Knowledge-Based Systems 158 (2018) 25–42

Contents lists available at ScienceDirect

Knowledge-Based Systems

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

Multi-granularity feature selection on cost-sensitive data with measurement errors and variable costs

Shujiao Liao[*,a,b], Qingxin Zhu[b], Yuhua Qian[c], Guoping Lin[a]

[a] School of Mathematics and Statistics, Minnan Normal University, Zhangzhou 363000, China
[b] School of Information and Software Engineering, University of Electronic Science and Technology of China, Chengdu 610054, China
[c] Institute of Big Data Science and Industry, Shanxi University, Taiyuan 030006, China

ARTICLE INFO

Keywords:
Feature-granularity selection
Measurement errors
Multi-granularity
Neighborhood
Rough sets
Variable costs

ABSTRACT

In real applications of data mining, machine learning and granular computing, measurement errors, test costs and misclassification costs often occur. Furthermore, the test cost of a feature is usually variable with the error range, and the variability of the misclassification cost is related to the object considered. Recently, some approaches based on rough sets have been introduced to study the error-based cost-sensitive feature selection problem. However, most of them consider only single-granularity cases, thus are not feasible for the case where the granularity diversity between different features should be taken into account. Motivated by this problem, we propose a multi-granularity feature selection approach which considers measurement errors and variable costs in terms of feature-value granularities. For a given feature, the feature-value granularity is evaluated by the error confidence level of the feature values. In this way, we build a theoretic framework called confidence-level-vector-based neighborhood rough set, and present a so-called heuristic feature-granularity selection algorithm, and a relevant competition strategy which can select both features and their respective feature-value granularities effectively and efficiently. Experiment results show that a satisfactory trade-off among feature dimension reduction, feature-value granularity selection and total cost minimization can be achieved by the proposed approach. This work would provide a new insight into the cost-sensitive feature selection problem from the multi-granularity perspective.

1. Introduction

Feature selection is one of the most frequently-used techniques in data mining, machine learning and granular computing [4,14,19,42,65]. A dataset often contains many features, thus posing great difficulty in processing. By using the feature selection technique, irrelevant or redundant features can be removed to reduce the data complexity. Consequently, the efficiency of data processing can be improved significantly [10]. Rough set theory [26,34,35,41] is a powerful mechanism to handle uncertain data. Feature selection is also called attribute reduction in rough set society [16,30,39,50].

Cost-sensitive learning has received much attention in data mining and machine learning [1,43,54,60,66,67]. Among various kinds of cost in cost-sensitive learning [48], test cost (also called feature cost) and misclassification cost are the most commonly considered. Usually, the feature values of an object could not be obtained for free. Test cost refers to the money, time, or other resources consumed in acquiring a data item of the object. In addition, an object may be misclassified into a class that it does not belong to. Misclassification cost is the penalty paid for the wrong decision. Cost-sensitive feature selection, also called cost-sensitive attribute reduction in rough set community, aims at finding a feature subset to minimize some types of cost and meanwhile to keep the properties of original decision system as many as possible [15,21,25,31,44,46,61].

In practical applications, it is hard to obtain the accurate value of a data item because the measurement errors are ubiquitous and ineffaceable. For a quantity in reality, its measurement errors usually satisfy a normal (or near-normal) distribution. The existence of measurement errors poses great difficulty in distinguishing two objects if their measured values are close to each other. In view of this problem, some researchers have addressed objects in groups instead of addressing them individually [2,12]. The groups are referred to as information granules. Objects with measured values closing to each other are drawn into the same granule. In this case, the sizes of information granules are related to the error ranges, or equivalently, the lengths of error intervals. Granularity selection, namely selecting the sizes of information granules, plays an important role in granular computing [63].

Recently, three main kinds of approaches have been presented to

* Corresponding author at: School of Mathematics and Statistics, Minnan Normal University, Zhangzhou 363000, China.
E-mail addresses: sjliao2011@163.com (S. Liao), qxzhu@uestc.edu.cn (Q. Zhu), jinchengqyh@126.com (Y. Qian), guoplin@163.com (G. Lin).

https://doi.org/10.1016/j.knosys.2018.05.020
Received 30 June 2017; Received in revised form 13 May 2018; Accepted 16 May 2018
Available online 23 May 2018

study the error-based cost-sensitive feature selection problem by using the rough set theory. The first kind [6,32] considers only test costs but not misclassification costs; the second kind [62,63] considers both test costs and misclassification costs, and the two types of costs are assumed to be fixed values; while in the third kind of approaches [23,64], both types of costs are seen to be variable, and all features are supposed to have the same feature-value granularity, namely have the same data precision. Unfortunately, these approaches are not feasible in some real-world scenarios. Firstly, test costs and misclassification costs often occur simultaneously in real applications, thus it is more realistic to take the two types of costs into consideration. Secondly, acquiring fine-grained data items usually costs more than acquiring coarse-grained ones, so the test cost of a feature is often monotonically decreasing with the enlargement of the feature values' error range. While the variability of the misclassification cost depends on the environment involved and the object considered. Taking the risk evaluation of granting credit as an example, if a customer is misclassified, both the cost (also called benefit if the cost is negative) of the customer and that of the finance company are usually constants. Taking the medical diagnosis as another example, for the misdiagnosis of a specific disease, the misclassification cost of the patient is often fixed, but that of the doctor is usually monotonically increasing with the total test cost paid by the patient. Concretely, if the patient is misdiagnosed with a total test cost of $100, he may require just a little compensation, namely the misclassification cost of the doctor is low; whereas if the patient is misdiagnosed with a total test cost of $1000, the misdiagnosis may make him angry and result in high misclassification cost of the doctor. Finally, in many real applications, different features may have different feature-value granularities, namely have different data precision. For example, electrocardiogram and color ultrasound are two different medical check-up items. Their metrics are not the same; naturally, the precision requirements are not necessarily identical for them. Therefore, the granularity diversity between different features, also called the multi-granularity characteristic of features, should be discussed in the research. However, most of existing rough-set-based feature selection approaches are essentially single-granularity approaches.

In actual applications, for a given dataset, if more necessary features are selected, or feature-value granularities get smaller (in this case, the similarity among the objects in each granule is enhanced), the total test cost will increase, while the misclassification rate will usually decrease. In this case, people cannot intuitively know how the total cost will change. Accordingly, it is complicated but important to choose suitable features and their corresponding feature-value granularities to achieve a trade-off between test costs and misclassification costs so that the total cost is as small as possible. Moreover, except the above-mentioned error-based cost-sensitive feature selection approaches, some existing papers of cost-insensitive feature selection [12,13,51] also addressed the granularity of feature values, but most of them have not taken the diversity between different features into consideration. Multi-granulation rough sets, which deal with multiple binary relations on the universe, have been studied extensively in recent years [18,27,38,53,56,59], but they have not touched the multi-granularity characteristic of features in the feature selection. Based on the above considerations, we introduce multi-granularity ideas into the cost-sensitive feature selection in this study.

In this paper, based on measurement errors and variable costs, we propose a multi-granularity feature selection approach to deal with the relationship among feature dimension, feature-value granularities and total cost. The approach aims at finding a suitable pair of feature subset and feature-value granularity vector to minimize the average total cost (the average value of total cost for the objects in the universe), and at the same time, to preserve the information of original decision system as much as possible. Differing from the previous methods, in the proposed approach the feature-value granularities between different features are not necessarily the same, thus we call the approach multi-granularity feature selection. Owing to the variability consideration of test costs and misclassification costs as well as the diversity of feature-value granularities between different features, the proposed approach is more versatile and practical than the existing error-based cost-sensitive feature selection approaches. Moreover, since most previous feature selection approaches, no matter whether cost-sensitive or cost-insensitive, are single-granularity in essence, this study would provide a new insight into the feature selection problem from the multi-granularity perspective.

In the proposed approach, for a given feature, the feature-value granularity is evaluated by the confidence level of the feature values' measurement errors. The measurement errors are assumed to satisfy a normal distribution, and the confidence level refers to the frequency that an observed interval contains a specific error value. So the confidence level is closely related to the data precision. In this context, we construct a confidence-level-vector-based neighborhood rough set model, in which features and their respective feature-value granularities are associated effectively. Under the new model, some fundamental concepts in neighborhood rough sets are redefined and discussed, such as the neighborhood granule, the lower and upper approximations, and the positive region. These concepts are closely relevant to the given feature subset and its corresponding confidence level vector. Moreover, some important properties in this model are also presented, such as three types of monotonicity in respect of the above-mentioned concepts. Then, some types of variable cost settings are introduced according to reality, in which the relationship among feature-value granularities, test costs and misclassification costs is considered. We also discuss how to compute the average total cost for any given feature-granularity pair (the pair of features and their respective feature-value granularities). Finally, we formally define the multi-granularity feature selection problem which takes measurement errors and variable costs into consideration.

A heuristic feature-granularity selection (the selection of features and their respective feature-value granularities) algorithm and a relevant competition strategy are proposed to deal with the multi-granularity feature selection problem. An addition-deletion strategy is adopted in the heuristic algorithm. Concretely, in the addition phase of the algorithm, for a given feature and its corresponding error confidence level, a feature-granularity significance (the significance of a feature and its feature-value granularity) function is designed by combining the size of incremental positive region with a δ-weighted test-cost-related value. The weight δ is set by the user to adjust the influence of the test cost to the feature-granularity significance. According to the significance values, best features and their corresponding best confidence levels are selected step by step. It is worthwhile to note that the above-mentioned monotonicities of the fundamental concepts in the confidence-level-vector-based neighborhood rough set model are fully used to make the process more efficient. Then in the deletion phase, the redundant feature-granularity elements (a feature-granularity element refers to a feature and its associated confidence level in the selected feature-granularity pair) are deleted to guarantee that the remaining feature-granularity pair has the minimal total cost. As for the competition strategy, it is presented to run the heuristic algorithm with different δ values and choose the best result. By using it, the users need not know the best setting for the weight δ in advance. Finally, some evaluation metrics are developed to study the performance of the proposed approach.

To evaluate the performance of the multi-granularity feature selection approach, a series of detailed experiments are undertaken on nine datasets from the UCI (University of California – Irvine) library [3]. Experimental results demonstrate that a satisfactory trade-off among feature dimension reduction, feature-value granularity selection and total cost minimization can be achieved by the approach. Both features and their respective feature-value granularities, which are often not the same between different features, can be obtained simultaneously through using the approach. This cannot be achieved by using the previous methods. The proposed multi-granularity approach

performs well not only on minimizing the total cost but also on the computational efficiency. Compared with the three kinds of error-based cost-sensitive feature selection approaches discussed above, the proposed approach is more effective and versatile. In addition, rational value ranges of extrinsic parameters are also given through in-depth experimental analyses.

The rest of the paper is organized as follows. In Section 2, we construct the confidence-level-vector-based neighborhood rough set model, and present some important notions and properties of this model. Section 3 designs the variable cost settings and gives the calculation method of average total cost. Then the multi-granularity feature selection problem is formally defined. In Section 4, we propose the heuristic feature-granularity selection algorithm and the competition strategy. Some relevant evaluation metrics are also given. Section 5 discusses the experiment settings and results thoroughly. Finally, we conclude the paper and outline further research ideas in Section 6.

2. Confidence-level-vector-based neighborhood rough set

As a technique of granular computing [2,8,29,37], classical rough set [35,36] and its extensions [5,9,11,13,24,40,45,47,52,68] handle the uncertainty and the granulation in information systems and decision systems. In view of the universality of measurement errors, we construct a confidence-level-vector-based neighborhood rough set (CVRS) model in this section. Fundamental notions and properties in this model are introduced, such as the concepts of neighborhood, lower and upper approximations, positive region, and their monotonicities with respect to (w.r.t.) the given pair of feature subset and error confidence level vector.

2.1. Preliminaries

In this subsection, we review some basic concepts related to granular computing and statistics.

In granular computing domains, an information granule is often represented by a neighborhood [13,17,28,49]. Let U be a nonempty finite object set called the universe. For each object $x \in U$, a neighborhood granule of x is a set $n(x)$ composed of some objects from U with a certain criterion. The elements in $n(x)$ are indistinguishable from x under the given criterion. When measurement errors are involved in the universe, the neighborhoods are formed according to the errors. An error-bound-based neighborhood has been defined as follows:

Definition 2.1. [64] Let U be a universe with measurement errors, and $e \in \mathbb{R}^+$ be an error bound. For each $x \in U$, the neighborhood of x w.r.t. the error bound is defined as

$$n_e(x) = \{x' \in U | |x' - x| \leq 2e\}. \tag{1}$$

Naturally, the error interval is $[-e, e]$, which the measurement error values should lie within. The reason why $2e$ instead of e is employed in Eq. (1) has been explained in [64]. It means that all objects with measured value differing from x by not exceeding $2e$ should be drawn into the neighborhood $n_e(x)$ together.

In Definition 2.1, both the error bound and the error interval are fixed. However, in real-world applications, the error range for the same data item often changes due to different observers or different instruments. Similar to those in [23,62,64], we suppose that the measurement errors follow a normal distribution. Confidence interval and confidence level are two commonly-used concepts in statistics [7,33]. The confidence interval is a type of interval estimation for a population parameter, while the confidence level determines how frequently an observed interval contains a given parameter value. Moreover, the left endpoint of confidence interval is called the lower confidence limit, and the right endpoint is called the upper confidence limit. The higher the confidence level is, the wider the confidence interval becomes. For a normal distribution, the confidence interval and confidence level satisfy the so-called "3-sigma" rule, which is shown in Fig. 1, where μ and σ are the mean and the standard deviation, respectively. From the figure, we can find that nearly all values (99.7%) lie within 3 standard deviations away from the mean.

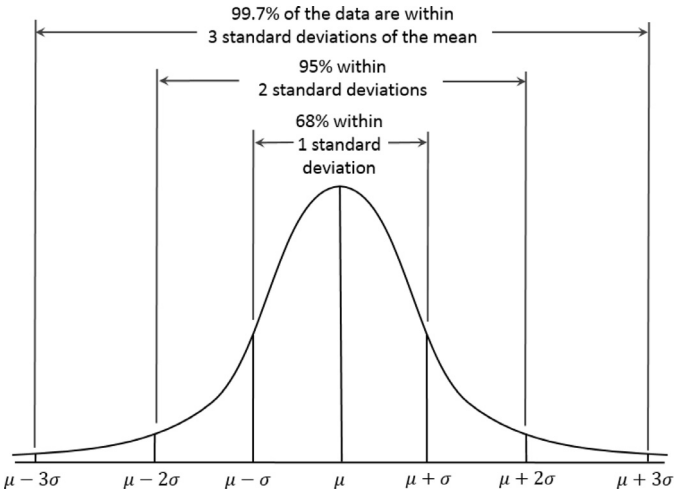


Fig. 1. The "3-sigma" rule.

For a normal distribution, if the confidence interval is known, the confidence level can be computed by using the cumulative distribution function [7]. Concretely, assuming that x is a normal random variable with mean μ and variance σ^2, the confidence level (i.e. the probability) of $x \in [a, b]$ is

$$p\{a \leq x \leq b\} = F(b) - F(a) = \Phi\left(\frac{b-\mu}{\sigma}\right) - \Phi\left(\frac{a-\mu}{\sigma}\right), \tag{2}$$

where F and Φ denote the cumulative distribution functions of normal distribution $N(\mu, \sigma^2)$ and standard normal distribution $N(0, 1)$, respectively. Besides, if the confidence level is known, the confidence interval can be calculated through the quantile function, which is the inverse of the cumulative distribution function [7]. Assuming that the confidence level is p, the quantile function of the standard normal distribution can be expressed as

$$\Phi^{-1}(p) = \sqrt{2}\, erf^{-1}(2p - 1),\ p \in (0, 1), \tag{3}$$

where $erf^{-1}(2p-1)$ is the inverse error function. For convenience, $\Phi^{-1}(p)$ is usually denoted by z_p. Assuming that x is a normal random variable with mean μ and variance σ^2, its quantile function value is

$$F^{-1}(p) = \mu + \sigma\Phi^{-1}(p) = \mu + \sigma z_p,\ p \in (0, 1). \tag{4}$$

x will exceed $\mu + \sigma z_p$ with probability $1 - p$, and will lie outside $[\mu - \sigma z_p, \mu + \sigma z_p]$ with probability $2(1 - p)$.

