symmetry

MDPI

Article

A Novel Attribute Reduction Algorithm for Incomplete Information Systems Based on a Binary Similarity Matrix

Yan Zhou and Yan-Ling Bao *

College of Mathematics and System Sciences, Xinjiang University, Urumqi 830046, China
* Correspondence: bao-yanling@xju.edu.cn

Abstract: With databases growing at an unrelenting rate, it may be difficult and complex to extract statistics by accessing all of the data in many practical problems. Attribute reduction, as an effective method to remove redundant attributes from massive data, has demonstrated its remarkable capability in simplifying information systems. In this paper, we concentrate on reducing attributes in incomplete information systems. We introduce a novel definition of a binary similarity matrix and present a method to calculate the significance of attributes in correspondence. Secondly, We develop a heuristic attribute reduction algorithm using a binary similarity matrix and attribute significance as heuristic knowledge. In addition, we use a numerical example to showcase the practicality and accuracy of the algorithm. In conclusion, we demonstrate through comparative analysis that our algorithm outperforms some existing attribute reduction methods.

Keywords: attribute reduction; binary similarity matrix; attribute significance; incomplete information system

Citation: Zhou, Y.; Bao, Y.-L. A Novel Attribute Reduction Algorithm for Incomplete Information Systems Based on a Binary Similarity Matrix. *Symmetry* **2023**, *15*, 674. https://doi.org/10.3390/sym15030674

Academic Editors: Wenyin Gong and Libao Deng

Received: 8 February 2023
Revised: 24 February 2023
Accepted: 1 March 2023
Published: 7 March 2023

1. Introduction

In the age of information explosion, abundant information rapidly gushes towards people like tidewater, but it is too deficient to adopt traditional knowledge to dispose of the information. How to mine, store and process a large amount of complex data is a critical problem that needs to be solved urgently. Extracting useful data information efficiently from incomplete information systems has become a hot topic in information science and technology areas. Attribute reduction, also known as feature selection, can find out as few attributes as possible to keep the classification of information tables and has been applied to many practical problems [1–7]. Specifically, Chen et al. [8] investigated feature selection by combining multiple feature selection results and pointed out that a combination of filter (i.e., principal component analysis) and wrapper (i.e., genetic algorithms) techniques by the union method is a better choice. Yuan et al. [9] studied mixed attribute reduction to maintain learning ability without decision information based on fuzzy rough sets. Xie et al. [10] considered the weight of each attribute in information systems with the help of a data binning method and information entropy theory, and further designed a novel attribute reduction method by using weighted neighborhood probabilistic rough sets.

So far, there exist two kinds of algorithms for attribute reduction. One is based on a discernibility matrix proposed by Skowron and Rauszer [11]. The central steps involve obtaining a discernibility matrix and transforming the conjunctive normal form to a disjunctive normal form. The attribute reduction can be obtained directly through the disjunctive normal form. Although the attribute reduction algorithm based on discernibility matrix can acquire all reductions simultaneously, storing the discernibility matrix may require a lot of time and storage space, and the transformation from conjunctive normal form to disjunctive normal form may encounter a combinatorial explosion problem [12]. The other is heuristic algorithm, which has low time complexity and is effective for large data sets, but it can obtain only one reduction at one time. For an information system, obtaining the optimal

reduction of attributes has been proved to be an NP-hard problem, which motivates many scholars to devote themselves to heuristic attribute reduction algorithms. There are many types of heuristic knowledge for heuristic attribute reduction algorithms, the most common is attribute significance, which can reflect the importance of an attribute for an information system. Specifically, Shen and Jiang [13] adopted information entropy to calculate attribute significance. Later on, Zhou et al. [14] proposed a method to acquire attribute significance with the help of a similarity matrix. Subsequently, Dai et al. [15] put forward a new conditional entropy and provided its application to attribute reduction. Lately, Hu et al. [16] utilized a dependency relationship of weighted neighborhood rough set (WNRS) model to evaluate attribute significance, and designed a greedy search heuristic algorithm based on WNRS. In addition, the discernibility matrix also behaved well in calculating attribute significance. However, it often requires large storage space during the attribute reduction process. To overcome this defect, many scholars considered a binary discernibility matrix, which is a generalization of the discernibility matrix. Instituto and Nacional [17] defined a more ordered matrix based on the binary discernibility matrix, which allows us to reduce the number of candidate attributes. Ma and Zhang [18] proposed a form of generalized binary discernibility matrix, and further built a new algorithm based on a generalized binary discernibility matrix. Li et al. [19] developed an attribute reduction algorithm based on an improved binary discernibility matrix. Compared with the discernibility matrix, binary discernibility matrix can effectively save storage space and obtain information to calculate attribute significance conveniently. Meanwhile, binary discernibility matrix has the advantages of visual representation and being easy to understand.

As mentioned above, both discernibility matrix and binary discernibility matrix have been applied to attribute reduction algorithms, and binary discernibility matrix outperformed discernibility matrix usually. On the other hand, attribute reduction algorithms based on similarity matrix have been studied systemically. Up to now, there have not been any reduction attribute algorithms based on binary similarity matrix. Therefore, in this paper, we concentrate on proposing a binary similarity matrix and further establish a heuristic attribute reduction algorithm based on the binary similarity matrix for incomplete information systems.

The structure of the paper is as follows. In Section 2, we firstly review some basic definitions about attribute reduction and define binary similarity matrix. In Section 3, we propose a method to calculate attribute significance based on binary similarity matrix and develop an attribute reduction algorithm for incomplete decision information systems. In Section 4, the effectiveness and advantages of the algorithm are demonstrated by some numerical examples. Section 5 concludes the paper.

2. Preliminaries

In this section, we recall some basic definitions about incomplete information systems, tolerance relations and binary similarity matrix.

