Information Sciences 681 (2024) 121219

Contents lists available at ScienceDirect

Information Sciences

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

Matrix-based incremental feature selection method using weight-partitioned multigranulation rough set

Weihua Xu *, Qinyuan Bu

College of Artificial Intelligence, Southwest University, Chongqing, 400715, PR China

ARTICLE INFO

Keywords:
Incremental feature selection
Matrix dynamic updating
Generalized multigranulation rough set
Conditional entropy

ABSTRACT

Incremental feature selection methods have gained increasing research attention as they improve the efficiency of feature selection for dynamic datasets. Multigranulation rough set, as an extension of rough set theory, allows for a comprehensive and rational analysis of problems from multiple hierarchical and granular perspectives. However, existing research on granularity partitioning relies on the decision maker's subjective experience, which lacks convincing power. In this paper, we propose a generalized multigranulation neighborhood rough set based on weight partition model, using a matrix form. We discuss several properties and define a new entropy measure to evaluate feature importance. A heuristic feature selection algorithm is developed based on this entropy to search for the optimal subset. Furthermore, we discuss dynamic updating mechanism and design two incremental feature selection algorithms. Finally, we conduct experiments on 12 public datasets to evaluate the performance of the proposed algorithms and validate their effectiveness and efficiency in feature selection for both static and dynamic datasets.

1. Introduction

Feature selection has played a pivotal and increasingly prominent role in the data preprocessing stage, witnessing widespread adoption across various domains such as data mining, image processing, and natural language processing. Its significance has grown significantly, enabling the transformation of raw data into distinct and meaningful features that capture essential information and facilitate the identification of different categories or the extraction of highly informative characteristics. By enhancing model performance and improving generalization capabilities, effective feature selection not only reduces data dimensionality but also mitigates redundancy, addresses overfitting, and enhances the accuracy of classification or regression tasks. Moreover, it facilitates the discovery of latent patterns and structures hidden within the data, thereby contributing to reducing model complexity and improving interpretability [1–3].

In feature selection, information entropy (IE) is used to measure the information content and uncertainty of features, playing an important role. The notion of IE, originally introduced by C.E. Shannon, serves as a measure of uncertainty in information and finds extensive applications in domains such as information theory, machine learning, and data compression. Xu et al. formulated the fuzzy dominance conditional entropy within the context of multi-source interval-valued systems, establishing a corresponding fusion model [4]. They devised a dynamic incremental heuristic algorithm to experimentally validate the efficacy and efficiency of the proposed model. Furthermore, Deng et al. devised a novel approach termed Label Learning Method (LDL), which incorporates

* Corresponding author.
E-mail addresses: chxuwh@gmail.com (W. Xu), bqy990515@163.com (Q. Bu).

https://doi.org/10.1016/j.ins.2024.121219
Received 23 January 2024; Received in revised form 20 May 2024; Accepted 18 July 2024

Available online 23 July 2024

bilateral similarity and constructs a novel form of fuzzy entropy [5]. They employed NRS to handle LDL and substantiated its feasibility through subsequent experimental assessments. Jiang et al. introduced a novel technique based on IE within the framework of NRS to tackle the issue of outlier detection in the field of data mining [6]. This approach offers a solution for identifying and handling abnormal data instances. In a similar vein, Wang et al. put forward a distance-based fuzzy rough entropy and utilized this entropy to construct a model for anomaly detection [7]. Additionally, they devised specific algorithms tailored to support the execution of the anomaly detection process.

Granular computing (GrC) [8,9] is an approach that addresses problems from multiple perspectives and multiple levels, and it has become a rapidly developing discipline widely applied in fields such as intelligent computing and machine learning. As a new concept and computational paradigm in information processing, it integrates theories and methods from various domains, such as rough set theory and concept cognitive learning (CCL), forming a vast knowledge system and coherent methodology. Zhang et al. introduced bidirectional concept cognitive learning in multi-granular decision tables based on granular computing, utilizing entropy to fuse information from different sources and designing three concept learning algorithms to apply the theory to practical problems [10]. Xu et al. established a new cognitive mechanism by analyzing the relationship of bidirectional concepts (2WL) and introduced the motion three-way decision into 2WL, constructing an enhanced version of 2WL called TCCL [11]. They validated the effectiveness and generalization ability of TCCL through experiments. Guo et al. combined the memory concept cognitive learning with CCL by introducing a recall-forgetting mechanism to discover knowledge. Experimental verification demonstrated that this method enhances the effectiveness of concept learning and reduces cognitive complexity [12]. Wang et al. fused multiple types of kernel functions, constructed the Variable Precision Multigranulation Kernel Rough Set, and proposed a new three-way decision method. Finally, the applicability of the model was validated through simulation analysis [13].

Rough set theory (RST), initially proposed by Pawlak, is a mathematical tool used to handle uncertain, incomplete, and fuzzy data. In recent years, with the advancement of RST, it has been employed to address various problems. Due to its ability to process data without relying on any prior knowledge, RST has been widely applied in feature selection. In situations where the relationships between attributes are unknown, RST can partition objects in the data into different equivalence classes using equivalence relations, enabling effective analysis and processing of data validity in such scenarios. However, real-life data is diverse, and the classical RST approach often leads to information loss, increased computational complexity, and limited efficacy. To overcome these limitations, scholars have delved into researching RST, introducing neighborhood relations, dominance relations, and other alternatives to equivalence relations. This has led to the development of numerous enhanced models, such as neighborhood rough set (NRS), variable precision rough set (VPRS), and multigranulation rough set (MGRS) [14–16]. These models address the shortcomings of classical RST and often exhibit higher efficiency and simplicity in data processing. This paper primarily focuses on an in-depth investigation of the neighborhood rough set and multigranulation rough set models. Classical RST models and their extensions are built on the foundation of single-granularity models. However, in real-life scenarios, single-granularity models exhibit significant limitations when addressing certain problems. Dividing data into the smallest units fails to capture relationships between data, lacking flexibility and generality. For instance, performing intersection operations on two mutually independent decisions is clearly impractical. Consequently, scholars have begun researching Multigranulation rough set models. Qian et al. first proposed the Multigranulation Rough Set model (MGRS) [17] and provided new metrics based on this model, such as approximate precision and approximate quality. They also applied the theory to practical problems through algorithm designs. To enhance the efficiency and problem-solving capabilities of MGRS, scholars have proposed various extensions of MGRS through extensive research. Traditional MGRS imposes either too strict or too loose requirements when approximating, lacking restrictive conditions. Therefore, Xu et al. introduced the Generalized Multigranulation Rough Set model (GMGRS) [18], defined the supporting feature function, introduced the parameter β for examination, and discussed new properties. Zhang et al. employed a multigranulation fusion strategy to acquire information from multiple source systems, constructed a dynamic updating mechanism, and verified the effectiveness of fusion operators through algorithm comparisons [19].

In real life, data frequently undergoes dynamic alterations. Data systems consistently adjust data to maintain information accuracy. Classical RST assumes dataset static during feature selection. However, when data changes, recalculating the entire dataset substantially prolongs runtime. To address these issues, Scholars have introduced extended models to efficiently manage dynamic datasets. Sang et al. addressed dynamic data by proposing the fuzzy dominance neighborhood rough set model, defining a new entropy as an evaluation criterion, and designing two heuristic algorithms for feature subset selection [20]. Yang et al. used sample selection to eliminate useless samples from the newly added data and designed a feature-based accelerator to incrementally select the best features, achieving feature selection [21]. Pan et al., based on the dominance neighborhood rough set, considered the importance of each feature and assigned corresponding weights to each feature. They proposed a matrix-based entropy and designed algorithms to select excellent features [22]. Sang et al. explored the matrix-based dominance conditional entropy and the updating rules for the dominance relation matrix and dominance diagonal matrix when objects vary, and developed two incremental feature selection algorithms [23]. Huang et al. proposed an incremental feature selection method using fuzzy rough set theory in hierarchical classification. They analyzed the incremental updates when new samples are added and designed two incremental algorithms based on existing non-incremental methods [24].

When addressing problems, it is usually necessary to think and solve them from multiple angles. Since classical RST can only consider problems from a single perspective, it has significant limitations in problem-solving. While GMGRS can overcome the aforementioned issues, it necessitates recalculating the dataset whenever the sample size changes, leading to significant time consumption. Inspired by the GMGRS, this study aims to investigate the updating issue when samples change and to use a novel model for feature selection. The key innovations and contributions of our research are as follows:

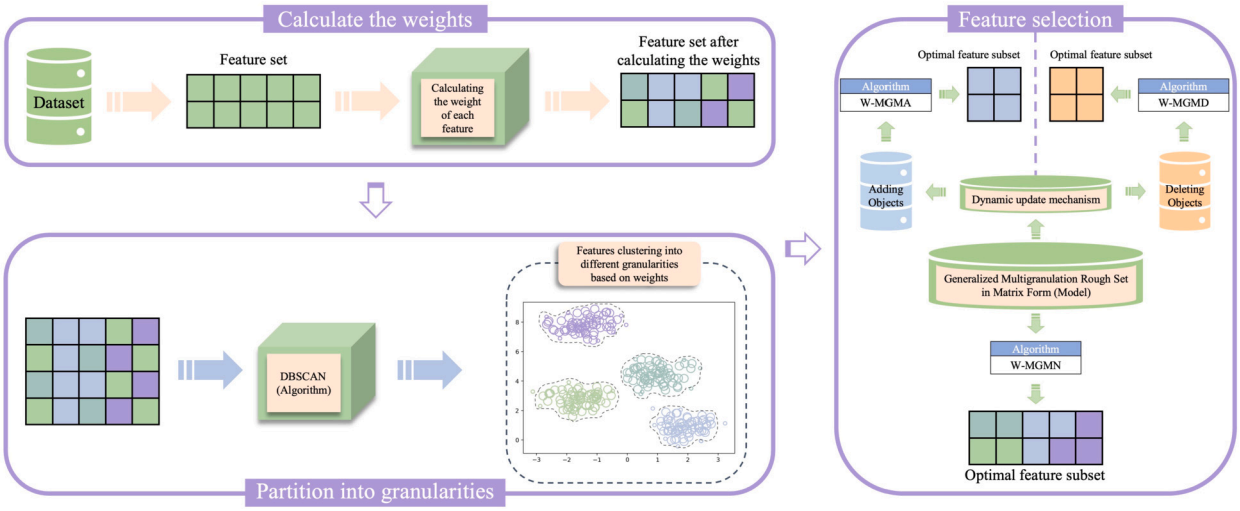

Fig. 1. The framework of this paper.

1. A generalized multi-granulation neighborhood rough set model based on weighted partitioning is proposed (W-GMNRS). This model assigns corresponding weights to each knowledge granule (feature). By considering the differences in weight values, it groups knowledge granules within the same weight threshold into a single granulation. This makes the granulation allocation more reasonable and provides a foundation for subsequent feature selection.
2. Within the W-GMNRS framework, a matrix-based conditional entropy (W-MGME) is introduced, its properties are outlined, and a feature selection algorithm leveraging W-MGME is developed.
3. We analyzed the dynamic updating mechanisms when objects change and designed two incremental feature selection algorithms to accelerate computation speed when dataset objects are altered.
4. Experimental evaluations conducted on a set of 12 publicly available datasets substantiate the effectiveness and classification efficiency of W-GMNRS. Additionally, the proposed incremental algorithm demonstrates significant reductions in time and resource consumption as the number of objects increases, while simultaneously achieving enhanced classification accuracy.

The remaining chapters of this paper are structured as follows: Chapter 2 provides an introduction to pertinent concepts and prior works. Building upon this foundation, Chapter 3 presents a methodology for computing the weights of knowledge granules and establishes the W-GMNRS model, while also discussing its associated properties. Subsequently, we design the conditional entropy based on this model, which employs matrix computations. Chapter 4 demonstrates the dynamic updating mechanism in response to variations in the number of objects, accompanied by a comprehensive analysis of the update process. Then, we devise two dynamic updating algorithms and apply them to feature selection. In Chapter 5, experimental results conducted on 12 publicly available datasets illustrate the efficacy and robustness of the proposed approach for feature selection. Finally, conclusions are drawn, and future research prospects are discussed in Chapter 6. The flowchart of this article is shown in Fig. 1.

2. Preliminaries

This Chapter provides an overview of some fundamental concepts related to classical RST, NRS, classical MGRS, and generalized multigranulation rough set (GMGRS) [25,26,17,18].

2.1. Rough set

Let $\mathcal{IS} = (U, N)$ be an information system, where $U = \{x_1, x_2, \cdots, x_n\}$ signifies a set of objects, and $N = A \cup D$ denotes the attribute set, in which $A = \{b_1, b_2, \cdots, b_s\}$ is the conditional attribute set and $D = \{d\}$ is the decision attribute set, and additionally, $A \cap D = \emptyset$. It is worth emphasizing that U and N both consist of finite non-empty sets. The equivalence relation is called R, and $[x]_R$ is the equivalence classes of object x with respect to R. For any target concept $X \subseteq U$, the upper approximation and the lower approximation are defined by

$$\begin{aligned} \underline{R}(X) &= \left\{ x \in U \;\middle|\; [x]_R \subseteq X \right\}, \\ \overline{R}(X) &= \left\{ x \in U \;\middle|\; [x]_R \cap X \neq \emptyset \right\}. \end{aligned} \tag{1}$$

Among them, X is a definable set when $\underline{R}(X) = \overline{R}(X)$. On the contrary, X is called a rough set.

2.2. Neighborhood rough set

In an $\mathcal{IS} = (U, A \cup \{d\})$, for $\forall x \in U$, $\forall b \in A$, the $f(x, b)$ denotes the attribute value of x on attribute b. Then $\mathcal{IS}$ is designated as a neighborhood information system (NIS) if and only if the values of A are numerical, and we called $\mathcal{NIS}$ in this paper. In $\mathcal{NIS}$, we opt to utilize the following distance function to characterize the distance between x_i and x_j within $B \subseteq A$:

$$\Delta_B(x_i, x_j) = \left(\sum_{b \in B} \left| f(x_i, b) - f(x_j, b) \right|^s \right)^{\frac{1}{s}}. \tag{2}$$

In this paper, we opt for the Euclidean distance for distance computation.

Given a neighborhood radius, δ, the neighborhood classes of $x \in U$ under neighborhood relation R_δ^B is defined by

$$[x]_{R_\delta^B} = \left\{ x_k \;\middle|\; \Delta_B(x, x_k) \le \delta, x_k \in U \right\}. \tag{3}$$

Given a $\mathcal{NIS} = (U, A \cup \{d\})$, the neighborhood classes $[x]_{R_\delta^B}$, for $\forall X \subseteq U$, $\forall B \subseteq A$, the definitions of upper approximation $\overline{R_\delta^B}(X)$ and lower approximation $\underline{R_\delta^B}(X)$ are

$$\begin{aligned} \underline{R_\delta^B}(X) &= \left\{ x \in U \;\middle|\; [x]_{R_\delta^B} \subseteq X \right\}, \\ \overline{R_\delta^B}(X) &= \left\{ x \in U \;\middle|\; [x]_{R_\delta^B} \cap X \neq \emptyset \right\}. \end{aligned} \tag{4}$$

In the realm of feature selection, NRS offers the capability to directly select attributes from numerical decision tables, obviating the necessity for the discretization process of numerical data while ensuring classification performance. Consequently, the utilization of neighborhood NRS is employed for data processing in this study.

2.3. Multigranulation rough set

In practical scenarios, domains are often partitioned by multiple relationships rather than a single one. The classical RST is insufficient to address such situations. To capture problem descriptions from multiple perspectives, Xu et al. introduced optimistic and pessimistic multigranulation rough sets [27].

Given a $\mathcal{NIS} = (U, A \cup \{d\})$, neighborhood class $[x]_{R_\delta^B}$, for $\forall X \subseteq U$, $\forall N_i \subseteq A (i = 1, 2, \cdots, m \le 2^{|A|})$, the definitions of the upper approximation and lower approximation of X under relation R_δ^B in optimistic multigranulation are as following:

$$\begin{aligned} \underline{\mathcal{OM}_\delta^{\sum_{i=1}^m N_i}}(X) &= \left\{ x \in U \;\middle|\; \bigvee_{i=1}^m \left([x]_{R_\delta^B} \subseteq X \right) \right\}, \\ \overline{\mathcal{OM}_\delta^{\sum_{i=1}^m N_i}}(X) &= \left\{ x \in U \;\middle|\; \bigwedge_{i=1}^m \left([x]_{R_\delta^B} \cap X \neq \emptyset \right) \right\}, \end{aligned} \tag{5}$$

and the upper approximation and lower approximation in pessimistic multigranulation are defined as

$$\begin{aligned} \underline{\mathcal{PM}_\delta^{\sum_{i=1}^m N_i}}(X) &= \left\{ x \in U \;\middle|\; \bigwedge_{i=1}^m \left([x]_{R_\delta^B} \subseteq X \right) \right\}, \\ \overline{\mathcal{PM}_\delta^{\sum_{i=1}^m N_i}}(X) &= \left\{ x \in U \;\middle|\; \bigvee_{i=1}^m \left([x]_{R_\delta^B} \cap X \neq \emptyset \right) \right\}. \end{aligned} \tag{6}$$

In these definitions, the symbols "$\bigwedge$" and "$\bigvee$" are used to represent "*and*" and "*or*" respectively. Furthermore, when $\underline{\mathcal{OM}_\delta^{\sum_{i=1}^m N_i}}(X) = \overline{\mathcal{OM}_\delta^{\sum_{i=1}^m N_i}}(X)$, X is considered optimistic and precise, the same applies to pessimistic multigranulation when $\underline{\mathcal{PM}_\delta^{\sum_{i=1}^m N_i}}(X) = \overline{\mathcal{PM}_\delta^{\sum_{i=1}^m N_i}}(X)$. On the contrary, X is rough.

