African Journal of Mathematics and Statistics Studies
ISSN: 2689-5323
Volume 3, Issue 1, 2020 (pp. 68-78)
www.abjournals.org

SOME VERSIONS OF K-MEANS CLUSTERING METHOD AND ITS COMPARATIVE STUDY IN LOW AND HIGH DIMENSIONAL DATA

Oti Eric Uchenna[1] and Onyeagu Sidney Iheanyi[2]

[1]Department of Statistics, School of Applied Sciences, Federal Polytechnic, Ekowe, Bayelsa State, Nigeria, Tel: +2348037979262

[2]Department of Statistics, Faculty of Physical Sciences, Nnamdi Azikiwe University, Awka, Anambra State, Nigeria

ABSTRACT: *In this paper we present some versions of k-means clustering method and compare the methods using simulated data, and also low and high dimensional data set in terms of their accuracy and minimized total intra-cluster variance. The versions of k-means clustering method discussed in this paper are namely: The Forgy's method, Lloyd's method, MacQueen's method, Hartigan and Wong's method, Likas' method and Faber's method. These methods minimize a given criterion by iteratively relocating points between clusters until a locally optimal partition is attained. In a basic iterative algorithm, such as k-means, convergence is local and the globally optimum solution cannot be guaranteed. From experimental results, it was observed that Likas' method and Faber's method performed better in our synthetic data; method like Likas' performed better in low dimensional data (iris data) while Hartigan and Wong's method did better in high dimensional data (yeast cell cycle data).*

KEYWORDS: Clustering, Cluster Centroids, Euclidean Distance, Minimum Distance Rule, Versions of K-means Clustering Method

INTRODUCTION

Data clustering (or just clustering), also called cluster analysis, segmentation analysis, taxonomy analysis, or unsupervised classification, is a method of creating groups of data points, or clusters, in such a way that data points in one cluster are very similar to each other and data points in different clusters are quite distinct (Anderberg, 1973; Hartigan, 1975; Gan et al., 2007; Everitt et al., 2011; Mirkin, 2013).

Clustering methods gained widespread acceptance in the sciences, and motivated world-wide research on clustering methods when Sokal and Sneath's (1963) made a publication of Principles of Numerical Taxonomy. The clustering method (cluster analysis) which is usually performed under a condition known as unsupervised learning is different from supervised classification (discriminant analysis). In supervised classification, observations are allocated to a known number of predefined groups, while in unsupervised learning; neither the number of groups nor the groups themselves are known in advance. Humans quite often perform clustering most of the time without noticing. For instance, sorting of documents into folders normally involves grouping similar ones together. Although, clustering as a method that is widely used in many disciplines is not without flaws, so Xu and Wunsch (2009) contributed that the inexactitude begins with the fact that there is no universally agreed and precise definition of the term cluster, so it then becomes subjective to the problem. Some scholars

argue that this lack of a key definition is the main reason why there are so many clustering algorithms today (Estivill-Castro, 2002).

Clustering methods can be divided into two main groups which are based on the structure of their output namely: hierarchical and non-hierarchical clustering methods. Hierarchical clustering methods produce a sequence of the sets of the clusters. The clusters are merged (agglomerative methods) or split (divisive methods) step-by-step based on the applied similarity measure. The results of a hierarchical clustering method entail that agglomerative and divisive methods can be displayed graphically using a tree diagram known as dendrogram. The dendrogram shows all the steps in the hierarchical procedure which includes the similarities or distances at which clusters are merged. While non-hierarchical or partitioning clustering methods partition the data, object set into clusters where every pair of object clusters is either distinct (exhaustive, non-overlapping, crisp, or hard clustering) or has some members in common (Non-exhaustive, overlapping, or soft clustering). Partitioning clustering begins with a starting cluster partition which is iteratively improved until a locally optimal partition is reached. Amongst the partitioning clustering methods, the k-means method is the most popularly and commonly used in practice. K-means clustering is used to divide a set of objects (items, cases, entities, or data points) into k subsets or clusters (partitions, classes, or groups). The separated clusters are disjoint and the members in a cluster are sufficiently similar or close to each other and are sufficiently distant or dissimilar to non-members in other clusters (Forgy, 1965; MacQueen, 1967; Hartigan and Wong 1979; Lloyd, 1982).

