How to cite this paper:

Seman, A., & Sapawi, A. M. (2018). Extensions to the k-AMH algorithm for numerical clustering. *Journal if Information and Communication Technology*, *17* (4), 587-599.

EXTENSIONS TO THE K-AMH ALGORITHM FOR NUMERICAL CLUSTERING

Ali Seman & Azizian Mohd Sapawi
Faculty of Computer and Mathematical Sciences
Universiti Teknologi MARA, Malaysia

aliseman@tmsk.uitm.edu.my; azizian@tmsk.uitm.edu.my

ABSTRACT

The *k*-AMH algorithm has been proven efficient in clustering categorical datasets. It can also be used to cluster numerical values with minimum modification to the original algorithm. In this paper, we present two algorithms that extend the *k*-AMH algorithm to the clustering of numerical values. The original *k*-AMH algorithm for categorical values uses a simple matching dissimilarity measure, but for numerical values it uses Euclidean distance. The first extension to the *k*-AMH algorithm, denoted *k*-AMH Numeric I, enables it to cluster numerical values in a fashion similar to *k*-AMH for categorical data. The second extension, *k*-AMH Numeric II, adopts the cost function of the fuzzy *k*-Means algorithm together with Euclidean distance, and has demonstrated performance similar to that of *k*-AMH Numeric I. The clustering performance of the two algorithms was evaluated on six real-world datasets against a benchmark algorithm, the fuzzy *k*-Means algorithm. The results obtained indicate that the two algorithms are as efficient as the fuzzy *k*-Means algorithm when clustering numerical values. Further, on an ANOVA test, *k*-AMH Numeric I obtained the highest accuracy score of 0.69 for the six datasets combined with *p*-value less than 0.01, indicating a 95% confidence level. The experimental results prove that the *k*-AMH Numeric I and *k*-AMH Numeric II algorithms can be

Received: 3 April 2018 Accepted: 19 August 2018 Published: 1 October 2018

effectively used for numerical clustering. The significance of this study lies in that the *k*-AMH numeric algorithms have been demonstrated as potential solutions for clustering numerical objects.

Keywords: Cluster analysis, partitional clustering algorithms, categorical and numerical data mining.

INTRODUCTION

Conventionally, clustering algorithms may fall under either of two clustering approaches: hierarchical approach or partitional approach. In the hierarchical approach, objects are arranged in a multi-level overlapping hierarchical form, whereas in the partitional approach, objects are assigned to a one-level non-overlapping partitioning (Gan et al., 2007). For clustering large and high-dimensional datasets, partitional clustering algorithms are generally more efficient than hierarchical clustering algorithms (Gan et al., 2007). Several well-established partitional clustering algorithms based on the center of a cluster, also known as center-based clustering algorithms (Gan et al., 2007) exist. They include the *k*-Means algorithm (MacQueen, 1967), which uses mean center clusters; the *k*-Modes algorithm (Huang, 1998), which uses mode center clusters; the *k*-Medoid algorithm (Kaufman & Rousseeuw, 1987), which uses object center clusters; and the *k*-Median algorithm (Meyerson et al., 2004), which uses median center clusters.

Researchers became interested in clustering categorical data 20 years ago, when Huang (1998) proposed the *k*-Modes algorithm to specifically handle categorical variables. The *k*-Modes algorithm is based on the k-Means algorithm, established for clustering numerical variables. It leverages the framework of the k-Means algorithm by exchanging mean with mode as the center of clusters and the simple similarity measure for calculating categorical variables, instead of the Euclidean distance of numerical variables. Variants that improve on these two seminal clustering algorithms (k-Means and k-Modes) have been proposed over the years. For example, k-Modes variants have been proposed for dealing with set-valued features (Cao et. al., 2017a) and matrix-object data (Cao et. al., 2017b). Similarly, bi-level and tri-level k-Means algorithms have been introduced to overcome the common issues of outliers and noisy data by the k-Means algorithm (Yu et. al., 2017). In addition, density canopy has been incorporated into the algorithm for determining the appropriate value for *k*, clusters, and initial cluster centers (Zhang, 2018).