2.4. Generalized multigranulation rough set

Among the aforementioned two types of MGRS, the optimistic MGRS is overly lenient in depict approximations, while the pessimistic MGRS is excessively stringent. Both approaches overlook the commonly encountered principle of minority following the majority in real-life situations. To overcome these limitations, Xu et al. proposed the generalized multigranulation rough set (GMGRS) [18]. In this paper, we extend the existing model by incorporating neighborhood relation, leading to the development of generalized multigranulation neighborhood rough set (GMNRS). To expound upon this model, a feature support function is initially defined.

Given a $\mathcal{NIS} = (U, A \cup \{d\})$, for $\forall X \subseteq U$, $\forall N_i \subseteq A (i = 1, 2, \cdots, m \leq 2^{|A|})$, the feature support function $S_X^{\delta, N_i}(x)$ is defined to describe the relationship of inclusion regarding the $[x]_{R_\delta^{N_i}}$ and X as:

$$S_X^{\delta, N_i}(x) = \begin{cases} 1, & [x]_{R_\delta^{N_i}} \subseteq X \\ 0, & others \end{cases}. \tag{7}$$

Through the scrutiny of the provided feature support function, it is discerned that in the case of optimistic MGRS, an approximation can solely be achieved when $S_X^{\delta, N_i}(x) = 1$. This criterion is excessively rigorous and may introduce superfluous information. Likewise, the conditions of pessimistic MGRS are too lenient to accurately characterize concepts. Consequently, GMGRS employs the parameter β to govern the approximation conditions. A smaller value of β signifies a more relaxed requirement, whereas a larger value of β implies a more stringent condition.

Given a $\mathcal{NIS} = (U, A \cup \{d\})$, for the feature support function $S_X^{\delta, N_i}(x)$, $\forall X \subseteq U$, $\forall N_i \subseteq A (i = 1, 2, \cdots, m \leq 2^{|A|})$, $\beta \in (0.5, 1]$, the definitions of upper approximation and lower approximation in GMGRS are as follows:

$$\begin{aligned} \underline{\mathcal{GM}_\delta}(X) &= \left\{ x \in U \middle| \frac{\sum_{i=1}^{m} S_X^{\delta, N_i}(x)}{m} \geq \beta \right\}, \\ \overline{\mathcal{GM}_\delta}(X) &= \left\{ x \in U \middle| \frac{\sum_{i=1}^{m} \left(1 - S_X^{\delta, N_i}(x)\right)}{m} > 1 - \beta \right\}. \end{aligned} \tag{8}$$

$< \overline{\mathcal{GM}_\delta}(X), \underline{\mathcal{GM}_\delta}(X) >$ is called GMGRS, β is referred to as the information level. When $\underline{\mathcal{GM}_\delta}(X) \neq \overline{\mathcal{GM}_\delta}(X)$, X is a rough set. It is worth noting that there is no explicitly fixed inclusion relationship between $\overline{\mathcal{GM}_\delta}(X)$ and $\overline{R_\delta^B}(X)$, and the same applies to $\underline{\mathcal{GM}_\delta}(X)$ and $\underline{R_\delta^B}(X)$ as well.

3. Matrix-based generalized multigranulation neighborhood rough set based on weighted partition

In this chapter, we will initially outline the methodology of assign weights to knowledge granules and assembling them into diverse granularities. Subsequently, we will proceed with the expansion of this representation using a matrix-based approach.

3.1. Generation of weights and knowledge granule partitioning

In real-world scenarios, each knowledge granule possesses a distinct level of importance. Existing MGRS approaches solely rely on granule quantity to determine granularity, disregarding the quality of knowledge granules. Knowledge granules of varying importance correspond to distinct weight values. It is erroneous to equate the impact of multiple low-weight knowledge granules with that of a single high-weight knowledge granule in decision-making. Consequently, based on the assigned weights of individual knowledge granules, we cluster them into diverse granularities, ensuring a more reasonable and practical granularity allocation process that enhances the applicability of the decision-making process.

Definition 1. Given a $\mathcal{NIS} = (U, A \cup \{d\})$, where $A = (b_1, b_2, \cdots, b_s)$. For $\forall b \in A$, $\forall x \in U$, $g(x, b)$ signifies the value of x pertaining to b. Let the coefficient matrix of A be

$$\mathcal{M} = \begin{pmatrix} g(x_1, b_1) & g(x_1, b_2) & \cdots & g(x_1, b_s) \\ g(x_2, b_1) & g(x_2, b_2) & \cdots & g(x_2, b_s) \\ \vdots & \vdots & \vdots & \vdots \\ g(x_n, b_1) & g(x_n, b_2) & \cdots & g(x_n, b_s) \end{pmatrix}, \tag{9}$$

the weight-vector of A be

$$\omega = \left(\omega_{b_1}, \omega_{b_2}, \cdots, \omega_{b_s}\right)^T, \tag{10}$$

the vector of D be

$$\xi = \left(g(x_1, d), g(x_2, d), \cdots, g(x_n, d)\right)^T. \tag{11}$$

In situations where the number of attributes surpasses the number of objects, the feature matrix may become non-invertible. To address this issue, we introduce a regularization parameter λ (set to 0.1 in this paper) and an identity matrix E. This enables us to obtain the closed-form solution for the weight-vector, which is given by:

$$\omega = (\mathcal{M}^T\mathcal{M} + \lambda E)^{-1}\mathcal{M}^T\xi. \tag{12}$$

Subsequently, the optimal weight-vector is sought by minimizing the loss function, which is defined as follows:

$$\mathcal{F} = \|\xi - \mathcal{M}\omega\|^2 + \lambda\|\omega\|^2. \tag{13}$$

By employing this method, we can derive the weight-vector for all knowledge granules and acquire the weight values corresponding to each knowledge granule, defined as follows:

$$\omega(b_i) = \omega_{b_i}, \quad i \in \{1, 2, \cdots, s\}. \tag{14}$$

Given a $\mathcal{NIS} = (U, A \cup \{d\})$, weight-vector for knowledge granules ω. For $\forall b \in A$, the mapping function h form the weight of knowledge granule to its corresponding granule, is defined as follows:

$$h(\omega(b_i)) = b_i, \quad i \in \{1, 2, \cdots, s\}. \tag{15}$$

From equation (12), we can deduce that as the weight value increases, the correlation with the decision attribute also increases.

Once the weights of knowledge granules are computed, we employ the Density-Based Spatial Clustering of Applications with Noise (DBSCAN) algorithm [28] from machine learning to perform granularity partitioning. This algorithm utilizes grid search to discover the optimal neighborhood radius and density threshold. It organizes data points into high-density regions while considering low-density regions as noise points. Consequently, based on the weight values of different knowledge granules, they are clustered into distinct granularities denoted as $P_i (i = 1, 2, \cdots, m \le 2^{|A|})$, and meet the following relationships:

1. $\forall P_i \subseteq A$, where $i = 1, 2, \cdots, m \le 2^{|A|}$;
2. $1 \le |P_i| \le |A|$, where $i = 1, 2, \cdots, m \le 2^{|A|}$ and "$|\cdot|$" represents the number of knowledge granules.

We apply the aforementioned approach to GMNRS and construct generalized multigranulation neighborhood rough set based on weight partition (W-GMNRS), which considers weight distributions. Next, we delve into the study of its characteristic support function and relationships from a matrix form.

3.2. Matrix form in generalized multigranulation neighborhood rough set based on weight partition (W-MGMNRS)

Definition 2. Given a $\mathcal{NIS} = (U, A \cup \{d\})$, for $P \subseteq A$, $P_i \subseteq P (i = 1, 2, \cdots, m \le 2^{|P|})$, $\forall x_t \in U (t = 1, 2, \cdots, n)$, let $R_\delta^{P_i}$ denote neighborhood relation of granularity P_i, the definition of the neighborhood relation matrix $\mathcal{M}_\delta^{P_i} = [\mathfrak{m}_{jk}^{\delta P_i}]_{n\times n}$ for P_i is as follows:

$$\mathcal{M}_\delta^{P_i} = \begin{pmatrix} \mathfrak{m}_{11}^{\delta P_i} & \mathfrak{m}_{12}^{\delta P_i} & \cdots & \mathfrak{m}_{1n}^{\delta P_i} \\ \mathfrak{m}_{21}^{\delta P_i} & \mathfrak{m}_{22}^{\delta P_i} & \cdots & \mathfrak{m}_{2n}^{\delta P_i} \\ \vdots & \vdots & \ddots & \vdots \\ \mathfrak{m}_{n1}^{\delta P_i} & \mathfrak{m}_{n2}^{\delta P_i} & \cdots & \mathfrak{m}_{nn}^{\delta P_i} \end{pmatrix},$$

where

$$\mathfrak{m}_{jk}^{\delta P_i} = \begin{cases} 1, & \Delta_{P_i}(x_j, x_k) \le \delta \\ 0, & \text{others} \end{cases} \tag{16}$$

signifies the fundamental element of $\mathcal{M}_\delta^{P_i}$, δ is the neighborhood radius, $\Delta_{P_i}(x_j, x_k)$ is the distance function under P_i.

Given a $\mathcal{NIS} = (U, A \cup \{d\})$, for $\forall x_t \in U (t = 1, 2, \cdots, n)$, let $f(x, d)$ denote the attribute value of x on d, the definition of the decision matrix $\mathcal{D}^d = [\mathfrak{D}_{jk}^d]_{n\times n}$ for d is as follows:

$$\mathcal{D}^d = \begin{pmatrix} \mathfrak{D}_{11}^d & \mathfrak{D}_{12}^d & \cdots & \mathfrak{D}_{1n}^d \\ \mathfrak{D}_{21}^d & \mathfrak{D}_{22}^d & \cdots & \mathfrak{D}_{2n}^d \\ \vdots & \vdots & \ddots & \vdots \\ \mathfrak{D}_{n1}^d & \mathfrak{D}_{n2}^d & \cdots & \mathfrak{D}_{nn}^d \end{pmatrix},$$

where

$$\mathfrak{D}_{jk}^d = \begin{cases} 1, & f(x_j, d) = f(x_k, d) \\ 0, & \text{others} \end{cases} \tag{17}$$

signifies the fundamental element of $\mathcal{D}^d$. In this paper, we solely focus on the consideration of a single decision.

Given a $\mathcal{NIS} = (U, A \cup \{d\})$, for $P \subseteq A$, $P_i \subseteq P (i = 1, 2, \cdots, m \le 2^{|P|})$, $\forall x_t \in U$, $[x_t]_{R_\delta^{P_i}}$ denotes the neighborhood class of x under P_i, let feature column vector $H_{x_t}(P_i) = (h_{G_1}(x_t), h_{G_2}(x_t), \cdots, h_{G_s}(x_t))^T$ and feature column vector $L_{x_t}(P_i) = (l_{G_1}(x_t), l_{G_2}(x_t), \cdots, l_{G_s}(x_t))^T$, where the feature elements are:

Table 1
An Information System.

U	b_1	b_2	b_3	b_4	b_5	b_6	b_7	b_8	d
x_1	0.40	0.92	0.61	0.73	0.84	0.55	0.33	0.72	1
x_2	0.72	0.53	0.82	0.42	0.64	0.97	0.85	0.47	1
x_3	0.63	0.46	0.45	0.87	0.76	0.57	0.38	0.95	1
x_4	0.81	0.75	0.96	0.38	0.51	0.68	0.49	0.87	2
x_5	0.93	0.64	0.54	0.76	0.47	0.89	0.79	0.58	2
x_6	0.52	0.87	0.76	0.65	0.91	0.34	0.45	0.76	2
x_7	0.76	0.37	0.48	0.92	0.69	0.74	0.86	0.59	2
x_8	0.47	0.61	0.83	0.54	0.35	0.96	0.67	0.78	3
x_9	0.67	0.73	0.57	0.39	0.89	0.43	0.98	0.67	3
x_{10}	0.92	0.56	0.68	0.78	0.41	0.62	0.73	0.85	3

$$h_{P_i}^{\sim X}(x_t)=\begin{cases}1, & [x_t]_{R_\delta^{P_i}}\cap X=\emptyset\\ 0, & [x_t]_{R_\delta^{P_i}}\cap X\neq\emptyset\end{cases},$$
$$l_{P_i}^{X}(x_t)=\begin{cases}1, & [x_t]_{R_\delta^{P_i}}\subseteq X\\ 0, & \text{others}\end{cases}. \tag{18}$$

Given a $\mathcal{NIS}=(U,A\cup\{d\})$. For $P\subseteq A$, $P_i\subseteq P(i=1,2,\cdots,m\leq 2^{|P|})$, $\forall x_t\in U$, let feature support function $H_{x_t}^{\sim X}(P)$ denotes the average level of inclusion of x_t relative to X across all attributes A, feature support function $L_{x_t}^{X}(P)$ denotes the average level of inclusion of x_t relative to the complement of X across all attributes A. The following is the delineation of their definition:

$$H_{x_t}^{\sim X}(P)=\frac{\sum_{i=1}^{m}(1-h_{P_i}^{\sim X}(x_t))}{m},$$
$$L_{x_t}^{X}(P)=\frac{\sum_{i=1}^{m}l_{P_i}^{X}(x_t)}{m}. \tag{19}$$

Given a $\mathcal{NIS}=(U,A\cup\{d\})$. For $P\subseteq A$, $\forall x_t\in U$, let $\beta\in(0.5,1]$, $G_P^O(X)=(g_P^O(x_t))_{n\times 1}$ and $G_P^N(X)=(g_P^N(x_t))_{n\times 1}$ denote the positive and negative domain vectors respectively. Their definitions are as follows:

$$g_P^O(x_t)=\begin{cases}1, & \text{If } L_{x_t}^{X}(P)\geq\beta\\ 0, & \text{others}\end{cases},$$
$$g_P^N(x_t)=\begin{cases}1, & \text{If } H_{x_t}^{\sim X}(P)>1-\beta\\ 0, & \text{others}\end{cases}. \tag{20}$$

Example 1. Table 1 is a NIS, where $U=\{x_1,x_2,\cdots,x_{10}\}$ is the object set, $A=\{b_1,b_2,\cdots,b_8\}$ is the condition attribute set and d is the decision attribute set. Assuming target set is $X=\{x_1,x_3,x_5,x_6,x_8\}$, neighborhood radius δ is 0.2, and the parameter β is 0.6. By employing Definition 1, we can derive the weight vector to be: $\omega=(0.22,0.05,0.01,-0.08,-0.10,0.21,0.31,0.07)^T$. Then, by utilizing DBSCAN algorithm, we can utilize the weights of each knowledge granule to classify them into the following granularities: $P_1=\{b_1,b_6,b_7\}$, $P_2=\{b_4,b_5\}$, and $P_3=\{b_2,b_3,b_8\}$.

Based on Definition 2, we can derive the decision matrix and the neighborhood relationship matrices corresponding to the granularities P_1, P_2, and P_3 as follows:

$$\mathcal{D}^d=\begin{pmatrix}1&1&1&0&0&0&0&0&0&0\\1&1&1&0&0&0&0&0&0&0\\1&1&1&0&0&0&0&0&0&0\\0&0&0&1&1&1&1&0&0&0\\0&0&0&1&1&1&1&0&0&0\\0&0&0&1&1&1&1&0&0&0\\0&0&0&1&1&1&1&0&0&0\\0&0&0&0&0&0&0&1&1&1\\0&0&0&0&0&0&0&1&1&1\\0&0&0&0&0&0&0&1&1&1\end{pmatrix},\mathcal{M}_\delta^{P_1}=\begin{pmatrix}1&0&1&0&0&0&0&0&0&0\\0&1&0&0&1&0&1&0&0&0\\1&0&1&1&0&0&0&0&0&0\\0&0&1&1&0&0&0&0&0&0\\0&1&0&0&1&0&1&0&0&0\\0&0&0&0&0&1&0&0&0&0\\0&1&0&0&1&0&1&0&0&1\\0&0&0&0&0&0&0&1&0&0\\0&0&0&0&0&0&0&0&1&0\\0&0&0&0&0&0&1&0&0&1\end{pmatrix},$$

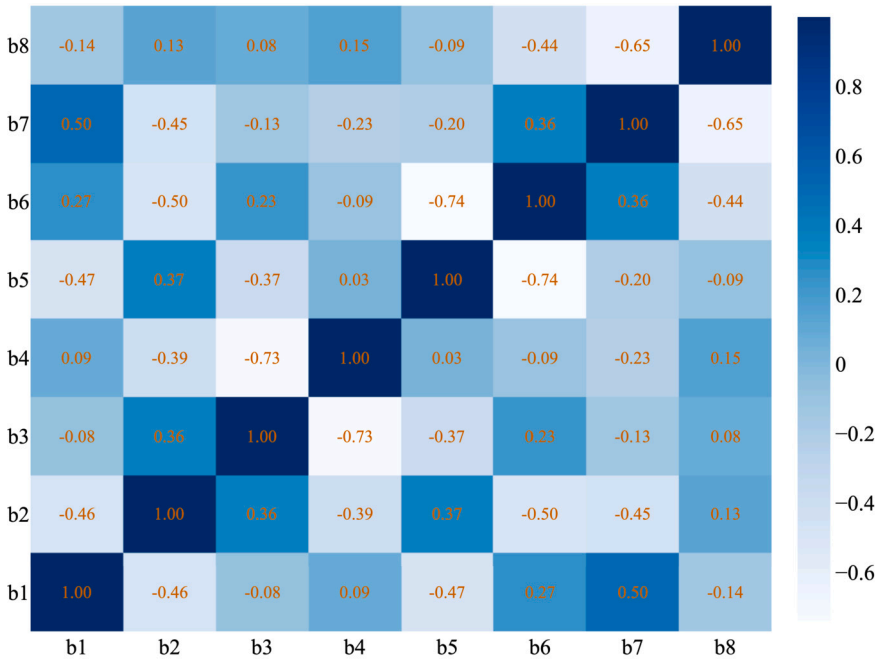

Fig. 2. Heatmap of Relationships Among Condition Attributes.

