ISSN: 2664-9640

Incremental Attribute Reduction Using Self-information based on Improved Generalized Decision Approximation for Dynamic Ordered Decision Systems

Bo Zou*

College of Computer and Information Science, Chongqing Normal University, Chongqing 401331, P.R. China

*Corresponding Author

Abstract

In the era of big data, massive amounts of ordered data exist in all walks of life. Incremental attribute reduction of ordered data has become a widely studied topic, which can identify essential attributes and reduce the dimensionality of attributes, as well as improve the classification ability of learning models. However, most of the existing incremental attribute reduction methods can only handle ordered decision system (ODS) with one-dimensional variations, and the application scenarios are relatively ideal. Therefore, in response to this situation, this paper introduces an incremental attribute reduction method based on the dominance-based rough set approach (DRSA) to effectively handle multi-dimensional variations of ODS. First, this paper uses self-information as an uncertainty measure, which can simultaneously consider certain and possible classification information to more accurately capture the preference relation between attributes and decisions. Secondly, the matrix calculation process is optimized by processing the dominance-based relation tiangular matrix (tDRM), which greatly saves time. In addition, a more efficient generalized decision (GD) method is introduced to calculate the upper and lower approximation, which further improves the computational efficiency of the algorithm. Finally, the verification on multiple datasets shows that the proposed algorithm can effectively remove redundant attributes and greatly save the reduction time.

Keywords

Dominance-based Rough Set Approach; Ordered Decision System; Self-information; Incremental Attribute Reduction; Multi-dimensional Variations.

1. Introduction

Introduced by Pawlak in 1982, rough set theory (RST) [1] is an important tool in the field of computation and knowledge discovery to deal with inaccurate, inconsistent and uncertain information without any prior knowledge. RST is based on equivalence relations, which divide objects into equivalence granules, as a way of characterizing approximation concepts, and is often used as a basis for attribute reduction due to its advantages. Since its introduction, many different rough set methods as well as attribute reduction methods relying on different rough set theories have been incrementally proposed to refine the theoretical edifice in the field of granular computing as well as to solve different problems in real life [2–8]. Through continuous development, the theories of fuzzy rough sets [9] and neighborhood rough sets [10] have been perfected and widely used in attribute reduction. However, these two types of rough sets do not work well when dealing with ordered datasets. In order to solve this problem, Greco proposed a DRSA specifically for dealing with the case of datasets with preferences. This method utilizes the concept of dominance relation, which can better handle ordered data and

ISSN: 2664-9640

improve the efficiency and accuracy of processing preferred datasets. Meanwhile, attribute reduction methods based on the DRSA method and its extended model have begun to receive more attention, and their emergence has brought new ideas and methods to the field of data analysis, providing more possibilities for data mining and knowledge discovery. Hu et al. delved into an extended version of DRSA, known as the fuzzy preference based rough set model, to handle numerical ordered data, and they introduced a corresponding feature selection algorithm [11]. Chen et al. explored the dominance neighborhood rough set (DNRS) model for hybrid ordered data and discussed a parallel attribute reduction method [12]. Yang et al. introduced an extended DNRS model incorporating fuzzy preference relations and developed three attribute reduction algorithms based on various criteria [13].

Feature selection, also known as attribute reduction in the field of granular computing [14–16], aims to improve classification accuracy and algorithm performance by selecting information-rich attributes and eliminating redundant attributes, which can effectively preprocess data in the field of machine learning and data mining [17–19]. RST-based attribute reduction methods have been extensively studied in the past decades [14–16], and some methods such as e.g. dependency [8, 20, 21], combination entropy, information entropy [22, 23], misclassification rate [24, 25] and a series of metrics have been explored by scholars to evaluate the importance of attributes. However, as people enter the era of big data, various industries are faced with a huge amount of ordered data, i.e., there is order or preference among data, and mining and analyzing [26, 27] ordered data is required in the fields of credit rating evaluation, medical diagnosis, financial risk prediction, etc., which in turn provides valuable information for decision makers' evaluation, recommendation, prediction, and decision making [28]. Therefore, attribute reduction of this kind of data is more urgent.

Incremental attribute reduction methods are tools that leverage existing knowledge to efficiently reduce the attributes of a dataset [29–32]. In the era of big data, ordered data is continuously undergoing dynamic variations, primarily including three types of single-dimensional varies: object set variations, attribute set variations, and attribute value variations [33]. Many scholars have conducted extensive research on these single-dimensional dynamic variations. When the object set varies, Zhang et al. created incremental feature selection techniques using a fuzzy rough set-based information entropy combined with an active object selection strategy [34]. Yang et al. Investigated approaches for incremental feature selection using an active object selection principle [35], and later the authors proposed a method tailored for dynamic heterogeneous data [36]. Shu et al. presented an incremental feature selection algorithm designed for dynamic hybrid data [37]. When the attribute set varies, a incremental attribute reduction method based on recognizable relations [38] for dynamically increasing attributes was proposed by Chen et al. Lang et al. investigated a incremental algorithm based on correlation families [39] for dealing with attribute reduction in dynamic coverage decision information systems. When the attribute value varies, Wei et al. outlined a dynamic strategy for feature selection employing a discernibility matrix [40], followed by the creation of an accelerated incremental algorithm using decision table compression techniques [41]. Cai et al. devised two incremental methods for attribute reduction, focusing on adjusting covering granularity [42]. Additionally, Dong and Chen devised an innovative incremental attribute reduction algorithm based on RST for decision tables undergoing simultaneous increases in both objects and attributes [29].

However, in real life, the dynamic variations of datasets are often multi-dimensional, including three situations: imultaneous variations in the object set and attribute set, simultaneous variations in the object set and attribute values, and simultaneous variations in the attribute set and attribute values [43]. Compared to these dynamic variations, one-dimensional dynamic variations scenarios are overly idealized. Some scholars have recognized this and have conducted research accordingly. In the case of simultaneous dynamic variations of the object set, attribute set and attribute values both at the same time, Wang et al. conducted a study on approximate set updating [43–45]. By reviewing the past research literature, it can be concluded that there are fewer studies on incremental attribute reduction for this kind of two-by-two variation multi-dimensional dynamically variation ordered data, therefore, in this paper, we will study and propose a incremental attribute reduction algorithm for simultaneous variation of object set and attribute set.

To achieve this goal, we propose a novel incremental attribute reduction algorithm based on the theory of granular computing, specifically for situations involving attribute sets and object updates. The main contributions of this study include three key aspects: First, self-information is employed as an uncertainty measure, while a non-monotonic attribute reduction strategy is developed to improve the robustness and accuracy of attribute reduction. Second, introducing the tDRM and designing a special treatment of the triangular matrix for efficiently computation of the upper and lower approximations in order to realize a more efficient attribute reduction process. Thirdly, we explore the updating mechanism of triangular matrix when objects and attributes increase at the same time, and develop an incremental attribute reduction algorithm for multi-dimensional variations ODS.

The rest of the paper is structured as follows: Section 2 will give a background on the basics of rough sets; Section 3 will detail our proposed algorithm; Section 4 will experimentally verify the effectiveness and efficiency of our algorithms; finally, the conclusion will emphasize the efficiency and effectiveness of our algorithms.

2. Preliminary

This section provides a concise overview of RST and DRSA .

A decision system is typically denoted by a quaternion as $S = \langle U, AT \cup D, V, f \rangle$, where U represents a universe, AT is the set of conditional attributes, and D is the set of decisions. V is defined as the union of all V_{a_k} where $a_k \in \{AT \cup D\}$. Here, V_{a_k} represents the domain of attribute a_k, and the function $f: U \times \{AT \cup D\} \to V$ is defined as an information function, and $f(x_i, a_k)$ denotes the value under attribute a_k. For each $a \in \{AT \cup D\}$, if V_a is increasing or decreasing in partial order, the decision system is said to be an ODS [46], denoted $S^{\succcurlyeq} = \langle U, AT \cup D, V, f \rangle$.

Definition 2.1. [47]Given an ODS $S^{\succcurlyeq} = \langle U, AT \cup D, V, f \rangle$, for any $P \subseteq AT$ and $x_i, x_j \in U$, the dominance-based relation is defined as:

$$\mathcal{D}_P^{\succeq} = \{ (x_i, x_j) \in U \times U \mid f(x_i, a) \nprec f(x_j, a), \forall a \in P \}. \tag{1}$$

The set of dominating and dominated of the object xi are defined as follows:

$$\mathcal{D}_P^{+}(x_i) = \{ x_j \in U \mid \langle x_j, x_i \rangle \in \mathcal{D}_P^{\succeq} \}, \tag{2}$$

$$\mathcal{D}_P^{-}(x_i) = \{ x_j \in U \mid \langle x_i, x_j \rangle \in \mathcal{D}_P^{\succeq} \}. \tag{3}$$

Definition 2.2. [47] Given an ODS $S^{\succcurlyeq} = \langle U, AT \cup D, V, f \rangle$, as the decision set D encompasses d distinct values, U can be partitioned into d equivalence classes based on the preference-order relation determined by D. Consequently, this collection of equivalence classes is termed the decision class and can be expressed as:

$$Cl = \{ Cl_n \mid n \in D \}. \tag{4}$$

Two comprehensive sets of classes necessitating approximation exist: the upward union set and the downward union set, defined as follows.

$$Cl_n^{\geqslant} = \bigcup_{n' \geq n} Cl_{n'}, Cl_n^{\leqslant} = \bigcup_{n' \leq n} Cl_{n'}, \forall n', n \in D, D = \{D_1, \dots, D_d\}. \quad (5)$$

Definition 2.3. [48] Given an ODS $S^{\succcurlyeq} = \langle U, AT \cup D, V, f \rangle$, for any $P \subseteq AT$ and $n \in D$, the lower and upper approximations of the upward union $Cl_n^{\succcurlyeq}$ are defined respectively:

$$\underline{\mathcal{D}_P}(Cl_n^{\geqslant}) = \{ x_i \in U \mid \mathcal{D}_P^+(x_i) \subseteq Cl_n^{\geqslant} \}, \quad (6)$$

$$\overline{\mathcal{D}_P}(Cl_n^{\geqslant}) = \{ x_i \in U \mid \mathcal{D}_P^+(x_i) \cap Cl_n^{\geqslant} \neq \emptyset \}; \quad (7)$$

the lower and upper approximations of the downward union $Cl_n^{\preccurlyeq}$ are defined respectively:

$$\underline{\mathcal{D}_P}(Cl_n^{\leqslant}) = \{x_i \in U | \mathcal{D}_P^-(x_i) \subseteq Cl_n^{\geqslant}\}, \quad (8)$$

$$\overline{\mathcal{D}_P}(Cl_n^{\leqslant}) = \{ x_i \in U \mid \mathcal{D}_P^-(x_i) \cap Cl_n^{\geqslant} \neq \emptyset \}. \quad (9)$$

Definition 2.4. [49] The measure of uncertainty for two random variables x_i and x_j, denoted as $I(x_i)$ and $I(x_j)$, is termed the self-information of xi and xj when it possesses the subsequent characteristics:

(1) Non-negative, i.e., $I(x) \nless 0$;

(2) $I(x) \begin{cases} \to \infty & P(x) \to 0 \\ = 0 & P(x) = 1; \end{cases}$

(3) Strict monotonic, i.e., $I(x_i) > I(x_j),\ if P(x_i) < P(x_j)$,

where $P(x)$ is the probability of event x occurring, and x denotes a collective term for x_i and x_j. Due to the properties that the metric has, combined with self-information, the following definitions are available.

