Knowledge-Based Systems 145 (2018) 289–297

Contents lists available at ScienceDirect

Knowledge-Based Systems

journal homepage: www.elsevier.com/locate/knosys

Improved K-means algorithm based on density Canopy

Geng Zhang[a], Chengchang Zhang[b,*], Huayu Zhang[b]

[a] Information & communication Department, China Electric Power Research Institute, Beijing, China
[b] College of Electroning Engineering, Chongqing University of Posts and Telecommunications, Chongqing, China

ARTICLE INFO

Article history:
Received 11 June 2017
Revised 22 January 2018
Accepted 27 January 2018
Available online 31 January 2018

Keywords:
Density Canopy
K-means algorithm
Weight product
Optimal value K
Initial seed

ABSTRACT

In order to improve the accuracy and stability of K-means algorithm and solve the problem of determining the most appropriate number K of clusters and best initial seeds, an improved K-means algorithm based on density Canopy is proposed. Firstly, the density of sample data sets, the average sample distance in clusters and the distance between clusters are calculated, choosing the density maximum sampling point as the first cluster center and removing the density cluster from the data sets. Defining the product of sample density, the reciprocal of the average distance between the samples in the cluster, and the distance between the clusters as weight product, the other initial seeds is determined by the maximum weight product in the remaining data sets until the data sets is empty. The density Canopy is used as the preprocessing procedure of K-means and its result is used as the cluster number and initial clustering center of K-means algorithm. Finally, the new algorithm is tested on some well-known data sets from UCI machine learning repository and on some simulated data sets with different proportions of noise samples. The simulation results show that the improved K-means algorithm based on density Canopy achieves better clustering results and is insensitive to noisy data compared to the traditional K-means algorithm, the Canopy-based K-means algorithm, Semi-supervised K-means++ algorithm and K-means-u* algorithm. The clustering accuracy of the proposed K-means algorithm based on density Canopy is improved by 30.7%, 6.1%, 5.3% and 3.7% on average on UCI data sets, and improved by 44.3%, 3.6%, 9.6% and 8.9% on the simulated data sets with noise signal respectively. With the increase of the noise ratio, the noise immunity of the new algorithm is more obvious, when the noise ratio reached 30%, the accuracy rate is improved 50% and 6% compared to the traditional K-means algorithm and the Canopy-based K-means algorithm.

1. Introduction

Clustering algorithm, which is one of the most classical algorithms of data mining, has been researched by many scholars. Clustering technology is widely used in many fields. In the commercial field, it can be used to analyze customers' behavior, providing an important basis for the development of commercial marketing strategy [1,2]. Besides, in the field of internet e-commerce, it can be used to analyze the characteristics of similar customers according to the user's browsing logs, so as to help internet merchants to provide better customer service [3]. In addition, clustering analysis has an important application in data mining for big data on smart grid user side [4–6]. By mining the effective information in user's electricity data and analyzing the user's electricity using behavior, the power consumption forecast is carried out. It is of great significance for grid companies to carry on the electric power dispatching [7,8]. According to the different clustering methods, the clustering algorithm can be divided into division-based method, hierarchical-based method, density-based method, mesh-based and model-based method [9].

K-means algorithm is a commonly used clustering algorithm based on division method [10–12], its procedure is simple and efficient, suiting for clustering analysis of big data sets. It uses distance as the similarity to divide the sample into several clusters. Within the same cluster, the similarity among samples is higher, and the dissimilarity among samples in different clusters is higher. At present, K-means algorithm mainly has two problems, which are the determination of cluster number (value K) and the selection of the initial clustering centers. Therefore, the research work of K-means algorithm is mainly focused on the above two aspects. By dividing the original data sets into several optimal subsets and selecting the initial clustering center in each subset, the authors gives a method of division clustering in [13], although the method improves the accuracy of clustering, it increases the complexity of

* Corresponding author.
E-mail addresses: zhanggeng@epri.sgcc.com.cn (G. Zhang), zhangcc@cqupt.edu.cn (C. Zhang).

https://doi.org/10.1016/j.knosys.2018.01.031
0950-7051/© 2018 Elsevier B.V. All rights reserved.

the algorithm, which is not suitable for clustering analysis of big datasets. A data sampling and K-means pre-clustering method has been proposed in [14], through multiple data sampling and generating a clustering result by K-means algorithm respectively, the clustering results are calculated intersection and constructed the weighted connected graph to obtain the clustering center. However, the method lacks the consideration of the overall sample distribution of the data sets, having some limitations and instability. Besides, a method of determining the upper limit of cluster number K by AP algorithm [15] is proposed in [16], but the specific method of determining the optimal K value is not given. Mao Dianhui proposes a method that Canopy algorithm and K-means algorithm are combined to determine the clustering input parameters in [17], using the maximum and minimum distance method [18,19] to solve the problem of determining the threshold T_1 and T_2 in Canopy clustering. However, the immunity of the algorithm to noise is weak. In addition, an improved method named Semi-supervised K-means++ algorithm was proposed in 2016 [20]. By marking up some of the data firstly, the rest was labeled according to the minimum cost, and the expected result can be received by account for the labels. But choosing the suitable labeled data, which has a certain impact on the final clustering results, is not easy. Therefore, the new method has some limits. Moreover, Fritzke proposed the K-means-u* algorithm to improve the limits of K-means++ algorithm in 2017 [21], however, it increases the complexity of the algorithm greatly, not suiting for the scenes having large amount of data.

Therefore, a new Canopy clustering method based on the density of samples is proposed in this paper. The optimal value K of the data sets and the initial clustering center are obtained by density Canopy algorithm, which are used as the input parameters of the K-means algorithm, solving the two difficult problems: the determination of value K and the selection of the initial clustering center [22,23]. The simulation tests on UCI website datasets [24] and simulated data sets with noise, show that the K-means clustering method based on new density Canopy can obtain better clustering results, at the same time, it is more robust to noise rejection.

This paper is organized as follows. In Section II, the improved K-means algorithm based on density Canopy is presented. In Section III, the simulation and the results are presented and discussed. Finally in Section IV, the relevant conclusions are drawn.

