Hindawi
Security and Communication Networks
Volume 2023, Article ID 9761814, 1 page
https://doi.org/10.1155/2023/9761814

Retraction

Retracted: An Ensemble Clustering Approach (Consensus Clustering) for High-Dimensional Data

Security and Communication Networks

Received 8 August 2023; Accepted 8 August 2023; Published 9 August 2023

This article has been retracted by Hindawi following an investigation undertaken by the publisher [1]. This investigation has uncovered evidence of one or more of the following indicators of systematic manipulation of the publication process:

(1) Discrepancies in scope

(2) Discrepancies in the description of the research reported

(3) Discrepancies between the availability of data and the research described

(4) Inappropriate citations

(5) Incoherent, meaningless and/or irrelevant content included in the article

(6) Peer-review manipulation

The presence of these indicators undermines our confidence in the integrity of the article's content and we cannot, therefore, vouch for its reliability. Please note that this notice is intended solely to alert readers that the content of this article is unreliable. We have not investigated whether authors were aware of or involved in the systematic manipulation of the publication process.

Wiley and Hindawi regrets that the usual quality checks did not identify these issues before publication and have since put additional measures in place to safeguard research integrity.

We wish to credit our own Research Integrity and Research Publishing teams and anonymous and named external researchers and research integrity experts for contributing to this investigation.

The corresponding author, as the representative of all authors, has been given the opportunity to register their agreement or disagreement to this retraction. We have kept a record of any response received.

References

[1] J. Yan and W. Liu, "An Ensemble Clustering Approach (Consensus Clustering) for High-Dimensional Data," *Security and Communication Networks*, vol. 2022, Article ID 5629710, 9 pages, 2022.

Hindawi
Security and Communication Networks
Volume 2022, Article ID 5629710, 9 pages
https://doi.org/10.1155/2022/5629710

Research Article

An Ensemble Clustering Approach (Consensus Clustering) for High-Dimensional Data

Jingdong Yan and Wuwei Liu

School of Management, Wuhan University of Technology, Wuhan 430070, Hubei, China

Correspondence should be addressed to Wuwei Liu; liuww02@whut.edu.cn

Received 4 April 2022; Revised 12 April 2022; Accepted 18 April 2022; Published 16 May 2022

Academic Editor: Mohammad Ayoub Khan

Due to the plurality of irrelevant attributes, sparse distribution, and complicated calculations in high-dimensional data, traditional clustering algorithms, such as K-means, do not perform well on high-dimensional data. To address the clustering problem of high-dimensional data, this paper studies an integrated clustering method for high-dimensional data. A method of subspace division based on minimum redundancy is proposed to solve the problem of subspace division of high-dimensional data; subspace division is improved by using the K-means algorithm. Additionally, this method uses mutual information between the characteristic variables of the data to replace the calculation in the K-means algorithm. The distance between the characteristic variables of the data is used to divide the data into subspaces according to the mutual information values between the characteristic variables of the data. To achieve high clustering accuracy and diversity based on clustering requirements, this paper uses a genetic algorithm as the consistency integration function. The fitness function is designed according to the clustering fusion target, and the selection operator is designed according to the maximum number of overlapping elements in the base clustering. The experimental results show that the clustering algorithm proposed in this paper outperforms other methods on most datasets and is an effective clustering integration algorithm. The proposed clustering algorithm is compared with other commonly used clustering fusion algorithms on datasets to prove the advantages of the proposed algorithm.

1. Introduction

Clustering algorithms were first proposed in a work on anthropological data in 1932. Interdisciplinary research on these algorithms has a history of more than 80 years [1–3]. Biologists, sociologists, economists, statisticians, mathematicians, engineers, computer scientists, medical researchers, and many other workers facing data processing have all contributed to the clustering method [4]. Since then, clustering has been one of the important research topics in related fields [5]. High-dimensional data clustering originated from medical and computer science, but with the development of big data technology, an increasing number of high-dimensional data have appeared in economic life. Cluster analysis is needed to analyze these high-dimensional economic data interpretations [6, 7]. Cluster analysis is used mainly for regression and classification, that is, to complete the prediction of new data with many labeled data (continuous data). In addition, in classification (discrete data), unsupervised learning is mainly cluster analysis, that is, distinguishing data without any manual intervention. Unlike classification, clustering groups unlabeled samples to discover the natural structure of the data [8, 9].

The main idea of ensemble learning is to first generate multiple learners through certain rules and then use an integration strategy to combine algorithms, and finally comprehensively judge and output the final result [10]. Generally, multiple learners in so-called ensemble learning are homogeneous "weak learners". Based on the weak learner, multiple learners are generated through sample set perturbation, input feature perturbation, output representation perturbation, and algorithm parameter perturbation [11]. After integration, a "strong learner" with good accuracy is obtained [12, 13]. With the deepening of ensemble

learning research, a broad definition of ensemble learning has been gradually accepted by scholars. Ensemble learning refers to the use of learning methods for multiple sets of learners without distinguishing the nature of the learners. According to this definition, multiple areas, such as multilearner systems, multiexpert mixes, and committee-based learning, can be incorporated into integrated learning. However, most studies on ensemble learning are still based on ensemble learning of homogeneous classifiers [14].

In ensemble learning, the differences between learners are considered to be one of the key factors affecting the results of ensembles [15]. The first step of clustering integration is to generate multiple clustering results through different types of clustering basis learners. The k-means principle is simple and fast, but it depends on the initial parameter settings to make the clustering results unstable and cannot effectively achieve target clustering of nonconvex shape distribution data. EM does not need to set the number of categories in advance, and the calculation results are stable and accurate, but the algorithm is relatively complex and the convergence is slow, thus making EM unsuitable for large-scale datasets and high-dimensional data [16]. The clustering result depends on subjective learning, and the hierarchical clustering has no target function. After the cluster is merged, it is irreversible, and the local optimum is used as the global optimal solution. Farthest First reduces the number of samples to be clustered and the number of categories during iteration, thereby streamlining the clustering results. Each algorithm has its own advantages and disadvantages, and the applicable scenarios differ; therefore, the algorithms need to be integrated to achieve complementary advantages [17].

