Knowledge-Based Systems 212 (2021) 106583

Contents lists available at ScienceDirect

Knowledge-Based Systems

journal homepage: www.elsevier.com/locate/knosys

Incremental attribute reduction approaches for ordered data with time-evolving objects

Binbin Sang [a,b], Hongmei Chen [a,b,*], Lei Yang [c], Dapeng Zhou [a,b], Tianrui Li [a,b], Weihua Xu [d]

[a] Institute of Artificial Intelligence, School of Information Science and Technology, Southwest Jiaotong University, Chengdu 611756, PR China
[b] National Engineering Laboratory of Integrated Transportation Big Data Application Technology, Southwest Jiaotong University, Chengdu 611756, PR China
[c] School of Mathematics, Southwest Jiaotong University, Chengdu 611756, PR China
[d] School of Artificial Intelligence, Southwest University, Chongqing 400715, PR China

ARTICLE INFO

Article history:
Received 3 July 2020
Received in revised form 13 September 2020
Accepted 29 October 2020
Available online 5 November 2020

Keywords:
Ordered decision system
Dominance-based rough set approach
Attribute reduction
Dominance conditional entropy
Incremental algorithm

ABSTRACT

Dominance-based rough set approach (DRSA) is widely applied to multi-criteria decision analysis and sorting problems for data with preference-ordered relation, where attribute reduction is an important research field. At present, based on DRSA, many traditional attribute reduction approaches are extended to process a static ordered data. In real-world applications, ordered data with time-evolving objects widely exist, which is called a dynamic ordered data. However, for dynamic ordered data, employing these existing approaches to compute reducts is very time-consuming, since they need to recalculate knowledge from scratch when multiple objects vary. Incremental updating method can effectively complete the dynamic learning task, because it can acquire new knowledge based on previous knowledge. Inspired by this, this work studies incremental attribute reduction approaches for dynamic ordered data in DRSA framework. First, matrix-based method for calculating the dominance conditional entropy is investigated. Next, the updating principles of the dominance relation matrix and dominance diagonal matrix are studied when objects vary. Finally, two incremental algorithms of attribute reduction are proposed when multiple objects are added to or deleted from an ordered decision system, respectively. Experiments on different datasets provided by University of California at Irvine (UCI) are conducted to evaluate the proposed algorithms. Experimental results show that the proposed incremental algorithms can effectively and efficiently accomplish the task of attribute reduction in dynamic ordered data.

1. Introduction

Feature selection, also known as attribute reduction in rough set theory (RST), has elicited widespread attention in data mining [1–4]. This approach aim to remove redundant or irrelevant attributes from complex data and achieve the goals of reducing dimensionality, avoiding overfitting, thereby saving the time and space cost of calculation. In real-life applications, datasets usually exhibit dynamic characteristics over time-evolving, i.e., dynamic datasets. This promotes the development of incremental techniques for attribute reduction [5–9]. Incremental attribute reduction approaches are widely studied, because they can effectively and efficiently complete attribute reduction tasks for dynamic datasets. However, the existing incremental approaches do not consider the monotonous ordered relation of samples in dynamic datasets. Motivated by this issue, this study focuses on incremental attribute reduction approaches for dynamic ordered datasets.

With the development of the information age, feature selection methods have been continuously improved and innovated as the complexity and diversity of data structures increase. Many excellent models and algorithms for feature selection have been proposed. Deep learning is an important machine learning method that can automatically learning features representation from complex data. Some commonly used deep learning based feature representation methods are convolutional neural networks (CNN) [10], restrict Boltzmann machine (RBM) [11] and recurrent neural networks (RNN) [12]. In recent years, deep learning has also been applied to feature selection. Zhao et al. presented a heterogeneous feature selection approach with multimodal deep neural networks and sparse group lasso algorithm [13]. Semwal et al. proposed a robust and accurate feature selection with deep learning approach for classification [14]. Chen et al. constructed feature selection of deep learning models for electroencephalogram-based a rapid serial visual presentation

* Corresponding author at: Institute of Artificial Intelligence, School of Information Science and Technology, Southwest Jiaotong University, Chengdu 611756, PR China.
E-mail address: hmchen@swjtu.edu.cn (H. Chen).

https://doi.org/10.1016/j.knosys.2020.106583
0950-7051/

target detection [15]. Furthermore, in real-world applications, deep learning based feature selection approaches are successfully used to financial forecast [16], remote sensing scene classification [17], and traffic classification [18], etc. Evolutionary algorithms simulate the behavior of living beings in nature, which is an effective way to solve the combination optimization problem in feature selection [19]. Genetic algorithm is one of most important evolutionary algorithm, which has been widely used in feature selection. Nag et al. studied feature extraction and selection approach for parsimonious classifiers with multi-objective genetic programming [20]. Labani et al. proposed a multi-objective genetic algorithm for text feature selection using the relative discriminative criterion [21]. Ma et al. designed a genetic programming based feature selection approach for classification [22]. Das et al. proposed an ensemble feature selection method using bi-objective genetic algorithm [23]. Li et al. constructed a multi-objective feature selection method using hybridization of a genetic algorithm and direct multi-search for key quality characteristic selection [24].

On feature selection, RST is an important theoretical basis [25–27]. RST proposed by Pawlak serves as an effective mathematical tool for dealing with inconsistent and uncertain information [28]. Objects with the same description based on equivalence relations compose the basic granules of knowledge. However, in ordinal classification tasks, RST ignores the dominance principle, which requires that objects with better descriptions should not get worse labels. In other words, RST does not consider inconsistencies from the criteria, namely, attributes with preference-ordered domains, which exist in credit level, article rank, and profit rate, etc. To offset this deficiency, Greco et al. proposed dominance-based rough set approach (DRSA) [29], which has been successfully applied to multi-criteria decision-making (MCDM) [30]. Since its inception, different extended DRSA models have been proposed, including monotonic variable consistency rough set approaches [31], stochastic dominance-based rough set model [32], soft dominance based rough sets [33], and generalized dominance rough set models [34], etc. Moreover, DRSA has been extended and applied to various types of ordered information systems [35–38]. Therefore, DRSA is an effective tool to cope with knowledge acquisition in ordered data, and it is the theoretical basis of this work.

Attribute reduction methods based on DRSA have been extensively studied in the past decades, and they are used to deal with static ordered dataset [39–43]. Although these methods can effectively remove redundant attributes from an ordered data, they ignore that ordered data usually evolve over time in real-life applications. For example, student's grade data is an example of ordered data. With the graduation and enrollment of students, this data has dynamic characteristics. For dynamic ordered datasets, employing these existing approaches to compute reducts are very time-consuming, since they need to recalculate knowledge from scratch when the dataset changes slightly. This defect increases the consumption of space and time. Accordingly, an effective and efficient attribute reduction method is urgently requested to process dynamic ordered datasets.

Incremental learning is an efficient approach, which can quickly acquire new knowledge from dynamic datasets by utilizing previous knowledge [44–46]. In the past decade, scholars have proposed numerous incremental learning algorithms for attribute reduction, and they can be generally divided into: objects-oriented, attributes-oriented, and attribute values-oriented.

- For the variation of objects. Liang et al. proposed an incremental updating feature subset approach via using information entropy [47]. Zhang et al. developed the incremental feature selection methods using a fuzzy rough set based information entropy with an active sample selection strategy [48]. Yang et al. studied incremental feature selection approaches with an active sample selection principle [49], and then the authors further presented an incremental feature selection method for dynamic heterogeneous data [50]. Shu et al. introduced an incremental feature selection algorithm for dynamic hybrid data [51]. For fused decision tables, Ye et al. designed an incremental feature selection approach via using the pseudo value of discernibility matrix [52]. Das et al. proposed a group incremental feature selection algorithm by using genetic algorithm [53].
- For the variation of attributes. Chen et al. introduced a discernible relations based incremental attribute reduction method for dynamically increasing attributes [54]. Wang et al. proposed an effective attribute reduction algorithm based on information entropy for dynamic datasets with attribute set changes [55]. In dynamic covering decision information systems, Lang et al. studied incremental algorithms based on related families [56]. Zeng et al. investigated a fuzzy rough set based incremental attribute reduction method on hybrid data [57].
- For the variation of attribute values. Wang et al. proposed an effective feature selection algorithm by using three representative entropies [58]. Wei et al. presented a dynamic feature selection approach by using discernibility matrix [59], and then they developed a accelerating incremental algorithm via using a technique of compressing decision table [60]. Cai et al. developed two incremental attribute reduction approaches for coarsening and refining covering granularity [61]. Furthermore, Dong and Chen designed a novel RST-based incremental attribute reduction algorithm for decision table with simultaneously increasing samples and attributes [62]. Jing et al. introduced incremental methods of calculating reducts for a decision table with simultaneously time-evolving objects and attributes [63].

It should be found that the aforementioned incremental attribute reduction approaches do not consider dynamic datasets with a preference-order relation. Thence, the existing incremental attribute reduction approaches are not suitable for dynamic ordered datasets, which motivates this study.

Uncertainty measure play a key role in attribute reduction approach, which is used to measure the importance of attribute and quantify the inconsistency in data. Information entropy, as a common uncertainty measure, has been widely concerned. Related researches were extended after information entropy was proposed by Shannon [64]. For ordered data, Hu et al. proposed ascending and decreasing rank conditional entropies, and they were used to evaluate the consistency degree of the ranking of objects under attributes and decisions [65]. In this study, we use the ascending rank conditional entropy (also called the dominance conditional entropy) as the uncertainty measure of attribute reduction approach.

Owing to the matrix form of information can simplify the calculation process and intuitively represent the construction of a method, matrix-based computing technology is widely used in incremental learning [66–69]. Moreover, the relation between objects in DRSA is a preference-ordered relation, which is antisymmetric. So the approximate space formed via DRSA is an irregular cover, not a regular division. Hence, using set representation to investigate issues in DRSA would be tedious and complicated, especially in a dynamic ordered data environment. Generally speaking, matrix-based technique is a simple and effective method for acquiring knowledge in covering-based approximate space, which can be effectively used for dynamic knowledge acquisition. Considering the advantages of matrix form, this work exploits matrix form to study the incremental mechanism of the dominance conditional entropy.

Based on the above discussion, we believe that how to employ incremental learning approach to effectively and efficiently select the necessary attributes in the dynamic ordered data is a topic worth discussing. Therefore, in this study, we develop incremental approaches of attribute reduction for dynamic ordered data in DRSA framework. The main contributions are three-folds: (1) We present the definitions of dominance relation matrix and dominance diagonal matrix in an ordered information system, and propose a method for calculating the dominance conditional entropy in matrix form. (2) The incremental calculation method of matrix-based dominance conditional entropy is proposed when multiple objects are added to or deleted from the ordered decision system. On this basis, we develop two incremental algorithms for attribute reduction. (3) Experiments on nine datasets from UCI show that the proposed algorithms are effective and efficient.

The paper is organized as follows. Section 2 reviews the related work. In Section 3, a matrix-based method for calculating dominance conditional entropy is presented and proved, and then a heuristic attribute reduction algorithm is introduced. In Section 4, the updating mechanisms of matrix-based dominance conditional entropy are proposed. On this basis, we develop two incremental attribute reduction algorithms when multiple objects are added to or deleted from an ordered decision system, respectively. Section 5 presents the experimental results on nine datasets, which demonstrate the effectiveness and efficiency of the proposed algorithms. Section 6 concludes the work and outline the future research.

2. Preliminaries

In this section, we briefly review some relevant knowledge in DRSA.

2.1. Ordered decision system and dominance-based rough set approach

Definition 2.1 (*[28], Information System*)**.** In RST, an information system is a 4-tuple $S = (U, AT, V, f)$, where $U = \{x_1, x_2, \ldots, x_n\}$ is a non-empty finite set of objects; AT is a non-empty finite set of attributes; $V = \bigcup_{a \in AT} V_a$, V_a is the domain of attribute a; $f : U \times AT \to V$ is the information function with $f(x, a) \in V_a$, $\forall a \in AT$ and $x \in U$.

In an information system, if the domain of an attribute is ordered in accordance with an increasing or decreasing preference, then the attribute is a criterion. The information system is called ordered information system (OIS) if all attributes are criterions. An OIS is denoted as $S^{\succeq} = (U, AT, V, f)$.

In real-world applications, decision makers usually know the order of criterion values within their domain or prior knowledge. Such as, for the test score and operating profit, the higher the better; for risk assessment, the lower the better with all other things being equal. In an OIS, the domain of criterion $a \in AT$ is completely pre-ordered by the relation $\succeq_a$: $x \succeq_a y \Leftrightarrow f(x, a) \geq f(y, a)$ (i.e., an increasing preference) or $x \succeq_a y \Leftrightarrow f(x, a) \leq f(y, a)$ (i.e., a decreasing preference), where $x, y \in U$. For simplicity and without any loss of generality, the following we only consider criteria with increasing preferences.

Definition 2.2 (*[32], Dominance Relation*)**.** Let $S^{\succeq} = (U, AT, V, f)$ be an OIS, $\forall P \subseteq AT$, $P \neq \emptyset$, the dominance relation D_P is defined as

$$D_P = \{(x, y) \in U \times U : f(x, a) \geq f(y, a), \forall a \in P\}. \quad (1)$$

Table 1
A part of academic transcript.

	x_1	x_2	x_3
a_1	98	87	78
a_2	95	88	65
d	A	B	C

Property 2.1 (*[32]*)**.** *D_P is a dominance relation, the following properties hold.*
(1) Reflexive: $\forall x \in U$, then xD_Px holds;
(2) Non-symmetric: $\forall x, y \in U$, if xD_Py, then yD_Px does not hold;
(3) Transitive: $\forall x, y, z \in U$, if xD_Py and yD_Pz, then xD_Pz holds.

Definition 2.3 (*[32], Knowledge Granules*)**.** Let $S^{\succeq} = (U, AT, V, f)$ be an OIS, $\forall P \subseteq AT, P \neq \emptyset$, two knowledge granules of x are called P-dominating set and P-dominated set, which are respectively defined as follows

$$D_P^+(x) = \{y \in U : yD_Px\}; \quad (2)$$

$$D_P^-(x) = \{y \in U : xD_Py\}. \quad (3)$$

Example 1. Table 1 is part of the academic transcript, where a_1 and a_2 represent professional course 1 and 2, respectively, and x_1, x_2, and x_3 represent three students. Table 1 is a typical OIS, where $P = \{a_1, a_2\}$, $U = \{x_1, x_2, x_3\}$, D_P is a dominance relation.

Next, we verify Property 2.1 based on Definition 2.2 as follows. (1) $\forall x \in U$, then xD_Px holds; (2) $x_1D_Px_2$ holds, but $x_2D_Px_1$ does not hold; (3) $x_1D_Px_2$ and $x_2D_Px_3$ hold, then $x_1D_Px_3$ also holds. According to Definition 2.3, P-dominating set and P-dominated set are respectively calculated as $D_P^+(x_1) = \{x_1\}$, $D_P^+(x_2) = \{x_1, x_2\}$, and $D_P^+(x_3) = \{x_1, x_2, x_3\}$; $D_P^-(x_1) = \{x_1, x_2, x_3\}$, $D_P^-(x_2) = \{x_2, x_3\}$, and $D_P^-(x_3) = \{x_3\}$.

Property 2.2 (*[32]*)**.** *For any $P_1, P_2 \subseteq AT$ and $\forall x \in U$, the following properties hold.*
(1) If $P_1 \subseteq P_2$, then $D_{P_2}^+(x) \subseteq D_{P_1}^+(x)$ and $D_{P_2}^-(x) \subseteq D_{P_1}^-(x)$;
(2) $D_{P_1}^+(x) \cap D_{P_2}^+(x) = D_{P_1 \cup P_2}^+(x)$ and $D_{P_1}^-(x) \cap D_{P_2}^-(x) = D_{P_1 \cup P_2}^-(x)$.

An OIS with decision is called an ordered decision system (ODS), which is denoted as $S^{\succeq} = (U, C \cup \{d\}, V, f)$, where AT is divided into the condition attribute set C and the decision attribute d. Note that $C \cup \{d\} = AT$ and $d \notin C$. According to decision attribute d, U can be divided into a family of equivalent classes, denoted by $Cl = \{Cl_n, n \in T\}$, where $T = \{1, 2, \ldots, |V_d|\}$. The decision classes are also preference-ordered, that is, $\forall r, s \in T$, if $r > s$, then $\forall x \in Cl_r$ is preferred to $\forall y \in Cl_s$. In DRSA, the sets to be approximated are upward and downward unions, which are respectively denoted as $Cl_n^{\succeq} = \bigcup_{n' \geq n} Cl_{n'}$ and $Cl_n^{\preceq} = \bigcup_{n' \leq n} Cl_{n'} (\forall n, n' \in T)$. The statement $x \in Cl_n^{\succeq}$ means that x belongs to at least class Cl_n, whereas $x \in Cl_n^{\preceq}$ means that x belongs to at most class Cl_n.

Definition 2.4 (*[32], Approximations*)**.** Let $S^{\succeq} = (U, C \cup \{d\}, V, f)$ be an ODS, for any $P \subseteq C$, the lower and upper approximations of $Cl_n^{\succeq}$ are respectively defined as follows

$$\underline{P}(Cl_n^{\succeq}) = \{x \in U : D_P^+(x) \subseteq Cl_n^{\succeq}\}; \quad (4)$$

$$\overline{P}(Cl_n^{\succeq}) = \{x \in U : D_P^-(x) \cap Cl_n^{\succeq} \neq \emptyset\}. \quad (5)$$

The lower and upper approximations of $Cl_n^{\preceq}$ are respectively defined as follows

$$\underline{P}(Cl_n^{\preceq}) = \{x \in U : D_P^-(x) \subseteq Cl_n^{\preceq}\}; \quad (6)$$

$$\overline{P}(Cl_n^{\preceq}) = \{x \in U : D_P^+(x) \cap Cl_n^{\preceq} \neq \emptyset\}. \quad (7)$$

Example 2. Continuing from Example 1. Table 1 is also an ODS, where the values of the decision d is ranked as $C \prec B \prec A$. The upward and downward unions are $Cl_1^{\succeq} = \{x_1, x_2, x_3\}$, $Cl_2^{\succeq} = \{x_1, x_2\}$, and $Cl_3^{\succeq} = \{x_1\}$, $Cl_1^{\preceq} = \{x_3\}$, $Cl_2^{\preceq} = \{x_2, x_3\}$, and $Cl_3^{\preceq} = \{x_1, x_2, x_3\}$. According to Definition 2.4, the approximations of the upward unions are calculated as $\underline{P}(Cl_1^{\succeq}) = \{x_1, x_2, x_3\}$, $\underline{P}(Cl_2^{\succeq}) = \{x_1, x_2\}$, $\underline{P}(Cl_3^{\succeq}) = \{x_1\}$, $\overline{P}(Cl_1^{\succeq}) = \{x_1, x_2, x_3\}$, $\overline{P}(Cl_2^{\succeq}) = \{x_1, x_2\}$, and $\overline{P}(Cl_3^{\succeq}) = \{x_1\}$. The approximations of the downward unions are calculated as $\underline{P}(Cl_1^{\preceq}) = \{x_3\}$, $\underline{P}(Cl_2^{\preceq}) = \{x_2, x_3\}$, $\underline{P}(Cl_3^{\preceq}) = \{x_1, x_2, x_3\}$, $\overline{P}(Cl_1^{\preceq}) = \{x_3\}$, $\overline{P}(Cl_2^{\preceq}) = \{x_2, x_3\}$, and $\overline{P}(Cl_3^{\preceq}) = \{x_1, x_2, x_3\}$.

