Applied Soft Computing Journal 170 (2025) 112716

Contents lists available at ScienceDirect

Applied Soft Computing

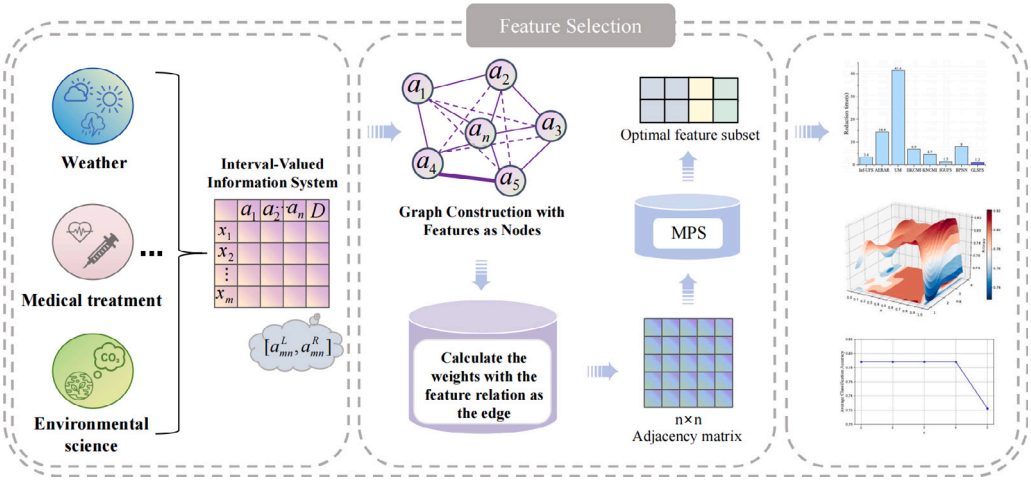
journal homepage: www.elsevier.com/locate/asoc

Graph-driven feature selection via granular-rectangular neighborhood rough sets for interval-valued data sets

Xiaoyan Zhang [a,b,c,*], Xuan Shen [a]

[a] College of Artficial Intelligence, Southwest University, Chongqing, 400715, PR China
[b] National & Local Joint Engineering Research Center of Intelligent Transmission and Control Technology, Chongqing, 400715, PR China
[c] Chongqing Key Laboratory of Brain-inspired Computing and Intelligent Chips, Chongqing, 400715, PR China

GRAPHICAL ABSTRACT



ARTICLE INFO

Keywords:
Feature selection
Graph theory
Interval-valued information system (IVIS)
Matrix power series (MPS)
Neighborhood rough set (NRS)

ABSTRACT

In the burgeoning landscape of big data analytics, interval-valued datasets are indispensable for modeling uncertainty and vagueness, with significant implications for sectors such as healthcare and environmental science. Feature selection, a linchpin in data mining, is paramount for streamlining data processing and bolstering predictive models. However, the literature on feature extraction within interval-valued information systems is notably sparse. This paper proposes a groundbreaking feature selection framework that skillfully addresses the complexities of interval-valued data. The method innovatively utilizes a fully connected weighted undirected graph to encapsulate interval data, combining graph-theoretic insights with granular-rectangular neighborhood rough set theory. By evaluating the significance of each attribute based on its importance to the entire information system, and applying matrix power series to accelerate computations, the framework ensures both robust classification performance and the elimination of redundancy, marking a significant advancement in this field. Through comparative experiments on 12 public datasets with 7 other algorithms, theoretical analysis, and experimental results demonstrate that the proposed method not only exhibits high effectiveness in handling interval-valued data but also further improves efficiency and classification performance. In addition, the method also shows significant advantages in reducing the dependence on prior knowledge and improving the interpretability of the model, which fully proves its applicability and reliability in large-scale data analysis.

* Corresponding author.
E-mail addresses: zxy19790915@163.com (X. Zhang), sx18908529566@163.com (X. Shen).

https://doi.org/10.1016/j.asoc.2025.112716
Received 22 July 2024; Received in revised form 22 November 2024; Accepted 3 January 2025
Available online 9 January 2025

1. Introduction

As big data and artificial intelligence technologies advance, the significance of interval values and interval analysis in data science and machine learning is escalating. Interval-valued data, characterized by its representation of each data point as a range between a lower and upper limit, offers a nuanced approach to capturing the intrinsic variability and uncertainty inherent in measurements, as exemplified by weather forecasts and medical diagnostics. This form of data is especially prevalent in sectors where precision is inherently constrained or data collection is fraught with uncertainty, such as environmental science, engineering, and economic analysis. The aggregation of such data into interval-valued information systems (IVIS) [1] has emerged as a burgeoning area of interest within the broader study of uncertainty in artificial intelligence [2,3]. With the ongoing enhancements in data acquisition and computational power, feature selection leveraging interval values is poised to expand its reach, becoming increasingly vital in fields that contend with high levels of data uncertainty.

Feature selection [4] stands as a pivotal area of inquiry within the realms of machine learning and data mining, dedicated to discerning the most salient subset of features from expansive datasets. This endeavor is instrumental in amplifying the efficiency of model learning, curtailing their intrinsic complexity, and fortifying their capacity for generalization. Beyond bolstering computational expedience, feature selection is paramount for augmenting the models' predictive fidelity by expunging extraneous or supererogatory features, thus circumventing the pitfalls of overfitting. Conventional feature selection methodologies, often marred by a dearth of interpretability, presuppose an underpinning of exact numerical data. Yet, in the context of interval-valued data, a paradigm shift is imperative, necessitating the recalibration or conceptual reengineering of these techniques to align with the nuances of imprecise datasets.

Pawlak's Rough Set Theory (RST) [5] is recognized as a powerful mathematical tool for addressing uncertainty and vagueness. It is renowned for its ability to interpretably handle incomplete, imprecise, and inconsistent datasets without requiring prior knowledge. RST has garnered significant attention in the artificial intelligence community, with its applications expanding to key areas such as pattern recognition [6] and decision-making analysis [7,8]. The use of RST for knowledge discovery in interval-valued datasets has also attracted growing interest [9–11]. However, since the original RST is primarily tailored for discrete datasets, researchers have introduced neighborhood relations to extend its applicability, resulting in interval-valued neighborhood rough set models [12,13]. Despite these advancements, reducing the computational complexity of these methods remains an urgent issue to address.

Graph-based methods offer a compelling solution, as they can capture and leverage structural features within data to reduce computational overhead [14,15]. However, graph-based feature selection methods still face a significant challenge: how to effectively minimize their dependence on prior knowledge [16,17]. In many studies, these methods often require substantial prior knowledge to construct the graph, which becomes especially problematic when data uncertainty is high and prior knowledge is limited.

Based on the above analysis, overcoming the limitations of feature selection in interval-valued information systems (IVIS) requires addressing two key challenges currently faced by this field:

(1) The challenge of extracting valuable information from uncertainty-laden interval-valued data while maintaining the authenticity and comprehensiveness of the information, reducing reliance on prior knowledge, and ensuring interpretability.
(2) In the context of rapidly increasing data volume and dimensionality, there is an urgent need to effectively reduce data dimensionality, improve computational efficiency, and alleviate time complexity to meet the application demands of large-scale datasets.

To address these challenges, this paper proposes a feature selection strategy specifically designed for IVIS, designed to navigate these complexities. It initiates by scrutinizing the pivotal role of attributes within the broader information system, influenced by dual determinants. The first is the attribute's contribution to the classification prowess of the IVIS, quantified through granular rectangular neighborhood rough set theory. The second is the attribute's uniqueness, where independence from other IVIS attributes signifies non-redundancy. This framework culminates in the construction of a weighted undirected fully connected graph, transmuting IVIS data into an adjacency matrix informed by these criteria. The matrix power series (MPS) is then ingeniously harnessed to demystify the computational intricacies.

Based on the analysis above, our research contributions are threefold:

(1) We improved the accuracy and interpretability of the feature selection method, making it adept at handling interval-valued data and leveraging its inherent uncertainty. By integrating granular-rectangular neighborhood rough sets, we utilized the granularity and hierarchical structure of the data to construct equivalence classes through space partitioning, thereby enhancing the model's interpretability. This strategy enhances the model's generalization ability, reduces the risk of overfitting, and improves interpretability.
(2) We engineer a reduction in computational complexity, infusing our feature selection process with graph-based methodologies and granular-rectangular neighborhood rough sets. The construction of a weighted undirected fully connected graph, coupled with the MPS, streamlines the computational workflow, facilitating rapid model execution even with large-scale datasets.
(3) We reduced the reliance on prior knowledge by constructing a relationship graph between features, while ensuring non-redundancy and high classification capability in the feature selection process.

The overarching structure of our research is delineated in Fig. 1. This work is meticulously structured into seven comprehensive sections. Section 2 provides a detailed overview of related work. Section 3 explores foundational studies related to Interval-Valued Information Systems (IVIS), Granular-Rectangular Rough Sets (GRRS), Matrix Power Series (MPS), and graph-theoretic concepts. In Section 4, based on the axioms of relevance and non-redundancy, we have constructed a graph tailored for IVIS and designed a novel supervised feature selection method. Section 5 introduces the algorithmic formulation of our Graph-based Local Search Feature Selection (GLSFS) approach. Section 6 substantiates the efficacy and applicability of our proposed method through an array of rigorous experiments. The paper culminates in Section 7, synthesizing our findings recognizing limitations, and charting pathways for future research efforts.

2. Related work

The information on various studies omitted in the introduction will be elaborated in detail in this section.

Feature selection methods based on RST have been widely applied to single-point datasets [18–21], with some researchers extending its application to interval-valued data. Jensen et al. [22] proposed an interval-valued fuzzy-rough set model for datasets with missing values. Li et al. [23] introduced a feature selection method based on interval dominance relations for interval-valued ordered data systems. To accommodate a wider range of data types, Lin and Yao [12] were the first to combine rough sets with neighborhood systems, elucidating the connection with fuzzy sets. Yao [13] further explored granular structures from the perspective of rough sets and neighborhood systems. Building on Lin's foundational neighborhood model [12], Hu et al. [24] proposed a Neighborhood Rough Set (NRS) model based

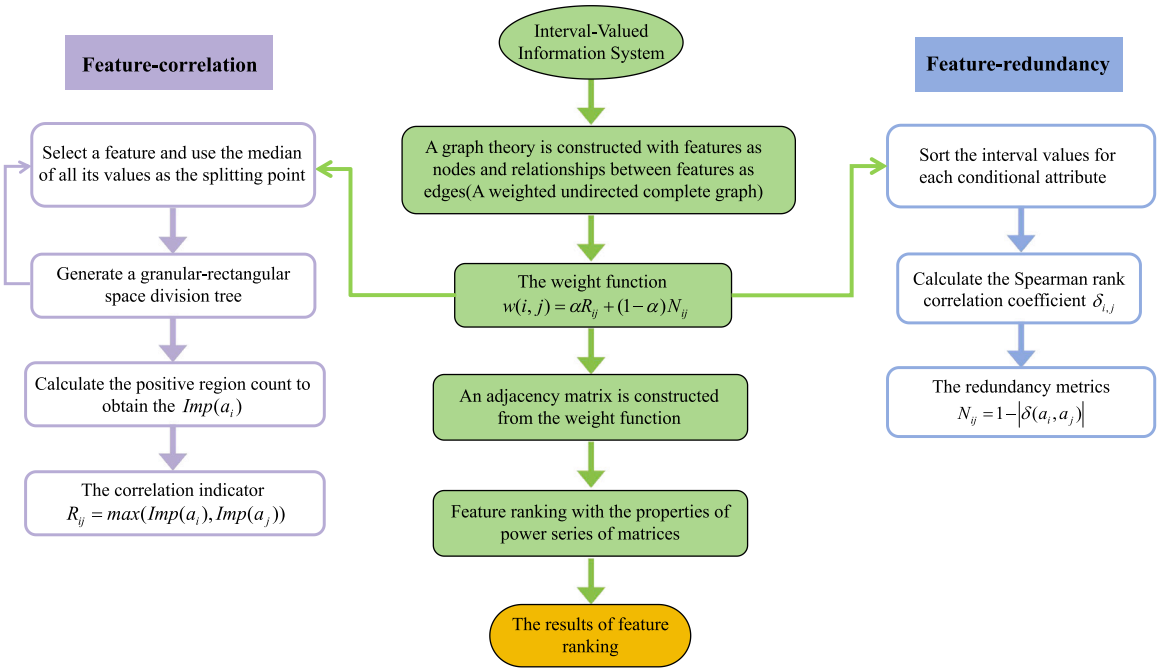


Fig. 1. The overarching structure of our research.

on neighborhood relations instead of equivalence relations, facilitating the handling of continuous data and leading to the development of interval-valued neighborhood rough set models. Zhang et al. [25] proposed a high-dimensional interval-valued feature selection algorithm based on a weighted interval-valued neighborhood rough set model. Sang et al. [26] investigated incremental feature selection methods for interval-valued ordered data using fuzzy dominance neighborhood rough sets.

Meanwhile, graph-based methods have garnered increasing attention for their unique advantages in data structure modeling and feature interaction analysis, and have been progressively applied to feature selection problems. Akhiat et al. [14] constructed a weighted graph with single features and feature pairs, using iterative partitioning and graph structure optimization for feature selection. Roffo et al. [15] proposed a feature filtering method that treats the feature set as paths within a graph. Xie et al. [16] enhanced feature selection by jointly learning graph structures and latent representations. Tang et al. [17] studied unsupervised feature selection methods involving multi-graph fusion. Ke et al. [27] proposed an adaptive disentangled representation learning method based on graph structural information. Dong et al. [28,29] improved the performance and efficiency of graph neural networks through techniques such as injection aggregation and principal component analysis for denoising.

Table 1 provides a classification and summary of the existing literature based on different methods and application scenarios. We can observe that several current feature selection methods exhibit significant limitations when dealing with uncertainty and complex data. First, the traditional rough set theory still difficult to deal with interval valued data effectively and has high computational complexity, which limits its application to large-scale data sets. Second, neighborhood rough set theory expands the application of classical rough sets by introducing distance metrics. However, the upper and lower approximations in these methods consist of sample points rather than equivalence classes, resulting in weaker model interpretability. Furthermore, the reliance on distance metrics leads to instability across different types of datasets. Finally, graph-based methods, when constructing feature relationships, are heavily dependent on prior knowledge and initial graph partitioning. This increases the reliance on prior information and results in high computational complexity, particularly when handling high-dimensional data.

The proposed GLSFS feature selection method effectively addresses the gaps and limitations of current research. It enhances the description and interpretability of uncertain knowledge in interval-valued information systems while significantly reducing dependency on prior knowledge and computational complexity. Table 2 summarizes the related works that provide valuable references and support for this study.

3. Preliminaries

In this section, we embark on an introduction to IVISs, subsequently delving into a review of pivotal definitions within the realms of Graph Theory, Granular-Rectangular Rough Set (GRRS), Spearman Rank Correlation Coefficient (SRCC), and the Matrix Power Series (MPS).

3.1. Interval-valued information system

Interval-valued data refers to the collection of interval-valued attributes that describe objects or instances in an information system. Each object is characterized by a set of interval-valued attributes rather than exact values. Interval-valued data allows for the representation of uncertainty and imprecision in the information system.

Let $IVIS = (U, AT, f)$ be an information system, where $U = \{x_1, x_2, \ldots, x_m\}$ is a non-empty finite set of objects, $AT = \{a_1, a_2, \ldots, a_n\}$ is a non-empty finite set of attributes, and $\forall x_i \in U$, $a_k \in AT$, the $f(x_i, a_k)$ yields an interval-valued number, specifically denoted as $f(x_i, a_k) = [a_k^L(x_i), a_k^R(x_i)]$. $a_k^L(x_i)$ and $a_k^R(x_i)$ are respectively referred to as the left and right boundaries of this interval, and they can also be symbolized as a_{ik}^L and a_{ik}^R for brevity. Notably, in the context of such an interval-valued framework, if the left and right boundaries coincide, i.e., $a_k^L(x_i) = a_k^R(x_i)$, then $f(x_i, a_k)$ simplifies to a single-valued output, thereby illustrating that a single-valued decision system is but a particular manifestation of the broader IVIS paradigm.

Drawing upon the existing literature [30], let us consider an interval-valued ordered information system denoted as $IVIS^{\preceq} = (U, AT, f)$, $\forall A \subseteq AT$, the dominance relation $R_A^{\preceq}$ is defined as

$$R_A^{\preceq} = \{(x_i, x_j) \in U \times U \mid (\forall a_k \in A)[a_k^L(x_i) \le a_k^L(x_j), a_k^R(x_i) \le a_k^R(x_j)]\}.$$

Table 1
The summary of existing methods.

