An Improved K-means Method with Density Distribution Analysis

Huiwen Xue^{1,*}, Haochen Li¹, and Yanfei Wang¹

¹College of Science, China Agricultural University, Tsinghua East Road, Beijing, China

Abstract. In this paper, a novel K-means clustering algorithm is proposed. Before running the traditional K-means, the cluster centers should be randomly selected, which would influence the time cost and accuracy. To solve this problem, we utilize density distribution analysis in the traditional K-means. For a reasonable cluster, it should have a dense inside structure which means the points in the same cluster should tightly surround the center, while separated away from other cluster centers. Based on this assumption, two quantities are firstly introduced: the local density of cluster center ρ_i and its desperation degree δ_i , then some reasonable cluster centers candidates are selected from the original data. We performed our algorithm on three synthetic data and a real bank business data to evaluate its accuracy and efficiency. Comparing with Traditional K-means and K-means++, the results demonstrated that the improved method performs better.

1 Introduction

Today we are living in a world full of data. To store and manage the information we obtain every day, a large amount of dataset is generated. Thus, data analysis technology is needed. Cluster analysis plays an important and indispensable role in understanding data and has a long and rich history in variety of research fields. It is defined as “a statistical classification technique for discovering whether the individuals of a population fall into different groups by making quantitative comparisons of multiple characteristics according to Webster [1]”. One of the most popular and simplest clustering algorithms is K-means. It is proposed in 1955 and has been extended in many different ways in the past 50 years [2]. Although thousands of clustering algorithms have published today, traditional K-means is still widely used in many fields. Easily implementation, simplicity and empirical success are the main reasons for its popularity [1-3].

The traditional K-means method is a typical partitioned algorithm. It can work very well for compact and hyper spherical clusters. However, despite its wide application, K-means suffers from a serious limitation. Before running the algorithm, there are three parameters: the number of clusters, cluster centers initialization and distance metric, have to be determined firstly [1, 4]. Mostly, K-means is utilized with the Euclidean metric to compute the distance between points and cluster centers. This is why K-means perform well in detecting spherical or ball shape clusters while perform poorly in grouping arbitrary shape data. Although Minkowski distance, Hamming distance, sup distance and Person correlation are also applied in some research field, the experimental and practical results are not satisfied [2]. The reason is the researchers have not considered the relationship between different features in the data, but assume they are orthogonal. It makes the distance compute only based on

the difference in the same features while not in different features. This kind of assumption is not reasonable sometimes because in real data different features usually always have connection with each other. To solve this problem, proper prior process “distance metric learning” is needed [4]. Mahalanobis distance is a kind of metric used to detect hyper ellipsoidal cluster, covariance matrix of different features has to be calculated. Although for different data, it can learn its own Mahalanobis distance, this comes at high computational cost. Meanwhile, covariance describes the linear relationship between two variables, it is not universally for all kinds of data [2, 5].

Another important shortage that restricts the application of K-means is the initial cluster centroids and the number of clusters K must be decided before analysis [1-8]. The analysis results heavily depend on the initial positions of the cluster centers, thus if a bad initialization is chosen the objective function is easily getting trapped in poor local minima [7, 8]. In the meantime, cluster analysis is unsupervised learning which means there are no any category labels that tag objects with prior identifiers, thus it is very hard to find a proper K. For this issue, several heuristic methods have been given. However, the most widely used method is running the algorithm independently for different values of K and to find the most meaningful partition [5, 6, 7, 8]. It is a kind of dynamic strategy. The data is clustered from 1 group to N groups, here N is the total number of data points. When K is set differently, the partition is different. A reasonable K would be select according to domain knowledge or based on the evaluation criterion of the original data.

In this paper, we will focus on how to determine the parameter K in K-means. Equivalently, we will find the correct cluster centroids before applying K-means method for analysis of dataset. The ideal inspired from Rodriguez and Laio’s brilliant clustering method which is based on fast search and find of density peaks [3]. In the next

section, we will introduce the basic idea about how to determine the cluster centers. Then, in section 3 we will perform our new method on two benchmark dataset to test it efficiency and effectiveness. Finally, we apply this improved algorithm on a real bank business data to detect the underlying group structure.

