

Article

Similarity-Based Three-Way Clustering by Using Dimensionality Reduction

Anlong Li ¹, Yiping Meng ^{2,*} and Pingxin Wang ²

¹ School of Computer, Jiangsu University of Science and Technology, Zhenjiang 212100, China; 221210701313@stu.just.edu.cn

² School of Science, Jiangsu University of Science and Technology, Zhenjiang 212100, China; wangpingxin@just.edu.cn

* Correspondence: ypmeng@just.edu.cn

Abstract: Three-way clustering uses core region and fringe region to describe a cluster, which divide the dataset into three parts. The division helps identify the central core and outer sparse regions of a cluster. One of the main challenges in three-way clustering is the meaningful construction of the two sets. Aimed at handling high-dimensional data and improving the stability of clustering, this paper proposes a novel three-way clustering method. The proposed method uses dimensionality reduction techniques to reduce data dimensions and eliminate noise. Based on the reduced dataset, random sampling and feature extraction are performed multiple times to introduce randomness and diversity, enhancing the algorithm's robustness. Ensemble strategies are applied on these subsets, and the k-means algorithm is utilized to obtain multiple clustering results. Based on these results, we obtain co-association frequency between different samples and fused clustering result using the single-linkage method of hierarchical clustering. In order to describe the core region and fringe region of each cluster, the similar class of each sample is defined by co-association frequency. The lower and upper approximations of each cluster are obtained based on similar class. The samples in the lower approximation of each cluster belong to the core region of the cluster. The differences between lower and upper approximations of each cluster are defined as fringe region. Therefore, a three-way explanation of each cluster is naturally formed. By employing various UC Irvine Machine Learning Repository (UCI) datasets and comparing different clustering metrics such as Normalized Mutual Information (NMI), Adjusted Rand Index (ARI), and Accuracy (ACC), the experimental results show that the proposed strategy is effective in improving the structure of clustering results.

Keywords: three-way clustering; co-association frequency; dimension reduction; similar classes

MSC: 68T37; 62A86



Citation: Li, A.; Meng, Y.; Wang, P. Similarity-Based Three-Way Clustering by Using Dimensionality Reduction. *Mathematics* **2024**, *12*, 1951. <https://doi.org/10.3390/math12131951>

Academic Editor: Antonella Basso

Received: 15 May 2024

Revised: 16 June 2024

Accepted: 18 June 2024

Published: 24 June 2024



Copyright: © 2024 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

As an unsupervised technique in data mining and machine learning, cluster analysis is widely used in various areas such as attribute reduction [1–4], feature selection [5–7], image processing [8,9], information granulation [10–12], and graph convolutional neural networks [13–15]. The primary objective of clustering is to organize heterogeneous data into meaningful groups based on their similarities, revealing the inherent structures and patterns within the dataset. To achieve this, various clustering algorithms [16] have been developed. However, it has been accepted that a single clustering algorithm cannot handle all types of data distribution effectively. Different algorithms or different parameters for an algorithm may lead to different clustering results. To enhance the robustness and stability of clustering algorithms, researchers have proposed ensemble clustering methods. In comparison to single clustering methods, ensemble clustering methods [17–22] integrate results from multiple foundational clustering algorithms, yielding more stable, robust,

and accurate clustering solutions. Nevertheless, existing ensemble clustering methods typically adopt a hard clustering strategy, where an element can belong to only one cluster or none, and clear boundaries exist between different clusters. However, in situations with insufficient information on data samples, hard clustering algorithms often lead to higher decision risks.

To address this issue, three-way decision theory [23,24] was introduced to describe uncertainties in information. This method divides the sample universe into three mutually exclusive regions and adopts different decision strategies for each region [25,26]. The three-way decision framework can be integrated with various computational models for learning uncertainty, such as rough set theory [27–29], Bayesian networks [30,31], and fuzzy particle swarm optimization [32,33]. Inspired by the idea of three-way decision, Yu [34] presented the framework of three-way clustering by using core and the fringe regions to character a cluster. These two sets partition the sample space into three parts, which capture three kinds of relationships between objects and a cluster, namely, belonging to, partially belonging to, and not belonging to [35–38].

Recently, three-way clustering [39] has garnered widespread research interest, leading to the development of various three-way clustering algorithms within this theoretical framework. Wang and Yao [40] proposed a three-way clustering framework called CE3, derived from mathematical morphology's erosion and dilation concepts. Li et al. [41] introduced sample's stability to identify and establish relationships in ensemble clustering. Yu et al. [42] proposed an efficient three-way clustering algorithm based on the idea of universal gravitation. Jia et al. [43] developed an automatic three-way clustering approach by combining the proposed threshold selection and the cluster number selection method. Wang et al. [44] proposed a three-way adaptive density peak clustering (3W-ADPC) method by integrating natural nearest neighbors with DPC.

Most of the existing three-way clustering algorithms are based on the original dataset, which is not suitable for high-dimensional datasets. The processing of high-dimensional data poses a fundamental yet highly challenging problem in the current field of data science. The purpose of dimensionality reduction is to decrease the data's dimensionality while retaining the most significant aspects of its characteristics. By reducing the data's dimensionality, we can simplify the complexity of data analysis, enhance model training speed, reduce storage requirements, and facilitate a clearer understanding and interpretation of the model's results. Various dimensionality reduction techniques are commonly employed to address this challenge, including Principal Component Analysis (PCA) [45–47], spectral clustering [48,49], factor analysis [50], and multidimensional scaling [51].

By integrating dimensionality reduction into three-way clustering, this paper presents an ensemble three-way clustering algorithm based on dimensionality reduction. The proposed method uses dimensionality reduction techniques to reduce data dimensions and eliminate noise. Based on the reduced dataset, random sampling and feature extraction are performed multiple times to introduce randomness and diversity, enhancing the algorithm's robustness. Ensemble strategies are applied on these subsets, and the k-means algorithm is utilized to obtain multiple clustering results. Based on these results, the frequency of different data points being assigned to the same cluster is calculated to derive the co-occurrence frequency. If the co-occurrence frequency between data points exceeds a certain threshold, they are defined as similar classes. Finally, a three-way clustering approach was introduced by using the proposed similar relations. The main contributions of this research are as follows:

- (1) Ensemble three-way clustering framework based on dimensionality reduction.

We introduce a novel ensemble three-way clustering framework that combines dimensionality reduction techniques with clustering ensemble methods. This framework reduces data dimensions, eliminates noise, and enhances clustering stability. By leveraging multiple clustering results, the method enhances the algorithm's robustness through randomness and diversity.

- (2) Integration of co-occurrence frequency, hierarchical clustering, and lifecycle analysis:

The proposed method calculates the co-occurrence frequency of data points being in the same cluster, aiding in accurately defining similar classes. It employs a single-linkage hierarchical clustering approach to fuse clustering results and constructs a dendrogram based on these probabilities. By analyzing the lifecycle of clusters, we determine the most stable clustering result, ensuring robustness and consistency.

These contributions collectively enhance the performance and applicability of three-way clustering algorithms, especially for high-dimensional datasets, providing a more accurate and stable clustering solution.

The remainder of this paper is organized as follows. In Section 2, we provide a comprehensive review of the concepts related to three-way clustering, the k-means algorithm, PCA, and data integration strategies. Section 3 outlines the methodology and algorithmic process employed in this study. The results and performance metrics obtained from the proposed algorithm on the UCI dataset are presented in Section 4. Section 5 encompasses the discussion of our findings and identifies areas for future improvement.