Definition 1 ([20]). *Suppose (U, A, V, f) is a quadruplet, $U = \{x_1, x_2, \cdots, x_n\}$ is a finite set of objects, $A = \{a_1, a_2, \cdots, a_m\}$ is a set of attributes, f: $U \times A \to V$ is an information function and $V = \cup_{a \in A} V_a$ is a set of values, then $S = (U, A, f, V)$ is called an information system. If there exist $x \in U, a \in A$ such that $f(x, a) = *$, "$*$" represents a missing value, then $S = (U, A, f, V)$ is an incomplete information system.*

Sometimes, $S = (U, A, f, V)$ can be abbreviated as $S = (U, A)$ while there are no confusions. If $A = C \cup D$ and $C \cap D = \emptyset$, then $(U, C \cup D)$ is called a decision information system, where C and D are termed as a condition attribute set and decision attribute set, respectively. In many practical applications, D often contains just one attribute, denoted by $D = \{d\}$.

Definition 2 ([11]). *Suppose $S = (U, A)$ is an incomplete information system, $C \subseteq A$, for any $x_i, x_j \in U$, tolerance relation R_C is defined as:*

$$x_i R_C x_j \iff \forall a \in C, f(x_i, a) = f(x_j, a) \text{ or } f(x_i, a) = * \text{ or } f(x_j, a) = *.$$

It is obvious that R_C is reflexive and symmetric, but not necessarily transitive, and $U/R_C = \{R_C(x_1), R_C(x_2), \cdots, R_C(x_{|U|})\}$ is a cover of U, where $R_C(x_i) = \{x_j | x_i R_C x_j\}(x_i, x_j \in U)$.

Definition 3 ([21]). *In an incomplete decision information system $S = (U, C \cup D)$, $U/R_C = \{R_C(x_1), R_C(x_2), \cdots, R_C(x_{|U|})\}$, $U/R_D = \{R_D(x_1), R_D(x_2), \cdots, R_D(x_{|U|})\}$, then $U_{CD} = \{x | R_C(x) \subseteq R_D(x)\}$ is called the consistent part of S. If $U_{CD} = U$, then S is consistent. Otherwise, S is inconsistent.*

Definition 4 ([22]). *Let $S = (U, C \cup D)$ be an incomplete decision information system, $\emptyset \neq T \subseteq C$. If T satisfies the following conditions*

(1) $U_{TD} = U_{CD}$,

(2) *For any* $\emptyset \neq T^{'} \subset T, U_{T^{'}D} \neq U_{CD}$,

then T is called a reduction of C.

In a decision information system $S = (U, C \cup D)$, the attribute reduction algorithm based on similarity matrix is effective and is simple to implement.

Definition 5 ([14]). *For an incomplete decision information system $S = (U, C \cup \{d\})$, $U = \{x_1, x_2, \cdots, x_n\}$, $C = \{a_1, a_2, \cdots, a_m\}$, the similarity matrix $M = [m(i,j)]_{n \times n}$ $(i, j = 1, 2, \cdots, n)$ is defined as*

$$m(i,j) = \begin{cases} \{a \in C | a(x_i) = a(x_j) \vee a(x_i) = * \vee a(x_j) = *\}, d(x_i) \neq d(x_j) \wedge \min\{|\sigma_C(x_i)|, |\sigma_C(x_j)|\} = 1, \\ \\ \emptyset, \qquad\qquad\qquad\qquad else. \end{cases}$$

where $\sigma_C(x_i) = \{d(y) | y \in R_C(x_i)\}$.

Definition 6 ([14]). *In an incomplete decision information system $S = (U, C \cup \{d\})$, $U = \{x_1, x_2, \cdots, x_n\}$, $C = \{a_1, a_2, \cdots, a_m\}$, the significance of attribute a_k is*

$$SGF(a_k) = -\sum_{i=1}^{n} \sum_{j=1}^{n} \lambda_{ij} / card(m(i,j)),$$

if $a_k (a_k \in C) \in m(i,j)$, then $\lambda_{ij} = 1$, otherwise $\lambda_{ij} = 0$, $card(m(i,j))$ represents the number of attributes contained in $m(i,j)$.

In Definition 5, all elements in the similarity matrix are attribute sets and all objects need to be distinguished, so a large space is needed to store the similarity matrix. To save storage space, we define a binary similarity matrix.

Definition 7. *In an incomplete decision information system $S = (U, C \cup \{d\})$, $U = \{x_1, x_2, \cdots, x_n\}$, $C = \{a_1, a_2, \cdots, a_m\}$. For x_i, $x_j \in U$ $(x_i \neq x_j)$, if $d(x_i) \neq d(x_j)$ and $\min\{|\sigma_C(x_i)|, |\sigma_C(x_j)|\} = 1$, then the binary similarity matrix $BSM = [m((i,j), a_k)](a_k \in C)$ is defined as,*

$$m((i,j), a_k) = \begin{cases} 1, & a_k(x_i) = a_k(x_j) \vee a_k(x_i) = * \vee a_k(x_j) = *, \\ \\ 0, & a_k(x_i) \neq a_k(x_j). \end{cases}$$

where $\sigma_C(x_i) = \{d(y) | y \in R_C(x_i)\}$.

In a binary similarity matrix, each element is represented by "0" or "1", and we just need to consider objects that satisfy the constraints. Therefore, the storage space of the binary similarity matrix is small, and the calculation of binary similarity matrix is simple.

Example 1. *Given an incomplete decision information system* $S = (U, C \cup \{d\})$ *(as shown in Table 1),* $U = \{x_1, x_2, x_3, x_4, x_5, x_6, x_7, x_8\}$, $C = \{a_1, a_2, a_3, a_4, a_5\}$. *The binary similarity matrix can be obtained by the following steps.*

Firstly, it is easy to obtain object pairs that satisfy condition $d(x_i) \neq d(x_j)$*:* (x_1, x_2), (x_1, x_5), (x_1, x_8), (x_2, x_3), (x_2, x_4), (x_2, x_6), (x_2, x_7), (x_3, x_5), (x_3, x_8), (x_4, x_5), (x_4, x_8), (x_5, x_6), (x_5, x_7), (x_6, x_8), (x_7, x_8).

