Applied Intelligence (2024) 54:7099–7117
https://doi.org/10.1007/s10489-024-05533-8

WalkNAR: A neighborhood rough sets-based attribute reduction approach using random walk

Haibo Li[1,3] · Wuyang Xiong[1] · Yanbin Li[1] · Xiaojun Xie[1,2]

Accepted: 17 May 2024 / Published online: 3 June 2024

Abstract
Neighborhood rough sets, as an effective tool for processing numerical data, is widely used in many fields, such as data mining, machine learning and decision-making system. However, most of the existing neighborhood rough set-based attribute reduction algorithms have low efficiency. To address the limitation, this paper has proposed an efficient positive region search algorithm based on multiple hash buckets and multiple granularity mechanisms. This algorithm achieves a more accurate neighborhood extent by superimposing the effects of multiple hash buckets, and accelerates positive region searching through the idea of multiple granularity. In addition, on the foundation the positive region search algorithm, we improved the existing algorithm and proposed an attribute reduction algorithm based on multi-hash bucket and multi-granularity. To further remove the redundant attributes, the two algorithms mentioned above are applied into a novel attribute reduction approach based on random walk. Experiments conducted on UCI datasets show that our attribute reduction algorithm has high efficiency. Moreover, attribute reduction approach we proposed can further compress the reduced attribute set, and the results maintain similar or even better classification accuracy.

Keywords Neighborhood rough sets · Attribute reduction · Random walk · Hash bucket

1 Introduction

Rough sets, presented in Pawlak's theory [15, 16], is used to analyze information with inaccuracy, inconsistency, and incompleteness. The main idea of this theory is using equivalence relations for classification and approximating uncertain knowledge with upper and lower approximation methods. However, traditional neighborhood rough set has many shortcomings. Therefore, many related concepts have emerged, such as neighborhood rough set [3, 37], multi-granulation rough set [4, 24], soft rough set [2, 20], and fuzzy rough set [22, 35, 36].

Among all the related concepts, the neighborhood rough set has been proved highly effective in the application of attribute reduction. Hu [8] proposed the neighborhood rough set based on the measurement of attribute distance between samples, which replaces the equivalence relation between samples with the determination of sample neighborhoods. Through this method, neighborhood rough set can process numerical data. To generate high quality attribute sets, forward attribute selection strategy was used for attribute reduction in Qing [19]. However, the computational process of neighborhood rough set has a high degree of complexity, so many works focus on speeding up the process of obtaining reduced attribute sets. Liu [38] proposed a fast hash attribute reduct Algorithm FHARA based on hash bucket partitioning. Wang [30] proposed an improved algorithm based on Liu's research, proposing a fast reduction algorithm EasiFFRA based on symmetry and decision filtering. Xia [18] proposed a pre-sorting method, which provides a large amount

✉ Xiaojun Xie
xxj@njau.edu.cn

Haibo Li
lihaibo@stu.xmu.edu.cn

Wuyang Xiong
xwy020822@163.com

Yanbin Li
yanbinli@njau.edu.cn

1 College of Artificial Intelligence, Nanjing Agricultural University, 1 Weigang Road, Nanjing 210095, Jiangsu, China

2 Center for Data Science and Intelligent Computing, Nanjing Agricultural University, 1 Weigang Road, Nanjing 210095, Jiangsu, China

3 School of Informatics, Xiamen University, 4221 Xiang'an South Road, Xiamen 361102, Fujian, China

of local and overall sample information in advance by sorting. Srirekha [21] introduced the concept of object ranking and implemented four attribute reduction methods based on object sorting by defining dual operators. Zou [41] proposed a rough set attribute reduction algorithm based on conditional entropy which is used for fatigue decision system of the aluminum welded joints. Chu [5] introduced Best-Worst method and constructed the neighborhood rough set attribute reduction model as well as took attribute correlation into consideration. In the above work, the geometric distribution information of the attribute set can be obtained additionally in the process of calculation, and the prior knowledge can be used to reduce unnecessary operations in the calculation.