2.2. Attribute reduction based on dominance conditional entropy

In this subsection, we introduce some basic concepts of dominance information entropy (DIE), dominance conditional entropy (DCE), and attribute reduction approach based on DCE in an ODS.

Definition 2.5 (*[65], DIE*)**.** Let $S^{\succeq} = (U, C \cup \{d\}, V, f)$ be an ODS, for any $A \subseteq C$, the dominance information entropy of U with respect to A is defined as

$$DH_A^{\succeq}(U) = -\frac{1}{|U|}\sum_{i=1}^{n} \log \frac{|D_A^+(x_i)|}{|U|}, \tag{8}$$

where $|*|$ represents the cardinality of set $*$. In addition, for any $A, B \subseteq C$, the dominance information entropy of U with respect to A and B is defined as

$$DH_{A\cup B}^{\succeq}(U) = -\frac{1}{|U|}\sum_{i=1}^{n} \log \frac{|D_A^+(x_i) \cap D_B^+(x_i)|}{|U|} = -\frac{1}{|U|}\sum_{i=1}^{n} \log \frac{|D_{A\cup B}^+(x_i)|}{|U|}. \tag{9}$$

Definition 2.6 (*[65], DCE*)**.** Let $S^{\succeq} = (U, C \cup \{d\}, V, f)$ be an ODS, for any $A \subseteq C$, the dominance conditional entropy of A to d is defined as

$$DH_{d|A}^{\succeq}(U) = -\frac{1}{|U|}\sum_{i=1}^{n} \log \frac{|D_d^+(x_i) \cap D_A^+(x_i)|}{|D_A^+(x_i)|} = -\frac{1}{|U|}\sum_{i=1}^{n} \log \frac{|D_{\{d\}\cup A}^+(x_i)|}{|D_A^+(x_i)|}. \tag{10}$$

From Definition 2.6, we find that DCE reflects the degree of ranking consistency of objects, which is determined by the information provided by the conditional attribute set and the decision attribute. Since the formula $\frac{|D_d^+(x_i)\cap D_A^+(x_i)|}{|D_A^+(x_i)|}$ essentially determines the degree of ranking consistency. According to Eq. (10), we easily find that the value of $DH_{d|A}^{\succeq}(U)$ is inversely proportional to the degree of ranking consistency, where $DH_{d|A}^{\succeq}(U) \geq 0$. Namely, the smaller the value of $DH_{d|A}^{\succeq}(U)$, the higher the degree of ranking consistency, which also indicates that conditional attribute set A provides more accurate ranking information for the object set, and vice versa.

In the attribute reduction process, the informative attributes can be obtained through the attribute significance measures, which are defined as follows.

Definition 2.7 (*[6], DCE-Based Inner Significance Measure*)**.** Let $S^{\succeq} = (U, C \cup \{d\}, V, f)$ be an ODS, $\forall A \subseteq C$ and $\forall a \in A$, the DCE-based inner significance measure of a in A is defined as

$$sig_{inner}^{\succeq U}(a, A, d) = DH_{d|A-\{a\}}^{\succeq}(U) - DH_{d|A}^{\succeq}(U). \tag{11}$$

According to the explanation of DCE, the higher is the inner significance measure, the more important is conditional attribute. It can select necessary condition attributes in entire condition attribute set. In addition, the core attribute set of the attribute set A is represented as $Core_A = \{a \in A | sig_{inner}^{\succeq U}(a, A, d) > 0\}$.

Definition 2.8 (*[6], DCE-Based Outer Significance Measure*)**.** Let $S^{\succeq} = (U, C \cup \{d\}, V, f)$ be an ODS, $\forall B \subseteq C$ and $\forall a \in (C - B)$, the DCE-based outer significance measure of a to B is defined as

$$sig_{outer}^{\succeq U}(a, B, d) = DH_{d|B}^{\succeq}(U) - DH_{d|B\cup\{a\}}^{\succeq}(U). \tag{12}$$

Similar to the inner significance measure, the outer significance measure can select the necessary condition attributes other than the selected condition attribute set.

Given an ODS $S^{\succeq} = (U, C \cup \{d\}, V, f)$ and $\forall a \in C$, according to the heuristic attribute reduction strategy, we can acquire that if $sig_{inner}^{\succeq U}(a, C, d) > 0$, then $a \in Core_C$, i.e., a is an indispensable attribute. Then, a reduct can be gained based on $Core_C$ by gradually adding selected attributes with the highest outer significance to $Core_C$. Next, we introduce the definition of attribute reduction based on DCE.

Definition 2.9 (*[6], Attribute Reduction*)**.** Let $S^{\succeq} = (U, C \cup \{d\}, V, f)$ be an ODS, $\forall B \subseteq C$, the attribute subset B is a reduct of $S^{\succeq}$ if it satisfies

(1) $DH_{d|B}^{\succeq}(U) = DH_{d|C}^{\succeq}(U)$ and

(2) $\forall a \in B$, $DH_{d|B-\{a\}}^{\succeq}(U) \neq DH_{d|B}^{\succeq}(U)$.

The condition (1) ensures the selected attribute subset has the same distinguish power as the whole attribute set; the condition (2) ensures that all attributes in the subset are indispensable by deleting redundant attributes in selected attribute subset. Therefore, the selected attribute subset is called a reduct if it satisfies both of these two conditions, otherwise, it is just a relative reduct.

3. Attribute reduction based on dominance conditional entropy in matrix form

In this section, first, we define the dominance relation matrix and dominance diagonal matrix of OIS. Then, a matrix-based method for calculating DCE (MDCE) is presented and proved. Final, we introduce a MDCE based heuristic attribute reduction algorithm.

3.1. Matrix-based calculation of DCE

Definition 3.1 (*Dominance Relation Matrix*)**.** Let $S^{\succeq} = (U, AT, V, f)$ be an OIS, for any $A \subseteq AT$, D_A is a dominance relation under A, the dominance relation matrix on U with respect to A is defined as $\mathbb{M}_U^{\succeq A} = [m_{(i,j)}^A]_{n\times n}$, where

$$m_{(i,j)}^A = \begin{cases} 1, & x_j D_A x_i; \\ 0, & otherwise. \end{cases} \tag{13}$$

Property 3.1. *$\mathbb{M}_U^{\succeq A} = [m_{(i,j)}^A]_{n\times n}$ is a dominance relation matrix, the following properties hold.*
(1) $m_{(i,i)}^A = 1$, where $i \in [1, n]$ and $i \in N^+$;
(2) $\sum_{j=1}^{n} m_{(i,j)}^A = |D_A^+(x_i)|$ and $\sum_{i=1}^{n} m_{(i,j)}^A = |D_A^-(x_j)|$, where $i, j \in [1, n]$ and $i, j \in N^+$.

Definition 3.2 (*"∩" Operation*)**.** Let $S^{\succeq} = (U, AT, V, f)$ be an OIS, for any $A, B \subseteq AT$, two dominance relation matrices on U with respect to A and B are denoted as $\mathbb{M}_U^{\succeq A} = [m_{(i,j)}^A]_{n\times n}$ and $\mathbb{M}_U^{\succeq B} = [m_{(i,j)}^B]_{n\times n}$. Then "∩" operation between $\mathbb{M}_U^{\succeq A}$ and $\mathbb{M}_U^{\succeq B}$ is defined as

$$\mathbb{M}_U^{\succeq A} \cap \mathbb{M}_U^{\succeq B} = [m_{(i,j)}^A \times m_{(i,j)}^B]_{n\times n}. \tag{14}$$

From Eq. (14), we can easily find that the intuitive meaning of "∩" operation is to obtain a new dominance relation matrix by multiplying the corresponding elements of any two dominance relation matrices $\mathbb{M}_U^{\succeq A}$ and $\mathbb{M}_U^{\succeq B}$. Its practical significance is to obtain the dominance relation matrix with respect to attribute sets A and B simultaneously. Subsequently, we introduce this property.

Proposition 3.1. *Let $S^{\succeq} = (U, AT, V, f)$ be an OIS, for any $A, B \subseteq AT$, then $\mathbb{M}_U^{\succeq A\cup B} = \mathbb{M}_U^{\succeq A} \cap \mathbb{M}_U^{\succeq B}$ holds.*

Proof. According to Definition 3.1, $\mathbb{M}_U^{\succeq A\cup B} = [m_{(i,j)}^{A\cup B}]_{n\times n}$. If $m_{(i,j)}^{A\cup B} = 1$, i.e., $x_j \in D_{A\cup B}^{+}(x_i)$. Then, we have $x_j \in D_A^{+}(x_i)$ and $x_j \in D_B^{+}(x_i)$, i.e., $m_{(i,j)}^{A} = 1$ and $m_{(i,j)}^{B} = 1$. So, we can get $m_{(i,j)}^{A\cup B} = m_{(i,j)}^{A} \times m_{(i,j)}^{B} = 1$, and vice versa. If $m_{(i,j)}^{A\cup B} = 0$, i.e., $x_j \notin D_{A\cup B}^{+}(x_i)$, that is, $x_j \notin D_A^{+}(x_i)$ or $x_j \notin D_B^{+}(x_i)$, i.e., $m_{(i,j)}^{A} = 0$ or $m_{(i,j)}^{B} = 0$. Thus, we can obtain $m_{(i,j)}^{A\cup B} = m_{(i,j)}^{A} \times m_{(i,j)}^{B} = 0$, and vice versa. In summary, we can get $m_{(i,j)}^{A\cup B} = m_{(i,j)}^{A} \times m_{(i,j)}^{B}$, i.e., $\mathbb{M}_U^{\succeq A\cup B} = \mathbb{M}_U^{\succeq A} \cap \mathbb{M}_U^{\succeq B}$ holds. □

Definition 3.3 (*Dominance Diagonal Matrix*). Let $S^{\succeq} = (U, AT, V, f)$ be an OIS, for any $A \subseteq AT$, the dominance diagonal matrix of the dominance relation matrix $\mathbb{M}_U^{\succeq A} = [m_{(i,j)}^{A}]_{n\times n}$ is defined as $\mathbb{D}_U^{\succeq A} = [d_{(i,j)}^{A}]_{n\times n}$, where

$$d_{(i,j)}^{A} = \begin{cases} \sum_{l=1}^{n} m_{(i,l)}^{A}, & 1 \le i, j \le n, i = j; \\ 0, & 1 \le i, j \le n, i \ne j. \end{cases} \tag{15}$$

In addition, the determinant of dominance diagonal matrix is expressed as $|\mathbb{D}_U^{\succeq A}| = \Pi_{i=j=1}^{n} d_{ij}^{A}$, the inverse matrix of the dominance diagonal matrix is represented as $(\mathbb{D}_U^{\succeq A})^{-1} = [\frac{1}{d_{(i,j)}^{A}}]_{n\times n}$, where

$$\frac{1}{d_{(i,j)}^{A}} = \begin{cases} \frac{1}{\sum_{l=1}^{n} m_{(i,l)}^{A}}, & 1 \le i, j \le n, i = j; \\ 0, & 1 \le i, j \le n, i \ne j. \end{cases} \tag{16}$$

Corollary 3.1 (*MDCE*). *Let $S^{\succeq} = (U, C\cup\{d\}, V, f)$ be an ODS, for any $A \subseteq C$, based on the dominance diagonal matrices $\mathbb{D}_U^{\succeq A}$ and $\mathbb{D}_U^{\succeq A\cup\{d\}}$, MDCE of A to d is denoted as*

$$MDH_{d|A}^{\succeq}(U) = -\frac{1}{|U|} \log |\mathbb{D}_U^{\succeq A\cup\{d\}} \cdot (\mathbb{D}_U^{\succeq A})^{-1}|. \tag{17}$$

Proof. Based on Definition 2.6, we can get $DH_{d|A}^{\succeq}(U) = -\frac{1}{|U|} \sum_{i=1}^{n} \log \frac{|D_{\{d\}\cup A}^{+}(x_i)|}{|D_A^{+}(x_i)|} = -\frac{1}{|U|} \log \frac{\Pi_{i=1}^{n} |D_{\{d\}\cup A}^{+}(x_i)|}{\Pi_{i=1}^{n} |D_A^{+}(x_i)|}$. According to Definitions 3.1 and 3.3, the dominance diagonal matrices $\mathbb{D}_U^{\succeq A} = [d_{(i,j)}^{A}]_{n\times n}$ and $\mathbb{D}_U^{\succeq A\cup\{d\}} = [d_{(i,j)}^{A\cup\{d\}}]_{n\times n}$, where $d_{(i,j)}^{A} = |D_A^{+}(x_i)|$ and $d_{(i,j)}^{A\cup\{d\}} = |D_{A\cup\{d\}}^{+}(x_i)|$. Because $|\mathbb{D}_U^{\succeq A\cup\{d\}} \cdot (\mathbb{D}_U^{\succeq A})^{-1}| = \Pi_{i=1}^{n} \frac{d_{(i,j)}^{A\cup\{d\}}}{d_{(i,j)}^{A}} = \frac{\Pi_{i=1}^{n} d_{(i,j)}^{A\cup\{d\}}}{\Pi_{i=1}^{n} d_{(i,j)}^{A}} = \frac{\Pi_{i=1}^{n} |D_{\{d\}\cup A}^{+}(x_i)|}{\Pi_{i=1}^{n} |D_A^{+}(x_i)|}$. Thus, we can get $DH_{d|A}^{\succeq}(U) = MDH_{d|A}^{\succeq}(U)$. In summary, the results of calculating the dominance conditional entropy based on matrix and non-matrix methods are consistent. □

From Eq. (17), we find that the core part of MDCE is $|\mathbb{D}_U^{\succeq A\cup\{d\}} \cdot (\mathbb{D}_U^{\succeq A})^{-1}|$, which intuitively reflects the proportion of the diagonal matrices $\mathbb{D}_U^{\succeq A\cup\{d\}}$ to $\mathbb{D}_U^{\succeq A}$. Its practical significance is consistent with Eq. (10). Subsequently, an example is used to explain how to calculate MDCE.

Example 3. Table 2 is a car evaluation table, which is an example of ODS. These cars are evaluated by four criteria: maximum speed, acceleration, climbing ability, and safety performance. In Table 2, $U = \{x_1, x_2, x_3, x_4, x_5, x_6, x_7\}$ stands for seven cars, $C = \{a_1, a_2, a_3, a_4\}$, where a_1 stands for maximum speed, a_2 stands for acceleration, a_3 stands for climbing ability, and a_4 stands for safety performance. The values of the different criteria are ranked as V_{a_1} : *low* $\prec$ *mid* $\prec$ *high* $\prec$ *v − high*, V_{a_2} : *low* $\prec$ *mid* $\prec$ *high*, V_{a_3} : *poor* $\prec$ *fair* $\prec$ *good* $\prec$ *v−good*, V_{a_4} : *fair* $\prec$ *good* $\prec$ *excellent*, and $V_d : D \prec C \prec B \prec A$.

Table 2
An example of ordered decision system.

U	a_1	a_2	a_3	a_4	d
x_1	Mid	High	Fair	Excellent	D
x_2	High	Low	Fair	Good	B
x_3	Low	Mid	Good	Excellent	B
x_4	Mid	High	poor	Excellent	C
x_5	High	Low	Fair	Good	A
x_6	Low	Mid	Good	Excellent	B
x_7	High	Low	Fair	Good	B

According to Definition 3.1, the dominance relation matrices $\mathbb{M}_U^{\succeq C}$ and $\mathbb{M}_U^{\succeq d}$ are calculated respectively as

$$\mathbb{M}_U^{\succeq C} = \begin{bmatrix} 1&0&0&0&0&0&0\\ 0&1&0&0&1&0&1\\ 0&0&1&0&0&1&0\\ 1&0&0&1&0&0&0\\ 0&1&0&0&1&0&1\\ 0&0&1&0&0&1&0\\ 0&1&0&0&1&0&1 \end{bmatrix}_{7\times 7},$$

$$\mathbb{M}_U^{\succeq d} = \begin{bmatrix} 1&1&1&1&1&1&1\\ 0&1&1&0&1&1&1\\ 0&1&1&0&1&1&1\\ 0&1&1&1&1&1&1\\ 0&0&0&0&1&0&0\\ 0&1&1&0&1&1&1\\ 0&1&1&0&1&1&1 \end{bmatrix}_{7\times 7}.$$

Taking $\mathbb{M}_U^{\succeq C}$ as an example, Property 3.1 is verified as follows
(1) For any $i \in [1, 7]$ and $i \in N^{+}$, $m_{(i,i)}^{C} = 1$;
(2) For any $i, j \in [1, 7]$ and $i, j \in N^{+}$, $\sum_{j=1}^{7} m_{(i,j)}^{C} = |D_C^{+}(x_i)|$ and $\sum_{i=1}^{7} m_{(i,j)}^{C} = |D_C^{-}(x_j)|$, e.g., when $i = 1$, $D_C^{+}(x_1) = \{x_1\}$, we have $\sum_{j=1}^{7} m_{(1,j)}^{C} = |D_C^{+}(x_1)| = 1$, when $j = 1$, $D_C^{-}(x_1) = \{x_1, x_4\}$, we have $\sum_{i=1}^{7} m_{(i,1)}^{C} = |D_C^{-}(x_1)| = 2$.

