Applied Intelligence
https://doi.org/10.1007/s10489-024-05411-3

Incremental feature selection approach to multi-dimensional variation based on matrix dominance conditional entropy for ordered data set

Weihua Xu[1] · Yifei Yang[1] · Yi Ding[1] · Xiyang Chen[2] · Xiaofang Lv[3]

Accepted: 20 March 2024

Abstract

Rough set theory is a mathematical tool widely employed in various fields to handle uncertainty. Feature selection, as an essential and independent research area within rough set theory, aims to identify a small subset of important features by eliminating irrelevant, redundant, or noisy ones. In human life, data characteristics constantly change over time and other factors, resulting in ordered datasets with varying features. However, existing feature extraction methods are not suitable for handling such datasets since they do not consider previous reduction results when features change and need to be recomputed, leading to significant time consumption. To address this issue, the incremental attribute reduction algorithm utilizes prior reduction results effectively reducing computation time. Motivated by this approach, this paper investigates incremental feature selection algorithms for ordered datasets with changing features. Firstly, we discuss the dominant matrix and the dominance conditional entropy while introducing update principles for the new dominant matrix and dominance diagonal matrix when features change. Subsequently, we propose two incremental feature selection algorithms for adding (IFS-A) or deleting (IFS-D) features in ordered data set. Additionally, nine UCI datasets are utilized to evaluate the performance of our proposed algorithm. The experimental results validate that the average classification accuracy of IFS-A and IFS-D under four classifiers on twelve datasets is 82.05% and 80.75%, which increases by 5.48% and 3.68% respectively compared with the original data.

Keywords Conditional entropy · Dominance matrix · Feature selection · Ordered data set · Rough set

1 Introduction

By integrating rough set theory (RST), feature selection effectively achieves data dimensionality reduction and offers precise semantic interpretation for the results, garnering continuous attention from numerous scholars [1–4]. Feature selection, also known as attribute reduction, is to eliminate redundant attributes in the data by using the constraints of specific metrics to improve the performance of subsequent learning algorithms. In life, data sets often change with time and other variables, and these changes are called dynamic data sets. Feature selection algorithms for dynamic data sets usually adopt incremental methods [5–9]. The incremental approach has attracted a lot of interest because it effectively utilizes existing reduction results and thus saves a lot of time and space overhead.

✉ Yi Ding
drying20@163.com

Extended author information available on the last page of the article

With the emergence of the big data and complex data era, there has been an increase in the complexity and diversity of data, leading to continuous advancements and innovations in feature selection methods. Numerous feature selection algorithms have been proposed by scholars. Recently, deep learning [10–12] has also been applied to feature selection problems. Zhao et al. [13] introduced a feature selection algorithm based on multi-dimensional deep neural networks (DNNs), combined with relatively rare population loops. Semwal et al. [14] described a highly robust feature extraction method that successfully addressed classification problems. Chen et al. [15] developed a target feature prediction method based on EEG analysis, which found successful applications across various domains such as economics field [16], remote control systems [17], and traffic field [18]. Inspired by Darwin's theory of evolution, evolutionary algorithms [19] simulate natural evolution behaviors to solve optimization problems including feature selection tasks. Nag et al. [20] investigated a simplified classifier approach for feature extraction and selection methods using multi-objective

Published online: 10 April 2024

genetic programming. Labani et al. [21] proposed and validated a text features-based multi-objective genetic algorithm incorporating relative criteria. Ma et al. [22] presented a classification-oriented genetic programming-based method for feature selection. Das et al. [23] described an algorithm that simultaneously addresses two objectives through joint feature extraction. Li et al. [24] demonstrated an approach that combines multi-objective features with genetic algorithms.

In 1982, Polish mathematician Z. Pawlak proposed the Rough Set Theory (RST) [25], which serves as a mathematical tool to handle uncertainty alongside probability theory, fuzzy set theory, and evidence theory. Simultaneously, RST provides an important theoretical foundation for feature selection problems [26–28]. Feature selection aims to eliminate redundant attributes and select relevant attribute information in order to achieve efficient learning tasks. Currently, numerous scholars have proposed various feature selection algorithms based on knowledge partitioning, closeness measures, mutual information, granularity analysis, among others. However, classical rough set theory is not suitable for handling ordered data. Therefore, the Greco team introduced a dominance relation-based feature extraction method [29] and applied it in multidimensional prediction scenarios [30]. Additionally, other models such as the monotone variable consistency rough set method [31], rough set model based on stochastic dominance [32], rough set model based on soft dominance [33], and frequently described rough set model [34] have been developed by scholars.

The above algorithm is effective when dealing with static data, but it is complex and redundant when reconstructing feature selection space when facing complex and changeable real data. Dynamic incremental learning is an effective approach for efficiently acquiring new information from dynamic datasets by leveraging prior knowledge. Over time, numerous scholars have proposed a plethora of attribute reduction algorithms for incremental learning, which can be broadly categorized into object-oriented [35–40], attribute-oriented [41–43], and attribute-value-oriented [44–49] approaches.

The change of data set samples prompted Liang et al. [35] to propose a dynamic update algorithm based on information value. Zhang et al. [36] demonstrated an active selection algorithm that dynamically selects sample features. Yang et al. [37, 38] developed a dynamic sample selection method based on the principle of active learning and also investigated the feature extraction method for variable heterogeneous data. Shu et al. [39] described a dynamic feature selection algorithm that integrates multiple types of information. Ye et al. [40] introduced and validated a dynamic algorithm based on matrix pseudo-values. Considering the change in attribute values within data set, Chen et al. [41] proposed an incremental feature selection method based on identifiable relations to dynamically add attributes. Wang et al. [42] devised an entropy-based algorithm to handle dynamically changing datasets effectively. Zeng et al. [43] explored an incremental attribute reduction method for mixed data using fuzzy rough sets. In order to handle the fluctuations in attribute values within a data set, Wang et al. [44] proposed an algorithm based on representative entropy. Wei et al. [45, 46] introduced a feature selection method that utilizes the discriminant matrix and further developed an accelerated incremental algorithm based on compressed decision table techniques. Cai et al. [47] presented a dynamic feature selection algorithm based on coarse and fine granularity approaches. Building upon this foundation, Dong and Chen [48] devised a novel RST-based incremental feature selection algorithm that simultaneously incorporates sample and attribute additions. Jing et al. [49] put forth an incremental approach for calculating decision table reductions amidst changes in entities and properties over the course of time.

By analyzing the aforementioned algorithms, including their application scenarios and experimental demonstration datasets, as well as reproducing and testing them on ordered data sets, it was discovered that algorithms which alter attribute characteristics are not used on ordered data sets. In light of this, there is an urgent need to propose an algorithm for changing attributes in ordered data sets. The proposal by Shannon [50] has led to extensive research on information entropy as a measure of uncertainty. Subsequent studies have applied this concept to data with sequential relationships, such as the ascending and descending conditional entropy proposed by Hu et al. [51]. Meanwhile, in discernibility-based rough set approach (DRSA) [52], the correlation between objects establishes an antisymmetric preference order, resulting in spatial irregularity. However, studying DRSA using collections becomes cumbersome and complex, particularly for non-static datasets. Therefore, this paper presents a multi-dimensional change incremental feature selection method for ordered datasets based on matrix dominant conditional entropy, the algorithms we propose in the following paper mainly uses the conditional entropy under the advantageous condition to construct the matrix efficiently and overcome the existing difficulties. We investigate the variation of matrix dominant conditional entropy as the number of attributes increases and decreases, respectively. Additionally, we suggest a corresponding feature selection algorithm to identify the important attributes. The main contributions of this paper are as follows: (1) Introducing a matrix-based technique for computing dominance conditional entropy in ordered data set and demonstrating its associated properties; (2) Proposing two dynamic incremental feature extraction algorithms, namely IFS-A and IFS-D; (3) Conducting experiments to verify the efficacy of our suggested approach, employing a set of nine UCI datasets.

The documentation is structured as follows: Section 2 introduces the fundamentals, including dominance relations and information entropy. In Section 3, we present the calculation methods for the dominant matrix, dominant diagonal matrix, and dominant conditional entropy of the matrix. Based on these methods, a static feature extraction algorithm is proposed. Section 4 investigates and proposes two incremental feature selection algorithms (IFS-A and IFS-D) in terms of attribute increase and decrease. Section 5 analyzes experimental results from multiple perspectives such as classification accuracy and time to demonstrate the effectiveness of our proposed method. Finally, we finalize our research and offer a glimpse into forthcoming obstacles in Section 6.

2 Preliminaries

In this section, we provide a comprehensive overview of the fundamental principles underlying DRSA.

2.1 Dominance relation

Definition 2.1 ([28]) Given a 4-tuple $I = (O, C, V, h)$, where O is a non-empty finite set of objects; C is a non-empty finite set of attributes; $V = \bigcup_{c \in C} V_c$, V_c is the domain of attribute c; $h : O \times C \to V$ is the information function with $h(o, c) \in V_c$, $\forall c \in C$ and $\forall o \in O$. If there is an ascending or descending sequence between any of the attributes, it is an ordered data set (ODS), which is denoted by $I^{\succeq} = (O, C, V, h)$.

Definition 2.2 ([32]) Let $I^{\succeq} = (O, C, V, h)$ be an ODS, $\forall A \subseteq C$, $A \neq \emptyset$, the dominance relation R_A is defined as follows:

$$R_A = \{(o_1, o_2) \in O \times O | h(o_1, c) \geq h(o_2, c), \forall c \in A\}. \quad (1)$$

Property 2.1 ([32]) For a dominance relation in an ordered data set R_A, we have:

(1) Reflexive: $\forall o \in O$, then $o R_A o$;
(2) Non-symmetric: $\forall o_1, o_2 \in O$, if there is $o_1 R_A o_2$, then we can't have $o_2 R_A o_1$;
(3) Transitive: $\forall o_1, o_2, o_3 \in O$, if $o_1 R_A o_2$ and $o_2 R_A o_3$, then $o_1 R_A o_3$.

Definition 2.3 ([32]) Given an ODS $I^{\succeq} = (O, C, V, h)$, $\forall A \subseteq C$, $A \neq \emptyset$, the two relational sets of o_1 are called A-dominating sets and A-dominated sets, respectively, and they are defined as follows:

$$R_A^+(o_1) = \{o_2 \in U | o_2 R_A o_1\}; \quad (2)$$

$$R_A^-(o_1) = \{o_2 \in U | o_1 R_A o_2\}. \quad (3)$$

Property 2.2 ([32]) For any $A, B \subseteq C$ and $\forall o \in O$, the subsequent characteristics are valid.

(1) Let $A \subseteq B$, then $R_B^+(o) \subseteq R_A^+(o)$ and $R_B^-(o) \subseteq R_A^-(o)$;
(2) $R_A^+(o) \cap R_B^+(o) = R_{A \cup B}^+(o)$ and $R_A^-(o) \cap R_B^-(o) = R_{A \cup B}^-(o)$.

Example 1 Table 1 shows the English and math scores of the three students and the corresponding ratings, where c_1 and c_2 represent English and math, respectively, and o_1, o_2, o_3 represent three students. Then $P = \{c_1, c_2\}$, $U = \{o_1, o_2, o_3\}$, D_P is a dominance relation. The dominant order relation for decision d is pass $\prec$ good $\prec$ perfect.

2.2 Entropy theory

In this section, we will provide a comprehensive overview of dominance entropy and introduce the attribute reduction method known as ordered decision data set (ODDS) for enhanced academic understanding.

Definition 2.5 ([51]) Let $I^{\succeq} = (O, C \cup \{d\}, V, h)$ be an ODDS, for $\forall A, B \subseteq C$, the dominance information entropy (DIE) about A and $A \cup B$ are defined as follows:

$$DE_A^{\succeq}(O) = -\frac{1}{|O|} \sum_{i=1}^{n} \log \frac{|R_A^+(o_i)|}{|O|}. \quad (4)$$

$$DE_{A \cup B}^{\succeq}(O) = -\frac{1}{|O|} \sum_{i=1}^{n} \log \frac{|R_A^+(o_i) \cap R_B^+(o_i)|}{|O|} = -\frac{1}{|O|} \sum_{i=1}^{n} \log \frac{|R_{A \cup B}^+(o_i)|}{|O|}. \quad (5)$$

Definition 2.6 ([51]) Let $I^{\succeq} = (O, C \cup \{d\}, V, h)$ be an ODDS, for $\forall A \subseteq C$, the dominance conditional entropy (DCE) is defined as follows:

$$DE_{d|A}^{\succeq}(O) = -\frac{1}{|U|} \sum_{i=1}^{n} \log \frac{|R_d^+(o_i) \cap R_A^+(o_i)|}{|R_A^+(o_i)|} = -\frac{1}{|O|} \sum_{i=1}^{n} \log \frac{|R_{\{d\} \cup A}^+(o_i)|}{|R_A^+(o_i)|}. \quad (6)$$

According to Definition 2.6, the hierarchical relationship reflected by DCE is manifested through the production of

Table 1 The grade table of the subject

	o_1	o_2	o_3
c_1	98	87	78
c_2	95	88	65
d	perfect	good	pass

consistent objects, which are closely associated with both the set of information condition attributes and decision attributes provided.

In the process of feature extraction, the importance of a certain feature can be evaluated and the importance between different features can be explored.

Definition 2.7 ([6]) Let $I^{\succeq} = (O, C \cup \{d\}, V, h)$ be an ODDS, for $\forall A \subseteq C$ and $\forall c \in A$, the internal significance measures of the dominance conditional entropy is defined as follows:

$$\phi(c, A, d) = DE^{\succeq}_{d|A-\{c\}}(O) - DE^{\succeq}_{d|A}(O). \tag{7}$$

Definition 2.8 ([6]) Let $I^{\succeq} = (O, C \cup \{d\}, V, h)$ be an ODDS, for $\forall B \subseteq C$ and $\forall c \in (C - B)$, the external significance measures of the dominance conditional entropy is defined as follows:

$$\psi(c, B, d) = DE^{\succeq}_{d|B}(O) - DE^{\succeq}_{d|B\cup\{c\}}(O). \tag{8}$$

The above two definitions are used to select the important condition attributes from the attribute set and to select the important attributes after removing an attribute. If $\phi(c, A, d) > 0$, attribute c holds greater importance.