2. Related Work

2.1. Three-Way Clustering

Traditional hard clustering depicts a cluster by one set with a sharp boundary. Only two relationships between the sample and cluster are considered, i.e., belonging to and not belonging to. For the samples inside the cluster, they belong to this cluster, and for the samples outside the cluster, they are not the elements of this cluster. Given a dataset $X = \{x_1, x_2, \dots, x_n\}$ with n samples and k clusters in traditional hard clustering, the clustering results can be represented as $C = \{C_1, C_2, \dots, C_k\}$, where $C_i, i = 1, 2, \dots, k$ satisfies

$$\begin{cases} C_i \neq \emptyset, i = 1, \dots, k \\ \bigcup_{i=1}^k C_i = U \\ C_i \cap C_j = \emptyset, i \neq j \end{cases}$$

In traditional clustering, each sample is unequivocally assigned to one cluster, and there are clear boundaries between different clusters. This two-way description of a cluster may not adequately show the uncertainty information in data. To address the limitation in traditional clustering, Yu [34,52] proposed three-way clustering by defining three types of membership relations between a sample and a cluster, namely, belonging to fully, belonging to partially and not belonging to. Three-way clustering utilizes the core region $Co(C_i)$ and the fringe region $Fr(C_i)$ to depict a cluster, and the universe is split by these two sets into three sections, $Co(C_i)$, $Fr(C_i)$, and $Tr(C_i) = U - Co(C_i) - Fr(C_i)$, which obey the following conditions:

$$\begin{cases} Co(C_i) \cup Tr(C_i) \cup Fr(C_i) = U \\ Co(C_i) \cap Tr(C_i) = \emptyset \\ Co(C_i) \cap Fr(C_i) = \emptyset \\ Fr(C_i) \cap Tr(C_i) = \emptyset \end{cases}$$

Three-way clustering results of dataset X are expressed as

$$C = \{(Co(C_1), Fr(C_1)), (Co(C_2), Fr(C_2)), \dots, (Co(C_k), Fr(C_k))\}.$$

2.2. PCA Dimensionality Reduction

As a powerful tool in the realm of data analysis, PCA [47] (Principal Component Analysis) offers a systematic approach to reduce the dimensionality of data while retaining the significant variance within the dataset. This not only makes data easier to visualize but also enhances the efficiency of subsequent analytical techniques. The fundamental idea of PCA involves a linear transformation that maps the original data onto a new coordinate system. The selection of this new coordinate system aims to maximize the variance of the data along specific axes. By choosing the first few principal components, the data can be projected onto these components, achieving dimensionality reduction.

In the computational process, the initial step involves calculating the covariance matrix of the original data. Subsequently, through eigenvalue decomposition, the eigenvalues and eigenvectors of the covariance matrix are obtained. Following this, a selection of the top eigenvalue-ordered eigenvectors forms the new coordinate system, representing the principal components. Finally, projecting the original data onto these principal components yields the reduced-dimensional data. Figure 1 illustrates the fundamental principle of PCA for dimensionality reduction. In Figure 1, the original distribution of the dataset is given on the plane, where the red and black dots represent different classes. Through PCA, these points are projected onto the principal component directions in the reduced-dimensional space, resulting in a new distribution of data. This process allows for the mapping of high-dimensional data into a lower-dimensional space while retaining the essential features of the original data, thus reducing dimensionality.

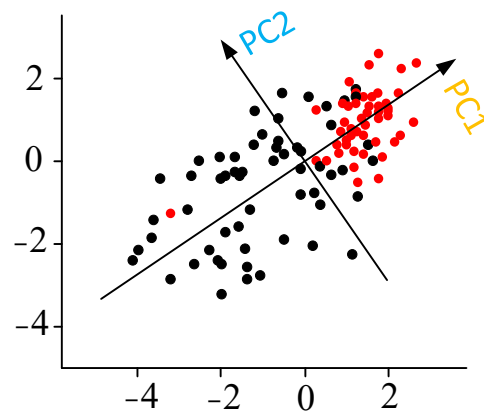


Figure 1. Illustration of Dimensionality Reduction by PCA.

The application of PCA for dimensionality reduction offers the advantage of preserving the crucial features of the data while reducing their dimensionality. This enhances computational efficiency for subsequent analyses, providing robust support for research endeavors.

2.3. K-Means Algorithm

K-means algorithm [53] is a widely used clustering method with the goal of partitioning a dataset into k clusters, such that samples in the same cluster have high similarity, and samples in distinct clusters have low similarity. The main idea of k -means algorithm involves determining the positions of cluster centers by minimizing a loss function, which incorporates the Euclidean distance between sample and cluster centers. Specifically, the algorithm initiates by randomly selecting k sample points as initial cluster centers. It iteratively performs two key steps, i.e., assigning each sample point to the closest cluster center in Euclidean distance, and updating the position of each cluster's center based on the samples assigned to it. This process repeats until the cluster centers no longer undergo significant changes, signifying convergence of the loss function. The mathematical formulation of the loss function is given by

$$J = \sum_{i=1}^n \sum_{j=1}^k w_{ij} \|\chi_i - \mu_j\|^2, \quad (1)$$

where w_{ij} is the indicator function, indicating whether the sample χ_i is assigned to cluster μ_j . By minimizing this loss function, k -means algorithm efficiently identifies optimal cluster center positions, facilitating effective data clustering.

2.4. Hierarchical Clustering

Hierarchical clustering builds a tree-like structure (dendrogram) to represent the nested grouping of data points. It can be divided into agglomerative and divisive methods [54]. Agglomerative hierarchical clustering starts with each data point as an individual cluster and iteratively merges the closest pairs of clusters until a single cluster is formed. Conversely, divisive hierarchical clustering starts with the whole dataset as a single cluster and recursively splits it into smaller clusters. A well-known variation is the single-linkage method, which defines the distance between two clusters as the minimum distance between any pair of points from the two clusters. This method is effective in identifying clusters with irregular shapes.

2.5. Clustering Ensemble and Co-Association Frequency

Although there are many clustering methods, it has been accepted that there is not one clustering method that can identify all kinds of data structure distribution. In order to solve this problem, Strehl and Ghosh [16] proposed the cluster ensemble algorithm, which combines multiple clustering results of a set of objects into one clustering result without accessing the original features of the objects. The framework of clustering ensemble can be depicted by Figure 2.

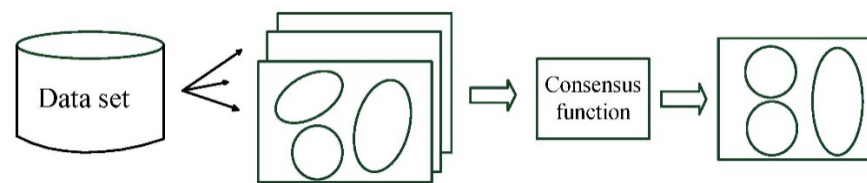


Figure 2. The framework of clustering ensemble.

The aim of clustering ensemble [55] is to consolidate multiple independent clustering results into a comprehensive outcome, aiming to overcome potential biases introduced by different clustering algorithms. Moreover, the rise of clustering ensemble has given birth to various clustering ensemble methods, such as the voting–merging approach proposed by Hornik [56]. This method leverages clustering ensemble algorithms to achieve more reliable and stable clustering results. It utilizes an unsupervised voting mechanism to amalgamate within the ensemble clustering, ultimately merging to derive the final clustering outcome. For family clustering results of a dataset, there are three types of relationships between two samples by qualitative observation. They may be always assigned to the same cluster, or they are assigned to the same cluster occasionally. The last circumstance is not assigned to the same group completely. In order to quantify a sample’s tendency of changing groups quantitatively, Li et al. [41] introduced a measurement named as co-association frequency by using the results of a family clustering.

Given a dataset $X = \{x_1, x_2, \dots, x_n\}$ with n samples and C^1, C^2, \dots, C^L are family clustering results on U , we use $C^l(x_i)$ to indicate the label of x_i induced by clustering result C^l . The co-association frequency p_{ij} , which represents that two samples x_i and x_j appear in the same cluster, is calculated by

$$p_{ij} = \frac{1}{L} \sum_{l=1}^L \prod(C^l(x_i), C^l(x_j)), \tag{2}$$

where

$$\prod(C^l(x_i), C^l(x_j)) = \begin{cases} 1 & C^l(x_i) = C^l(x_j) \\ 0 & C^l(x_i) \neq C^l(x_j) \end{cases}$$

We use an example to illustrate p_{ij} . Figure 3 is a dataset X with 6 samples and four clustering results C^1, C^2, C^3, C^4 of X . The samples x_1 and x_2 consistently remain in the same cluster across all results, indicating that co-association frequency $p_{12} = 1$. On the

other hand, x_1 and x_3 are assigned to the same cluster only in C^1 and C^2 , showing that co-association frequency $p_{13} = 0.5$. The samples x_1 and x_5 are grouped into different clusters across all four clustering results, indicating that co-association frequency $p_{15} = 0$.