In data mining and machine learning, decision system is a fundamental notion defined as follows:

Definition 2.2. [55] A *decision system* (DS) S is the 5-tuple:

$$S = (U, C, d, V = \{V_a | a \in C \cup \{d\}\}, I = \{I_a | a \in C \cup \{d\}\}),$$

where U is a finite nonempty set of objects called the universe, C is the set of conditional attributes (also called as features), d is the decision attribute, V_a is the set of values for each $a \in C \cup \{d\}$, and I_a: $U \to V_a$ is an information function for each $a \in C \cup \{d\}$.

Note that, in real applications, many decision systems only have one decision attribute, so our work focuses on this kind of decision systems. If the involved DS has more than one decision attribute, we can construct multiple new decision systems, each with only one decision attribute.

2.2. Fundamental concepts in confidence-level-vector-based neighborhood rough set

Now we present the fundamental concepts in the CVRS model, which is started from the decision system. Since each object in a decision system has a series of features, and the data precisions between different features are not necessarily the same, we construct an error confidence level vector corresponding to all features. Based on the confidence level vector, a new kind of decision system is defined as follows:

Definition 2.3. A *confidence-level-vector-based decision system* (CVDS) S is the 6-tuple:

$$S = (U, C, d, V, I, P), \tag{5}$$

where U, C, d, V, I have the same meanings as in Definition 2.2, and $P = \left(p_{a_1}, p_{a_2}, \ldots, p_{a_{|C|}}\right)$ is a confidence level vector, in which $p_{a_i} \in (0, 1)$ is the error confidence level for the feature values of feature a_i.

Note that, for brevity, we often omit the term "feature values" when mentioning confidence level in the following context. For example, p_a is called as the confidence level for feature a, or the confidence level corresponding to feature a. An exemplary CVDS consists of the decision system shown in Table 1 and the confidence level vector shown in Table 2. From the two tables, we know that $U = \{x_1, x_2, \cdots, x_6\}$, $C = \{a_1, a_2, a_3\}$, $P = (0.5, 0.8, 0.6)$.

Since the confidence interval refers to the measurement errors in this paper, the upper confidence limit is regarded as the upper error bound. As told in [64], the confidence interval for a normal distribution is $(-\infty, +\infty)$ if the confidence level is 1; combined with the "3-sigma" rule, the maximal confidence level for each attribute is supposed to be 99.7%. Let $e(a, p_a)$ denote the upper error bound w.r.t. feature a and its corresponding confidence level p_a, then according to Eq. (4), we have

$$e(a, p_a) = \sigma_a z_{p_a},\ p_a \in (0, 0.997]. \tag{6}$$

So the maximum $e(a, p_a)$ is $e(a, 0.997) = 3\sigma_a$. We let the standard deviation be

$$\sigma_a = k \cdot max|a(x_i) - \overline{a(x)}|,\ 1 \le i \le |U|, \tag{7}$$

where $k > 0$ is a constant, $a(x_i)$ is the feature value of object x_i w.r.t. feature a, and $\overline{a(x)} = \frac{1}{|U|}\sum_{i=1}^{|U|} a(x_i)$ is the average feature value of a for all objects (note that the standard deviation σ_a could also be given in other forms). Then, for each feature a and its corresponding confidence level p_a, the upper error bound can be computed according to Eqs. (6) and (7).

Neighborhoods play an important role in the CVRS model. Analogously to Definition 2.1, the neighborhood based on a single feature and its corresponding confidence level is defined as follows:

Definition 2.4. Let be a CVDS, $x \in U$ and $a \in C$. The neighborhood of x with reference to feature a and confidence level p_a is defined as

$$n_{(a,p_a)}(x) = \{x' \in U | |a(x') - a(x)| \le 2e(a, p_a)\},\ p_a \in (0, 0.997]. \tag{8}$$

Before introducing the neighborhood corresponding to multiple features, we give a remark about subvector as follows:

Table 1
An example of numeric decision system.

	a_1	a_2	a_3	d
x_1	0.71	0.31	0.21	1
x_2	0.61	0.34	0.11	1
x_3	0.58	0.27	0.25	1
x_4	0.82	0.29	0.23	2
x_5	0.68	0.44	0.55	2
x_6	0.55	0.38	0.05	2

Table 2
An example of confidence level vector.

a	a_1	a_2	a_3
p_a	0.5	0.8	0.6

Remark 2.5. Traditional subvector is defined as a contiguous part of a larger vector. To facilitate our discussions, we relax the restriction in this paper, i.e., we suppose that the subvector can also be constituted by two or more discontiguous parts of the entire vector. For example, given two vectors $V = (1, 2, 3, 4, 5, 6)$ and $V' = (1, 3, 6)$, we say that V' is the subvector of V, which is denoted as $V' \sqsubseteq V$ or $V \sqsupseteq V'$.

Let be a CVDS, $B \subseteq C$. In the following context, we let P_B denote the confidence level subvector corresponding to B, i.e., $P_B \sqsubseteq P$ and each component of P_B corresponds to a feature in B. Naturally, the neighborhood of $x \in U$ induced by B and P_B is the intersection of the neighborhoods induced by each feature $a \in B$ and its corresponding confidence level p_a, i.e., the confidence-level-vector-based neighborhood is

$$n_{(B,P_B)}(x) = \bigcap_{a \in B} n_{(a,p_a)}(x). \tag{9}$$

Two exemplary 2-dimensional neighborhoods, whose feature-granularity pairs are (B, P_B) and (B, P_B') respectively, are shown in Fig. 2, where $B = \{a_1, a_2\}$, $P_B = (p_1, p_2)$ and $P_B' = (p_1', p_2')$. From the figure, it is known that the neighborhoods change with the feature-granularity pairs. The related properties will be discussed further in Section 2.3. Moreover, it is easy to obtain the following proposition for neighborhoods:

Proposition 2.6. *Let be a CVDS, $B \subseteq C$, and P_B be the corresponding confidence level subvector. Then for any object in U, its neighborhood based on B and P_B satisfies*

(1) *reflexivity:* $\forall x \in U, x \in n_{(B,P_B)}(x)$;
(2) *symmetry:* $\forall x_i, x_j \in U$, *if* $x_j \in n_{(B,P_B)}(x_i)$, $x_i \in n_{(B,P_B)}(x_j)$.

According to the reflexivity of neighborhoods, we have $U \subseteq \{n_{(B,P_B)}(x) | x \in U\}$, so $\{n_{(B,P_B)}(x) | x \in U\}$ is a covering of U. We denote it as $C(B, P_B)$ for brevity, i.e.,

$$C(B, P_B) = \{n_{(B,P_B)}(x) | x \in U\}. \tag{10}$$

Note that, if B only contains one feature a, we write $C(B, P_B)$ as $C(a, p_a)$ instead of $C(\{a\}, (p_a))$ for brevity, which is similar to that for neighborhood. This kind of notation is also used for other concepts

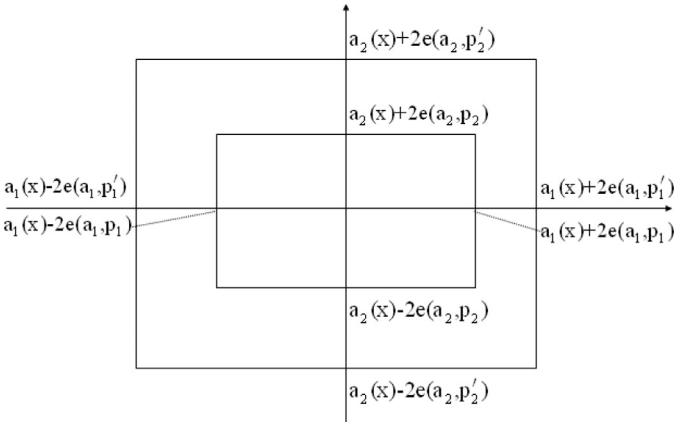


Fig. 2. Two exemplary 2-dimensional neighborhoods with feature-granularity pairs (B, P_B) and (B, P_B') respectively, where $B = \{a_1, a_2\}$, $P_B = (p_1, p_2)$ and $P_B' = (p_1', p_2')$.

Table 3
The upper error bounds.

a	a_1	a_2	a_3
p_a	0.5	0.8	0.6
z_{p_a}	0.6745	1.2816	0.8416
$e(a, p_a)$	0.0218	0.0261	0.0533

introduced in the following context.

Based on B and P_B, a neighborhood relation $R_{(B,P_B)}$ on the universe U can be induced. It can be written as a relation matrix $M(R_{(B,P_B)}) = (r_{ij})_{|U|\times|U|}$, where $r_{ij} = 1$ if $x_j \in n_{(B,P_B)}(x_i)$, or equivalently, $x_i \in n_{(B,P_B)}(x_j)$; otherwise $r_{ij} = 0$. According to Proposition 2.6, it is easy to know that $R_{(B,P_B)}$ satisfies reflexivity: $r_{ii} = 1$, and symmetry: $r_{ij} = r_{ji}$.

Lower and upper approximations, positive region and boundary region are fundamental issues in rough set theory. We redefine them in the CVRS model as follows.

Definition 2.7. Let be a CVDS, and let $U/\{d\}$ denote the partitions of the universe U induced by the decision attribute d. Suppose that $B \subseteq C$ and P_B is the corresponding confidence level subvector. We call $< U, R_{(B,P_B)}>$ a neighborhood approximation space. For any $X \in U/\{d\}$, the lower and upper approximations of X in $< U, R_{(B,P_B)}>$ are defined as

$$\underline{N_{(B,P_B)}}(X) = \{x \in U | n_{(B,P_B)}(x) \subseteq X\}, \tag{11}$$

$$\overline{N_{(B,P_B)}}(X) = \{x \in U | n_{(B,P_B)}(x) \cap X \neq \varnothing\}. \tag{12}$$

Obviously, $\underline{N_{(B,P_B)}}(X) \subseteq X \subseteq \overline{N_{(B,P_B)}}(X)$. The boundary region of X in $< U, R_{(B,P_B)}>$ is defined as

$$BN_{(B,P_B)}(X) = \overline{N_{(B,P_B)}}(X) - \underline{N_{(B,P_B)}}(X). \tag{13}$$

Definition 2.8. Let be a CVDS, $B \subseteq C$, and P_B be the corresponding confidence level subvector. Suppose that $U/\{d\} = \{X_1, X_2, ..., X_K\}$, where X_i is the object subset with decision class i. Then the lower and upper approximations of decision $\{d\}$ in the neighborhood approximation space $< U, R_{(B,P_B)}>$ are defined as

$$\underline{N_{(B,P_B)}}(\{d\}) = \bigcup_{i=1}^{K} \underline{N_{(B,P_B)}}(X_i),\ \overline{N_{(B,P_B)}}(\{d\}) = \bigcup_{i=1}^{K} \overline{N_{(B,P_B)}}(X_i). \tag{14}$$

The decision boundary region of $\{d\}$ in $< U, R_{(B,P_B)}>$ is defined as

$$BN_{(B,P_B)}(\{d\}) = \overline{N_{(B,P_B)}}(\{d\}) - \underline{N_{(B,P_B)}}(\{d\}). \tag{15}$$

The lower approximation $\underline{N_{(B,P_B)}}(\{d\})$ is also called as the positive region, which is denoted by $\overline{POS_{(B,P_B)}}(\{d\})$. For each object in the positive region, its neighborhood granule consistently belongs to one of the decision classes. In contrast, for each object in the boundary region, the samples in its neighborhood granule come from two or more classes. So the objects in the positive region can be certainly classified into one class, while those in the boundary region cannot. One objective of multi-granularity feature selection is to make the positive region as large as possible. It is easy to obtain the following proposition:

Proposition 2.9. *Let be a CVDS, $B \subseteq C$, and P_B be the corresponding confidence level subvector. Then we have*

(1) $\underline{POS_{(\varnothing,P_\varnothing)}}(\{d\}) = \varnothing$;
(2) $\overline{N_{(B,P_B)}}(\{d\}) = U$;
(3) $\underline{POS_{(B,P_B)}}(\{d\}) \bigcap BN_{(B,P_B)}(\{d\}) = \varnothing$;
(4) $\underline{POS_{(B,P_B)}}(\{d\}) \bigcup BN_{(B,P_B)}(\{d\}) = U$.

We give an example to illustrate the concepts discussed above.

Example 2.10. A CVDS is composed of Tables 1 and 2, where $U = \{x_1, x_2, ..., x_6\}$, $C = \{a_1, a_2, a_3\}$, $P = (0.5, 0.8, 0.6)$ and $U/\{d\} = \{\{x_1, x_2, x_3\}, \{x_4, x_5, x_6\}\}$. Let $X_1 = \{x_1, x_2, x_3\}$, $X_2 = \{x_4, x_5, x_6\}$. And let $k = 0.2$ in Eq. (7), then the upper error bounds for each (a, p_a) pair can be calculated according to Eqs. (6)–(7). The results are displayed in Table 3.

Let
$B_1 = \{a_1, a_2\}$, $P_1 = (0.5, 0.8)$, $B_2 = \{a_1, a_3\}$, $P_2 = (0.5, 0.6)$, $B_3 = \{a_2, a_3\}$ and $P_3 = (0.8, 0.6)$. Based on the upper error bounds, we can compute the neighborhoods for each feature-granularity pair according to Eqs. (8)–(9). The results are listed in Table 4.

According to Table 4, a series of coverings of U are obtained, which are

$C(a_1, 0.5) = \{\{x_1, x_5\}, \{x_2, x_3\}, \{x_2, x_3, x_6\}, \{x_4\}, \{x_3, x_6\}\}$,
$C(a_2, 0.8) = \{\{x_1, x_2, x_3, x_4\}, \{x_1, x_2, x_4, x_6\}, \{x_1, x_3, x_4\}, \{x_5\}, \{x_2, x_6\}\}$,
$C(a_3, 0.6) = C(B_3, P_3)$
$= \{\{x_1, x_2, x_3, x_4\}, \{x_1, x_2, x_6\}, \{x_1, x_3, x_4\}, \{x_5\}, \{x_2, x_6\}\}$,
$C(B_1, P_1) = C(B_2, P_2) = C(C, P) = \{\{x_1\}, \{x_2\}, \{x_3\}, \{x_4\}, \{x_5\}, \{x_6\}\}$. So the coverings induced by (B_1, P_1), (B_2, P_2) and (C, P) are all the partitions of U essentially. The lower and upper approximations for each feature-granularity pair can also be computed according to Table 4. The results are shown in Table 5.

Based on the obtained lower and upper approximations, the positive regions and boundary regions are calculated for different feature-granularity pairs. $POS_{(a_1,0.5)}(\{d\}) = \{x_2, x_4\}$, $BN_{(a_1,0.5)}(\{d\}) = \{x_1, x_3, x_5, x_6\}$; $POS_{(a_2,0.8)}(\{d\}) = POS_{(a_3,0.6)}(\{d\}) = POS_{(B_3,P_3)}(\{d\}) = \{x_5\}$, $BN_{(a_2,0.8)}(\{d\}) = BN_{(a_3,0.6)}(\{d\}) = BN_{(B_3,P_3)}(\{d\}) = \{x_1, x_2, x_3, x_4, x_6\}$; $POS_{(B_1,P_1)}(\{d\}) = POS_{(B_2,P_2)}(\{d\}) = POS_{(C,P)}(\{d\}) = U$, $BN_{(B_1,P_1)}(\{d\}) = BN_{(B_2,P_2)}(\{d\}) = BN_{(C,P)}(\{d\}) = \varnothing$. Hence, the feature subsets $B_1 = \{a_1, a_2\}$ and $B_2 = \{a_1, a_3\}$ have the same approximate power as the entire feature set C at the given confidence levels. They can characterize all samples in the universe.

2.3. Three types of monotonicity with respect to above-mentioned concepts

In this subsection, we discuss three types of monotonicity for the fundamental concepts mentioned in Section 2.2. The first type of monotonicity relates to the addition of features.

Table 4
The neighborhoods.