Definition 2.9 Given $I^{\succeq} = (O, C \cup \{d\}, V, h)$ is an ODDS, for $\forall A \subseteq C$, the attribute set A is a reduct subset of $I^{\succeq}$ if the following holds:

(1) $DE^{\succeq}_{d|A}(O) = DE^{\succeq}_{d|C}(O)$;
(2) $\forall c \in A$, $DE^{\succeq}_{d|A-\{c\}}(O) \neq DE^{\succeq}_{d|A}(O)$.

The condition mentioned above (1) is utilized to ensure that the classification capability of the chosen subset of attributes is comparable to that of the origin feature set. The purpose of condition (2) is to progressively eliminating unnecessary attributes from the chosen subset, thereby ensuring its non-redundancy and indispensability for each attribute in the set. As a result, if both conditions mentioned earlier are satisfied by the selected attribute subset, it can be referred to as a reduction; otherwise, it would be considered a relative reduction.

3 Static feature selection method based on matrix dominance conditional entropy

In this section, we first define the dominant matrix of ODDS. Then, the matrix dominance conditional entropy (MDCE) and the static feature selection algorithm (SFS) based on MDCE are proposed.

3.1 Matrix dominant conditional entropy

Definition 3.1 Let $I^{\succeq} = (O, C \cup \{d\}, V, h)$ be an ODDS, for $\forall A \subseteq C$, R_A is a dominance relation, the dominant matrix is described like $\mathbb{X}_O^{\succeq A} = \left[x^A_{(i,j)}\right]_{n\times n}$, where

$$x^A_{(i,j)} = \begin{cases} 1, & o_j R_A o_i; \\ 0, & \text{otherwise}. \end{cases} \tag{9}$$

Property 3.1 $\mathbb{X}_O^{\succeq A} = \left[x^A_{(i,j)}\right]_{n\times n}$ is a dominant matrix, characterized by the following properties.

(1) $x^A_{(i,i)} = 1$, where $i \in \{1, 2, \ldots, n\}$;
(2) $\sum_{j=1}^n x^A_{(i,j)} = \left|R^+_A(o_i)\right|$ and $\sum_{i=1}^n x^A_{(i,j)} = \left|R^-_A(o_j)\right|$, where $i, j \in \{1, 2, \ldots, n\}$.

Definition 3.2 Let $I^{\succeq} = (O, C \cup \{d\}, V, h)$ be an ODDS, for $\forall A, B \subseteq C$, and given two dominant matrices $\mathbb{X}_O^{\succeq A} = \left[x^A_{(i,j)}\right]_{n\times n}$ and $\mathbb{X}_O^{\succeq B} = \left[x^B_{(i,j)}\right]_{n\times n}$. We define $\mathbb{X}_O^{\succeq A} \cap \mathbb{X}_O^{\succeq B}$ as follows:

$$\mathbb{X}_O^{\succeq A} \cap \mathbb{X}_O^{\succeq B} = \left[x^A_{(i,j)} \times x^B_{(i,j)}\right]_{n\times n}. \tag{10}$$

Above formula provides a method for obtaining new dominant matrices $\mathbb{X}_O^{\succeq A}$ and $\mathbb{X}_O^{\succeq B}$, which is of practical significance as it allows the simultaneous acquisition of dominant matrices for attribute sets A and B.

Proposition 3.1 *Let $I^{\succeq} = (O, C \cup \{d\}, V, h)$ be an ODDS, for $\forall A, B \subseteq C$, then $\mathbb{X}_O^{\succeq A\cup B} = \mathbb{X}_O^{\succeq A} \cap \mathbb{X}_O^{\succeq B}$ establishes.*

Proof According to Definition 3.1, $\mathbb{X}_O^{\succeq A\cup B} = \left[x^{A\cup B}_{(i,j)}\right]_{n\times n}$. If $x^{A\cup B}_{(i,j)} = 1$, $o_j \in R^+_{A\cup B}(o_i)$. Then, we have $o_j \in R^+_A(o_i)$ and $o_j \in R^+_B(o_i)$, $x^A_{(i,j)} = 1$ and $x^B_{(i,j)} = 1$. Then $x^{A\cup B}_{(i,j)} = x^A_{(i,j)} \times x^B_{(i,j)} = 1$, and the reverse is also true. If $x^{A\cup B}_{(i,j)} = 0$, i.e., $o_j \notin R^+_{A\cup B}(o_i)$, that is, $o_j \notin R^+_A(o_i)$ or $o_j \notin R^+_B(o_i)$, i.e., $x^A_{(i,j)} = 0$ or $x^B_{(i,j)} = 0$. Consequently, we obtain $x^{A\cup B}_{(i,j)} = x^A_{(i,j)} \times x^B_{(i,j)} = 0$, the same holds true if the situation is reversed. Finally, there are $x^{A\cup B}_{(i,j)} = x^A_{(i,j)} \times x^B_{(i,j)}$, i.e., $\mathbb{X}_O^{\succeq A\cup B} = \mathbb{X}_O^{\succeq A} \cap \mathbb{X}_O^{\succeq B}$ holds. □

Definition 3.3 Given $I^{\succeq} = (O, C \cup \{d\}, V, h)$ is an ODDS, for $\forall A \subseteq C$, the dominant diagonal matrix and its inverse matrix for $\mathbb{X}_O^{\succeq A} = \left[x^A_{(i,j)}\right]_{n\times n}$ are defined as $\mathbb{D}_O^{\succeq A} = \left[d^A_{(i,j)}\right]_{n\times n}$ and $\left(\mathbb{D}_O^{\succeq A}\right)^{-1} = \left[\frac{1}{d^A_{(i,j)}}\right]_{n\times n}$, respectively, where

$$d^A_{(i,j)} = \begin{cases} \sum_{l=1}^n x^A_{(i,l)}, & i, j \in [1, n], i = j; \\ 0, & i, j \in [1, n], i \neq j. \end{cases} \tag{11}$$

$$\frac{1}{d^A_{(i,j)}} = \begin{cases} \frac{1}{\sum_{l=1}^{n} x^A_{(i,l)}}, & i, j \in [1, n], i = j; \\ 0, & i, j \in [1, n], i \neq j. \end{cases} \tag{12}$$

At the same time we define the $|\cdot|$ operation as $\left|\mathbb{D}_O^{\succeq A}\right| = \Pi_{i=j=1}^{n} d^A_{ij}$.

Corollary 3.1 Let $I^{\succeq} = (O, C \cup \{d\}, V, h)$ be an $ODDS$, for $\forall A \subseteq C$. The matrix dominance conditional entropy of condition attribute A under decision attribute d is defined as follows:

$$MDE^{\succeq}_{d|A}(O) = -\frac{1}{|O|} \log \left| \mathbb{D}_O^{\succeq A \cup \{d\}} \times \left(\mathbb{D}_O^{\succeq A}\right)^{-1} \right|. \tag{13}$$

Proof According to Definition 2.6, we have $DE^{\succeq}_{d|A}(O) = -\frac{1}{|O|} \sum_{i=1}^{n} \log \frac{\left|R^{+}_{\{d\} \cup A}(o_i)\right|}{\left|R^{+}_{A}(o_i)\right|} = -\frac{1}{|O|} \log \frac{\Pi_{i=1}^{n} \left|R^{+}_{\{d\} \cup A}(o_i)\right|}{\Pi_{i=1}^{n} \left|R^{+}_{A}(o_i)\right|}$. According to Definitions 3.1 and 3.3, the dominance diagonal matrices $\mathbb{D}_O^{\succeq A} = \left[d^A_{(i,j)}\right]_{n \times n}$ and $\mathbb{D}_O^{\succeq A \cup \{d\}} = \left[d^{A \cup \{d\}}_{(i,j)}\right]_{n \times n}$, where $d^A_{(i,j)} = \left|R^{+}_{A}(o_i)\right|$ and $d^{A \cup \{d\}}_{(i,j)} = \left|R^{+}_{A \cup \{d\}}(o_i)\right|$. Because $\left| \mathbb{D}_O^{\succeq A \cup \{d\}} \times \left(\mathbb{D}_O^{\succeq A}\right)^{-1} \right| = \Pi_{i=1}^{n} \frac{d^{A \cup d}_{(i,j)}}{d^A_{(i,j)}} = \frac{\Pi_{i=1}^{n} d^{A \cup |d}_{(i,j)}}{\Pi_{i=1}^{n} d^A_{(i,j)}} = \frac{\Pi_{i=1}^{n} \left|R^{+}_{\{d\} \cup A}(o_i)\right|}{\Pi_{i=1}^{n} \left|R^{+}_{A}(o_i)\right|}$. Thus, we can get $DE^{\succeq}_{d|A}(O) = MDE^{\succeq}_{d|A}(O)$. The results obtained from both matrix and non-matrix methods for calculating the dominance conditional entropy are found to be identical.

The core component of MDCE, as derived from formula (13), is given by $\left| \mathbb{D}_O^{\succeq A \cup \{d\}} \cdot \left(\mathbb{D}_O^{\succeq A}\right)^{-1} \right|$. Here, the dimensions of the diagonal matrix are indicated from $\mathbb{D}_O^{\succeq A \cup \{d\}}$ to $\mathbb{D}_O^{\succeq A}$. Its interpretation resembles that of formula (6). Finally, an illustrative example is provided to demonstrate the computational approach for Corollary 3.1. □

Example 2 The ordered decision data set presented in Table 2 serves to elucidate the subsequent analysis. In Table 2, $U = \{o_1, o_2, o_3, o_4, o_5, o_6, o_7\}$, $C = \{c_1, c_2, c_3, c_4\}$. The different feature rankings are like this $V_{c_1} : 1 \prec 2 \prec 3$, $V_{c_2} : 1 \prec 2 \prec 3$, $V_{c_3} : 1 \prec 2 \prec 3$, $V_{c_4} : 2 \prec 3$, and $V_d : 1 \prec 2 \prec 3 \prec 4$. According to Definition 3.1, the dominant matrices $\mathbb{X}_O^{\succeq C}$ and $\mathbb{X}_O^{\succeq d}$ are as follows:

Table 2 An ordered decision data set

U	c_1	c_2	c_3	c_4	d
o_1	2	3	2	3	1
o_2	3	1	2	2	2
o_3	1	2	3	3	2
o_4	2	3	1	3	3
o_5	3	1	2	2	4
o_6	1	2	3	3	2
o_7	3	1	2	2	2

$$\mathbb{X}_O^{\succeq C} = \begin{bmatrix} 1&0&0&0&0&0&0 \\ 0&1&0&0&1&0&1 \\ 0&0&1&0&0&1&0 \\ 1&0&0&1&0&0&0 \\ 0&1&0&0&1&0&1 \\ 0&0&1&0&0&1&0 \\ 0&1&0&0&1&0&1 \end{bmatrix}_{7\times 7}, \quad \mathbb{X}_O^{\succeq d} = \begin{bmatrix} 1&1&1&1&1&1&1 \\ 0&1&1&0&1&1&1 \\ 0&1&1&0&1&1&1 \\ 0&1&1&1&1&1&1 \\ 0&0&0&0&1&0&0 \\ 0&1&1&0&1&1&1 \\ 0&1&1&0&1&1&1 \end{bmatrix}_{7\times 7}.$$

Taking $\mathbb{X}_O^{\succeq C}$ as an exemplar, the verification of Property 3.1 is conducted in the following manner.

(1) For $\forall i \in \{1, 2, \ldots, 7\}$, $x^C_{(i,i)} = 1$;

(2) For $\forall i, j \in \{1, 2, \ldots, 7\}$, $\sum_{j=1}^{7} x^C_{(i,j)} = \left|R^{+}_{C}(o_i)\right|$ and $\sum_{i=1}^{7} x^C_{(i,j)} = \left|R^{-}_{C}(o_j)\right|$. While $i = 1$, $R^{+}_{C}(o_1) = \{o_1\}$, there is $\sum_{j=1}^{7} x^C_{(1,j)} = \left|R^{+}_{C}(o_1)\right| = 1$; while $j = 1$, $R^{-}_{C}(o_1) = \{o_1, o_4\}$, we have $\sum_{i=1}^{7} x^C_{(i,1)} = \left|R^{-}_{C}(o_1)\right| = 2$.