Methods	Categories	Disadvantages
Rough set theory	For single-valued datasets [18–21]	Not applicable to interval-valued datasets.
	Using attribute significance for interval-valued datasets [22,23]	The computational cost is high.
Neighborhood rough set theory	Domain relations using distance measures [12,13,24]	The lower and upper approximations consist of sample points, rather than equivalence classes, and thus lose interpretability. And it relies on distance metrics.
	Using weighed neighborhood rough set [25]	
	Utilizing conditional entropy [26]	The computational cost is high.
Graph theory	Build a weighted graph [14,15]	Depends on the initial partition of the graph.
	A priori information about the graph structure is used to construct the feature space [16,17,27]	The dependence on prior knowledge is high.
	Graph neural network denoising based on PCA [28,29]	High computational complexity and excessive memory consumption.

Table 2
Summary of related works supporting the research.

Authors	Research content	Reference
Roffo et al.	The weighted graph was constructed for feature selection.	[15]
Qian et al.	Interval ordered information systems.	[30]
Hu et al.	Neighborhood rough set for feature selection.	[31]
Xia et al.	The granular ball neighborhood rough set.	[32,33]
Xia et al.	The granular-rectangular rough set.	[34]
Fieller et al.	Tests for rank correlation coefficients.	[35]

Afterwards, we can use the dominance relationship to determine the ordinal position of objects with respect to attributes.

3.2. Graph theory

As reported in the literature [15], by constructing a weighted undirected complete graph $G = \langle V, E\rangle$, the interval-valued information system can be described, where the nodes in the graph represent features $A = \{a_1, a_2, \ldots, a_n\}$ and the edges E formed represent relationships between features. Setting a weight function $w(i, j)$ on the edges between feature a_i and a_j, and generating the adjacency matrix Q based on this weight function $w(i, j)$. The graph G employs its adjacency matrix Q as a means to encapsulate the intricate relationships between nodes and the information they convey. Each entry $Q(i, j)$ within this matrix is expressed through a corresponding weight function denoted as $w(.,.)$, facilitating a nuanced representation of the interconnections.

$$Q(i, j) = w(i, j).$$

The weight function $w(.,.)$ captures the relationship between feature a_i and a_j by incorporating a weighted linear combination of two evaluation metrics, defined as

$$w(i, j) = \alpha R_{ij} + (1 - \alpha)N_{ij}. \tag{1}$$

where R_{ij} represents the significance of features to the information system, describing their contribution to the classification problem. Meanwhile, N_{ij} represents the lack of correlation between pairs of features. Therefore, the weight function $w(i, j)$ can be utilized to assess the importance of features to the information system and the non-redundancy among features.

3.3. Granular-rectangular rough set

Neighborhood rough set (NRS) [31] are an extension of traditional rough set theory, which primarily relies on exact equivalence relations to partition data. However, NRS introduces the concept of neighborhoods, allowing for a certain degree of uncertainty and ambiguity. NRS uses neighborhood relations to describe the relationships between samples. These neighborhood relations are derived entirely from the data distribution and do not require any prior knowledge. Additionally, NRS can directly handle continuous data. However, since the upper and lower approximations in NRS are composed of sample points rather than equivalence classes, it loses interpretability.

The classical NRS model requires computing the neighborhood for each sample, resulting in relatively low efficiency. The GBNRS (Granular Ball Neighborhood Rough Set) [32,33] model proposes a new neighborhood model that partitions the dataset using the computation approach of granular balls, where all samples within each granular ball share the same neighborhood. Iterative refinement of these granules, by controlling either the quality of the granules or a threshold on the number of samples within them, results in the formation of robust granules with a purity of one. The center of a granular ball is defined as a generated positive region. Granules with a purity of less than one are considered boundary regions. Additionally, its upper and lower approximations are described by equivalence classes, maintaining interpretability.

The GRRS [34] divides the data set into multiple subspaces using spatial partitioning, each of which corresponds to a specific range in the data, defined as

Definition 1. $IS = (U, AT, D)$ is given as a decision information system, $\forall A \subseteq AT$ and $A = \{a_1, a_2, \ldots, a_m\}$, S is used as the spatial partition of the attribute set A on the target sample, $S = \{S_1, S_2, \ldots, S_n\}$. The

subspace $S_k(0 < k < n)$ is represented as $\{x_k \mid a_{1_{k-}} \le x_{k1} < a_{1_{k+}}, a_{2_{k-}} \le x_{k2} < a_{2_{k+}}, \ldots, a_{m_{k-}} \le x_{km} < a_{m_{k+}}\}$, where x_{km} denotes the value of x_k in the mth dimension, and $a_{m_{k-}}$ and $a_{m_{k+}}$ denote the upper and lower bounds of that space. S is a granular-rectangular space partition.

Parent granular spaces possess a coarser granularity compared to their child granular spaces, and as the space is refined, the upper and lower bounds of each granular rectangle become increasingly tight. These subspaces are interconnected, forming a hierarchical tree structure. Within this structure, each node represents a subspace, and the connections between nodes depict the relationships between child and parent spaces. This hierarchical layout allows for a clear visualization of data distribution, facilitating quick localization of specific subspaces based on attributes. Similar to GBNRS, the tree structure formed by the division in GRRS ensures that all data points within a subspace share the same neighborhood. The shared neighborhood aids in distinguishing between internal and boundary points within each subspace. Unlike traditional NRS and GBNRS, which use Euclidean distance, GRRS employs a unique method for defining neighborhood radius. In GRRS, the neighborhood radius is no longer a continuous variable but is transformed into a discrete value. Specifically, the neighborhood radius is defined by ascending from a leaf node up to its parent space, increasing by one with each level ascended. This discretization simplifies the computational process and aligns the definition of neighborhoods more closely with practical application needs.

3.4. Spearman rank correlation coefficient

Spearman rank correlation coefficient [35] is a statistic used to measure the nonlinear relationship between two variables. It is based on the ranks or orders of variables, rather than the actual numerical values of variables.

Given δ as the SRCC between two vectors a and b, where

$$\delta = 1 - \frac{6\sum_{i=1}^{m} d_i^2}{m(m^2-1)}. \tag{2}$$

the sample size is denoted by m, while d signifies a specific type of difference sequence between a and b.

The value of SRCC ranges from −1 to 1. When the absolute value of the coefficient approaches 1, it indicates a strong correlation between the two variables, a and b. Conversely, when the absolute value of the coefficient is close to 0, it indicates that there is almost no linear relationship between the two variables.

3.5. Matrix power series

Let $A_0, A_1, A_2, \ldots, A_k, \ldots$ be a sequence of matrices, where $A_k = (a_{ij}^{(k)}) \in C^{m\times n}$, then the sum formula $A_0 + A_1 + A_2 + \cdots + A_k + \cdots$ is called an MPS [36], which can be written as $\sum_{k=0}^{\infty} A_k$

For any positive integer N, let $B^{(N)} = \sum_{k=0}^{N} A_K$ be the partial sum of the matrix series. If the matrix sequence $\{B^{(N)}\}$, which is made of $B^{(N)} = \sum_{k=0}^{N} A_K$, converges [37] and has a limit, that is, $\lim_{N\to\infty} B^{(N)} = B$, then the matrix series $\sum_{k=0}^{\infty} A_k$ converges, and let call S the sum of this matrix series, that is, $S = \sum_{k=0}^{\infty} A_k$.

Proposition 1. *If the partial sums of a matrix sequence $\sum_{k=0}^{N} A_k$ converge to a matrix B as $N \to \infty$, then the series of each element $\sum_{k=0}^{\infty} a_{ij}^{(k)}$ in the matrix also converges, and it converges to the corresponding element B_{ij} of matrix B.*

Proof. Assume that the partial sums of the matrix sequence $\sum_{k=0}^{N} A_k$ converge to the matrix B, i.e., $\lim_{n\to\infty} \sum_{k=0}^{N} A_k = B$. The convergence of the matrix implies that each of its elements also converges according to its index. Therefore, the element $a_{ij}^{(k)}$ of matrix A_k: $\lim_{n\to\infty} \sum_{k=0}^{N} a_{ij}^{(k)} = B_{ij}$. This indicates that the partial sum sequence of each element $\sum_{k=0}^{N} a_{ij}^{(k)}$ converges to the corresponding element B_{ij} of matrix B. Since the partial sum of each element converge, we can infer the convergence of the infinite series: $\sum_{k=0}^{\infty} a_{ij}^{(k)} = \lim_{n\to\infty} \sum_{k=0}^{N} a_{ij}^{(k)} = B_{ij}$. Therefore, the infinite series $\sum_{k=0}^{\infty} a_{ij}^{(k)}$ of each element $a_{ij}^{(k)}$ converges to B_{ij}.

Employing the convergence properties of MPS substantially streamlines the computation process when summing up certain specialized matrix series.

4. The graph-based local search feature selection (GLSFS) model

Our feature selection method, GLSFS, evaluates the importance of attributes in the entire information system, as illustrated in Fig. 2.

The method consists of three main parts. First, we construct a weighted undirected fully connected graph based on graph theory, where the original attributes are represented as nodes and the relationships between attributes are represented as edges, with weight functions assigned to the edges to intuitively reflect the interrelationships among the attributes. Second, the calculation of the weights is influenced by two key factors: on one hand, the contribution of each attribute to classification ability is assessed using the GRRS, reflecting the relevance of the attribute to the information system and its importance; on the other hand, the non-redundancy among attributes is measured by calculating the SRCC after sorting the interval-valued data, ensuring that the selected features are both significant and redundant. Finally, these two factors are transformed into the adjacency matrix of the graph, and the MPS is applied to compute the feature ranking from the adjacency matrix. The optimal feature subset is obtained by flexibly setting the cutoff proportion to truncate the ranking sequence.

4.1. Graph building for IVIS

In the context of feature selection based on intervals, we establish a weighted undirected complete graph $G = \langle V, E\rangle$. Each node in the graph represents a set of features $A = \{a_1, a_2, \ldots, a_n\}$, and the edges E between nodes indicate relationships between the features. Subsequently, we assign a weight function $w(.,.)$ to the edges, which allows us to compute the element $Q(i,j)$. Using this $Q(i,j)$, we generate an adjacency matrix Q that describes the importance of features in the information system and the relationships between features in the graph G. Each entry $Q(i,j)$ within this matrix is expressed through a corresponding weight function $w(.,.)$, defined as

$$w(i,j) = \alpha R_{ij} + (1-\alpha)N_{ij}. \tag{3}$$

R_{ij} represents the relevance of features, reflecting their importance to the information system. Therefore, a larger value of R_{ij} is desirable, denoting a higher importance of the feature. Its value is based on the count of the positive domain in granular rectangles. In rough set theory, the larger the positive domain, the better the classification capability of the attribute set, *i.e.* , $R_{ij} = max(Imp(a_i), Imp(a_j))$, where $Imp(a_i)$ refers to the positive domain ratio for feature $\{a_i\}$. N_{ij} signifies the non-redundancy between features. Following the principle of redundancy minimization, we aim for as low redundancy as possible. Hence, to ensure the monotonicity of criterion $w(.,.)$, the second term is set as $N_{ij} = 1- \mid \delta(h_i, h_j) \mid$. This setup underscores the balance between the significance of feature representation and the minimization of redundancy within the feature set. α is a parameter $\in [0,1]$. The α will be incrementally adjusted with a step size of 0.1 in subsequent experiments. Its optimal value will be determined through a series of five-fold cross-validation trials conducted during the training phase of the classification task.

4.2. The calculation of weight functions

To facilitate the construction of graph theory, we provide detailed descriptions of correlation indicators and redundancy metrics for the weight function.

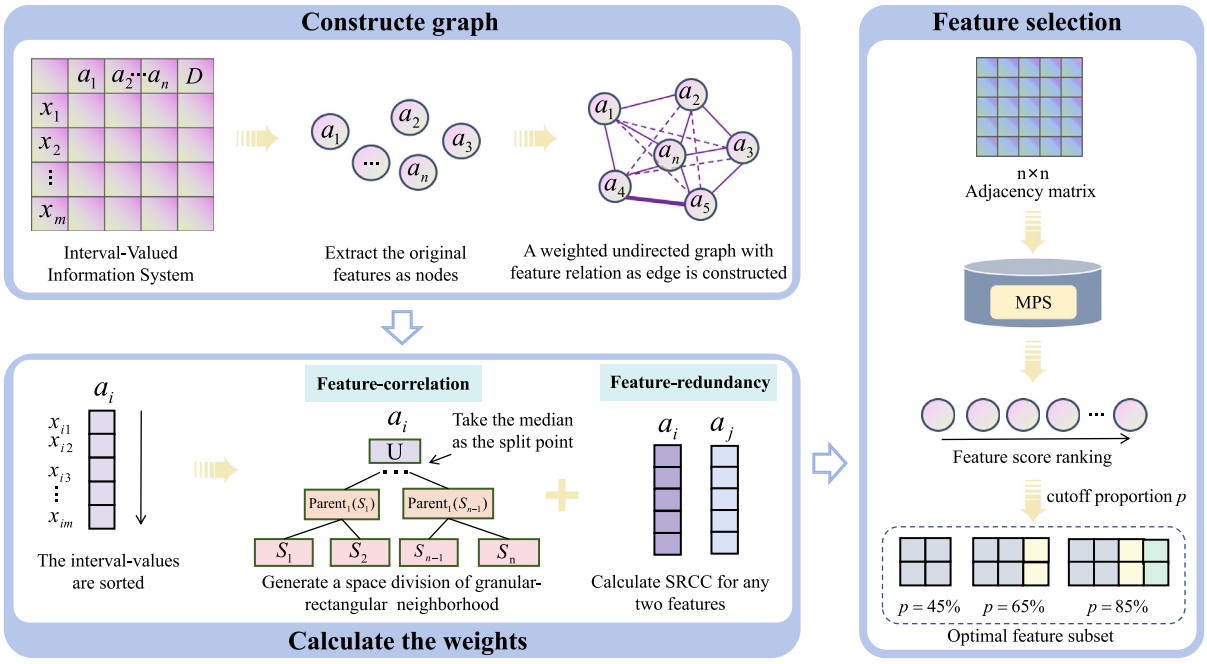


Fig. 2. The mind map of GLSFS model. In the constructed graph, each node represents an original attribute, and edges between nodes indicate relationships with weights assigned based on correlation and redundancy. The process of space division for weight calculation is illustrated in Fig. 3. The adjacency matrix is derived from these weights, and MPS is used for feature ranking. The top p percent of features are selected as the optimal subset, while the rest are discarded as less relevant.

4.2.1. The correlation indicator

Initially, we present the specific calculation method for the importance of features based on the GRRS under an interval-valued information system.

In traditional rough set theory, datasets are partitioned into several equivalence classes where each class contains samples with identical values across all attributes. However, this method is less effective with continuous attributes due to the wide range of values, making it challenging to find identical values. To address this, GRRS has proven highly effective for feature selection with continuous data. Instead of specific values, GRRS utilizes value ranges to segment the dataset into subsets, where samples within each subset are consistent within a given attribute range. This approach has been adapted to interval-valued information systems for feature selection. Most existing NRS models rely on Euclidean distance metrics, which may fail when attribute weights differ. To overcome this issue, the concept of granular-rectangular neighborhoods is introduced, which eliminates the need for distance metrics and reduces the search space for neighborhood radii, significantly enhancing the accuracy and efficiency of NRS. This method constructs neighborhood radii by describing the relationship between child and parent spaces without relying on any distance metrics. Specifically, if two data points are close within the value range of each attribute, falling within the same or adjacent granular rectangles, they are considered to be in the same domain.

GRRS partitions the dataset into multiple subspaces. The parent granular space has a coarser granularity than the child granular space. As the space is continuously refined, the upper and lower bounds of each dimension in each granular rectangle become increasingly tight. Eventually, the space forms a tree-like structure, with the root node containing all samples and all leaf nodes constituting a granular rectangular space partition. First, we perform a spatial division of the interval-valued information system. When sorting interval-valued data, traditional sorting methods are inadequate for handling nested interval values [36]. Therefore, a method for sorting interval values is proposed to obtain an ordered set of all sample values under a specific feature.

Definition 2. Let $IVIS = (U, AT, F)$ be an interval-valued information system, $\forall A \subseteq AT$, $a_k \in A$, $f(x_i, a_k) = [a_{ik}^L, a_{ik}^R]$ is an interval-valued number. Let $g = (g^L, g^R)$ and $h = (h^L, h^R)$ be two interval values. $g \preceq h$ if and only if

$$\left(\frac{h^L + h^R}{2} - \frac{g^L + g^R}{2}\right) > \theta$$

and the dominance relation $R_A^{\preceq}$ is defined as

$$R_A^{\preceq} = \{(x_i, x_j) \in U \times U \mid \frac{a_{jk}^L + a_{jk}^R}{2} - \frac{a_{ik}^L + a_{ik}^R}{2} \geq \theta, \theta \geq 0\}.$$

where θ is a nonnegative number, which signifies the permitted extent of deviation, acting as a tolerance threshold.