Secondly, it is obvious that $R_C(x_1) = \{x_1\}$, $R_C(x_2) = \{x_2, x_3\}$, $R_C(x_3) = \{x_2, x_3\}$, $R_C(x_4) = \{x_4\}$, $R_C(x_5) = \{x_5, x_6, x_7\}$, $R_C(x_6) = \{x_5, x_6, x_7\}$, $R_C(x_7) = \{x_5, x_6, x_7\}$, $R_C(x_8) = \{x_2, x_3, x_6, x_8\}$. Therefore, $\sigma_C(x_1) = \{1\}$, $\sigma_C(x_2) = \{1, 2\}$, $\sigma_C(x_3) = \{1, 2\}$, $\sigma_C(x_4) = \{1\}$, $\sigma_C(x_5) = \{1, 2\}$, $\sigma_C(x_6) = \{1, 2\}$, $\sigma_C(x_7) = \{1, 2\}$, $\sigma_C(x_8) = \{1, 2\}$.

Finally, we can have the object pairs satisfying $d(x_i) \neq d(x_j)$ *and* $\min\{|\sigma_C(x_i)|, |\sigma_C(x_j)|\} = 1$ *are* (x_1, x_2), (x_1, x_5), (x_2, x_4), (x_4, x_5), (x_1, x_8), (x_4, x_8). *Therefore, a binary similarity matrix is obtained as shown in Table 2.*

Table 1. Incomplete decision table.

	a_1	a_2	a_3	a_4	a_5	d
x_1	1	3	2	0	2	1
x_2	*	1	*	1	0	2
x_3	*	1	*	1	0	1
x_4	1	3	2	1	0	1
x_5	3	*	*	3	1	2
x_6	*	0	0	*	*	1
x_7	3	1	1	3	1	1
x_8	2	*	*	*	*	2

Table 2. Binary similarity matrix.

	a_1	a_2	a_3	a_4	a_5
(x_1, x_2)	1	0	1	0	0
(x_1, x_5)	0	1	1	0	0
(x_2, x_4)	1	0	1	1	1
(x_4, x_5)	0	1	1	0	0
(x_1, x_8)	0	1	1	1	1
(x_4, x_8)	0	1	1	1	1

3. Heuristic Attribute Reduction Algorithm Based on Binary Similar Matrix

In this part, we apply the binary similarity matrix to define attribute significance and then design a heuristic attribute reduction algorithm with attribute significance as heuristic knowledge.

In order to calculate attribute significance conveniently, we add a bottom row to the binary similarity matrix to describe the number of "1" in the column where $a_k (a_k \in C)$ is located, representing the number of object pairs (x_i, x_j) belonging to the same tolerance class under attribute a_k. Take Table 2 as an example, the extended binary similarity matrix (as shown in Table 3) can be acquired easily.

Table 3. Extended binary similarity matrix 1.

	a_1	a_2	a_3	a_4
(x_1, x_2)	1	0	1	0
(x_1, x_5)	0	1	1	0
(x_2, x_4)	1	0	1	1
(x_4, x_5)	0	1	1	0
(x_1, x_8)	0	1	1	1
(x_4, x_8)	0	1	1	1
t	2	4	6	3

Definition 8. *Given an incomplete decision information system $S = (U, C \cup \{d\})$ with binary similarity matrix BSM, the significance attribute of a_k ($a_k \in C$) is defined as $CS(a_k) = -\frac{t_k}{|BSM|}$, where t_k is the number of "1" in the column a_k, and $|BSM|$ is the number of object pairs in the binary similarity matrix.*

In Definition 8, if $|BSM|$ is fixed, then the larger $CS(a_k)$ is, the smaller t_k is, indicating less object pairs in terms of tolerance relation induced by attribute a_k. In addition, the larger $CS(a_k)$ is, it indicates the stronger ability of a_k to distinguish object pairs, that is, the more significant a_k is. Therefore, $CS(a_k)$ directly reflects the significance of a_k.

In order to reduce the time complexity, most attribute reduction algorithms start from core attributes. Next, Theorem 1 shows how to determine the core attributes of an information system according to binary similarity matrix.

Theorem 1. *Given an incomplete decision system $S = (U, C \cup \{d\})$. For objects x_i, $x_j \in U$ $(x_i \neq x_j)$, if there is only one $a_k \in C$ satisfying $m((x_i, x_j), a_k) = 0$, then a_k is a core attribute, i.e., $a_k \in core(C)$.*

Proof. According to Definition 7, we have $d(x_i) \neq d(x_j)$ and $\min\{|\sigma_C(x_i)|, |\sigma_C(x_j)|\} = 1$. As $\min\{|\sigma_C(x_i)|, |\sigma_C(x_j)|\} = 1$, then $|\sigma_C(x_j)| = 1$ or $|\sigma_C(x_i)| = 1$. If $|\sigma_C(x_j)| = 1$, then $R_C(x_j) \subseteq R_{\{d\}}(x_j)$, so $x_j \in U_{CD}$. If $|\sigma_C(x_i)| = 1$, then $R_C(x_i) \subseteq R_{\{d\}}(x_i)$, hence $x_i \in U_{CD}$. Because $d(x_i) \neq d(x_j)$, we have $x_j \notin R_{\{d\}}(x_i)$ and $x_i \notin R_{\{d\}}(x_j)$. In addition, if there is only one $a_k \in C$ such that $m((x_i, x_j), a_k) = 0$, it indicates that objects x_i and x_j satisfy the condition $a_k(x_i) \neq a_k(x_j)$, so $a_l(x_i) = a_l(x_j)(a_l \in (C - \{a_k\})$, and $x_j \notin R_C(x_i)$, $x_i \notin R_C(x_j)$. Let $C - \{a_k\} = E$, obviously, $x_j \in R_E(x_i)$, $x_i \in R_E(x_j)$. Since $U_{ED} = \{x | R_E(x) \subseteq R_{\{d\}}(x)\}$, we have $x_i \notin U_{ED}$ and $x_j \notin U_{ED}$, which implies $U_{ED} \subset U_{CD}$. Therefore, $a_k \in core(C)$. □

According to Theorem 1, it is easy to obtain $core(C) = \{a_1, a_2\}$ in Example 1.

Next, we propose a new attribute reduction algorithm based on binary similarity matrix, which takes attribute significance as heuristic knowledge. The main process is as follows.

The binary similarity matrix in our algorithm needs less storage space compared with some existing similarity matrixes. On the other hand, our attribute reduction method is simpler and more direct than previous similarity matrix-based methods. In addition, in some heuristic attribute reduction algorithms, multiple tests are needed to verify whether a reduct is obtained. In our heuristic attribute reduction algorithm, the sign of the end of the algorithm is that the binary similarity matrix does not contain "1", and multiple tests are not needed. Therefore, the time complexity can be reduced significantly.