According to Definition 3.2, the matrix $\mathbb{X}_O^{\succeq C \cup \{d\}}$ representing the dominance relation is computed in the following manner.

$$\mathbb{X}_O^{\succeq C \cup \{d\}} = \mathbb{X}_O^{\succeq C} \cap \mathbb{X}_O^{\succeq d} = \begin{bmatrix} 1\times 1 & 0\times 1 & 0\times 1 & 0\times 1 & 0\times 1 & 0\times 1 & 0\times 1 \\ 0\times 0 & 1\times 1 & 0\times 1 & 0\times 0 & 1\times 1 & 0\times 1 & 1\times 1 \\ 0\times 0 & 0\times 1 & 1\times 1 & 0\times 0 & 0\times 1 & 1\times 1 & 0\times 1 \\ 1\times 0 & 0\times 1 & 0\times 1 & 1\times 1 & 0\times 1 & 0\times 1 & 0\times 1 \\ 0\times 0 & 1\times 0 & 0\times 0 & 0\times 0 & 1\times 1 & 0\times 0 & 1\times 0 \\ 0\times 0 & 0\times 1 & 1\times 1 & 0\times 0 & 0\times 1 & 1\times 1 & 0\times 1 \\ 0\times 0 & 1\times 1 & 0\times 1 & 0\times 0 & 1\times 1 & 0\times 1 & 1\times 1 \end{bmatrix}$$

$$= \begin{bmatrix} 1&0&0&0&0&0&0 \\ 0&1&0&0&1&0&1 \\ 0&0&1&0&0&1&0 \\ 0&0&0&1&0&0&0 \\ 0&0&0&0&1&0&0 \\ 0&0&1&0&0&1&0 \\ 0&1&0&0&1&0&1 \end{bmatrix}_{7\times 7}.$$

Consequently, based on Definition 3.3, the dominance relation is determined by calculating the diagonal matrices $\mathbb{D}_O^{\succeq C}$ and $\mathbb{D}_O^{\succeq C \cup \{d\}}$, as well as the inverse matrix $\left(\mathbb{D}_O^{\succeq C}\right)^{-1}$, resulting in

$$\mathbb{D}_O^{\succeq C} = \begin{bmatrix} 1&0&0&0&0&0&0 \\ 0&3&0&0&0&0&0 \\ 0&0&2&0&0&0&0 \\ 0&0&0&2&0&0&0 \\ 0&0&0&0&3&0&0 \\ 0&0&0&0&0&2&0 \\ 0&0&0&0&0&0&3 \end{bmatrix}_{7\times 7}, \mathbb{D}_O^{\succeq C \cup \{d\}} = \begin{bmatrix} 1&0&0&0&0&0&0 \\ 0&3&0&0&0&0&0 \\ 0&0&2&0&0&0&0 \\ 0&0&0&1&0&0&0 \\ 0&0&0&0&1&0&0 \\ 0&0&0&0&0&2&0 \\ 0&0&0&0&0&0&3 \end{bmatrix}_{7\times 7},$$

$$\left(\mathbb{D}_O^{\succeq C}\right)^{-1}=\begin{bmatrix}\frac{1}{1}&0&0&0&0&0&0\\0&\frac{1}{3}&0&0&0&0&0\\0&0&\frac{1}{2}&0&0&0&0\\0&0&0&\frac{1}{2}&0&0&0\\0&0&0&0&\frac{1}{3}&0&0\\0&0&0&0&0&\frac{1}{2}&0\\0&0&0&0&0&0&\frac{1}{3}\end{bmatrix}_{7\times 7}.$$

Finally, we have $MDE_{d|C}^{\succeq}(O)=-\frac{1}{7}\log\left|\mathbb{D}_O^{\succeq C\cup\{d\}}\times\left(\mathbb{D}_O^{\succeq C}\right)^{-1}\right|=0.3693.$

Corollary 3.2 Let $I^{\succeq}=(O,C\cup\{d\},V,h)$ be an $ODDS$, for $\forall A\subseteq C$ and $\forall c\in A$, the internal significance measure of matrix dominance conditional entropy is defined as follows:

$$\phi_M(c,A,d)=MDE_{d|(A-\{c\})}^{\succeq}(O)-MDE_{d|A}^{\succeq}(O). \tag{14}$$

Corollary 3.3 Let $I^{\succeq}=(O,C\cup\{d\},V,h)$ be an $ODDS$, for $\forall B\subseteq C$ and $\forall c\in(C-B)$, the external significance measure of matrix dominance conditional entropy is defined as follows:

$$\psi_M(c,B,d)=MDE_{d|B}^{\succeq}(O)-MDE_{d|B\cup\{c\}}^{\succeq}(O). \tag{15}$$

Clearly, the internal and external significance measures of the matrix dominance conditional entropy are similar to Definitions 2.8 and 2.9 in the previous section.

3.2 Static feature selection algorithm based on MDCE

The feature selection algorithm based on MDCE in ODDS is presented in this section. Regarding this algorithm, it recalculates the reduction from scratch when there are changes in the reduction data object. Therefore, although not classified as a dynamic attribute reduction algorithm like the feature incremental algorithm, it serves as a basic for the subsequent paper's incremental feature selection algorithm.

The subsequent steps outline Algorithm 1. Step 2 involves calculating the matrix dominance conditional entropy of the ODDS. Next six steps aim to identify significant important attributes and obtain preliminary reduced subsets. Steps 10–16 primarily focus on determining if there exist important core attributes among the initially filtered attributes, until it is verified that any remaining attributes are unnecessary. The last for loop involve eliminating redundant attributes from the existing attribute set to ensure indispensability of each attribute. The time complexity and space complexity of Algorithm 1 can be expressed as $O(|C||O|^2+2|C|^2|O|^2+|B|^2|O|^2)$ and $O(|O|^2+|C||O|^2)$, respectively.

Algorithm 1 Static feature selection algorithm.

Input: An ODDS $I^{\succeq}=(O,C\cup\{d\},V,h)$.
Output: A reduct Red_O.
1 Initialize $Red_O\leftarrow\emptyset$;
2 Calculate $MDE_{d|C}^{\succeq}(O)$ in O via using (17);
3 **for** each$c_k\in$ C **do**
4 Calculate $\phi_M\,(c_k,C,d)$ via using (18);
5 **if** $\phi_M\,(c_k,C,d)>0$, **then**
6 $Red_O\leftarrow Red_O\cup\{c_k\}$;
7 **end**
8 **end**
9 $B\leftarrow \mathrm{Red}_U$;
10 **while** $MDE_{d|B}^{\succeq}(O)\neq MDE_{d|C}^{\succeq}(O)$ **do**
11 **for** each $c_l\in C$ **do**
12 Calculate $\psi_M\,(c_l,B,d)$ via using (19);
13 **end**
14 Select $c_{max}=\max\{\psi_M\,(c_l,B,d)\,,c_l\in(C-B)\}$;
15 $B\leftarrow B\cup\{c_{max}\}$;
16 **end**
17 **for** each c $\in B$ **do**
18 **if** $MDE_{d|(B-\{c\})}^{\succeq}(O)=MDE_{d|B}^{\succeq}(O)$, **then**
19 $B\leftarrow B-\{c\}$;
20 **end**
21 **end**
22 $Red_O\leftarrow B$;
23 **return** Red_O;

4 Incremental multi-objective feature selection method based on matrix dominance conditional entropy

The features in an ordered data set may increase or decrease over time. The process of statically computing reductions can be complicated by the presence of repeated computations. Therefore, this section proposes and elaborates on two types of incremental attribute reduction algorithms that utilize previously obtained results to save time and space while reducing algorithmic complexity.

4.1 An incremental multi-objective feature selection method when adding features

In this section, we initially examine the change process of matrix dominance conditional entropy as the attribute increases and present the corresponding feature selection algorithm.

4.1.1 Matrix dominance conditional entropy adjust principle when adding features

The addition of new features will inevitably alter the dominance relationship between the original objects. It is evident that in order for the original dominant object to maintain its advantage, it must also uphold its superiority under the influence of these new features. Hence, modifications are made to the dominant matrix.

Proposition 4.1 *Let $I^{\succeq} = (O, C \cup \{d\}, V, h)$ be an ODDS, where $C = \{c_1, c_2, \ldots, c_n\}$. For $\forall A \subseteq C$, assume as the dominant matrix on O concerning A is $\mathbb{X}_O^{\succeq A} = \left[x_{(i,j)}^A\right]_{n\times n}$, the feature set $A^+ = \left\{c_{n+1}, c_{n+2}, \ldots, c_{n+n'}\right\}$ is added to $I^{\succeq}$. The adjust dominant matrix is defined as $\mathbb{X}_O^{\succeq A\cup A^+} = \left[x_{(i,j)}^{\prime A\cup A^+}\right]_{n\times n}$, where*

$$x_{(i,j)}^{\prime A\cup A^+} = \begin{cases} 1, & h(o_j, c) \geq h(o_i, c), \forall c \in A \cup A^+; \\ 0, & \textit{otherwise}. \end{cases} \tag{16}$$

The rationale for updating the dominant matrix is provided by Proposition 4.1 as multiple features are incorporated. The fundamental idea is to assess whether the recently incorporated conditional features of the primary dominant entity maintain their dominance according to the adjust matrix representing dominant relationships. Illustrative examples are presented below.

Example 4 A noval attribute set $C^+ = \{c_5\}$ is added to Table 2. There is $c_5 = \{1, 2, 2, 1, 3, 2, 2\}$, where feature rankings is $V_{c_5} : 1 \prec 2 \prec 3$. The newly introduced matrix $\mathbb{X}_O^{\succeq C\cup C^+}$ represents the dominance relation among conditional attributes and can be denoted as

$$\mathbb{X}_O^{\succeq C\cup C^+} = \begin{bmatrix} 1&0&0&0&0&0&0 \\ 0&1&0&0&1&0&1 \\ 0&0&1&0&0&1&0 \\ 1&0&0&1&0&0&0 \\ 0&0&0&0&1&0&0 \\ 0&0&1&0&0&1&0 \\ 0&1&0&0&1&0&1 \end{bmatrix}_{7\times 7}.$$

Proposition 4.2 *Let $I^{\succeq} = (O, C \cup \{d\}, V, h)$ be an ODDS, where $C = \{c_1, c_2, \ldots, c_n\}$. For $\forall A \subseteq C$, then the dominant diagonal matrix is $\mathbb{D}_O^{\succeq A} = \left[d_{(i,j)}^A\right]_{n\times n}$, the feature set $A^+ = \left\{c_{n+1}, c_{n+2}, \ldots, c_{n+n'}\right\}$ is added to $I^{\succeq}$. The adjust dominant diagonal matrix is defined as $\mathbb{D}_O^{\succeq A\cup A^+} = \left[d_{(i,j)}^{\prime A\cup A^+}\right]_{n\times n}$, where*

$$d_{(i,j)}^{\prime A\cup A^+} = \begin{cases} d_{(i,j)}^A - x_{(i,j)}^{\prime A\cup A^+}, & h(o_j, c) \geq h(o_i, c), \forall c \in A \cup A^+; \\ d_{(i,j)}^A, & h(o_j, c) < h(o_i, c), \forall c \in A \cup A^+. \end{cases} \tag{17}$$

Example 5 Building upon Example 4, given matrices $\mathbb{X}_O^{\succeq C\cup C^+}$ and $\mathbb{D}_O^{\succeq C}$, we can apply Proposition 4.2 to adjust matrix $\mathbb{D}_O^{\succeq C\cup C^+}$ as

$$\mathbb{D}_U^{\succeq C\cup C^+} = \begin{bmatrix} 1-0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 3-0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 2-0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 2-0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 3-2 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 2-0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 3-0 \end{bmatrix}$$

$$= \begin{bmatrix} 1&0&0&0&0&0&0 \\ 0&3&0&0&0&0&0 \\ 0&0&2&0&0&0&0 \\ 0&0&0&2&0&0&0 \\ 0&0&0&0&1&0&0 \\ 0&0&0&0&0&2&0 \\ 0&0&0&0&0&0&3 \end{bmatrix}_{7\times 7}.$$

Next, we will outline the detailed steps for calculating a new MDCE after incorporating multiple attribute characteristics. As a new object set O^+, we are aware that the origin matrices are $\mathbb{X}_O^{\succeq A}$, $\mathbb{X}_{O\cup O^+}^{\succeq A}$, $\mathbb{D}_O^{\succeq A}$, and $\mathbb{D}_{O\cup O^+}^{\succeq A}$. When A^+ is added to $I^{\succeq}$, we can readily acquire the adjust dominant diagonal matrices $\mathbb{D}_O^{\succeq A\cup A^+}$ and $\mathbb{D}_{O\cup O^+}^{\succeq A}$. Therefore, by applying Corollary 3.1, the $MDE_{d|A\cup A^+}^{\succeq}(O)$ can be performed effortlessly.

4.1.2 An incremental feature selection algorithm when adding features

The incremental feature selection algorithm (IFS-A) is presented in Algorithm 2, drawing inspiration from the updating principle of MDCE.

The specific procedures outlined in Algorithm 2 are presented below. Steps 2-4 progressively compute the updated dominant matrix and its dominant diagonal matrix. Then determines the adjusted MDCE based on Corollary 3.1. Steps 6-10 primarily aim to ascertain if the new MDCE matches both the MDCE of the initial attribute subset (i.e., raw reduction) and that below this complete attribute set, keeping the original subset of attributes unchanged if necessary. Steps 11-16 arrange the eliminated attributes in descending order to form a fresh set and modify the selected attribute subset until reaching Step 12. Steps 17-22 eliminate redundant attributes from the existing attribute set to ensure indispensability of each attribute within it. Finally, steps 23-24 present the ultimate reduction outcome. In brief, the time and space complexity are $O\big(|O||C^+||C'| + (|C'| - |B|)|O|^2 + |B|^2|O|^2\big)$

Algorithm 2 IFS-A algorithm

Input:
(1) A raw ODDS $I^{\succeq} = (O, C \cup \{d\}, V, h)$, where $C = \{c_1, c_2, \ldots, c_n\}$. New attributes set $C^+ = \{c_{n+1}, c_{n+2}, \ldots, c_{n+n}\}$;
(2) The original reduct Red_O on O;
(3) The original dominant matrices $\mathbb{X}_O^{\succeq C} = \left[x_{(i,j)}^C\right]_{n\times n}$, $\mathbb{X}_O^{\succeq C\cup\{d\}} = \left[x_{(i,j)}^{C\cup\{d\}}\right]_{n\times n}$, $\mathbb{X}_O^{\succeq Red_O} = \left[x_{(i,j)}^{Red_O}\right]_{n\times n}$, and $\mathbb{X}_O^{\succeq d} = \left[x_{(i,j)}^{d}\right]_{n\times n}$;
(4) The original dominance diagonal matrices $\mathbb{D}_O^{\succeq C} = \left[d_{(i,j)}^C\right]_{n\times n}$, $\mathbb{D}_O^{\succeq C\cup\{d\}} = \left[d_{(i,j)}^{C\cup\{d\}}\right]_{n\times n}$, $\mathbb{D}_O^{\succeq Red_O} = \left[d_{(i,j)}^{Red_O}\right]_{n\times n}$ and $\mathbb{D}_O^{\succeq Red_O\cup\{d\}} = \left[d_{(i,j)}^{Red_O\cup\{d\}}\right]_{n\times n}$.
Output: A new reduct $Red_{O'}$.