$$\mathcal{M}_{\delta}^{P_2}=\begin{pmatrix}1&0&1&0&0&1&0&0&0&0\\0&1&0&1&0&0&0&0&0&0\\1&0&1&0&0&0&1&0&0&0\\0&1&0&1&0&0&0&1&0&0\\0&0&0&0&1&0&0&0&0&1\\1&0&0&0&0&1&0&0&0&0\\0&0&1&0&0&0&1&0&0&0\\0&0&0&1&0&0&0&1&0&0\\0&0&0&0&0&0&0&0&1&0\\0&0&0&0&1&0&0&0&0&1\end{pmatrix},\mathcal{M}_{\delta}^{P_3}=\begin{pmatrix}1&0&0&0&0&1&0&0&1&0\\0&1&0&0&0&0&0&0&0&0\\0&0&1&0&0&0&0&0&0&0\\0&0&0&1&0&0&0&1&0&0\\0&0&0&0&1&0&0&0&1&0\\1&0&0&0&0&1&0&0&0&0\\0&0&0&0&0&0&1&0&0&0\\0&0&0&1&0&0&0&1&0&1\\1&0&0&0&1&0&0&0&1&0\\0&0&0&0&0&0&0&1&0&1\end{pmatrix}.$$

The feature column vector of x_i is as follows:

$$H_{x_1}(P)=(0,0,0)^T, H_{x_2}(P)=(1,1,0)^T, H_{x_3}(P)=(0,0,0)^T, H_{x_4}(P)=(0,0,0)^T, H_{x_5}(P)=(0,0,0)^T,$$
$$H_{x_6}(P)=(0,0,0)^T, H_{x_7}(P)=(1,0,0)^T, H_{x_8}(P)=(0,0,0)^T, H_{x_9}(P)=(0,1,1)^T, H_{x_{10}}(P)=(0,0,1)^T;$$
$$L_{x_1}(P)=(0,1,1)^T, L_{x_2}(P)=(0,0,0)^T, L_{x_3}(P)=(1,0,0)^T, L_{x_4}(P)=(0,0,0)^T, L_{x_5}(P)=(0,0,0)^T,$$
$$L_{x_6}(P)=(1,1,1)^T, L_{x_7}(P)=(0,0,0)^T, L_{x_8}(P)=(0,0,1)^T, L_{x_9}(P)=(0,0,0)^T, L_{x_{10}}(P)=(0,0,0)^T.$$

Based on the above, we can derive the two feature support functions for x_i as follows:

$$H_{x_1}^{\sim X}(P)=1, H_{x_2}^{\sim X}(P)=\frac{1}{3}, H_{x_3}^{\sim X}(P)=1, H_{x_4}^{\sim X}(P)=1, H_{x_5}^{\sim X}(P)=1,$$
$$H_{x_6}^{\sim X}(P)=1, H_{x_7}^{\sim X}(P)=\frac{2}{3}, H_{x_8}^{\sim X}(P)=1, H_{x_9}^{\sim X}(P)=\frac{1}{3}, H_{x_{10}}^{\sim X}(P)=\frac{2}{3};$$
$$L_{x_1}^{X}(P)=\frac{2}{3}, L_{x_2}^{X}(P)=0, L_{x_3}^{X}(P)=\frac{1}{3}, L_{x_4}^{X}(P)=0, L_{x_5}^{X}(P)=0,$$
$$L_{x_6}^{X}(P)=1, L_{x_7}^{X}(P)=0, L_{x_8}^{X}(P)=\frac{1}{3}, L_{x_9}^{X}(P)=0, L_{x_{10}}^{X}(P)=0.$$

Then, the positive and negative domain vectors can be derived as follows:

$$G_P^O(X)=(1,0,0,0,0,1,0,0,0,0),$$
$$G_P^N(X)=(1,0,1,1,1,1,1,1,0,1).$$

Fig. 2 illustrates a heatmap showcasing the interrelationships among the condition attributes. We utilized the Pearson correlation coefficient method for assessing the correlation among conditional attributes, employing the following calculation formula:

$$r=\frac{\sum_{i=1}^{n}(x_i-\bar{x})(y_i-\bar{y})}{\sqrt{\sum_{i=1}^{n}(x_i-\bar{x})^2\sum_{i=1}^{n}(y_i-\bar{y})^2}},$$

where x_i and y_i represent the values of two attributes within the same sample, $\bar{x}$ and $\bar{y}$ denote their respective means, n indicates the number of samples. The positive or negative values indicate the presence of positive or negative associations between the condition attributes, with higher values indicating stronger associations. Among b_1, b_6 and b_7, they exhibit positive correlations with each other and are the top three attributes with the highest values. Similarly, the remaining attributes demonstrate similar relationships. Hence, when partitioning the granularities based on the weights, we consider not only the relationships between the condition attributes and the decision attribute but also the relationships among the condition attributes themselves. Therefore, this partitioning approach exhibits strong interpretability and rationality.

3.3. Conditional entropy of W-MGMNRS

The notion of information entropy serves as a measure to evaluate the level of uncertainty in a discrete sample space. Consequently, entropy and its extensions have seen extensive applications across diverse domains. Within the context of feature selection, entropy plays a pivotal role. In this paper, we introduce a novel entropy measure named W-MGME, built upon the W-MGMNRS.

Definition 3. Given a $\mathcal{NIS}=(U,A\cup\{d\})$, for $P\subseteq A$, $P_i\subseteq P(i=1,2,\cdots,m\leq 2^{|P|})$, $U/d=\{d_1,d_2,\cdots,d_l\}$, the W-MGME is defined as

$$\mathcal{IE}(P,d)=-\sum_{i=1}^{m}\frac{\left|\bigcup_{k=1}^{l}G_{P_i}^{O}(d_k)\right|}{|U|}\log_2\frac{\left|\mathcal{M}_{\delta}^{P_i}\cap\mathcal{D}^d\right|}{\left|\mathcal{M}_{\delta}^{P_i}\right|}. \tag{21}$$

In equation (21), $\frac{\left|\bigcup_{k=1}^{l}G_{P_i}^{O}(d_k)\right|}{|U|}$ denotes the degree of dependence of d with respect to P_i. $\frac{\left|\mathcal{M}_{\delta}^{P_i}\cap\mathcal{D}^d\right|}{\left|\mathcal{M}_{\delta}^{P_i}\right|}$ can be regarded as a variable that reflects the extent of consistency in sample ranking based on d and all the condition attributes contained in granularity P_i. It is apparent that this variable is inversely related to W-MGME. As the value of $\mathcal{IE}(P,d)$ decreases, the significance of P_i increases.

3.4. Feature selection in matrix representation

When facing with high-dimensional data in feature selection, employing matrices can significantly reduce computation time and enhance algorithm efficiency. In this study, we employ interior significance measure and exterior significance measure to assess the importance of attributes. In the following, we will provide their respective definitions.

Definition 4. Given a $\mathcal{NIS}=(U,A\cup\{d\})$, for $P'\subseteq P\subseteq A$, $P_i\subseteq P'(i=1,2,\cdots,m\leq 2^{|P'|})$, $U/d=\{d_1,d_2,\cdots,d_l\}$, the interior significance measure of P_i is defined as

$$\mathcal{S}_{int}(P_i,P,d)=\mathcal{IE}(P-P_i,d)-\mathcal{IE}(P,d). \tag{22}$$

A higher value of $\mathcal{S}_{int}(P_i,P,d)$ indicates the relatively greater importance of the condition attribute subset P_i with respect to P.

Given a $\mathcal{NIS}=(U,A\cup\{d\})$, for $P'\subseteq P\subseteq A$, $P_i\subseteq P-P'(i=1,2,\cdots,m\leq 2^{|P-P'|})$, $U/d=\{d_1,d_2,\cdots,d_l\}$, the exterior significance measure of P_i is defined as

$$\mathcal{S}_{out}(P_i,P,d)=\mathcal{IE}(P,d)-\mathcal{IE}(P\cup P_i,d). \tag{23}$$

The exterior significance measure plays a vital role in the selection of feature subsets by identifying the critical features that have an impact on the decision outcome.

We propose the non-incremental heuristic feature selection algorithm, which called W-MGMN, aiming to select significant feature subsets using the interior and exterior significance measures defined in Definition 4. The selection process is outlined in Algorithm 1.

In algorithm W-MGMN, the time complexity of Step 2 is $O(mn)$, where m represents the number of features and n denotes the number of objects. Steps 3 and 4 have a time complexity of $O(m^2n^2)$. In Steps $5-10$, the selection of the core feature subset based on interior significance measure has a time complexity of $O(m^3n^2)$. In Steps $12-18$, sequentially adding the features with the highest exterior significance measure to the feature subset has a time complexity of $O(m^3n^2)$. Finally, in Steps $19-23$, redundant features are eliminated to obtain the optimal feature subset with a time complexity of $O(m^3n^2)$.

4. Matrix update approximation multigranulation neighborhood rough set based on weight partition

In the era of big data, where information is constantly evolving, new information is being incorporated, and outdated information is being discarded. When dealing with such dynamic data, employing static methods for analysis would require a significant amount

Algorithm 1: W-MGMN algorithm.

Input: A $\mathcal{NIS} = (U, A \cup \{d\})$, δ, $\beta \in (0.5, 1]$.
Output: An optimal feature subset B.

Initialize $B \leftarrow \emptyset$, $C \leftarrow \emptyset$;
Calculate the weights of each attributes and partition them;
Calculate the $\mathcal{IE}(P, d)$ for feature set A;
Calculate the $\mathcal{IE}(P - P_i, d)$ for $P_i \subseteq A$;
for $i = 1$ *to* $|A|$ **do**
  Calculate $\mathcal{S}_{int}(P_i, P, d)$ by Equation (22);
  if $\mathcal{S}_{int}(P_i, P, d) > 0$ **then**
    $C \leftarrow C \cup P_i$;
  end
end
Let $B \leftarrow C$;
while $\mathcal{IE}(B, d) > \mathcal{IE}(A, d)$ **do**
  for $j = 1$ *to* $|A - B|$ **do**
    Calculate $\mathcal{S}_{out}(P_j, B, d)$ by Equation (23);
  end
  Select $P_j = argmax_{P_j \subseteq A-B} \mathcal{S}_{out}(P_j, B, d)$;
  $B \leftarrow B \cup P_j$;
end
for $k = 1$ *to* $|B|$ **do**
  if $\mathcal{IE}(B - P_k, d) \leq \mathcal{IE}(B, d)$ **then**
    $B \leftarrow B - P_k$;
  end
end
return: The optimal feature subset B.

of time. Hence, in this paper, we utilize matrix form to update the approximations when changes in data structure occur. Our primary focus lies in understanding the changes that occur when objects are added or removed.

4.1. Updating mechanism of W-MGME when adding objects

As the quantity of objects in the data increases, the neighborhood relation matrix of each granularity undergoes alterations and necessitates prompt updates. This subsection will explore the subject from this viewpoint.

Proposition 1. *Given a $\mathcal{NIS} = (U, A \cup \{d\})$, for $P \subseteq A$, $P_i \subseteq P (i = 1, 2, \cdots, m \leq 2^{|P|})$, $\forall x_t \in U (t = 1, 2, \cdots, n)$, the neighborhood relation matrix $\mathcal{M}_\delta^{P_i} = [\mathfrak{m}_{j,k}^{\delta P_i}]_{n \times n}$ for P_i. Following the addition of n' objects, the updated neighborhood relation matrix $\mathcal{M}'^{P_i}_\delta$ is as follows:*

$$\mathcal{M}'^{P_i}_\delta = \left(\begin{array}{ccc|ccc} \mathfrak{m}_{11}^{\delta P_i} & \cdots & \mathfrak{m}_{1,n}^{\delta P_i} & \mathfrak{m}'^{\delta P_i}_{1,n+1} & \cdots & \mathfrak{m}'^{\delta P_i}_{1,n+n'} \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ \mathfrak{m}_{n,1}^{\delta P_i} & \cdots & \mathfrak{m}_{n,n}^{\delta P_i} & \mathfrak{m}'^{\delta P_i}_{n,n+1} & \cdots & \mathfrak{m}'^{\delta P_i}_{n,n+n'} \\ \hline \mathfrak{m}_{n+1,1}^{\delta P_i} & \cdots & \mathfrak{m}_{n+1,n}^{\delta P_i} & \mathfrak{m}'^{\delta P_i}_{n+1,n+1} & \cdots & \mathfrak{m}'^{\delta P_i}_{n+1,n+n'} \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ \mathfrak{m}'^{\delta P_i}_{n+n',1} & \cdots & \mathfrak{m}'^{\delta P_i}_{n+n',n} & \mathfrak{m}'^{\delta P_i}_{n+n',n+1} & \cdots & \mathfrak{m}'^{\delta P_i}_{n+n',n+n'} \end{array}\right),$$

where

$$\mathfrak{m}'^{\delta P_i}_{j,n+k} = \begin{cases} 1, & If\ x_{n+k} \in [x_j]_{R_\delta^{P_i}} \\ 0, & others \end{cases}, \ (j \in \{1, 2, \cdots, n\}, k \in \{1, 2, \cdots, n'\})$$

$$\mathfrak{m}'^{\delta P_i}_{n+j,k} = \begin{cases} 1, & If\ x_k \in [x_{n+j}]_{R_\delta^{P_i}} \\ 0, & others \end{cases}, \ (j \in \{1, 2, \cdots, n'\}, k \in \{1, 2, \cdots, n\}) \tag{24}$$

$$\mathfrak{m}'^{\delta P_i}_{n+j,n+k} = \begin{cases} 1, & If\ x_{n+k} \in [x_{n+j}]_{R_\delta^{P_i}} \\ 0, & others \end{cases}. \ (j \in \{1, 2, \cdots, n'\}, k \in \{1, 2, \cdots, n'\})$$

Proposition 2. *Given a $\mathcal{NIS} = (U, A \cup \{d\})$, for $\forall x_t \in U (t = 1, 2, \cdots, n)$, the decision matrix $\mathcal{D}^d = [\mathfrak{D}_{j,k}^d]_{n \times n}$ for d, let $f(x, d)$ denote the attribute value of x on d. Following the addition of n' objects, the updated decision matrix $\mathcal{D}'^d$ is as follows:*

$$D'^{d}=\left(\begin{array}{ccc|ccc} \mathfrak{D}^{d}_{11} & \cdots & \mathfrak{D}^{d}_{1,n} & \mathfrak{D}'^{d}_{1,n+1} & \cdots & \mathfrak{D}'^{d}_{1,n+n'} \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ \mathfrak{D}^{d}_{n,1} & \cdots & \mathfrak{D}^{d}_{n,n} & \mathfrak{D}'^{d}_{n,n+1} & \cdots & \mathfrak{D}'^{d}_{n,n+n'} \\ \hline \mathfrak{D}'^{d}_{n+1,1} & \cdots & \mathfrak{D}'^{d}_{n+1,n} & \mathfrak{D}'^{d}_{n+1,n+1} & \cdots & \mathfrak{D}'^{d}_{n+1,n+n'} \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ \mathfrak{D}'^{d}_{n+n',1} & \cdots & \mathfrak{D}'^{d}_{n+n',n} & \mathfrak{D}'^{d}_{n+n',n+1} & \cdots & \mathfrak{D}'^{d}_{n+n',n+n'} \end{array}\right),$$

where

$$\begin{aligned} \mathfrak{D}'^{d}_{j,n+k}&=\begin{cases}1, & f(x_j,d)=f(x_{n+k},d)\\ 0, & others\end{cases},\ (j\in\{1,2,\cdots,n\},k\in\{1,2,\cdots,n'\})\\ \mathfrak{D}'^{d}_{n+j,k}&=\begin{cases}1, & f(x_{n+j},d)=f(x_k,d)\\ 0, & others\end{cases},\ (j\in\{1,2,\cdots,n'\},k\in\{1,2,\cdots,n\})\\ \mathfrak{D}'^{d}_{n+j,n+k}&=\begin{cases}1, & f(x_{n+j},d)=f(x_{n+k},d)\\ 0, & others\end{cases}.\ (j\in\{1,2,\cdots,n'\},k\in\{1,2,\cdots,n'\}) \end{aligned} \tag{25}$$

Once the neighborhood relation matrix is updated, we will proceed to explore the inclusion relationship between the updated neighborhood class and the target set. We will begin by examining the dynamic updating mechanism of t he feature matrix, as outlined below.

