Hindawi
Computational Intelligence and Neuroscience
Volume 2022, Article ID 3958423, 13 pages
https://doi.org/10.1155/2022/3958423

Research Article

Improvement of K-Means Algorithm and Its Application in Air Passenger Grouping

Donghua Yu,[1] **Shuhua Dong**,[1] **and Shuang Yao**[2]

[1]*Department of Computer Science and Engineering, Shaoxing University, Shaoxing 312000, China*
[2]*College of Economics and Management, China Jiliang University, Hangzhou, China*

Correspondence should be addressed to Shuang Yao; alloniam@163.com

Received 26 April 2022; Revised 11 June 2022; Accepted 3 August 2022; Published 12 September 2022

Academic Editor: Maciej Lawrynczuk

The k-means is one of the most popular clustering analysis algorithm and widely used in various fields. Nevertheless, it continues to have some shortcomings, for example, extremely sensitive to the initial center points selection and the special points such as noise or outliers. Therefore, this paper proposed initial center points' selection optimization and phased assignment optimization to improve the k-means algorithm. The experimental results on 15 real-world and 10 synthetic datasets show that the improved k-means outperforms its main competitor k-means ++ and under the same setting conditions, namely, using the default parameters,its clustering performance is better than Affinity Propagation, Mean Shift, and DBSCAN. The proposed algorithm was applied to analyze the airline seat selection data to air passengers grouping. The clustering results, as well as absolute deviation rate analysis, realized customer grouping and found out suitable audience group for the recommendation of seat selection services.

1. Introduction

Clustering is to divide the dataset into nonoverlapping subsets, such that the objects in the cluster are as similar as possible, and the objects between the clusters are as dissimilar as possible [1]. There are numerous kinds of clustering algorithms, such as AP [2], DPC [3–6], which show excellent clustering performance. However, as one of the most classic clustering algorithm, the k-means aimed to partition the given dataset into K subsets so as to minimize the within-cluster sum of squared distances continues to be one of the most popular clustering algorithms [7]. Its efficiency and simplicity of implementation make it successfully applied in various fields, such as image [8, 9], education [10], bioinformatics [11], medical [12], partial multiview data [13], agricultural data [14], fuzzy decision-making [15].

Optimizing the initial center points may be one of the most effective methods to improve the performance of k-means algorithm. The study of Fränti and Sieranoja [16] reported that (a) the k-means clustering algorithm can be significantly improved by using a better initialization technique and by repeating (re-starting) the algorithm; (b) when the data have overlapping clusters, k-means can improve the results of the initialization technique; (c) when the data have well separated clusters, the performance of k-means depends completely on the goodness of the initialization; (d) initialization using simple furthest point heuristic (Maxmin) reduces the clustering error of k-means from 15% to 6%, on average. With the popularity of deep learning in various fields, optimizing data representation is also a means to improve clustering performance, especially in the face of high-dimensional data. The robust deep k-means (RDKM) algorithm [17] exploit the hierarchical information of multiple-level attributes with using the deep structure to hierarchically perform k-means.

The k – means + + [18] provided a simple and effective initial center points optimization method called D^2 – sampling. It adds new center point one by one and assigns different selection probabilities to each potential center point. Since then, especially after being embedded in scikit-learn as the default k-means algorithm, it has almost become the first choice based on partitioning clustering algorithms. However, due to k-means ++ randomly selects the first center point uniformly and randomly adds

subsequent center points according to the probability, some special data distribution can also lead to k-means ++ poor results, even unreasonable clustering results. For example, a dataset with five clusters is synthesized and some noise points half-circle surrounding them are added. The clustering result of k-means ++ was shown in Figure 1, where each color represents a cluster. The desired clustering result should be that the points in the upper left corner are divided into five clusters, but the actual result is that the points in the lower (green points) are clustered into a single cluster to be a wrong result. In this paper, some methods were proposed to solve this problem.

Cluster analysis is one of the basic methods of data knowledge discovery. With the development of airline business, ancillary services that satisfy passengers' personal requirement are becoming more and more important for airlines [19, 20]. However, owing to the impact of COVID-19, the airline market faced a dramatic regression (2019–2021), compelling airlines to seek revenue other than from flight tickets [21, 22]. Therefore, establishing ancillary services is significantly important for airlines due to the ability to increase the airline's revenue. In this paper, the improved k-means algorithm is applied into cluster analysis an airline seat selection dataset, which aims to group airline passengers to serve the establishment of auxiliary services.

Based on the above analysis and application requirements, this paper proposed an improved k-means algorithm, called as k-means2o, based on initial center points selection optimization and phased assignment optimization, and realized the clustering analysis on airline seat selection dataset. The main contributions are summarized as follows:

(1) Two optimization methods are proposed for the k-means algorithm: initial center points selection and phased assignment. In the initial center points selection optimization, this method inherits the center point incremental strategy of k-means ++ [18], K-MC ^{2} [23] and AFK-MC ^{2} [24], but redefines the first center point selection strategy and the subsequent center point incremental strategy. In the phased assignment optimization, the Tukey's rule is adopted to divide dataset into core and noncore sets to realize two-stage assignment, then two assignment strategies are proposed corresponding to the core and noncore sets, respectively.

(2) Four popular algorithms, k-means ++ [18], affinity propagation [2], mean shift [25], and DBSCAN [26], are used to verify the effectiveness and the performance improvement of k-means2o based on 15 real-world and 10 synthetic datasets. Further, the impact of core and noncore sets on the clustering result is analyzed.

(3) The improved k-means algorithm is applied to an airline seat selection dataset, and the passenger groups who are more willing to pay for seat selection are found out. The absolute deviation rate adr is defined to analyze the significance of passenger grouping. This provides valuable information for auxiliary services.

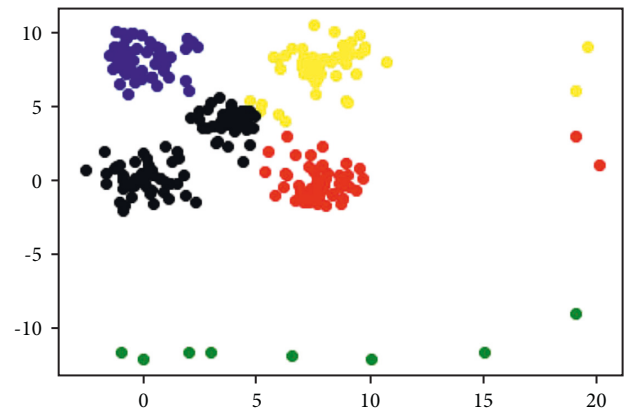

Figure 1: The k − means + + clustering result on synthesized dataset.

2. Related Works

There are many possible ways to optimize the initial center points. The k-means ++ [18] provided D^2 − sampling method which assigns different selection probabilities to each potential center point. Bachem et al. [23] replaced the D^2 − sampling in k-means ++ with MCMC-sampling and obtained a nearly linear improved k-means algorithm K-MC^2. However, this algorithm defines two data-dependent hypothesis $\alpha(X)$, $\beta(X)$, which will have an important impact on the clustering result and the algorithm complexity. Subsequently, Bachem et al. [24] solved the hypothesis defect of the K-MC^2 algorithm. They extended a regular term based on D^2 − sampling of k-means ++. This new algorithm is called AFK-MC^2. Whether it is K-MC^2 or AFK-MC^2, they all follow the first center point selection strategy of the k − means + + algorithm, namely that it first samples an initial center uniformly at random. At the same time, they all have similar center point selection methods, that is, a point farther from the currently selected center points has a greater probability of being chosen as the next center point. For more information on the optimization method of the initial center point, please consult the literature [27].

