0.636	0.626	0.616	0.626	0.641	**0.687**	0.644
		Accuracy	0.719	0.685	0.719	0.719	**0.735**	0.711	0.720	0.686	0.695

effect on the accuracy of KM and CM algorithms, but the accuracy of the KCM and CM algorithms with Euclidean and Minkowski distances was better than that with Manhattan distance.

Using the real datasets revealed that the Iris dataset yielded higher precision values for clusters with the three distances. The clusters formed by the combined KCM-Euc were observed to be more distinct. Using the Lens dataset led to poor precision levels but acceptable accuracy values for all the combinations. With the Wine dataset, medium precision levels were achieved with all the combinations. CM-Man and KCM-Euc yielded the most compact clusters, while KCM-Man yielded the most distinct ones. In general, the Iris dataset not only formed the most compact and distinct clusters, but also yielded higher precision and accuracy levels for KM, CM, and KCM clusters with the three distances than did the Lens or Wine datasets.

Finally, we recall that the time computation in a clustering method depends on the algorithm and its implementation, programming language, and hardware. Therefore, based on the complexity of the clustering problem one can consider the best of them.

Data Availability

The codes and data are available upon request.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

References

[1] A. C. Rencher, *Method of Multivariate Analysis*, John Wiley & Sons, 2nd edition, 2002.

[2] Z. Huang, "Extensions to the k-means algorithm for clustering large data sets with categorical values," *Data Mining and Knowledge Discovery*, vol. 2, no. 3, pp. 283–304, 1998.

[3] C.-T. Chang, J. Z. Lai, and M.-D. Jeng, "A fuzzy K-means clustering algorithm using cluster center displacement," *Journal of Information Science and Engineering*, vol. 27, no. 3, pp. 995–1009, 2011.

[4] T. Volmurgan, "Austria performance comparison between K-means and fuzzy C-means," *Wulfenia Journal*, vol. 19, pp. 234–241, 2012.

[5] T. C. Havens, J. C. Bezdek, C. Leckie, L. O. Hall, and M. Palaniswami, "Fuzzy C-means algorithms for very large data," *IEEE Transactions on Fuzzy Systems*, vol. 20, no. 6, pp. 1130–1146, 2012.

[6] B. Panda, S. Sahoo, and S. K. Patnaik, "A comparative study of hard and soft clustering using swarm optimization," *International Journal of Scientific & Engineering Research*, vol. 4, pp. 785–790, 2013.

[7] N. Grover, "A study of various fuzzy clustering algorithms," *International Journal of Engineering Research*, vol. 3, no. 3, pp. 177–181, 2014.

[8] D. J. Bora and D. A. Gupta, "A comparative study between fuzzy clustering algorithm and hard clustering algorithm," *International Journal of Computer Trends and Technology*, vol. 10, no. 2, pp. 108–113, 2014.

[9] M. Fajardo, A. McBratney, and B. Whelan, "Fuzzy clustering of Vis-NIR spectra for the objective recognition of soil morphological horizons in soil profiles," *Geoderma*, vol. 263, pp. 244–253, 2014.

[10] M. R. Anderberg, *Cluster Analysis for Applications*, Academic Press, New York, NY, USA, 1973.

[11] J. MacQueen, "Some methods for classification and analysis of multivariate observations," in *Proceedings of the 5th Berkeley Symposium on Mathematical Statistics and Probability*, vol. 1, pp. 281–297, 1967.

[12] A. Banharnsakun, "A MapReduce-based artificial bee colony for large-scale data clustering," *Pattern Recognition Letters*, vol. 93, pp. 78–84, 2017.

[13] G. Gan, C. Ma, and J. Wu, *Data Clustering Theory: Algorithms and Applications*, SIAM, Virginia, 2007.

[14] J. C. Dunn, "A fuzzy relative of the ISODATA process and its use in detecting compact well-separated clusters," *Journal of Cybernetics*, vol. 3, no. 3, pp. 32–57, 1973.

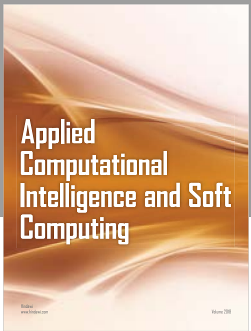
[15] J. C. Bezdek, *Pattern Recognition with Fuzzy Objective Function Algorithms*, Plenum Press, New York, NY, USA, 1981.

