A Novel Improved $K$-Means Algorithm Based on Parameter Adaptive Selection

To cite this article: Xiaodi Huang et al 2020 J. Phys.: Conf. Ser. 1549 042005

View the article online for updates and enhancements.
A Novel Improved \textit{K-Means} Algorithm Based on Parameter Adaptive Selection

Xiaodi Huang\textsuperscript{a}, Minglun Ren\textsuperscript{*} and Xiaoxi Zhu\textsuperscript{b}

School of Management, Hefei University of Technology, Hefei, 230009, China

\textsuperscript{*}Corresponding author e-mail: renml@hfut.edu.cn, \textsuperscript{a}hxd@mail.hfut.edu.cn, \textsuperscript{b}zhuxiaoxi@hfut.edu.cn

Abstract. As a classical clustering algorithm, \textit{K-means} has been widely applied due to its features of simple mathematical thinking, fast convergence rate, less complexity, and easy to implementation. However, \textit{K-means} algorithm always requires users to set the desired number of clusters in advance, and the initial cluster centers are usually generated in a random way. When dealing with unknown datasets that users do not have enough domain-assisted knowledge, such parameters setting strategies not only increases the burden on users, but also makes clustering quality difficult to guarantee. Therefore, in view of the high sensitivity of \textit{K-means} clustering process to initial parameters, this paper propose an improved DDWK-\textit{means} (Distance-Density-Weight \textit{K-means}) algorithm. Based on the distance-density feature and the method of inertia weight of particle swarm optimization algorithm, the optimal initial cluster centers not only can be determined adaptively according to the structural characteristics of the dataset itself without introducing artificial parameters, but also can be adjusted dynamically due to the threshold change of clustering quality metric. We make an experimental study with five standard datasets from UCI (University of California Irvine), and the results indicate that the DDWK-\textit{means} algorithm exhibits a significantly improvement in clustering efficiency and stability.

1. Introduction

As an important research method in data mining, clustering approach is centered on unsupervised learning using appropriate metric based on data characteristics [1]. Currently, many clustering algorithms have been proposed, which are inter-connected and overlapped [2]. According to the form of the clustering results, these algorithms can be roughly classified as hierarchical approach and partitioning approach. For hierarchical approach, Data objects are gathered into clusters at different hierarchies, where the same object can belong to different clusters [3, 4]. Partitioning clustering algorithms divide data objects into multiple mutually exclusive clusters, where the same object can only belongs to a single cluster [5, 6].

\textit{K-means} algorithm, being a well-known partitioning clustering algorithm, is popular and widely used due to its simplicity [7]. Based on the randomly selected initial cluster centers, it uses the distance as the metric to divide data objects into multiple mutually exclusive clusters, where the same object can only belongs to a single cluster [8]. Then, based on the relocation iteration of cluster centers, achiving the goals that the objects within the same cluster are highly similar and the objects among clusters are
maximum different [9]. The main drawback of the algorithm is that it is difficult to select the appropriate desired number of cluster and initial cluster centers for unknown datasets [10]. Currently, various researches show that the robustness and applicability of K-means algorithm can be obviously improved by optimizing the initial parameter based on the data characteristics of the dataset itself [11, 12]. The existing mainstream improvement research can be roughly divided into the following categories.

Parameter selection strategy based on the distance characteristic. Chen et al. [13] selected two data with the greatest distance from each other as initial clustering centers to form initial clusters. Then each cluster splits continuously within the preset threshold, and the desired number of cluster and the initial cluster centers will be determined adaptively. Zhu et al. [14], according to the fact that the intra-class distance between data objects in the same cluster is less than the inter-class distance among different clusters [15, 16], objects whose similarity are higher than a fixed threshold will be regarded as core objects. Then, the most dissimilar objects within the core objects will be selected as initial cluster centers. Based on the prior knowledge of specific application background [17], Yang et al. [18] modified the distance measure of conventional k-means algorithm to make the algorithm more pertinent when selecting initial parameters. Such kind of research only improves the algorithm based on distance metric, and does not control the neighborhood density of data objects. The clustering process is easily affected by noise and anomaly, and clustering results are lack of stability.

Parameter selection strategy based on the density characteristic. With the neighborhood density of data objects as the metric, data objects whose density is higher than the average level will be selected as initial cluster centres. When dealing with non-uniformly distributed datasets, the clustering results based on density metric will be significantly improved [19, 20]. Redmond et al. [21] estimated the neighborhood density of data object of different regions based on the kd-tree (k-dimensional tree), the initial clustering center is selected due to the difference of relative density among data objects. This kind of improved research based on density metric can effectively avoid the interference of noise and anomaly, and objects with tighter data relations will be selected as initial cluster centers. However, for relocation iteration, due to the lack of distance constraints, clustering results may fall into local optimum because the initial cluster center is too close.

