Rep. Fac. Sci. Engrg.
Saga Univ.
36-1 (2007),25-31

Reports of the Faculty of Science and Engineering,
Saga University, Vol. 36, No.1, 2007

Hierarchical K-means: an algorithm for centroids initialization for K-means

By
Kohei Arai* and Ali Ridho Barakbah,**

Abstract: Initial starting points those generated randomly by K-means often make the clustering results reaching the local optima. The better results of K-means clustering can be achieved after computing more than one times. However, it is difficult to decide the computation limit, which can give the better result. In this paper, we propose a new approach to optimize the initial centroids for K-means. It utilizes all the clustering results of K-means in certain times, even though some of them reach the local optima. Then, we transform the result by combining with Hierarchical algorithm in order to determine the initial centroids for K-means. The experimental results show how effective the proposed method to improve the clustering results by K-means.

Key words: Clustering, K-means clustering, Initial centroid determination, Hierarchical algorithm

1. Introduction

Clustering is an effort to classify similar objects in the same groups. Cluster analysis constructs good cluster when the members of a cluster have a high degree of similarity to each other (internal homogeneity) and are not like members of other clusters (external homogeneity) (Grow, 1999; Castro, 2002). It means process to define a mapping f:$D \rightarrow C$ from some data $D=\{d_1,d_2,..,d_n\}$ to some clusters $C=\{c_1,c_2,..,c_n\}$ on similarity between d_i. There many applications of clustering diverse in many fields, such as data mining, pattern recognition, image classification, biological sciences, marketing, city-planning, document retrievals, etc.

The most well known methods for clustering is K-means developed by Mac Queen in 1967. The simplicity of K-means made this algorithm used in various fields. K-means is a partition clustering method that separates data into k mutually excessive groups. By iterative such partitioning, K-means minimizes the sum of distance from each data to its clusters. K-means method is very popular because of its ability to cluster huge data, and also outliers, quickly and efficiently. It remains a basic framework for developing numerical or conceptual clustering systems because various possibilities of distance and prototype choice (Ralambondrainy, 1995).

However, K-means algorithm is very sensitive in initial starting points. K-means generates initial cluster randomly. When random initial starting points close to the final solution, K-means has high possibility to find out the cluster center. Otherwise, it will lead to incorrect clustering results (Cheung, 2003). Because of initial starting points generated randomly, K-means does not guarantee the unique clustering results. (Shehroz and Ahmad, 2004). K-means method is difficult to reach global optimum, but only in local minimum (Kövesi, 2001).

Several methods proposed to solve the cluster initialization for K-means. A recursive method for initializing the means by running K clustering problems is discussed by Duda and Hart (1973). A variant of this method consists of taking the entire data and then randomly perturbing it K times (Shehroz and Ahmad, 2004). Bradley and Fayyad (1998) proposed an algorithm that refines initial points by analyzing distribution of the data and probability of data density (Bredley and Fayyad, 1998). Penã 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. Shehroz 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. The experiment results of CCIA performed the effectiveness

Received on Apr.28 2007

*Department of Information Science
**EEPIS:Electric Engineering Politechnique in Surabaya

and robustness this method to solve the several clustering problems.

In Section 2, we describe the K-means algorithm and distortion of the method. In Section 3, we propose Hierarchical K-means algorithm as a new approach to determine the centroids initialization for K-means algorithm. We describe our proposed method how to designate the initial cluster centers. Section 4 performs the experimental results on the normal data distribution as well as real world data set. In Section 5, we draw conclusions.

These introductions give you guidelines for preparing papers for the Reports of the Faculty of Science and Engineering, Saga University. The reports are printed by photo-offset reproduction of the material prepared by the authors.

2. K-means algorithm

2.1. Basic theory

Let $A = \{a_i \mid i = 1,..., n\}$ be attributes of n-dimensional vector and $X = \{x_i \mid i = 1,..., r\}$ be each data of *A*. The K-means separates *X* into *K* partitions called clusters $S = \{s_i \mid i = 1,..., K\}$ where $M \in X$ is $M = \{m_i \mid i = 1,..., n(s_i)\}$ as members of *S*. Each cluster has cluster center $C = \{c_i \mid i = 1,..., k\}$.