In the actual process, it is necessary to consider the dynamic changes of the dataset and the need for multi-dimensional measurements. The multi-granularity mechanism is a good way to take into account the impact of different situations. By fusing information from different situations, it is possible to further accelerate the operation of the algorithm or obtain a set of attributes with excellent properties. Liu [13] introduced multi-granularity mechanisms and achieved result fusion through multi-granularity constraints to achieve attribute reduction. Li [42] also used multi-granularity mechanisms to adjust the degree of mutual influence between two different neighborhood radii. The algorithm of GBNRS proposed by Xia [31] adaptively generates different neighborhoods for each sample, which is more flexible and variable than traditional methods. Dai [40] deleted objects in the dataset to change the granularity. Su [23] proposed an incremental update mechanism for the positive region and right neighborhood that is suitable for dynamic datasets. Yang [33] used a matrix-based mechanism to solve the problem of multi-granularity. Li [12] accelerated the reduction process by constructing a multi-granularity reduction result with multiple neighborhood radii. Tallon [25] introduced positive approximation mechanism and proposed an acceleration strategy for neighborhood based multi-granularity attribute reduction. Zhao [39] noticed that the multi-granularity model can be used in continuous parameters and tried to accelerate the process of multi-granularity attribute reduction. Jiang [9] designed an accelerator for varying neighborhood radius, which realizes high-speed computing under a multi-granularity mechanism. Yang [32] tried to fuse three-way decision, granular computing and multi-granularity approach and propose the multilevel neighborhood granular structures which has very good performance. In Liu et al. [14], three-way decision was introduced into multi-granularity attribute reduction and data-aware multi-granularity structure is automatically induced from self-contained distance space to define a novel feature evaluation criterion. However, the multi-granularity mechanism can improve the quality of the reduced attribute set, but it remains to be proved whether it can reduce the redundant components in the results.

Most method explored relatively less in generating a relative attribute reduct. To address this issue, most studies have designed algorithms to obtain high quality attribute set. Kang [11] proposed an inconsistent gray decision system attribute reduction based on variable precision gray multi-granularity rough set. Guo [7] proposed a double fuzzy consistency measure to simultaneously mine the valuable information in upper and lower approximations from the absolute quantitative perspective and in the boundary from the relative quantitative perspective. Abdolrazzagh-Nezhad [1] proposed a continuous optimization algorithm using cultural algorithm with a dual inheritance system to generate new individuals and planning a novel heuristic to discrete population and belief spaces. Peng [17] noticed that robust variable granular-ball model has shown good effectiveness in label noise environment and used adaptive granular-balls of different sizes to perform efficient attribute reduction. In Ju et al. [10], nearest mutual neighbor-based personalized information granularity is introduced in attribute reduction and Attribute reduction is transformed into an optimization issue. Dynamic attribute reduction was proposed in Yang et al. [34] to cope with changes and it incremental updates attributes to generate attribute sets. Gao [6] tried to remove irrelevant examples to simultaneously exclude redundant attributes. However, most methods aim to directly obtain high-quality solutions, but in reality, this is not always possible. We want to obtain better solutions through iterative processes using randomization. This approach is more stable and also transforms attribute reduction into an optimization problem. Self-information can describe the uncertainty of a signal very well, which is consistent with the idea of fuzzy approximation, so part of the works [26, 28] combines it with attribute selection and achieves good results. In addition, some works [27, 29] has shown that fuzzy set can be used to describe the fuzzy relationships and uncertainties between attributes so as to consider the association between attributes more comprehensively in attribute reduction.

In this paper, we try to synthesize the interaction between multiple hash buckets to further effectively reduce the positive region search range of a single sample. Besides, we found that all samples in the same hash bucket have the same search range, which suggests that we can perform positive region search on all samples in a bucket at the same time, which is a very novel multi-granularity mechanism. Based on the above two mechanisms, we propose a multi-hash bucket and multi-granularity positive region search algorithm (MMPRSA) to accelerate positive region search and apply it to random walk. Finally, an efficient attribute reduction algorithm (WalkNAR) is obtained. WalkNAR can perform multiple iterations of the

existing set of attributes, trying to remove individual attribute or replace two of them with a new one, so as to obtain a more reduced result. The diagram of the algorithms we designed is shown in Fig. 1. The main contributions of the paper are the following:

1. A method based on multiple hash bucket has been proposed and this method can reduce the range of neighborhoods of a single sample by performing intersection operations between different independent hash buckets so as to avoid a large number of computations of distance between samples.
2. An efficient method, based on multiple granularity is applied and it means some samples can be omitted when processing other samples. This method considers that the search space generated by samples with the same hash bucket index is consistent, so as to transform the positive region search problem of a single sample into a positive region search problem of all samples in the bucket.
3. A forward search strategy, gradually adding attribute sets that maximize the positive region using a greedy strategy, is used for a multi-hash bucket and multi-granularity attribution reduction to obtaining the relative reduction attribute set.
4. Finally, a local search method to explore local solutions through random walk with different strategies is conduct to obtain higher-quality attribute sets. The random walk strategy involves removing a single attribute and replacing two of them with a new one.