1 Initialize $B \leftarrow Red_O$, $C' \leftarrow C \cup C^+$, $\mathbb{X}_O^{\succeq C'} \leftarrow \mathbb{X}_O^{\succeq C}$, $\mathbb{D}_O^{\succeq C'} \leftarrow \mathbb{D}_O^{\succeq C}$, $\mathbb{D}_O^{\succeq C'\cup\{d\}} \leftarrow \mathbb{D}_O^{\succeq C\cup\{d\}}$;
2 Compute new dominant matrices $\mathbb{X}_O^{\succeq C'} \leftarrow \left[x_{(i,j)}^{\prime C}\right]_{n\times n}$, $\mathbb{X}_O^{\succeq B} \leftarrow \left[x_{(i,j)}^{\prime B}\right]_{n\times n}$, $\mathbb{X}_O^{\succeq d} \leftarrow \left[x_{(i,j)}^{\prime d}\right]_{n\times n}$ via using Proposition 4.1;
3 Compute dominant matrices $\mathbb{X}_O^{\succeq C'\cup\{d\}}$ and $\mathbb{X}_O^{\succeq B\cup\{d\}}$;
4 Compute new dominance diagonal matrices $\mathbb{D}_O^{\succeq C} = \left[d_{(i,j)}^C\right]_{n\times n}$, $\mathbb{D}_O^{\succeq C\cup\{d\}} = \left[d_{(i,j)}^{C\cup\{d\}}\right]_{n\times n}$, $\mathbb{D}_O^{\succeq Red_O} = \left[d_{(i,j)}^{Red_O}\right]_{n\times n}$ and $\mathbb{D}_O^{\succeq B\cup\{d\}} \leftarrow \left[d_{(i,j)}^{\prime B\cup\{d\}}\right]_{n\times n}$ via using Proposition 4.2;
5 Compute new MDCE MDCE $MDE^{\succeq}_{d|C'}(O)$ and $MDE^{\succeq}_{d|B}(O)$;
6 **if** $MDE^{\succeq}_{d|C'}(O) = MDE^{\succeq}_{d|B}(O)$, **then**
7 | go to step17;
8 **else**
9 | go to step11;
10 **end**
11 For each $c \in (C' - B)$, compute $\psi_M(c, B, d)$, then save the result as $\left\{c'_0, c'_1, \ldots, c'_{|C'-B|}\right\}$;
12 **while** $MDE^{\succeq}_{d|C'}(O) \neq MDE^{\succeq}_{d|B}(O)$ **do**
13 | **for** $c_z \in C' - B$ **do**
14 | | Select $B \leftarrow B \cup \left\{c'_z\right\}$, then calculate $MDE^{\succeq}_{d|B}(O)$;
15 | **end**
16 **end**
17 **for** each $c \in B$ **do**
18 | calculate $MDE^{\succeq}_{d(B-\{c\})}(O)$;
19 | **if** $MDE^{\succeq}_{d|(B-\{c\})}(O) = MDE^{\succeq}_{d|B}(O)$, **then**
20 | | $B \leftarrow B - \{c\}$;
21 | **end**
22 **end**
23 $Red_{O'} \leftarrow B$;
24 **return** $Red_{O'}$;

and $\mathrm{O}\left(|O|^2 + (|C'| - |B|)|O|^2\right)$, respectively. The specific comparison between SFS and IFS-A is shown in Table 3.

From Table 3, it is evident that the IFS-A algorithm exhibits lower time and space complexity compared to the SFS algorithm. This discrepancy arises because the SFS algorithm recomputes reductions from scratch whenever there are changes in features, whereas the IFS-A algorithm leverages previous reduction results, thereby significantly reducing both time and space complexity. Consequently, employing the IFS-A algorithm can considerably expedite reduction calculations for extensive datasets.

4.2 An incremental multi-objective feature selection method when deleting features

The change process of the dominant conditional entropy of the matrix is preliminarily investigated in this section when the attribute is deleted, and a corresponding feature selection algorithm is provided.

4.2.1 Matrix dominance conditional entropy adjust principle when deleting features

Similar to Section 4.1.1, this section introduces the updating mechanism of matrix dominance conditional entropy when ODDS deleting features.

Proposition 4.3 *Let $I^{\succeq} = (O, C \cup \{d\}, V, h)$ be an ODDS, where $C = \{o_1, o_2, \ldots, o_n\}$. For $\forall A \subseteq C$, suppose that the dominant matrix is $\mathbb{X}_O^{\succeq A} = \left[x_{(i,j)}^A\right]_{n\times n}$, the feature set $A^- = \left\{o_{q1}, o_{q2}, \ldots, o_{qn'}\right\}$ is deleted from $I^{\succeq}$. The adjust dominant matrix is defined as $\mathbb{X}_O^{\succeq A-A^-} = \left[x_{(i,j)}^{\prime A-A^-}\right]_{n\times n}$, where*

$$x_{(i,j)}^{\prime A-A^-} = \begin{cases} 1, & h(o_j, c) \geq h(o_i, c), \forall c \in A - A^-; \\ 0, & otherwise. \end{cases} \tag{18}$$

Example 6 A feature set $C^- = \{c_3, c_4\}$ is deleted from Table 2. The new conditional attribute dominant matrix $\mathbb{X}_O^{\succeq C-C^-}$ can be expressed as

$$\mathbb{X}_O^{\succeq C-C^-} = \begin{bmatrix} 1&0&0&1&0&0&0 \\ 0&1&0&0&1&0&1 \\ 1&0&1&1&0&1&0 \\ 1&0&0&1&0&0&0 \\ 0&0&0&0&1&0&0 \\ 1&0&1&1&0&1&0 \\ 0&1&0&0&1&0&1 \end{bmatrix}_{7\times 7}.$$

Proposition 4.4 *Let $I^{\succeq} = (O, C \cup \{d\}, V, h)$ be an ODDS, where $C = \{c_1, c_2, \ldots, c_n\}$. For $\forall A \subseteq C$, given the dominant diagonal matrix $\mathbb{D}_O^{\succeq A} = \left[d_{(i,j)}^A\right]_{n\times n}$, then feature set $A^- = \left\{c_{q1}, c_{q2}, \ldots, c_{qn'}\right\}$ is deleted from $I^{\succeq}$. The adjust dominant diagonal matrix is defined as $\mathbb{D}_O^{\succeq A-A^-} = \left[d_{(i,j)}^{\prime A-A^-}\right]_{n\times n}$, where*

$$d_{(i,j)}^{\prime A-A^-} = \begin{cases} d_{(i,j)}^A + x_{(i,j)}^{\prime A-A^-}, & h(o_j, c) \geq h(o_i, c), \forall c \in A - A^-; \\ d_{(i,j)}^A, & h(o_j, c) < h(o_i, c), \forall c \in A - A^-. \end{cases} \tag{19}$$

Table 3 Time and space complexity comparison of SFS and IFS-A

Algorithm	SFS	IFS-A
Time complexity	$\mathrm{O}\big(\lvert C'\rvert\,\lvert O\rvert^2+2\lvert C'\rvert^2\,\lvert O\rvert^2+\lvert B\rvert^2\,\lvert O\rvert^2\big)$	$\mathrm{O}\big(\lvert O\rvert\,\lvert C^+\rvert\,\lvert C'\rvert+(\lvert C'\rvert-\lvert B\rvert)\,\lvert O\rvert^2+\lvert B\rvert^2\,\lvert O\rvert^2\big)$
Space complexity	$\mathrm{O}\big(\lvert O\rvert^2+\lvert C'\rvert\,\lvert O\rvert^2\big)$	$\mathrm{O}\big(\lvert O\rvert^2+(\lvert C'\rvert-\lvert B\rvert)\,\lvert O\rvert^2\big)$

Example 7 Building upon Example 6, given matrices $\mathbb{X}_O^{\succeq C-C^-}$ and $\mathbb{D}_O^{\succeq C}$, we can apply Proposition 4.4 to adjust matrix $\mathbb{D}_O^{\succeq C-C^-}$ as

$$\mathbb{D}_O^{\succeq C-C^-}=\begin{bmatrix}1+1 & 0 & 0 & 0 & 0 & 0 & 0\\ 0 & 3+0 & 0 & 0 & 0 & 0 & 0\\ 0 & 0 & 2+2 & 0 & 0 & 0 & 0\\ 0 & 0 & 0 & 2+0 & 0 & 0 & 0\\ 0 & 0 & 0 & 0 & 3+0 & 0 & 0\\ 0 & 0 & 0 & 0 & 0 & 2+2 & 0\\ 0 & 0 & 0 & 0 & 0 & 0 & 3+0\end{bmatrix}$$

$$=\begin{bmatrix}2&0&0&0&0&0&0\\0&3&0&0&0&0&0\\0&0&4&0&0&0&0\\0&0&0&2&0&0&0\\0&0&0&0&3&0&0\\0&0&0&0&0&4&0\\0&0&0&0&0&0&3\end{bmatrix}_{7\times 7}.$$

4.2.2 An incremental feature selection algorithm when deleting features

Motivated by the refreshing principle of MDCE, Algorithm 3 introduces a novel approach called IFS-D, which is a multi-feature incremental feature selection algorithm when deleting features in ODDS.

The step-by-step procedure of Algorithm 3 is outlined below. Steps 2-3 progressively calculate the updated dominant matrix and its corresponding diagonal matrix. Then we determine the new MDCE. Next five steps primarily aim to ascertain if the new MDCE matches both the original attribute subset's MDCE (raw reduction) and the overall attribute set's new MDCE, thereby preserving the raw subset of attributes when necessary. Arrange the eliminated attributes in descending order and update the selected attribute subset through steps 10-15. Finally, we eliminate redundant attributes from the existing set to ensure that each attribute remains essential.

The time and space complexity are $\mathrm{O}(|O|+(|C'|-|B|)|O|^2+|B|^2|O|^2)$ and $\mathrm{O}\big(|O|^2+(|C'|-|B|)|O|^2\big)$ in general, respectively. We have also conducted a comparison between the complexity of the SFS and the IFS-D, with the corresponding findings presented in Table 4.

The time and space complexity of the IFS-D algorithm is demonstrated to be smaller than that of the SFS algorithm, as shown in Table 4. This disparity arises from the fact that when there are changes in features, the SFS algorithm recomputes reductions from scratch, whereas the IFS-D algorithm leverages previous reduction results, resulting in a significant reduction in both time and space complexity. Consequently, for large-scale data reduction calculations, the IFS-D algorithm offers substantial time savings.

5 Experiments

A set of experiments are carried out to verify the efficiency of the incremental algorithm proposed for attribute features. Table 5 summarizes the nine UCI datasets used in these experiments. All algorithms in this study were implemented using Python within Anaconda Navigator environment and executed on a computer equipped with an AMD Ryzen 7 4800H CPU (2.90 GHz) with Radeon Graphics, 8 GB memory, and running on a 64-bit Windows 10 operating system.

We conducted a comparative experiment to assess the effectiveness of our suggested algorithm. We compared our IFS-A and IFS-D algorithms with four feature selection algorithms, namely SFS, DRSQR, FEAR, and NRSAR. The SFS algorithm is shown in Algorithm 1. DRSQR is a quick reduction algorithm built upon dominant RST. FEAR utilizes fuzzy entropy for attribute reduction while NRSAR employs neighborhood rough sets for reducing attributes. The paper employs four classical classifiers, namely BayesNet (BN), RandomTree (RT), K-NearestNeighbor (KNN), and Adaptive boosting (Adaboost), to assess the impact of attribute reduction on classification accuracy through 10-fold cross-validation.

Table 4 Time and space complexity comparison of SFS and IFS-D

Algorithm	SFS	IFS-D
Time complexity	$\mathrm{O}(\lvert C'\rvert\,\lvert O\rvert^2+2\lvert C'\rvert^2\,\lvert O\rvert^2+\lvert B\rvert^2\,\lvert O\rvert^2)$	$\mathrm{O}(\lvert O\rvert+(\lvert C'\rvert-\lvert B\rvert)\,\lvert O\rvert^2+\lvert B\rvert^2\,\lvert O\rvert^2)$
Space complexity	$\mathrm{O}(\lvert O\rvert^2+\lvert C'\rvert\,\lvert O\rvert^2)$	$\mathrm{O}(\lvert O\rvert^2+(\lvert C'\rvert-\lvert B\rvert)\,\lvert O\rvert^2)$

5.1 Experimental effect analysis of IFS-A algorithm

The algorithm IFS-A is analyzed from three perspectives: classification accuracy, algorithm time, and index performance evaluation.

Algorithm 3 IFS-D algorithm

Input:
(1) A raw ODDS $I^{\succeq} = (O, C \cup \{d\}, V, h)$, where $C = \{c_1, c_2, \ldots, c_n\}$.
$C^{-} = \left\{c_{q1}, c_{q2}, \ldots, c_{qn'}\right\}$ is a deleted feature set ;
(2) The original reduct Red_O on O;
(3) The original dominant matrices $\mathbb{X}_O^{\succeq C} = \left[x_{(i,j)}^{C}\right]_{n\times n}$, $\mathbb{X}_O^{\succeq C\cup\{d\}} = \left[x_{(i,j)}^{C\cup\{d\}}\right]_{n\times n}$, $\mathbb{X}_O^{\succeq Red_O} = \left[x_{(i,j)}^{Red_O}\right]_{n\times n}$ and $\mathbb{X}_O^{Red_O\cup\{d\}} = \left[x_{(i,j)}^{Red_O\cup\{d\}}\right]_{n\times n}$;
(4) The original dominance diagonal matrices $\mathbb{D}_O^{\succeq C} = \left[d_{(i,j)}^{C}\right]_{n\times n}$, $\mathbb{D}_O^{\succeq C\cup\{d\}} = \left[d_{(i,j)}^{C\cup\{d\}}\right]_{n\times n}$, $\mathbb{D}_O^{\succeq Red_O} = \left[d_{(i,j)}^{Red_O}\right]_{n\times n}$ and $\mathbb{D}_O^{\succeq Red_O\cup\{d\}} = \left[d_{(i,j)}^{Red_O\cup\{d\}}\right]_{n\times n}$.
Output: A new reduct $Red_{U'}$.