Phased assignment, generally speaking, is to divide the data into different stages to complete the cluster label assignment, or assign the cluster labels to only part of the data, and the remaining part will be removed as outliers, noise, etc. Zhou et al. [28] proposed a three-stage k-means algorithm to cluster data and detect outliers. In the first stage, the fuzzy c-means algorithm is applied to cluster the data. In the second stage, local outliers are identified, and the cluster centers are recalculated. In the third stage, certain clusters

are merged, and global outliers are identified. Im et al. [29] proposed the NK-means algorithm which emphasizes the removal of noise/outliers and is a two-stage k-means algorithm. In the first stage, a greedy algorithm is utilized to remove abnormal points. In the second stage, the center points are optimized in the constructed core set, and cluster label is assigned to each point. In term of preprocessing techniques, k-means ++ is utilized as an additional filtering step to remove out z of data points as outliers before applying the conventional k-means. The clustering process is only performed on the remaining data which are outlier-free. The outliers data are completely removed and not classified to any known cluster as collected initially. The KMOR algorithm is proposed by Gan and Ng [30] assigns outliers to an additional cluster. This algorithm redefines the clustering objective function and takes into account the SSE between outliers and center points. However, it introduces two new parameters to adjust outlier number. The k-means-sharp is proposed by Olukanmi et al. [31] to eliminate the outliers' influences from the clusters' centroid. The detected outliers are completely excluded from the mean measurement only, but they are involved later in the clustering process. However, the data point with all attributes is eliminated completely from centroid measurement. In this case, the algorithm cannot recognize an outlier's presence in every attribute independently. This is because the single value of the distance metric represents the entire vector instead the single attribute be removed. Therefore, an empty cluster may occur in case of the presence of at least one outlier in each data point [32]. The phased assignment is not only used to optimize the k-means algorithm. For example, Yu et al. [33] also adopted a two-stage assignment strategy based on boundary conditions to optimize the DPC clustering algorithm. For a dataset to be clustered, in many cases, users do not care whether it contains outliers, because the outliers themselves are difficult to define, but they definitely want to assign them cluster labels. Wang et al. [34] proposed an improved integrated clustering learning strategy based on three-stage affinity propagation algorithm with density peak optimization theory (DPKT-AP). In the first stage, the clustering center point was selected by density peak clustering. In the second stage, the k-means algorithm was used to cluster the data samples. In the third stage, DPKT-AP used the AP algorithm to merge and cluster the spherical subgroups.

3. Proposed K-Means Algorithm

Suppose a given dataset $X = \{x_1, x_2, \ldots, x_n\}, x_i \in \mathbb{R}^m$, and divide it into K mutually disjoint sets $C = \{C_1, \ldots, C_K\}$, so that $\cup_i^K C_i = X$ and $C_i \cap C_j = \Phi, \forall i, j, i \neq j$.

3.1. Initial Center Points Optimization. Like the k-means ++ algorithm, the k-means2o adopts a strategy of increasing center points one by one until the desired K points are reached. However, the difference is that the new algorithm redefines the selection of the first center point and subsequent center points. For this purpose, first, define the distance function $\mathrm{d}(x, S)$ between the point x and the set S:

$$\mathrm{d}(x, S) = \min_{x_j \in S} \mathrm{d}\left(x, x_j\right), \tag{1}$$

where $\mathrm{d}(x, x_j)$ represents the distance between two points x, x_j. In this paper, Euclidean distance is selected.

Let $c_i, i = 1, \ldots, K$ represent the center point of cluster $c_i, i = 1, \ldots, K$, then the first center point c_1 is selected as follows:

$$c_1 = \frac{1}{\left|S_{\text{core}}\right|} \sum_{x_i \in S_{\text{core}}} x_i, \tag{2}$$

where $|S_{\text{core}}|$ represents the number of elements in the core set S_{core}. Then, the (2) shows that c_1 is the mean value of the core set S_{core}.

Let $C^k = \{c_1, \ldots, c_k\}$ represents a set containing k center points, then the selection method of $k+1$ th center point c_{k+1} is as follows:

$$c_{k+1} = \mathrm{argmax}_{x_i \in S_{\text{core}}} \mathrm{d}\left(x_i, C^k\right), \tag{3}$$

then $C^{k+1} = C^k \bigcup \{c_{k+1}\}$. Equation (3) shows that c_{k+1} is the point farthest from the selected center points in the core set S_{core}. The whole process above is shown in Figure 2.

3.2. Phased Assignment. The k-means2o is mainly divided into two stages to complete the clustering. The first stage is to assign cluster label to the core set S_{core}, and the second stage is to assign cluster label to the noncore set S_{noncore}. The Tukey's rule is adopted to divide the dataset X into sets $S_{\text{core}}, S_{\text{noncore}}$. Tukey's rule is one of the most robust used techniques for anomaly detection in univariate data [35].

In the first stage, the k-means2o establishes the Tukey's rule for each attribute of the data, and then the judgment results in all dimensions are integrated to determine whether the sample point x belongs to the core set S_{core}.

First, calculate the first quartile Q_1 and third quartile Q_3 on each attribute:

$$\begin{aligned} Q_1^j &= x_i^j \mid i = \mathrm{round}\left((n+1) \times 0.25\right), \\ Q_3^j &= x_i^j \mid i = \mathrm{round}\left((n+1) \times 0.75\right). \end{aligned} \tag{4}$$

Then, calculate the upper and lower bounds $B_{\text{upper}}, B_{\text{lower}}$ as follows:

$$\begin{aligned} B_{\text{lower}}^j &= Q_1^j - r \times \mathrm{IQR}^j, \\ B_{\text{upper}}^j &= Q_3^j + r \times \mathrm{IQR}^j, \end{aligned} \tag{5}$$

where $\mathrm{IQR}^j = Q_3^j - Q_1^j$ and r is a scale factor.

Finally, calculate the core set S_{core} and noncore set S_{noncore} as follows:

$$\begin{aligned} S_{\text{core}} &= \left\{x_i \in X \mid B_{\text{lower}}^j \leq x_{ij} \leq B_{\text{upper}}^j, \forall j\right\}, \\ S_{\text{noncore}} &= X - S_{\text{core}}, \end{aligned} \tag{6}$$

Equation (6) shows that this paper will evaluate each attribute of the data individually, and then integrate all m

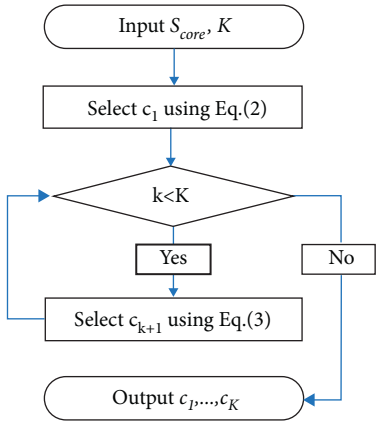

Figure 2: The flow chart of initial center points optimization.

attributes to determine whether it belongs to the core set S_{core}. As long as any attribute does not satisfy the inequality constraints, it will be judged as belonging to S_{noncore}. According to equations (3) and (6), it is obvious that c_2 almost will be the point in the noncore set S_{noncore}, that is, $c_2 \in S_{\text{noncore}}$, and $c_i, i > 2$ will also select the point in the noncore set S_{noncore} with a high probability.

The scale factor r in equation (5) is a predefined adjustable parameter. If you have sufficient prior knowledge of dataset, you can set it depending on experience. If not, it recommends to set $r = 1.5$. Although in the field of anomaly detection research, $r = 1.5$ is often regarded as the boundary value of the outlier. In cluster analysis, points in S_{noncore} cannot be regarded as outliers and discarded, and they still need to be assigned cluster labels. Whether points in S_{core} or in S_{noncore}, in the final clustering result, it is necessary to assign cluster labels which are also one of the goals of cluster analysis. On the 15 real datasets in this paper, each sample has an exact class label, but the S_{noncore} of almost all datasets are not empty. After constructing S_{core}, it is more helpful to obtain a more excellent initial center points. Not only that S_{core} effectively assists the selection of the initial center points but also has a positive effect on the update of center points.

When we obtain S_{core}, use the initial center points selection method described in Section 3.1 to select the initial center points set C^K from S_{core}, and then use the traditional center points update method of k-means to complete clustering in S_{core}. Obtain the optimal clustering center points set $\hat{C}^K$ and clusters $\overline{C}_1, \ldots, \overline{C}_K$. The first stage of clustering ends.

$$x_i \in \overline{C}_k \Leftrightarrow \mathrm{d}\left(x_i, \hat{C}^K\right) = \mathrm{d}\left(x_i, \hat{c}_k\right), x_i \in S_{\text{core}}. \tag{7}$$

In the second stage, points in S_{noncore} will be assigned cluster label. With the help of the optimal clusters $\overline{C}_1, \ldots, \overline{C}_K$ obtained in the first stage, determine the cluster label of $\forall x_i \in S_{\text{noncore}}$:

$$x_i \in C_k \Leftrightarrow \mathrm{d}\left(x_i, S_{\text{core}}\right) = \mathrm{d}\left(x_i, \overline{C}_k\right), x_i \in S_{\text{noncore}}, \tag{8}$$

where $\mathrm{d}(x_i, S_{\text{core}}), \mathrm{d}(x_i, \overline{C}_k)$ are defined in (1), and C_k is the $k-$tk cluster.

