Asian Journal of Information Technology 12 (2): 77-82, 2013
ISSN: 1682-3915

An Efficient K-Means Initialization Using Minimum-Average-Maximum (MAM) Method

[1]S. Dhanabal and [2]S. Chandramathi
[1]Jansons Institute of Technology, Coimbatore, Tamilnadu, India
[2]Hindusthan College of Engineering and Technology, Coimbatore, Tamilnadu, India

Abstract: Data clustering is the process of grouping of data which are close together. The most popular clustering algorithm used in various domains is K-means. However, K-means algorithm has four main drawbacks: it converges to the local optimum solutions. The results obtained are strongly depends upon the selection of initial seeds, number of clusters need to be known in advance and it does not provide approximation guarantee. Various initialization methods were proposed to improve the performance of K-means algorithm. As the convergence of data points are only based on the selection of initial centroids, researchers are proposing an efficient algorithm for finding the initial centroids by considering distance on extreme ends, called K-means Minimum-Average-Maximum (K-MAM) Method. The proposed algorithm is tested with some of the UCI repository datasets and are compared with K-means and K-means++ algorithms. The results show that the proposed algorithm converges very fast with better accuracy.

Key words: Clustering, K-means, initialization techniques, K-means extreme end initialization, India

INTRODUCTION

Cluster analysis or clustering is the way of assigning a set of similar objects into groups (called clusters). A good clustering method will result in high quality clusters with high intra-cluster similarity and low inter-cluster similarity. It has applications in many areas including engineering, medicine, biology, nuclear science, radar scanning, research and development planning, data mining and image segmentation. The performance of clustering algorithms varies from one dataset to another. Choosing a single best clustering algorithm for all datasets is a tedious task as it depends on the nature of application and patterns to be extracted. Various algorithms have been proposed in the literature (Jain *et al*., 1999; Tan *et al*., 2006) to solve the clustering problem. Data clustering algorithms can be classified as hierarchical or partitional (A Tutorial on Clustering Algorithms http://www.elet.polimi.it/upload/matteucc/Clustering/tutorial_html/). In hierarchical clustering the data are partitioned as a series of partitions that may run from a single cluster containing all objects to n clusters each, containing a single object. Hierarchical clustering is subdivided into agglomerative methods which proceed by series of fusions of n objects into groups and divisive methods which separate n objects successively into finer groupings. Partitional clustering, on the other hand, attempts to directly decompose the data set into a set of disjoint clusters. The criterion function that the clustering algorithm tries to minimize may emphasize the local structure of the data as by assigning clusters to peaks in the probability density function or the global structure. Typically the global criteria involve minimizing some measure of dissimilarity in the samples within each cluster while maximizing the dissimilarity of different clusters.

The most popular, unsupervised, partition clustering algorithm is k-means. It is used to find k clusters which minimize SSE (Sum Squared Error). The performance of K-means algorithm is fully depend on the initial seed. If the initial partitions are not chosen carefully, the computation will run the chance of converging to a local minimum rather than the global minimum solution. The initialization step is therefore very important. To combat this problem it might be a good idea to run the algorithm several times with different initializations. If the results converge to the same partition then it is likely that a global minimum has been reached. This, however has the drawback of time consuming and computationally expensive. Various initialization techniques for K-means is proposed by so many researchers at various point of time. But no method best suits for all the datasets. So, researchers are proposing an algorithm called K-means MAM which can converge very fast.

K-MEANS CLUSTERING ALGORITHM

One of the most popular partitioning algorithms is K-means which is simple and fast (MacQueen, 1967). The algorithm begins with random initial centroids and keeps

Corresponding Author: S. Chandramathi, Hindusthan College of Engineering and Technology, Coimbatore, Tamilnadu, India

reassigning the patterns to clusters based on the similarity between the pattern and the cluster centroids until a convergence criterion is met after some number of iterations. The K-means algorithm is popular because it is easy to implement and its time complexity is O(n) where n is the number of patterns. Since, there are at most k^n possible clustering, the process will always terminate. The basic algorithm works as follows:

Algorithm 1 (K-means algorithm):

```
Arbitrarily choose K center locations ($C_1, ..., C_K$)
Assign each $X_i$ to its nearest cluster centre $C_i$
Update each cluster centre $C_i$ as the mean of all $X_i$ that have been assigned as closest to it
Calculate $D = \sum_{i=0}^{n} \min_{j=1..k} d(Xi, Ci)$
If the value of D has converged, then return ($C_1, ..., C_K$); else go to Step 2
```

