Hierarchical K-Means Clustering Algorithm Based on Silhouette and Entropy

Wuzhou Dong[1], JiaDong Ren[1], and Dongmei Zhang[2]

[1] College of Information Science and Engineering, Yanshan University, Qinhuangdao, China
[2] Qinhuangdao Port CO., LTD Qinhuangdao, China
dongwz@hebeea.edu.cn, jdren@ysu.edu.cn, zhangdongdongmei@126.com

Abstract. Hierarchical K-means clustering is one of important clustering task in data mining. In order to address the problem that the time complexity of the existing HK algorithms is high and most of algorithms are sensitive to noise, a hierarchical K-means clustering algorithm based on silhouette and entropy(HKSE) is put forward. In HKSE, the optimal cluster number is obtained through calculating the improved silhouette of the dataset to be clustered, so that time complexity can be reduced from O(n2) to $O(k \times n)$. Entropy is introduced in the hierarchical clustering phase as the similarity measurement avoiding distance calculation in order to reduce outlier effect on the cluster quality. In the post processing phase, the outlier cluster is identified by computing the weighted distance between clusters. Experiment results show that HKSE is efficient in reducing time complexity and sensitivity to noise.

Keywords: Hierarchical clustering, Silhouette, Entropy.

1 Introduction

Clustering is one of the analyzing method [1] in data mining and pattern recognition field. Up to now, researchers have proposed many clustering algorithms, among which the partition and hierarchical clustering methods [2] are most common.

K-means [3] is a representative classic partition algorithm. In this algorithm, K is obtained through minimizing objective function. K-means has higher efficiency compared with hierarchical method. However, the number of clusters K needs to be fixed iteratively. Common method trial-and-error depends on specific clustering algorithm. In addition, computation efficiency of trial-and-error is not high when determining the number of clusters. In order to address these problems, L.F. Chen proposed a method named COPS [4] based on hierarchical method to determine the optical number in large dataset clustering. In COPS, the optical number of clusters was obtained by constructing cluster quality curve. However, the parameter would affect the clustering results.

C.R. Lin pointed that K-Means had linear time complexity, but the clustering quality was not good. In order to improve the clustering efficiency, C.R. Lin presented a new algorithm CSM [5, 6], which combined K-Means with hierarchical

H. Deng et al. (Eds.): AICI 2011, Part I, LNAI 7002, pp. 339–347, 2011.

clustering method. Although CSM had better efficiency, K had to be fixed in advance. Sid LAMROUS introduced silhouette [7] as a measurement into the proposed divided hierarchical clustering algorithm, which was based on non-binary tree. Each node of the tree had m(m∈ [2,5]) sub-clusters. The node was divided by using K-means algorithm. For each node, the number of sub-cluster was determined through using silhouette. J.F. Lu proposed a K-means initialization method, which reduced and sampled dataset with weighted hierarchy structure. This method could find cluster centroid better, and it also extended clustering into high dimensional data space. However, the number of clusters K should be selected before clustering [8]. In order to address the problem that the cluster centroid and the number of clusters K should be selected before clustering, Chen proposed HK (Hierarchical K-means) algorithm [9]. In HK, the number of initial clusters and the centroid of cluster were firstly fixed by using agglomerative hierarchical clustering algorithm. Then the clustering was improved with K-means. HK had high time complexity in hierarchical clustering phase. In order to improve algorithm efficiency, W.C. Li proposed a method, in which cluster number was determined by using silhouette. This proposed method was superior compared with HK, but its time cost reached to O(n2). In addition, the proposed method was sensitive to noise [10].

In this paper, we propose HKSE, a new method for Hierarchical K-means Clustering algorithm based on silhouette and entropy. In HKSE, IS (Improved Silhouette), WDMC (Weighted Distance Matrix between Clusters) are defined. Optimal number of clusters is determined through computing the average Improved Silhouette of the dataset, so time complexity can be reduced. Entropy is introduced to HKSE as a similarity measurement to reduce sensitivity to noise. Clusters are weighted according to the size of clusters to improve the clustering quality.