Proposition 3. *Given a $\mathcal{NIS}=(U,A\cup\{d\})$, for $U=\{x_1,x_2,\cdots,x_n\}$, $P\subseteq A$, $P_i\subseteq P(i=1,2,\cdots,m\le 2^{|P|})$, following the addition of n' objects, the domain is transformed into U'. Subsequently, the feature matrix of U' is updated as:*

$$H^{\sim X}_{U'}(P)=\left(\begin{array}{cccc} h^{\sim X}_{P_1}(x_1) & h^{\sim X}_{P_2}(x_1) & \cdots & h^{\sim X}_{P_s}(x_1) \\ h^{\sim X}_{P_1}(x_2) & h^{\sim X}_{P_2}(x_2) & \cdots & h^{\sim X}_{P_s}(x_2) \\ \vdots & \vdots & \vdots & \vdots \\ h^{\sim X}_{P_1}(x_n) & h^{\sim X}_{P_2}(x_n) & \cdots & h^{\sim X}_{P_s}(x_n) \\ \hline h^{\sim X}_{P_1}(x_{n+1}) & h^{\sim X}_{P_2}(x_{n+1}) & \cdots & h^{\sim X}_{P_s}(x_{n+1}) \\ \vdots & \vdots & \vdots & \vdots \\ h^{\sim X}_{P_1}(x_{n+n'}) & h^{\sim X}_{P_2}(x_{n+n'}) & \cdots & h^{\sim X}_{P_s}(x_{n+n'}) \end{array}\right), L^{X}_{U'}(P)=\left(\begin{array}{cccc} l^{X}_{P_1}(x_1) & l^{X}_{P_2}(x_1) & \cdots & l^{X}_{P_s}(x_1) \\ l^{X}_{P_1}(x_2) & l^{X}_{P_2}(x_2) & \cdots & l^{X}_{P_s}(x_2) \\ \vdots & \vdots & \vdots & \vdots \\ l^{X}_{P_1}(x_n) & l^{X}_{P_2}(x_n) & \cdots & l^{X}_{P_s}(x_n) \\ \hline l^{X}_{P_1}(x_{n+1}) & l^{X}_{P_2}(x_{n+1}) & \cdots & l^{X}_{P_s}(x_{n+1}) \\ \vdots & \vdots & \vdots & \vdots \\ l^{X}_{P_1}(x_{n+n'}) & l^{X}_{P_2}(x_{n+n'}) & \cdots & l^{X}_{P_s}(x_{n+n'}) \end{array}\right),$$

where, for $\forall t\in\{1,2,\cdots,n'\}$, $\forall i\in\{1,2,\cdots,s\}$, we can derive:

$$\begin{aligned} h^{\sim X}_{P_i}(x_{n+t})&=\begin{cases}1, & [x_{n+t}]_{R^{P_i}_{\delta}}\cap X=\emptyset\\ 0, & [x_{n+t}]_{R^{P_i}_{\delta}}\cap X\neq\emptyset\end{cases},\\ l^{X}_{P_i}(x_{n+t})&=\begin{cases}1, & [x_{n+t}]_{R^{P_i}_{\delta}}\subseteq X\\ 0, & others\end{cases}. \end{aligned} \tag{26}$$

Given a $\mathcal{NIS}=(U,A\cup\{d\})$, for $P\subseteq A$, $\forall x_t\in U(t=1,2,\cdots,n)$, $\beta\in(0.5,1]$, following the addition of n' objects, the updated positive and negative domain vectors are as follows:

$$\begin{aligned} G^{O}_{P}(X)&=(g^{O}_{P}(x_1),g^{O}_{P}(x_2),\cdots,g^{O}_{P}(x_n),\cdots,g^{O}_{P}(x_{n+n'})),\\ G^{N}_{P}(X)&=(g^{N}_{P}(x_1),g^{N}_{P}(x_2),\cdots,g^{N}_{P}(x_n),\cdots,g^{N}_{P}(x_{n+n'})). \end{aligned}$$

In the case of object addition, we can efficiently update the feature matrix by directly applying Proposition 3. This strategy eliminates the need to recalculate previously computed, and significantly reduces the required processing time, thereby enhancing computational efficiency.

4.2. Updating mechanism of W-MGME when deleting objects

Apart from object addition, there is also be cases where certain objects are deleted. Similar to the scenario of object addition, when objects are deleted, the neighborhood relationship matrix of each granularity experiences alterations. This subsection will examine the subject from this standpoint.

Proposition 4. *Given a $\mathcal{NIS}=(U,A\cup\{d\})$, for $P\subseteq A$, $P_i\subseteq P(i=1,2,\cdots,m\le 2^{|P|})$, $\forall x_t\in U(t=1,2,\cdots,n)$, the neighborhood relation matrix $\mathcal{M}_\delta^{P_i}=[\mathfrak{m}_{j,k}^{\delta P_i}]_{n\times n}$ for P_i. Following the deletion of n' objects, the updated neighborhood relation matrix $\mathcal{M}'^{P_i}_\delta$ is as follows:*

$$\mathcal{M}'^{P_i}_\delta=\left(\begin{array}{ccc|ccc}\mathfrak{m}_{11}^{\delta P_i}&\cdots&\mathfrak{m}_{1,n-n'-1}^{\delta P_i}&\mathfrak{m}'^{\delta P_i}_{1,n-n'}&\cdots&\mathfrak{m}'^{\delta P_i}_{1,n}\\ \vdots&\ddots&\vdots&\vdots&\ddots&\vdots\\ \mathfrak{m}_{n-n'-1,1}^{\delta P_i}&\cdots&\mathfrak{m}_{n-n'-1,n-n'-1}^{\delta P_i}&\mathfrak{m}'^{\delta P_i}_{n-n'-1,n-n'}&\cdots&\mathfrak{m}'^{\delta P_i}_{n-n'-1,n}\\ \hline \mathfrak{m}_{n-n',1}^{\delta P_i}&\cdots&\mathfrak{m}_{n-n',n-n'-1}^{\delta P_i}&\mathfrak{m}'^{\delta P_i}_{n-n',n-n'}&\cdots&\mathfrak{m}'^{\delta P_i}_{n-n',n}\\ \vdots&\ddots&\vdots&\vdots&\ddots&\vdots\\ \mathfrak{m}'^{\delta P_i}_{n,1}&\cdots&\mathfrak{m}'^{\delta P_i}_{n,n-n'-1}&\mathfrak{m}'^{\delta P_i}_{n,n-n'}&\cdots&\mathfrak{m}'^{\delta P_i}_{n,n}\end{array}\right),$$

where

$$\begin{aligned}\mathfrak{m}'^{\delta P_i}_{j,n-k}&=\begin{cases}1, & If\ x_{n-k}\in[x_j]_{R_\delta^{P_i}}\\ 0, & others\end{cases},\ (j\in\{1,2,\cdots,n\},k\in\{1,2,\cdots,n'\})\\ \mathfrak{m}'^{\delta P_i}_{n-j,k}&=\begin{cases}1, & If\ x_k\in[x_{n-j}]_{R_\delta^{P_i}}\\ 0, & others\end{cases},\ (j\in\{1,2,\cdots,n'\},k\in\{1,2,\cdots,n\})\\ \mathfrak{m}'^{\delta P_i}_{n-j,n-k}&=\begin{cases}1, & If\ x_{n-k}\in[x_{n-j}]_{R_\delta^{P_i}}\\ 0, & others\end{cases}.\ (j\in\{1,2,\cdots,n'\},k\in\{1,2,\cdots,n'\})\end{aligned}\tag{27}$$

Proposition 5. *Given a $\mathcal{NIS}=(U,A\cup\{d\})$, for $\forall x_t\in U(t=1,2,\cdots,n)$, the decision matrix $\mathcal{D}^d=[\mathfrak{D}_{j,k}^d]_{n\times n}$ for d, let $f(x,d)$ denote the attribute value of x on d. Following the deletion of n' objects, the updated decision matrix $\mathcal{D}'^d$ is as follows:*

$$\mathcal{D}'^d=\left(\begin{array}{ccc|ccc}\mathfrak{D}_{11}^d&\cdots&\mathfrak{D}_{1,n-n'-1}^d&\mathfrak{D}'^d_{1,n-n'}&\cdots&\mathfrak{D}'^d_{1,n}\\ \vdots&\ddots&\vdots&\vdots&\ddots&\vdots\\ \mathfrak{D}_{n-n'-1,1}^d&\cdots&\mathfrak{D}_{n-n'-1,n-n'-1}^d&\mathfrak{D}'^d_{n-n'-1,n-n'}&\cdots&\mathfrak{D}'^d_{n-n'-1,n}\\ \hline \mathfrak{D}'^d_{n-n',1}&\cdots&\mathfrak{D}'^d_{n-n',n-n'-1}&\mathfrak{D}'^d_{n-n',n-n'}&\cdots&\mathfrak{D}'^d_{n-n',n}\\ \vdots&\ddots&\vdots&\vdots&\ddots&\vdots\\ \mathfrak{D}'^d_{n,1}&\cdots&\mathfrak{D}'^d_{n,n-n'-1}&\mathfrak{D}'^d_{n,n-n'}&\cdots&\mathfrak{D}'^d_{n,n}\end{array}\right),$$

where

$$\begin{aligned}\mathfrak{D}'^d_{j,n-k}&=\begin{cases}1, & f(x_j,d)=f(x_{n-k},d)\\ 0, & others\end{cases},\ (j\in\{1,2,\cdots,n\},k\in\{1,2,\cdots,n'\})\\ \mathfrak{D}'^d_{n-j,k}&=\begin{cases}1, & f(x_{n-j},d)=f(x_k,d)\\ 0, & others\end{cases},\ (j\in\{1,2,\cdots,n'\},k\in\{1,2,\cdots,n\})\\ \mathfrak{D}'^d_{n-j,n-k}&=\begin{cases}1, & f(x_{n-j},d)=f(x_{n-k},d)\\ 0, & others\end{cases}.\ (j\in\{1,2,\cdots,n'\},k\in\{1,2,\cdots,n'\})\end{aligned}\tag{28}$$

Proposition 6. *Given a $\mathcal{NIS}=(U,A\cup\{d\})$, for $U=\{x_1,x_2,\cdots,x_n\}$, $P\subseteq A$, $P_i\subseteq P(i=1,2,\cdots,m\le 2^{|P|})$, following the deletion of n' objects, the domain is transformed into U'. Subsequently, the feature matrix of U' is updated as:*

$$H^{\sim X}_{U'}(P)=\left(\begin{array}{cccc}h^{\sim X}_{P_1}(x_1)&h^{\sim X}_{P_2}(x_1)&\cdots&h^{\sim X}_{P_s}(x_1)\\ h^{\sim X}_{P_1}(x_2)&h^{\sim X}_{P_2}(x_2)&\cdots&h^{\sim X}_{P_s}(x_2)\\ \vdots&\vdots&\vdots&\vdots\\ \hline h^{\sim X}_{P_1}(x_{n-n'-1})&h^{\sim X}_{P_2}(x_{n-n'-1})&\cdots&h^{\sim X}_{P_s}(x_{n-n'-1})\\ \hline h^{\sim X}_{P_1}(x_{n-n'})&h^{\sim X}_{P_2}(x_{n-n'})&\cdots&h^{\sim X}_{P_s}(x_{n-n'})\\ \vdots&\vdots&\vdots&\vdots\\ h^{\sim X}_{P_1}(x_n)&h^{\sim X}_{P_2}(x_n)&\cdots&h^{\sim X}_{P_s}(x_n)\end{array}\right),L^X_{U'}(P)=\left(\begin{array}{cccc}l^X_{P_1}(x_1)&l^X_{P_2}(x_1)&\cdots&l^X_{P_s}(x_1)\\ l^X_{P_1}(x_2)&l^X_{P_2}(x_2)&\cdots&l^X_{P_s}(x_2)\\ \vdots&\vdots&\vdots&\vdots\\ \hline l^X_{P_1}(x_{n-n'-1})&l^X_{P_2}(x_{n-n'-1})&\cdots&l^X_{P_s}(x_{n-n'-1})\\ \hline l^X_{P_1}(x_{n-n'})&l^X_{P_2}(x_{n-n'})&\cdots&l^X_{P_s}(x_{n-n'})\\ \vdots&\vdots&\vdots&\vdots\\ l^X_{P_1}(x_n)&l^X_{P_2}(x_n)&\cdots&l^X_{P_s}(x_n)\end{array}\right),$$

where, for $\forall t\in\{1,2,\cdots,n'\}$, $\forall i\in\{1,2,\cdots,s\}$, we can derive:

Algorithm 2: W-MGMA algorithm.

Input: A $\mathcal{NIS}=(U,A\cup\{d\})$, and its optimal feature subset B, δ, $\beta\in(0.5,1]$, $U'=\{x_{n+1},x_{n+2},\cdots,x_{n+n'}\}$, matrices $\mathcal{M}_{\delta,U}^{P_i^A}(P_i^A\subseteq A)$, $\mathcal{D}_U^{d,A}$, $\mathcal{M}_{\delta,U}^{P_j^B}(P_j^B\subseteq B)$, $\mathcal{D}_U^{d,B}$

Output: A new optimal feature subset B'.

1 Add object set $\widehat{U}\leftarrow U\cup U'$;
2 Update matrices $\mathcal{M}_{\delta,\widehat{U}}^{P_i}\leftarrow\mathcal{M}_{\delta,U}^{P_i}$, $\mathcal{D}_{\widehat{U}}^{d,A}\leftarrow\mathcal{D}_U^{d,A}$, $\mathcal{M}_{\delta,\widehat{U}}^{P_j^B}\leftarrow\mathcal{M}_{\delta,U}^{P_j^B}$, $\mathcal{D}_{\widehat{U}}^{d,B}\leftarrow\mathcal{D}_U^{d,B}$;
3 Calculate the new $\mathcal{IE}_{\widehat{U}}(P^A,d)$ for feature set A and the new $\mathcal{IE}_{\widehat{U}}(P^B,d)$ for feature set B;
4 **if** $\mathcal{IE}_{\widehat{U}}(P^B,d)>\mathcal{IE}_{\widehat{U}}(P^A,d)$ **then**
5 | turn to Step 9;
6 **else**
7 | turn to Step 15;
8 **end**
9 For each $P_k^{A-B}\subseteq\{P^A-P^B\}$, using Equation (22) to calculate $S_{out}^{\widehat{U}}(P_k^{A-B},B,d)$ and then arrange these granularities in descending order based on their exterior significance measures as $\{P_1^{A-B},P_2^{A-B},\cdots,P_{|A-B|}^{A-B}\}$;
10 **while** $\mathcal{IE}_{\widehat{U}}(P^B,d)>\mathcal{IE}_{\widehat{U}}(P^A,d)$ **do**
11 | **for** $l=1$ *to* $|A-B|$ **do**
12 | | select $B\leftarrow B\cup\{P_l^{A-B}\}$ and calculate $\mathcal{IE}_{\widehat{U}}(P^B,d)$;
13 | **end**
14 **end**
15 **for** *each* $P^B\subseteq B$ **do**
16 | Calculate $\mathcal{IE}_{\widehat{U}}(B-P^B,d)$;
17 | **if** $\mathcal{IE}_{\widehat{U}}(B-P^B,d)\le\mathcal{IE}_{\widehat{U}}(B,d)$ **then**
18 | | $B\leftarrow B-P^B$
19 | **end**
20 **end**
21 $B'\leftarrow B$;
22 **return**: The optimal feature subset B'.

$$h_{P_i}^{\sim X}(x_{n-t})=\begin{cases}1, & [x_{n-t}]_{R_\delta^{P_i}}\cap X=\emptyset\\ 0, & [x_{n-t}]_{R_\delta^{P_i}}\cap X\neq\emptyset\end{cases},$$
$$l_{P_i}^{X}(x_{n-t})=\begin{cases}1, & [x_{n-t}]_{R_\delta^{P_i}}\subseteq X\\ 0, & others\end{cases}. \tag{29}$$

Given a $\mathcal{NIS}=(U,A\cup\{d\})$, for $P\subseteq A$, $\forall x_t\in U(t=1,2,\cdots,n)$, $\beta\in(0.5,1]$, following the addition of n' objects, the updated positive and negative domain vectors are as follows:

$$G_P^O(X)=(g_P^O(x_1),g_P^O(x_2),\cdots,g_P^O(x_{n-n'}),\cdots,g_P^O(x_n)),$$
$$G_P^N(X)=(g_P^N(x_1),g_P^N(x_2),\cdots,g_P^N(x_{n-n'}),\cdots,g_P^N(x_n)).$$

4.3. Updating algorithm when adding or deleting objects

To facilitate efficient feature selection during object addition or deletion, we have devised two algorithms, namely W-MGMA and W-MGMD. The specific algorithmic processes are presented in Algorithm 2 and Algorithm 3.

In algorithm W-MGMA, let m represent the number of features, $\tilde{m}$ denotes the number of the previous feature subset B, n denotes the number of original objects and $\tilde{n}$ denotes the number of objects added. Step 1 involves adding objects to the original INS. Next, in Step $2-3$, the new neighborhood relation matrix and decision matrix are updated using Proposition 1 and Proposition 2, then calculate the W-MGME before and after the addition of objects, with a time complexity of $O(m^2(n+\tilde{n})\tilde{n})$. Steps $4-8$ determine whether the W-MGME of the previous feature subset B is greater than that of the original feature set A. Steps $9-14$ sort the remaining granularities in descending order based on their corresponding external significance measures and update the feature subset, with a time complexity of $O((m-\tilde{m})^2(n+\tilde{n})^2)$. Steps $15-20$ remove redundant attributes from the feature subset B, with a time complexity of $O(\tilde{m}^3\tilde{n}^2)$. Finally, the optimal feature subset is outputted.

In algorithm W-MGMD, m, n, and $\tilde{m}$ have the same meaning as in Algorithm W-MGMA. $\tilde{n}$ denotes the number of objects deleted. In Step 1, the original INS undergoes object deletion. Subsequently, in Steps $2-3$, the new neighborhood relation matrix and decision matrix are updated based on Proposition 4 and Proposition 5, respectively, and then, the W-MGME is calculated before and after the deletion of objects. This calculation involves a time complexity of $O(m^2(n-\tilde{n})n)$. Steps $4-8$ determine whether the W-MGME of the previous feature subset B is less than or equal to the W-MGME of the original feature set A. In Steps $9-14$, redundant attributes are eliminated from the feature subset B. The time complexity of this operation is $O(\tilde{m}^3\tilde{n}^2)$. During Steps $15-20$, the remaining granularities are sorted in a descending order based on their corresponding external significance measures. Subsequently, the feature subset is updated. The computational complexity of this procedure is $O((m-\tilde{m})^2(n-\tilde{n})^2)$. Finally, the optimal feature subset is generated.

Algorithm 3: W-MGMD algorithm.

Input: A $\mathcal{NIS} = (U, A \cup \{d\})$, and its optimal feature subset B, δ, $\beta \in (0.5, 1]$, $U' = \{x_1, x_2, \cdots, x'_n\}$, matrices $\mathcal{M}^{P_i^A}_{\delta,U}(P_i^A \subseteq A)$, $\mathcal{D}^{d,A}_U$, $\mathcal{M}^{P_j^B}_{\delta,U}(P_j^B \subseteq B)$, $\mathcal{D}^{d,B}_U$.
Output: A new optimal feature subset B'.