2. K-means algorithm based on density Canopy

2.1. Canopy algorithm principle

The canopy algorithm is an unsupervised pre-clustering algorithm introduced by McCallum et al. [25], It is often used as preprocessing steps for the K-means algorithm or the Hierarchical clustering algorithm. As shown in Fig. 1, Canopy algorithm sets two distance thresholds T_1 and T_2, selects the initial cluster center randomly, and calculates the Euclidean Distance between sample and initial center. The sample will be classified into the corresponding cluster according to thresholds. Finally, the clustering data sets are divided into n clusters. The cluster number and the clustering center of Canopy algorithm are used as the input parameters of K-means algorithm to complete the data sets clustering.

As shown in Algorithm 1, the flow of the Canopy algorithm is as follows:

Step 1 Giving the datasets $D = \{x_1, x_2, ..., x_n\}$, setting the thresholds T_1 and T_2, where $T_1 > T_2$.

Step 2 Taking the sample S from the data sets D and calculating the Euclidean Distance d between the remaining sample points and point S respectively. If $d < T_1$, the point will be added to the current Canopy sets.

Step 3 It completes the task of comparing the calculated distance d with T_2. If $d < T_2$, the sample point is removed from the data sets D, which is no longer added to the other Canopy sets.

Step 4 Repeating Step2 and Step3 until D is empty.

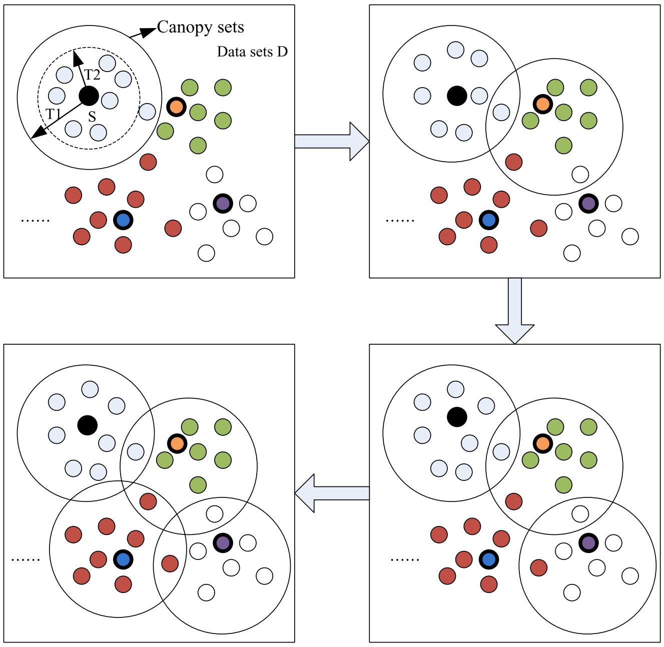
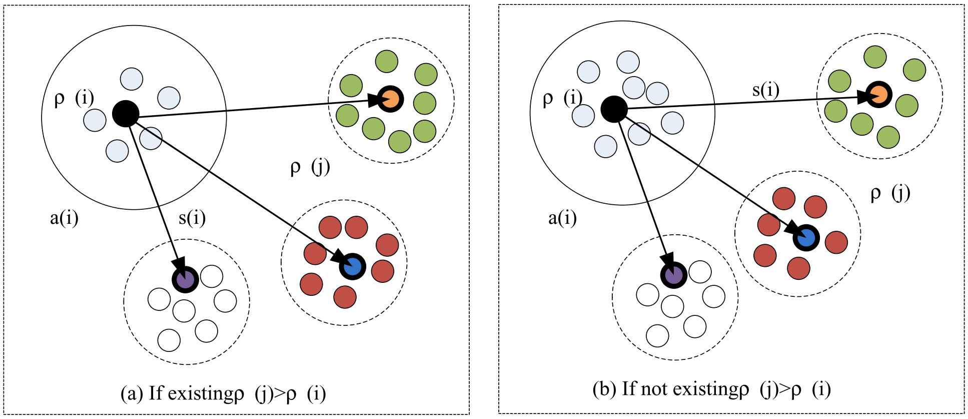


Fig. 1. The principle of the Canopy algorithm.

Algorithm 1 Canopy clustering algorithm.

```
Input: Data sets D
Output: The cluster number K and initial cluster centers of datasets
initialize the ArrayList and set $T_1$, $T_2$
select sample S randomly
FOR(each sample $a \in D$){
    IF(data sets D!=null)
      {compute $d$;
        IF($d < T_1$) {
            Canopy sets $C_i \leftarrow$ sample $a$;
            };
        ELSE IF($d < T_2$) {
            remove sample $a$;
            }
        }
      ELSE {
          break;
        }
  }
END FOR
PRINTF(value K, Initial Center);
```

The thresholds T_1 and T_2 are difficult to be determined in Canopy algorithm, and the value of the threshold has great influence on clustering results. If the threshold value is too large, the data that belongs to different classes will be incorrectly grouped into the same class. If it is too small, the data that belongs to the same class will be divided into several classes. Next, a new Canopy algorithm with density parameters is proposed to solve this problem.



(a) If existingρ (j)>ρ (i)

(b) If not existingρ (j)>ρ (i)

Fig. 2. Diagram of maximum weight method.

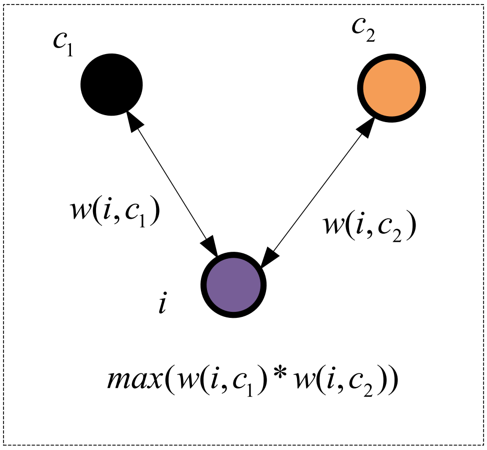


Fig. 3. Diagram of maximum weight method to obtain the best clustering centers.