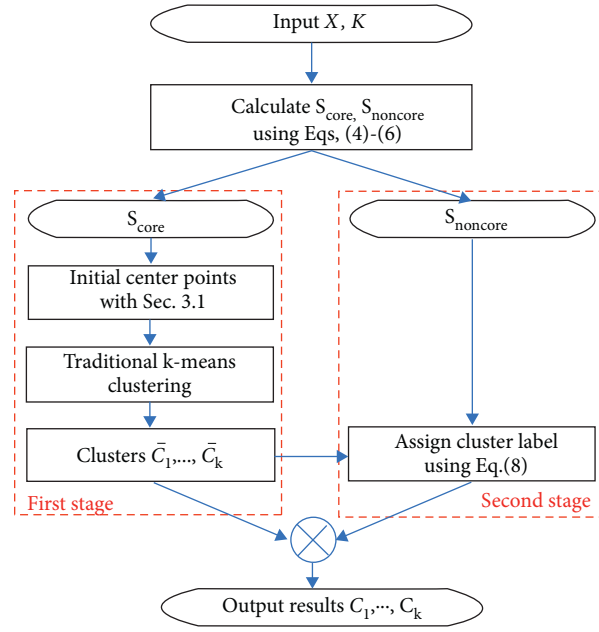

Figure 3: The flow chart of phased assignment.

The whole process above is shown in Figure 3.

3.3. Algorithm Flow and Complexity Analysis. The k-means2o algorithm that optimizes the initial center points selection and phased assignment are performed. The algorithm 1 shows its detail process. The steps 1–15 corresponds to the first stage, including that the Step 1 determines $S_{\text{core}}, S_{\text{noncore}}$, and the Steps 2–4 optimize the initial center points. The Steps 16–19 correspond to the second stage.

According to the detailed steps in algorithm 1, the complexity of k-means2o algorithm is analyzed with data size n, attribute m, and cluster number K. The number of iterations is denoted as t, and its maximum value is max_iter. Step 1 generates $S_{\text{core}}, S_{\text{noncore}}$ with $O(nm)$. Steps 2–5 select initial center points with $O(nK)$. Steps 6–13 are a traditional k-means clustering process; however, Step 8 is a new label assignment strategy, so the complexity of these steps becomes $O(n^2t)$. In summary, the complexity of the k-means2o algorithm is $O(n^2t)$.

4. Performance Analysis of the Proposed Algorithm

In this section, the improved k-means algorithm, k-means2o, testing and verification for clustering performance compared with the well-known k-means ++ [18] which is the most commonly used partition-based algorithm with different initializations of the centroids to reduce the sensitivity. Then, the performance of the k-means2o will be compared with affinity propagation (AP) [2], mean shift (MS) [25], DBSCAN [26]. Although the latter obtain excellent clustering performance on some special datasets, they require to preset one or more important parameter(s), which is a very difficult task. The k-means2o is designed with

Input: Dataset X, cluster number K, scale factor r
Output: Clustering results $C = \{C_1, \ldots, C_K\}$, center points set $\widehat{C}^K$, sum of squared error SSE
(1) Using (6) divide dataset X into $S_{\text{core}}, S_{\text{noncore}}$
(2) Using (2) generate c_1
(3) For $i = 2$ to K do
(4) Using (3) generate c_i
(5) End for
(6) For $j = 1$ to max_iter do
(7) for $\forall x \in S_{\text{core}}$ do
(8) According to the principle of the nearest distance between x and C^K, classify x into the corresponding cluster
(9) end for
(10) if SSE does not change then
(11) break
(12) end if
(13) end for
(14) Update the center points set C^K and compute SSE
(15) Compute the optimal center points $\widehat{C}^K$
(16) for $\forall x \in S_{\text{noncore}}$ do
(17) According to the principle of the nearest distance between x and S_{core}, classify x into the corresponding cluster
(18) end for
(19) Compute SSE
(20) Return clustering results $C = \{C_1, \ldots, C_K\}$, center points set $\widehat{C}^K$, sum of squared error SSE

Algorithm 1: k-means2o.

Python and k-means ++, AP, MS, DBSCAN are called from scikit-learn [36].

4.1. Datasets and Evaluation Metrics. A total of 15 real-world datasets used in the experiments were taken from UCI [37]. The data size n, attribute m, and cluster number K are summarized in Table 1 and Table 2 shows 10 synthetic datasets from references [38, 39], where the K1 dataset is synthesized by this paper, see Figure 1. All datasets are publicly available1.

An appropriate and uniform evaluation index is both required and meaningful to compare the different clustering algorithms. Therefore, the quality was measured via the accuracy (ACC), the Adjusted Rand Index (ARI) [40], the Normalized Mutual Information (NMI) [41] and the Fowlkes–Mallows Index (FMI) [42] between the produced clusters and the truth categories. Larger evaluation index values indicate improved clustering performance, and all index upper bounds = 1, representing perfectly correct clustering:

$$\begin{aligned} \text{ACC} &= \frac{\sum_{i=1}^{n} \delta(\text{label\_true}, \text{map}(\text{label\_pred}))}{n}, \\ ARI &= \frac{RI - E[RI]}{\max(RI) - E[RI]}, \\ \text{NMI} &= \frac{MI(U,V)}{\text{mean}(H(U), H(V))}, \\ \text{FMI} &= \frac{TP}{\sqrt{(TP+FP)(TP+FN)}}. \end{aligned} \tag{9}$$

where U, V are predicted label and true label.

Table 1: Real-world datasets.

Dataset	n	m	K	Dataset	n	m	K
Breast-cancer	569	30	2	Banknote	1372	4	2
Bupa	345	6	2	Compound	399	2	6
Ct	221	36	2	Hayes-roth	132	5	3
Iris	150	4	3	Libras	360	90	15
Parkinsons	195	22	2	Penbased	10992	16	10
Vowel	990	10	11	Waveform21	5000	21	3
Waveform40	5000	40	3	Wdbc	569	30	2
Wine	178	13	3				

Table 2: Synthetic datasets.

Dataset	n	m	K	Dataset	n	m	K
Aggregation	788	2	7	circlesA3	300	2	3
D31	3100	2	31	Flame	240	2	2
Jain	373	2	2	K1	262	2	5
R15	600	2	15	S1	5000	2	15
S3	5000	2	15	Spiral	312	2	3

4.2. Experimental Results and Discussion. The experimental datasets were clustered using k-means ++ and k-means2o. The ACC, ARI, NMI, and FMI of them are listed in Tables 3 and 4, where k-++ represents k-means ++ and k-2o represents k-means2o. The best clustering performance evaluation values are shown in bold, and 1 means that the clustering result is completely correct. The value 0.0000 in the table represents its real metric value < 0.0001.

From Table 3, the k-means ++ and k-means2o simultaneously obtained the maximum FMI value for 8 of the 15 datasets. This shows that the two algorithms have the same performance, and further performance comparison and

Table 3: Clustering results of k-means ++ and k-means2o on real-world datasets.