Definition 2.5. [50] Given an ODS $S^{\succcurlyeq} = \langle U, AT \cup D, V, f \rangle$, for any $P \subseteq AT$, $n \in D$, the self-information can be defined as:

$$I_P(Cl_n^{\geqslant}) = -\rho_P(Cl_n^{\geqslant}) ln\big(\alpha_P(Cl_n^{\geqslant})\big), \quad (10)$$

Where $\alpha_P(Cl_n^{\succcurlyeq})$ is roughness accuracy, denoted as the ratio of the number of upper and lower approximation sets of $Cl_n^{\succcurlyeq}$ under the attribute subset P, and $\rho_P(Cl_n^{\succcurlyeq})$ is the roughness, denoted as $1 - \alpha_P(Cl_n^{\succcurlyeq})$. The equation is the self-information of a class, and the self-information of the whole ODS with respect to the decision set D is denoted as $I_P(Cl^{\succcurlyeq}) = -\sum_{n=Cl_1^{\succcurlyeq}}^{Cl_d^{\succcurlyeq}} \rho_P(Cl_n^{\succcurlyeq}) ln\left(\alpha_P(Cl_n^{\succcurlyeq})\right)$.

Property 2.1. Let $S^{\succcurlyeq} = \langle U, AT \cup D, V, f \rangle$ be an ODS, for $P_1 \subseteq P_2 \subseteq AT$, there is $I_{P_1}(Cl^{\succeq}) \geq I_{P_2}(Cl^{\succeq})$.

Proof. Since $P_1 \subseteq P_2$, we have $\mathcal{D}_{P_2}^+(x) \subseteq \mathcal{D}_{P_1}^+(x)$ for $x \in U$; $\underline{\mathcal{D}_{P_1}}(Cl_n^{\geqslant}) \subseteq \underline{\mathcal{D}_{P_2}}(Cl_n^{\geqslant})$ and $\overline{\mathcal{D}_{P_2}}(Cl_n^{\geqslant}) \subseteq \overline{\mathcal{D}_{P_1}}(Cl_n^{\geqslant})$ by Eqs. (6) and (7). By the definition of roughness $\alpha_P(Cl_n^{\succcurlyeq})$, we have $0 \leq$

ISSN: 2664-9640

$\alpha_{P_1}(Cl_n^{\succcurlyeq}) \leq \alpha_{P_2}(Cl_n^{\succcurlyeq}) \leq 1$, and $0 \leq 1-\alpha_{P_2}(Cl_n^{\succcurlyeq}) \leq \alpha_{P_1}(Cl_n^{\succcurlyeq}) \leq 1$, i.e., $0 \leq \rho_{P_2}(Cl_n^{\succcurlyeq}) \leq \rho_{P_2}(Cl_n^{\succcurlyeq}) \leq 1$ and $ln\alpha_{P_1}(Cl_n^{\succcurlyeq}) \leq ln\alpha_{P_2}(Cl_n^{\succcurlyeq})$. Therefore, $-\rho_{P_2}(Cl_n^{\succcurlyeq})ln\left(\alpha_{P_2}(Cl_n^{\succcurlyeq})\right) \leq$ $-\rho_{P_1}(Cl_n^{\succcurlyeq})ln\left(\alpha_{P_1}(Cl_n^{\succcurlyeq})\right)$, i.e., $I_{P_1}(Cl^{\succcurlyeq}) \geq I_{P_2}(Cl^{\succcurlyeq})$ holds and the property is proved.

3. Proposed Method and Algorithm

3.1. Incremental Mechanism

Definition 3.1. Given an ODS $S^{\succcurlyeq} = \langle U, AT \cup D, V, f\rangle$, for $P \subseteq AT$, the GD of P for $\forall x_i \in U$ is denoted as $GD = \langle P_l, P_u\rangle(x_i)$ where $P \subseteq AT$, and for P_l and P_u,

$$P_{l(x_i)} = \min\{f\left(x_j, D_k\right) \mid \langle x_j, x_i\rangle \in D_P^{\succcurlyeq}, x_j \in U\}, \tag{11}$$

$$P_{u(x_i)} = \max\{f\left(x_j, D_k\right) \mid \langle x_i, x_j\rangle \in D_P^{\succcurlyeq}, x_j \in U\}; \tag{12}$$

The GD is introduced here because the process of calculating upper and lower approximations by ordinary methods is too cumbersome. This method simplifies the calculation of the upward and downward approximations. At the same time, it also saves the time overhead of calculating the upper and lower approximations. We have also refined the GD. The following definition shows the principle of calculating the up and down approximation by GD.

Definition 3.2. [51]Given an ODS $S^{\succcurlyeq} = \langle U, AT \cup D, V, f\rangle$, for any $P \subseteq AT$ and $n \in D$, the lower and upper approximations of the approximation of the upward union $Cl_n^{\geq}$ and downward union $Cl_n^{\leq}$ are redefined respectively:

$$\underline{D_P}(Cl_n^{\succcurlyeq}) = \{x \in U \mid P_l(x_i) \geq Cl_n^{\succcurlyeq}\}, \tag{13}$$

$$D_P(Cl_n^{\succcurlyeq}) = \{x \in U \mid P_u(x_i) \geq Cl_n^{\succcurlyeq}\}, \tag{14}$$

$$D_P(Cl_n^{\preccurlyeq}) = \{x \in U \mid P_u(x_i) \leq Cl_n^{\preccurlyeq}\}, \tag{15}$$

$$D_P(Cl_n^{\preccurlyeq}) = \{x \in U \mid P_l(x_i) \leq Cl_{n\}}^{\preccurlyeq}. \tag{16}$$

According to the definition of lower approximation, lower approximation represents accurate information. In a set, if the largest values are all smaller than the value to be compared, there is no element in the current set that is larger than that value; conversely, if the total smallest values of the set are all larger than the value to be compared, there is no element in the current set that is smaller than that value, in ODS, this situation can be understood as a lower approximation of the value, and it is important to note that at this point, the granularity needs to be specified. By analyzing the definition of upper approximation, upper approximation denotes ambiguous and possible information. Similarly, imagine that in a collection, at a specific granularity, if the largest value is larger than the value to be compared, then the current value can be an element of the upper approximation of that value. And vice versa.

In this study, we focus on the case where the upward union $Cl_n^{\succcurlyeq}$.

Proposition 3.1. Given an ODS $S^{\succeq} = \langle U, AT \cup D, V, f \rangle$, for any $P \subseteq AT$ and $x_i, x_j \in U$, we define dominance-based relation matrix (DRM) as:

$$\mathbb{M}_U^{\succeq P} = \begin{bmatrix} m_{(1,1)}^{\succeq P} & m_{(1,2)}^{\succeq P} & \cdots & m_{(1,n)}^{\succeq P} \\ m_{(2,1)}^{\succeq P} & m_{(2,2)}^{\succeq P} & \cdots & m_{(2,n)}^{\succeq P} \\ \vdots & \vdots & \vdots & \vdots \\ m_{(n,1)}^{\succeq P} & m_{(n,1)}^{\succeq P} & \cdots & m_{(n,n)}^{\succeq P} \end{bmatrix}, \tag{17}$$

Where

$$m_{(i,j)}^{\succeq P} = \begin{cases} 0, & \langle x_i, x_j \rangle \notin \mathcal{D}_P^{\succeq}, \\ 1, & \langle x_i, x_j \rangle \in \mathcal{D}_P^{\succeq} \\ 2, & \langle x_j, x_i \rangle \in \mathcal{D}_P^{\succeq}. \\ 3, & f(x_j, P) = f(x_i, P). \end{cases} \tag{18}$$

To compute the lower approximation and upper approximation of the current ODS by the matrix method, the previous GD needs to be further improved with the following arithmetic rules:

$$P_{l(x_i)} = \min\{f(x_j, D_k) \mid m_{(i,j)}^{\succcurlyeq P} \in \{1,3\}, x_j \in U, m_{(i,j)}^{\succcurlyeq P} \in M_U^{\succcurlyeq P}\}, \tag{19}$$

$$P_u(x_i) = \max\{f(x_j, D_k) \mid m_{\{(i,j)\}}^{\succcurlyeq P} \in \{2,3\}, x_j \in U, m_{\{(i,j)\}}^{\succcurlyeq P} \in M_U^{\succcurlyeq P}\}; \tag{20}$$

Proof. In an ordinary matrix, the matrix elements are generally composed of 1 or 0, where 1 means $\langle x_i, x_j \rangle \in D_P^{\succcurlyeq}$, 0 means $\langle x_j, x_i \rangle \in D_P^{\succcurlyeq}$, and if $f(x_j, P) = f(x_i, P)$, the element is denoted as 1. In our method, we use two elements to denote the dominance relation between different objects, i.e., $(01)_2$ means$\langle x_i, x_j \rangle \in D_P^{\succcurlyeq}$, $(10)_2$ means $\langle x_j, x_i \rangle \in D_P^{\succcurlyeq}$, and if $f(x_j, P) = f(x_i, P)$, the element is $(11)_2$. In decimal, $(01)_2$, $(10)_2$, and $(11)_2$ in binary are respectively are equal to $(1)_{10}$, $(2)_{10}$ and $(3)_{10}$.