4. Numerical Illustration

4.1. Attribute Reduction of Incomplete Decision Tables

To verify the effectiveness of the proposed algorithm, an incomplete decision table (as shown in Table 4) in [14] is adopted. Next, we apply our heuristic algorithm to calculate the attribute reduction of Table 4.

Example 2 ([14]). *Given an incomplete decision table $S = (U, C \cup \{d\})$ (as shown in Table 4), $U = \{x_1, x_2, \cdots, x_{12}\}$, $C = \{a_1, a_2, \cdots, a_8\}$.*

Step 1. Firstly, we apply a Boolean matrix to calculate U_{CD}. According to Table 4, the Boolean relation matrices of $R_{a_1}, R_{a_2}, \cdots, R_{a_8}$, and R_d can be acquired as below.

Table 4. Incomplete decision Table 2.

	a_1	a_2	a_3	a_4	a_5	a_6	a_7	a_8	d
x_1	3	2	1	1	1	0	$*$	$*$	0
x_2	2	3	2	0	$*$	1	3	1	0

Table 4. *Cont*.

	a_1	a_2	a_3	a_4	a_5	a_6	a_7	a_8	d
x_3	2	3	2	0	1	$*$	3	1	1
x_4	$*$	2	$*$	1	$*$	2	0	1	1
x_5	$*$	2	$*$	1	1	2	0	1	1
x_6	2	3	2	1	3	1	$*$	1	1
x_7	3	$*$	$*$	3	1	0	2	$*$	0
x_8	$*$	0	0	$*$	$*$	0	2	0	1
x_9	3	2	1	3	1	1	2	1	1
x_{10}	1	$*$	$*$	$*$	1	0	$*$	0	0
x_{11}	$*$	2	$*$	$*$	1	$*$	0	1	0
x_{12}	3	2	1	$*$	$*$	0	2	3	0

$$M_{R_{a_1}} = \begin{bmatrix} 1&0&0&1&1&0&1&1&1&0&1&1\\ 0&1&1&1&1&1&0&1&0&0&1&0\\ 0&1&1&1&1&1&0&1&0&0&1&0\\ 1&1&1&1&1&1&1&1&1&1&1&1\\ 1&1&1&1&1&1&1&1&1&1&1&1\\ 0&1&1&1&1&1&0&1&0&0&1&0\\ 1&0&0&1&1&0&1&1&1&0&1&1\\ 1&1&1&1&1&1&1&1&1&1&1&1\\ 1&0&0&1&1&0&1&1&1&0&1&1\\ 0&0&0&1&1&0&0&1&0&1&1&0\\ 1&1&1&1&1&1&1&1&1&1&1&1\\ 1&0&0&1&1&0&1&1&1&0&1&1 \end{bmatrix}, M_{R_{a_2}} = \begin{bmatrix} 1&0&0&1&1&0&1&0&1&1&1&1\\ 0&1&1&0&0&1&1&0&0&1&0&0\\ 0&1&1&0&0&1&1&0&0&1&0&0\\ 1&0&0&1&1&0&1&0&1&1&1&1\\ 1&0&0&1&1&0&1&0&1&1&1&1\\ 0&1&1&0&0&1&1&0&0&1&0&0\\ 1&1&1&1&1&1&1&1&1&1&1&1\\ 0&0&0&0&0&0&1&1&0&1&0&0\\ 1&0&0&1&1&0&1&0&1&1&1&1\\ 1&1&1&1&1&1&1&1&1&1&1&1\\ 1&0&0&1&1&0&1&0&1&1&1&1\\ 1&0&0&1&1&0&1&0&1&1&1&1 \end{bmatrix},$$

$$M_{R_{a_3}} = \begin{bmatrix} 1&0&0&1&1&0&1&0&1&1&1&1\\ 0&1&1&1&1&1&1&0&0&1&1&0\\ 0&1&1&1&1&1&1&0&0&1&1&0\\ 1&1&1&1&1&1&1&1&1&1&1&1\\ 1&1&1&1&1&1&1&1&1&1&1&1\\ 0&1&1&1&1&1&1&0&0&1&1&0\\ 1&1&1&1&1&1&1&1&1&1&1&1\\ 0&0&0&1&1&0&1&1&0&1&1&0\\ 1&0&0&1&1&0&1&0&1&1&1&1\\ 1&1&1&1&1&1&1&1&1&1&1&1\\ 1&1&1&1&1&1&1&1&1&1&1&1\\ 1&0&0&1&1&0&1&0&1&0&1&1 \end{bmatrix}, M_{R_{a_4}} = \begin{bmatrix} 1&0&0&1&1&1&0&1&0&1&1&1\\ 0&1&1&0&0&0&0&1&0&1&1&1\\ 0&1&1&0&0&0&0&1&0&1&1&1\\ 1&0&0&1&1&1&0&1&0&1&1&1\\ 1&0&0&1&1&1&0&1&0&1&1&1\\ 1&0&0&1&1&1&0&1&0&1&1&1\\ 0&0&0&0&0&0&1&1&1&1&1&1\\ 1&1&1&1&1&1&1&1&1&1&1&1\\ 0&0&0&0&0&0&1&1&1&1&1&1\\ 1&1&1&1&1&1&1&1&1&1&1&1\\ 1&1&1&1&1&1&1&1&1&1&1&1\\ 1&1&1&1&1&1&1&1&1&1&1&1 \end{bmatrix},$$

