Three-Way Ensemble Clustering for Incomplete Data

PINGXIN WANG\(^1,2\) AND XIANGJIAN CHEN\(^3\)

\(^1\)School of Science, Jiangsu University of Science and Technology, Zhenjiang 212003, China  
\(^2\)College of Mathematics and Information Science, Hebei Normal University, Shijiazhuang 050024, China  
\(^3\)School of Computer Science, Jiangsu University of Science and Technology, Zhenjiang 212003, China

Corresponding author: Pingxin Wang (pingxin_wang@hotmail.com)

This work was supported in part by the National Natural Science Foundation of China under Grant 61503160 and Grant 61572242, and in part by the Natural Science Foundation of the Jiangsu Higher Education Institutions of China under Grant 15KJB110004.

\section*{ABSTRACT}
There are many incomplete data sets in all fields of scientific studies due to random noise, data lost, limitations of data acquisition, data misunderstanding etc. Most of the clustering algorithms cannot be used for incomplete data sets directly because objects with missing values need to be preprocessed. In this paper, we present a new imputation algorithm for incomplete data and a three-way ensemble clustering algorithm based on the imputation result. In the proposed imputation algorithm, the objects with nonmissing values are firstly clustered by using hard clustering methods. For each missing objects, the mean attribute's value of each cluster are used to fill the missing attribute's value, respectively. Perturbation analysis of cluster centroid is applied to search the optimal imputation. As an application of proposed imputation method, we develop a three-way ensemble clustering algorithm by using the ideas of clustering ensemble and three-way decision. The objects with the same cluster label in different clustering results are assigned the core region of corresponding cluster while the objects with different clustering labels are assigned to the fringe region. Therefore, a three-way clustering is naturally formed. The experimental results on UCI data sets can verify that the algorithm is effective in revealing cluster structures.

\section*{INDEX TERMS} Three-way decision, three-way clustering, incomplete data, ensemble clustering.

\section{I. INTRODUCTION}
As a new idea of artificial intelligence in recent years, granular computing [1]–[3] is relatively modern theory in simulating human being's thinking and problem solving. One of the fundamental problem of granular computing is information granulation [4]–[6]. Many strategies of information granulation [7]–[9] have been proposed to meet different user's demands, in which clustering analysis is a widely used one [10]. As an important field in data mining, clustering [11] tries to find the internal underlying structure of data set. Generally, it means that similar objects are assigned to the same cluster while dissimilar objects are assigned to different clusters.

Though there are a large variety of clustering algorithms, most of the existing algorithms like k-means [12] are only adapted to complete datasets. However, in the actual scenario, some data values are missing due to random noise, data lost, limitations of data acquisition, data misunderstanding etc. The object with missing values is generally referred to as incomplete data. Due to the existences of missing values, traditional clustering algorithms cannot be directly applied to incomplete datasets. Therefore, how to handle incomplete data set is a problem to be solved in cluster analysis. In order to solve the problem of incomplete data clustering, Hathaway and Bezdek [13] put forward four incomplete data clustering methods based on fuzzy k-means clustering algorithm (FCM), which are Whole Data Strategy (WDS), Partial Distance Strategy (PDS), Optimal Completion Strategy (OCS) and Nearest Prototype Strategy (NPS), respectively. On the basis of FCM, Li \textit{et al}. [14] studied a robust FCM clustering algorithm to deal with incomplete data. Shi \textit{et al}. [15] introduced a clustering ensemble algorithm for mixed data. Yu \textit{et al}. [16] proposed a three-way decision clustering algorithm for incomplete data. In this paper, we use ensemble clustering to present a new three-way clustering algorithm for incomplete datasets.

As for incomplete datasets, the single clustering algorithm cannot achieve a good clustering result because of a large number of missing data. In order to overcome this problem,
we use ensemble clustering technique to combine multiple clustering results into a probably better one in this paper. The idea of ensemble clustering [17] is to integrate multiple clustering results into an alternative approach for improving the quality of the results. In fact, as we know, different clustering algorithm may lead to different clustering results. It is difficult to judge which clustering result is better than others due to the lack of supervised information. Therefore, the ensemble clustering technique emerged as a powerful tool for data clustering. Compared with a single clustering algorithm, ensemble clustering can significantly improve the robustness, stability and quality of clustering results. In recent years, ensemble clustering algorithm has been receiving increasing attention and many ensemble clustering algorithms have been developed [18]–[24].

The concept of three-way decision was outlined by Yao [25]. The main idea of three-way decision is to divide the universe into the positive, negative and boundary regions, denoting the regions of acceptance, rejection and non-commitment for ternary classifications. Since the introduction of three-way decision, many developments and applications [26]–[32] are proposed in different fields and disciplines. Inspired by the theory of three-way decision, Yu proposed the concept of three-way clustering [33]. In three-way clustering, a clustering is no longer represented by a single set with clear boundaries, but by three regions called core region, fringe region and trivial region, namely, which reflects the three types of relationship between an object and a cluster, namely, belong-to definitely, uncertain and not belong-to definitely. For an incomplete dataset, it is hard to assign the elements with missing attribute values to one cluster because of insufficient information. Three-way clustering can be used to solve this problem.