High-dimensional problems are almost ubiquitous, and mathematical models established to solve such problems generally use many unknown parameters; even the number of unknown parameters (or explanatory variables) is much larger than the number of sample sizes [18, 19]. The introduction of high-dimensional features poses a great challenge for traditional statistical methods. When the number of unknown parameters increases sharply, it is difficult for researchers to guarantee the accuracy of traditional statistical methods in estimation and testing, while the explanatory variables increase greatly, researchers are prone to deviations in the analysis of specific variables, and the model explains the phenomenon under study. The strength cannot be guaranteed [20]; additionally, the increase in the model dimension will make the model optimization and solution extremely complicated, thus resulting in the difficulty of analysis. Additionally, when too many variables exist in the model, false correlations will frequently appear in the free combination of variables; this will bring about collinearity problems and conceal the inherent causality or correlation of the data itself [21]. Noise accumulation is also a very easy problem for analyzing high-dimensional data. This problem is due mainly to the random error accompanied by random variable data generation. When the model introduces too many variables, it also introduces too much noise and too many real signals. The effect cannot be reflected due to the accumulation of these noises, and the judgment made by researchers on the true data relationship is also affected [22]. In summary, pseudocorrelation and noise accumulation are the two most common problems that need to be avoided when statistically modeling high-dimensional data. If these two types of problems cannot be effectively dealt with in statistical modeling, they will easily fall into the dimensionality trap and greatly reduce the reliability of model identification and statistical inference [23, 24].

This paper proposes a minimum redundancy method for partitioning data into subspaces. Before introducing the minimum redundancy method, the characteristics of three commonly used correlation coefficients to measure the correlation between variables and how to use the regression linear equation to reflect the quantitative relationship between variables are introduced. The cluster fusion algorithm optimized by the genetic algorithm proposed in this paper has three main steps: (1) use K-means to set the same k value to generate base clusters; (2) unify the base cluster labels; and (3) use genetic algorithms. As a consistent integration function, base clustering evolves to obtain the final result.

2. Proposed Method

2.1. Cluster Integration Scheme. The integration method proposed in this article has two layers:

In the first layer, we use K-means on the dataset multiple times to obtain m basis clusters $\Pi = \{\pi_1, \pi_2, \ldots, \pi_m\}$. Each basis cluster is merged independently and iteratively until the number of specific classes K' is obtained, where $K' > K_{\text{real}}$ and K_{real} is the number of real classes. The combined base cluster $\Pi^{c'} = \{\pi_1', \pi_2', \ldots, \pi_m'\}$ is used to generate a refined association matrix.

In the second layer, each base cluster π_i is divided and merged through the refined association matrix to generate the final clustering result. The method of dividing π_i is to calculate the intraclass homogeneity of each base class in π_i through the refined association matrix CM, use this index to find the base class with lower homogeneity, and use a variant K of the base class. The K-means method performs secondary division to ensure that the homogeneity of the base class after the division is high and obtains the base cluster $\Pi^{\prime\prime} = \{\pi_1', \pi_2', \ldots, \pi_m'\}$ after the division. The merge method of π_i'' calculates the similarity between all base classes in π_i'' through the refined association matrix CM. Through this similarity measure, the single-linkage method is used to merge the base clusters. The higher the similarity is, the higher the degree of similarity. The number is the number of real classes, which is the final clustering result.

2.1.1. Generate Association Matrix. CM differs from the traditional association matrix in that CM contains more class structure information, less information is lost in converting the graphic distribution to digital description, and CM can better reflect the distribution relationship of objects in the dataset. The class structure information is rarely reflected in the initial base clustering because the base cluster is composed of divided small Gaussian clusters. Each Gaussian cluster has a simple structure and few data objects.

The generated correlation matrix is also insufficiently precise. Therefore, the base clustering needs to be merged.

The closer the merge is to the true division, the more class structure information is reflected, and the more accurate the generated association matrix is. Concerning the merging process, whether as part of the integration method or a common merging-based clustering, heterogeneous objects may merge in the next stage.

In the integrated method proposed in this paper, to obtain high homogeneity, many classes are generated through base clustering. For each base cluster, the similarity between cohesion classes is used to use the single-linkage method to base the cluster. The base classes in M are combined to a specific number of classes K', where K'> Kreal, and then a refined association matrix is generated by the merged base cluster. Cohesion similarity is a new method for measuring the similarity between classes. Cohesion similarity differs essentially from the similarity between classes proposed previously. Cohesion similarity considers the distribution of all objects between two classes and is unaffected by a few discrete objects. Cohesion similarity has globality and is a suitable method for measuring the similarity between classes. Assuming two classes C_i and C_j in an initial basis cluster, their cohesion similarity is defined as follows:

$$\text{cohesion}(Ci, Cj) = \frac{\sum_{p \in C_i, C_j} \min(fi(v), fi(v))}{|C_i| + |C_j|}, \tag{1}$$

where $|C_i|$ is the number of C_i points. In this method, $f(v)$ is the probability density function, d is the dimension of the data points, and u is the mean of the class. The probability density function $f(v)$ is defined as follows:

$$f(v) = (2\pi)^{-d/2} (\det\Psi)^{-1/2} \exp\left[-\frac{1}{2}\Delta^2(v)\right]. \tag{2}$$

In the above function:

$$\Delta^2(v) = (v-u)^T \Psi^{-1} (v-u). \tag{3}$$

According to the cohesion (Ci, Cj) defined above, the most similar pair of classes in the base cluster is determined to be merged first, and the similarity between classes is updated. The larger of the similarities between the two classes and other classes is determined. Then, the pair with the most similarity in the base cluster is refound to merge and update the similarity; these iterations are repeated until the maximum number of merges with the specified class is reached.