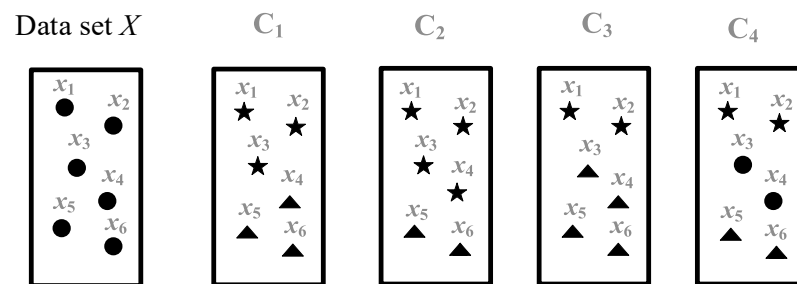


Figure 3. An example of a dataset with four clustering results.

According to the above definition, we can obtain the co-association matrix of Figure 3 as Table 1.

Table 1. The co-association frequency matrix of Figure 3.

p_{ij}	x_1	x_2	x_3	x_4	x_5	x_6
x_1	1	1	0.5	0.25	0	0
x_2	1	1	0.5	0.25	0	0
x_3	0.5	0.5	1	0.75	0.25	0.25
x_4	0.25	0.25	0.75	1	0.5	0.25
x_5	0	0	0.25	0.5	1	1
x_6	0	0	0.25	0.25	1	1

Co-association frequency [57,58] is to measure the probability that two data samples are assigned to the same cluster in multiple clustering results. Specifically, if two samples are consistently assigned to the same cluster across multiple clustering results, their co-association frequency is 1. If two samples are not assigned to the same group completely, their co-association frequency is 0. By calculating the co-occurrence probability for all data points, a co-association frequency matrix is obtained. This matrix provides information about the similarity of data points. By setting a threshold for co-association frequency, samples with frequencies above the threshold are grouped into the same similarity class. This approach integrates information from multiple clustering runs, not relying solely on a single clustering result, thereby enhancing a comprehensive understanding of the data structure.

3. Similarity-Based Three-Way Clustering by Using Dimensionality Reduction

In this section, we propose a similarity theory [43,59] based on data dimensionality reduction and similarity-based three-way clustering. In contrast to traditional algorithms, our approach first employs the PCA algorithm for data preprocessing, transforming high-dimensional data into low-dimensional data. It incorporates an ensemble strategy by randomly extracting subsets of features from the samples in multiple iterations, generating diverse basic clustering results using the traditional k-means clustering algorithm. Subsequently, we calculate the co-association frequency between samples to derive similarity classes. By extracting only partial features of the samples, we significantly reduce the computational complexity compared to the existing traditional ensemble clustering methods. The algorithm proposed in this paper involves three main steps: the generation of basic clusters by using dimensionality-reduced data, the computation of co-association frequency and similarity classes, and the integration of these results into three-way clustering.

3.1. Dimensionality Reduction

In this study, we employed data dimensionality reduction techniques, specifically utilizing Principal Component Analysis (PCA) to reduce the dimensions of the data. PCA is a commonly used dimensionality reduction method, aiming to map the original data onto a lower-dimensional subspace while retaining the maximum variance in the data. Through PCA, we can transform high-dimensional data into lower-dimensional space, thereby enhancing our understanding of the intrinsic structure of the data.

To begin with, consider a dataset comprising n samples and D features, represented by matrix X , where each row corresponds to a sample, and each column represents a feature. Our objective is to project this D -dimensional dataset onto a K -dimensional subspace (where $K < D$) and obtain a new feature matrix Z . The specific steps of dimensionality reduction by using PCA are as follows:

Step 1: Data normalization: The first step involves centralizing the original data by subtracting the mean of each feature, resulting in the centered matrix X' .

Step 2: Covariance Matrix Computation: The covariance matrix represents the correlations between data features, with the specific formula

$$\Omega = \frac{1}{N} X'^T \bullet X'. \quad (3)$$

Step 3: Eigenvalue and Eigenvector Computation: Eigenvalue decomposition is applied to the covariance matrix Ω , yielding eigenvalues $\lambda_1, \lambda_2, \dots, \lambda_D$ and their corresponding eigenvectors v_1, v_2, \dots, v_D .

$$\Omega v = \lambda_i v_i, i = 1, 2, \dots, D. \quad (4)$$

Step 4: Selection of Top K Eigenvectors: The eigenvectors corresponding to the top K largest eigenvalues are chosen, forming the projection matrix V .

Step 5: Data Projection: The centered original data matrix X' is projected onto the selected K -dimensional subspace, resulting in the reduced feature matrix Z , where each row represents a sample, and each column represents a reduced feature. The specific formula is

$$Z = X'V. \quad (5)$$

Through the aforementioned steps, we obtain the reduced-dimensional data matrix. In this low-dimensional space, we conduct fundamental clustering operations. This data-driven foundational clustering method allows for clustering analysis in lower dimensions while preserving the primary features of the data. The key advantage of this approach lies in its ability to facilitate data visualization, reduce computational complexity, and enhance clustering effectiveness through dimensionality reduction.

Next, we randomly select parts of the sample's features to obtain different clustering results. For a multidimensional dataset, different subsets of features try to describe the dataset from different views. Thus, a set of diverse clustering results will be obtained when distinguishing subsets of features are employed. Suppose that we randomly extract parts of the features and apply the k-means clustering method to divide the dataset into k clusters. This process is repeated L times, yielding multiple clustering results C^1, C^2, \dots, C^L . The process of foundational clustering based on data dimensionality reduction is outlined in Algorithm 1.

Algorithm 1: Foundational Clustering Based on Data Dimensionality Reduction

Input: Original data matrix $X = \{x_1, x_2, \dots, x_n\}$, number of iteration L

Output: Reduced-dimensional dataset Z , Corresponding clustering results C^1, C^2, \dots, C^L .

- 1 **For** $i = 1, \dots, L$ **do the following:**
- 2 Apply the dimensionality reduction method to X and obtain the reduced-dimensional dataset Z .
- 3 Randomly extract a subset of features from Z .
- 4 Use the k-means algorithm to obtain the clustering result C^i .
- 5 **End**
- 6 **Return** Multiple clustering results C^1, C^2, \dots, C^L and reduced dimensional dataset Z .

3.2. Clustering Ensemble

From multiple clustering iterations, we obtain basic clustering results C^1, C^2, \dots, C^L . Subsequently, we present a method for integrating the basic clustering results by using the co-occurrence frequency matrix. The aim is to employ the single-link method of hierarchical clustering to generate a more robust clustering result.

For a dataset $X = \{x_1, x_2, \dots, x_n\}$ with n samples and C^1, C^2, \dots, C^L are family clustering results of X , we can construct an $n \times n$ co-association frequency matrix P , whose elements p_{ij} represents the frequency that two samples x_i and x_j are simultaneously assigned to the same cluster.

We view p_{ij} as the similarity between samples and utilize the single-linkage of hierarchical clustering to obtain an ensemble clustering result. In the process of clustering, each data sample is treated as an independent cluster, and then gradually the most similar cluster is merged based on their co-association frequencies. Clusters with the highest similarity are merged to form a new cluster node. This process iterates until the cluster result with the highest lifetime is chosen as the final merged result.

The schematic representation of the single-linkage clustering dendrogram is illustrated in Figure 4. Different colors in Figure 4 represent different clusters at present, and each color represents a set of samples with high similarity. This bottom-up merging strategy ensures that we fully consider the degree of association between samples, resulting in more accurate clustering results. By measuring the similarity between different clusters and visualizing them as a dendrogram, we could intuitively observe the structure and hierarchy of the clustering results. In the dendrogram, higher connecting points represented stronger associations between clusters with higher co-occurrence frequencies. These results were relatively stable and less susceptible to noise or changes in the data. Therefore, such clustering results were more reliable and better able to reflect the true structure and patterns of the data.

By constructing a single-linkage clustering dendrogram using co-association frequencies and selecting the clustering result with the highest lifetime as the final fusion result, we obtain more stable clustering results, thereby enhancing our understanding of the features and inherent structure of the dataset. The process of ensemble clustering is outlined in Algorithm 2.