Definition 3.3. Given an ODS $S^{\succcurlyeq} = \langle U, AT \cup D, V, f \rangle$, for any $P \subseteq AT$, we define tDRM as

$$\mathbb{TM}_U^{\succeq P} = \begin{bmatrix} tm_{(1,1)}^{\succeq P} & tm_{(1,2)}^{\succeq P} & \cdots & tm_{(1,n)}^{\succeq P} \\ & tm_{(2,2)}^{\succeq P} & \cdots & tm_{(2,n)}^{\succeq P} \\ & & \ddots & \vdots \\ & & & tm_{(n,n)}^{\succeq P} \end{bmatrix}_{n \times n}. \tag{21}$$

Since a triangular matrix is used here to save time, the matrix elements need to be treated as follows when solving the upper and lower approximations of $Cl^{\succcurlyeq}$:

$$tm_{(i,j)}^{\succeq P} = \begin{cases} m_{(i,j)}^{\succeq P}, & i \in [1,n] \cap j \in [i,n]; \\ \sim m_{(j,i)}^{\succeq P}, & j \in [1,n-1] \cap i \in [j+1,n]. \end{cases} \tag{22}$$

Definition 3.4. Given an ODS $S^{\succcurlyeq} = \langle U, AT \cup D, V, f \rangle$, due to $Cl = D$, the triangular-matrix-based self-information of $Cl^{\succcurlyeq}$ can also be defined as

$$\mathbb{MI}_P^U(Cl^{\succeq}) = -\sum_{n=Cl_1^{\succeq}}^{Cl_d^{\succeq}} \left(1 - \frac{|\underline{\underline{\mathcal{D}_P}}(Cl_n^{\succeq})|}{|\overline{\overline{\mathcal{D}_P}}(Cl_n^{\succeq})|}\right) \ln\left(\frac{|\underline{\underline{\mathcal{D}_P}}(Cl_n^{\succeq})|}{|\overline{\overline{\mathcal{D}_P}}(Cl_n^{\succeq})|}\right). \tag{23}$$

Further explanation of Eq. 22: Operator " " denotes a bitwise NOT operation, assuming the tDRM is complete, there is $m(i,j) =\sim m(j,i)$, and $1(01)_2$ and $2(10)_2$ are inverse by bit when expressed in binary. To find $P_l(x_i)$ and $P_u(x_i)$ in the upper triangular matrix, the relational identity of all objects with respect to the current object is used. In this case, the first element of the row of the matrix in which the current object is located, i.e., $m(i,i)$, and all the elements of the column in which it is located that are either $(1)_{10}$ or $(2)_{10}$ need to be interchanged when comparing them, i.e., $(1)_{10}$ converted to $(2)_{10}$, and $(2)_{10}$ converted to $(1)_{10}$.

The update mechanism of the proposed algorithm is described next.

Proposition 3.2. Mechanism for updating the matrix when a new set of objects $U^+ = (x_{n+1}, x_{n+2}, \dots, x_{n+n^+})$and a new set of attributes$AT^+ = (a_{m+1}, a_{m+2}, \dots, a_{m+m^+})$ are added to the original ordered decision system, which can be divided into two stages.

(1) When processing the newly arrived attributes in the original object, $AT' = AT \cup AT^+$, the original tDRM $\mathbb{TM}_{U'}^{\succeq AT'}$ is updated as follows:

$$tm'^{\succcurlyeq P}_{(i,j)} = \begin{cases} 0, & \langle x_i, x_j \rangle \notin \mathcal{D}_{AT^+}^{\succcurlyeq}, \\ tm_{(i,j)}^{\succcurlyeq AT} \& 1, & \langle x_i, x_j \rangle \in \mathcal{D}_{AT^+}^{\succcurlyeq}, \\ tm_{(i,j)}^{\succcurlyeq AT} \& 2, & \langle x_j, x_i \rangle \in \mathcal{D}_{AT^+}^{\succcurlyeq}, \\ tm_{(i,j)}^{\succcurlyeq AT}, & f(x_j, P) = f(x_i, P). \end{cases} \tag{24}$$

(2) When processing new objects and new attributes, $U' = U \cup U^+$, the updated tDRM $\mathbb{TM}_{U'}^{\succeq AT'}$ makes the following updates:

$$tm''^{\succcurlyeq P}_{(i,j)} = \begin{cases} 0, & \langle x_i, x_j \rangle \notin \mathcal{D}_{AT'}^{\succcurlyeq}, \\ 1, & \langle x_i, x_j \rangle \in \mathcal{D}_{AT'}^{\succcurlyeq}, \\ 2, & \langle x_j, x_i \rangle \in \mathcal{D}_{AT'}^{\succcurlyeq}, \\ 3, & f(x_j, AT') = f(x_i, AT'). \end{cases} \tag{25}$$

Note that the attribute set and the object set are added at the same time, so the principles are explained separately for convenience.

Proof. (1) Due to the nature of the dominance relation, an object is included in its set of dominancing or dominanced, i.e., $D_P^+(x_i) = \{x_j \in U \mid \langle x_j, x_i \rangle \in D_P^{\succcurlyeq}\}$ or $D_P^-(x_i) = \{x_j \in U \mid$

$\langle x_i, x_j \rangle \in D_P^{\succcurlyeq}\}$, only if it completely dominates or dominanced the current object, otherwise x_i is dominance independent to x_j with respect to P; in the operation of and with with binary with bit 2, there are $01\&01 = 01$, $01\&10 = 00$, $01\&11 = 01$ and $10\&11 = 10$, and when converted to decimal, there are $1\&1 = 1$, $1\&2 = 0$, $1\&3 = 1$ and $2\&3 = 2$, and in the setup of Proposition 3.1, the elements in the matrix are all represented by the two-digit binary code, which is also in accordance with the rules of arithmetic, and, therefore, the first three terms of Eq. 24 hold; and, in the operation of and, since all digits of any number are equal to itself, i.e., xx&11 = xx, when the operation of and is performed in binary, the last term of Eq. 24 holds, i.e., Proposition 3.2 (1) holds.

(2) By Definition 18, Proposition 3.2(2) can be proved directly.

Note that this method can also be used for tDRM $\mathbb{M}_{U'}^{\succeq B}$ when solving for the outer significance of a, where R denotes a temporary reduct in updated ODS and $a \in (AT' - B)$.

It should be noted that the division into two steps is for the readers to understand easily, but in fact, the matrix is updated in one step.

After calculating the uTDRM, the approximation set is easy to calculate, and then the uncertainty measure can also be obtained, specifically refer to Example 3. In the process of attribute reduction, the inner and outer significance is a measure of the degree of importance of an attribute in the ODS, and the specific significance is as follows: the inner significance is often used to measure the degree of redundancy of an attribute a, and the attribute a will be retained if $sig_{inner}^{\succcurlyeq U}(a, P, D) > 0$, where $a \in P \subseteq AT$; the outer significance is often used to measure the importance of an attribute, and then judge whether to add reduct. inner and outer significance is defined as follows, respectively:

Definition 3.5. Given an ODS $S^{\succcurlyeq} = \langle U, AT \cup D, V, f \rangle$, for any attribute subset $P \subseteq AT$ and any attribute element $a \in P$, the representation of the inner significance and the outer significance of a using $\mathbb{MI}_P^U$ is formulated as

$$sig_{inner}^{\succeq U}(a, P, D) = \mathbb{MI}_{P-\{a\}}^{U}(Cl_n^{\succeq}) - \mathbb{MI}_P{}^{U}(Cl_n^{\succeq}) \tag{26}$$

and

$$sig_{outer}^{\succeq U}(a, P, D) = \mathbb{MI}_P^U(Cl_n^{\succeq}) - \mathbb{MI}_{P\cup\{a\}}^{U}(Cl_n^{\succeq}), \tag{27}$$

respectively.

3.2. Heuristic Self-information based Attribute Reduction Algorithm

When the set of objects and attributes of an ODS varies, the traditional approach, i.e., the heuristic approach, recalculates the approximation of that ODS, which is non-incremental and more time-consuming compared to the incremental attribute reduction approach. This subsection presents a self-informed matrix-based heuristic attribute reduction algorithm, the details of which are shown in Algorithm 1.

Algorithm 1 HAR MDSI algorithm

Input: An ODS $S^{\succeq} = \langle U, AT \cup D, V, f \rangle$.

Output: A reduct $R_{S^{\succeq}}$.

1: A reduct $R_{S^{\succeq}} \leftarrow \emptyset$;
2: Calculate $\mathbb{MI}_{AT}^{U}(Cl_n^{\succeq})$ via using Definition 3.4
3: **for** $a \in AT$ **do**

4: Calculate $sig_{inner}^{\succeq U}(a, AT, D) = \mathbb{MI}_{AT-\{a\}}^{U}(Cl^{\succeq}) - \mathbb{MI}_{AT}^{U}(Cl^{\succeq})$ by Eq. 26;
5: **If** $sig_{inner}^{\succeq U}(a, AT, D) > 0$ **then:**
6: $R_{S^{\succeq}} \leftarrow R_{S^{\succeq}} \cup \{a\}$;
7: **end if**
8: end for
9: Let $B \leftarrow R_{S^{\succeq}}$;
10: **while** $\mathbb{MI}_{B}^{U}(Cl^{\succeq}) \neq \mathbb{MI}_{AT}^{U}(Cl^{\succeq})$ **do**
11: **for** $a \in (AT - B)$ **do**
12: Calculate $sig_{outer}^{\succeq U}(a, B, D) = \mathbb{MI}_{B}^{U}(Cl^{\succeq}) - \mathbb{MI}_{B\cup\{a\}}^{U}(Cl^{\succeq})$ by Eq. 27;
13: **end for**
14: $a_{max} = \arg\max\{ sig_{outer}^{\succeq U}(a, B, D) \mid a \in (AT - B)\}$;
15: $B \leftarrow B \cup \{a_{max}\}$;
16: **end while**
17: **for** $a \in B$ do
18: Calculate $\mathbb{MI}_{B-\{a\}}^{U}(Cl^{\succeq})$ via using Eq. 26;
19: **if** $\mathbb{MI}_{B-\{a\}}^{U}(Cl^{\succeq}) = \mathbb{MI}_{B}^{U}(Cl^{\succeq})$ **then**
20: $B \leftarrow (B - \{a\})$;
21: **end if**
22: **end for**
23: $R_{S^{\succeq}} \leftarrow B$;
24: **return** $Red_{S^{\succeq}}$.