2 Methods

2.1 Detection of cluster centers

As we mentioned above, the cluster centers should be the points that surrounded closely by their neighbors, this property describes the inside structure of a group is compact and the centers have very high local density [9, 10]. Meanwhile, centers in different clusters should be far away from each other which means they have a large relatively distance from each other. Based on this assumption, Rodriguez and Laio proposed clustering method by fast search and find of density peaks [3]. Here we introduce its basic idea and apply it to detect the cluster center. Let $X = \{x_i\}$, $i = 1, 2, \dots, n$, be the set of n dimensional points to be grouped. For each data point x_i , two quantities would be compute: its local density ρ_i and the distance δ_i from other points with higher density. To obtain these two values, we have to compute the distances d_{ij} between data points and set a threshold d_c . It's clear that varying d_c would affect ρ_i and δ_i . Empirically, we choose d_c so that the average number of neighbors of a point is 2% of the total number of points in the dataset. The local density ρ_i of point x_i is defined as:

$$\rho_i = \sum \chi(d_{ij} - d_c) \quad (1)$$

Here $\chi(x) = 1$ if $x < 0$ and $\chi(x) = 0$ otherwise. From this definition, ρ_i is actually equal to the number of points that are closer than d_c to point x_i . δ_i is obtain by computing the minimum distance between point x_i and any other point with higher density, the definition is as followed:

$$\delta_i = \min_{j: \rho_j > \rho_i} (d_{ij}) \quad (2)$$

We give both two values to each data point and plot them in a plan, then we will find the cluster centers [3]. The proper cluster centers should have peak values in both two quantities.

Fig.1. Four clusters centers in K-means data from MATLAB.

We use K-means data in MATLAB to show how powerful this method is. In “Fig. 1(b)”, clearly the four black squares have significantly high value in local density ρ and distance from other points with higher density. Thus, we decide that there are four cluster centers and the data should be grouped into four parts. The result is reasonable according to the introduction in MATLAB about this dataset which is shown in Fig. 1(a).

2.2 K-means algorithm

K-means is a typical partitioned algorithm. In this section we will introduce its basic idea. K-means algorithm finds a partition via minimizing the squared error between the empirical mean of a cluster and the points inside the cluster. Here, Let $C = \{c_k, k = 1, 2, \dots, K\}$ be the set of K clusters. Let v_k be the center of cluster c_k and usually it is computed as the mean of the all points in c_k . The squared error between v_k and all points is defined as an objective function:

$$J(c_k) = \sum_{x_i \in c_k} a_{ik} \|x_i - v_k\|^2 \quad (3)$$

where a_{ik} is equal to 1 if x_i is a point in cluster c_k and a_{ik} is 0 otherwise. The sum of the squared error over all K cluster is defined as:

$$J(C) = \sum_{k=1}^K \sum_{x_i \in c_k} a_{ik} \|x_i - v_k\|^2 \quad (4)$$

The goal of K-means is minimizing the sum of the squared error via finding the best a_{ik} for every point x_i . Minimizing the objective function above is known as a NP-hard problem. It can only converge to a local minimum. Usually, K-means begin with an initial partition with K clusters via select cluster centers randomly and then assign points to clusters so as to reduce the squared error. The main steps of K-means are as follows [1, 10]:

- 1) Select K initial cluster centers and begin computing $J(C)$
- 2) Generated a new partition via assigning each point to its closest cluster center.
- 3) Compute new mean of every cluster as the new cluster centers.
- 4) Repeat steps 2 and 3 until cluster centers do not move.

2.3 Silhouette value

We utilize silhouette measure [11] to assess the quality of clusters. To calculate the silhouette value $s(i)$ of any point, firstly we must estimate two scalars $a(x_i)$ and $b(x_i)$. Suppose point x_i belongs to cluster A , when cluster A contains other genes apart from x_i , then we can compute:

$$a(i) = \begin{matrix} \text{average distance of } x_i \text{ to} \\ \text{all other points of cluster } A. \end{matrix} \quad (5)$$

Then we consider any other cluster C which is different from A , and compute:

$$d(i, C) = \begin{matrix} \text{average distance of } x_i \text{ to} \\ \text{all points of cluster } C. \end{matrix} \quad (6)$$

After computing $d(i, C)$ for all cluster $C \neq A$, we select the smallest of those values and denote it by:

$$b(i) = \min\{d(i, C)\}, C \neq A \quad (7)$$

Suppose cluster B is the cluster which this minimum is obtained, that is, $d(i, B) = b(i)$, then we call it the neighbor of point x_i . Now $s(i)$ can be obtained by combining $a(i)$ and $b(i)$ as follows:

$$s(i) = \frac{b(i) - a(i)}{\max\{b(i) - a(i)\}} \quad (8)$$

From the above definition we can easily see that $s(i)$ locates in $[-1, 1]$, when $s(i)$ is close to 1, it implies that the ‘within’ distance $a(i)$ is much smaller than the smallest ‘between’ distance $b(i)$. Therefore, we can consider x_i is tight with its cluster and it is ‘well-clustered’. Another situation is $s(i)$ is around 0 which means $a(i)$ and $b(i)$ are almost equal, hence it is not clear whether x_i should belong to either cluster A or B . This situation is considered as an ‘intermediate case’. However, the worst situation is $s(i)$ is close to -1. It shows $a(i)$ is much larger than $b(i)$, thus gene x_i is much closer to B than to A . Therefore, we consider this is a “bad cluster”.