The rest of the paper is organized as follow: In Sect. 2, basic concepts about neighborhood rough set are introduced. Section 3 exposes the multi-hash bucket and multi-granularity positive region search algorithm for attribute reduction. The neighborhood rough set-based attribute reduction approach using random walk we proposed is introduced in Sect. 4. Experimental results on UCI data sets are presented in Sect. 5. Some conclusions are drawn in Sect. 6.

2 Preliminaries

We first recall some concepts and results that will be used throughout the paper. The concepts mentioned can all be found within the reference [8].

A *decision table* can be expressed as $S = (U, C, D, \mathcal{V}, f)$, where $U = \{x_1, x_2, \ldots, x_{|U|}\}$ is a finite nonempty set of objects, called *universe*, $C = \{a_1, a_2, \ldots, a_{|C|}\}$ is a finite nonempty set of *conditional attributes*, D is the *decision attributes*, and $f : U \times (C \cup D) \to \mathcal{V}$ is an *information function*, where $\mathcal{V}$ is *domain* of attributes $C \cup D$. Moreover, for every $a \in C \cup D$, we have $\mathcal{V}_a = \{f(x, a) \mid x \in U\}$. An example of a simple decision table is given in Table 1, where $U = \{x_1, x_2, \ldots, x_{11}\}$, $C = \{a_1, a_2\}$, and $D = \{d\}$.

Definition 1 As defined in Hu et al. [8], given a m-dimensinal real space Ω, a neighborhood radius δ, $\forall x_i \in \Omega$ and $B \subseteq C$, the neighborhood $\delta_B(x_i)$ of x_i in feature space B is defined as

$$\delta_B(x_i) = \{x_j \mid x_j \in U, \Delta^B(x_i, x_j) \leqslant \delta\}$$

where Δ is a mapping $\Delta : \Omega \to \Omega$, Δ satisfies:

1. Non-negativity:$\Delta(x_1, x_2) \geqslant 0$, $\Delta(x_1, x_2) = 0$ if and only if $x_1 = x_2$;

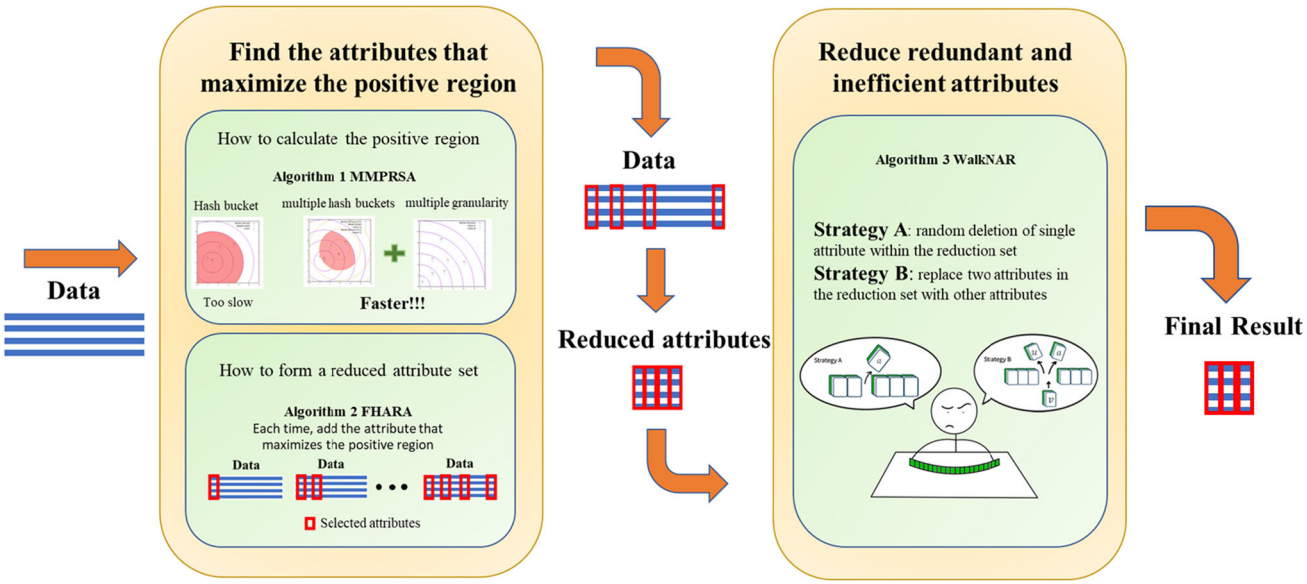


Fig. 1 The diagram of the algorithms

2. Symmetry:$\Delta(x_1, x_2) = \Delta(x_2, x_1)$;
3. Triangle inequality:$\Delta(x_1, x_3) \leqslant \Delta(x_1, x_2) + \Delta(x_2, x_3)$.

Δ is usually expressed by the L_2-norm:

$$\Delta_B(x_i, x_j) = (\sum_{k=1}^{s} \mid f(x_i, a_k) - f(x_j, a_k) \mid^2)^{\frac{1}{2}}$$

Example 1 Considering the decision system in Table 1, assuming L_2-norm is used for the distance function $f(x_i, x_j)$, the distance among samples can be easily compute:

$$f(x_1, x_2) = \mathbf{0.1077}, f(x_1, x_3) = \mathbf{0.5099}, f(x_1, x_4) = \mathbf{0.3000}$$
$$f(x_2, x_3) = \mathbf{0.6708}, f(x_2, x_4) = \mathbf{0.5657}, f(x_3, x_4) = \mathbf{0.2236}$$

Assuming $\delta = 0.3$ then the neighborhood set of x_i are:

$$\delta(x_1) = \{x_1, x_2, x_4\}, \delta(x_2) = \{x_1, x_2\}, \delta(x_3) = \{x_3, x_4\}, \delta(x_4) = \{x_1, x_3, x_4\}$$

Definition 2 As defined in Hu et al. [8], given a universe U, a neighborhood relation N over U and a subset X of U, the mathematical expressions of lower and upper approximations of X in $\langle U, N\rangle$ are as fellows:

$$\underline{N}X = \{x_i \mid \delta(x_i) \subseteq X, x_i \in U\}$$

$$\overline{N}X = \{x_i \mid \delta(x_i) \cap X \neq \emptyset, x_i \in U\}$$

Further more, $X_1, X_2, \ldots, X_s$ are the division by D to U, then the lower approximation and upper approximation of conditional attribute subset B relative to D are:

$$\underline{N_B}D = \bigcup_{i=1}^{s} \underline{N_B}X_i$$

$$\overline{N_B}D = \bigcup_{i=1}^{s} \overline{N_B}X_i$$

$\underline{N_B}D$ is also known as the positive region $POS_B(D)$.

Example 2 Given a subset $X = \{x_1, x_2\}$ of U, a conditional attribute subset $B = C$, then the lower approximation and upper approximation of B relative to D can be given as follow:

$$\underline{N_B}D = \{x_2\}$$

$$\overline{N_B}D = \{x_1, x_2, x_4\}$$

We can easily know from the example that the following rule is true:

$$\underline{N_B}D \subseteq X \subseteq \overline{N_B}D$$

$\underline{N_B}D$ includes all the samples whose neighborhoods with same class intersect with X,and $\overline{N_B}D$ represents the samples whose neighborhoods with same class are included in X.

Table 1 Demo data

U	a_1	a_2	d
x_1	0.13	0.21	0
x_2	0.53	0.11	0
x_3	0.23	0.71	1
x_4	0.13	0.51	1

3 MMPRSA

In this section we have introduced two methods used in our positive region search algorithm to avoid redundant operations.

3.1 Multi-hash bucket positive region search algorithm

Based on the characteristics of the spherical neighborhood of the neighborhood rough set, Liu et al. proposed an attribute reduction algorithm FHARA based on hash bucket division, which generated hash buckets according to the distance between the randomly selected sample center, so as to limit the neighborhood range of each sample to adjacent buckets and reduce the search range of the positive region.

