Hindawi
Complexity
Volume 2019, Article ID 3213808, 17 pages
https://doi.org/10.1155/2019/3213808

WILEY | Hindawi

Research Article

Fuzzy Set-Valued Information Systems and the Algorithm of Filling Missing Values for Incomplete Information Systems

Zhaohao Wang and Xiaoping Zhang

School of Mathematics and Computer Science, Shanxi Normal University, Linfen 041000, Shanxi, China

Correspondence should be addressed to Zhaohao Wang; wzhao2019@126.com

Received 13 July 2019; Accepted 5 November 2019; Published 10 December 2019

Academic Editor: Lingzhong Guo

How to effectively deal with missing values in incomplete information systems (IISs) according to the research target is still a key issue for investigating IISs. If the missing values in IISs are not handled properly, they will destroy the internal connection of data and reduce the efficiency of data usage. In this paper, in order to establish effective methods for filling missing values, we propose a new information system, namely, a fuzzy set-valued information system (FSvIS). By means of the similarity measures of fuzzy sets, we obtain several binary relations in FSvISs, and we investigate the relationship among them. This is a foundation for the researches on FSvISs in terms of rough set approach. Then, we provide an algorithm to fill the missing values in IISs with fuzzy set values. In fact, this algorithm can transform an IIS into an FSvIS. Furthermore, we also construct an algorithm to fill the missing values in IISs with set values (or real values). The effectiveness of these algorithms is analyzed. The results showed that the proposed algorithms achieve higher correct rate than traditional algorithms, and they have good stability. Finally, we discuss the importance of these algorithms for investigating IISs from the viewpoint of rough set theory.

1. Introduction

The classical rough model [1] can be used to deal with complete information systems. In practice, the lack of some data in IISs [2–9] is inevitable. For example, because the data collection process may be imperfect, human or objective conditions result in data loss or unavailability. For data mining, these missing data may have a very important impact on final decision. Therefore, how to infer unknown information from known information has important theoretical and practical significance.

Kryszkiewicz [10] defined tolerance relation in IISs to investigate IISs by using rough set approach. This tolerance relation assumed that the missing attribute values in IISs could be represented by a set of all possible values of the corresponding attributes from an optimistic perspective. Based on Kryszkiewicz's research, Leung and Li [11] presented a method for obtaining the relative reduction in IISs. Subsequently, Stefanowski and Tsoukias [9] established a new rough set model based on the other relations in IISs. Authors [8, 11–17] gave different methods to induce binary relations from IISs, and studied IISs by means of rough set theory. They had two main ways to treat the missing values. One was to delete the missing values, and the other was to take the missing values as generic values.

Based on the probability theory, Yuan et al. [18] filled the missing values in IISs by obtaining the sample that is the closest to the missing data sample in terms of Euclidean distance and correlation. Chen and Shao [19] used the Jackknife variance estimate to investigate the missing values. In addition, there are other methods to handle missing values in IISs. Wang et al. [20] addressed the missing values in IISs by means of the Hopfield neural network approach. Salama et al. [21] proposed a topology method to retrieve missing values in IISs. Clearly, these methods of filling missing values were founded through the other theories, such as, neural network and topology. In this paper, we establish a new method to fill missing values by means of rough set theory. Next, we state the motivation of giving this method. We know that the indiscernibility relation is a basic

concept in rough set theory. Given a complete information system, we can establish an indiscernibility relation. Two objects are viewed as indiscernible if they have the same values for each attribute. Therefore, we think that if two objects possess more the same values of attributes, then they have the higher degree of indiscernibility. Based on the observation, we provide a method to fill missing values. By using this method, we can convert the missing values into fuzzy set values by evaluating the relationship between the attribute values of different objects, and then we can transform fuzzy set values into set values or real values according to the principle of maximum membership degree in fuzzy set theory. It is worth noting that, in order to construct this method, we established a new information system, namely, the fuzzy set-valued information system (FSvIS) which plays an important role in the method.

The rest of this paper is organized as follows. In Section 2, some basic concepts and notations of rough sets and fuzzy sets are given. In Section 3, we propose the fuzzy set-valued information system (FSvIS), and we induce some binary relations from FSvISs. Furthermore, we investigate the connections between these binary relations. In Section 4, we provide two methods of filling missing values. One is to fill missing values with fuzzy set values, and the other is to fill missing values with set values (or real values). In Section 5, we perform several experiments to analyze the effectiveness of the proposed methods. In Section 6, we apply the proposed methods of filling missing values to investigate IISs. Section 7 concludes this paper.

2. Basic Concepts and Properties

In this section, we review some basic concepts and notations in rough sets and fuzzy sets.

2.1. Basic Concepts for Rough Sets. In this subsection, we review some basic concepts related to general binary relations and information systems [22–24].

Definition 1 (see [23]). A general binary relation on a nonempty set U is a subset of $U \times U$. R is called

(1) Reflexive, if for any $x \in U$, $(x, x) \in R$

(2) Symmetric, if for any $x, y \in U$, $(x, y) \in R$ implies $(y, x) \in R$

(3) Transitive, if for any $x, y, z \in U$, $(x, y) \in R$ and $(y, z) \in R$ imply $(x, z) \in R$

Generally, if R satisfies reflexive and symmetric, it is called a *similarity relation*; if R satisfies reflexive, symmetric, and transitive, then it is called an *equivalence relation*.

Let R be a general binary relation on U, for $x \in U$, and the successor neighbourhood $R(x)$ of x with respect to R is defined by

$$R(x) = \{y \in U \mid (x, y) \in R\}. \tag{1}$$

A triple (U, Att, V) is called an *information system*, where U is a finite nonempty set of objects called the universe, Att is a finite nonempty set of attributes, and $V = \cup_{a \in \mathrm{Att}} V_a$, where V_a called the domain of a is a nonempty set of values of attribute $a \in \mathrm{Att}$. If there exist $x \in U$ and $a \in \mathrm{Att}$ such that the value $a(x)$ of x under a is a missing value (a null of unknown value), denoted as "$*$," that is, $\exists a \in \mathrm{Att}$, $* \in V_a$, then the information system is called an *incomplete information system* (IIS).

In order to investigate the IIS by using rough set approach, Kryszkiewic [13] presented a way to induce a relation in the IIS (U, Att, V) as follows for $B \subseteq \mathrm{Att}$:

$$T_B = \{(x, y) \in U \times U \mid a(x) = a(y) \vee a(x) = * \vee a(y) = *, \forall a \in B\}. \tag{2}$$

It is easy to check that T_B is reflexive and symmetric, that is to say, T_B is a similarity relation on U.

In this paper, we call (U, R) a *generalized approximation space*, where R is a binary relation on a finite nonempty set U.

Definition 2 (see [1]). Given a generalized approximation space (U, R) and $X \subseteq U$, the lower approximation and upper approximation of X are defined as follows:

$$\underline{\mathrm{apr}}_R(X) = \{x \in U \mid R(x) \subseteq X\},$$
$$\overline{\mathrm{apr}}_R(X) = \{x \in U \mid R(x) \cap X \neq \varnothing\}. \tag{3}$$

In [23], Wang et al. constructed an uncertainty measure in generalized approximation spaces, which is defined as follows:

Definition 3. Let (U, R) be a generalized approximation space. The entropy of R is defined as follows:

$$H(R) = -\frac{1}{|U|} \sum_{x \in U} \log \frac{|R(x)|}{|U|}. \tag{4}$$

Proposition 1 (see [23]). *Let R_1 and R_2 be binary relations on U. If $R_1 \subseteq R_2$, then $H(R_1) \geq H(R_2)$.*

2.2. Basic Concepts for Fuzzy Sets. In this section, we introduce some basic concepts and measures about fuzzy sets.

A fuzzy subset A of a nonempty set U is a map from U to $[0, 1]$ [25]. The collection of all fuzzy subsets of U is denoted as $\mathscr{F}(U)$. Similarity measure is an important concept in fuzzy set theory, and it is defined as follows:

Definition 4 (see [26]). A function $S : \mathscr{F}(U) \times \mathscr{F}(U) \longrightarrow [0, 1]$ is called a similarity measure on $\mathscr{F}(U)$, if S satisfies the following properties:

(1) $S(U, \varnothing) = 0$ and $S(A, A) = 1$ for all $A \in \mathscr{F}(U)$

(2) $S(A, B) = S(B, A)$ for all $A, B \in \mathscr{F}(U)$

(3) For all $A, B, C \in \mathscr{F}(U)$, $A \subseteq B \subseteq C$, then $S(A, C) \leq S(B, C)$ and $S(A, C) \leq S(A, B)$

Particularly, a similarity measure S is called a strictly similarity measure if it also satisfies

(4) $S(A, B) = 1$ if and only if $A = B$, for all $A, B \in \mathscr{F}(U)$

Let $U = \{x_1, x_2, \ldots x_n\}$ and $A, B \in \mathscr{F}(U)$. The most popular similarity measures include:

(1) Hamming similarity measure [27]:

$$S_H(A,B) = 1 - \frac{1}{n}\sum_{i=1}^{n}\left|A(x_i) - B(x_i)\right|. \quad (5)$$

(2) Euclidean similarity measure [27]:

$$S_E(A,B) = 1 - \frac{1}{\sqrt{n}}\left[\sum_{i=1}^{n}\left(A(x_i) - B(x_i)\right)^2\right]^{1/2}. \quad (6)$$

(3) Max-min similarity measure [28]:

$$S_N(A,B) = \frac{\sum_{i=1}^{n}\left(A(x_i) \wedge B(x_i)\right)}{\sum_{i=1}^{n}\left(A(x_i) \vee B(x_i)\right)}. \quad (7)$$

Remark 1. In this paper, we always assume that $\sum_{i=1}^{n}(A(x_i) \vee B(x_i)) \neq 0$.

3. Fuzzy Set-Valued Information Systems (FSvISs)

In this section, we replace the real number in the real-valued information system with fuzzy set and propose a more general information system, that is, the fuzzy set-valued information system. It can be seen as a generalization of the probabilistic set-valued information system defined by Huang et al. [29].

Definition 5. A fuzzy set-valued information system (FSvIS) is a triple $(U, \mathrm{Att}, \mathscr{F}(V))$, where U is a nonempty set, Att is a set of attributes, and V is the basic set of attribute values. In addition, for all $a \in \mathrm{Att}$ and $x \in U$, the value $a(x)$ of x under a is a fuzzy subset of V, that is, $a(x) \in \mathscr{F}(V)$.

In some cases, if the attribute values are uncertain or missing, then it is reasonable to describe them with fuzzy set values. For example, in IISs, we may fill the missing values with fuzzy set values. In this paper, we will investigate IISs by means of FSvISs.

Example 1. Table 1 gives a FSvIS $(U, \mathrm{Att}, \mathscr{F}(V))$, where $U = \{x_1, x_2, x_3, x_4, x_5\}$, $\mathrm{Att} = \{a_1, a_2, a_3\}$, and $V = \{-1, 0, 1\}$. In Table 1, $a_1(x_1) = (0.23/-1) + (0.42/0) + (0.76/1)$ represents the value of the object x_1 under attribute a_1. $(a_1(x_1))(-1) = 0.23$ is the grade of membership of -1 in $a_1(x_1)$.

3.1. The Similarity Relations in FSvISs. The rough set approach is applied for rule extractions and attribute reductions in information systems. The key problem is how to construct binary relations from information systems. Next, we will establish some similarity relations in FSvISs. Then, we establish the relationships between them.

It is well known that, in fuzzy set theory, similarity measure is an important concept to evaluate the similarity degree between fuzzy sets.