METHODS

There are six versions of k-means clustering method considered in this paper and each version aim to classify points or objects to be analyzed into well separated groups or clusters. The six versions of k-means clustering method are: Forgy's method, Lloyd's method, MacQueen's method, Hartigan and Wong's method, Likas' method and Faber's method. These six versions were chosen because they are the most widely used k-means clustering method and they all use Euclidean distance as their metrics.

To be able to use any of the six versions, you first need to know how many clusters are present in your data; multiple runs or trials will be necessary to find the best number of clusters. These six versions of k-means clustering method have two phases of iteration namely: the assignment or initialization phase which involves an iterative process where each data point is assigned to its nearest centroid using any metric of choice; the next is the centroid update phase, where clusters centroids are updated given the partition obtained by the previous phase. The iterative process stops when no data point change clusters or some maximum number of iterations is reached.

Forgy's Method

Forgy (1967) proposed a batch algorithm which is also an offline centroid clustering model, seldom known referred to as traditional k-means algorithm. The algorithm is based on the minimization of the average squared Euclidean distance between the data points and the cluster's center known as centroid. A centroid is the center of a geometric object and it is seen as a generalization of the mean. Batch algorithm is an algorithm where a transformative

step is applied to all data-point (case) at once, where c is the cluster center in the Euclidean distance and x is the case it is compared to, i is the dimension of $x(or\ c)$ being compared and k is the total number of dimensions. That is,

$$d_{euc} = \sqrt{\sum_{i=1}^{k}(c_i - x_i)^2} \quad (1)$$

being the most common distance. Forgy's method starts (begins) with the choosing of k instance or initialization of data set uniformly at random and assigns the rest of the data points to the closest cluster (Peña et al,1999). It also treats the data set as a continuous distribution. Let a given data set be that$\{x_1, x_2, \ldots, x_n\} \epsilon R^d$, where R^d is the real d-dimensional data space (or the Euclidean d-dimensional data space), the algorithm tries to find a set of k cluster centers,$c = \{c_1, c_2, \ldots, c_k\} \epsilon R^d$, which is a solution to the minimization problem. The error function for a continuous distribution is defined as

$$E = \sum_{i=1}^{k} \int f(x) d\left(c_i, x_{ij}\right) dx \quad (2)$$

In the above equation, $\sum_{i=1}^{k} is$ a summation sign,$\int is$ an integral sign, $f(x)$ is the probability density function and $d\left(c_i, x_{ij}\right)$ is the distance function. K-means method can converge to a local optimum; Different initial points will conclusively lead to different convergence of centroid; hence, it is appropriate to start with a reasonable initial partition in order to achieve high quality clustering solution. However, there is no efficient and universal technique for obtaining such initial partitions theoretically. The Forgy's method has a major drawback which lies with choosing an outlier as an initial cluster center; it is possible that no other data point is assigned to it, and hence the cluster with the outlier as its center remains empty.

Algorithm 1: The Forgy's (Traditional) Algorithm.

1. Begin with any desired initial configuration. Go to step 2 if beginning with a set of seed points; go to step 3 if beginning with a partition of the data units.
2. Assign each data unit to the cluster with the nearest seed point. The seed point's remains fixed for a full cycle through the entire data set.
3. Compute new seed points as the centroids of the cluster of the data units.
4. Repeat step 2 and 3 until the process converges; that is, continue until no data units changes their cluster membership at step 2.

Lloyd's Method

Lloyd (1982) proposed a method that is widely known as the standard k-means algorithm; it is also a batch algorithm that is based on the minimization of the average squared Euclidean distance between the data items and the cluster's center like the Forgy's method. The dissimilarity between the Lloyd algorithm and the Forgy algorithm is that the Lloyd algorithm treats the data set as discrete distribution while the Forgy algorithm treats the data set as a continuous distribution. While the similarity between them is that they have the same procedure. The error function for a discrete distribution is defined as

$$E = \sum_{i=1}^{k} \sum_{j=1}^{n} f(x)\, d(c_i, x_{ij}) \quad (3)$$

In Equation (3) above, $d(c_i, x_{ij})$ is the distance function of the data point x_{ij} and cluster center c_i.