Delete object set $\widehat{U} \leftarrow U - U'$;
Update matrices $\mathcal{M}^{P_i}_{\delta,\widehat{U}} \leftarrow \mathcal{M}^{P_i}_{\delta,U}$, $\mathcal{D}^{d,A}_{\widehat{U}} \leftarrow \mathcal{D}^{d,A}_U$, $\mathcal{M}^{P_j^B}_{\delta,\widehat{U}} \leftarrow \mathcal{M}^{P_j^B}_{\delta,U}$, $\mathcal{D}^{d,B}_{\widehat{U}} \leftarrow \mathcal{D}^{d,B}_U$;
Calculate the new $\mathcal{IE}_{\widehat{U}}(P^A, d)$ for feature set A and the new $\mathcal{IE}_{\widehat{U}}(P^B, d)$ for feature set B;
if $\mathcal{IE}_{\widehat{U}}(P^B, d) \le \mathcal{IE}_{\widehat{U}}(P^A, d)$ **then**
  turn to Step 9;
else
  turn to Step 15;
end
for *each* $P^B \subseteq B$ **do**
  Calculate $\mathcal{IE}_{\widehat{U}}(B - P^B, d)$;
  if $\mathcal{IE}_{\widehat{U}}(B, d) \le \mathcal{IE}_{\widehat{U}}(B - P^B, d)$ **then**
    $B \leftarrow B - P^B$
  end
end
For each $P_k^{A-B} \subseteq \{P^A - P^B\}$, using Equation (22) to calculate $S^{\widehat{U}}_{out}(P_k^{A-B}, B, d)$ and then arrange these granularities in descending order based on their exterior significance measures as $\{P_1^{A-B}, P_2^{A-B}, \cdots, P_{|A-B|}^{A-B}\}$;
while $\mathcal{IE}_{\widehat{U}}(P^B, d) > \mathcal{IE}_{\widehat{U}}(P^A, d)$ **do**
  for $l = 1$ *to* $|A - B|$ **do**
    select $B \leftarrow B \cup \{P_l^{A-B}\}$ and calculate $\mathcal{IE}_{\widehat{U}}(P^B, d)$;
  end
end
$B' \leftarrow B$;
return: The optimal feature subset B'.

Table 2
Time Complexity Comparison of W-MGMN, W-MGMA and W-MGMD.

Algorithms	Time complexity
W-MGMN	$O(m^2(n+\tilde{n}) + m^3(n+\tilde{n})^2 + m^3(n+\tilde{n})^2 + m^3(n+\tilde{n})^2)$
W-MGMA	$O(m^2\tilde{n}(n+\tilde{n}) + (m-\tilde{m})^2(n+\tilde{n})^2 + \tilde{m}^3(n+\tilde{n})^2)$
W-MGMN	$O(m^2(n-\tilde{n}) + m^3(n-\tilde{n})^2 + m^3(n-\tilde{n})^2 + m^3(n-\tilde{n})^2)$
W-MGMD	$O(m^2\tilde{n}(n-\tilde{n}) + (m-\tilde{m})^2(n-\tilde{n})^2 + \tilde{m}^3(n-\tilde{n})^2)$

Table 3
The Detailed Introduction of The Datasets.

No.	Datasets	Abbreviation	Objects	Features	Classes	Type
1	Statlog	Statlog	2310	19	7	Real
2	Abalone	Abalone	4178	9	3	Real
3	Shill Bidding	Shill	6321	13	2	Real
4	Central Nervous System	Nervous	60	7130	2	Real
5	DLBCL-Harvard	DLBCL	77	7131	2	Real
6	LungCancer-BAWHospital	LungB	181	12533	2	Real
7	LungCancer-DanaFarberCancerInstitute	LungD	203	12600	5	Real
8	Prostate Cancer	Prostate	102	12601	2	Real
9	Breast Cancer	Breast	97	16185	2	Real
10	Gene Expression Cancer RNA-Seq	Gene	801	20531	5	Real
11	Ovarian Cancer NCI Q-Star	Ovarian	215	37333	3	Real
12	Condition Monitoring of Hydraulic Systems	Condition	2205	43680	3	Real

Table 2 demonstrates the overall time complexity of W-MGMN, W-MGMA and W-MGMD respectively. It is evident from the table that the time complexity of W-MGMA and W-MGMD are significantly lower compared to the time complexity of the W-MGMN. Therefore, the W-MGMA and W-MGMD demonstrate significantly higher efficiency in feature selection compared to the W-MGMN.

5. Experimental decision and analysis

In this chapter, we selected 12 datasets from UCI (as shown in Table 3) and conducted a series of experiments to validate the effectiveness of the proposed feature algorithm. The experimental algorithms were implemented in Python and executed on a personal computer with the following specifications: Apple M1 CPU, 16 GB of RAM, and Sonoma 14.2.1 operating system.

5.1. Experimental design

Before commencing the experiments, we conducted data preprocessing by employing the technique of max-min scaling normalization to rescale the data within the range of 0 to 1. The procedure is as follows:

$$b(x_j) = \frac{b(x_j) - \min b(x)}{\max b(x) - \min b(x)},$$

where $\min b(x)$ and $\max b(x)$ represent the maximum and minimum values of attribute b, respectively.

To address missing values within the dataset, we employed the K-Nearest Neighbors algorithm to approximate these values by referencing neighboring samples. We determined similarity between a missing value and other samples through Euclidean distance calculation. This process allowed us to pinpoint the K most similar samples as the closest neighbors to the missing value. Ultimately, we filled in the missing values by averaging the values of these nearest neighbor samples.

To evaluate the effectiveness of the designed algorithm, we used four classifiers: Recursive Neural Network (RNN), Random Forest (RF), Naive Bayes (NB), and Decision Tree (DT) to assess the classification accuracy after performing feature selection. In the DT classifier, the max depth was set to 3. Similarly, in the RF classifier, the max depth was set to 3. For the RNN classifier, the batch size was set to 32, and the number of iterations was set to 100. The NB classifier was utilized with default parameters. For the classification process, we applied ten-fold cross-validation, randomly dividing the dataset into ten mutually exclusive subsets. In each iteration, one subset was used as the validation set, while the remaining nine subsets were employed for model training and evaluation. This process was repeated ten times, and the average results were considered as the final evaluation metrics.

Initially, we conducted a time analysis for feature selection on 12 datasets, investigating the influence of various neighborhood radius δ and parameter β on the algorithm's effectiveness. Based on these results, we selected suitable parameters for subsequent experiments. Subsequently, we compared the classification accuracy and runtime of our proposed algorithm with six different feature selection algorithms on identical datasets to validate its efficacy. Lastly, we employed p-value tests to examine the disparities between our algorithm and the comparative approaches. The specific details of the selected comparative algorithms are outlined as follows:

1. *Hybrid Feature Selection Method Based on Harmony Search and Naked Mole-Rat Algorithms (HS-NMR)* [29]: The paper employs the Harmony Search (HS) algorithm and a novel nature-inspired algorithm known as the Naked Mole-Rat (NMR) algorithm to devise a novel hybrid feature selection approach for identifying the optimal subset of features.
2. *Hybrid two-stage feature selection method for microarray data (MMBDE)* [30]: Introducing a two-stage hybrid feature selection method, MMBDE, based on the improved minimum redundancy maximum relevance (mRMR) and enhanced binary differential evolution (BDE) algorithms.
3. *Hierarchical Harris hawks optimizer for feature selection (EHHO)* [31]: This algorithm employs the Harris's Hawk Optimization (HHO) technique and incorporates a hierarchical framework to address intricate problems, thereby devising an algorithm dedicated to feature selection.
4. *Dynamic Salp swarm algorithm (DSSA)* [32]: This algorithm is built upon the Shark Search Algorithm (SSA) and proposes an enhanced version that incorporates a local search algorithm to enhance the capabilities of SSA. This modification aims to address the challenges of maintaining population diversity and avoiding local optima commonly encountered by SSA.
5. *Feature Selection using PSO-MI (HDFS)* [33]: This method is based on particle swarm optimization and proposes a novel feature selection algorithm for handling high-dimensional datasets.
6. *Fast Genetic Algorithm for Feature Selection (CHC_{QX})* [34]: This method utilizes a qualitative approximation variant called CHC_{QX}, which is based on the genetic algorithm-based CHC algorithm, for feature selection.
7. *Embedded chaotic whale survival algorithm (ECWSA)* [35]: This algorithm incorporates chaos theory to steer the movement patterns of whales during the search process, thereby enhancing the effectiveness of feature selection.
8. *competitive grey wolf optimizer algorithm (OBCGWO)* [36]: This algorithm enhances the competitive binary grey wolf optimizer by introducing the opposition-based competitive grey wolf optimizer, allowing it to execute feature selection efficiently within a continuous search space.

5.2. Performance verification of the W-MGMN algorithm

In this subsection, we will utilize 12 datasets to evaluate the performance of W-MGMN by comparing its feature selection and classification capabilities with those of other algorithms.

To begin with, we examine the influence of various parameters on algorithm effectiveness, using the results obtained under the RNN classifier as an example. Neighborhood radius δ is varied from 0 to 1 with a step size of 0.2, while parameter β is selected from the range of 0.6 to 1 with an increment of 0.2. The classification results corresponding to different parameter values under the RNN classifier are displayed in Fig. 3. From the figure, we can observe that the variation in accuracy is more significant when δ changes compared to when β changes. Therefore, we can conclude that δ has a greater impact on the results than β. Additionally, when δ is within the range of 0 to 0.2 and β is set to 0.6, the accuracy of the experimental results is generally the highest. Based on these findings, for the subsequent experiments, we will select δ as 0.1 and β as 0.6.

Table 4 presents the number of features selected by W-MGMN and other algorithms across the 12 datasets. It is evident from the table that all algorithms have selected a smaller number of features compared to the original dataset. Notably, W-MGMN exhibits the lowest average number of selected features, indicating its effectiveness in performing feature selection.

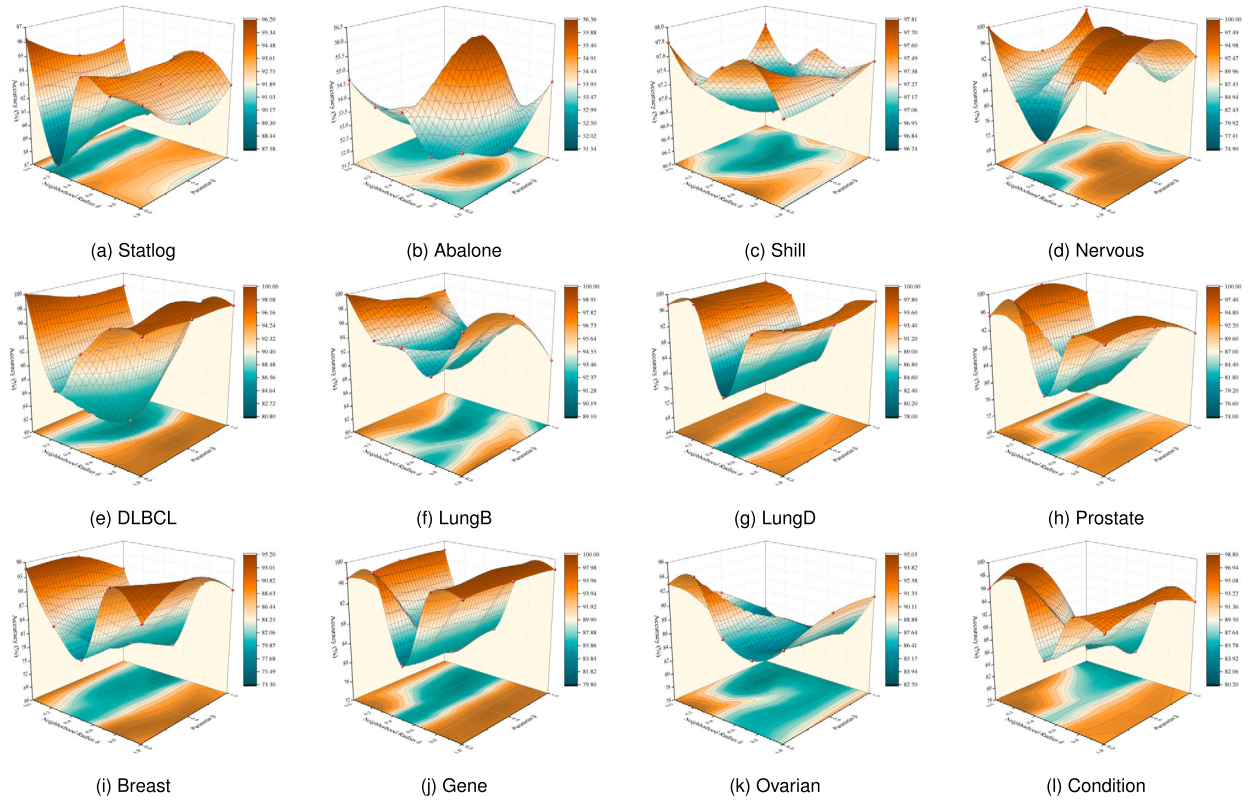

Fig. 3. Classification Accuracy of Different Parameters δ and β under RNN Classifier.

Table 4
Number of Selected Features by Different Algorithms when $\delta = 0.1, \beta = 0.6$.

Datasets	RAW	HS-NMR	MMBDE	EHHO	DSSA	HDFS	CHC_{QX}	ECWSA	OBCGWO	W-MGMN
Statlog	19	9	4	10	11	11	14	7	7	**3**
Abalone	9	2	4	5	6	6	**2**	**2**	4	4
Shill	14	3	3	4	3	5	7	4	3	**2**
Nervous	7130	1057	115	142	3446	3445	3586	1563	1392	**76**
DLBCL	7131	671	108	229	3396	3427	3562	**10**	1498	143
LungB	12533	445	69	1666	6131	5847	6248	364	2690	**25**
LungD	12600	425	188	171	6270	5590	6272	203	2780	**130**
Prostate	12601	391	233	289	6212	5753	6307	3210	2603	**193**
Breast	16185	3213	63	**19**	7872	7467	8066	3565	3739	46
Gene	20531	2165	203	744	7966	7496	8270	115	3654	**11**
Ovarian	37333	17048	312	**33**	18486	18263	9345	4338	11052	182
Condition	43680	8888	352	22272	21595	21247	16050	518	11287	**224**
Average	14147.08	2697.25	137.83	2132	6782.83	6546.42	5644.08	1558.28	3392	**86.58**

Subsequently, we assess and compare the feature selection and classification performance of W-MGMN and alternative algorithms across the four designated classifiers. The classification accuracy of each algorithm after feature selection is presented in Tables 5, 6, 7 and 8. The "RAW" column represents the original classification accuracy of the datasets before feature selection. The highest classification accuracy is shown in bold. The values on the left of "±" indicate the average, while the values on the right indicate the variance. Based on the results, it is evident that the W-MGMN algorithm consistently achieves higher classification accuracy compared to the original dataset. Moreover, in the majority of the datasets, W-MGMN outperforms the other six algorithms in terms of accuracy. Hence, W-MGMN effectively eliminates redundant features from the data and improves the classification accuracy.

Based on the number of selected features and the achieved classification accuracy, it can be inferred that W-MGMN performs efficient dimensionality reduction, identifies the optimal subset of features, and enhances the classification precision during data processing.

Table 5
Classification Accuracy of Different Algorithms under RNN Classifier when $\delta = 0.1, \beta = 0.6$.

Datasets	RAW	HS-NMR	MMBDE	EHHO	DSSA	HDFS	CHC_{QX}	ECWSA	OBCGWO	W-MGMN
Statlog	87.62±0.08	93.33±0.10	80.95±0.08	92.14±0.10	92.38±0.10	**95.95±0.11**	91.67±0.10	91.67±0.10	89.05±0.10	94.05±0.10
Abalone	51.91±0.01	54.82±0.04	52.51±0.04	51.91±0.04	54.67±0.05	54.19±0.04	51.08±0.03	52.75±0.04	51.79±0.05	**56.22±0.05**
Shill	92.41±0.14	97.24±0.24	91.23±0.19	**99.60±0.24**	99.05±0.24	99.45±0.24	98.18±0.24	98.50±0.24	98.42±0.24	97.39±0.23
Nervous	57.89±0.01	66.67±0.16	66.67±0.02	66.67±0.10	83.33±0.01	75.00±0.02	58.33±0.05	66.67±0.17	66.67±0.11	**91.67±0.15**
DLBCL	56.25±0.23	87.50±0.20	68.75±0.11	87.50±0.19	81.25±0.08	81.25±0.08	81.25±0.08	87.50±0.07	93.75±0.20	**99.75±0.23**
LungB	78.38±0.00	98.00±0.24	98.00±0.23	75.68±0.12	81.08±0.11	78.38±0.12	89.19±0.10	97.30±0.23	92.68±0.15	**99.30±0.23**
LungD	78.38±0.00	90.24±0.14	97.56±0.14	92.68±0.14	58.54±0.07	85.37±0.09	85.37±0.11	92.68±0.13	97.30±0.23	**100.0±0.00**
Prostate	38.10±0.00	76.19±0.14	94.00±0.15	80.95±0.14	47.62±0.00	66.67±0.00	57.14±0.00	72.38±0.02	90.48±0.19	**95.24±0.24**
Breast	57.89±0.00	73.68±0.11	47.37±0.07	52.63±0.01	89.63±0.00	42.11±0.01	42.11±0.00	82.11±0.01	63.16±0.16	**94.74±0.18**
Gene	40.37±0.01	95.00±0.16	95.76±0.16	95.00±0.16	92.80±0.03	30.43±0.01	38.51±0.01	95.76±0.16	96.00±0.16	**96.89±0.15**
Ovarian	60.47±0.00	67.44±0.16	79.07±0.17	74.42±0.06	85.81±0.00	55.81±0.00	88.37±0.05	81.40±0.14	69.77±0.17	**93.02±0.22**
Condition	65.75±0.05	80.11±0.09	90.06±0.19	69.06±0.03	89.06±0.04	65.75±0.03	87.96±0.02	91.16±0.20	91.16±0.18	**97.79±0.24**
Average	64.57±0.05	81.69±0.15	80.16±0.13	78.19±0.12	79.64±0.06	69.54±0.06	72.43±0.07	84.15±0.13	83.42±0.15	**93.01±0.17**

Table 6
Classification Accuracy of Different Algorithms under RF Classifier when $\delta = 0.1, \beta = 0.6$.