Let $(U, \mathrm{Att}, \mathscr{F}(V))$ be a FSvIS, S be a similarity measure and $\lambda \in [0,1]$. There is a common method to construct binary relation in terms of similarity measure as follows:

$$R_B^{S^\lambda} = \{(x,y) \in U \times U \mid S(a(x), a(y)) \geq \lambda, \quad \forall a \in B\}, \quad (8)$$

where $\varnothing \neq B \subseteq \mathrm{Att}$. Clearly, $R_B^{S^\lambda}$ is a binary relation on U. The successor neighbourhood of $x \in U$ can be computed as follows:

$$R_B^{S^\lambda}(x) = \{y \in U \mid S(a(x), a(y)) \geq \lambda, \quad \forall a \in B\}. \quad (9)$$

In the following section, we limit $B \neq \varnothing$.

By (1) of Definition 4, $R_B^{S^\lambda}$ is reflexive. In addition, the symmetry of $R_B^{S^\lambda}$ is clear. Therefore, the following result is obvious.

Proposition 2. *Let $(U, Att, \mathscr{F}(V))$ be a FSvIS, S be a similarity measure, $B \subseteq Att$, and $\lambda \in [0,1]$. Then, the binary relation $R_B^{S^\lambda}$ is reflexive and symmetric.*

Proposition 2 shows that $R_B^{S^\lambda}$ is a similarity relation.

Remark 2. By equations (5)–(8), we can obtain three similarity relations: $R_B^{S_H^\lambda}$, $R_B^{S_E^\lambda}$, and $R_B^{S_N^\lambda}$.

Proposition 3. *Let $(U, Att, \mathscr{F}(V))$ be a FSvIS, S be a similarity measure, $B \subseteq Att$, and $\lambda \in [0,1]$. The following statements hold:*

(1) If $\varnothing \neq C \subseteq B$, then $R_B^{S^\lambda} \subseteq R_C^{S^\lambda}$

(2) If $0 \leq \lambda_1 \leq \lambda_2 \leq 1$, then $R_B^{S^{\lambda_2}} \subseteq R_B^{S^{\lambda_1}}$

Proof

(1) We only need to prove that $\forall x \in U$, $R_B^{S^\lambda}(x) \subseteq R_C^{S^\lambda}(x)$. $\forall y \in R_B^{S^\lambda}(x)$, by equation (9), we have that $\forall a \in B$, $S(a(x), a(y)) \geq \lambda$. By $C \subseteq B$, it is clear that $\forall a \in C$, $S(a(x), a(y)) \geq \lambda$. It follows from equation (7) that $y \in R_C^{S^\lambda}(x)$. Hence, $R_B^{S^\lambda}(x) \subseteq R_C^{S^\lambda}(x)$. Consequently, $R_B^{S^\lambda} \subseteq R_C^{S^\lambda}$.

(2) We only need to prove that $\forall x \in U$, $R_B^{S^{\lambda_2}}(x) \subseteq R_B^{S^{\lambda_1}}(x)$. $\forall y \in R_B^{S^{\lambda_2}}(x)$, by equation (9), we have that $\forall a \in B$, $S(a(x), a(y)) \geq \lambda_2$. By $\lambda_1 \leq \lambda_2$, it is clear that $\forall a \in B$, $S(a(x), a(y)) \geq \lambda_1$. It follows from equation (8) that $y \in R_B^{S^{\lambda_1}}(x)$. Hence, $R_B^{S^{\lambda_2}}(x) \subseteq R_B^{S^{\lambda_1}}(x)$. Consequently, $R_B^{S^{\lambda_2}} \subseteq R_B^{S^{\lambda_1}}$.

In the following, we establish the relationships among $R_B^{S_H^\lambda}$, $R_B^{S_E^\lambda}$, and $R_B^{S_N^\lambda}$. Firstly, we provide the connections among the similarity measures given by equations (5)–(7). □

Proposition 4. *Let U be a nonempty set. The following statements hold:*

(1) $\forall A_1, A_2 \in \mathscr{F}(U)$, $S_N(A_1, A_2) \leq S_H(A_1, A_2)$

(2) $\forall A_1, A_2 \in \mathscr{F}(U)$, $S_E(A_1, A_2) \leq S_H(A_1, A_2)$

Proof

(1) We may assume that $U = \{x_1, x_2, \ldots, x_n\}$. Let $A_1, A_2 \in \mathscr{F}(U)$. It is easy to verify that

Table 1: A fuzzy set-valued information system.

Objects	a_1	a_2	a_3
x_1	(0.23/−1) + (0.42/0) + (0.38/1)	(0.45/−1) + (0.22/0) + (0.15/1)	(0.32/−1) + (0.21/0) + (0.58/1)
x_2	(0.15/−1) + (0.23/0) + (0.76/1)	(0.15/−1) + (0.26/0) + (0.89/1)	(0.28/0) + (0.85/1)
x_3	(0.25/−1) + (0.43/0) + (0.36/1)	(0.43/−1) + (0.26/0) + (0.12/1)	(0.33/−1) + (0.22/0) + (0.57/1)
x_4	(0.21/−1) + (0.41/0) + (0.35/1)	(0.14/−1) + (0.25/0) + (0.87/1)	(0.34/−1) + (0.20/0) + (0.59/1)
x_5	(0.37/−1) + (0.86/0)	(0.42/−1) + (0.24/0) + (0.13/1)	(0.73/−1) + (0.15/0) + (0.23/1)

$$\begin{aligned}&\sum_{i=1}^{n}\left(A_1(x_i)\vee A_2(x_i)\right)-\sum_{i=1}^{n}\left(A_1(x_i)\wedge A_2(x_i)\right)\\&\quad=\sum_{i=1}^{n}\left[\left(A_1(x_i)\vee A_2(x_i)\right)-\left(A_1(x_i)\wedge A_2(x_i)\right)\right]\\&\quad=\sum_{i=1}^{n}\left|A_1(x_i)-A_2(x_i)\right|,\end{aligned}\tag{10}$$

that is,

$$\begin{aligned}&\sum_{i=1}^{n}\left(A_1(x_i)\vee A_2(x_i)\right)-\sum_{i=1}^{n}\left(A_1(x_i)\wedge A_2(x_i)\right)\\&\quad=\sum_{i=1}^{n}\left|A_1(x_i)-A_2(x_i)\right|.\end{aligned}\tag{11}$$

In addition, it is clear that

$$\frac{1}{n}\sum_{i=1}^{n}\left(A_1(x_i)\vee A_2(x_i)\right)\le 1.\tag{12}$$

Therefore, by equations (11) and (12), we have that

$$\begin{aligned}&\left[\sum_{i=1}^{n}\left|A_1(x_i)-A_2(x_i)\right|\right]\left[\frac{1}{n}\sum_{i=1}^{n}\left(A_1(x_i)\vee A_2(x_i)\right)\right]\\&\quad\le\sum_{i=1}^{n}\left|A_1(x_i)-A_2(x_i)\right|\\&\quad=\sum_{i=1}^{n}\left(A_1(x_i)\vee A_2(x_i)\right)-\sum_{i=1}^{n}\left(A_1(x_i)\wedge A_2(x_i)\right).\end{aligned}\tag{13}$$

Thus,

$$\begin{aligned}&\sum_{i=1}^{n}\left(A_1(x_i)\wedge A_2(x_i)\right)\le\sum_{i=1}^{n}\left(A_1(x_i)\vee A_2(x_i)\right)\\&\quad-\left[\sum_{i=1}^{n}\left|A_1(x_i)-A_2(x_i)\right|\right]\left[\frac{1}{n}\sum_{i=1}^{n}\left(A_1(x_i)\vee A_2(x_i)\right)\right].\end{aligned}\tag{14}$$

By Remark 1, $\sum_{i=1}^{n}\left(A_1(x_i)\vee A_2(x_i)\right)\neq 0$. Therefore,

$$\frac{\sum_{i=1}^{n}\left(A_1(x_i)\wedge A_2(x_i)\right)}{\sum_{i=1}^{n}\left(A_1(x_i)\vee A_2(x_i)\right)}\le 1-\frac{1}{n}\sum_{i=1}^{n}\left|A_1(x_i)-A_2(x_i)\right|.\tag{15}$$

By equations (5) and (7), we conclude that $S_N(A_1, A_2)\le S_H(A_1, A_2)$.

(2) Let $A_1, A_2\in\mathcal{F}(U)$. Next, we will use mathematical induction to prove $S_E(A_1, A_2)\le S_H(A_1, A_2)$. If $n=1$, it is clear that $S_E(A_1, A_2)=S_H(A_1, A_2)$, which implies that $S_E(A_1, A_2)\le S_H(A_1, A_2)$ is true.

Assume that $S_E(A_1, A_2)\le S_H(A_1, A_2)$ is true when $n=k$. By equations (5) and (6), we have that

$$\begin{aligned}&1-\frac{1}{\sqrt{k}}\left[\sum_{i=1}^{k}\left(A_1(x_i)-A_2(x_i)\right)^2\right]^{1/2}\\&\quad\le 1-\frac{1}{k}\sum_{i=1}^{k}\left|A_1(x_i)-A_2(x_i)\right|,\end{aligned}\tag{16}$$

where $U=\{x_1, x_2, \ldots, x_k\}$. This implies that

$$\left[\sum_{i=1}^{k}\left|A_1(x_i)-A_2(x_i)\right|\right]^2\le k\cdot\sum_{i=1}^{k}\left(A_1(x_i)-A_2(x_i)\right)^2.\tag{17}$$

Next, we shall prove that the conclusion is true when $n=k+1$. By equation (5) and (6), we only need to prove that

$$\begin{aligned}&\frac{1}{k+1}\sum_{i=1}^{k+1}\left|A_1(x_i)-A_2(x_i)\right|\\&\quad\le\frac{1}{\sqrt{k+1}}\left[\sum_{i=1}^{k+1}\left(A_1(x_i)-A_2(x_i)\right)^2\right]^{1/2},\end{aligned}\tag{18}$$

that is,

$$\left[\sum_{i=1}^{k+1}\left|A_1(x_i)-A_2(x_i)\right|\right]^2\le(k+1)\sum_{i=1}^{k+1}\left(A_1(x_i)-A_2(x_i)\right)^2.\tag{19}$$

For simplicity, we write $M_i=\left|A_1(x_i)-A_2(x_i)\right|$. Hence, we only need to prove that

$$\left[\sum_{i=1}^{k+1}M_i\right]^2\le(k+1)\sum_{i=1}^{k+1}M_i^2.\tag{20}$$

In addition, equation (17) can be written by

$$\left[\sum_{i=1}^{k}M_i\right]^2\le k\cdot\sum_{i=1}^{k}M_i^2.\tag{21}$$

By equation (21), it is clear that

$$\begin{aligned}\left[\sum_{i=1}^{k+1} M_i\right]^2 &= \left[\sum_{i=1}^{k} M_i + M_{k+1}\right]^2 \\ &= \left[\sum_{i=1}^{k} M_i\right]^2 + 2M_{k+1}\left[\sum_{i=1}^{k} M_i\right] + M_{k+1}^2 \\ &\le k \cdot \sum_{i=1}^{k} M_i^2 + 2M_{k+1}\left[\sum_{i=1}^{k} M_i\right] + M_{k+1}^2 \\ &= k \cdot \sum_{i=1}^{k} M_i^2 + \sum_{i=1}^{k}\left[2M_{k+1}M_i\right] + M_{k+1}^2 \\ &\le k \cdot \sum_{i=1}^{k} M_i^2 + \sum_{i=1}^{k}\left[M_{k+1}^2 + M_i^2\right] + M_{k+1}^2 \\ &= k \cdot \sum_{i=1}^{k} M_i^2 + kM_{k+1}^2 + \sum_{i=1}^{k} M_i^2 + M_{k+1}^2 \\ &= k \cdot \sum_{i=1}^{k+1} M_i^2 + \sum_{i=1}^{k+1} M_i^2 = (k+1)\sum_{i=1}^{k+1} M_i^2.\end{aligned} \tag{22}$$

This completes the proof. □

According to Proposition 4 and equation (8), the following result is obvious.

