Journal of Southeast University (English Edition) Vol. 23, No. 3, pp. 435 – 438 Sept. 2007 ISSN 1003—7985

Improved *k*-means clustering algorithm

Xia Shixiong Li Wenchao Zhou Yong Zhang Lei Niu Qiang

(School of Computer Science and Technology, China University of Mining and Technology, Xuzhou 221008, China)

Abstract: In allusion to the disadvantage of having to obtain the number of clusters of data sets in advance and the sensitivity to selecting initial clustering centers in the *k*-means algorithm, an improved *k*-means clustering algorithm is proposed. First, the concept of a silhouette coefficient is introduced, and the optimal clustering number K_{opt} of a data set with unknown class information is confirmed by calculating the silhouette coefficient of objects in clusters under different K values. Then the distribution of the data set is obtained through hierarchical clustering and the initial clustering-centers are confirmed. Finally, the clustering is completed by the traditional *k*-means clustering. By the theoretical analysis, it is proved that the improved *k*-means clustering algorithm has proper computational complexity. The experimental results of IRIS testing data set show that the algorithm can distinguish different clusters reasonably and recognize the outliers efficiently, and the entropy generated by the algorithm is lower.

Key words: clustering; *k*-means algorithm; silhouette coefficient

As an important research branch of data mining, cluster analysis aims to divide data objects into groups based on their attributes and relations. And the objects have high similarity to one another within the same groups and have high dissimilarity to the objects in other groups[1].

The most well-known and commonly used clustering algorithms are *k*-means, *k*-medoids and their variations. The computational complexity of the *k*-means algorithm is low and the *k*-means algorithm may only find local optimum rather than the global[2]. Considering the innate limitation of the *k*-means method, this paper proposes an improved *k*-means method. The improved method can not only effectively decide appropriate number of clusters, but also properly select initial points for *k*-means. In addition, the improved *k*-means algorithm has good clustering results.

1 Criteria of Clustering

1.1 Entropy

Entropy depicts the dispersal of objects belonging to the same class being merged into different clusters. According to the distribution of classes, we can calculate the entropy of each cluster by[3]

$$e_i = -\sum_{j=1}^{L} p_{ij}\log_2 p_{ij} \tag{1}$$

where $p_{ij} = \frac{m_{ij}}{m_i}$ is the probability of objects of class j belonging to cluster i; m_i is the number of objects in cluster i; m_{ij} is the number of objects of class j in cluster i; and L is the number of classes. The total entropy of clusters is the weighted sum of each cluster's entropy, and is calculated by

$$e = \sum_{i=1}^{k} \frac{m_i}{m} e_i \tag{2}$$

where k is the number of clusters, and m is the number of objects in the data set.

1.2 Overall similarity

A good result of clustering should show dense and independent traits. Therefore, overall similarity adopts inner cohesion of clusters to estimate the quality of clustering, and is defined by

$$\text{similarity}_i = \frac{\sum_{x \in C_i} \text{dist}(x, c_i)}{m_i} \tag{3}$$

where C_i denotes cluster i, x is one object in C_i; c_i is the centroid of c_i; $\text{dist}(x, c_i)$ is the distance between x and c_i; m_i is the number of objects in cluster i.

Just as entropy, the total overall similarity can be calculated by the weighted sum of each cluster, namely,

$$\text{similarity} = \sum_{i=1}^{k} \frac{m_i}{m} \text{similarity}_i \tag{4}$$

1.3 Silhouette coefficient

The silhouette coefficient is a function that measures the similarity of an object with objects of their

Received 2007-05-18.

Foundation items: The National Natural Science Foundation of China (No. 50674086), Specialized Research Fund for the Doctoral Program of Higher Education (No. 20060290508), the Youth Scientific Research Foundation of China University of Mining and Technology (No. 2006A047).