Fuzzy clustering algorithms are inspired by the concept of fuzzy sets, introduced by Zadeh (1965). The fuzzy *k*-Means (also known as fuzzy

c-Means) algorithm, introduced by Bezdek (1981), gained popularity as the pioneer fuzzy-based clustering algorithm. Several other fuzzy-based algorithms exist for solving clustering problems, e.g., the fuzzy Covariance Clustering (Gustafson & Kessel, 1978) and fuzzy *c*-Elliptotypes (Bezdek, 1981) algorithms for numerical problems, and the fuzzy *k*-Modes (Huang & Ng, 1999), *k*-Population (Kim et al., 2005), and new fuzzy *k*-Modes (Ng & Jing, 2009) algorithms for categorical problems.

Recently, a new algorithm called the *k*-Approximate Modal Haplotype (*k*-AMH) algorithm, which manipulates objects as the center of clusters, was exclusively introduced for clustering categorical values, particularly the DNA datasets of Y-Short Tandem Repeats (Y-STR) (Seman et al., 2012a). The *k*-AMH algorithm comes with its own medoid mechanism as the center of its clusters and has proven to be efficient in clustering Y-STR (Seman et al., 2012b) and other categorical data, such as soybean and voting (Seman et al., 2013). The algorithm relies on the maximization of its cost function and employs the fuzzy clustering framework as incorporated by the fuzzy *k*-means-type and fuzzy *k*-modes-type algorithms. This clustering framework still leaves the *k*-AMH algorithm open for further extension and improvement; for example, *k*-AMH I, II, and III were extended to improve the previous Y-STR clustering results (Seman et al., 2015).

This study presents two extension algorithms aimed at generalizing the *k*-AMH algorithm for clustering numerical values: (1) the original *k*-AMH algorithm extended directly to cluster numerical values by using Euclidean distance and (2) another *k*-AMH algorithm with a new cost function, adapted from the fuzzy *k*-Means algorithm cost function, combined with Euclidean distance. Thus, in the ensuing sections, the fundamental features of the *k*-AMH algorithm is first described, followed by descriptions of the two proposed algorithms for clustering numerical values. Finally, the performances of these two algorithms versus that of the fuzzy *k*-means algorithm on six real-world datasets are discussed.

k-AMH ALGORITHM

Let $X = \{X_1, X_2, ..., X_n\}$ be a set of n objects with categorical values and $H = \{H_1, H_2, ..., H_k\}$ be a set of objects as the medoid of clusters. The aim is to find k clusters from n objects; the algorithm selects an object, h, for each cluster. Each x is tested and replaced in succession to obtain h via the maximization processes of $P(W, D)$, as described in Eq. (1).

$$P(W,D)^r > P(W,D)^t, r \neq t; \forall t, 1 \leq t \leq (n-k) \quad (1)$$

$P(W, D)$ is maximized and defined in Eq. (2):

$$P(W,D) = \sum_{l=1}^{k}\sum_{i=1}^{n} w_{li}^{\alpha} d_{li} \tag{2}$$

where $w_{li}^{\alpha} \in W$ is a $(k \times n)$ matrix that describes the degree of fuzziness values of the object, which contains values of zero and one, as described in Eq. (3).

$$w_{li}^{\alpha} = \begin{cases} 1 & X_i = H_l \\ 0 & X_i = H_z, z \neq i \\ \left[\sum_{z=1}^{k} \frac{d(X_i, H_i)}{d(X_i, H_z)}\right]^{\frac{-\alpha}{\alpha-1}} & Otherwise \end{cases} \tag{3}$$

where k ($\leq n$) is a predefined number of clusters, H is the medoid such that $H = \{H_1, H_2, ..., H_k\} \in X$, a $\in [1, \infty)$ is an alpha value that is typically greater than 1.0 and $d(X_i, H_z)$ is the distance calculated between object X_i and medoid H_z, as described in Eq. (4) and (4a):

$$d(X,H) = \sum_{j=1}^{m} \gamma\left(x_j, h_j\right) \tag{4}$$

subject to

$$\gamma(x_j, h_j) = \begin{cases} 0, & x_j = h_j \\ 1, & x_j \neq h_j \end{cases} \tag{4a}$$

where m is the number of attributes.