K-means algorithm can be described as follows:

1. Initiate its algorithm by generating random initial cluster centers c_k.
2. Calculate the distance *d(x,c)* between vector x_i to cluster center c_k. Euclidean distance can be used to express the distance.
3. Separate x_i into s_k which has minimum *d(x,c)*.
4. Determine the new cluster centers defined as:

$$c_i = \frac{1}{p}\sum_{j=1}^{p} m(s_i, j), \qquad \text{where } p=n(s_i) \qquad (1)$$

5. Go back to step 2 until *Ci* = *Ci*-1.

It may stop in the t iteration with a threshold ε (Kövesi et al, 2001) if K-means reaches as:

$$\left|\frac{C^t - C^{t-1}}{C^t}\right| < \varepsilon \qquad (2)$$

2.2. Distortion aggregate

To calculate the distortion of K-means method, let E:*X*→*S* be encode function to cluster *X* into *S*, and D:*S*→*X* be the decode function. The distortion of clustering can be defined as:

$$Distortion = \left|\sum_{i=1}^{r}\left(x_i - D\left[E\left(x_i\right)\right]\right)\right| \qquad (3)$$

The correct clustering has x_i=*D[E(x_i)]*, so that *Distortion* is 0. The good clustering performs minimum *Distortion*. Therefore, it try to make *Distortion* as minimum as possible. Referring Eq.1 and $M \in X$, the effort to minimize *Distortion* can be set by minimizing *P* as:

$$P = \left|\left(m(s_i, j) - c_j\right)\right| \qquad (4)$$

where c_k is the cluster center of *m(s_i,k)*. Therefore, the determining of initial cluster centers for K-means is very important because it can determines the distortion and the precision of clustering results.

3. Hierarchical K-means

In this paper, a new approach to determine initial centroids for K-means is proposed, called as Hierarchical K-means. The approach combines the K-means and Hierarchical algorithm.

3.1. Basic concept

The better result of K-means clustering can be achieved after making some experiments. However, it is difficult to decide the limitation of experiments those which one can give the better result. We do not know what the numbers of experiments those have been done are enough to get the best result or perhaps the next experiment will achieve the better result. This kind of uncertainty makes the K-means algorithm somewhat difficult to be applied in real clustering cases.

Actually, the clustering result of K-means can be considerable as valuable input to get the better result, even though it reaches the local optima, because it reflects the partitioned feature space based on the certain initial points which were generated randomly.

We utilize all clustering results by K-means in order to determine the desired initial centroids. Let us illustrate the clustering case of Ruspini data set in Fig. 1.

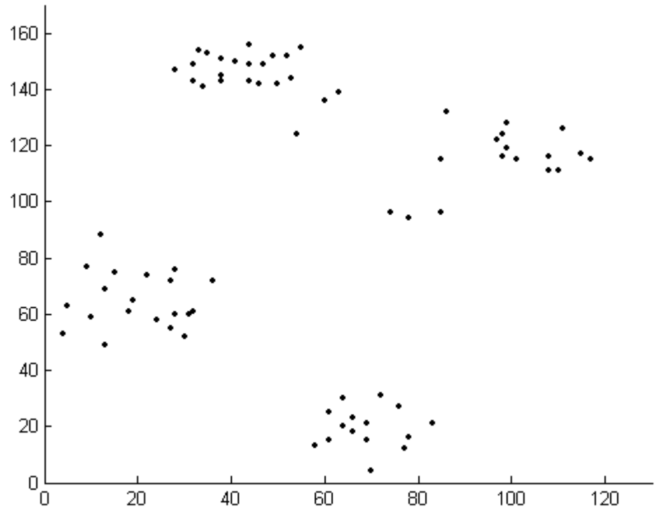


Fig. 1. Ruspini data set

Then we apply K-means algorithm to solve the clustering case of Ruspini data set. Because K-means generates random initial starting points, the clustering result is not unique, and we do not know which one is best result. By computing in certain times, some different clustering results of Ruspini data set can be produced. For each computation, we record each final centroids. Fig. 2. shows the different final centroids from 10 computations by K-means.