Based on the above discussion, we present a three-way ensemble clustering algorithm for incomplete data in this paper. We firstly propose an imputation incomplete data clustering algorithm based on hard clustering algorithm. In the proposed algorithm, we cluster the objects with non-missing values and use the mean attribute’s value of each cluster to fill the missing attribute’s value, respectively. Perturbation analysis of cluster centroid is applied to search the optimal imputation. As an application of proposed imputation method, we develop a three-way ensemble clustering algorithm based on the ideas of clustering ensemble and three-way decision. The objects with the same cluster label in different clustering results are assigned the core region of corresponding cluster while the objects with different clustering labels are assigned to the fringe region. Therefore, a three-way clustering is naturally formed.

The rests of this paper are organized as follows. Section 2 reviews incomplete information system, ensemble clustering, three-way decision. Section 3 introduces the proposed three-way ensemble clustering algorithm for incomplete data. Section 4 reviews two clustering performance measurements. Experiment results are presented in Section 5 and concluding remarks are reported in Section 6.

### II. PRELIMINARIES

In this section, we mainly review some related contents: incomplete information system, three-way decision and three-way clustering, ensemble clustering.

#### A. AN INCOMPLETE INFORMATION SYSTEM

The information system is also known as the knowledge representation system. It can be represented as $S = \{U, A, V, f\}$ or it can be abbreviated as $S = \{U, A\}$. $U = \{x_1, x_2, \ldots, x_n\}$ is a finite nonempty collection of samples, referred to as universe, where $n$ denotes the number of samples in the universe. $A = \{a_1, a_2, \ldots, a_m\}$ is a finite set of nonempty features, where $m$ represents the number of object features. $V = \{V_1, V_2, \ldots, V_h\}$ is the set of object feature values, $V_i$ is the possible feature values of $a_k$. $f$ is an information function, $f : V_{ik} = f(x_i, a_k) \in V_{ik}$. $V_{ik}$ represents the value of the sample $x_i$ on the feature $a_k$. For example, $x_i$ is the $i$-th object with $m$ features, namely, $x_i = \{x_{i1}, x_{i2}, \ldots, x_{im}\}$, where $x_{il}(i \leq n, l \leq m)$ represents the value of $l$-th feature of sample $x_i$.

When some attribute values are missing, the information system $S$ is called as incomplete information system. Table 1 is an example of an incomplete information system, which includes 8 samples and each sample has 6 attributes. The missing values are represented by $\ast$.

#### B. THREE-WAY DECISION AND THREE-WAY CLUSTERING

Rough set theory [34] uses three pair-wise disjoint sets to approximate a set, which are the positive, boundary, and negative regions, respectively. In order to interpret the three types of decision rules, Yao [35] introduced the Bayes risk decision-making into rough sets and proposed the decision-theoretic rough set model, then proposed the theory of three-way decision [25]. Three-way decision gives the definite decision to the object with definite information and to give the delayed decision to the object with insufficient information, which extends binary decision in order to overcome some drawbacks of binary-decisions. Three-way decision investigates scientifically a common problem-solving and information-processing practice. The idea of three-way decision is consistent with human’s cognitions to solve the problem in the

| Attributes | $a_1$ | $a_2$ | $a_3$ | $a_4$ | $a_5$ | $a_6$ |
|------------|------|------|------|------|------|------|
| $x_1$      | 3    | 2    | 5    | 9    | $\ast$| 1    |
| $x_2$      | 4    | 3    | $\ast$| 7    | $\ast$| 2    |
| $x_3$      | $\ast$| $\ast$| 8    | 11   | 13   | $\ast$|
| $x_4$      | 6    | 3    | 4    | 7    | 5    | 9    |
| $x_5$      | 12   | $\ast$| $\ast$| 11   | 1    | $\ast$|
| $x_6$      | 14   | 3    | $\ast$| $\ast$| $\ast$| 2    |
| $x_7$      | $\ast$| 3    | $\ast$| 7    | $\ast$| 4    |
| $x_8$      | 11   | 16   | 5    | 8    | 14   | 3    |
real world [36]. Many soft computing models for learning uncertain concepts, such as interval sets, rough sets, fuzzy sets and shadowed sets, have the tri-partitioning properties and can be reinvestigated within the framework of three-way decision [25]. Recently, Yao [36] used a TAO model to describe the research on three-way decision, which can be depicted by Fig. 1. Since the introduction of three-way decision, many developments and applications are proposed in different fields and disciplines, such as information filtering [37], text classification [38], risk decision [26], government decision [39], formal concept analysis [29]–[32], [40], sequential three-way decision [41], [42], etc.

Inspired by the idea of three-way decision, Yu [33] and Yu et al. [43], [44] proposed a framework of three-way clustering analysis. In the representation of two-way clustering, a cluster is represented by a single set, where the objects in the set belong to this cluster definitely and the objects not in the set do not belong to this cluster definitely. In contrast to the general crisp representation of a cluster, three-way clustering represents a three-way cluster $C_i$ as

$$C_i = (Co(C_i), Fr(C_i)).$$

The representation of three-way clustering by using a pair of sets can chapter the three types of relationships between an element and a cluster, namely, belong-to definitely, not belong to definitely, and uncertain. It is more felicitous than the use of a crisp set.

Here, we summarize some basic concepts of three-way clustering. Assume that

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

is a family three-way clusters of universe $U = \{x_1, x_2, \ldots, x_n\}$, where $Co(C_i) \subset U$, and $Fr(C_i) \subset U$. Let $Tr(C_i) = U - (Co(C_i) \cup Fr(C_i))$. The three sets, $Co(C_i)$, $Fr(C_i)$ and $Tr(C_i)$ naturally form the Core Region, Fringe Region, and Trivial Region, respectively, of a cluster. That is:

Core Region ($C_i) = Co(C_i)$,

Fringe Region ($C_i) = Fr(C_i)$,

Trivial Region ($C_i) = Tr(C_i)$.