Algorithm 2: Ensemble Clustering Results	
	Input: Reduced data matrix C^1, C^2, \dots, C^L
	Output: C_1, C_2, \dots, C_L
1	Compute the co-occurrence frequency matrix P by (2).
2	Obtain the single-linkage dendrogram of P .
3	Achieve ensemble clustering results C with the highest lifetime.
4	Return ensemble clustering results $C = (C_1, C_2, \dots, C_k)$.

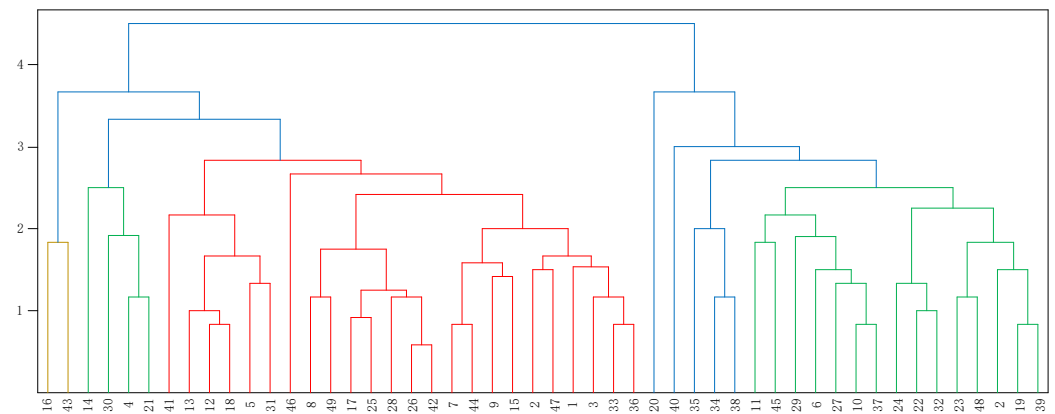


Figure 4. Single-linkage dendrogram of hierarchical clustering.

3.3. Similar Classes Based on Co-Association Frequency

This section introduces three-way clustering models based on the co-occurrence frequency derived from clustering ensemble, proposing a similarity relationship under the framework of co-association frequency. Firstly, we give the definition of similar relation between x_i and x_j .

Definition 1. For a dataset $X = \{x_1, x_2, \dots, x_n\}$ with n samples and C^1, C^2, \dots, C^L are family clustering results of X , p_{ij} is the co-association frequency between samples x_i and x_j . The similarity relation $Sim_\theta(x_i, x_j)$ based on a threshold θ is defined as:

$$Sim_\theta(x_i, x_j) = \{(x_i, x_j) \in X \times X | p_{ij} \geq \theta\}, \tag{6}$$

where $0 \leq \theta \leq 1$ is a pre-defined parameter. For $x_i \in X$, the similar class is computed by:

$$Sim_\theta(x_i) = \{x_j \in X | p_{ij} \geq \theta\}. \tag{7}$$

We still use Figure 3 as an example. If we take $\theta = 0.7$, then $Sim_\theta(x_1) = \{x_1, x_2\}$, $Sim_\theta(x_2) = \{x_1, x_2\}$, $Sim_\theta(x_3) = \{x_3, x_4\}$, $Sim_\theta(x_4) = \{x_3, x_4\}$, $Sim_\theta(x_5) = \{x_5, x_6\}$, and $Sim_\theta(x_6) = \{x_5, x_6\}$.

From the above definition, we can find that the similar class $Sim_\theta(x_i)$ has the following properties:

$$1 \ x_i \in Sim_\theta(x_i); \tag{8}$$

$$2 \ \text{if } x_j \in Sim_\theta(x_i), \text{ then } x_i \in Sim_\theta(x_j); \tag{9}$$

$$3 \ \bigcup_{i=1}^n Sim_\theta(x_i) = X. \tag{10}$$

Clearly, the set of similar classes $\{Sim_\theta(x_i) | x_i \in X\}$ forms a covering of dataset X . For any subset $C \subseteq X$, $0 \leq \theta \leq 1$, the lower and upper approximations based on the co-association frequency are defined as follows:

$$\underline{Apr}_\theta(C) = \{x_i \in X | Sim_\theta(x_i) \subseteq C\}; \tag{11}$$

$$\overline{Apr}_\theta(C) = \{x_i \in X | Sim_\theta(x_i) \cap C \neq \emptyset\}. \tag{12}$$

Furthermore, we can use the positive region $Pos_\theta(C)$ and the fringe region $Bnd_\theta(C)$ to describe the objective subset C . So, we define $Pos_\theta(C)$ and $Bnd_\theta(C)$, of C as

$$Pos_\theta(C) = \underline{Apr}_\theta(C) \tag{13}$$

$$Bnd_\theta(C) = \overline{Apr}_\theta(C) - \underline{Apr}_\theta(C) \tag{14}$$

Usually, the positive region $Pos_\theta(C)$ contains the samples that belong to C definitely, and the fringe region $Bnd_\theta(C)$ contains the samples that belong to C possibly. Based on the definitions and properties of $Pos_\theta(C)$ and $Bnd_\theta(C)$, for any cluster $C_i \subseteq X$, it is straightforward to obtain the core region $Co(C_i)$ and the fringe region $Fr(C_i)$ by

$$Co(C_i) = Pos_\theta(C_i), \tag{15}$$

$$Fr(C_i) = Bnd_\theta(C_i). \tag{16}$$

Algorithm 3 illustrates the calculation of core region $Co(C_i)$ and the fringe region $Fr(C_i)$ based on co-association frequency.

Algorithm 3: Finding core region and fringe region

Input: Co-association frequency p_{ij} , ensemble clustering results C_1, C_2, \dots, C_k ,

threshold θ ;

Output: $C = \{(Co(C_1), Fr(C_1)), (Co(C_2), Fr(C_2)), \dots, (Co(C_k), Fr(C_k))\}$

1 **For** $i = 1, \dots, L$ **do the following:**

2 | Compute $Sim_\theta(x_i) = \{x_j \in X | p_{ij} \geq \theta\}$.

3 | **End**

4 | **For** $i = 1, \dots, k$ **do**

 | $Co(C_i) = \{x_i \in X | Sim_\theta(x_i) \subseteq C_i\}$

5 | $Fr(C_i) = \{x_i \in X | Sim_\theta(x_i) \cap C_i \neq \emptyset\} - Co(C_i)$

6 | **End**

7 **Return** $C = \{(Co(C_1), Fr(C_1)), (Co(C_2), Fr(C_2)), \dots, (Co(C_k), Fr(C_k))\}$

3.4. Similarity-Based Three-Way Clustering by Using Dimensionality Reduction

The stepwise execution of Algorithms 1–3 forms the framework of the proposed similarity-based three-way clustering by using dimensionality reduction, as illustrated in Algorithm 4.

Algorithm 4: Similarity-based three-way clustering algorithm

Input: Original data matrix $X = \{x_1, x_2, \dots, x_n\}$, Number of iterations of clusters L
Dimensionality reduction method (PCA), Threshold θ

Output: $C = \{(Co(C_1), Fr(C_1)), (Co(C_2), Fr(C_2)), \dots, (Co(C_k), Fr(C_k))\}$

- 1 **Initialize:** \leftarrow Algorithm 1; **Return** C^1, C^2, \dots, C^L ;
- 2 **Ensemble:** \leftarrow Algorithm 2; **Return** C_1, C_2, \dots, C_k ;
- 3 **Identify Core and Fringe Regions:** \leftarrow Algorithm 3;
- 4 **Return** $C = \{(Co(C_1), Fr(C_1)), (Co(C_2), Fr(C_2)), \dots, (Co(C_k), Fr(C_k))\}$.

In this framework, we first generate a set of base clustering results by employing dimensionality reduction techniques (Algorithm 1). Subsequently, by calculating co-association frequencies, we utilize the single-linkage of hierarchical clustering to obtain ensemble clustering results (Algorithm 2). Finally, by defining the similar classes of each sample, we derive the core and fringe regions, further adjusting the clustering structure to yield more accurate and representative three-way clustering outcomes.

The uniqueness of this framework lies in its integration of data dimensionality reduction, co-association frequency computation, and definition of similar classes, providing a comprehensive revelation of the intrinsic structure during the clustering ensemble process. Algorithm 4 outlines the overall process of the three-way clustering framework, demonstrating how optimized clustering results are generated through multiple iterations to better reflect the characteristics of the original data.

