Ali Ridho Barakbah, Afrida Helen, "Optimized K-means: an algorithm of initial centroids optimization for K-means",
Seminar on Soft Computing, Intelligent System, and Information Technology (SIIT) 2005, July 28-29, 2005,
Petra Christian University, Surabaya.

Seminar on Soft Computing, Intelligent System, and Information Technology (SIIT) 2005
July 28-29, 2005, Petra Christian University, Surabaya.

Optimized K-means: an algorithm of initial centroids optimization for K-means

Ali Ridho Barakbah and Afrida Helen
Electronics Engineering Polytechnic Institute of Surabaya - ITS
Email: ridho@eepis-its.edu, helen@eepis-its.edu

Abstract

Performance of K-means algorithm which depends highly on initial starting points can be trapped in local minima and led to incorrect clustering results. The lack of K-means algorithm that generates the initial centroids randomly does not consider the placement of them spreading in the feature space. In this paper we propose a new approach to optimize the initial centroids for K-means. This approach spreads the initial centroids in the feature space so that the distance among them are as far as possible. Started from the center of the data, this approach chooses each initial centroids those reside in distant position among them. The experimental results show the improved solution using the proposed approach.

Keywords: *clustering; K-means algorithm; initial centroids*

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 each other (internal homogeneity) and are not like members of other clusters (external homogeneity) [4, 11]. It means that the 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. The applications of clustering is diversely in many fields such as data mining, pattern recognition, image classification, biological sciences, marketing, city-planning, document retrievals, etc.

The most well known, widely used and fast methods for clustering is K-means clustering developed by Mac Queen in 1967. The simplicity of K-means clustering made this algorithm used in various fields. K-means clustering is a partitioning clustering method that separates data into k mutually excessive groups. Through such the iterative partitioning, K-means clustering minimizes the sum of distance from each data to its clusters. K-means clustering is very popular because of its ability to cluster a kind of 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 [5].

However, K-means clustering is very sensitive to the designated initial starting points as cluster centers. K-means clustering generates initial clusters randomly. If a randomly designated initial starting point close to a final cluster center, then K-means clustering can find the final cluster center. It, however is not always. If a designated initial point is far from the final cluster center, it will lead to incorrect clustering results [12]. Because of initial starting points generated randomly, K-means clustering does not guarantee the unique clustering results [9]. K-means clustering is difficult to reach global optimum, but only to one of local minima [2].

Several methods proposed to solve the cluster initialization for K-means clustering. A recursive method for initializing the means by running k clustering problems is discussed by Duda and Hart (1973). A variation of this method consists of taking the entire data into account and then randomly perturbing it k times [9]. Bradley and Fayyad (1998) proposed an algorithm that refines initial points by analyzing distribution of the data and probability of data density [7]. Penã et al. (1999) presented empirical comparison for fthe initialization methods for K-means clustering 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 clustering [6].

In this paper we propose a new approach to solve determination of optimized initial starting points for K-means, called as Optimized K-means. This approach optimizes the initial centroids for K-means by spreading the initial centroids in the feature space so that the distance among them are as far as possible. The experimental results will show the performance of this approach.

Ali Ridho Barakbah, Afrida Helen, "Optimized K-means: an algorithm of initial centroids optimization for K-means",
Seminar on Soft Computing, Intelligent System, and Information Technology (SIIT) 2005, July 28-29, 2005,
Petra Christian University, Surabaya.

2. Basic theory of K-means

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

K-means clustering algorithm can be described as follows:

1. Initiate its algorithm by generating random starting points of initial cluster centers c_k.
2. Calculate the distance $d(x,c)$ between vector x_i to cluster center c_k. Euclidean distance used to 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:

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

5. Go back to step 2 until $Ci = Ci\text{-}1$.

It may stop in the t iteration with a threshold ε [2] if cluster center has been updated by the distance below ε:

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

3. Optimized K-means

3.1. Basic concept

The ideal initial centroids reside near average gravity of the cluster members. It means that the grouping process of the members by K-means will collect them in the same cluster because the closeness of distance between the centroid and them. Nevertheless placement of initial centroids can also be set in the surrounding of the members as far as the members still consider the same centroid as nearest centroids. Figure 1 shows the same clusters even though the placement of centroid is not near average gravity of the cluster members.

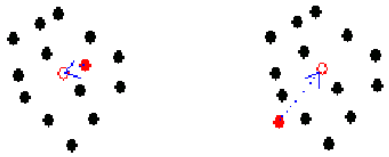

Figure 1. Different placements of initial centroid may make the same clusters.