Subsequently, according to Definition 3.2, the dominance relation matrix $\mathbb{M}_U^{\succeq C\cup\{d\}}$ is calculated as

$$\mathbb{M}_U^{\succeq C\cup\{d\}} = \mathbb{M}_U^{\succeq C} \cap \mathbb{M}_U^{\succeq d}$$

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

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

Next, according to Definition 3.3, the diagonal matrices $\mathbb{D}_U^{\succeq C}$, $\mathbb{D}_U^{\succeq C\cup\{d\}}$, and the inverse matrix $(\mathbb{D}_U^{\succeq C})^{-1}$ are calculated as

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

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

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

Finally, according to Corollary 3.1, MDCE of C to d can be calculated via using matrices $\mathbb{D}_U^{\succeq C\cup\{d\}}$ and $(\mathbb{D}_U^{\succeq C})^{-1}$ as $MDH_{d|C}^{\succeq}(U) = -\frac{1}{7}\log|\mathbb{D}_U^{\succeq C\cup\{d\}}\cdot(\mathbb{D}_U^{\succeq C})^{-1}| = 0.3693$.

Corollary 3.2 (*MDCE-Based Inner Significance Measure*)**.** *Let $S^{\succeq} = (U, C\cup\{d\}, V, f)$ be an ODS, for any $B\subseteq C$ and $\forall a\in B$, MDCE-based inner significance measure of a in B is denoted as*

$$Msig_{inner}^{\succeq U}(a, B, d) = MDH_{d|(B-\{a\})}^{\succeq}(U) - MDH_{d|B}^{\succeq}(U). \tag{18}$$

The meaning of the inner significance measure based on DCE and MDCE are consistent, and the results calculated by Eqs. (11) and (18) are also the same.

Corollary 3.3 (*MDCE-Based Outer Significance Measure*)**.** *Let $S^{\succeq} = (U, C\cup\{d\}, V, f)$ be an ODS, for any $B\subseteq C$ and $\forall a\in(C-B)$, MDCE-based outer significance measure of a to B is denoted as*

$$Msig_{outer}^{\succeq U}(a, B, d) = MDH_{d|B}^{\succeq}(U) - MDH_{d|B\cup\{a\}}^{\succeq}(U). \tag{19}$$

The DCE-based and MDCE-based outer significance measures have the same meaning, and the results calculated by Eqs. (12) and (19) are also consistent. Note that in the incremental attribute reduction algorithm, we will use Corollary 3.3 to construct a sequence of all candidate attributes to accelerate the reduct selection.

3.2. Heuristic attribute reduction algorithm based on MDCE

This subsection introduces a heuristic attribute reduction (HAR) algorithm based on MDCE in ODS. This algorithm calculates a reduct from scratch when objects vary, which retrains the dynamic ODS as a new one. Thus, this algorithm is a non-incremental attribute reduction algorithm, which is compared with incremental algorithms. The detailed steps are introduced in Algorithm 1.

Algorithm 1 HAR algorithm

Input: An ODS $S^{\succeq} = (U, C\cup\{d\}, V, f)$.
Output: A reduct Red_U.
1: Initialize $Red_U \leftarrow \emptyset$;
2: Calculate MDCE $MDH_{d|C}^{\succeq}(U)$ in U via using Corollary 3.1;
3: **for** $k = 1$ to $|C|$ **do**
4: Calculate $Msig_{inner}^{\succeq U}(a_k, C, d)$ via using Corollary 3.2;
5: **if** $Msig_{inner}^{\succeq U}(a_k, C, d) > 0$, **then**
6: $Red_U \leftarrow Red_U \cup \{a_k\}$;
7: **end if**
8: **end for**
9: Let $B \leftarrow Red_U$;
10: **while** $MDH_{d|B}^{\succeq}(U) \neq MDH_{d|C}^{\succeq}(U)$ **do**
11: **for** $l = 1$ to $|C - B|$ **do**
12: Calculate $Msig_{outer}^{\succeq U}(a_l, B, d)$ via using Corollary 3.3;
13: **end for**
14: Select $a_0 = max\{Msig_{outer}^{\succeq U}(a_l, B, d), a_l \in (C - B)\}$;
15: $B \leftarrow B \cup \{a_0\}$
16: **end while**
17: **for** each $a \in B$ **do**
18: **if** $MDH_{d|(B-\{a\})}^{\succeq}(U) = MDH_{d|B}^{\succeq}(U)$, **then**
19: $B \leftarrow B - \{a\}$;
20: **end if**
21: **end for**
22: $Red_U \leftarrow B$;
23: **return** Red_U;

The detailed explanation of the steps in Algorithm 1 and their time complexity are given as follows. Step 2 calculates MDCE from scratch and its time complexity is $O(|C||U|^2)$. Steps 3–8 obtain the indispensable attribute a_k, i.e., a_k is a *core* attribute of the ODS, and its time complexity is $O(|C|^2|U|^2)$. Steps 10–16 find the best candidate attribute from remaining attributes $C - B$ to attribute subset B until Step 10 no longer holds, i.e., the relative reduct B is obtained, and its time complexity is $O(|C|^2|U|^2)$. Steps 17–21 delete redundant attributes from relative reduct B and its time complexity is $O(|B|^2|U|^2)$. In summary, the time complexity of Algorithm 1 is $O(|C||U|^2 + |C|^2|U|^2 + |C|^2|U|^2 + |B|^2|U|^2)$. Furthermore, the space complexity of Algorithm 1 is $O(|U|^2 + |C||U|^2)$.

Example 4. The process for calculating the reduct of ODS in Table 2 by using Algorithm 1 is shown in Table 3.

4. The incremental mechanism for attribute reduction with the variation of multiple objects

In an ODS, the variation of multiple objects can be divided into two types: adding multiple objects and deleting multiple objects. In this section, two incremental attribute reduction algorithms based on MDCE are proposed. Since the calculation of MDCE plays a key role in attribute reduction algorithms, which directly affects the efficiency of the algorithms. When objects vary, recomputing MDCE is time-consuming, especially in large data. To overcome this deficiency, we propose two incremental updating methods based on dominance relation matrix for calculating MDCE when objects vary in an ODS. On this basis, incremental algorithms for attribute reduction are proposed in this section.

4.1. An incremental method for obtaining attribute reduction when adding multiple objects

In this section, the updating principle of MDCE is firstly presented when adding multiple objects. Then, algorithm for updating attribute reduction in an ODS is designed. In each subsection, examples are given to illustrate the given method.

4.1.1. The updating principle of MDCE $w.r.t.$ adding objects

This subsection introduces the dominance relation matrix based incremental updating method to compute new MDCE when multiple objects are added to an ODS. The key step of the method is to update the dominance relation matrix and dominance diagonal matrix. Next, we introduce the update principles of the correlation matrices.

Proposition 4.1 (*Update Dominance Relation Matrix*)**.** *Given an OIS $S^{\succeq} = (U, AT, V, f)$, where $U = \{x_1, x_2, \ldots, x_n\}$. $\forall A\subseteq AT$, suppose that the dominance relation matrix on U with respect to A is $\mathbb{M}_U^{\succeq A} = [m_{(i,j)}^{A}]_{n\times n}$, the object set $U^+ = \{x_{n+1}, x_{n+2}, \ldots, x_{n+n'}\}$ is added to $S^{\succeq}$. The updated dominance relation matrix on $U\cup U^+$ with respect*

Table 3
The illustration of the calculation process of Algorithm 1.

Steps	Operations
1	Initialize $Red_U = \emptyset$.
2	Calculate MDCE of the entire attribute set C to d as $MDH^{\succeq}_{d\|C}(U) = 0.3693$, then turn to Steps 3–9.
3–9	For any $a \in C$, calculate inner significance as $Msig^{\succeq U}_{inner}(a_1, C, d) = 0.2180$, $Msig^{\succeq U}_{inner}(a_2, C, d) = 0$, $Msig^{\succeq U}_{inner}(a_3, C, d) = 0.2857$, $Msig^{\succeq U}_{inner}(a_4, C, d) = 0$. If $Msig^{\succeq U}_{inner}(a, C, d) > 0$, put a into Red_U, then $Red_U = \{a_1, a_3\}$. Let $B \leftarrow Red_U$, then turn to Step 10.
10	Calculate MDCE of B to d as $MDH^{\succeq}_{d\|B}(U) = 0.2724$. Because the condition $MDH^{\succeq}_{d\|C}(U) \neq MDH^{\succeq}_{d\|B}(U)$ is satisfied, then turn to Steps 11–15.
11–15	For any $a \in (C - B)$, calculate outer significance as $Msig^{\succeq U}_{outer}(a_2, B, d) = 0.0969$, $Msig^{\succeq U}_{outer}(a_4, B, d) = 0.0969$, and then put the a with the maximal outer significance into B, i.e., $B = \{a_1, a_3\} \cup \{a_2\}$ (If they are equal, then choose the one with the smaller subscript). After obtaining the new attribute subset B, then go to verify the loop condition, i.e., turn to Step 10.
10	Calculate MDCE of B to d as $MDH^{\succeq}_{d\|B}(U) = 0.3693$. Because the condition $MDH^{\succeq}_{d\|C}(U) \neq MDH^{\succeq}_{d\|B}(U)$ is not satisfied, then loop is break and calculation step turn to Steps 17–21.
17–21	For any $a \in B$, calculate $MDH^{\succeq}_{d\|(B-\{a\})}(U)$ as $MDH^{\succeq}_{d\|(B-\{a_1\})}(U) = 0.5873$, $MDH^{\succeq}_{d\|(B-\{a_2\})}(U) = 0.2724$, $MDH^{\succeq}_{d\|(B-\{a_3\})}(U) = 0.6550$. Because $\forall a \in B$, $MDH^{\succeq}_{d\|(B-\{a\})}(U) \neq MDH^{\succeq}_{d\|B}(U)$ holds, there is no redundant attribute in B, i.e., $B = \{a_1, a_2, a_3\}$, then turn to Steps 22–23.
22–23	Output the final reduct as $Red_U = \{a_1, a_2, a_3\}$.

to A is denoted as $\mathbb{M}^{\succeq A}_{U \cup U^+} = [m'^{A}_{(i,j)}]_{(n+n')\times(n+n')}$, *where*

$$m'^{A}_{(i,j)} = \begin{cases} m^{A}_{(i,j)}, & (1 \le i \le n) \wedge (1 \le j \le n); \\ 1, & x_j D_A x_i, (n+1 \le i \le n+n') \vee (n+1 \le j \le n+n'); \\ 0, & otherwise. \end{cases} \tag{20}$$

Proof. Assuming that U^+ is added to U, then $U \cup U^+ = \{x_1, x_2, \ldots, x_n, x_{n+1}, x_{n+2}, \ldots, x_{n+n'}\}$. Based on Definition 3.1, the dominance relation matrix $\mathbb{M}^{\succeq A}_{U \cup U^+}$ can be divided into four parts, i.e., $\mathbb{M}^{\succeq A}_{U \cup U^+} = \left[\begin{array}{c|c} [\mathbb{M}^{1\succeq A}_{U}]_{n\times n} & [\mathbb{M}^{2\succeq A}_{U,U^+}]_{n\times n'} \\ \hline [\mathbb{M}^{3\succeq A}_{U^+,U}]_{n'\times n} & [\mathbb{M}^{4\succeq A}_{U^+}]_{n'\times n'} \end{array}\right]$, where $\mathbb{M}^{1\succeq A}_{U}$ represents the dominance relation matrix of $U \times U$ under A, $\mathbb{M}^{2\succeq A}_{U,U^+}$ represents the dominance relation matrix of $U \times U^+$ under A, $\mathbb{M}^{3\succeq A}_{U^+,U}$ represents the dominance relation matrix of $U^+ \times U$ under A, $\mathbb{M}^{4\succeq A}_{U^+}$ represents the dominance relation matrix of $U^+ \times U^+$ under A. For $\mathbb{M}^{1\succeq A}_{U} = [m^{1A}_{(i,j)}]_{n\times n}$, if $x_j D_A x_i, (1 \le i \le n) \wedge (1 \le j \le n)$ holds, then $m^{1A}_{(i,j)} = 1$; otherwise, $m^{1A}_{(i,j)} = 0$. For $\mathbb{M}^{2\succeq A}_{U,U^+} = [m^{2A}_{(i,j)}]_{n\times n'}$, if $x_j D_A x_i, (1 \le i \le n) \wedge (n+1 \le j \le n+n')$ holds, then $m^{2A}_{(i,j)} = 1$; otherwise, $m^{2A}_{(i,j)} = 0$. For $\mathbb{M}^{3\succeq A}_{U^+,U} = [m^{3A}_{(i,j)}]_{n'\times n}$, if $x_j D_A x_i, (n+1 \le i \le n+n') \wedge (1 \le j \le n)$ holds, then $m^{3A}_{(i,j)} = 1$; otherwise, $m^{3A}_{(i,j)} = 0$. For $\mathbb{M}^{4\succeq A}_{U^+} = [m^{4A}_{(i,j)}]_{n'\times n'}$, if $x_j D_A x_i, (n+1 \le i \le n+n') \wedge (n+1 \le j \le n+n')$ holds, then $m^{4A}_{(i,j)} = 1$; otherwise, $m^{4A}_{(i,j)} = 0$. Obviously, $m^{1A}_{(i,j)} = m^{A}_{(i,j)}$ always holds for $(1 \le i \le n) \wedge (1 \le j \le n)$. From the above description, we find that $\mathbb{M}^{2\succeq A}_{U,U^+}$, $\mathbb{M}^{3\succeq A}_{U^+,U}$, and $\mathbb{M}^{4\succeq A}_{U^+}$ can be represented by a characteristic function, i.e., $m'^{A}_{(i,j)} = \begin{cases} 1 & x_j D_A x_i, (n+1 \le i \le n+n') \vee (n+1 \le j \le n+n') \\ 0 & otherwise \end{cases}$. In summary, we can get the characteristic function of $\mathbb{M}^{\succeq A}_{U \cup U^+}$, i.e., Eq. (20). □

Proposition 4.1 provides the principle of updating the dominance relation matrix when adding multiple objects. Its basic idea is to add three new dominance relation matrices to the original dominance relation matrix to achieve the purpose of updating the dominance relation matrix. Next, we use an example to illustrate it.

Example 5. Continuing from Example 3, $U^+ = \{x_8, x_9, x_{10}\}$ is added to Table 2, where $x_8 = \{mid, high, poor, excellent, C\}$, $x_9 = \{v-high, high, v-good, excellent, C\}$, and $x_{10} = \{low, mid, good, excellent, A\}$. $\mathbb{M}^{\succeq C}_{U \cup U^+}$ can be updated by using Proposition 4.1 as

$$\mathbb{M}^{\succeq C}_{U \cup U^+} = \left[\begin{array}{c|c} [\mathbb{M}^{1\succeq C}_{U}]_{7\times 7} & [\mathbb{M}^{2\succeq C}_{U,U^+}]_{7\times 3} \\ \hline [\mathbb{M}^{3\succeq C}_{U^+,U}]_{3\times 7} & [\mathbb{M}^{4\succeq C}_{U^+}]_{3\times 3} \end{array}\right] = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 1 & 1 \\ 1 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 1 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 0 & 1 & 0 & 1 & 0 & 1 & 0 \\ 1 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 1 & 1 \end{bmatrix}_{10\times 10},$$

where $\mathbb{M}^{1\succeq C}_{U} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 & 1 \end{bmatrix}_{7\times 7}$,

$$\mathbb{M}^{2\succeq C}_{U,U^+} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 1 & 0 \\ 0 & 1 & 1 \\ 1 & 1 & 0 \\ 0 & 1 & 0 \\ 0 & 1 & 1 \\ 0 & 1 & 0 \end{bmatrix}_{7\times 3},$$

$$\mathbb{M}^{3\succeq C}_{U^+,U} = \begin{bmatrix} 1 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 & 0 \end{bmatrix}_{3\times 7},$$

$$\mathbb{M}^{4\succeq C}_{U^+} = \begin{bmatrix} 1 & 1 & 0 \\ 0 & 1 & 0 \\ 0 & 1 & 1 \end{bmatrix}_{3\times 3}.$$

Proposition 4.2 (*Update Dominance Diagonal Matrix*)**.** *Given an OIS* $S^{\succeq} = (U, AT, V, f)$*, where* $U = \{x_1, x_2, \ldots, x_n\}$*. For any* $A \subseteq AT$*, let the dominance diagonal matrix on U with respect to A is* $\mathbb{D}^{\succeq A}_{U} = [d^{A}_{(i,j)}]_{n\times n}$*, the object set* $U^+ = \{x_{n+1}, x_{n+2}, \ldots, x_{n+n'}\}$ *is added to*

$S^{\succeq}$. The updated dominance diagonal matrix on $U \cup U^{+}$ with respect to A is denoted as $\mathbb{D}^{\succeq A}_{U\cup U^{+}} = [d'^{A}_{(i,j)}]_{(n+n')\times(n+n')}$, where

$$d'^{A}_{(i,j)} = \begin{cases} d^{A}_{(i,j)} + \sum\limits_{l=n+1}^{n+n'} m'^{A}_{(i,l)}, & 1 \le i,j \le n,\ i=j; \\ \sum\limits_{l=1}^{n+n'} m'^{A}_{(i,l)}, & n+1 \le i,j \le n+n',\ i=j; \\ 0, & 1 \le i,j \le n+n',\ i \ne j. \end{cases} \quad (21)$$

Proof. Assuming that U^{+} is added to U, then $U \cup U^{+} = \{x_1, x_2, \ldots, x_n, x_{n+1}, x_{n+2}, \ldots, x_{n+n'}\}$. According to Definition 3.3, we can get $\mathbb{D}^{\succeq A}_{U\cup U^{+}} = [d'^{A}_{(i,j)}]_{(n+n')\times(n+n')}$. Actually, for any $1 \le i,j \le n+n', i \ne j$, $d'^{A}_{(i,j)} = 0$ always holds. So we can get that $d'^{A}_{(i,j)}$ remain unchanged for any $1 \le i,j \le n, i \ne j$, i.e., $d'^{A}_{(i,j)} = d^{A}_{(i,j)}$. Based on Definition 3.3, for any $1 \le i,j \le n, i = j$, we can obtain $d'^{A}_{(i,j)} = \sum_{l=1}^{n+n'} m'^{A}_{(i,l)} = \sum_{l=1}^{n} m'^{A}_{(i,l)} + \sum_{l=n+1}^{n+n'} m'^{A}_{(i,l)}$. For any $1 \le i,l \le n$, $m'^{A}_{(i,l)} = m^{A}_{(i,l)}$ always hold. Hence, we can get $d'^{A}_{(i,j)} = \sum_{l=1}^{n} m^{A}_{(i,l)} + \sum_{l=n+1}^{n+n'} m'^{A}_{(i,l)} = d^{A}_{(i,j)} + \sum_{l=n+1}^{n+n'} m'^{A}_{(i,l)}$. Besides, for any $n+1 \le i,j \le n+n'$, $i=j$, according to Definition 3.3, the $d'^{A}_{(i,j)}$ should be calculated, i.e., $d'^{A}_{(i,j)} = \sum_{l=1}^{n+n'} m'^{A}_{(i,l)}$. In summary, we can get the characteristic function of $\mathbb{D}^{\succeq A}_{U\cup U^{+}}$, i.e., Eq. (21). □

Example 6. Continuing from Example 5, known matrices $\mathbb{M}^{\succeq C}_{U\cup U^{+}}$ and $\mathbb{D}^{\succeq C}_{U}$, we can update matrix $\mathbb{D}^{\succeq C}_{U\cup U^{+}}$ by using Proposition 4.2 as

$$\mathbb{D}^{\succeq C}_{U\cup U^{+}} = \begin{bmatrix} 1+1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 3+1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 2+2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 2+2 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 3+1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 2+2 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 3+1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 4 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 4 \end{bmatrix}$$

$$= \begin{bmatrix} 2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 4 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 4 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 4 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 4 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 4 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 4 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 4 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 4 \end{bmatrix}_{10\times 10}.$$

Here we explain how to calculate a new MDCE by updating the dominance relation matrix and its dominance diagonal matrix when adding objects. For any $A \subseteq C$, the known original matrices are $\mathbb{M}^{\succeq A}_{U}$, $\mathbb{M}^{\succeq A\cup\{d\}}_{U}$, $\mathbb{D}^{\succeq A}_{U}$, and $\mathbb{D}^{\succeq A\cup\{d\}}_{U}$. When U^{+} is added to $S^{\succeq}$, according to Propositions 4.1 and 4.2, we can get the updated dominance diagonal matrices $\mathbb{D}^{\succeq A}_{U\cup U^{+}}$ and $\mathbb{D}^{\succeq A\cup\{d\}}_{U\cup U^{+}}$. Therefore, it is easy to calculate MDCE $MDH^{\succeq}_{d|A}(U \cup U^{+})$ by Corollary 3.1.