$$\vdots \qquad\qquad \vdots$$

$$M_{R_d} = \begin{bmatrix} 1&1&0&0&0&0&1&0&0&1&1&1\\ 1&1&0&0&0&0&1&0&0&1&1&1\\ 0&0&1&1&1&1&0&1&1&0&0&0\\ 0&0&1&1&1&1&0&1&1&0&0&0\\ 0&0&1&1&1&1&0&1&1&0&0&0\\ 0&0&1&1&1&1&0&1&1&0&0&0\\ 1&1&0&0&0&0&1&0&0&1&1&1\\ 0&0&1&1&1&1&0&1&1&0&0&0\\ 0&0&1&1&1&1&0&1&1&0&0&0\\ 1&1&0&0&0&0&1&0&0&1&1&1\\ 1&1&0&0&0&0&1&0&0&1&1&1\\ 1&1&0&0&0&0&1&0&0&1&1&1 \end{bmatrix}, \bigwedge_{i=1}^{8} M_{R_{a_i}} = \begin{bmatrix} 1&0&0&0&0&0&0&0&0&0&1&1\\ 0&1&1&0&0&0&0&0&0&0&0&0\\ 0&1&1&0&0&0&0&0&0&0&0&0\\ 0&0&0&1&1&0&0&0&0&0&1&0\\ 0&0&0&1&1&0&0&0&0&0&1&0\\ 0&0&0&0&0&1&0&0&0&0&0&0\\ 0&0&0&0&0&0&1&1&0&0&0&1\\ 0&0&0&0&0&0&1&1&0&1&0&0\\ 0&0&0&0&0&0&0&0&1&0&0&0\\ 0&0&0&0&0&0&0&1&0&1&0&0\\ 1&0&0&1&1&0&0&0&0&0&1&0\\ 1&0&0&0&0&0&1&0&0&0&0&1 \end{bmatrix}.$$

According to matrix $\bigwedge_{i=1}^{8} M_{R_{a_i}}$, it is obvious that $R_C(x_1) = \{x_1, x_{11}, x_{12}\}$, $R_C(x_2) = \{x_2, x_3\}$, $R_C(x_3) = \{x_2, x_3\}$, $R_C(x_4) = \{x_4, x_5, x_{11}\}$, $R_C(x_5) = \{x_4, x_5, x_{11}\}$, $R_C(x_6) = \{x_6\}$, $R_C(x_7) = \{x_7, x_8, x_{12}\}$, $R_C(x_8) = \{x_7, x_8, x_{10}\}$, $R_C(x_9) = \{x_9\}$, $R_C(x_{10}) = \{x_8, x_{10}\}$, $R_C(x_{11}) = \{x_1, x_4, x_5, x_{11}\}$, $R_C(x_{12}) = \{x_1, x_7, x_{12}\}$.

According to matrix M_{R_d}, we have $R_{\{d\}}(x_1) = \{x_1, x_2, x_7, x_{10}, x_{11}, x_{12}\}$, $R_{\{d\}}(x_2) = \{x_1, x_2, x_7, x_{10}, x_{11}, x_{12}\}$, $R_{\{d\}}(x_3) = \{x_3, x_4, x_5, x_6, x_8, x_9\}$, $R_{\{d\}}(x_4) = \{x_3, x_4, x_5, x_6, x_8, x_9\}$, $R_{\{d\}}(x_5) = \{x_3, x_4, x_5, x_6, x_8, x_9\}$, $R_{\{d\}}(x_6) = \{x_3, x_4, x_5, x_6, x_8, x_9\}$, $R_{\{d\}}(x_7) = \{x_1, x_2, x_7, x_{10}, x_{11}, x_{12}\}$, $R_{\{d\}}(x_8) = \{x_3, x_4, x_5, x_6, x_8, x_9\}$, $R_{\{d\}}(x_9) = \{x_3, x_4, x_5, x_6, x_8, x_9\}$, $R_{\{d\}}(x_{10}) = \{x_1, x_2, x_7, x_{10}, x_{11}, x_{12}\}$, $R_{\{d\}}(x_{11}) = \{x_1, x_2, x_7, x_{10}, x_{11}, x_{12}\}$, $R_{\{d\}}(x_{12}) = \{x_1, x_2, x_7, x_{10}, x_{11}, x_{12}\}$, then $U_{CD} = \{x | R_C(x) \subseteq R_{\{d\}}(x)\} = \{x_1, x_6, x_9, x_{12}\}$.

Secondly, it is easy to obtain $\sigma_C(x_1) = \{0\}$, $\sigma_C(x_2) = \{0, 1\}$, $\sigma_C(x_3) = \{0, 1\}$, $\sigma_C(x_4) = \{0, 1\}$, $\sigma_C(x_5) = \{0, 1\}$, $\sigma_C(x_6) = \{1\}$, $\sigma_C(x_7) = \{0, 1\}$, $\sigma_C(x_8) = \{0, 1\}$, $\sigma_C(x_9) = \{1\}$, $\sigma_C(x_{10}) = \{0, 1\}$, $\sigma_C(x_{11}) = \{0, 1\}$, $\sigma_C(x_{12}) = \{0\}$.

Finally, according to Definition 7, we can obtain a binary similarity matrix (as shown in Table 5).

Step 2. According to Theorem 1, we can obtain $core(C) = \{a_4, a_6, a_7\}$.

Step 3. Test whether $core(C)$ is a reduction of C. Similar to the method of calculating U_{CD}, $U_{core(C)D} = \{x_1, x_9, x_{12}\} \neq U_{CD} = \{x_1, x_6, x_9, and x_{12}\}$. Thus, we delete the columns where attributes a_4, a_6 and a_7 are located in Table 5 to obtain Table 6. Initialization: $DE = \emptyset$.

Step 4. According to Table 6, we have $|BSM| = 20$. Calculate the CS of attributes in Table 6: $CS(a_1) = -\frac{t_1}{|BSM|} = -\frac{12}{20}$, $CS(a_2) = -\frac{t_2}{|BSM|} = -\frac{12}{20}$, $CS(a_3) = -\frac{t_3}{|BSM|} = -\frac{13}{20}$, $CS(a_5) = -\frac{t_5}{|BSM|} = -\frac{16}{20}$, $CS(a_8) = -\frac{t_8}{|BSM|} = -\frac{12}{20}$.