In order to overcome the problems of the above two improvement strategies and further improve the effectiveness of the clustering process and result, current optimization researches focus on the construction of improvement strategy based on the double metric of distance and neighborhood density. Zhang et al. [22] proposed an improved algorithm based on the product of sample density, the reciprocal of the average distance between the samples in the cluster, and the distance between the clusters, and all the objects meeting the threshold requirements will be selected as the initial cluster centers. Nguyen et al. [23] used the similarity distance of scalable spatial density to measure the similarity between data objects. Experiment results show that more reasonable initial clustering centers can be obtained and the distribution of complex datasets can be better reflected. In addition, some other initial parameters improvement strategies based on fuzzy theory [24-26], feature weight [27, 28], and swarm intelligence algorithm [29-31].

This paper is organized as follows. In section 2, we briefly described the process of conventional K-means algorithm. In Section 3, we give a detailed elaboration on each part of DDWK-means algorithm. In Section 4, the experiment and the results are presented and discussed. Finally, we give conclusions in Section 5.

2. The conventional K-means algorithm

For a dataset D containing n objects, the main objective of K-means algorithm is to partition the dataset into a predefined k number of clusters, and minimizing the sum of squared errors within each cluster by relocated iteration. The different between two objects is calculated by Euclidean distance as shown in Eq.1. The sum of squared errors within each cluster is calculated by Eq.2, ci represents each cluster center, p represents other objects in the cluster.

\[ d(a_i, a_{ij}) = \sqrt{(\beta_{i1} - \beta_{j1})^2 + (\beta_{i2} - \beta_{j2})^2 + \cdots + (\beta_{im} - \beta_{jm})^2} \]  

(1)
Based on elbow method, all integers in $[2: \sqrt{n}]$ will be selected successively as the desired number of cluster $k$. The process of conventional K-means algorithm can be described as follows:

Step 1. Due to each desired number of cluster, $k$ data objects will be randomly selected from the dataset to serve as initial centers.

Step 2. According to Eq.1, remaining objects will be distributed to their nearest centers to form initial clusters, and recording the sum of squared errors $E$ due to Eq.2.

Step 3. Mean values of each cluster are calculated to serve as new centers.

Step 4. Relocated iteration: New clusters will be formed based on new centers, and recording the sum of squared errors of new clusters.

Step 5. Iteration will go forward until the maximum preset number of times is reached or the sum of squared error of all clusters remain stable.

Step 6. Clustering results with different desired number of clusters will be compared to select the best one.

The improved DDWK-means algorithm proposed in this paper aims to optimize initial parameters of the conventional algorithm to select more reasonable desired number of clusters and initial cluster centers. In order to facilitate experimental comparison, we also use Eq.1 and Eq.2 to calculate the distance between data objects and the sum of squared errors within each cluster.

### 3. The process of DDWK-means algorithm

The DDWK-means algorithm proposed in this paper consists of five parts: threshold parameters self-adaptive generation, candidate cluster centers selection, initial cluster centers screening, conditional relocation iteration, and clustering result evaluation. The detailed process of DDWK-means algorithm is shown in Fig.1 as below.

![Figure 1. The detailed process of DDWK-means algorithm](image-url)

#### 3.1. The Self-adaptive Generation of Threshold Parameters

The conventional k-means algorithm determines the optimal parameters based on all possible clustering results. When dealing with large-scale complex datasets, the algorithm will be insufficient in scalability due to the surge of operation redundancy. In this section of DDWK-means algorithm, firstly, the range
of desired number of cluster is divided into equal intervals, then the upper boundary value of each interval is taken as the representative of the whole interval according to the fuzzy theory [32, 33], and threshold parameters of the algorithm will be set accord to these representative values. In addition, circular judgment criterion is introduced to judge the rationality of equal intervals based on clustering accuracy. Detailed steps are given in following Rules 1 and 2.