$d_{li} \in D$, a is a $(k \times n)$ matrix that assigns a value of 1.0 or 0.5, known as the dominant weight, described in Eq. (5), subject to Eq. (5a), and (5b):

$$d_{li} = \begin{cases} 1.0, & if\ w_{li}^{\alpha} = \max^{w_{li}^{\alpha}, 1 \leq l \leq k} \\ 0.5, & otherwise \end{cases} \tag{5}$$

subject to

$$1.5 \leq \sum_{l=1}^{k} d_l \ \leq_i k, 1 \leq i \leq n \tag{5a}$$

$$0.5 < \sum_{i=1}^{n} d_{li} < n, 1 \le l \le k \quad (5b)$$

The optimization steps of the *k*-AMH algorithm are as follows.

Step 1: Select the initial medoid, $H^{(1)} \in X$, randomly. Calculate $P(W, D)$. Set $q=1$.
Step 2: Select $X^{(t+1)}$ such that $P(W, D)^{t+1}$ is maximized to replace $H^{(1)} \leftarrow X^{(t+1)}$.
Step 3: Set $q = q+1$. If $q = n$, stop; otherwise, go to Step 2.

The algorithm above can be further simplified to the following steps.

Step 1: Choose *k* initial objects randomly as medoids.
Step 2: Calculate distance $d(X_i, H_z)$ between object *x* and medoid *h* using Eq. (4) and subject to Eq. (4a).
Step 3: Based on the distance calculated in Step 2, calculate partition matrix w_{li} using Eq. (3).
Step 4: Based on the partition matrix calculated in Step 3, assign a weighting dominant of 1.0 or 0.5 using Eq. (5) and subject to Eq. (5a) and (5b).
Step 5: Calculate cost function $P(W, D)$ using Eq. (2).
Step 6: If the current cost function (Step 5) is greater than the previous cost function, replace medoid *h* for each *x* until the final medoids are obtained for all clusters using Eq. (1).
Step 7: Assign the objects to their corresponding crisp clusters.

EXTENDED *k*-AMH ALGORITHMS FOR NUMERICAL VALUES

k-AMH Numeric I

This algorithm is based precisely on the original *k*-AMH algorithm. However, Euclidean distance is used instead of similarity measure for the clustering of numerical values, as described in Eq. (4) and (4a).

$$d_{\text{euc}}(x, y) = \left[\sum_{j=1}^{d} (x_j - y_j)^2\right]^{\frac{1}{2}} \quad (6)$$

Therefore, the *k*-AMH Numeric I uses the clustering procedures of *k*-AMH algorithm as described above.

k-AMH Numeric II

This algorithm was introduced with a minor modification to the original cost function of the *k*-AMH algorithm. The cost function employed by the fuzzy *k*-Means algorithm (Bezdek, 1981) is used in this algorithm to iteratively replace objects in succession towards *h*. Therefore, the replacement is based on the minimization of the cost function, *P*(*W, D*) and as described in Eq. (7) and (8), and Euclidean distance as the distance measure (Eq. (6)).

$$P(W,D)^{\mathrm{r}} < P(W,D)^{\mathrm{t}}, r \neq t; \forall t, 1 \leq t \leq (n-k) \tag{7}$$

P(*W, D*) is a cost function, as described above in Eq. (2) and expanded in Eq. (8):

$$P(W,D) = \sum_{l=1}^{k} \sum_{i=1}^{n} w_{li}^{\propto} \, d(X_i, D_l) \tag{8}$$

subject to Eq. (3) and (6).

RESULTS AND DISCUSSION

In this section, we discuss the experimental results obtained and compare the clustering performance of each of the two *k*-AMH numeric algorithms to that of the well-established fuzzy *k*-Means algorithm.

Clustering Performance

We used the external criterion employed by Huang (1998) to evaluate his algorithm, the *k*-Modes algorithm, in evaluating the clustering performance of the proposed algorithms. The external criterion is one of three types of criteria that can be used in discovering inherent data structures of clustering results (Jain & Dubes, 1988). This criterion measures the degree of correspondence between the clusters and the classes assigned a priori.

The accuracy scores were manually obtained using the misclassification matrix Huang (1998) employed to analyze the correspondence between the clusters and the classes of the instances. This method was mainly used to measure the performance of clustering algorithms, as described by (9):

$$r = \frac{\sum_{i=1}^{k} g_i}{n} \tag{9}$$

where k is the number of clusters, g_i is the number of objects occurring in both cluster i and its corresponding class/cluster, and n is the number of objects.