Biography: Xia Shixiong(1961—), male, professor, xiasx@cumt.edu.cn.

clusters compared to the objects of other clusters. For object i, the value of the silhouette coefficient is defined by[4]

$$\text{silhouette}_i = \frac{b_i - a_i}{\max(a_i, b_i)} \tag{5}$$

where a_i is the mean of the distance between the object i and the objects of their clusters, and b_i is the minimum of the average distance between the object i and the objects in other clusters.

2 Improved *k*-Means Clustering Algorithm

2.1 Description and process of algorithm

The improved *k*-means clustering algorithm introduces the concept of the silhouette coefficient first, and computes the mean silhouette coefficient of all the clusters to obtain the optimal number of clusters K_{opt}. The *k*-means algorithm can trust in the results of the hierarchical algorithm completely. Instead of restarting *k*-means based on the initial information provided by the hierarchical algorithm, we just accomplish *k*-means based on the hierarchical algorithm. We set a threshold for *k*-means. If the distance between a certain object and the centroids of all the clusters exceeds the threshold, we consider this object to be an outlier; otherwise, we merge this object into the closest cluster and update the centroid of the cluster dynamically.

The process of the improved *k*-means algorithm is depicted as follows:

① Classify the original data set first, and calculate the silhouette coefficients respectively based on different K values. Then choose the K value corresponding to the maximum of the silhouette coefficient as the optimal cluster number K_{opt}.

② Repeat.

③ Choose two clusters (objects) with the closest distance and merge them into a new cluster.

④ Calculate the mean value of two clusters (objects) as the centroid of the new cluster.

⑤ Until remaining K_{opt} + REAR clusters in a data set.

⑥ Repeat.

⑦ Calculate each cluster's overall similarity according to Eq. (3).

⑧ Choose the cluster corresponding to minimal overall similarity. Then distribute objects in this cluster to other clusters which have the closest distance to objects, and update the centroid of clusters dynamically.

⑨ Until remaining K_{opt} clusters in data set.

⑩ Repeat.

⑪ Choose each object in the data set in turn.

⑫ If the object is already merged into K_{opt} clusters in step ⑨, then this object remains in the original cluster.

⑬ Else calculate the distances between this object and the centroids of existing K clusters.

⑭ If the distances exceed the threshold set by the user, then this object is considered as an outlier.

⑮ Else merge this object into the cluster with the closest distance and update the centroid of the cluster.

⑯ Until no object changes in the whole data set.

The improved *k*-means clustering algorithm refers to two parameters: REAR and E_{ps}. If we directly obtain K_{opt} clusters in step ⑤, this will cause several centroids of clusters to be scattered, so we terminate hierarchical clustering until remaining K_{opt} + REAR clusters. The value of REAR is set according to the percentage of hierarchical clustering. The goal of threshold E_{ps} is to recognize few outliers in the data set accurately.

2.2 Analysis of computational complexity of algorithm

The improved *k*-means clustering algorithm can be partitioned into three phases: Step ① finds the optimal number of clusters K_{opt}; steps ② to ⑨ adopt the agglomerate algorithm to produce initial information for *k*-means; steps ⑩ to ⑯ adopt *k*-means to accomplish a cluster. In the first phase, we need to calculate the silhouette coefficient repeatedly to obtain K_{opt}. So the computational complexity of this step is $O(I \times n)$, where I represents the times of calculating the silhouette coefficient and is not more than $\sqrt{n}$[5]. In the second phase, the computational complexity of hierarchical clustering is mainly focused on steps ② to ⑤. And steps ③ to ④ need to be executed $n - (K + \text{REAR})$ times repeatedly. Given certain iteration, step ③ needs to spend $O((n-i+1)^2)$ on searching for the proximity matrix, and step ④ needs to spend $O(n-i+1)$ on updating the proximity matrix. And the computational complexity of *k*-means in the third phase does not exceed $O(n^2)$. Therefore, the whole computational complexity of the improved *k*-means clustering algorithm is $O((n-C)n^2)$. If the proximity matrix is stored in an ordinal linked list, the expense of searching for the proximity matrix in step ③ can be reduced to $O(n-i+1)$. Considering the additional expense for maintaining the structure of the ordinal linked list, the final computational complexity of this algorithm is $O((n-C)n\log n)$, where $C = K_{\text{opt}}$ + REAR. Obviously, the computational complexity of the improved *k*-means clustering algorithm is lower than the hierarchical clustering algorithm $O(n^2 \log n)$.