Rule 1: For a dataset containing \( n \) objects, all possible values of \( k \) which represent the desired number of cluster form a set, which is calculated by Eq.3.

\[
k = \lfloor k \rfloor, k' = \begin{cases} \sqrt{n}/p, & \text{if } \sqrt{n}/p > 2 \\ 2, & \text{if } \sqrt{n}/p \leq 2 \end{cases}, p = N^+, p \in [1:10]
\]  

Based on the value of \( k \), there are about \( \lfloor n/k \rfloor \) data objects in each cluster on average. The Euclidean distance as shown in Eq.1 is chose as the measurement to calculate the centroid of the dataset. The radius of the centroid which contains at least \( \lfloor n/k \rfloor \) data objects will be set as the measurement radius \( r \) and \( \lfloor n/k \rfloor \) data objects will be taken as the density threshold \( p \) of each cluster. If the optimal cluster number exists in an interval, the corresponding neighborhood radius and neighborhood density must contain the parameters determined by the upper boundary value of the interval. Therefore, the real cluster will not be omitted with the density threshold of \( p = \lfloor n/k \rfloor \). When dealing with datasets which have background knowledge or special clustering accuracy requirements, these situations can be solved by adjusting the number of equant intervals. The clustering efficiency can be improved by reducing the range of parameters on a large scale.

In the last part of the algorithm: clustering result evaluation, if all clustering results of the algorithm parameters determined according to Rule 1 do not meet the threshold requirement of the metric, the parameters will be re-selected according to Rule 2.

Rule 2: The upper boundary values of the two intervals with the best clustering quality in the previous round will be chose to form a new parameter selection range: \( \left( \sqrt{n}/p_1, \sqrt{n}/p_j \right) \), and the new set which consist all possible new values of \( k \) will be calculated by Eq.4 during the new rang.

\[
k = \lfloor k \rfloor, k' = \sqrt{n}/p, p = N^+, p \in [1:10]
\]  

Based on the new set of desired number of cluster, the new neighborhood radius and density threshold are determined for clustering calculation again. In the part of clustering result evaluation, if there exists clustering result satisfying the threshold requirement, the algorithm will go to the next part, otherwise it will continue to iterate according to Rule 2.

3.2. Candidate Cluster Centers Selection

The best clustering results should meet the criteria that objects within cluster are highly similar and objects among clusters differ to the greatest extent, that is, the distance between cluster centers should be as far as possible to avoid concentrating in a small range and falling into local optimal [34]. Meanwhile, in order to avoid the influence of noise and anomaly data, cluster centers should have relatively high neighborhood density [35, 36]. From this, we can deduce Inference 1 as follows:

Inference 1: The optimal clustering results of conventional datasets should satisfy that the neighborhood density of each cluster centers are relatively high and the distance between different cluster centers are relatively far.

According to Inference 1, candidate cluster centers will be selected based on the metric of “neighborhood density-relative distance”, and the initial clusters will be formed based on the selected candidate cluster centers. By deleting the sequential formed clusters, the repeated computation will be reduced obviously. The detailed steps are as follows:
Step 1. For a dataset $D$, the data object $c_1$ which has the highest neighborhood within the measurement radius $\sigma$ is selected as the first candidate cluster center.

Step 2. The first cluster $C_1$ is formed by aggregating $S_1$ data within the measurement radius $\sigma$.

Step 3. The new dataset $D_1$ is formed by removing the first cluster, in which remains $|D| - |S_1|$ data objects.

Step 4. In dataset $D_1$, Euclidean distance is taken as a measure to search for the object with the farthest distance to $c_1$. If the neighborhood density of this object meets the density threshold requirement $\rho=[n/k]$, it will be taken as the second candidate cluster center. Otherwise, the second distant object from $c_1$ will be selected for neighborhood density measurement. At last, the object that meets the density threshold requirement and is far enough away from $c_1$ is selected as the second candidate cluster center $c_2$.

Step 5. The second cluster $C_2$ is formed by aggregating $S_2$ data objects within the measurement radius $\sigma$.

Step 6. The new dataset $D_2$ is formed by removing the second cluster, in which remains $|D_1| - |S_2|$ data objects.

Step 7. The relative distance ($Rd$) to existing centers of the remaining data objects are calculated by Eq.5. In dataset $D_2$, Euclidean distance is taken as the measure to search for data with the largest $Rd$. If the neighborhood density of the data meets the density threshold requirement $\rho=[n/k]$, it will be taken as the third candidate cluster center. Otherwise, the density measurement will be carried out on the object with second largest $Rd$. The object which meets the density threshold requirement and the $Rd$ is large enough will be selected as the third candidate cluster center.

$$Rd_i = \min(dist(a, x_i), dist(a, x_2)) \quad (5)$$

Step 8. The third cluster $C_3$ is formed by aggregating $S_3$ data objects within the measurement radius $\sigma$.