Theorem 1. *Let $(U, Att, \mathscr{F}(V))$ be a FSvIS, $B \subseteq Att$ and $\lambda \in [0,1]$. Then, the following statements hold:*

(1) $R_B^{S_N^\lambda} \subseteq R_B^{S_H^\lambda}$

(2) $R_B^{S_E^\lambda} \subseteq R_B^{S_H^\lambda}$

3.2. The Uncertainty Measures of FSvISs. In Section 3.1, we establish three similarity relations in FSvISs. If we use the rough set approach to investigate FSvISs, we usually need to choose reasonable similarity relations according to the actual condition. Therefore, in this section, we discuss the uncertainty measures of these similarity relations so as to provide evidence for the choice of similarity relations.

Proposition 5. *Let $(U, Att, \mathscr{F}(V))$ be a FSvIS, $B \subseteq Att$ and $\lambda \in [0,1]$. The following statements hold:*

(1) $\forall X \subseteq U$, $\underline{apr}_{R_B^{S_N^\lambda}}(X) \supseteq \underline{apr}_{R_B^{S_H^\lambda}}(X)$ *and* $\underline{apr}_{R_B^{S_E^\lambda}}(X) \supseteq \underline{apr}_{R_B^{S_H^\lambda}}(X)$

(2) $\forall X \subseteq U$, $\overline{apr}_{R_B^{S_N^\lambda}}(X) \subseteq \overline{apr}_{R_B^{S_H^\lambda}}(X)$ *and* $\overline{apr}_{R_B^{S_E^\lambda}}(X) \subseteq \overline{apr}_{R_B^{S_H^\lambda}}(X)$

Proof. It is straightforward from Theorem 1 and Definition 2. □

Proposition 6. *Let $(U, Att, \mathscr{F}(V))$ be a FSvIS, $B \subseteq Att$ and $\lambda \in [0,1]$. The following statements hold:*

(1) $H(R_B^{S_N^\lambda}) \ge H(R_B^{S_H^\lambda})$

(2) $H(R_B^{S_E^\lambda}) \ge H(R_B^{S_H^\lambda})$

Proof. It is straightforward from Theorem 1 and Proposition 1. □

4. Algorithms of Filling Missing Values in IISs

We know that complete information systems can be investigated by the rough set approach. In general, in order to discuss an IIS by means of rough set theory, we need to fill missing values in the IIS. That is to say, we first need to transform the IIS into a complete information system. In this section, we provide some methods to fill missing values in IISs. Note that data are often divided into two types: discrete data and continuous data. Next, we study the issue of filling missing data under two cases.

4.1. Algorithm of Filling Missing Values in IISs of Discrete Data. Clearly, the missing values possess the property of uncertainty; therefore, it is reasonable to use fuzzy set values (or set values) to fill missing values in IISs. In this section, we provide two schemes, namely, replacing the missing values with fuzzy set values and replacing the missing values with set values.

4.1.1. Filling the Missing Values with Fuzzy Set Values. Next, we provide a method to fill missing values in IISs of discrete data. We replace the missing values with fuzzy set values. In fact, this method can transform IISs into FSvISs.

In the IIS given by Table 2, the value domain of a_1 is $\{L, H, N, *\}$, and the value of x_2 under attribute a_1 is the missing value, that is, $a_1(x_2) = *$. We think that this missing value may be L or H or N. We cannot determine which one is $a_1(x_2)$, but we can find a way to evaluate the degree that L (or H or N) is $a_1(x_2)$. That is, we can replace the missing values with fuzzy sets on $\{L, H, N\}$. Next, we outline *the main idea of filling missing data*. The indiscernibility relation is a basic concept in rough set theory. Given a complete information system, we can establish an indiscernibility relation. Two objects are viewed as indiscernible if they have the same values for each attribute. Therefore, we think that if two objects possess more the same values of attributes, then they have the higher degree of indiscernibility. For example, in Table 2, $a_1(x_2) = *$. x_2 and x_4 have the same values of five attributes $(\{a_2, a_3, a_4, a_5, a_6\})$; x_2 and x_3 have the same values of two attributes $(\{a_2, a_6\})$. Thus, x_2 and x_4 have the higher degree of indiscernibility. That is to say, the possibility degree of $a_1(x_2) = a_1(x_4) = H$ is more than that of $a_1(x_2) = a_1(x_3) = N$. Based on this observation, we obtain Algorithm 1.

Remark 3. In Step 2 of Algorithm 1, $D(x_l, x_i)$ describes how many attributes for x_l and x_i have the same value. Thus, it can be used to characterize the degree of indiscernibility of x_l

Table 2: An IIS of discrete data.

Objects	a_1	a_2	a_3	a_4	a_5	a_6
x_1	L	N	N	L	N	N
x_2	*	H	H	H	H	H
x_3	N	H	N	N	N	H
x_4	H	H	H	H	H	H
x_5	N	H	N	N	N	H

and x_i. In Step 3, $|t^{a_k}|/|U|$ can be considered as probability of the elements whose attribute values are t in U.

Example 2. In Table 3, $a_1(x_5) = *$. Clearly, $V^*_{a_1} = \{L, H, N\}$.

Step 1: Take $L \in V^*_{a_1}$. It is easy to compute that $L^{a_1} = \{x_1, x_4, x_7, x_8\}$.

Step 2: We can compute that

$$D(x_5, x_1) = D(x_5, x_4) = D(x_5, x_7) = 0,$$
$$D(x_5, x_8) = \frac{1}{3}. \tag{23}$$

Thus, $D(x_5, L^{a_1}) = \max\{0, 1/3\} = 1/3$.

Step 3: $(a_1^F(x_5))(L) = 1/3$.

Similarly, we can compute that $(a_1^F(x_5))(H) = 1/3$ and $(a_1^F(x_5))(N) = 2/3$. Therefore, we fill the missing value $a_1(x_5)$ with the following fuzzy set:

$$a_1^F(x_5) = \frac{1/3}{L} + \frac{1/3}{H} + \frac{2/3}{N}. \tag{24}$$

4.1.2. Filling the Missing Values with Set Values. Based on the discussion of Section 4.1.1, we can replace a missing value with a fuzzy set. In fact, we can transform the fuzzy set into a set by means of the maximum membership degree law. Let (U, Att, V) be an IIS of discrete data. Assume that $a_k(x_l) = *$, where $x_l \in U$ and $a_k \in \text{Att}$. By Algorithm 1, we obtain the fuzzy set $a_k^F(x_l)$. Thus, we can use the following set to fill the missing value $a_k(x_l)$:

$$a_k^S(x_l) = \left\{t \in V^*_{a_k} \mid \left(a_k^F(x_l)\right)(t) = M\right\},$$
$$\text{where } M = \max\left\{\left(a_k^F(x_l)\right)(t) \mid t \in V^*_{a_k}\right\}. \tag{25}$$

Example 3. In Example 2, we obtain that $a_1^F(x_5) = ((1/3)/L) + ((1/3)/H) + ((2/3)/N)$. Thus, the maximal membership degree M is 2/3, that is, $M = 2/3$. By equation (25), we have that $a_1^S(x_5) = \{N\}$. That is to say, we can fill the missing value $a_1(x_5)$ with the set $\{N\}$. In Table 4, we know that $a_1(x_5)$ should be N. This coincides with the filling values by our algorithm.

In Table 3, $a_1(x_9)$ and $a_2(x_7)$ are also missing. By Algorithm 1, we can obtain that

$$a_1^F(x_9) = \frac{1/3}{L} + \frac{1/3}{H} + \frac{1/3}{N},$$
$$a_2^F(x_7) = \frac{2/3}{L} + \frac{1/3}{H} + \frac{1/3}{N}. \tag{26}$$

Table 3: Missing dataset.

Objects	a_1	a_2	a_3
x_1	L	N	N
x_2	H	L	H
x_3	N	H	N
x_4	L	L	L
x_5	*	H	H
x_6	N	H	H
x_7	L	*	L
x_8	L	H	N
x_9	*	N	H
x_{10}	H	L	N

Thus, we have that $a_1^S(x_9) = \{L, H, N\}$ and $a_2^S(x_7) = \{L\}$.

4.2. Algorithm of Filling Missing Values in IISs of Continuous Data. Similar to the discussion of Section 4.1, we investigate the corresponding issues of IISs of continuous data in this section.

4.2.1. Filling the Missing Values with Fuzzy Set Values. Similar to Algorithm 1, we give Algorithm 2 to fill the missing value in IISs of continuous data.

Example 4. In this example, we discuss the Iris information system given by Table 5 from UCI. Suppose that $a_1(x_5)$ and $a_2(x_7)$ in Table 5 are missing. We obtain Table 6. Next, we use the IIS given by Table 6 to illustrate Algorithm 2.

In Table 6, $a_1(x_5) = *$. Clearly, $V^*_{a_1} = \{4.3, 4.9, 5.4, 5.7, 5.8, 6.3, 6.6\}$. We take the thresholds $\lambda_1 = 0.2$ and $\lambda_2 = \lambda_3 = \lambda_4 = 0.5$.

Step 1: Take $4.3 \in V^*_{a_1}$. It is easy to compute that $4.3^{a_1}_{\lambda_1} = \{x_i \in U \mid |a_1(x_i) - 4.3| \le 0.2\} = \{x_1\}$.

Step 2: Since $|a_2(x_1) - a_2(x_5)| = 0.5 \le \lambda_2 = 0.5$, $|a_3(x_1) - a_3(x_5)| = 0.3 \le \lambda_3 = 0.5$, and $|a_4(x_1) - a_4(x_5)| = 0.2 \le \lambda_4 = 0.5$, it follows that $\{a_j \in \text{Att} \mid |a_j(x_5) - a_j(x_1)| \le \lambda_j\} = \{a_2, a_3, a_4\}$, and thus $D(x_5, x_1) = 3/4$. This implies that $D(x_5, 4.3^{a_1}_{\lambda_1}) = \max\{3/4\} = 0.75$.

Step 3: $(a_1^F(x_5))(4.3) = 0.75$.

Similarly, we can compute that $(a_1^F(x_5)(4.9)) = 0$, $(a_1^F(x_5))(5.4) = 0.75$, $(a_1^F(x_5))(5.7) = 0.75$, $(a_1^F(x_5))(5.8) = 0.75$, $(a_1^F(x_5))(6.3) = 0.25$, and $(a_1^F(x_5))(6.6) = 0$. Therefore, we fill the missing value $a_1(x_5)$ with the following fuzzy set:

$$a_1^F(x_5) = \frac{0.75}{4.3} + \frac{0}{4.9} + \frac{0.75}{5.4} + \frac{0.75}{5.7} + \frac{0.75}{5.8} + \frac{0.25}{6.3} + \frac{0}{6.6}. \tag{27}$$

4.2.2. Filling the Missing Values with Real Values. Based on the discussion of Section 4.2.1, we can replace a missing value with a fuzzy set. Clearly, we can transform the fuzzy set

Let (U, Att, V) be an IIS of discrete data. Assume that $a_k(x_l) = *$, where $x_l \in U$ and $a_k \in \mathrm{Att}$. $V^*_{a_k}$ denotes the set $V_{a_k} - \{*\}$, that is, $V^*_{a_k} = V_{a_k} - \{*\}$. We shall use a fuzzy set of $V^*_{a_k}$ to represent the missing value $a_k(x_l)$, and we denote the fuzzy set by $a^F_k(x_l)$. Thus, $\forall t \in V^*_{a_k}$, we need to compute the membership degree $(a^F_k(x_l))(t)$. Next, we establish the steps of filling the missing value $a_k(x_l)$ as follows:

Step 1: $\forall t \in V^*_{a_k}$, compute $t^{a_k} = \{x_i \in U \mid a_k(x_i) = t\}$

Step 2: Compute $D(x_l, t^{a_k}) = \max[D(x_l, x_i) \mid x_i \in t^{a_k}]$, where $D(x_l, x_i) = \left|\left\{a_j \in Att \mid a_j(x_l) = a_j(x_i)\right\}\right| / |\mathrm{Att}|$

Step 3: Assign a value to $(a^F_k(x_l))(t)$, $(a^F_k(x_l))(t) = \begin{cases} D(x_l, t^{a_k}), D(x_l, t^{a_k}) \neq 0, \\ |t^{a_k}|/|U|, D(x_l, t^{a_k}) = 0. \end{cases}$

Algorithm 1: Filling the missing values in IISs of discrete data with fuzzy set values.