After sorting the interval values, the next step is to select the split point for the target splitting attribute. We choose the median of all object values under the target feature as the split point. This approach, when applied to continuous numerical features, can effectively partition the dataset into two relatively balanced subsets. It contributes to minimizing skewness in the decision tree and enhancing the generalization capability of the model. From this, by iteratively performing spatial divisions to form a complete tree structure, we define the indistinguishable relationship of granular-rectangular and its equivalence classes.

Definition 3. Let $IVIS = (U, AT, D, f)$ be an interval-valued information system. $\forall x, y \in U$ and $a \in AT$, S_i is a granular-rectangular in a granular-rectangular division, and the indistinguishable relationship of granular-rectangular INDGR (a) of the attribute a on S_i is defined as

$$INDGR(a) = \{(x, y) \in U \times U \mid f(x, a) \in S_i, f(y, a) \in S_i, \forall a \in AT\}$$

where f is a mapping function that represents the value of the object under attribute a. A granular-rectangle equivalent class is defined as

$$[x]_{GR(a)} = \{y \in U \mid (x, y) \in INDGR(a)\}.$$

$\{[x]_{GR(a)} \mid x \in U\}$ is a granular-rectangular division of U, denoted $U/INDGR(a)$ and abbreviated $U/GR(a)$.

As shown in Fig. 3(a), a tree is a succinct method to represent granular-rectangular divisions. Initially, the entire dataset acts as the root node of the decision tree. First, an attribute is selected and all

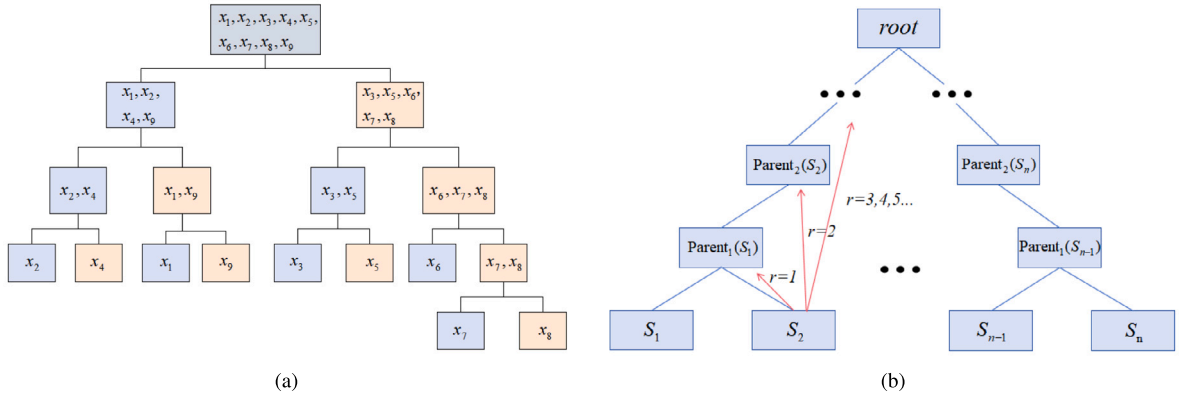


Fig. 3. (a)An example of the tree structure formed by space division. (b)The neighborhood radius of granular-rectangular space division.

interval values under this attribute are sorted to determine the median, which then serves as the splitting point. This operation divides the dataset into two subspaces, each corresponding to the samples that are either below or above the median. This process is recursively applied to each subspace until the space is completely divided, ultimately forming a tree structure. In this structure, each node represents an equivalence class, and all the leaf nodes together constitute a granular-rectangular division of space. Clearly, the closer two nodes are, the higher the similarity of the samples they contain. Based on this, we can describe the neighborhood radius under space division.

As can be seen in Fig. 3(b), the neighborhood radius r is defined by the level of the parent space and is a smaller discrete value. As shown in Fig. 3(a), in the subspace (leaf node) where sample x_2 is located, the neighborhood with radius 1 includes the subspaces containing samples x_2 and x_4, while the neighborhood with radius 2 includes the subspaces containing samples x_1, x_2, x_4, and x_9. In our experiments, we set the step size to 1 and searched for the optimal neighborhood radius within the range of 1 to 5. Clearly, the neighborhood of a granular rectangle is defined based on equivalence classes. Accordingly, we provide the definition of upper and lower approximations described using equivalence classes.

Definition 4. Let $IVIS = (U, AT, D, f)$ be an interval-valued information system, where U is the data point set, AT is the conditional attribute set, and D is the decision attribute set. Let $S = \{S_1, S_2, \ldots, S_M\}$ be a granular-rectangular division generated on U. D divides U into N equivalence classes, $X_1, X_2, \ldots, X_N$. According to the S, $\forall a \in AT$, the lower approximation and the upper approximation of the decision attribute set D under the conditional attribute a are respectively defined as

$$\underline{GRa}(D) = \bigcup_{i=1}^{N} \underline{GRa}(X_i). \tag{4}$$

$$\overline{GRa}(D) = \bigcup_{i=1}^{N} \overline{GRa}(X_i). \tag{5}$$

where $\underline{GRa}(X_i) = \{S_j \mid parent_r(S_j) \subseteq X_i, S_j \subset U\}$, $\overline{GRa}(X_i) = \{S_j \mid parent_r(S_j) \cap X_i \neq \varnothing, S_j \subset U\}$.

The lower approximation, denoted as $\underline{GRa}(D)$, consists of granular-rectangular equivalence classes whose neighborhood can be fully contained within an equivalence class partitioned by the decision attribute set D. On the other hand, the upper approximation, denoted as $\overline{GRa}(D)$, comprises granular-rectangular equivalence classes whose neighborhood intersects with the equivalence class that requires representation and is also partitioned by the decision attribute set D. In rough set theory, the positive region is equivalent to the lower approximation, as it consists of all objects that can be unambiguously classified into a particular concept based on the available attributes. Based on Eq. (4), we obtain the positive region of a granular-rectangular division as

$$POS^{S}_{GRa}(D) = \underline{GRa}(D). \tag{6}$$

In rough set theory, the accuracy of a model depends on the size of the positive, negative, and boundary regions. The accuracy is weaker when the positive region is smaller and the negative region is larger. Additionally, the smaller the boundary region of the rough model, the greater its ability to handle uncertainty. The positive region is an important concept used to describe the set of objects for which the value of the decision attribute is determined, reliable, and not constrained by the conditions. The larger the positive region, the higher the likelihood that the number of samples whose neighborhoods are completely contained in the equivalence classes partitioned by the decision attribute set D will increase. This demonstrates that the division formed by the condition attribute a_i is a better fit for the decision boundary. In other words, there is a greater dependency of the decision attribute set D on the condition attribute a_i. From this, the positive domain count can be computed, and it is utilized to measure the importance of an attribute to the information system, serving as a relevance indicator in the graph theory weighting function. The definition is as follows:

Definition 5. Let $IVIS = (U, AT, D)$ be an interval-valued information system. $\forall A \subseteq AT$, $a_i \in A$, the importance of a_i relative to A is defined as

$$Imp(a_i) = \frac{POS_{a_i}(D)}{|U|}. \tag{7}$$

According to Eq. (7), the importance of attribute a_i is defined as the ratio of its positive domain count to the total number of objects. This measure demonstrates the impact of attribute a_i on the classification capability of the information system. Finally, we use $R_{ij} = max(Imp(a_i), Imp(a_j))$ normalized as a relevance indicator for subsequent calculations.

4.2.2. The redundancy metrics

We utilize the SRCC to characterize the redundancy among features. According to Definition 6, the dominance relations under a subset of attributes can be determined, which leads to the definition of the dominating and dominated sets.

Definition 6. Let $IVIS = (U, AT, F)$ be an interval-valued information system, $\forall A \subseteq AT$, the knowledge granules about $x_i \in U$ induced by $R^{\preceq}_A$ can be divided into dominating and dominated sets in terms of A are defined as

$$R^{+}_A(x_i) = \{x_j \in U \mid x_i R^{\preceq}_A x_j\},$$

$$R^{-}_A(x_i) = \{x_j \in U \mid x_j R^{\preceq}_A x_i\}.$$

The rank order of the object x_i within attribute h_j can be determined by utilizing the cardinality of the dominating or dominated sets of x_i. Further definitions of the calculation method for the SRCC are provided.

Definition 7. Let $IVIS = (U, AT, F)$ be an interval-valued information system, $\forall a_i, a_j \in AT$, SRCC of a_i and a_j is defined as

$$\delta_{i,j} = 1 - \frac{6\sum_{k=1}^{m}(\mid R^{-}_{\{a_i\}}(x_k) \mid - \mid R^{-}_{\{a_j\}}(x_k) \mid)^2}{m(m^2-1)}. \tag{8}$$

SRCC can reveal the redundancy between pairwise features by measuring the strength and direction of the monotonic relationship between two variables. It helps identify whether features in the dataset carry similar information. When two features have a high SRCC value, it indicates a strong correlation between them, suggesting redundancy. Therefore, we desire minimal redundancy, which is why the redundancy metrics in the weighting function is denoted as $N_{ij} = 1 - \mid \delta(a_i, a_j) \mid$.

4.3. Feature ranking process

In this study, the feature ranking process within graph theory is achieved through the utilization of matrix power series properties. Compared to other path selection methods like shortest path selection and random walk, MPS systematically integrates multiple paths between nodes through high-order matrix powers, capturing the global influence of features. The shortest path considers only the optimal path, neglecting relationships from other potential paths, while random walk, due to its randomness, can lead to instability in high-dimensional data. MPS comprehensively accounts for all path contributions between features, making the ranking more robust and adaptable.

Then we further explain the process of using MPS to achieve feature ranking. Let $\theta = \{\vec{v}_0 = i, \vec{v}_1, \ldots, \vec{v}_{t-1}, \vec{v}_t = j\}$ denote a path of length t between nodes $\vec{v}_0$ and $\vec{v}_t$, which represent features a_i and a_j respectively in a weighted undirected fully connected graph $G = \langle V, E\rangle$. Assuming that the count of nodes n in graph G exceeds the length t of the path θ, the path can then be straightforwardly considered as a subset of the feature set AT.

Subsequently, the overall weight associated with θ is defined as follows:

$$\mu_\theta = \prod_{k=0}^{t-1} Q(\vec{v}_k, \vec{v}_{k+1}). \tag{9}$$

With μ_θ representing all the feature pairs it encompasses, and given that there may be multiple paths of length t connecting nodes $\vec{v}_0$ and $\vec{v}_t$, let us define the set $\psi^t_{i,j}$ as comprising all paths of this length between the two nodes $\vec{v}_0$ and $\vec{v}_t$. Consequently, we utilize the following sum:

$$\Delta_t(i,j) = \sum_{\theta \in \psi^t_{i,j}} \mu_\theta \tag{10}$$

to articulate the contribution of all these paths. Following the principles of matrix algebra, it follows that

$$\Delta_t = Q^t. \tag{11}$$

Proposition 2. *Let $G = \langle V, E\rangle$ be a weighted undirected fully connected graph where V is the set of nodes and E is the set of edges, with each edge weight being positive. For any two nodes $u, v \in V$, define $d(u,v)$ as the shortest path length between u and v. Let $diam(G)$ denote the diameter of graph G, i.e., $diam(G) = \max_{u,v\in V} d(u,v)$. For sufficiently large t, the number of paths P from u to v of length at least t, denoted by $\mid P_t(u,v) \mid$, increases with t, but the limit of the average path length between u and v is $diam(G)$.*

Proof. Since G is connected, the shortest path length $d(u,v)$ exists and is finite for any pair of nodes u, v. The diameter of graph G, $diam(G)$ is the maximum shortest path length. Considering all paths from node u to node v with length t, as t approaches infinity, the number of such paths may increase indefinitely. Despite the existence of longer paths, the shortest path length $d(u,v)$ provides a fixed lower bound. When t exceeds $diam(G)$, the shortest path length between any pair of nodes will not surpass $diam(G)$. Therefore, for any sufficiently large t, the length of paths P will converge to $diam(G)$.

In the context of feature selection, feature a_i and feature a_j may be directly related or interconnected through other features. When the number of intermediary connections between a_i and feature a_j is fixed at $t-1$, $\psi^t_{i,j}$ encompasses all such potential scenarios. Thus, we can assess the individual feature score for feature a_i at a specified path length t as follows:

$$\rho_t(i) = \sum_{j\in V} \Delta_t(i,j) = \sum_{j\in V} Q^t(i,j). \tag{12}$$

Moreover, by incorporating all possible path lengths, the set of paths can be viewed as covering all the subsets of feature set AT, which can be calculated as

$$\rho(i) = \sum_{t=1}^{\infty} \rho_t(i) = \sum_{t=1}^{\infty}(\sum_{j\in V} \Delta_t(i,j)) = \sum_{t=1}^{\infty}(\sum_{j\in V} Q^t(i,j)). \tag{13}$$

Eq. (13) assesses the value of feature a_i across all subsets of features. Clearly, the larger the value $\rho(i)$, the better. Direct computation of ρ as per Eq. (13), however, is unfeasible due to the potentially infinite nature of the calculations involved. Therefore, it is sensible to investigate whether the computation can be simplified by utilizing the convergence properties of power series in algebra. Let Γ denote the power series for the adjacency matrix Q: $\Phi = \sum_{t=1}^{\infty} Q^t$. It should be noted that Φ can be utilized to derive $\rho(i)$ as

$$\rho(i) = \sum_{t=1}^{\infty} \rho_t(i) = [(\sum_{t=1}^{\infty} E^t)\beta]_i = [\Phi\beta]_i. \tag{14}$$

where β represents a one-dimensional vector of ones, and the square brackets denote the extraction of a vector element at the specified index i.

The issue is that the power series $\sum_{t=1}^{\infty} Q^t$ might not converge. In such instances, regularization becomes necessary through the use of generating functions [38], which are typically employed to assign a consistent value to a potentially divergent series. Various types of generating functions exist [39]. The generating function for the t-path is defined as follows:

$$\breve{\rho}(i) = \sum_{t=1}^{\infty} c^t \rho_t(i) = \sum_{t=1}^{\infty}\sum_{j\in V} c^t \Delta_t(i,j). \tag{15}$$

where c is a real-valued regularization factor, and c^t can be interpreted as the weight for paths of length t. The parameter c is defined as $c = 0.9/\varphi(Q)$, where $\varphi(Q)$ is the spectral radius of Q, ensuring the convergence of the infinite sum.

Proposition 3. *For any given adjacency matrix Q of a weighted undirected fully connected graph G, there exists a regularization factor c, and $\varphi(Q)$ denotes the spectral radius of Q. If c satisfies $0 < c < (1/\varphi(Q))$, then the matrix power series $\sum(cQ)^t$ are used for feature ranking converges.*

Proof. Assuming $\{\lambda_0, \lambda_1, \ldots, \lambda_{n-1}\}$ as the eigenvalues of the matrix Q and given the identity matrix E, according to linear algebra, we can define the spectral radius $\varphi(S) = max_{\lambda_i\in\{\lambda_0,\ldots,\lambda_{n-1}\}}(\mid \lambda_i \mid)$. For the theory of convergence of matrix series, we have $\sum_{t\to\infty} Q^t = 0 \Leftrightarrow \varphi(Q) < 1 \Leftrightarrow \sum_{t=1}^{\infty} Q^t = (E-Q)^{-1} - E$. Then, Gelfand's formula states that, we have $\varphi(Q) = \lim_{t\to\infty} \parallel Q^t \parallel^{1/t}$. For each pair of matrices Q and M, we have $\varphi(QM) \le \varphi(Q)\varphi(M)$. When $M = cE$, we have $\varphi(cQ) = \varphi((cE)Q) \le \varphi(cE)\varphi(Q) = c\varphi(Q)$. Thus, by choosing c, which satisfies $0 < c < (1/\varphi(Q))$, the power series of the matrix in the

definition of $\breve{\rho}(i)$ would be convergent because $0 < \varphi(cQ) = c\varphi(Q) < (1/\varphi(Q))\varphi(Q) = 1$. Therefore, we obtain $\breve{\Phi} = \sum_{t=1}^{\infty}(cQ)^t = (E-cQ)^{-1} - E$. In all the experiments conducted in this article, we consistently use $c = (0.9/\varphi(Q))$ with c belonging to the interval $(0, (1/\varphi(Q)))$.

Utilizing the convergence property of a matrix power series, $\breve{\rho}(i)$ can be efficiently computed, thereby facilitating the derivation of a unique score for each feature through marginalization.

$$\breve{\rho}(i) = [\breve{\Phi}\beta]_i. \tag{16}$$

As per the article's definition of feature scores, the higher the value, the stronger the relevance and the lower the redundancy of the feature compared to others, indicating better classification capability. Consequently, features with elevated scores ought to be retained as the reduced set, discernible from the descending sequence of the $\breve{\rho}$ vector.