The proposed approach offers a powerful tool for clustering ensemble, aiding in the precise capture of complex relationships and distribution patterns in clustering analysis. The three-way clustering framework provides valuable insights seeking to uncover intricate structures within their datasets.

4. Experimental Analyses

4.1. Data Descriptions

In this section, we conduct some experiments to evaluate the effectiveness of the proposed algorithm. We employ datasets from 13 UCI machine learning repositories [60], spanning diverse domains such as biology, medicine, and finance. The detailed information about these datasets is presented in Table 2, including the number of clusters and other relevant details. The software used for implementation includes MATLAB2019a for statistical and matrix computations and Python 3.9 with libraries such as NumPy, SciPy, and scikit-learn for data processing and machine learning tasks, ensuring robust and efficient analysis.

Table 2. Datasets Used in Experiments.

ID	Datasets	Numbers	Dimensions	Categories
1	Seeds	270	7	3
2	Credit	1493	9	3
3	Ionosphere	351	34	2
4	Libras	360	90	15
5	Ecoil	210	7	3
6	Segmentation	2310	19	7
7	Thyroid	215	9	3
8	Wdbc	569	30	2
9	Wine	178	13	3
10	Waveform	5000	40	3
11	Iris	150	4	3

Table 2. Cont.

ID	Datasets	Numbers	Dimensions	Categories
12	Yeast	1484	8	10
13	Dermatology	366	34	6

4.2. Evaluation Indices

- (1) Adjusted Rand Index (ARI) [61,62] serves as a prominent external metric for assessing clustering performance in comparison to ground truth labels. The ARI, an extension of the Rand Index (RI), is designed to overcome the limitations of the RI by adjusting for chance agreements.

ARI adjusts the RI using the following formula:

$$ARI = \frac{RI - E[RI]}{\max(RI) - E[RI]}, \tag{17}$$

where $E[RI]$ represents the expected Rand Index under random conditions. The Rand Index (RI) is calculated by the formula:

$$RI = \frac{a + b}{a + b + c + d}. \tag{18}$$

a : the number of sample pairs that belong to the same cluster in both the ground truth and clustering results.

b : the number of sample pairs that belong to different clusters in both the ground truth and clustering results.

c : the number of sample pairs that belong to the same cluster in the ground truth but to different clusters in the results.

d : the number of sample pairs that belong to different clusters in the ground truth but to the same cluster in the results.

ARI values provide insights into the agreement between clustering results and ground truth labels, with 1 indicating perfect agreement, 0 suggesting performance no better than random assignment, and negative values indicating worse than random allocation. The introduction of ARI offers a comprehensive and objective means for evaluating clustering algorithms, facilitating a more accurate understanding of their performance.

- (2) Adjusted Mutual Information (AMI) [63,64] is an internal metric commonly used to assess the performance of clustering results. It is designed to measure the similarity between clustering results and a ground truth (typically, actual labels) by quantifying the information gain between two distributions.

The computation of AMI involves the following formula:

$$AMI(U, V) = \frac{MI(U, V) - E[MI(U, V)]}{\max(H(U), H(V)) - E[MI(U, V)]}, \tag{19}$$

where $MI(U, V)$ represents the mutual information between U and V . $E[MI(U, V)]$ is the expected mutual information under random conditions. $H(U)$ and $H(V)$ are the entropies of U and V , respectively.

The numerator of AMI is an adjusted value of mutual information, while the denominator is an adjusted value of entropy. The values of AMI range from $[0, 1]$, where 1 indicates a perfect match, 0 denotes random matching, and negative values signify matching below random levels.

- (3) Accuracy (ACC) [65] is a common metric used to assess the performance of a classification model. It measures the proportion of samples that the model correctly

classifies and serves as a simple and intuitive performance indicator. The formula for calculating ACC is as follows:

$$ACC = \frac{TP + TN}{TP + TN + FP + FN} \quad (20)$$

where TP (True Positives) represents the number of samples correctly classified as the positive class, TN (True Negatives) represents the number of samples correctly classified as the negative class, FP (False Positives) represents the number of samples actually belonging to the negative class but misclassified as the positive class, FN (False Negatives) represents the number of samples actually belonging to the positive class but misclassified as the negative class.

The range of ACC is $[0, 1]$, where 1 indicates perfect classification and 0 indicates classification failure. While ACC is an intuitive and easy-to-understand metric, it may have limitations when dealing with class imbalance.

4.3. Experimental Performances

Firstly, the PCA dimensionality reduction method is applied to high-dimensional datasets to obtain processed low-dimensional data. Subsequently, a clustering ensemble strategy is employed for the low-dimensional data. This involves randomly sampling subsets of data and features and running the traditional k-means clustering strategy for 50 iterations on all datasets. Then, an automatic hierarchical clustering method is used to form the clustering structure, and the merged results can be visualized using a dendrogram. Finally, the upper and lower approximations of similar classes are derived, and the core and fringe regions of each cluster are determined. Additionally, similarity threshold θ is 0.7 in the experiments.

Because NMI, ARI, and ACC are only adopted to the hard clustering results, three-way clustering results cannot calculate these values directly. In order to present the performances of our proposed algorithm, this study uses the core regions to form a clustering result, then calculate the NMI, ARI, and ACC by using the core region to represent the corresponding cluster. The clustering ensemble strategy is executed 50 times on all datasets, with an ensemble size of 50, to calculate the average NMI, ARI, and ACC values. The performances of the proposed algorithm on these three indicators are displayed in Table 3 and Figures 5–7. To compare clustering effects, the performances of k-means, FCM, and DBSCAN are also presented in Table 3 and Figures 5–7. The best performances for each dataset are highlighted in bold.

Table 3. The performances of different algorithms.

Datasets	Algorithm	ARI	AMI	ACC
Seeds	K-means	0.7500	0.7054	0.9095
	FCM	0.7161	0.6915	0.8952
	DBSCAN	0.7021	0.4396	0.3667
	Ours	0.8198	0.7685	0.9356
Credit	K-means	0.0091	0.032	0.3741
	FCM	0.0272	0.0317	0.3917
	DBSCAN	0.0110	0.0006	0.389
	Ours	0.1116	0.1073	0.3987
Ionosphere	K-means	0.011	0.0006	0.5783
	FCM	0.1713	0.1272	0.7094
	DBSCAN	0.2174	0.1426	0.3932
	Ours	0.2500	0.2017	0.7833

Table 3. Cont.

Datasets	Algorithm	ARI	AMI	ACC
Libras	K-means	0.1837	0.1842	0.3389
	FCM	0.0597	0.33	0.1778
	DBSCAN	0.0025	0.2215	0.1000
	Ours	0.5193	0.7144	0.6182
Ecoil	K-means	0.4542	0.5709	0.5565
	FCM	0.3679	0.5619	0.497
	DBSCAN	0.0080	0.005	0.4256
	Ours	0.3937	0.4999	0.6100
Segmentation	K-means	0.0331	0.0736	0.2455
	FCM	0.3875	0.5062	0.6100
	DBSCAN	0.1067	0.3301	0.2939
	Ours	0.5501	0.6996	0.6720
Thyroid	K-means	0.2145	0.3911	0.5721
	FCM	0.4294	0.176	0.786
	DBSCAN	0.3123	0.0356	0.4465
	Ours	0.5964	0.5628	0.8950
Wdbc	K-means	0.0019	0.0052	0.5202
	FCM	0.7299	0.6138	0.9279
	DBSCAN	0.0274	0.0145	0.6098
	Ours	0.6441	0.5295	0.9320
Wine	K-means	0.4483	0.4485	0.6461
	FCM	0.3492	0.4075	0.6854
	DBSCAN	0.2700	0.3137	0.5169
	Ours	0.5831	0.6674	0.8118

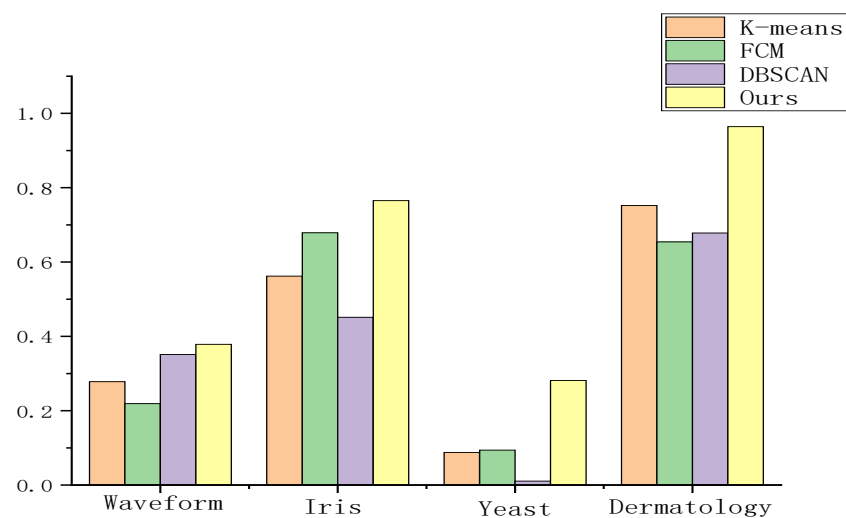


Figure 5. ARI Comparison of Algorithms Embedded in K-means.