U	$n_{(a_1,0.5)}(x)$	$n_{(a_2,0.8)}(x)$	$n_{(a_3,0.6)}(x)$	$n_{(B_1,P_1)}(x)$	$n_{(B_2,P_2)}(x)$	$n_{(B_3,P_3)}(x)$	$n_{(C,\ P)}(x)$
x_1	$\{x_1, x_5\}$	$\{x_1, x_2, x_3, x_4\}$	$\{x_1, x_2, x_3, x_4\}$	$\{x_1\}$	$\{x_1\}$	$\{x_1, x_2, x_3, x_4\}$	$\{x_1\}$
x_2	$\{x_2, x_3\}$	$\{x_1, x_2, x_4, x_6\}$	$\{x_1, x_2, x_6\}$	$\{x_2\}$	$\{x_2\}$	$\{x_1, x_2, x_6\}$	$\{x_2\}$
x_3	$\{x_2, x_3, x_6\}$	$\{x_1, x_3, x_4\}$	$\{x_1, x_3, x_4\}$	$\{x_3\}$	$\{x_3\}$	$\{x_1, x_3, x_4\}$	$\{x_3\}$
x_4	$\{x_4\}$	$\{x_1, x_2, x_3, x_4\}$	$\{x_1, x_3, x_4\}$	$\{x_4\}$	$\{x_4\}$	$\{x_1, x_3, x_4\}$	$\{x_4\}$
x_5	$\{x_1, x_5\}$	$\{x_5\}$	$\{x_5\}$	$\{x_5\}$	$\{x_5\}$	$\{x_5\}$	$\{x_5\}$
x_6	$\{x_3, x_6\}$	$\{x_2, x_6\}$	$\{x_2, x_6\}$	$\{x_6\}$	$\{x_6\}$	$\{x_2, x_6\}$	$\{x_6\}$

Table 5
The lower and upper approximations.

		$(a_1, 0.5)$	$(a_2, 0.8)$	$(a_3, 0.6)$	(B_1, P_1)	(B_2, P_2)	(B_3, P_3)	(C, P)
$\underline{N_{(B,P_B)}}(X)$	X_1	$\{x_2\}$	$\emptyset$	$\emptyset$	X_1	X_1	$\emptyset$	X_1
	X_2	$\{x_4\}$	$\{x_5\}$	$\{x_5\}$	X_2	X_2	$\{x_5\}$	X_2
$\overline{N_{(B,P_B)}}(X)$	X_1	$\{x_1, x_2, x_3, x_5, x_6\}$	$\{x_1, x_2, x_3, x_4, x_6\}$	$\{x_1, x_2, x_3, x_4, x_6\}$	X_1	X_1	$\{x_1, x_2, x_3, x_4, x_6\}$	X_1
	X_2	$\{x_1, x_3, x_4, x_5, x_6\}$	U	U	X_2	X_2	U	X_2

Theorem 2.11. *(Type-1 monotonicity). Let be a CVDS,* $B_1 \subseteq B_2 \subseteq C$, $P_{B_1} \sqsubseteq P_{B_2}$. *We have*

(1) $\forall x \in U$, $n_{(B_1,P_{B_1})}(x) \supseteq n_{(B_2,P_{B_2})}(x)$;
(2) $R_{(B_1,P_{B_1})} \supseteq R_{(B_2,P_{B_2})}$;
(3) $\forall X \in U/\{d\}$, $\underline{N_{(B_1,P_{B_1})}}(X) \subseteq \underline{N_{(B_2,P_{B_2})}}(X)$, $\overline{N_{(B_1,P_{B_1})}}(X) \supseteq \overline{N_{(B_2,P_{B_2})}}(X)$;
(4) $POS_{(B_1,P_{B_1})}(\{d\}) \subseteq POS_{(B_2,P_{B_2})}(\{d\})$, $BN_{(B_1,P_{B_1})}(\{d\}) \supseteq BN_{(B_2,P_{B_2})}(\{d\})$.

Proof. (1) According to Eqs. (8) and (9), $\forall x' \in n_{(B_2,P_{B_2})}(x)$, we have $|a(x') - a(x)| \le 2e(a, p_a)$, $\forall a \in B_2$. Since $B_1 \subseteq B_2$, we have $|a(x') - a(x)| \le 2e(a, p_a)$, $\forall a \in B_1$. So $x' \in n_{(B_1,P_{B_1})}(x)$, then we have $n_{(B_1,P_{B_1})}(x) \supseteq n_{(B_2,P_{B_2})}(x)$.

(2) Let r_{ij}, s_{ij} denote the elements in relation matrices $M(R_{(B_1,P_{B_1})})$ and $M(R_{(B_2,P_{B_2})})$, respectively. If $s_{ij} = 1$, $x_j \in n_{(B_2,P_{B_2})}(x_i)$, we have $x_j \in n_{(B_1,P_{B_1})}(x_i)$ according to (1), so $r_{ij} = 1$. Then we have $R_{(B_1,P_{B_1})} \supseteq R_{(B_2,P_{B_2})}$.

(3) $\forall x \in \underline{N_{(B_1,P_{B_1})}}(X)$, $n_{(B_1,P_{B_1})}(x) \subseteq X$. According to (1), $n_{(B_2,P_{B_2})}(x) \subseteq n_{(B_1,P_{B_1})}(x) \subseteq X$, so $x \in \underline{N_{(B_2,P_{B_2})}}(X)$, we have $\underline{N_{(B_1,P_{B_1})}}(X) \subseteq \underline{N_{(B_2,P_{B_2})}}(X)$. Similarly, we have $\overline{N_{(B_1,P_{B_1})}}(X) \supseteq \overline{N_{(B_2,P_{B_2})}}(X)$.

(4) Assume $U/\{d\} = \{X_1, X_2, ..., X_K\}$. According to (3), $\underline{N_{(B_1,P_{B_1})}}(X_i) \subseteq \underline{N_{(B_2,P_{B_2})}}(X_i)$, $i = 1, 2, ..., K$. Because $POS_{\left(B_j,P_{B_j}\right)}(\{d\}) = \bigcup_{i=1}^{K} \underline{N_{\left(B_j,P_{B_j}\right)}}(X_i)$, $j = 1, 2$, we have $POS_{(B_1,P_{B_1})}(\{d\}) \subseteq POS_{(B_2,P_{B_2})}(\{d\})$. Similarly, $\overline{N_{(B_1,P_{B_1})}}(\{d\}) \supseteq \overline{N_{(B_2,P_{B_2})}}(\{d\})$ according to (3). Since $BN_{\left(B_j,P_{B_j}\right)}(\{d\}) = \overline{N_{\left(B_j,P_{B_j}\right)}}(\{d\}) - POS_{\left(B_j,P_{B_j}\right)}(\{d\})$, $j = 1, 2$, we have $BN_{(B_1,P_{B_1})}(\{d\}) \supseteq BN_{(B_2,P_{B_2})}(\{d\})$. □

Theorem 2.11 shows that the positive region increases monotonically with the adding of features, while the boundary region decreases monotonically. Since not only the features but also their respective feature-value granularities should be considered in the multi-granularity feature selection problem, we discuss the second type of monotonicity, which refers to the change of confidence levels. Before that, we define an order relation $\preceq$ for the vectors with the same dimension.

Definition 2.12. Given two vectors $X = (x_1, x_2, ..., x_n)$ and $Y = (y_1, y_2, ..., y_n)$, $X \preceq Y$ if $x_i \le y_i$, $i = 1, 2, ..., n$.

Based on the order relation, the second type of monotonicity is given as follows:

Theorem 2.13. *(Type-2 monotonicity). Let be a CVDS,* $B \subseteq C$, *and* P_1, P_2 *be two confidence level subvectors corresponding to B, which satisfies* $P_1 \preceq P_2$. *We have*

(1) $\forall x \in U$, $n_{(B,P_1)}(x) \subseteq n_{(B,P_2)}(x)$;
(2) $R_{(B,P_1)} \subseteq R_{(B,P_2)}$;
(3) $\forall X \in U/\{d\}$, $\underline{N_{(B,P_1)}}(X) \supseteq \underline{N_{(B,P_2)}}(X)$, $\overline{N_{(B,P_1)}}(X) \subseteq \overline{N_{(B,P_2)}}(X)$;
(4) $POS_{(B,P_1)}(\{d\}) \supseteq POS_{(B,P_2)}(\{d\})$, $BN_{(B,P_1)}(\{d\}) \subseteq BN_{(B,P_2)}(\{d\})$.

Proof. (1) Let p_a^1, p_a^2 respectively denote the components of P_1, P_2 corresponding to feature $a \in B$. $\forall x' \in n_{(B,P_1)}(x)$, we have $|a(x') - a(x)| \le 2e(a, p_a^1)$, $\forall a \in B$. Since $P_1 \preceq P_2$, we have $p_a^1 \le p_a^2$, $2e(a, p_a^1) \le 2e(a, p_a^2)$, then $|a(x') - a(x)| \le 2e(a, p_a^2)$, $\forall a \in B$. So $x' \in n_{(B,P_2)}(x)$, $n_{(B,P_1)}(x) \subseteq n_{(B,P_2)}(x)$. □

The proof of (2)-(4) is similar to that for Theorem 2.11, so we omit it for brevity. From Theorem 2.13, we know that the positive region decreases monotonically with the increase of confidence levels, while the boundary region increases monotonically. Theorem 2.11 and Theorem 2.13 can significantly improve the efficiency of the heuristic algorithm designed in Section 4.1.

Note that, in real applications, for each feature $a \in C$, there is usually a highest data precision which could be achieved by using the best measurement instruments. At this time, the error interval and error confidence level are both minimal. Assuming that the minimal error is $e_a > 0$ for feature a, then according to Eq. (2), the minimal confidence level is

$$p_a^{\square} = \Phi\left(\frac{e_a}{\sigma_a}\right) - \Phi\left(\frac{-e_a}{\sigma_a}\right) = 2\Phi\left(\frac{e_a}{\sigma_a}\right) - 1, \tag{16}$$

where Φ denotes the cumulative distribution function of the standard normal distribution, and σ_a is computed according to Eq. (7). Obviously, $p_a^{\square}$ is intrinsic to feature a. Let $P_{\square} = \left(p_{a_1}^{\square}, p_{a_2}^{\square}, ..., p_{a_{|C|}}^{\square}\right)$ denote the minimal-confidence-level vector corresponding to the entire feature set C. Based on the above two theorems, we have the third type of monotonicity as follows:

Theorem 2.14. *(Type-3 monotonicity). Let be a CVDS,* $B \subseteq C$, P_B *be the confidence level subvector corresponding to B, and* $P_{\square}$ *be the minimal-confidence-level vector corresponding to C. We have*

(1) $\forall x \in U$, $n_{(B,P_B)}(x) \supseteq n_{(C,P_{\square})}(x)$;
(2) $R_{(B,P_B)} \supseteq R_{(C,P_{\square})}$;
(3) $\forall X \in U/\{d\}$, $\underline{N_{(B,P_B)}}(X) \subseteq \underline{N_{(C,P_{\square})}}(X)$, $\overline{N_{(B,P_B)}}(X) \supseteq \overline{N_{(C,P_{\square})}}(X)$;
(4) $POS_{(B,P_B)}(\{d\}) \subseteq POS_{(C,P_{\square})}(\{d\})$, $BN_{(B,P_B)}(\{d\}) \supseteq BN_{(C,P_{\square})}(\{d\})$.

Proof. The proof for $POS_{(B,P_B)}(\{d\}) \subseteq POS_{(C,P_{\square})}(\{d\})$:

Obviously, $P_{\square} \preceq P$, so $POS_{(C,P)}(\{d\}) \subseteq POS_{(C,P_{\square})}(\{d\})$ according to Theorem 2.13. Moreover, since $B \subseteq C$, $P_B \sqsubseteq P$, we have $POS_{(B,P_B)}(\{d\}) \subseteq POS_{(C,P)}(\{d\})$ according to Theorem 2.11. Therefore, $POS_{(B,P_B)}(\{d\}) \subseteq POS_{(C,P_{\square})}(\{d\})$. □

For brevity, we only give the proof for $POS_{(B,P_B)}(\{d\}) \subseteq POS_{(C,P_{\square})}(\{d\})$. The proofs of other formulas are similar.

3. Variable-cost-based multi-granularity feature selection problem

In this section, we start from designing several kinds of variable cost settings. In these cost settings, the relationship among feature-value granularities, test costs and misclassification costs is considered, in which the feature-value granularity of a feature is evaluated by the confidence level of the feature values' measurement errors. Then, we present the calculation method of average total cost for any given pair of features and confidence levels. Finally, the variable-cost-based multi-granularity feature selection problem is formally defined.

3.1. Variable cost settings

As we know, for a feature, the test cost often increases monotonically with the increase of data precision. Meanwhile, the data precision decreases monotonically with the increase of feature-value granularity, or equivalently the error confidence level of feature values. Therefore, the test cost is monotonically decreasing as the confidence level gets large. As for the misclassification cost, its variability depends on the environment involved and the object considered. As discussed earlier, in some cases the misclassification cost is a fixed value, while in other cases it is monotonically increasing as the total test cost becomes large. For these reasons, we design the test cost functions and the misclassification cost functions in the following forms.

For a CVDS $S = (U, C, d, V, I, P)$, let tc denote the test cost function, and $tc(a)$ denote the highest test cost of feature a, namely, the test cost paid for obtaining the highest data precision for feature a. Then the highest total test cost for each object is $tc(C) = \sum_{a\in C} tc(a)$. Given feature a and its corresponding confidence level p_a, the test cost function can be presented in different forms according to the application backgrounds. For example, a linear-function-form test cost is

$$tc(a, p_a) = tc(a)\cdot(1 - \lambda_a p_a),\ p_a \in (0, 0.997], \tag{17}$$

where $\lambda_a \in [0, 1]$ is the adjusting factor of the test cost w.r.t. the confidence level; and a piecewise-constant-function-form test cost is

$$tc(a, p_a) = TC_i(a),\ p_a \in [p_{i-1}, p_i)(i = 1, 2, ..., m), \tag{18}$$

where m is the number of segments, $p_0 > 0$, $p_m < 1$, and $TC_i(a)$ are constant values satisfying $TC_1(a) > TC_2(a) > \cdots > TC_m(a) > 0$. Then, given a feature-granularity pair (B, P_B), the corresponding total test cost is

$$tc(B, P_B) = \sum_{a\in B} tc(a, p_a). \tag{19}$$

Finally, let (k, l) denote the misclassification from class k to class l, which is called a misclassified class pair, and let $mc(B, P_B)_{(k,\ l)}$ denote the misclassification cost of (k, l) based on (B, P_B) pair. Obviously, if $k = l$, $mc(B, P_B)_{(k,l)} = 0$. While if $k \neq l$, $mc(B, P_B)_{(k,\ l)}$ can be given in multiple forms according to reality. For example, a constant-form misclassification cost is

$$mc(B, P_B)_{(k,l)} = MC_{(k,l)}, \tag{20}$$

where $MC_{(k,\ l)} > 0$ is a constant; a linear-function-form misclassification cost is

$$mc(B, P_B)_{(k,l)} = \gamma_{(k,l)}\cdot tc(B, P_B), \tag{21}$$

where $\gamma_{(k,\ l)} > 0$ is a penalty factor; and a piecewise-constant-function-form misclassification cost is

$$mc(B, P_B)_{(k,l)} = MC_j^{(k,l)},\ tc(B, P_B) \in [TTC_{j-1}, TTC_j)(j = 1, 2, ..., n), \tag{22}$$

where n is the number of segments, all TTC_j and $MC_j^{(k,l)}$ are constant values, $TTC_0 \geq 0$, and $0 < MC_1^{(k,l)} < MC_2^{(k,l)} < \cdots < MC_n^{(k,l)}$.

We give an example of variable cost settings as follows:

Example 3.1. A CVDS is constituted by Tables 1 and 2, where $C = \{a_1, a_2, a_3\}$ and $P = (0.5, 0.8, 0.6)$. Given $tc(a_1) = 23$, $tc(a_2) = 97$, $tc(a_3) = 14$, $\lambda_{a_1} = 0.2$, $\lambda_{a_2} = 0.4$ and $\lambda_{a_3} = 0.1$, we have $tc(C) = \sum_{i=1}^{3} tc(a_i) = 134$. Let $B = \{a_2, a_3\}$, then $P_B = (0.8, 0.6)$. Since $tc(a_2, 0.8) = 97 \times (1 - 0.4 \times 0.8) = 65.96$ and $tc(a_3, 0.6) = 14 \times (1 - 0.1 \times 0.6) = 13.16$, we have $tc(B, P_B) = 65.96 + 13.16 = 79.12$. Then let $\gamma_{(1,2)} = 50$ and $\gamma_{(2,1)} = 10$, we have $mc(B, P_B)_{(1,2)} = 50 \times 79.12 = 3956$, $mc(B, P_B)_{(2,1)} = 10 \times 79.12 = 791.2$.