In real, the algorithm requires very few iterations than any other clustering algorithms. Despite being used in a wide area of applications, the K-means algorithm is not exempt of drawbacks. Some of these drawbacks have been extensively reported in the literature. The most important are listed:

- It does not provide approximation guarantee (Forgey, 1965)
- It converges to the local optimum solutions (Selim and Ismail, 1984)
- The results obtained from this algorithm are strongly dependent to its initial points (Selim and Ismail, 1984)
- Number of clusters need to be known in advance (Fathian and Amiri, 2008)

To overcome the earlier drawbacks especially to provide approximation guarantee, many researchers tried to resolve the problem of selecting the initial centre through various methods.

LITERATURE REVIEW

Milligan (1980) shows the strong dependence of the K-means algorithm on initial clustering and suggests that good final cluster structures can be obtained using Ward's Hierarchical Method (Ward, 1963) to provide the K-means algorithm with initial clusters. Fisher (1996) proposes creating the initial clusters by constructing an initial hierarchical clustering based upon the research (Fisher, 1987). Higgs *et al.* (1997) and Snarey *et al.* (1997) suggest using a MaxMin algorithm in order to select a subset of the original database as the initial centroids to establish the initial clusters. In a recent study, Meila and Heckerman (1998) present some experimental results of an instance of the EM algorithm reminiscent of the K-means with three different initialization methods (being one of them a Hierarchical Agglomerative Clustering Method).

Bradley and Fayyad (1998) developed an algorithm for complex clustering methods by refining the initial seeds for the K-Means algorithm. The K-means++ Method (Arthur and Vassilvitskii, 2007) interpolates between MacQueen's Second Method and the Maxmin Method. It chooses the first center randomly and the ith (i {2, 3, ..., K}) center is chosen to be x' X with a probability of:

$$\frac{md(x^2)}{\sum_{j=1}^{n} md(x^2)}$$

Where:

md (x) = The minimum-distance from a point
x = The earlier selected centers

This method yields an υ(logK) approximation. The greedy K-means++ Method probabilistically selects log(K) centers in each round and then greedily selects the center that most reduces the SSE. This modification aims to avoid the unlikely event of choosing two centers that are close to each other. Pena *et al.* (1999) presented empirical comparison for four initialization methods for K-means algorithm and concluded that the random and Kaufman initialization method outperformed the other two methods with respect to the effectiveness and the robustness of K-means algorithm. Khan and Ahmad (2004) proposed Cluster Center Initialization Algorithm (CCIA) to solve cluster initialization problem. CCIA is based on two observations which some patterns are very similar to each other. It initiates with calculating mean and standard deviation for data attributes and then separates the data with normal curve into certain partition. CCIA uses K-means and density-based multi scale data condensation to observe the similarity of data patterns before finding out the final initial clusters.

Astrahan (1970) suggests using the nearest neighbour density when choosing seeds for the K-means algorithm. They use initial points that are both well-separated and have a large number of observations within a multidimensional sphere of the initial points. Hartigan and Wong (1979) developed an initialization method that involves choosing seeds that are of varying distances from the overall mean. All of the observations are ordered based on their distance to the overall mean and initial seeds are chosen as quantiles of these ordered observations. The specific quantiles correspond to equal increments in probabilities.

In a study by Faber (1994), K randomly sampled data points are selected as initial seeds. The reasoning behind this method is that denser portions of the dataset will likely be chosen as seeds. Researchers note that each random sample will produce different values for (Hartigan and Wong, 1979), therefore, researchers use an improved version of this method where several random samples are compared. The partition which minimizes (Hartigan and Wong, 1979) is chosen as the best candidate. Faber (1994) proposed a method which involves repeated division of the dataset into K clusters with seeds chosen as the centroids corresponding to the division that minimizes the method proposed by Bradley and Fayyad (1998). Steinley (2003) developed an initialization method that involves finding the dimension with maximum variance and dividing this into K groups where the corresponding data point for the median of each group will initialize the K-means algorithm. Hand and Krzanowski (2005) proposed an extension to the Faber Method starting with a random set of seeds. They suggest performing several iterations such that at each iteration random samples of points are placed in other groups. Mirkin (2005) used a Max Min procedure for initialization. This procedure attempts to find initial seeds that represent real observations and are well separated from all other seeds.