In Algorithm HAR_MDSI, Step 2 calculates the uncertainty measure $\mathbb{MI}_{AT}^{U}(Cl_{n}^{\succeq})$ of the current ODS under AT by Definition 3.4; Steps 3-9 add the attribute elements with inner significance greater than 0 to the initial reduct $R_{S^{\succeq}}$ and temporarily denote the reduct $R_{S^{\succeq}}$ by B instead; Steps 10-16 add the attribute with the largest outer significance to the temporary reduct B until the $\mathbb{MI}_{B}^{U}(Cl^{\succeq})$ under B and the $\mathbb{MI}_{AT}^{U}(Cl^{\succeq})$ under AT are the same; Steps 17-22 remove the redundant attributes in the temporary reduct B by inner significance; finally, Steps 23-24 obtain the final reduct $R_{S^{\succeq}}$.

Time Complexity Analysis: in Step 2, when calculating the uncertainty measure, the matrix used is the complete matrix, i.e., DRM, and according to Definitions 3.1 and 3.2, the calculation of the upper and lower approximations needs to find the GD of all the objects, which requires a sorting operation, and then compares the GD of each object with the decision attributes, and the time complexity of this sub-step is $O(|U|^2 log|U| + 2|U|)$, thus the time complexity of Step 2 is $O(|U|^2 + |U|^2 log|U| + 2|U|)$; the time complexity of Steps 3-9 is $O\big((|AT|)(|U|^2 + |U|^2 log|U| + 2|U|)\big)$; the time complexity of Steps 10-16 is $O\big((|AT - B|^2)(|U|^2 log|U| + 2|U|)\big)$; similarly, the time complexity of Steps 17-23 is $O\big((|B|)(|U|^2 log|U| + 2|U|)\big)$. Thus, the time complexity of Algorithm 1 is calculated to be $O\big(|U|^2 + |U|^2 log|U| + 2|U| + (|AT|)(|U|^2 + |U|^2 log|U| + 2|U|) + (|AT - B|^2)(|U|^2 log|U| + 2|U|) + (|B|)(|U|^2 log|U| + 2|U|)\big)$.

3.3. Incremental Self-information-based Attribute Reduction Algorithm

Traditional attribute reduction algorithms require re-computation from start to finish when both the object set and the attribute set vary at the same time, and the knowledge learned earlier is completely overwritten, and there is unnecessary time overhead. On the contrary, incremental attribute reduction performs attribute reduction on the updated ODS based on the original reduct, where the original knowledge is fully utilized and the time overhead is reduced. Algorithm 2 shows the specific steps of incremental attribute reduction proposed in this paper.

Algorithm 2 IAR tMDSI algorithm

Input: An ODS (ODS) $S^{\succeq} = \langle U, AT \cup D, V, f \rangle$; the $U^{+} = (x_{n+1}, x_{n+2}, \ldots, x_{n+n^{+}})$, the $AT^{+} = (a_{m+1}, a_{m+2}, \ldots, a_{m+m^{+}})$,the initial $R_{S^{\succeq}}$ on $S^{\succcurlyeq}$.

Output: A reduct $R_{S^{\succeq}}$ on $S^{\succcurlyeq}$.

1: Let $B' \leftarrow R_{S^{\succeq}}, U' \leftarrow U \cup U^{+}, AT' \leftarrow AT \cup AT^{+}$;
2: Update $\mathbb{MI}^{U}_{AT^{+}}(Cl^{\succeq})$;
3: **for** $a \in AT^{+}$ **do**
4:     Concurrent calculate $sig^{\succeq U}_{inner}(a, AT^{+}, D) = \mathbb{MI}^{U}_{AT^{+}-\{a\}}(Cl^{\succeq}) - \mathbb{MI}^{U}_{AT^{+}}(Cl^{\succeq})$ by Eq. 26;
5:     **if** $sig^{\succeq U}_{inner}(a, AT^{+}, D) > 0$ **then**
6:         $B' \leftarrow B' \cup \{a\}$;
7:     **end if**
8: **end for**
9: **if** $\mathbb{MI}^{U'}_{B'}(Cl^{\succeq}) \neq \mathbb{MI}^{U'}_{AT'}(Cl^{\succeq})$ **then**
10:     Calculate the outer significance of each attribute $a \in (AT' - B')$ concurrently and construct an attribute sequence like $a'_{1}, a'_{2}, \ldots, a'_{|AT'-B'|}$ according to the descending sequence of $sig^{\succeq U'}_{outer}(a, B', D)$;
11:     **for** $a \in (AT' - B')$ **do**
12:         Choose $B' \leftarrow B' \cup \{a\}$ and calculate $\mathbb{MI}^{U'}_{B'}(Cl^{\succeq})$;
13:         **if** $\mathbb{MI}^{U'}_{B'}(Cl^{\succeq}) = \mathbb{MI}^{U'}_{AT'}(Cl^{\succeq})$ **then**
14:             break;
15:         **end if**
16:     **end for**
    end if
17: **for** $a \in B'$ **do**
18:     Calculate $sig^{\succeq U'}_{inner}(a, B', D) = \mathbb{MI}^{U'}_{B'-\{a\}}(Cl^{\succeq}) - \mathbb{MI}^{U'}_{B'}(Cl^{\succeq})$ by Eq. 26;
19:     **if** $sig^{\succeq U'}_{inner}(a, B', D) = 0$ **then**
20:         $B' \leftarrow (B' - \{a\})$;
21:     **end if**
22: **end for**
23: $R_{S'^{\succeq}} \leftarrow B'$;
24: **return** $R_{S'^{\succeq}}$.

In Algorithm IAR_tMDSI, Step 1 replaces the original reduct $R_{S^{\succcurlyeq}}$ with the temporary reduct B', and the ODS is updated simultaneously, including the object set and the attribute set; Step 2 computes the uncertainty metric $\mathbb{MI}^{U}_{AT^{+}}(Cl^{\succeq})$ of the current ODS under AT' by Definition 3.4, but it should be noted that this part is realized by the incremental approach, embodied in changing the DRM to tDRM via Definition 3.3 and updating the tDRM via Proposition 3.2; Steps 3-8 add in parallel attribute elements a with an inner significance greater than 0 to the temporary reduct B', where $a \in AT^{+}$; Step 9-17 calculates the outer significance of each attribute element in attribute subset $(AT' - B')$ and creates a sequence in descending order, adding the first attribute in the constructed list, i.e., the attribute element with the largest outer significance, to the temporary reduct B' each time the loop occurs until the $\mathbb{MI}^{U'}(Cl^{\succeq})$ under B' is the same as that under AT', and here also the incremental approach is used, and its incremental at the algorithmic level; Steps 18-23 remove the redundant attributes in the temporary reduct B' by inner significance; finally, Steps 23-24 get the final reduct.

Time Complexity Analysis: in our proposed algorithm, tDRM is used instead of DRM, which saves half of the time compared to DRM. In finding the attribute elements with internal importance greater than 0 in the added attribute AT^{+}, since each computation is performed independently, a parallel approach is considered, which provides a more significant speed-up compared to the traditional approach, since only one attribute element internal importance needs to be computed in the time scale. In Steps 9-17, the attribute element added each time is the first element of the descending queue (each time it is added, the current first element is removed from the queue), rather than recalculating the maximum value, so the time complexity is $O\big((|AT' - B|)(|U'/2|^2 + U'/2^2 log|U'/2| + |U'|)\big)$. The overall time complexity of Algorithm 2 is then $O\big((|U'|/2)^2 + (|U'|/2)^2 log(|U'|/2) + 2(|U'|/2) + (|AT' - B'|)((|U'|/2)^2 + (|U'|/2)^2 log(|U'|/2) + |U'|) + (|B'|)((|U'|/2)^2 + (|U'|/2)^2 log(|U'|/2) + |U'|)\big)$.

Table 1. Comparison of time complexity of different algorithms.

Algorithm	Time Complexity
HAR_MDSI	$O(\lvert U'\rvert^2 + \lvert U'\rvert^2 log\lvert U'\rvert + 2\lvert U'\rvert + (\lvert AT'\rvert)(\lvert U'\rvert^2 + \lvert U'\rvert^2 log\lvert U'\rvert + 2\lvert U'\rvert) + (\lvert AT' - B'\rvert^2)(\lvert U'\rvert^2 + \lvert U'\rvert^2 log\lvert U'\rvert + 2\lvert U'\rvert) + (\lvert B'\rvert)(\lvert U'\rvert^2 + \lvert U'\rvert^2 log\lvert U'\rvert + 2\lvert U'\rvert))$
IAR_tMDSI	$O((\lvert U'\rvert/2)^2 + (\lvert U'\rvert/2)^2 log(\lvert U'\rvert/2) + 2(\lvert U'\rvert/2) + (\lvert AT' - B'\rvert)((\lvert U'\rvert/2)^2 + (\lvert U'\rvert/2)^2 log(\lvert U'\rvert/2) + \lvert U'\rvert) + (\lvert B'\rvert)((\lvert U'\rvert/2)^2 + (\lvert U'\rvert/2)^2 log(\lvert U'\rvert/2) + \lvert U'\rvert))$

4. Experiment and Analysis

Table 2. Detailed description of data and two algorithms effects.