These subsets have the following properties

$$Tr(C_i) \cup Co(C_i) \cup Fr(C_i) = U,$$

$$Co(C_i) \cap Fr(C_i) = \emptyset,$$

$$Co(C_i) \cap Tr(C_i) = \emptyset,$$

$$Fr(C_i) \cap Tr(C_i) = \emptyset.$$

If $Fr(C_i) = \emptyset$, the representation of $C_i$ turns into $C_i = Co(C_i)$ and $Tr(C_i) = U - Co(C_i)$. This is a representation of two-way clustering. This means two-way clustering is a special case of three-way cluster in which the fringe regions are the empty set.

Under the different applications, there are different requirements on $Co(C_i)$ and $Fr(C_i)$. In this paper, we adopt the following properties:

1) $Co(C_i) \neq \emptyset$,

2) $\bigcup_{i=1}^{k} (Co(C_i) \cup Fr(C_i)) = U$,

3) $Co(C_i) \cap Co(C_j) = \emptyset, i \neq j$.

Property (1) demands that each cluster cannot be empty. Property (2) states that it is possible that an element belongs to more than one cluster. Property (3) requires that the core regions of clusters are pairwise disjoint.

Since the proposal of three-way clustering, some three-way clustering approaches are developed. For example, Yu et al. [44] proposed a three-way clustering method to solve the problem of incremental overlapping clustering; Wang et al. [45] proposed a three-way k-means method by integrating k-means and three-way decisions; Wang and Yao [46] proposed a three-way clustering method based on mathematical morphology. Afridi et al. [47] presented a three-way clustering approach for handling missing data by using game-theoretic rough set model. All the above results enrich the theories and models of three-way clustering.

C. ENSEMBLE CLUSTERING

There are many clustering algorithms in cluster analysis, and each algorithm has its own method to discover the underlying data structure. However, different algorithm may lead to different clustering results. It has been accepted that a single clustering algorithm cannot handle all types of data distribution effectively. Due to a lack of any prior knowledge about cluster shapes, choosing a specialized clustering method is not easy. Therefore, some researchers are focus on integrating multiple clustering results, which is named as ensemble clustering.

Cluster ensemble methods attempt to find a better, more robust clustering solution by fusing as much information as possible among several distinct executions of a partitioning algorithm over the data. Compared with a single clustering algorithm, the clustering ensemble can significantly improve the robustness, stability, and quality of a clustering.
The ensemble clustering problem was first proposed by Strehl and Ghosh in [17], which is described as combining multiple clustering results of a set of objects without accessing the original features. Since the introduction of ensemble clustering, many developments are proposed in different fields and disciplines [26]–[32]. In general, the research on ensemble clustering mainly includes three aspects: ensemble generation, ensemble selection and ensemble integration. Ensemble generation is the first step in clustering ensemble algorithms in this step the set of clusterings that will be combined is generated. Appropriate generation process is very important because clustering result will be conditioned by the initial clusterings obtained in this step. Usually, there are no constraints about how the partitions must be obtained. Therefore, we can use different clustering algorithms or the same algorithm with different parameters initialization in the generation process. Next, we use k-means ensemble clustering as an example to show the process of ensemble clustering.

Suppose a data set $U = \{x_1, x_2, \ldots, x_n\}$ with $n$ objects, after clustering with different clustering algorithms or different parameters or initialization values, we get clustering result $X_1, X_2, \ldots, X_t$, where $t$ is the ensemble sizes, which indicates the number of clusterings.

K-means algorithm was introduced by Macqueen [12], which is one of the most widely employed clustering algorithms for its efficiency and performance. The computational complexity of this algorithm is lower than the complexity of most other clustering algorithms. Detailed steps of k-means can be found in Algorithm 1.

**Algorithm 1 k-Means Clustering Algorithm**

**Input:** Dataset: $U = \{x_1, x_2, \ldots, x_n\} \subset R^m$, number of clusters: $k$;

**Output:** Clustering result: $C = \{C_1, C_2, \ldots, C_k\}$.

1. randomly select $k$ objects from the dataset, where $k < n$,
2. assign each of object to one of $k$ centers, according to the shortest distance principle. i.e., $C_i = \{x_j | d(x_j, z_i) \leq d(x_j, z_p), (p \neq i), j = 1, 2, \ldots, n\}$.
3. when all of objects are assigned, recalculate the value of centers according to the formula: $z_i = (\sum_{x_j \in C_i} x_j) / |C_i| (i = 1, 2, \ldots, k)$.
4. repeat Step 2 and Step 3 until the centers no longer change or satisfy some stop conditions.
5. return $C = \{C_1, C_2, \ldots, C_k\}$.

K-means algorithm has the advantages of simplicity, efficiency, keeps scalability and high efficiency when dealing with large data sets. However, it has some inevitable shortcomings like: (1) the value of $k$ has to be given in advance; (2) it is sensitive to the initial value, different initial centers lead to different clustering results; (3) it is easy to fall into the local optimal rather than the global optimal result. Therefore, we perform ensemble clustering by randomly initializing the centers value to obtain relatively stable clustering result.