Step 9. The new dataset $D_3$ will be formed by removing the second cluster, in which remains $|D_2| - |S_3|$ data.

Based on above steps, the selection go farward until any object in the dataset $D$ does not meet the density threshold requirements, terminate the algorithm and sequential output all candidate cluster centers $c_1, c_2, ..., c_k$.

3.3. Initial Cluster Centers Screening

The process of selecting candidate cluster centers based on Inference 1 is essentially a cluster sampling method, which divides the whole data set into several non-overlapping clusters, and then takes cluster as the sampling unit to extract samples. Hence, there may be a problem of relative quality degradation of samples in the process of selecting candidate cluster centers [37]. That is, with the sample extraction, the size of the sample pool is continuously shrinking. Although the subsequent samples still meet the threshold requirements, the relative quality of the samples decreases in the order of their generation due to the reduction of the contrast space [38, 39]. In existing researches of such improvement, the quality differences between candidate cluster centers are hardly taken into account, they usually participate in subsequent iterations with the same weight. The phenomenon of inferior cluster centers expelling high-quality cluster centers may occur, which affects the robustness of clustering process and result.

As shown in Figure 2, we also find the similar situation in the experiment, the dataset is expected to be divided into four clusters, and the real cluster centers should be $c_1$, $c_2$, $c_3$, $c_4$. Data objects $c_3$ and $c_4$ are "pseudo cluster centers" that selected after the size of the dataset is significantly reduced. From this, we can deduce Inference 2 as follows.

**Inference 2:** In the processing of candidate cluster centers selection, the quality of the first center is the best, and as the size of the dataset shrinks gradually, the quality of the subsequent centers may show a decreasing trend.
The quality of the candidate cluster centers show a decreasing trend

In order to solve this problem, in the part of initial cluster center screening, we introduce the inertia weight of particle swarm optimization algorithm [40, 41]. The detailed steps of initial cluster centers screening are as follows:

Step 1. The linear decreasing weight will be given according to the order of candidate cluster centers generation: The weights of each cluster center are calculated according to Eq.6, and the weight distances of other data objects to each cluster center are calculated according to Eq.7.

\[ w = w_{\text{max}} - \frac{n \cdot (w_{\text{max}} - w_{\text{min}})}{n_{\text{max}}} \]  

(6)

The \( n_{\text{max}} \) represents the total number of candidate cluster centers, \( n \) is the position of generation order of current candidate cluster center, \( w_{\text{max}}=0.9, w_{\text{min}}=0.4 \).

\[ \text{dist}(o_1, o_2) = \rho \sqrt{(x_1 - y_1)^2 + (x_2 - y_2)^2 + \cdots + (x_n - y_n)^2}, \rho = \frac{1}{w} \]  

(7)

Step 2. A once weight clustering is performed based on all the candidate cluster centers.

Step 3. Density evaluation is performed on all the formed clusters, if there are inferior clusters, that is, the density within the cluster does not meet the threshold requirement \( \rho = \lceil n/k \rceil \), the algorithm go to step 3, if there is no inferior cluster, go to step 5.

Step 4. The original candidate cluster center forming the inferior cluster will be deleted, and a new weight clustering will be imposed on the remaining original candidate cluster centers that satisfy the density threshold requirement.

Step 5. Density evaluation is performed on all the new formed clusters, if there still exist inferior clusters, the algorithm continues to process according to step 3 until all the inferior cluster centers are screened out. If there is no inferior cluster, go to step 5.

Step 6. The centers of the clusters formed by the weight clustering will be calculated to serve as initial cluster centers of the next part of DDWK-means algorithm.

According to the order of generation, the earlier a candidate cluster center is produced, the greater the weight will assigned to it, the accuracy and iteration times will be optimized by accelerating their local convergence speed. The candidate cluster centers with lower ranking will be given smaller weight, the convergence to local optimal results will be avoided by enhancing their global searching ability. The set of candidate cluster centers will be checked by weight iteration to further screen out the inferior cluster centers and improve the stability of the algorithm.

3.4. Conditional Relocation Iteration

In this part, in order to avoid the existence of inferior cluster centers in initial cluster centers, we set the clustering quality control conditions in the relocation iteration of the conventional algorithm. That is, after each cluster center relocation iteration, a density metric is added to mark the inferior cluster that
does not meet the threshold requirement. If the labeled inferior cluster show a downward trend in clustering quality (cluster density continues to decline) during the subsequent iterations, the inferior cluster will be deleted, and relocation iteration will be only carried on the remaining clusters that satisfying the density threshold. The detailed steps are as follows:

Step 1. Relocation iteration is carried out on initial cluster centers which were screened out in the previous part.

