A new algorithm for choosing initial cluster centers for k-means

Jiangang Qiao, Yonggang Lu*
School of Information Science and Engineering
Lanzhou University
Lanzhou, Gansu 730000, China
*Corresponding Author: ylu@lzu.edu.cn

***Abstract*—The k-means algorithm is widely used in many applications due to its simplicity and fast speed. However, its result is very sensitive to the initialization step: choosing initial cluster centers. Different initialization algorithms may lead to different clustering results and may also affect the convergence of the method. In this paper, we propose a new algorithm for improving the initialization of the cluster centers by reducing dimensions followed by moving cluster centers towards high density regions. Our algorithm is compared with three other initialization algorithms for k-means. And the effectiveness of our approach is shown by a series of carefully designed experiments.**

Keywords-K-means, cluster analysis

I. INTRODUCTION

Because of the importance of clustering algorithm in data mining and pattern recognition, many different clustering algorithms have been developed. With the fast development of technologies, such as data storage, camera acquisition, and medical equipment data acquisition, the increasingly inflation of the data, puts forwards the requirements for data clustering with high speed. Although the k-means algorithm has been proposed a long time ago [[1]], it is still the most widely used clustering method for its simplicity and fast speed which makes it a suitable method for clustering large amounts of data [[2]].

However, the clustering result of k-means is severely affected by the choice of initial cluster centers, especially in the case of a large number of clusters. The traditional k-means method chooses initial cluster centers arbitrarily, which may affect its accuracy in the clustering [[3], [4]].

K-means++ [[3]] is an improved version of k-means. The first center is chosen from the data set randomly, the other centers are chosen with a probability: the further a point is from the selected centers the larger the probability for the point to be chosen as a new center. The improvement not only speeds up the convergence of the clustering process but yields a better clustering result than k-means [[3]]. However, k-means++ may choose outliers or data points located at low density area as cluster centers, which may lead to sub-optimal solutions.

To overcome the disadvantages of the randomness in selecting initial cluster centers, another method for choosing cluster centers has been proposed by Murat Erisoglu, et al. [[4]]. This method first chooses two main dimensions that best represent the distribution of the dataset, and then computes Euclidean distances between each data point and the centroid of the data in the subspace defined by the two selected dimensions. The first cluster center is the data point with the longest distance from the centroid in the subspace. The *i*-th cluster center is the data point with the maximum combined distance from the previous (i-1) cluster centers. The algorithm has been shown to be effective in improving the k-means method when applied to some real data sets [[4]]. However, in our study of the algorithm, it is found that when it is applied to some synthetic data sets containing noise, the algorithm sometimes chooses noise data point far away from the centroid as the cluster center. Moreover, when the algorithm is launched multiple times on the same data set, it produces the same result.

To overcome the drawbacks of the available methods, we propose a new algorithm for initializing the cluster centers for k-means algorithm. After reducing the number of the dimensions of data sets, the candidate cluster centers are chosen similarly as k-means++, the final cluster centers are chosen by moving the candidate centers towards high density area.

The rest of the paper is organized as follows. The proposed algorithm for choosing initial cluster centers is introduced in Section 2. The experimental results are presented in Section 3. Conclusions are drawn in Section 4.

II. PROPOSED ALGORITHM

In this section, the proposed algorithm for choosing initial cluster centers is described. It consists of three parts.

A. Reducing the number of dimensions

First, in order to speed up the process of choosing the initial cluster centers, a two dimensional subspace is selected from the feature space, in other words, two main variables which are most representative for the original data are selected for initializing the cluster centers, which is similar to the method used by Murat Erisoglu, et al. [[4]]. The first variable is the one which has the maximum absolute value of the coefficient of variation, where the coefficient of variation is determined by

$$CV_j = \left| \frac{s(x_j)}{\overline{x_j}} \right|, j = 1,2, ..., p \qquad (1)$$

where $s(x_i)$ is the standard deviation, $\overline{x_j}$ is the mean of the *j*-th attribute variable, and p is the number of features. Then, we make use of the correlation coefficient of the variables to select the second main variable. The correlation coefficient is defined as

Published by Atlantis Press, Paris, France.
© the authors

$$CC_{jj'} = \frac{\sum_{i=1}^{n}(x_{ij} - \overline{x_j})(x_{ij'} - \overline{x_{j'}})}{\sum_{i=1}^{n}(x_{ij} - \overline{x_j})^2 \sum_{i=1}^{n}(x_{ij'} - \overline{x_{j'}})^2} \quad (2)$$

The lowest absolute value of correlation coefficient $CC_{jj'}$ means that the j-th and j'-th attribute columns are most independent. The variable which is most independent from the first main variable is selected as the second main variable. In the following, we will describe the cluster center selection process based on these two variables.