Datasets	Samples	NoA	IAR_TMDSI			HAR_MDSI			RRT(%)	NoD
			NoR(%)	RR(%)	Time(ms)	NoR(%)	RR(%)	Time(ms)		
Car	1728	6	1	16.67	1.40	6	100	14.34	9.77	4
Glass	214	10	1	10.00	0.04	1	10	0.20	17.81	7
Card	2126	21	14	66.67	37.59	13	61.90	73.64	51.04	3
White	4898	11	11	100.00	126.74	11	100	239.07	53.01	7
452	280	23	8.21	8.41	1	0.36	137.87	6.10	2	
Wilt	4839	5	4	80.00	114.26	5	100	116.34	98.21	2
Page	5473	10	6	60.00	124.64	6	60	227.85	54.70	5
Spam	4601	57	37	64.91	392.97	35	61.40	1193.02	32.94	2
Wave	5000	21	20	95.24	303.09	16	76.19	557.31	54.38	3
Seme	1593	256	94	36.72	211.89	1	0.39	1627.72	13.02	10
Mice	1077	68	57	83.82	44.33	21	30.88	67.91	65.28	8
Secom	1567	590	14	2.37	217.56	1	0.17	7606.37	2.86	2
Average	2797.33	111.25	23.5	52.05	131.91	9.75	50.11	988.47	38.26	-

In this section, we evaluate the performance of Algorithm IAR_tMDSI in terms of both effectiveness and efficiency. In terms of effectiveness, we evaluate the performance of Algorithm IAR_tMDSI with algorithms HFS_DRS [15], FS_DRS [15], FS_FDRS [11], and HAR_MDSI in terms of approximate size and classification accuracy, where algorithm HAR_MDSI is the algorithm in this paper.1 In terms of efficiency, we compare the performance of algorithms HFS_DRS, FS_DRS, FS_FDRS, AR_ MDSI and IAR_tMDSI algorithms in terms of computation time and speed-up ratio. All datasets are publicly available datasets downloaded from the UCI database; some of the original datasets in Table 2 could not be used directly in the experiments because they contain missing values. To solve this problem, we preprocessed these datasets. For datasets with fewer missing values, such as the mouse protein expression dataset and the postoperative (Mice) dataset, we chose to remove objects containing missing values to ensure the integrity of the data and the accuracy of the experimental results. The platform on which all comparison experiments in this section were run was a computer configured with a 64-bit Microsoft Windows 10 system with an Inter(R) Core(TM) i9-10900X CPU at 3.70 GHz and 128 GB of RAM, and all algorithms were implemented by Matlab R2020a. Thespecific experimental design is described as follows.

4.1. Effectiveness Evaluations

As mentioned earlier, attribute reduction is a key stage in data preprocessing. Its purpose is to select informationrich attributes in a dataset while eliminating redundant and inconsistent attributes. Its overall goal is to improve classification accuracy and algorithm performance. Therefore, this section will evaluate the efficacy of the algorithm with a focus on classification accuracy.

Table 3. The comparison of classification accuracies of different algorithms on NB and SVM. (%)

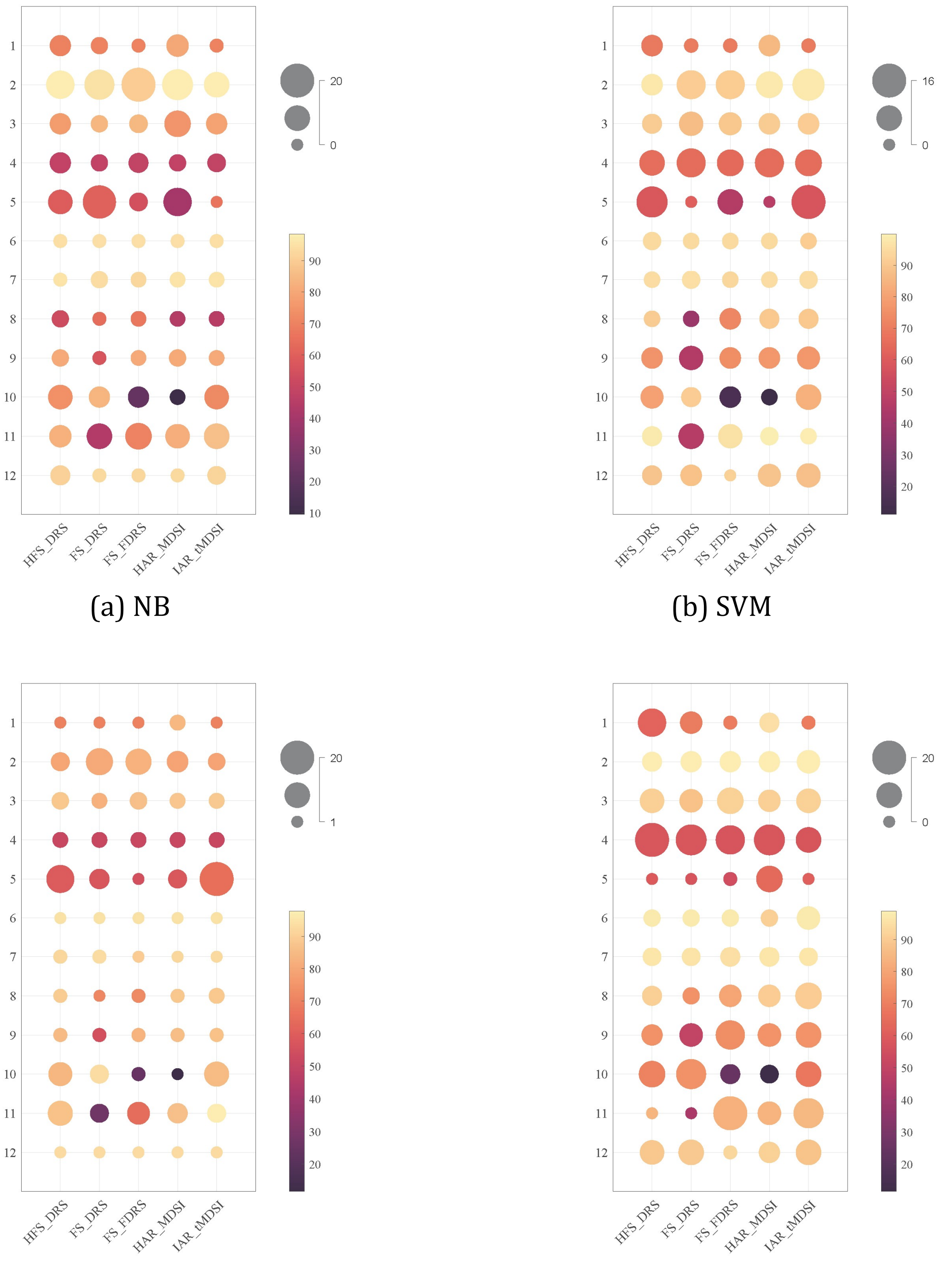
Dataset	NB					SVM				
	HFS_DRS	FS_DRS	FS_FDRS	HAR_MDSI	IAR_tMDSI	HFS_DRS	FS_DRS	FS_FDRS	HAR_MDSI	IAR_tMDSI
Car	69.92±0.26	69.97±0.12	70.02±0.00	**80.34±0.31**	70.02±0.00	70.02±0.00	70.02±0.00	70.02±0.00	**84.00±0.15**	70.02±0.00
Glass	98.46±0.62	95.28±0.75	90.23±0.97	**98.46±0.76**	98.27±0.50	79.25±0.24	80.37±0.58	**82.62±0.53**	79.21±0.33	79.07±0.20
Card	76.60±0.26	84.19±0.12	**84.69±0.19**	75.25±0.51	78.70±0.27	88.02±0.17	82.15±0.15	86.65±0.18	87.76±0.15	**88.71±0.13**
White	48.35±0.26	48.35±0.15	**48.71±0.20**	48.35±0.15	48.49±0.19	52.06±0.12	52.02±0.12	**52.11±0.11**	52.02±0.12	52.04±0.11
Arrhy	59.58±0.44	60.62±0.91	54.73±0.19	39.87±0.64	**66.75±1.15**	59.05±0.64	57.35±0.29	55.09±0.00	57.85±0.22	**65.55±0.98**
Wilt	94.63±0.01	**94.64±0.02**	**94.64±0.02**	**94.64±0.02**	94.59±0.01	**94.61±0.00**	**94.61±0.00**	**94.61±0.00**	**94.61±0.00**	**94.61±0.00**
Page	95.51±0.04	94.07±0.12	92.89±0.06	**95.54±0.07**	95.50±0.05	92.38±0.05	**93.43±0.09**	89.79±0.01	92.35±0.05	92.39±0.05
Spam	52.58±0.14	64.36±0.05	**67.29±0.07**	44.61±0.09	45.27±0.06	**89.27±0.06**	71.72±0.01	72.11±0.07	88.15±0.09	88.56±0.10
Wave	80.39±0.11	55.80±0.04	80.45±0.06	**80.80±0.11**	80.71±0.09	85.28±0.10	55.05±0.07	83.27±0.06	85.75±0.08	**86.97±0.09**
Seme	73.84±0.41	**83.68±0.27**	22.53±0.26	9.45±0.08	72.15±0.43	83.92±0.45	**93.52±0.22**	24.83±0.07	11.51±0.03	84.88±0.49
Mice	82.47±0.31	43.80±0.46	70.39±0.52	82.10±0.42	**86.64±0.47**	86.99±0.49	27.21±0.21	64.50±0.35	86.30±0.28	**97.78±0.23**
Secom	91.40±0.24	**93.43±0.00**	92.47±0.00	93.36±0.02	91.51±0.15	**93.36±0.00**	**93.36±0.00**	93.36±0.00	**93.36±0.00**	**93.36±0.00**
Average	76.98±0.26	74.01±0.25	72.42±0.21	70.23±0.27	77.38±0.28	81.18±0.19	72.57±0.15	72.41±0.12	76.07±0.12	82.83±0.20

Table 4. The comparison of classification accuracies of different algorithms on KNN and CART. (%)