4.1.2. An incremental attribute reduction algorithm *w.r.t.* adding objects

In this subsection, an incremental attribute reduction algorithm while adding multiple objects (IAR-A) based on the updating principle of MDCE is given in Algorithm 2. Then, the time and space complexity of the proposed algorithm are analyzed. Lastly, we demonstrate the process of the proposed algorithm by an example.

Algorithm 2 IAR-A algorithm

Input:
(1) An original ODS $S^{\succeq} = (U, C \cup \{d\}, V, f)$, where $U = \{x_1, x_2, \ldots, x_n\}$, $U^{+} = \{x_{n+1}, x_{n+2}, \ldots, x_{n+n'}\}$ is an added object set;
(2) The original reduct Red_U on U;
(3) The original dominance relation matrices $\mathbb{M}^{\succeq C}_{U} = [m^{C}_{(i,j)}]_{n\times n}$, $\mathbb{M}^{\succeq Red_U}_{U} = [m^{Red_U}_{(i,j)}]_{n\times n}$, and $\mathbb{M}^{\succeq d}_{U} = [m^{d}_{(i,j)}]_{n\times n}$;
(4) The original dominance diagonal matrices $\mathbb{D}^{\succeq C}_{U} = [d^{C}_{(i,j)}]_{n\times n}$, $\mathbb{D}^{\succeq C\cup\{d\}}_{U} = [d^{C\cup\{d\}}_{(i,j)}]_{n\times n}$, $\mathbb{D}^{\succeq Red_U}_{U} = [d^{Red_U}_{(i,j)}]_{n\times n}$, and $\mathbb{D}^{\succeq Red_U\cup\{d\}}_{U} = [d^{Red_U\cup\{d\}}_{(i,j)}]_{n\times n}$.
Output: A new reduct $Red_{U'}$ on $U \cup U^{+}$

1: Initialize $B \leftarrow Red_U$, $U' \leftarrow U \cup U^{+}$, $\mathbb{M}^{\succeq C}_{U'} \leftarrow \mathbb{M}^{\succeq C}_{U}$, $\mathbb{M}^{\succeq B}_{U'} \leftarrow \mathbb{M}^{\succeq B}_{U}$, $\mathbb{M}^{\succeq d}_{U'} \leftarrow \mathbb{M}^{\succeq d}_{U}$, $\mathbb{D}^{\succeq C}_{U'} \leftarrow \mathbb{D}^{\succeq C}_{U}$, $\mathbb{D}^{\succeq C\cup\{d\}}_{U'} \leftarrow \mathbb{D}^{\succeq C\cup\{d\}}_{U}$, $\mathbb{D}^{\succeq B}_{U'} \leftarrow \mathbb{D}^{\succeq B}_{U}$, and $\mathbb{D}^{\succeq B\cup\{d\}}_{U'} \leftarrow \mathbb{D}^{\succeq B\cup\{d\}}_{U}$;
2: Compute new dominance relation matrices $\mathbb{M}^{\succeq C}_{U'} \leftarrow [m'^{C}_{(i,j)}]_{(n+n')\times(n+n')}$, $\mathbb{M}^{\succeq B}_{U'} \leftarrow [m'^{B}_{(i,j)}]_{(n+n')\times(n+n')}$, and $\mathbb{M}^{\succeq d}_{U'} \leftarrow [m'^{d}_{(i,j)}]_{(n+n')\times(n+n')}$ by using Proposition 4.1;
3: Compute dominance relation matrices $\mathbb{M}^{\succeq C\cup\{d\}}_{U'}$ and $\mathbb{M}^{\succeq B\cup\{d\}}_{U'}$ by using Proposition 3.1;
4: Compute new dominance diagonal matrices $\mathbb{D}^{\succeq C}_{U'} \leftarrow [d'^{C}_{(i,j)}]_{(n+n')\times(n+n')}$, $\mathbb{D}^{\succeq C\cup\{d\}}_{U'} \leftarrow [d'^{C\cup\{d\}}_{(i,j)}]_{(n+n')\times(n+n')}$, $\mathbb{D}^{\succeq B}_{U'} \leftarrow [d'^{B}_{(i,j)}]_{(n+n')\times(n+n')}$, and $\mathbb{D}^{\succeq B\cup\{d\}}_{U'} \leftarrow [d'^{B\cup\{d\}}_{(i,j)}]_{(n+n')\times(n+n')}$ by using Proposition 4.2;
5: Compute new MDCE $MDH^{\succeq}_{d|C}(U')$ and $MDH^{\succeq}_{d|B}(U')$ by using Corollary 3.1;
6: **if** $MDH^{\succeq}_{d|C}(U') = MDH^{\succeq}_{d|B}(U')$, **then**
7: go to step 17;
8: **else**
9: go to step 11;
10: **end if**
11: For each $a \in (C - B)$, compute $Msig^{\succeq U'}_{outer}(a, B, d)$ by using Corollary 3.3, then construct a descending sequence by $Msig^{\succeq U'}_{outer}(a, B, d)$, and record the results by $\{a'_1, a'_2, \ldots, a'_{|C-B|}\}$;
12: **while** $MDH^{\succeq}_{d|C}(U') \ne MDH^{\succeq}_{d|B}(U')$ **do**
13: **for** $h = 1$ to $|C - B|$ **do**
14: select $B \leftarrow B \cup \{a'_h\}$ and compute $MDH^{\succeq}_{d|B}(U')$;
15: **end for**
16: **end while**
17: **for** each $a \in B$ **do**
18: compute $MDH^{\succeq}_{d|(B-\{a\})}(U')$;
19: **if** $MDH^{\succeq}_{d|(B-\{a\})}(U') = MDH^{\succeq}_{d|B}(U')$, **then**
20: $B \leftarrow B - \{a\}$;
21: **end if**
22: **end for**
23: $Red_{U'} \leftarrow B$;
24: **return** $Red_{U'}$;

The detailed description of the steps in Algorithm 2 and their time complexity are given as follows. Steps 2–4 calculate new dominance relation matrices and its dominance diagonal matrices in incremental manners by using Propositions 4.1 and 4.2 and its time complexity is $O(|C\|U^{+}\|U'|)$. Step 5 computes new MDCE by using Corollary 3.1. Steps 6–10 determine whether the new MDCE under the original attribute subset (i.e. original reduct) is equal to that of under the entire attribute set; if yes, then keep the original attribute subset unchanged. Steps 11–16 arrange the remaining attributes in descending order, and incrementally update the selected attribute subset until Step 12 no longer holds, its time complexity is $O((|C| - |B|)|U'|^2)$. Steps 17–22 delete redundant attributes from the selected attribute subset and its time complexity is $O(|B|^2|U'|^2)$. Steps 23–24 output

Table 4
The comparisons of the time and space complexity of algorithms HAR and IAR-A.

Algorithm	HAR	IAR-A
Time complexity	$O(\|C\|\|U'\|^2+\|C\|^2\|U'\|^2+\|C\|^2\|U'\|^2+\|B\|^2\|U'\|^2)$	$O(\|C\|\|U^+\|\|U'\|+(\|C\|-\|B\|)\|U'\|^2+\|B\|^2\|U'\|^2)$
Space complexity	$O(\|U'\|^2+\|C\|\|U'\|^2)$	$O(\|U'\|^2+(\|C\|-\|B\|)\|U'\|^2)$

a final reduct. In summary, the time complexity of Algorithm 2 is $O(|C||U^+||U'|+(|C|-|B|)|U'|^2+|B|^2|U'|^2)$. Moreover, the space complexity of Algorithm 2 is $O(|U'|^2+(|C|-|B|)|U'|^2)$. The time and space complexity of algorithms HAR and IAR-A are summarized in Table 4.

Obviously, as shown in Table 4, the time complexity of algorithm IAR-A is usually much less than that of algorithm HAR. The main reason is that algorithm HAR calculates a new reduct from scratch when multiple objects are added to the original ODS. In contrast, algorithm IAR-A uses previous knowledge to quickly calculate MDCE by the updating mechanism, then calculates a new reduct based on the greedy search strategy. In real-life applications, the number of samples in ODS is much larger than the number of attributes, i.e., $|U| \gg |C|$. Therefore, algorithm IAR-A has a more significant time-saving effect on calculating reduct for large-scale data. Moreover, the space complexity of algorithm IAR-A is slightly smaller than that of algorithm HAR.

Subsequently, we present an example to demonstrate the detailed steps for calculating a new reduct by using Algorithm 2.

Example 7. Continuing from Example 4, the known knowledge of the original ODS includes the reduct $Red_U = \{a_1, a_2, a_3\}$; the dominance relation matrices $\mathbb{M}_U^{\succeq C}$, $\mathbb{M}_U^{\succeq Red_U}$, and $\mathbb{M}_U^{\succeq d}$; and dominance diagonal matrices $\mathbb{D}_U^{\succeq C}$, $\mathbb{D}_U^{\succeq C\cup\{d\}}$, $\mathbb{D}_U^{\succeq Red_U}$, and $\mathbb{D}_U^{\succeq Red_U\cup\{d\}}$. The object set $U^+ = \{x_8, x_9, x_{10}\}$ is added to U. The calculation process of Algorithm 2 is shown in Table 5.

4.2. An incremental method for obtaining attribute reduction when deleting multiple objects

In this section, the updating principle of MDCE is firstly presented when deleting multiple objects. Then, algorithm for updating attribute reduction in an ODS is proposed and the time and space complexity of it are analyzed.

4.2.1. The updating principle of MDCE w.r.t. deleting multiple objects

This subsection presents a dominance relation matrix based incremental updating method for computing new MDCE when multiple objects are deleted from an ODS. The process of calculating new MDCE is similar to that presented in Section 4.1, but the method for updating the relevant matrices is different. The main reason is that when deleting multiple objects from the original ODS, we do not need to calculate new dominance relation matrices. We only need to move the position of the matrix elements according to the position of the deleted objects, and then obtain new dominance relation matrices. The following paragraphs introduce the updating principles of the dominance relation matrix and its dominance diagonal matrix.

Proposition 4.3 (*Update Dominance Relation Matrix*)**.** *Given an OIS $S^{\succeq} = (U, AT, V, f)$, where $U = \{x_1, x_2, \dots, x_n\}$. $\forall A \subseteq AT$, suppose that the dominance relation matrix on U with respect to A is $\mathbb{M}_U^{\succeq A} = [m^A_{(i,j)}]_{n\times n}$, the object set $U^- = \{x_{q_1}, x_{q_2}, \dots, x_{q_{n'}}\}$ is deleted from $S^{\succeq}$. The updated dominance relation matrix on $U - U^-$ with respect to A is denoted as $\mathbb{M}_{U-U^-}^{\succeq A} = [m'^A_{(i,j)}]_{(n-n')\times(n-n')}$, where*

$$m'^A_{(i,j)} = \begin{cases} m^A_{(i+k_r-1,j+k_c-l)}, & (q_{k_r-1}-k_r+2 \le i < q_{k_r}-k_r+1) \\ & \wedge(q_{k_c-1}-k_c+2 \le j < q_{k_c}-k_c+1); \\ m^A_{(i+k_r-1,j+n')}, & (q_{k_r-1}-k_r+2 \le i < q_{k_r}-k_r+1) \\ & \wedge(q_{n'}-n'+1 \le j \le n-n'); \\ m^A_{(i+n',j+k_c-l)}, & (q_{n'}-n'+1 \le i \le n-n') \\ & \wedge(q_{k_c-1}-k_c+2 \le j < q_{k_c}-k_c+1); \\ m^A_{(i+n',j+n')}, & (q_{n'}-n'+1 \le i \le n-n') \\ & \wedge(q_{n'}-n'+1 \le j \le n-n'), \end{cases} \tag{22}$$

where $1 \le k_r, k_c \le n'$.

Proof. Assuming that $U^- = \{x_{q_1}, x_{q_2}, \dots, x_{q_{n'}}\}$ is deleted from $S^{\succeq}$ and $1 \le q_1 < q_2 < \cdots < q_{n'} \le n$. Firstly, we consider updating the row coordinate of element $m^A_{(i,j)}$. For any $q_{k_r-1} \le i < q_{k_r}$ $(1 \le k_r \le n')$, it is easy to know that k_r-1 rows are deleted before ith row from $\mathbb{M}_U^{\succeq A}$. Thus, the element $m^A_{(i,j)}$ for any $q_{k_r-1} \le i < q_{k_r}$ will be shifted forward by k_r-1 rows, i.e., $m^A_{(i,*)} = m'^A_{(i-k_r+1,*)}$. Then, we can obtain $m'^A_{(i,*)} = m^A_{(i+k_r-1,*)}$ for any $q_{k_r-1}-k_r+2 \le i < q_{k_r}-k_r+1$. In addition, for any $q_{n'}-n'+1 \le i \le n-n'$, the element $m^A_{(i,j)}$ will be shifted forward by n' rows, i.e., $m^A_{(i,*)} = m'^A_{(i-n',*)}$, which indicates $m'^A_{(i,*)} = m^A_{(i+n',*)}$. Next, we consider updating the column coordinate of element $m^A_{(i,j)}$. The updating mechanism of column coordinates is similar to that of row coordinates. Thus, we can get the update results as follows. For any $q_{k_c-1}-k_c+2 \le j < q_{k_c}-k_c+1$, we can obtain $m'^A_{(*,j)} = m^A_{(*,j+k_c-1)}$. For any $q_{n'}-n'+1 \le i \le n-n'$, we can obtain $m'^A_{(*,j)} = m^A_{(*,j+n')}$. Therefore, when the dominance relation matrix $\mathbb{M}_{U-U^-}^{\succeq A}$ is updated, there are four cases for updating the coordinates of the elements $m'^A_{(i,j)}$, as follows. (1) if $(q_{k_r-1}-k_r+2 \le i < q_{k_r}-k_r+1) \wedge (q_{k_c-1}-k_c+2 \le j < q_{k_c}-k_c+1)$ holds, then $m'^A_{(i,j)} = m^A_{(i+k_r-1,j+k_c-l)}$; (2) if $(q_{k_r-1}-k_r+2 \le i < q_{k_r}-k_r+1) \wedge (q_{n'}-n'+1 \le j \le n-n')$ holds, then $m'^A_{(i,j)} = m^A_{(i+k_r-1,j+n')}$; (3) if $(q_{n'}-n'+1 \le i \le n-n') \wedge (q_{k_c-1}-k_c+2 \le j < q_{k_c}-k_c+1)$ holds, then $m'^A_{(i,j)} = m^A_{(i+n',j+k_c-l)}$; (4) if $(q_{n'}-n'+1 \le i \le n-n') \wedge (q_{n'}-n'+1 \le j \le n-n')$ holds, then $m'^A_{(i,j)} = m^A_{(i+n',j+n')}$. In summary, we can get the characteristic function of $\mathbb{M}_{U-U^-}^{\succeq A}$, i.e., Eq. (22). □

Example 8. Continuing from Example 3, the object set $U^- = \{x_1, x_5\}$ is deleted from Table 2. $\mathbb{M}_{U-U^-}^{\succeq C}$ can be updated by using Proposition 4.3 as

$$\mathbb{M}_{U-U^-}^{\succeq C} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 & 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 1 \end{bmatrix}_{5\times 5},$$

where the red marked elements represent the deleted elements.

Proposition 4.4 (*Update Dominance Diagonal Matrix*)**.** *Given an OIS $S^{\succeq} = (U, AT, V, f)$, where $U = \{x_1, x_2, \dots, x_n\}$. For any $A \subseteq AT$,*

Table 5
The illustration of the calculation process of Algorithm 2.