Step 2. Density measurement is performed on the formed clusters. If the density of each cluster meet the density threshold requirement, their inferior cluster markers will be eliminated, and the processing is continued according to step 1 until the clustering result is stable or reaches the maximum number of iterations. If there are clusters do not meet the density threshold requirement, they will be marked as inferior clusters and go to step 3.

Step 3. After recording the density in the current inferior cluster, if the cluster is marked for the first time, it go to step 1. If the cluster already has marks, it transfer to step 4.

Step 4. According to the quality control threshold $\varsigma$, if the number of times the inferior cluster is labeled does not exceed the threshold, it go to step 1, otherwise, it go to step 5. (Parameter $\varsigma$ represents the maximum times for a cluster allowed to be labeled as inferior cluster, The smaller the value of parameter $\varsigma$ is, the higher the control precision is, and the faster the converge rate is, but it is also easier to delete the real cluster by mistake.)

Step 5. If a cluster is continuously below the density threshold and the records of cluster density is decreasing, it will be deleted, otherwise it transfer to step 1.

By conditional relocation iteration, the possible inferior cluster centers can be eliminated as early as possible to avoid redundant iterations, and the convergence rate and stability of the algorithm will be optimized obviously.

3.5. Clustering Result Evaluation

In this part, we choose normalized mutual information (NMI) as the metric to evaluate the clustering results obtained by different parameters. The quality control threshold $NMI_{\text{accept}}$ of clustering result should be set according to the requirement of practical application. If all the clustering results do not meet the threshold, re-clustering will be carried out according to Rule 2. If there are clustering results that meet the requirements, the optimal clustering result is selected according to the value of NMI for each of them.

4. Experiment and analysis

4.1. Experimental Datasets

In the part of experiment, DDWK-means algorithm and conventional K-means algorithm were respectively repeated run 30 times on five datasets that selected from UCI (University of California Irvine)\(^1\). The maximal, minimal, and average value of each index were recorded. Both algorithms were implemented using Python3.0 on an Intel i7 of 3.60 GHz with 8 GB RAM. All the datasets were preprocessed by normalization. The detailed information of each dataset are shown in Table 1.

| Dataset               | Scale | Attribute | Cluster |
|-----------------------|-------|-----------|---------|
| Iris                  | 150   | 4         | 3       |
| Seeds                 | 210   | 7         | 3       |
| Survival              | 306   | 3         | 2       |
| Knowledge Modeling    | 403   | 5         | 4       |
| Perfume               | 560   | 2         | 20      |

\(^1\) The link of UCI dataset official website: http://archive.ics.uci.edu/ml/datasets.php.
4.2. Measurement Index
In the experiment, the performance of DDWK-means and conventional K-means algorithms are evaluated with internal and external measurements. The internal measurement uses silhouette coefficient, while the external measurement uses normalized mutual information.

Silhouette coefficient (SC). Evaluated the clustering quality by calculating the cohesion and separation of all the data. When the value of SC approaches 1, the clustering result is reasonable. Otherwise, if the value approaches −1, the clustering result is unreasonable.

NMI (Normalized Mutual Information). Evaluated the clustering quality by comparing the distribution of actual label with the data distribution after clustering. The receivable value of NMI should be in the range of [0, 1], a larger value corresponding to a greater agreement.

4.3. Experimental Results
The maximal number of the iteration for each algorithm was preset to 100. For DDWK-means algorithm, the \( NMI_{\text{accept}} \) was set to 0.55 and the \( \varsigma \) was set to 3. The mean values of 30 test results for each index are shown in Table 2. All the values were calculated to three decimal places.

| Dataset   | Conventional K-means | DDWK-means |
|-----------|----------------------|-------------|
| Iris      | SC Max: 0.689, Min: 0.599 | SC Max: 0.770, Min: 0.669 |
| Seeds     | SC Max: 0.698, Min: 0.605 | SC Max: 0.714, Min: 0.615 |
| Survival  | SC Max: 0.679, Min: 0.588 | SC Max: 0.737, Min: 0.647 |
| Knowledge | SC Max: 0.668, Min: 0.579 | SC Max: 0.620, Min: 0.575 |
| Perfume   | SC Max: 0.619, Min: 0.589 | SC Max: 0.648, Min: 0.618 |

4.4. Experiment Analysis
As shown in Table 2, compared with conventional K-means algorithm, our proposed algorithm eliminates the randomness of parameter selection, hence the clustering results are always unique, which makes the clustering process more stable. In addition, besides Perfume dataset, the clustering qualities of other datasets are significantly improved, which means that our algorithm is much more efficient than the conventional algorithm.