Dataset	ACC		ARI		NMI		FMI	
	k-++	k-2o	k-++	k-2o	k-++	k-2o	k-++	k-2o
Breast-cancer	0.8541	**0.8910**	0.4914	**0.6062**	0.4647	**0.5276**	0.7915	**0.8286**
Banknote	**0.6122**	0.5954	**0.0485**	0.0356	**0.0303**	0.0239	**0.5517**	0.5231
Bupa	**0.8550**	0.5768	0.0000	**0.0058**	0.0000	**0.0112**	**0.6192**	0.5136
Compound	**0.6566**	0.6365	**0.5378**	0.5043	**0.7191**	0.6557	**0.6422**	0.6181
Ct	0.8235	**0.8325**	0.4160	**0.4399**	0.3296	**0.3485**	0.7078	**0.7199**
Hayes–Roth	0.4393	**0.4469**	0.0202	**0.0226**	0.0287	**0.0317**	0.3501	**0.3519**
Iris	**0.8933**	**0.8933**	**0.7302**	**0.7302**	**0.7581**	**0.7581**	**0.8208**	**0.8208**
Libras	0.4277	**0.4416**	**0.3199**	0.2760	**0.6066**	0.5716	**0.3734**	0.3389
Parkinsons	**0.7230**	0.6307	0.0000	**0.0625**	0.0000	**0.0493**	**0.7444**	0.5889
Penbased	**0.7674**	0.6035	**0.5992**	0.4907	**0.6927**	0.6723	**0.6412**	0.5582
Vowel	0.3636	**0.3645**	0.2028	**0.2204**	0.4141	**0.4337**	0.2789	**0.2868**
Waveform21	0.5016	**0.5018**	0.2536	**0.2547**	0.3622	**0.3654**	0.5039	**0.5047**
Waveform40	0.5146	**0.5160**	0.2516	**0.2530**	0.5023	**0.5035**	0.3605	**0.3632**
Wdbc	0.8541	**0.8910**	0.4914	**0.6062**	0.4647	**0.5276**	0.7915	**0.8286**
Wine	**0.7022**	**0.7022**	**0.3711**	0.3675	**0.4287**	0.4164	**0.5835**	0.5809
Maximum	7	10	6	10	6	10	8	8

The best clustering performance evaluation values are shown in bold.

analysis of other evaluation indicators are required. From the view of ARI in Table 3, the most significant and direct conclusion is that the k-means2o outperforms the k-means ++ on most datasets, and the performance of the two algorithms is also very close on a few datasets that are inferior to k-means ++. Specifically, the k-means2o achieved the maximum ARI value for 10 of the 15 datasets, as well as the NMI and it obtained the same result, and the k-means ++ achieved the best clustering performance only on 6 datasets in ARI, as well as in NMI. For banknote, iris, wine datasets, the k-means2o is only inferior to k-means ++ with a small gap. For ACC evaluation, it comes to the exact same conclusion as NMI and ARI, that is, the k-means2o clustering performance is better than the k-means ++.

For the synthetic datasets in Table 2, the four evaluation metrics in Table 4 show that k-means ++ and k-means2o have similar clustering performance. For datasets with spherical cluster distribution, such as D31, R15, S1, and S3, the clustering results of the two algorithms are close to the real cluster partition, while for datasets with nonspherical distribution such as spiral, flame, circlesA3, the clustering performance of them drops sharply. When the size of the distribution area of spherical clusters is significantly different, the performance difference between k-means ++ and k-means2o can be revealed. For example, in the aggregation dataset, the two algorithms' clustering results are shown in Figure 4. The evaluation values of ARI, NMI, and FMI all show that k-means ++ is better than k-means2o, but ACC gives the opposite conclusion. Figure 4(a) shows that k-means ++ selects seven center points in six real clusters, and two different clusters (green points in the figure) are wrongly classified into the same cluster. Figure 4(b) shows that k-means2o can select center points in seven real clusters, respectively.

Further, the performance of the k-means2o will be compared with AP, MS, and DBSCAN. The ARI and NMI of these algorithms are listed in Table 5, and the ACC and FMI are listed in Table 6. The values larger than the one of the k-means2o are marked in bold. The three comparison algorithms all use default parameters. Considering better performance, the data are normalized here. From the perspective of ARI values, compared with AP, MS, and DBSACN, the k-means2o obtained better clustering performance on 12,14,13 datasets, respectively. The evaluation results of NMI are similar to ARI, except for the AP algorithm. The AP's measurement results of NMI and ARI are very different, which may be tied to the number of error clusters given by the AP algorithm. The ACC evaluation conclusion is consistent with ARI, but FMI and NMI reach opposite conclusions. For the MS algorithm, its FMI value is better than k-means2o algorithm in 9 out of 15 datasets, while for the AP algorithm, its FMI value on all datasets is smaller than k-means2o algorithm. Based on the four evaluation metrics, the k-means2o algorithm is superior to the comparison methods in at least three of these metrics on most datasets. Therefore, k-means2o has better clustering performance.

As for the abnormal conclusion given by a certain evaluation metric for a specific algorithm, for example, the NMI evaluation metric for the AP algorithm, the FMI evaluation metric for the MS algorithm, it may be caused by too many or too few clusters. Table 7 shows that the AP and MS algorithms give the wrong number of clusters on any datasets, and the former far exceeds the true number of clusters, while the latter divides more than half of the datasets into one cluster. Undeniably, the AP, MS, and DBSCAN algorithms provide a method to identify the number of clusters. If the parameters for the AP algorithm, damping factor, and preference value are carefully adjusted, it maybe achieves better clustering performance in these real-world datasets. In those clustering algorithms that contain parameters, careful selection of parameters is often time-consuming and requires prior knowledge. Therefore, these algorithms have poor universality.

The performance of all five algorithms can be directly compared in Figure 5. In this radar chart, each axis

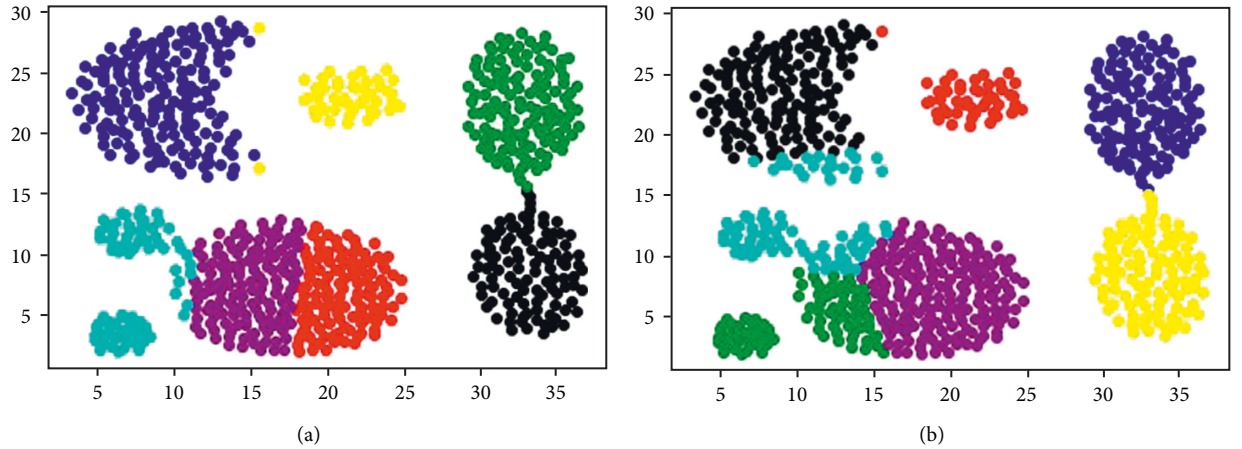

Figure 4: Clustering results on aggregation dataset: (a) k-means ++; (b) k-means2o.

Table 4: Clustering results of k-means ++ and k-means2o on synthetic datasets.

Dataset	ACC		ARI		NMI		FMI	
	k-++	k-2o	k-++	k-2o	k-++	k-2o	k-++	k-2o
Aggregation	0.7855	**0.8680**	**0.7624**	0.7438	**0.8792**	0.8373	**0.8159**	0.7992
CirclesA3	0.5833	**0.5866**	0.1283	**0.1311**	0.3616	**0.3633**	0.4903	**0.4915**
D31	**0.9764**	0.9290	**0.9522**	0.9059	**0.9669**	0.9498	**0.9538**	0.9090
Flame	**0.8375**	**0.8375**	**0.4534**	**0.4534**	**0.3987**	**0.3987**	**0.7363**	**0.7363**
Jain	**0.7855**	**0.7855**	**0.3241**	**0.3241**	**0.3690**	**0.3690**	**0.7005**	**0.7005**
K1	0.7824	**0.9923**	0.7318	**0.9809**	0.8160	**0.9765**	0.7962	**0.9847**
R15	**0.9966**	**0.9966**	**0.9927**	**0.9927**	**0.9942**	**0.9942**	**0.9932**	**0.9932**
S1	**0.9938**	0.9936	**0.9867**	0.9863	**0.9866**	0.9861	**0.9876**	0.9872
S3	**0.8568**	**0.8568**	**0.7270**	**0.7270**	0.7959	**0.7962**	**0.7453**	**0.7453**
Spiral	**0.3461**	**0.3461**	**0.0000**	**0.0000**	**0.0007**	0.0005	0.3276	**0.3277**
Maximum	7	8	8	7	7	6	7	7

The best clustering performance evaluation values are shown in bold.