2.2. Subspace Clustering

2.2.1. Subspace Division Method Based on Minimal Redundancy between Variables. When dividing the subspace, multiple methods can be used for division. The more common method is the exhaustive method, that is, the size of the mutual information between each pair of feature variables in the data calculated in the first step. A feature variable is used to classify the subspaces. However, the overhead of this method is relatively large, so in this paper, we will use an improved K-means algorithm to partition subspaces with minimal redundancy between feature variables. The traditional K-means algorithm is a distance-based unsupervised clustering algorithm. Its central idea is to assign each data sample point to the class closest to the center point of a certain class. The objective function of the K-means algorithm is as follows:

$$J_{K-\text{means}} = \sum_{j=1}^{k} \sum_{x_i \in C_j} \left\| x_i - u_j \right\|^2. \tag{4}$$

In the above formula, u_j represents the center point of C_j, such as sample point x_i. $J_{K-\text{means}}$ is the sum of the square of the distance between the data sample point and its corresponding class center point. The goal of the K-means algorithm is to minimize $J_{K-\text{means}}$. The flow of the K-means algorithm is as follows:

Input: cluster number k, dataset $X = \{x_0, x_1, x_2, \ldots, x_{N-1}\}$;

Output: cluster partition $C = \{c_0, c_1, \ldots, c_{K-1}\}$;

(1) Randomly select K data samples as the initial clustering center;

(2) Calculate the distances between $x_0, x_1, x_2, \ldots, x_{N-1}$ and the center points $c_0, c_1, \ldots, c_{K-1}$. If the difference between c_i and c_i is the smallest, mark its class as i;

(3) For all instances marked as i, calculate the average value as the new center point ci;

(4) Repeat steps 2 and 3 until the change in the c_i value is less than a given threshold or the number of iterations reaches a maximum;

This paper uses the improved K-means algorithm for subspace partitioning and replaces the mutual information values between the characteristic variables of the calculated data with the distance between the points of the calculated data samples in the k-means algorithm. Mutual information is defined in the field of information theory as a way to measure the correlation between two event sets. However, in clustering integration, mutual information can be regarded as a kind of index information with symmetry and different distribution. The final output of the algorithm is our use of minimal redundancy. The subspace divided by the principle is the feature subset with the least redundancy in the data. The specific process of the MRFS subspace division method is as follows: First, arbitrarily select K from the data feature variables (K is the number of subspaces divided into N), feature variables as the initial target feature variables, and then use the remaining feature variables and the selected target feature variables to calculate mutual information. According to the size of the mutual information value, assign the feature variables to be most similar to them (the mutual information value is the largest), complete an iteration of the feature variables, calculate the sum of the mutual information values between the feature variables in each category, and then take the feature variable with the largest mutual information value from the original target feature variable in each class as the new target. Feature variables, and then perform a second iteration of the feature variables; after the

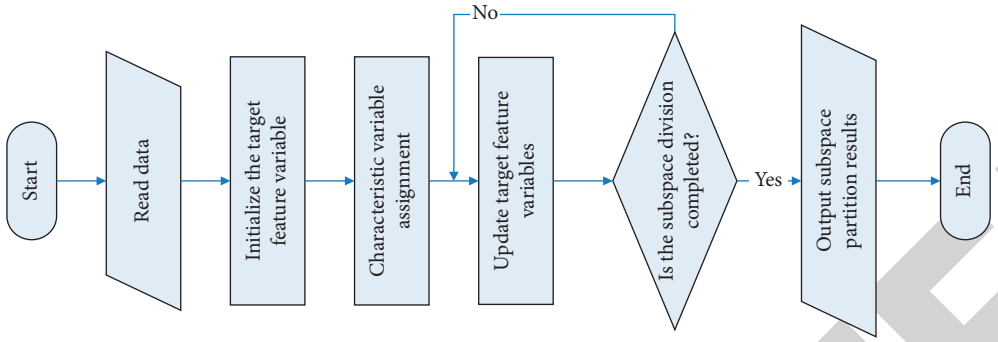

Figure 1: Molecular space flow chart.

second iteration, calculate the sum of the mutual information values between the feature variables in each category. Compare the size of the sum of the mutual information values obtained in the second iteration with the sum of the mutual information values obtained in the previous iteration, and repeat this process until the difference between the sum of the last mutual information value and the sum of the previous mutual information value is constant. The objective function of the minimum redundancy method is defined as follows:

$$H = \sum_{j=1}^{k} \sum_{x_i \in G_j} I\left(x_i; v_j\right). \tag{5}$$

In the above formula, v_j represents the target feature variable of the class to which the feature variable belongs. In dividing the data into subspaces by using the minimum redundancy method, the selection of the threshold is very important. The appropriateness of the selected threshold directly determines the quality of the subspace partition. In this paper, the threshold value is not uniformly selected as a threshold value but is selected according to the size of the experimental dataset. When the scale of the experimental data is large, the sum of the mutual information between the feature variables in each subspace is relatively large. When we set the threshold, we need to choose a larger value. The sum of the information is relatively small, and we need to choose a smaller value when setting the threshold. In this way, dynamically setting the experimental threshold according to the data scale can improve the final experimental results to a certain extent. The flow chart of the minimum redundant subspace partitioning method is shown in Figure 1.