As shown in Table 3, for Perfume dataset which expects to divide 560 data objects into 20 clusters, the DDWK-means algorithm proposed in this paper is weaker than the conventional K-means algorithm in both quality evaluation metric of SC and NMI.

| Dataset | K-means | DDWK-means |
|---------|---------|------------|
| SC      | Max: 0.619, Min: 0.589, Mean: 0.601 | Max: 0.556 |
| NMI     | 0.648, 0.618, 0.622 | 0.590 |

After careful analysis of each part of DDWK-means algorithm, we found that the maximum cluster number \( \sqrt{n}=24 \) set for Perfume dataset is very close to the real cluster number \( (k=\sqrt{n}) \). Although the density threshold \( p=\lceil n/k \rceil \) and the measurement radius \( \varsigma \) based on this setting are close to the real value, in the process of candidate cluster centers selection, a large number of data objects that far exceeding the value of density threshold have been delimited by several early generated candidate cluster centers, which makes it impossible to select enough candidate cluster centers, thus leading to a large deviation in the number of initial cluster centers. Moreover, this deviation cannot be improved by the following steps, which greatly reduces the accuracy of the clustering result.
4.5. The Improvement for Multi-cluster Dataset

For multi-cluster dataset, which means the number of real clusters is close to the maximum clusters \( k \approx \sqrt{n} \), the main problem of DDWK-means algorithm is that the second part of the algorithm cannot pick out enough candidate cluster centers. Therefore, when dealing with multi-cluster dataset, we add density limit control to the part of candidate cluster centers selection to ensure that a sufficient number of candidate cluster centers can be found. The algorithm is adjusted as follows: for each selected candidate cluster center, only the nearest \( \sqrt{n} \) data objects are deleted from the current dataset to form the cluster, the algorithm will go forward until there is no object satisfying the density threshold.

On the two multi-cluster datasets of Perfume dataset and Absenteeism dataset, clustering experiments were conducted based on the adjusted DDWK-means algorithm and conventional K-means algorithm. The detailed information of each dataset are shown in Table 4.

| Dataset     | Scale | Attribute | Cluster |
|-------------|-------|-----------|---------|
| Perfume     | 560   | 2         | 20      |
| Absenteeism | 740   | 21        | 22      |

Each algorithm is respectively repeated run 30 times and the maximal, minimal, and average value of each index were recorded in Table 5. All the datasets were preprocessed by normalization and all the values were calculated to three decimal places.

| Dataset     | SC Max | SC Min | SC Mean | NMI Max | NMI Min | NMI Mean | SC | NMI |
|-------------|--------|--------|---------|---------|---------|----------|----|-----|
| Perfume     | 0.619  | 0.589  | 0.601   | 0.648   | 0.618   | 0.622    | 0.688 | 0.734 |
| Absenteeism | 0.604  | 0.536  | 0.579   | 0.617   | 0.564   | 0.592    | 0.653 | 0.717 |

The experimental results show that the adjusted DDWK-means algorithm is superior to the conventional K-means algorithm in SC and NMI metrics, and the clustering results are still stable, which proves the effectiveness of the algorithm in multi-cluster dataset clustering.

5. 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 desired number of clusters and initial cluster centers in conventional algorithm, an improved DDWK-means algorithm based on “distance-density-weight” is proposed in this paper. Firstly, based on the concept of upper boundary of fuzzy theory, the range of desired number of clusters is divided equally, and the control parameters such as neighborhood density and measurement radius are determined by taking the upper boundary of each sub-segment as the representative value. It improves the robustness of the algorithm by avoiding the uncertainty which is caused by random selection and artificial setting. Secondly, data objects that meet the requirements will be sequentially selected as candidate cluster centers and form initial clusters. By continuously deleting the formed clusters, the size of the dataset and the computational complexity can be significantly reduced. Candidate cluster centers selected based on such rules are unique, and the clustering results are relatively stable which has no need for repeated experiments. Thirdly, in order to screen out better cluster centers, based on the inertia weight theory, all the candidate cluster centers will be given linear decreasing weights in order of their generation to speed up the gathering of data objects near the high-quality centers. After once weight iteration, inferior cluster centers that do not meet the density threshold requirement are screened out, and the remaining centers will be taken as initial cluster centers for next step. Based on this, the phenomenon of inferior cluster centers expelling high-quality cluster centers can be avoided by
eliminating the inferior candidate cluster centers that is caused by the change of dataset size. Then, in the part of conditional relocation iteration, both the convergence rate and the iterative redundancy will be optimized by adding the intra-cluster density evaluation to the relocation iteration process to screen out the inferior cluster centers as early as possible. Finally, according to the metric of clustering quality, the optimal algorithm parameters and clustering result can be adaptively determined by measuring each clustering result from different parameters.