It is notable that, although sometimes the misclassification costs are fixed, the test costs are always variable, so the designed cost settings are said to be variable. Moreover, for simplicity, we only introduce some types of cost functions in this paper. One could also present other types of cost functions according to practical problems. Besides, for convenience, each object in the universe is supposed to have the same total test cost, and have the same misclassification cost w.r.t the same misclassified class pair.

3.2. Calculation method of average total cost

We begin with discussing the calculation method of total misclassification cost for all objects in the universe, which is crucial for computing the average total cost. A relevant method has been introduced in [64], in which the total misclassification cost is obtained by computing the sum of the misclassification costs of each neighborhood granule in the covering of the universe. However, the obtained value often exceeds the real value. The reason is that, an object often belongs to more than one neighborhood granule in the covering, which results in repetitive computations for the misclassification costs.

To overcome this inexpedience, we present a new method for computing the total misclassification cost. Let be a CVDS, $x \in U$, $B \subseteq C$, and P_B be the confidence level subvector corresponding to B, and let $mc(x, B, P_B)$ denote the misclassification cost of x based on B and P_B. The calculation process of the total misclassification cost and average total cost based on B and P_B is stated as follows, in which Eqs. (17)–(22) are used to compute test costs and misclassification costs. (1) For each object $x \in U$, classify it according to its neighborhood $n_{(B,P_B)}(x)$, and obtain the misclassification cost $mc(x, B, P_B)$. There are two cases.

A) If $\forall\ y \in n_{(B,P_B)}(x)$, $d(y) = d(x)$, we can classify x into the right class, so $mc(x, B, P_B) = 0$.

B) If $\exists\ y \in n_{(B,P_B)}(x)$, $d(y) \neq d(x)$, since the objects in the same neighborhood granule are indistinguishable, they are assumed to have the same decision value in the classification. So we can classify x into the class which minimizes the total misclassification cost for all objects in $n_{(B,P_B)}(x)$. Naturally, the corresponding $mc(x, B, P_B)$ can be obtained.

(2) Compute the total misclassification cost (TMC) and average misclassification cost (AMC) for all objects in U.

$$TMC(U, B, P_B) = \sum_{x\in U} mc(x, B, P_B), \tag{23}$$

$$AMC(U, B, P_B) = \frac{TMC(U, B, P_B)}{|U|}. \tag{24}$$

(3) Compute the average total cost for all objects in U. Since the total test cost for each object is supposed to be the same and is equal to $tc(B, P_B)$, the average total cost (ATC) is

$$ATC(U, B, P_B) = tc(B, P_B) + AMC(U, B, P_B). \tag{25}$$

Based on Examples 2.10 and 3.1, we show the calculation process of average total cost in the following example:

Example 3.2. Since $B = B_3$, from Table 4, we have $n_{(B,P_B)}(x_1) = \{x_1, x_2, x_3, x_4\}$, $n_{(B,P_B)}(x_2) = \{x_1, x_2, x_6\}$, $n_{(B,P_B)}(x_3) = n_{(B,P_B)}(x_4) = \{x_1, x_3, x_4\}$, $n_{(B,P_B)}(x_5) = \{x_5\}$, $n_{(B,P_B)}(x_6) = \{x_2, x_6\}$.

Combined with Table 1, each object is classified according to its neighborhood, and the misclassification costs can be obtained. Concretely, for x_1, if we guess all objects in $n_{(B,P_B)}(x_1)$ belong to class "1", the total misclassification cost for the objects in $n_{(B,P_B)}(x_1)$ is 791.2; on the contrary, if they are classified into class "2", the corresponding total misclassification cost is 3956×3, so we choose class "1" to obtain a less total misclassification cost. Consequently, x_1 is correctly classified and $mc(x_1, B, P_B) = 0$. For x_4, if we categorize all objects in $n_{(B,P_B)}(x_4)$ into class "1", the total misclassification cost corresponding to $n_{(B,P_B)}(x_4)$

Input: (1) a confidence-level-vector-based decision system (U, C, d, V, I, P), the test cost function for each feature, and the misclassification cost function for each misclassified class pair;
(2) the weight δ; for each feature $a \in C$, the confidence level's minimal value p_a^0 and the step-size s_a.
Output: the selected feature subset B and confidence level vector P_B
Method: addition-deletion

//Step 1: Initialize five global variables
Set $B = \emptyset, P_B = (), POS_{(B,P_B)}(\{d\}) = \emptyset, CA = C$, and $S = U$, where B is the selected feature subset, P_B is the selected confidence level vector ("()" denotes an empty vector), $POS_{(B,P_B)}(\{d\})$ is the positive region, CA is the set of unselected features, and S is the set of the objects out of the positive region.
//Step 2: Add feature-granularity elements which have the maximal significances into (B, P_B) step by step
while $(|S| > 0)$ **do**
  for (each $a \in CA$) **do**
    for (each $x \in S$) **do**
      sign_x = true;//sign_x is a global variable used in Algorithm 2
    end for
    if $(\delta == 0)$ **then**
      $p_a^* = p_a^0$;
      FGScomputing(S, B, P_B, a, p_a^*);//Invoke Algorithm 2, and return $IPR_{(B,P_B)}(a, p_a^*)$ and $FGS_{(B,P_B)}(a, p_a^*)$
    else
      for $(p_a = p_a^0; p_a \le 0.997; p_a = p_a + s_a)$ **do**
        FGScomputing(S, B, P_B, a, p_a);//Invoke Algorithm 2, and return $IPR_{(B,P_B)}(a, p_a)$ and $FGS_{(B,P_B)}(a, p_a)$
      end for
      Select p_a^* satisfying $FGS_{(B,P_B)}(a, p_a^*) = \max(FGS_{(B,P_B)}(a, p_a))$;
    end if
  end for
  Select a' satisfying $FGS_{(B,P_B)}(a', p_{a'}^*) = \max(FGS_{(B,P_B)}(a, p_a^*))$;
  if $(FGS_{(B,P_B)}(a', p_{a'}^*) > 0)$ **then**
    $B = B \cup \{a'\}$; $P_B = P_B \sqcup (p_{a'}^*)$; $POS_{(B,P_B)}(\{d\}) = POS_{(B,P_B)}(\{d\}) \cup IPR_{(B,P_B)}(a', p_{a'}^*)$; $CA = CA - \{a'\}$; $S = S - IPR_{(B,P_B)}(a', p_{a'}^*)$;//Update the five variables, in which $P_B = P_B \sqcup (p_{a'}^*)$ refers to Definition 4.1
  else
    exit while;
  end if
end while
//Step 3: Delete redundant feature-granularity elements to make the remaining feature-granularity pair have the minimal average total cost (ATC)
$cmtc = ATC(U, B, P_B)$;//$cmtc$ denotes currently minimal ATC, where ATC is computed according to Section3.2
while (ture) **do**
  for (each $a \in B$) **do**
    Compute $ATC(U, B - \{a\}, P_B - (p_a))$, where $P_B - (p_a)$ refers to Definition 4.2;
  end for
  Select a' satisfying $ATC(U, B - \{a'\}, P_B - (p_{a'})) = \min(ATC(U, B - \{a\}, P_B - (p_a)))$;
  if $(cmtc > ATC(U, B - \{a'\}, P_B - (p_{a'})))$ **then**
    $cmtc = ATC(U, B - \{a'\}, P_B - (p_{a'}))$; $B = B - \{a'\}$; $P_B = P_B - (p_{a'})$;//Update the three variables
  else
    exit while;
  end if
end while
return B, P_B;

Algorithm 1. A δ-weighted heuristic feature-granularity selection algorithm.

is 791.2; conversely, the corresponding total misclassification cost is 3956×2, thus we also choose class "1". In this case, x_4 is misclassified and $mc(x_4, B, P_B) = 791.2$. Similarly, we have $mc(x_2, B, P_B) = mc(x_3, B, P_B) = mc(x_5, B, P_B) = 0$, and $mc(x_6, B, P_B) = 791.2$.

According to Eqs. (23) and (24), $TMC(U, B, P_B) = 0 + 0 + 0 + 791.2 + 0 + 791.2 = 1582.4$, $AMC(U, B, P_B) = \frac{1582.4}{6} \approx 263.73$. Then according to Eq. (25), $ATC(U, B, P_B) = tc(B, P_B) + AMC(U, B, P_B) = 79.12 + 263.73 = 342.85$.

If we use the method in [64], the total misclassification cost is

$791.2 \times 4 = 3164.8$, and the average total cost is $\frac{3164.8}{6} + 79.12 \approx 527.47 + 79.12 = 606.59$. Both the two cost values are much larger than those obtained by our method because the misclassification costs of x_4 and x_6 are repetitively computed. This loophole has been coped with by our method.

3.3. The variable-cost-based multi-granularity feature selection problem

Traditional cost-sensitive feature selection aims at finding a feature subset to minimize the total cost, or equivalently the average total cost, and meanwhile to preserve the information of original decision system as much as possible. The scale of preserved information is often measured with the size of the positive region. In the multi-granularity feature selection, not only features but also feature-value granularities are taken into account based on measurement errors and variable costs. We focus on finding a suitable pair of feature subset and confidence level vector to achieve a good trade-off among feature dimension reduction, feature-value granularity selection and total cost minimization. The variable-cost-based multi-granularity feature selection problem can be formally defined as follows:

Problem 3.3. The variable-cost-based multi-granularity feature selection problem.

Input: a CVDS $S = (U, C, d, V, I, P)$, the test cost function for each feature, and the misclassification cost function for each misclassified class pair;

Output: the pair of selected feature subset B and confidence level vector P_B;

Optimization objectives: (1) $\min(ATC(U, B, P_B))$; and (2) $\max(|POS_{(B,P_B)}(\{d\})|)$.

4. Algorithm design

Since not only features but also their respective feature-value granularities are considered in the variable-cost-based multi-granularity feature selection problem, the problem is more complicated than the existing cost-sensitive feature selection problems. In this section, we design a novel approach to deal with the new problem, which mainly contains a δ-weighted heuristic feature-granularity selection algorithm and a relevant competition strategy. Firstly, the heuristic algorithm is introduced, which follows an addition-deletion strategy. In the algorithm, the monotonicities of the fundamental concepts in the CVRS model are fully used to improve the efficiency. Then, the competition strategy is adopted to run the heuristic algorithm within a given range of δ to choose the best result. Finally, some metrics are presented to evaluate the performance of the proposed approach.

4.1. The δ-weighted heuristic feature-granularity selection algorithm

There are a series of heuristic feature selection algorithms in the literature of rough set applications [13,22,39]. In particular, the weighted heuristic algorithms, in which the weights are often used to adjust the influence ratio of the feature cost to the feature significance, are proposed to deal with the cost-sensitive feature selection problems [31,62]. These weighted heuristic algorithms have been manifested to be effective and efficient, but they have not touched variable costs and feature-value granularities. In this subsection, we introduce a δ-weighted heuristic feature-granularity selection algorithm to address the variable-cost-based multi-granularity feature selection problem. Note that, according to the essence, the feature-granularity selection algorithm is also called multi-granularity feature selection algorithm in Section 5.3 to facilitate the statements regarding the comparison with the existing single-granularity algorithms. There are two operations w.r.t vectors in the proposed algorithm. We define them as follows:

Input: the set S which includes the objects out of the positive region; the selected feature subset B and confidence level vector P_B; an unselected feature a and its corresponding confidence level p_a
Output: the incremental positive region (IPR) and the FGS value
Method: FGScomputing

1: $IPR_{(B,P_B)}(a, p_a) = \emptyset$;
2: **for** (each $x \in S$) **do**
3: **if** (sign_x == true) **then**
4: Compute $n_{(B\cup\{a\},P_B\sqcup(p_a))}(x)$;//Compute the neighborhood of x with respect to the new feature-granularity pair
5: **if** ($\exists X \in U/\{d\}$, such that $n_{(B\cup\{a\},P_B\sqcup(p_a))}(x) \subseteq X$) **then**
6: $IPR_{(B,P_B)}(a, p_a) = IPR_{(B,P_B)}(a, p_a) \cup \{x\}$;//Update the incremental positive region
7: **else**
8: sign_x = false;
9: **end if**
10: **end if**
11: **end for**
12: Compute $tc(a, p_a)$ according to the test cost function of feature a;
13: $FGS_{(B,P_B)}(a, p_a) = |IPR_{(B,P_B)}(a, p_a)| \cdot [tc(a, p_a)]^6$;//Compute the feature-granularity significance
14: return $IPR_{(B,P_B)}(a, p_a), FGS_{(B,P_B)}(a, p_a)$;

Algorithm 2. An algorithm to compute the feature-granularity significance (FGS).

Definition 4.1. Given a vector $X = (x_1, x_2, ...,x_n)$ and a number y, $X \sqcup (y) = (x_1, x_1, ..., x_n, y)$ denotes a new vector obtained by extending vector X and adding y as its last component.

Definition 4.2. Given a vector $X = (x_1, x_2, ...,x_{i-1}, x_i, x_{i+1}, ...,x_n)$, $X - (x_i) = (x_1, x_2, ..., x_{i-1}, x_{i+1}, ..., x_n)$ denotes a new vector obtained by deleting component x_i from vector X.

The proposed heuristic feature-granularity selection algorithm is composed of Algorithms 1 and 2, in which Algorithm 1 is the main framework, and Algorithm 2 is invoked by Algorithm 1. Note that, there are two remarks about the parameters in the input of Algorithm 1, which are given as follows:

Remark 4.3. As a matter of fact, the entire confidence level vector P in the input of Algorithm 1 is not given any specified values, namely it is just a symbol in the heuristic algorithm. For each feature $a \in C$, we only give the minimal confidence level p_a^0 and the maximal confidence level 0.997 at the beginning of the algorithm. Note that, p_a^0 can be set to be $p_a^{\square}$ (the intrinsic minimal value of confidence level computed by Eq. (16)) or a larger value. We will discuss which setting is better in Section 5.

Remark 4.4. Although it seems that there are a series of parameters in the input of Algorithm 1, most of them are intrinsic to the given decision system in real applications, except the weight δ, the confidence level' minimal value p_a^0 and the step-size s_a. Concretely, given a decision system (U, C, d, V, I) in real world, the test cost functions and the misclassification cost functions introduced in Section 3.1 are automatically generated and could not be controlled by users, namely they are intrinsic parameters.

Algorithm 1 follows an addition-deletion strategy discussed in [57] and it mainly contains three steps. First, Step 1 is shown in line 1, in which several global variables are explicitly initialized. Then, as shown in lines 2–23, Step 2 is the addition phase, in which features and their corresponding confidence levels which make the designed feature-granularity significance maximal in each iteration of the while loop are chosen step by step until the positive region could not be expanded any more. Algorithm 2 is invoked in this step. After this step, a feature-granularity pair (B, P_B) is obtained. Finally, as listed in lines 24–35, Step 3 is the deletion phase, in which we delete redundant feature-granularity elements in (B, P_B) according to the average total costs whose computation method has been introduced in Section 3.2. The result of the algorithm is an optimal feature-granularity pair that has the minimal average total cost.