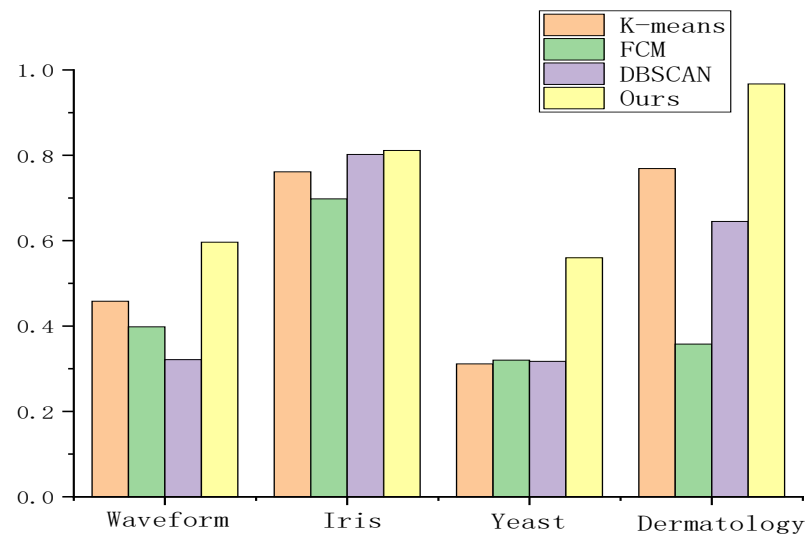


Figure 6. AMI Comparison of Algorithms Embedded in K-means.

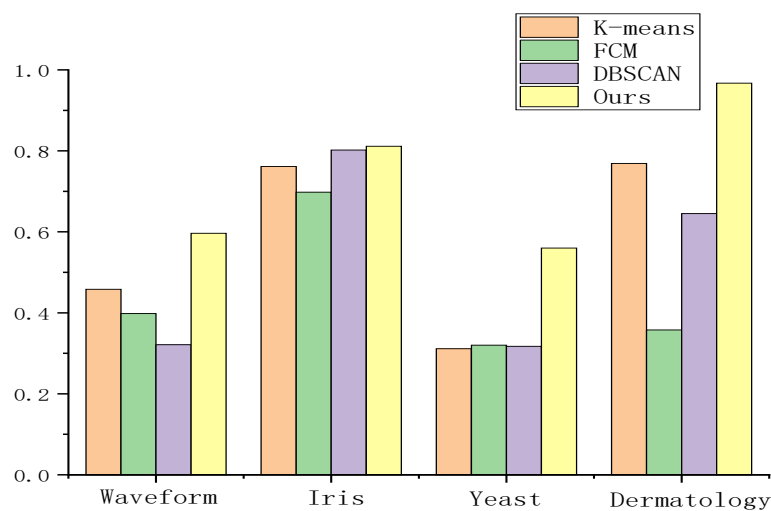


Figure 7. ACC Comparison of Algorithms Embedded in K-means.

Through a comparative analysis of the data presented in Table 3 and Figures 5–7, the following conclusions can be drawn:

- (1). By comparing the performance of our proposed three-way clustering algorithm with traditional clustering methods, such as k-means, FCM (Fuzzy C-Means), and DBSCAN (Density-Based Spatial Clustering of Applications with Noise), on AMI, ARI, and ACC, it can be found that our proposed algorithm demonstrates significant advantages on most datasets. Taking the Libras dataset as an example, after running the proposed algorithm, the resulting AMI, ARI, and ACC values are 0.5193, 0.7144, and 0.6182, respectively. In contrast, the AMI, ARI, and ACC values for the traditional k-means algorithm are only 0.1837, 0.1842, and 0.3389, respectively. This improvement is attributed to the dimensionality reduction of original high-dimensional data, mapping it to a lower-dimensional space, thus reducing data complexity. The introduction of co-occurrence probability enables more precise delineation of similar classes, allocating data points to core and fringe regions, better capturing the inherent structure of the data.
- (2). By comparing the proposed three-way clustering algorithm with other algorithms in terms of AMI, ARI, and ACC, we observed significant improvements in the proposed algorithm relative to others. Specifically, across all datasets, the proposed algorithm exhibited an average improvement of approximately 20% to 30% in ARI and ACC,

and an average increase of about 15% to 35% in AMI. There are several potential reasons behind these improvements. Firstly, the proposed three-way clustering algorithm adopts an ensemble strategy, integrating concepts of data dimensionality reduction, co-occurrence frequencies, and similarity classes, thereby offering a more comprehensive consideration of the inherent structure of the data. Secondly, leveraging the single-linkage method of hierarchical clustering, the proposed three-way clustering algorithm effectively captures the degree of correlation among data points, resulting in more precise classification of data points into clusters. Additionally, by selecting the clustering result with the highest lifetime as the final merged result, the proposed three-way algorithm ensures the stability and consistency of the clustering results, rendering it more suitable for various data types and complex structures. The suboptimal performance on the Wdbc dataset may be due to algorithm sensitivity to different parameter settings, and parameter selection may vary across different datasets. Although our proposed algorithm shows significant improvements, certain algorithms may perform better under specific conditions due to their inherent characteristics. For example, algorithms like DBSCAN are particularly effective for datasets with noise and density variations, while hierarchical clustering can capture nested cluster structures. By comparing the actual runtime with the computational time complexity, it is concluded that the proposed algorithm strikes a balance between accuracy and computational efficiency. Although it is not the fastest, its robustness and ability to handle high-dimensional and noisy data make it a valuable tool in practical applications.

In summary, the proposed three-way clustering algorithm amalgamates ideas from data dimensionality reduction, co-occurrence frequency calculation, and similar class partitioning. Compared to traditional clustering algorithms, it demonstrates advantages in more nuanced data analysis and accurate clustering results, making it more feasible and effective in practical applications.

5. Conclusions

The theoretical contribution of this paper lies in the proposal of a novel three-way clustering framework that integrates dimensionality reduction, co-occurrence frequencies, and similarity classes with three-way clustering. The objective is to efficiently cluster heterogeneous data from multiple sources by leveraging inherent structural information. Initially, we employ principal component analysis (PCA) to reduce the dimensionality of the data, mapping high-dimensional data into a lower-dimensional space. This not only decreases computational complexity but also enhances clustering efficiency.

Subsequently, we introduce the concept of co-occurrence frequencies, considering the co-occurrence relationships between samples. By applying a threshold to the co-occurrence probability, samples are classified into similar classes, combined with the division into core and fringe regions. This ensures that the proposed algorithm not only accurately describes the intrinsic structure of the data but also exhibits robustness. The experimental results show that the proposed algorithm can improve clustering accuracy, particularly when dealing with complex data structures and significant noise interference. To further enhance the clustering process, we integrate these co-occurrence probabilities with a single-linkage hierarchical clustering method. This fusion enables us to construct a dendrogram that captures the similarity between different clusters. Lifecycle analysis is then employed to select the most stable clustering result, ensuring consistency and robustness.