Table 4: Original dataset of Table 3.

Objects	a_1	a_2	a_3
x_1	L	N	N
x_2	H	L	H
x_3	N	H	N
x_4	L	L	L
x_5	N	H	H
x_6	N	H	H
x_7	L	L	L
x_8	L	H	N
x_9	N	N	H
x_{10}	H	L	N

Table 5: Original dataset of Table 6.

Objects	a_1	a_2	a_3	a_4
x_1	4.3	3	1.1	0.1
x_2	5.8	4	1.2	0.2
x_3	5.7	4.4	1.5	0.4
x_4	5.4	3.9	1.3	0.4
x_5	5.1	3.5	1.4	0.3
x_6	5.7	3.8	1.7	0.31
x_7	5.7	2.8	4.5	1.3
x_8	6.3	3.3	4.7	1.6
x_9	4.9	2.4	3.3	1
x_{10}	6.6	2.9	4.6	1.3

into real value by means of the maximum membership degree law. Let (U, Att, V) be an IIS of continuous data. Assume that $a_k(x_l) = *$, where $x_l \in U$ and $a_k \in \mathrm{Att}$. By Algorithm 2, we obtain the fuzzy set $a^F_k(x_l)$. Thus, we can use the following real value to fill the missing value $a_k(x_l)$:

$$a^R_k(x_l) = \frac{\sum_{t \in \mathrm{smd}} t}{|\mathrm{smd}|}, \tag{28}$$

where $\mathrm{smd} = \left\{t \in V^*_{a_k} \mid (a^F_k(x_l))(t) = M\right\}$ and $M = \max\left\{(a^F_k(x_l))(t) \mid t \in V^*_{a_k}\right\}$.

Example 5. From Example 4, we know that

$$a^F_1(x_5) = \frac{0.75}{4.3} + \frac{0}{4.9} + \frac{0.75}{5.4} + \frac{0.75}{5.7} + \frac{0.75}{5.8} + \frac{0.25}{6.3} + \frac{0}{6.6}. \tag{29}$$

Table 6: Missing dataset.

Objects	a_1	a_2	a_3	a_4
x_1	4.3	3	1.1	0.1
x_2	5.8	4	1.2	0.2
x_3	5.7	4.4	1.5	0.4
x_4	5.4	3.9	1.3	0.4
x_5	*	3.5	1.4	0.3
x_6	5.7	3.8	1.7	0.31
x_7	5.7	*	4.5	1.3
x_8	6.3	3.3	4.7	1.6
x_9	4.9	2.4	3.3	1
x_{10}	6.6	2.9	4.6	1.3

By equation (28), we compute that

$$\begin{aligned} M &= 0.75, \\ \mathrm{smd} &= \{4.3, 5.4, 5.7, 5.8\}. \end{aligned} \tag{30}$$

Thus, we obtain that $a^R_1(x_5) = (4.3 + 5.4 + 5.7 + 5.8)/4 = 5.3$.

Remark 4. By Table 5, we know that $a_1(x_5)$ should be 5.1. By Example 5, we fill the value $a^R_1(x_5) = 5.3$ under the assumption that $a_1(x_5)$ and $a_2(x_7)$ are missing. The deviation of $a_1(x_5)$ and $a^R_1(x_5) = 5.3$ is within 0.2. This indicates that the method of filling missing value is effective.

5. Experiments and Effectiveness Analysis

In this section, we employ several experiments to show the effectiveness of the algorithms given by Section 4. We compare the proposed methods with a representative algorithm. The summary information of experimental datasets is shown in Table 7. Adult dataset and Abalone dataset are taken from UCI (http://archive.ics.uci.edu/ml/datasets.php).

5.1. Effectiveness Analysis of the Algorithm of Filling Missing Values in IISs of Discrete Data. In this part, we will conduct two groups of experiments. They are used to compare the effectiveness of methods of filling missing values from different points of view. Frequency Estimator-based filling method (Algorithm FE) [30] is a common method of filling missing data. In this section, a comparison of the proposed methods with Algorithm FE is given.

Let (U, Att, V) be an IIS of continuous data. Assume that $a_k(x_l) = *$, where $x_l \in U$ and $a_k \in \text{Att}$. $V^*_{a_k}$ denotes the set $V_{a_k} - \{*\}$. We shall use a fuzzy set of $V^*_{a_k}$ to represent the missing value $a_k(x_l)$, and we denote the fuzzy set by $a^F_k(x_l)$. Thus, $\forall t \in V^*_{a_k}$, and we need to compute the membership degree $(a^F_k(x_l))(t)$. Next, we establish the steps of filling the missing value $a_k(x_l)$ as follows:

Step 1: $\forall t \in V^*_{a_k}$, compute $t^{a_k}_{\lambda_k} = \{x_i \in U \mid |a_k(x_i) - t| \le \lambda_k\}$, where λ_k is a threshold on a_k

Step 2: Compute $D(x_l, t^{a_k}_{\lambda_k}) = \max\left\{D(x_l, x_i) \mid x_i \in t^{a_k}_{\lambda_k}\right\}$, where $D(x_l, x_i) = \left|\left\{a_j \in \text{Att} \mid |a_j(x_l) - a_j(x_i)| \le \lambda_j\right\}\right| / |\text{Att}|$.

Step 3: Assign a value to $(a^F_k(x_l))(t)$, $(a^F_k(x_l))(t) = \begin{cases} D(x_l, t^{a_k}_{\lambda_k}), D(x_l, t^{a_k}_{\lambda_k}) \ne 0, \\ |t^{a_k}_{\lambda_k}|/|U|, D(x_l, t^{a_k}_{\lambda_k}) = 0. \end{cases}$

Algorithm 2: Filling the missing values in IISs of continuous data with fuzzy set values.

Table 7: Detailed information of the datasets.

Index	Dataset	Data type	Objects	Attributes
1	Adult dataset	Nominal, numeric	48842	9
2	Abalone dataset	Numeric	4177	8

In Section 4.1, we give the method of filling the missing values with fuzzy set values. Furthermore, we obtain the method of filling the missing values with set values. By combining Section 4.1.1 and Section 4.1.2, we design Algorithm FMvSV to fill the missing values with set values (Algorithm 3).

Next, we provide a comparative study of the effectiveness for Algorithms FMvSV and FE. We first give a quantitative index of the effectiveness for filling missing values as follows.

Definition 6. Given a complete information system (U, Att) of discrete data, suppose that the values $a_{k_1}(x_{w_1})$, $a_{k_2}(x_{w_2}), \ldots, a_{k_q}(x_{w_q})$ are missing, and the filling set values are denoted as $a^S_{k_1}(x_{w_1}), a^S_{k_2}(x_{w_2}), \ldots, a^S_{k_q}(x_{w_q})$, respectively. Then, the correct rate of filling values is defined by

$$\text{CR} = \frac{\sum_{i=1}^{q} p_i}{q}, \tag{31}$$

where q is the number of the missing values, and

$$p_i = \begin{cases} 1 - \dfrac{\left|a^S_{k_i}(x_{w_i})\right| - 1}{\left|V^*_{a_{k_i}}\right|}, & a_{k_i}(x_{w_i}) \in a^S_{k_i}(x_{w_i}); \\ 0, & \text{Others.} \end{cases} \tag{32}$$

Example 6. In Table 4, $a_1(x_5) = N$, $a_1(x_9) = N$ and $a_2(x_7) = L$. In Example 3, suppose that $a_1(x_5)$, $a_1(x_9)$, and $a_2(x_7)$ in Table 4 are missing. Then, we obtain that $a^S_1(x_5) = \{N\}$, $a^S_1(x_9) = \{L, H, N\}$ and $a^S_2(x_7) = \{L\}$. Thus, by Definition 6, we can compute that

$$\begin{aligned} p_1 &= 1 - \frac{\left|a^s_1(x_5)\right| - 1}{\left|V^*_{a_1}\right|} = 1 - \frac{1-1}{3} = 1, \\ p_2 &= 1 - \frac{\left|a^s_1(x_9)\right| - 1}{\left|V^*_{a_1}\right|} = 1 - \frac{3-1}{3} = \frac{1}{3}, \\ p_3 &= 1 - \frac{\left|a^s_2(x_7)\right| - 1}{\left|V^*_{a_2}\right|} = 1 - \frac{1-1}{3} = 1. \end{aligned} \tag{33}$$

Therefore, the correct rate of filling values is $\text{CR} = (p_1 + p_2 + p_3)/3 = 0.778$.

In this section, we use some subsets of Adult dataset (see Table 7) to experiment. We need to experiment with the discrete value. We randomly select some subsets of discrete values in Adult dataset. Table 8 gives three subsets of Adult dataset.

Experiment 1. The effects of experiment times on correct rates of filling values.

In this experiment, we mainly compare the efficiency of Algorithms FMvSV and FE by the dataset AD200 in Table 8. The steps are as follows:

(i) 2.5% attribute values are randomly selected from AD200 and supposed that they are missing

(ii) By means of Algorithms FMvSV and FE, we can fill these missing values, and we can obtain the correct rate of every algorithm

The steps (i) and (ii) are repeated ten times, and the corresponding results are summarized in Table 9. Similarly, we also consider the cases of 5%, 7.5%, 10%, 12.5%, 15%, 17.5%, 20%, 22.5%, 25%, 27.5%, and 30% missing values in AD200. The results are shown in Figures 1 and 2.

Figures 1 and 2 show the following facts:

(i) The correct rate of Algorithm FMvSV is 10% – 20% higher than that of Algorithm FE.

(ii) The number of missing values is given, but the missing values in AD200 are not necessarily the same in each experiment. The correct rates of Algorithm FMvSV have little change in each experiment when the number of missing values in AD200 is a fixed value. However, in the similar case, the correct rates of Algorithm FE fluctuate obviously in each experiment. This indicates that Algorithm FMvSV does well in stability.

In Table 9, the mean value of ten correct rates related to Algorithm FMvSV can be considered as the correct rate of Algorithm FMvSV for 2.5% missing values. The correct rate of Algorithm FE for 2.5% missing values can be obtained similarly. Furthermore, for 5%, 7.5%, 10%, 12.5%, 15%, 17.5%, 20%, 22.5%, 25%, 27.5%, and 30% missing values in AD200, we also compute the correct rates of Algorithms FMvSV and FE. The results are shown in Table 10 and Figure 3.