1 Initialize $B \leftarrow Red_O, C' \leftarrow C - C^{-}, \mathbb{X}_O^{\succeq C'} \leftarrow \mathbb{X}_O^{\succeq C}, \mathbb{X}_O^{\succeq C'\cup\{d\}} \leftarrow \mathbb{X}_O^{\succeq C\cup\{d\}}, \mathbb{D}_O^{\succeq C'} \leftarrow \mathbb{D}_O^{\succeq C}, \mathbb{D}_O^{\succeq C'\cup\{d\}} \leftarrow \mathbb{D}_O^{\succeq C\cup\{d\}}$;
2 Compute new dominant matrices $\mathbb{X}_O^{\succeq C'} \leftarrow \left[x'^{C}_{(i,j)}\right]_{n\times n}, \mathbb{X}_O^{\succeq B} \leftarrow \left[x'^{B}_{(i,j)}\right]_{n\times n}, \mathbb{X}_O^{\succeq C'\cup\{d\}} \leftarrow \left[x_{(i,j)}^{\succeq C'\cup\{d\}}\right]_{n\times n}$ and $\mathbb{X}_O^{\succeq B'\cup\{d\}} \leftarrow \left[x_{(i,j)}^{\succeq B'\cup\{d\}}\right]_{n\times n}$ via using Proposition 4.3 ;
3 Compute new dominance diagonal matrices $\mathbb{D}_O^{\succeq C'} \leftarrow \left[d'^{C}_{(i,j)}\right]_{n\times n}, \mathbb{D}_O^{\succeq C'\cup\{d\}} \leftarrow \left[d'^{C\cup\{d\}}_{(i,j)}\right]_{n\times n}, \mathbb{D}_O^{\succeq B} \leftarrow \left[d'^{B}_{(i,j)}\right]_{n\times n}, \mathbb{D}_O^{\succeq B\cup\{d\}} \leftarrow \left[d'^{B\cup\{d\}}_{(i,j)}\right]_{n\times n}$ via using Proposition 4.4;
4 Calculate new MDCE MDCE $MDE^{\succeq}_{dC'}(O)$ and $MDE^{\succeq}_{d|B}(O)$;
5 **if** $MDE^{\succeq}_{dC'}(O) = MDE^{\succeq}_{d|B}(O)$, **then**
6 go to step16;
7 **else**
8 go to step10;
9 **end**
10 For each $c \in \left(C' - B\right)$, calculate $\psi_M(c, B, d)$, then save the result as $\left\{c'_0, c'_1, \ldots, c'_{|C'-B|}\right\}$;
11 **while** $MDE^{\succeq}_{d|C'}(O) \succeq MDE^{\succeq}_{d|B}(O)$ **do**
12 **for** $c_z \in C' - B$ **do**
13 Select $B \leftarrow B \cup \left\{c'_z\right\}$ then calculate $MDE^{\succeq}_{d|B}(O)$;
14 **end**
15 **end**
16 **for** each $c \in B$ **do**
17 calculate $MDE^{\succeq}_{d(B-\{c\})}(O)$;
18 **if** $MDE^{\succeq}_{d|[B-\{c\})}(O) = MDE^{\succeq}_{d|B}(O)$, **then**
19 $B \leftarrow B - \{c\}$;
20 **end**
21 **end**
22 $Red_{O'} \leftarrow B$;
23 **return** $Red_{O'}$;

5.1.1 Comparison of classification accuracy

The IFS-A algorithm proposed in this paper is compared to four other algorithms to evaluate its classification accuracy. Each data set listed in Table 5 is divided into two parts: a random selection of 50% features as the raw feature set, and the remaining 50% features added to it. The IFS-A, SFS, DRSQR, FEAR, and NRSAR algorithms are utilized to compute fresh reductions based on these sets. Experimental results can be found in Tables 6 and 7, where 'Origin' represents the classification effect of the full set of attributes. The numbers in parentheses following the classification accuracy in Table 6 indicate the reduced number of attributes. Additionally, Tables 7, 10, and 11 follow a same format as Table 6.

Based on the chart provided above, it can be observed that algorithm IFS-A consistently outperforms other algorithms in terms of classification accuracy across all scenarios. Moreover, its average score significantly surpasses others, indicating a remarkably high level of accuracy for the IFS-A algorithm.

5.1.2 Comparison of feature selection time

The effectiveness of IFS-A is verified in this section through a comparison of computation time and speed-up ratio. To conduct our tests, we created five test sets for each data set listed in Table 5. Initially, we randomly selected 50% of the attributes as the origin attribute set. Then, from the remaining 50%, we added attributes to the origin attribute set to create a dynamic data set for testing purposes. Specifically, we randomly selected 10%, 20%, 30%, 40%, and 50% of the remaining attributes and added them to the original attribute set. It is worth noting that since both BCC and Abalone data sets have less than ten attribute (9 for BCC and 8 for Abalone), we initially selected only four attributes for BCC and three attributes for Abalone. We then gradually added one attribute at a time. By comparing the computation times for various algorithms applied to these data sets, Fig. 1 illustrates computation times for all algorithms. The X-axis represents the size of the added attribute set, while the Y-axis represents computation time.

From Fig. 1, it is evident that as the attribute feature set increases, the computation time for these five algorithms also increases. Each subgraph clearly demonstrates that algorithm IFS-A has significantly lower computation time compared to other algorithms. This difference is particularly noticeable when dealing with large data sets, where algorithm IFS-A proves to be highly efficient in terms of saving time. Hence, we can conclude that algorithm IFS-A exhibits exceptional efficiency.

Subsequently, we validate the effectiveness of the IFS-A algorithm by analyzing its acceleration ratio. By utilizing the data presented in Fig. 1, we calculate the acceleration ratio of

Table 5 Details of the datasets

ID	Abbreviation	Datasets	Objects	Attributes	Classes
1	Abalone	Abalone	4177	8	3
2	BCC	Breast Cancer Coimbra	116	9	2
3	Codon	Codon_usage	13028	68	10
4	DMT	Detect Malware Types	7107	280	8
5	Eye	EEG Eye State	14980	14	2
6	Letter	Letter-recognition	20000	16	26
7	OLD	Ozone Level Detection	2536	73	2
8	RLS	Rocket League Skillshots	297	18	7
9	Wine	Wine	178	13	3

IFS-A compared to four other algorithms. The results of our experiments are depicted in Fig. 2. These algorithms consistently exhibit high speed ratios across various datasets. However, due to a potentially dense curve, identifying trends may be challenging. To address this issue, we present our results in a tridimensional context. To illustrate, Fig. 2(a) shows that the X-axis corresponds to the capacity of the added attribute set. The Y-axis represents BCC, Wine, RLS and OLD datasets respectively; their experimental results fall within a value range of [0-3]. Meanwhile,the Z-axis displays experimental results for DMT, Codon, Eye and Letter data sets with a value range between [0-250]. Subsequently, in Fig. 2(b), (c), (d), as well as Fig. 4(a), (b), (c) and (d), follow a similar structure as Fig. 2(a).

The results presented in Fig. 2 clearly demonstrate that algorithm IFS-A exhibits a superior acceleration ratio compared to the other algorithms across each data set. This indicates that algorithm IFS-A outperforms the remaining four algorithms in terms of speed for each empirical data set. Moreover, for large-scale data sets, algorithm IFS-A demonstrates a significantly higher level of efficiency, surpassing the other four algorithms by dozens or even hundreds of times. These findings once again validate the exceptional efficiency achieved by algorithm IFS-A.

5.1.3 Analysis of evaluation index

The algorithm is divided into two categories based on the measurement index on the sample, which are the measurement based on classification and the measurement based on ranking. This paper chooses to evaluate the effectiveness of the algorithm based on Average Precision (AP) and Ranking Loss (RL) in ranking metrics.

AP reflects the average probability of the predicted label ranking for all samples, where the one before the relevant label is also the relevant label. The formula for AP is expressed as follows:

$$AP(f) = \frac{1}{n}\sum_{i=1}^{n}\frac{1}{|R_i|}\sum_{l\in R_i}\frac{\left\{k \in R_i \mid r_f(o_i,k) \leqslant r_f(o_i,l)\right\}}{r_f(o_i,l)} \tag{20}$$

Table 6 Comparison accuracy(%) of five algorithms on BN and RT

Datasets	BN Origin	NRSAR	FEAR	DRAQR	SFS	IFS-A	RT Origin	NRSAR	FEAR	DRAQR	SFS	IFS-A
Abalone	51.27	57.02(3)	54.11(2)	53.32(3)	51.82(1)	**57.32(2)**	80.07	82.15(3)	74.13(2)	79.81(3)	86.25(1)	**88.76(2)**
Bcc	64.21	61.37(4)	64.98(5)	57.42(9)	63.25(5)	**65.71(6)**	80.03	81.17(4)	69.98(5)	**85.17(9)**	66.74(5)	82.22(6)
Codon	76.34	64.92(11)	72.18(13)	61.40(23)	75.72(14)	**75.81(8)**	77.4	72.89(11)	71.54(13)	80.84(23)	71.08(14)	**84.66(8)**
DMT	81.55	82.85(89)	74.27(94)	81.99(65)	77.52(66)	**84.29(77)**	76.44	79.28(89)	74.48(94)	79.46(65)	83.42(66)	**83.99(77)**
Eye	81.74	77.63(10)	73.19(8)	67.25(8)	88.75(5)	**92.36(6)**	60.11	55.39(10)	54.18(8)	57.62(8)	59.38(5)	**61.25(6)**
Letter	73.28	69.97(8)	71.11(9)	69.77(8)	71.92(5)	**74.71(6)**	79.77	71.89(8)	74.13(9)	77.93(8)	78.19(5)	**82.46(6)**
OLD	69.87	65.28(24)	**74.29(38)**	71.18(39)	67.99(41)	72.56(27)	74.98	73.36(24)	81.77(38)	82.68(39)	64.29(41)	**87.94(27)**
RLS	81.64	83.21(15)	82.29(11)	79.66(12)	84.58(14)	**87.77(9)**	77.27	81.19(15)	83.24(11)	82.22(12)	79.97(14)	**90.03(9)**
Wine	98.15	81.30(2)	85.44(11)	92.34(10)	**98.15(12)**	**98.15(12)**	84.90	86.89(2)	66.81(11)	87.19(10)	79.28(12)	**93.47(12)**
Average	75.34	71.51	72.43	70.48	75.52	**78.74**	76.77	76.02	72.25	79.21	74.29	**83.86**

Table 7 Comparison accuracy(%) of five algorithms on KNN and Adaboost

Datasets	KNN Origin	NRSAR	FEAR	DRAQR	SFS	IFS-A	Adaboost Origin	NRSAR	FEAR	DRAQR	SFS	IFS-A
Abalone	92.16	77.35(3)	74.18(2)	88.45(3)	81.47(1)	**95.29(2)**	75.29	74.98(3)	73.78(2)	77.97(3)	84.25(1)	**87.73(2)**
Bcc	77.67	74.69(4)	67.77(5)	69.96(9)	73.90(5)	**80.74(6)**	79.57	74.19(4)	**81.76(5)**	74.87(9)	69.77(5)	81.31(6)
Codon	71.07	69.50(11)	69.97(13)	72.09(23)	75.81(14)	**79.69(8)**	77.63	72.89(11)	71.01(13)	70.18(23)	68.80(14)	**81.57(8)**
DMT	84.49	85.69(79)	86.74(84)	83.97(76)	87.22(89)	**90.01(57)**	83.29	79.28(79)	77.45(84)	82.49(76)	84.47(89)	**91.78(57)**
Eye	76.29	73.46(10)	77.28(8)	81.11(8)	79.99(5)	**82.33(6)**	69.92	59.37(10)	58.82(8)	63.35(8)	65.81(5)	**70.01(6)**
Letter	73.99	67.89(8)	72.39(9)	77.71(8)	76.58(5)	**82.11(6)**	73.19	66.81(8)	73.79(9)	75.21(8)	74.86(5)	**80.09(6)**
OLD	73.46	71.16(35)	72.29(34)	**75.58(42)**	74.42(41)	75.09(38)	82.23	84.47(35)	83.39(34)	85.57(42)	79.99(41)	**87.57(38)**
RLS	67.72	59.88(11)	63.87(12)	66.99(8)	69.20(12)	**68.81(14)**	69.25	71.10(11)	72.24(12)	74.49(8)	67.72(12)	**77.43(14)**
Wine	90.21	65.36(2)	96.08(11)	91.78(10)	98.15(12)	**99.35(12)**	70.56	67.72(2)	71.83(11)	72.44(10)	75.28(12)	**79.54(12)**
Average	78.56	71.66	75.62	78.63	79.64	**83.71**	75.66	72.31	73.79	75.17	74.55	**81.89**