As shown in lines 3–17 of Algorithm 1, there is a two-layer selection in the addition phase of the heuristic algorithm. Firstly, for each unselected feature a, we try the confidence level p_a from the minimal value p_a^0 to the maximal value 0.997 with a step-size $s_a \in (0, 1)$, and choose the best value p_a^* which makes the feature-granularity significance maximal. Next, for all unselected features, their maximal feature-granularity significance values are compared to select the best feature a'. The feature-granularity significance function plays an important role in the process. Let B and P_B respectively denote the selected feature subset and confidence level vector, then the incremental positive region (IPR) induced by feature $a \in C - B$ and confidence level p_a is denoted as

$$IPR_{(B,P_B)}(a, p_a) = POS_{(B\cup\{a\},P_B\sqcup(p_a))}(\{d\}) - POS_{(B,P_B)}(\{d\}). \tag{26}$$

According to Theorem 2.11, $|IPR_{(B,P_B)}(a, p_a)| \geq 0$. Then the δ-weighted feature-granularity significance (FGS) function is defined as

$$FGS_{(B,P_B)}(a, p_a) = |IPR_{(B,P_B)}(a, p_a)| \cdot [tc(a, p_a)]^{\delta}, \tag{27}$$

where the exponent $\delta \leq 0$ is a parameter set by the user to adjust the weight ratio between the size of incremental positive region $|IPR_{(B,P_B)}(a, p_a)|$ and the test cost $tc(a, p_a)$. Because we choose the feature-granularity pair with the maximal feature-granularity significance in the algorithm, it is known from Eq. (27) that the feature-granularity pair which can expand the positive region more largely or which has cheaper test cost is preferred. In fact, the positive region is related to the total misclassification cost. If an object belongs to the positive region, its neighborhood granule is consistent, namely, the objects in the neighborhood have the same decision value, so the misclassification cost is 0 for the object. Therefore, maximizing the positive region is helpful to minimize the total misclassification cost. Essentially, test costs and misclassification costs are both considered in the feature-granularity significance function. The algorithm aims at finding a good trade-off between them. Specially, if $\delta = 0$, test costs are not considered in the feature-granularity significance function.

In particular, the above-mentioned monotonicities w.r.t. the fundamental concepts in CVRS are well used to make the addition phase of the heuristic algorithm more efficient. Firstly, as shown in lines 4–6 of Algorithm 1 and lines 3–10 of Algorithm 2, we use global variables $\{\text{sign}_x\}$ to judge whether to continue computing the neighborhood of x in Algorithm 2 when p_a increases. Concretely, assuming that $n_{(B\cup\{a\},P_B\sqcup(p_a))}(x) \nsubseteq X$, $\forall X \in U/\{d\}$, we label "sign_x = false" to avoid computing $n_{(B\cup\{a\},P_B\sqcup(p_a+s_a))}(x)$ because $n_{(B\cup\{a\},P_B\sqcup(p_a+s_a))}(x) \nsubseteq X$ according to Theorem 2.13, and $x \notin IPR_{(B,P_B)}(a, p_a + s_a)$ in this case. So we need not consider this kind of objects when computing $IPR_{(B,P_B)}(a, p_a + s_a)$. In this way, the time consumption of calculating the incremental positive region can be reduced effectively. Secondly, as shown in lines 7–15 of Algorithm 1, selecting the best confidence level p_a^* for $\delta = 0$ is much faster than that for $\delta < 0$. If $\delta = 0$, i.e., only $|IPR_{(B,P_B)}(a, p_a)|$ is considered in the significance function $FGS_{(B,P_B)}(a, p_a)$, p_a^* is equal to the minimal confidence level p_a^0. The reason is that $FGS_{(B,P_B)}(a, p_a)$ is equal to $|IPR_{(B,P_B)}(a, p_a)|$ and the latter is maximal at this time according to Theorem 2.13. Accordingly, if $\delta = 0$, we can immediately choose $p_a^* = p_a^0$ and compute $FGS_{(B,P_B)}(a, p_a^*)$, while the calculation of $FGS_{(B,P_B)}(a, p_a)$ is not needed when $p_a > p_a^0$. Because we often run the heuristic algorithm within a range of δ including $\delta = 0$ for comparison by using the competition strategy introduced in the following context, this design can improve the efficiency of our approach. Finally, as shown in line 19 of Algorithm 1, by using $S = S - IPR_{(B,P_B)}(a', p_{a'}^*)$, the objects needed to be judged whether they belong to the positive region get fewer and fewer as the feature-granularity selection goes on. Given $B_1 \subseteq B_2 \subseteq C$ and $P_{B_1} \sqsubseteq P_{B_2}$, if $x \in POS_{(B_1,P_{B_1})}(\{d\})$, we have $x \in POS_{(B_2,P_{B_2})}(\{d\})$ according to Theorem 2.11. It means that we just need to discuss the objects in $U - POS_{(B_1,P_{B_1})}(\{d\})$ when computing $POS_{(B_2,P_{B_2})}(\{d\})$ because the objects in $POS_{(B_1,P_{B_1})}(\{d\})$ are necessarily in $POS_{(B_2,P_{B_2})}(\{d\})$. Consequently, the computation will be reduced significantly.

Now we analyze the time complexity of the heuristic feature-granularity selection algorithm. Because the algorithm is composed of Algorithms 1 and 2, and Algorithm 2 is invoked by Algorithm 1, we first analyze the time complexity of Algorithm 2. In Algorithm 2, the key step is to compute the neighborhood of object $x \in S$, which is shown in line 4. According to Eqs. (6)–(9), the time complexity of this step is $O(|U||B \cup \{a\}|) = O(|U|(|B| + 1))$. In the worst case, $\forall x \in S$, "sign_x = = true", so the time complexity of Algorithm 2 is $O(|S||U|(|B| + 1))$. Then, we combine Algorithms 1 and 2 to analyze the time complexity of the addition phase of the heuristic algorithm. Assuming that $|B| = k$ after this phase and that selecting a feature-granularity element averagely leads to $|U|/k$ objects added into the positive region, we have $|S| = |U|\left(1 - \frac{i}{k}\right) = |U|\frac{k-i}{k} (i = 0, 1, 2, ..., k - 1, k)$ with the while loop going on. In this phase, the key step is to compute the incremental positive region and the feature-granularity significance, i.e. to invoke Algorithm 2, which is shown in line 9 of Algorithm 1 if $\delta = 0$, and line

12 if $\delta < 0$. If $\delta = 0$, the total computational time of the addition phase is

$$O(|C|\cdot|U|^2+2(|C|-1)\cdot|U|^2\frac{k-1}{k}+3(|C|-2)\cdot|U|^2\frac{k-2}{k} +\cdots+k\cdot(|C|-k+1)\cdot|U|^2\cdot\frac{1}{k}\Big) = O(\frac{|U|^2}{k}\sum_{i=0}^{k-1}(i+1)(k-i)(|C|-i)); \tag{28}$$

and if $\delta < 0$, let n_p=$\max_{a\in C}\left\lceil\frac{0.997-p_a^0}{s_a}\right\rceil$, then in the worst case, the total computational time of this phase is

$$O(\frac{n_p|U|^2}{k}\sum_{i=0}^{k-1}(i+1)(k-i)(|C|-i)). \tag{29}$$

In practice, it is often found that most of objects are grouped into the positive region at the beginning of the addition phase, and the above-mentioned monotonicities can further improve the efficiency, so the computational time of this phase is usually much less than that shown in Eq. (28) or Eq. (29). Finally, we analyze the time complexity of the deletion phase of the heuristic algorithm. The key step in this phase is to compute the average total cost for given feature-granularity pair, which is shown in line 27 of Algorithm 1. This step is mainly composed of two computations. The first one, whose time complexity is $O(|B-\{a\}||U|^2) = O((|B|-1)|U|^2)$, computes the neighborhoods for each object in U; and the second one, whose time complexity is $O(|U||V_d|)$ according to Section 3.1 and 3.2, calculates the average total cost. Assuming that $l < k$ features and their associated confidence levels are removed in the deletion phase, the total computational time of this phase is

$$O(k\cdot((k-1)\cdot|U|^2+|U||V_d|)+(k-1)\cdot((k-2)\cdot|U|^2 +|U||V_d|)+\cdots+(k-l)\cdot((k-l-1)\cdot|U|^2+|U||V_d|)) = O(\sum_{i=0}^{l}(k-i)((k-i-1)|U|^2+|U||V_d|)).$$

According to the above analysis, if $\delta = 0$, the total computational time of the heuristic algorithm is

$$O(\frac{|U|^2}{k}\sum_{i=0}^{k-1}(i+1)(k-i)(|C|-i)+\sum_{i=0}^{l}(k-i)((k-i-1)|U|^2 +|U||V_d|)); \tag{30}$$

and if $\delta < 0$, its counterpart is

$$O(\frac{n_p|U|^2}{k}\sum_{i=0}^{k-1}(i+1)(k-i)(|C|-i)+\sum_{i=0}^{l}(k-i)((k-i-1)|U|^2 +|U||V_d|)). \tag{31}$$

As discussed above, for the heuristic algorithm, the computational time of its addition phase is often much less than that formulated in Eq. (28) when $\delta = 0$ and Eq. (29) when $\delta < 0$, so its total computational time is usually less than that shown in Eq. (30) or Eq. (31), which will be validated in Section 5.3.

4.2. The competition strategy

As mentioned above, the parameter δ in Eq. (27) is used to adjust the weight ratio between the size of incremental positive region and the test cost. Different δ settings result in different feature-granularity selection results, but the users do not know which δ value will generate less average total cost in advance. To tackle this problem, we propose a competition strategy. The strategy is similar to that in [31] in terms of designing ideas, but the optimization objective function has changed because of the new environment.

Concretely, we specify a set of δ values and compute the feature-granularity selection results for each δ respectively by using the heuristic algorithm. Through comparison, the feature-granularity pair with minimal average total cost will be chosen. Formally, let (B_δ, P_δ) denote the feature-granularity pair constructed by the heuristic algorithm with the weight δ, and L denote the set of user-specified δ values, then the minimal average total cost and the corresponding optimal feature-granularity pair can be obtained by the following equation:

$$ATC_L = min_{\delta\in L}ATC(B_\delta, P_\delta), \tag{32}$$

where $ATC(B_\delta, P_\delta)$ is the abbreviation of $ATC(U, B_\delta, P_\delta)$. This kind of abbreviations is also used in Section 4.3.

By using the competition strategy, the heuristic algorithm needs to be run for $|L|$ times, but this is acceptable for relatively small $|L|$ since the heuristic algorithm is fast, which will be validated in Section 5.3. If $|L|$ is large, we can run the program on several computers in parallel to reduce the time consumption. Although the competition strategy is simple, by using it, users do not have to know the best setting of δ in advance, and the quality of the feature-granularity selection results can be enhanced effectively, which will be manifested in Section 5.

4.3. Evaluation metrics

To compare our approach with the existing approaches, some evaluation metrics are introduced, which is started from the difference ratio of cost. Let c denote a type of cost, for a dataset in a particular environment, the cost difference ratio (DR) is defined as

$$DR_c = \frac{c_2-c_1}{c_1} = \frac{c_2}{c_1} - 1,\ c_1 \neq 0, \tag{33}$$

where c_1 and c_2 denote the costs obtained by two different approaches, respectively. Obviously, $DR_c \geq -1$. If $c_2 \leq c_1$, $-1 \leq DR_c \leq 0$; otherwise, $DR_c > 0$. The smaller $|DR_c|$ is, the closer c_2 is to c_1. Specially, $DR_c = -1$ when $c_2 = 0$, and $DR_c = 0$ when $c_2 = c_1$. Take the average total cost as an example. Let ATC_1 and ATC_2 denote the average total costs obtained by Approach 1 and Approach 2 of feature selection, respectively. If $ATC_1 = 200$ and $ATC_2 = 210$, we have $DR_{ATC} = \frac{210-200}{200} = 0.05$, so Approach 1 performs better than Approach 2 on minimizing the average total cost.

Note that, a concept called cost exceeding factor was introduced in [31] to compute the exceeding ratio between the cost obtained by a heuristic feature selection algorithm and the minimal cost obtained by the exhaustive algorithm, so the cost exceeding factor is no less than 0. Compared with the cost exceeding factor, the defined cost difference ratio has a wider value range, and it is also more general because any two feature selection algorithms can be compared by using it.

As will be shown in Section 5.3, our approach is mainly compared with three state-of-art feature selection or feature-granularity selection approaches from the perspective of minimizing average total costs in the experiments, thus we discuss the relevant metrics for the approaches here. For example, let (B, P_B) and (R, p) respectively denote the feature-granularity selection results obtained by our approach and the approach proposed in [64] for a decision system with a particular cost setting. Then the difference ratio between (B, P_B) and (R, p) in terms of average total cost is

$$DR_{ATC}(B, P_B) = \frac{ATC(B, P_B)}{ATC(R, p)} - 1. \tag{34}$$

Since we compare the two approaches by running them with different cost settings for each decision system in Section 5.3, the corresponding average difference ratio (ADR) can be obtained as follows:

$$ADR_{ATC} = \frac{\sum_{i=1}^{K}DR_{ATC}(B_i, P_{B_i})}{K} = \frac{1}{K}\sum_{i=1}^{K}\left(\frac{ATC(B_i, P_{B_i})}{ATC(R_i, p_i)} - 1\right), \tag{35}$$

where (B_i, P_{B_i}) and (R_i, p_i) are the results of feature-granularity selection

Table 6
Data information.

Dataset	Domain	Samples	Features	Classes
Diab	Clinic	768	8	2
German	Finance	1000	20	2
Heart	Clinic	303	13	5
Image	Graphics	2310	18	7
Iono	Physics	351	34	2
Liver	Clinic	345	6	2
Sonar	Physics	208	60	2
Wdbc	Clinic	569	30	2
Wpbc	Clinic	198	33	2

undertaken on the *i*th cost setting by the two approaches, respectively. The average difference ratio index compares the overall performance of different feature selection or feature-granularity selection algorithms from a statistical perspective.

5. Experiments

As discussed in Section 4, the proposed multi-granularity feature selection approach mainly contains the δ-weighted heuristic feature-granularity selection algorithm and the relevant competition strategy. In this section, we try to answer the following questions by experimentation:

(1) Is the heuristic feature-granularity selection algorithm appropriate for the multi-granularity feature selection problem?
(2) Can the competition strategy enhance the quality of the results?
(3) Is there an optimal setting or a rational value range for each extrinsic parameter in the heuristic algorithm for any dataset?
(4) Does the multi-granularity feature selection approach perform better than the existing single-granularity approaches?

We start from generating the parameter values, which mainly include those of test cost functions and misclassification cost functions, for experiments according to reality. Then we give some representative results of multi-granularity feature selection and analyze the results. Finally, we further study the performance of our approach through making comparisons from two perspectives. One is comparing between our approach and the existing approaches, the other is comparing among different values for the extrinsic parameters, especially the weight δ. All the algorithms are running on the same computation platform (CPU: Intel(R) Core(TM) i7-6500U CPU @ 2.50 GHz; RAM: 8.00 GB; OS: Windows 10).

5.1. Data generation

We test the performance of the multi-granularity feature selection approach on nine standard datasets in the UCI repository, whose basic information is listed in Table 6. To handle the data more easily, data items are normalized onto [0,1]. For convenience, missing values are directly set to be 0.5. The constant k in Eq. (7) is set to be 0.05. To facilitate the understanding of experiment results w.r.t. the feature values' multi-granularity, both the confidence level's minimal value p_a^0 and the step-size s_a are respectively set to be the same among all features in the following experiments (when this assumption does not hold, it is known that the results are similar by experimentation).

Since the UCI datasets have no intrinsic test costs and misclassification costs, we create the two kinds of cost functions for experimentation according to Section 3.1. Naturally, Eqs. (17)–(22) can be used to generate multiple different cost settings. Here we mainly introduce one type of cost setting for brevity, and if not specified, the results mentioned below corresponds to this type of cost setting. First, the values of test cost parameters are set according to the number of features. For the datasets whose feature number is less than 15, the highest test costs $tc(a)$ in Eq. (17) are set to be uniformly distributed random integers lying within [20,100]; and for other datasets, the counterparts are set to be lying within [20,200]. The test cost adjusting factors λ_a in the equation are set to be uniformly distributed random decimals lying within [0,1]. Then, the values of misclassification cost parameters are set according to the application background of the dataset. For the medical datasets Diab, Heart, Liver, Wdbc and Wpbc, misclassification cost functions are set by using Eq. (21), in which the misclassification cost penalty factors $\gamma_{(m,\ n)}$ are supposed to be integers lying within [10,100]; for other four datasets, misclassification cost functions are set by using Eq. (20), in which the constant misclassification costs $MC_{(m,\ n)}$ are assumed to be integers lying within [5000,50000]. Note that, in order to be close to reality, the values of misclassification cost parameters are set carefully. For one example, there are two classes "recur" and "nonrecur" for the objects in Wpbc dataset, which are abbreviated as "R" and "N" respectively. Generally speaking, the cost of misclassifying an object from class "R" to class "N" is larger than that for misclassifying from class "N" to class "R", because the former will delay the treatment of the life-threatening disease, while the latter may only cause unnecessary spending and psychological burden. So we set $\gamma_{(R,N)} = 100$ and $\gamma_{(N,R)} = 10$. For another example, there are seven classes in Image dataset, and the misclassification consequences are similar between different classes, so all the misclassification costs are set to be 10,000 for convenience.