3 Results and discussion

In this part, we firstly test our combined method on three UCI clustering analysis benchmark data: Iris, Wine, Seed which are widely tested in different clustering methods [12–14] and make a comparison with traditional K-means and K-means++. Then we apply our method on a real bank business data: aiming to detect its potential group structure. Because the clustering structure of the bank business data is unknown, therefore we just analyze the data distribution itself while do not consider its real background.

3.1 Testing on benchmark data

We firstly test our approach on the four popular benchmark data. Specially, we will focus on the clustering analysis on Iris data and illustrate the analysis results in Fig. 2. As we know that Iris data is one of the most classical datasets for analysis of pattern recognition, classification and clustering. There are 150 samples and divided into 3 groups. Every sample has four features. In this analysis, we pretend that the number of clusters k is unknown in advance, thus we have to estimate its value by calculating ρ and δ for every sample. As shown in Fig. 2 (a), there are three extreme values of δ and these three samples are the three centers of the clusters. It is much clearly in Fig 2 (b) that the three cluster centers are sample 50, sample 70 and sample 121 respectively. Fig. 2(c) is the final cluster center position we obtained after running the traditional K-means method and our combined approach just for one time. Here, we pick two features sepal length and sepal width as the coordinates in the plane. The three different color points represent three different three clusters. The red round points are cluster center computed by our approach while the dark triangle points are cluster center obtained by the traditional K-means method. Apparently, as cluster centers, the red round points are more reasonable because they are located in the central position of the three groups. On the contrary, it is obviously wrong if the dark triangle points are considered as the center of cluster because two of the centers are the same group and it is unacceptable. Meanwhile, in Fig 2(d), the evaluation obtained from the distribution of silhouette value of each cluster suggests this improved K-means method is more effective because all silhouette value of samples is over zero. On the view of data distribution, there are no wrong groups.

Table 1 is the detail of analysis on the three benchmark data. We compared the three center initialization algorithm: traditional K-means algorithm, K-means ++ algorithm and the improved algorithm. The results suggest the new improved K-means algorithm perform much better than the two previous K-means algorithms. Firstly, in terms of computing time, compared with the traditional K-means algorithm, the improved algorithm saved about three quarters of the computing time; At the same time, improved algorithm also increased accuracy (especially in Wine data).

Fig. 2. Analysis on Iris Data

Table 1. Clustering analysis on four benchmark dataset

Data Sets	Size	K	Time(s) (Traditional/Kmeans++/Improved)	Accuracy (Traditional/ Kmeans++/Improved)	Silhouette Value (Traditional/ Kmeans++/Improved)
Iris	150	3	0.866/0.697/0.149	88.67% /89.33%/88.67%	0.7342/0.7355/0.7342
Wine	178	3	0.603/0.126/0.147	38.20%/38.20% /70.22%	0.7274/0.7274/0.7323
Seed	210	3	0.468/0.113/0.119	89.05% / 89.05%/89.52%	0.6584 /0.6584/0.6632

3.2 Analysis on bank customer data

Clustering analysis is widely applied in financial economics. In this part, we apply the improved K-means on bank business data to classify bank customer. In this data, 40 bank card customers are selected. Considering that when banks classify customers, they are more concerned with the features that can play a key role in cluster analysis. So, the following six features are selected to segment: age, gender, education, monthly income, monthly transaction amount and monthly transactions.

Table 2. Different values of the features

Age	Sex	Education	Monthly income	Monthly transaction amount	Monthly transactions
30	male	undergraduate	5000	3000	6
3	2	6	4	3	4

Firstly, for convenience, we combine the different values of the features in the range of 1-10. (Table 2) After the actual experiment, we find that the survey data divided into four groups is better than three ones, and the classification of the description data is also more detailed, so we suppose that the number of clusters is 4. Secondly, we still use the above improved ideas ---selecting the cluster center first, (Fig3 (a)) and then clustering the 40 samples. At last, the results were tested using the silhouette value method and the box-plot. In Fig3 (b), there is no silhouette value less than 0, so the classification of four clusters are very good, especially the third cluster and the fourth one, silhouette values of which are above 0.5, showing that the clustering results in theory is very reasonable.