2.2. Density Canopy algorithm

2.2.1. Basic concept

For the giving data sets $D = \{x_1, x_2, ..., x_n\}$, the sample element m in D is denoted as $x_m = \{x_{m1}, x_{m2}, ..., x_{mr}\}, 1 \le m \le n,$ where r is attribute numbers of x_m, and $d(x_p,\ x_q)$ represents the Euclidean Distance between two elements $x_p = \{x_{p1}, x_{p2}, ..., x_{pr}\}$ and $x_q = \{x_{q1}, x_{q2}, ..., x_{qr}\}$.

Definition 1. The average distance of all sample elements in datasets D is defined as:

$$MeanDis(D) = \frac{2}{n(n-1)} \sum_{i=1}^{n} \sum_{j=i+1}^{n} d(x_i, x_j) \tag{1}$$

Definition 2. The density of the sample element i in datasets D is defined as:

$$\rho(i) = \sum_{j=1}^{n} f\left[d_{ij} - MeanDis(D)\right] \tag{2}$$

where $f(x) = \{\begin{matrix}1, x < 0\\ 0, x \ge 0\end{matrix};$

Definition 3. According to Eq. (2), $\rho(i)$ is the number of samples meeting the condition that the distance from other samples to point i is less than $MeanDis(D)$. The samples that meet the condition form a cluster, and the average distance between samples in cluster is defined as:

$$a(i) = \frac{2}{\rho(i)[\rho(i) - 1]} \sum_{i=1}^{\rho(i)} \sum_{j=i+1}^{\rho(i)} d(x_i, x_j) \tag{3}$$

Definition 4. The clusters distance $s(i)$ represents the distance between the sample element i and the other sample element j with a higher local density. If the local density of sample points i is the maximum, $s(i)$ is defined as $max\{d(i,\ j)\}$; if existing $\rho(j) > \rho(i)$, it is defined as $\underset{j:\rho(j)>\rho(i)}{min}\{d(i,\ j)\}$, that is:

$$s(i) = \begin{cases} \underset{j:\rho(j)>\rho(i)}{min}\{d(i,\ j)\}, \exists j, \rho(j) > \rho(i) \\ max\{d(i,\ j)\}, \nexists j, \rho(j) > \rho(i) \end{cases} \tag{4}$$

Definition 5. The datasets D is divided into k clusters. The center of the cluster $C_j (j \le k)$ is c_j. The sum of squared errors of the clustering result is the sum of squared distance between each cluster's samples and the center of their cluster. That is:

$$E = \sum_{i=1}^{n} \sum_{j=1}^{k} (x_i - c_j)^2, x_i \in C_j \tag{5}$$

2.2.2. Maximum weight product method

Definition 6. The product of $\rho(i)$, $\frac{1}{a(i)}$ and $s(i)$ is defined as product weight. That is:

$$w = \rho(i) * \frac{1}{a(i)} * s(i) \tag{6}$$

The thresholds are selected randomly in traditional Canopy algorithm, and it has great impact on clustering results. This paper proposes maximum weight product method, which can reduce the instability caused by the randomness and improve clustering accuracy.

The maximum weight product method is as shown in Fig. 2. At first, the density of the samples is calculated according to Eq. (2), setting the maximum value of the density as the first cluster center, and adding all sample points satisfying the conditions that the distance between the sample and the initial cluster center is less than $MeanDis(D)$ in Definition 2 to the current cluster, and removing those samples from the data sets. Besides, the weight product w of remaining elements is calculated according to Eqs. (2)–(4) and (6), finding the maximum value and selecting the corresponding sample as the second cluster center. Finally, repeat the above steps until the datasets D is empty.

In addition, the larger the value $\rho(i)$, the more the elements around the point i and the more concentrated the elements. The smaller the value $a(i)$, the greater the value $1/a(i)$, the tighter the elements in a cluster. The greater the value $s(i)$, the greater the degree of dissimilarity between the two clusters.

The most appropriate number of cluster in optimal partition will be determined according to Maximum weight product method proposed in this paper. The specific steps are as follows:

Step 1 Giving the Data sets, the density of all samples is calculated referring to Eq. (2), choosing the maximum density sample c_1 as the first clustering center, and the center c_1 is added to the sets C, which is $C = \{c_1\}$. At the same time, all the samples that satisfying the condition that the distance between the remaining samples and the first clustering center is less than MeanDis(D), are removing from Data sets.

Step 2 Calculate the $\rho(i)$, $a(i)$ and $s(i)$ of the samples in the rest of Data sets, and the second clustering center c_2 will be determined according to the Maximum weight product method, and the center is added to the sets C as well, therefore $C = \{c_1, c_2\}$. Similarly, all the samples that satisfying the prescript condition are removing from Datasets.

Step 3 Calculate the distance between the remaining samples and each points in the sets C, if satisfying the condition: $max(w(i,\ c_1)^*w(i,\ c_2))$, the point i will be set to the third clustering center c_3 and add it to the sets C. Removing all the samples that satisfying the prescript condition as well. The diagram of maximum weight method to obtain the best clustering centers is as shown in Fig. 3.

Step 4 Similarly, if sample j satisfying the condition: $max(w(i, c_1) * w(i, c_2) * \ldots * w(i, c_{k-1}))$, the point j will be set the kth clustering center. Remove all the samples that satisfying the prescript condition.

Step 5 Repeat the step4 until the Data sets is empty.

Finally, the datasets will be divided into several subsets referring to Maximum weight product method proposed in paper. Calculating the average value of all samples in each subsets as the clustering center. Therefore, the most appropriate number of cluster in optimal partition will be determined.

The density-based method is insensitive to noisy data. The possible outliers could be found and removed by $\rho(i)$ and $s(i)$. For outliers, it has the characteristics of discrete, low density and deviating from the normal samples. Therefore, when $\rho(i)$ is small and $s(i)$ is large, the sample point is considered as an abnormal point. The removal of abnormal noise points can guarantee the accuracy of clustering, so as to enhance the stability of clustering.