5. The GLSFS algorithm

Algorithm 1 completes feature ranking following the process outlined below. Specifically, in Algorithm 1, we initially treat the entire attribute set AT as the attribute set C, which facilitates the iteration over the initial attributes. Setting the positive region count to zero ensures accurate recounting after each spatial division. From steps 3 to 6, for each attribute a_i in set C, an iterative process generates a space division tree. For each space division, the samples in the positive region are counted, and the $Imp(a_i)$ is computed according to Definition 5. From steps 7 to 12, the SRCC between features is calculated, and the weights on the graph edges are derived based on the calculated $Imp(a_i)$ and $\delta(a_i, a_j)$ values. Finally, from steps 18 to 25, the eigenvalues of the adjacency matrix Q are computed, with the largest eigenvalue used to calculate c, yielding the feature ranking results. Based on a predefined cutoff proportion p, the top-ranked features are selected to form the final optimal feature subset B.

Subsequently, we calculated the time complexity of this algorithm. Let M denote the number of samples and N denote the number of conditional attributes. The time complexity of the algorithm 1 consists of three components. From steps 3 to 6, the generation of the space partitioning tree and the computation of $Imp(a_i)$ have a time complexity of $O(nmlogm)$. From steps 7 to 12, the computation of SRCC and the adjacency matrix Q has a time complexity of $O(mn^2)$. Finally, using matrix transformations and inversions for feature ranking through MPS has a time complexity of $O(n^3)$. Thus, the overall time complexity of Algorithm 1 is $O(mn^2 + n^3)$. Given this complexity, our algorithm demonstrates significant advantages in accelerating computations and effectively reducing time consumption when processing large-scale sample datasets.

Next, a brief example is provided to illustrate the algorithm described above.

Example: Table 3 shows an $IVIS = (U, AT, F)$, this IVIS is an interval-valued dataset obtained by preprocessing a real-valued dataset that contains information from 9 patients treated with immunotherapy for warts. The attributes include gender, age, duration of treatment, number and size of warts, immune response (induration diameter), and treatment outcome. $U = \{x_1, x_2, \dots, x_9\}$ is a set of objects, $AT = \{a_1, a_2, \dots, a_7\}$ is a set of conditional attributes, D is the decision attribute. When $\alpha = 0.7$, $\theta = 0.001$ and $r = 1$, we have

$POS_{a_1}(D) = 0$, $POS_{a_2}(D) = 6$, $POS_{a_3}(D) = 6$, $POS_{a_4}(D) = 6$,

$POS_{a_5}(D) = 0$, $POS_{a_6}(D) = 3$, $POS_{a_7}(D) = 7$,

$Imp(a_1) = 0.0000$, $Imp(a_2) = 0.6667$, $Imp(a_3) = 0.6667$, $Imp(a_4) = 0.6667$,

$Imp(a_5) = 0.0000$, $Imp(a_6) = 0.3333$, $Imp(a_7) = 0.7778$.

Afterwards, perform min–max normalization and take the maximum value.

$R_{12} = R_{21} = 0.8571$, $R_{13} = R_{31} = 0.8571$, $R_{14} = R_{41} = 0.8571$,

Algorithm 1: $GLSFS$.

Input:

1. A Interval-valued information system $IVIS = (U, AT, D)$, where $U = \{x_1, x_2, \dots, x_m\}$, $AT = \{a_1, a_2, \dots, a_n\}$;
2. The parameters α and θ, the radius of neighborhood r, where $\alpha \in [0, 1]$, $\theta \geq 0$;
3. A cutoff proportion p, where $p \in (0, 1)$, representing the proportion of features to select.

Output: The optimal feature subset B, the subset of features comprising the top p proportion of features based on the feature score ranking $\breve{\rho}$.

```
1  begin
2      C is initialized to AT.
3      for a_i in C do
4          Generate a space division tree on a_i;
5          Calculate Imp(a_i) according to Definition 5;
6      end
7      for i ← 1 to n do
8          for j ← 1 to n do
9              for k ← 1 to m do
10                 Calculate R^-_{a_i}(x_k) and R^-_{a_j}(x_k);
11             end
12             compute δ(a_i, a_j);
13             R_ij ← max(Imp(a_i), Imp(a_j));
14             N_ij ← 1− | δ(a_i, a_j) |;
15             Q(i, j) ← αR_ij + (1 − α)N_ij;
16         end
17     end
18     calculate adjacency matrix Q;
19     c = 0.9/φ(Q);
20     Φ̆ = (E − cQ)^{-1} − E;
21     ρ̆(i) = [Φ̆β]_i;
22     Sort ρ̆ in descending order;
23     Select the top p proportion of features as the feature subset B;
24     return B.
25 end
```

$R_{15} = R_{51} = 0.0000$, $R_{16} = R_{61} = 0.4286$, $R_{17} = R_{71} = 1.0000$,

$R_{23} = R_{32} = 0.8571$, $R_{24} = R_{42} = 0.8571$, $R_{25} = R_{52} = 0.8571$,

$R_{26} = R_{62} = 0.8571$, $R_{27} = R_{72} = 1.0000$, $R_{34} = R_{43} = 0.8571$,

$R_{35} = R_{53} = 0.8571$, $R_{36} = R_{63} = 0.8571$, $R_{37} = R_{73} = 1.0000$,

$R_{45} = R_{54} = 0.8571$, $R_{46} = R_{64} = 0.8571$, $R_{47} = R_{74} = 1.0000$,

$R_{56} = R_{65} = 0.4286$, $R_{57} = R_{75} = 1.0000$, $R_{67} = R_{76} = 1.0000$.

According to Definition 7, we can obtain

$\delta_{1,1} = \delta_{2,2} = \delta_{3,3} = \delta_{4,4} = \delta_{5,5} = \delta_{6,6} = \delta_{7,7} = 1.0000$,

$\delta_{12} = \delta_{21} = 0.2174$, $\delta_{13} = \delta_{31} = 0.6062$, $\delta_{14} = \delta_{41} = -0.3044$,

$\delta_{15} = \delta_{51} = 0.4743$, $\delta_{16} = \delta_{61} = -0.2598$, $\delta_{17} = \delta_{71} = -0.6957$,

$\delta_{23} = \delta_{32} = 0.1590$, $\delta_{24} = \delta_{42} = 0.5210$, $\delta_{25} = \delta_{52} = 0.4125$,

$\delta_{26} = \delta_{62} = -0.8285$, $\delta_{27} = \delta_{72} = -0.6427$, $\delta_{34} = \delta_{43} = -0.3264$,

$\delta_{35} = \delta_{53} = -0.3195$, $\delta_{36} = \delta_{63} = -0.2500$, $\delta_{37} = \delta_{73} = -0.6193$,

$\delta_{45} = \delta_{54} = 0.7792$, $\delta_{46} = \delta_{64} = -0.5774$, $\delta_{47} = \delta_{74} = 0.2311$,

$\delta_{56} = \delta_{65} = -0.3651$, $\delta_{57} = \delta_{75} = 0.1833$, $\delta_{67} = \delta_{76} = 0.5690$.

And we may acquire the adjacency matrix Q by computation according to Eq. (3):

Table 3
Interval-valued information system of immunotherapy.

(U, AT, D)	a_1	a_2	a_3	a_4	a_5	a_6	a_7	D
x_1	(−0.9951, 0.9951)	(−0.4300, 0.7700)	(−0.3864, 0.6064)	(0.2378, 1.2022)	(0.1656, 1.8344)	(−0.2777, 0.3777)	(0.1965, 1.2235)	1
x_2	(−0.9951, 0.9951)	(−0.5800, 0.6200)	(0.3636, 1.3564)	(−0.4222, 0.5422)	(−0.8344, 0.8344)	(−0.2177, 0.4377)	(−0.1735, 0.8535)	1
x_3	(−0.9951, 0.9951)	(−0.3100, 0.8900)	(−0.1764, 0.8164)	(−0.0422, 0.9222)	(0.1656, 1.8344)	(−0.2477, 0.4077)	(−0.1035, 0.9235)	1
x_4	(0.0049, 1.9951)	(−0.5000, 0.7000)	(−0.0464, 0.9464)	(−0.4222, 0.5422)	(−0.8344, 0.8344)	(−0.0877, 0.5677)	(−0.1035, 0.9235)	1
x_5	(0.0014, 1.9986)	(−0.1046, 0.9246)	(0.3877, 1.6123)	(−0.1023, 0.6623)	(0.2520, 1.7480)	(−0.1588, 0.2188)	(−0.4366, 0.5166)	0
x_6	(0.0049, 1.9951)	(−0.1600, 1.0400)	(−0.0164, 0.9764)	(−0.4222, 0.6442)	(−0.8344, 0.8344)	(−0.2977, 0.3577)	(−0.5035, 0.5235)	1
x_7	(−0.9986, 0.9986)	(−0.0546, 0.9746)	(−0.2523, 0.9723)	(−0.0523, 0.7123)	(0.2520, 1.7480)	(−0.1288, 0.2488)	(−0.4066, 0.5466)	0
x_8	(0.0014, 1.9986)	(0.1154, 1.1446)	(0.2977, 1.5223)	(0.1777, 0.9423)	(−0.2480, 1.2480)	(−0.1688, 0.2088)	(−0.4166, 0.5366)	0
x_9	(0.0049, 1.9951)	(−0.4300, 0.7700)	(0.1836, 1.1764)	(−0.2622, 0.7022)	(−0.8344, 0.8344)	(−0.2277, 0.4277)	(−0.4235, 0.6035)	1

$$Q = \begin{pmatrix} 0.0000 & 0.8348 & 0.7181 & 0.8087 & 0.1577 & 0.5221 & 0.7913 \\ 0.8348 & 0.6000 & 0.8523 & 0.7437 & 0.7762 & 0.6514 & 0.8072 \\ 0.7181 & 0.8523 & 0.6000 & 0.8021 & 0.8041 & 0.8250 & 0.8142 \\ 0.8087 & 0.7437 & 0.8021 & 0.6000 & 0.6662 & 0.7268 & 0.9307 \\ 0.1577 & 0.7762 & 0.8041 & 0.6662 & 0.0000 & 0.4905 & 0.9450 \\ 0.5221 & 0.6514 & 0.8250 & 0.7268 & 0.4905 & 0.3000 & 0.8293 \\ 0.7913 & 0.8072 & 0.8142 & 0.9307 & 0.9450 & 0.8293 & 0.7000 \end{pmatrix}.$$

Subsequently, we proceed to obtain the eigenvalues of matrix Q and carry out the subsequent computations as follows:

$$\{\lambda_0, \lambda_1, \lambda_2, \lambda_3, \lambda_4, \lambda_5, \lambda_6\} = \{4.9236, -1.0753, -0.4545, -0.3245, -0.1609, -0.0451, -0.0633\}$$

$$c = 0.9/\varphi(Q) = 0.9/4.9236 = 0.1828$$

$$\breve{\Phi} = (E - cQ)^{-1} - E$$

$$= \begin{pmatrix} 0.7988 & 1.1587 & 1.1686 & 1.1640 & 0.8320 & 0.9575 & 1.2522 \\ 1.1587 & 1.4156 & 1.4957 & 1.4511 & 1.1560 & 1.2273 & 1.5817 \\ 1.1686 & 1.4957 & 1.4899 & 1.4977 & 1.1902 & 1.2876 & 1.6237 \\ 1.1640 & 1.4511 & 1.4977 & 1.4365 & 1.1472 & 1.2495 & 1.6141 \\ 0.8320 & 1.1560 & 1.1902 & 1.1472 & 0.8108 & 0.9589 & 1.2860 \\ 0.9575 & 1.2273 & 1.2876 & 1.2495 & 0.9589 & 1.0008 & 1.3678 \\ 1.2522 & 1.5817 & 1.6237 & 1.6141 & 1.2860 & 1.3678 & 1.7066 \end{pmatrix}$$

$$\breve{p}(i) = [\breve{\Phi}\beta]_i = (7.3318, 9.4861, 9.7534, 9.5601, 7.3810, 8.0494, 10.4321)_i^T.$$

Ultimately, we can determine the feature ranking as $a_7, a_3, a_4, a_2, a_6, a_5$ and a_1, and by setting the cutoff proportion $p = 0.65$, we select the top 65% of the ranked features as the optimal feature subset B. Thus, we obtain $B = \{a_7, a_3, a_4, a_2\}$.

6. Experiments and analysis

To evaluate the efficacy and performance of the proposed GLSFS algorithm in feature selection, a comprehensive set of experiments was conducted in this section.

Below is a detailed description of the computer configuration used for the experiments: The CPU is an AMD R7-5800H with a base frequency of 3.2 GHz and a boost frequency of 4.4 GHz. The system has 16 GB of memory and runs on a 64-bit Windows 11 operating system. All algorithms were written in Python and executed on the PyCharm platform. To carry out the experiments, we downloaded 12 datasets from the UCI Machine Learning Repository and Kaggle, as shown in Table 4. The code is released as an open library called GLSFS: https://github.com/weathin/GLSFS. Our experiment is specifically divided into the following steps:

(1) *Data preprocessing*: Since the features of all 12 datasets were in real numbers, we first converted them into interval value datasets to meet the requirements of the algorithm. Given a real-valued information system $IS = (U, AT, f)$. Subsequently, for each $a \in AT$ and $x \in U$, we set $a(x)^L = a(x) - 2std$ and $a(x)^R = a(x) + 2std$, where std represents the standard deviation of the information values for objects that belong to the same class as object x under the attribute.

Table 4
Summary of the experimental datasets.

No.	Datasets	Abbreviation	Objects	Attributes	Classes
1	Immunotherapy	Immunotherapy	90	8	2
2	Ionosphere	Ionosphere	351	34	2
3	Dermatology	Dermatology	358	35	6
4	Connectionist Bench	sonar	208	61	2
5	Turkish Music Emotion	Acoustic	400	51	2
6	German credit card	GCC	1000	17	2
7	Autism screening data for toddlers	AST	1054	19	2
8	Contraceptive method choice	cmc	1473	9	3
9	Abalone	Abalone	4177	9	29
10	Shill Bidding	Bid	6321	10	2
11	Nursery	Nursery	12 960	9	5
12	Letter-recognition	letter	20 000	17	26

(2) *Performance evaluation*: The analysis of time complexity and the comparison of reduction time, classification accuracy, and the number of reduced features with the existing seven feature selection algorithms.

To facilitate the comparison of the GLSFS algorithm with other algorithms, we set the proportion of features selected by the GLSFS algorithm in the obtained feature ranking results to 45%, 65%, and 85%, respectively. We used four classifiers, KNN, SVM [42], Bayes, and CART, to evaluate the classification performance of these three different feature proportions, along with the other seven comparison algorithms. The results obtained by the classifiers were compared with the classification results of the original dataset to determine the classification accuracy of each algorithm. All algorithms were validated based on the number of selected features and classification accuracy using five-fold cross-validation on the training set.

(3) *Statistical Testing*: Two hypothesis testing experiments were designed to further analyze the differences in classification performance between our algorithm and the comparison algorithms.

(4) *Parameter sensitivity analysis*: The sensitivity of the three parameters α, θ, and r used in constructing the algorithm was analyzed, and the impact of different parameter combinations on the algorithm's final classification performance was explored.

In our tested algorithm, the parameter α is introduced to control the balance between the relevance principle and the non-redundancy principle in the graph theory weight function, with its range set from 0 to 1 and a step size of 0.1. Parameter θ represents the allowable degree of deviation when sorting the sample interval values, and is set to 0, 0.001, 0.01, and 0.1, respectively. Additionally, parameter r represents the neighborhood radius, which is a discrete value, and in the experiment, the neighborhood radius r is searched within the range of 1 to 5 with a step size of 1.

(5) *Robustness evaluation*: The robustness of the GLSFS algorithm was evaluated using noise experiments.

We added Gaussian noise with noise levels (i.e., noise standard deviations) of 0.1, 0.2, 0.3, 0.4, and 0.5 to each dataset in Table 4, and then performed feature selection on the noise-contaminated datasets.

(6) *Uncertainty measurement of subset*: The relationship between uncertainty and the feature subset is analyzed to further explore the interpretability of the GLSFS.

6.1. Performance evaluation of GLSFS algorithm

In this subsection, we conducted a detailed analysis of the experimental results on the classification performance of the GLSFS algorithm. During the comparison of algorithm performance, we employed seven distinct feature selection algorithms, which are listed below.