Algorithm.2: The Lloyd's (Standard) Algorithm.

1. Choose k data objects representing the cluster centroids.
2. Assign each data object of the entire data set to the cluster having the closest centroid.
3. Compute new centroid for each cluster by averaging the data observations belonging to the cluster.
4. If at least one of the centroids has changed, go to step 2, otherwise go to step 5
5. Output the clusters.

MacQueen's Method

MacQueen (1967) proposed the MacQueen's algorithm, and it is often referred to as basic k-means algorithm, which is an online (or incremental) algorithm. The MacQueen's method is similar to the Forgy's and Lloyd's Methods, but the main difference is that the centroids are updated by re-calculating the points (cases) any time it is moved. Once the initial centroids have been chosen in the same way like the Lloyd's algorithm, the iterations follows: For each case (x_i) in turn, after arbitrarily partitioning of points (items) into clusters, we compute the coordinates ($\bar{x}_i'^s$) of the cluster centroid (mean), likewise is the Euclidean distance computed for each point from the group centroids and reassign each point to the nearest group. If a point is moved from its initial position, the cluster centroid must be recalculated or updated before computing the squared distances.

If the centroid of a case belongs to the nearest subspace, no change is made. If another centroid is closest to the subspace, the case is re-assigned to the other centroid and the centroids for both the old and new subspaces (centers) are recalculated as the mean of the cases. When we see that each point is currently assigned to the clusters with the nearest centroid, the process stops.

Algorithm 3: The MacQueen's (Basic) Algorithm.

1. Choose k points as initial cluster centroids.
2. Assign each object to the cluster that has the closest centroid.
3. When all objects have been assigned, re-compile the positions of the k centroids.
4. If at least there is a change in one of the centroids, repeat step 2 and 3, otherwise go to step 5.
5. Output the clusters.

African Journal of Mathematics and Statistics Studies
ISSN: 2689-5323
Volume 3, Issue 1, 2020 (pp. 68-78)

www.abjournals.org

Hartigan and Wong's Method

Hartigan and Wong's method is a non-Lloyd heuristic that updates centers considering each point, rather than after each pass over the entire data set (Hartigan and Wong, 1979). Hartigan and Wong (1979) proposed the conventional k-means algorithm popularly known as Hartigan and Wong's algorithm.

It follows that the algorithm searches for the partition of data space with locally optimal within-cluster sum of squares error (SSE), which means that it may assign a case to another subspace, even if it currently belongs to the subspace of the closest centroid; doing so minimizes the total within-cluster sum of square (Morissette and Chartier, 2013). The initialization of the cluster centers is done in the same way as that of Lloyd's and Forgy's algorithm. The points (cases) are designated (assigned or allotted) to the centroid nearest to them and the centroids are calculated as the mean of the designated data points. The iterative steps are as follows:

Step 1. For each point I($I = 1, \dots, M$), find its closest and second closest cluster centers, $IC1(I)$ and $IC2(I)$, respectively. Assign point I to cluster $IC1(I)$.

Step 2. Update the cluster centers to be the average of the points contained within them.

Step 3. Initially, all clusters belong to the live set (specified number of k).

Step 4. This is the optimal-transfer (OPTRA) stage: Consider each point I ($I = 1, 2, \dots, M$) in turn. If cluster L ($L = 1, 2, \dots, K$) is updated in the last quick-transfer (QTRAN) stage, then the cluster belongs to the live set throughout this stage. Otherwise, at each step, it is not in the live set if it has not been updated in the last M optimal-transfer steps. Let point I be in cluster L1. If L1 is in the live set, do step 4a; otherwise, do step 4b.