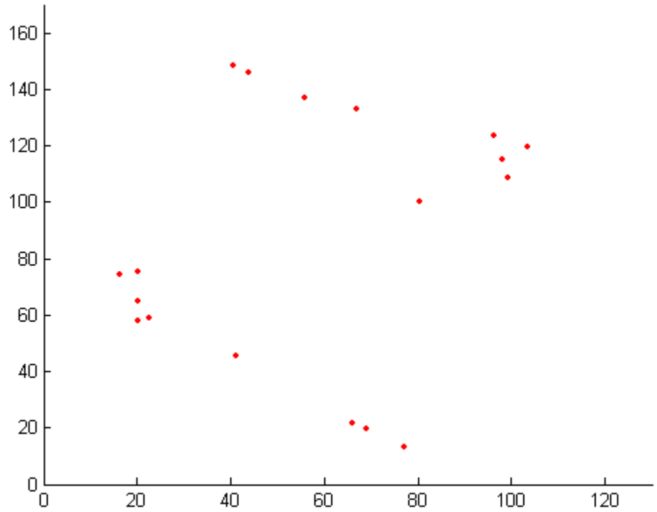


Fig. 2. The final centroids from 10 computations

After we get the all final centroids from certain computations, we apply the hierarchical clustering algorithms. However, we have tried in some data sets of clustering cases with different hierarchical algorithms (Single, Centroid, Complete and Average Linkage) without having any change in the results. The final centroids after applying hierarchical algorithm can be used for initial centroids for K-means.

For simple clustering cases, like Ruspini data set, Hierarchical K-means does not perform the superiority. But for the complex clustering cases with large numbers of data set and many dimensional attributes those are very difficult to be visualized, Hierarchical K-means can show the good performance, both in precision and speed. The experiment results in Section 4 will perform the clustering results of Hierarchical K-means, compared than the other clustering algorithms.

3.2. Algorithm

In this subsection we present execution steps of our proposed Hierarchical K-means algorithm to determine initial centroids for K-means. The algorithm is described as follows:

1. Set $X = \{x_i \mid i = 1,...,r\}$ as each data of *A*, where $A = \{a_i \mid i = 1,...,n\}$ is attribute of n-dimensional vector.
2. Set *K* as the predefined number of clusters.
3. Determine *p* as numbers of computation
4. Set *i*=1 as initial counter
5. Apply K-means algorithm.
6. Record the centroids of clustering results as $C_i = \{c_{ij} \mid j=1,\dots,K\}$
7. Increment *i*=*i*+1
8. Repeat from step 5 while *i*<*p*.
9. Assume $C = \{C_i \mid i=1,\dots,p\}$ as new data set, with *K* as predefined number of clusters
10. Apply hierarchical algorithm
11. Record the centroids of clustering result as $D = \{d_i \mid i=1,\dots,K\}$

Then, we can apply $D = \{d_i \mid i=1,\dots,K\}$ as initial cluster centers for K-means clustering. The experiment results will perform the accuracy of our proposed method.

4. Experimental results

In order to analyze the accuracy of our proposed method, we apply Hierarchical K-means to three kinds of experiments: random normal data distribution, real world data sets and image clustering case.

4.1. Random normal data distribution

This kind of experiment can express the ability of the proposed method to solve clustering cases with normal data distribution. In the experiment, we use two-dimensional data set (x and y). Then, we use *normrnd* function in Matlab to generate the random normal data distribution. We determine 9 nodes as random mean, (20,20), (50,20), 80,20), (20,50), (50,50), (80,50), (20,80), (50,80) and (80,80), with minimal number of nodes = 6. We set the standard deviation = 5 because we want to intent well-separated clusters. We determine numbers of data ≥ 10 for each nodes. With this model, it can generate thousands of different combination for normal data distribution.