5.2. Representative results and the analyses

We let $\delta = -4, ..., -0.5, 0$, and let the pair (p_a^0, s_a) be (0.1,0.1), (0.1,0.2) and (0.2,0.15), respectively. Note that, the aim of testing different values of p_a^0 is to find whether the confidence level's intrinsic minimal value $p_a^{\square}$ is a good parameter assignment in the proposed heuristic algorithm, and whether there exists a value which is larger than $p_a^{\square}$ and performs better than $p_a^{\square}$. For each dataset in Table 6 and each (p_a^0, s_a) pair, we generate 1000 different cost settings, then run the heuristic algorithm with these data settings for all δ values. In order to

Table 7
A representative feature-granularity selection result for Liver dataset with $(p_a^0, s_a) = (0.1, 0.1)$, where p_a^0 and s_a are respectively the minimal value and the step-size of the confidence level, TTC denotes the total test cost for each object, and AMC and ATC respectively denote the average misclassification cost and average total cost for all objects.

δ	TTC	AMC	ATC	Feature subset	Confidence level vector
−4	152.0487	92.5217	244.5705	{1,2,3,4,6}	(0.997,0.9,0.9,0.8,0.997)
−3.5	156.5427	45.2174	201.76	{1,2,3,4,6}	(0.997,0.9,0.6,0.8,0.997)
−3	125.0852	14.4928	139.5779	{1,2,4,6}	(0.9,0.9,0.3,0.997)
−2.5	125.0852	14.4928	139.5779	{1,2,4,6}	(0.9,0.9,0.3,0.997)
−2	86.4529	52.3478	138.8007	{2,4,6}	(0.3,0.7,0.997)
−1.5	88.0332	35.7101	123.7434	{2,4,6}	(0.3,0.7,0.9)
−1	109.6507	3.1594	112.8101	{2,3,6}	(0.3,0.1,0.9)
−0.5	94.9563	13.6232	**108.5795**	{2,4,6}	(0.3,0.3,0.9)
0	102.106	20.6957	122.8016	{2,3}	(0.1,0.1)

Table 8
A representative feature-granularity selection result for Wpbc dataset with $(p_a^0, s_a) = (0.1, 0.1)$.

δ	TTC	AMC	ATC	Feature subset	Confidence level vector
−4	42.1923	20.202	62.3943	{6,7,12,14,20,31}	(0.997,0.997,0.997,0.997,0.8,0.997)
−3.5	42.1923	20.202	62.3943	{6,7,12,14,20,31}	(0.997,0.997,0.997,0.997,0.8,0.997)
−3	42.1923	20.202	62.3943	{6,7,12,14,20,31}	(0.997,0.997,0.997,0.997,0.8,0.997)
−2.5	42.6963	15.1515	57.8479	{6,7,12,14,20,31}	(0.997,0.997,0.997,0.997,0.7,0.997)
−2	38.2619	10.101	48.3629	{6,10,12,20,31}	(0.997,0.9,0.997,0.6,0.997)
−1.5	38.2619	10.101	48.3629	{6,10,12,20,31}	(0.997,0.9,0.997,0.6,0.997)
−1	28.9484	7.5758	**36.5241**	{6,12,20,31}	(0.997,0.997,0.1,0.997)
−0.5	39.9784	10.101	50.0794	{9,12,29,31}	(0.997,0.997,0.1,0.997)
0	128.8477	0	128.8477	{4,7}	(0.1,0.1)

save space, we only list the representative results of Liver and Wpbc with $(p_a^0, s_a) = (0.1, 0.1)$, which are shown in Tables 7,8, where TTC denotes the total test cost for each object. The boldface numbers in the fourth columns of the tables are the minimal average total costs, and the integers in the fifth columns are the indexes of selected features.

The following observations could be made from the results:

(1) Only a part of candidate features are selected; more importantly, even though both p_a^0 and s_a are respectively set to be the same among all features, the best confidence levels p_a^* between different chosen features are often not the same except that p_a^* for each selected feature is p_a^0 when $\delta = 0$ (This has been explained in Section 4.1). It validates that the heuristic algorithm can effectively solve the multi-granularity feature selection problem.

(2) Although the feature-granularity selection results may be the same between two adjacent values of δ in some cases, they change with the value of δ in general, and the competition strategy could be used to further improve the quality of the results. By using the competition strategy, the minimal average total cost and the best feature-granularity pair can be obtained within the given range of δ.

(3) In general, with the increase of δ value, the dimension of selected features reduces gradually apart from some exceptions. The reason is that, the bigger δ is, the larger proportion the size of incremental positive region occupies in the feature-granularity significance according to Eq. (27). Hence, less features are usually needed to maximize the positive region in the addition phase of the heuristic algorithm.

To investigate the influence of parameters p_a^0 and s_a, we also test our heuristic algorithm with the same data settings (i.e. the same settings of decision systems and cost functions) under the three (p_a^0, s_a) pairs. Two examples of Wpbc dataset are given in Table 9, in which the boldface numbers are the minimal average total costs. It is found from the table that, in the first example, the minimal average total cost for $(p_a^0, s_a) = (0.1, 0.2)$ is less than that for $(p_a^0, s_a) = (0.1, 0.1)$, and the corresponding cost for $(p_a^0, s_a) = (0.2, 0.15)$ is more than that for $(p_a^0, s_a) = (0.1, 0.1)$. While in the second example, the minimal average total cost for $(p_a^0, s_a) = (0.1, 0.2)$ is equal to that for $(p_a^0, s_a) = (0.1, 0.1)$, and the corresponding cost for $(p_a^0, s_a) = (0.2, 0.15)$ is less than that for $(p_a^0, s_a) = (0.1, 0.1)$. In fact, with the increase of p_a^0 and/or s_a, the average total cost for each δ value may increase, decrease, or stay the same, so the minimal average total cost may also grow, drop or remain unchanged. Hence, there is not an obvious change rule of the minimal average total cost with the change of (p_a^0, s_a).

Combining Tables 7,8 with Table 9, we find that the average total costs are usually smaller when δ takes value within interval $[-3, 0)$, and the minimal average total cost and optimal feature-granularity pair often fall within this range of δ. The reason is that, if $\delta = 0$, test costs are not taken into account for computing the feature-granularity significance in Eq. (27); while the δ-weighted test-cost-related value usually far outweighs the size of incremental positive region if $\delta < -3$. In these two cases, a good trade-off between test costs and misclassification costs often cannot be achieved. Hence, to obtain better results, the value range of δ is that $\delta < 0$ and $|\delta|$ is not too large ($|\delta| \leq 3$ here).

In summary, our approach can effectively solve the multi-granularity feature selection problem. A good trade-off among feature dimensionality reduction, feature-value granularity selection and total cost minimization can be obtained by the approach. We will further investigate the performance of our approach and the influence of the extrinsic parameters in the next subsection.

5.3. Comparisons and analyses

In this subsection, we study the performance of the proposed measurement errors and variable costs based multi-granularity feature selection approach by making comparisons from two aspects. One aspect is that, we compare our multi-granularity approach with three state-of-art error-based cost-sensitive feature selection approaches mentioned in Section 1, all of which are essentially single-granularity approaches [6,63,64]. The other aspect is that, the performance of the designed heuristic algorithm is compared among different values of extrinsic parameters to study the influence of these parameters. Two main metrics are employed to compare the performance. One metric is the average difference ratio introduced in Section 4.3, which is used to compare the effectiveness from the viewpoint of total cost minimization. The other metric is the run-time for comparing the computational efficiency.

To study the influence of different values of extrinsic parameters, we first discuss detailedly the comparisons between our multi-granularity approach and the single-granularity approach in [64]. For brevity, if not specified, in this subsection the phrase "single-granularity approach/algorithm" refers to the one in [64]. In order to facilitate the comparisons, we uniformly use Eqs. (17)–(22) to construct the cost settings for the compared approaches, and uniformly use the calculation method presented in Section 3.2 to compute the average total cost. Let $\delta = -4, -3.75, ..., -0.25, 0$, and let $(p_a^0, s_a) = (0.1, 0.1), (0.1, 0.2), (0.2, 0.15)$. For each dataset and each (p_a^0, s_a) pair, 1000 different cost settings are generated; and for each data setting, we run our multi-granularity feature selection algorithm (i.e., the heuristic feature-granularity selection algorithm proposed in Section 4.1) with each δ value and run the single-granularity feature selection algorithm.[1] The average difference ratios w.r.t. average total cost are computed, which are depicted in Figs. 3–5. Note that, according to Eq. (35), the less the average difference ratio is, the better our algorithm is in terms of minimizing the total cost. In particular, if the average difference ratio is less than 0, the average total cost obtained by our multi-granularity algorithm is less than that of the single-granularity algorithm. From the figures, we observe the follows:

(1) For each (p_a^0, s_a) pair, there is at least one average difference ratio less than or close to 0 for all datasets. Especially in Diab, German, Heart, Image and Liver, a series of average difference ratios are

[1] For each (p_a^0, s_a) pair, we generate 50 different cost settings for German dataset and 100 different cost settings for Image dataset, as on each cost setting the single-granularity algorithm takes at least 1 h and 0.15 h for German and Image respectively.

Table 9
Two exemplary changes of minimal average total costs with the increase of p_a^0 and/or s_a.

No.	(p_a^0, s_a)	$\delta = -4$	$\delta = -3.5$	$\delta = -3$	$\delta = -2.5$	$\delta = -2$	$\delta = -1.5$	$\delta = -1$	$\delta = -0.5$	$\delta = 0$
1	(0.1,0.1)	86.0389	86.0389	86.0389	75.8952	75.8952	68.6087	**63.1366**	**63.1366**	125.8582
	(0.1,0.2)	73.1694	73.1694	**58.3503**	**58.3503**	**58.3503**	**58.3503**	66.8415	66.8415	125.8582
	(0.2,0.15)	84.8507	84.8507	84.8507	84.8507	71.2236	**66.774**	**66.774**	**66.774**	121.0185
2	(0.1,0.1)	100.0722	100.0722	70.8993	67.5602	65.7547	65.7547	65.7547	**63.7621**	334.0319
	(0.1,0.2)	70.8993	70.8993	70.8993	65.7547	65.7547	65.7547	65.7547	**63.7621**	334.0319
	(0.2,0.15)	90.6442	85.8204	85.8204	70.2147	70.2147	70.2147	67.4919	**62.8244**	270.4155

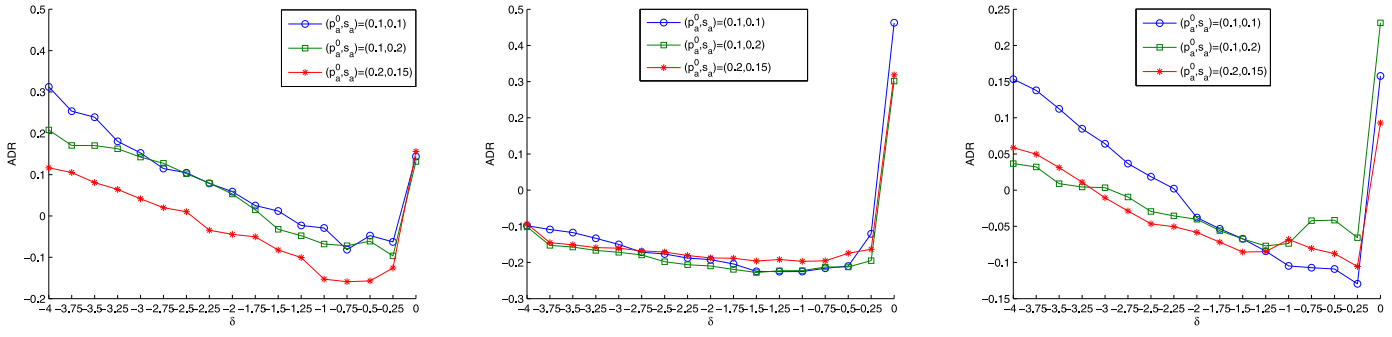


Fig. 3. Average difference ratios (ADR): (a) Diab, (b) German, (c) Heart.

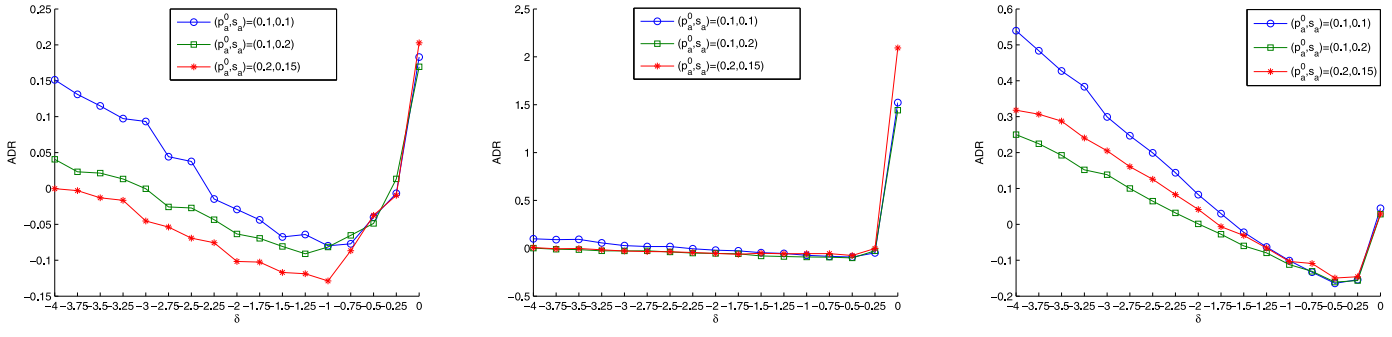


Fig. 4. Average difference ratios: (a) Image, (b) Iono, (c) Liver.

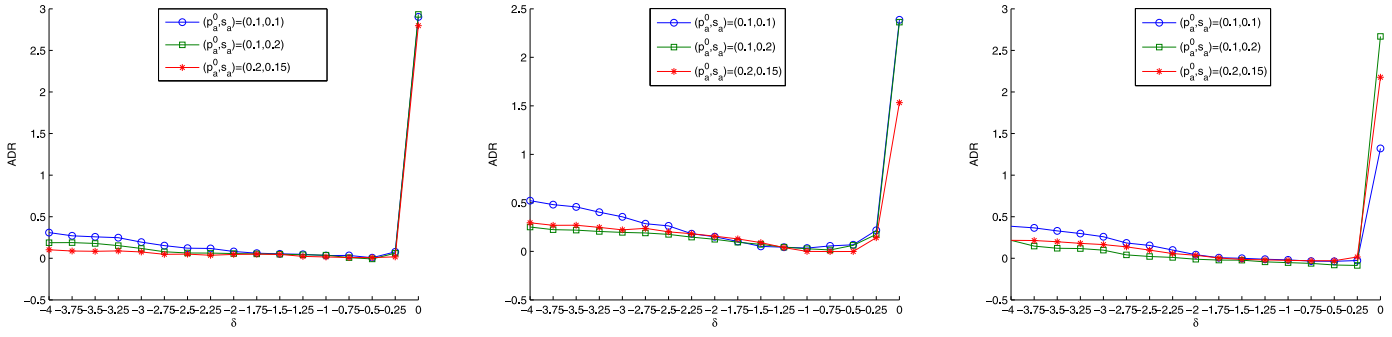
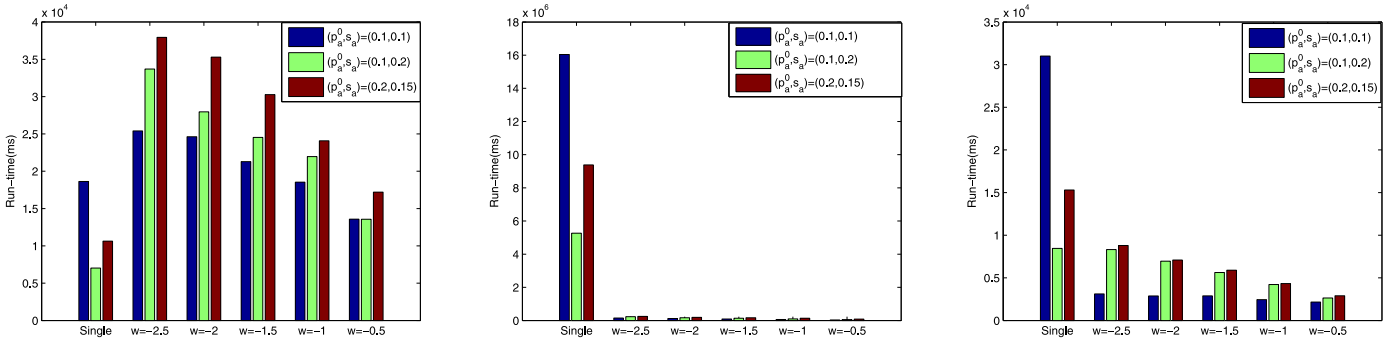


Fig. 5. Average difference ratios: (a) Sonar, (b) Wdbc, (c) Wpbc.

dramatically less than 0 for each (p_a^0, s_a) pair. Naturally, all the minimal average difference ratios are less than or close to 0 in these datasets. Therefore, the minimal average total costs obtained by our multi-granularity feature selection approach are less than or close to those obtained by the approach in [64], which are minimal in the single-granularity context. Hence, our approach performs well on minimizing the total cost.