In this paper, we use the clustering algorithm of k-means to obtain a set of base clustering results $X_1, X_2, \ldots, X_t$, ($t$ is the ensemble sizes) by generating different initial centers. The procedure can be described by Fig. 2.

**III. THREE-WAY ENSEMBLE CLUSTERING ALGORITHM FOR INCOMPLETE DATA**

Suppose that $U = \{x_1, x_2, \ldots, x_n\} \subset R^m$ is an incomplete dataset with $n$ objects. The missing values is constrained by two conditions: each original feature vector $x_i$ retains at least one component and each feature has at least one value presented in the incomplete data set. In this section, we present a new imputation algorithm for incomplete data and a three-way ensemble clustering algorithm based on the imputation result. Firstly, the data set $U$ can be classified into two disjoint subsets. One set $U_W$ requires each object with non-missing values and the other set $U_M$, where $U_W \cup U_M = U$. Secondly, Algorithm 1 is used to cluster the set $U_W$ and the mean attribute’s value of each cluster is used to fill the objects in $U_M$. Perturbation analysis of cluster centroid is applied to search the optimal imputation. Based on proposed imputation method, a three-way ensemble clustering algorithm is proposed by integrating the ideas of clustering ensemble and three-way decision. The specific process of three-way ensemble clustering algorithm for incomplete data is shown in Fig. 3, where Algorithm 2 and Algorithm 3 are described specifically in the following Section A and Section B, respectively.

**A. MISSING VALUE FILLING ALGORITHM**

In the study of incomplete data clustering, the effective filling of incomplete data is the key to improve the accuracy of clustering result. There are many ways to fill incomplete data. For example: mean imputation, regression imputation, multiple imputation, etc. In this subsection, we present an improved mean imputation incomplete data clustering algorithm based on hard clustering algorithm. Firstly, we use
k-means algorithm to cluster the objects with non-missing values \( C_W = \{C_1, C_2, \ldots, C_k\}\) of set \( U_W \). Secondly, as for an object \( x_i \) with missing values in \( U_M \), the missing value \( x_i^d = * \) (the \( d \)-th feature of \( x_i, d \in \{1, 2, \ldots, m\} \)) is filled by the corresponding attribute mean of all samples in each cluster \( C_i (i = 1, 2, \ldots, k) \) of \( U_W \), respectively. Thirdly, in order to search the optimal imputation, we compare the impacts of each filled object \( x_i \) on corresponding cluster centroid by adding \( x_i \) with \( |C_i|/k \) times into \( C_i \) and compute the new cluster center of \( C_i' \). Then, recalculate the cluster centroid and compute the difference between the new and the old center. Finally, select the minimum value among these differences, and assign \( x_i \) to the corresponding cluster, the missing value can be determined at the same time. Algorithm 2 is designed to describe the process of the proposed imputation method.

**Algorithm 2 Missing Value Filling Algorithm**

**Input:** Clustering result of complete data \( C_W = \{C_1, C_2, \ldots, C_k\} \).

**Output:** Clustering result after imputation \( C' = \{C'_1, C'_2, \ldots, C'_k\} \).

1. if \( U_M \neq \emptyset, x_i \in U_M \), then
2. for \( j \leftarrow 1 \) to \( k \), do
3. set \( x_i^d = \frac{1}{|C_j|} \sum_{x \in C_j} x^d \), where \( x_i^d = * \) is the \( d \)-th attribute of object \( x_i \).
4. add \( x_i \) with \( |C_j|/k \) times into \( C_j \) and obtain the new cluster \( C_j' \).
5. recalculate the cluster centroid by \( z_i^d = \frac{1}{|C'_j|} \sum_{x \in C'_j} x \)
   \[ (i = 1, 2, \ldots, k), \]
   and compute the difference \( d_i = |z_i - z_i'| \).
6. end for
7. select the minimum value among \( d_i \) \((i = 1, 2, \ldots, k)\), and assign \( x_i \) to \( C_j \), the missing value of object \( x_i \) can determined at the same time.
8. until the set \( U_M \) has no object with missing values
9. end if
10. return \( C' = \{C'_1, C'_2, \ldots, C'_k\} \)

**TABLE 2. Different represents of the same clustering results.**

| \( C_1 \) | \( C_2 \) | \( C_3 \) |
|---|---|---|
| \( v_1 \) | 1 | 2 | 3 |
| \( v_2 \) | 1 | 2 | 3 |
| \( v_3 \) | 2 | 3 | 2 |
| \( v_4 \) | 2 | 3 | 2 |
| \( v_5 \) | 3 | 1 | 1 |
| \( v_6 \) | 3 | 1 | 1 |

2) \( \bigcup_{i=1}^{k} (Co(C_i) \cup Fr(C_i)) = U \),
3) \( Co(C_i) \cap Co(C_j) = \emptyset, i \neq j \).

In this subsection, we present a three-way ensemble clustering algorithm for incomplete system based on our proposed imputation method. This process can be divided into two major steps: cluster label matching and three-way clustering based on voting method.

Suppose that we use algorithm 1 to obtain a set of base clustering results \( C_{W1}, C_{W2}, \ldots, C_{Wm} \) by selecting different initial centers and use algorithm 2 to obtain clustering result after imputation \( C'_1, C'_2, \ldots, C'_k \). The number of identical objects covered by each pair of clusters \( C'_i \) and \( C'_j \) (\( 1 \leq i, j \leq k \)) are recorded in the overlap matrix of \( k \times k \). Although we have obtained the clustering results of \( U, C'_i \) cannot be directly used for the conclusion of the next stage due to the lack of priori category information. As an example, we consider the data set \( V = \{v_1, v_2, v_3, v_4, v_5, v_6\} \) and let \( C_1, C_2, C_3 \) be three clustering results of \( V \) which are shown in Table 2. Although the objects are expressed in different orders, they represent the same clustering result. In order to combine the clustering results, the cluster labels must be matched to establish the correspondence between each other.