Table 5: ARI and NMI evaluation results on real-world datasets with AP, MS, DBSCAN.

Dataset	ARI				NMI			
	AP	MS	DBSCAN	k-2o	AP	MS	DBSCAN	k-2o
Breast-cancer	0.0574	0.2275	0.0687	0.6062	0.2692	0.2439	0.0415	0.5276
Banknote	0.0491	0.0000	0.0000	0.0356	0.2973	0.0000	0.0000	0.0239
Bupa	0.0000	0.0000	0.0037	0.0058	0.0287	0.0237	0.0130	0.0112
Compound	0.3023	0.7189	0.0000	0.4133	0.6289	0.7692	0.0000	0.6240
Ct	0.1006	0.0000	0.3271	0.4399	0.2529	0.0601	0.2563	0.3485
Hayes-Roth	0.0350	0.0000	0.0589	0.0226	0.2010	0.0000	0.1168	0.0317
Iris	0.3381	0.5681	0.0000	0.7302	0.5706	0.7336	0.0000	0.7581
Libras	0.2882	0.0000	0.0058	0.2760	0.6375	0.0000	0.1449	0.5716
Parkinsons	0.0305	0.0000	0.0000	0.0625	0.1889	0.0641	0.0227	0.0493
Penbased	0.1037	0.0000	0.0008	0.4907	0.5929	0.0000	0.0380	0.6723
Vowel	0.1315	0.0000	0.0000	0.2204	0.5525	0.0000	0.0000	0.4337
Waveform21	0.0168	0.0000	0.0000	0.2547	0.2345	0.0000	0.0003	0.3654
Waveform40	0.0187	0.0000	0.0000	0.2530	0.2052	0.0000	0.0000	0.3632
Wdbc	0.0574	0.2275	0.0687	0.6062	0.2692	0.2439	0.0415	0.5276
Wine	0.2689	0.0000	0.4228	0.3675	0.5264	0.0000	0.5263	0.4164
Number	3	1	2	—	8	2	1	—

Table 6: ACC and FMI evaluation results on real-world datasets with AP, MS, DBSCAN.

Dataset	ACC				FMI			
	AP	MS	DBSCAN	k-2o	AP	MS	DBSCAN	k-2o
Breast-cancer	0.1353	0.7153	0.6626	0.8910	0.2491	0.7117	0.6775	0.8286
Banknote	0.1013	0.5554	0.5554	0.5954	0.2230	0.7112	0.7112	0.5231
Bupa	0.0841	0.5188	0.9913	0.5768	0.1540	0.6224	0.7073	0.5136
Compound	0.3584	0.7393	0.3960	0.6365	0.4530	0.8159	0.4972	0.6181
Ct	0.2172	0.4977	0.7873	0.8325	0.3267	0.6724	0.6633	0.7199
Hayes-Roth	0.1515	0.3864	0.3561	0.4469	0.1981	0.5876	0.3436	0.3519
Iris	0.4133	0.6667	0.3333	0.8933	0.5145	0.7715	0.5735	0.8208
Libras	0.3889	0.0667	0.1278	0.4416	0.3333	0.2531	0.2429	0.3389
Parkinsons	0.1590	0.6513	0.6154	0.6307	0.2465	0.6857	0.6436	0.5889
Penbased	0.1018	0.1041	0.1195	0.6035	0.2416	0.3164	0.3080	0.5582
Vowel	0.1657	0.0909	0.0909	0.3645	0.2266	0.3000	0.3000	0.2868
Waveform21	0.0368	0.3392	0.3374	0.5018	0.1067	0.5773	0.5613	0.5047
Waveform40	0.0650	0.3384	0.3384	0.5160	0.1124	0.5773	0.5773	0.5035
Wdbc	0.1353	0.7153	0.6626	0.8910	0.2491	0.7117	0.6775	0.8286
Wine	0.3427	0.3989	0.6966	0.7022	0.4604	0.5813	0.6482	0.5809
Number	0	2	1	—	0	9	7	—

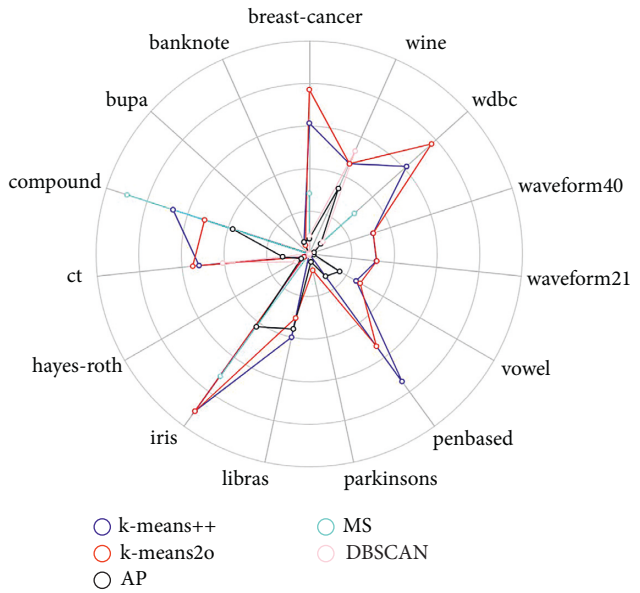

Figure 5: Radar chart of ARI values on the real-world datasets.

represents a dataset, and its value is the cluster evaluation ARI value. According to the previous analysis, the k-means2o has the best performance, and its corresponding red line in the radar chart reaches the maximum value on more polar axes, that is, farther away from the center point.

4.3. Comparative Analysis of Different Initialization Methods. In this subsection, the effects of three different initialization methods on the performance of the k-means clustering algorithm are compared. These three methods are represented by Random, D^2-sampling, New respectively, see the header of Table 8. Random means randomly initializing the center point. D^2-sampling means assigning a selection probability to each noncenter point and randomly selecting the center point. New means the center point initialization optimization method proposed in this paper. In fact, the k-means algorithm based on D^2-sampling is the famous k-means ++ algorithm.

Table 7: The number of real and 3 algorithms predicted clusters in the real-world datasets.

Dataset	AP	MS	DBSCAN	Real
Breast-cancer	43	12	2	2
Bupa	32	14	2	2
Ct	20	7	2	2
Iris	9	2	1	3
Parkinsons	21	5	2	2
Vowel	85	1	1	11
Waveform40	157	1	1	3
Wine	14	1	3	3
Banknote	45	1	1	2
Compound	15	3	1	6
Hayes-roth	16	1	4	3
Libras	30	1	6	15
Penbased	199	1	7	10
Waveform21	148	1	2	3
Wdbc	43	12	2	2

The initial center points optimization plays an important role in the performance improvement of k-means2o. However, Table 8 shows that only using the initialization method proposed in this paper cannot improve the clustering performance. From the evaluation value of ARI, the optimal initialization method is D^2-sampling, followed by Random, and the worst is New which is the initialization method proposed in this paper. Except for tiny numerical differences on individual datasets, the NMI evaluation shows similar conclusions. Combined with the conclusion of k-means2o performance improvement, it is the combination of initial center point optimization and phased assignment that improves the performance of k-means2o, not just the center points optimization.

Table 8: Results of different initialization methods of k-means algorithm.