Most of the initialization methods that mentioned above do not constitute only initialization methods but also clustering methods themselves. When these methods are used with the K-means algorithm, it results in a hybrid clustering algorithm. Thus, these initialization methods suffer from the same problem as the K-means algorithm and they have to be provided with an initial clustering.

In this study, researchers are proposing a novel approach to find the initial seed for K-means algorithm by considering the extreme ends and then taking the centroid as the maximum, average and minimum distances.

PROPOSED ALGORITHM

In most of the initialization methods, it is observed that initial centroids are chosen randomly or by using a systematic approach. In those cases, the initial seeds may either fall in a dense area or it will give more outliers as a result. This is because points are chosen as maximum or somewhere else between maximum and minimum in most of the cases. In K-means algorithm, sometimes the data points which are far away from the centroid may be discarded due to high SSE value even though the point belongs to that centroid. To overcome this problem, researchers are proposing a method that can consider the points at the extreme ends and then finds the centroid for K-means. In the proposed method, researchers are implementing it in two phases: finding centroid using K-MAM Method and using normal K-means algorithm for finding the clusters. The two phases are depicted in the Fig. 1.

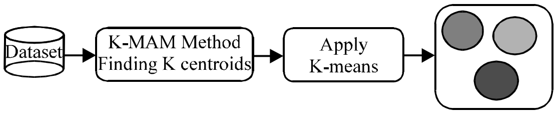

Fig. 1: Two phases of K-MAM algorithm

Initially, researchers have to choose the first data point (D_1) as the initial seed and calculate the distance to all other points. Almost all the initialization methods, explained earlier, choose the initial seed randomly or using some mathematical formulation. If researchers choose the initial seed randomly, again there is a chance of having more outliers. So, researchers took the first data point P_1 as the initial seed. After that researchers calculated the distance from the initial seed to all other data points using Euclidean distance Eq. 1:

$$D(x, y) = \left[\sum_{i=1}^{n} (x_i - y_i)^2 \right]^{1/2} \quad (1)$$

After calculating the distance, choose the maximum distance and mark it as P_2. Then, from P_2, again find the distance using Eq. 1 to all other data points and choose the minimum distance, say P_3. This is chosen because this point is the nearest point from P_2. This will give the minimum point on one end. Then, from P_3, again find the distance using Eq. 1 to all other data points and choose the maximum distance, say P_4. This is chosen because this point is the farthest point from P_3. This will give the maximum point on the other end. Then, calculate the mean, say P_5 by adding P_3 and P_4 and divided by number of rows in the dataset. The mean point may be chosen approximately nearest to the midpoint. Then, add these centroids to the centroid set. If $k = 2$ then choose P_3 and P_4 as centroids whereas if the number of clusters, say $k = 3$, then choose P_3, P_4 and P_5 as centroids and proceed as normal K-means. If $k>3$, then choose centroids by calculating the average between minimum and average and average and maximum. In all the calculations, researchers kept the maximum, average and minimum which is chosen at first. Proceed in this way until k cluster centroids are found. An example is depicted in Table 1 when $k>3$. Initially, it takes little time to find the maximum and minimum distance centroids. But when $k>2$, it calculates the centroids by considering the max seed and min seed. This reduces the time for calculating distances to every other point each time. After getting the centroids, proceed with simple K-means clustering algorithm. The convergence criterion for the K-means algorithm is minimizing the Sum Squared Errors (SSE). The equation to calculate SSE value is given Eq. 2:

$$SSE = \sum_{i=1}^{k} \sum_{x \in C_i} dist(C_i, X)^2 \quad (2)$$

Table 1: Example for finding the centroids when k>3

Centroids	3	5	7	10
1	1	1	1	1
2	-	-	2	2
3	-	3	-	3
4	-	-	4	4
5	5	5	5	5
6	-	-	-	6
7	-	-	7	7
8	-	8	-	8
9	-	-	9	9
10	10	10	10	10

where, dist is the standard Euclidean distance between two objects in the Euclidean distance. The overall implementation is given in the algorithm 1 and 2. The main advantage of this algorithm, researchers can find the centroids between minimum and maximum point of the given data set.