In general, the number of identical objects covered by the corresponding cluster labels should be the largest. Therefore, according to this heuristic, we can match the cluster label. Firstly, choose the cluster label that covers the largest number of identical objects to establish a correspondence and remove the result from the \( k \times k \) overlap matrix. Finally, repeat the above process until all the cluster labels have established the corresponding relationship. For convenience, we choose the clustering result \( C'_i \) as matching criterion and match the other clustering results with \( C'_i \) during label matching. The implementation steps are shown in steps 1-6 of Algorithm 3.

If the label of the cluster is already matched, an updated cluster results \( C_1, C_2, \ldots, C_t \) are obtained. In the next steps, we use the voting-based method to achieve three-way cluster results. Firstly, compute the unions of clusters with the same label in all cluster members, i.e., \( C'_1 \cup C'_2 \cup \ldots \cup C'_j (j = 1, 2, \ldots, k) \). If the object \( x \in C'_1 \cup C'_2 \cup \ldots \cup C'_j \), we view \( C'_j \) as one vote if \( x \in C'_j (i = 1, 2, \ldots, t) \) and count the votes of the object \( x \). Finally, we get all the objects’ votes and denote the vote of the object \( x \) as count(\( x \)). If count(\( x \)) \( \geq t/2 \),
we assign this object to the core region of the cluster $C_j$, otherwise, we assign it to the fringe region of the cluster $C_j$. The implementation steps are shown in steps 8-16 of Algorithm 3.

**Algorithm 3** Three-Way Ensemble Clustering Algorithm

**Input:** Ensembling clustering result after imputation $C'_1, C'_2, \ldots, C'_j$

**Output:**
$C_{final} = \{(Co(C_1), Fr(C_1)), (Co(C_2), Fr(C_2)), \ldots, (Co(C_k), Fr(C_k))\}$

1. **for** $l \leftarrow 2$ to $t$, **do**
2. 2. form the matrix $A = (a_{ij})_{k \times k}, a_{ij}$ is equal to the number of the identical objects between the $i$-th cluster of $C'_1$ and the $j$-th cluster of $C'_l$.
3. **for** $i \leftarrow 1$ to $k$, **do**
4. 4. find the max value of $a_{ij}$ and let $j = i$, that is to unify the cluster label of each clustering result.
5. **end for**
6. **end for**
7. through step 1-6, we get the new clustering result $C_1, C_2, \ldots, C'_l$ after label matching.
8. **for** $j \leftarrow 1$ to $k$, **do**
9. 9. compute the unions $C'_1 \cup C'_2 \cup \ldots \cup C'_j$, if $x \in C'_i \cup C'_j \cup \ldots \cup C'_l$, **then**
10. 10. view $C'_j$ as one vote if $x \in C'_j(i = 1, 2, \ldots)$ and count the votes of the object $x$: count($x$),
11. **if** count($x$) $\geq \frac{t}{2}$, **then**
12. 12. assign $x$ to the core region of the corresponding cluster $C_j$ i.e., $x \in Co(C_j)$;
13. **else**
14. 14. assign $x$ to the fringe region of the corresponding cluster $C_j$, i.e., $x \in Fr(C_j)$;
15. **end if**
16. **end for**
17. return $\{(Co(C_1), Fr(C_1)), (Co(C_2), Fr(C_2)), \ldots, (Co(C_k), Fr(C_k))\}$

**C. GENERATION OF THE FINAL CLUSTERING**

For an incomplete system, Algorithm 1 shows the clustering results of objects with non-missing values. Algorithm 2 helps us get the best filling value for incomplete data; To generate the final three-way clustering result $C_{final}$, we repeat the processes of algorithm 1 and algorithm 2 by selecting different initial centers and obtain a set of clustering results after imputation $C'_1, C'_2, \ldots, C'_m$. Base on the strategy of ensemble clustering, three-way ensemble clustering results can be obtained by Algorithm 3. The whole process can be shown as Algorithm 4.

**IV. CLUSTERING PERFORMANCE MEASUREMENT**

Generally, clustering performance measurement is also known as “validity index”. It is similar to the effect of performance metrics of supervised learning. As for validity index, we need to adopt it to evaluate the clustering result on one side. On the other side, it can be viewed as an optimization goal during the process of clustering when the validity index is determined.

**A. ACCURACY**

The Accuracy is a frequently-used evaluation index and easy to understand. It represents the ratio between the number of correctly partitioned objects and the total number of samples. The greater value of Accuracy means the more objects are correctly divided.

**Definition 1 Accuracy (ACC Hereafter):**

$$ACC = \frac{1}{n} \sum_{i=1}^{k} \theta_i$$

where $n$ denotes the total number of objects in data set, $\theta_i$ represents the amount of objects that are exactly divided into the $i$-th cluster and the letter $k$ represents the clustering number.