The feature space of the data is divided into subspaces by using the minimum redundancy method. The results of the subspace partition are used to cluster the partition results to obtain the base clusterer, and then the consensus clusterer is selected to perform the base clusterer. Integration is used to obtain the final cluster integration result. Therefore, the working principle of subspace clustering integration based on the smallest redundant feature subset is shown in Figure 2:

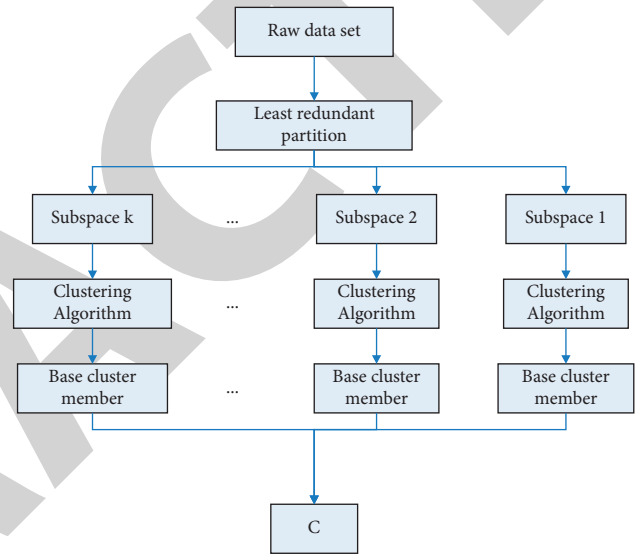

Figure 2: Principle of clustering ensemble with minimum redundant feature subsets.

2.2.2. Laplace Feature Mapping Method. Feature selection refers to removing redundant information from the original data and selecting representative feature subsets. For the clustering problem of high-dimensional data, unsupervised feature selection is usually used to preprocess the data, and then cluster on the selected feature subset. Unsupervised feature selection can be divided into three categories: filter method, wrapper method, and embedded method. Filter-based methods usually use indirect measurement to measure the importance of feature subsets, rather than the commonly used error function. Such methods usually use some statistical characteristics to select important features. For example, Laplacian score is a typical filter-based method. It measures the importance of features with the local retention ability of each feature and selects a representative feature subset by calculating the Laplacian score of each feature.

The Laplace feature mapping method makes up for the shortcomings of principal component analysis and increases the success rate of the latter by 30%. The Laplace feature mapping method combines mainly the neighborhood information of the dataset to construct a similarity graph. Each data point is a node in the similarity graph, and the connection between nodes can be obtained according to their neighboring points. The graph obtained in this way is a discrete approximation of a low-dimensional manifold. The

minimization of the graph-based objective function ensures that the mappings of similar points in the discrete manifold approximation in low-dimensional space are also close to each other. Suppose graph $G = (V, E)$ is composed of dataset X; V and E represent the vertex set and the edge set, respectively; the vertex of the graph is each data point; and the similarity or distance of each point is the edge of the graph. Assuming the mapping relationship between the high and low dimensions of $y = (y_1, y_2, \ldots, y_n)^T$, the objective function is as follows:

$$E(y) = \frac{1}{2}\sum_{i,j}\left(y_i, y_j\right)^2 W_{ij}. \tag{6}$$

Minimizing $E(y)$ means that the closer points in the low-dimensional manifold in feature extraction also maintain a small distance in the high-dimensional space. W in the above formula represents a weight matrix. Generally, a Gaussian kernel function is used to solve the distance between points. Assuming Ni represents the K-nearest neighbor of point xi, then:

$$W_{ij} = \begin{cases} exp\left(-\frac{\left\|x_i - x_j\right\|2}{t}\right), & x_j \in N_i, \\ 0, & \text{other.} \end{cases} \tag{7}$$

Suppose L represents the Laplacian of the weight matrix W and satisfies $L = D - W$, $D_{ii} = \sum_j W_{ij}$; and suppose L represents a semipositive definite matrix. Then:

$$\begin{aligned} E(y) &= \frac{1}{2}\sum_{i,j}\left(y_i - y_j\right)^2, \\ W_{ij} &= \frac{1}{2}\sum_{i,j}\left(y_i^2 + y_j^2 - 2y_i y_j\right), \\ W_{ij} &= y^T L y. \end{aligned} \tag{8}$$

When uniformly collecting data from low-dimensional manifolds in high-dimensional space, the Laplacian matrix in the manifold can be approximated by the Laplacian matrix on the high-dimensional data graph. However, the first eigenvector of the Laplacian matrix in the high-dimensional data graph is the discrete approximation of the Laplacian eigenfunction in the manifold. By calculating the eigenvectors of the Laplacian matrix, eigenvectors $v_0, v_l \ldots, v_d, v_0$ of d minimum eigenvalues can be obtained. The eigenvalue of $v0$ is 0, the eigenvectors $v_l \ldots, v_d$ are used to represent the low-dimensional coordinates after dimensionality reduction, and $y_i = (v_{ki})|_{k=1}^{d}$ can be obtained.

2.3. Cluster Fusion. This paper combines genetic algorithms to achieve cluster fusion. First, a basis cluster is randomly selected from m ($m \geq 2$) basis clusters as a benchmark. In general, the first basis cluster is selected as the benchmark. Then, the remaining m-1 basis clusters are matched with the benchmark, and cluster label transformation is performed. If $B_i = \{C_1^i, C_2^i, \ldots, C_k^i\}$ is taken as the benchmark, the base cluster $B_j = \{C_i^j, C_2^j, \ldots, C_k^j\}$ is matched, where C_k^j represents the data contained in the k-th cluster in the base cluster B_j. First, a $k \times k$ matrix is established to record the number of data overlapped by the clusters between the base cluster B_i and the base cluster B_j. Then, the maximum value in the matrix is selected, a matching relationship between the row and column is established, and the maximum row and column are assigned to -1. This step is repeated until the cluster labels in the base cluster B_j have a corresponding relationship with the base cluster Bf. Finally, according to the corresponding relationship, the cluster labels in the base cluster B_j are changed to the corresponding cluster labels in the benchmark B_i. Thus far, cluster label conversion based on the number of overlapping data of the base cluster B_i and B_j has been completed.

The clustering fusion algorithm (CEGA) based on the genetic algorithm has three main steps: (1) set the same k value with K-means to generate base clusters; (2) unify the base cluster labels; and (3) use the genetic algorithm. As a consistent integration function, the base clustering evolves to obtain the final result.