In addition, the experiments were conducted for each algorithm and dataset based on the alpha values used by them, i.e., 1.1, 1.2, 1.3, 1.4, 1.5, 1.6, 1.7, 1.8, 1.9, and 2.0. To benchmark the performances of all algorithms, we rigorously conducted a 100-run experiment for each algorithm, dataset, and alpha value. Consequently, we obtained 18,000 accuracy scores (= 3 algorithms × 6 datasets × 100-run experiment × 10 alpha values).

Real-world Datasets

To benchmark the results, six numerical datasets were used to evaluate the performance of the algorithms; specifically, the Haberman, Pima, Wine, Seed, Iris, and User Knowledge datasets from the UCI repository (Lichman, 2013). Table 1 gives a summary of all the datasets and presents the number of objects, classes, and attributes for each set.

Table 1

Summary of Numerical Datasets

Dataset	Description	No. of Instances	No. of Classes	No. of Attributes
1. Haberman	Haberman's survival dataset used for breast cancer studies. The original dataset contained 306 items with no filtration, divided into two classes with three attributes.	306	2	3
2. Pima	Pima Indians Diabetes Data Set donated by National Institute of Diabetes and Digestive and Kidney Diseases. The original dataset contained 768 items, but was filtered to 393 by excluding all missing values. The number of classes and attributes are three and eight, respectively.	393	3	8
3. Wine	The Wine dataset used for chemical analysis of wines grown in a specific area of Italy. The original dataset contained 178 items with no filtration, divided into three classes with thirteen attributes.	178	3	13

(continued)

Dataset	Description	No. of Instances	No. of Classes	No. of Attributes
4. Seed	The Seed dataset used for comparing three different varieties of wheat: Kama, Rosa, and Canadian. The original dataset contained 210 items with no filtration, divided into three classes with seven attributes.	210	3	7
5. Iris	The Iris dataset used for analyzing the three types of Iris plants. The original dataset contained 150 items with no filtration, divided into three classes with four attributes.	150	3	4
6. User knowledge	The Users' knowledge dataset used for studying students' knowledge status about electrical DC machines. The dataset was filtered to 255 by excluding zero values from the 258-item training set. The number of classes and attributes are four and five, respectively.	255	4	5

Clustering Results

The performance of the clustering algorithms can be observed by comparing the accuracy scores recorded through the 100-run experiments for each algorithm and dataset. Figure 1 shows box plot diagrams plotted from 1000 accuracy scores for each algorithm and dataset through their alpha values. At a glance, it is clear that the two *k*-AMH numeric algorithms are competitive. In general, for each dataset, the performances of the *k*-AMH numeric algorithms are at least on par with the performances of the fuzzy *k*-Means algorithm. In fact, *k*-AMH Numeric I obviously outperforms the fuzzy *k*-Means algorithm on dataset 1, whereas *k*-AMH Numeric II demonstrates its competitiveness on datasets 3, 4, and 5. However, there is no algorithm is able to discover the inherent grouping structures consistently in all datasets. This is because the datasets used for the experiments are real datasets.