(1) *Original Dataset*: The entire set of conditional attributes present in the original dataset is utilized for the purpose of classification.
(2) *Infinite Feature Selection (Inf-UFS) [15]*: The Inf-UFD algorithm utilizes a filter feature selection framework, where feature subsets are regarded as paths in a graph, with nodes representing features and edges indicating the correlation and redundancy between features.
(3) *Unsupervised Attribute Reduction Based on α-Approximate Equal Relation (AERAR) [40]*: The AERAR algorithm employs approximate equivalence relations to effectuate unsupervised attribute reduction.
(4) *θ-Rough Degree-Based Method (UM) [41]*: The UM algorithm utilizes θ-Rough degree to handle uncertainty and performs unsupervised attribute reduction.
(5) *Hybrid-Kernel-Based Fuzzy Complementary Mutual Information Method (HKCMI) [42]*: The HKCMI algorithm utilizes fuzzy complementary entropy, corresponding complementary conditional entropy, and fuzzy complementary mutual information to score features.
(6) *k-nearest Neighborhood Conditional Mutual Information Method (KNCMI) [43]*: The KNCMI algorithm evaluates the importance of features by combining δ-neighborhood and K-nearest neighbor methods, employing an iterative strategy, and utilizing information entropy for feature selection.
(7) *Unsupervised Method Based on Graph Theory(IGUFS) [44]*: Unsupervised feature selection is performed by establishing a graph where features are represented as nodes, and pairwise relationships between features are represented as edges. Features are scored based on the principles of correlation and redundancy.
(8) *Neural Networks (MIV-BPNN)*: The mean impact value (MIV) serves as the chosen feature evaluation metric, gauging the significance of each variable in relation to the dependent variable. It is employed to quantify the degree to which input neurons influence the output neuron, with the absolute magnitude of MIV indicating the relative importance of this influence.
(9) The graph-based local search feature selection method(GLSFS) proposed in this article.

In the comparative analysis, we have evaluated several feature selection algorithms, with AERAR, UM, and IGUFS specifically tailored for IVIS, while HKCMI, Inf-FS, KNCMI, and MIV-BPNN were originally designed for single-value information systems. To incorporate the latter group into our experiments, we adapted them by transforming the original IVIS into single-value information systems, achieving this conversion by extracting the midpoint value from each interval.

6.1.1. Time complexity and reduction time analysis

We provide a detailed analysis of the time complexity of the GLSFS algorithm and seven other algorithms, along with their reduction times across different datasets.

(1) *Time complexity analysis*: Table 5 shows the time complexity of our GLSFS algorithm and seven other comparison algorithms.

For other algorithms, Inf-UFS and IGUFS have a time complexity that includes an n^3 term, meaning their computational cost increases significantly as the number of features grows. As a result, their performance may be severely limited when dealing with datasets that have a large number of features. AERAR and HKCMI exhibit a time complexity of $O(m^2n)$, indicating a quadratic dependency on the number of samples, which makes their computational cost quite high when working with large sample sizes. The UM algorithm, with a complexity of $O(m^2n^2)$, is suitable for small-scale datasets, but its computational cost may become unmanageable with larger datasets. BPNN combines the computations of neural networks, with a complexity of $O(m_{iter}m(n+h))$, where the cost depends on the number of iterations m_{iter} and the number of hidden neurons h. Despite its powerful capabilities, it also comes with significant computational overhead.

Table 5
Time complexity analysis of feature selection algorithms.

Algorithm	Time complexity
Inf-UFS	$O((1+m)n^3)$
AERAR	$O(m^2n)$
UM	$O(m^2n^2)$
HKCMI	$O(m^2n)$
KNCMI	$O(m^2n)$
IGUFS	$O(mn^3)$
BPNN	$O(m_{iter}m(n+h))$
GLSFS	$O(mn^2+n^3)$

The m represents the number of samples, n denotes the initial number of features, m_{iter} indicates the number of iterations, and h refers to the total number of hidden neurons in the neural network.

GLSFS demonstrates a time complexity of $O(mn^2+n^3)$, and strikes a good balance between the number of samples m and the number of features n. When dealing with large datasets that have both a high number of samples and features, GLSFS is able to maintain a high level of computational efficiency.

Therefore, the performance of GLSFS in terms of time complexity makes it a strong algorithmic choice for large-scale and high-dimensional datasets. It offers effective feature selection capabilities in big data problems encountered in real-world scenarios.

(2) *The reduction time*: To demonstrate the efficiency of the algorithm, we compared the reduction times of eight algorithms across 12 different datasets and plotted the results in Fig. 4. For algorithms with run times exceeding 24 h, we set their values to 0, excluding them from the comparison.

From Fig. 4, we can observe the distribution trends of reduction times for the eight feature selection algorithms across different datasets. Compared to other algorithms, GLSFS shows significantly lower reduction times on most datasets, particularly on large-scale datasets such as AST and letter, where its advantage is more pronounced. Overall, GLSFS consistently maintains low run times across various types of datasets, highlighting its efficiency and adaptability. This stable, low-time performance demonstrates that GLSFS has strong computational efficiency in feature selection, effectively reducing time costs and providing a highly efficient solution for large-scale data processing.

6.1.2. Comparison of classification accuracy

We conducted a comparative analysis of the classification accuracy between the GLSFS algorithm and seven other algorithms. During the experiments, any algorithm whose dataset reduction time exceeded 24 h was terminated due to excessive time costs.

Tables 6–9 display the classification accuracy of these seven algorithms along with our three proposed variants of the GLSFS algorithm (featuring 45%, 65%, and 85% of the features) across KNN, SVM, Bayes, and CART classifiers. The bold numbers in each row of Tables 3–6 represent the highest classification accuracy achieved among the ten algorithms for that dataset. We consider some datasets (Acoustic, AST, cmc, Dermatology, GCC, Immunotherapy, Ionosphere, and sonar) as standard datasets, while others (Abalone, Bid, Nursery, and letter) are classified as large-sample datasets. For the standard datasets, GLSFS achieved the best classification accuracy in 12 records at 45% feature proportion, in 9 records at 65% feature proportion, and in 15 records at 85% feature proportion. However, among the Inf-UFS, AERAR, UM, HKCMI, KNCMI, IGUFS, and BPNN algorithms, only AERAR, HKCMI, KNCMI, and BPNN achieved the best accuracy in 1, 3, 4, and 1

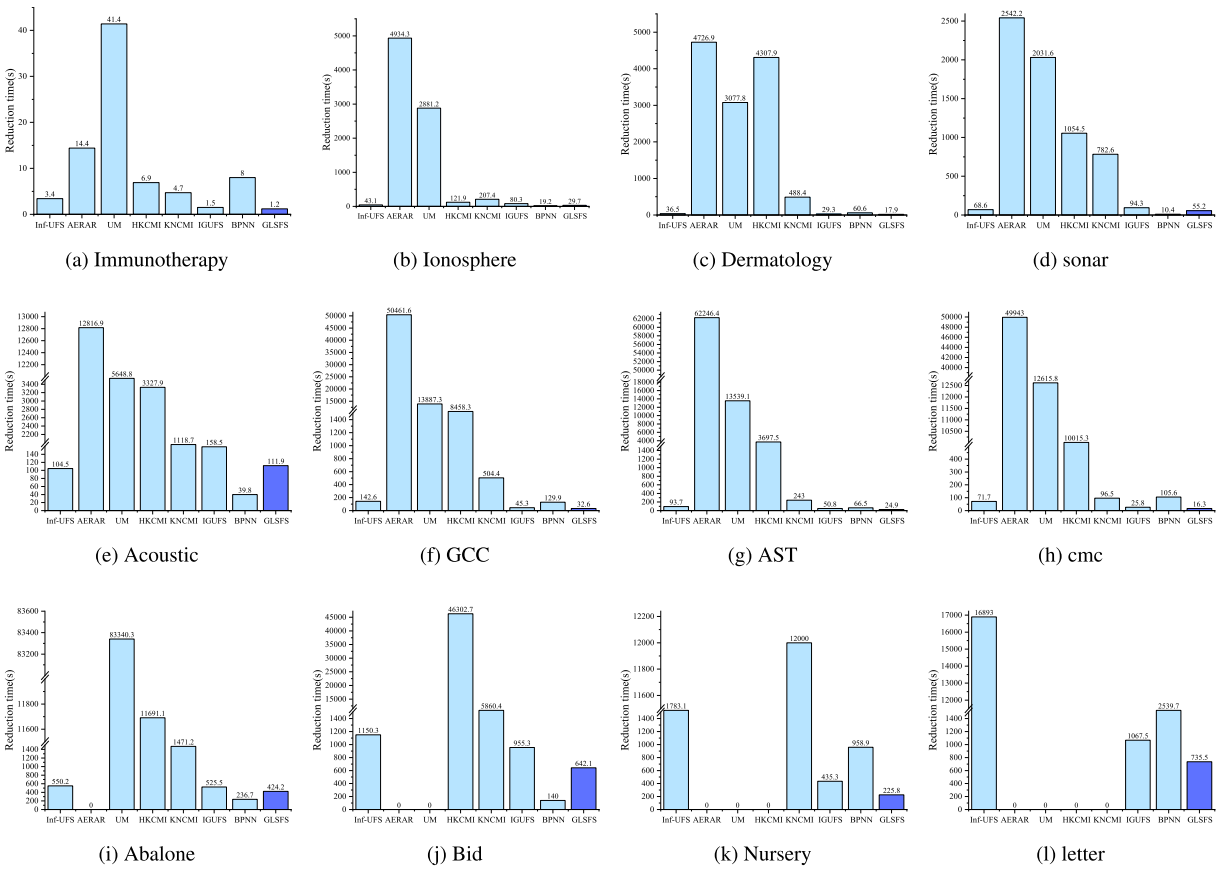


Fig. 4. The reduction time of feature selection algorithms.

records respectively. Therefore, particularly at 45% and 85% feature proportions, the GLSFS algorithm generally outperforms the selected comparative algorithms in terms of average classification accuracy across the eight standard datasets. Regarding the four large-sample datasets used in the experiments, out of 16 records (excluding algorithms without classification accuracy data and comparing only those with available data), GLSFS at 45% feature proportion achieved the best classification accuracy in 5 records, GLSFS at 65% in another 5 records, and GLSFS at 85% in 4 records. The other comparative algorithms managed to reach the best accuracy in only 3 records.

Therefore, the GLSFS algorithm provides high classification accuracy across most datasets at feature proportions of 45%, 65%, and 85%. This demonstrates that the algorithm maintains good performance at different levels of feature retention, particularly achieving optimal accuracy in most cases at an 85% feature proportion. Additionally, the GLSFS algorithm has shown its capability to handle large-scale datasets, as evidenced by the classification accuracy data in the tables. While comparative algorithms such as AERAR, HKCMI, and KNCMI also displayed commendable performance on some datasets, the GLSFS algorithm generally offers more consistent and reliable results across all datasets. Furthermore, the GLSFS algorithm maintains high accuracy across various types of classifiers, demonstrating its good applicability and scalability. This provides important evidence for algorithm selection and deployment in practical applications.

Based on the above analysis, our introduced GLSFS algorithm is capable of either enhancing or sustaining the classification accuracy achieved with the original dataset.

6.1.3. Comparison of reduction numbers

We have presented a comparison of the average number of selected features between the GLSFS algorithm and other comparative algorithms, as shown in Table 10. From the table, it is apparent that these feature selection methods can effectively reduce the number of attributes at different levels. However, a significant advantage of the GLSFS algorithm over others is its ability to flexibly adjust the quantity or percentage of selected features based on practical needs. For instance, For datasets that inherently possess a relatively limited number of features, like those pertaining to Immunotherapy and cmc, should the objective be to attain enhanced classification accuracy, it is advisable to consider an appropriate increase in the percentage of features that are selected, for example, to 65% or 85%. Conversely, for datasets that encompass a larger quantity of features, such as Sonar and Acoustic, it is feasible to decrease the percentage of selected features, for instance, down to 45%, in order to potentially streamline the model while maintaining satisfactory classification performance.

The different feature proportions offered by the GLSFS algorithm provide a flexible strategy to meet the needs of various datasets, allowing adjustments in the proportion of feature selection based on specific requirements for classification accuracy and reduction efficiency. For example, if the goal is to maximize reduction efficiency by significantly reducing the number of attributes with minimal impact on classification accuracy, a lower feature proportion, such as 45% or less, can be selected. For instance, on the Ionosphere dataset, selecting 45% of features is more appropriate than 85%, as reducing the number of attributes to 14, which reduces the attribute reduction proportion by about 40% from the original data, only decreases the classification accuracy from 91.71% to 90.28% using the KNN classifier. However, most alternative feature selection algorithms, including AERAR and UM, commonly produce a fixed set of reduction outcomes that are not capable of being tailored to specific, real-world needs.

Table 6
The classification accuracy of selected features on KNN(%).

Data sets	Original	Inf-UFS	AERAR	UM	HKCMI	KNCMI	IGUFS	BPNN	GLSFS(45%)	GLSFS(65%)	GLSFS(85%)
Immunotherapy	80.00 ± 6.66	72.72 ± 15.23	76.15 ± 9.98	78.75 ± 12.52	70.83 ± 14.97	76.53 ± 14.33	78.75 ± 15.19	75.55 ± 2.72	78.88 ± 4.15	78.88 ± 4.15	**81.11 ± 6.08**
Ionosphere	91.16 ± 2.13	58.21 ± 5.12	52.38 ± 3.01	74.86 ± 10.52	88.00 ± 5.24	89.14 ± 6.36	88.00 ± 5.97	83.19 ± 3.30	90.28 ± 1.80	**92.57 ± 1.06**	91.71 ± 1.39
Dermatology	97.76 ± 1.13	86.60 ± 1.84	94.65 ± 3.01	23.87 ± 6.37	60.37 ± 4.94	95.78 ± 2.30	91.30 ± 4.57	94.96 ± 2.08	93.54 ± 2.07	**98.03 ± 0.89**	**98.31 ± 1.04**
sonar	80.19 ± 2.75	64.80 ± 5.30	**84.52 ± 3.57**	50.69 ± 9.17	64.71 ± 8.49	84.10 ± 7.98	82.14 ± 8.37	80.77 ± 4.28	**84.52 ± 5.95**	79.69 ± 1.53	78.26 ± 2.82
Acoustic	67.75 ± 5.88	38.74 ± 5.68	65.62 ± 5.62	28.82 ± 8.74	41.88 ± 10.22	60.42 ± 9.33	52.14 ± 11.16	39.00 ± 2.66	67.66 ± 3.22	**69.68 ± 3.48**	**69.17 ± 1.67**
GCC	72.87 ± 2.42	63.28 ± 4.15	72.17 ± 1.67	64.56 ± 11.12	63.47 ± 3.33	70.37 ± 2.21	70.77 ± 3.26	64.20 ± 2.83	**78.67 ± 2.65**	73.57 ± 0.99	74.47 ± 1.20
AST	95.63 ± 1.18	62.17 ± 4.85	95.63 ± 1.18	66.47 ± 4.74	66.09 ± 5.70	**97.34 ± 1.58**	90.22 ± 2.37	68.22 ± 2.27	96.44 ± 0.71	94.49 ± 0.50	94.58 ± 0.50
cmc	45.44 ± 1.58	41.85 ± 3.57	45.44 ± 1.58	40.90 ± 4.25	51.21 ± 4.97	43.96 ± 4.46	39.54 ± 5.39	43.58 ± 2.19	**66.03 ± 2.02**	**66.10 ± 1.42**	**66.10 ± 2.41**
Abalone	20.70 ± 0.25	10.46 ± 1.54	–	17.49 ± 4.06	20.55 ± 1.57	21.94 ± 1.97	19.66 ± 2.12	22.95 ± 1.25	**25.62 ± 0.39**	21.54 ± 0.04	22.84 ± 0.60
Bid	98.48 ± 0.25	86.51 ± 0.87	–	–	86.41 ± 0.65	98.66 ± 0.58	88.89 ± 0.99	89.86 ± 0.73	**99.66 ± 0.20**	**99.22 ± 0.37**	98.53 ± 0.35
Nursery	**91.76 ± 0.25**	33.35 ± 1.23	–	–	–	83.74 ± 1.94	73.33 ± 1.71	76.70 ± 2.45	69.35 ± 0.32	75.28 ± 1.40	85.46 ± 7.83
letter	**93.38 ± 0.08**	39.58 ± 4.88	–	–	–	–	89.94 ± 0.46	70.99 ± 0.93	93.06 ± 0.16	91.83 ± 0.47	**93.08 ± 0.43**
Average	77.93 ± 2.03	54.85 ± 4.52	73.32 ± 3.70	49.60 ± 7.94	61.35 ± 6.01	74.98 ± 4.79	72.06 ± 5.13	67.50 ± 2.31	**78.64 ± 1.97**	**78.40 ± 1.36**	**79.47 ± 2.19**

Table 7
The classification accuracy of selected features on SVM(%).