Datasets	RAW	HS-NMR	MMBDE	EHHO	DSSA	HDFS	CHC_{QX}	ECWSA	OBCGWO	W-MGMN
Statlog	86.95±0.12	85.86±2.55	76.19±1.74	87.86±2.69	86.57±2.12	83.10±2.35	87.33±2.57	86.81±0.94	91.71±0.33	**92.67±0.01**
Abalone	53.08±0.03	54.15±1.75	53.27±2.35	54.46±1.88	54.32±2.20	53.91±1.14	50.80±2.87	52.62±1.47	54.51±0.41	**55.64±0.00**
Shill	89.32±0.01	98.18±0.45	91.35±1.29	98.20±0.42	98.18±0.45	98.20±0.42	98.24±0.41	98.20±0.42	98.26±0.24	**98.37±0.00**
Nervous	58.33±1.81	61.67±1.00	**65.00±1.40**	**65.00±0.72**	58.33±3.44	63.33±6.33	61.67±0.15	63.33±0.53	60.00±1.28	**65.00±2.47**
DLBCL	76.61±0.99	76.61±2.71	**84.46±1.45**	80.54±0.47	78.04±1.98	81.79±1.16	79.11±0.78	81.79±0.39	81.79±0.16	80.54±1.03
LungB	97.30±1.81	98.33±1.55	96.73±3.56	98.89±2.22	98.89±2.22	98.89±0.22	98.89±2.22	93.39±0.54	91.17±0.41	**100.0±0.00**
LungD	78.31±0.36	85.81±0.41	86.29±0.99	85.76±0.69	86.69±4.39	86.14±6.23	86.17±5.36	78.29±0.78	**94.44±0.00**	90.14±0.24
Prostate	87.45±1.63	92.36±0.70	90.27±0.51	90.36±0.31	89.36±3.75	90.36±0.48	89.36±0.98	85.27±0.48	87.45±2.77	**92.27±1.05**
Breast	63.78±2.59	73.44±0.91	70.00±0.41	62.78±0.27	72.22±3.30	70.56±3.75	69.44±1.61	72.00±0.96	74.22±0.34	**74.56±1.59**
Gene	69.04±0.59	99.50±0.06	97.75±1.09	97.88±1.94	99.00±0.94	98.88±1.18	99.38±0.84	94.38±1.87	98.63±0.87	**99.63±0.00**
Ovarian	71.28±2.00	73.55±1.55	76.47±0.86	75.17±1.46	75.50±2.07	74.03±1.92	74.98±0.35	74.35±0.62	73.12±2.60	**76.75±1.77**
Condition	94.36±0.06	94.47±1.91	92.49±2.90	94.58±1.13	94.47±2.04	94.80±2.21	94.14±1.92	92.38±1.05	94.14±1.04	**95.69±0.03**
Average	77.15±1.00	82.83±1.30	81.69±1.50	82.62±1.18	82.63±2.49	81.67±2.15	82.46±1.67	81.07±0.84	81.04±0.87	**85.10±0.68**

Table 7
Classification Accuracy of Different Algorithms under NB Classifier when $\delta = 0.1, \beta = 0.6$.

Datasets	RAW	HS-NMR	MMBDE	EHHO	DSSA	HDFS	CHC_{QX}	ECWSA	OBCGWO	W-MGMN
Statlog	69.76±0.07	75.38±2.38	71.86±2.78	83.05±3.20	76.52±2.48	80.05±2.82	78.90±2.92	81.95±1.97	**83.33±0.27**	81.76±0.05
Abalone	44.71±0.79	51.00±0.05	51.43±2.85	51.53±1.48	51.50±1.70	51.71±1.60	48.86±2.39	50.17±0.31	50.96±0.10	**51.74±0.04**
Shill	97.04±0.00	**98.04±0.43**	90.95±1.24	**98.04±0.43**	**98.04±0.43**	97.04±0.43	97.02±0.68	97.04±0.42	98.04±0.43	97.28±0.00
Nervous	66.67±2.22	68.00±0.75	65.00±1.72	68.00±0.26	60.00±5.28	61.67±2.42	58.33±0.17	70.00±0.00	69.67±0.33	**70.00±4.33**
DLBCL	78.93±2.17	83.86±0.73	79.11±1.14	85.36±1.20	79.11±3.00	80.36±3.15	78.93±3.62	84.14±0.19	78.93±1.75	**85.71±1.79**
LungB	94.59±0.00	97.28±0.60	96.70±3.67	98.39±0.40	98.89±2.22	98.36±0.50	94.44±1.67	95.09±0.15	92.69±0.49	**98.92±0.05**
LungD	89.67±0.07	88.81±0.24	92.79±1.02	87.81±0.65	89.69±4.01	90.69±3.96	90.67±3.38	92.74±0.22	91.87±0.48	**93.62±0.29**
Prostate	62.82±1.86	76.82±0.03	75.82±0.65	77.73±0.64	72.82±3.65	62.82±3.65	61.82±3.45	76.82±0.03	85.82±3.65	**95.27±0.92**
Breast	62.78±3.46	80.89±0.79	79.00±0.72	66.56±0.60	80.89±0.79	63.89±0.63	62.78±1.65	70.22±0.09	80.89±1.36	**81.00±1.98**
Gene	68.29±0.40	90.26±0.62	93.13±1.89	93.13±2.23	78.28±2.39	79.03±4.65	79.14±3.30	93.01±0.83	90.77±0.34	**93.50±0.07**
Ovarian	65.06±0.96	73.61±0.06	72.61±0.69	76.58±0.04	73.54±0.60	65.52±3.84	71.70±0.46	71.68±0.94	72.64±0.61	**89.70±0.40**
Condition	78.79±0.08	80.45±2.34	77.13±1.88	80.12±2.37	79.34±2.66	79.90±2.96	80.23±2.99	82.11±0.23	82.89±0.65	**87.73±0.19**
Average	73.30±1.01	80.37±0.75	78.88±1.69	80.53±1.13	77.47±2.29	74.54±2.44	75.24±2.23	80.56±0.45	81.54±0.87	**85.52±0.84**

5.3. Performance verification of the W-MGMA algorithm

In this subsection, we will compare the computational efficiency and feature selection capability differences between W-MGMN and W-MGMA, aiming to evaluate the effectiveness and efficiency of W-MGMA.

Prior to the comparison, we divided each preprocessed dataset into an initial object set, U, and an object addition set, U'. We then randomly selected objects from U' at different proportions $(10\%, 20\%, 30\%, 40\%, 50\%)$ and added them to U, resulting in dynamic test datasets. Subsequently, we employed various neighborhood radius δ and parameters, β, to calculate the computation time of the W-MGMA operation. The range for δ was chosen from 0 to 1 with an increment of 0.2, while β ranged from 0.6 to 1 with an increment of 0.2. The results are depicted in Fig. 4. Upon analyzing the runtime results across the 12 datasets, we observed that the computation time was longest and most compelling when δ ranged from 0 to 0.2 and β was set to 0.6. Therefore, for subsequent experiments, we selected $\delta = 0.1$ and $\beta = 0.6$ as the parameter values.

Table 8
Classification Accuracy of Different Algorithms under DT Classifier when $\delta = 0.1, \beta = 0.6$.

Datasets	RAW	HS-NMR	MMBDE	EHHO	DSSA	HDFS	CHC_{QX}	ECWSA	OBCGWO	W-MGMN
Statlog	54.67±0.03	88.95±1.59	84.00±1.94	**91.10±1.41**	89.62±1.78	89.71±2.11	89.04±0.19	89.38±2.02	90.81±0.51	82.43±0.05
Abalone	44.17±0.77	53.24±0.99	53.68±2.11	53.17±1.98	53.20±1.63	53.41±1.09	51.04±2.60	53.41±1.09	55.04±0.01	**55.11±0.07**
Shill	97.28±0.00	98.29±0.49	91.79±1.37	99.45±0.33	99.51±0.26	99.43±0.31	99.53±0.22	99.45±0.11	99.51±0.26	**99.57±0.00**
Nervous	50.00±6.11	64.00±1.79	55.09±0.28	63.33±2.60	**66.67±0.82**	50.00±2.54	60.00±0.20	60.00±1.34	66.67±2.87	**66.67±3.33**
DLBCL	78.04±2.37	85.96±1.00	76.79±2.85	82.50±3.58	74.11±0.56	77.50±2.38	85.54±2.15	84.46±0.75	83.29±0.29	**86.79±1.54**
LungB	91.75±0.31	93.42±1.84	93.78±3.69	94.51±0.05	94.53±0.33	94.42±0.78	**97.28±0.36**	86.26±0.73	95.14±0.45	94.53±0.23
LungD	75.36±0.55	87.29±0.48	89.07±0.60	86.21±0.77	83.26±5.90	**89.10±0.72**	88.10±0.52	83.24±0.98	88.09±0.15	**89.10±0.65**
Prostate	86.36±1.20	86.45±0.99	85.36±0.35	83.64±0.31	87.36±2.26	75.55±1.86	82.55±0.03	86.45±0.84	87.45±0.08	**88.27±0.74**
Breast	62.11±2.86	61.11±1.41	66.22±0.09	64.11±0.94	64.67±0.10	**68.33±0.49**	60.11±3.86	64.11±1.41	61.00±1.99	64.22±4.76
Gene	87.51±0.06	95.38±1.38	92.76±2.14	96.01±0.01	97.88±2.31	97.25±2.25	98.13±1.60	87.89±0.86	97.00±0.32	**98.00±0.04**
Ovarian	59.13±1.10	72.21±1.02	70.99±1.29	70.95±0.08	71.32±0.32	60.00±3.66	68.40±0.16	72.56±1.21	72.06±0.15	**73.72±1.24**
Condition	93.59±0.06	94.25±1.32	93.26±2.08	93.70±0.15	93.58±2.43	94.28±2.19	94.03±2.55	93.59±0.67	94.15±0.85	**94.36±0.05**
Average	73.33±1.29	81.71±1.43	79.40±1.57	81.56±0.93	81.35±1.56	77.49±1.61	81.15±1.2	80.07±1.00	82.52±0.66	**82.73±1.06**

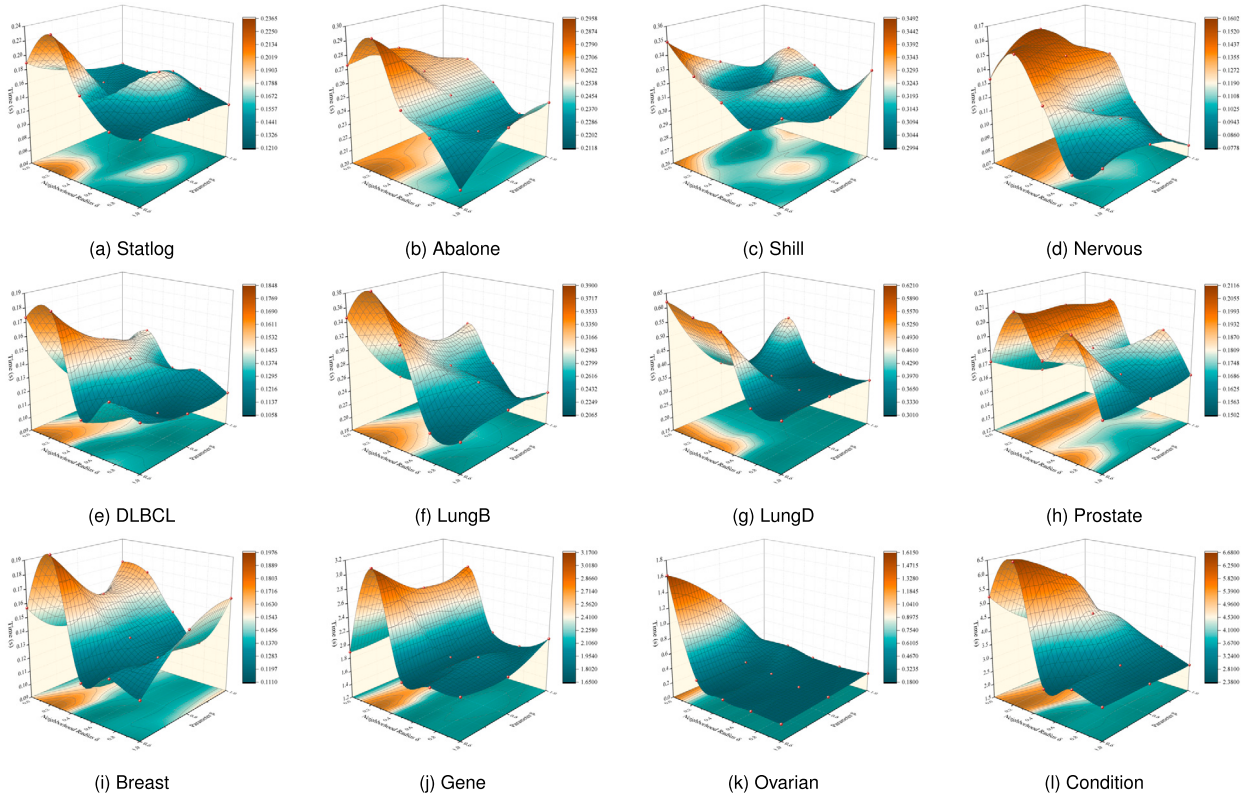

Fig. 4. Computation Time of Different Parameters δ and β When Adding Objects.

Subsequently, utilizing the 12 datasets, we computed the runtime for adding objects at proportions of 10%, 20%, 30%, 40%, and 50%, as depicted in Table 9. Fig. 6 provides a visual representation of the contrasting runtime between the two algorithms. Upon analyzing the outcomes, we noted that across all 12 datasets, irrespective of their object quantity or attribute quantity, the computation time of W-MGMA was significantly lower compared to W-MGMN, thus affirming the superior efficacy of the W-MGMA. Furthermore, we observed that the computation efficiency of W-MGMA did not exhibit linear fluctuations with dataset size.

Tables 10 and 11 respectively demonstrate the classification accuracy of W-MGMA's feature selection during object addition under RNN, RF, NB, and DT classifiers. From these results, we observe that in the majority of datasets, the classification accuracy of W-MGMA is comparable to that of W-MGMN and, at times, even surpasses the classification accuracy of W-MGMN. Hence, we can deduce that W-MGMA is highly effective.

Based on the efficiency and effectiveness of the W-MGMA, it can be inferred that the computational time for feature selection is considerably lower than that of W-MGMN. Additionally, the chosen outcomes are comparable to, and in some cases even superior to, those of W-MGMN. Consequently, when adding objects to a NIS, the incremental algorithm W-MGMA can efficiently and effectively conduct feature selection.

Table 9
Runtime of Different Algorithms as Objects Increase when $\delta = 0.1, \beta = 0.6$.

Datasets	10%		20%		30%		40%		50%	
	W-MGMN	W-MGMA	W-MGMN	W-MGMA	W-MGMN	W-MGMA	W-MGMN	W-MGMA	W-MGMN	W-MGMA
Statlog	0.826	0.076	1.258	0.106	1.511	0.132	1.833	0.159	2.136	0.181
Abalone	3.816	0.119	4.534	0.191	5.064	0.261	5.547	0.282	6.003	0.335
Shill	1.977	0.185	2.283	0.234	2.617	0.327	3.219	0.361	3.798	0.418
Nervous	2.251	0.696	2.774	0.893	3.049	0.942	3.598	1.050	4.121	1.112
DLBCL	3.839	0.090	4.352	0.098	4.970	0.110	5.969	0.121	6.809	0.171
LungB	6.426	0.505	7.574	0.670	8.344	0.680	9.597	0.743	11.33	0.880
LungD	7.016	0.211	8.974	0.252	9.968	0.331	11.62	0.455	14.04	0.904
Prostate	7.053	0.097	8.149	0.153	9.424	0.247	10.07	0.362	11.70	0.484
Breast	4.955	0.095	6.696	0.131	7.833	0.160	8.663	0.393	9.495	0.499
Gene	175.1	2.436	247.6	4.981	350.0	5.693	459.1	6.409	511.7	7.180
Ovarian	33.04	0.447	44.09	0.456	52.81	1.740	64.23	2.197	74.85	2.654
Condition	398.4	8.649	586.9	11.41	690.9	14.59	970.8	16.30	1159	20.11

Table 10
Classification Accuracy of W-MGMA under Different Classifiers when $\delta = 0.1, \beta = 0.6$.

Datasets	RNN (Adding Objects)					RF (Adding Objects)				
	10%	20%	30%	40%	50%	10%	20%	30%	40%	50%
Statlog	92.62±0.10	91.90±0.10	92.14±0.10	**93.71±0.00**	93.39±0.07	92.43±0.03	92.57±0.04	92.48±0.02	**92.81±0.03**	92.19±0.02
Abalone	51.20±0.04	54.31±0.04	54.43±0.04	55.62±0.05	**57.54±0.05**	55.59±0.07	54.99±0.06	55.33±0.07	55.45±0.05	**55.78±0.05**
Shill	97.31±0.23	97.08±0.24	97.39±0.23	97.31±0.23	**97.87±0.23**	98.31±0.00	98.29±0.00	98.28±0.00	98.26±0.00	**98.31±0.00**
Nervous	84.21±0.12	89.47±0.11	83.33±0.05	91.67±0.17	**94.74±0.19**	78.33±3.36	**81.67±3.58**	80.00±3.22	80.00±3.22	80.00±2.67
DLBCL	87.50±0.10	87.50±0.05	**100.0±0.00**	85.00±0.06	87.50±0.07	**90.89±1.08**	86.79±2.17	85.36±1.57	88.21±1.51	89.64±1.62
LungB	94.59±0.10	91.89±0.16	94.59±0.16	97.30±0.13	**100.0±0.00**	99.44±0.03	**100.0±0.00**	100.0±0.00	99.44±0.03	100.0±0.00
LungD	94.59±0.10	95.12±0.15	95.12±0.15	92.68±0.12	**100.0±0.00**	90.17±0.28	90.69±0.15	89.19±0.23	91.67±0.18	**92.17±0.19**
Prostate	90.48±0.19	89.47±0.11	90.48±0.13	**100.0±0.00**	100.0±0.00	93.36±1.38	**94.27±0.31**	94.27±0.93	94.27±0.27	94.27±0.37
Breast	84.21±0.12	**94.74±0.16**	84.21±0.17	89.47±0.16	94.74±0.19	84.33±0.93	85.44±0.65	84.56±1.26	82.22±1.00	**87.67±1.15**
Gene	91.99±0.12	91.30±0.14	94.41±0.13	93.17±0.13	**95.65±0.14**	**93.38±0.08**	90.63±0.20	91.26±0.14	90.63±0.18	92.76±0.09
Ovarian	88.37±0.21	83.72±0.21	90.70±0.21	91.24±0.24	**95.35±0.22**	82.81±1.14	83.38±0.69	83.83±0.79	**89.29±0.18**	87.90±0.23
Condition	92.82±0.23	93.37±0.23	**98.90±0.24**	97.24±0.23	96.69±0.23	94.36±0.03	94.36±0.03	94.37±0.04	**94.47±0.03**	94.25±0.02

Table 11
Classification Accuracy of W-MGMA under Different Classifiers when $\delta = 0.1, \beta = 0.6$.