2.3. Density Canopy-based K-means algorithm

In this paper, an improved K-means algorithm based on density Canopy is proposed. Firstly, the data sets is pre-clustered by density Canopy algorithm, obtaining the optimal value K and the initial clustering center as K-means algorithm input parameters, then the flow of K-means algorithm is executed.

As shown in Algorithm 2, the improved algorithm performs as follows:

Step 1 It completes the determination of Value K and the initial cluster center from line 1 to 19. The optimal K and the initial clustering center are obtained by optimizing the density Canopy algorithm.

Step 2 It performs the Clustering task from line 21 to 27. Calculate the Euclidean distance between samples in the remaining data sets and the initial cluster center, and add the samples to the cluster of the corresponding cluster center in the light of the minimum distance principle.

Step 3 It completes the calculation of the new Clustering center from line 30 to 31. Calculate the average distance of elements in cluster, and update it as the new cluster center;

Step 4 Comparing the updated cluster center to original one, if there is no change, the algorithm is terminated, getting the final result of clustering, otherwise, the algorithm goes to Step 2.

The algorithm flow is shown in Fig. 4.

Algorithm 2 Density Canopy-based K-means algorithm.

```
Input: Data sets D
Output: Clustering results of data sets
1. initialize the ArrayList;
2. compute MeanDis(D);
3.      FOR(each sample $i \in D$){
4.        compute $\rho(i)$;
5.        }
6.         WHILE(data sets D!=null){
7.        //
8.         select Center $c_1 \leftarrow$sample Max$\rho(i)$;
9.         remove Cluster $C_1$;
10.          FOR((each sample $i \in D$)&&(cluster $C_i \notin D$)){
11.         compute $a(i)$;
12.          compute $s(i)$;
13.                IF(MaxWeight $w = \rho(i) * 1/a(i) * s(i)$
14.                   Center $c_i \leftarrow$sample (MaxWeight $w$) $i$;
15.                remove Cluster $C_i$;
16.                }
17.                        }
18.         END FOR;
19.         PRINTF(Value K, Initial Center);
20.         //
21.         K-means input(Value K, Initial Center);
22.         WHILE(NEW Center== Original Center){
23.         FOR(each sample $i \in D$){
24.         compute $d_{ik}$(sample $i$ to Center $c_k$);
25.         IF (MinDis($k$) = $d_{ik}$){
26.          Center $c_k \leftarrow$sample $i$;
27.         }
28.                }
29.         END FOR;
30.          compute NEW Center $c_i$=
31.                Mean(sample ($i$ &&($i \in Cluster\ C_i$)));
32.         }
33. PRINTF(Cluster $C_i$);
```

Table 1
Parameters of UCI data sets.

Data sets	Data size	Attributes	Cluster number
Soybean-small	47	35	4
Iris	150	4	3
Wine	178	13	3
Segmentation	2310	19	7
Ionoshpere	351	34	2
Pima	768	8	2
Segmentation-T	13,000	19	7

3. Simulation and discussion

3.1. UCI data sets simulation experiment

The experimental data in this section are derived from the UCI website, selecting the following seven testing data sets: Soybean-small, Iris, Wine, Segmentation, Ionoshpere, Pima Indians Diabetes and Segmentation-T.

As shown in Table 1, each data set has different number of samples, and each sample has different number of attributes, through which to test the effectiveness of the improved algorithm. The segmentation-T is the data sets adding a certain amount of analog value on the basis of the segmentation, which is used to test the clustering effect of the improved algorithm for larger data sets.

The clustering effect is measured by the following parameters: the time required to complete the clustering, the sum of squared