Data sets	Original	Inf-UFS	AERAR	UM	HKCMI	KNCMI	IGUFS	BPNN	GLSFS(45%)	GLSFS(65%)	GLSFS(85%)
Immunotherapy	78.88 ± 8.89	78.88 ± 8.88	78.88 ± 8.88	78.75 ± 12.52	78.75 ± 12.52	79.86 ± 13.79	68.47 ± 13.09	78.88 ± 2.22	**81.11 ± 8.88**	80.00 ± 8.88	80.00 ± 7.85
Ionosphere	93.42 ± 3.20	63.14 ± 3.42	57.61 ± 7.12	74.86 ± 10.52	91.43 ± 3.13	92.29 ± 3.39	90.57 ± 5.12	93.15 ± 2.46	**94.85 ± 2.45**	94.00 ± 2.32	**94.28 ± 2.77**
Dermatology	97.21 ± 2.49	85.19 ± 3.01	94.69 ± 3.69	35.40 ± 5.46	33.96 ± 6.98	33.41 ± 6.35	50.51 ± 6.57	96.92 ± 2.23	94.93 ± 1.70	**98.03 ± 1.05**	**98.03 ± 1.12**
sonar	**83.11 ± 4.25**	65.72 ± 6.29	81.19 ± 3.92	57.98 ± 10.06	70.62 ± 8.97	62.81 ± 10.22	70.05 ± 7.32	82.22 ± 2.83	**83.10 ± 1.23**	82.14 ± 1.75	**83.60 ± 1.89**
Acoustic	78.69 ± 7.37	45.31 ± 6.32	74.75 ± 4.70	32.83 ± 5.64	28.33 ± 6.49	29.58 ± 6.11	62.13 ± 10.24	37.00 ± 3.40	**79.45 ± 2.14**	77.19 ± 3.64	77.19 ± 3.64
GCC	76.37 ± 2.15	70.28 ± 7.37	71.67 ± 1.12	68.36 ± 4.78	69.96 ± 5.43	69.56 ± 4.95	70.13 ± 4.97	70.20 ± 0.50	76.77 ± 2.53	75.97 ± 2.20	**77.87 ± 2.23**
AST	98.76 ± 0.87	69.13 ± 3.46	98.86 ± 1.33	69.13 ± 4.08	28.33 ± 6.49	98.77 ± 1.28	88.88 ± 2.99	69.07 ± 1.66	**99.76 ± 0.23**	98.76 ± 0.64	98.86 ± 0.57
cmc	50.81 ± 2.20	46.83 ± 3.05	45.43 ± 1.64	43.21 ± 3.79	49.80 ± 3.90	46.88 ± 4.24	44.70 ± 3.02	43.71 ± 1.85	46.81 ± 2.32	53.40 ± 2.11	**54.01 ± 1.91**
Abalone	25.43 ± 1.24	16.04 ± 0.91	–	19.16 ± 1.26	**27.16 ± 2.19**	26.75 ± 1.72	26.92 ± 2.19	23.86 ± 0.50	26.62 ± 0.56	25.85 ± 0.04	26.12 ± 0.58
Bid	97.97 ± 0.20	89.22 ± 0.98	–	–	85.13 ± 0.98	99.29 ± 0.33	90.09 ± 1.45	90.21 ± 0.33	**99.43 ± 0.18**	98.73 ± 0.30	97.90 ± 0.19
Nursery	**95.73 ± 0.05**	36.89 ± 0.64	–	–	–	89.22 ± 0.56	77.08 ± 1.38	83.60 ± 0.29	76.79 ± 0.36	83.75 ± 0.34	85.49 ± 7.83
letter	**92.67 ± 0.55**	37.33 ± 1.35	–	–	–	–	**92.03 ± 0.41**	64.51 ± 0.73	73.99 ± 0.02	88.35 ± 0.40	91.26 ± 0.56
Average	**80.75 ± 2.79**	58.66 ± 3.81	75.39 ± 4.05	53.30 ± 6.46	56.35 ± 5.71	66.31 ± 4.84	69.30 ± 4.90	69.44 ± 1.58	77.80 ± 1.88	79.68 ± 1.97	**80.38 ± 2.60**

Table 8
The classification accuracy of selected features on NB(%).

Data sets	Original	Inf-UFS	AERAR	UM	HKCMI	KNCMI	IGUFS	BPNN	GLSFS(45%)	GLSFS(65%)	GLSFS(85%)
Immunotherapy	80.00 ± 8.31	77.86 ± 0.90	64.74 ± 11.69	78.75 ± 12.52	74.31 ± 13.96	77.64 ± 13.04	78.75 ± 12.52	76.66 ± 6.47	**85.55 ± 6.47**	84.44 ± 4.15	83.33 ± 8.16
Ionosphere	88.85 ± 5.81	56.82 ± 5.80	55.10 ± 4.08	74.86 ± 10.52	84.00 ± 5.60	87.71 ± 7.12	87.43 ± 5.74	83.48 ± 2.84	86.57 ± 2.77	86.28 ± 2.45	**89.42 ± 4.37**
Dermatology	86.80 ± 2.86	50.82 ± 8.56	70.79 ± 3.22	35.40 ± 5.46	65.71 ± 6.54	**89.37 ± 5.77**	67.07 ± 7.28	85.98 ± 2.03	87.64 ± 2.91	85.97 ± 2.24	86.80 ± 2.82
sonar	65.66 ± 7.39	65.74 ± 6.94	71.42 ± 4.76	57.02 ± 7.99	63.76 ± 10.53	63.31 ± 10.35	71.50 ± 9.25	66.88 ± 8.66	**73.81 ± 1.19**	69.04 ± 2.73	66.63 ± 3.37
Acoustic	75.18 ± 4.72	36.25 ± 5.87	78.12 ± 1.87	29.56 ± 4.24	51.64 ± 7.26	73.17 ± 5.11	34.83 ± 7.52	37.00 ± 3.40	72.18 ± 1.62	**78.75 ± 3.75**	73.94 ± 2.69
GCC	72.37 ± 2.31	70.11 ± 3.69	73.77 ± 0.90	69.66 ± 5.39	69.86 ± 4.11	72.47 ± 4.30	72.67 ± 4.59	71.90 ± 1.98	**74.77 ± 1.44**	**74.37 ± 1.01**	72.77 ± 1.83
AST	97.24 ± 0.96	67.88 ± 3.73	97.24 ± 0.96	69.13 ± 4.08	69.13 ± 4.08	95.73 ± 2.00	91.55 ± 2.61	95.82 ± 1.31	**99.28 ± 0.71**	94.30 ± 1.06	96.01 ± 0.89
cmc	47.07 ± 3.16	45.50 ± 1.66	47.07 ± 3.16	43.62 ± 3.25	**51.16 ± 3.35**	46.88 ± 3.79	44.16 ± 3.77	47.11 ± 1.89	43.88 ± 2.12	46.87 ± 2.76	50.33 ± 3.62
Abalone	23.75 ± 0.07	16.03 ± 1.49	–	10.72 ± 2.15	22.53 ± 2.38	25.48 ± 2.15	22.87 ± 2.39	22.50 ± 1.22	**26.29 ± 0.79**	**26.09 ± 0.25**	25.43 ± 0.07
Bid	97.04 ± 0.46	89.18 ± 1.03	–	–	89.32 ± 0.69	96.77 ± 0.73	87.64 ± 0.97	90.21 ± 0.33	**98.03 ± 0.27**	**98.03 ± 0.27**	97.42 ± 0.35
Nursery	63.77 ± 0.54	38.79 ± 1.00	–	–	–	60.48 ± 1.55	58.24 ± 1.54	60.01 ± 0.25	57.29 ± 0.09	**64.31 ± 0.34**	63.37 ± 0.52
letter	64.36 ± 0.51	37.58 ± 3.96	–	–	–	–	56.57 ± 0.90	59.51 ± 0.81	54.66 ± 0.11	64.23 ± 0.21	**65.68 ± 0.37**
Average	71.84 ± 3.09	54.38 ± 3.72	69.78 ± 3.83	52.08 ± 6.18	64.14 ± 5.85	71.73 ± 5.08	64.44 ± 4.92	66.42 ± 2.60	**71.66 ± 1.71**	**72.72 ± 1.77**	**72.59 ± 2.42**

Table 9
The classification accuracy of selected features on CART(%).

Data sets	Original	Inf-UFS	AERAR	UM	HKCMI	KNCMI	IGUFS	BPNN	GLSFS(45%)	GLSFS(65%)	GLSFS(85%)
Immunotherapy	**85.55 ± 9.29**	72.86 ± 8.34	64.72 ± 10.82	78.75 ± 12.52	68.61 ± 6.34	69.44 ± 12.97	74.17 ± 12.17	65.55 ± 4.15	81.11 ± 9.29	83.33 ± 6.47	**85.55 ± 4.44**
Ionosphere	83.71 ± 5.39	48.57 ± 3.95	47.10 ± 7.58	74.86 ± 10.52	88.57 ± 4.61	91.71 ± 4.86	90.86 ± 5.12	89.17 ± 4.20	90.00 ± 1.71	90.57 ± 0.57	**92.28 ± 1.93**
Dermatology	92.41 ± 2.87	84.07 ± 6.20	86.51 ± 2.30	34.83 ± 5.97	67.17 ± 7.74	95.23 ± 3.55	87.90 ± 4.96	94.68 ± 2.39	91.00 ± 1.06	**95.50 ± 1.13**	**95.79 ± 1.02**
sonar	65.24 ± 8.58	55.20 ± 4.66	71.42 ± 4.76	52.64 ± 6.06	63.38 ± 8.59	74.36 ± 10.80	75.26 ± 9.13	73.05 ± 3.97	**80.95 ± 7.14**	76.71 ± 1.21	75.79 ± 1.56
Acoustic	59.14 ± 6.32	38.33 ± 5.68	68.75 ± 1.24	36.08 ± 7.46	49.37 ± 4.00	68.14 ± 6.94	45.61 ± 6.46	56.25 ± 2.49	68.67 ± 1.15	68.17 ± 1.45	**69.15 ± 0.81**
GCC	67.66 ± 3.83	60.25 ± 2.81	69.66 ± 2.07	68.36 ± 4.78	67.56 ± 5.68	68.57 ± 4.37	69.46 ± 3.85	66.70 ± 2.78	**71.67 ± 0.37**	70.87 ± 0.59	70.67 ± 0.63
AST	**100.00 ± 0.00**	55.41 ± 3.85	97.24 ± 0.96	69.61 ± 3.82	69.23 ± 3.84	**100.00 ± 0.00**	90.02 ± 3.00	**100.00 ± 0.00**	**100.00 ± 0.00**	**100.00 ± 0.00**	**100.00 ± 0.00**
cmc	46.75 ± 3.31	42.86 ± 5.11	46.40 ± 1.63	41.58 ± 3.59	**55.78 ± 3.97**	43.21 ± 3.7	44.23 ± 3.13	44.80 ± 0.89	45.38 ± 1.91	47.82 ± 1.48	48.70 ± 1.63
Abalone	18.61 ± 2.56	10.89 ± 0.65	–	19.16 ± 1.26	25.31 ± 1.78	**25.48 ± 2.15**	24.93 ± 2.86	20.49 ± 0.99	22.77 ± 0.23	21.42 ± 0.07	21.00 ± 0.01
Bid	99.69 ± 0.17	81.46 ± 1.54	–	–	88.99 ± 0.86	99.49 ± 0.28	90.22 ± 0.68	99.69 ± 0.18	99.63 ± 0.24	99.63 ± 0.22	**99.84 ± 0.07**
Nursery	**99.28 ± 0.36**	37.81 ± 0.93	–	–	–	87.71 ± 0.81	77.24 ± 1.38	83.85 ± 0.27	76.79 ± 0.44	83.75 ± 0.27	83.08 ± 0.86
letter	87.15 ± 0.20	40.33 ± 3.35	–	–	–	–	44.43 ± 0.89	86.97 ± 0.43	79.99 ± 0.15	**87.48 ± 0.67**	**87.89 ± 0.33**
Average	75.43 ± 3.57	52.34 ± 3.92	68.98 ± 3.92	52.87 ± 6.22	64.40 ± 4.74	74.85 ± 4.58	67.86 ± 4.47	73.43 ± 1.66	**75.66 ± 1.97**	**77.10 ± 1.16**	**77.48 ± 1.11**

Table 10
Average number of selected attributes.

Data sets	Original	Inf-UFS	AERAR	UM	HKCMI	KNCMI	IGUFS	BPNN	GLSFS (45%)	GLSFS (65%)	GLSFS (85%)
Immunotherapy	7	3	7	2	3	3	4	3	3	4	5
Ionosphere	33	14	28.4	2	5	15	17	15	14	21	28
Dermatology	34	15	15.4	2	6	28	18	15	15	22	28
sonar	60	27	50	2	2	41	31	27	21	39	51
Acoustic	50	22	17	2	3	49	26	22	22	32	42
GCC	16	9	13.4	2	3	15	11	9	7	10	13
AST	18	8	17	2	2	6	10	8	8	11	15
cmc	8	4	8	2	6	6	5	4	3	5	6
Abalone	8	3	–	2	3	5	5	4	3	5	6
Bid	9	6	–	2	3	6	6	4	4	5	7
Nursery	8	5	–	–	–	4	5	3	3	5	6
letter	16	10	–	–	–	–	9	7	7	10	13
Average	25.5	10.5	19.5	2	3.6	16.2	12.3	10.1	10.2	15.6	20.5

Table 11
Average ranking of classification accuracies of algorithms.

	Inf-UFS	AERAR	UM	HKCMI	KNCMI	IGUFS	BPNN	GLSFS (45%)	GLSFS (65%)	GLSFS (85%)
KNN	2.13	6.13	2.75	3.25	6.75	4.88	4.25	**8.38**	**8.38**	**8.38**
SVM	4.50	5.50	2.25	3.50	4.25	3.88	5.25	**8.75**	**8.13**	**9.38**
NB	3.00	6.00	2.25	4.13	6.00	5.38	5.50	**8.00**	**7.13**	**7.88**
CART	2.25	4.75	2.63	3.88	6.38	5.38	5.13	**8.00**	**8.50**	**9.38**

Therefore, the GLSFS algorithm demonstrates strong advantages in flexibility and practicality in feature selection, allowing it to adjust the feature selection strategy according to different application requirements. This makes it highly applicable in practical settings, especially suitable for processing large datasets or complex datasets with a high number of features. Overall, the GLSFS algorithm is not only efficient but also possesses clear advantages in feature selection for IVIS classification tasks.

6.2. Statistical testing

In this subsection, we utilize two statistical testing methods to further analyze the experimental results of various algorithms: Friedman's test (F-test) and the Wilcoxon test (W-test). These tests are employed to validate the effectiveness of our algorithm. Due to some comparative algorithms failing to produce results within 24 h for large sample datasets, our subsequent analyses focus on the standard datasets previously mentioned. Table 8 displays the average rankings based on classification accuracy for the Inf-UFS, AERAR, UM, HKCMI, KNCMI, IGUFS, BPNN, and GLSFS algorithms across the KNN, SVM, Bayes, and CART classifiers. This approach provides a comprehensive overview of how each algorithm performs under various classification conditions, offering insights into their relative strengths and weaknesses.

The Friedman test, a non-parametric statistical method, tests the null hypothesis that all experimental algorithms exhibit equivalent classification performance. The significance level for this test is set at 0.05. If the P-value exceeds 0.05, the null hypothesis stands; if it is below 0.05, the null hypothesis is rejected. In our comparative analysis involving 8 datasets and 10 algorithms, the Friedman value, which assesses the classification performance of these algorithms, is derived from the average rankings shown in Table 11. According to the results displayed in Table 12, all P-values from the Friedman test fall below the 0.05 significance level, leading to the rejection of the null hypothesis. This indicates that the classification performance of the algorithms under comparison is not equivalent.

Furthermore, the Wilcoxon test is utilized to delve deeper into the relative performance and distinctions among all the comparative algorithms. This test assesses whether the rank differences between two sample datasets are symmetrically distributed around the median, which helps determine whether the two samples originate from the same distribution. In this study, we undertake three distinct sets of hypothesis tests, each aligned with the three distinct optimal feature subset ratios introduced by our method. For each set, the null hypothesis is formulated to assert that the classification accuracy attained by our approach does not surpass that of the seven comparative methods under consideration. The resulting p-values from these rigorous tests are comprehensively presented in Table 13.

Table 12
Result of the Friedman test.

	Friedman value	χ_F^2	P Value
KNN	14.67	48.74	1.85×10^{-7}
SVM	15.33	49.43	1.38×10^{-7}
NB	5.62	32.07	1.94×10^{-4}
CART	30.98	58.73	2.35×10^{-9}

The rejection of the null hypothesis demonstrates that our model attains classification accuracy that surpasses the comparative models, thereby validating the efficacy of our approach. As evidenced by the data in the tables, the test results for both the KNN and SVM classifiers reject the null hypothesis when our model's feature selection ratios are configured at 45% and 65%. This confirms that our model's classification accuracy substantially exceeds that of the models from other comparative algorithms. For the NB and CART classifiers, only the hypothesis tests for the AERAR, KNCMI, and BPNN comparison algorithms do not reject the null hypothesis, potentially due to the limited sample sizes involved.