Dataset	KNN					CART				
	HFS_DRS	FS_DRS	FS_FDRS	HAR_MDSI	IAR_tMDSI	HFS_DRS	FS_DRS	FS_FDRS	HAR_MDSI	IAR_tMDSI
Car	69.62±0.21	70.02±0.02	70.02±0.00	**86.08±0.20**	70.02±0.00	62.45±0.61	69.20±0.33	70.02±0.00	**95.13±0.22**	70.02±0.00
Glass	97.90±0.25	91.82±0.50	91.31±0.55	98.18±0.41	**98.27±0.70**	98.74±0.23	98.50±0.30	98.50±0.30	98.50±0.30	**98.83±0.40**
Card	90.91±0.18	87.28±0.33	90.02±0.28	91.19±0.22	**91.79±0.21**	91.09±0.42	87.70±0.35	91.60±0.52	91.28±0.32	**91.61±0.45**
White	**65.93±0.39**	65.76±0.50	64.82±0.41	65.76±0.50	65.72±0.42	**58.13±0.99**	57.91±0.75	57.37±0.65	57.91±0.75	57.82±0.46
Arrhy	58.87±0.63	**61.68±1.13**	46.24±0.36	47.63±1.10	57.59±0.79	59.27±2.35	57.52±1.42	55.09±0.00	**64.60±0.52**	61.13±2.11
Wilt	**94.99±0.11**	94.94±0.10	94.94±0.10	94.94±0.10	91.31±0.09	97.97±0.11	**98.08±0.14**	98.03±0.13	91.29±0.13	**98.08±0.36**
Page	95.59±0.09	**96.02±0.10**	94.17±0.08	95.54±0.08	95.57±0.12	**96.73±0.16**	96.44±0.15	95.06±0.21	96.71±0.24	96.70±0.18
Spam	**90.88±0.10**	39.30±0.09	72.94±0.21	90.18±0.15	90.43±0.17	**91.16±0.21**	75.17±0.13	80.34±0.32	90.87±0.31	90.87±0.51
Wave	76.57±0.24	45.52±0.34	74.87±0.22	77.28±0.23	**77.45±0.27**	75.17±0.28	50.29±0.37	74.06±0.68	**75.49±0.40**	75.47±0.50
Seme	79.99±0.25	**91.70±0.18**	16.03±0.24	11.09±0.06	83.82±0.39	71.02±0.52	**75.05±0.72**	24.57±0.20	11.47±0.16	67.87±0.48
Mice	99.08±0.20	46.50±0.40	96.62±0.32	99.23±0.14	**99.89±0.09**	84.28±1.19	43.46±1.25	82.90±0.97	83.88±0.39	**85.26±0.70**
Secom	89.27±0.18	88.53±0.24	**92.51±2.70**	89.16±0.29	87.95±0.30	89.20±0.40	89.94±0.48	**93.36±0.00**	91.54±0.29	88.23±0.49
Average	84.13±0.23	73.26±0.33	75.38±0.46	78.86±0.29	**84.15±0.29**	81.27±0.62	74.94±0.53	76.74±0.33	79.06±0.34	**81.82±0.55**

ISSN: 2664-9640

4.1.1. Comparison of the Algorithms IAR tMDSI and HAR MDSI in Terms of Reduct Size



(a) NB (b) SVM

(c) KNN (d) CART

Figure 1. Comparison of different attribute reduction algorithms under different classifiers.

In Table 2, a detailed description of the datasets required for the experiments is shown. The number of reducts (NoR), the reduct ratio (RR, $RR=NoR/NoA$), the time and the ratio of the running time (RRT, $RRT = T_{IAR\_tMDSI}/T_{HAR\_MDSI}$, T_* is the running time of Algorithm *) of the two algorithms computed by the algorithms IAR_tMDSI and HAR_MDSI to add the remaining 50% of the data (attributes and objects) in each dataset to the original 50% of the data, are included in columns 5 through 11, where NoR is the number of reduct, NoA is the number of attribute and NoD is the number of decision set.

ISSN: 2664-9640

Comparing the contents of columns 6 and 9, it can be seen that the dataset White fails to compute the reduct by Algorithm IAR_tMDSI, while the datasets Car, White, and Wilt fail to compute the reduct by algorithm HAR_MDSI, thus, compared to algorithm HAR_MDSI, the Algorithm IAR_tMDSI can more efficiently perform attribute reduction. Moreover, Algorithm IAR_tMDSI saves more time cost than algorithm HAR_MDSI, as can be seen from the column RRT, which is less than 60% for the 12 datasets, and even less than 20% for the Datasets Car, Glass, Arrhy, Seme, and Secom, which suggests that Algorithm IAR_tMDSI significantly reduces the attribute reduction in time cost.

4.1.2. Comparison of Five Algorithms in Terms of Classification Accuracy

Here we use the reduct computed by the algorithms HFS_DRS, FS_DRS, FS_FDRS, HAR_MDSI and IAR_tMDSI to test the classification performance, and four classifiers, namely, Naive Bayes (NB), Support Vector Machines (SVM), K-nearest Neighbor (KNN, K=3) and Classification and Regression Tree (CART) are used to test the classification accuracy of the resulting reducts and the original attribute set on the basis of the 10-fold cross-validation, respectively, and the results are shown in Tables 3 and 4.

In each table, in each row we denote in bold the value with the highest classification accuracy, observing that the reducts computed by the Algorithm IAR_tMDSI maintain or even improve the classification accuracy of the dataset compared to the other algorithms, as is well represented by their average values. Also, in terms of classifiers, the Algorithm performs particularly well in SVM, KNN and CART, as reflected by the number of datasets with the highest classification accuracy. It is to be particularly noted that for the Datasets Arrhy and Mice, the reducts computed by the Algorithm IAR_tMDSI are significantly higher than the reducts computed by the other algorithms in terms of classification accuracy. thus the effectiveness of the Algorithm IAR_tMDSI is further demonstrated.

Previously we quantitatively analyzed the effectiveness of different algorithms, however, in the actual classification process, different factors will lead to certain drawings. Even after using 10-fold cross-validation, this is not consistent with effective proof. Therefore, here we further conduct a qualitative analysis of the data.

We use Matlab to express the values of these four tables graphically, as shown in the Figure 1. Among them, the abscissa represents the restoration of the original attribute set and the calculation results of different algorithms, which are replaced by the corresponding algorithm name for the convenience of display; the ordinate is the data set serial number, which corresponds to the data set in Table 4 from top to bottom; the rectangle The color bar represents the color at the corresponding accuracy. The lighter the color, the higher the accuracy; the gray ball is used to represent the standard deviation. Since the range of the standard deviation is [0,1], the interval is divided into 20 parts, so that the range represents the corresponding standard deviation, this processing method is to limit the bubble size.

It can be observed from the four pictures that the y-axis bubble colors under the Algorithm IAR_tMDSI are all similar to the y-axis bubble colors of the reducts computed by other algorithms, and even some are lighter. At the same time, the bubbles are also relatively small, indicating that the Algorithm IAR_tMDSI is ensuring While the accuracy is similar, it also has good stability. Therefore, the effectiveness of Algorithm IAR_tMDSI is proved.

4.2. Efficiency Evaluations

In this section, we evaluate the efficiency of the Algorithm IAR_tMDSI by comparing the computation time and speed-up ratio of the Algorithm IAR_tMDSI with the other four algorithms (HAR_MDSI, HFS_DRS, FS_DRS and FS_FDRS). For each dataset in Table 4, it is first randomly shuffled by row, and then the data except the last column is randomly shuffled by column, and then the first 50% of the objects and attributes are selected as the initial dataset, and then the remaining data is added to the initial dataset at different proportions to construct different test sets. It should be noted that since 50% of the two dimensions of the dataset are selected at the same time, the remaining dataset size is actually 75% of the size of the full dataset, as shown in the Figure 4. Therefore, in operation, we can obtain six new datasets of different sizes according to the proportions of the first 50%, 60%, 70%, 80%, 90% and 100% of the full dataset for evaluating the performance of the algorithm.