Steps	Operations
1	Add object set as $U' \leftarrow U \cup U^+$. Initialize original reduct as $B \leftarrow Red_U$. Initialize known matrices as $\mathbb{M}_{U'}^{\succeq C} \leftarrow \mathbb{M}_U^{\succeq C}$, $\mathbb{M}_{U'}^{\succeq B} \leftarrow \mathbb{M}_U^{\succeq B}$, $\mathbb{M}_{U'}^{\succeq d} \leftarrow \mathbb{M}_U^{\succeq d}$, $\mathbb{D}_{U'}^{\succeq C} \leftarrow \mathbb{D}_U^{\succeq C}$, $\mathbb{D}_{U'}^{\succeq C\cup\{d\}} \leftarrow \mathbb{D}_U^{\succeq C\cup\{d\}}$, $\mathbb{D}_{U'}^{\succeq B} \leftarrow \mathbb{D}_U^{\succeq B}$, and $\mathbb{D}_{U'}^{\succeq B\cup\{d\}} \leftarrow \mathbb{D}_U^{\succeq B\cup\{d\}}$.
2	Update dominance relation matrices as $\mathbb{M}_{U'}^{\succeq C}$, $\mathbb{M}_{U'}^{\succeq B}$, and $\mathbb{M}_{U'}^{\succeq d}$, then turn to Step 3.
3	Calculate dominance relation matrices as $\mathbb{M}_{U'}^{\succeq C\cup\{d\}}$ and $\mathbb{M}_{U'}^{\succeq B\cup\{d\}}$ on the basis of the updated dominance relation matrices in Step 2, then turn to Step 4.
4	Update dominance diagonal matrices as $\mathbb{D}_{U'}^{\succeq C}$, $\mathbb{D}_{U'}^{\succeq C\cup\{d\}}$, $\mathbb{D}_{U'}^{\succeq B}$, and $\mathbb{D}_{U'}^{\succeq B\cup\{d\}}$ on the basis of the updated dominance relation matrices in Steps 2–3, then turn to Step 5.
5	Calculate two new MDCEs as $MDH_{d\|C}^{\succeq}(U') = 0.6490$, $MDH_{d\|B}^{\succeq}(U') = 0.6490$ on the basis of the updated dominance diagonal matrices in Step 4, then turn to Steps 6–10.
6–10	Because the condition $MDH_{d\|C}^{\succeq}(U') = MDH_{d\|B}^{\succeq}(U')$ is satisfied, then the next step turns to Step 17.
17–22	For each $a \in B$, calculate $MDH_{d\|(B-\{a\})}^{\succeq}(U')$ as $MDH_{d\|B-\{a_1\}}^{\succeq}(U') = 0.6490$, $MDH_{d\|B-\{a_2\}}^{\succeq}(U') = 0.6105$, $MDH_{d\|B-\{a_3\}}^{\succeq}(U') = 0.8912$. Because $MDH_{d\|(B-\{a_1\})}^{\succeq}(U') = MDH_{d\|B}^{\succeq}(U')$, a_1 is a redundant attribute, and then delete it from B, i.e., $B = \{a_2, a_3\}$, then turn to Steps 23–24.
23–24	Output the final reduct as $Red_{U'} = \{a_2, a_3\}$.

let the dominance diagonal matrix on U with respect to A is $\mathbb{D}_U^{\succeq A} = [d^A_{(i,j)}]_{n\times n}$, the object set $U^- = \{x_{q_1}, x_{q_2}, \ldots, x_{q_{n'}}\}$ is deleted from $S^{\succeq}$. The updated dominance diagonal matrix on $U - U^-$ with respect to A is denoted as $\mathbb{D}_{U-U^-}^{\succeq A} = [d'^A_{(i,j)}]_{(n-n')\times(n-n')}$, where

$$d'^A_{(i,j)} = \begin{cases} d^A_{(i+k-1,j+k-1)} - \sum_{t=1}^{n'} m^A_{(i+k-1,q_t)}, & q_{k-1} - k + 2 \le i, \\ & j < q_k - k + 1, i = j; \\ d^A_{(i+n',j+n')} - \sum_{t=1}^{n'} m^A_{(i+n',q_t)}, & q_{n'} - n' + 1 \le i, \\ & j \le n - n', i = j; \\ 0, & 1 \le i, j \le n - n', i \ne j, \end{cases} \tag{23}$$

where $1 \le k \le n'$.

Proof. Assuming that U^- is deleted from $S^{\succeq}$, then $U - U^- = \{x_1, x_2, \ldots, x_{n-n'}\}$. We can get $\mathbb{D}_{U-U^-}^{\succeq A} = [d'^A_{(i,j)}]_{(n-n')\times(n-n')}$ according to Definition 3.3. Actually, for any $1 \le i, j \le n - n', i \ne j$, $d'^A_{(i,j)} = 0$ always holds. So we can get that $d'^A_{(i,j)}$ remains unchanged for any $1 \le i, j \le n - n', i \ne j$, i.e., $d'^A_{(i,j)} = d^A_{(i,j)} = 0$. For any $1 \le i, j \le n - n', i = j$, we can get $d'^A_{(i,j)} = \sum_{l=1}^{n-n'} m'^A_{(i,l)} = \sum_{l=1}^{n} m'^A_{(i,l)} - \sum_{t=1}^{n'} m'^A_{(i,t)}$. Since $m'^A_{(i,j)} = m^A_{(i,j)}$ is true for any $1 \le i, j \le n$, we can acquire $d'^A_{(i,j)} = \sum_{l=1}^{n} m^A_{(i,l)} - \sum_{t=1}^{n'} m^A_{(i,t)} = d^A_{(i,j)} - \sum_{t=1}^{n'} m^A_{(i,t)}$. Based on Proposition 4.3 and Definition 3.3, for any $q_{k-1} \le i, j < q_k, i = j$, the row and column coordinates of the element $d^A_{(i,j)}$ should be shifted forward by $k-1$ positions simultaneously. Thus, we can obtain $d'^A_{(i,j)} = d^A_{(i+k-1,j+k-1)} - \sum_{t=1}^{n'} m^A_{(i+k-1,q_t)}$ for any $q_{k-1} - k + 2 \le i, j < q_k - k + 1, i = j$. On the other hand, for any $q_{n'} - n' + 1 \le i, j \le n - n', i = j$, the row and column coordinates of the element $d^A_{(i,j)}$ should be shifted forward by n' positions simultaneously. So we can get $d'^A_{(i,j)} = d^A_{(i+n',j+n')} - \sum_{t=1}^{n'} m^A_{(i+n',q_t)}$ for any $q_{n'} - n' + 1 \le i, j \le n - n', i = j$. In summary, we can get the characteristic function of $\mathbb{D}_{U-U^-}^{\succeq A}$, i.e., Eq. (23). □

Example 9. Continuing from Example 8, on the basis of $\mathbb{D}_U^{\succeq C}$, $\mathbb{D}_{U-U^-}^{\succeq C}$ can be updated by using Proposition 4.4 as

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

$$= \begin{bmatrix} 2 & 0 & 0 & 0 & 0 \\ 0 & 2 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 2 & 0 \\ 0 & 0 & 0 & 0 & 2 \end{bmatrix}_{5\times 5},$$

where the red marked elements represent the deleted elements, the blue marked elements represent the changed elements.

4.2.2. An incremental attribute reduction algorithm w.r.t. deleting objects

An incremental attribute reduction algorithm while deleting multiple objects (IAR-D) is designed in Algorithm 3, which is based on the updating principle of MDCE. Then, we analyze the time and space complexity of the proposed algorithm. Lastly, an example is presented to demonstrate computation process of algorithm IAR-D.

The detailed description of the steps in Algorithm 3 and their time complexity are given as follows. Steps 2–3 compute new dominance relation matrices and its dominance diagonal matrices in an incremental manner by using Propositions 4.3 and 4.4, and its time complexity is $O(|U^-||U|)$. Step 4 computes new MDCE by using Corollary 3.1. Steps 5–9 determine whether the new MDCE under the original attribute subset (i.e. original reduct) is equal to that of under the entire attribute set; if yes, then keep the original attribute subset unchanged. Steps 10–15 arrange the remaining attributes in descending order, and incrementally update the selected attribute subset until Step 11 no longer holds, its time complexity is $O((|C| - |B|)|U'|^2)$. Steps 16–21 delete redundant attributes from the selected attribute subset and its time complexity is $O(|B|^2|U'|^2)$. Steps 22–23 output a final reduct. In summary, the time complexity of Algorithm 3 is $O(|U^-||U| + (|C| - |B|)|U'|^2 + |B|^2|U'|^2)$. In addition, the space complexity of Algorithm 3 is $O(|U'|^2+(|C|-|B|)|U'|^2)$. The following we compare

Algorithm 3 IAR-D algorithm

Input:
(1) An original ODS $S^{\succeq} = (U, C \cup \{d\}, V, f)$, where $U = \{x_1, x_2, \ldots, x_n\}$, $U^{-} = \{x_{q_1}, x_{q_2}, \ldots, x_{q_{n'}}\}$ is a deleted object set;
(2) The original reduct Red_U on U;
(3) The original dominance relation matrices $\mathbb{M}_U^{\succeq C} = [m_{(i,j)}^{C}]_{n\times n}$, $\mathbb{M}_U^{\succeq C\cup\{d\}} = [m_{(i,j)}^{C\cup\{d\}}]_{n\times n}$, $\mathbb{M}_U^{\succeq Red_U} = [m_{(i,j)}^{Red_U}]_{n\times n}$, and $\mathbb{M}_U^{\succeq Red_U\cup\{d\}} = [m_{(i,j)}^{Red_U\cup\{d\}}]_{n\times n}$.
(4) The original dominance diagonal matrices $\mathbb{D}_U^{\succeq C} = [d_{(i,j)}^{C}]_{n\times n}$, $\mathbb{D}_U^{\succeq C\cup\{d\}} = [d_{(i,j)}^{C\cup\{d\}}]_{n\times n}$, $\mathbb{D}_U^{\succeq Red_U} = [d_{(i,j)}^{Red_U}]_{n\times n}$, and $\mathbb{D}_U^{\succeq Red_U\cup\{d\}} = [d_{(i,j)}^{Red_U\cup\{d\}}]_{n\times n}$.

Output: A new reduct $Red_{U'}$ on $U - U^{-}$

1: Initialize $B \leftarrow Red_U$, $U' \leftarrow U - U^{-}$, $\mathbb{M}_{U'}^{\succeq C} \leftarrow \mathbb{M}_U^{\succeq C}$, $\mathbb{M}_{U'}^{\succeq C\cup\{d\}} \leftarrow \mathbb{M}_U^{\succeq C\cup\{d\}}$, $\mathbb{M}_{U'}^{\succeq B} \leftarrow \mathbb{M}_U^{\succeq B}$, $\mathbb{M}_{U'}^{\succeq B\cup\{d\}} \leftarrow \mathbb{M}_U^{\succeq B\cup\{d\}}$, $\mathbb{D}_{U'}^{\succeq C} \leftarrow \mathbb{D}_U^{\succeq C}$, $\mathbb{D}_{U'}^{\succeq C\cup\{d\}} \leftarrow \mathbb{D}_U^{\succeq C\cup\{d\}}$, $\mathbb{D}_{U'}^{\succeq B} \leftarrow \mathbb{D}_U^{\succeq B}$, and $\mathbb{D}_{U'}^{\succeq B\cup\{d\}} \leftarrow \mathbb{D}_U^{\succeq B\cup\{d\}}$;
2: Compute new dominance relation matrices $\mathbb{M}_{U'}^{\succeq C} \leftarrow [m_{(i,j)}^{\prime C}]_{(n-n')\times(n-n')}$, $\mathbb{M}_{U'}^{\succeq C\cup\{d\}} \leftarrow [m_{(i,j)}^{\prime C\cup\{d\}}]_{(n-n')\times(n-n')}$, $\mathbb{M}_{U'}^{\succeq B} \leftarrow [m_{(i,j)}^{\prime B}]_{(n-n')\times(n-n')}$, and $\mathbb{M}_{U'}^{\succeq B\cup\{d\}} \leftarrow [m_{(i,j)}^{\prime B\cup\{d\}}]_{(n-n')\times(n-n')}$ by using Proposition 4.3;
3: Compute new dominance diagonal matrices $\mathbb{D}_{U'}^{\succeq C} \leftarrow [d_{(i,j)}^{\prime C}]_{(n-n')\times(n-n')}$, $\mathbb{D}_{U'}^{\succeq C\cup\{d\}} \leftarrow [d_{(i,j)}^{\prime C\cup\{d\}}]_{(n-n')\times(n-n')}$, $\mathbb{D}_{U'}^{\succeq B} \leftarrow [d_{(i,j)}^{\prime B}]_{(n-n')\times(n-n')}$, and $\mathbb{D}_{U'}^{\succeq B\cup\{d\}} \leftarrow [d_{(i,j)}^{\prime B\cup\{d\}}]_{(n-n')\times(n-n')}$ by using Proposition 4.4;
4: Compute new MDCE $MDH_{d|C}^{\succeq}(U')$ and $MDH_{d|B}^{\succeq}(U')$ by using Corollary 3.1;
5: **if** $MDH_{d|C}^{\succeq}(U') = MDH_{d|B}^{\succeq}(U')$, **then**
6: go to step 16;
7: **else**
8: go to step 10;
9: **end if**
10: For each $a \in (C - B)$, compute $Msig_{outer}^{\succeq U'}(a, B, d)$ by using Corollary 3.3, then construct a descending sequence by $Msig_{outer}^{\succeq U'}(a, B, d)$, and record the results by $\{a'_1, a'_2, \ldots, a'_{|C-B|}\}$;
11: **while** $MDH_{d|C}^{\succeq}(U') \neq MDH_{d|B}^{\succeq}(U')$ **do**
12: **for** $h = 1$ to $|C - B|$ **do**
13: select $B \leftarrow B \cup \{a'_h\}$ and compute $MDH_{d|B}^{\succeq}(U')$;
14: **end for**
15: **end while**
16: **for** each $a \in B$ **do**
17: compute $MDH_{d|(B-\{a\})}^{\succeq}(U')$;
18: **if** $MDH_{d|(B-\{a\})}^{\succeq}(U') = MDH_{d|B}^{\succeq}(U')$, **then**
19: $B \leftarrow B - \{a\}$;
20: **end if**
21: **end for**
22: $Red_{U'} \leftarrow B$;
23: **return** $Red_{U'}$;

the time and space complexity of algorithms HAR and IAR-D in Table 6.

From Table 6, the time complexity of algorithm IAR-D is much lower than that of algorithm HAR. The main reason is that algorithm IAR-D uses the previous knowledge when calculating the new reduct, while algorithm HAR calculates a new reduct from scratch, which does not use the previous knowledge. Thus, algorithm HAR is very time consuming for computing a new reduct. Moreover, the storage space required by algorithm HAR is also greater than that of algorithm IAR-D.

The following we present an example to demonstrate the detailed steps for calculating a new reduct by using Algorithm 3.

Example 10. Continuing from Example 4, the known knowledge of the original ODS includes the reduct $Red_U = \{a_1, a_2, a_3\}$; the dominance relation matrices $\mathbb{M}_U^{\succeq C}$, $\mathbb{M}_U^{\succeq C\cup\{d\}}$, $\mathbb{M}_U^{\succeq Red_U}$, and $\mathbb{M}_U^{\succeq Red_U\cup\{d\}}$; and dominance diagonal matrices $\mathbb{D}_U^{\succeq C}$, $\mathbb{D}_U^{\succeq C\cup\{d\}}$, $\mathbb{D}_U^{\succeq Red}$, and $\mathbb{D}_U^{\succeq Red\cup\{d\}}$. The object set $U^{-} = \{x_1, x_5\}$ is deleted from U. The calculation process of Algorithm 3 is shown in Table 7.

5. Experimental analysis

In this section, a series of experiments have been carried out to demonstrate the effectiveness and efficiency of the proposed incremental algorithms. In these experiments, the summary of nine employed datasets are shown in Table 8, seven of which are obtained from UCI. In order to evaluate the proposed incremental algorithms in larger datasets, two artificial datasets are also provided, i.e., AD1 and AD2. In datasets Mice Protein Expression and Dermatology, we delete some objects and attributes with missing values. Some datasets have been used for updating approximations in dynamic ordered datasets [68–71] and attribute reduction work [40,72]. In this paper, all algorithms are coded in Java language and run on a computer with 3.20 GHz CPU Intel(R) Core(TM) i7-8700, 16.0 GB of memory, and 64-bit Windows 10 operation system.

In order to evaluate the performance of the proposed incremental attribute reduction algorithms IAR-A and IAR-D, we compare the proposed algorithms with four existing attribute reduction algorithms HAR, DRSQR [40], FEAR [73], and NRSAR [74]. Algorithm HAR is a general heuristic attribute reduction algorithm using dominance conditional entropy given in Algorithm 1. Algorithm DRSQR is a dominance-based rough set based QuickReduct algorithm. Algorithm FEAR is a fuzzy entropy based attribute reduction algorithm. Algorithm NRSAR is an attribute reduction algorithm using neighborhood entropy based on neighborhood rough set. Moreover, we use four classifiers BayesNet, RandomTree, OLM, and OSDL in Weka to test the classification effect of the reduct generated using them. 10-fold cross-validation is adopted in classification.

5.1. Performance evaluations of algorithm IAR-A when adding multiple objects

In this subsection, we evaluate the performance of algorithm IAR-A in terms of effectiveness and efficiency. In terms of effectiveness, we compare algorithm IAR-A with other four algorithms from classification accuracy. In terms of efficiency, we compare algorithm IAR-A with other four algorithms from two aspects: computational time and speed-up ratio. The specific experimental design is shown below.

5.1.1. Effectiveness comparison

This subsection compares the effectiveness of algorithm IAR-A with other four algorithms. The dynamic datasets are simulated by the following way. We randomly select 50% of the objects from each dataset in Table 8 as the original object set, and the remaining 50% objects are treated as the added objects. Algorithms IAR-A, HAR, DRSQR, FEAR, and NRSAR are used to calculate a new reduct when the remaining 50% of the objects are added to the original 50% object set. Subsequently, we separately tested the classification accuracy of the reducts generated by using these five algorithms. The experimental results are presented in Tables 9 and 10, where "Raw" represents the classification accuracy of the raw attribute set. Note that in Table 9, the number in bracket after each classification accuracy result indicates the size of the generated reduct. In the following, the structure of Tables 10, 11, and 12 is similar to that of Table 9.

As indicated in Tables 9 and 10, the classification accuracy of the reducts generated using algorithm IAR-A in the four classifiers is equal or even slightly higher than that of the reducts generated

Table 6
The comparisons of the time and space complexity of algorithms HAR and IAR-D.