Dataset	ARI			NMI		
	Random	D^2-Sampling	New	Random	D^2-Sampling	New
Breast-cancer	0.4914	0.4914	0.4914	0.4914	0.4914	0.4914
Banknote	0.0485	0.0485	0.0485	0.0485	0.0485	0.0485
Bupa	0.0000	0.0000	0.0000	0.0000	0.0001	0.0001
Compound	0.5328	0.5378	0.4133	0.7220	0.7191	0.6240
Ct	0.4160	0.4160	0.0000	0.3296	0.3296	0.0219
Hayes-Roth	0.0160	0.0202	0.0202	0.0250	0.0287	0.0287
Iris	0.7302	0.7302	0.7302	0.7581	0.7581	0.7581
Libras	0.3062	0.3199	0.2868	0.5896	0.6066	0.5767
Parkinsons	0.0853	0.0000	0.0001	0.0505	0.0001	0.0001
Penbased	0.5442	0.5992	0.4265	0.6835	0.6927	0.6486
Vowel	0.2180	0.2028	0.2058	0.4332	0.4141	0.4067
Waveform21	0.2536	0.2536	0.2535	0.3622	0.3622	0.3622
Waveform40	0.2516	0.2516	0.2515	0.3605	0.3605	0.3605
Wdbc	0.4914	0.4914	0.4914	0.4647	0.4647	0.4647
Wine	0.3711	0.3711	0.3711	0.4287	0.4287	0.4287
Maximum	11	13	7	11	12	9

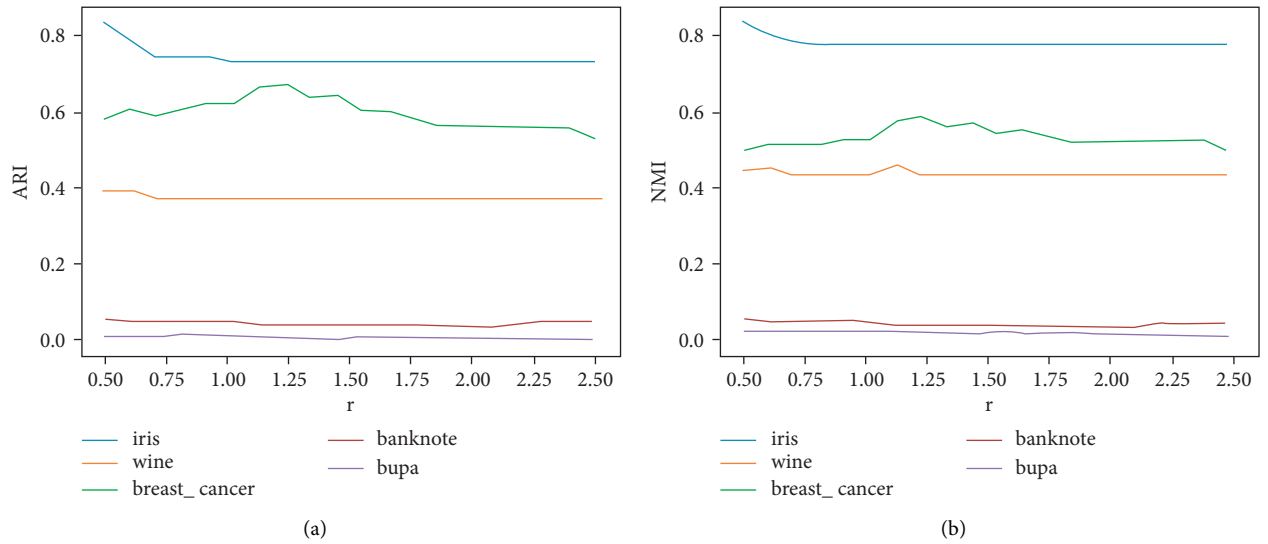

Figure 6: Clustering evaluation for different r on five datasets: (a) ARI curve and (b) NMI curve.

4.4. Impact Analysis of Core and Noncore Sets. This paper uses Tukey's rule to realize the division of S_{core} and $S_{noncore}$. Therefore, a scale factor r needs to be given. Tukey's rule comes from the field of anomaly detection. Generally, the scale factor is set to 1.5. Points that do not meet the conditions of the scale factor are called outliers. In most cases, these points are directly abandoned. This idea is introduced into cluster analysis and used in the data preprocessing stage. As a result, the points detected as abnormal will be discarded and not assign cluster label. There will be great hidden trouble in this way. Table 9 shows the number of elements in S_{core} and $S_{noncore}$ in 15 real-world datasets when $r = 1.5$. Except that the $S_{noncore}$ of compound dataset is empty, the $S_{noncore}$ of the remaining 14 datasets are not empty. However, as well as we known, all points in these datasets are labeled with class labels. Therefore, it is unreasonable to abandon these suspected outliers simply and rudely. For this reason, this paper proposes a two-stage assignment method, whose first stage assigns cluster label to the points in S_{core} and second stage assigns the points in the $S_{noncore}$. For the compound dataset, the empty $S_{noncore}$ indicates that Tukey's rule has no effect on this dataset and will directly lead to the failure of the second stage assignment.

The k-means2o algorithm relies on a predefined scale factor r, so it is necessary to perform a sensitivity test of this parameter. Therefore, we took the iris, wine, breast_cancer, banknote, and bupa datasets as an example to investigate the effects of different r on ARI and NMI, as shown in Figure 6. Its shows that the ARI and NMI curves of the five datasets do not fluctuate drastically, so the clustering performance of the k-means2o algorithm based on the scale factor r is relatively robust. Nevertheless, the scale factor r still has a slight impact on the clustering performance. For example, in the iris dataset, when $r = 0.5$, its ARI and NMI values reach

0.8340 and 0.8191, respectively. This clustering result is better than k-means ++, see Table 3 (the values are ARI = 0.7302 and NMI = 0.7581).

In the above analysis, the k-means2o outperforms k-means ++, AP, MS, and DBSCAN. Combined with the fact, almost all $S_{noncore}$ of these datasets in Table 9 are nonempty. These results show that the combination optimization of the initial center point and the core subset works and improves the k-means clustering performance.

5. The Application of K-Means2o

In this section, the k-means2o is applied into cluster to analyze the airline seat selection dataset provided by Neusoft. According to the meaning of clustering, the samples in the same cluster are as similar as possible, and the samples between different clusters are as dissimilar as possible. If most samples in the same cluster have a certain property, it can be inferred that other samples in the same cluster are also most likely to have the same property. If the most passengers in the cluster are willing to accept some of the personalized recommendation service, such as paying for seat selection, the same service should be recommended to other passengers in the cluster, and a clearer audience group will increase the personalized recommendation service success rate. For the airline seat selection dataset, the appropriate clusters number is required to be determined first.

The silhouette coefficient is a simple and effective method to determine the appropriate clusters number for the k-means algorithm. The silhouette coefficient of the k-means2o algorithm on this dataset is shown in Figure 7. The figure shows that the SSE change tends to be gentle from 16 clusters. Therefore, the optimal number of clusters would be selected as 16. Then, the k-means2o is applied and divides the data into 16 clusters. The number of passengers in each cluster is shown in the column named as size in Table 10. The 3rd, 4th, and 5th columns of Table 10 (payment, no-payment, payment ratio), respectively, show the number of paid passengers, the number of nonpaid passengers and the proportion of paid ones in the airline seat selection. The absolute deviation rate adr in the last column is defined as follows:

$$\mathrm{adr}_c = \frac{|r_c - r|}{r}, \tag{10}$$

where r_c is the payment rate in cluster c and r is the payment rate in the dataset. The larger the adr value, the more significant the difference between the payment behavior of passengers in the cluster and the whole dataset.

The clustering results show that the number of passengers in each cluster is not close. The cluster with the largest number of passengers is $C0$, with 2580, while the smallest one is $C13$, with 379.

Further, the significant differences are explored between clusters. Figure 8 shows the kernel density estimation curves of three attributes, pax_fcny, pax_tax, recent_gap_day. On the whole, these curves in each cluster are not completely coincident, and there are significant differences, which show

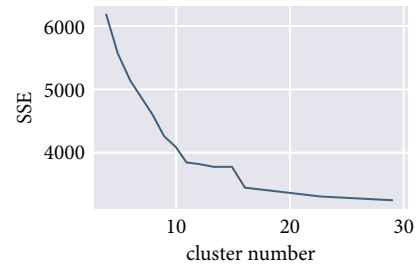

Figure 7: Silhouette coefficient of the k-means2o algorithm on passenger seat selection dataset.

Table 9: The number of core and noncore subsets elements $|S_{\mathrm{core}}|, |S_{noncore}|$ in k-means2o clustering.