Table 10 and Figure 3 show that the correct rates of Algorithm FMvSV monotone decrease with the increase of

Input: An IIS (U, Att, V) of discrete data, where $U = \{x_1, x_2, \ldots, x_n\}$ and $\text{Att} = \{a_1, a_2, \ldots, a_m\}$
The missing values: $a_{k_1}(x_{w_1}), a_{k_2}(x_{w_2}), \ldots, a_{k_q}(x_{w_q})$
Output: The filling set values: $a^S_{k_1}(x_{w_1}), a^S_{k_2}(x_{w_2}), \ldots, a^S_{k_q}(x_{w_q})$

(1) **for** $i = 1$ **to** q **do**
(2) **for** every t in $V^*_{a_{k_i}}$ **do**
(3) Compute $t^{a_{k_i}} = \left\{x \in U \mid a_{k_i}(x) = t\right\}$
(4) **for** every x in $t^{a_{k_i}}$ **do**
(5) Compute $D(x_{w_i}, x) = \left|\left\{a \in \text{Att} \mid a(x_{w_i}) = a(x)\right\}\right|/|\text{Att}|$
(6) **end**
(7) Compute $D(x_{w_i}, t^{a_{k_i}}) = \max\left\{D(x_{w_i}, x) \mid x \in t^{a_k}\right\}$
(8) **if** $D(x_{w_i}, t^{a_{k_i}}) \neq 0$ **then**
(9) $(a^F_{k_i}(x_{w_i}))(t) = D(x_{w_i}, t^{a_{k_i}})$
(10) **else**
(11) $(a^F_{k_i}(x_{w_i}))(t) = |t^{a_{k_i}}|/|U|$
(12) **end if**
(13) **end**
(14) $M = \max\left\{(a^F_{k_i}(x_{w_i}))(t) \mid t \in V^*_{a_{k_i}}\right\}$
(15) $a^S_{k_i}(x_{w_i}) = \left\{t \in V^*_{a_{k_i}} \mid (a^F_{k_i}(x_{w_i}))(t) = M\right\}$
(16) **end**

Algorithm 3: Algorithm FMvSV: filling the missing values with set values.

Input: An IIS (U, Att) of continuous data, where $U = \{x_1, x_2, \ldots, x_n\}$ and $\text{Att} = \{a_1, a_2, \ldots, a_m\}$.
The missing values: $a_{k_1}(x_{w_1}), a_{k_2}(x_{w_2}), \ldots, a_{k_q}(x_{w_q})$. The threshold: $\lambda_1, \lambda_2, \ldots, \lambda_m$.
Output: The filling real values: $a^R_{k_1}(x_{w_1}), a^R_{k_2}(x_{w_2}), \ldots, a^R_{k_q}(x_{w_q})$.

(1) **for** $i = 1$ **to** q **do**
(2) **for** every t in $V^*_{a_{k_i}}$ **do**
(3) Compute $t^{a_{k_i}}_{\lambda_{k_i}} = \left\{x \in U \mid |a_{k_i}(x) - t| \leq \lambda_{k_i}\right\}$
(4) **for** every x in $t^{a_{k_i}}_{\lambda_{k_i}}$ **do**
(5) Compute $D(x_{w_i}, x) = \left|\left\{a_l \in \text{Att} \mid |a_l(x_{w_i}) - a_l(x)| \leq \lambda_l\right\}\right|/|\text{Att}|$
(6) **end**
(7) Compute $D(x_{w_i}, t^{a_{k_i}}_{\lambda_{k_i}}) = \max\left\{D(x_{w_i}, x) \mid x \in t^{a_{k_i}}_{\lambda_{k_i}}\right\}$
(8) **if**$D(x_{w_i}, t^{a_{k_i}}_{\lambda_{k_i}}) \neq 0$ **then**
(9) $(a^F_{k_i}(x_{w_i}))(t) = D(x_{w_i}, t^{a_{k_i}}_{\lambda_{k_i}})$
(10) **else**
(11) $(a^F_{k_i}(x_{w_i}))(t) = |t^{a_{k_i}}_{\lambda_{k_i}}|/|U|$
(12) **end if**
(13) **end**
(14) $M = \max\left\{(a^F_{k_i}(x_{w_i}))(t) \mid t \in V^*_{a_{k_i}}\right\}$; $\text{smd} = \left\{t \in V^*_{a_{k_i}} \mid (a^F_{k_i}(x_{w_i}))(t) = M\right\}$
(15) $a^R_{k_i}(x_{w_i}) = \sum_{t \in \text{smd}} t/|\text{smd}|$
(16) **end**

Algorithm 4: Algorithm FMvRV: filling the missing values with real values.

missing values. However, the monotonicity of the correct rates of Algorithm FE is not obvious. In addition, Table 10 and Figure 3 also indicate that the effect of Algorithm FMvSV is better than that of Algorithm FE. Furthermore, when the missing values are increased to 30%, the correct rate of Algorithm FMvSV still achieves 64%.

Experiment 2. The effects of data size on correct rates of filling values.

In this experiment, we use AD200, AD400, and AD800 to discuss the effects of data size on correct rates of Algorithms FMvSV and FE. For 5% – 60% missing values, similar to the calculating method of Table 10, we can obtain

Table 8: The subsets of Adult dataset used in Experiments 1 and 2.

The name of dataset	The type of datasets	The type of attribute values	The number of objects	The number of attributes
AD200	Complete	Discrete	200	9
AD400	Complete	Discrete	400	9
AD800	Complete	Discrete	800	9

Table 9: Comparison of the correct rates of Algorithms FMvSV and FE in terms of AD200 with 2.5% missing values.

Times	1	2	3	4	5	6	7	8	9	10
FMvSV	0.87	0.777	0.807	0.762	0.799	0.788	0.815	0.804	0.776	0.789
FE	0.687	0.6	0.6	0.556	0.644	0.733	0.644	0.622	0.6	0.644

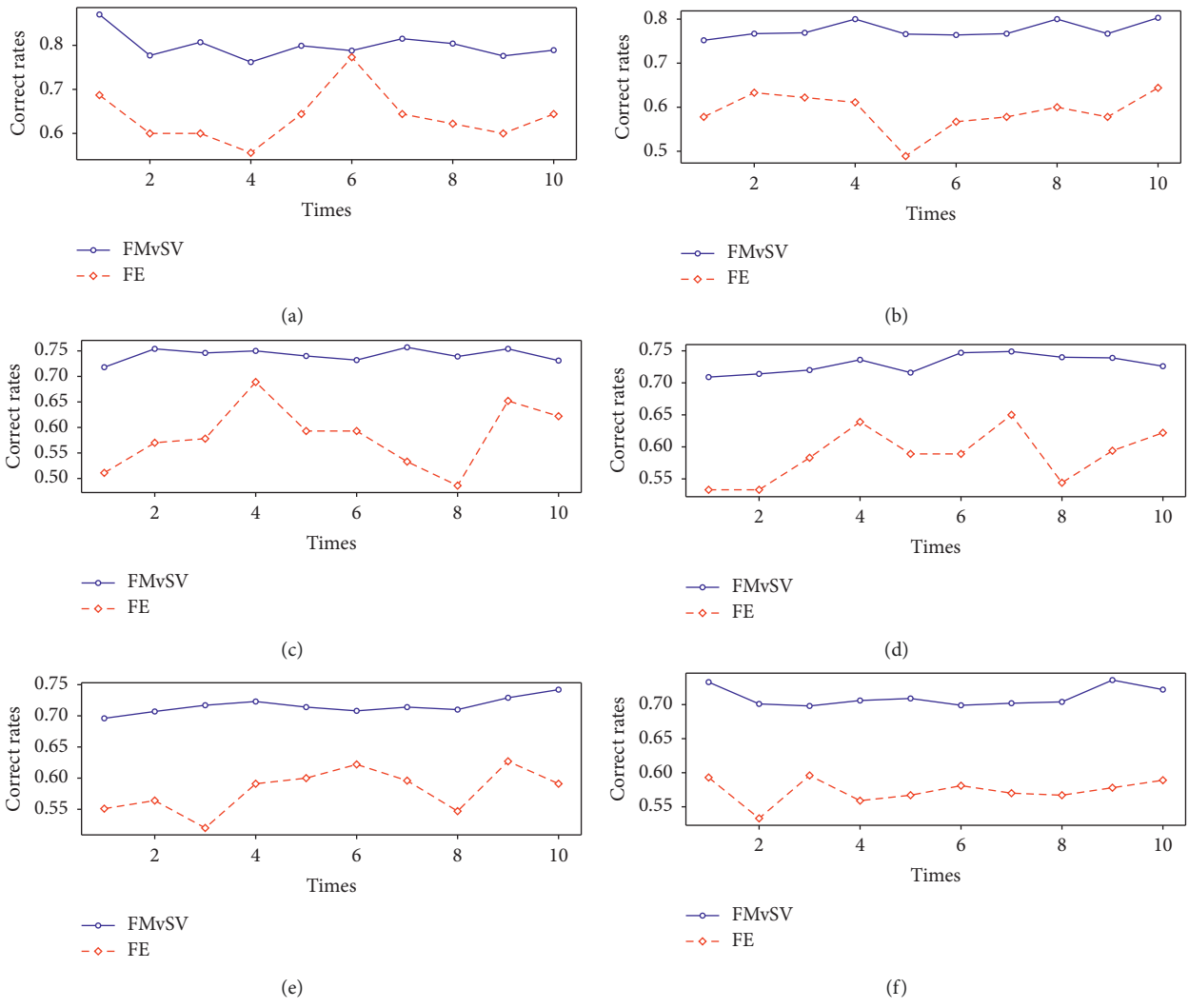

Figure 1: Comparison of the correct rates of Algorithms FMvSV and FE in terms of AD200 with 2.5% – 15% missing values. (a) 2.5% missing values. (b) 5% missing values. (c) 7.5% missing values. (d) 10% missing values. (e) 12.5% missing values. (f) 15% missing values.

the correct rate of Algorithm FMvSV (or FE) in this experiment. The results are shown in Figure 4.

Figure 4 reflects the following facts:

(i) When the data size increases under the same missing rate, the correct rate of Algorithm FMvSV remains basically the same and is higher than that of Algorithm FE. Therefore, for Algorithm FMvSV, we can divide a dataset into several small datasets, and then fill missing values to improve efficiency of it.

(ii) It is easy to see that as the data size increases, the difference between the correct rates of Algorithms FMvSV and FE becomes larger. This illustrates that