From the development of the clustering algorithm, a clustering algorithm is widely used and can be used in many practical problems, such as the Internet, e-commerce, biological information, and other fields. So far, the clustering algorithm has been developed for a long time. Although the algorithm has been proposed continuously, it still has certain limitations in practical application. For example, clustering on complex data will be difficult. At the same time, most clustering algorithms have the defect of high time complexity, and often need to consume a lot of time or space complexity in exchange for a slight improvement of the clustering effect. As shown in Figure 3.

3. Experiments

3.1. Experimental Dataset and Data Collation. The experimental data in this paper come from the UCI dataset, which is a commonly used standard set of machine learning test data. The dataset is a machine learning database maintained by the University of California, Irvine. Most of the tests for machine learning algorithms use UCI datasets. The important point is the word "standard". Newly compiled machine learning programs can be tested with UCI datasets. Similar machine learning algorithms can also be used for higher tests. The official website address of the UCI dataset is as follows: website: UCI Machine Learning Repository.

This article uses MATLAB to organize the data. The procedure for doing so is as follows: First, from the official website, download the dataset, such as the previously downloaded iris.data or a .txt file (self-named iris.txt) that you copied yourself. Create a new.m file in the folder where the file is located. The function of the finishing program is to read the original file data, change the English mark of the last column to a number of 1–3 for each category, and place the mark in the first column; additionally, resave the file consecutively as a new .txt file, a .mat file, and an .xls file. The last column of the .ris file is English letters. Directly using

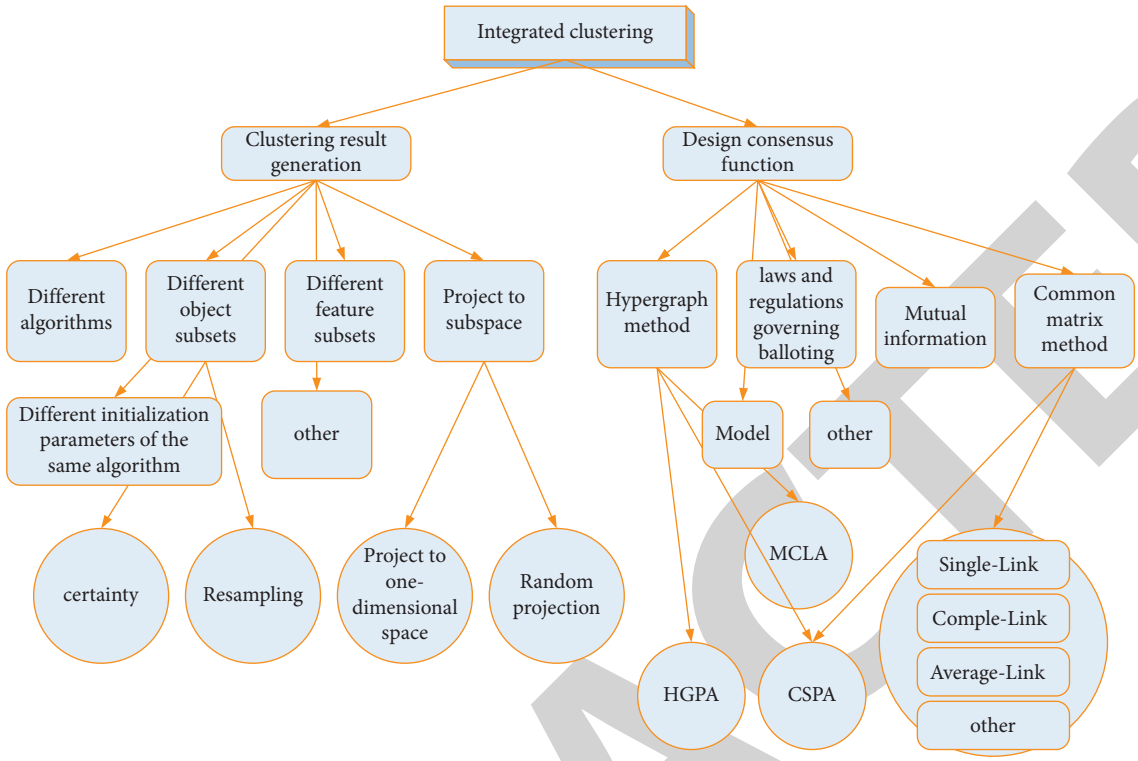

Figure 3: Various algorithms of integrated clustering.

MATLAB's load() function will cause an unknown error in the last column of text, so the textscan() function is used here. The MATLAB program for collating the Iris dataset is as follows. First, textscan() is used to read a cell array of $1 * 5$ cells, each element is an array of $150 * 1$ double, and all the data are stored in a column. Obviously, the last element is an array of all the tags. We iterate through these tags and record the index of the data with the same kind of tags. According to the index of all the data of each type recorded, the data of each type can be removed and renumbered.

3.2. Evaluation Index. To evaluate the quality of the clustering results, this paper uses three classic indicators: clustering accuracy (ACC), normalized mutual information (NMI), and the adjusted Rand index (ARI). The clustering accuracy is the comparison formula of each clustering result and the real result:

$$\mathrm{ACC} = \frac{N_{\mathrm{cor}}}{N}, \tag{9}$$

where N is the total number of documents and Ncor is the number of documents that are correctly clustered; the mutual information is worth calculating the degree of correlation between two random variables as follows:

$$I(X,Y) = \sum_{x,y} p(x,y)\log\frac{p(x,y)}{p(x)p(y)}. \tag{10}$$

Normalized mutual information is the normalization of mutual information to 0~1, usually divided by the maximum entropy. The calculation formula is as follows:

$$\begin{aligned} U(X,Y) &= 2R, \\ &= 2\frac{I(X;Y)}{H(X)+H(Y)}, \end{aligned} \tag{11}$$

where $H(X)$ is the maximum entropy. The RI coefficient needs to give the actual category information. Assuming K is the clustering result, a represents the logarithms of elements of the same category in C and K, and b represents the logarithms of elements of different categories in C and K as follows:

$$RI = \frac{a+b}{C_2^{n_{\mathrm{samples}}}}. \tag{12}$$

For random results, RI cannot guarantee a score close to zero. To make it possible to satisfy the requirement that "the index should be close to zero when the clustering results are randomly generated", an adjusted Rand index (ARI), which has a higher degree of discrimination, is proposed. The specific definition of the ARI is as follows:

$$ARI = \frac{RI - E[RI]}{\max(RI) - E(RI)}. \tag{13}$$

The upper bounds of the three evaluation indexes are all 1, and the larger the value is, the better the clustering result.