Dataset	$\lvert S_{\mathrm{core}}\rvert$	$\lvert S_{noncore}\rvert$	Dataset	$\lvert S_{\mathrm{core}}\rvert$	$\lvert S_{noncore}\rvert$
Breast-cancer	398	171	Banknote	1280	92
Bupa	280	65	Compound	399	0
Ct	164	57	Hayes-roth	102	30
Iris	146	4	Libras	356	4
Parkinsons	148	47	Penbased	10482	510
Vowel	960	30	waveform21	4740	260
Waveform40	4116	884	Wdbc	398	171
Wine	161	17			

Table 10: The airline seat selection dataset clustering result.

Cluster	Size	Payment	No-payment	Payment ratio (%)	adr (%)
$C0$	2580	185	2395	7.17	13.99
$C1$	1771	127	1644	7.17	13.99
$C2$	1139	30	1109	2.63	58.19
$C3$	1256	107	1149	8.52	35.45
$C4$	1596	43	1553	2.69	57.23
$C5$	1582	112	1470	7.08	12.56
$C6$	761	22	739	2.89	54.05
$C7$	1935	166	1769	8.58	36.41
$C8$	1930	140	1790	7.23	14.94
$C9$	1623	67	1556	4.13	34.34
$C10$	2023	176	1847	8.70	38.31
$C11$	695	21	674	3.02	51.99
$C12$	1438	58	1380	4.03	35.93
$C13$	379	8	371	2.11	66.45
$C14$	909	67	842	7.37	17.17
$C15$	1815	146	1669	8.04	27.82

that the data distribution of each cluster is different. This conclusion is consistent with the expectation of cluster analysis, that is, the samples between clusters are dissimilar as much as possible. From a single attribute point of view, the discrimination of pax_fcny attribute is the most significant, with different mean point, peak point, and data span. Followed by pax_tax attribute. The third one is recent_gap_day attribute. Its mean and span are very similar, but the peak point is still different. The difference of peak points indicates that there are differences in the

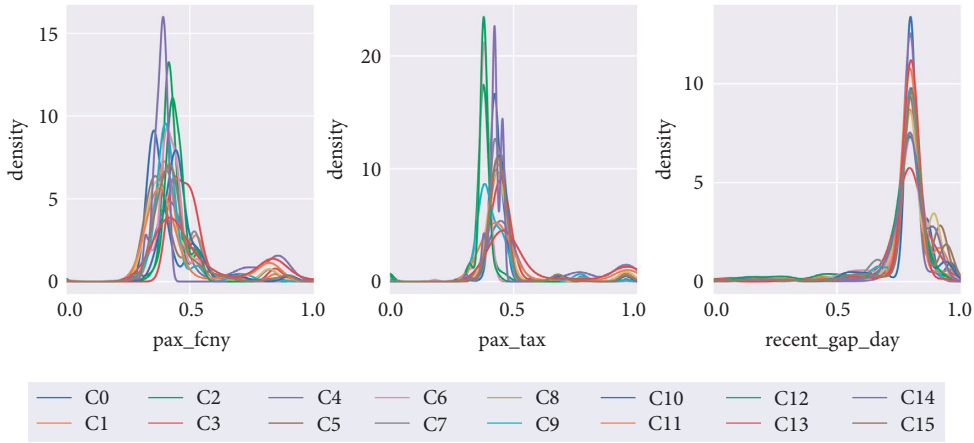

Figure 8: Kernel density estimation curves of pax_fcny, pax_tax, and recent_gap_day attributes.

concentration of data distribution in the cluster. The larger the peak value, more points are distributed near the mean value.

Table 10 discusses the k-means2o algorithm clustering results of the airline seat selection dataset from the similarities within clusters and dissimilarities between the clusters. The clustering results will be a good reference basis for customer grouping. Air passenger grouping will enable the decision-makers to more accurately find the audience of the personalized recommendation service, such as payment for airline seat selection. The dataset shows the label of payment for airline seat selection. The adr value of each cluster is greater than 12%, which is significantly different from the payout rate of the entire dataset of 6.29%. The cluster with the largest adr value is C13, reaching 66.45%, and the one with the smallest *adr* value is c5, reaching 12.56%. These results show that passenger payment behavior within clusters is more agglomerated compared to the entire dataset. Since the payment rate of C13 is 2.11%, it is a reverse difference. In other words, the *adr* = 66.45% indicates that passengers in C13 are extremely unwilling to pay for seat selection, and the willingness to pay is significantly lower than the overall level. In 9 of the 16 clusters, the ratio of paying for airline seat selection exceeds 5%. According to the precise recommendation or personalized marketing strategy, enterprises should pay more attention to the passengers in these nine clusters, and their marketing is more likely to succeed. Compared with the passengers in other clusters, the ones in these clusters will be more willing to accept such recommendations and enhance their stickiness. On designing a recommendation system, this clustering result will become a good auxiliary prior information.

6. Conclusion

In this paper, two optimization methods for k-means are initial center points selection and phased assignment were proposed, and then the improved k-means algorithm, k-means2o, were proposed. In contrast to the previously introduced algorithms, k-means ++, K-MC^2, and AFK-MC^2, the new initial center points selection optimization redefines the first center point selection strategy and the subsequent center point incremental strategy. The phased assignment optimization adopted the Tukey's rule to divide dataset into core and noncore sets, then two assignment strategies were proposed corresponding to the core and noncore sets, respectively. These two optimization methods complement each other to form combinatorial optimization. The experimental results on 15 real-world and 10 synthetic datasets show that the k-means2o outperforms its main competitor k-means ++, and under the same setting conditions, namely using the default parameters, the clustering performance of k-means2o is better than affinity propagation, mean shift, and DBSCAN.

The improved k-means algorithm, k-means2o, is applied to analyze the airline seat selection dataset. Combined with the data label of paying for seat selection, the clustering results realize customer grouping, and find suitable audience group for the recommendation of seat selection services. Through the analysis of the newly defined absolute deviation rate *adr* index, the appropriate groups for service recommendation are found, and the groups that are not suitable for recommendation are distinguished. Therefore, the airline enterprises can use limited resources to promote the groups with high-payment willingness, improve the success rate, and avoid promoting seat selection services to the groups with low-payment willingness which not only wastes resources but also causes passengers' disgust.

After a lot of experimental tests, the k-means2o algorithm, like other algorithms, cannot be adapted to all fields and situations, such as high-dimensional sparse data. If the data are a huge number of attributes or higher dimensions, it will easily lead to fewer samples in S_{core}, and in extreme cases, it may be less than the number of clusters. The Olivetti Face image data with $112 * 92 = 10304$ dimension have been tested and found that $|S_{core}| < 40$, that is, the number of samples in the core set is less than the number of clusters; therefore, the clustering fails. Due to the division of the core and noncore sets, the k-means2o algorithm is not suitable for huge number of attributes or higher dimensions. We will continue to study this problem and hope to solve this problem in the future.

Data Availability

The data are available at https://gitee.com/ydh-usx/k-means2o-data/tree/master/data.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This work was supported by the National Natural Science Foundation of China (No. 62002227) and the School-level scientific research project of Shaoxing University (No. 2021LG004).

References

[1] J. Han, J. Pei, and M. Kamber, *Data Mining: Concepts and Techniques*, Elsevier, Amsterdam, Netherlands, 2011.

[2] B. J. Frey and D. Dueck, "Clustering by passing messages between data points," *Science*, vol. 315, no. 5814, pp. 972–976, 2007.

[3] A. Rodriguez and A. Laio, "Clustering by fast search and find of density peaks," *Science*, vol. 344, no. 6191, pp. 1492–1496, 2014.

[4] M. Parmar, D. Wang, A. H. Tan, C. Miao, J. Jiang, and Y. Zhou, "A novel density peak clustering algorithm based on squared residual error," in *Proceedings of the 2017 International Conference on Security, Pattern Analysis, and Cybernetics (SPAC)*, pp. 43–48, IEEE, Shenzhen, China, 2017.

[5] M. Parmar, D. Wang, X. Zhang et al., "REDPC: a residual error-based density peak clustering algorithm," *Neurocomputing*, vol. 348, pp. 82–96, 2019.

[6] M. D. Parmar, W. Pang, D. Hao et al., "FREDPC: a feasible residual error-based density peak clustering algorithm with the fragment merging strategy," *IEEE Access*, vol. 7, pp. 89789–89804, 2019.