Fig 3. Analysis on bank business data

For banks, the first cluster of customers belongs to the middle-aged groups. They are mostly well-educated with bachelor or master degrees; monthly income is stable in the 3000-8000 yuan and the monthly transaction amount are less than 5,000 yuan. The proportion of the first category is very large, which is the mid-range customer groups. Banks need to pay more attention to their

directions. The second cluster of customers belongs to the older groups. They are mainly male with high academic qualifications, high income and frequent transaction; however, their monthly amount of consumption is still less than million. The proportion of the second category is not high, which is the potential customer groups with the potential to rise to high-end customers. Banks may take measures to facilitate their conversion to high-end. The third cluster of customers are not high academic qualifications, their monthly income is very low, the amount of monthly transaction are about 3,000 yuan and the number of transaction is more than small. They belong to low-end customers. Although there is no potential value at present, the proportion of the group is still in a large number, so banks can be more publicity for this group. The fourth group of customers are highly educated with middle-age. Their monthly incomes are more than 20,000 yuan, the monthly transaction amount are more than 10,000 yuan, and monthly transaction are more than 10 times. They are high-end customers, although the proportion of the group is smaller, the banks should focus on providing them with personalized service.

4 Conclusions

In K-means cluster analysis, it is crucial to choose a proper cluster number K and proper cluster centers before applying the clustering algorithm. A proper K and correct cluster centers can make the iteration of the algorithm faster and more precise. To solve this issue, we propose an improved K-means method combine with the distribution analysis of dataset. The core idea of this approach is finding good cluster centers via computing two value of each point: its local density ρ_i and the distance δ_i from other points with higher density. For a good cluster center, we have the assumption that it should have both high values of ρ and δ . By this simple method, we find the number of clusters which also show the potential group structure of a dataset. We test this approach on three machine learning benchmark datasets. The experiment results suggest the improved method perform much better than the traditional K-means and K-means++. Meanwhile, we apply our method on clustering analysis of bank business data, it also performs well. Using this improved method, we successfully help the bank to classify the customers and give the bank some advice to manage the customers.

Acknowledgment

This work is supported in part by the University Students Innovation Fund of China Agricultural University.

References

1. Jain, A.K., "Data clustering: 50 years beyond K-means" *Pattern Recognition letters* 31, pp. 651-666 (2010)

2. Xu, R. and Wunsch, D., "Survey of Clustering Algorithms", *IEEE Transactions on Neural Networks*, Vol.16, No.3, pp:645-677 (2005)
3. Rodriguez, A. and Lai, A., "Clustering by fast search and find of density peaks", *Science* Vol **344** Issue 6191, pp. 1492-1496 (2014)
4. Schwammle V. and Jensen O. N., "A simple and fast method to determine the parameters for fuzzy c-means cluster analysis", *Bioinformatics*, Vol. **26**, No.22, pp: 2841-2848 (2011)
5. Tzortzis, C., and Likas, A., "The Min-Max k-Means clustering algorithm", *Pattern Recognition*, Vol. **47**, pp. 2505-2516 (2014)
6. Ball, G and Hall, D., "ISODATA, a novel method of data analysis and pattern classification.", *Technical report NTIS AD 699616*. Stanford Research Institute, Stanford, CA, 1965.
7. Kleinberg, J. "An impossibility theorem for clustering", In: *NIPS* **15**, pp. 463-470. (2002)
8. Wallace, C.S and Freeman, P.R., "Estimation and Inference by compact coding" *JRSSB*, **49**, , pp.240-251 (1987)
9. Fukuyama, Y. and Sugeno, M. "A new method of choosing the number of clusters for the fuzzy c-means method." *Proc. 5th fuzzy syst. Symp.*, PP. 247(1989)
10. MacQueen, J., "Some methods for classification and analysis of multivariate observations" In: *Fifth Berkeley Symposium on Mathematics, Statistics and Probability*. University of California Press, pp: 281-297 (1965)
11. Kaufman, L. and Rousseeuw, P.J. "Finding groups in data: an introduction to cluster analysis." *Wiley series in Probability and Statistics* (2005)
12. Mao, J. and Jain, A.K., "A self-organizing network for hyper-ellipsoidal clustering (HEC)", *IEEE Transactions on Neural Networks*, Vol. **7**, pp. 16-29 (1996)
13. Figureueredo, M. and Jain, A.K., "Unsupervised learning of finite mixture models" *IEEE Transaction on Pattern Analysis and Machine Intelligence*, Vol **24**, No. 3, pp. 381-396 (2002)
14. Tavazoie, S., Hughes, J.D., Campbell, m.J., Cho, R.J. and Church, G.M., "Systematics determination of genetic network architecture", *Nature Genetics*, Vol. **22**, 281-285 (1999)