The lack of K-means algorithm that generates the initial centroids randomly does not consider the placement of them spreading in the feature space. It makes the initial centroids may be placed closely so that one of them can be ignored. Therefore the initial centroids generated by K-means may be trapped in the local optima. We proposed in this paper how to place the initial centroids spreadly in the feature space.

3.2. Spreading the initial centroids

First of all we determine position m as the center of data in the feature space by calculating the average value of the data. Then we look for the nearest data to the center, called as c_1, and the distance between c_1 and m is called as d_1. We choose c_1 as the first initial centroid for K-means. Figure 2 performs the determination of first initial centroid.

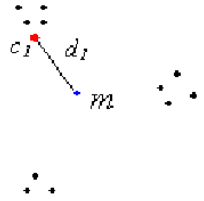

Figure 2. Determining the first initial centroid for K-means

Then, we look again for the second nearest data to m as the second initial centroids for K-means, called as c_2, and the distance between c_2 and m is called as d_2. In order to spread the initial centroids, we set rules for c_2 that fulfills $d_2 > d_1$ and $d(c_2, c_1) >= d_1$. It prevents choosing the second initial centroid near the first one. Figure 3 illustrates applying the rule to determine the second initial centroid.

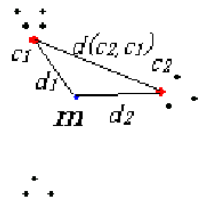

Figure 3. Illustration of applying the rule to determine the second initial centroid.

Then, we look again for the second nearest data to m as the second initial centroids for K-means, called as c_2, and the distance between c_2 and m is called as d_2. In order

Ali Ridho Barakbah, Afrida Helen, "Optimized K-means: an algorithm of initial centroids optimization for K-means",
Seminar on Soft Computing, Intelligent System, and Information Technology (SIIT) 2005, July 28-29, 2005,
Petra Christian University, Surabaya.

to spread the initial centroids, we set rules for c_2 that fulfills $d_2 > d_1$ and $d(c_2, c_1) >= d_1$. It prevents choosing the second initial centroid near the first one. Figure 3 illustrates applying the rule to determine the second initial centroid.

We apply the rule $d_i > d_p$ and $d(c_i, c_p) >= d_p$, where p=1…i-1, to determine c_i. Thus, the next initial centroids will scatter spreadly in the feature space.

3.3. Avoiding maximum d_m

If d_m is a maximum d_i and $m < n$, where n is number of clusters, the next initial centroids can not be determined. Let see Figure 4 to illustrate this condition.

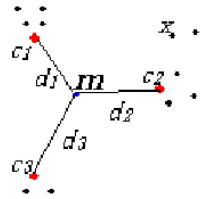

Figure 4. Condition when the next initial centroid can not be determined.

Let we assume number of clusters we want to set is 4. Shown in Figure 4, if we set x as c_4, it does not fulfill $c_4 - c_2 >= d_2$. Thus, we can not determine d_4 because it does not fulfill the rule.

To avoid this condition, we enhance the rule with involving the furthest distance in the feature space. Let d_m is the furthest d_i and $m<n$ where n=number of clusters, we accumulate the distance of x_i, where i=1..n, to d_j, where j=1..m. Then we look for the highest value of accumulated distance, called as d_{m+1}, and the data will be c_{m+1}. We can repeat it until number of clusters=n. Figure 5 performs the result to find the next initial centroid.

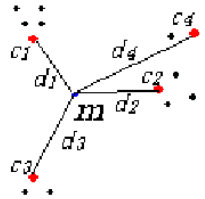

Figure 5. The result of avoiding maximum d_m

3.4. Algorithm

In this section, the following execution steps of Optimized K-means is proposed:

1. Determine position m as the center of data in the feature space.
2. Determine *totalcluster* and initialize $k = 0$.
3. Compute d_i, where i = 1..n, and select d_k as minimum d_i.
4. If $d_k > d_p$ and $d(c_k, c_p) >= d_p$, where p=1…k-1, determine c_k. Otherwise, go to step 6 to avoid maximum d_m, where m=k.
5. If k = *totalcluster*, finish. Otherwise, go to step 3.
6. Let d_m is the furthest d_i and m<*totalcluster*, then accumulate the distance of d_i to d_j, where i=1..*totalcluster* and j=1..m.
7. Find the highest value of accumulated distance, called as d_{m+1}, and the data will be c_{m+1}.
8. Increment m, and $k = m$.
9. If k = *totalcluster*, finish. Otherwise, go to step 6.