This paper is organized as follows: In section II, we give the basic concepts and definitions. In section III, we present our hierarchical clustering algorithm called HKSE. Section IV shows the experimental results of the clustering algorithm. Finally we conclude the paper in section V.

2 Basic Concepts and Definitions

Definition 1. IS (Improved Silhouette).

Let S be a dataset consisting of clusters C1, C2…Ct. The distance between each object oi(oi∈ Cj, j∈ [1,t]) and the centroid of its own cluster is denoted as ai. bi is the minimum distance between oi and each centroid of the other t-1 clusters. The IS(oi) is defined as formula(1).

$$IS(o_i) = \frac{b_i - a_i}{\max(a_i, b_i)} \tag{1}$$

In formula(1), the meanings of ai and bi have changed compared with the traditional silhouette computation. Both ai and bi denote the distance to the cluster centroid.

The average IS of dataset corresponding to different partition is calculated. The maximal IS of the dataset corresponds to the optimal partition of the dataset.

We take point A in Fig. 1 as an example to show the IS computation of a data point.

(1) Obtain the centroids of clusters C1, C2, C3 respectively: Centroid1= $\left(\frac{1+1+1+2+2}{5}, \frac{0+1+1+2+2}{5}\right)$ = (1.4, 1.2), Centroid2=(4.4, 4.8), Centroid3=(6.5, 0.8333);

(2) Calculate aA, the distance between A and the centroid of its own cluster: $a_A = \sqrt{(1-1.4)^2 + (0-1.2)^2}$ =1.2649. The distances between A and each centroid of C2 and C3 can be obtained similarly, and they are 5.8822 and 5.5628 respectively. Since bA denotes the minimum distance according to the definition of IS, thus let bA=5.8822.

(3) The IS of A can be obtained based on formula

(4) IS(A)=(bA-aA)/max(aA, bA)=0.7850.

The IS values of other points can be calculated similarly.

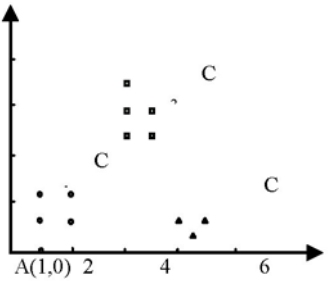

Fig. 1. Silhouette calculation process

Definition 2. WDMC(Weighted Distance Matrix between Clusters)

Let C1,C2,…,Ck be clusters of data set S. |Ci| is the number of data points in Ci. wi is assigned according to the data points number of Ci. wi is defined as formula(2).

$$\text{wi}=\frac{|C_i|}{|C_1|+|C_2|+...+|C_k|} \quad (\text{i}=1,2,\ldots,\text{k}) \tag{2}$$

Weighted distance between Ci and Cj is defined as formula(3).

$$\text{wdij}=\frac{1}{w_i}\sum_{j=1}^{k} \left|centroid_i - centroid_j\right| \tag{3}$$

Where wi=min(wi, wj), and $\left|centroid_i - centroid_j\right|$ is Euclidean distance between centroids of Ci and Cj.

For Ci(i=1,2,…,k), the weighted Euclidean distance between Ci and other k-1 cluster centroids is calculated. WDMC Mc can be obtained as formula(4).

$$\text{Mc}=\begin{bmatrix} wd_{11} & wd_{12} & \dots & wd_{1n} \\ wd_{21} & wd_{22} & \dots & wd_{2n} \\ \dots & \dots & \dots & \dots \\ wd_{n1} & \dots & \dots & wd_{nn} \end{bmatrix} (\text{wdij} \neq \text{wdji}) \tag{4}$$

WDMC reflects different importance of cluster through assigning different weight according to cluster size. For Ci, suppose its weight is small and distance between Ci and most clusters is large, the probability that Ci is outlier cluster become high. Thus, outlier cluster can be recognized through comparing weighted distance between clusters and distance threshold. wdij is not equal to wdji, for different cluster size has different effect to weighted distance.

3 Hierarchical K-Means Clustering Algorithm Based on Silhouette and Entropy

3.1 Find-K Algorithm

In HKSE, we plot the curve about the average IS of dataset to be clustered and the number of partitions. The optimal number of clusters is determined by the maximum of the curve, since the average IS of a dataset not only reflects the density of clusters, but also the dissimilarity between clusters.