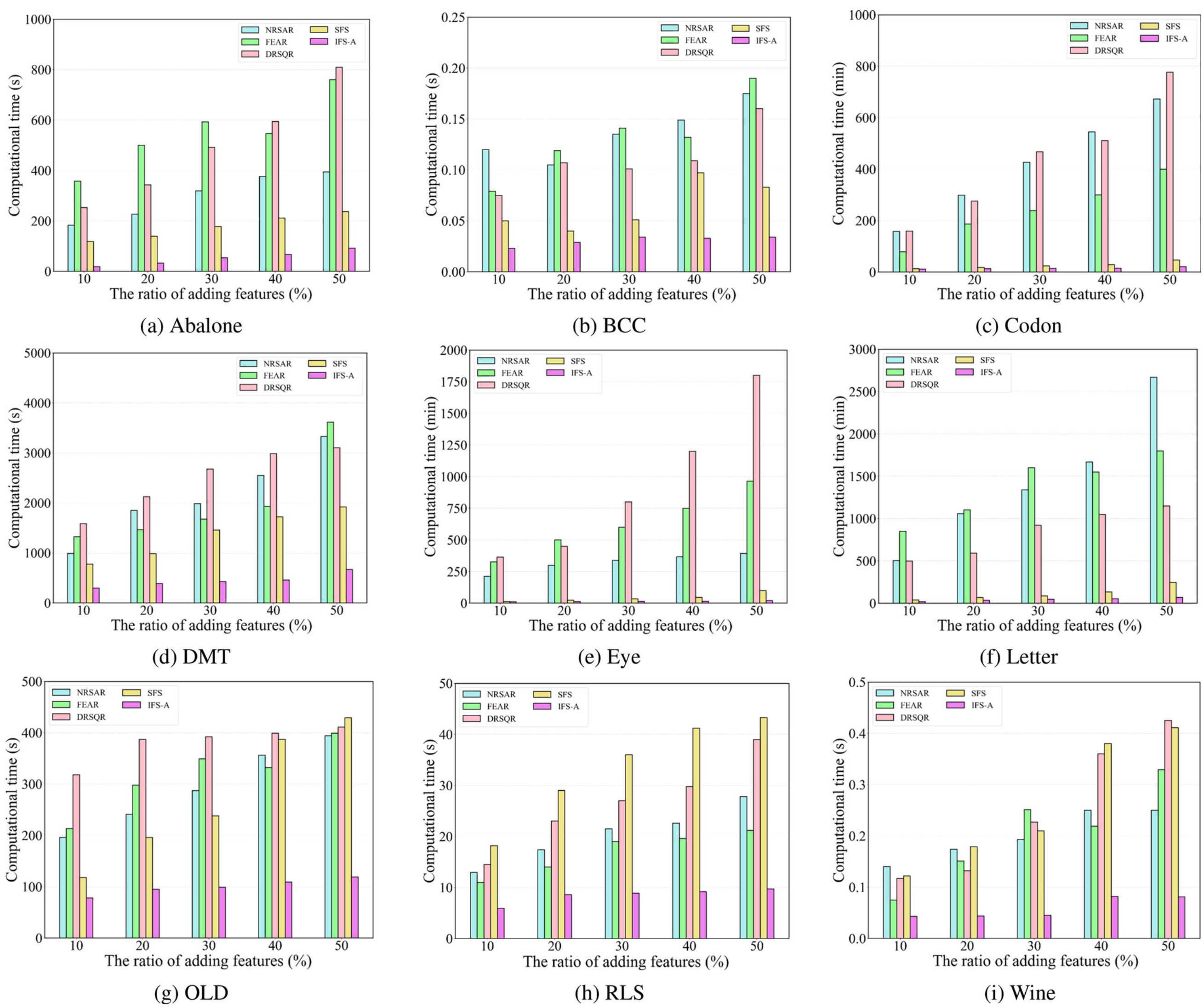
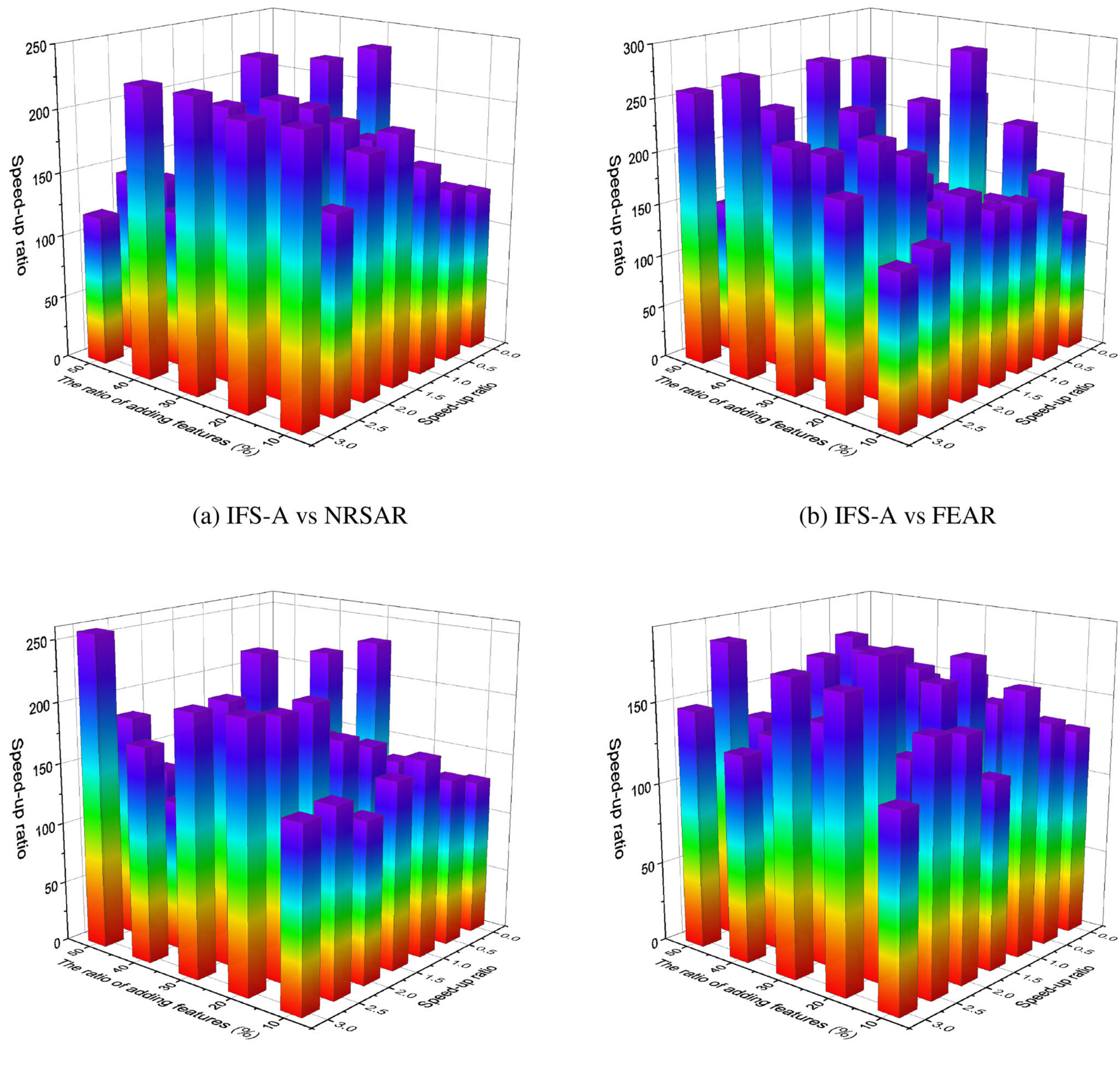


Fig. 1 The time taken by various algorithms when features are added at different ratios



(a) IFS-A vs NRSAR

(b) IFS-A vs FEAR

(c) IFS-A vs DRSQR

(d) IFS-A vs SFS

Fig. 2 The speed-up ratios between IFS-A and other four algorithms

where f denotes the different algorithms, n denotes the number of samples, $r_f(o, l)$ represents the ordinal position of label l among all the predicted labels.

RL reflects the average probability of an irrelevant label being ranked before a relevant label in the predicted label ranking for all samples. The RL formula is expressed as follows:

$$RL(f) = \frac{1}{n}\sum_{i=1}^{n}\frac{1}{|R_i|\,|\overline{R}_i|}\left|\left\{(l,k) \mid r_f(o_i,l) \geqslant r_f(o_i,k),\,(l,k) \in R_i \times \overline{R}_i\right\}\right| \tag{21}$$

where $\overline{R}_i$ denotes the complement of R_i.

The performance of the IFS-A algorithm and the other four algorithms on nine datasets is presented in Tables 8 and 9. These tables showcase the average performance values of the four classifiers based on two evaluation indexes. A higher value for the AP evaluation index indicates superior algorithm performance, while a lower value for the RL evaluation index suggests better algorithm performance. Based on these results, it can be observed that the IFS-A algorithm demonstrates superior performance compared to others.

Table 8 Comparing the performance of algorithms using the AP evaluation metric

Datasets	NRSAR	FEAR	DRAQR	SFS	IFS-A
Abalone	0.3927±0.0356	0.3468±0.0557	0.2213±0.0562	0.7388±0.0192	**0.7422±0.0031**
Bcc	0.5928±0.0213	0.5130±0.0197	0.5213±0.0211	0.3267±0.0325	**0.6001±0.0112**
Codon	0.4147±0.0426	0.3927±0.0233	0.3527±0.0477	0.2419±0.0334	**0.5122±0.0123**
DMT	0.5219±0.0164	0.4287±0.0243	0.4372±0.0366	0.2769±0.0170	**0.5388±0.0117**
Eye	0.3436±0.0142	0.4435±0.0119	0.4178±0.0385	0.4656±0.0287	**0.4772±0.0174**
Letter	0.4005±0.0133	0.5112±0.0344	0.5231±0.0441	0.7211±0.0291	**0.7388±0.0196**
OLD	0.4387±0.0147	0.5032±0.0478	0.6198±0.0367	0.4232±0.0189	**0.6278±0.0369**
RLS	0.4143±0.0223	0.4248±0.0267	0.4331±0.0758	0.4041±0.0363	**0.4892±0.0737**
Wine	0.5327±0.0184	0.5243±0.0145	0.4769±0.0231	0.5627±0.0190	**0.5797±0.0107**
Average	0.4502	0.4542	0.4448	0.4623	**0.5896**

Table 9 Comparing the performance of algorithms using the RL evaluation metric

Datasets	NRSAR	FEAR	DRAQR	SFS	IFS-A
Abalone	0.4033±0.2694	0.3834±0.2305	0.3953±0.2451	0.3993±0.2656	**0.3488±0.2280**
Bcc	0.2628±0.4424	0.2617±0.4151	0.2638±0.4500	0.2583±0.4075	**0.2369±0.3709**
Codon	0.5738±0.2754	0.5398±0.2473	0.5571±0.2826	0.5656±0.3558	**0.5132±0.2700**
DMT	0.6435±0.5327	0.4623±0.3116	0.5372±0.4242	0.4686±0.3582	**0.3179±0.2447**
Eye	0.7626±0.4820	0.7503±0.4508	0.7494±0.3829	0.7528±0.4514	**0.7159±0.4871**
Letter	0.4917±0.3243	0.4665±0.3927	0.4854±0.2452	0.4958±0.2351	**0.4355±0.1752**
OLD	0.4168±0.3927	0.4359±0.3267	0.3687±0.2153	0.5254±0.3245	**0.3154±0.5423**
RLS	0.6173±0.2160	0.5793±0.2034	0.6289±0.2894	0.5760±0.2168	**0.5723±0.3329**
Wine	0.4909±0.3428	0.4676±0.2873	0.4734±0.4204	0.4892±0.3466	**0.4341±0.2797**
Average	0.5181	0.4830	0.4955	0.5034	**0.4322**

Table 10 Comparison accuracy(%) of five algorithms on BN and RT

Datasets	BN Origin	NRSAR	FEAR	DRAQR	SFS	IFS-D	RT Origin	NRSAR	FEAR	DRAQR	SFS	IFS-D
Abalone	91.13	87.65(3)	86.78(2)	88.19(2)	83.37(1)	**92.01(2)**	70.56	72.20(3)	74.75(2)	66.98(2)	64.78(1)	**75.99(2)**
Bcc	88.97	79.88(4)	76.54(4)	85.67(8)	86.62(5)	**89.47(6)**	84.48	79.77(4)	82.21(4)	**85.79(8)**	75.92(5)	73.39(6)
Codon	84.76	75.62(21)	73.64(17)	82.57(13)	81.90(15)	**84.82(9)**	74.72	70.79(21)	68.97(17)	70.45(13)	71.47(15)	**79.28(9)**
DMT	75.59	82.28(73)	81.54(64)	82.20(43)	76.69(52)	**84.47(77)**	82.64	85.41(73)	86.64(64)	83.26(43)	79.94(52)	**87.74(77)**
Eye	84.16	78.54(7)	75.53(8)	81.56(6)	82.05(3)	**85.38(4)**	58.24	57.73(7)	57.64(8)	50.59(6)	52.94(3)	**60.11(4)**
Letter	83.99	77.89(6)	79.31(7)	87.17(3)	83.46(5)	**87.51(9)**	71.88	67.71(6)	69.98(7)	71.11(3)	67.78(5)	**73.89(9)**
OLD	83.46	79.91(27)	83.65(29)	**86.87(34)**	79.95(38)	85.51(39)	76.92	66.73(27)	71.98(29)	79.92(34)	82.18(38)	**83.33(39)**
RLS	81.42	78.42(14)	74.52(13)	78.54(11)	75.59(10)	**84.58(9)**	74.59	71.64(14)	76.27(13)	74.75(11)	80.06(10)	**83.77(9)**
Wine	82.97	77.39(2)	79.55(9)	84.22(8)	85.88(10)	**86.94(10)**	85.59	74.98(2)	63.97(9)	87.53(8)	81.17(10)	**88.04(10)**
Average	84.05	79.73	79.01	84.11	81.72	**86.74**	75.51	71.88	72.49	74.49	72.92	**78.39**

Table 11 Comparison accuracy(%) of five algorithms on KNN and Adaboost

Datasets	KNN Origin	NRSAR	FEAR	DRAQR	SFS	IFS-D	Adaboost Origin	NRSAR	FEAR	DRAQR	SFS	IFS-D
Abalone	65.36	67.88(3)	64.89(2)	70.13(2)	64.97(1)	**72.11(2)**	75.62	77.64(3)	69.94(2)	72.79(2)	74.98(1)	**76.85(2)**
Bcc	82.23	81.11(4)	79.56(4)	78.87(8)	81.87(5)	**83.54(6)**	80.01	**84.41(4)**	82.64(4)	83.38(8)	82.74(5)	83.99(6)
Codon	72.65	70.79(21)	71.29(17)	73.85(13)	71.78(15)	**75.43(9)**	77.27	75.83(21)	72.55(17)	73.02(13)	73.42(15)	**80.01(9)**
DMT	60.03	63.26(64)	64.27(75)	59.92(66)	64.68(78)	**69.91(73)**	65.29	70.87(64)	72.34(75)	64.32(66)	69.75(78)	**74.65(73)**
Eye	73.34	76.46(7)	74.95(8)	68.38(6)	80.08(3)	**81.39(4)**	71.65	62.76(7)	70.54(8)	70.78(6)	68.79(3)	**72.77(4)**
Letter	71.83	72.99(6)	70.94(7)	70.83(3)	75.77(5)	**76.74(9)**	79.22	69.82(6)	75.21(7)	75.53(3)	77.48(5)	**80.12(9)**
OLD	86.89	**89.91(21)**	84.78(17)	83.96(34)	86.78(35)	89.07(27)	90.57	91.42(21)	92.03(17)	86.78(34)	89.96(35)	**93.31(27)**
RLS	65.57	66.93(10)	61.43(13)	71.13(11)	73.49(10)	**75.76(11)**	78.95	75.87(10)	76.83(13)	81.15(11)	82.65(10)	**84.77(11)**
Wine	64.80	65.24(2)	63.68(9)	59.97(8)	65.51(10)	**66.86(10)**	77.69	62.65(2)	71.44(9)	73.98(8)	80.06(10)	**83.55(10)**
Average	71.41	72.73	70.64	70.78	73.88	**76.76**	77.36	74.59	75.95	75.75	77.31	**81.11**