4. Results and Discussion

4.1. Dataset Homogeneity Analysis. To obtain different base clusters to generate the association matrix, this paper uses K-means for multiple clustering, where the number of

clusters K is set to $1.5\sqrt{N}$. After the base clustering is merged to a specific number of classes K', many experiments on multiple datasets have found that the base clustering is rarely mistakenly merged before it is merged into $K' = K_{\text{real}} + 5$, where K_{real} is the true of the dataset. The number of classes and the many erroneous merges in the subsequent merging step greatly reduce the accuracy of the base clustering, and undoubtedly, the accuracy of the association matrix will also be affected. Therefore, to improve the accuracy of the association matrix, $K' = K_{\text{real}} + 5$ is set here. As shown in Table 1.

4.2. Effectiveness Analysis of the Algorithm. To verify the effectiveness of the execution efficiency optimization strategy, $K = 4$, 8, 10, and 14 were used to cluster the sample set by using the traditional K-means algorithm, DK-means algorithm, and the algorithm proposed in this paper. The experimental comparison results are shown in Figure 4. As Figure 4 shows, the clustering time overhead of the proposed algorithm is lower than that of the DK-means algorithm and the traditional K-means algorithm, and the optimization efficiency is highest at $K = 5$. These results show that the execution efficiency of the high-dimensional data clustering algorithm with the improved strategy is higher than that of the DK-means algorithm, thus proving the effectiveness of the improved strategy. In addition, the proposed algorithm reduces the time cost of iteration by reducing the number of invalid calculations during each iteration. Therefore, the comparison results of the clustering time costs also show that the convergence speed of the proposed clustering algorithm outpaces that of the traditional K-means algorithm. Because the execution efficiency optimization strategy makes up for the time waste caused by the DK-means algorithm's selecting the initial clustering center, in the actual experimental process, the initial clustering center of the traditional K-means algorithm was selected better and did not fall into the local optimal solution. Under these circumstances, the traditional K-means algorithm does not necessarily consume more clustering time than does the proposed algorithm. However, due to the instability of the traditional K-means algorithm, this is not reflected in the comparison results of the average of 100 experiments.

4.3. Comparative Analysis of Different Methods. We choose the clustering results of several mature clustering integration methods for comparison: the connection-based integration algorithm (LCE) and hypergraph-based integration algorithm (JMLR). For connection-based integration, we use three spectral clustering-based algorithms: weighted connected-triple (WCT), weighted triple quality (WTQ), and hypergraph-based integration. We choose three clustering integration algorithms: the cluster-based similarity partitioning algorithm (CSPA) and the hypergraph partitioning algorithm (HGPA).

This paper uses these algorithms to perform experiments on the six datasets mentioned above and calculates the average accuracy of the clustering results for a comprehensive comparison. Table 2 and Figure 5 show the average

Table 1: Homogeneity of datasets merged into different stages.

	$K+6$	$K+5$	$K+4$	$K+3$	$K+2$	$K+1$	K
Iris	1	1	0.99	0.981	0.96	0.92	0.85
Ionosphere	0.96	0.959	0.94	0.92	0.915	0.9	0.86
Jain	0.98	0.98	0.97	0.965	0.96	0.93	0.81

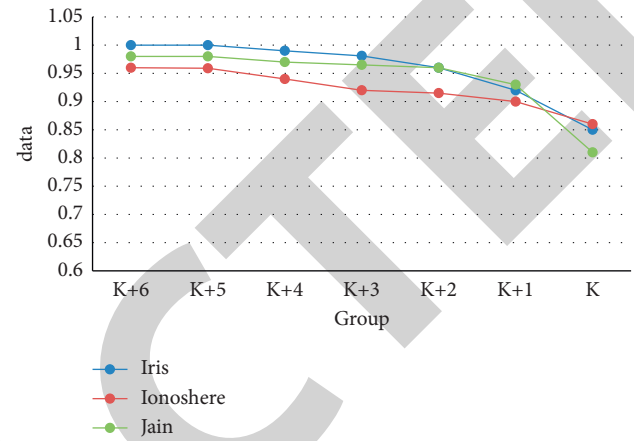

Figure 4: Homogeneity of the dataset merged into different stages.

Table 2: Accuracy of 300 clustering results on 6 datasets.