Step 4a. Compute the minimum of the quantity, $R2 = [NC(L) * D(I,L)^2 \ ]/[NC(L) + 1]$, over all clusters $L(L \neq L1,\ L = 1,2, \dots, K)$ where the number of points in cluster L is denoted by $NC(L)$; while number of points in cluster $L1$ be $NC(L1)$; $D(I,L)$ is the Euclidean distance between point I and cluster L; $D[I, L(I)]$is the Euclidean distance between I and the cluster mean of the cluster containing I; $D(I,L)^2$ is the squared Euclidean distance between point I and cluster L. Let L2 be the cluster with the smallest R2. If this value is greater than or equal to $R1 = [NC(L1) * D(I,L1)^2]/[NC(L1) - 1]$, no reallocation is necessary and L2 is the new $IC2(I)$. Otherwise, point I is allocated to cluster L2, and L1 is the new IC2 (I). Cluster centers are updated to be the means of points assigned to them if reallocation has taken place. The two clusters that are involved in the transfer of point I at this particular step are now in the live set.

Step 4b. This step is the same as step 4a, except that the minimum R2 is computed only over clusters in the live set.

Step 5. Stop if the live set is empty; otherwise, go to step 6; after one pass through the data set.

African Journal of Mathematics and Statistics Studies
ISSN: 2689-5323
Volume 3, Issue 1, 2020 (pp. 68-78)

www.abjournals.org

Step 6. This is the quick-transfer (QTRAN) Stage: Consider each point $I(I = 1, 2, \dots, M)$ in turn. Let $L1 = IC1(I)$ and $L2 = IC2(I)$. It is not necessary to check the point I if both the clusters $L1$ and $L2$ have not changed in the last M steps. Compute the values:

$$R1 = [NC(L1) * D(I, L1)^2]/[NC(L1) - 1] \text{ and}$$

$$R2 = [NC(L2) * D(I, L2)^2]/[NC(L2) + 1]$$

If R1 is less than R2; point I remains in cluster $L1$. Otherwise, switch $IC1(I)$ and $IC2(I)$ and update the centers of clusters $L1$ and $L2$. The two clusters are also noted for their involvement in a transfer at this step.

Step 7. If no transfer took place in the last M steps, go to step 4, otherwise go to step 6.

Algorithm 4: The Hartigan and Wong's (Conventional) Algorithm.

1. Choose the number of cluster, k, and tentative centroids, $c_1, c_2, \dots, c_k$.
2. Observe an entity $i \in I$ coming either randomly or according to a pre-specified (dynamically) changing order.
3. d_{ij}= distance between case i and cluster j;
4. $d_{ij} = \arg min_{\ 1 \le j \le k}\, d_{ij}$
5. Assign cases *i* to cluster n_i ;
6. Re-compute the cluster means of any changed cluster above;
7. If no further change of cluster membership occurs in a complete iteration;
8. Output results.

Likas' Method

Likas et al. (2003**)** proposed a global k-means clustering algorithm, which constitutes a deterministic effective global clustering algorithm for the minimization of the clustering error that employs the basic k-means algorithm as a local search procedure. The basic idea behind the global k-means algorithm is that an optimal solution for a clustering problem with k clusters can be obtained by carrying out a series of local searches using the Basic k-means algorithm. At each local search, the $k - 1$ cluster centers are always initially placed at their optimal positions corresponding to the clustering problem with $k - 1$ clustering (Gan et al., 2007). The remaining kth cluster center is initially placed at several positions within the data space. Since for $k = 1$ the optimal solution is known, it can be iteratively applied to the above procedure to find optimal solutions for all m-clustering problems $m = 1, \dots, K$. (Likas et al., 2003). The k-clustering problem aims at partitioning the dataset into k disjoint subsets (clusters) $c_1, c_2, \dots, c_k$, such that the clustering criterion is optimized. The most widely used clustering criterion is the sum of the squared Euclidean distances between each data point, x_i, and the centroid, m_j. This criterion is called clustering error and it depends on the cluster center, $m_1, m_2, \dots, m_k$:

African Journal of Mathematics and Statistics Studies
ISSN: 2689-5323
Volume 3, Issue 1, 2020 (pp. 68-78)

www.abjournals.org

$$E(m_1, m_2, \dots, m_k) = \sum_i^n \sum_j^k d_{euc}^2(x_i, m_j) \quad (4)$$

In Equation (4) above, x_i and m_j, are the data point and the cluster center (centroid) while $d_{euc}^2(\dots)$ is the squared Euclidean distance which is one of the most widely used clustering criterion. This method does not depend on any initial values. Instead of selecting initial values randomly for all cluster centers as is the case with most global clustering algorithms, the method proceeds in an incremental way attempting to optimally add one new cluster center at each stage of the iteration.