The practical contribution of this paper is the improvement in clustering accuracy. Experimental results demonstrate that the proposed algorithm significantly enhances clustering precision, especially when handling complex data structures and substantial noise interference. This proves its practical effectiveness in various real-world scenarios. The method shows significant advantages across multiple datasets, highlighting its versatility and robustness in dealing with diverse and high-dimensional data. This adaptability makes it suitable for a wide range of applications, from bioinformatics to market segmentation.

Although the algorithm demonstrates significant advantages across multiple datasets during experimental validation, it does not consistently exhibit the expected improvements on certain specific datasets. This discrepancy may arise due to a partial mismatch between data characteristics and algorithm design, necessitating further exploration and refinement.

In future research, we will focus on the following aspects:

(1). Adaptability of parameter selection:

The subjective nature of parameter thresholds in the algorithm may impact the stability of experimental results. To enhance algorithm robustness, considering more objective and adaptive parameter selection methods to accommodate different dataset requirements and application scenarios is essential.

(2). Improving the Quality of Base Clustering:

The generation of base clustering using different feature subsets may lead to poor-quality results, negatively affecting the final ensemble clustering outcome. To enhance the quality of base clustering, we can employ automatic evaluation mechanisms based on the data's intrinsic structure or utilize advanced clustering performance metrics. Additionally, introducing other methods such as setting evaluation functions will help eliminate the impact of low-quality base clustering, effectively improving the overall performance of ensemble clustering.

(3). Adaptation Improvements for Specific Datasets:

The observation that the algorithm did not consistently exhibit expected improvements on specific datasets suggests a potential mismatch between data characteristics and algorithm design. Further work can include adapting the algorithm specifically for certain datasets, enhancing its generality and adaptability.

Author Contributions: Conceptualization, Y.M.; Data curation, A.L.; Formal analysis, A.L.; Funding acquisition, P.W.; Methodology, P.W.; Supervision, Y.M. All authors have read and agreed to the published version of the manuscript.

Funding: This work is supported by the National Natural Science Foundation of China (nos. 62076111, 61906078) and the Key Laboratory of Oceanographic Big Data Mining and Application of Zhejiang Province (no. OBDMA202002).

Data Availability Statement: The data used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Yang, X.; Yao, Y. Ensemble selector for attribute reduction. *Appl. Soft Comput.* **2018**, *70*, 1–11. [[CrossRef](#)]
2. Jiang, Z.; Yang, X.; Yu, H.; Liu, D.; Wang, P.; Qian, Y. Accelerator for multi-granularity attribute reduction. *Knowl.-Based Syst.* **2019**, *177*, 145–158. [[CrossRef](#)]
3. Li, J.; Yang, X.; Song, X.; Li, J.; Wang, P.; Yu, D.J. Neighborhood attribute reduction: A multi-criterion approach. *Int. J. Mach. Learn. Cybern.* **2019**, *10*, 731–742. [[CrossRef](#)]
4. Liu, K.; Yang, X.; Yu, H.; Fujita, H.; Chen, X.; Liu, D. Supervised information granulation strategy for attribute reduction. *Int. J. Mach. Learn. Cybern.* **2020**, *11*, 2149–2163. [[CrossRef](#)]
5. Xu, S.; Yang, X.; Yu, H.; Yu, D.-J.; Yang, J.; Tsang, E.C. Multi-label learning with label-specific feature reduction. *Knowl.-Based Syst.* **2016**, *104*, 52–61. [[CrossRef](#)]
6. Liu, K.; Yang, X.; Fujita, H. An efficient selector for multi-granularity attribute reduction. *Inf. Sci.* **2019**, *505*, 457–472. [[CrossRef](#)]
7. Liu, K.; Yang, X.; Yu, H.; Mi, J.; Wang, P.; Chen, X. Rough set based semi-supervised feature selection via ensemble selector. *Knowl.-Based Syst.* **2020**, *165*, 282–296. [[CrossRef](#)]
8. Xu, M.; Li, C.; Zhang, S.; Le Callet, P. State-of-the-art in 360 video/image processing: Perception, assessment and compression. *IEEE J. Sel. Top. Signal Process.* **2020**, *14*, 5–26. [[CrossRef](#)]
9. Tov, O.; Alaluf, Y.; Nitzan, Y.; Patashnik, O.; Cohen-Or, D. Designing an encoder for StyleGAN image manipulation. *ACM Trans. Graph.* **2021**, *40*, 1–14. [[CrossRef](#)]
10. Yang, X.; Qi, Y.; Song, X.; Yang, J. Test cost sensitive multigranulation rough set: Model and minimal cost selection. *Inf. Sci.* **2013**, *250*, 184–199. [[CrossRef](#)]