The focus of future research should be on datasets that are large in scale but have few real clusters. For such dataset, if the number of its real cluster of is much smaller than the expected maximum number of clusters (k≪√n), our proposed algorithm tends to generate a large number of redundant iterations in the part of threshold parameters self-adaptive generation. In addition, when the difference between data objects of the dataset cannot be measured by Euclidean distance and neighborhood density, although the core idea of the algorithm is unchanged, it is necessary to redesign the selection rules of candidate cluster centers.

Acknowledgments

Fund: National Natural Science Foundation of China, No. 71531008, 71521001, 71490720

References

[1] Rostami, A.S., et al., Survey on clustering in heterogeneous and homogeneous wireless sensor networks. Journal of Supercomputing, 2017 (13): p. 1-47.
[2] Mazzeo, G.M., E. Masciari, and C. Zaniolo, A Fast and Accurate Algorithm for Unsupervised Clustering Around Centroids. Information Sciences, 2017. 400.
[3] Kamis, N.H., F. Chiclana, and J. Levesley, Geo-uninorm Consistency Control Module for Preference Similarity Network Hierarchical Clustering Based Consensus Model. Knowledge-Based Systems, 2018.
[4] Zhang, F., et al., UD-HMM: An unsupervised method for shilling attack detection based on hidden Markov model and hierarchical clustering. Knowledge-Based Systems, 2018.
[5] Berkhin, P., A Survey of Clustering Data Mining Techniques. Grouping Multidimensional Data, 2006. 43 (1): p. 25–71.
[6] Yue, S., et al., A new validity index for evaluating the clustering results by partitional clustering algorithms. Soft Computing, 2016. 20 (3): p. 1127-1138.
[7] Celebi, M.E., Partitional Clustering Algorithms. Applied Statistics, 2008. 23 (6): p. 38-41.
[8] Choi, D.-W. and C.-W. Chung, A K-Partitioning Algorithm for Clustering -Scale Spatio-Textual Data. Information Systems, 2017. 64 (C): p. 1-11.
[9] Capó, M., A. Pérez, and J.A. Lozano, An efficient approximation to the K-means clustering for massive data. Knowledge-Based Systems, 2017. 117: p. 56-69.
[10] Barbakh, W. and C. Fyfe, Local vs global interactions in clustering algorithms: Advances over K-means. International Journal of Knowledge-based and Intelligent Engineering Systems, 2008. 12 (2): p. 83-99.
[11] Fränti, P. and S. Sieranoja, K-means properties on six clustering benchmark datasets. Applied Intelligence, 2018. 48 (12): p. 4743-4759.
[12] Rebollo-Monedero, D., et al., A modification of the k-means method for quasi-unsupervised learning. Knowledge-Based Systems, 2013. 37: p. 176-185.
[13] Chen, G.P., W.P. Wang, and J. Huang, Improved Initial Clustering Center Selection Method for K-means Algorithm. Journal of Chinese Computer Systems, 2012. 33 (6): p. 1320-1323.
[14] Zhu, M., W. Wang, and J. Huang, Improved initial cluster center selection in K-means clustering. Engineering Computations, 2014. 31 (8): p. 1661-1667.
[15] Tang, H.Q. and H.J. Wang, Application of Improved K-Means Algorithm to Analysis of Online Public Opinions. Journal of Personality & Social Psychology, 2010. 98 (4): p. 550-8.
[16] Goyal, M. and S. Kumar, Improving the Initial Centroids of k-means Clustering Algorithm to Generalize its Applicability. Journal of the Institution of Engineers, 2014. 95 (4): p. 345-350.
[17] Yang, Y.M., et al., *Determining the Number of Instars in Simulium quinquecessiatrum (Diptera: Simuliidae) Using k-Means Clustering via the Canberra Distance*. Journal of medical entomology, 2018. 55 (4): p. 808-816.

[18] Yang, J., et al., *An Initialization Method Based on Hybrid Distance for k-Means Algorithm*. Neural Computation, 2017. 29 (11): p. 1-24.