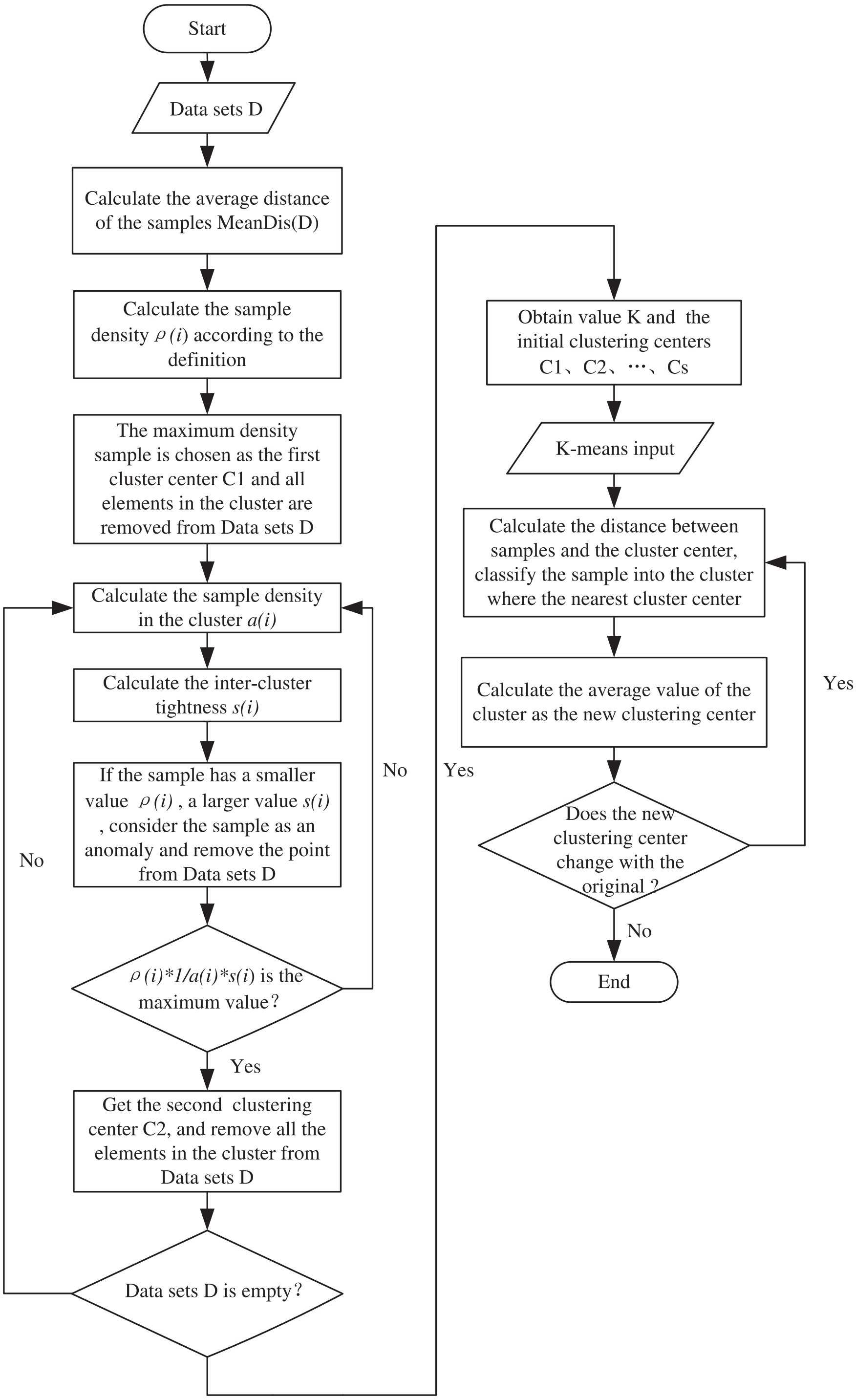

Start
Data sets D
Calculate the average distance of the samples MeanDis(D)
Calculate the sample density ρ(i) according to the definition
The maximum density sample is chosen as the first cluster center C1 and all elements in the cluster are removed from Data sets D
Calculate the sample density in the cluster a(i)
Calculate the inter-cluster tightness s(i)
If the sample has a smaller value ρ(i), a larger value s(i), consider the sample as an anomaly and remove the point from Data sets D
ρ(i)*1/a(i)*s(i) is the maximum value?
No
Yes
Get the second clustering center C2, and remove all the elements in the cluster from Data sets D
Data sets D is empty?
No
Yes
Obtain value K and the initial clustering centers C1、C2、…、Cs
K-means input
Calculate the distance between samples and the cluster center, classify the sample into the cluster where the nearest cluster center
Calculate the average value of the cluster as the new clustering center
Does the new clustering center change with the original ?
Yes
No
End

Fig. 4. Algorithm flow chart.

errors of the clustering results (Calculated according to Eq. (5)), the accuracy of the clustering results, and the three parameters that measure the validity of the clustering: Rand Index, Jaccard coefficient, and Adjust Rand Index. It is a comparison of the sum of squared errors of clustering results and the clustering time on UCI data sets using five different clustering algorithms (Traditional K-means algorithm, K-means algorithm based on Canopy, K-means algorithm based on density Canopy, Semi-supervised K-means++ algorithm, K-means-u* algorithm) in Table 2.

From the data analysis and comparison in Table 2, the following conclusions can be drawn:

(1) Traditional K-means algorithm, Semi-supervised K-means++ algorithm and K-means-u* algorithm has the longer time to complete data clustering. As the traditional method selects the initial center randomly, the number of iterations is large when the algorithm reaches stable, so the execution time is longer, moreover, Semi-supervised K-means++ algorithm and K-means-u* algorithm have a higher algorithm complexity, therefore, the execution time is also longer. The K-means clustering using Canopy algorithm as data preprocessing is obviously superior to the traditional K-means algorithm. Because the algorithm achieves the input parameters of K-means through Canopy al-

Table 2
The clustering time T (s) and the sum of squared errors of the clustering results on UCI data sets.

Data sets	Traditional K-means algorithm		The K-means algorithm based on Canopy		The K-means algorithm based on density Canopy	
	T	E	T	E	T	E
Soybean-small	6.012	246.233	3.961	207.76	1.833	200.21
Iris	7.019	97.21	4.938	79.65	2.828	75.32
Wine	7.013	2.39E + 06	5.963	2.35E + 06	2.864	2.33E + 06
Segmentation	8.115	3.22E + 06	6.425	2.92E + 06	3.319	2.89E + 06
Ionoshpere	9.081	2.71E + 03	7.891	2.36E + 03	4.726	2.28E + 03
Pima Indians Diabetes	8.150	5.63E + 06	6.621	5.13E + 06	3.991	5.11E + 06
Segmentation-T	33.105	1.61E + 07	29.75	1.55E + 07	25.76	1.41E + 07

Data sets	Semi-supervised K-means±+		K-means-u* algorithm	
	T	E	T	E
Soybean-small	7.124	203.122	9.267	202.46
Iris	8.264	78.43	10.934	75.19
Wine	8.175	2.34E + 06	10.586	2.33E + 06
Segmentation	9.514	2.91E + 06	11.855	2.9E + 06
Ionoshpere	10.268	2.35E + 03	12.873	2.33E + 03
Pima Indians Diabetes	9.586	5.1E + 06	13.863	5.08E + 06
Segmentation-T	35.248	1.51E + 07	40.753	1.49E + 07

Table 3
Parameters of the data sets.

	Cluster A	Cluster B	Cluster C
Means	$u_X^A = 0$, $u_Y^A = 0$	$u_X^B = 5$, $u_Y^B = 1$	$u_X^C = 5$, $u_Y^C = -2$
Standard deviation	$\sigma_x = \sigma_y = 2$	$\sigma_x = 1, \sigma_y = 2$	$\sigma_x = \sigma_y = 1$

gorithm, then complete the clustering of the data sets and the number of iterations is less when the algorithm reaches stable, therefore it is more efficient than traditional K-means algorithm.

(2) In terms of the sum of squared errors of the clustering results, the clustering effect of density Canopy-based K-means algorithm is the best. The traditional K-means selects the initial cluster center randomly, with the largest squared errors and the worst clustering result.

The comparison of the parameters that measure the clustering results is shown in Fig. 5.