Method	The proposed algorithm	WCT	WTQ	CSPA	HGPA
Iris	0.921	0.923	0.966	0.909	0.932
Ionosphere	0.682	0.677	0.623	0.648	0.680
Jain	1	1	1	1	1
Spiral	1	0.833	0.914	0.856	0.900
S1	0.952	0.963	0.961	0.911	0.897
R15	1	1	1	1	1

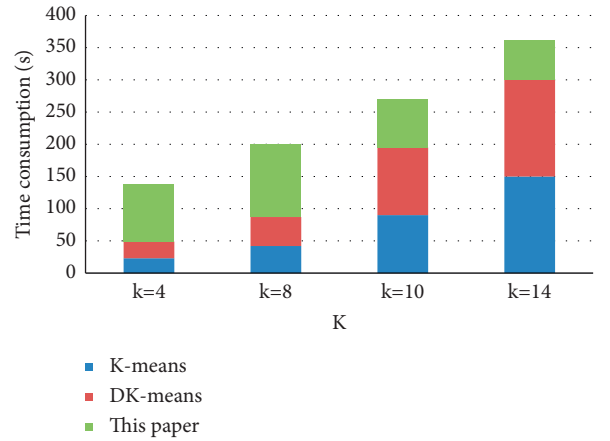

Figure 5: Comparison of the algorithms' time overheads.

index of clustering accuracy 300 times for 3 artificial datasets and 3 real datasets. The closer the index is to 1, the greater the consistency and the better the clustering effect. The method proposed in this paper has a better integration effect on the six datasets of Iris, Ionosphere, Jain, Spiral, S1, and R15. Both the table and graph show that the proposed

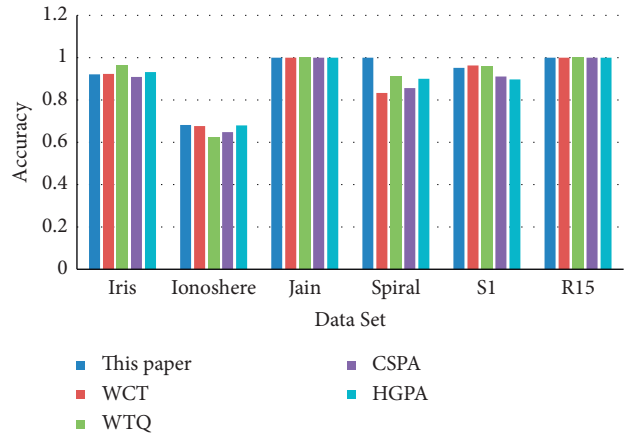

Figure 6: Accuracy of 300 clustering results on 6 datasets.

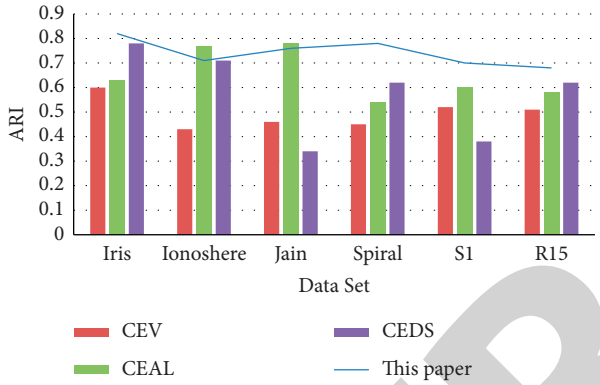

Figure 7: Comparison of commonly used cluster fusion algorithms and the improved cluster fusion algorithm of the genetic algorithm proposed in this paper.

method outperforms the other algorithms. The integration of the S1 and Ionosphere datasets is close to the best algorithm. In summary, although the proposed method is slightly less effective than the other methods on some datasets, the proposed method is more effective than the other methods on most datasets and is an effective clustering integration algorithm.

4.4. Performance Analysis of Clustering Fusion Algorithm. In this paper, we compare the average ARI values of the common clustering fusion algorithm and the genetic algorithm improved clustering fusion algorithm proposed in this paper. The CEV algorithm is a clustering fusion based on the voting method, the CEAL algorithm is a clustering fusion based on single-chain agglomeration clustering, and the proposed algorithm is a clustering fusion based on Dempster–Shafer evidence theory. The results of running four cluster fusion algorithms on six UCI datasets are shown in Figure 6. This figure shows that the proposed algorithm significantly outperforms the other three cluster fusion algorithms on the Iris, Spiral, S1, and R15 datasets; on the Ionosphere and Jain datasets, the algorithm slightly outperforms the CEDS algorithm. Therefore, the proposed algorithm is the best performing algorithm among the four cluster fusion algorithms. Except for the Ionosphere and Jain datasets, the CEV algorithm performs significantly worse than other algorithms, so the CEV algorithm is the worst performer among the 4 algorithms. The results show that the proposed CEGA algorithm improves the genetic algorithm as a consistent integration function and improves the fitness of the base clustering by performing crossover operations and gene mutations to improve the fitness of the base clustering (the data in the clusters are similar, and the data between clusters are slightly similar) to obtain the global optimal result. This is why the CEGA algorithm is superior to other cluster fusion algorithms as shown in Figure 7.

5. Conclusions

Clustering technology uses the similarity calculation method to determine the similarity between data samples. The dataset is divided into different classes according to the similarity between different data samples. The rules for dividing classes are data sample points that belong to different classes. The similarity between these data sample points is the smallest, and the similarity between the data sample points belonging to the same category is the largest. Social and human production practices have produced much big data. Big data has the following characteristics: a large data volume, diverse data types, low value density, and a fast processing speed. The large volume of data is reflected mainly in the number and dimensions of data, and subspace clustering is one of the methods used to solve high-dimensional data. In another aspect, the "dimensional effect" of big data greatly reduces the clustering effectiveness of many traditional clustering algorithms. The processing of high-dimensional data has become one of the main tasks facing machine learning at this stage.

The proposed algorithm introduces mainly how to divide the subspace of the data feature variables according to the minimum redundancy, calculate the mutual information values between the feature variables, use the maximum correlation between variables, and add the minimum redundancy to limit the choice of feature variables.

Addressing the disadvantages of the basic clustering accuracy of general clustering fusion algorithms that are affected by the clustering algorithm, this paper proposes the use of a genetic algorithm that can simulate the natural evolution process to search for the optimal solution as a consistent integration function and adaptability to the genetic algorithm. Functions and selection operators were improved. To encode the basic clusters generated by the clustering algorithm to complete the cluster label conversion, the initial population was at first determined. Then, the selection operator proposed in this paper selects chromosomes for crossover and mutation operations to evolve the population, and the chromosome fitness function designed in this paper selects the elites in the population. [25–28].