**B. FOWLKES_MALLOWX_SCORES**

The Fowlkes_Mallows_Scores index (FMI) [48] is defined as the geometric mean of the precision and recall. It measures the similarity between two clustering results. The score ranges from 0 to 1. A high value indicates a good similarity between two clusters.
### Experimental results of ACC on UCI datasets.

| Datasets   | Miss Rate% | Average value | Best value |
|------------|------------|---------------|------------|
|            |            | OCS-FCM | NPS-FCM | Algorithm 4 | OCS-FCM | NPS-FCM | Algorithm 4 | OCS-FCM | NPS-FCM |
| Iris       | 5          | 0.892   | 0.853   | 0.707       | 0.991   | 0.867   | 0.840       |
|            | 10         | 0.887   | 0.847   | 0.642       | 0.987   | 0.847   | 0.760       |
|            | 15         | 0.840   | 0.837   | 0.784       | 0.873   | 0.847   | 0.753       |
|            | 20         | 0.823   | 0.809   | 0.722       | 0.887   | 0.827   | 0.753       |
|            | 25         | 0.815   | 0.738   | 0.801       | 0.872   | 0.767   | 0.787       |
|            | 30         | 0.815   | 0.741   | 0.757       | 0.867   | 0.760   | 0.780       |
| Wine       | 5          | 0.944   | 0.946   | 0.843       | 0.966   | 0.949   | 0.916       |
|            | 10         | 0.948   | 0.924   | 0.631       | 0.961   | 0.933   | 0.674       |
|            | 15         | 0.928   | 0.916   | 0.718       | 0.938   | 0.916   | 0.893       |
|            | 20         | 0.931   | 0.908   | 0.815       | 0.949   | 0.910   | 0.927       |
|            | 25         | 0.898   | 0.893   | 0.811       | 0.910   | 0.904   | 0.883       |
|            | 30         | 0.902   | 0.888   | 0.689       | 0.933   | 0.899   | 0.876       |
| Glass      | 5          | 0.662   | 0.557   | 0.563       | 0.793   | 0.584   | 0.621       |
|            | 10         | 0.665   | 0.583   | 0.492       | 0.792   | 0.598   | 0.542       |
|            | 15         | 0.668   | 0.576   | 0.614       | 0.731   | 0.584   | 0.762       |
|            | 20         | 0.702   | 0.540   | 0.684       | 0.826   | 0.570   | 0.764       |
|            | 25         | 0.757   | 0.488   | 0.611       | 0.899   | 0.519   | 0.738       |
|            | 30         | 0.701   | 0.465   | 0.677       | 0.824   | 0.495   | 0.739       |
| WDBC       | 5          | 0.929   | 0.927   | 0.866       | 0.937   | 0.930   | 0.916       |
|            | 10         | 0.924   | 0.920   | 0.734       | 0.930   | 0.926   | 0.894       |
|            | 15         | 0.924   | 0.919   | 0.811       | 0.933   | 0.919   | 0.856       |
|            | 20         | 0.921   | 0.919   | 0.830       | 0.933   | 0.919   | 0.882       |
|            | 25         | 0.920   | 0.919   | 0.918       | 0.930   | 0.919   | 0.898       |
|            | 30         | 0.918   | 0.914   | 0.890       | 0.935   | 0.919   | 0.898       |
| Pendigits  | 5          | 0.757   | 0.622   | 0.692       | 0.778   | 0.674   | 0.745       |
|            | 10         | 0.755   | 0.552   | 0.472       | 0.781   | 0.583   | 0.708       |
|            | 15         | 0.754   | 0.552   | 0.677       | 0.767   | 0.611   | 0.771       |
|            | 20         | 0.784   | 0.569   | 0.560       | 0.795   | 0.614   | 0.654       |
|            | 25         | 0.765   | 0.603   | 0.569       | 0.773   | 0.614   | 0.671       |
|            | 30         | 0.776   | 0.595   | 0.536       | 0.779   | 0.597   | 0.691       |
| Page Blocks| 5          | 0.578   | 0.439   | 0.394       | 0.578   | 0.461   | 0.435       |
|            | 10         | 0.522   | 0.424   | 0.397       | 0.522   | 0.429   | 0.412       |
|            | 15         | 0.509   | 0.433   | 0.367       | 0.509   | 0.438   | 0.407       |
|            | 20         | 0.553   | 0.444   | 0.332       | 0.553   | 0.450   | 0.352       |
|            | 25         | 0.513   | 0.465   | 0.352       | 0.513   | 0.472   | 0.382       |
|            | 30         | 0.517   | 0.478   | 0.356       | 0.517   | 0.484   | 0.441       |

**Definition 2** The Fowlkes_Mallows_Scores Index (FMI Hereafter):

$$FMI = \frac{TP}{\sqrt{(TP + FP) \times (TP + FN)}}$$

where TP is the number of True Positive (i.e. the number of pair of points that belongs to the same clusters in both labels true and labels_pred), FP is the number of False Positive (i.e. the number of pair of points that belongs to the same clusters in labels true and not in labels_pred) and FN is the number of
TABLE 5. Experimental results of FMI on UCI datasets.