The comparison of the parameters of clustering results in Fig. 5 shows that the three parameters of new algorithm are optimal and the accuracy is the highest. Moreover, the clustering accuracy is 30.7 percent above traditional K-means algorithm, 6.1 percent above K-means algorithm based on Canopy, 5.3 percent above Semi-supervised K-means++ algorithm and 3.7 percent above K-means-u* algorithm.

In this paper, K-means algorithm based on density Canopy is proposed. By calculating the density of datasets, the most compact cluster in datasets is found. The initial clustering centers are determined by the maximum weight product method, and the best value K is determined. This method takes the distribution of all samples into consideration, making clustering more objective. At the same time, it solves the problem of the traditional Canopy algorithm that the threshold T_1 and T_2 are difficult to determine. Thus, the clustering results of the new algorithm are more accurate, the convergence rate is faster, achieving the global optimization of clustering.

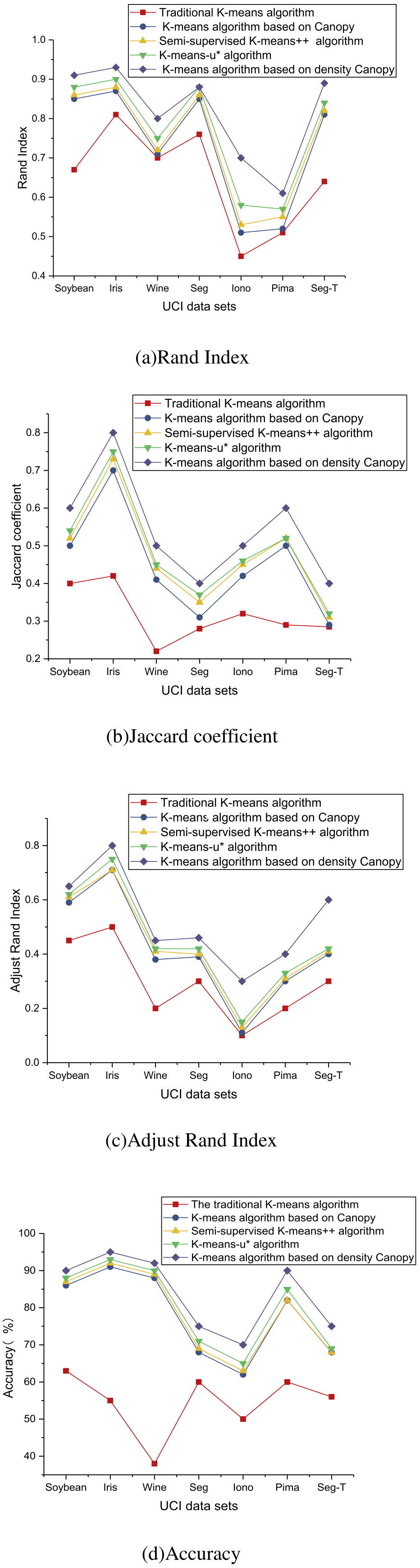
3.2. Simulated data experiment

This section uses artificial two-dimensional sample data which is added different proportions noise signal (0%, 5%, 10%, 15%, 20%, 25%, 30%) for verification experiments. It has seven data sets in total, testing the noise immunity of the proposed al-

Table 4
The clustering time T (s) and the sum of squared errors of the clustering results on simulated datasets.

Noise ratio /%	Traditional K-means algorithm		The K-means algorithm based on Canopy		The K-means algorithm based on density Canopy	
	T	E	T	E	T	E
0	8.023	1026.21	7.162	735.73	2.967	628.17
5	9.019	1045.37	7.026	864.28	3.879	742.61
10	10.018	1021.12	9.023	802.56	4.865	686.15
15	11.022	1498.73	10.011	1087.39	5.753	900.18
20	13.029	1149.46	12.189	1098.57	5.855	916.73
25	16.031	1003.88	14.021	1001.51	7.783	896.26
30	20.025	1056.73	17.09	1016.26	7.882	928.64

Noise ratio /%	Semi-supervised K-means±+		K-means-u* algorithm	
	T	E	T	E
0	8.156	728.31	9.753	715.34
5	10.567	855.72	12.683	846.25
15	14.238	1055.28	16.862	1047.35
20	15.398	1056.21	18.364	1008.42
25	17.881	995.26	19.245	987.26
30	23.831	1000.86	25.866	975.29



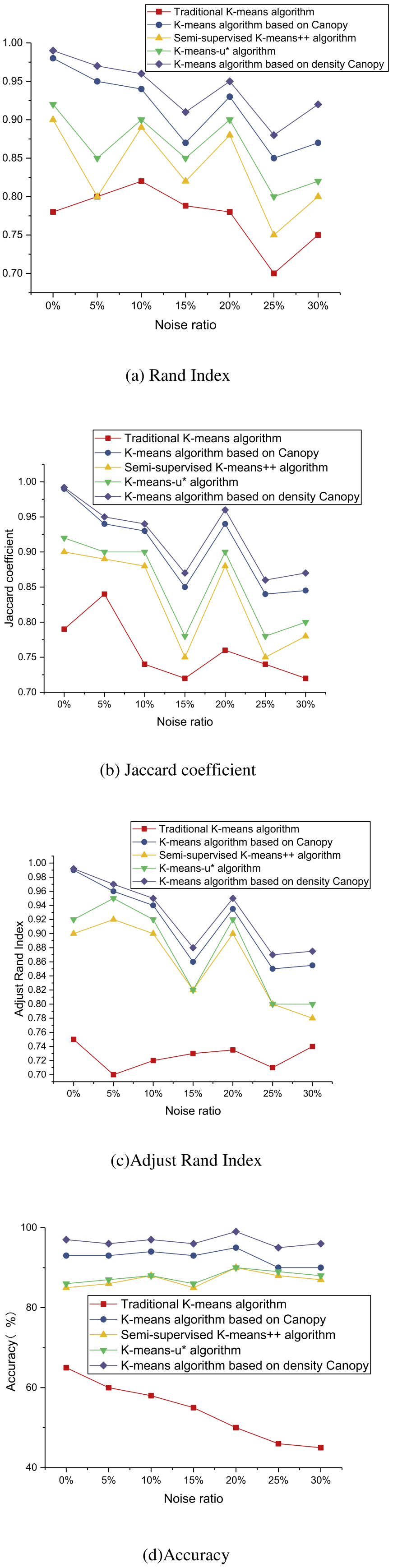
(a)Rand Index

(b)Jaccard coefficient

(c)Adjust Rand Index

(d)Accuracy

Fig. 5. Clustering results of UCI datasets.