Algorithm 1 (Cluster K-MAM (Cluster K)):

```
Select the first data point $P_1$ as the initial seed.
Calculate the distance $D_1$ from $P_1$ to all the data points and select the maximum distance, say $P_2$.
Calculate the distance $D_2$ from P $_2$to all the data points and select the minimum distance, say $P_3$
Calculate the distance $D_3$ from P $_3$ to all the data points and select the maximum distance, say $P_4$
If (k = 2)
        Centroid C = {$P_3$, $P_4$};
If (k = 3)
{
   Calculate $P_5$ = ($P_{3+}$ $P_4$)/n, where n = # of rows.
   Centroid C ={$P_3$, $P_4$, $P_5$}
}
If (k>3)
{
for (i = 4 to n)
 {
   Calculate mean between $P_{i-1}$ and $P_{i+1}$
   Add it to C{}
   If (N(C) = k)  // where N(C) = No. of centroids in C, k = No. of clusters
       Break;
else
       Calculate mean between $P_i$ and $P_{i+1}$
       Add it to C{}
}
Return C
```

Algorithm 2 (Modified K-means):

```
1. C = K-MAM(Cluster K)
2. Assign each $X_i$ to its nearest cluster centre $C_i$.
3. Update each cluster centre $C_i$ as the mean of all $X_i$ that have been assigned as closest to it.
4. Calculate $\text{dist} = \sum_{i=0}^{n} \min_{j=1..k} d(Xi,Ci)$
5. If the value of dist has converged, then stop; else go to Step 2
```

EXPERIMENTAL RESULT

In this study, researchers compare the performance of the algorithm based on four criteria, two on effectiveness and two on efficiency. The effectiveness criteria are SSE and normalization and efficiency criteria are number of iterations and CPU time.

Sum Squared Errors (SSE): This is the objective function of K-means algorithm. The convergence of the dataset is either based on the number of iterations reached or SSE value is greater than the threshold value.

Normalization: Researchers normalize the dataset1, dataset3 and dataset4 using Min Max Normalization (Han and Kamber, 2006) which performs a linear transformation of original data. This is done so that the attribute data are scaled to fall within a small specified range such as [0.0-1.0]. The Eq. 3 is given:

$$V' = \frac{v - \min A}{\max A - \min A}(\text{new\_max}_A\text{-new\_min}_A) + \text{new\_min}_A \tag{3}$$

where, $\min_A$ and $\max_A$ are the minimum and maximum values of an attribute. It maps a value of v to V' in the range [new_min_A new_max_A]. This normalization prevents certain features to dominate the analysis because of their large numerical values.

Number of iterations: K-means requires number of iterations until reaching convergence when it is initialized by the centroid.

CPU time: This is the total time taken by the CPU the initialization and clustering phases. The experiments are conducted on a PC with an Intel Core i3 processor (2.4 GHz) and 4G byte of memory running the Windows 7 Home premium operating system. The implementation of the algorithm is done in. NET platform using C# language. All the four algorithms are run on three different datasets. The datasets are all well-known iris, wine and blood transfusion service centre dataset taken from UCI Machine Learning Laboratory (Blake and Merz, 1998).

Dataset1: This is the Iris data set which is perhaps the best-known database to found in the pattern recognition literature. The data set contains three classes of 50 instances each where each class refers to a type of iris plant. One class is linearly separable from the other 2; the latter are not linearly separable from each other. There are 150 instances with four numeric attributes in iris data set. There is no missing attribute value. The attributes of the iris data set are sepal length in cm, sepal width in cm, petal length in cm and petal width in cm.

Dataset2: This is the wine data set which also taken from UCI laboratory. These data are the results of a chemical analysis of wines grown in the same region in Italy but derived from three different cultivars. The analysis determined the quantities of 13 constituents found in each of the three types of wines. The dataset consists of three classes in which each class contains 59, 71 and 48 instances, respectively.

Dataset3: This is a Haberman dataset taken from UCI Repository. This data set contains cases from study conducted on the survival of patients who had undergone surgery for breast cancer. There are two classes of Survival status, the patient survived 5 years or longer and the patient died within 5 years. The data set consists of 306 examples with 3 attributes.

Dataset4: This is the blood transfusion service centre dataset which is also taken from UCI Laboratory. This is the donor database of Blood Transfusion Service Centre in Hsin-Chu City in Taiwan. The centre passes their blood transfusion service bus to one university in Hsin-Chu to gather blood donated about every 3 months. There are 748 instances with 5 attributes and a binary variable representing whether he/she donated blood in March 2007 (1 stand for donating blood; 0 stands for not donating blood). The performance of the proposed method is analysed based on the following two effectiveness criteria and two efficiency criteria.