The best solution obtained from the N runs is considered as the solution $[m_1^*(k), m_2^*(k), \dots, m_K^*(k)]$ of the K-clustering problem. By proceeding in the above fashion, we finally obtain a solution with m clusters having also found solution for all K-clustering problems with $k < m$ (Likas et al., 2003). The global K-means algorithm for the computation of $q \leq n$ cluster in the data set A can be described as follows.

Algorithm 5: The Global Likas Algorithm.

1. (Initialization) compute the centroid x_1, of the set A:

   $$x_1 = \frac{1}{n}\sum_{i=1}^{n} m_i,\ m_i \in A, \qquad i = 1,2,\cdots,n \text{ and set } k = 1.$$

2. Set $k = k + 1$ and consider the centers, $x_1, x_2, \cdots, x_{k-1}$, from the previous iteration.

3. Consider each point m, of A as a starting point for the k cluster center; thus obtain n initial solution with k points ($x_1, x_2, \cdots, x_{k-1}$, m); apply Basic K-means algorithm to each of them; keep the best k-partition obtained and its centers, $x_1, x_2, \cdots, x_k$.

4. If $k = n$ then stop, otherwise go to step 2.

Faber's Method

Faber (1994) proposed the Faber's method which is popularly known as the continuous k-means algorithm. The continuous k-means algorithm is faster than the standard k-means algorithm and it is also different from the standard k-means algorithm in two ways. First, the reference points in the continuous k-means algorithm are chosen as a random sample from the whole population of data point, while in the standard k-means algorithm the initial reference points are chosen more or less arbitrarily. Secondly, the way the data points are treated during the update process. During the iteration, the standard k-means algorithm examines all of the data points in sequence while the continuous k-means algorithm examines only a random sample of data points. If the data set is very large and the sample is a representative of the data set, the continuous k-means algorithm should converge much faster than the algorithm that examines every point in sequence. To be precise, the continuous k-means algorithm adopts MacQueen's method of updating the centroids during the initial partitioning, when the data points are first assigned to clusters (Faber, 1994).

Theoretically, random sampling represents a return to Macqueen's original concept of the algorithm as a method of clustering data over a continuous space. In Macqueen's formulation, the error measure E_i for each region R_i is given by

$$E_i = \int_{x \in R_i} f(x) \parallel x - z_i \parallel^2 dx \quad (5)$$

where $f(x)$ is the probability distribution function, which is a continuous function defined over the space, x is the data point and z_i is the centroid of the region R_i; while E_i is the total error measure.

Large set of discrete data point can be seen as a large sample as well as a good estimate of the continuous probability density $f(x)$. Then it suffices that a random sample of the data set can also be a good estimate of $f(x)$. Such a sample yields a representative set of cluster centroids and a reasonable estimate of the error measure without using all the points in the original data set.

RESULTS AND DISCUSSION

This section shows the performance comparison of the modified k-means method and the existing six k-means clustering methods using R statistical software (R version 3.2.2) support window 64-bit system. We conducted experiments using one simulated data set and two real-life data sets to ensure the efficiency of the proposed modified k-means method. The number of clusters k used two and three, since research has proven that the optimal number of clusters k will either be two, three, or four using methods like elbow, the silhouette and the gap statistic methods (Kaufman and Rousseeuw, 1990).

The performance of the proposed method was evaluated using total intra-cluster variance and accuracy parameters.