3 Experimental Results and Analysis

3.1 Calculation of cluster number K_{opt}

Given that two-dimensional testing data set includes 600 objects (as shown in Fig. 1), we can obtain the proper number of clusters in a data set by calculating silhouette coefficient of the whole objects. In order to implement this easily, we calculate the silhouette coefficient of the centroid of each cluster instead of all of the objects in the cluster. We can, thus, obtain the mean silhouette coefficient of all the clusters according to formula (5), and use silhouette $(k) = \frac{1}{k}\sum_{i=1}^{k} \overline{\text{silhouette}}_i$ to obtain the silhouette coefficient of the whole data set. Fig. 2 shows an example of silhouette coefficient with allusion to different K values.

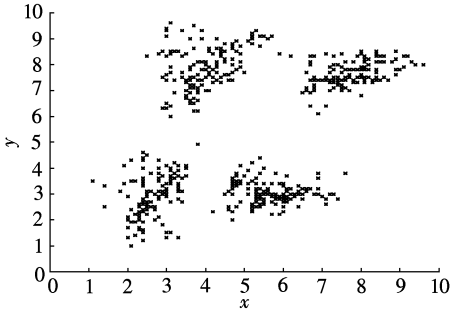

Fig. 1 Distribution of two-dimensional testing data set

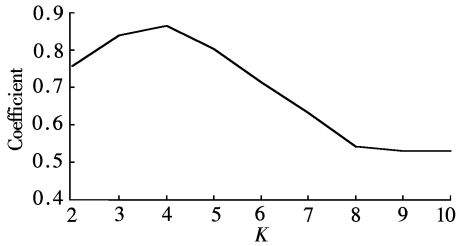

Fig. 2 The mean silhouette coefficient for different numbers of clusters

As shown in Fig. 2, when $K = 4$, the mean silhouette value 0.764 0 is the greatest. So we can conclude that this data set may consist of four clusters with the most possibility. At the same time, the chart has an obvious inflexion when $K = 8$. That is because the distance between the centroid of the cluster and the objects in other clusters is reduced rapidly, once splitting each physical cluster into two parts.

3.2 Experimental results

In this experiment, we choose the IRIS data set which is used specially to test clustering algorithms popularly[6]. And we compare the results of the improved *k*-means clustering with the traditional *k*-means clustering. Considering that it is a fact that IRIS consists of three classes of objects: Setosa, Versicolour and Virginica, so the improved *k*-means clustering algorithm executes directly from step ②.

In order to observe the effects of clustering distinctly, we choose the two most sensible attributes (petal length and petal width) from IRIS attributes and place them into the two-dimensional coordinate system. Then we adopt traditional and improved *k*-means algorithms to accomplish clustering respectively. In order to diminish the fluctuation of clustering results caused by the algorithm, we choose the best result from 10 different results generated by *k*-means. The improved *k*-means clustering algorithm just executes once. And we terminate hierarchical clustering at around 40% to 60%. In this experiment, we set 7 as REAR value, and 1.0 cm as our threshold. As shown in Tab. 1, we can contrast the entropy of clusters generated by different algorithms to compare their performances.