Datasets	NB (Adding Objects)					DT (Adding Objects)				
	10%	20%	30%	40%	50%	10%	20%	30%	40%	50%
Statlog	74.67±0.14	78.86±0.12	85.29±0.07	87.43±0.09	**88.52±0.09**	82.24±0.06	82.43±0.06	82.29±0.06	82.24±0.06	**84.19±0.06**
Abalone	52.05±0.05	52.00±0.04	**52.96±0.04**	52.96±0.04	52.05±0.03	55.06±0.07	55.06±0.08	**55.09±0.08**	55.02±0.08	55.04±0.07
Shill	97.28±0.00	97.28±0.00	97.21±0.00	**97.36±0.00**	97.28±0.00	99.56±0.00	99.56±0.00	99.31±0.00	**99.57±0.00**	99.57±0.00
Nervous	**78.33±3.92**	70.00±7.67	75.00±3.47	75.00±4.58	70.00±4.33	70.00±3.78	78.33±3.36	73.33±2.33	80.00±3.22	**80.00±4.89**
DLBCL	92.13±0.14	90.14±0.51	93.39±0.75	92.31±0.41	**93.39±0.75**	81.96±2.20	80.54±2.28	83.04±1.94	84.46±2.20	**86.79±1.67**
LungB	**98.92±0.05**	98.92±0.05	98.92±0.05	98.92±0.05	98.92±0.05	**99.47±0.02**	99.47±0.02	98.92±0.05	98.92±0.05	99.47±0.02
LungD	85.14±0.36	86.81±0.81	**93.22±0.49**	90.37±0.15	91.32±0.66	**80.76±0.67**	76.33±0.39	80.74±0.53	79.81±0.51	80.71±0.88
Prostate	94.25±0.61	94.33±0.40	95.57±0.93	94.19±0.27	**95.91±0.37**	81.45±0.44	82.36±0.35	84.45±0.36	82.36±0.55	**86.27±0.57**
Breast	91.47±0.30	90.39±0.27	90.02±0.30	90.52±0.41	**91.85±0.39**	71.67±0.67	72.78±1.36	75.78±0.47	74.00±2.42	**76.00±0.79**
Gene	97.25±0.12	96.14±0.08	97.75±0.02	98.01±0.24	**98.13±0.81**	87.88±0.18	**87.89±0.18**	87.76±0.20	87.26±0.20	87.88±0.18
Ovarian	88.38±0.37	91.51±0.37	91.01±0.16	91.84±0.23	**92.91±0.18**	72.16±0.65	**73.07±0.49**	72.16±0.65	73.07±0.49	73.07±0.49
Condition	**87.73±0.19**	87.73±0.19	87.73±0.19	87.73±0.19	87.73±0.19	93.15±0.06	93.37±0.05	93.48±0.05	93.48±0.05	**93.59±0.05**

5.4. Performance verification of the W-MGMD algorithm

This subsection aims to analyze and compare the differences in computational efficiency and feature selection capabilities between W-MGMN and W-MGMA, thereby evaluating the effectiveness and efficiency of W-MGMD. The experimental methodology is similar to the previous subsection.

Before conducting the comparison, we start with each preprocessed dataset as the initial object set, denoted as U. Then, we randomly select objects from U at different proportions $(10\%, 20\%, 30\%, 40\%, 50\%)$ and remove them from U, resulting in dynamic testing datasets. Next, we calculate the runtime of W-MGMA operation using different neighborhood radius δ and parameter β. We consider a range of 0 to 1 for δ with a step size of 0.2, and a range of 0.6 to 1 for parameter β with a step size of 0.2. Subsequently, we present the results in Fig. 5. Analyzing the outcomes from the 12 datasets, we observe that the longest runtime occurs when δ is

Table 12
Runtime of Different Algorithms as Objects Decrease when $\delta = 0.1, \beta = 0.6$.

Datasets	10%		20%		30%		40%		50%	
	W-MGMN	W-MGMD	W-MGMN	W-MGMD	W-MGMN	W-MGMD	W-MGMN	W-MGMD	W-MGMN	W-MGMD
Statlog	1.524	0.153	1.227	0.129	0.937	0.104	0.744	0.083	0.593	0.070
Abalone	5.804	0.330	5.018	0.251	4.226	0.203	3.388	0.122	3.180	0.068
Shill	3.563	0.376	2.737	0.326	2.324	0.281	1.977	0.225	1.747	0.151
Nervous	4.562	0.526	4.174	0.478	3.679	0.432	3.061	0.374	2.321	0.335
DLBCL	5.624	0.071	4.874	0.046	4.348	0.047	3.933	0.043	3.439	0.035
LungB	10.34	0.751	9.329	0.670	8.582	0.618	7.314	0.535	6.325	0.276
LungD	12.70	0.542	11.24	0.325	9.242	0.231	8.863	0.220	7.360	0.188
Prostate	11.68	0.104	10.15	0.101	9.227	0.071	8.540	0.062	7.184	0.044
Breast	9.079	0.095	8.085	0.067	7.683	0.053	6.504	0.045	4.830	0.042
Gene	509.3	8.565	444.5	7.442	341.2	5.161	250.9	4.041	168.4	3.436
Ovarian	71.89	1.754	61.30	1.473	50.95	0.845	42.38	0.720	29.42	0.432
Condition	1036	19.47	940.8	16.19	668.7	14.05	543.7	10.78	334.2	8.054

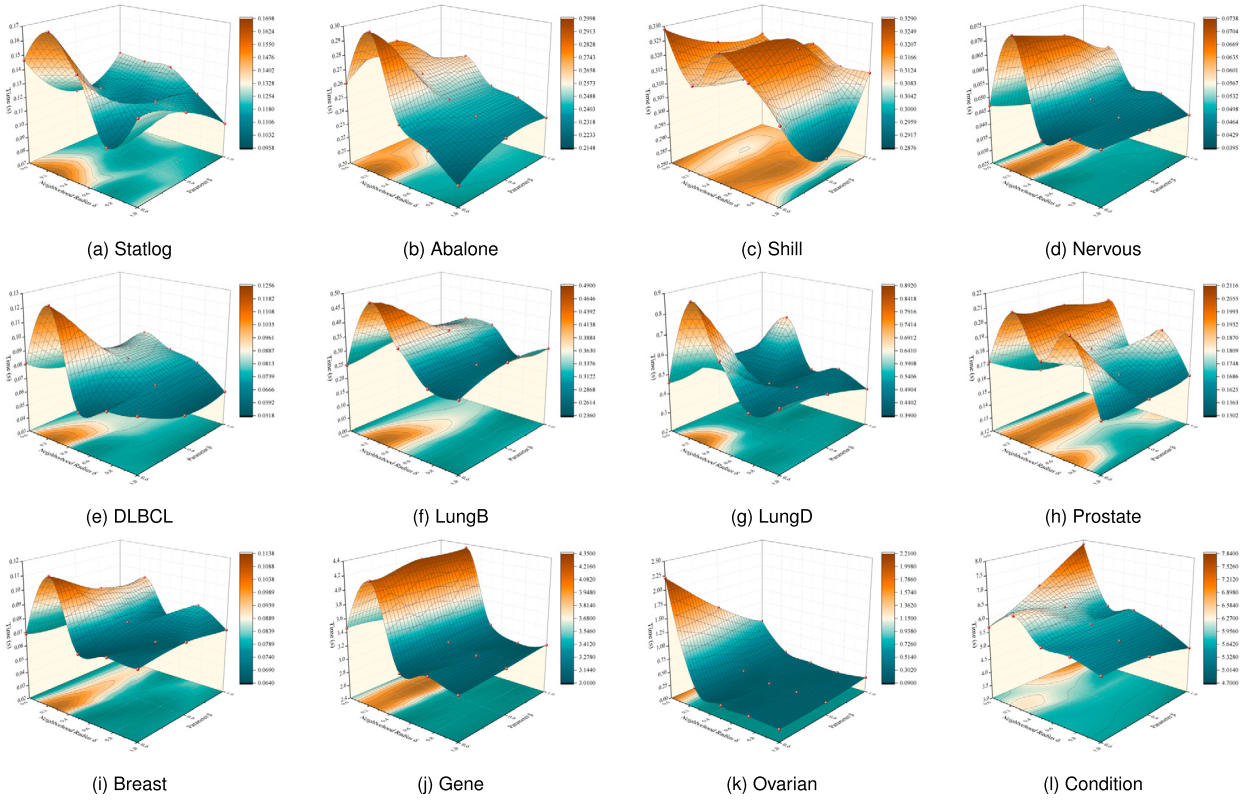

Fig. 5. Computation Time of Different Parameters δ and β When Deleting Objects.

between 0 and 0.2, and β is set to 0.6, which provides the most convincing results. Consequently, for the subsequent experiments, we select $\delta = 0.1$ and $\beta = 0.6$ as the parameter settings.

Table 12 presents the runtime for removing objects at different proportions $(10\%, 20\%, 30\%, 40\%, 50\%)$ across the 12 datasets. These results are visually represented in Fig. 7, allowing for a clear comparison of the disparities in runtime between the two algorithms. Based on the analysis of the results, we can observe that for all 12 datasets, regardless of whether they have a large number of objects or a high number of attributes, W-MGMD consistently exhibits significantly lower computation time compared to W-MGMN. This substantiates the W-MGMD outperforms the W-MGMN in terms of efficiency. Similarly, the computational efficiency of the W-MGMD does not follow a linear relationship with the size of the dataset.

The classification accuracy of feature selection using W-MGMD under RNN, RF, NB, and DT classifiers during objects removal is presented in Tables 13 and 14. Upon analyzing the results in the tables, we can observe that in most datasets, the classification

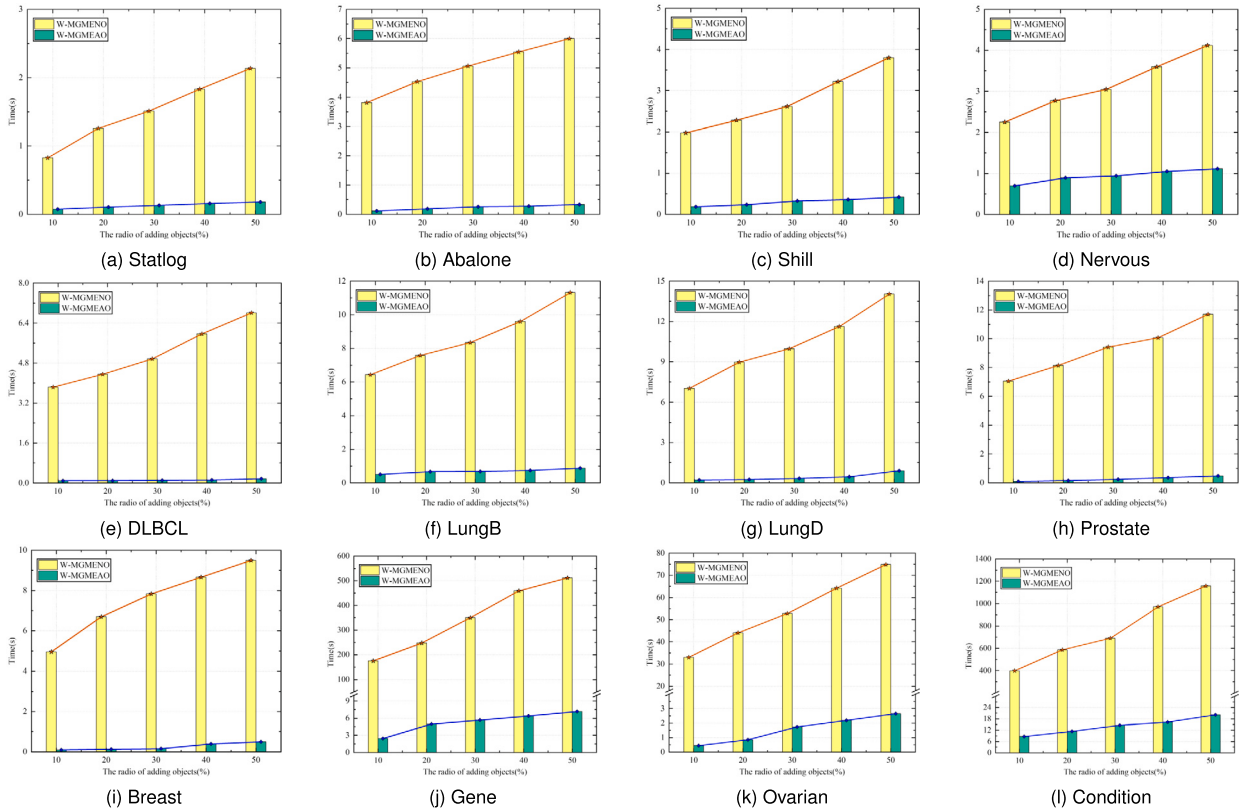

Fig. 6. Computation Time of Different Algorithms when Adding Objects ($\delta = 0.1, \beta = 0.6$).

accuracy of feature selection using W-MGMD is comparable to, and sometimes even surpasses, that of W-MGMN. Therefore, we can conclude that the W-MGMD is effective in terms of feature selection.

Based on the analysis of efficiency and effectiveness between W-MGMD and W-MGMN, we can draw the following conclusions: W-MGMD demonstrates significantly reduced runtime during feature selection compared to the W-MGMN, while similar, or even superior selection outcomes. Therefore, when the objective entails removing objects from a NIS, W-MGMD proves to be an efficient and effective method for feature selection. It accomplishes the task within a shorter time and the selected features exhibit no significant degradation compared to W-MGMN, and in some cases, may even outperform it.

5.5. Statistical testing

To enhance the comparison of various algorithms' experimental outcomes, we utilized both the Friedman and Wilcoxon test to assess the validity of the algorithmic comparisons.

The Friedman test is a non-parametric hypothesis test used to compare whether the means of multiple related samples are equal. Its null hypothesis posits that all algorithms exhibit comparable classification performance. The formula is defined as

$$\chi_F^2 = \frac{12n}{k(k+1)}\left(\sum_{j=1}^{k} R_j^2 - \frac{k(k+1)^2}{4}\right),$$

where n and k indicate the number of samples and algorithms being compared respectively, R_i denotes the average ranking of the classification accuracy results for algorithm i across various classifiers.

Table 15 shows the average rankings of the HS-NMR, MMBD, EHHO, DSSA, HDFS, CHC, and W-MGMN algorithms under the RNN, RF, NB, and DT classifiers. It is evident that W-MGMN consistently ranks first across all four classifiers, demonstrating the superiority of the proposed algorithm.

Table 16 displays the results of the Wilcoxon test used to assess the relative performance and differences between W-MGMN and eight other algorithms. The significance level was set at 0.05. If the P-value exceeds 0.05, it suggests that the performance difference between the two algorithms is not significant. Conversely, if the P-value is below 0.05, it indicates a significant difference in performance between the two algorithms. As evident from the table, all P-values are under 0.05, demonstrating statistically significant differences between W-MGMN and the other eight algorithms.

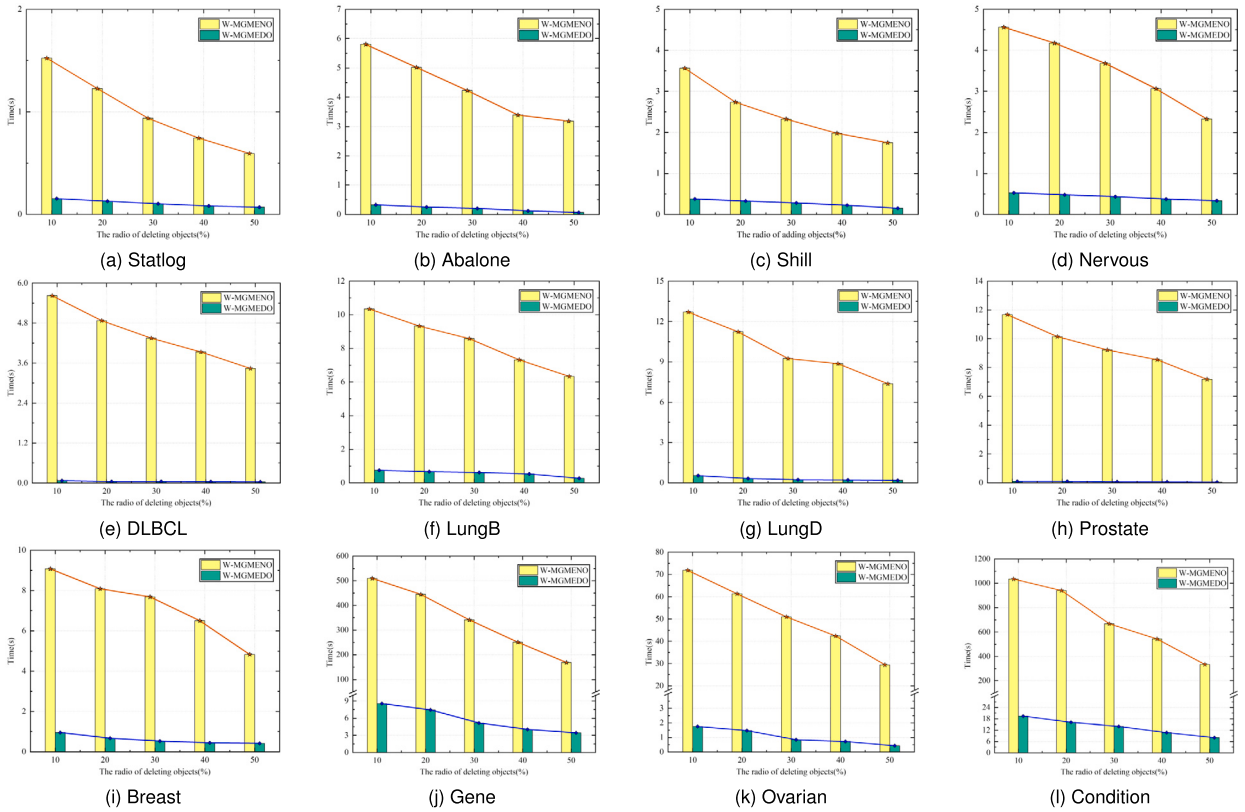

Fig. 7. Computation Time of Different Algorithms when Deleting Objects ($\delta = 0.1, \beta = 0.6$).