[19] Fan, G.L., et al., *Application of K-means algorithm to web text mining based on average density optimization*. Journal of Digital Information Management, 2016.

[20] Khan, S.S. and A. Ahmad, *Cluster center initialization algorithm for K-means clustering*. 2004: Elsevier Science Inc. 1293-1302.

[21] Redmond, S.J. and C. Heneghan, *A method for initialising the K-means clustering algorithm using kd-trees*. Pattern Recognition Letters, 2007. 28 (8): p. 965-973.

[22] Zhang, G., C. Zhang, and H. Zhang, *Improved K-means algorithm based on density Canopy*. Knowledge-Based Systems, 2018. 145: p. 289-297.

[23] Nguyen, B. and B.D. Baets, *Kernel-Based Distance Metric Learning for Supervised k-Means Clustering*. IEEE Transactions on Neural Networks and Learning Systems, 2019. PP (99): p. 1-12.

[24] Peters, G., *Some refinements of rough k-means clustering*. 2006: Elsevier Science Inc. 1481-1491.

[25] Mitra, S., H. Banka, and W. Pedrycz, *Rough\&\#8211;Fuzzy Collaborative Clustering*. IEEE Transactions on Systems Man & Cybernetics Part B Cybernetics A Publication of the IEEE Systems Man & Cybernetics Society, 2006. 36 (4): p. 795.

[26] Vatanen, T., et al. *Semi-supervised detection of collective anomalies with an application in high energy particle physics*. in International Joint Conference on Neural Networks. 2012.

[27] Huang, X., et al., *A new weighting k-means type clustering framework with an l2-norm regularization*. Knowledge-Based Systems, 2018. 151: p. 165-179.

[28] Zhang, G.-Y., et al., *TW-Co-k-means: Two-level weighted collaborative k-means for multi-view clustering*. Knowledge-Based Systems, 2018. 150: p. 127-138.

[29] Manju, V.N. and A.L. Fred, *AC coefficient and K-means cuckoo optimisation algorithm-based segmentation and compression of compound images*. Iet Image Processing, 2018. 12 (2): p. 218-225.

[30] Reddy, T.N. and K.P. Supreethi, *Optimization of K-means algorithm: Ant colony optimization*. in 2017 International Conference on Computing Methodologies and Communication (ICCMC). 2017.

[31] Rahman, M.A. and M.Z. Islam, *A hybrid clustering technique combining a novel genetic algorithm with K-Means*. Knowledge-Based Systems, 2014. 71: p. 345-365.

[32] Wang, Y., et al., *An Improved Result on Exponential Stabilization of Sampled-Data Fuzzy Systems*. IEEE Transactions on Fuzzy Systems, 2018. PP (99): p. 1-1.

[33] Wu, H.N. and S. Feng, *Mixed Fuzzy/Boundary Control Design for Nonlinear Coupled Systems of ODE and Boundary-Disturbed Uncertain Beam*. IEEE Transactions on Fuzzy Systems, 2018. 26 (6): p. 1-1.

[34] Zhang, T. and F. Ma, *Improved rough k-means clustering algorithm based on weighted distance measure with Gaussian function*. International Journal of Computer Mathematics, 2017. 94 (4): p. 663-675.

[35] Du, M., S. Ding, and Y. Xue, *A robust density peaks clustering algorithm using fuzzy neighborhood*. International Journal of Machine Learning & Cybernetics, 2017 (12): p. 1-10.

[36] Ding, S., et al., *An entropy-based density peaks clustering algorithm for mixed type data employing fuzzy neighborhood*. Knowledge-Based Systems, 2017. 133.

[37] Alemazkoor, N. and H. Meidani, *A near-optimal sampling strategy for sparse recovery of polynomial chaos expansions*. Journal of Computational Physics, 2018. 371: p. 137-151.

[38] Dong, F., et al., *Adapting the Sample Size in Particle Filters Through KLD-Sampling*. International Journal of Robotics Research, 2016. 22 (12): p. 985-1003.

[39] Bubeck, S., R. Eldan, and J. Lehec, *Sampling from a Log-Concave Distribution with Projected
Langevin Monte Carlo. Discrete & Computational Geometry, 2017: p. 1-27.

[40] Sabeti, M., R. Boostani, and B. Davoodi, Improved particle swarm optimisation to estimate bone age. Iet Image Processing, 2018. 12 (2): p. 179-187.

[41] Adeli, A. and A. Broumandnia, Image steganalysis using improved particle swarm optimization based feature selection. Applied Intelligence, 2018. 48 (4): p. 1-14.