Based on the earlier criteria, first, researchers run the K-means algorithm. As K-means converges very fast but the results are depends on the initial centroid, researchers have taken the average of 10 runs. The average SSE value is taken and is compared with the K-means++ and K-MAM Methods. It is shown in the Table 2. In this case, the proposed method has very low SSE value with the dataset1, 2 and 4 whereas in dataset3, the SSE value is equal to the K-means++ Method.

Table 3 gives the comparison of number of iterations taken by the initialization methods. From Table 3, it is proved that the proposed method outperforms well with all the given datasets except dataset2 where the proposed method takes one extra iteration compared to the other methods. But it is acceptable when it is compared in terms of accuracy.

Table 4, Fig. 2 and depicts the comparison of CPU time of various initialization methods. Researchers have taken the average CPU time for K-means for 10 runs whereas the proposed method outperforms well compared to the other methods. In dataset2, K-means++ has little bit low CPU time compared to the method and K-means.

Finally, researchers compared the performance of the proposed method with other methods. It shows that based on the CPU time and accuracy, the proposed method outperforms well compared to the other methods. This is shown in the Table 5. Table 6 gives the overall performance percentage of the proposed method compared with K-means and K-means++ in terms of SSE, No. of iterations, CPU time and accuracy on tested datasets.

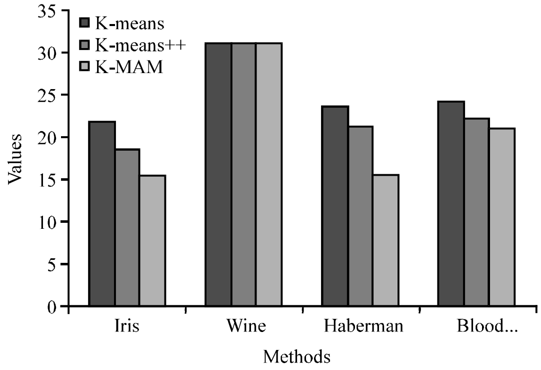

Fig. 2: Comparison of CPU time of initialization methods

Table 2: Comparison of SSE values

Initialization Methods	Iris	Wine	Haberman	Blood transfusion
K-means	8.83E+01	2.46E+06	3.09E+04	3.80E+01
K-means++	7.95E+02	2.43E+05	3.06E+04	3.68E+01
K-MAM	5.31E+03	2.42E+05	3.06E+04	3.06E+04

Table 3: Comparison of number of iterations

Dataset	Iris	Wine	Haberman	Blood transfusion
K-means	7	8	8	6
K-means++	6	8	7	6
K-MAM	5	9	4	5

Table 4: Comparison of CPU time in milli seconds

Dataset	Iris	Wine	Haberman	Blood transfusion
K-means	21.84	31.20	23.65	24.31
K-means++	18.70	31.10	21.29	22.36
K-MAM	15.60	31.20	15.60	21.24

Table 5: Comparison of accuracy

Dataset	Iris	Wine	Haberman	Blood transfusion
K-means	85.90	87.75	92.91	89.97
K-means++	91.20	92.50	95.70	91.70
K-MAM	96.00	94.38	96.40	92.11

Table 6: Overall performance percentage of K-MAM compared with K-means and K-means++

	K-means (%)				K-means++ (%)			
Performance criteria	Dataset1	Dataset2	Dataset3	Dataset4	Dataset1	Dataset2	Dataset3	Dataset4
SSE	39.88	1.79	0.94	19.45	33.20	0.53	-0.08	16.86
CPU time	28.57	0.00	34.03	12.62	16.57	-0.32	26.71	5.02
No. of iterations	28.57	-12.50	50.00	16.67	16.67	-12.50	42.86	16.67
Increased accuracy	10.52	7.02	3.62	2.32	5.00	1.99	0.73	0.45

CONCLUSION

Clustering is the process of grouping of data which are close together. The drawback of K-means algorithm is the convergence of the algorithm is based on the selection of initial centroids. So, researchers proposed a method called, K-MAM Method which considers the data point on exterior ends as well as the interior points. The experimental results shows that the proposed algorithm outperforms well in the sense of minimizing the SSE value, number of iteration, CPU time and accuracy. The proposed algorithm is well suited for low dimensional datasets.

In the future research, as the algorithm outperforms well in the case of low dimensional datasets and researchers are working to improve it to suit the high dimensional datasets. This can be done by using dimension reduction methods to improve the efficiency of the algorithm for high dimensional dataset.