Algorithm Find-K
Input: S
Output: K

```
begin
1: partition S into t clusters: C1, C2…Ct, according to the geometry distribution of S
2: repeat
3: {
4:   for (i=1; i<=t; i++)
5:     {for (each object x in Ci)
6:        {calculate ISi(x), the improved silhouette of x;
7:        calculate IS_i , the average Improved Silhouette of S, and IS_i = (1/n) Σ_{i=1}^{t} Σ_{x∈C_i} IS_i(x) ;
8:           plot the curve about t and IS_i in the 2-dimensional coordinate system;
9:         }
10:       t:=t+1;
11:    }
12: } until (the curve reaches the maximum)
13: K: =t;
end
```

In algorithm Find-K, dataset S is firstly partitioned into t clusters: C1, C2…Ct, according to the geometry distribution of S. IS is introduced into the algorithm

Find-K. In Find-K, The closer the improved silhouette of a cluster to 1, the more likely the objects belong to the same cluster. The curve, which is about cluster number t and the average IS, is plotted. The number of clusters corresponding to the maximum of the curve is the optimal number of clusters.

3.2 Outlier-C Algorithm

Cluster effect is positive to the number of data points that the cluster contains. In other words, the more data points the cluster contains, the bigger weight of the cluster will be. Outlier clusters always contain a small quantity of data points. In this paper, K+μ clusters are weighted. Outlier cluster can be recognized by comparing weighted distance between clusters with weighted distance threshold. Thus, clustering sensitivity to the outliers can be reduced.

Algorithm Outlier-C
Input: K+μ clusters
Output: K clusters

begin
1: While number of clusters is not equal to k do
2: Calculate weighted Eucliden distance between centroids of K+μ clusters respectively
3: Construct Mc for k+μ clusters /*Mc is weighted distance matrix for k+μ cluster */
4: Array the data of Mc in ascending sequence;
5: Put the arrayed data in a sequence list Q;
6: Take f as distance threshold between k+μ clusters.

$$f = \frac{\sum_{i=1}^{k+\mu}\sum_{j=1}^{k+\mu} wd(C_i, C_j)}{(\mathrm{k}+\mu)^2}$$

7: Find cluster Ci that has farthest distance to f.
8: Take Ci as critical cluster.
9: Dispose the clusters behind Ci in Q.
10: EndWhile
11: Return(k clusters)
end

θ, hierarchical clustering extent parameter, is introduced to Algorithm Outlier-C in order to dispose outlier clusters. Thus, the sensitivity to noise can be reduced. The value of μ is determined by θ.

In algorithm Outlier-C, the average weighted distance between clusters is taken as similarity degree threshold. Thus, these clusters, which are far more than threshold and have fewer data points compared with most of clusters, can be considered as outlier clusters. Each element value in Mc is arrayed. The problem of searching and comparing in matrix is converted into binary searching problem. The time efficiency of algorithm is improved.

3.3 HKSE Algorithm

Traditional clustering algorithms always adopt distance as similarity measurement. However, clustering quality of traditional clustering algorithm is not good, for distance is sensitive to outlier. Entropy, one of a measurement, can reflect the degree that the cluster is composed by data points of same category. The closer the entropy of a cluster is to 0, the bigger the purity of cluster belonging to same cluster will be. In HKSE, entropy is considered as similarity measurement. Agglomerative hierarchical clustering is instructed by the entropy change after combined with new data point. The parameters of HKSE are as follows: S is the dataset to be clustered; n is the number of data object; μ is a parameter; K is the number of clusters.

Algorithm HKSE
Input: S, n, μ
Output: K

begin
1: Call Find-K
2: label each data object in S as a single cluster
3: repeat
4: for each cluster C, Put C in other clusters
5: Calculate the incremental entropy of each cluster supposing the data object is joined
6: Assign the data object to the cluster whose incremental entropy is least
7: re-label the clusters obtained from step 6
6: n=n-1
7: until n=K+ μ
8: Call Outlier-C
end

In HKSE, the cluster combination is directed by the entropy calculation. Supposed O is a data point. Let Δi and Δj denote entropy increment after O is put in Ci and Cj respectively. If Δ i< Δ j, O is assigned to Ci, otherwise to Cj. Compared with algorithm HK, HKSE takes entropy as similarity measurement instead of distance. Our proposed algorithm considers data point density of cluster in order to reduce the sensitivity to noise.