Algorithm	HAR	IAR-D
Time complexity	$O(\|C\|\|U'\|^2+\|C\|^2\|U'\|^2+\|C\|^2\|U'\|^2+\|B\|^2\|U'\|^2)$	$O(\|U^-\|\|U\|+(\|C\|-\|B\|)\|U'\|^2+\|B\|^2\|U'\|^2)$
Space complexity	$O(\|U'\|^2+\|C\|\|U'\|^2)$	$O(\|U'\|^2+(\|C\|-\|B\|)\|U'\|^2)$

Table 7
The illustration of the calculation process of Algorithm 3.

Steps	Operations
1	Delete object set as $U' \leftarrow U - U^-$. Initialize original reduct as $B \leftarrow Red_U$. Initialize known matrices as $\mathbb{M}_{U'}^{\succeq C} \leftarrow \mathbb{M}_{U}^{\succeq C}$, $\mathbb{M}_{U'}^{\succeq B} \leftarrow \mathbb{M}_{U}^{\succeq B}$, $\mathbb{M}_{U'}^{\succeq d} \leftarrow \mathbb{M}_{U}^{\succeq d}$, $\mathbb{D}_{U'}^{\succeq C} \leftarrow \mathbb{D}_{U}^{\succeq C}$, $\mathbb{D}_{U'}^{\succeq C\cup\{d\}} \leftarrow \mathbb{D}_{U}^{\succeq C\cup\{d\}}$, $\mathbb{D}_{U'}^{\succeq B} \leftarrow \mathbb{D}_{U}^{\succeq B}$, $\mathbb{D}_{U'}^{\succeq B\cup\{d\}} \leftarrow \mathbb{D}_{U}^{\succeq B\cup\{d\}}$.
2	Update dominance relation matrices as $\mathbb{M}_{U'}^{\succeq C}$, $\mathbb{M}_{U'}^{\succeq B}$, $\mathbb{M}_{U'}^{\succeq C\cup\{d\}}$, and $\mathbb{M}_{U'}^{\succeq B\cup\{d\}}$, then turn to Step 3.
3	Update dominance diagonal matrices as $\mathbb{D}_{U'}^{\succeq C}$, $\mathbb{D}_{U'}^{\succeq C\cup\{d\}}$, $\mathbb{D}_{U'}^{\succeq B}$, and $\mathbb{D}_{U'}^{\succeq B\cup\{d\}}$, then turn to Step 4.
4	Calculate two new MDCEs as $MDH_{d\|C}^{\succeq}(U')=0$, $MDH_{d\|B}^{\succeq}(U')=0$ on the basis of the updated dominance diagonal matrices in Step 4, then turn to Steps 5–9.
5–9	Because the condition $MDH_{d\|C}^{\succeq}(U')=MDH_{d\|B}^{\succeq}(U')$ is satisfied, then turn to Step 16.
16–21	For each $a \in B$, calculate $MDH_{d\|(B-\{a\})}^{\succeq}(U')$ as $MDH_{d\|B-\{a_1\}}^{\succeq}(U')=0$, $MDH_{d\|B-\{a_2\}}^{\succeq}(U')=0$, $MDH_{d\|B-\{a_3\}}^{\succeq}(U')=0.2340$. Because the conditions $MDH_{d\|(B-\{a_1\})}^{\succeq}(U')=MDH_{d\|B}^{\succeq}(U')$ and $MDH_{d\|(B-\{a_2\})}^{\succeq}(U')=MDH_{d\|B}^{\succeq}(U')$ are satisfied, then a_1 and a_2 are redundant attributes, and then delete them from B, i.e., $B=\{a_3\}$, then turn to Steps 22–23.
22–23	Output the final reduct as $Red_{U'}=\{a_3\}$.

Table 8
The description of datasets.

No.	Datasets	Abbreviation	Objects	Attributes	Classes
1	Wine	Wine	178	13	3
2	Leaf	Leaf	340	15	30
3	Ionosphere	Iono	351	34	2
4	Dermatology	Derm	358	34	6
5	Libras Movement	Libras	360	90	15
6	Mice Protein Expression	Mice	1077	68	8
7	Cardiotocography	Card	2093	21	3
8	Artificial Data 1	AD1	7180	34	2
9	Artificial Data 2	AD2	8501	34	2

using other four algorithms for most datasets, as fully illustrated by their average values. This finding proves that the generated reduct using algorithm IAR-A is feasible. Hence, we can conclude from Tables 9 and 10 that algorithm IAR-A is effective.

5.1.2. Efficiency comparison

In this subsection, to test the efficiency of algorithm IAR-A, we compare algorithm IAR-A with other four algorithms in terms of computational time and speed-up ratio. The dynamic change of datasets is simulated in the following way. For each dataset in Table 8, five testing sets are constructed. First, 50% of the objects are randomly selected as the original object set. Subsequently, we randomly add objects from the remaining 50% objects to the original object set to obtain dynamic datasets for testing (i.e., 10%, 20%, 30%, 40%, and 50% of the objects from the remaining 50% objects are randomly selected and added to the original object set). Then, we compare the time consumed via using different algorithms on these datasets. Fig. 1 shows the detailed change trend of these five algorithms with different size of datasets. The abscissa represents the size of added datasets, and the ordinate represents the computational time. For some datasets, the calculation time span of these five algorithms is relatively large. This situation causes some curves to be very dense in one coordinate system, and it is difficult to distinguish them. To solve this issue, we added corresponding subgraphs to the figures of dense curves to clearly show the change of these curves.

For example, Fig. 1(b) shows that the curves of algorithms NRSAR, FEAR, HAR, and IAR-A are too dense, so we add a subgraph to Fig. 1(b), to clearly show the changes trend of these four algorithms. The following, the structures of Figs. 1(h), 3(b), 3(c), 3(e), 3(f), 3(h), and 3(i) are similar to those of Fig. 1(b).

From Fig. 1, the computational time of these five algorithms increases as the size of added object sets increase. It can be observed from each sub-figure that the computational time of algorithm IAR-A is smaller than that of other four algorithms. In particular, for datasets AD1 and AD2, the computational time of algorithm IAR-A is significantly shorter than that of other four algorithms. This indicates that algorithm IAR-A can compute a reduct in a much shorter time, especially for large datasets, the time-saving effect is more obvious. The main reason is that algorithm IAR-A uses previous knowledge, which avoids some re-calculation. Conversely, other four algorithms retrain the changed datasets from scratch, which do not use the knowledge already generated in the original dataset and do a lot of repeated calculations. Therefore, algorithm IAR-A is more efficient than other four algorithms.

Subsequently, we again demonstrate the efficiency of algorithm IAR-A from the aspect of speed-up ratio. We compute the speed-up ratio that algorithm IAR-A relates to other four algorithms on the basis of the results shown in Fig. 1. The experimental results are shown in Fig. 2, where the abscissa denotes the size of the added datasets and the ordinate denotes the value of the speed-up ratio. For different datasets, the speed-up ratio span of these algorithms is relatively large. Similarly, this situation also causes the curves to be very dense in one coordinate system, and it is difficult to distinguish them. To solve this issue, we show some experimental results with a small numerical range in the form of subgraph. For example, Fig. 2(a) shows the experimental results of datasets Derm, Card, AD1, and AD2, and their numerical range is [1, 90]. The subgraph in Fig. 2(a) shows the experimental results of datasets Wine, Leaf, Iono, Libras, and Mice, and their numerical range is [1, 8]. The following, the structures of Figs. 2(b), 2(c), 2(d), 4(a), 4(b), 4(c), and 4(d) are similar to those of Fig. 2(a).

From Fig. 2, it can be observed that all speed-up ratios are greater than 1. This indicates that for all datasets, algorithm IAR-A is faster than the other four algorithms. What is more, for most datasets, algorithm IAR-A is at least nearly 2 times or more faster than the other four algorithms. It is worth pointing out that for some large datasets, such as AD1 and AD2, algorithm IAR-A is even tens of times faster than algorithms NRSAR, FEAR, and DRSQR. The experimental results again prove that the efficiency of algorithm IAR-A.

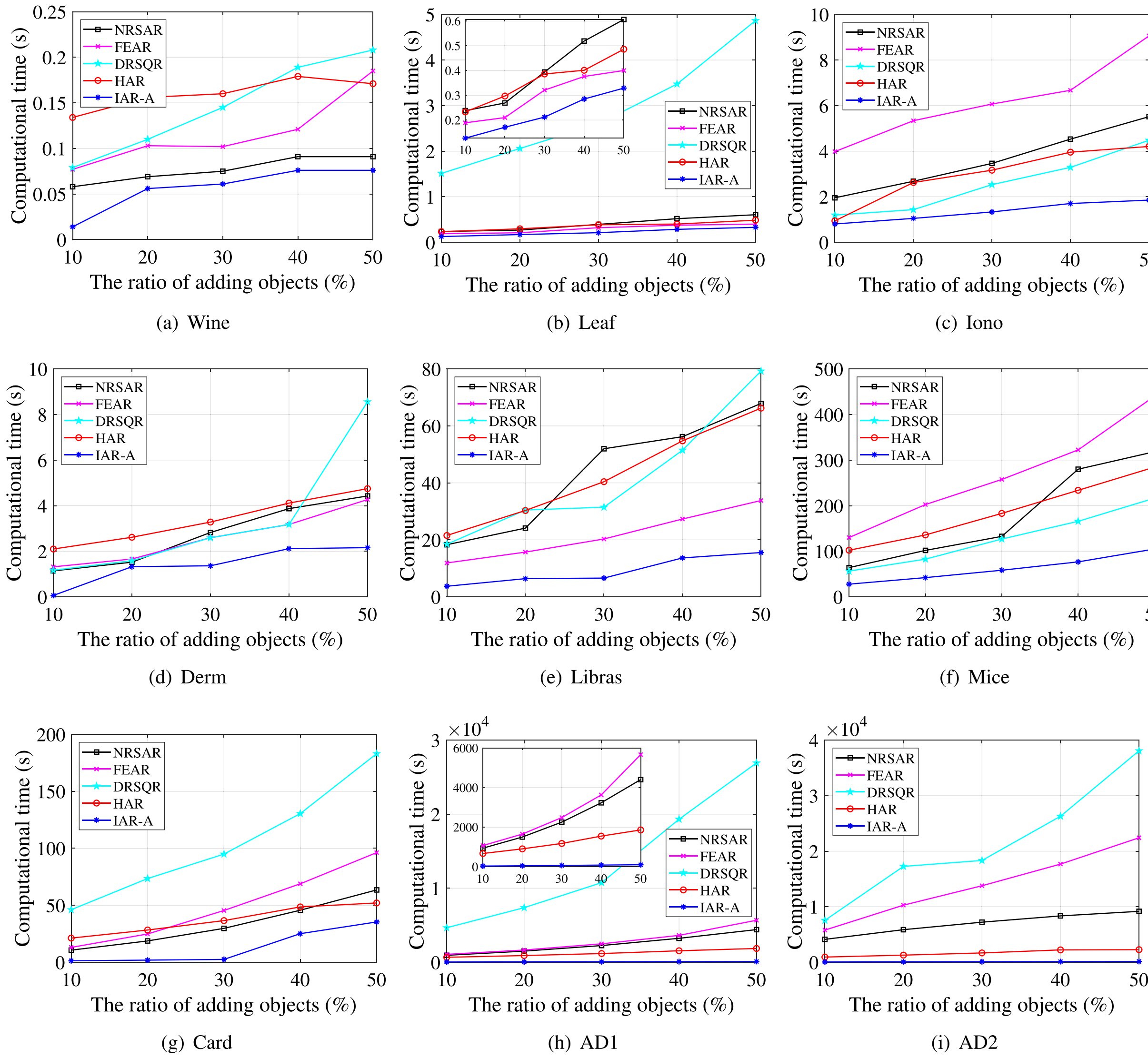


Fig. 1. The computational time of different algorithms versus different ratios of adding objects.

Table 9
The comparison of classification accuracies of different algorithms on BayesNet and RandomTree (%)

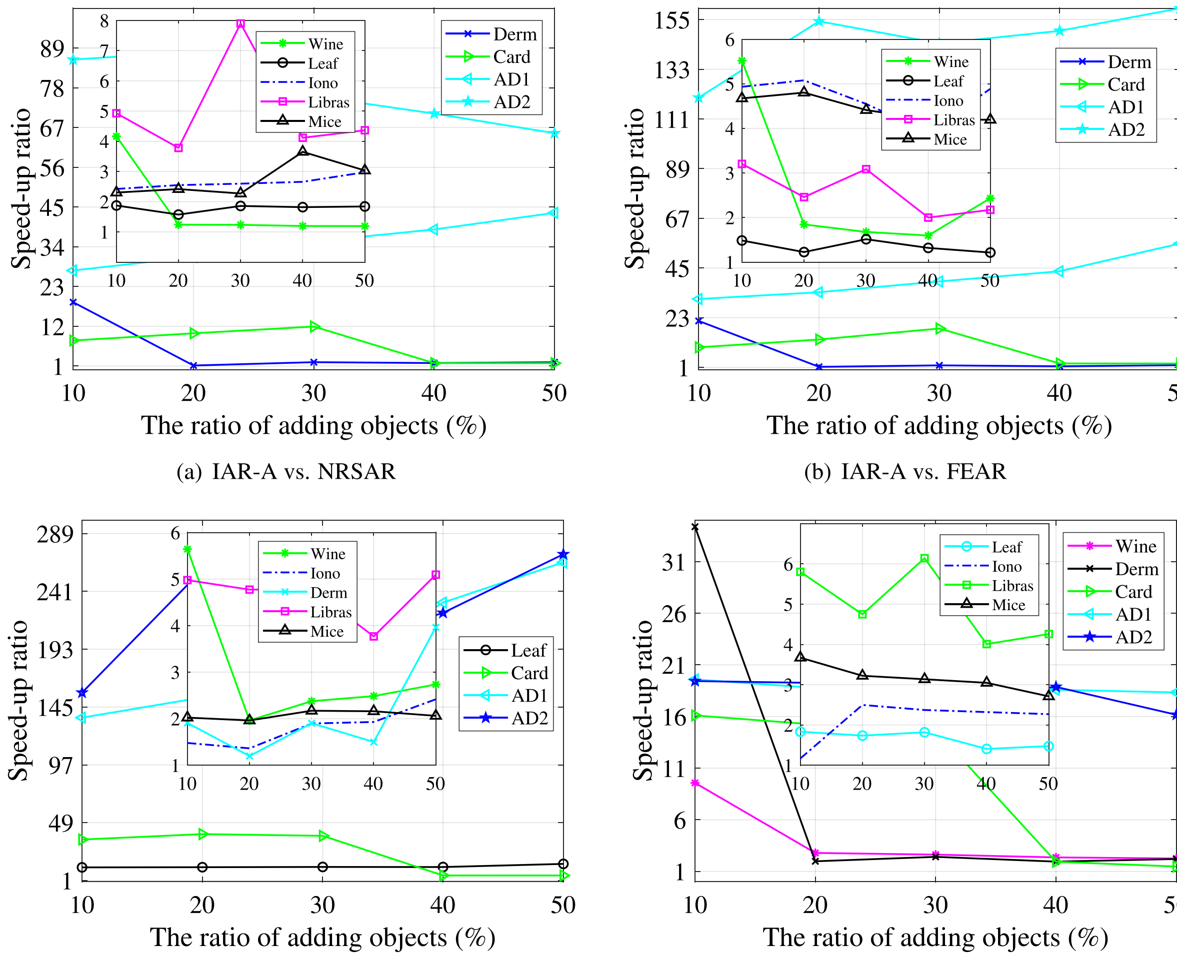
Datasets	BayesNet						RandomTree					
	Raw	NRSAR	FEAR	DRAQR	HAR	IAR-A	Raw	NRSAR	FEAR	DRAQR	HAR	IAR-A
Wine	98.88	80.90 (1)	85.44 (11)	94.94 (11)	98.88 (12)	**98.88 (12)**	91.01	72.47 (1)	84.94 (11)	82.70 (11)	91.01 (12)	**91.01 (12)**
Leaf	63.82	65.00 (13)	62.65 (13)	19.41 (1)	61.47 (9)	**68.82 (9)**	59.12	56.76 (13)	60.18 (13)	19.12 (1)	53.82 (9)	**60.29 (9)**
Iono	89.46	74.93 (1)	**90.31 (17)**	88.32 (21)	88.32 (23)	88.03 (23)	88.60	74.93 (1)	88.03 (17)	86.89 (21)	84.90 (23)	**88.76 (23)**
Derm	97.75	69.57 (4)	98.45 (11)	98.16 (19)	88.03 (20)	**98.62 (21)**	90.14	70.28 (4)	90.71 (11)	**100 (19)**	85.76 (20)	90.70 (21)
Libras	62.50	56.38 (27)	62.77 (31)	56.38 (27)	60.33 (33)	**63.90 (33)**	66.11	61.66 (27)	**67.22 (31)**	65.27 (27)	65.56 (33)	65.00 (33)
Mice	83.10	65.55 (5)	**84.02 (43)**	75.11 (11)	81.34 (27)	81.06 (27)	81.15	81.15 (5)	80.50 (43)	83.37 (11)	79.57 (27)	**84.27 (27)**
Card	86.77	78.30 (1)	86.71 (17)	85.52 (13)	87.14 (13)	**87.14 (13)**	91.30	78.30 (1)	91.30 (17)	91.20 (13)	91.59 (13)	**91.59 (13)**
AD1	99.68	74.69 (1)	98.79 (15)	99.35 (23)	99.76 (23)	**99.76 (23)**	100	74.69 (1)	100 (15)	100 (23)	100 (23)	**100 (23)**
AD2	99.69	74.59 (1)	99.69 (16)	99.69 (23)	99.92 (23)	**99.92 (23)**	100	74.59 (1)	100 (16)	100 (23)	100 (23)	**100 (23)**
Average	86.85	71.10	85.43	79.65	85.02	**87.35**	85.27	71.65	84.76	80.95	83.58	**85.74**

5.1.3. Summary

From the comparisons of effectiveness and efficiency between five different algorithms, it can be concluded that our algorithm IAR-A is better than the other four algorithms. The computational time required to obtain a feasible reduct by algorithm IAR-A is much shorter than that of the other four algorithms. Therefore, when multiple objects are added to an ODS, we can obtain a feasible reduct by algorithm IAR-A more efficiently.

5.2. Performance evaluations of algorithm IAR-d when deleting multiple objects

This subsection evaluates the performance of algorithm IAR-D in terms of effectiveness and efficiency. Algorithm IAR-D and other four algorithms are compared in the same scheme as the previous subsection. The specific experimental design is described as follows.