Tab. 1 The entropy of clusters generated by different algorithms

Algorithm	Entropy			
	Cluster 1	Cluster 2	Cluster 3	Weighted sum
k-means	0	0.591 7	0.266 8	0.301 2
Improved *k*-means	0	0.322 8	0.253 9	0.189 3

As shown in Fig. 3, the efforts generated by the improved *k*-means are obviously better than those generated by the traditional. This is because the initial centers are selected at random in the traditional *k*-means, but the improved *k*-means selects the initial centers de-

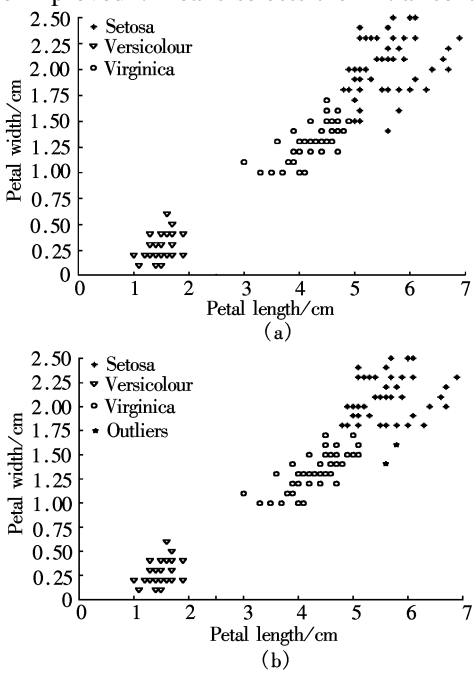

Fig. 3 The clusters generated by different clustering algorithms. (a) Clusters generated by *k*-means; (b) Clusters generated by improved *k*-means

pending on the distribution of data sets. Moreover, this algorithm can identify outliers effectively. Therefore, the results generated by the improved *k*-means clustering algorithm avoid the local optimum to ensure high stability.

4 Conclusion

After analyzing the drawbacks of *k*-means, this paper proposes an improved *k*-means algorithm. This algorithm conquers the diversity of result clusters, and optimizes the quality of clustering. Finally, this paper validates the efficiency of the algorithm by testing cases. When one object cannot be dispatched to a single cluster, we need to distribute the object to several clusters based on different possibilities.

References

[1] Han Jiawei, Kamber Micheline. *Data mining concepts and techniques* [M]. 2nd ed. Beijing: China Machine Press, 2001. (in Chinese)

[2] Xu Rui, Wunsch Ⅱ Donald. Survey of clustering algorithms [J]. *IEEE Transactions on Neural Networks*, 2005, **16**(3): 634 - 678.

[3] Su Ting, Dy Jennifer. A deterministic method for initializing *k*-means clustering[C]//*Proceedings of the* 16*th IEEE International Conference on Tools with Artificial Intelligence* (*ICTAI* 2004). Boca Raton, FL, USA, 2004: 784 - 786.

[4] Kanungo Tapas, Mount David M, Netanyahu Nathan S, et al. An efficient *k*-means clustering algorithm: analysis and implementation[J]. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2002, **24**(7): 881 - 892.

[5] Ramze R M, Lelieveldt B P F, Reiber J H C. A new cluster validity index for the fuzzy *c*-mean[J]. *Pattern Recognition Letters*, 1998, **19**(3/4): 237 - 246.

[6] Fisher R A. Iris plants database[EB/OL]. (1988-07)[2007-04-30]. http://www.ics.uci.edu/~mlearn/MLRepository.html.

一种改进的 *k*-means 聚类算法

夏士雄　李文超　周　勇　张　磊　牛　强

(中国矿业大学计算机科学与技术学院, 徐州 221008)

摘要:针对 *k*-means 算法事先必须获知聚类数目以及难以确定初始中心的缺点,提出了一种改进的 *k*-means 聚类算法. 首先引入轮廓系数的概念,通过计算不同 K 值下簇集中各对象的轮廓系数确定事先未知分类信息的数据集中所包含的最优聚类数 K_{opt};然后通过凝聚层次聚类的方法获得数据集的分布,确定初始聚类中心;最后利用传统的 *k*-means 方法完成聚类. 理论分析表明,所提出的算法具有适度的计算复杂度. IRIS 测试数据集的实验结果表明了该算法能够合理区分不同类型的簇集,且可以有效地识别离群点,聚合后的结果簇集具有较低的熵值.

关键词:聚类;*k*-means 算法;轮廓系数

中图分类号:TP18