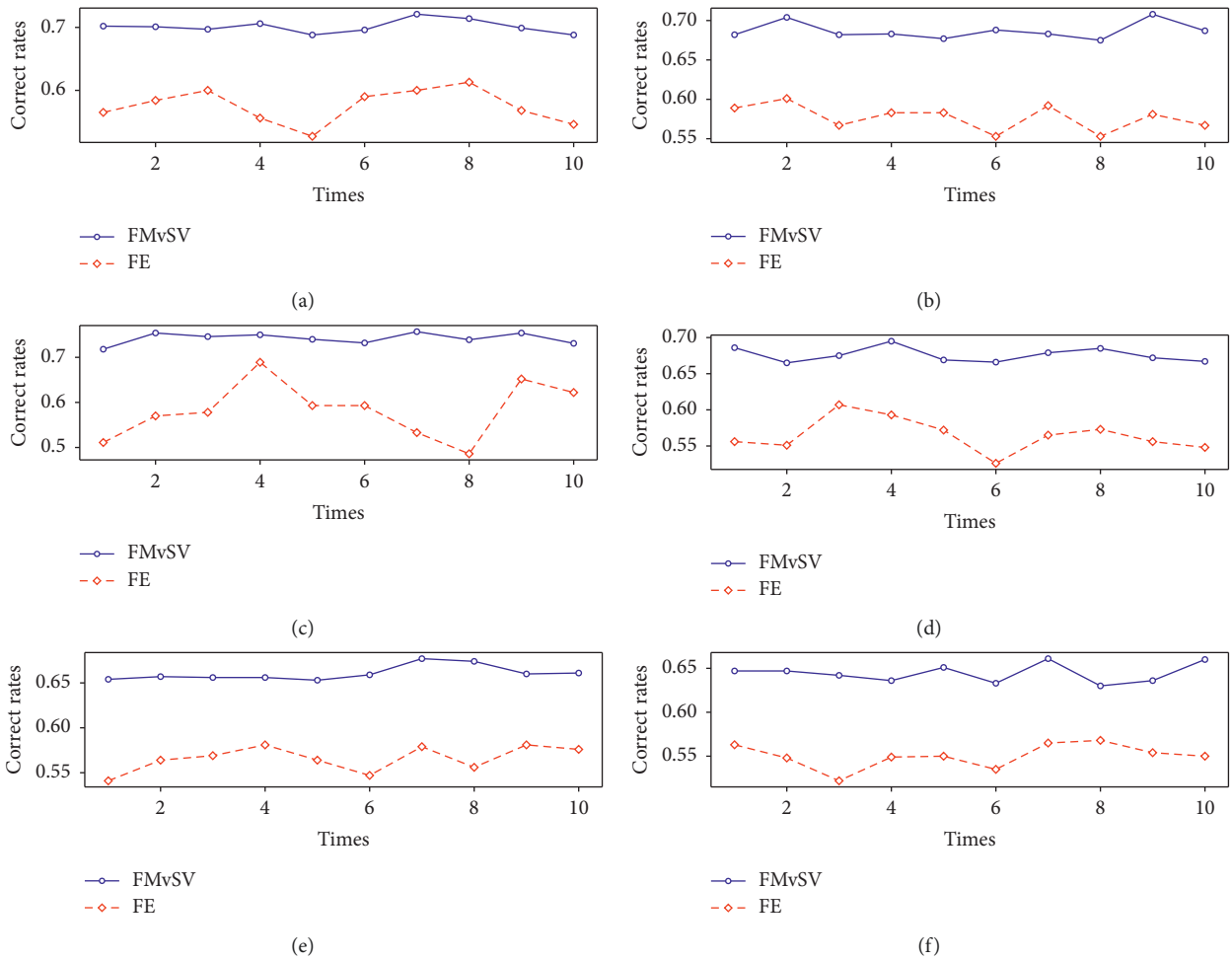

Figure 2: Comparison of the correct rates of Algorithms FMvSV and FE in terms of AD200 with 17.5% – 30% missing values. (a) 17.5% missing values. (b) 20% missing values. (c) 22.5% missing values. (d) 25% missing values. (e) 27.5% missing values. (f) 30% missing values.

Table 10: Comparison of the correct rates of Algorithms FMvSV and FE in terms of AD200 with 2.5% – 30% missing values.

Missing values (%)	2.5	5	7.5	10	12.5	15	17.5	20	22.5	25	27.5	30
FMvSV	0.80	0.78	0.74	0.73	0.72	0.71	0.70	0.69	0.68	0.68	0.66	0.64
FE	0.63	0.59	0.58	0.59	0.58	0.57	0.58	0.58	0.57	0.58	0.57	0.55

as the data size increases, the advantage of Algorithm FMvSV is obvious, that is, Algorithm FMvSV has an advantage in processing big dataset and dynamically increasing dataset.

5.2. Effectiveness Analysis of the Algorithm of Filling Missing Values in IISs of Continuous Data. In this section, we also conduct two groups of experiments. They are still used to compare the effectiveness of algorithms of filling missing values from different points of view. Mean-based filling method (Algorithm MEAN) [31] is a common method of filling missing data for an IIS of continuous data. Next, a comparison of the proposed methods with Algorithm MEAN is provided.

In Section 4.2, we obtain the method of filling the missing values with real values. By combining Section 4.2.1 and Section 4.2.2, we design Algorithm FMvRV to fill the missing values with real values (Algorithm 4).

Next, we provide a comparative study of the effectiveness for Algorithms FMvRV and MEAN. We first give a quantitative index of the effectiveness for filling missing values as follows.

Definition 7. Given a complete information system (U, Att) of continuous data, suppose that the values $a_{k_1}(x_{w_1})$, $a_{k_2}(x_{w_2}), \ldots, a_{k_q}(x_{w_q})$ are missing, and the filling set values are denoted as $a^R_{k_1}(x_{w_1}), a^R_{k_2}(x_{w_2}), \ldots, a^R_{k_q}(x_{w_q})$, respectively. Then, the correct rate of filling values is defined by

$$\mathrm{CRC} = \frac{\left|\left\{(k_i, w_i)\right| \left|a_{k_i}\left(x_{w_i}\right) - a^R_{k_i}\left(x_{w_i}\right)\right| \le \lambda_{k_i}, i = 1, 2, \ldots, q\right\}\right|}{q}, \tag{34}$$

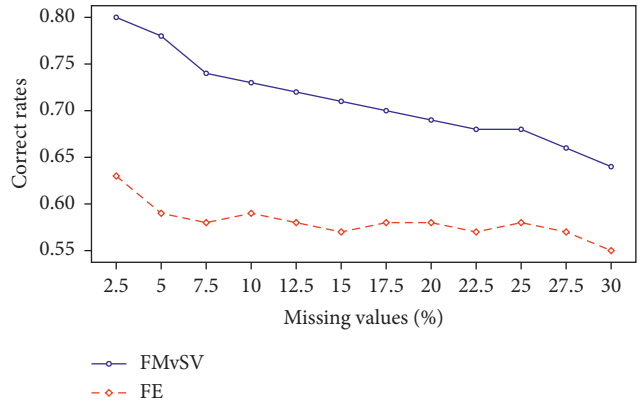

Figure 3: Comparison of correct rates of Algorithm FMvSV and Algorithm FE.

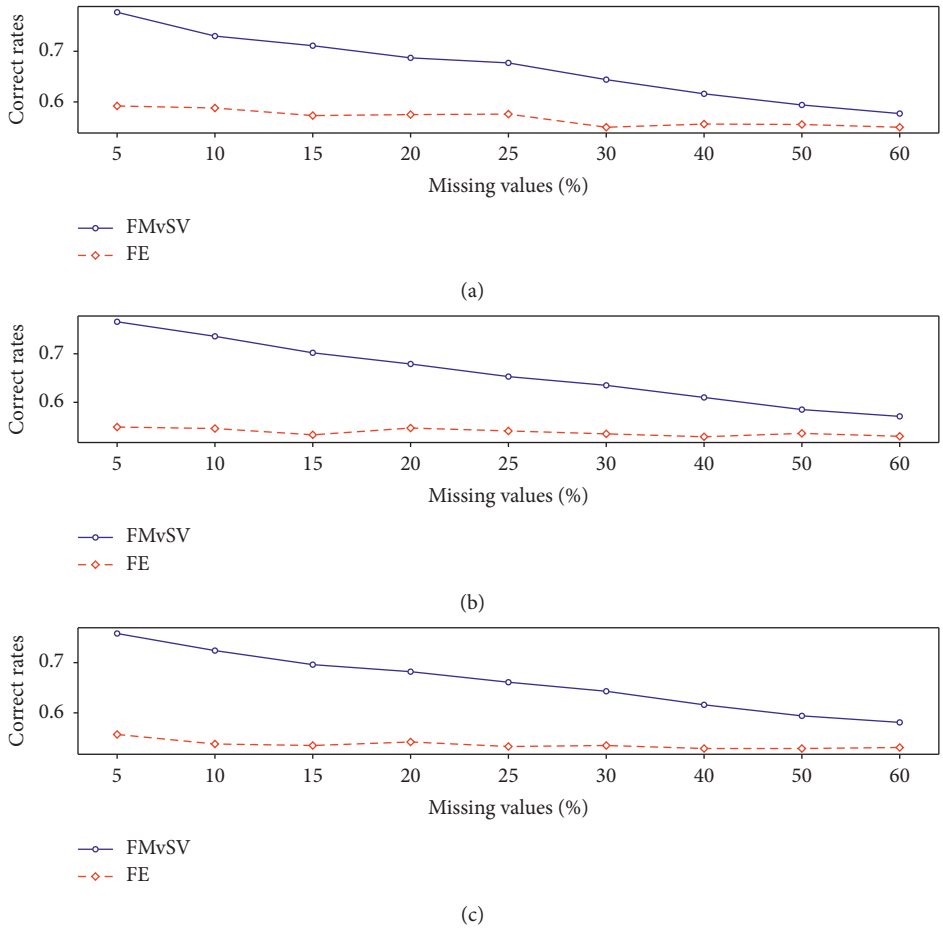

Figure 4: Comparison of the correct rates of Algorithms FMvSV and FE in terms of (a) AD200, (b) AD400, and (c) AD800.

where q is the number of the missing values, and $\lambda_1, \lambda_2, \ldots,$ and λ_q are thresholds corresponding to a_{k_1}, a_{k_2}, $\ldots,$ and a_{k_q}.

Example 7. From Example 5, we know that the thresholds are $\lambda_1 = 0.2, \lambda_2 = \lambda_3 = \lambda_4 = 0.5$, and $a_1^R(x_5) = 5.3$. Similarly, we can compute that $a_2^R(x_7) = 3.15$. It is clear that $|a_1(x_5) - a_1^R(x_5)| = |5.1 - 5.3| = 0.2 \leq \lambda_1 = 0.2$ and $|a_2(x_7) - a_2^R(x_7)| = |2.8 - 3.15| = 0.35 \leq \lambda_2 = 0.5$. Therefore, we can calculate the correct rate: $\mathrm{CRC} = |\{(1,5), (2,7)\}|/2 = 2/2 = 1$.

In this section, we use some subsets of Abalone dataset (see Table 8) to experiment. Table 11 gives three subsets of Abalone dataset.

Experiment 3. The effects of experiment times on correct rates of filling values about continuous data.

In this experiment, we mainly compare the efficiency of Algorithms FMvRV and MEAN by AB200 in Table 11. The steps are as follows:

(i) 5% attribute values are randomly selected from AB200 and supposed that they are missing

(ii) By means of Algorithms FMvRV and MEAN, we can fill these missing values, and we can obtain the correct rate of every algorithm

The steps (i) and (ii) are repeated ten times, and the corresponding results are summarized in Table 12. Similarly, we also consider the cases of 10%, 15%, 20%, 25%, and 30% missing values in AB200. The results are shown in Figure 5.

Figure 5 shows that Algorithm FMvRV is more stable than Algorithm MEAN. Furthermore, the correct rate of Algorithm FMvRV is better than that of Algorithm MEAN. This indicates that Algorithm FMvRV can carry out more accurate forecast of missing values. It is meaningful to explore the correct classification of incomplete datasets.

In Table 12, the mean value of ten correct rates related to Algorithm FMvRV can be viewed as the correct rate of Algorithm FMvRV for 5% missing values. The correct rate of Algorithm MEAN for 5% missing values can be computed similarly. Furthermore, for 10%, 15%, 20%, 25%, and 30% missing values in AB200, we also compute the correct rates of Algorithms FMvRV and MEAN. The results are shown in Figure 6.

Figure 6 shows that the correct rates of Algorithm FMvRV monotone almost decrease with the increase of missing values. However, the monotonicity of the correct rates of Algorithm MEAN is not obvious. In addition, Figure 6 also indicates that the effect of Algorithm FMvRV is better than that of Algorithm MEAN. Furthermore, when the missing values are increased to 30%, the correct rate of Algorithm FMvRV is more than 85%. However, now, the correct rate of Algorithm MEAN is less than 60%. This indicates that Algorithm FMvRV is more conducive to predicting the missing values.

Experiment 4. The effects of data size on correct rates of filling values about continuous data.

In this experiment, we use AB200, AB400, and AB800 to discuss the effects of data size on correct rates of Algorithms FMvRV and MEAN. For 10% – 30% missing values, similar to the calculating method of Table 12, we can obtain the correct rate of Algorithm FMvRV (or MEAN) in this experiment. The results are shown in Figure 7.

Figure 7 reflects the following facts:

(i) When the data size increases, the correct rate of Algorithm FMvRV is higher than that of Algorithm MEAN.

(ii) When the missing values are less than 30%, the correct rates of Algorithm FMvRV are almost unchanged and close to 90%. Now, the data size has little effect on the correct rates of Algorithm FMvRV. This illustrates that Algorithm FMvRV has obvious advantages in processing big dataset when the missing values are less than 30%.