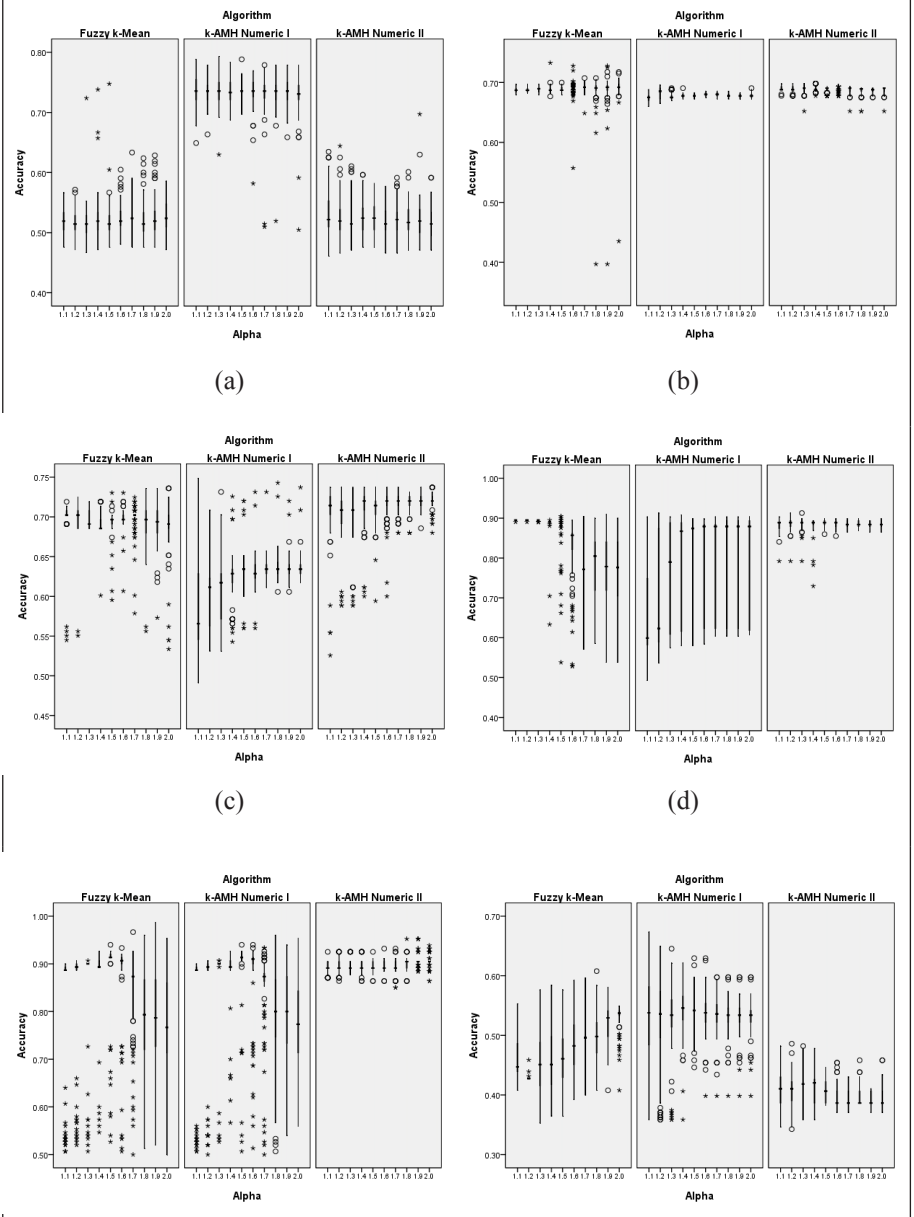

Figure 1. Overall observation through box plots based on clustering accuracy scores for each algorithm and alpha value: (a) Haberman dataset, (b) Pima dataset, (c) Wine dataset, (d) Seed dataset, (e) Iris dataset, (f) User Knowledge dataset.

Figure 2 shows box plots of the accuracy scores of the fuzzy k-Means, k-AMH Numeric I, and k-AMH Numeric II algorithms plotted based on the combined accuracy scores of all datasets and alpha values. In general, there are no obvious differences among the three box plots. The box plots are comparatively tall, which suggests that the overall accuracy scores are optimal. Furthermore, the medians, which are generally close to the average accuracy scores, are all considerably at the same level. The average accuracy scores recorded were 0.675, 0.694, and 0.685, whereas the minimum scores were 0.35, 0.36, and 0.34 and the maximum were 0.99, 0.96, and 0.95 for fuzzy k-Means, k-AMH Numeric I, and k-AMH Numeric II, respectively.