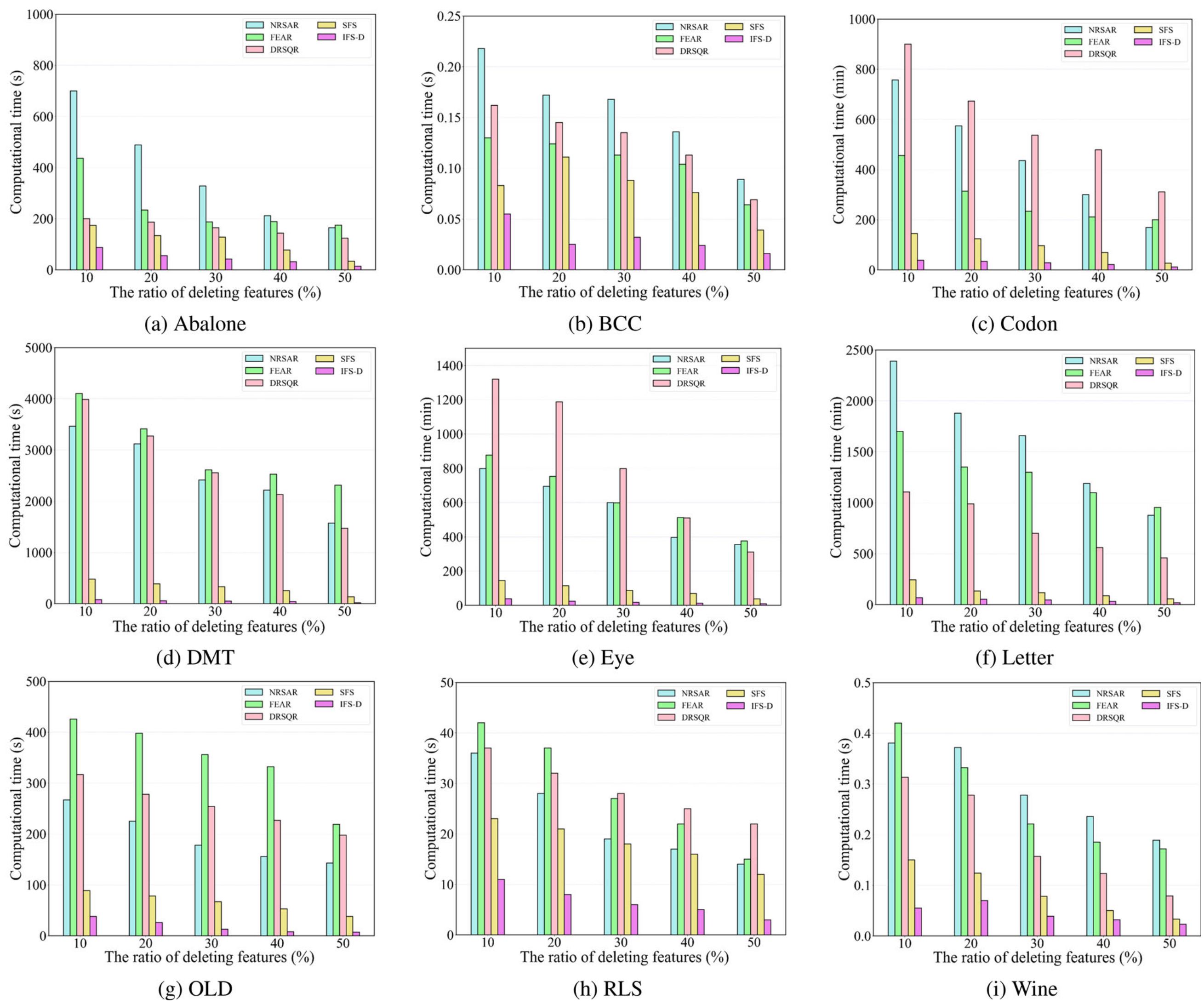
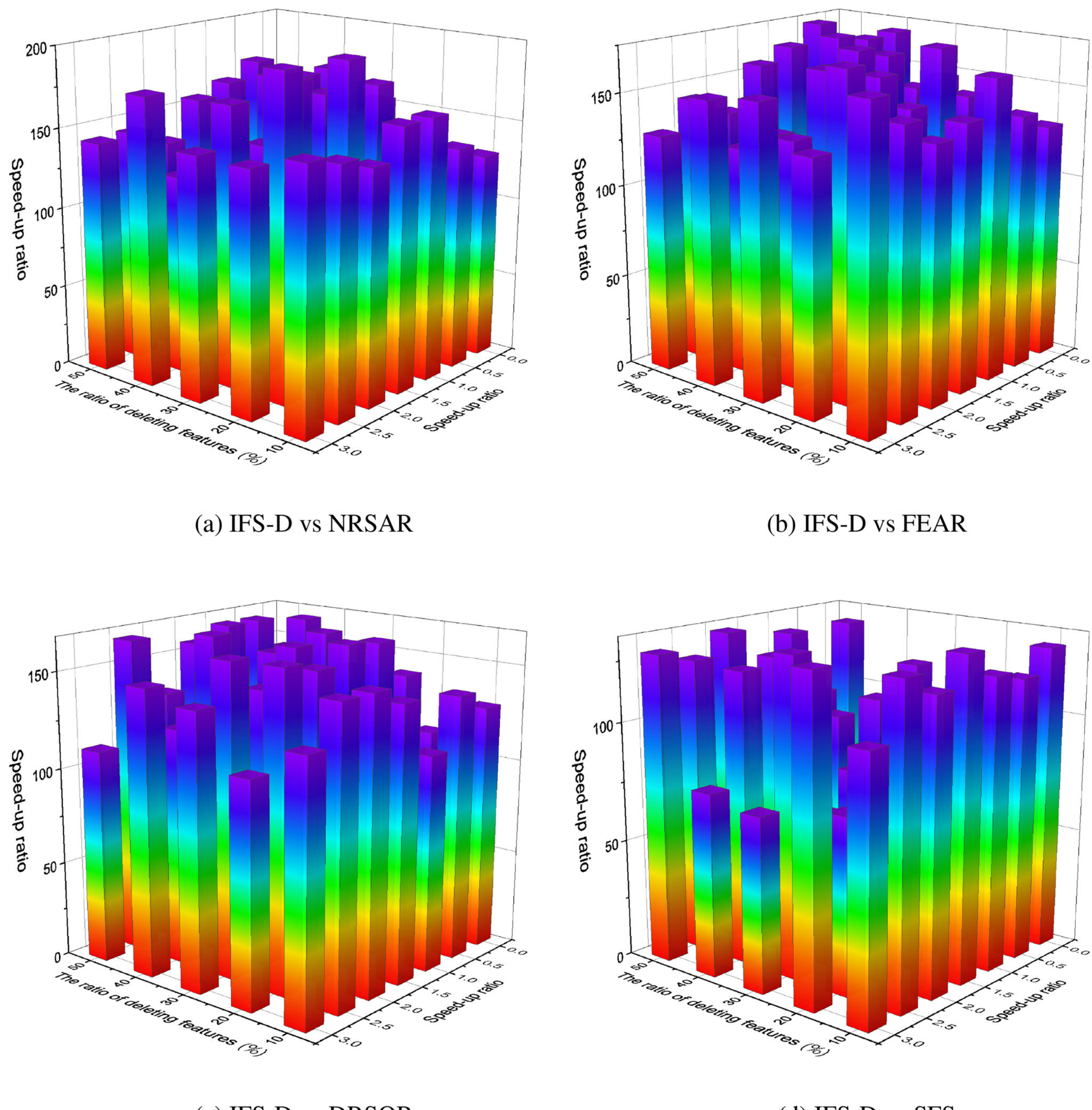


Fig. 3 The time taken by various algorithms when features are added at different ratios



(a) IFS-D vs NRSAR

(b) IFS-D vs FEAR

(c) IFS-D vs DRSQR

(d) IFS-D vs SFS

Fig. 4 The speed-up ratios between IFS-D and other four algorithms

5.1.4 Overview of IFS-A

By conducting comparative experiments to evaluate the classification efficiency, feature selection time, and other indicators of various algorithms, it can be inferred that the IFS-A algorithm outperforms other algorithms in terms of performance. In comparison with alternative algorithms, the computation time required by the IFS-A algorithm for achieving feasible reduction is significantly shorter while yielding more accurate results.

5.2 Experimental effect analysis of IFS-D algorithm

The algorithm IFS-D is analyzed from three perspectives: classification accuracy, algorithm time, and index performance evaluation.

Table 12 Comparing the performance of algorithms using the AP evaluation metric

Datasets	NRSAR	FEAR	DRAQR	SFS	IFS-D
Abalone	0.4960±0.0241	0.4637±0.0209	0.4723±0.0437	0.4883±0.0192	**0.4997±0.0204**
Bcc	0.1363±0.0264	0.1367±0.0260	0.1363±0.0264	0.1353±0.0253	**0.1411±0.0218**
Codon	0.8103±0.0219	0.7713±0.0314	0.7783±0.0259	0.7970±0.0321	**0.8276±0.0294**
DMT	0.4267±0.2386	0.4752±0.1531	0.4637±0.2412	0.5271±0.1537	**0.5877±0.1321**
Eye	0.7557±0.0220	0.7367±0.0198	0.7490±0.0286	0.7610±0.0241	**0.7755±0.0431**
Letter	0.4726±0.1216	0.4511±0.0237	0.4168±0.0356	0.4223±0.0145	**0.4889±0.0227**
OLD	0.6592±0.0179	0.6325±0.0286	0.6418±0.0375	0.7141±0.0425	**0.7227±0.0315**
RLS	0.7429±0.2165	0.6787±0.0121	0.7132±0.1928	0.7146±0.0190	**0.7698±0.0421**
Wine	0.4797±0.0220	0.4633±0.0275	0.4607±0.0321	0.4740±0.0271	**0.4845±0.0252**
Average	0.5533	0.5344	0.5369	0.5593	**0.5886**

5.2.1 Comparison of classification accuracy

In this section, we compare the classification accuracy of our proposed IFS-D algorithm with four other algorithms. For each data set in Table 5, we randomly select 50% of the features as the initial feature set, while deleting the remaining 50%. We then apply algorithms IFS-D, SFS, DRSQR, FEAR, and NRSAR to compute new reductions based on the modified feature set. The experimental results can be found in Tables 10 and 11, where 'Origin' represents the classification accuracy of the original attribute set. Based on the chart provided above, it can be observed that algorithm IFS-D consistently outperforms other algorithms in terms of classification accuracy across all scenarios. Moreover, its average score significantly surpasses others, indicating a remarkably high classification accuracy for the HRA-D algorithm.

5.2.2 Comparison of feature selection time

In this section, we assess the efficiency of the IFS-D algorithm and conduct a comparison with four other algorithms in terms of computation time and acceleration ratio. For each data set listed in Table 5, we generate five test sets. Initially, we randomly select 50% of the attributes to form the origin feature set. Subsequently, we randomly eliminate attributes from the remaining 50% to create dynamic data sets for testing purposes (specifically, random selection and deletion of 10%, 20%, 30%, 40%, and 50% of the rest attributes in the original set). Figure 3 illustrates detailed variations in all algorithms when attributes changes. The x-axis represents the capacity of deleted attributes sets, while the y-axis represents computation time. From Fig. 3, it is evident that as attribute sets decrease consistently, so does computation time for all five algorithms. However, algorithm IFS-D exhibits significantly lower computation times compared to other algorithms across all subgraphs. This effect is particularly pronounced for large data sets, indicating a substantial time-saving advantage offered by algorithm IFS-D's high efficiency.

Following that, we proceed to validate the effectiveness of the IFS-D algorithm based on its acceleration ratio. By examining the results presented in Fig. 3, we computed the acceleration ratio of IFS-D relative to the others. The experimental findings are depicted in Fig. 4. As evident from Fig. 4, IFS-D consistently demonstrates a positive acceleration ratio

Table 13 Comparing the performance of algorithms using the RL evaluation metric

Datasets	NRSAR	FEAR	DRAQR	SFS	IFS-D
Abalone	0.0508±0.0011	0.0478±0.0015	0.0488±0.0019	0.0504±0.0012	**0.0443±0.0016**
Bcc	0.0287±0.0030	0.0287±0.0029	0.0287±0.0029	0.0285±0.0030	**0.0281±0.0028**
Codon	0.0653±0.0025	0.0647±0.0024	0.0641±0.0025	0.0649±0.0026	**0.0638±0.0023**
DMT	0.0397±0.0014	0.0369±0.0017	0.0463±0.0022	0.0427±0.0011	**0.0299±0.0002**
Eye	0.0357±0.0009	0.0356±0.0009	0.0357±0.0009	0.0356±0.0009	**0.0350±0.0008**
Letter	0.0264±0.0001	0.0287±0.0013	0.0267±0.0003	0.0457±0.0015	**0.0190±0.0011**
OLD	0.0417±0.0011	0.0478±0.0013	0.0428±0.0017	0.0452±0.0016	**0.0348±0.0007**
RLS	0.0397±0.0002	0.0537±0.0018	0.0478±0.0054	0.0576±0.0007	**0.0243±0.0014**
Wine	0.0442±0.0024	0.0418±0.0021	0.0426±0.0029	0.0441±0.0025	**0.0396±0.0024**
Average	0.0414	0.0429	0.0426	0.0461	**0.0354**

across all datasets, indicating its superior speed compared to the employed alternative algorithms. Furthermore, for larger datasets in particular, IFS-D surpasses the performance of the other four algorithms by significant margins ranging from tens to even hundreds of times. These discoveries reaffirm that maintaining algorithmic efficiency remains a prominent characteristic of IFS-D.