Definition 3 As shown in Fig. 2, given the decision table $DT = \langle U, C \cup D, \mathcal{V}, f\rangle$ assuming that the δ is the neighborhood radius, a point X called sample center is randomly selected in the sample space, then the above method maps samples to the mutually exclusive hash bucket $B_0, B_1, \cdots, B_c$ according to δ and the distance from the samples to X. The possible range of the neighborhood of the corresponding sample can be narrowed to B_q and adjacent buckets B_{q-1}, B_{q+1}.

Since the sample is divided according to the distance from the sample center, there are:

$$\begin{cases} B_q = B(x_i) = \{x_j \mid \lceil \Delta\left(x_j, X\right)/\delta \rceil = \lceil \Delta(x_i, X)/\delta \rceil\} \\ q = \lceil \Delta(x_i, X)/\delta \rceil \end{cases} \quad (1)$$

It is easy to draw the following conclusions:

$$\forall i, j \in [0, b], i \neq j, B_i \cap B_j = \emptyset \quad (2)$$

$$\bigcup_{i=1}^{b} B_i = U \quad (3)$$

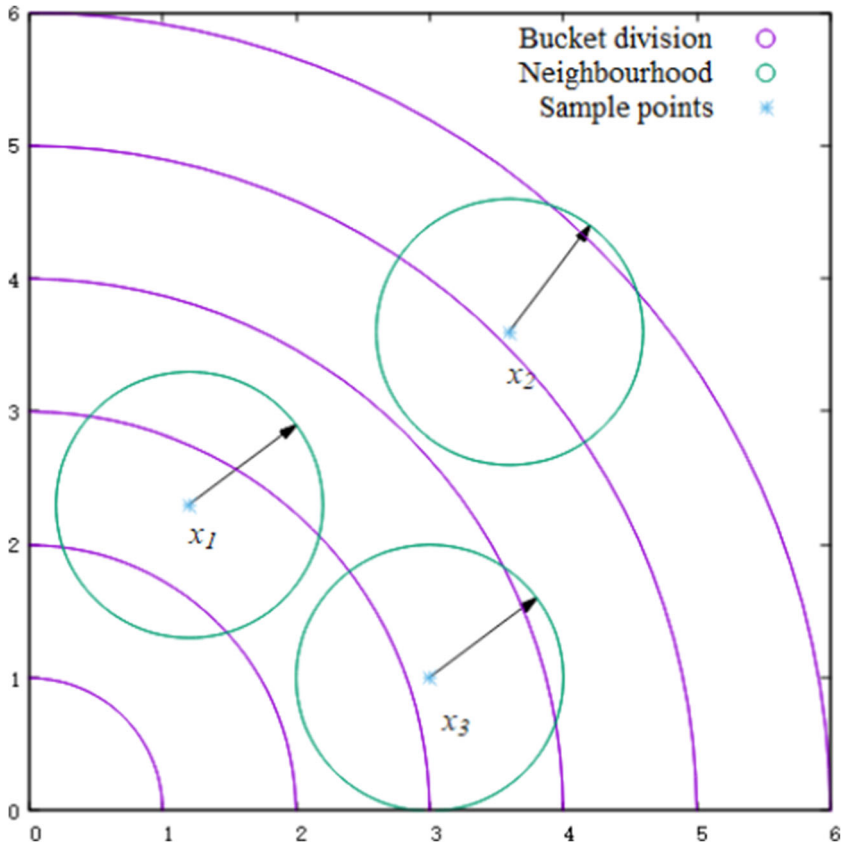


Fig. 2 Hash bucket division

Theorem 1 *As delineated in the reference [38], given a decision table* $DT = \langle U, C \cup D, \mathcal{V}, f \rangle$*, assuming that the* δ *is the neighborhood radius,* $B_0 \ldots B_k$ *are the buckets, then* $\forall x_i \in B_q (q = 1, 2, \ldots, k-1)$ *the neighborhoods of* x_i *are all included in* $B_q \cup B_{q-1} \cup B_{q+1}$.