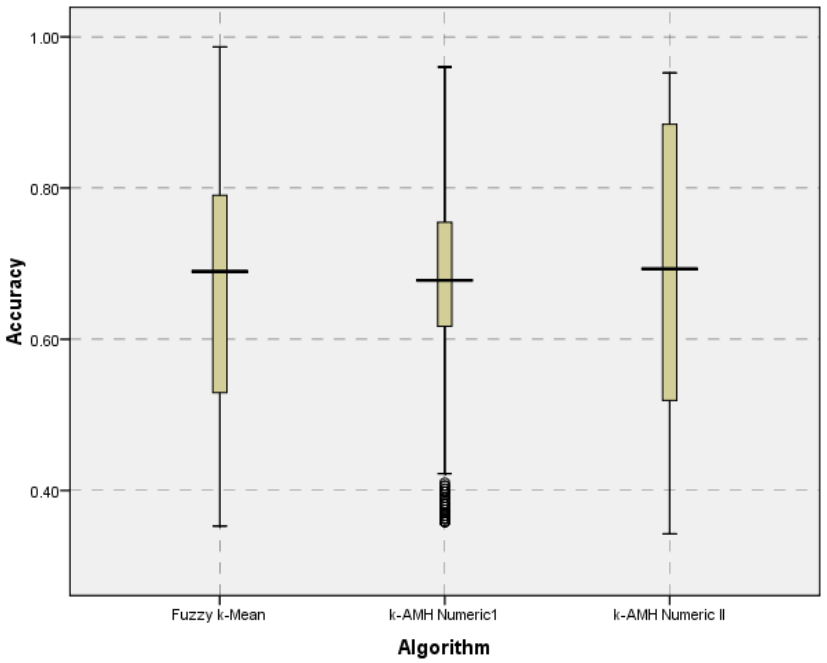

Figure 2. Box plot showing the accuracy scores on the six datasets combined.

For further analysis, a one-way ANOVA test was also used to elucidate whether any differences exist among the three algorithms. The assumption of homogeneity of variance was violated; therefore, the Welch *F*-ratio was reported, in which F (2, 11, 732.9) = 17,997, $p < 0.01$, $\omega^2 < 0.01$. Thus, the comparison of the three algorithms was based on the Games-Howell Post Hoc test. Table 2 shows the comparison results obtained for the Games-Howell Post Hoc test. At a five percent level of significance, the *k*-AMH Numeric I

algorithm (M = 0.694, 95% CI [0.691, 0.697]) is significantly different from the average accuracy scores for both the fuzzy *k*-Means and the *k*-AMH Numeric II algorithms, with p-value < 0.01. Furthermore, the average accuracy score of *k*-AMH Numeric II (M = 0.685, 95% CI [0.680, 0.689]) is also significantly different from that of the fuzzy *k*-Means algorithm (p-value = 0.01). In conclusion, the *k*-AMH Numeric I and II algorithms are marginally better that than the fuzzy *k*-Means algorithm.

Table 2

Multiple Comparisons of the Fuzzy k-Means, k-AMH Numeric I, and k-AMH Numeric II Algorithms for Combined Datasets

Average accuracy Games-Howell Post Hoc Test						
(I) Algorithm	(J) Algorithm	Mean (I-J)	Std. Error	*p-value*	95% Confidence Level	
					Lower Bound	Upper Bound
Fuzzy *k*-Means	*k*-AMH Numeric I	-0.02*	0.003	**<0. 01**	-0.03	0.01
	k-AMH Numeric II	0.00*	0.003	**0. 01**	-0.02	0.00
k-AMH Numeric I	Fuzzy *k*-Mean	0.02*	0.003	**<0.01**	0.01	0.02
	k-AMH Numeric II	0.01*	0.003	**0.04**	0.02	0.02

***Note:** $p < 0.05$.

CONCLUSION

The *k*-AMH algorithm has already been proven efficient for the clustering of categorical data. Moreover, it can also be generalized and extended to cluster numerical data because it uses a clustering framework similar to the one incorporated by the fuzzy *k*-Means algorithm. The main difference is simply that it uses medoid-based cluster centers instead of the centroid used by the *k*-Means algorithm. From the rigorous experiments conducted on the six real-world datasets, the performances of both *k*-AMH numeric algorithms are very promising. The *k*-AMH Numeric algorithms are considerably efficient in clustering numerical objects and their performances are at least on par with that of the well-established fuzzy *k*-Means algorithm. Furthermore, on certain datasets—specifically, datasets 1, 2, 4, and 5—the two algorithms obviously outperform the fuzzy *k*-Means algorithm. The results presented via box plots demonstrate their efficiency—specifically in terms of higher accuracy scores

and median values—as well as their consistency, specifically showing fewer outliers and shorter box plots. Hence, the *k*-AMH numeric algorithms can be viewed as potential solutions for clustering numerical objects.