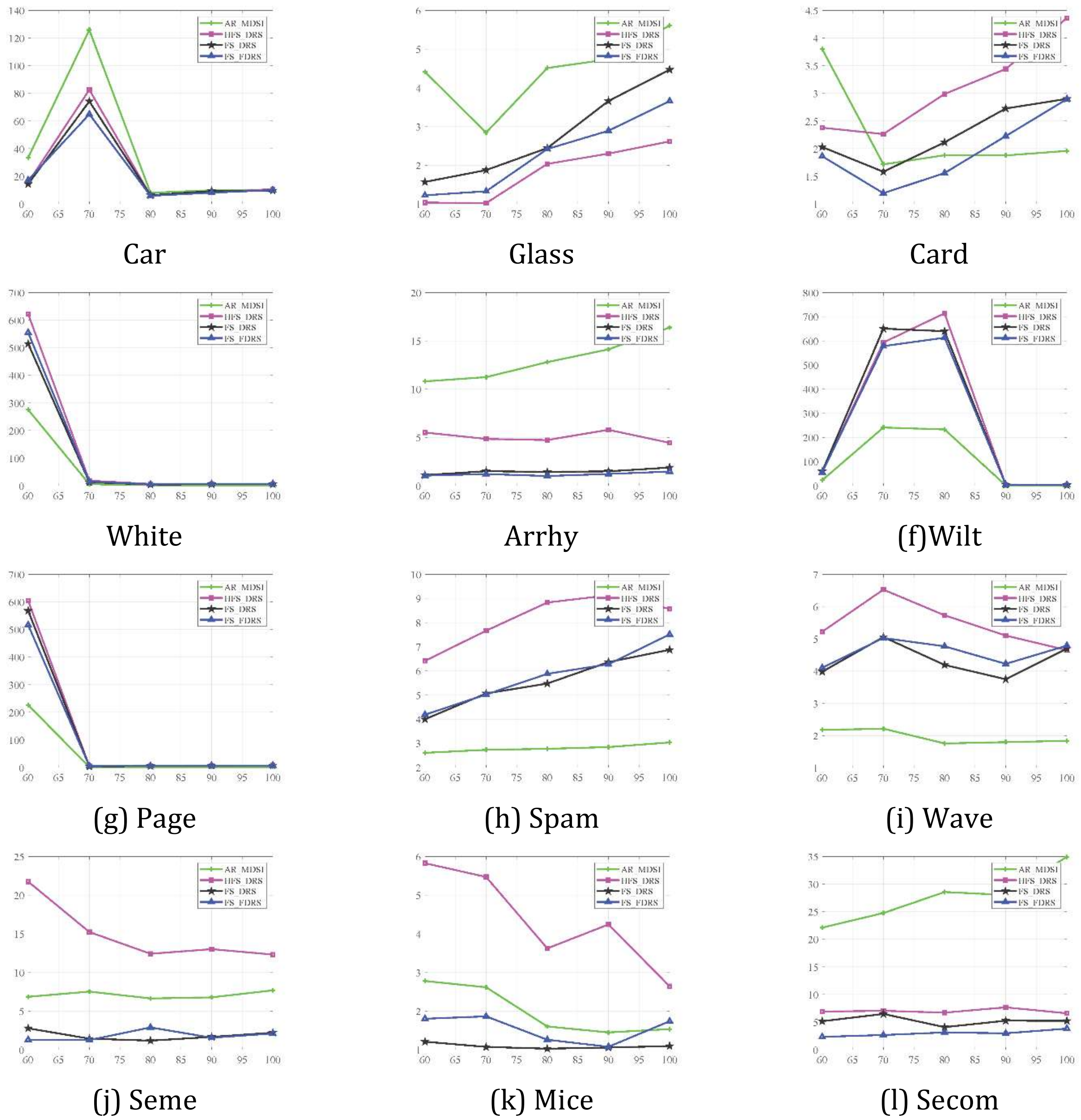
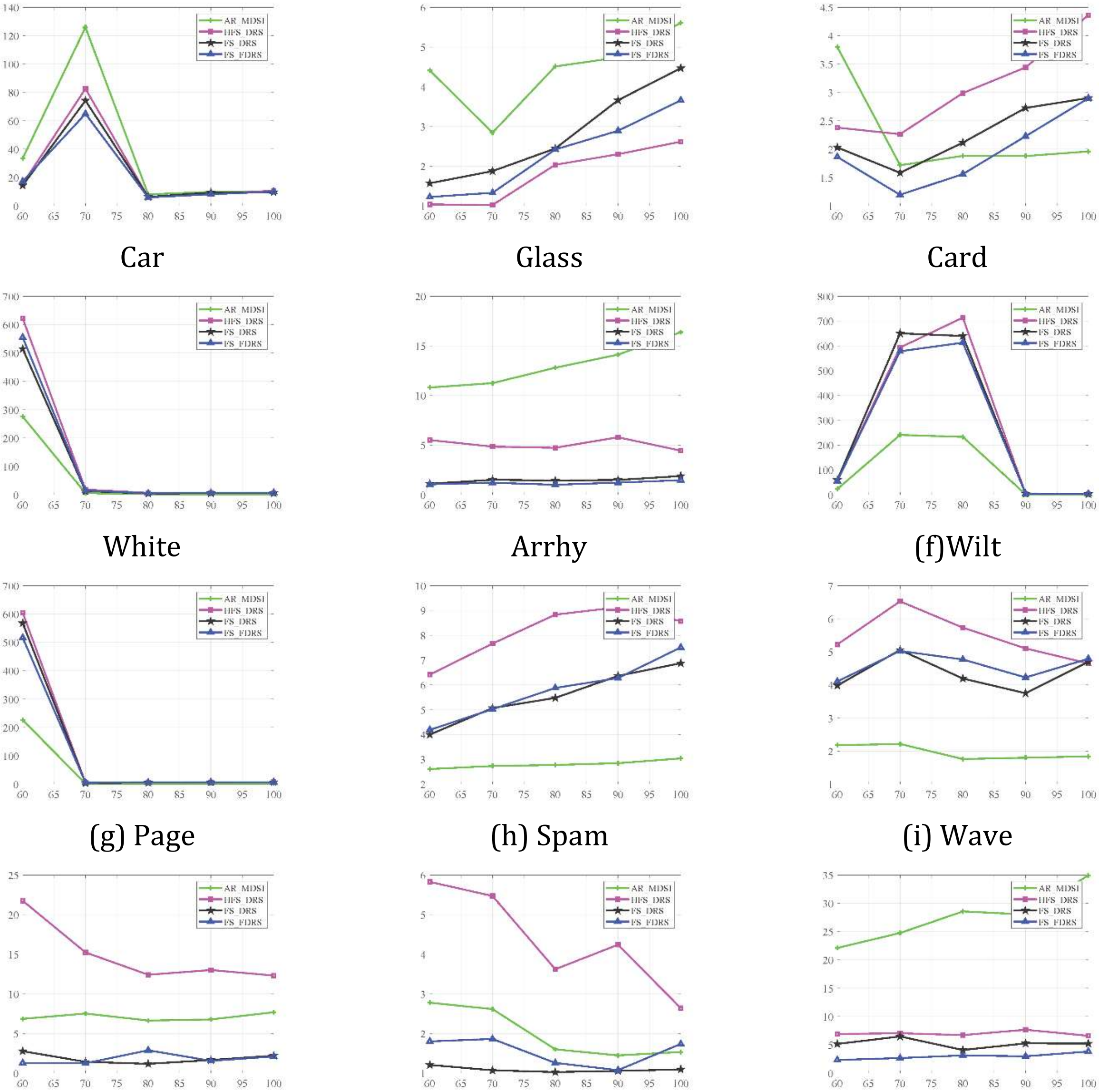


Figure 2. Comparison of the computation time of algorithm IAR tMDSI and other attribute reduction algorithms when adding objects and attributes of different sizes simultaneously



Car Glass Card

White Arrhy (f)Wilt

(g) Page (h) Spam (i) Wave

(j) Seme (k) Mice (l) Secom

Figure 3. The speed-up ratio of different algorithms when adding multiple objects and attributes.

4.2.1. Comparison of the IAR tMDSI Algorithm with the Other Four Algorithms in Terms of Computation Time

For the Algorithm IAR_tMDSI, 50% of the initial data set is needed because the reduction of the initial data needs to be calculated at the beginning, while for other algorithms, this is not necessary because they have to start from scratch each time. The remaining five data sets of different sizes for each data set are used to determine the running time of different algorithms when adding objects and attributes of different sizes at the same time. Figure 2 shows the detailed time curves of different reduction algorithms as the size of the data set increases, where the horizontal axis represents the size of the objects and attributes added at the same time, and the vertical axis represents the running time of the algorithm. The curves of different algorithms are marked in the upper right corner of each figure.

It is obvious that in all the subgraphs of Figure 2, Algorithm IAR_tMDSI has the smallest curve variation. At the same time, its value is also the smallest among all the curves. As the proportion of added data increases, the growth trend of the running timeline of Algorithm IAR_tMDSI is slower than that of other algorithms. It can be concluded that Algorithm IAR_tMDSI can perform attribute simplification in a shorter time.

4.2.2. Evaluation of the Efficiency of Algorithm IAR tMDSI in Terms of Speed-up Ratio

Note that the speed-up ratio requires at least two algorithms, one of which plays the role of the control group. It is calculated as follows, and the resulting method, for the Algorithm IAR_tMDSI, the results of the acceleration ratios of the different algorithms are shown in Figure 3.

As shown in Figure 3, in all sub-graphs, we can see that the Algorithm IAR_tMDSI runs at least twice as fast as other algorithms. On some datasets, IAR_tMDSI performs even better. For example, on Car, Arrhy, Spam, Wave, Seme, Mice, and Secom datasets, IAR_tMDSI is about ten times faster than other algorithms, and even tens or hundreds of times faster in some datasets. The experimental results once again prove that the Algorithm IAR_tMDSI has superior performance compared to other algorithms.

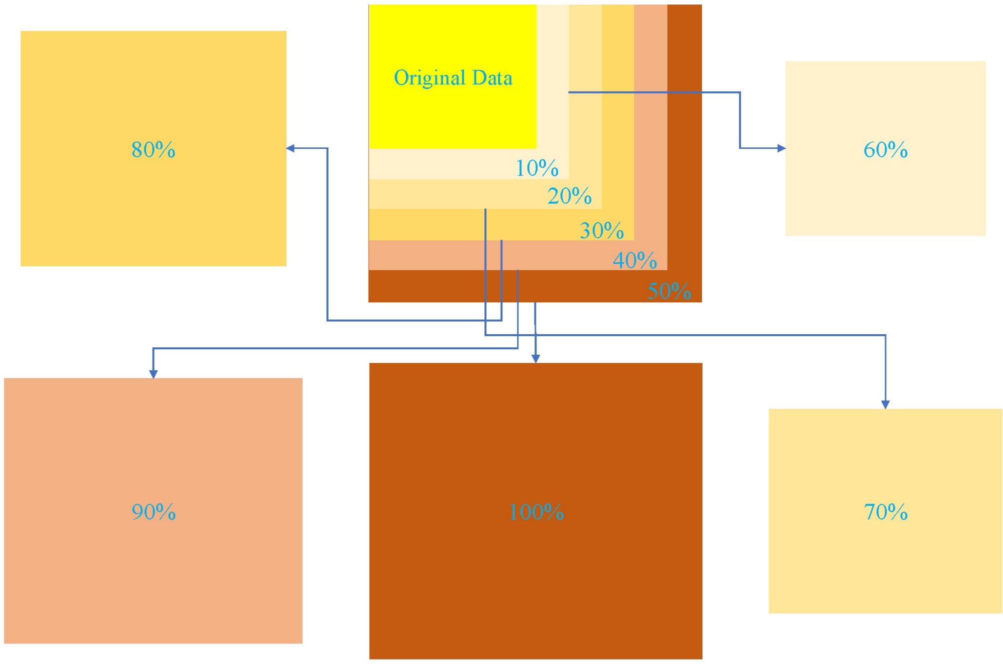


Figure 4. Description of adding objects and attributes of different scales at the same time

5. Conclusion

The DRSA considers preference relations on top of equivalence relations and can effectively handle information in the prioritization domain. This paper explores the key research area of continuous attribute reduction for ordered data, with a focus on identifying preference dynamics between attributes and decisions as well as dealing with the constant addition of new data. Recognizing the inefficiencies and limitations of existing methods, especially those constrained to one-dimensional variations, the paper introduces a pioneering approach to continuous attribute reduction designed to efficiently deal with ordered data exhibiting multi-dimensional variations.

The article employs self-information as an uncertainty metric that not only demonstrates a nuanced understanding of the intricacies involved, but also takes into account upper and lower approximation bounds. The strategic choice of diagonal matrix structure helps to simplify the matrix computation, thus saving a lot of time. In addition, the introduction of more efficient

ISSN: 2664-9640

upper and lower approximation computation methods as well as multi-threaded operations further improves the computational efficiency of the algorithm.

Finally, by comparing different datasets with different algorithms, the proposed algorithm contributes significantly to the effectiveness and efficiency of attribute reduction for ordered data. The results highlight the ability of the algorithm in navigating complex multi-dimensional variations, which allows it to become an important tool for data mining of ordered data.