After process, it will generate the initial centroids c_k, where k=1, 2, ..., *totalcluster*. Then, we can apply it as initial cluster centers for K-means clustering. The experiment results will perform the accuracy of the proposed method.

4. Experimental results

We apply our proposed approach in this paper to solve clustering cases with random normal data distribution and local normal data distribution.

For random normal data distribution, we analyze the performance using variance ratio in the experiment. The following variance ratio, v is defined as a performance measure in the experiments. Variance constraint [3] can express the density of the clusters with variance within cluster and variance between clusters [8, 10]. The ideal cluster has minimum variance within clusters, called as v_w, to express internal homogeneity and maximum variance between clusters, called as v_b, to express external homogeneity [1]. The variance ratio can be determined as:

$$v = \frac{v_w}{v_b} x100\% \qquad (3)$$

For local normal data distribution, we analyze the performance using error ratio in the experiment. The error ratio is used for preclassified data in order to analyze the performance [9]. The error ratio can be determined as follows:

$$Error = \frac{Numberofmisclassified}{Numberofpatterns} x100\% \qquad (4)$$

Ali Ridho Barakbah, Afrida Helen, "Optimized K-means: an algorithm of initial centroids optimization for K-means",
Seminar on Soft Computing, Intelligent System, and Information Technology (SIIT) 2005, July 28-29, 2005,
Petra Christian University, Surabaya.

We made 1000 experiments for each data forms, and we compare the performance with K-means using random initialization. Table 1 performs the experiment results between our proposed approach and K-means using random initialization.

	Optimized K-means (%)	K-means using random init. (%)
Variance ratio	0.49556	0.91253
Error ratio	32.3	60.7

Table 1. The experiment results

Shown in Table 1 that our proposed method can improve the performance of initial centroids for K-means. Optimized K-means can reduce 54.3% of variance for random normal data distribution and 53.2% of error ratio for local normal data distribution.

We use Cluster Center Proximity Index (CCPI) to measure the degree of closeness between the initial cluster centers and the desired cluster centers [9] which defined as:

$$CCPI = \frac{1}{K * m} \sum_{s=1}^{K} \sum_{j=1}^{m} \left| \frac{f_{sj} - C_{sj}}{f_{sj}} \right| \qquad (5)$$

where f_{sj} is jth attribute value of the desired sth cluster center and C_{sj} is jth attribute value of the initial sth cluster center. Table 2 performs the comparison of average CCPI between K-means using random initialization and Optimized K-means for 1000 cases using local normal data distribution.

	Average CCPI
Optimized K-means	0.5581
K-means using random initialization	0.9268

Table 2. The comparison of average CCPI

Shown in Table 2 that Optimized K-means can improve 60.2% closeness of initial centroids. It means that our proposed approach can generate the closer initial centroids for K-means compared with random initialization.

4. Conclusion

We have presented a new approach, called as Optimized K-means, to determine the initial starting points for K-means algorithm. This approach optimizes the initial centroids for K-means by spreading them in the feature space so that the distance among them are as far as possible. From the experiments, the proposed approach can improve the performance of initial centroids for K-means. It can reduce 54.3% of variance for random normal data distribution and 53.2% of error ratio for local normal data distribution. Moreover, it can improve 60.2% closeness of initial centroids for K-means compared with random initialization.

References

[1] A. R. Barakbah, K. Arai, "Identifying moving variance to make automatic clustering for normal dataset", *Proc. IECI Japan Workshop 2004 (IJW 2004)*, Musashi Institute of Technology, Tokyo, 2004.

[2] B. Kövesi, J.M. Boucher, S. Saoudi, "Stochastic K-means algorithm for vector quantization", *Pattern Recognition Lett*, 22, 603-610, 2001.

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

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

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

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

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

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

[9] S.S. Khan, A. Ahmad, "Cluster center initialization algorithm for K-means clustering", *Pattern Recognition Lett* (Accepted), 2004.

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

[11] V.E. Castro, "Why so many clustering algorithms-a position paper", *ACM SIGKDD Explorations Newsletter*, Vol. 4, Issue 1, pp. 65-75, 2002.

[12] Y.M. Cheung, "k*-Means: A new generalized k-means clustering algorithm", *Pattern Recognition Lett*, 24, 2883-2893, 2003.