4 Experimental Results

Our experiments are conducted on a computer with 2.4Ghz Intel CPU and 512M main memory. The operating system of the computer is Microsoft Windows XP.

HKSE is compared with HK to evaluate the performance of HKSE. All the algorithms are implemented in Visual C++.

We perform our experiments on UCI data sets iris, breast-cancer, credit-g and letter. The efficiency and performance of HKSE and NHK have been compared in our experiment. Parameter θ denotes the fulfillment extent of hierarchical clustering. The value of θ in traditional HK algorithm is in the range of [0.4,0.6] according to literature[11]. In this paper, our experiments are conducted under the premise that the value of θ is 0.5. Parameters of four data sets are shown in Table 1.

Table 1. Parameters of the testing sequence data sets

1. Dataset	1. name	1. Size	1. clusters
1. D1	2. Iris	2. 150	2. 3
1. D2	3. breast-cancer	3. 277	3. 2
1. D3	4. credit-g	4. 900	4. 2
1. D4	5. letter	5. 20000	5. 26

4.1 The Efficiency Analysis

We perform our experiments on UCI data sets D1, D2, D3 and D4. NHK and HKSE are carried out on the data sets respectively in order to compare algorithm performance. Running time result is shown in figure 2.

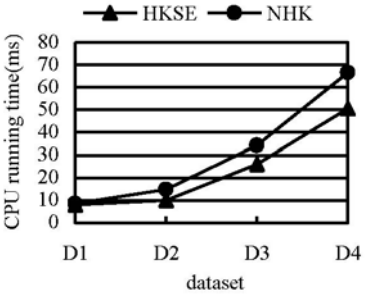

Fig. 2. HKSE and NHK comparison in terms of running time

In NHK, distance between each data point needs to be calculated in order to obtain the optimal cluster number. Thus, the time complexity of HK reaches to O(n2). In HKSE, silhouette is introduced to simplify the calculation of cluster number. We only need to calculate distance between each cluster centroid. The CPU running time is reduced largely. For k clusters, time complexity can be reduced to O(k×n). The time performance superiority is especially obvious when the data set scale is large.

4.2 The Accuracy Analysis

2%、4%、6%、8% and 10% noise are inserted into D3 respectively. After algorithms NHK and HKSE are ran with D3, the experiment results are shown in figure 3. Clustering quality is measurement of clustering algorithm. In the agglomerative hierarchical clustering phase of HKSE, entropy is adopted as similarity measurement instead of distance. Data object is combined according to entropy increment. Thus, distance calculation is avoided. Sensitivity to noise can be reduced. Thus, clustering result is improved.

From figure 3, we can draw the conclusion that both NHK and HKSE can obtain the correct clustering result when the cluster distribution is clear and noise is few in

data set to be clustered. The misclassification percentages of both algorithms are low,. However, when noises in the data set to be clustered become more, clustering with HKSE can obtain better result, for entropy is used to direct the combination of clusters without calculation of distance. Thus, the misclassification rate of HKSE is lower than that of NHK.

From 1% to 5% outlier data points are added in data set D1 respectively. HKSE and NHK are run in D1 respectively. The clustering result is shown in figure 4.