Total intra-cluster variance: The total intra-cluster variance is defined as the sum of squared distance between points and the corresponding centroid. That is; $W(C_K) = \sum_{x_i \epsilon c_k} (x_i - \mu_k)^2$ where

- x_i is the data point belonging to the cluster c_k.
- μ_k is the mean value of the points assigned to the cluster c_k.

Accuracy: Accuracy is defined as the ratio of the total number of correctly classified instances divided by total number of correctly plus incorrectly classified instances.

Simulated Data

The simulated data was generated randomly from a Gaussian (Normal) distribution with dimension of 300 rows and 2 columns (categories or attributes) that are divided into two and three clusters (that is, k = 2, 3). We chose 300 true centers uniformly at random given the above dimension. The point from the Gaussian distributions has a variance of 1 around each true center. Thus, this obtained a number of well separated Gaussians with the true centers providing a good approximation to the optimal clustering.

Shown below is the summary table of the results of experiments and data analysis of six versions of k-means clustering method when the number of clusters k is two and three respectively:

Table 1: Summary results of simulated data when the number of clusters k = 2 and 3.

Methods	When K = 2			When K = 3		
	Mean	Standard Deviation	Accuracy (%)	Mean	Standard Deviation	Accuracy (%)
Forgy	1.584	0.4949	80.0	2.248	0.7476	83.7
Lloyd	1.496	0.5020	79.1	1.920	0.8092	79.0
MacQueeen	1.504	0.5020	79.1	2.296	0.7831	81.4
Hartigan & Wong	1.504	0.5020	79.1	2.144	0.8299	78.3
Likas	1.776	0.4186	84.5	2.544	0.6780	88.6
Faber	1.760	0.4288	83.3	2.048	0.9233	72.0

From the above results of the simulation generated randomly, when the number of clusters k = 2, the Likas' method performed better than the other methods with minimum standard deviation of 0.4186 and 84.5 percent of accuracy, considering the fact that the variance (the total within-cluster sum of squares) is minimized; it measures the compactness (i.e. goodness) of the clustering which is meant to be as small as possible, also, high accuracy indicates how better the method is. When the number of clusters k = 3, Likas' method performed best with a standard deviation of 0.6780 and accuracy rate of 88.6 percent,

Real-Life Data

To understand how efficient these methods are under more practical circumstances, we run a number of experiments on two data sets which consist of the iris data set, and the yeast cell cycle data set. The first data set is from UC-Irvine Machine Learning Repository, and the second data set is from Stanford genomic resources (Stanford University). Each experiment involves solving k-means problem on a set of points in a real dimensional space.

Iris Data Set

The iris flower data set is a multivariate data set with 150 rows (instances) which is divided into 3 instances each, where each class refers to a type of iris plant (iris setosa, iris versicolor, and iris virginica): the number of columns (attributes) is 4 which consist of sepal length, sepal width, petal length and petal width (Fisher, 1936). The summary table of the results of experiments when the number of clusters k is two and three is shown in the Table 3.2 below:

Table 2: Summary results of iris data when the number of clusters k = 2 and 3.

Methods	When K = 2			When K = 3		
	Mean	Standard Deviation	Accuracy (%)	Mean	Standard Deviation	Accuracy (%)
Forgy	1.3533	0.4796	83.50	1.560	0.8067	82.00
Lloyd	1.6467	0.4796	83.50	2.4933	0.7396	85.20
MacQueeen	1.3533	0.4796	83.50	1.9333	0.5983	91.50
Hartigan & Wong	1.6467	0.4796	83.50	2.080	0.8633	79.10
Likas	1.6333	0.4835	84.70	2.6533	0.7234	86.00
Faber	1.8611	0.3459	89.23	2.4667	0.9193	77.45

From the above experiments and summary table on iris data set, it is observed that when the number of clusters k = 2, Faber's method outperformed the other methods with minimum standard deviation of 0.3459 and 89.23 percent accuracy. When the number of clusters k = 3, the MacQueen's method performed better than every other method with minimal standard deviation of 0.5983 and 91.50 percent accuracy.