We made 1000 experiments and used Centroid Linkage as Hierarchical algorithm in Hierarchical K-means with 10 computation times for K-means (this is also used for other data set experiments) . For each experiment, we compared our proposed method with some Hierarchical algorithms (Single Linkage, Centroid Linkage, Complete Linkage and Average Linkage), Fuzzy c-means and K-means using random initialization. For Fuzzy c-means, we use 1.5 for degree of fuzziness. For K-means, we compute 1000 iteration times and take the average results.

We represent variance factors as performance measure in the experiments. Variance constraint (Veenman et al, 2002) can express the density of the clusters with variance within cluster and variance between clusters (Ray and Turi, 1999; Ming and Hou, 2004). The ideal cluster has minimum variance within clusters (V_w) to express internal homogeneity and maximum variance between clusters (V_b) to express external homogeneity (Barakbah and Arai, 2004).

$$V = \frac{Vw}{Vb} x100\% \qquad (5)$$

Good cluster can be represented by low V. Table 1 performs the error comparison between four methods from experiments results.

Table 1 Comparison error using random normal data distribution

	V (%)
Single linkage	0.7345
Centroid linkage	0.5452
Complete linkage	0.5524
Average linkage	0.5455
Fuzzy c-means	0.6962
K-means using random initialization	1.0730
Hierarchical K-means	0.5562

We can see that the error of our proposed method has low V, and is very close to V computed by Centroid and Average Linkage. It means that our proposed method can be considerable to solve the well-separated clustering cases.

4.2. Real world data sets

In order to analyze the accuracy of our proposed method, we try to make experiments using a number of real world data sets. The data sets, which we use, are Iris data, Wine data, Fossil data, Ruspini data, Letter Recognition data and New Thyroid data.

We use raw data sets in the experiments, because we concern in the accuracy of the methods to solve clustering cases. If we normalize the data, even though it is usual to get the better clustering results, the clustering results are not only depend on clustering methods, but also are depend on normalization methods. Therefore, we decide to not normalize the data in order to ensure that the clustering results are absolutely depending on the accuracy of clustering methods.

We compute the real world data sets with several methods, Single Linkage, Centroid Linkage, Complete Linkage, Average Linkage, Fuzzy c-means, K-means using random initialization and our proposed method. In some data sets, we add CCIA (Sheroz and Ahmad, 2004) as the other comparison method, even though its clustering result computed after normalizing the data to lie between 0 and 1. For Fuzzy c-means, we use 1.5 for degree of fuzziness. For K-means, we compute 1000 iteration times and take the average results.

We represent error percentage as performance measure in the experiments. It is calculated from number of misclassified patterns and the total number of patterns in the data sets.

$$Error = \frac{Numberofmisclassified}{Numberofpatterns} x100\% \qquad (6)$$

4.2.1. Iris data set

We obtained this data set from UCI Repository. This data set contains information about Iris flowers. There are three classes of Iris flowers, namely Iris Setosa, Iris Versicolor and Iris Virginica. The data set consists of 150 examples with 4 attributes. One class is well separable the other two. The others have a large overlap.

Table 2 Iris data set

	Error (%)
Single Linkage	32
Centroid Linkage	9.3333
Complete Linkage	16
Average Linkage	9.3333
Fuzzy c-means	13.524
K-means using random initialization	17.7027
Hierarchical K-means	10.6667
K-means using CCIA	11.33

4.2.2. Wine data set

We obtained this data set from UCI Repository. The data is the result of a chemical analysis of wines grown in a region in Italy but derived from three different cultivars. There are three classes. The dataset consists of 178 examples each with 13 continuous attributes. The data set contains distribution 59 examples of class 1, 71 examples for class 2 and 48 examples for class 3.

Table 3 Wine data set

	Error (%)
Single Linkage	57.3034
Centroid Linkage	38.764
Complete Linkage	32.5843
Average Linkage	38.764
Fuzzy c-means	30.3371
K-means using random initialization	32.5236
Hierarchical K-means	29.7753
K-means using CCIA	5.05

The high error happened with Hierarchical K-means compared with CCIA because the raw data actually has far difference scale among attributes. There is an attribute that has high scale of value compared to the others. For this case, the data is usually better to standardize before clustering. Table 4 performs the error of Hierarchical K-means after normalizing the data using 4 different normalization methods.