| Datasets       | Miss Rate% | Average value | Best value |
|----------------|------------|---------------|------------|
|                |            | OCS-FCM       | NPS-FCM    | OCS-FCM    | NPS-FCM    |
|                |            | Algorithm 4   |            |            |            |
| Iris           | 5          | 0.799         | 0.776      | 0.660      | 0.886      | 0.789      | 0.791      |
|                | 10         | 0.784         | 0.756      | 0.543      | 0.832      | 0.764      | 0.574      |
|                | 15         | 0.747         | 0.756      | 0.526      | 0.829      | 0.774      | 0.573      |
|                | 20         | 0.711         | 0.700      | 0.411      | 0.804      | 0.724      | 0.547      |
|                | 25         | 0.704         | 0.628      | 0.430      | 0.787      | 0.660      | 0.526      |
|                | 30         | 0.697         | 0.514      | 0.396      | 0.782      | 0.563      | 0.562      |
| Wine           | 5          | 0.888         | 0.878      | 0.759      | 0.910      | 0.899      | 0.898      |
|                | 10         | 0.884         | 0.855      | 0.516      | 0.932      | 0.868      | 0.628      |
|                | 15         | 0.859         | 0.839      | 0.504      | 0.923      | 0.848      | 0.566      |
|                | 20         | 0.862         | 0.824      | 0.510      | 0.921      | 0.837      | 0.559      |
|                | 25         | 0.811         | 0.778      | 0.521      | 0.837      | 0.809      | 0.573      |
|                | 30         | 0.758         | 0.751      | 0.497      | 0.857      | 0.770      | 0.571      |
| Glass          | 5          | 0.466         | 0.404      | 0.379      | 0.563      | 0.406      | 0.387      |
|                | 10         | 0.463         | 0.405      | 0.377      | 0.558      | 0.411      | 0.396      |
|                | 15         | 0.437         | 0.443      | 0.352      | 0.533      | 0.465      | 0.359      |
|                | 20         | 0.435         | 0.426      | 0.349      | 0.479      | 0.467      | 0.381      |
|                | 25         | 0.432         | 0.415      | 0.324      | 0.469      | 0.452      | 0.351      |
|                | 30         | 0.427         | 0.412      | 0.302      | 0.456      | 0.442      | 0.303      |
| WDBC           | 5          | 0.873         | 0.866      | 0.795      | 0.883      | 0.879      | 0.842      |
|                | 10         | 0.873         | 0.865      | 0.635      | 0.886      | 0.879      | 0.718      |
|                | 15         | 0.870         | 0.863      | 0.555      | 0.888      | 0.873      | 0.581      |
|                | 20         | 0.873         | 0.863      | 0.677      | 0.886      | 0.873      | 0.729      |
|                | 25         | 0.865         | 0.863      | 0.698      | 0.886      | 0.873      | 0.729      |
|                | 30         | 0.850         | 0.849      | 0.673      | 0.890      | 0.871      | 0.727      |
| Pendigits      | 5          | 0.677         | 0.438      | 0.297      | 0.677      | 0.484      | 0.396      |
|                | 10         | 0.617         | 0.366      | 0.266      | 0.617      | 0.397      | 0.271      |
|                | 15         | 0.656         | 0.346      | 0.221      | 0.656      | 0.351      | 0.265      |
|                | 20         | 0.621         | 0.349      | 0.220      | 0.621      | 0.355      | 0.238      |
|                | 25         | 0.595         | 0.351      | 0.220      | 0.595      | 0.353      | 0.234      |
|                | 30         | 0.569         | 0.359      | 0.243      | 0.569      | 0.362      | 0.265      |
| Page Blocks    | 5          | 0.592         | 0.503      | 0.504      | 0.592      | 0.516      | 0.525      |
|                | 10         | 0.589         | 0.497      | 0.486      | 0.589      | 0.503      | 0.494      |
|                | 15         | 0.590         | 0.503      | 0.477      | 0.590      | 0.509      | 0.487      |
|                | 20         | 0.588         | 0.510      | 0.462      | 0.588      | 0.512      | 0.491      |
|                | 25         | 0.589         | 0.515      | 0.452      | 0.589      | 0.519      | 0.465      |
|                | 30         | 0.576         | 0.522      | 0.463      | 0.576      | 0.526      | 0.472      |

False Negative (i.e. the number of pair of points that belongs to the same clusters in labels_pred and not in labels True).

V. EXPERIMENTAL ILLUSTRATION

To test the performance of our proposed algorithm, six data sets from UCI Machine Learning repository [49] are employed in this section. Table 3 summarizes the detailed information of these datasets.

The incomplete data set $U_M$ are obtained by randomly selecting some missing feature values. The random selection of missing values is constrained by two conditions: (1) each original feature vector retains at least one component;
(2) each feature has at least one value present in the incomplete data set. In most cases, the higher the missing rate in the dataset, the lower the accuracy of the clustering results. Because the higher the missing rate, the accuracy of incomplete data filling will also decrease, which directly leads to the performance degradation of the clustering algorithm. Therefore, Algorithm 4 is only suitable for low missing rates. In this paper, the missing rates are set from 5% to 30%.

Because the evaluation indices ACC and FMI are only adapted to hard clustering algorithm. In order to identify the quality of our proposed three-way clustering, we use all the core regions to form a clustering result and compute ACC and FMI by using core region to represent corresponding cluster. The total number of the objects is to exclude the objects in the fringe regions when calculating ACC. Table 4 and Table 5 list the experimental results of ACC and FMI on UCI datasets, respectively. For comparing the clustering effect, the performances of OCS-FCM and NPS-FCM [13] are also presented in Table 4 and Table 5, where the bold data represents the best results. We get the average value and the best value of ACC and FMI through 100 times experimental tests under different missing rates.