11. Xu, W.; Guo, Y. Generalized multigranulation double-quantitative decision-theoretic rough set. *Knowl.-Based Syst.* **2016**, *105*, 190–205. [[CrossRef](#)]
12. Li, W.; Xu, W.; Zhang, X.; Zhang, J. Updating approximations with dynamic objects based on local multigranulation rough sets in ordered information systems. *Artif. Intell. Rev.* **2021**, *55*, 1821–1855. [[CrossRef](#)]
13. Zhang, S.; Tong, H.; Xu, J.; Maciejewski, R. Graph convolutional networks: A comprehensive review. *Comput. Soc. Netw.* **2019**, *6*, 11. [[CrossRef](#)]
14. Zhou, J.; Cui, G.; Hu, S.; Zhang, Z.; Yang, C.; Liu, Z.; Wang, L.; Li, C.; Sun, M. Graph neural networks: A review of methods and applications. *AI Open* **2020**, *1*, 57–81. [[CrossRef](#)]
15. Coley, C.W.; Jin, W.; Rogers, L.; Jamison, T.F.; Jaakkola, T.S.; Green, W.H.; Barzilay, R.; Jensen, K.F. A graph-convolutional neural network model for the prediction of chemical reactivity. *Chem. Sci.* **2019**, *10*, 370–377. [[CrossRef](#)] [[PubMed](#)]
16. Strehl, A.; Ghosh, J. Cluster ensembles—a knowledge reuse framework for combining multiple partitions. *J. Mach. Learn. Res.* **2003**, *3*, 583–617.
17. Fred, A.L.N.; Jain, A.K. Combining multiple clusterings using evidence accumulation. *IEEE Trans. Pattern Anal. Mach. Intell.* **2005**, *27*, 835–850. [[CrossRef](#)] [[PubMed](#)]
18. Zhou, Z.; Tang, W. Cluster ensemble. *Knowl.-Based Syst.* **2006**, *19*, 77–83. [[CrossRef](#)]
19. Huang, D.; Lai, J.; Wang, C. Ensemble clustering using factor graph. *Pattern Recognit.* **2016**, *50*, 131–142. [[CrossRef](#)]
20. Huang, D.; Wang, C.; Lai, J. Locally weighted ensemble clustering. *IEEE Trans. Cybern.* **2018**, *48*, 1460–1473. [[CrossRef](#)]
21. Xu, L.; Ding, S. A novel clustering ensemble model based on granular computing. *Appl. Intell.* **2021**, *51*, 5474–5488. [[CrossRef](#)]
22. Zhou, P.; Wang, X.; Du, L.; Li, X.J. Clustering ensemble via structured hypergraph learning. *Inf. Fusion* **2022**, *78*, 171–178. [[CrossRef](#)]
23. Yao, Y. The superiority of three-way decisions in probabilistic rough set models. *Inf. Sci.* **2011**, *81*, 1080–1096. [[CrossRef](#)]
24. Yao, Y. Three-way decisions and cognitive computing. *Cogn. Comput.* **2016**, *8*, 543–554. [[CrossRef](#)]
25. Yao, Y. Tri-level thinking: Models of three-way decision. *Int. J. Mach. Learn. Cybern.* **2020**, *11*, 947–959. [[CrossRef](#)]
26. Yao, Y. The geometry of three-way decision. *Appl. Intell.* **2021**, *51*, 6298–6325. [[CrossRef](#)]
27. Pawlak, Z. Rough sets. *Int. J. Comput. Inf. Sci.* **1982**, *11*, 314–356. [[CrossRef](#)]
28. Yang, X.; Liang, S.; Yu, H.; Gao, S.; Qian, Y. Pseudo-label neighborhood rough set: Measures and attribute reductions. *Int. J. Approx. Reason.* **2019**, *105*, 112–129. [[CrossRef](#)]
29. Dou, H.; Yang, X.; Song, X.; Yu, H.; Wu, W.Z.; Yang, J. Decision-theoretic rough set: A multi-cost strategy. *Knowl.-Based Syst.* **2016**, *91*, 71–83. [[CrossRef](#)]
30. Darwiche, A. Bayesian networks. *Found. Artif. Intell.* **2008**, *3*, 467–509.
31. Daly, R.; Shen, Q.; Aitken, S. Learning Bayesian networks: Approaches and issues. *Knowl. Eng. Rev.* **2011**, *26*, 99–157. [[CrossRef](#)]
32. Yang, X.; Liu, D.; Yang, X.; Liu, K.; Li, T. Incremental fuzzy probability decision-theoretic approaches to dynamic three-way approximations. *Inf. Sci.* **2021**, *550*, 71–90. [[CrossRef](#)]
33. Li, C.; Zhou, J.; Kou, P.; Xiao, J. A novel chaotic particle swarm optimization based fuzzy clustering algorithm. *Neurocomputing* **2012**, *83*, 98–109. [[CrossRef](#)]
34. Yu, H. A framework of three-way cluster analysis. In Proceedings of the International Joint Conference on Rough Sets, Olsztyn, Poland, 3–7 July 2017; pp. 300–312.
35. Wu, T.; Fan, J.; Wang, P. An improved three-way clustering based on ensemble strategy. *Mathematics* **2022**, *10*, 1457. [[CrossRef](#)]
36. Wang, P.; Yang, X. Three-way clustering method based on stability theory. *IEEE Access* **2021**, *9*, 33944–33953. [[CrossRef](#)]
37. Wang, P.; Shi, H.; Yang, X.; Mi, J. Three-way k-means: Integrating k-means and three-way decision. *Int. J. Mach. Learn. Cybern.* **2019**, *10*, 2767–2777. [[CrossRef](#)]
38. Fan, J.; Wang, P.; Jiang, C.; Yang, X.; Song, J. Ensemble learning using three-way density-sensitive spectral clustering. *Int. J. Approx. Reason.* **2022**, *149*, 70–84. [[CrossRef](#)]
39. Wang, P.; Yang, X.; Ding, W.; Zhan, J.; Yao, Y. Three-way clustering: Foundations, survey and challenges. *Appl. Soft Comput.* **2024**, *151*, 111131. [[CrossRef](#)]
40. Wang, P.; Yao, Y. CE3: A three-way clustering method based on mathematical morphology. *Knowl.-Based Syst.* **2018**, *155*, 54–65. [[CrossRef](#)]
41. Li, F.; Qian, Y.; Wang, J.; Dang, C.; Jing, L. Clustering ensemble based on sample's stability. *Artif. Intell.* **2019**, *273*, 37–55. [[CrossRef](#)]
42. Yu, H.; Chang, Z.; Wang, G.; Chen, X. An efficient three-way clustering algorithm based on gravitational search. *Int. J. Mach. Learn. Cyber.* **2020**, *11*, 1003–1016. [[CrossRef](#)]
43. Jia, X.; Rao, Y.; Li, W.; Yang, S.; Yu, H. An automatic three-way clustering method based on sample similarity. *Int. J. Mach. Learn. Cybern.* **2021**, *12*, 1545–1556. [[CrossRef](#)]
44. Wang, P.; Wu, T.; Yao, Y. A three-way adaptive density peak clustering (3W-ADPC) method. *Appl. Intell.* **2023**, *53*, 23966–23982. [[CrossRef](#)]
45. Vittoria, B.; Lucia, M.C.; Domenico, V. A short review on minimum description length: An application to dimension reduction in PCA. *Entropy* **2022**, *24*, 269. [[CrossRef](#)] [[PubMed](#)]
46. Goparaju, B.; Rao, S.B. A DDoS attack detection using PCA dimensionality reduction and support vector machine. *Int. J. Commun. Netw. Inf. Sec.* **2022**, *14*, 1–8. [[CrossRef](#)]

47. Boushaba, A.; Cauet, S.; Chamroo, A.; Etien, E.; Rambault, L. Comparative study between physics-informed CNN and PCA in induction motor broken bars MCSA Detection. *Sensors* **2022**, *22*, 9494. [[CrossRef](#)] [[PubMed](#)]
48. Geophys, J. A tutorial on spectral clustering. *Stat. Comput.* **2007**, *17*, 395–416.
49. Ng, A.; Jordan, M.; Weiss, Y. On spectral clustering: Analysis and an algorithm. *Adv. Neural Inf. Process. Syst.* **2001**, *14*, 849–856.
50. Ford, J.K.; MacCallum, R.C.; Tait, M. The application of exploratory factor analysis in applied psychology: A critical review and analysis. *Pers. Psychol.* **1986**, *39*, 291–314. [[CrossRef](#)]
51. Torgerson, W.S. Multidimensional scaling: I. Theory and method. *Psychometrika* **1952**, *17*, 401–419. [[CrossRef](#)]
52. Yu, H.; Jiao, P.; Yao, Y.Y.; Wang, G.Y. Detecting and refining overlapping regions in complex networks with three-way decisions. *Inf. Sci.* **2016**, *373*, 21–41. [[CrossRef](#)]
53. MacQueen, J. Some methods for classification and analysis of multivariate observations. *Berkeley Symp. Math. Stat. Probab.* **1967**, *5*, 281–297.
54. Unrau, R.C.; Krieger, O.; Gamsa, B.; Stumm, M. Hierarchical clustering: A structure for scalable multiprocessor operating system design. *J. Supercomput.* **1995**, *9*, 105–134. [[CrossRef](#)]
55. Shi, N.; Liu, X.; Guan, Y. Research on k-means clustering algorithm: An improved k-means clustering algorithm. In Proceedings of the 2010 Third International Symposium on Intelligent Information Technology and Security Informatics, Jian, China, 2–4 April 2010; IEEE: Piscataway, NJ, USA, 2010; pp. 63–67.
56. Dimitriadou, E.; Weingessel, A.; Hornik, K. Voting-merging: An ensemble method for clustering. In Proceedings of the 2010 International Conference on Artificial Neural Networks, Vienna, Austria, 21–25 August 2010; pp. 217–224.
57. Fan, J.; Wang, X.; Wu, T.; Zhu, J.; Wang, P. Three-way ensemble clustering based on sample's perturbation theory. *Mathematics* **2022**, *10*, 2598. [[CrossRef](#)]
58. Wang, Y.; Liu, J.; Huang, Y.; Feng, X. Using hashtag graph-based topic model to connect semantically-related words without co-occurrence in microblogs. *IEEE Tran. Knowl. Data Eng.* **2016**, *28*, 1919–1933. [[CrossRef](#)]
59. Abdalrada, A.S.; Abawajy, J.; Al-Quraishi, T.; Islam, S.M.S. Machine learning models for prediction of co-occurrence of diabetes and cardiovascular diseases: A retrospective cohort study. *J. Diab. Meta. Disord.* **2022**, *21*, 251–261. [[CrossRef](#)] [[PubMed](#)]
60. Bache, K.; Lichman, M. *UCI Machine Learning Repository*; University of California: Irvine, CA, USA, 2013.
61. Shi, H.; Wang, P.; Yang, X.; Yu, H. An improved mean imputation clustering algorithm for incomplete data. *Neural Process. Lett.* **2021**, *54*, 3537–3550. [[CrossRef](#)]
62. Jiang, W.; Xu, X.; Wen, Z.; Wei, L. Applying the similarity theory to model dust dispersion during coal-mine tunneling. *Process Saf. Environ.* **2021**, *148*, 415–427. [[CrossRef](#)]
63. Hoffman, M.; Steinley, D.; Brusco, M.J. A note on using the adjusted rand index for link prediction in networks. *Soc. Netw.* **2015**, *42*, 72–79. [[CrossRef](#)]
64. Steinley, D.; Brusco, J.M. A note on the expected value of the Rand index. *Brit. J. Math. Stat. Psychol.* **2018**, *71*, 287–299. [[CrossRef](#)]
65. Amodio, S.; D'Ambrosio, A.; Iorio, C.; Siciliano, R. Adjusted concordance index: An extension of the adjusted rand index to fuzzy partitions. *J. Classif.* **2021**, *38*, 112–128.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.