Table 4 Error of Hierarchical K-means after normalizing the wine data

Normalization method	Error (%)
Min-max (0-1)	3.9326
Z-Score	3.3708
Sigmoid	2.809
Softmax	2.809

4.2.3. Fossil data set

The Fossil data is obtained from Chernoff (Yi-tzuu, 1978). It consists of 87 nummulitidae specimens from Eocene yellow limestone formation of northwestern Jamaica. There are three 6 attributes with 3 classes which the distribution is 40 examples of class 1, 34 examples of class 2 and 13 examples of class 3.

Table 5 Fossil data set

	Error (%)
Single Linkage	13.7931
Centroid Linkage	11.4943
Complete Linkage	14.9425
Average Linkage	9.1954
Fuzzy c-means	11.5057
K-means using random initialization	8.5931
Hierarchical K-means	3.4483
K-means using CCIA	0

We can see in Table 5 that K-means using CCIA computed a better result compared to Hierarchical K-means. However, we try to make experiments with normalizing the data. Table 6 performs the error of Hierarchical K-means after normalizing the data using 4 different normalization methods.

Table 6 Error of Hierarchical K-means after normalizing the fossil data

Normalization method	Error (%)
Min-max (0-1)	0
Z-Score	0
Sigmoid	0
Softmax	0

4.2.4. Ruspini data set

The Ruspini data set represents a simple, well-known example that is commonly used as a benchmark problem in evaluating clustering methods and is widely available, incorporated as a built-in data object in both R and S-plus statistics packages (Pearson et al, 2004). The data set consists of 75 bi-variate attribute vectors. There are four classes. The data set contains 23, 20, 17 and 15 in classes 1, 2, 3 and 4 respectively.

Table 7 Ruspini data set

	Error (%)
Single Linkage	0
Centroid Linkage	0
Complete Linkage	4
Average Linkage	0
Fuzzy c-means	0
K-means using random initialization	13.6973
Hierarchical K-means	0
K-means using CCIA	4

4.2.5. Letter recognition data set

This data set obtained from UCI Repository. The objective is to identify each of a large number of black-and-white rectangular pixel displays as one of the 26 capital letters in the English alphabet. The character images were based on 20 different fonts and each letter within these 20 fonts was randomly distorted to produce a file of 20,000 unique stimuli. Each stimulus was converted into 16 primitive numerical attributes (statistical moments and edge counts), which were then scaled to fit into a range of integer values from 0 through 15. The training data consists of first 16000 items and then used the resulting model to predict the letter category for the remaining 4000. For experimental purpose we have taken 595 patterns of letter A and 597 patterns of letter D from the training data set, as CCIA has done.

Table 8 Letter recognition data set

	Error (%)
Single Linkage	49.8322
Centroid Linkage	48.1544
Complete Linkage	42.7852
Average Linkage	6.8792
Fuzzy c-means	13.1711
K-means using random initialization	8.2323
Hierarchical K-means	8.2215
K-means using CCIA	8.55

4.2.6. New thyroid data set

The new thyroid data set is also obtained from UCI Repository. The data set contains information about classification whether a patient's thyroid to the class euthyroidism, hypothyroidism or hyperthyroidism. The diagnosis (the class label) was based on a complete medical record, including anamnesis, scan etc. The data set consists 5 attributes, with 215 examples. The distribution is 150 of class euthyroidism, 35 of class hypothyroidism and 30 of class hyperthyroidism.