5.2.3 Analysis of evaluation index

According to the experimental evaluation index of Section 5.1.3, Tables 12 and 13 present the experimental results of the IFS-D and other algorithms on nine datasets using two evaluation metrics (averaging the effects of the four classifiers). A higher value for the AP evaluation metric indicates superior algorithm performance, while smaller values for RL evaluation metrics indicate better algorithm performance.

It is evident from the aforementioned findings that algorithm IFS-D outperforms the others.

5.2.4 Overview of IFS-D

The comparative study conducted on the algorithm, taking into account its effectiveness, efficiency, and performance evaluation, leads to the finding that our proposed IFS-D algorithm surpasses other algorithms. In terms of computation time needed for obtaining feasible reduction, the IFS-D algorithm demonstrates significantly shorter compared to alternative approaches while achieving superior accuracy outcomes.

6 Conclusion

In this paper, we present an incremental feature selection algorithm based on matrix dominance conditional entropy. Firstly, we provide an introduction to the fundamental knowledge of feature selection. Secondly, we introduce the concepts related to the research, such as dominance relation matrix and dominance conditional entropy. Then, we propose two novel algorithms for incremental attribute reduction: IFS-A and IFS-D. Finally, we validate the effectiveness of our algorithm through experiments.

The exploration of dynamic feature selection algorithms in more intricate dynamic data environments is an immensely significant avenue for further investigation. Specifically, our forthcoming research will concentrate on three key areas: (1) devising an incremental algorithm for feature selection to effectively handle fluctuations in the number of objects within a dataset; (2) implementing the dynamic attribute reduction algorithm into the prevailing fuzzy rough set model; and (3) conducting additional examinations on incremental methods for reducing attributes in set-valued decision information systems.

Acknowledgements This work is supported by the National Natural Science Foundation of China (No. 62376229) and the Natural Science Foundation of Chongqing (No. CSTB2023NSCQ-LZX0027).

Author Contributions Weihua Xu: Conceptualization, Funding acquisition, Investigation, Methodology, Project administration, Supervision, Validation. Yifei Yang: Data curation, Methodology, Software, Visualization, Writing-original draft, Writing-review & editing. Yi Ding: Investigation, Language polishing, Methodology, Typography, Validation, Writing-review & editing. Xiyang Chen: Investigation, Methodology. Xiaofang Lv: Investigation, Methodology.

Data Availability No data was used for the research described in the article.

Declarations

Competing interest The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

1. Lin Y, Hu Q, Liu J, Li J, Wu X (2017) Streaming feature selection for multilabel learning based on fuzzy mutual information. IEEE Trans Fuzzy Syst 25(6):1491–1507
2. Hu Q, Zhang L, Zhou Y, Pedrycz W (2018) Large-scale multimodality attribute reduction with multi-kernel fuzzy rough sets. IEEE Trans Fuzzy Syst 26(1):226–238
3. Lin Y, Li Y, Wang C, Chen J (2018) Attribute reduction for multilabel learning with fuzzy rough set. Knowl-Based Syst 152:51–61
4. Tan A, Wu W, Qian Y, Liang J, Chen J, Li J (2019) Intuitionistic fuzzy rough set-based granular structures and attribute subset selection. IEEE Trans Fuzzy Syst 27(3):527–539
5. Chen Degang, Yang Yanyan, Dong Ze (2016) An incremental algorithm for attribute reduction with variable precision rough sets. Appl Soft Comput 45:129–149
6. Jing Y, Li T, Luo C, Horng S, Wang G, Yu Z (2016) An incremental approach for attribute reduction based on knowledge granularity. Knowl-Based Syst 104:24–38
7. Yang Y, Chen D, Hui W (2017) Active sample selection based incremental algorithm for attribute reduction with rough sets. IEEE Trans Fuzzy Syst 25(4):825–838
8. Lang G, Cai M, Fujita H, Xiao Q (2018) Related families-based attribute reduction of dynamic covering decision information systems. Knowl-Based Syst 162:161–173
9. Shu W, Qian W, Xie Y (2019) Incremental approaches for feature selection from dynamic data with the variation of multiple objects. Knowl-Based Syst 163:320–331
10. LeCun Y, Boser B, Denker JS, Henderson D, Howard RE, Hubbard W, Jackel LD (1989) Backpropagation applied to handwritten zip code recognition. Neural Comput 1(4):541–551
11. Hinton GE, Osindero S, Yee YW (2006) A fast learning algorithm for deep belief nets. Neural Comput 18(7):1527–1554
12. Williams RJ, Zipser D (1989) A learning algorithm for continually running fully recurrent neural networks. Neural Comput 1(2):270–280

13. Zhao L, Hu Q, Wang WW (2015) Heterogeneous feature selection with multi-modal deep neural networks and sparse group lasso. IEEE Trans Multimed 17(11):1936–1948
14. Semwal VB, Mondal K, Nandi GC (2017) Robust and accurate feature selection for humanoid push recovery and classification: deep learning approach. Neural Comput Appl 28(3):565–574
15. Chen J, Mao Z, Zheng R, Huang Y, He L (2019) Feature selection of deep learning models for EEG-based RSVP target detection. IEICE Trans Inf Syst 102–D(4):836–844
16. Niu T, Wang J, Lu H, Yang W, Du P (2020) Developing a deep learning framework with two-stage feature selection for multivariate financial time series forecasting. Expert Syst Appl 148:1–17
17. Zou Q, Ni L, Zhang T, Wang Q (2015) Deep learning based feature selection for remote sensing scene classification. IEEE Geosci Remote Sens Lett 12(11):2321–2325
18. Shi H, Li H, Zhang D, Cheng C, Cao X (2018) An efficient feature generation approach based on deep learning and feature selection techniques for traffic classification. Comput Netw 132:81–98
19. Ding W, Lin C, Pedrycz W (2020) Multiple relevant feature ensemble selection based on multilayer co-evolutionary consensus mapreduce. IEEE Trans Cybern 50(2):425–439
20. Nag K, Pal NR (2020) Feature extraction and selection for parsimonious classifiers with multiobjective genetic programming. IEEE Trans Evol Comput 24(3):454–466
21. Labani M, Moradi P, Jalili M (2020) A multi-objective genetic algorithm for text feature selection using the relative discriminative criterion. Expert Syst Appl 149:1–21
22. Ma J, Gao X (2020) A filter-based feature construction and feature selection approach for classification using genetic programming. Knowl-Based Syst 196:1–14
23. Das AK, Das S, Ghosh A (2017) Ensemble feature selection using bi-objective genetic algorithm. Knowl-Based Syst 123:116–127
24. Li A, Xue B, Zhang M (2020) Multi-objective feature selection using hybridization of a genetic algorithm and direct multisearch for key quality characteristic selection. Inform Sci 523:245–265
25. Pawlak Z (1982) Rough sets. Int J Comput Inf Sci 11(5):341–356
26. Sun L, Zhang X, Qian Y, Xu J, Zhang S (2019) Feature selection using neighborhood entropy-based uncertainty measures for gene expression data classification. Inform Sci 502:18–41
27. Dai J, Hu Q, Hu H, Huang D (2018) Neighbor inconsistent pair selection for attribute reduction by rough set approach. IEEE Trans Fuzzy Syst 26(2):937–950
28. Wang C, Wang Y, Shao M, Qian Y, Chen D (2020) Fuzzy rough attribute reduction for categorical data. IEEE Trans Fuzzy Syst 28(5):818–830
29. Greco S, Matarazzo B, Slowinski R (1999) Rough approximation of a preference relation by dominance relations. European J Oper Res 117(1):63–83
30. Greco Salvatore, Matarazzo Benedetto, Slowinski Roman (2001) Rough sets theory for multicriteria decision analysis. European J Oper Res 129(1):1–47
31. Błaszczyński J, Greco S, Słowiński R, Szelg M (2009) Monotonic variable consistency rough set approaches. Internat J Approx Reason 50(7):979–999
32. Kotłowski W, Dembczyński K, Greco S, Słowiński R (2008) Stochastic dominance-based rough set model for ordinal classification. Inform Sci 178(21):4019–4037
33. Ali A, Ali MI, Rehman N (2019) Soft dominance based rough sets with applications in information systems. Internat J Approx Reason 113:171–195
34. Zhang X, Chen D, Tsang ECC (2017) Generalized dominance rough set models for the dominance intuitionistic fuzzy information systems. Inform Sci 378:1–25
35. Liang J, Wang F, Dang C, Qian Y (2014) A group incremental approach to feature selection applying rough set technique. IEEE Trans Knowl Data Eng 26:294–308
36. Zhang X, Mei C, Chen D, Yang Y, Li J (2020) Active incremental feature selection using a fuzzy rough set-based information entropy. IEEE Trans Fuzzy Syst 28(5):901–915
37. Yang Y, Chen D, Hui W, Wang X (2018) Incremental perspective for feature selection based on fuzzy rough sets. IEEE Trans Fuzzy Syst 26(3):1257–1273
38. Yang Y, Song S, Chen D, Zhang X (2020) Discernible neighborhood counting based incremental feature selection for heterogeneous data. Int J Mach Learn Cybern 11(5):1115–1127
39. Shu W, Qian W, Xie Y (2020) Incremental feature selection for dynamic hybrid data using neighborhood rough set. Knowl-Based Syst 194:1–15
40. Liu Y, Zheng L, Xiu Y, Yin H, Zhao S, Wang X, Chen H, Li C (2020) Discernibility matrix based incremental feature selection on fused decision tables. Internat J Approx Reason 118:1–26
41. Chen D, Dong L, Mi J (2020) Incremental mechanism of attribute reduction based on discernible relations for dynamically increasing attribute. Soft Comput 24(1):321–332
42. Wang F, Liang J, Qian Y (2013) Attribute reduction: A dimension incremental strategy. Knowl-Based Syst 39:95–108
43. Zeng A, Li T, Liu D, Zhang J, Chen H (2015) A fuzzy rough set approach for incremental feature selection on hybrid information systems. Fuzzy Sets Syst 258:39–60
44. Wang F, Liang J, Dang C (2013) Attribute reduction for dynamic data sets. Appl Soft Comput 13(1):676–689
45. Wei W, Wu X, Liang J, Cui J, Sun Y (2018) Discernibility matrix based incremental attribute reduction for dynamic data. Knowl-Based Syst 140:142–157
46. Wei W, Song P, Liang J, Wu X (2019) Accelerating incremental attribute reduction algorithm by compacting a decision table. Int J Mach Learn Cybern 10(9):2355–2373
47. Cai M, Lang G, Fujita H, Li Z, Yang T (2019) Incremental approaches to updating reducts under dynamic covering granularity. Knowl-Based Syst 172:130–140
48. Dong L, Chen D (2020) Incremental attribute reduction with rough set for dynamic datasets with simultaneously increasing samples and attributes. Int J Mach Learn Cybern 11(6):1339–1355
49. Jing Y, Li T, Fujita H, Wang B, Cheng N (2018) An incremental attribute reduction method for dynamic data mining. Inform Sci 465:202–218
50. Shannon C, Weaver W (1948) The mathematical theory of communication. Bell Syst. Tech. J. 27(3/4):373–423
51. Hu Q, Che X, Zhang L, Zhang D, Guo M, Yu D (2012) Rank entropy based decision trees for monotonic classification. IEEE Trans Knowl Data Eng 24(11):2052–2064
52. Inuiguchi M, Yoshioka Y, Kusunoki Y (2009) Variable-precision dominance-based rough set approach and attribute reduction. Internat J Approx Reason 50(8):1199–1214

Weihua Xu received the M.Sc. degree in mathematics from the School of Mathematics and Information Sciences, Guangxi University, Nanning, China, in 2004, and the Ph.D. degree in mathematics from the School of Sciences, Xi'an Jiaotong University, Xi'an, China, in 2007. He is currently a Professor with the College of Artificial Intelligence, Southwest University, Chongqing, China. He serves on the editorial boards of several international journals and has published four monographs and over 220 articles in international journals. His current research interests include granular computing, cognitive computing and information fusion.

Yifei Yang is currently working toward the M.Sc. degree in Data science and Big Data technology at the the College of Artificial Intelligence, Southwest University, Chongqing, China. His research interests cover data statistics and analysis, convolutional neural network, attribute reduction.

Yi Ding received the B.Sc. Degree in Electronic Information Engineering from Westa College of Southwest University, Chongqing, China, in 2021. He is currently working toward the M.Sc. degree in Computer Science and Technology at the the College of Artificial Intelligence, Southwest University, Chongqing, China. His research interests cover concept-cognitive learning, feature selection, granular computing and knowledge discovery.

Xiyang Chen is currently working toward the M.Sc. degree in Computer Technology at the College of Computer Science and Technology, Xi'an University of Science and Technology, Xi'an, China. His research interests include machine learning, deep learning, Knowledge Graphs, and applications in coal mine safety.

Xiaofang Lv is currently working toward the B.Sc. degree in Bioscience at the the College of Life Science, Southwest University, Chongqing, China. Her research interests cover biostatistics, biological diversity, brain cognitive science.

Authors and Affiliations

Weihua Xu[1] · **Yifei Yang**[1] · **Yi Ding**[1] · **Xiyang Chen**[2] · **Xiaofang Lv**[3]

Weihua Xu
chxuwh@gmail.com

Yifei Yang
Yifei.Yang.yyf2002@gmail.com

Xiyang Chen
1192766550@qq.com

Xiaofang Lv
15194235153@163.com

[1] College of Artificial Intelligence, Southwest University, Chongqing 400715, People's Republic of China

[2] College of Computer Science and Technology, Xi'an University of Science and Technology, Xi'an 710600, People's Republic of China

[3] College of Life Sciences, Southwest University, Chongqing 400715, People's Republic of China