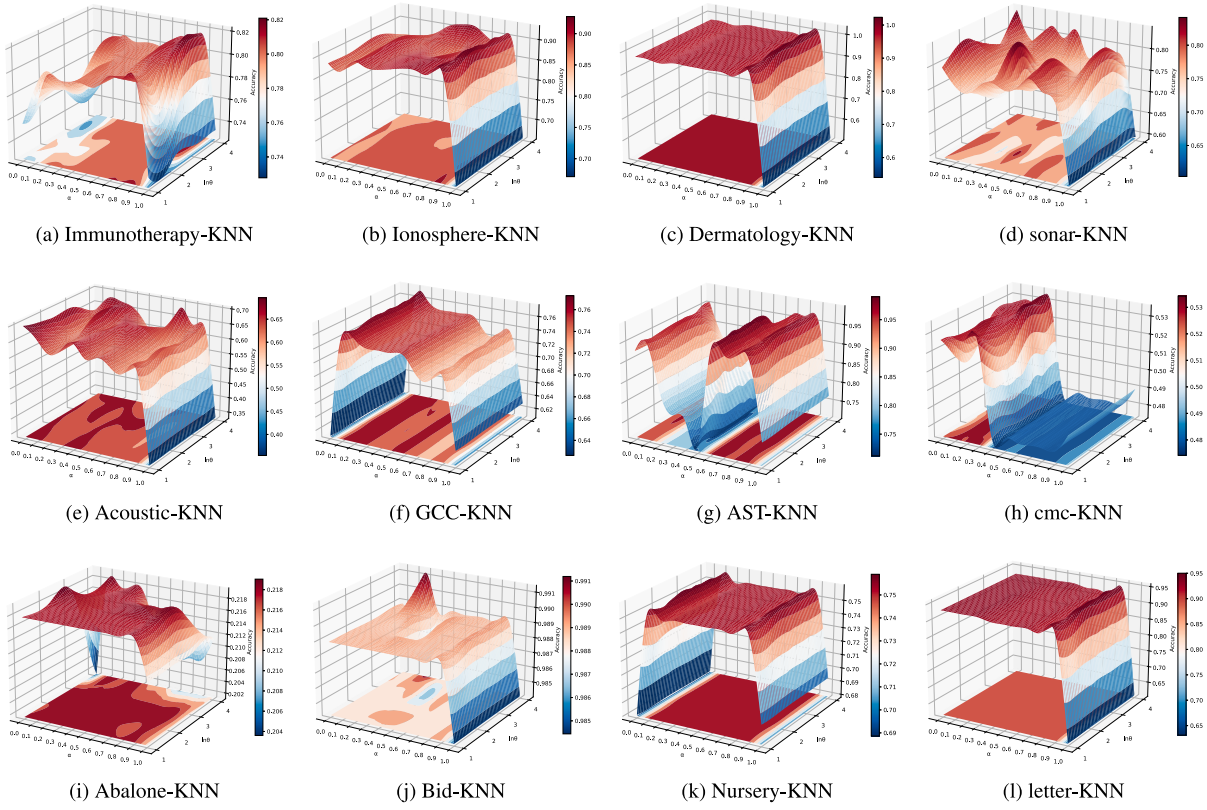
6.3. Parameter sensitivity analysis of GLSFS algorithm

(1) *Sensitivity analysis of parameters α and θ*: To analyze the impact of varying parameter combinations of α and θ on the accuracy of classification achieved by the KNN classifier using the proposed GLSFS algorithm, we plotted the classification accuracy results for 12 datasets at a neighborhood radius $r = 1$, with various combinations of α and θ, as shown in Fig. 5.

The charts clearly demonstrate that selecting different parameter combinations is crucial, as they significantly affect the classification accuracy. From the charts, it is evident that the optimal parameter combinations vary across datasets. For instance, the GLSFS algorithm performs better with higher α values for datasets like Ionosphere,

Table 13
P-value of the Wilcoxon test.

	KNN			SVM			NB			CART		
	45%	65%	85%	45%	65%	85%	45%	65%	85%	45%	65%	85%
Inf-UFS	< 0.05	< 0.05	< 0.05	< 0.05	< 0.05	< 0.05	< 0.05	< 0.05	< 0.05	< 0.05	< 0.05	< 0.05
AERAR	< 0.05	< 0.05	< 0.05	< 0.05	< 0.05	< 0.05	0.25	0.31	0.46	< 0.05	< 0.05	< 0.05
UM	< 0.05	< 0.05	< 0.05	< 0.05	< 0.05	< 0.05	< 0.05	< 0.05	< 0.05	< 0.05	< 0.05	< 0.05
HKCMI	< 0.05	< 0.05	< 0.05	< 0.05	< 0.05	< 0.05	< 0.05	< 0.05	< 0.05	< 0.05	< 0.05	< 0.05
KNCMI	< 0.05	< 0.05	< 0.05	< 0.05	< 0.05	< 0.05	0.38	0.38	0.07	0.23	0.6	< 0.05
IGUFS	< 0.05	< 0.05	0.05	< 0.05	< 0.05	< 0.05	< 0.05	0.05	0.07	< 0.05	< 0.05	< 0.05
BPNN	< 0.05	< 0.05	< 0.05	< 0.05	< 0.05	< 0.05	0.05	0.10	< 0.05	0.06	< 0.05	< 0.05



(a) Immunotherapy-KNN (b) Ionosphere-KNN (c) Dermatology-KNN (d) sonar-KNN

(e) Acoustic-KNN (f) GCC-KNN (g) AST-KNN (h) cmc-KNN

(i) Abalone-KNN (j) Bid-KNN (k) Nursery-KNN (l) letter-KNN

Fig. 5. The impact of parameters α and θ on classification accuracy for twelve datasets when $r = 1$ on the KNN classifier.

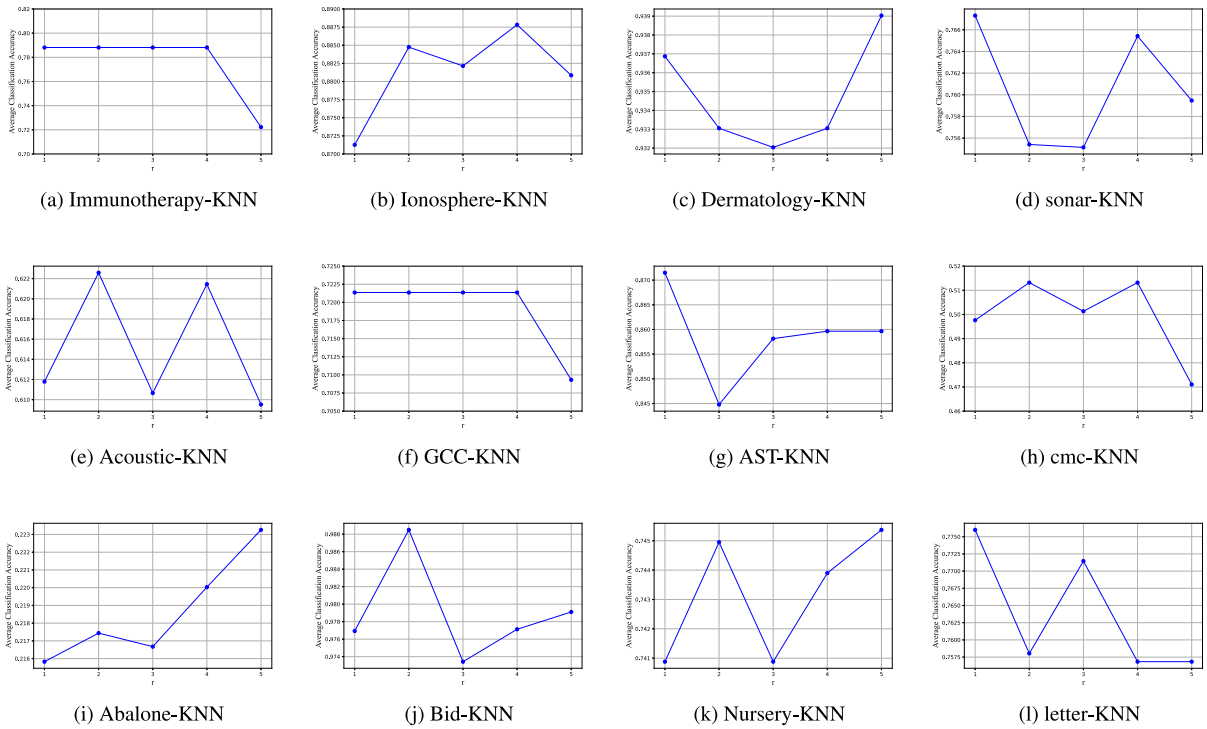
Dermatology, and AST, while lower α values are more suitable for datasets such as Acoustic, GCC, and cmc. Particularly when α is close to 0.9, the GLSFS algorithm achieves superior classification performance for most datasets, indicating that increasing the weight of the relevance principle in the graph theory weight function is essential for enhancing classification performance. Conversely, lower α values yield better results for datasets like Acoustic and GCC, suggesting that reducing the weight of the relevance principle can effectively minimize feature redundancy, thereby improving classification accuracy. The optimal θ values also show considerable variation between datasets. For example, higher θ values (e.g., 0.1) significantly improve classification accuracy in Dermatology and Ionosphere datasets. On other datasets like Bid and Nursery, lower θ values (such as 0.01 or lower) are sufficient to achieve good performance, likely due to the specific distribution and type of features in these datasets.

Furthermore, for most datasets, classification accuracy reaches its peak within specific combination regions of α and θ, further emphasizing the importance of parameter tuning to optimize algorithm performance based on the specific characteristics and requirements of each dataset. The charts also show that when parameters approach their upper or lower limits, the performance of most datasets deteriorates, indicating that over-reliance on a single principle may lead to a decline in performance.

In conclusion, different combinations of parameters significantly affect the final classification accuracy. Therefore, it is crucial to carefully select the appropriate combinations of α and θ to achieve the best possible classification accuracy for the GLSFS algorithm. This flexible adjustment of parameters provides customized solutions for different types of datasets, thereby enhancing overall classification efficiency and accuracy.

(2) *Sensitivity analysis of neighborhood radius r*: We discuss further experiments to understand the relationship between the neighborhood radius and GLSFS performance.

In our approach, we considerably reduced the search range for the neighborhood radius by converting it from a real number to an integer. We plotted the changes in average classification accuracy across 12 datasets as the neighborhood radius was searched within the range of 1 to 5, as shown in Fig. 6. From the figure, it can be observed that different neighborhood radius values r have a certain impact on the KNN classification accuracy across various datasets, with most fluctuations



(a) Immunotherapy-KNN (b) Ionosphere-KNN (c) Dermatology-KNN (d) sonar-KNN

(e) Acoustic-KNN (f) GCC-KNN (g) AST-KNN (h) cmc-KNN

(i) Abalone-KNN (j) Bid-KNN (k) Nursery-KNN (l) letter-KNN

Fig. 6. The average classification accuracy variation for different neighborhood radius(r) on the KNN classifier.

remaining around 0.01 or 0.02. This indicates that the classification performance of the model does not vary significantly with different radius values, although there are some local improvements in accuracy at certain specific values. For instance, in the Acoustic and Bid datasets, the accuracy shows fluctuations with r changes. Furthermore, different datasets exhibit varied trend responses. For example, Immunotherapy shows that accuracy remains relatively stable before decreasing as r increases; meanwhile, Abalone displays a gradual increase in accuracy as r increases. These results suggest that it is necessary to adjust r to optimize the classification for specific datasets. Priority should be given to selecting neighborhood radius values that show stable or locally optimal performance across multiple datasets, while avoiding excessively large or small radius values that may negatively impact model performance.

6.4. Robustness evaluations of GLSFS

In this subsection, we evaluate the robustness of the feature selection algorithm based on the feature subsets.

To evaluate the robustness of the GLSFS feature selection algorithm and demonstrate the advantages of data processing, Gaussian noise with noise levels (i.e., standard deviations of the noise) of 0.1, 0.2, 0.3, 0.4, and 0.5 was added to the feature values of each dataset listed in Table 4. Gaussian noise is a common type of random noise, characterized by a normal distribution with a defined mean and variance. As a result, the same algorithm can generate six feature subsets: one from the original clean data and five from datasets with varying noise levels. The similarity between these subsets serves as a measure of the algorithm's robustness.

For an algorithm that outputs feature subsets, let $E_i = \{e_1^i, e_2^i, \ldots, e_{|E|}^i\}$ and $F_j = \{f_1^j, f_2^j, \ldots, f_{|F|}^j\}$ be the feature subsets obtained at noise levels i and j, respectively. The similarity between these two feature subsets is calculated as

$$S(E_i, F_j) = \frac{|\ E_i \cap F_j\ |}{|\ E_i\ | + |\ F_j\ | - |\ E_i \cap F_j\ |}. \tag{17}$$

At different noise levels, the same algorithm generated six sets of feature subsets. The similarity between each pair of subsets was calculated to construct a similarity matrix $S = [s_{ij}]_{6\times6}$. Each element s_{ij} in the matrix represents the degree of similarity between the feature subsets generated under noise levels i and j. To assess the algorithm's robustness, the average value of all elements in matrix S was used as the overall evaluation metric. A higher average value indicates greater stability of the algorithm in handling noise interference.

Since the GLSFS algorithm selects feature subsets by truncating based on the ranking results, 65% of the top-ranked features were chosen as the feature subset in this experiment. To further validate the performance and robustness of GLSFS, we compared it against seven other algorithms. Table 14 presents the experimental results, with the highest similarity value in each row highlighted in bold.

From the Table 14, we can observe that GLSFS outperforms other feature selection algorithms on most datasets. Particularly on datasets such as Ionosphere, Abalone, and Nursery, GLSFS demonstrates significant advantages, indicating its strong adaptability to various datasets and feature selection requirements. In terms of average performance, GLSFS shows superior results in the average similarity metric compared to other algorithms, reflecting its robust noise resistance. Even at high noise levels, GLSFS can reliably select feature subsets similar to the original, ensuring consistency and effectiveness in data processing.

It is worth noting, however, that on certain datasets (such as Dermatology and Sonar), GLSFS does not show a significant advantage over other algorithms. This may be due to specific characteristics in feature distribution or noise sensitivity within these datasets, which affect the performance of GLSFS in feature selection.

Overall, GLSFS exhibits stable robustness and efficiency in handling high-dimensional, complex datasets and noise interference, providing strong support for large-scale data processing and accurate classification.

6.5. Uncertainty measurement of subset

In this subsection, to evaluate the uncertainty of the feature subsets selected by our feature selection method GLSFS, we used the concept of roughness. Roughness is a measure in rough set theory that is well-suited for quantifying uncertainty by measuring boundary regions. A

Table 14
Similarity of feature subsets at different noise levels.

Data sets	Inf-UFS	AERAR	UM	HKCMI	KNCMI	IGUFS	BPNN	GLSFS (65%)
Immunotherapy	0.72	0.70	0.69	0.67	0.59	**0.75**	0.66	0.74
Ionosphere	0.69	0.39	0.37	0.44	0.55	0.49	0.67	**0.70**
Dermatology	0.75	0.44	0.52	0.38	**0.77**	0.66	0.59	0.72
sonar	0.57	0.55	0.59	0.48	0.54	0.69	**0.70**	0.61
Acoustic	0.69	0.54	**1.00**	0.46	0.66	0.66	0.58	0.60
GCC	0.78	0.78	0.65	0.40	0.70	**0.90**	0.59	0.68
AST	0.66	0.76	0.54	0.48	0.69	**0.85**	0.62	0.69
cmc	0.72	0.78	0.68	0.68	0.70	0.89	0.69	**1.00**
Abalone	0.66	–	0.72	0.68	0.69	0.66	0.69	**0.91**
Bid	0.78	–	–	0.61	0.68	0.80	0.70	**0.92**
Nursery	0.76	–	–	–	0.80	0.82	0.75	**1.00**
letter	0.95	–	–	–	–	0.94	**1.00**	0.75
Average	0.73	0.62	0.64	0.53	0.67	0.76	0.69	**0.78**

lower roughness value indicates a more precise boundary and lower uncertainty, suggesting that the selected features effectively capture relevant information for classification.

Let $f(x_i, a_k) = (a_{ik}^L, a_{ik}^R)$ and $f(x_j, a_k) = (a_{jk}^L, a_{jk}^R)$ be two interval values of samples x_i and x_j under the attribute a_k, The similarity of two samples under attribute a_k is defined as

$$Sim(x_i, x_j) = 1 - \min\left(1, \max\left(\frac{a_{ik}^R - a_{jk}^L}{(a_{ik}^R - a_{ik}^L) + (a_{jk}^R - a_{jk}^L)}, 0\right)\right) - \min\left(1, \max\left(\frac{a_{jk}^R - a_{ik}^L}{(a_{jk}^R - a_{jk}^L) + (a_{ik}^R - a_{ik}^L)}, 0\right)\right). \quad (18)$$

Let $IVIS = (U, AT, D, f)$ be an interval-valued information system, where U is the data point set, $A = \{a_1, a_2, \ldots, a_n\} \subseteq AT$ is the conditional attribute set and D is the decision attribute set. D divides U into N equivalence classes, $X_1, X_2, \ldots, X_N$, the lower approximation and the upper approximation of the decision attribute set D under A are respectively defined as

$$\underline{A}^{\epsilon} X(D) = \bigcup_{j=1}^{N} \{x_i \in U \mid S_A^{\epsilon}(x_i) \subseteq X_j\}. \quad (19)$$

$$\overline{A}^{\epsilon} X(D) = \bigcup_{j=1}^{N} \{x_i \in U \mid S_A^{\epsilon}(x_i) \cap X_j \neq \varnothing\}. \quad (20)$$

where $S_A^{\epsilon}(x_i) = \{x_j \mid \sqrt{\sum_{k=1}^{n} (Sim(x_i, x_j))^2} \geq \epsilon, A \subseteq AT, x_j \in U\}$ represents the collection of samples within set A that have a similarity with the interval-valued sample x_i less than ϵ.