Table 9 New thyroid data set

	Error (%)
Single Linkage	29.7674
Centroid Linkage	27.907
Complete Linkage	28.3721
Average Linkage	26.0465
Fuzzy c-means	14.4186
K-means using random initialization	20.9126
Hierarchical K-means	13.9535

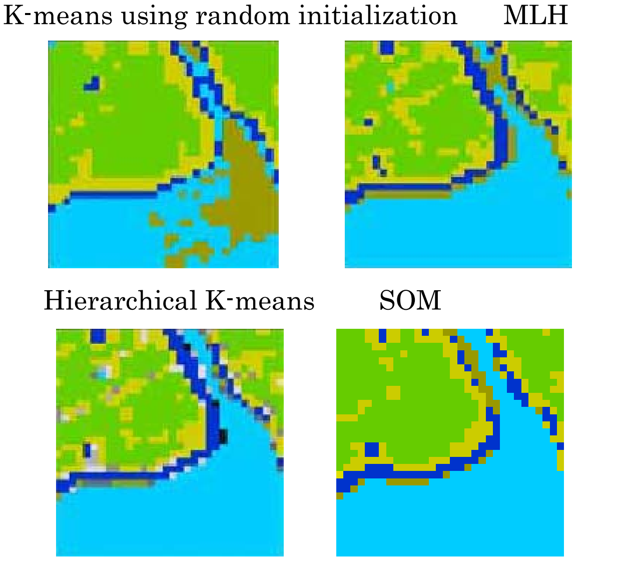
4.3. Image clustering case

We also try to apply our proposed method for multi bands image clustering case. The imagery data (Fig. 7) that we use contains information about Landsat TM data of Saga, Japan, acquired on May 1989. There are five classes, Ariake sea, road, paddy field, bare soil and artificial construction.

Fig. 8 performs the comparison results between our proposed method, K-means using random initialization, Maximum Likelihood (MLH) and SOM (Arai, 2004). It is found that the clustering result from our proposed method can make better separated cluster than the others.

5. Conclusions

It is widely reported that the K-means algorithm suffers from initial cluster centers. Our main purpose is to optimize the initial centroids for K-means algorithm. Therefore, in this paper, we proposed Hierarchical K-means algorithm. It utilizes all the clustering results of K-means in certain times, even though some of them reach the local optima. Then, we transform the all centroids of clustering result by combining with Hierarchical algorithm in order to determine the initial centroids for K-means. This algorithm is better used for the complex clustering cases with large numbers of data set and many dimensional attributes. Hierarchical K-means bargains the advantage of K-means algorithm in speed and hierarchical algorithm in precision. Experimental results with random normal data distribution, real world data sets, and multi band image clustering performs the accuracy and improved clustering results as compared to some clustering methods.


K-means using random initialization

MLH

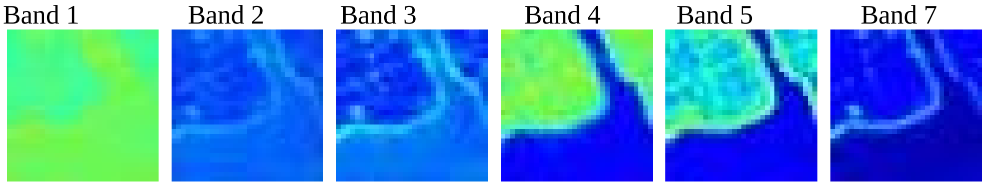
Hierarchical K-means

SOM

Fig. 8. Comparison of clustering results between 4 methods

Table 10 Computation time

	Iris (ms)	Wine (ms)	Fossil (ms)	Ruspini (ms)	Letter (s)	New thyroid (ms)
Single Linkage	78	156	31	15	38.766	188
Centroid Linkage	78	156	31	15	38.734	188
Complete Linkage	78	156	31	15	38.687	188
Average Linkage	297	562	78	47	209.625	875
Fuzzy c-means	39.344	140	27.386	15	0.406	219
K-means using random init.	3.168	5.133	1.977	2.459	0.0051	6.057
Hierarchical K-means	47	62	31	31	0.453	62


Band 1

Band 2

Band 3

Band 4

Band 5

Band 7

Fig. 7. Landsat imagery data of Saga, Japan (32x32)

References

Arai, K.,(2004) Learning processes of image clustering method with density maps derived from Self-Organizing Mapping(SOM), Journal of Japanese Society for Photogrametry and Remote Sensing、43、5、62-67.