REFERENCES

Arthur, D. and S. Vassilvitskii, 2007. k-means++: The advantages of careful seeding. Proceedings of the 18th Annual ACM-SIAM Symposium of Discrete Analysis, Jan. 7-9, New Orleans, Louisiana, pp: 1027-1035.

Astrahan, M.M., 1970. Speech Analysis by Clustering, or the Hyperphoneme Method. Defense Technical Information Center, Palo Alto, CA.

Blake, C.L. and C.J. Merz, 1998. UCI Repository of Machine Learning Databases. 1st Edn., University of California, Irvine, CA.

Bradley, P.S. and U.M. Fayyad, 1998. Refining initial points for K-means clustering. Proceedings of the 15th International Conference on Machine Learning, July 24-27, Morgan Kaufmann, San Francisco, pp: 91-99.

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

Fathian, M. and B. Amiri, 2008. A honeybee-mating approach for cluster analysis. Adv. Manuf. Tech., 1: 809-821.

Fisher, D., 1996. Iterative optimization and simplification of hierarchical clustering. J. Artif. Intell. Res., 4: 147-179.

Fisher, D.H., 1987. Knowledge acquisition via incremental conceptual clustering. Machine Learning, 2: 139-172.

Forgey, E., 1965. Cluster Analysis of multivariate data: Efficiency versus interpretability of classification. Biometrics, 21: 768-780.

Han, J. and M. Kamber, 2006. Data Mining-Concepts and Techniques. 2nd Edn., Morgan Kaufmann Publishers, USA., ISBN: 1558609016, pp: 800.

Hand, D.J. and W.J. Krzanowski, 2005. Optimising k-means clustering results with standard software packages. Comput. Stat. Data Anal., 49: 969-973.

Hartigan, J.A. and M.A. Wong, 1979. Algorithm AS 136: A K-means clustering algorithm. J. Royal Statist. Soc. Ser. C Applied Statist., 28: 100-108.

Higgs, R.E., K.G. Bemis, I.A. Watson and J.H. Wikel, 1997. Experimental designs for selecting molecules from large chemical databases. J. Chem. Inf. Comput. Sci., 37: 861-870.

Jain, A.K., M.N. Murty and P.J. Flynn, 1999. Data clustering: A review. ACM Comput. Surveys, 31: 264-323.

Khan, S.S. and A. Ahmad, 2004. Cluster center initialization for K-mean clustering. Pattern Recognit. Lett., 25: 1293-1302.

MacQueen, J., 1967. Some methods for classification and analysis of multivariate observations. Proceedings of the 5th Berkeley Symposium on Mathematical Statistics and Probability, January 17-20, 1967, Berkeley, CA., USA., pp: 281-297.

Meila, M. and D. Heckerman, 1998. An experimental comparison of several clustering and initialization methods. Proceedings of the 14th Conference on Uncertainty in Artificial Intelligence, July 24-26, 1998, Morgan Kaufmann, San Francisco, CA, pp: 386-395.

Milligan, G.W., 1980. An examination of the effect of six types of error perturbation on fifteen clustering algorithms. Psychometrika, 45: 325-342.

Mirkin, B., 2005. Clustering for Data Mining: A Data Recovery Approach. Chapman and Hall, London.

Pena, J.M., J.A. Lozano and P. Larranaga, 1999. An empirical comparison of four initialization methods for the K-Means algorithm. Pattern Recognit. Lett., 20: 1027-1040.

Selim, S.Z. and M.A. Ismail, 1984. K-means type algorithms: A generalized convergence theorem and characterization of local optimality. IEEE Trans. Pattern Anal. Mach. Intellg., 6: 81-87.

Snarey, M., N.K. Terrett, P. Willet and D.J. Wilton, 1997. Comparison of algorithms for dissimilarity-based compound selection. J. Mol. Graphics Modell., 15: 372-385.

Steinley, D., 2003. Local optima in k-means clustering: What you don't know may hurt you. Psychol. Methods, 8: 294-304.

Tan, P.N., M. Steinbach and V. Kumar, 2006. Cluster Analysis: Basic Concepts and Algorithms. In: Introduction to Data Mining, Tan, P.N., M. Steinbach and V. Kumar (Eds.)., Pearson Addison Wesley, Boston.

Ward, Jr. J.H., 1963. Hierarchical grouping to optimize an objective function. J. Am. Stat. Assoc., 58: 236-244.