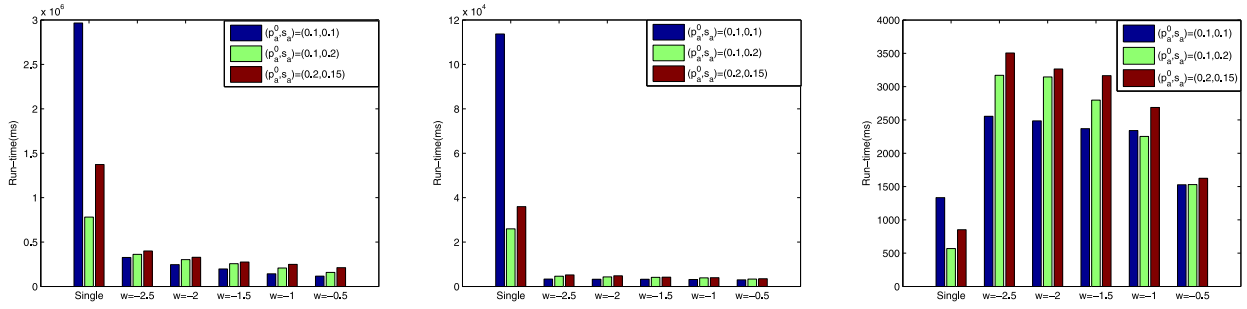
(2) According to Eq. (35), we know that all the minimal average total costs and optimal feature-granularity pairs obtained by our approach fall within $\delta \in [-1.5, -0.25]$ for each dataset and each (p_a^0, s_a) pair. Moreover, when $\delta = 0$ or $\delta < -3$, the average total costs are often larger than those corresponding to $\delta \in [-3, 0)$. These observations are in accord with those in the penultimate paragraph of Section 5.2, and a smaller value range is obtained for the weight δ, namely $\delta \in [-1.5, -0.25]$.

(3) There is not a universally best setting for δ. However, there is



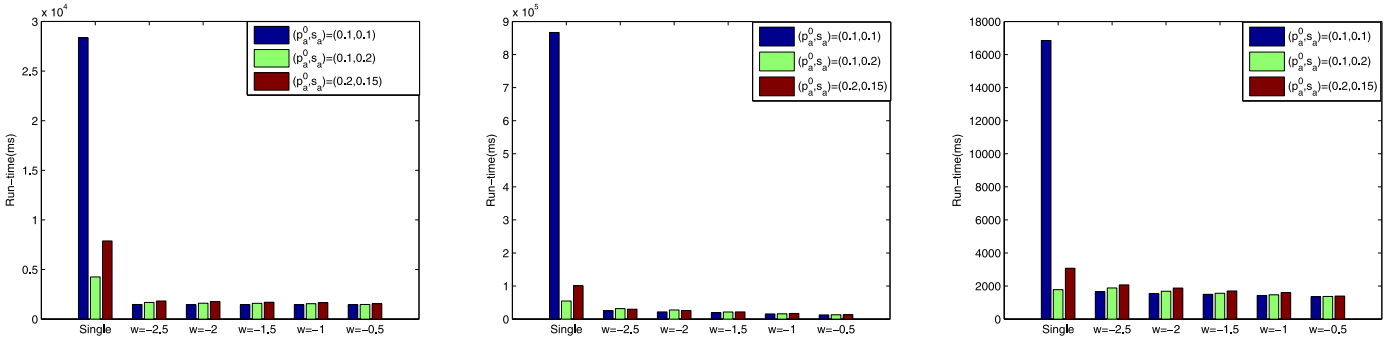
(a) (b) (c)

Fig. 6. Comparisons of average run-time: (a) Diab, (b) German, (c) Heart, where "single" denotes the single-granularity feature selection algorithm proposed in [64], and "w = *" denotes the proposed heuristic multi-granularity algorithm with weight $\delta = *$.



(a) (b) (c)

Fig. 7. Comparisons of average run-time: (a) Image, (b) Iono, (c) Liver.



(a) (b) (c)

Fig. 8. Comparisons of average run-time: (a) Sonar, (b) Wdbc, (c) Wpbc.

also not a big change for the average difference ratios when $\delta \in [-1.5, -0.25]$ in most cases. It means that good feature-granularity selection results could usually be achieved within this range of δ even when the competition strategy is not adopted.

Note that, for a dataset, if the average difference ratios corresponding to one (p_a^0, s_a) pair are lower than those corresponding to another two (p_a^0, s_a) pairs, it does not mean that the former (p_a^0, s_a) value is better. The reason is that, as discussed in Section 5.2, with the increase of p_a^0 and/or s_a, the average total costs obtained by our heuristic multi-granularity algorithm may grow, drop or remain unchanged; and it is found in the experiments that those costs gained by the single-granularity algorithm may also increase, decrease or stay the same. Moreover, for each dataset, the test cost settings are generated randomly and they are not necessarily the same among the three (p_a^0, s_a) pairs. Thus we cannot know which (p_a^0, s_a) value is the best according to the average difference ratio index. Fortunately, the question could be answered by the comparisons of average run-time in the above-mentioned experiments. The results are shown in Figs. 6–8, in which the unit of run-time is 1 ms. From the figures, we note the follows:

(1) In general, our heuristic multi-granularity feature selection algorithm performs well on the computational efficiency. Although sometimes our algorithm runs more slowly than the single-granularity algorithm when the dataset only has several features, it runs much faster than the latter when the dataset has many features; meanwhile, it can obtain multiple feature-value granularities for selected features while the latter cannot. Besides, as discussed above, good feature-granularity selection results could usually be obtained within $\delta \in [-1.5, -0.25]$ even when the competition strategy is not used. Even if the competition strategy is adopted, the multi-granularity feature selection approach is still efficient because the range of δ value is short ($\delta \in [-1.5, -0.25]$).

(2) With the increase of the (p_a^0, s_a) value, the run-time of our heuristic multi-granularity algorithm often gets large. The reason is that, big (p_a^0, s_a) will generate large error intervals. In this case, more

Table 10
A representative feature-granularity selection result for Wpbc dataset with $(p_a^0, s_a) = (0.1, 0.1)$ and another type of cost setting.

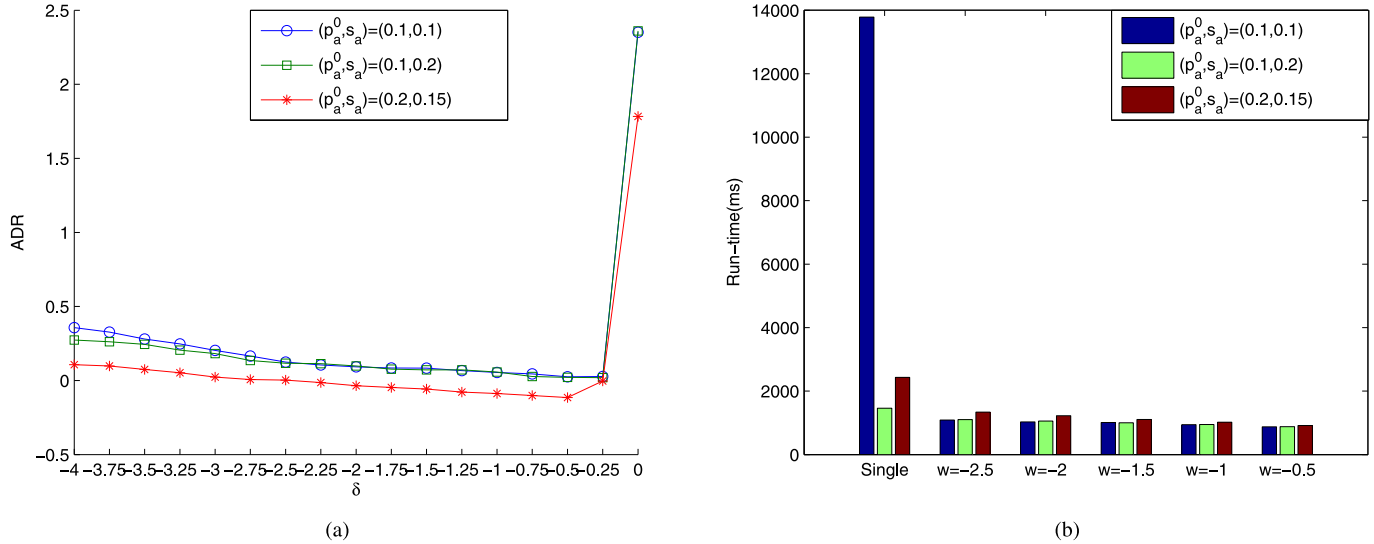
δ	TTC	AMC	ATC	Feature subset	Confidence level vector
−4	415.3408	202.0202	617.361	{5,13,27,30}	(0.9,0.9,0.9,0.9)
−3.5	415.3408	202.0202	617.361	{5,13,27,30}	(0.9,0.9,0.9,0.9)
−3	528.3196	50.5051	578.8246	{5,16,22,27,30}	(0.9,0.7,0.9,0.7,0.9)
−2.5	528.3196	50.5051	578.8246	{5,16,22,27,30}	(0.9,0.7,0.9,0.7,0.9)
−2	359.5862	50.5051	410.0912	{5,22,27}	(0.1,0.9,0.7)
−1.5	359.5862	50.5051	410.0912	{5,22,27}	(0.1,0.9,0.7)
−1	348.588	0	**348.588**	{5,27,30}	(0.1,0.5,0.9)
−0.5	348.588	0	**348.588**	{5,27,30}	(0.1,0.5,0.9)
0	1,167.5526	0	1,167.5526	{4,7}	(0.1,0.1)

feature-granularity elements are usually needed to reduce the increase of misclassification rate induced by the large error intervals in the addition phase of the algorithm, and these feature-granularity elements are required to be checked whether to be redundant in the deletion phase. Consequently, the run-time increases for both the addition phase and the deletion phase of the algorithm. So (0.1,0.1) is the best among the three (p_a^0, s_a) values for our heuristic multi-granularity algorithm in terms of computational efficiency. Besides, from Table 9, it is known that there is usually not a big change of minimal average total costs between the three (p_a^0, s_a) values. Hence, if allowed, one could choose p_a^0 to be the confidence level's intrinsic minimal value $p_a^{\square}$, and choose s_a to be a rational small value.

(3) For the same dataset and the same (p_a^0, s_a) value, the run-time of our algorithm often grows with the decrease of δ although the growth may be slow sometimes. The reason is that, the less the δ value is, the smaller proportion the size of incremental positive region occupies in the feature-granularity significance according to Eq. (27). In this case, more feature-granularity elements are often needed to maximize the positive region in the addition phase of the algorithm. Similarly with that in (2), the run-time of the algorithm will increase. Hence, $[-1.5, -0.25]$ is a desirable value range of δ in terms of the computational efficiency.

It is notable that, for simplicity only one type of cost setting is introduced in Section 5.1, but as a matter of fact we test our approach with multiple types of cost settings in the experiments. And it is found from the experimental results that our multi-granularity feature selection approach always has a good performance, especially when comparing with the existing single-granularity approach in [64]. Taking one type of cost setting for Wpbc dataset as an example, we use Eq. (18) to generate piecewise-constant-function-form test costs and let them be random integers lying within [100,1000], and let misclassification costs be constant values, $mc_{(R,N)} = 100000$, $mc_{(N,R)} = 10000$. This type of cost setting is different from that one introduced in Section 5.1. Then we conduct the same experiments as those in Section 5.2 and Section 5.3 with this type of cost setting. A representative feature-granularity selection result is listed in Table 10, and the results of comparing between the multi-granularity approach and the single-granularity approach are shown in Fig. 9. It can be found that these results follow the rules obtained in Section 5.2 and 5.3. The good performance of the proposed multi-granularity feature selection approach is due to the nice algorithm design.

As mentioned above, except the approach in [64], in fact we also compare our multi-granularity feature selection approach with two other single-granularity error-based cost-sensitive feature selection approaches in [6,63]. Experiments whose procedure is the same as that discussed in this subsection are conducted on the nine datasets listed in Table 6. In particular, to facilitate the comparisons, three different values are respectively taken for the parameters of the two compared single-granularity approaches. It is known from the experimental results that the proposed approach performs much better than the two existing approaches on total cost minimization, and its computational efficiency is comparable to that of the two ones. In order to save the space, we only display the results of Heart dataset, which are depicted in Fig. 10. From the average difference ratios shown in the figure, it is known that most of average total costs obtained by our multi-granularity algorithm are rather less than those obtained by the two existing single-granularity algorithms. In addition, our algorithm runs faster than the algorithm in [63], but it runs more slowly than that in [6]. The reason why our multi-granularity algorithm runs more slowly than some single-granularity algorithms is that, the multi-granularity



(a) (b)

Fig. 9. Comparison results on Wpbc dataset with another type of cost setting: (a) average difference ratio, (b) average run-time.

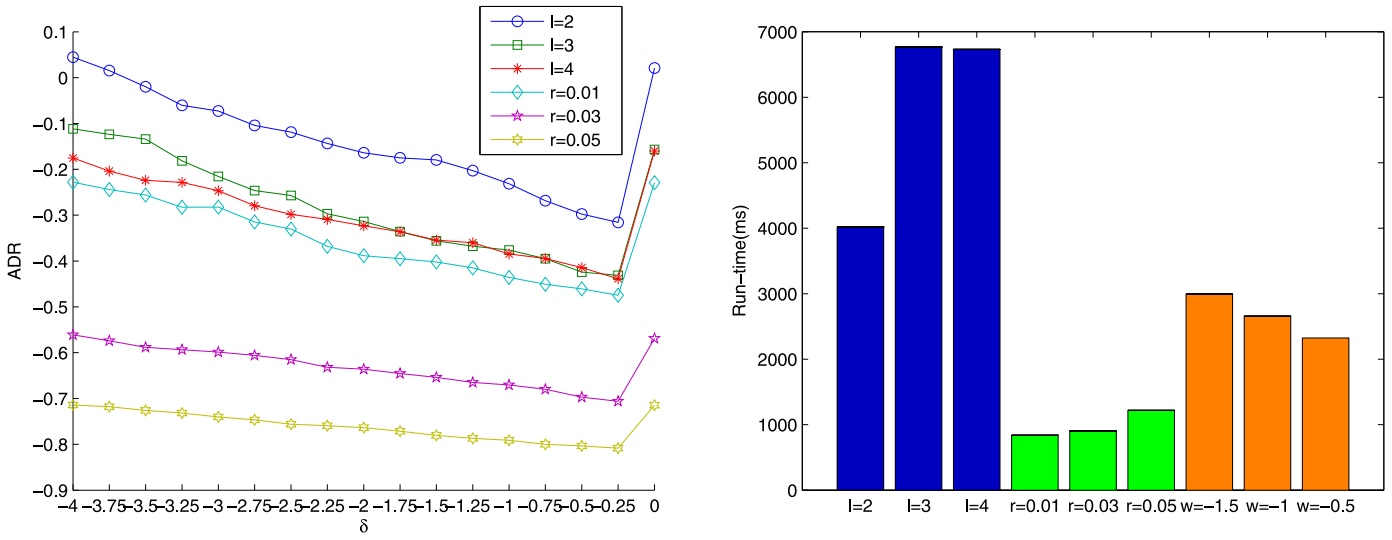


Fig. 10. Comparison results on Heart dataset: (a) average difference ratios, (b) average run-time, where "l = *" denotes the single-granularity feature selection algorithm in [63] with the level of confidence l = *, "r = *" denotes the single-granularity algorithm in [6] with neighborhood radius r = *, and "w = *" denotes the proposed multi-granularity algorithm with weight δ = *.

algorithm selects not only features but also their respective feature-value granularities, while single-granularity algorithms select only features, or features plus a single feature-value granularity for all selected features. In general, the proposed multi-granularity feature selection approach performs well not only on minimizing the total cost but also on the computational efficiency. In these two aspects, the results of our approach are better than or comparable to those of state-of-art single-granularity error-based cost-sensitive feature selection approaches. Meanwhile our approach can obtain multiple feature-value granularities for selected features, while the single-granularity approaches cannot. Hence, our multi-granularity approach is more effective and versatile than the single-granularity approaches. In addition, $[-1.5, -0.25]$ is a rational value range for the weight δ, and smaller values are preferred for the confidence level pair (p_a^0, s_a).