Arai, K., Barakbah, A.R., 2004. Method for shape independent clustering in case of numerical clustering together with condensed clustering. The 8th World Multi-Conference on Systemics, Cybernetics And Informatics (SCI 2004), Orlando, Florida.

Barakbah, A.R., K. Arai,, 2004. Identifying moving variance to make automatic clustering for normal data set. Proc. IECI Japan Workshop 2004 (IJW 2004), Musashi Institute of Technology, Tokyo.

Bradley, P.S., Fayyad, U.M., 1998. Refining initial points for K-means clustering. Proc. 15th Internat. Conf. on Machine Learning (ICML’98).

Castro, V.E., 2002. Why so many clustering algorithms-a position paper. ACM SIGKDD Explorations Newsletter, Volume 4, Issue 1, pp. 65-75.

Cheung, YM., 2003. k*-Means: A new generalized k-means clustering algorithm. Pattern Recognition Lett. 24, 2883-2893.

Cowgill, M.C., Harvey, R.J., 1999. A genetic algorithm approach to cluster analysis. Computers and Mathematics with Applications 37, 99-108.

Fraley, C., Raftery, A.E. 1998. How many clusters? Which clustering method? Answer via model-based cluster analysis. The Computer Journal. Vol. 41, No. 48.

Growe, G.A., 1999. Comparing Algorithms and Clustering Data: Components of The Data Mining Process, thesis, department of Computer Science and Information Systems, Grand Valley State University.

Halkidi, M., Batistakis, Y., Vazirgiannis, M., 2001. Clustering algorithms and validity measures. Proceedings of The 13th International Conference on Scientific and Statistical Database Management, July 18-20, IEEE Computer Society, George Mason University, Fairfax, Virginia, USA.

Khan, S.S., Ahmad, A., 2004. Cluster center initialization algorithm for K-means clustering. Pattern Recognition Lett. .

Kövesi, B., Boucher, JM., Saoudi, S., 2001. Stochastic K-means algorithm for vector quantization. Pattern Recognition Lett. 22, 603-610.

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

Lletí, R., Ortiz, M.C., Sarabia, L.A., Sánchez, M.S., 2004. Selecting variables for k-means cluster analysis by using a genetic algorithm that optimises the silhouettes. Analytica Chimica Acta .

Maulik, U., Bandyopadhyay, S., 2000. Genetic algorithm-based clustering technique. Pattern Recognition Lett. 33, 1455-1465.

Michael K. Ng, 2000. A note on constrained k-means algorithms. Pattern Recognition Lett. 33, 515-519.

Ming, W.H., Hou, C.J., 2004. Cluster analysis and visualization. Workshop on Statistics and Machine Learning, Institute of Statistical Science, Academia Sinica.

Pearson, R. K., Zylkin, T., Schwaber, J.S., Gonye, G.E., 2004. Quantitative evaluation of clustering results using computational negative controls. Proc. 2004 SIAM International Conference on Data Mining, Lake Buena Vista, Florida.

Penã, J.M., Lozano, J.A., Larrañaga, P., 1999. An empirical comparison of four initilization methods for the K-means algorithm. Pattern Recognition Lett. 20, 1027-1040.

Ralambondrainy, H., 1995. A conceptual version of the K-means algorithm. Pattern Recognition Lett. 16, 1147-1157.

Ray, S., Turi, R.H., 1999. Determination of number of clusters in K-means clustering and application in colour image segmentation. 4th ICAPRDT Proc., pp.137-143.

Sun, Y., Zhu, Q., Chen, Z., 2002. An iterative initial-points refinement algorithm for categorical data clustering. Pattern Recognition Lett. 23, 875-884.

Tseng, L.Y., Yang, S.B., 2001. A genetic approach to the automatic clustering problem. Pattern Recognition 34, 415-424.

UCI Repository (http://www.sgi.com/tech/mlc/db/)

Veenman, C.J., Reinders, M.J.T., Backer, E., 2002. A maximum variance cluster algorithm. IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 24, no. 9, pp. 1273-1280.

Yi-tsuu, C., 1978. Interactive Pattern Recognition. Marcel Dekker Inc., New York and Basel.