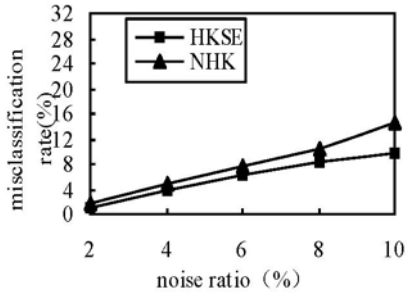

Fig. 3. Misclassification rate graph with the varing noise ratio

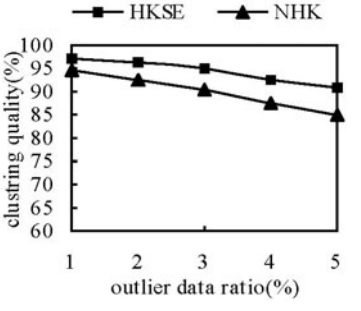

Fig. 4. The cluster quality graph with the varing outlier ratio

In post process of HKSE, clusters are weighted according to cluster size. Outlier cluster is recognized by comparing weighed distance between clusters and distance threshold. Thus, sensitivity to outlier data point of clustering is reduced. From figure 4, conclusion can be draw that HKSE is better than NHK in the ability of dealing with outlier data points.

5 Conclusions

Existing HK clustering algorithm has problem that time complexity is high and is sensitive to noise. In order to address the problem, HKSE is proposed in this paper. In HKSE, the IS curve is draw incrementally. The optimal cluster number is determined by maximum in IS curve. The time complexity can be reduced from O(n2) to O(k$\times$n). In the hierarchical clustering process, entropy is introduced as the similarity measurement in order to avoid distance calculation. Thus, the sensitivity to noise is reduced. In the post phase of HKSE, clusters are weighted according to cluster size. Experiments result shows that HKSE is efficient in reducing time complexity and sensitivity to noise.

Acknowledgments. This work is supported by the Natural Science Foundation of Hebei Province P. R. China under Grant No.F2010001298. The authors also gratefully acknowledge the valuable comments and suggestions of the reviewers, which have improved the presentation.

References

1. Dong, F.Y., Liu, J.j., Liu, B.: Study on improved grey integrated clustering method and its application. In: IEEE International Conference on Grey Systems and Intelligent Services, pp. 702–707 (2009)
2. Liu, L., Huang, L.H., Lai, M.Y.: Projective ART with buffers for the high dimensional space clustering and an application to discover stock. Associations Neurocomputing 72, 1283–1295 (2009)
3. Li, M.J., Ng, M.K., Cheung, Y.M.: Agglomerative fuzzy K-Means clustering algorithm with selection of number of clusters. IEEE Transactions on Knowledge and Data Engineering 20, 1519–1534 (2008)
4. Chen, L.F., Jiang, Q.S., Wang, S.R.: A Hierarchical Method for Determining the Number of Clusters. Journal of Software 19, 62–72 (2008)
5. Lin, C.R., Chen, M.S.: A Robust and Efficient Clustering Algorithm based on Cohesion Self-Merging. In: Inf. Conf. 8th ACM SIGKDD on Knowledge Discovery and Data Mining, pp. 582–587 (2002)
6. Lin, C.R., Chen, M.S.: Combining Partitional and Hierarchical Algorithms for Robust and Efficient Data Clustering with Cohesion Self-Merging. IEEE Transaction On Knowledge and Data Engineering 17, 145–159 (2005)
7. Lamrous, S., Taileb, M.: Divisive Hierarchical K-Means. In: International Conference on Computional Intelligence for Modeling Control and Automation, and International Conference on Intelligent Agent, Web Technologies and Internet Commerce, pp. 18–23 (2006)
8. Lu, J.F., Tang, J.B., Tang, Z.M.: Hierarchical initialization approach for K-Means clustering. Pattern Recognition Letters, 787–795 (2008)
9. Chen, T.S., Tsai, T.H., Chen, Y.T.: A Combined K-means and Hierarchical Clustering Method for Improving the Clustering Efficiency of Microarray. In: Proceeding of 2005 International Symposition on Intelligence Signal Processing and Communication System, pp. 405–408 (2005)
10. Li, W.C., Zhou, Y., Xia, S.X.: A Novel Clustering Algorithm Based on Hierarchical and K-means Clustering. In: Proceedings of the 26th Chinese Control Conference, pp. 605–609 (2007)
11. Chen, B., Tai, P.C., Harrison, R.: Novel Hybrid Hierarchical-K-means Clustering Method (H-K-means) for Microarray Analysis. In: Proceedings of the 2005 IEEE Computational Systems Bioinformatics Conference, pp. 105–108 (2005)