Proof According to Eq. (1), $x_i \in B_q \rightarrow (q-1)\theta < \Delta(x_i, X) \leqslant q\theta$, so $\forall x_j \notin B_q \cup B_{q-1} \cup B_{q+1}$, we have $\Delta(x_i, X) - \Delta(x_j, X) \geq \delta$, because we know the triangle inequality, we have $\Delta(x_i, x_j) \geq \Delta(x_i, X) - \Delta(x_j, X)$, so $\Delta(x_i, x_j) \geq \delta$. □

Since there is no intersection between adjacent buckets, there are no redundant samples when calculating the positive region, assuming that the total number of samples $|U|$ is evenly divided into h buckets, the number of samples to be traversed in a single search is reduced from $|U|$ to $3|U|/h$. Let the number of selected conditional attributes is m, then the computational times of positive region of the entire sample set is $O(m|U|^2/h)$, so this method can effectively improve the efficiency of calculating positive region.

Liu thought that b can tend to $|U|$. However, considering that the data in the sample space is not uniformly distributed, there is an inconsistent degree of aggregation, and the neighborhood radius δ has restriction on the number of buckets generated. The computational times of the positive region cannot be approached to $O(m|U|)$ in the actual operation.

As shown in Eq. (1), using hash buckets to divide the sample space is essentially to map the sample to a one-dimensional space, and we only need to search for samples in adjacent buckets at this one-dimensional space, which naturally sorts the samples according to distance, so that the adjacent relationship in the one-dimensional space can directly correspond to the neighborhood range of the sample. However, mapping such a large number of samples into one-dimensional space will cause the normalized sample points to be too dense, although it can still have an acceleration effect, but the number of samples in each bucket will surge, resulting in poor algorithm efficiency.

The selection of the sample center has no effect on the consistency and correctness of the algorithm. Assuming the neighborhood search area of x_i is $Search(x_i)$, so for the sample point x_i, the bucket corresponding to the x_i is B_q, the corresponding neighborhood search area for x_i is:

$$Search(x_i) = B_q \cup B_{q-1} \cup B_{q+1} \tag{4}$$

There is a simple method that we can choose multiple sample centers and taking the intersection between different search ranges formed by multiple divisions, there is no need to consider the number of attributes of the samples, so this method can avoid the calculation of distance calculation caused by too many attributes.