Similar ideas as those of k-means++ and Mean Shift algorithm [[5]] are used in determining initial cluster centers. Given a data set X, let v_1 denotes the first main variable selected in the previous step, v_2 the second. We then have a new data set X' consisting of only variables v_1 and v_2. A radius parameter R is first computed using the data in X', then the parameter R is used to choose the cluster centers.

B. *determination of a radius R:*

1) Randomly select 100 data points in X'.

2) For each data point, compute the distance between it and its nearest neighbor point in X'.

3) The radius R is four times of the maximum of the 100 distances computed in step 2.

C. *Choosing the candidate cluster centers*

1) Arbitrarily choose a data point in X' as the first candidate center, then calculate the mean of the data points within the circle centered at the candidate center and having the radius R. Similar to the Mean Shift method, a data point nearest from the mean is selected as the new position of the cluster center. This process is repeated until the center converges to a stable position.

2) Calculating the distances between all the data points and the cluster centers found previously. For each data point, the shortest distance from it to the centers is divided by the maximum distance from all the data to the centers, and the ratio of the distances are used as the probability for the data point to be selected as the next candidate cluster center. The candidate center is then chosen using the probability from all the data points, similar as k-means++.

3) The new candidate center is shifted similarly as the first candidate center until it converges to a new position.

4) If the new position found after shifting is not the same as the previous cluster centers, it is selected as the new cluster center. Otherwise, the original position of the candidate center before shifting is used.

5) Repeat Step 2 through Step 4 until all the k cluster centers are found.

5) Choosing k cluster centers in X which correspond to the selected cluster centers in X'.

III. EXPERIMENTS

A. *Comparison of the methods*

The random initial cluster centers in k-means method sometimes leads to bad clustering result. Similarly, the first cluster center of k-means++ is selected arbitrarily, so it can be an outlier or a data point located at low density area, which is also possible for other cluster centers.

The proposed algorithm utilizes a method similar to Mean Shift to move the candidate cluster centers to high density area, avoiding the issue of choosing cluster centers at low density area. Fig. 1 shows the shifting process. The four circles with a dot in the middle indicate the initial candidate centers, the four circles with asterisks in the middle represent the final cluster centers, and the other circles show the process of center shifting.

B. *Experiments on several data sets*

The performance of our proposed method is compared with k-means, k-means++ and a method proposed by Murat Erisoglu, et al. [[4]] on three synthetic data sets. Each data set is generated by Gaussian distribution function. FM Index [错误!未找到引用源。] is used to evaluate the clustering results of the four methods. By running each algorithm 100 times on the identical data set, the maximum and average of FM indices are recorded in Table I through Table III.

The first group of data sets is produced by changing the total number of data points while fixing the number of dimensions to 5 and the number of clusters to 8. The clustering results on this data set are showed in Table I. Our method produces the highest maximum FM indices when the number of data points is 5000 and 20000. The average FM indices from our method are the highest in the four algorithms applied on four data sets.

The second group of data sets is generated by varying the number of clusters and fixing the number of data points to 10000 in 5-dimension space. Table II shows the FM indices of the clustering results. The average FM indices of all the methods are decreasing with the increasing of the number of clusters. The maximum and the average FM indices produced by our method are higher than these of the other three algorithms when the number of the clusters reaches 10 and then 20. And it is noticed that the increase of the speed of our method is lower than the increase of the speed of Murat Erisoglu's method [4] for this group of data sets. Murat Erisoglu's method has best result when the number of the clusters is 5. Because Murat Erisoglu's method is not a randomized method, it always produces the same results for the same data set; it produces the best results for some data sets, while it doesn't for most of the other cases.

The third group of data sets is generated by varying the number of dimensions while fixing the total number of data points to 10000 and the number of clusters to 10. Table III shows the results for the data sets. Due to the increased distances between the centroids of different clusters included in four synthetic data sets in high-dimensional space, the maximum and average FM indices are increasing as the number of dimensions increasing. Our method has the best performance indicated by the average FM Indices on four data sets. Considered together, our method has a stable performance on a large part of tested data sets.

TABLE I. 5 DIMENSION, 8 CLUSTERS

The num of points		5000	10000	15000	20000
Time	Proposed algorithm	22.36	44.86	78.02	109.40
	K-means++ algorithm	8.95	18.16	38.73	56.36

© the authors

	Murat's algorithm	8.71	13.54	53.09	89.65
	K-means algorithm	5.67	19.17	36.29	53.67
Max FM	Proposed algorithm	0.8663	0.7813	0.8855	0.8112
	K-means++ algorithm	0.8651	0.7892	0.8855	0.8106
	Murat's algorithm	0.7280	0.6271	0.6450	0.6428
	K-means algorithm	0.8651	0.7857	0.8855	0.8106
Avg FM	Proposed algorithm	0.7309	0.6912	0.6625	0.6822
	K-means++ algorithm	0.7072	0.6863	0.6490	0.6685
	Murat's algorithm	0.7280	0.6271	0.6450	0.6428
	K-means algorithm	0.7059	0.6878	0.6584	0.6703