From the experimental results recorded in Table 4 and Table 5, we can get the following results. The results of the average and best value of ACC and FMI of Algorithm 4 is better than the OCS-FCM and NPS-FCM: The results of ACC and FMI of Algorithm 4 are relatively stable, with the increase of missing rate, the values of ACC and FMI are decreased gradually. Because we only consider the core region in fringe regions when we compute ACC and FMI, the performances will be fluctuating for different missing rates. The value of ACC and FMI of NPS-FCM are unstable and fluctuating, which indicates that the result of filling in the missing value based on the nearest neighbor varies greatly, especially the data set of Pendigits. It is not difficult to find after analysis: the comparison method is based on the fuzzy C-means method, which is difficult to obtain good results on non-spherical data sets; Algorithm 4 is based on the method of ensemble clustering, which can significantly improve the robustness, stability and quality of clustering results.

Through the comparison and analysis of experimental results, we can conclude that the Algorithm 4 is feasible and can effectively to deal with the problem of incomplete data clustering.

VI. CONCLUSION
Because of a large number of samples with missing feature values, the single clustering algorithm cannot achieve a good clustering result for incomplete datasets. In order to overcome this problem, we use ensemble clustering technique to combine multiple clustering results into a probably better one. Three-way clustering uses core region and fringe region to chapter the three types of relationships between an element and a cluster, which is more appropriate than the use of a set. Integrating three-way clustering and ensemble clustering, we present a three-way ensemble clustering algorithm for incomplete data in this paper.

In the proposed algorithm, we cluster the objects with non-missing values and use the mean attribute’s value of each cluster to fill the missing attribute’s value, respectively. Perturbation analysis of cluster centroid is applied to search the optimal imputation. Based on proposed imputation method, we develop a three-way ensemble clustering algorithm by integrating the ideas of clustering ensemble and three-way decision. The objects with the same cluster label in different clustering results are assigned the core region of corresponding cluster while the objects with different clustering labels are assigned to the fringe region. Therefore, a three-way clustering is naturally formed. Experimental results on UCI datasets demonstrate the effectiveness of the proposed method.

In this paper, we use the clustering algorithm of k-means to obtain a set of base clustering results by generating different initial centers. For the sake of simplicity, the algorithm in this paper sets the precondition that the number of clusters k in each cluster member is the same. How to apply the algorithm to the different value of k is our planned future work. Another direction of our future research is to extend our algorithm to noise data and analysis noise level.

REFERENCES
[1] H. Fujita, T. Li, and Y. Yao, “Advances in three-way decisions and granular computing,” Knowl.-Based Syst., vol. 91, pp. 1–3, Jan. 2016.
[2] Y. Yao, “A triarchic theory of granular computing,” Granular Comput., vol. 1, no. 2, pp. 145–157, Jun. 2016.
[3] L. A. Zadeh, “Toward a theory of fuzzy information granulation and its centrality in human reasoning and fuzzy logic,” Fuzzy Sets Syst., vol. 90, no. 2, pp. 111–127, Sep. 1997.
[4] Z. Pawlak, “Some issues on rough sets,” in Transactions on Rough Sets I, vol. 3100. Berlin, Germany: Springer, 2004, pp. 1–58.
[5] Y. Qian, H. Cheng, J. Wang, J. Liang, W. Pedrycz, and C. Dang, “Grouping granular structures in human granulation intelligence,” Inf. Sci., vol. 382–383, pp. 150–169, Mar. 2017.
[6] H. Fujita, A. Gaeta, V. Loia, and F. Orciuoli, “Resilience analysis of critical infrastructures: A cognitive approach based on granular computing,” IEEE Trans. Cybern., vol. 49, no. 5, pp. 1835–1848, May 2019. [Online]. Available: http://ieeexplore.ieee.org/document/8327881/
[7] Q. Hu, D. Yu, J. Liu, and C. Wu, “Neighborhood rough set based heterogeneous feature subset selection,” Inf. Sci., vol. 178, no. 18, pp. 3577–3594, Sep. 2008.
[8] X. Yang, J. Yang, C. Wu, and D. Yu, “Dominance-based rough set approach and knowledge reductions in incomplete ordered information system,” Inf. Sci., vol. 178, no. 4, pp. 1219–1234, Feb. 2008.
[9] X. Chen, D. Li, P. Wang, and X. Yang, “A deep convolutional neural network with fuzzy rough sets for FER,” IEEE Access, vol. 8, pp. 2772–2779, 2019, doi: 10.1109/ACCESS.2019.2960769.
[10] W. Pedrycz, Granular Computing Analysis and Design of Intelligent Systems. Boca Raton, FL, USA: CRC Press, 2013.
[11] R. Xu and D. C. Wunsch, “Survey of clustering algorithms,” IEEE Trans. Neural Netw., vol. 16, no. 3, pp. 645–678, May 2005.
[12] I. Macqueen, “Some methods for classification and analysis of multivariate observations,” in Proc. 5th Berkeley Symp. Math. Statist. Probab. Berkeley, CA, USA: Univ. California Press, 1967, pp. 197–281.
[13] R. J. Hathaway and J. C. Bezdek, “Fuzzy c-means clustering of incomplete data,” IEEE Trans. Syst., Man Cybern., B (Cybern.), vol. 31, no. 5, pp. 735–744, Oct. 2001.
[14] J. Li, S. Song, Y. Zhang, and K. Li, “A robust fuzzy c-means clustering algorithm for incomplete data,” in Proc. Int. Conf. Intell. Comput. Sustain. Energy Environ., 2017, pp. 3–12.