6. Application of the Algorithms of Filling Missing Values in Investigating IISs

When we apply the rough set approach to investigate an IIS, a key step is to induce a binary relation from the IIS. For an IIS, we can provide three ways to obtain a binary relation from the IIS. Let (U, Att, V) be an IIS and $B \subseteq Att$. Then, the three ways are as follows:

(1) By equation (2), we can obtain the binary relation T_B.

(2) By Algorithm 1, we can fill the missing values in IISs with fuzzy set values. Then we can also view the other values of attributes as fuzzy set values, for example, in Table 3 of Example 2, $a_1(x_1) = L$, we can see $a_1(x_1)$ as the fuzzy set value $a_1(x_1) = (1/L) + (0/H) + (0/N)$. Based on this discussion, we can transform an IIS into a FSvIS. Thus, according to equation (8), we can obtain the binary relation $R_B^{S^\lambda}$.

(3) In an IIS of discrete data, if the value $a(x)$ of x under attribute a is not missing, we can view $a(x)$ as a set value $\{a(x)\}$. Based on this consideration, we can use Algorithm FMvSV to transform an IIS into a set-valued information system. Then, we can obtain the following binary relation [32]:

$$T_B^{sv} = \left\{(x, y) \in U^2 \mid a(x) \cap a(y) \neq \varnothing, \quad \forall a \in B\right\}. \quad (35)$$

In this section, through a comparative research on these binary relations induced from the same IIS, we further show that our algorithms are meaningful for the studies of IISs. We choose three datasets, i.e., Mammographic dataset, Abalone dataset, and Car dataset, to carry out the comparative research. The summary information of Mammographic dataset and Abalone dataset is shown in Table 13. The Car dataset is shown in Table 14 [33]. Mammographic dataset and Abalone dataset are taken from UCI (http://archive.ics.uci.edu/ml/datasets.php).

Firstly, we introduce a new measure to evaluate the similarity degree between binary relations.

Definition 8. Let R_1 and R_2 be binary relations on a nonempty set U. The similarity degree of R_1 and R_2 is defined as

$$\mathrm{SD}(R_1, R_2) = \frac{|R_1 \cap R_2|}{|R_1 \cup R_2|}. \quad (36)$$

Example 8. For the car dataset given by Table 14, we can obtain the binary relations T_B, T_B^{sv}, and $R_B^{S_N^\lambda}$ as follows:

Table 11: The subsets of Abalone dataset used in Experiments 3 and 4.

The name of dataset	The type of dataset	The type of attribute value	The number of object	The number of attribute
AB200	Complete	Continuous	200	8
AB400	Complete	Continuous	400	8
AB800	Complete	Continuous	800	8

Table 12: Comparison of the correct rates of Algorithms FMvRV and MEAN in terms of AB200 with 5% missing values.

Times	1	2	3	4	5	6	7	8	9	10
FMvRV	0.925	0.838	0.888	0.925	0.95	0.938	0.925	0.9	0.913	0.925
MEAN	0.575	0.463	0.588	0.613	0.625	0.55	0.563	0.65	0.588	0.6

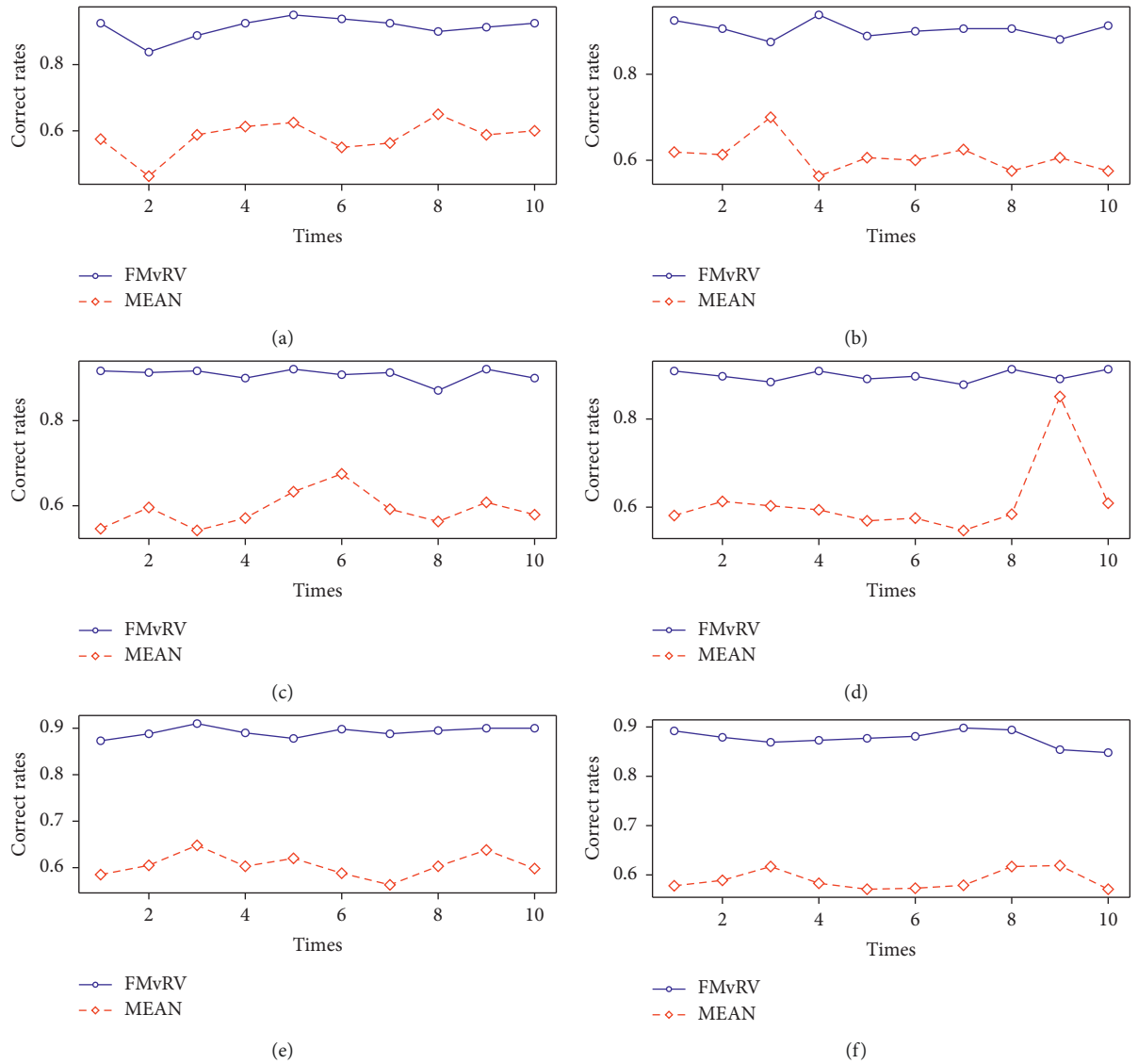

Figure 5: Comparison of the correct rates of Algorithms FMvRV and MEAN in terms of AB200 with 5% – 30% missing values. (a) 5% missing values. (b) 10% missing values. (c) 15% missing values. (d) 20% missing values. (e) 25% missing values. (f) 30% missing values.

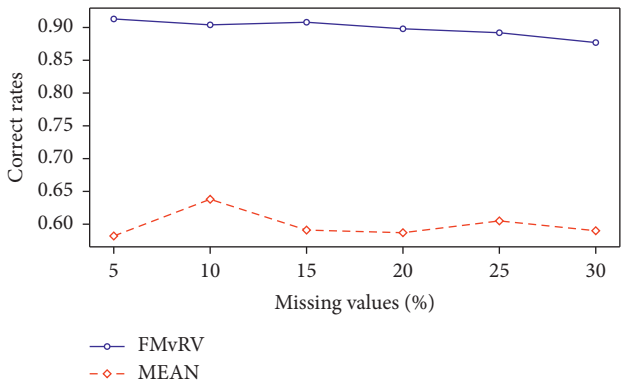

Figure 6: Comparison of correct rates of Algorithm FMvRV and Algorithm MEAN.

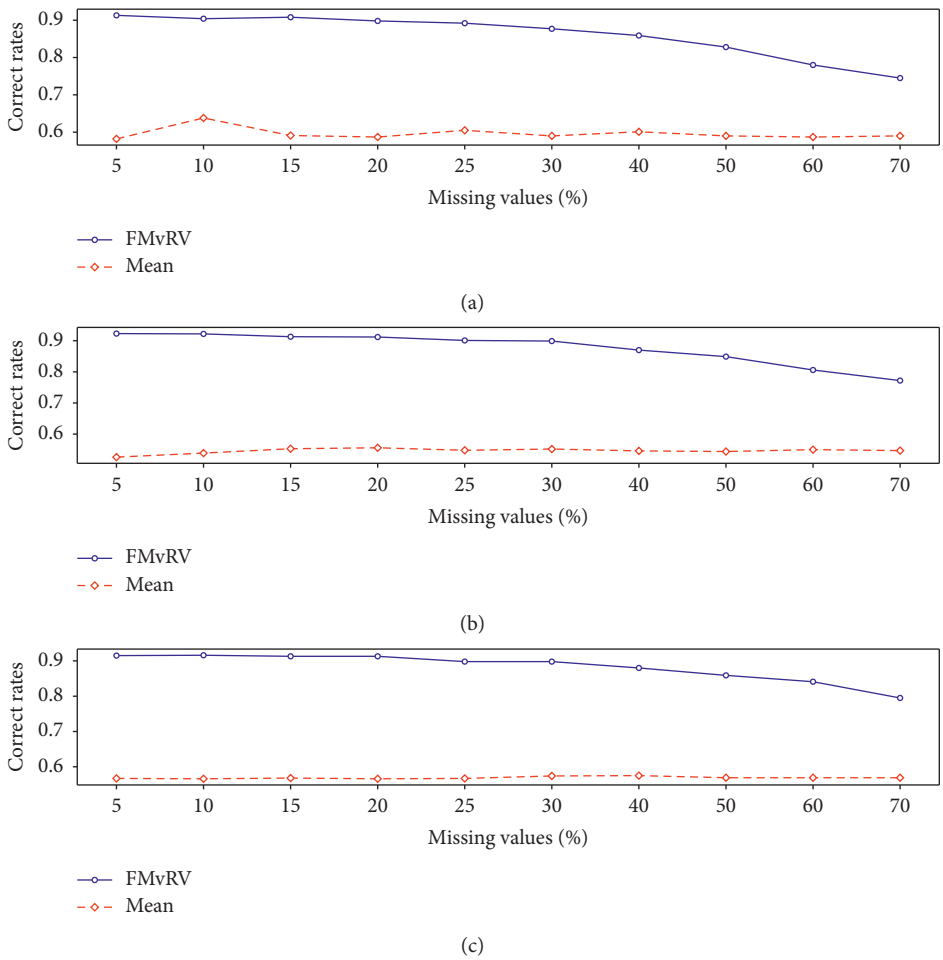

Figure 7: Comparison of the correct rates of Algorithms FMvRV and MEAN in terms of (a) AB200, (b) AB400, and (c) AB800.

Table 13: Detailed information of the datasets.