TABLE II. 10000 DATA POINTS, 5 DIMENSION

The num of clusters		5	8	10	20
Time	Proposed algorithm	27.40	35.24	37.20	63.27
	K-means++ algorithm	3.57	9.10	11.00	23.30
	Murat's algorithm	2.86	13.91	20.24	43.29
	K-means algorithm	3.13	8.20	10.20	17.30
Max FM	Proposed algorithm	0.9802	0.9728	0.9835	0.9682
	K-means++ algorithm	0.9802	0.9728	0.9835	0.9686
	Murat's algorithm	0.9800	0.8173	0.7494	0.7795
	K-means algorithm	0.9802	0.9728	0.9835	0.9680
Avg FM	Proposed algorithm	0.9576	0.8956	0.9039	0.8720
	K-means++ algorithm	0.9487	0.8954	0.8931	0.8542
	Murat's algorithm	0.9800	0.8173	0.7494	0.7795
	K-means algorithm	0.9468	0.8811	0.8560	0.8332

TABLE III. 10000 DATAPOINTS, 10 CLUSTERS

The num of Dim		2	5	10	15
Time	Proposed algorithm	41.43	42.23	50.93	54.42
	K-means++ algorithm	15.11	16.08	20.14	26.71
	Murat's algorithm	16.67	17.96	39.51	23.20
	K-means algorithm	13.57	14.83	19.15	20.12
Max FM	Proposed algorithm	0.8223	0.9017	0.9463	0.9243
	K-means++ algorithm	0.8224	0.9093	0.9507	0.9244
	Murat's algorithm	0.7658	0.8258	0.7352	0.8381
	K-means algorithm	0.8221	0.8985	0.9465	0.9243
Avg FM	Proposed algorithm	0.7955	0.8468	0.8391	0.8402
	K-means++ algorithm	0.7940	0.8455	0.8369	0.8363
	Murat's algorithm	0.7658	0.8258	0.7352	0.8381
	K-means algorithm	0.7859	0.8364	0.8175	0.8368

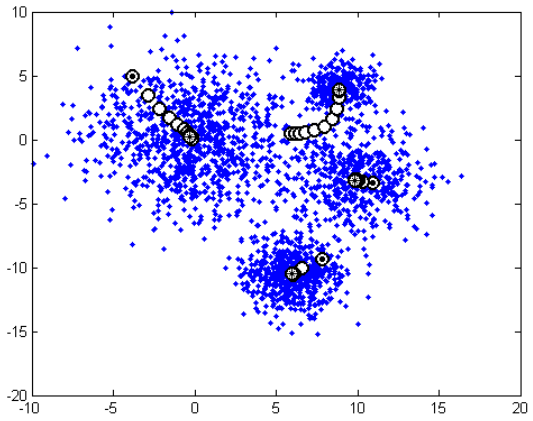

Figure 1. An example for shifting cluster centers in our method

IV. CONCLUSION

Because of the fast speed and wide applications of k-means, several algorithms are proposed in recent years to optimize the initialization of cluster centers in k-means. Most of the current methods may select initial cluster centers in low density region which may result in bad clustering results. So we propose a new method for choosing cluster centers using a combination of dimension reduction, selecting cluster centers with a probability, and followed by shifting the cluster centers towards high density area.

By reducing the number of dimensions to 2, we speed the process of the initialization. By shifting the candidate centers to high density area, the final clustering results can be improved. Experiments show the effectiveness of our method compared with three other methods. There are still problems with our proposed method: the speed of our method is not as good as k-means. So our future work will be concentrating on how to make the method more efficient.

V. ACKNOWLEDGMENT

This work is supported by the National Science Foundation of China (Grants No. 61272213).

VI. REFERENCES

[1] E. Forgey. Cluster Analysis of Multivariate Data: Efficiency vs. Interpretability of Classification. Biometrics, 21:768, 1965.

[2] Sariel Har-Peled and Bardia Sadri. How fast is the k-means method? In SODA '05, pages 877–885. 2005.

[3] Arthur, D., Vassilvitskii. k-Means++: The Advantages of Careful Seeding. Technical Report, Stanford. 2006.

© the authors

[4] MuratErisoglu, NazifCalis, Sadullah Sakallioglu. A new algorithm for initial cluster centers in k-means algorithm, Pattern Recognition Letters, 32, pp. 1701-1705. 2010.

[5] D. Comaniciu, and P. Meer, Mean shift: a robust approach toward feature space analysis, IEEE Trans. Pattern Analysis and Machine Intelligence, 24(5), pp. 603-619. 2002.

[6] Fowlkes, E. B.; Mallows, C. L.. "A Method for Comparing Two Hierarchical Clusterings". Journal of the American Statistical Association 78 (383): 553. (1 September 1983).

© the authors