ACKNOWLEDGMENT

This study was supported by the Fundamental Research Grant Scheme, Ministry of Higher Education, Malaysia (Reference No.: 600-RMI/FRGS 5/3 (37/2104)) and the Institute of Research Management and Innovation, Universiti Teknologi MARA.

REFERENCES

Bezdek, J. C. (1981). *Pattern recognition with Fuzzy objective function algorithms*. Norwell, MA: Kluwer Academic Publishers.

Cao, F., Huang J. Z., Liang, J., Zhao, X., Meng, Y., Feng, K., & Qian, Y. (2017a). An algorithm for clustering categorical data with set-valued features. *IEEE Transactions on Neural Networks and Learning Systems,* 99, 1–14.

Cao, F., Yu, L., Huang, J. Z., & Liang, J. (2017b). *k*-mw-modes: An algorithm for clustering categorical matrix-object data. *Applied Soft Computing Journal*, 57, 605–614.

Gan, G., Ma, C., & Wu, J. (2007). *Data clustering: Theory, algorithms, and applications*. Society for Industrial and Applied Mathematics (SIAM).

Gustafson, D. E., & Kessel, W. C. (1978). Fuzzy clustering with a Fuzzy covariance matrix. *IEEE Conference on Decision and Control including the 17th Symposium on Adaptive Processes*, *17*, 761–766.

Huang, Z. (1998). Extensions to the k-Means algorithm for clustering large datasets with categorical values. *Data Mining and Knowledge Discovery*, *2*, 283–304.

Huang, Z., & Ng, M. (1999). A Fuzzy k-Modes algorithm for clustering categorical data. *IEEE Transactions on Fuzzy Systems*, *7*, 446–452.

Jain, A. K., & Dubes, R. C. (1988). *Algorithm for clustering data*. New Jersey: Prentice-Hall Inc.

Kaufman, J., & Rousseeuw, P. J. (1987). Clustering by means of Medoid. In Y. Dodge (Ed.), *Statistical data analysis based on the L1 norm*. Elsevier/ North-Holland, Amsterdam, (pp. 405–416).

Kim, D. W., Lee, Y., K., Lee, D., & Lee, K. H. (2005). k-Populations algorithm for clustering categorical data. *Pattern Recognition*, *38*, 1131–1134.

Lichman, M. (2013). *UCI machine learning repository*. University of California, School of Information and Computer Science, Irvine, CA.

MacQueen, J. B. (1967). Some methods for classification and analysis of multivariate observations. *The 5th Berkeley Symposium on Mathematical Statistics and Probability*, *1*, 281–297.

Meyerson, A., O'callaghan, L., & Plotkin, S. (2004). A k-Median algorithm with running time independent of data size. *Machine Learning*, *56*, 61–87.

Ng, M. K., & Jing, L. (2009). A new Fuzzy k-Modes clustering algorithm for categorical data. *International Journal of Granular Computing, Rough Sets and Intelligent Systems*, *1*(1), 105–119.

Seman, A., Bakar, Z. A., & Isa, M. N. (2012). An efficient clustering algorithm for partitioning y-short tandem repeats data. *BMC Res Notes*, *5*, 557.

Seman, A., Bakar, Z. A., & Isa, M. N. (2012). Evaluation of k-Modes-Type algorithms for clustering y-short tandem repeats data. *Trends in Bioinformatics*, *5*, 47–52.

Seman, A., Bakar, Z. A., Sapawi, A. M., & Othman I. R. (2013). A medoid-based method for clustering categorical data. *Journal of Artificial Intelligence*, *6,* 257–265.

Seman, A., Sapawi A. M., & Salleh, M. Z. (2015). Towards development of clustering applications for large-scale comparative genotyping and kinship analysis using Y-short tandem repeats. *Journal of Integrative Biology*, *19*, 361–367.

Yu, S. -S., Chu, S. -W., Wang, C. -M., Chan, Y. -K., & Chang, T. -C. (2017). Two improved *k*-means algorithms. *Applied Soft Computing Journal*, doi: https://doi.org/10.1016/j.asoc.2017.08.032

Zadeh, L. (1965). Fuzzy sets. *Information and Control*, *8*(3), 338–353.

Zhang, G., Zhang, C., & Zhang, H. (2018). Improved *k*-means algorithm based on density canopy. *Knowledge-Based Systems*, 1–9.