Table 13
Classification Accuracy of W-MGMD under Different Classifiers when $\delta = 0.1, \beta = 0.6$.

Datasets	RNN (Deleting Objects)					RF (Deleting Objects)				
	10%	20%	30%	40%	50%	10%	20%	30%	40%	50%
Statlog	**95.24±0.11**	93.33±0.11	93.57±0.11	91.81±0.10	90.00±0.08	92.10±0.02	**92.67±0.03**	92.33±0.03	92.43±0.02	91.62±0.04
Abalone	**58.97±0.05**	52.27±0.04	56.82±0.05	54.19±0.05	52.75±0.04	55.30±0.07	55.45±0.07	**55.61±0.06**	55.28±0.06	55.14±0.06
Shill	97.55±0.23	**97.71±0.23**	97.41±0.22	97.31±0.23	97.49±0.23	98.34±0.00	98.26±0.00	98.32±0.00	98.29±0.00	**98.39±0.00**
Nervous	89.47±0.14	78.95±0.14	**94.74±0.15**	84.21±0.15	89.47±0.15	78.36±0.06	78.33±2.25	**80.00±3.22**	78.33±2.25	76.67±2.33
DLBCL	**93.75±0.05**	93.75±0.09	87.50±0.06	93.75±0.08	81.25±0.12	90.89±1.08	**93.57±0.73**	89.82±1.20	85.36±2.38	88.04±1.29
LungB	97.30±0.12	97.30±0.14	**100.0±0.00**	100.0±0.00	94.59±0.13	99.44±0.03	98.36±0.06	96.70±0.07	**99.44±0.03**	98.36±0.06
LungD	95.12±0.16	95.12±0.15	**97.56±0.16**	90.56±0.15	**92.68±0.15**	92.14±0.10	90.67±0.26	89.67±0.16	91.19±0.17	89.67±0.21
Prostate	85.71±0.15	90.48±0.17	**95.24±0.18**	95.24±0.19	90.48±0.16	94.36±1.38	**95.18±0.60**	93.27±0.89	94.27±0.67	94.27±0.26
Breast	**94.74±0.15**	92.45±0.02	89.47±0.17	84.21±0.15	89.47±0.15	**89.56±0.60**	86.33±0.96	87.56±0.79	82.22±1.39	87.44±0.59
Gene	90.06±0.13	**97.52±0.14**	93.79±0.13	91.30±0.13	91.93±0.13	91.38±0.14	**91.76±0.12**	91.51±0.14	91.51±0.17	89.76±0.08
Ovarian	**97.67±0.22**	86.05±0.20	88.37±0.22	90.70±0.21	93.02±0.21	85.17±0.57	84.29±0.85	83.29±0.29	**87.45±0.43**	83.83±0.53
Condition	**97.24±0.23**	97.24±0.23	96.69±0.23	90.61±0.22	94.48±0.23	**94.70±0.03**	94.58±0.04	94.70±0.04	94.47±0.03	94.47±0.03

6. Conclusions and future work

With the rapid advancement of technology, there has been a significant increase in the size of data, leading to a proliferation of redundant features. Effectively processing such large-scale data poses significant challenges to traditional rough set theory. To address this, multigranulation rough sets offer a rational approach by analyzing and handling problems from multiple levels and granularities. Feature selection, as a dimensionality reduction technique, aims to identify the most informative subset of features from the original data, thereby enhancing the performance and effectiveness of machine learning models. It has emerged as an efficient information preprocessing method. In this paper, we explore the incremental feature selection method using a matrix-based approach within the W-GMNRS framework. This method addresses the shortcomings of existing rough set models in handling dynamic datasets, such as slow runtime and lackluster performance. It enhances the efficiency of feature selection, conserves computational resources, and strengthens the application of rough set theory models. To validate this method, we conducted experiments on 12 publicly available datasets, and the results demonstrate the following:

Table 14
Classification Accuracy of W-MGMD under Different Classifiers when $\delta = 0.1, \beta = 0.6$.

Datasets	NB (Deleting Objects)					DT (Deleting Objects)				
	10%	20%	30%	40%	50%	10%	20%	30%	40%	50%
Statlog	83.87±0.05	**85.90±0.10**	78.86±0.12	78.76±0.08	77.19±0.03	**82.38±0.06**	82.38±0.06	82.29±0.06	82.33±0.06	82.29±0.06
Abalone	52.05±0.03	52.00±0.04	52.05±0.03	**52.45±0.02**	52.00±0.04	**55.06±0.08**	55.04±0.08	55.06±0.08	55.04±0.07	55.02±0.08
Shill	**97.28±0.00**	97.23±0.00	97.22±0.00	97.28±0.00	97.21±0.00	**99.56±0.00**	99.54±0.00	99.49±0.00	99.42±0.00	99.36±0.00
Nervous	77.28±0.13	71.67±5.03	73.33±4.00	71.67±3.92	**78.33±2.25**	70.92±0.05	75.00±5.14	**81.67±1.92**	78.33±0.58	66.67±1.11
DLBCL	**93.39±0.75**	93.39±0.75	93.39±0.75	93.39±0.75	93.39±0.75	85.71±2.10	83.39±1.87	**85.89±1.79**	83.04±1.94	81.79±3.00
LungB	**98.92±0.05**	97.28±0.13	97.28±0.13	97.28±0.13	97.81±0.07	**98.92±0.05**	98.92±0.05	96.14±0.13	95.58±0.11	96.14±0.13
LungD	92.45±0.74	92.22±0.12	**93.03±0.76**	91.32±0.65	90.77±0.63	**81.29±0.62**	79.76±0.93	79.31±0.58	80.65±0.45	80.74±0.48
Prostate	95.92±0.47	95.49±0.05	95.82±0.58	**95.96±0.67**	94.54±0.26	**87.27±0.59**	81.45±1.44	86.36±0.60	83.36±0.79	82.36±0.75
Breast	**91.65±0.26**	91.44±0.28	90.97±0.23	90.55±0.23	91.39±0.50	**76.00±1.08**	73.89±0.65	74.89±1.35	73.89±2.19	75.89±1.05
Gene	96.13±0.01	96.13±0.01	**97.75±0.02**	97.13±0.02	91.26±0.06	**88.01±0.19**	87.63±0.21	87.64±0.18	87.26±0.20	82.89±0.13
Ovarian	**92.35±0.37**	92.04±0.37	90.56±0.37	89.84±0.37	88.63±0.37	**73.07±0.49**	72.64±0.71	73.07±0.49	73.07±0.49	72.64±0.71
Condition	**87.73±0.19**	87.73±0.19	87.73±0.19	87.73±0.19	87.73±0.19	**93.48±0.05**	93.37±0.05	93.26±0.05	93.37±0.05	93.15±0.05

Table 15
Average Rank of the Classification Accuracy of Different Algorithms $\delta = 0.1, \beta = 0.6$.

Classifiers	HS-NMR	MMBDE	EHHO	DSSA	HDFS	CHC_{QX}	ECWSA	OBCGWO	W-MGMN
RNN	5.08	5.42	5.50	5.08	6.25	6.96	4.42	4.71	1.58
RF	5.54	5.75	4.71	5.25	4.95	5.58	6.71	4.88	1.63
NB	4.88	6.00	3.88	5.66	6.21	7.63	4.79	4.33	1.63
DT	5.46	6.42	5.83	5.00	5.25	5.21	5.79	3.71	2.33

Table 16
P Value of the Wilcoxon Test when $\delta = 0.1, \beta = 0.6$.

Classifiers	HS-NMR	MMBDE	EHHO	DSSA	HDFS	CHC_{QX}	ECWSA	OBCGWO
RNN	0.0005	0.0005	0.0024	0.0015	0.0034	0.0009	0.0010	0.0015
RF	0.0009	0.0262	0.0050	0.0004	0.0049	0.0005	0.0014	0.0050
NB	0.0024	0.0004	0.0269	0.0034	0.0005	0.0004	0.0040	0.0012
DT	0.0342	0.0269	0.0342	0.0141	0.0754	0.0129	0.0200	0.0109

1. When applying W-MGMN for feature selection, the obtained classification results not only surpass those of the original dataset but also outperform other feature selection algorithms. This demonstrates the strong feature selection capability of W-MGMN.
2. The proposed dynamic feature selection algorithms demonstrates good classification results when there are additions or removals of objects. They also exhibit significantly lower runtime compared to W-MGMN, validating the effectiveness and efficiency of the dynamic algorithms.

Incremental feature selection algorithms are extensively applied across various sectors due to their practicality and versatility. Our goal is to deploy our algorithm in widely encountered scenarios, such as medical diagnostics, financial risk evaluation, and industrial monitoring. By extracting the most relevant features from vast amounts of dynamic information, our algorithm helps professionals better analyze data.

When information is updated, the number of features may increase or decrease. In such cases, the proposed method becomes ineffective. Therefore, based on the findings of this study, further exploration into incremental update mechanisms when the number of features changes could enhance the model's adaptability, allowing it to perform better feature selection in the face of changes in both samples and features.

CRediT authorship contribution statement

Weihua Xu: Validation, Project administration, Methodology, Investigation, Funding acquisition, Conceptualization. **Qinyuan Bu:** Data curation, Methodology, Software, Visualization, Writing – original draft.

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.

Data availability

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

Acknowledgement

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

References

[1] X. Yang, H. Chen, T. Li, C. Luo, A noise-aware fuzzy rough set approach for feature selection, Knowl.-Based Syst. 250 (2022) 109092.
[2] W. Xu, M. Huang, Z. Jiang, Y. Qian, Graph-based unsupervised feature selection for interval-valued information system, IEEE Trans. Neural Netw. Learn. Syst. (2023).
[3] W. Li, H. Zhou, W. Xu, X.-Z. Wang, W. Pedrycz, Interval dominance-based feature selection for interval-valued ordered data, IEEE Trans. Neural Netw. Learn. Syst. (2022).
[4] W. Xu, Y. Pan, X. Chen, W. Ding, Y. Qian, A novel dynamic fusion approach using information entropy for interval-valued ordered datasets, IEEE Trans. Big Data (2022).
[5] Z. Deng, T. Li, D. Deng, K. Liu, P. Zhang, S. Zhang, Z. Luo, Feature selection for label distribution learning using dual-similarity based neighborhood fuzzy entropy, Inf. Sci. 615 (2022) 385–404.
[6] F. Jiang, Y. Sui, C. Cao, An information entropy-based approach to outlier detection in rough sets, Expert Syst. Appl. 37 (9) (2010) 6338–6344.
[7] S. Wang, Z. Yuan, C. Luo, H. Chen, D. Peng, Exploiting fuzzy rough entropy to detect anomalies, Int. J. Approx. Reason. (2023) 109087.
[8] Y. Yao, et al., Granular computing: basic issues and possible solutions, in: Proceedings of the 5th Joint Conference on Information Sciences, vol. 1, 2000, pp. 186–189.
[9] Q. Kong, X. Zhang, W. Xu, B. Long, A novel granular computing model based on three-way decision, Int. J. Approx. Reason. 144 (2022) 92–112.
[10] X. Zhang, D. Guo, W. Xu, Two-way concept-cognitive learning with multi-source fuzzy context, Cogn. Comput. (2023) 1–23.
[11] W. Xu, D. Guo, J. Mi, Y. Qian, K. Zheng, W. Ding, Two-way concept-cognitive learning via concept movement viewpoint, IEEE Trans. Neural Netw. Learn. Syst. (2023).
[12] D. Guo, W. Xu, Y. Qian, W. Ding, M-fccl: memory-based concept-cognitive learning for dynamic fuzzy data classification and knowledge fusion, Inf. Fusion 100 (2023) 101962.
[13] T. Wang, B. Sun, C. Jiang, Kernel similarity-based multigranulation three-way decision approach to hypertension risk assessment with multi-source and multi-level structure data, Appl. Soft Comput. (2023) 110470.
[14] J. Zhang, T. Li, D. Ruan, D. Liu, Neighborhood rough sets for dynamic data mining, Int. J. Intell. Syst. 27 (4) (2012) 317–342.
[15] M. Beynon, Reducts within the variable precision rough sets model: a further investigation, Eur. J. Oper. Res. 134 (3) (2001) 592–605.
[16] S.S. Kumar, H.H. Inbarani, Optimistic multi-granulation rough set based classification for medical diagnosis, Proc. Comput. Sci. 47 (2015) 374–382.
[17] Y. Qian, J. Liang, Y. Yao, C. Dang, Mgrs: a multi-granulation rough set, Inf. Sci. 180 (6) (2010) 949–970.
[18] W. Xu, X. Zhang, Q. Wang, A generalized multi-granulation rough set approach, in: Bio-Inspired Computing and Applications: 7th International Conference on Intelligent Computing, ICIC 2011, Revised Selected Papers 7, Zhengzhou, China, August 11–14, 2011, Springer, 2012, pp. 681–689.
[19] X. Zhang, X. Huang, W. Xu, Matrix-based multi-granulation fusion approach for dynamic updating of knowledge in multi-source information, Knowl.-Based Syst. 262 (2023) 110257.
[20] B. Sang, H. Chen, L. Yang, T. Li, W. Xu, C. Luo, Feature selection for dynamic interval-valued ordered data based on fuzzy dominance neighborhood rough set, Knowl.-Based Syst. 227 (2021) 107223.
[21] Y. Yang, D. Chen, X. Zhang, Z. Ji, Y. Zhang, Incremental feature selection by sample selection and feature-based accelerator, Appl. Soft Comput. 121 (2022) 108800.
[22] Y. Pan, W. Xu, Q. Ran, An incremental approach to feature selection using the weighted dominance-based neighborhood rough sets, Int. J. Mach. Learn. Cybern. 14 (4) (2023) 1217–1233.
[23] B. Sang, H. Chen, L. Yang, D. Zhou, T. Li, W. Xu, Incremental attribute reduction approaches for ordered data with time-evolving objects, Knowl.-Based Syst. 212 (2021) 106583.
[24] W. Huang, Y. She, X. He, W. Ding, Fuzzy rough sets-based incremental feature selection for hierarchical classification, IEEE Trans. Fuzzy Syst. (2023).
[25] Z. Pawlak, Rough sets, Int. J. Comput. Inf. Sci. 11 (1982) 341–356.
[26] Q. Hu, D. Yu, J. Liu, C. Wu, Neighborhood rough set based heterogeneous feature subset selection, Inf. Sci. 178 (18) (2008) 3577–3594.
[27] W. Xu, W. Sun, X. Zhang, W. Zhang, Multiple granulation rough set approach to ordered information systems, Int. J. Gen. Syst. 41 (5) (2012) 475–501.
[28] B. Ma, C. Yang, A. Li, Y. Chi, L. Chen, A faster dbscan algorithm based on self-adaptive determination of parameters, Proc. Comput. Sci. 221 (2023) 113–120.
[29] S. Guha, A. Das, P.K. Singh, A. Ahmadian, N. Senu, R. Sarkar, Hybrid feature selection method based on harmony search and naked mole-rat algorithms for spoken language identification from audio signals, IEEE Access 8 (2020) 182868–182887.
[30] W. Xie, Y. Chi, L. Wang, K. Yu, W. Li, Mmbde: a two-stage hybrid feature selection method from microarray data, in: 2021 IEEE International Conference on Bioinformatics and Biomedicine (BIBM), IEEE, 2021, pp. 2346–2351.
[31] L. Peng, Z. Cai, A.A. Heidari, L. Zhang, H. Chen, Hierarchical Harris hawks optimizer for feature selection, J. Adv. Res. (2023).
[32] M. Tubishat, S. Ja'afar, M. Alswaitti, S. Mirjalili, N. Idris, M.A. Ismail, M.S. Omar, Dynamic salp swarm algorithm for feature selection, Expert Syst. Appl. 164 (2021) 113873.
[33] K. Robindro, S.S. Devi, U.B. Clinton, L. Takhellambam, Y.R. Singh, N. Hoque, Hybrid distributed feature selection using pso-mi, Data Sci. Manag. (2023).
[34] M.G. Altarabichi, S. Nowaczyk, S. Pashami, P. Sheikholharam Mashhadi, Fast genetic algorithm for feature selection—a qualitative approximation approach, in: Proceedings of the Companion Conference on Genetic and Evolutionary Computation, 2023, pp. 11–12.
[35] R. Guha, M. Ghosh, S. Mutsuddi, R. Sarkar, S. Mirjalili, Embedded chaotic whale survival algorithm for filter–wrapper feature selection, Soft Comput. 24 (17) (2020) 12821–12843.
[36] J. Too, A.R. Abdullah, Opposition based competitive grey wolf optimizer for emg feature selection, Evol. Intell. 14 (4) (2021) 1691–1705.