Pawlak [41] proposed a numerical measure for evaluating the uncertainty of a rough set X: roughness. It is defined as

$$\gamma_A^{\epsilon}(X) = 1 - \frac{|\underline{A}^{\epsilon} X(D)|}{|\overline{A}^{\epsilon} X(D)|}. \quad (21)$$

Roughness $\gamma_A^{\epsilon}(X)$ reflects the degree of knowledge certainty of set X under relation $S_A^{\epsilon}(x_i)$ and can be effectively used to measure uncertainty.

Compared to the roughness of the original information system, we measure the roughness of the obtained feature subset by selecting the top 65% of the features. To present the measurement results, a line chart comparing the roughness values is drawn, as shown in Fig. 7. The results show that the feature subsets generated by the GLSFS algorithm maintain roughness levels close to those of the original system on most datasets. This indicates that GLSFS effectively preserves the structure and informational integrity of the original information system while selecting features. This trend demonstrates GLSFS's significant advantage in reducing data uncertainty, allowing it to retain system knowledge while reducing feature dimensions, further enhancing the interpretability of the model.

However, on a few datasets (such as GCC and AST), the roughness of the feature subset is slightly higher than that of the original system, which may be due to the specific distribution of data features or the influence of noise, leading to a slight impact on GLSFS's selection performance on these datasets.

In summary, the GLSFS exhibits robust performance on most datasets, generating compact and efficient feature subsets without significantly increasing uncertainty. Its contributions to uncertainty control and interpretability improvement highlight its potential advantages in handling complex and high-dimensional datasets.

7. Conclusion and outlook

7.1. Conclusions

In this study, we integrate graph theory and matrix algebra with neighborhood rough sets to provide a thorough evaluation of attribute significance within interval-valued information systems. Specifically, our approach balances both the contributions of attributes to classification accuracy and their interrelationships, offering a comprehensive framework for feature evaluation. We present a novel feature ranking methodology based on principles of relevance and redundancy, along with three distinct selection ratios to enhance the flexibility of our feature selection algorithm.

The method was rigorously tested against seven benchmark algorithms on 12 publicly available datasets, and the results confirmed the effectiveness of the Graph-based Local Search Feature Selection (GLSFS) algorithm for interval-valued information systems. The method demonstrated significant improvements in classification accuracy and computational efficiency across diverse datasets, highlighting not only the effectiveness of GLSFS in feature selection but also its potential for real-world applications where efficient and accurate data processing is essential.

7.2. Future work and limitations

While our approach offers substantial improvements, several limitations persist. The primary challenge lies in noise resistance; although our method exhibits stronger anti-interference capabilities compared to most benchmark algorithms, its robustness in highly noisy environments still requires further enhancement. Another limitation is the current application scope, as this study focuses solely on static interval-valued datasets. In real-world applications, datasets often possess dynamic characteristics, with attributes or instances evolving over time.

To address these challenges, future work will explore the extension of this methodology to dynamic interval-valued datasets. This extension will focus on developing an adaptive feature selection approach capable of efficient attribute reduction when the dataset evolves. By integrating dynamic data adaptation mechanisms, we aim to enhance the noise resistance of the GLSFS algorithm, thereby improving its robustness in real-world scenarios. Through these extensions, we hope to improve the applicability and resilience of our feature selection algorithm across diverse and dynamic data environments (see Fig. 8).

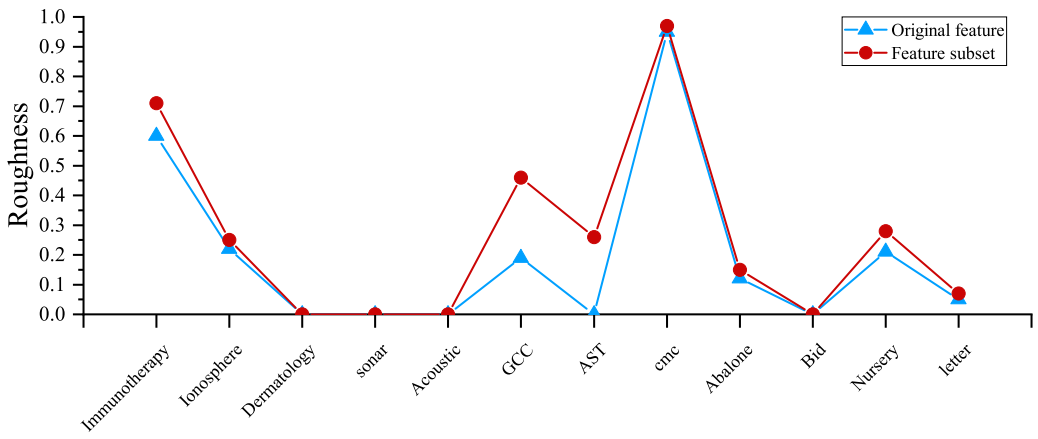


Fig. 7. Roughness of the original feature and the feature subset.

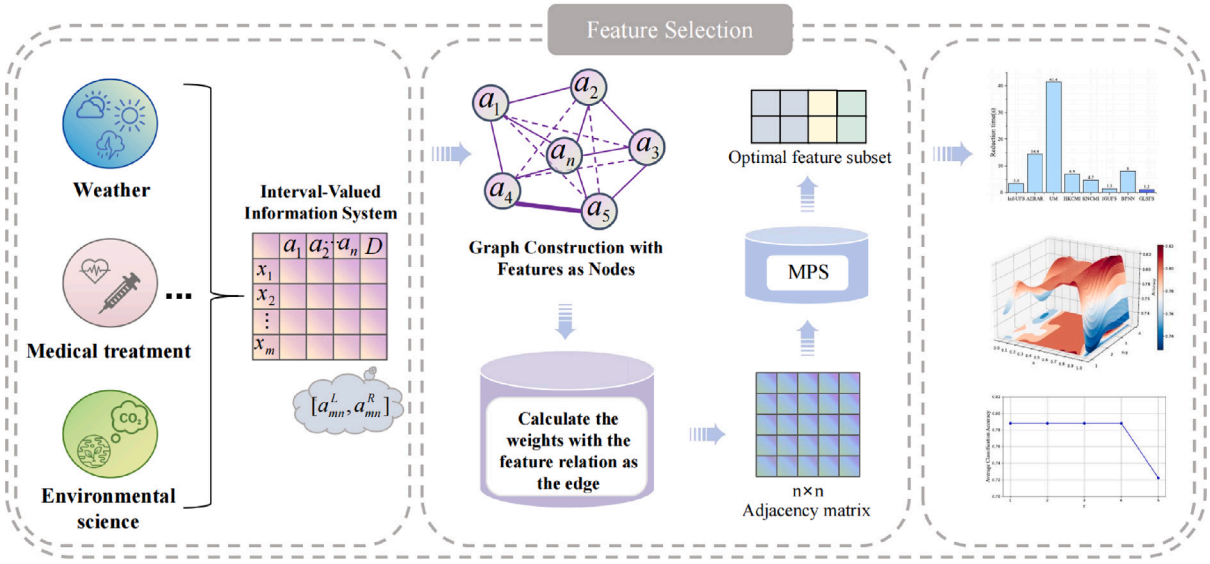


Fig. 8. Graphical summary.

CRediT authorship contribution statement

Xiaoyan Zhang: Validation, Supervision, Project administration, Methodology, Investigation, Funding acquisition, Conceptualization. **Xuan Shen:** Writing – review & editing, Writing – original draft, Visualization, Software, Methodology, Formal analysis, Data curation.

Declaration of competing interest

We wish to confirm that there are no known conflicts of interest associated with this publication and there has been no significant financial support for this work that could have influenced its outcome.

We confirm that the manuscript has been read and approved by all named authors and that there are no other persons who satisfied the criteria for authorship but are not listed. We further confirm that the order of authors listed in the manuscript has been approved by all of us.

We confirm that we have given due consideration to the protection of intellectual property associated with this work and that there are no impediments to publication, including the timing of publication, with respect to intellectual property. In so doing we confirm that we have followed the regulations of our institutions concerning intellectual property.

Acknowledgments

This work was supported by the National Natural Science Foundation of China (Grant NO.12371465) and the Chongqing Natural Science Foundation (Grant NO. CSTB2023NSCQ-MSX1063).

Data availability

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

References

[1] H. Zapata, H. Bustince, S. Montes, et al., Interval-valued implications and interval-valued strong equality index with admissible orders, Internat. J. Approx. Reason. 88 (2017) 91–109.
[2] X. Qi, H. Guo, Z. Artem, et al., An interval-valued data classification method based on the unified representation frame, IEEE Access 8 (2020) 17002–17012.
[3] J. Dai, W. Wang, J.S. Mi, Uncertainty measurement for interval-valued information systems, Inform. Sci. 251 (2013) 63–78.
[4] Y. Li, T. Li, H. Liu, Recent advances in feature selection and its applications, Knowl. Inf. Syst. 53 (2017) 551–577.
[5] Z. Pawlak, Rough sets, Int. J. Comput. Inf. Sci. 11 (1982) 341–356.
[6] S. Roy, P. Shivakumara, N. Jain, et al., Rough-fuzzy based scene categorization for text detection and recognition in video, Pattern Recognit. 80 (2018) 64–82.
[7] T. Shaheen, B. Mian, M. Shabir, et al., A novel approach to decision analysis using dominance-based soft rough sets, Int. J. Fuzzy Syst. 21 (2019) 954–962.

[8] G. Lang, D. Miao, M. Cai, Three-way decision approaches to conflict analysis using decision-theoretic rough set theory, Inform. Sci. 406 (2017) 185–207.
[9] W. Xu, Y. Pan, X. Chen, et al., A novel dynamic fusion approach using information entropy for interval-valued ordered datasets, IEEE Trans. Big Data (2022).
[10] W. Xu, K. Cai, D.D. Wang, A novel information fusion method using improved entropy measure in multi-source incomplete interval-valued datasets, Internat. J. Approx. Reason. 164 (2024) 109081.
[11] J. Dai, Y. Yan, Z. Li, et al., Dominance-based fuzzy rough set approach for incomplete interval-valued data, J. Intell. Fuzzy Systems 34 (1) (2018) 423–436.
[12] T.Y. Lin, Neighborhood systems-A qualitative theory for fuzzy and rough sets, Adv. Mach. Intell. Soft Comput. 4 (1997) 132–155.
[13] Y. Yao, Rough sets, neighborhood systems and granular computing, in: Engineering solutions for the next millennium. 1999 IEEE Canadian Conference on Electrical and Computer Engineering (Cat. No. 99TH8411), Vol. 3, IEEE, 1999, pp. 1553–1558.
[14] Y. Akhiat, Y. Asnaoui, M. Chahhou, et al., A new graph feature selection approach, in: 2020 6th IEEE Congress on Information Science and Tech..
[15] G. Roffo, S. Melzi, U. Castellani, et al., Infinite feature selection: a graph-based feature filtering approach, IEEE Trans. Pattern Anal. Mach. Intell. 43 (12) (2020) 4396–4410, nology (CiSt). IEEE, 2021: 156-161..
[16] X. Xie, Z. Cao, F. Sun, Joint learning of graph and latent representation for unsupervised feature selection, Appl. Intell. 53 (26) (2023) 25282–25295.
[17] C. Tang, X. Zheng, W. Zhang, et al., Unsupervised feature selection via multiple graph fusion and feature weight learning, Sci. China Inf. Sci. 66 (5) (2023) 152101.
[18] R.W. Swiniarski, A. Skowron, Rough set methods in feature selection and recognition, Pattern Recognit. Lett. 24 (6) (2003) 833–849.
[19] Y. Chen, D. Miao, R. Wang, A rough set approach to feature selection based on ant colony optimization, Pattern Recognit. Lett. 31 (3) (2010) 226–233.
[20] J.R. Anaraki, M. Eftekhari, Rough set based feature selection: a review, in: The 5th Conference on Information and Knowledge Technology, IEEE, 2013, pp. 301–306.
[21] N. Zhong, J. Dong, S. Ohsuga, Using rough sets with heuristics for feature selection, J. Intell. Inf. Syst. 16 (2001) 199–214.
[22] R. Jensen, Q. Shen, Interval-valued fuzzy-rough feature selection in datasets with missing values, in: 2009 IEEE International Conference on Fuzzy Systems, IEEE, 2009, pp. 610–615.
[23] W. Li, H. Zhou, W. Xu, et al., Interval dominance-based feature selection for interval-valued ordered data, IEEE Trans. Neural Netw. Learn. Syst. (2022).
[24] Q.-H. Hu, Numerical attribute reduction based on neighborhood granulation and rough approximation.
[25] X. Zhang, Z. Jiang, W. Xu, Feature selection using a weighted method in interval-valued decision information systems, Appl. Intell. 53 (9) (2023) 9858–9877.
[26] B. Sang, H. Chen, L. Yang, et al., Feature selection for dynamic interval-valued ordered data based on fuzzy dominance neighborhood rough set, Knowl.-Based Syst. 227 (2021) 107223.
[27] Q. Ke, X. Jing, M. Woźniak, et al., APGVAE: Adaptive disentangled representation learning with the graph-based structure information, Inform. Sci. 657 (2024) 119903.
[28] W. Dong, J. Wu, X. Zhang, et al., Improving performance and efficiency of graph neural networks by injective aggregation, Knowl.-Based Syst. 254 (2022) 109616.
[29] W. Dong, M. Woźniak, J. Wu, et al., Denoising aggregation of graph neural networks by using principal component analysis, IEEE Trans. Ind. Inform. 19 (3) (2022) 2385–2394.
[30] Y. Qian, J. Liang, C. Dang, Interval ordered information systems, Comput. Math. Appl. 56 (8) (2008) 1994–2009.
[31] Q. Hu, D. Yu, J. Liu, et al., Neighborhood rough set based heterogeneous feature subset selection, Inform. Sci. 178 (18) (2008) 3577–3594.
[32] S. Xia, H. Zhang, W. Li, et al., GBNRS: A novel rough set algorithm for fast adaptive attribute reduction in classification, IEEE Trans. Knowl. Data Eng. 34 (3) (2020) 1231–1242.
[33] S. Xia, C. Wang, G. Wang, et al., GBRS: A unified granular-ball learning model of Pawlak rough set and neighborhood rough set, IEEE Trans. Neural Netw. Learn. Syst. (2023).
[34] S. Xia, S. Wu, X. Chen, et al., GRRS: Accurate and efficient neighborhood rough set for feature selection, IEEE Trans. Knowl. Data Eng. (2022).
[35] E.C. Fieller, H.O. Hartley, E.S. Pearson, Tests for rank correlation coefficients. I, Biometrika 44 (3/4) (1957) 470–481.
[36] G. Bierman, Power series evaluation of transition and covariance matrices, IEEE Trans. Autom. Control 17 (2) (1972) 228–232.
[37] N.J. Young, The rate of convergence of a matrix power series, Linear Algebra Appl. 35 (1981) 261–278.
[38] R.L. Graham, Concrete Mathematics: A Foundation for Computer Science, Pearson Education India, 1994.
[39] E. Elizalde, Ten Physical Applications of Spectral Zeta Functions, Springer, 2012.
[40] X. Liu, J. Dai, J. Chen, et al., Unsupervised attribute reduction based on α-approximate equal relation in interval-valued information systems, Int. J. Mach. Learn. Cybern. 11 (9) (2020) 2021–2038.
[41] J. Dai, W. Wang, J.S. Mi, Uncertainty measurement for interval-valued information systems, Inform. Sci. 251 (2013) 63–78.
[42] Z. Yuan, H. Chen, X. Yang, et al., Fuzzy complementary entropy using hybrid-kernel function and its unsupervised attribute reduction, Knowl.-Based Syst. 231 (2021) 107398.
[43] W. Xu, Z. Yuan, Z. Liu, Feature selection for unbalanced distribution hybrid data based on k-nearest neighborhood rough set, IEEE Trans. Artif. Intell. (2023).
[44] W. Xu, M. Huang, Z. Jiang, et al., Graph-based unsupervised feature selection for interval-valued information system, IEEE Trans. Neural Netw. Learn. Syst. (2023).