Yeast Cell Cycle Data Set

The yeast cell cycle data set is a multivariate data with 6601 rows (instances) of genes and 18 attributes (columns). The output of the experiments when the number of clusters k = 2 and 3 will be summarized in Table 3.3 below:

Table 3: Summary results of yeast cell cycle data when the number of clusters k = 2 and 3.

Methods	When K = 2			When K = 3		
	Mean	Standard Deviation	Accuracy (%)	Mean	Standard Deviation	Accuracy (%)
Forgy	1.0786	0.2692	91.45	2.5946	0.7794	67.80
Lloyd	1.9209	0.2699	91.30	2.1763	0.4891	86.35
MacQueeen	1.0785	0.2689	91.80	2.1307	0.4310	88.10
Hartigan & Wong	1.9215	0.2689	91.80	1.2389	0.5040	83.79
Likas	1.3286	0.3394	86.70	2.7285	0.6273	75.95
Faber	1.8611	0.3459	87.16	1.8749	0.6595	73.50

From the experimental result of the yeast cell data set, it was observed that when the number of clusters k = 2, MacQueen's method and Hartigan and Wong's performed better than the other methods with a minimal standard deviation of 0.2689 with a high rate of accuracy of 93.60 percent. When the number of clusters k = 3, the MacQueen's method outperformed every other method with standard deviation of 0.4310 and 88.10 percent.

CONCLUSION

In this paper we have presented some versions of k-means clustering method and also compared them using simulated and real-life data, it was observed from the results of our simulation that Likas' method yielded excellent results when the number of cluster k = 2 and 3. In iris data experimental results, Faber's method performed better when the number of cluster k = 2 and MacQueen's method when k = 3, while in yeast cell cycle experimental results, MacQueen and Hartigan & Wong's method performed better when the number of clusters k = 2 and MacQueen's method when the number of clusters k = 3.

REFERENCES

Anderberg, M. R. (1973). Cluster Analysis for Applications. New York: Academic Press.

Estivill-Castro, V. (2002). Why so many clustering algorithms: a position paper. ACM SIGKDD Explorations Newsletter, 4(1):65-75.

Everitt, B., Landau, S., Leese, M., and Stajl, D. (2011). Cluster analysis, 5th edition, John Wiley and Sons.

Faber, V. (1994). Clustering and the continuous k-means algorithm. Los Alamos Science, 22:138-144.

Forgy, E. W. (1965). Cluster analysis of multivariate data: efficiency versus interpretability of classification. Biometrics, Vol. 21, pp. 768-769.

Gan, G., Ma, C., and Wu, J. (2007). Data clustering: Theory, Algorithms, and Applications. ASA-SIAM Series.

Hartigan, J. A. (1975). Clustering Algorithms. John Wiley & Sons. Inc., New York.

Hartigan, J. A. and Wong, M. A. (1979). Algorithm AS 136.A k-means clustering algorithm. Applied Statistics. 28, 100-108.

Kaufman, L. and Rousseeuw, P. J. (1990). Finding Groups in Data: An Introduction to Cluster Analysis. John Wiley & Sons, Inc.

Likas, A., Vlassis, N. and Verbeek, J. (2003). The global k-means clustering algorithm. Pattern Recognition, 36(2), 451-461.

Lloyd, S. (1982). Least squares quantization in PCM. IEEE Transaction on information Theory, 28(2), 129-137.

MacQueen, J. (1967). Some methods for classification and analysis of multivariate observations. In Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability, Vol. 1, pp.281-297. Berkeley, CA: University of California Press.

Mirkin, B. (2013). Clustering: A Data Recovery Approach, Second Edition (Chapman and Hall/CRC Computer Science and Data Analysis).

Morissette, L. and Chartier, S. (2013). The k-means clustering technique: General considerations and implementation in Mathematica. Tutorials in Quantitative Methods for Psychology, Vol. 9(1), 15-24.

Sokal, R. and Sneath, P. (1963). Principles of Numerical Taxonomy: San Francisco: California.

Xu, R. and Wunsch, D. C. (2009). Clustering IEEE Computational Intelligence Society, Series Press.John Wiley and Sons.