Step 5. Obviously, CS of a_5 is the smallest, so delete the column where a_5 is located and the row with value "1" in this column, and obtain Table 7, then $DE = \{a_5\}$. Because Table 7 has "1", go to Step 4. Calculate the CS of attributes in Table 7: $CS(a_1) = -\frac{1}{4}$, $CS(a_2) = -\frac{2}{4}$, $CS(a_3) = -\frac{3}{4}$, $CS(a_8) = -\frac{3}{4}$. It is obvious that CS of a_3 and a_8 are the smallest. If we delete the column where a_3 is located and the row with value "1" in that column to obtain Table 8, then $DE = \{a_5, a_3\}$. Table 8 has "1", then go to Step 4 again. Perform Step 4 again. Calculate the CS of attributes in Table 8: $CS(a_1) = 0$, $CS(a_2) = 0$, $CS(a_8) = -1$. Then, delete the column where a_8 is located and the row with value "1" in that column, and we have $DE = \{a_5, a_3, a_8\}$. BSM does not have "1", so then go to Step 6.

Table 5. Extended binary similarity matrix 2.

	a_1	a_2	a_3	a_4	a_5	a_6	a_7	a_8
(x_1, x_3)	0	0	0	0	1	1	1	1
(x_1, x_4)	1	1	1	1	1	0	1	1
(x_1, x_5)	1	1	1	1	1	0	1	1
(x_1, x_6)	0	0	0	1	0	0	1	1
(x_1, x_8)	1	0	0	1	1	1	1	1
(x_1, x_9)	1	1	1	0	1	0	1	1
(x_2, x_6)	1	1	1	0	1	1	1	1
(x_2, x_9)	0	0	0	0	1	1	0	1
(x_3, x_{12})	0	0	0	1	1	1	0	0
(x_4, x_{12})	1	1	1	1	1	0	0	0
(x_5, x_{12})	1	1	1	1	1	0	0	0
(x_6, x_7)	0	1	1	0	0	0	1	1
(x_6, x_{10})	0	1	1	1	0	0	1	0
(x_6, x_{11})	1	0	1	1	0	1	1	1
(x_6, x_{12})	0	0	0	1	1	0	1	0
(x_7, x_9)	1	1	1	1	1	0	1	1

Table 5. *Cont*.

	a_1	a_2	a_3	a_4	a_5	a_6	a_7	a_8
(x_8, x_{12})	1	0	0	1	1	1	1	0
(x_9, x_{10})	0	1	1	1	1	0	1	0
(x_9, x_{11})	1	1	1	1	1	1	0	1
(x_9, x_{12})	1	1	1	1	1	0	1	0
t	12	12	13	15	16	7	15	12

Table 6. Extended binary similarity matrix 3.

	a_1	a_2	a_3	a_5	a_8
(x_1, x_3)	0	0	0	1	1
(x_1, x_4)	1	1	1	1	1
(x_1, x_5)	1	1	1	1	1
(x_1, x_6)	0	0	0	0	1
(x_1, x_8)	1	0	0	1	1
(x_1, x_9)	1	1	1	1	1
(x_2, x_6)	1	1	1	1	1
(x_2, x_9)	0	0	0	1	1
(x_3, x_{12})	0	0	0	1	0
(x_4, x_{12})	1	1	1	1	0
(x_5, x_{12})	1	1	1	1	0
(x_6, x_7)	0	1	1	0	1
(x_6, x_{10})	0	1	1	0	0
(x_6, x_{11})	1	0	1	0	1
(x_6, x_{12})	0	0	0	1	0
(x_7, x_9)	1	1	1	1	1
(x_8, x_{12})	1	0	0	1	0
(x_9, x_{10})	0	1	1	1	0
(x_9, x_{11})	1	1	1	1	1
(x_9, x_{12})	1	1	1	1	0
t	12	12	13	16	12

Table 7. Extended binary similarity matrix 4.

	a_1	a_2	a_3	a_8
(x_1, x_6)	0	0	0	1
(x_6, x_7)	0	1	1	1
(x_6, x_{10})	0	1	1	0
(x_6, x_{11})	1	0	1	1
t	1	2	3	3

Table 8. Extended binary similarity matrix 5.

	a_1	a_2	a_8
(x_1, x_6)	0	0	1
t	0	0	1

Step 6. $T = C - DE = \{a_1, a_2, a_4, a_6, a_7\}$.

In Table 7, select the column where a_8 is located and the row with value "1" in that column to obtain Table 9. Then, according to Table 9, delete the column where a_2 is located and the row with value "1" in that column, and we can obtain $T = \{a_1, a_3, a_4, a_6, a_7\}$, then delete a_3 and in the last row, we also obtain $T = \{a_1, a_2, a_4, a_6, a_7\}$.

Table 9. Extended binary similarity matrix 6.

	a_1	a_2	a_3
(x_6, x_{10})	0	1	1
t	0	1	1

In order to verify the effectiveness of our algorithm, it is necessary to test whether T is a reduction of C. When $T = \{a_1, a_3, a_4, a_6, a_7\}$, $U_{TD} = \{x_1, x_9, x_{12}\} \neq U_{CD}$. When $T = \{a_1, a_2, a_4, a_6, a_7\}$, we have $U_{TD} = \{x_1, x_6, x_9, x_{12}\} = U_{CD}$, and there are no $\emptyset \neq T^{'} \subset T$ such that $U_{T^{'}D} = U_{CD}$. Thus, $T = \{a_1, a_2, a_4, a_6, a_7\}$ is a reduction of attributes set C.

Moreover, the reduction result in reference [14] is $T = \{a_1, a_3, a_4, a_5, a_6, a_8\}$. It is obvious that our result contains five attributes, and their result contains six attributes. What is more, it can be proved that our result is also a reduction of C from the point of literature [14]. The purpose of attribute reduction is to classify objects with as few attributes as possible. From the discussion above, our algorithm is not only effective but also efficient.

4.2. Attribute Reduction of Complete Decision Tables

Since complete decision systems are a special case of incomplete decision systems, a decent attribute reduction algorithm for incomplete decision systems should be also available for complete decision systems. In the following, we demonstrate the effectiveness of the proposed attribute reduction algorithm for complete decision systems.

In a complete decision system $S = (U, A)$, $C \subseteq A$, for any $x_i, x_j \in U$, the equivalence relation R_C is: $x_i R_C x_j \iff \forall a \in C, f(x_i, a) = f(x_j, a)$ [22]. According to Definition 2 and Definition 7, it is not difficult to find that when a decision system is complete, the construction of the binary similarity matrix is the same as that of the incomplete decision table. Next, we give the other steps of Algorithm 1 to obtain the reduction of Table 10 cited from the literature [22].