Data Availability

No data were used to support this study.

Ethical Approval

This article does not contain any studies with human participants performed by any of the authors.

Conflicts of Interest

All authors declare that they have no conflicts of interest.

References

[1] S. Aghabozorgi, A. Seyed Shirkhorshidi, and T. Ying Wah, "Time-series clustering - a decade review," *Information Systems*, vol. 53, no. C, pp. 16–38, 2015.

[2] I. S. Dhillon, S. Mallela, and D. S. Modha, "Information-theoretic co-clustering," in *Proceedings of the ACM KDD Conference*, pp. 89–98, New York, NY, USA, August 2015.

[3] S. Wang, D. Wang, C. Li, Y. Li, and G. Ding, "Clustering by fast search and find of density peaks with data field," *Chinese Journal of Electronics*, vol. 25, no. 3, pp. 397–402, 2016.

[4] C. F. Olson, "Parallel algorithms for hierarchical clustering," *Pattern Analysis & Machine Intelligence IEEE Transactions on*, vol. 12, no. 11, pp. 1088–1092, 2016.

[5] N. M. Kopelman, J. Mayzel, M. Jakobsson, N. A. Rosenberg, and I. Mayrose, "Clumpak: a program for identifying clustering modes and packaging population structure inferences across K," *Molecular Ecology Resources*, vol. 15, no. 5, pp. 1179–1191, 2015.

[6] T. P. Jacobs, J. Michelsen, J. S. Polay, A. C. D'Adamo, and R. E. Canfield, "Giant cell tumor in paget's disease of bone.Familial and geographic clustering," *Cancer*, vol. 44, no. 2, pp. 742–747, 1979.

[7] D. Feldman, M. Schmidt, and C. Sohler, "Turning big data into tiny data: constant-size coresets for K-Means, PCA and projective clustering," 2018, https://arxiv.org/abs/1807.04518.

[8] J. A. Aslam, E. Pelekhov, and D. Rus, "The star clustering algorithm for static and dynamic information organization," *Graph Algorithms and Applications 5*, vol. 8, no. 1, pp. 95–129, 2006.

[9] S. A. Hardy, L. Boddy, C. W. Morris, and M. F. Wilkins, "Comparison of five clustering algorithms to classify phytoplankton from flow cytometry data," *Cytometry*, vol. 44, no. 3, pp. 210–217, 2015.

[10] C. T. Forbes, L. Zangori, and C. V. Schwarz, "Empirical validation of integrated learning performances for hydrologic phenomena: 3rd-grade students' model-driven explanation-construction," *Journal of Research in Science Teaching*, vol. 52, no. 7, pp. 895–921, 2015.

[11] G. Rayner and T. Papakonstantinou, "Student perceptions of their workplace preparedness: making work-integrated learning more effective," *Asia-Pacific Journal of Cooperative Education*, vol. 16, no. 1, pp. 13–24, 2015.

[12] T. Winchester-Seeto, A. D. Rowe, and J. Mackaway, "Sharing the load: understanding the roles of academics and host supervisors in work-integrated learning," *Asia-Pacific Journal of Cooperative Education*, vol. 17, no. 2, pp. 101–118, 2016.

[13] C. Cameron, "The strategic and legal risks of work-integrated learning: an enterprise risk management perspective," *Social Science Electronic Publishing*, vol. 18, no. 3, pp. 243–256, 2017.

[14] W. Parker, "WE-D-204-03: CAMPEP residencies in a Canadian context: comprehensive cancer centers and integrated learning environments," *Medical Physics*, vol. 43, p. 3808, 2016.

[15] P. Phatisena, T. Eaksanti, P. Wichantuk et al., "Behavioral modification regarding liver fluke and cholangiocarcinoma with a health belief model using integrated learning," *Asian Pacific Journal of Cancer Prevention*, vol. 17, no. 6, pp. 2889–2894, 2016.

[16] C. Bilsland and H. Nagy, "Work-integrated learning in Vietnam: perspectives of intern work supervisors," *Asia-Pacific Journal of Cooperative Education*, vol. 16, no. 3, pp. 185–198, 2015.

[17] S. Ferns, L. Russell, and J. Kay, "Enhancing industry engagement with work-integrated learning: capacity building for industry partners," *Asia-Pacific Journal of Cooperative Education*, vol. 2016, no. 174, pp. 363–375, 2016.

[18] L. Jing, K. Tian, and J. Z. Huang, "Stratified feature sampling method for ensemble clustering of high dimensional data," *Pattern Recognition*, vol. 48, no. 11, pp. 3688–3702, 2015.

[19] A. Kaur and A. Datta, "A novel algorithm for fast and scalable subspace clustering of high-dimensional data," *Journal of Big Data*, vol. 2, no. 1, p. 17, 2015.

[20] L. M. Weber and M. D. Robinson, "Comparison of clustering methods for high-dimensional single-cell flow and mass cytometry data," *Cytometry, Part A*, vol. 89, no. 12, pp. 1084–1096, 2016.

[21] M. Śmieja, K. Hajto, and J. Tabor, "Efficient mixture model for clustering of sparse high dimensional binary data," *Data Mining and Knowledge Discovery*, vol. 33, no. 12, 2017.

[22] S. Chormunge and S. Jena, "Efficiency and effectiveness of clustering algorithms for high dimensional data," *International Journal of Computer Application*, vol. 125, no. 11, pp. 35–40, 2015.

[23] X. U. Xueli and Z. Xuejing, "Application of sparse spectral clustering algorithm in high-dimensional data," *Journal of University of Science and Technology of China*, vol. 47, no. 4, pp. 311–319, 2017.

[24] J. Shao, X. Wang, Q. Yang, C. Plant, and C Böhm, "Synchronization-based scalable subspace clustering of high-dimensional data," *Knowledge and Information Systems*, vol. 52, no. 1, pp. 83–111, 2016.