6. Conclusion and further work

In recent years, some researchers have studied cost-sensitive feature selection based on rough set theory. However, most of existing approaches are essentially single-granularity, which are not feasible in some real applications. In this paper, multi-granularity ideas have been introduced into the area of cost-sensitive feature selection to study the measurement errors and variable costs based multi-granularity feature selection problem. We first built a confidence-level-vector-based neighborhood rough set model, in which feature set and feature-value granularity vector are associated effectively. Fundamental notions and properties are discussed thoroughly in this new model. Then, several kinds of variable cost settings were constructed according to reality, in which the relationship among feature-value granularities, test costs and misclassification costs was considered; and the computation method of average total cost was developed. Finally, we proposed the multi-granularity feature selection approach which mainly contains a heuristic feature-granularity selection algorithm and a relevant competition strategy. In the feature-granularity selection algorithm, features and their respective feature-value granularities are selected simultaneously; and the competition strategy can further improve the performance of the algorithm when necessary. Experimental results have indicated that a desirable trade-off among feature dimension reduction, feature-value granularity selection and total cost minimization can be obtained by using the proposed approach. The multi-granularity feature selection approach is more effective and versatile than the state-of-art single-granularity error-based cost-sensitive feature selection approaches.

In summary, this work provides a new insight into the research concerning multi-granularity ideas, cost-sensitive learning, and feature selection problem. In the future, we will study the extended models to cope with more complex data, such as composite information systems or decision systems [20,58]. Our another future work is to design parallel or incremental algorithms to deal with the multi-granularity feature selection of large or even super-large data.

Acknowledgments

We are grateful to the anonymous reviewers for their valuable comments and suggestions. This work is supported in part by the National Natural Science Foundation of China under Grant Nos. 61672332 and 61603173, the Natural Science Foundation of Fujian Province, China under Grant Nos. 2016J01315 and 2017J01771, the Education Department of Fujian Province under Grant No. JAT160291, the Institute of Meteorological Big Data-Digital Fujian and Fujian Key Laboratory of Data Science and Statistics.

References

[1] D. Ardagna, C. Francalanci, M. Trubian, A multi-model algorithm for the cost-oriented design of internet-based systems, Inf. Sci. (Ny) 176 (21) (2006) 3105–3131.
[2] A. Bargiela, W. Pedrycz, Granular Computing: An Introduction, Kluwer Academic Publishers, Boston, 2002.
[3] C.L. Blake, C.J. Merz, UCI repository of machine learning databases, 1998, http://www.ics.uci.edu/~mlearn/mlrepository.html.
[4] M. Dash, H. Liu, Consistency-based search in feature selection, Artif. Intell. 151 (2003) 155–176.
[5] D. Dubois, H. Prade, Rough fuzzy sets and fuzzy rough sets, Int. J. Gen. Syst. 17 (2–3) (1990) 191–209.
[6] A.J. Fan, H. Zhao, W. Zhu, Test-cost-sensitive attribute reduction on heterogeneous data for adaptive neighborhood model, Soft Comput. 176 (21) (2016) 4813–4824.
[7] R.A. Fisher, On the mathematical foundations of theoretical statistics, Philosoph. Trans. R. Soc. Lond. Ser. A, Contain. Pap. Math. Phys. Character 222 (1922) 309–368.
[8] A. Gacek, Granular modelling of signals: a framework of granular computing, Inf. Sci. (Ny) 176 (21) (2013) 1–11.
[9] S. Greco, M. Inuiguchi, R. Slowinski, Dominance-based rough set approach using possibility and necessity measures, Proceedings of Rough Sets and Current Trends in Computing, volume 2475, LNCS, 2002, pp. 85–92.
[10] I. Guyon, A. Elisseeff, An introduction to variable and feature selection, J. Mach. Learn. Res. 3 (2003) 1157–1182.

[11] J.P. Herbert, J.T. Yao, Game-theoretic rough sets, Fundam. Inform. 108 (3–4) (2011) 267–286.
[12] Q.H. Hu, J.F. Liu, D.R. Yu, Mixed feature selection based on granulation and approximation, Knowl. Based Syst. 176 (21) (2008) 294–304.
[13] Q.H. Hu, D.R. Yu, J.F. Liu, C.X. Wu, Neighborhood rough set based heterogeneous feature subset selection, Inf. Sci. (Ny) 178 (18) (2008) 3577–3594.
[14] A. Jain, D. Zongker, Feature selection: evaluation, application, and small sample performance, IEEE Trans. Pattern Anal. Mach. Intell. 19 (2) (1979) 153–158.
[15] X.Y. Jia, W.H. Liao, Z.M. Tang, L. Shang, Minimum cost attribute reduction in decision-theoretic rough set models, Inf. Sci. (Ny) 219 (2013) 151–167.
[16] Y. Jiang, Y. Yu, Minimal attribute reduction with rough set based on compactness discernibility information tree, Soft Comput. 20 (6) (2016) 2233–2243.
[17] W. Jin, A.K. Tung, J.W. Han, W. Wang, Ranking outliers using symmetric neighborhood relationship, Proceedings of the Tenth Pacific-Asia Conference on Knowledge Discovery and Data Mining, (2006), pp. 577–593.
[18] H.R. Ju, H.X. Li, X.B. Yang, X.Z. Zhou, B. Huang, Cost-sensitive rough set: a multi-granulation approach, Knowl. Based Syst. 123 (2017) 137–153.
[19] K. Kira, L.A. Rendell, The feature selection problem: traditional methods and a new algorithm, Proceedings of the Tenth National Conference on Artificial Intelligence, AAAI, 1992, pp. 129–134.
[20] S.Y. Li, T.R. Li, J. Hu, Update of approximations in composite information systems, Knowl. Based Syst. 83 (2015) 138–148.
[21] X.J. Li, H. Zhao, W. Zhu, An exponent weighted algorithm for minimal cost feature selection, Int. J. Mach. Learn. Cybern. 7 (5) (2016) 689–698.
[22] J.Y. Liang, F. Wang, C.Y. Dang, Y.H. Qian, A group incremental approach to feature selection applying rough set technique, IEEE Trans. Knowl. Data Eng. 26 (2) (2014) 294–308.
[23] S.J. Liao, Optimal feature subset selection with errors and variable costs, J. Minnan Normal Univ. 176 (21) (2015) 21–30.
[24] S.J. Liao, Q.X. Zhu, R. Liang, On the properties and applications of inconsistent neighborhood in neighborhood rough set models, IEICE Trans. Inf. Syst. E101-D (3) (2018) 709–718.
[25] S.J. Liao, Q.X. Zhu, F. Min, Cost-sensitive attribute reduction in decision-theoretic rough set models, Math. Probl. Eng. 2014 (2014) 1–9.
[26] G.P. Lin, J.Y. Liang, Y.H. Qian, Multigranulation rough sets: from partition to covering, Inf. Sci. (Ny) 241 (2013) 101–118.
[27] G.P. Lin, Y.H. Qian, J.J. Li, Nmgrs: neighborhood-based multigranulation rough sets, Int. J. Approxim. Reason. 53 (2012) 1080–1093.
[28] T.Y. Lin, Neighborhood systems and approximation in database and knowledge base systems, Proceedings of the Fourth International Symposium on Methodologies of Intelligent Systems, (1989), pp. 75–86. ACM, October
[29] T.Y. Lin, Granular computing - structures, representations, and applications, Lecture Notes in Artificial Intelligence, volume 2639, (2003), pp. 16–24.
[30] Z.Q. Meng, Z.Z. Shi, A fast approach to attribute reduction in incomplete decision systems with tolerance relation-based rough sets, Inf. Sci. (Ny) 179 (16) (2009) 2774–2793.
[31] F. Min, H.P. He, Y.H. Qian, W. Zhu, Test-cost-sensitive attribute reduction, Inf. Sci. (Ny) 181 (2011) 4928–4942.
[32] F. Min, W. Zhu, Attribute reduction of data with error ranges and test costs, Inf. Sci. (Ny) 211 (2012) 48–67.
[33] J. Neyman, Outline of a theory of statistical estimation based on the classical theory of probability, Philosoph. Trans. R. Soc. Lond. Ser. A Math. Phys. Sci. 176 (21) (1937) 333–380.
[34] I.K. Park, G.S. Choi, Rough set approach for clustering categorical data using information-theoretic dependency measure, Inf. Syst. 48 (2015) 289–295.
[35] Z. Pawlak, Rough sets, Int. J. Comput. Inf. Sci. 11 (1982) 341–356.
[36] Z. Pawlak, A. Skowron, Rudiments of rough sets, Inf. Sci. (Ny) 177 (1) (2007) 3–27.
[37] W. Pedrycz, Allocation of information granularity in optimization and decision-making models: towards building the foundations of granular computing, Eur. J. Oper. Res. 232 (1) (2014) 137–145.
[38] Y.H. Qian, J.Y. Liang, C.Y. Dang, Incomplete multigranulation rough set, IEEE Trans. Syst. Man Cybernet. Part A 40 (2) (2010) 420–431.
[39] Y.H. Qian, J.Y. Liang, W. Pedrycz, C.Y. Dang, Positive approximation: an accelerator for attribute reduction in rough set theory, Artif. Intell. 174 (9–10) (2010) 597–618.
[40] Y.H. Qian, J.Y. Liang, Y.Y. Yao, C.Y. Dang, Mgrs: a multi-granulation rough set, Inf. Sci. (Ny) 180 (2010) 949–970.
[41] Y.H. Qian, H. Zhang, Y.L. Sang, J.Y. Liang, Multigranulation decision-theoretic rough sets, Int. J. Approxim. Reason. 55 (2014) 225–237.
[42] M.S. Raza, U. Qamar, An incremental dependency calculation technique for feature selection using rough sets, Inf. Sci. 343–344 (2016) 41–65.
[43] C. Scott, M. Davenport, Regression level set estimation via cost-sensitive classification, IEEE Trans. Signal Process. 55 (6) (2007) 2752–2757.
[44] W.H. Shu, H. Shen, Multi-criteria feature selection on cost-sensitive data with missing values, Pattern Recognit. 51 (2016) 268–280.
[45] A.H. Tan, J.J. Li, Y.J. Lin, G.P. Lin, Matrix-based set approximations and reductions in covering decision information systems, Int. J. Approxim. Reason. 59 (2015) 68–80.
[46] A.H. Tan, W.Z. Wu, Y.Z. Tao, A set-cover-based approach for the test-cost-sensitive attribute reduction problem, Soft Comput. 21 (20) (2017) 6159–6173.
[47] E.C. Tsang, D.G. Chen, D.S. Yeung, Approximations and reducts with covering generalized rough sets, Comput. Math.Appl. 56 (1) (2008) 279–289.
[48] P.D. Turney, Types of cost in inductive concept learning, Proceedings of the Seventeenth Workshop on Cost-Sensitive Learning, ICML, (2000), pp. 1–7.
[49] C.Z. Wang, M.W. Shao, Q. He, Y.H. Qian, et al., Feature subset selection based on fuzzy neighborhood rough sets, Knowl. Based Syst. (2016), http://dx.doi.org/10.1016/j.knosys.2016.08.009.
[50] G.Y. Wang, X.A. Ma, H. Yu, Monotonic uncertainty measures for attribute reduction in probabilistic rough set model, Int. J. Approxim. Reason. 59 (2015) 41–67.
[51] W.Z. Wu, Y. Leung, Theory and applications of granular labelled partitions in multi-scale decision tables, Inf. Sci. (Ny) 181 (2011) 3878–3897.
[52] W.Z. Wu, J.S. Mi, W.X. Zhang, Generalized fuzzy rough sets, Inf. Sci. (Ny) 151 (2003) 263–282.
[53] W.H. Xu, Q.R. Wang, X.T. Zhang, Multi-granulation fuzzy rough sets in a fuzzy tolerance approximation space, Int. J. Fuzzy Syst. 13 (4) (2011) 246–259.
[54] X.B. Yang, Y.S. Qi, X.N. Song, J.Y. Yang, Test cost sensitive multigranulation rough set: model and minimal cost selection, Inf. Sci. (Ny) 250 (2013) 184–199.
[55] Y.Y. Yao, A partition model of granular computing, Lect. Notes Comput. Sci. 3100 (2004) 232–253.
[56] Y.Y. Yao, Y.H. She, Rough set models in multigranulation spaces, Inf. Sci. (Ny) 327 (2016) 40–56.
[57] Y.Y. Yao, Y. Zhao, J. Wang, On reduct construction algorithms, Proceedings of Rough Set and Knowledge Technology, 4062 LNAI, 2006, pp. 297–304.
[58] J.B. Zhang, T.R. Li, H.M. Chen, Composite rough sets for dynamic data mining, Inf. Sci. (Ny) 257 (2014) 81–100.
[59] Q.H. Zhang, K. Xu, G.Y. Wang, Fuzzy equivalence relation and its multigranulation spaces, Inf. Sci. (Ny) 44–57 (2016) 346–347.
[60] S.C. Zhang, Cost-sensitive classification with respect to waiting cost, Knowl. Based Syst. 23 (5) (2010) 369–378.
[61] Y. Zhang, D.W. Gong, J. Cheng, Multi-objective particle swarm optimization approach for cost-based feature selection in classification, IEEE/ACM Trans. Comput. Biol. Bioinf. 14 (1) (2017) 64–75.
[62] H. Zhao, F. Min, W. Zhu, Cost-sensitive feature selection of numeric data with measurement errors, J. Appl. Math. 2013 (2013) 1–13.
[63] H. Zhao, P. Wang, Q.H. Hu, Cost-sensitive feature selection based on adaptive neighborhood granularity with multi-level confidence, Inf. Sci. (Ny) 366 (2016) 134–149.
[64] H. Zhao, W. Zhu, Optimal cost-sensitive granularization based on rough sets for variable costs, Knowl. Based Syst. 65 (2014) 72–82.
[65] N. Zhong, J.Z. Dong, S. Ohsuga, Using rough sets with heuristics to feature selection, J. Intell. Inf. Syst. 176 (21) (2001) 199–214.
[66] Q.F. Zhou, H. Zhou, T. Li, Cost-sensitive feature selection using random forest: selecting low-cost subsets of informative features, Knowl. Based Syst. 95 (2016) 1–11.
[67] Y.H. Zhou, Z.H. Zhou, Large margin distribution learning with cost interval and unlabeled data, IEEE Trans. Knowl. Data Eng. 28 (7) (2016) 1749–1763.
[68] W. Ziarko, Variable precision rough set model, J. Comput. Syst. Sci. 46 (1) (1993) 39–59.