Step 1. According to Definition 3, it is easy to obtain $U_{CD} = \{x_1, x_2, x_4\}$, and according to Definition 7, the binary similarity matrix is obtained (as shown in Table 11).

Step 2. According to Theorem 1, it is not difficult to obtain $core(C) = \{a_3\}$ from Table 11.

Step 3. Test whether $core(C)$ is a reduction of C. Similar to the method of calculating U_{CD}, $U_{core(C)D} = \emptyset \neq U_{CD} = \{x_1, x_2, x_4\}$. Thus, we delete the column where attribute a_3 is located in Table 11 to obtain Table 12. Initialization: $DE = \emptyset$.

Algorithm 1 Heuristic attribute reduction algorithm based on binary similarity matrix.

Input: An incomplete decision information system $S = (U, C \cup \{d\})$.

Output: A reduction T of C.

Step 1: Calculate U_{CD} and the binary similarity matrix BSM of S.

Step 2: Determine $core(C)$ according to Theorem 1.

Step 3: If $U_{core(C)D} = U_{CD}$, then $T = core(C)$, the algorithm ends. Otherwise, delete the columns where core attributes are located to obtain a new BSM. Initialization: $DE = \emptyset$.

Step 4: Calculate the CS of attributes in the latest BSM.

Step 5: Add attribute a_k with the smallest CS to DE, delete the column where a_k is located and the row with value '1' in the column. If there is no '1' in BSM, go to Step 6, otherwise, go to Step 4.

Step 6: $T = C - DE$, and test whether T is a reduction of attributes. The algorithm ends.

Table 10. Binary similarity matrix 2.

	a_1	a_2	a_3	a_4	d
x_1	0	1	1	1	2
x_2	1	0	0	0	1
x_3	1	0	1	1	1
x_4	1	0	0	1	1
x_5	1	0	1	1	2

Table 11. Extended binary similarity matrix 7.

	a_1	a_2	a_3	a_4
(x_1, x_2)	0	0	0	0
(x_2, x_3)	0	0	1	1
(x_1, x_4)	0	0	0	1
(x_2, x_5)	1	1	0	0
(x_4, x_5)	1	1	0	1
t	2	2	1	3

Table 12. Extended binary similarity matrix 8.

	a_1	a_2	a_4
(x_1, x_2)	0	0	0
(x_2, x_3)	0	0	1
(x_1, x_4)	0	0	1
(x_2, x_5)	1	1	0
(x_4, x_5)	1	1	1
t	2	2	3

Step 4. According to Table 12, we have $|BSM| = 5$. Calculate the CS of attributes in Table 12: $CS(a_1) = -\frac{t_1}{|BSM|} = -\frac{2}{5}$, $CS(a_2) = -\frac{t_2}{|BSM|} = -\frac{2}{5}$, $CS(a_4) = -\frac{t_3}{|BSM|} = -\frac{3}{5}$.

Step 5. Obviously, CS of a_4 is the smallest, so delete the column where a_4 is located and the row with value "1" in this column, and obtain Table 13, then $DE = \{a_4\}$. Because Table 13 has "1", go to Step 4. Calculate the CS of attributes in Table 13: $CS(a_1) = -\frac{1}{2}$, $CS(a_2) = -\frac{1}{2}$. From Table 13, it is not difficult to see that regardless of deleting a_1 or a_2, the Table does not have "1", then $DE = \{a_1, a_4\}$ or $DE = \{a_2, a_4\}$.

Table 13. Extended binary similarity matrix 9.

	a_1	a_2
(x_1, x_2)	0	0
(x_2, x_5)	1	1
t	2	2

Step 6. $T = \{a_2, a_3\}$ or $T = \{a_1, a_3\}$. The algorithm ends.

Table 10 is a complete decision table cited from [22], and the attribute reduction result in [22] is $T = \{a_2, a_3\}$ or $T = \{a_1, a_3\}$. Our result is the same. So our method is effective. The method in [22] is based on the discernibility matrix. Although all the reductions can be obtained, their method may have a combination explosion problem for large data sets.

4.3. Attribute Reduction of Fuzzy Numerical Decision Tables

The method proposed in this paper is also applicable to fuzzy numerical decision tables (such as Table 14). For a fuzzy decision information system, we can not directly

classify the objects according to the information. Liu et al. [22] introduced a threshold as follows,

$$TH = (max_{i=1}^{|U|} V_a(x_i) - min_{i=1}^{|U|} V_a(x_i)) \times 10\%,$$

where $max_{i=1}^{|U|} V_a(x_i)$ is the maximum attribute value of V_a, $min_{i=1}^{|U|} V_a(x_i)$ is the minimum attribute value of V_a. A binary relation R is defined as: $x_i R x_j$ if and only if $|V_a(x_i) - V_a(x_j)| < TH$. It is easy to check that R is reflexive and symmetric, but not transitive.

Table 14. Fuzzy numerical decision table.

	a_1	a_2	a_3	a_4	a_5	d
x_1	0.9871	0.4352	0.1987	0.8765	0.5436	0.3656
x_2	0.1131	0.5634	0.2134	0.8643	0.6578	0.3452
x_3	0.8675	0.5897	0.3101	0.8321	0.5784	0.3432
x_4	0.4563	0.6235	0.2567	0.8107	0.6785	0.3672
x_5	0.4786	0.4356	0.2675	0.7896	0.7012	0.3213
x_6	0.5342	0.4765	0.2834	0.8876	0.6132	0.3425
x_7	0.8765	0.6123	0.2457	0.8654	0.5324	0.3531
x_8	1	0.5471	0.3721	0.8743	0.5452	0.3456

After classifying objects according to the method above, the corresponding binary similarity matrix can be constructed according to Definition 7, and then the attribute reduction of the decision table can be obtained according to the steps of Algorithm 1.