(a) IAR-A vs. NRSAR (b) IAR-A vs. FEAR (c) IAR-A vs. DRSQR (d) IAR-A vs. HAR

Fig. 2. The speed-up ratios that algorithm IAR-A relates to different algorithms.

Table 10
The comparison of classification accuracies of different algorithms on OLM and OSDL (%)

Datasets	OLM						OSDL					
	Raw	NRSAR	FEAR	DRAQR	HAR	IAR-A	Raw	NRSAR	FEAR	DRAQR	HAR	IAR-A
Wine	30.90	33.15 (1)	32.58 (11)	34.83 (11)	42.58 (12)	**42.58 (12)**	46.07	38.76 (1)	44.94 (11)	41.12 (11)	47.75 (12)	**47.75 (12)**
Leaf	4.12	4.12 (13)	4.12 (13)	4.94 (1)	5.00 (9)	**5.41 (9)**	13.82	14.12 (13)	14.12 (13)	3.53 (1)	8.53 (9)	**14.41 (9)**
Iono	63.00	64.10 (1)	65.53 (17)	63.82 (21)	63.82 (23)	**66.53 (23)**	50.71	35.90 (1)	40.46 (17)	43.59 (21)	45.58 (23)	**53.02 (23)**
Derm	86.05	49.57 (4)	80.70 (11)	87.18 (19)	86.05 (20)	**88.11 (21)**	83.38	35.77 (4)	51.83 (11)	83.94 (19)	81.69 (20)	**88.59 (21)**
Libras	24.72	23.33 (27)	25.00 (31)	24.72 (27)	24.72 (33)	**25.27 (33)**	17.22	13.61 (27)	16.66 (31)	18.61 (27)	19.44 (33)	**19.44 (33)**
Mice	14.02	16.52 (5)	14.39 (43)	**26.18 (11)**	17.82 (27)	16.80 (27)	16.80	12.72 (5)	24.60 (43)	**43.45 (11)**	32.86 (27)	29.61 (27)
Card	79.93	66.36 (1)	80.64 (17)	80.55 (13)	86.23 (13)	**86.23 (13)**	72.47	78.30 (1)	67.46 (17)	77.49 (13)	76.63 (13)	**79.63 (13)**
AD1	**94.72**	63.89 (1)	85.62 (15)	94.65 (23)	93.47 (23)	93.47 (23)	88.12	36.10 (1)	80.04 (15)	88.12 (23)	88.75 (23)	**88.75 (23)**
AD2	93.47	63.88 (1)	83.81 (16)	90.28 (23)	93.47 (23)	**93.47 (23)**	88.06	36.11 (1)	81.34 (16)	87.63 (23)	88.75 (23)	**88.75 (23)**
Average	54.55	42.77	52.49	56.35	57.02	**57.54**	52.96	33.49	46.83	54.16	54.44	**56.66**

5.2.1. Effectiveness comparison

In this subsection, we compare the effectiveness of algorithm IAR-D with other four algorithms. The dynamic datasets are simulated in the following way. For each dataset in Table 8, we randomly select the 50% objects as deleted multiple objects. Algorithms IAR-D, HAR, DRSQR, FEAR, and NRSAR are employed to compute a new reduct when the selected 50% objects are deleted. Subsequently, we separately tested the classification accuracy of the reducts generated by using these five algorithms. The experimental results are recorded in Tables 11 and 12.

From Tables 11 and 12, we find that the classification accuracy of the reducts generated using the proposed algorithm IAR-D in the four classifiers is very close or even slightly higher than that of the reducts generated using other four algorithms for most datasets, as fully illustrated by their average values. This finding proves that the reduct generated by using algorithm IAR-D have the same or even higher classification ability than the other four algorithms. Hence, the experimental results indicate that algorithm IAR-D is effective.

5.2.2. Efficiency comparison

In this subsection, to demonstrate the efficiency of our algorithm IAR-D, we compare algorithm IAR-D with other four algorithms in terms of computational time and speed-up ratio. The dynamic change of datasets is simulated in the following way. For each dataset in Table 8, five testing sets are constructed.

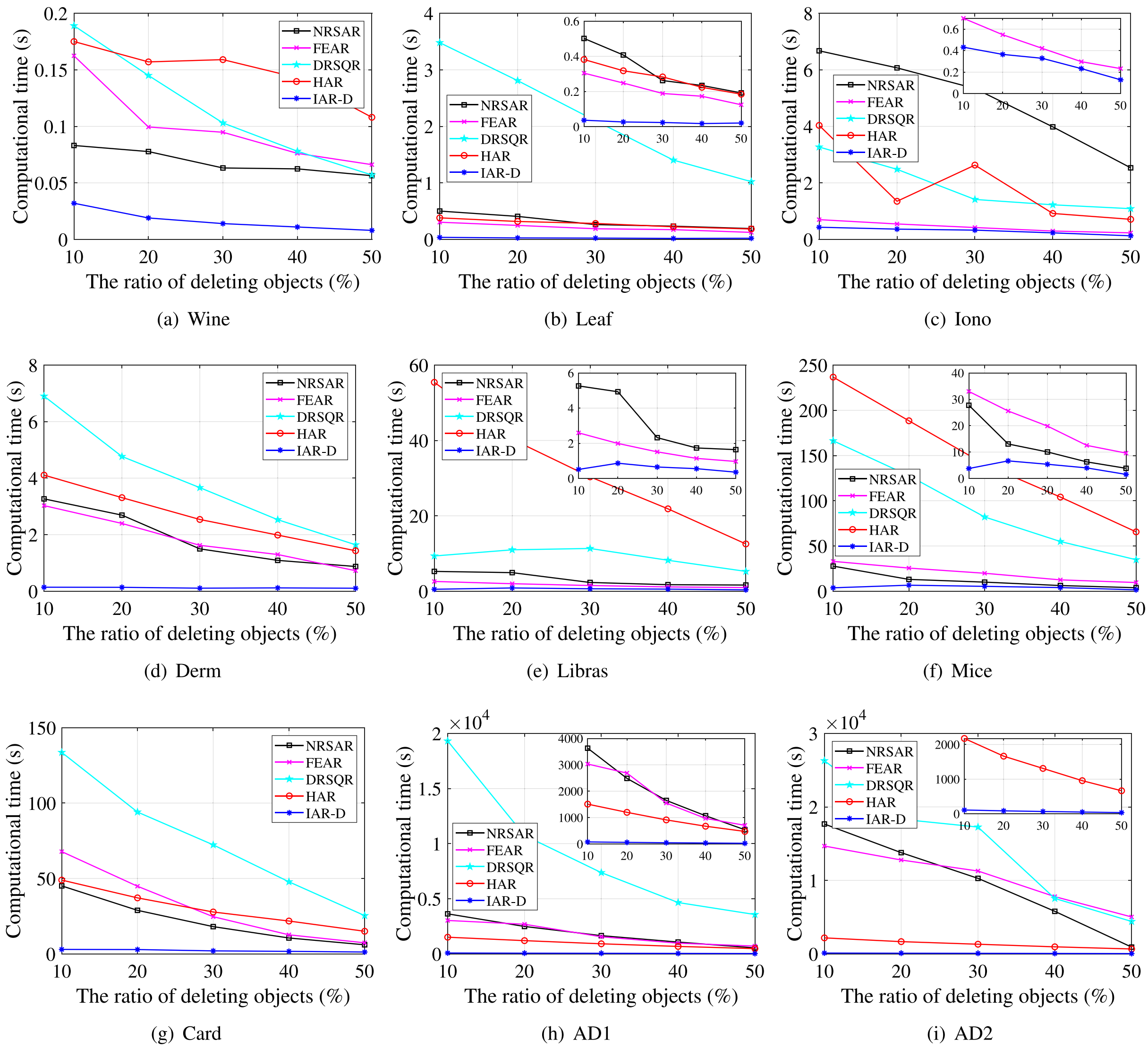


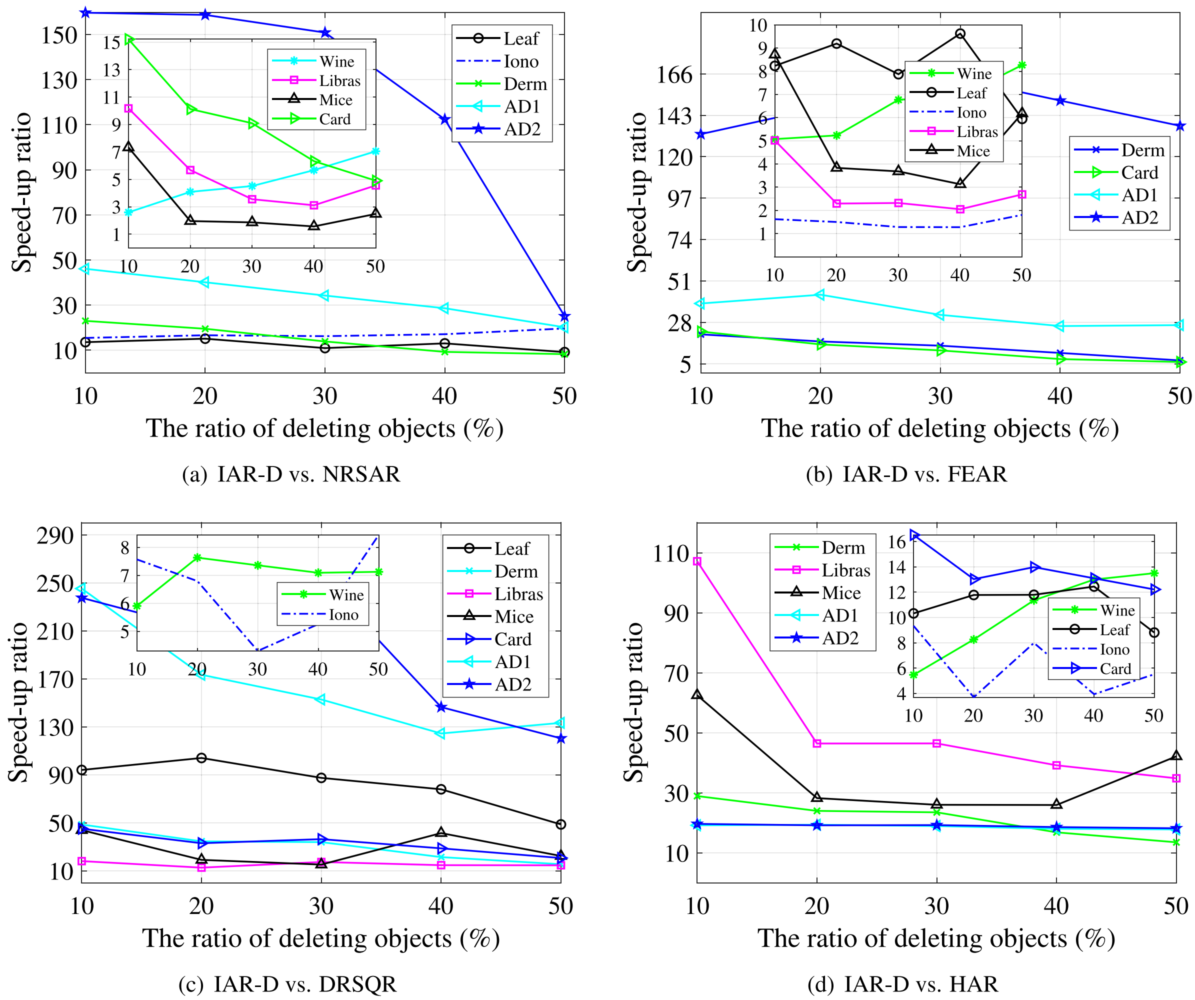
Fig. 3. The computational time of different algorithms versus different ratios of deleting objects.

Table 11
The comparison of classification accuracies of different algorithms on BayesNet and RandomTree (%)

Datasets	BayesNet						RandomTree					
	Raw	NRSAR	FEAR	DRAQR	HAR	IAR-D	Raw	NRSAR	FEAR	DRAQR	HAR	IAR-D
Wine	90.63	58.42 (1)	93.25 (9)	50.56 (1)	93.26 (9)	**93.26 (9)**	82.13	48.31 (1)	84.38 (9)	48.31 (1)	88.76 (9)	**88.76 (9)**
Leaf	**58.24**	60.00 (13)	57.64 (13)	52.94 (8)	50.59 (8)	50.59 (8)	**58.24**	52.35 (13)	50.00 (13)	46.47 (8)	52.35 (8)	52.35 (8)
Iono	**92.00**	76.00 (1)	90.85 (11)	88.57 (4)	86.86 (16)	86.29 (16)	84.57	76.00 (1)	80.42 (11)	84.14 (4)	83.43 (16)	**85.00 (16)**
Derm	90.21	65.36 (4)	96.08 (11)	96.08 (16)	96.09 (18)	**96.97 (18)**	81.62	63.68 (4)	88.82 (11)	81.06 (16)	87.15 (18)	**88.83 (18)**
Libras	30.44	36.66 (23)	30.00 (28)	14.44 (2)	35.00 (14)	**38.33 (19)**	**57.22**	51.66 (23)	47.77 (28)	27.77 (2)	47.78 (14)	49.44 (19)
Mice	**76.39**	53.90 (4)	74.72 (40)	71.00 (15)	65.80 (19)	70.63 (18)	70.56	69.70 (4)	70.67 (40)	70.79 (15)	69.15 (19)	**71.19 (18)**
Card	84.16	70.22 (1)	84.74 (13)	84.97 (12)	87.69 (13)	**87.92 (12)**	91.13	70.13 (1)	**91.79 (13)**	91.12 (12)	89.41 (13)	89.31 (12)
AD1	99.00	74.95 (1)	90.85 (16)	98.99 (21)	99.76 (23)	**99.76 (23)**	100	74.95 (1)	100 (16)	100 (21)	100 (23)	**100 (23)**
AD2	99.13	75.38 (1)	98.96 (16)	99.03 (22)	99.22 (23)	**99.22 (23)**	100	75.38 (1)	100 (16)	100 (22)	100 (23)	**100 (23)**
Average	80.02	63.43	79.68	72.95	79.36	**80.33**	**80.61**	64.68	79.32	72.18	79.78	80.54

We randomly delete objects from the raw dataset proportionally to obtain dynamic datasets for testing (i.e., 10%, 20%, 30%, 40%, and 50% of the raw dataset as the deleted object sets are randomly selected and then them are deleted successively), and then respectively record the running time of different algorithms on these datasets. The detailed change trend lines of different algorithms with different size of datasets are shown in Fig. 3.

Fig. 3 clearly shows that as the proportion of deleted object set increases, the running time of all algorithms decreases. It can be observed from each sub-figure that the computational time of algorithm IAR-D is significantly lower than that of other four algorithms. This indicates that algorithm IAR-D can obtain a reduct in a much shorter time. The main reason should be owed to the proposed algorithm IAR-D obtains a new reduct based on previous knowledge, which avoids some recalculation. Conversely, the other four algorithms are employed to retrain the changed dataset as a new one from scratch, which do not use the knowledge of generated from the original dataset. So they do a



(a) IAR-D vs. NRSAR

(b) IAR-D vs. FEAR

(c) IAR-D vs. DRSQR

(d) IAR-D vs. HAR

Fig. 4. The speed-up ratios that algorithm IAR-D relates to other different algorithms.

Table 12
The comparison of classification accuracies of different algorithms on OLM and OSDL (%)

Datasets	OLM						OSDL					
	Raw	NRSAR	FEAR	DRAQR	HAR	IAR-D	Raw	NRSAR	FEAR	DRAQR	HAR	IAR-D
Wine	32.58	33.71 (1)	33.71 (9)	29.43 (1)	33.71 (9)	**33.71 (9)**	50.56	53.93 (1)	57.30 (9)	53.93 (1)	59.55 (9)	**59.55 (9)**
Leaf	5.88	5.88 (13)	5.88 (13)	5.88 (8)	5.88 (8)	**5.88 (8)**	20.00	20.00 (13)	20.58 (13)	**27.46 (8)**	20.00 (8)	20.00 (8)
Iono	64.57	66.28 (1)	64.00 (11)	**68.00 (4)**	66.85 (16)	66.28 (16)	28	23.71 (1)	28.57 (11)	31.42 (4)	30.28 (16)	**32.00 (16)**
Derm	37.43	44.13 (4)	**59.77 (11)**	51.95 (16)	41.89 (18)	45.81 (18)	64.80	38.54 (4)	46.36 (11)	71.65 (16)	65.36 (18)	**73.74 (18)**
Libras	18.33	18.33 (23)	19.44 (28)	12.77 (2)	20.00 (14)	**20.44 (19)**	16.11	15.00 (23)	9.44 (28)	12.77 (2)	10.55 (14)	**17.11 (19)**
Mice	13.38	15.05 (4)	13.38 (40)	14.58 (15)	15.24 (19)	**17.65 (18)**	19.51	7.62 (4)	28.81 (40)	**35.02 (15)**	30.85 (19)	30.29 (18)
Card	72.99	67.55 (1)	70.09 (13)	72.33 (12)	74.04 (13)	**74.14 (12)**	77.19	70.22 (1)	81.20 (13)	84.44 (12)	84.63 (13)	**88.77 (12)**
AD1	94.31	63.95 (1)	87.66 (16)	94.12 (21)	94.31 (23)	**94.31 (23)**	80.01	36.04 (1)	75.62 (16)	86.07 (21)	86.07 (23)	**86.07 (23)**
AD2	94.75	64.58 (1)	88.28 (16)	94.68 (22)	94.75 (23)	**94.75 (23)**	82.23	35.41 (1)	78.16 (16)	87.27 (22)	87.27 (23)	**87.27 (23)**
Average	48.25	42.16	49.13	49.30	49.63	**50.33**	48.71	33.39	47.34	54.47	52.73	**54.98**

lot of repeated calculations. From the above analysis, we conclude that algorithm IAR-D is more efficient than other four algorithms.

Afterwards, the efficiency of algorithm IAR-D is verified again by calculating the speed-up ratio that algorithm IAR-D relates to other four algorithms. Similarly, the speed-up ratio of each dataset is calculated according to the results shown in Fig. 3, and the results are shown in Fig. 4.

From Fig. 4, we find that all speed-up ratios are greater than 1. This finding proves that algorithm IAR-D is faster than the other four algorithms for all datasets. Especially compared with algorithms DRSQR and HAR, for all datasets, algorithm IAR-D is at least four times faster than them. Furthermore, for large datasets AD1 and AD2, algorithm IAR-D is significantly dozens or even more than a hundred times faster than the other four algorithms. The experimental results again verify that algorithm IAR-D exhibits better efficiency than the other four algorithms.