Index	Dataset	The type of datasets	The type of attribute values	The number of objects	The number of attributes
1	Mammographic	Incomplete	Discrete	961	5
2	Breast cancer	Incomplete	Discrete	286	10

$$
\begin{aligned}
&T_B(x_1)=\{x_1\},\\
&T_B(x_2)=\{x_2,x_6\},\\
&T_B(x_3)=\{x_3\},\\
&T_B(x_4)=\{x_4,x_5\},\\
&T_B(x_5)=\{x_4,x_5,x_6\},\\
&T_B(x_6)=\{x_2,x_5,x_6\};\\
&T_B^{sv}(x_1)=\{x_1\},\\
&T_B^{sv}(x_2)=\{x_2,x_6\},\\
&T_B^{sv}(x_3)=\{x_3\},\\
&T_B^{sv}(x_4)=\{x_4,x_5\},\\
&T_B^{sv}(x_5)=\{x_4,x_5\},\\
&T_B^{sv}(x_6)=\{x_2,x_6\};\\
&R_B^{S_N^\lambda}(x_1)=\{x_1\},\\
&R_B^{S_N^\lambda}(x_2)=\{x_2\},\\
&R_B^{S_N^\lambda}(x_3)=\{x_3\},\\
&R_B^{S_N^\lambda}(x_4)=\{x_4\},\\
&R_B^{S_N^\lambda}(x_5)=\{x_5\},\\
&R_B^{S_N^\lambda}(x_6)=\{x_6\};
\end{aligned}
\tag{37}
$$

where we take $\lambda = 0.6$. Similarly, for every dataset given by Table 13, we can also compute the corresponding binary relations T_B, T_B^{sv}, and $R_B^{S_N^\lambda}$, where we choose $\lambda = 0.6$. Then, by Definition 8, we calculate the similarity degrees between T_B, T_B^{sv}, and $R_B^{S_N^\lambda}$. The result is shown in Table 15.

Table 15 reflects the following facts.

We know that the binary relation induced by a dataset can be considered as the classification result of objects, where the elements in a successor neighbourhood with respect to the binary relation are a class. In this example, for Breast cancer dataset, the similarity degrees between relations are almost close to 1. This means that missing data have less impact on the classification of Breast cancer datasets. Thus, we may ignore these missing values in addressing this dataset. In contrast, the relations induced by Car dataset have low similarity degrees. This shows that missing values in Car dataset play an important role in the classification of this dataset. A natural question is which relation is better to investigate Car dataset. In Table 15, we can see that the similarity degree between T_B^{sv} and T_B is higher than that between T_B and $R_B^{S_N^\lambda}$. Furthermore, the similarity degree between T_B^{sv} and $R_B^{S_N^\lambda}$ is higher than that between T_B and $R_B^{S_N^\lambda}$. This indicates that T_B^{sv} is a good choice to be used to investigate Car dataset. Note that T_B^{sv} is determined by using Algorithm FMvSV. This illustrates that Algorithm FMvSV is important for the studies of IISs.

At the end of this section, we apply the uncertainty measure to estimate the importance of the proposed algorithm. In Example 8, we list three binary relations T_B, T_B^{sv}, and $R_B^{S_N^\lambda}$ with respect to Car dataset. By Definition 3, we can compute their entropies, which are shown in Table 16.

Table 14: The incomplete information system about car dataset.

	Price	Mileage	Size	Max-speed
x_1	High	Low	Full	Low
x_2	Low	*	Full	Low
x_3	*	*	Compact	Low
x_4	High	*	Full	High
x_5	*	*	Full	High
x_6	Low	High	Full	*

Table 15: Similarity degrees between binary relations T_B, T_B^{sv}, and $R_B^{S_N^\lambda}$.

Dataset	$SD(T_B, R_B^{S_N^\lambda})$	$SD(T_B, T_B^{sv})$	$SD(T_B^{sv}, R_B^{S_N^\lambda})$
Car	0.5	0.83	0.6
Mammographic	0.709	0.998	0.71
Breast cancer	0.933	1	0.933

Table 16: The entropies of T_B, T_B^{sv}, and $R_B^{S_N^\lambda}$.

Dataset	$H(T_B)$	$H(T_B^{sv})$	$H(R_B^{S_N^\lambda})$
Car	1.723	1.918	2.585

We know that entropy can measure the granularity of a binary. Proposition 1 shows that the finer the binary relation is, the higher the entropy of it is. Conversely, if the entropy of the binary relation is high, then the binary relation should be fine. Thus, Table 16 indicates that T_B^{sv} and $R_B^{S_N^\lambda}$ are finer than T_B. That is to say, T_B^{sv} and $R_B^{S_N^\lambda}$ can provide more information for the studies of IISs. According to the above discussion, we know that T_B^{sv} and $R_B^{S_N^\lambda}$ are obtained in terms of the proposed algorithms. This illustrates that the proposed algorithms are useful for investigating IISs.

Finally, a similar discussion can also be made about continuous dataset. We omit it here.

7. Conclusion

This paper established the FSvIS, which is an extension of the PSvIS. By means of the FSvIS, we constructed some algorithms to fill missing values in IISs. We carried out several experiments to analyze the effectiveness of these algorithms. The experiment results indicated that these algorithms are useful to investigate the IISs. There are still many interesting issues worth studying. First, we will further study the relationship between FSvISs and the existing information systems and study the application of FSvISs. Second, we can apply uncertainty measures for fuzzy relations, which are established by [34], to investigate the fuzzy set-valued information system which is defined by this paper. Finally, we will conduct a more comprehensive analysis of the impact of missing values on IISs.

Data Availability

The data used to support the findings of this study are included within the article.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This work was supported by the Foundation of Shanxi Normal University (grant no. 872022).

References

[1] Z. A. Pawlak, "Rough sets," *International Journal of Computer and Information Sciences*, vol. 11, no. 5, pp. 341–356, 1982.

[2] G. Cattaneo and D. Ciucci, "Investigation about time monotonicity of similarity and preclusive rough approximations in incomplete information systems," *Rough Sets and Current Trends in Computing*, vol. 3066, pp. 38–48, 2004.

[3] J. Dai and Q. Xu, "Approximations and uncertainty measures in incomplete information systems," *Information Sciences*, vol. 198, pp. 62–80, 2012.

[4] S. Greco, B. Matarazzo, and R. Słowinski, "Handling missing values in rough set analysis of multi-attribute and multi-criteria decision problems," in *Lecture Notes in Computer Science*, vol. 1711, pp. 146–157, Springer, Berlin, Germany, 1999.

[5] J. W. Grzymala-Busse and W. Rzasa, "Local and global approximations for incomplete data," in *Rough Sets and Current Trends in Computing*, vol. 4259, pp. 244–253, Springer, Berlin, Germany, 2006.

[6] R. Jensen and Q. Shen, "Interval-valued fuzzy-rough feature selection in datasets with missing values," in *Proceedings of 2009 IEEE International Conference on Fuzzy Systems*, pp. 610–615, Jeju Island, South Korea, August 2009.

[7] Z. Meng and Z. Shi, "A fast approach to attribute reduction in incomplete decision systems with tolerance relation-based rough sets," *Information Sciences*, vol. 179, no. 16, pp. 2774–2793, 2009.

[8] Y. Qian, J. Liang, D. Li, F. Wang, and N. Ma, "Approximation reduction in inconsistent incomplete decision tables," *Knowledge-Based Systems*, vol. 23, no. 5, pp. 427–433, 2010.

[9] J. Stefanowski and A. Tsoukias, "Incomplete information tables and rough classification," *Computational Intelligence*, vol. 17, no. 3, pp. 545–566, 2001.

[10] M. Kryszkiewicz, "Rough set approach to incomplete information systems," *Information Sciences*, vol. 177, pp. 41–73, 2007.

[11] Y. Leung and D. Y. Li, "Maximal consistent block technique for rule acquisition in incomplete information systems," *Information Sciences*, vol. 153, pp. 85–106, 2003.

[12] J. Dai, "Rough set approach to incomplete numerical data," *Information Sciences*, vol. 241, pp. 43–57, 2013.

[13] M. Kryszkiewicz, "Rough set approach to incomplete information systems," *Information Sciences*, vol. 112, no. 1–4, pp. 39–49, 1998.

[14] D. Liu, T. Li, and J. Zhang, "A rough set-based incremental approach for learning knowledge in dynamic incomplete information systems," *International Journal of Approximate Reasoning*, vol. 55, no. 8, pp. 1764–1786, 2014.

[15] J. Liang and Z. Xu, "The algorithm on knowledge reduction in incomplete information systems," *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, vol. 10, no. 1, pp. 95–103, 2002.

[16] M.-W. Shao and W.-X. Zhang, "Dominance relation and rules in an incomplete ordered information system," *International Journal of Intelligent Systems*, vol. 20, no. 1, pp. 13–27, 2005.

[17] W. Xu, Y. Li, and X. Liao, "Approaches to attribute reductions based on rough set and matrix computation in inconsistent ordered information systems," *Knowledge-Based Systems*, vol. 27, pp. 78–91, 2012.

[18] J. Yuan, M. Chen, T. Jiang, and T. Li, "Complete tolerance relation based parallel filling for incomplete energy big data," *Knowledge-Based Systems*, vol. 132, pp. 215–225, 2017.

[19] J. Chen and J. Shao, "Jackknife variance estimation for nearest-neighbor imputation," *Journal of the American Statistical Association*, vol. 96, no. 453, pp. 260–269, 2001.

[20] S. Wang, "Classification with incomplete survey data: a Hopfield neural network approach," *Computers and Operations Research*, vol. 32, no. 10, pp. 2583–2594, 2005.

[21] A. S. Salama and O. G. El-Barbary, "Topological approach to retrieve missing values in incomplete information systems," *Journal of the Egyptian Mathematical Society*, vol. 25, no. 4, pp. 419–423, 2017.

[22] Z. Pawlak and A. Skowron, "Rudiments of rough sets," *Information Sciences*, vol. 177, no. 1, pp. 3–27, 2007.

[23] C. Wang, Q. He, M. Shao, Y. Xu, and Q. Hu, "A unified information measure for general binary relations," *Knowledge-Based Systems*, vol. 135, pp. 18–28, 2017.

[24] Y. Yao, "Constructive and algebraic methods of the theory of rough sets," *Information Sciences*, vol. 109, no. 1–4, pp. 21–47, 1998.

[25] L. A. Zadeh, "Fuzzy sets," *Information and Control*, vol. 8, no. 3, pp. 338–353, 1965.

[26] W. Zeng and H. Li, "Inclusion measures, similarity measures, and the fuzziness of fuzzy sets and their relations," *International Journal of Intelligent Systems*, vol. 21, no. 6, pp. 639–653, 2006.

[27] Y. Li, K. Qin, and X. He, "Some new approaches to constructing similarity measures," *Fuzzy Sets and Systems*, vol. 234, pp. 46–60, 2014.

[28] G. Deng, Y. Jiang, and J. Fu, "Monotonic similarity measures between fuzzy sets and their relationship with entropy and inclusion measure," *Fuzzy Sets and Systems*, vol. 287, pp. 97–118, 2016.

[29] Y. Huang, T. Li, C. Luo, H. Fujita, and S.-j. Horng, "Dynamic variable precision rough set approach for probabilistic set-valued information systems," *Knowledge-Based Systems*, vol. 122, pp. 131–147, 2017.

[30] X. Zhu, S. Zhang, Z. Jin, Z. Zhang, and Z. Xu, "Missing value estimation for mixed-attribute data sets," *IEEE Transactions on Knowledge and Data Engineering*, vol. 23, no. 1, pp. 110–121, 2011.

[31] B. Twala, M. Cartwright, and M. Shepperd, "Ensemble of missing data techniques to improve software prediction accuracy," in *Proceedings of the 28th International Conference on Software Engineering*, pp. 909–912, Shanghai, China, May 2006.

[32] Y. Y. Yao, "Information granulation and rough set approximation," *International Journal of Intelligent Systems*, vol. 16, no. 1, pp. 87–104, 2001.

[33] M. Kryszkiewicz, "Rules in incomplete information systems," *Information Sciences*, vol. 113, no. 3-4, pp. 271–292, 1999.

[34] C. Wang, Y. Huang, M. Shao, and D. Chen, "Uncertainty measures for general fuzzy relations," *Fuzzy Sets and Systems*, vol. 360, pp. 82–96, 2019.