5. Conclusions

In this paper, we introduce the definition of binary similarity matrix in incomplete decision information systems and define the attribute significance based on binary similarity matrix. What is more, a heuristic attribute reduction algorithm with attribute significance as heuristic knowledge is developed. It has been demonstrated that the algorithm is not only effective, but also efficient. In addition, it can be applied to consistent as well as inconsistent decision information systems. More importantly, the algorithm presented in this paper can hopefully provide more penetration into attribute reduction. In many cases, the data in decision tables may be affected by uncertainty and imprecise factors. In the future, we will apply a binary similarity matrix to hesitant fuzzy numerical decision tables to deal with fuzzy pattern recognition problems.

Author Contributions: Methodology, Y.-L.B.; writing–original draft preparation, Y.-L.B. and Y.Z.; writing–review and editing, Y.-L.B.; supervision, Y.-L.B. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the Scientific Research Program of the Higher Education Institution of Xinjiang (No. XJEDU2022P008).

Data Availability Statement: No new data were created or analyzed in this study. Data sharing is not applicable to this article.

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

References

1. Akram, M.; Ali, G.; Alcantud, J.C.R. Attributes reduction algorithms for m-polar fuzzy relation decision systems. *Int. J. Approx. Reason.* **2022**, *140*, 232–254. [CrossRef]
2. Al-juboori, A.M.; Alsaeedi, A.H.; Nuiaa, R.R.; Alyasseri, Z.A.A.; Sani, N.S.; Hadi, S.M.; Mohammed, H.J.; Musawi, B.A.; Amin, M.M. A hybrid cracked tiers detection system based on adaptive correlation features selection and deep belief neural networks. *Symmetry* **2023**, *15*, 358. [CrossRef]
3. He, J.L.; Qu, L.D.; Wang, Z.H.; Chen, Y.Y.; Luo, D.M. Attribute reduction in an incomplete categorical decision information system based on fuzzy rough sets. *Artif. Intell. Rev.* **2022**, *55*, 5313–5348. [CrossRef]
4. Hu, M.; Tsang, E.C.; Guo, Y.T.; Xu, W.H. Fast and robust attribute reduction based on the separability in fuzzy decision systems. *IEEE Trans. Cybern.* **2021**, *52*, 5559–5572. [CrossRef] [PubMed]

5. Kalaivanan, K.; Vellingiri, J. Normalized hellinger feature selection and soft margin boosting classification for water quality prediction. *Expert Syst.* **2022**. [CrossRef]
6. Singh, H.; Singh, B.; Kaur, M. An efficient feature selection method based on improved elephant herding optimization to classify high-dimensional biomedical data. *Expert Syst.* **2022**, *39*, 13038. [CrossRef]
7. Yang, Q.W.; Gao, Y.L.; Song, Y.J. A tent lévy flying sparrow search algorithm for wrapper-based feature selection: A COVID-19 case study. *Symmetry* **2023**, *15*, 316. [CrossRef]
8. Chen, C.W.; Tsai, Y.H.; Chang, F.R.; Lin, W.C. Ensemble feature selection in medical datasets: Combining filter, wrapper, and embedded feature selection results. *Expert Syst.* **2020**, *37*, 12553. [CrossRef]
9. Yuan, Z.; Chen, H.M.; Li, T.R.; Yu, Z.; Sang, B.B.; Luo, C. Unsupervised attribute reduction for mixed data based on fuzzy rough sets. *Inf. Sci.* **2021**, *572*, 67–87. [CrossRef]
10. Xie, J.J.; Hu, B.Q.; Jiang, H.B. A novel method to attribute reduction based on weighted neighborhood probabilistic rough sets. *Int. J. Approx. Reason.* **2022**, *144*, 1–17. [CrossRef]
11. Skowron, A.; Rauszer, C. The discernibility matrices and functions in information systems. *Intell. Decis. Support* **1992**, *21*, 331–362.
12. Liu, G.L. Attribute reduction algorithms determined by invariants for decision tables. *Cogn. Comput.* **2022**, *14*, 1818–1825. [CrossRef]
13. Shen, Q.; Jiang, Y. Attribute reduction of multi-valued information system based on conditional information entropy. In Proceedings of the 2008 International Conference on Granular Computing, Hangzhou, China, 26–28 August 2008; pp. 562–565.
14. Zhou, J.; E, X.; Li, Y.H.; Wang, Z.; Liu, Z.X.; Bai, X.Y.; Huang, X.Y. A new attribute reduction algorithm dealing with the incomplete information system. In Proceedings of the International Conference on Cyber-Enabled Distributed Computing and Knowledge Discovery, Zhangjiajie, China, 10–11 October 2009; pp. 12–19.
15. Dai, J.H.; Wang, W.T.; Tian, H.W.; Liu, L. Attribute selection based on a new conditional entropy for incomplete decision systems. *Knowl. Based Syst.* **2013**, *39*, 207–213. [CrossRef]
16. Hu, M.; Tsang, E.C.; Guo, Y.T.; Chen, D.G.; Xu, W.H. A novel approach to attribute reduction based on weighted neighborhood rough sets. *Knowl. Based Syst.* **2021**, *220*, 106908. [CrossRef]
17. Instituto; Nacional. A new algorithm for computing reducts based on the binary discernibility matrix. *Intell. Data Anal.* **2016**, *20*, 317–337. [CrossRef]
18. Ma, F.M.; Zhang, T.F. Generalized binary discernibility matrix for attribute reduction in incomplete information systems. *J. China Univ. Posts Telecommun.* **2017**, *4*, 57–75.
19. Li, J.; Wang, X.; Fan, X.W. Improved binary discernibility matrix attribute reduction algorithm in customer relationship management. *Procedia Eng.* **2019**, *7*, 473–476. [CrossRef]
20. Qian, Y.H.; Liang, J.Y.; Li, D.Y.; Wang, F.; Ma, N.N. Approximation reduction in inconsistent incomplete decision tables. *Knowl. Based Syst.* **2010**, *23*, 427–433. [CrossRef]
21. Liu, G.L.; Hua, Z.; Chen, Z.H. A general reduction algorithm for relation decision systems and its application. *Knowl. Based Syst.* **2017**, *119*, 87–93. [CrossRef]
22. Liu, G.L.; Li, L.; Yang, J.T.; Feng, Y.B.; Zhu, K. Attribute reduction approaches for general relation decision systems. *Pattern Recognit. Lett.* **2015**, *65*, 81–87. [CrossRef]