5.2.3. Summary

We can draw the conclusion that the incremental algorithm IAR-D outperforms the other four algorithms by comparing their effectiveness and efficiency. The computational time required to obtain a feasible reduct by algorithm IAR-D is much less than that of the other four algorithms. Accordingly, a feasible reduct can be obtained more efficiently by using algorithm IAR-D when deleting multiple objects from an ODS.

6. Conclusion and future work

In this study, we investigate incremental attribute reduction approaches for dynamic ordered data in DRSA framework. First, we present the definitions of dominance relation matrix and dominance diagonal matrix in an OIS, and then propose a matrix-based computation method of dominance conditional entropy, i.e., MDCE, which is used as an uncertainty measure in attribute reduction algorithms. Second, the updating principles of MDCE are introduced when multiple objects vary. On this basis, we develop two incremental attribute reduction algorithms i.e., IAR-A and IAR-D. Final, the experiments are conducted to compare the effectiveness and efficiency of the proposed incremental algorithms with the other four algorithms. Experimental results show that the proposed incremental algorithms can efficiently calculate an effective reduct from dynamic ordered data.

The variation of an ODS may be many-sided. How to further use the incremental learning mechanism to complex dynamic data environment is an urgent problem to be solved. In complex monotonic classification tasks, objects are typically characterized by means of multimodality fuzzy attributes. Further extending DRSA method to this type of monotonic classification task is a meaningful work and is deserved to be studied. Concretely, the following our future research work has three aspects. (1) We will develop incremental approaches for attribute reduction in dynamic ordered data with attribute set and attribute value varying over time, respectively. (2) We will extend the proposed incremental approaches for attribute reduction to dominance-based fuzzy rough set model. (3) We will study DRSA model for multi-modality fuzzy monotonic classification tasks.

CRediT authorship contribution statement

Binbin Sang: Methodology, Validation, Writing - original draft, Writing - review & editing. **Hongmei Chen:** Conceptualization, Resources, Visualization, Supervision, Project administration, Funding acquisition. **Lei Yang:** Formal analysis, Data curation. **Dapeng Zhou:** Software, Data curation. **Tianrui Li:** Resources, Supervision, Funding acquisition. **Weihua Xu:** Resources.

Declaration of 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.

Acknowledgments

This work is supported by the National Natural Science Foundation of China (Nos. 61976182, 61572406, 61573292, 61602327, 61876157, 61976245), Key Program for International S&T Cooperation of Sichuan Province, China (No. 2019YFH0097), Applied Basic Research Programs of Science and Technology Department of Sichuan Province, China (2019YJ0084).

References

[1] Yaojin Lin, Qinghua Hu, Jinghua Liu, Jinjin Li, Xindong Wu, Streaming feature selection for multilabel learning based on fuzzy mutual information, IEEE Trans. Fuzzy Syst. 25 (6) (2017) 1491–1507.
[2] Qinghua Hu, Lingjun Zhang, Yucan Zhou, Witold Pedrycz, Large-scale multimodality attribute reduction with multi-kernel fuzzy rough sets, IEEE Trans. Fuzzy Syst. 26 (1) (2018) 226–238.
[3] Yaojin Lin, Yuwen Li, Chenxi Wang, Jinkun Chen, Attribute reduction for multi-label learning with fuzzy rough set, Knowl.-Based Syst. 152 (2018) 51–61.
[4] Anhui Tan, Weizhi Wu, Yuhua Qian, Jiye Liang, Jinkun Chen, Jinjin Li, Intuitionistic fuzzy rough set-based granular structures and attribute subset selection, IEEE Trans. Fuzzy Syst. 27 (3) (2019) 527–539.
[5] Degang Chen, Yanyan Yang, Ze Dong, An incremental algorithm for attribute reduction with variable precision rough sets, Appl. Soft Comput. 45 (2016) 129–149.
[6] Yunge Jing, Tianrui Li, Chuan Luo, Shijinn Horng, Guoyin Wang, Zeng Yu, An incremental approach for attribute reduction based on knowledge granularity, Knowl.-Based Syst. 104 (2016) 24–38.
[7] Yanyan Yang, Degang Chen, Wang Hui, Active sample selection based incremental algorithm for attribute reduction with rough sets, IEEE Trans. Fuzzy Syst. 25 (4) (2017) 825–838.
[8] Guangming Lang, Mingjie Cai, Hamido Fujita, Qimei Xiao, Related families-based attribute reduction of dynamic covering decision information systems, Knowl.-Based Syst. 162 (2018) 161–173.
[9] Wenhao Shu, Wenbin Qian, Yonghong Xie, Incremental approaches for feature selection from dynamic data with the variation of multiple objects, Knowl.-Based Syst. 163 (2019) 320–331.
[10] Y. LeCun, B. Boser, J.S. Denker, D. Henderson, R.E. Howard, W. Hubbard, L.D. Jackel, Backpropagation applied to handwritten zip code recognition, Neural Comput. 1 (4) (1989) 541–551.
[11] G.E. Hinton, S. Osindero, Y.W. Yee, A fast learning algorithm for deep belief nets, Neural Comput. 18 (7) (2006) 1527–1554.
[12] R.J. Williams, D. Zipser, A learning algorithm for continually running fully recurrent neural networks, Neural Comput. 1 (2) (1989) 270–280.
[13] Lei Zhao, Qing Hua Hu, Wen Wu Wang, Heterogeneous feature selection with multi-modal deep neural networks and sparse group lasso, IEEE Trans. Multimed. 17 (11) (2015) 1936–1948.
[14] Vijay Bhaskar Semwal, Kaushik Mondal, Gora Chand Nandi, Robust and accurate feature selection for humanoid push recovery and classification: deep learning approach, Neural Comput. Appl. 28 (3) (2017) 565–574.
[15] Jingxia Chen, Zijing Mao, Ru Zheng, Yufei Huang, Lifeng He, Feature selection of deep learning models for EEG-based RSVP target detection, IEICE Trans. Inf. Syst. 102-D (4) (2019) 836–844.
[16] Tong Niu, Jianzhou Wang, Haiyan Lu, Wendong Yang, Pei Du, Developing a deep learning framework with two-stage feature selection for multivariate financial time series forecasting, Expert Syst. Appl. 148 (2020) 1–17.
[17] Qin Zou, Lihao Ni, Tong Zhang, Qian Wang, Deep learning based feature selection for remote sensing scene classification, IEEE Geosci. Remote Sens. Lett. 12 (11) (2015) 2321–2325.
[18] Hongtao Shi, Hongping Li, Dan Zhang, Chaqiu Cheng, Xuanxuan Cao, An efficient feature generation approach based on deep learning and feature selection techniques for traffic classification, Comput. Netw. 132 (2018) 81–98.
[19] Weiping Ding, Chinteng Lin, Witold Pedrycz, Multiple relevant feature ensemble selection based on multilayer co-evolutionary consensus mapreduce, IEEE Trans. Cybern. 50 (2) (2020) 425–439.
[20] Kaustuv Nag, Nikhil R. Pal, Feature extraction and selection for parsimonious classifiers with multiobjective genetic programming, IEEE Trans. Evol. Comput. 24 (3) (2020) 454–466.
[21] Mahdieh Labani, Parham Moradi, Mahdi Jalili, A multi-objective genetic algorithm for text feature selection using the relative discriminative criterion, Expert Syst. Appl. 149 (2020) 1–21.
[22] Jianbin Ma, Xiaoying Gao, A filter-based feature construction and feature selection approach for classification using genetic programming, Knowl.-Based Syst. 196 (2020) 1–14.
[23] Asit K. Das, Sunanda Das, Arka Ghosh, Ensemble feature selection using bi-objective genetic algorithm, Knowl.-Based Syst. 123 (2017) 116–127.
[24] Anda Li, Bing Xue, Mengjie Zhang, Multi-objective feature selection using hybridization of a genetic algorithm and direct multisearch for key quality characteristic selection, Inform. Sci. 523 (2020) 245–265.
[25] Lin Sun, Xiaoyu Zhang, Yuhua Qian, Jiucheng Xu, Shiguang Zhang, Feature selection using neighborhood entropy-based uncertainty measures for gene expression data classification, Inform. Sci. 502 (2019) 18–41.
[26] Jianhua Dai, Qinghua Hu, Hu Hu, Debiao Huang, Neighbor inconsistent pair selection for attribute reduction by rough set approach, IEEE Trans. Fuzzy Syst. 26 (2) (2018) 937–950.
[27] Changzhong Wang, Yan Wang, Mingwen Shao, Yuhua Qian, Degang Chen, Fuzzy rough attribute reduction for categorical data, IEEE Trans. Fuzzy Syst. 28 (5) (2020) 818–830.
[28] Zdzislaw Pawlak, Rough sets, Int. J. Comput. Inf. Sci. 11 (5) (1982) 341–356.
[29] Greco, Matarazzo, Slowinski, Rough approximation of a preference relation by dominance relations, European J. Oper. Res. 117 (1) (1999) 63–83.
[30] Salvatore Greco, Benedetto Matarazzo, Roman Slowinski, Rough sets theory for multicriteria decision analysis, European J. Oper. Res. 129 (1) (2001) 1–47.
[31] Jerzy Blaszczyński, Salvatore Greco, Roman Słowiński, Marcin Szelg, Monotonic variable consistency rough set approaches, Internat. J. Approx. Reason. 50 (7) (2009) 979–999.
[32] Wojciech Kotlowski, Krzysztof Dembczyński, Salvatore Greco, Roman Słowiński, Stochastic dominance-based rough set model for ordinal classification, Inform. Sci. 178 (21) (2008) 4019–4037.

[33] Abbas Ali, Muhammad Irfan Ali, Noor Rehman, Soft dominance based rough sets with applications in information systems, Internat. J. Approx. Reason. 113 (2019) 171–195.
[34] Xiaoxia Zhang, Degang Chen, E.C.C. Tsang, Generalized dominance rough set models for the dominance intuitionistic fuzzy information systems, Inform. Sci. 378 (2017) 1–25.
[35] Hongying Zhang, Yee Leung, Lei Zhou, Variable-precision-dominance-based rough set approach to interval-valued information systems, Inform. Sci. 244 (2013) 75–91.
[36] Wensheng Du, Baoqing Hu, Dominance-based rough set approach to incomplete ordered information systems, Inform. Sci. 346 (2016) 106–129.
[37] Xibei Yang, Yong Qi, Dongjun Yu, Hualong Yu, Jingyu Yang, α-DOminance relation and rough sets in interval-valued information systems, Inform. Sci. 294 (2015) 334–347.
[38] Bing Huang, Huaxiong Li, Dakuan Wei, Dominance-based rough set model in intuitionistic fuzzy information systems, Knowl.-Based Syst. 28 (2012) 115–123.
[39] Wenbin Qian, Wenhao Shu, Attribute reduction in incomplete ordered information systems with fuzzy decision, Appl. Soft Comput. 73 (2018) 242–253.
[40] Wensheng Du, Baoqing Hu, A fast heuristic attribute reduction approach to ordered decision systems, European J. Oper. Res. 264 (2) (2018) 440–452.
[41] Weihua Xu, Yuan Li, Xiuwu Liao, Approaches to attribute reductions based on rough set and matrix computation in inconsistent ordered information systems, Knowl.-Based Syst. 27 (2012) 78–91.
[42] Hongying Zhang, Shuyun Yang, Feature selection and approximate reasoning of large-scale set-valued decision tables based on α-dominance-based quantitative rough sets, Inform. Sci. 378 (2017) 328–347.
[43] Robert Susmaga, Reducts and constructs in classic and dominance-based rough sets approach, Inform. Sci. 271 (2014) 45–64.
[44] Jianhang Yu, Minghao Chen, Weihua Xu, Dynamic computing rough approximations approach to time-evolving information granule interval-valued ordered information system, Appl. Soft Comput. 60 (2017) 18–29.
[45] Xin Yang, Tianrui Li, Dun Liu, Hamido Fujita, A multilevel neighborhood sequential decision approach of three-way granular computing, Inform. Sci. 538 (2020) 119–141.
[46] Yuanjian Zhang, Duoqian Miao, Witold Pedrycz, Tianna Zhao, Jianfeng Xu, Ying Yu, Granular structure-based incremental updating for multi-label classification, Knowl.-Based Syst. 189 (2020) 1–15.
[47] Jiye Liang, Feng Wang, Chuangyin Dang, Yuhua Qian, A group incremental approach to feature selection applying rough set technique, IEEE Trans. Knowl. Data Eng. 26 (2014) 294–308.
[48] Xiao Zhang, Changlin Mei, Degang Chen, Yanyan Yang, Jinhai Li, Active incremental feature selection using a fuzzy rough set-based information entropy, IEEE Trans. Fuzzy Syst. 28 (5) (2020) 901–915.
[49] Yanyan Yang, Degang Chen, Wang Hui, Xizhao Wang, Incremental perspective for feature selection based on fuzzy rough sets, IEEE Trans. Fuzzy Syst. 26 (3) (2018) 1257–1273.
[50] Yanyan Yang, Shiji Song, Degang Chen, Xiao Zhang, Discernible neighborhood counting based incremental feature selection for heterogeneous data, Int. J. Mach. Learn. Cybern. 11 (5) (2020) 1115–1127.
[51] Wenhao Shu, Wenbin Qian, Yonghong Xie, Incremental feature selection for dynamic hybrid data using neighborhood rough set, Knowl.-Based Syst. 194 (2020) 1–15.
[52] Ye Liu, Lidi Zheng, Yeliang Xiu, Hong Yin, Suyun Zhao, Xizhao Wang, Hong Chen, Cuiping Li, Discernibility matrix based incremental feature selection on fused decision tables, Internat. J. Approx. Reason. 118 (2020) 1–26.
[53] Asit K. Das, Shampa Sengupta, Siddhartha Bhattacharyya, A group incremental feature selection for classification using rough set theory based genetic algorithm, Appl. Soft Comput. 65 (2018) 400–411.
[54] Degang Chen, Lianjie Dong, Jisheng Mi, Incremental mechanism of attribute reduction based on discernible relations for dynamically increasing attribute, Soft Comput. 24 (1) (2020) 321–332.
[55] Feng Wang, Jiye Liang, Yuhua Qian, Attribute reduction: A dimension incremental strategy, Knowl.-Based Syst. 39 (2013) 95–108.
[56] Guangming Lang, Duoqian Miao, Mingjie Cai, Zhifei Zhang, Incremental approaches for updating reducts in dynamic covering information systems, Knowl.-Based Syst. 134 (2017) 85–104.
[57] Anping Zeng, Tianrui Li, Dun Liu, Junbo Zhang, Hongmei Chen, A fuzzy rough set approach for incremental feature selection on hybrid information systems, Fuzzy Sets and Systems 258 (2015) 39–60.
[58] Feng Wang, Jiye Liang, Chuangyin Dang, Attribute reduction for dynamic data sets, Appl. Soft Comput. 13 (1) (2013) 676–689.
[59] Wei Wei, Xiaoying Wu, Jiye Liang, Junbiao Cui, Yijun Sun, Discernibility matrix based incremental attribute reduction for dynamic data, Knowl.-Based Syst. 140 (2018) 142–157.
[60] Wei Wei, Peng Song, Jiye Liang, Xiaoying Wu, Accelerating incremental attribute reduction algorithm by compacting a decision table, Int. J. Mach. Learn. Cybern. 10 (9) (2019) 2355–2373.
[61] Mingjie Cai, Guangming Lang, Hamido Fujita, Zhenyu Li, Tian Yang, Incremental approaches to updating reducts under dynamic covering granularity, Knowl.-Based Syst. 172 (2019) 130–140.
[62] Lianjie Dong, Degang Chen, Incremental attribute reduction with rough set for dynamic datasets with simultaneously increasing samples and attributes, Int. J. Mach. Learn. Cybern. 11 (6) (2020) 1339–1355.
[63] Yunge Jing, Tianrui Li, Hamido Fujita, Baoli Wang, Ni Cheng, An incremental attribute reduction method for dynamic data mining, Inform. Sci. 465 (2018) 202–218.
[64] Claude Shannon, Warren Weaver, The mathematical theory of communication, Bell Syst. Tech. J. 27 (3/4) (1948) 373–423.
[65] Qinghua Hu, Xunjian Che, Lei Zhang, David Zhang, Maozu Guo, Daren Yu, Rank entropy based decision trees for monotonic classification, IEEE Trans. Knowl. Data Eng. 24 (11) (2012) 2052–2064.
[66] Yanyong Huang, Tianrui Li, Chuan Luo, Hamido Fujita, Shijinn Horng, Matrix-based dynamic updating rough fuzzy approximations for data mining, Knowl.-Based Syst. 119 (2017) 273–283.
[67] Chuan Luo, Tianrui Li, Zhang Yi, Hamido Fujita, Matrix approach to decision-theoretic rough sets for evolving data, Knowl.-Based Syst. 99 (2016) 123–134.
[68] Shu Wang, Tianrui Li, Chuan Luo, Hamido Fujita, Efficient updating rough approximations with multi-dimensional variation of ordered data, Inform. Sci. 372 (2016) 690–708.
[69] Shu Wang, Tianrui Li, Chuan Luo, Hongmei Chen, Hamido Fujita, Domain-wise approaches for updating approximations with multi-dimensional variation of ordered information systems, Inform. Sci. 478 (2019) 100–124.
[70] Shaoyong Li, Tianrui Li, Dun Liu, Incremental updating approximations in dominance-based rough sets approach under the variation of the attribute set, Knowl.-Based Syst. 40 (2013) 17–26.
[71] Shaoyong Li, Tianrui Li, Incremental update of approximations in dominance-based rough sets approach under the variation of attribute values, Inform. Sci. 294 (2015) 348–361.
[72] Wensheng Du, Baoqing Hu, Attribute reduction in ordered decision tables via evidence theory, Inform. Sci. 364 (2016) 91–110.
[73] Qinghua Hu, Daren Yu, Zongxia Xie, Information-preserving hybrid data reduction based on fuzzy-rough techniques, Pattern Recognit. Lett. 27 (5) (2006) 414–423.
[74] Qinghua Hu, Daren Yu, Zongxia Xie, Numerical attribute reduction based on neighborhood granulation and rough approximation, J. Softw. 19 (3) (2008) 640–649 (in Chinese).