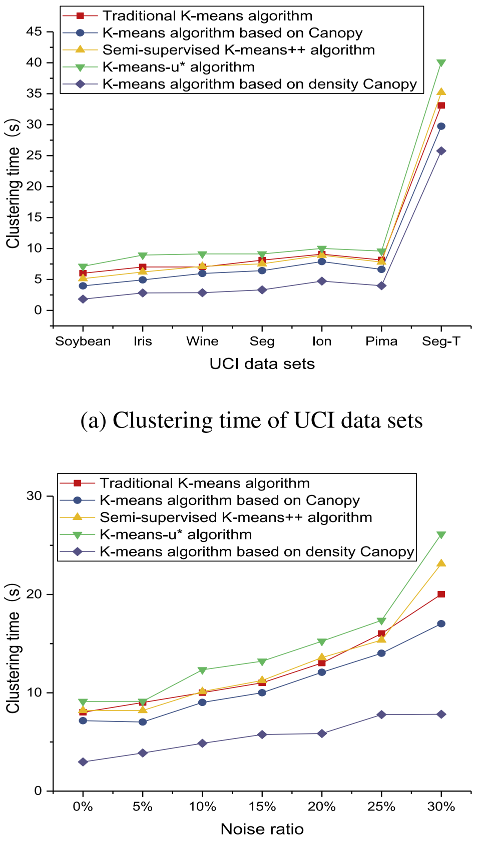
(a) Rand Index

(b) Jaccard coefficient

(c)Adjust Rand Index

(d)Accuracy

Fig. 6. Clustering results of simulated datasets.



(a) Clustering time of UCI data sets

(b) Clustering time of simulated datasets

Fig. 7. The comparison of clustering time.

gorithm. Data elements (X,Y) obey the normal distribution, $(X,Y) \sim N(u_1, u_2, \sigma_1^2, \sigma_2^2, \rho)$, where $\rho = 0$, then $X \sim N(u_1, \sigma_1^2)$, $Y \sim N(u_2, \sigma_2^2)$. Each data set is divided into three categories, and each category has 100 sample elements. The relevant parameters of the data sets are shown in Table 3. The data sets of cluster B is added noise signal, and the noise data sets follows the normal distribution: $(X_c, Y_c) \sim N(5, 1, 2, 2, 0)$.

Five algorithms (Traditional K-means algorithm, K-means algorithm based on Canopy and K-means algorithm based on density Canopy, Semi-supervised K-means++ algorithm, K-means-u* algorithm) are carried out in the seven data sets with different proportion of noise data respectively, getting the relevant simulation results. The clustering time and the sum of squared errors of clustering results on the simulated data sets are as shown in Table 3.

The simulation results in Table 4 show that the traditional K-means algorithm, Semi-supervised K-means++ algorithm and K-means-u* algorithm have longer the clustering time. The K-means algorithm based on density Canopy is slightly better than the K-means algorithm based on Canopy. In terms of the sum of squared errors of the clustering results, the new algorithm is also optimal, the convergence rate is the fastest and its anti-noise interference performance is the best.

The comparison of parameters that measure the clustering results is as shown in Fig. 6, including Rand index, Jaccard coefficient, Adjust Rand Index and clustering accuracy.

By comparing these curves in Fig. 6 can be seen that the K-means algorithm based on density Canopy has the best clustering effect for the data sets containing noise signal. The accuracy rate is 44.3 percent above the traditional K-means algorithm, 3.6 percent above K-means algorithm based on Canopy, 9.6 percent above Semi-supervised K-means++ algorithm, 8.9 percent above K-means-u* algorithm and the anti-noise performance of the new algorithm is more obvious as the noise ratio increases. When the noise ratio reaches 30 per cent, the clustering accuracy of the new algorithm is 6 percent above K-means algorithm based on Canopy.

In a word, the results show that the clustering effect based on density Canopy is the best, the accuracy is the highest and the anti-noise performance is the best for the data sets with noise signal.

3.3. Algorithm complexity discussions

In view of the traditional K-means algorithm, the time complexity of the algorithm can be expressed as $O(n\mathrm{KT})$, where n is the number of samples in the data sets, K is the number of classes of the data sets, and T is the number of cycling times of algorithm. The time complexity of K-means clustering algorithm based on density Canopy in this paper can be expressed as $O(n^2 + n\mathrm{S} + n\mathrm{K}t)$, where t is the number of iterations of K-means algorithm, S is the number of cycling times of density Canopy algorithm getting value K and initial cluster center, $O(n^2)$ is the time complexity of the improved density Canopy algorithm, and $\mathrm{S} < t$. So, the time complexity of the density Canopy algorithm is mainly determined by the number of data sets: n. The time complexity of the K-means clustering algorithm based on density Canopy is simplified to $O(n^2 + n\mathrm{K}t)$. The input parameters of K-means are the best in this paper, so the number of iterations of K-means algorithm is less than the traditional K-means algorithm. Therefore, when dealing with small data sets, the proposed algorithm can still have perfect time performance. When the data set increases to a certain extent, the time complexity of the algorithm will be determined mainly by $O(n^2)$.

The comparison of clustering time of five algorithms on UCI and simulated data sets is shown as in Fig. 7.