References

[1] Z. Pawlak, Rough sets, International journal of computer & information sciences 11 (1982) 341–356.

[2] J. Xie, B. Q. Hu, H. Jiang, A novel method to attribute reduction based on weighted neighborhood probabilistic rough sets, International Journal of Approximate Reasoning 144 (2022) 1–17.

[3] M. Hu, E. C. Tsang, Y. Guo, D. Chen, W. Xu, A novel approach to attribute reduction based on weighted neighborhood rough sets, Knowledge- Based Systems 220 (2021) 106908.

[4] J. Dai, H. Hu, W.-Z. Wu, Y. Qian, D. Huang, Maximal-discernibility-pair-based approach to attribute reduction in fuzzy rough sets, IEEE Transactions on Fuzzy Systems 26 (4) (2017) 2174–2187.

[5] C. Wang, M. Shao, B. Sun, Q. Hu, An improved attribute reduction scheme with covering based rough sets, Applied Soft Computing 26 (2015) 235–243.

[6] C. Wang, Y. Huang, M. Shao, X. Fan, Fuzzy rough set-based attribute reduction using distance measures, Knowledge-Based Systems 164 (2019) 205–212.

[7] Z. Yuan, H. Chen, T. Li, Z. Yu, B. Sang, C. Luo, Unsupervised attribute reduction for mixed data based on fuzzy rough sets, Information Sciences 572 (2021) 67–87.

[8] T. Sheeja, A. S. Kuriakose, A novel feature selection method using fuzzy rough sets, Computers in Industry 97 (2018) 111–116.

[9] E. C. Tsang, D. Chen, D. S. Yeung, X.-Z. Wang, J. W. Lee, Attributes reduction using fuzzy rough sets, IEEE Transactions on Fuzzy systems 16 (5) (2008) 1130–1141.

[10] Q. Hu, D. Yu, J. Liu, C. Wu, Neighborhood rough set based heterogeneous feature subset selection, Information Sciences 178 (18) (2008) 3577–3594.

[11] Q. Hu, D. Yu, M. Guo, Fuzzy preference based rough sets, Information Sciences 180 (10) (2010) 2003–2022.

[12] H. Chen, T. Li, Y. Cai, C. Luo, H. Fujita, Parallel attribute reduction in dominance-based neighborhood rough set, Information Sciences 373 (2016) 351–368.

[13] S. Yang, H. Zhang, B. De Baets, M. Jah, G. Shi, Quantitative dominance-based neighborhood rough sets via fuzzy preference relations, IEEE Transactions on Fuzzy Systems 29 (3) (2019) 515–529.

[14] J. Dai, Q. Hu, H. Hu, D. Huang, Neighbor inconsistent pair selection for attribute reduction by rough set approach, IEEE Transactions on Fuzzy Systems 26 (2) (2017) 937–950.

[15] W. S. Du, B. Q. Hu, A fast heuristic attribute reduction approach to ordered decision systems, European Journal of Operational Research 264 (2) (2018) 440–452.

[16] Y. Lin, Y. Li, C. Wang, J. Chen, Attribute reduction for multi-label learning with fuzzy rough set, Knowledge-Based Systems 152 (2018) 51–61.

[17] W. Ding, C.-T. Lin, Z. Cao, Shared nearest-neighbor quantum game-based attribute reduction with hierarchical coevolutionary spark and its application in consistent segmentation of neonatal cerebral cortical surfaces, IEEE Transactions on Neural Networks and Learning Systems 30 (7) (2018) 2013–2027.

[18] W. Ding, W. Pedrycz, I. Triguero, Z. Cao, C.-T. Lin, Multigranulation supertrust model for attribute reduction, IEEE Transactions on Fuzzy Systems 29 (6) (2020) 1395–1408.

[19] Y. Chen, K. Liu, J. Song, H. Fujita, X. Yang, Y. Qian, Attribute group for attribute reduction, Information Sciences 535 (2020) 64–80.

[20] Y. Liu, S. Zhao, H. Chen, C. Li, Y. Lu, Fuzzy rough incremental attribute reduction applying dependency measures, in: Web and Big Data: First International Joint Conference, APWeb-WAIM 2017, Beijing, China, July 7–9, 2017, Proceedings, Part I 1, Springer, 2017, pp. 484–492.

[21] C. Wang, M. Shao, Q. He, Y. Qian, Y. Qi, Feature subset selection based on fuzzy neighborhood rough sets, Knowledge-Based Systems 111 (2016) 173–179.

[22] X. Zhang, C. Mei, D. Chen, J. Li, Feature selection in mixed data: A method using a novel fuzzy rough set-based information entropy, Pattern Recognition 56 (2016) 1–15.

[23] P. Wang, L. Qu, Q. Zhang, Information entropy based attribute reduction for incomplete heterogeneous data, Journal of Intelligent & Fuzzy Systems 43 (1) (2022) 219–236.

[24] R. Kohavi, G. H. John, Wrappers for feature subset selection, Artificial Intelligence 97 (1-2) (1997) 273–324.

[25] M. Robnik-Sˇikonja, I. Kononenko, Theoretical and empirical analysis of relieff and rrelieff, Machine Learning 53 (2003) 23–69.

[26] C.-C. Chen, S.-T. Li, Credit rating with a monotonicity-constrained support vector machine model, Expert Systems with Applications 41 (16) (2014) 7235–7247.

[27] J.-R. Cano, P. A. Gutie´rrez, B. Krawczyk, M. Woz´niak, S. Garc´ıa, Monotonic classification: An overview on algorithms, performance measures and data sets, Neurocomputing 341 (2019) 168–182.

[28] S. Chakhar, A. Ishizaka, A. Labib, I. Saad, Dominance-based rough set approach for group decisions, European Journal of Operational Research 251 (1) (2016) 206–224.

[29] L. Dong, D. Chen, Incremental attribute reduction with rough set for dynamic datasets with simultaneously increasing samples and attributes, International Journal of Machine Learning and Cybernetics 11 (2020) 1339–1355.

[30] Y. Yang, D. Chen, X. Zhang, Z. Ji, Covering rough set-based incremental feature selection for mixed decision system, Soft Computing 26 (6) (2022) 2651–2669.

[31] W. Ding, T. Qin, X. Shen, H. Ju, H. Wang, J. Huang, M. Li, Parallel incremental efficient attribute reduction algorithm based on attribute tree, Information Sciences 610 (2022) 1102–1121.

[32] X. Yang, M. Li, H. Fujita, D. Liu, T. Li, Incremental rough reduction with stable attribute group, Information Sciences 589 (2022) 283–299.

[33] B. Sang, H. Chen, L. Yang, D. Zhou, T. Li, W. Xu, Incremental attribute reduction approaches for ordered data with time-evolving objects, Knowledge-Based Systems 212 (2021) 106583.

[34] X. Zhang, C. Mei, D. Chen, Y. Yang, J. Li, Active incremental feature selection using a fuzzy-rough-set-based information entropy, IEEE Transactions on Fuzzy Systems 28 (5) (2019) 901–915.

[35] Y. Yang, D. Chen, H. Wang, X. Wang, Incremental perspective for feature selection based on fuzzy rough sets, IEEE Transactions on Fuzzy Systems 26 (3) (2017) 1257–1273.

[36] Y. Yang, S. Song, D. Chen, X. Zhang, Discernible neighborhood counting based incremental feature selection for heterogeneous data, Inter- national Journal of Machine Learning and Cybernetics 11 (2020) 1115–1127.

[37] W. Shu, W. Qian, Y. Xie, Incremental feature selection for dynamic hybrid data using neighborhood rough set, Knowledge-Based Systems 194 (2020) 105516.

[38] D. Chen, L. Dong, J. Mi, Incremental mechanism of attribute reduction based on discernible relations for dynamically increasing attribute, Soft Computing 24 (2020) 321–332.

[39] G. Lang, D. Miao, M. Cai, Z. Zhang, Incremental approaches for updating reducts in dynamic covering information systems, Knowledge- Based Systems 134 (2017) 85–104.

[40] W. Wei, X. Wu, J. Liang, J. Cui, Y. Sun, Discernibility matrix based incremental attribute reduction for dynamic data, Knowledge-Based Systems 140 (2018) 142–157.

[41] W. Wei, P. Song, J. Liang, X. Wu, Accelerating incremental attribute reduction algorithm by compacting a decision table, International Journal of Machine Learning and Cybernetics 10 (2019) 2355–2373.

[42] M. Cai, G. Lang, H. Fujita, Z. Li, T. Yang, Incremental approaches to updating reducts under dynamic covering granularity, Knowledge-Based Systems 172 (2019) 130–140.

[43] S. Wang, T. Li, C. Luo, H. Fujita, Efficient updating rough approximations with multi-dimensional variation of ordered data, Information Sciences 372 (2016) 690–708.

[44] S. Wang, T. Li, C. Luo, H. Chen, H. Fujita, Domain-wise approaches for updating approximations with multi-dimensional variation of ordered information systems, Information Sciences 478 (2019) 100–124.

[45] S. Wang, T. Li, C. Luo, J. Hu, H. Fujita, T. Huang, A novel approach for efficient updating approximations in dynamic ordered information systems, Information Sciences 507 (2020) 197–219.

[46] H. Chen, T. Li, C. Luo, S.-J. Horng, G. Wang, A decision-theoretic rough set approach for dynamic data mining, IEEE Transactions on Fuzzy Systems 23 (6) (2015) 1958–1970.

[47] S. Greco, B. Matarazzo, R. Slowinski, Rough sets theory for multicriteria decision analysis, European Journal of Operational Research 129 (1) (2001) 1–47.

[48] S. Greco, B. Matarazzo, R. Slowinski, Rough approximation by dominance relations, International Journal of Intelligent Systems 17 (2) (2002) 153–171.

[49] C. E. Shannon, A mathematical theory of communication, ACM SIGMOBILE Mobile Computing and Communications Review 5 (1) (2001) 3–55.

[50] L. Yang, K. Qin, B. Sang, C. Fu, A novel incremental attribute reduction by using quantitative dominance-based neighborhood self- information, Knowledge-Based Systems 261 (2023) 110200.

[51] S. Li, T. Li, D. Liu, Dynamic maintenance of approximations in dominance-based rough set approach under the variation of the object set, International Journal of Intelligent Systems 28 (8) (2013) 729–751.