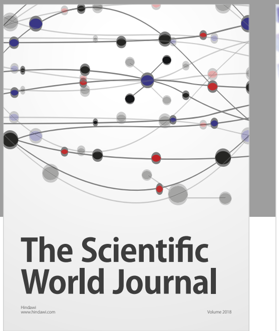
[16] G. J. Klir and B. Yuan, *Fuzzy Sets and Fuzzy Logic: Theory and Applications*, Prentice Hall, New York, NY, USA, 1995.

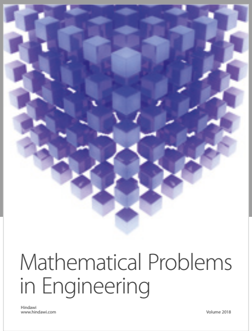
[17] R. Suganya and R. Shanthi, "Fuzzy C-means algorithm - a review," *International Journal of Scientific and Research Publications*, vol. 2, pp. 440–442, 2012.

[18] M. S. Yang, "Convergence properties of the generalized fuzzy C-means clustering algorithms," *Computers & Mathematics with Applications*, vol. 25, no. 9, pp. 3–11, 1993.

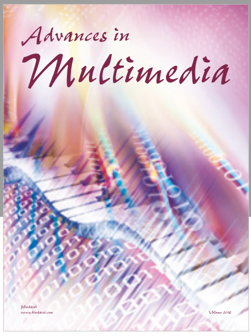
[19] V. Omelchenko, "Parameter estimation of sub-Gaussian stable distributions," *Kybernetika*, vol. 50, no. 6, pp. 929–949, 2014.

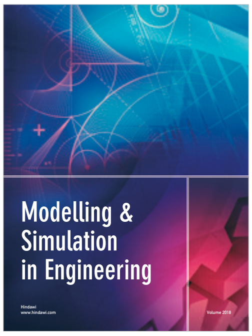

Applied Computational Intelligence and Soft Computing


The Scientific World Journal


Mathematical Problems in Engineering

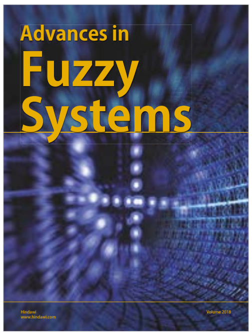
Journal of Engineering

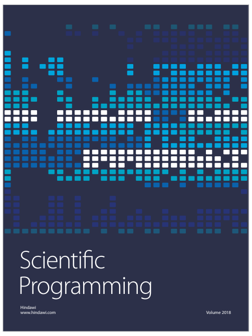

Advances in Multimedia

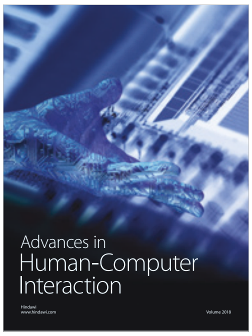

Modelling & Simulation in Engineering

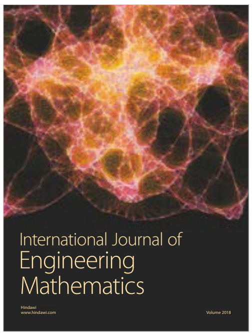
Advances in Artificial Intelligence

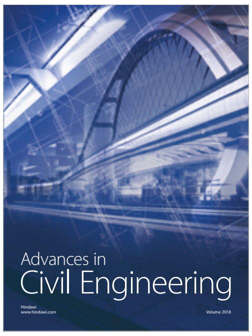
International Journal of Reconfigurable Computing

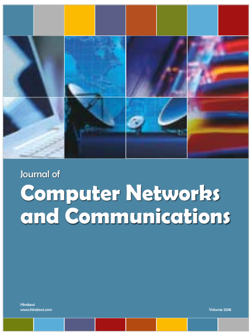

Advances in Fuzzy Systems


Scientific Programming


Advances in Human-Computer Interaction

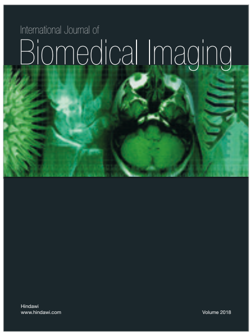

International Journal of Engineering Mathematics


Advances in Civil Engineering

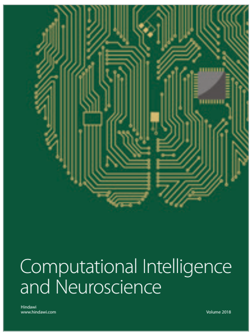

Journal of Computer Networks and Communications

Journal of Robotics

International Journal of Computer Games Technology


International Journal of Biomedical Imaging

Journal of Electrical and Computer Engineering


Computational Intelligence and Neuroscience