4. Conclusions and future work

K-means algorithm is one of the most typical methods of data mining. Aiming at the two disadvantages about the determination of the value K and initial clustering center in traditional K-means algorithm, an improved K-means algorithm based on density Canopy is proposed in this paper. In the improved algorithm, the density parameter is added. By defining the density of the samples in the data sets, the average distance between the samples in the cluster and the distance between the clusters, the value K and the initial clustering center of the clustering are obtained according to the proposed maximum weight product method, taking them as the input parameters of the K-means algorithm, which can improve the accuracy of clustering. At the same time, the introduction of density parameters in the algorithm enhances the anti-noise performance of the algorithm and ensures the reliability of the algorithm. Finally, the new algorithm, the traditional K-means algorithm, the K-means algorithm based on Canopy, the Semi-supervised K-means++ algorithm, the K-means-u* algorithm are simulated on UCI and the simulation datasets respectively. The simulation results show that the new algorithm has higher accuracy, better clustering effect and stronger anti-noise ability according to the comparison of Rand index, Jaccard coefficient, Adjust Rand Index, the accuracy rate and the sum of squared errors of the clustering results.

The future work will focus on dimensionality reduction for high-dimensional data sets and the parallelization of algorithms. Today is a Big data Era, high-dimensional data is ubiquitous, although it has more accurate description for the target object, it presents a huge challenge to data processing. Data dimensionality reduction for achieving more accurate and fast data clustering is very important. At the same time, when dealing with big data sets, how to combine Hadoop parallelization technology to improve the efficiency of the algorithm is worth further exploration.

Acknowledgment

This work was supported in part by Scientific Research Project of State Grid Corporation of China(JS71-16-001), National Natural Science Foundation of China (61001089), Scientific and Technological Research Program of Chongqing Municipal Education Commission(KJ1500407), and Youth Science Foundation of Chongqing University of Posts and Telecommunications (A2014-107).

References

[1] C. Aguwa, M.H. Olya, L. Monplaisir, Modeling of fuzzy-based voice of customer for business decision analytics, Knowl.-Based Syst. 125 (6) (2017) 136–145.

[2] D.L. García, Àngela Nebot, A. Vellido, Intelligent data analysis approaches to churn as a business problem: a survey, Knowl. Inf. Syst. 51 (3) (2017) 719–774.

[3] G. Vinodhini, R.M. Chandrasekaran, A sampling based sentiment mining approach for e-commerce applications, Inf. Process. Manage. 53 (1) (2017) 223–236.

[4] F. Mcloughlin, A. Duffy, M. Conlon, A clustering approach to domestic electricity load profile characterization using smart metering data, Appl. Energy 141 (2015) 190–199.

[5] A. Al-Wakeel, J. Wu, N. Jenkins, K-means based load estimation of domestic smart meter measurements, Appl. Energy (2016).

[6] A. Al-Wakeel, J. Wu, K-means based cluster analysis of residential smart meter measurements, Energy Procedia 88 (2016) 754–760.

[7] Quan-Sheng Dou, Zhong-Zhi Shi, Ping Jiang, et al., Application of associated clustering and classification method in electric power load forecasting, Chin. J. Comput. 35 (12) (2012) 2645–2651.

[8] Y. Wang, Q. Chen, C. Kang, et al., Clustering of electricity consumption behavior dynamics toward big data applications, IEEE Trans. Smart Grid 7 (5) (2016) 2437–2447.

[9] J. Sun, J. Liu, L. Zhao, Clustering algorithms research, J. Softw. 19 (19) (2008) 48–61.

[10] B.U. Yuan-Yuan, Z.R. Guan, Research of clustering algorithm based on K-means, J. Southwest Univ. Nationalities (2009).

[11] Jun Wang, Shi-Tong Wang, Zhao-Hong Deng, A novel clustering algorithm based on feature weighting distance and soft subspace learning, Chin. J. Comput. 35 (8) (2012) 1655–1665.

[12] J. Xie, S. Jiang, W. Xie, et al., An efficient global K-means clustering algorithm, J. Comput. 6 (2) (2011) 271–279.

[13] Jian-Pei Zhang, Yu Yang, Jing Yang, et al., Algorithm for initialization of K-means clustering center based on optimized-division, J. Syst. Simul. 21 (9) (2009) 2586–2590.

[14] Xiao-Feng Lei, Kun-Qing Xie, Fang Lin, et al., An efficient clustering algorithm based on local optimality of K-means, J. Softw. 19 (7) (2008) 1683–1692.

[15] B.J. Frey, D. Dueck, Clustering by passing messages between data points, Science 315 (5814) (2007) 972–976.

[16] Shi-Bing Zhou, Zhen-Yuan Xu, Xu-Qing Tang, New method for determining optimal number of clusters in K-means clustering algorithm, Comput. Eng. Appl. 46 (16) (2010) 27–31.

[17] D.H. Mao, Improved Canopy-Kmeans algorithm based on MapReduce, Comput. Eng. Appl. (2012).

[18] G. Tzortzis, A. Likas, G. Tzortzis, The MinMax k-Means clustering algorithm, Pattern Recognit. 47 (7) (2014) 2505–2516.

[19] D. Wu, Y. Zhang, F. Yang, et al., Improved k-means algorithm based on optimizing initial cluster centers, Icic Express Lett. 7 (3) (2013) 991–996.

[20] J. Yoder, C.E. Priebe, Semi-supervised k-means++, J. Stat. Comput. Simul. (3) (2016).

[21] Fritzke B. The k-means-u* algorithm: non-local jumps and greedy retries improve k-means++ clustering[*J*]. 2017.

[22] G.L. Fan, Y.W. Liu, J.Q. Tong, et al., Application of K-means algorithm to web text mining based on average density optimization, J. Digital Inf. Manage. (2016).

[23] S.Z. Ali, N. Tiwari, S. Sen, A novel method for clustering using k-means and Apriori algorithm, in: Proceeding of International Conference on Advances in Electrical, Electronics, Information, Communication and Bio-Informatics, Chennai, India, 2016, pp. 59–62.

[24] Frank L., Asuncion A. UCI machine learning repository[EB/OL].[2011-11-02]. http://archive.ics.uci.edu/ml/datasets.html.

[25] A. Mccallum, K. Nigam, L.H. Ungar, Efficient clustering of high-dimensional data sets with application to reference matching, in: International Conference on Knowledge Discovery and Data Mining, DBLP, 2000, pp. 169–178.