[7] H. Xie, L. Zhang, C. P. Lim et al., "Improving k-means clustering with enhanced firefly algorithms," *Applied Soft Computing*, vol. 84, Article ID 105763, 2019.

[8] N. Shah, D. Patel, and P. Fränti, "k-means image segmentation using mumford–shah model," *Journal of Electronic Imaging*, vol. 30, no. 06, Article ID 063029, 2021.

[9] A. R. Khan, S. Khan, M. Harouni, R. Abbasi, S. Iqbal, and Z. Mehmood, "Brain tumor segmentation using k-means clustering and deep learning with synthetic data augmentation for classification," *Microscopy Research and Technique*, vol. 84, no. 7, pp. 1389–1399, 2021.

[10] A. Moubayed, M. Injadat, A. Shami, and H. Lutfiyya, "Student engagement level in an e-learning environment: clustering using k-means," *American Journal of Distance Education*, vol. 34, no. 2, pp. 137–156, 2020.

[11] X. Qian, M. Di Renzo, and A. Eckford, "K-means clustering-aided non-coherent detection for molecular communications," *IEEE Transactions on Communications*, vol. 69, no. 8, pp. 5456–5470, 2021.

[12] Z. Xu, D. Shen, T. Nie, Y. Kou, N. Yin, and X. Han, "A cluster-based oversampling algorithm combining smote and k-means for imbalanced medical data," *Information Sciences*, vol. 572, pp. 574–589, 2021.

[13] H. Liu, J. Wu, T. Liu, D. Tao, and Y. Fu, "Spectral ensemble clustering via weighted k-means: theoretical and practical evidence," *IEEE Transactions on Knowledge and Data Engineering*, vol. 29, no. 5, pp. 1129–1143, 2017.

[14] A. A. Aldino, D. Darwis, A. T. Prastowo, and C. Sujana, "Implementation of k-means algorithm for clustering corn planting feasibility area in south lampung regency," *In Journal of Physics: Conference Series*, vol. 1751, Article ID 012038, 2021.

[15] Z.-S. Chen, X. Zhang, W. Pedrycz, X.-J. Wang, K.-S. Chin, and L. Martínez, "K-means clustering for the aggregation of HFLTS possibility distributions: N-two-stage algorithmic paradigm," *Knowledge-Based Systems*, vol. 227, Article ID 107230, 2021.

[16] P. Fränti and S. Sieranoja, "How much can k-means be improved by using better initialization and repeats?" *Pattern Recognition*, vol. 93, pp. 95–112, 2019.

[17] S. Huang, Z. Kang, Z. Xu, and Q. Liu, "Robust deep k-means: an effective and simple method for data clustering," *Pattern Recognition*, vol. 117, Article ID 107996, 2021.

[18] S. Vassilvitskii, "K-means: The advantages of careful seeding," in *Proceedings of the Eighteenth Annual ACM-SIAM Symposium on Discrete Algorithms*, pp. 1027–1035, SODA 2007, New Orleans, Louisiana, USA, 2006.

[19] D. Warnock-Smith, J. F. O'Connell, and M. Maleki, "An analysis of ongoing trends in airline ancillary revenues," *Journal of Air Transport Management*, vol. 64, pp. 42–54, 2017.

[20] P. Chiambaretto, "Air passengers' willingness to pay for ancillary services on long-haul flights," *Transportation Research Part E: Logistics and Transportation Review*, vol. 147, Article ID 102234, 2021.

[21] B. Vinod, "Airline revenue planning and the covid-19 pandemic," *Journal of Tourism Futures*, vol. 8, no. 2, pp. 245–253, 2021.

[22] G. Gunardi and R. S. Moody, H. k Martono, Covid-19: the impact on air transportation tariff in Indonesia," in *Proceedings of the International Conference on Economics, Business, Social, and Humanities (ICEBSH 2021)*, pp. 344–349, Atlantis Press, Noord-Holland, The Netherlands, 2021.

[23] O. Bachem, M. Lucic, S. Hamed Hassani, and A. Krause, "Approximate k-means++ in sublinear time," in *Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence*, pp. 1459–1467, AAAI, Palo Alto, California, U.S, 2016.

[24] O. Bachem, M. Lucic, H. Hassani, and A. Krause, "Fast and provably good seedings for k-means," *Advances in Neural Information Processing Systems*, vol. 29, pp. 55–63, 2016.

[25] D. Comaniciu and P. Meer, "Mean shift: a robust approach toward feature space analysis," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, no. 5, pp. 603–619, 2002.

[26] E. Schubert, J. Sander, M. Ester, H. P. Kriegel, and X. Xu, "Dbscan revisited, revisited: why and how you should (still) use dbscan," *ACM Transactions on Database Systems*, vol. 42, no. 3, pp. 1–21, 2017.

[27] M. E. Celebi, H. A. Kingravi, and P. A. Vela, "A comparative study of efficient initialization methods for the k-means clustering algorithm," *Expert Systems with Applications*, vol. 40, no. 1, pp. 200–210, 2013.

[28] Y. Zhou, Y. Hong, and X. Cai, "A novel k-means algorithm for clustering and outlier detection," in *Proceedings of the 2009 Second International Conference on Future Information Technology and Management Engineering*, pp. 476–480, IEEE, Sanya, China, 2009.

[29] I. Sungjin, M. Montazer Qaem, B. Moseley, X. Sun, and R. Zhou, "Fast noise removal for k-means clustering," in *Proceedings of the International Conference on Artificial*

Intelligence and Statistics, pp. 456–466, PMLR, New York City, NY, USA, 2020.

[30] G. Gan and M. K. P. Ng, "K-means clustering with outlier removal," *Pattern Recognition Letters*, vol. 90, pp. 8–14, 2017.

[31] O. Peter and B. Twala, "K-means-sharp: modified centroid update for outlier-robust k-means clustering," in *Proceedings of the 2017 Pattern Recognition Association of South Africa and Robotics and Mechatronics (PRASA-RobMech)*, pp. 14–19, IEEE, Bloemfontein, South Africa, 2017.

[32] N. H. M. M. Shrifan, M. F. Akbar, and N. A. Mat Isa, "An Adaptive Outlier Removal Aided K-Means Clustering Algorithm Journal of King Saud University-Computer and Information Sciences," vol. 34, no. 8, 2021.

[33] D. Yu, G. Liu, M. Guo, X. Liu, and S. Yao, "Density peaks clustering based on weighted local density sequence and nearest neighbor assignment," *IEEE Access*, vol. 7, pp. 34301–34317, 2019.

[34] L. Wang, W. Sun, X. Han et al., "An improved integrated clustering learning strategy based on three-stage affinity propagation algorithm with density peak optimization theory," *Complexity*, vol. 2021, Article ID 6666619, 12 pages, 2021.

[35] N. Huyghues-Beaufond, S. Tindemans, P. Falugi, M. Sun, and G. Strbac, "Robust and automatic data cleansing method for short-term load forecasting of distribution feeders," *Applied Energy*, vol. 261, Article ID 114405, 2020.

[36] F. Pedregosa, G. Varoquaux, A. Gramfort et al., "Scikit-learn: machine learning in Python," *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011.

[37] D. Dua and C. Graff, *UCI Machine Learning Repository*, vol. 588, 2017.

[38] L. Liu and D. Yu, "Density peaks clustering algorithm based on weighted k-nearest neighbors and geodesic distance," *IEEE Access*, vol. 8, pp. 168282–168296, 2020.

[39] D. Yu, G. Liu, M. Guo, and X. Liu, "An improved k-medoids algorithm based on step increasing and optimizing medoids," *Expert Systems with Applications*, vol. 92, pp. 464–473, 2018.

[40] D. Steinley, "Properties of the hubert-arable adjusted rand index," *Psychological Methods*, vol. 9, no. 3, pp. 386–396, 2004.

[41] N. Xuan Vinh, J. Epps, and J. Bailey, "Information theoretic measures for clusterings comparison: variants, properties, normalization and correction for chance," *Journal of Machine Learning Research*, vol. 11, pp. 2837–2854, 2010.

[42] E. B. Fowlkes and C. L. Mallows, "A method for comparing two hierarchical clusterings," *Journal of the American Statistical Association*, vol. 78, no. 383, pp. 553–569, 1983.