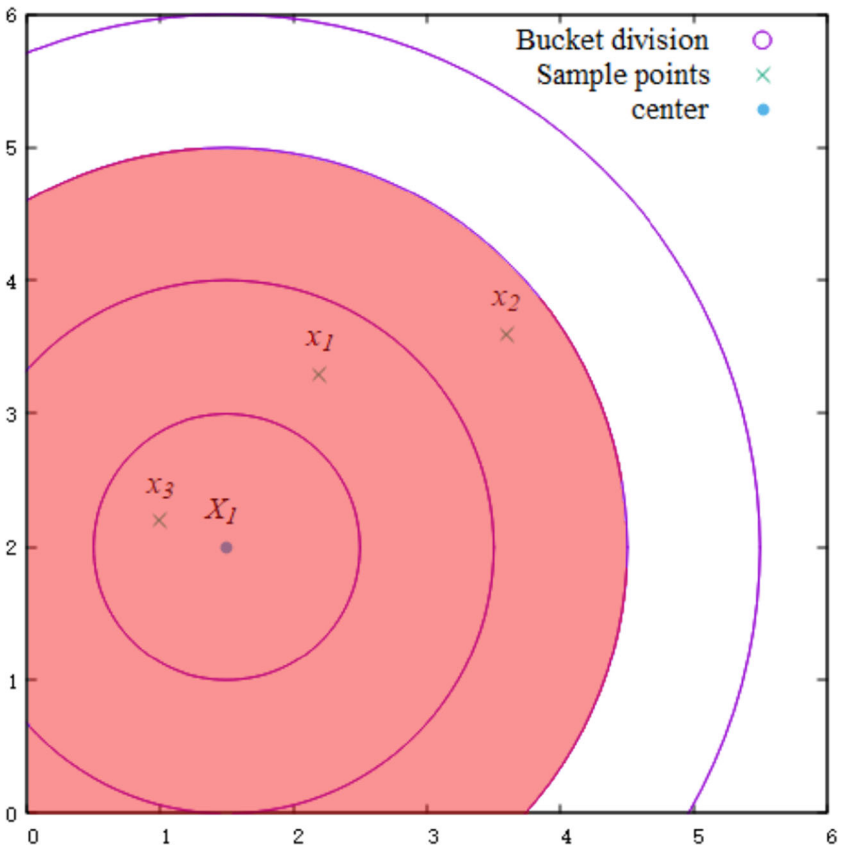


Fig. 3 Single-bucket-based search area reduction

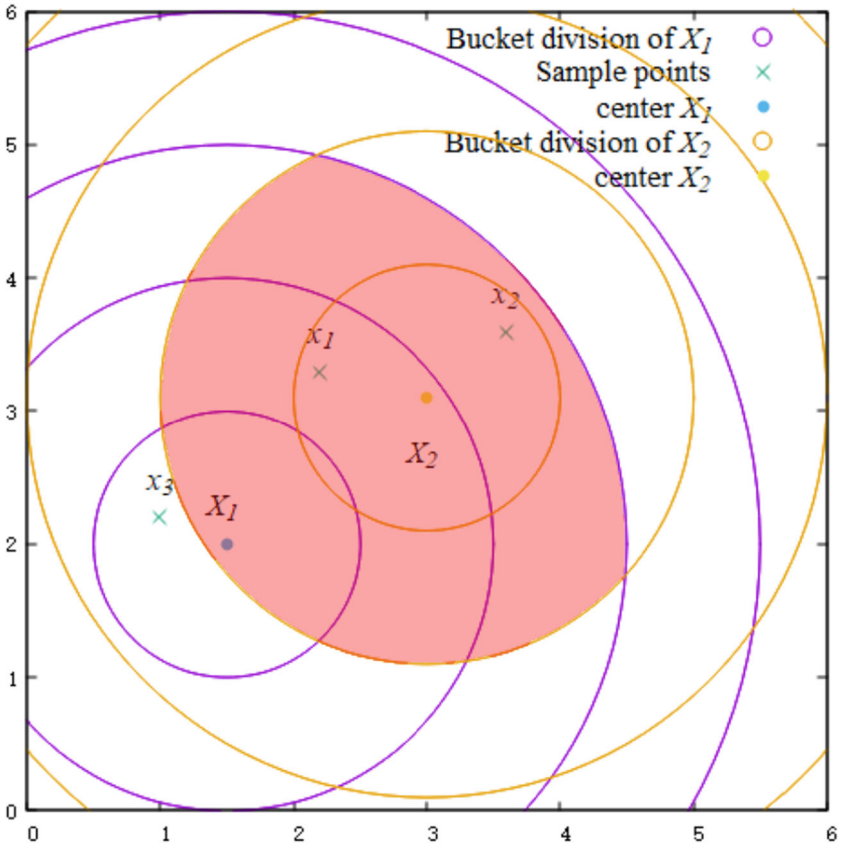


Fig. 4 Multi-bucket-based search area reduction

Example 3 As shown in Figs. 3, 4 and Table 2. In Fig. 3 we marked the area within the point search range as a shaded area, let sample center $X = \{X_1\}$, then we have $Search(x_1) = \{x_1, x_2, x_3\}$. If two sample centers are taken for calculation, let $X = \{X_1, X_2\}$ and then we have $Search(x_1) = Search(x_1, X_1) \cap Search(x_1, X_2) = \{x_1, x_2, x_3\} \cap \{x_1, x_2\} = \{x_1, x_2\}$.

Since different sample centers have no effect on the algorithm results, multiple sample centers X can be selected to form different divisions of the sample space, and then different neighborhood search ranges can be generated for the same sample, and the search range can be further narrowed by taking the intersection operation, thereby reducing the amount of operation.

Theorem 2 *Assuming that the sample centers of b samples are $X = \{X_1, \ldots, X_b\}$, and for X_k, the corresponding bucket of x_i and X_k is defined as $B_{q(i,k)k}$, then Eq. (1) and Eq. (4) can be rewritten as follows:*

$$\begin{cases} B_{q(i,k)k} = \{x_j \mid \lceil \Delta(x_j, X_k)/\delta \rceil = \lceil \Delta(x_i, X_k)/\delta \rceil\} \\ q = \lceil \Delta(x_i, X_k)/\delta \rceil \end{cases} \tag{5}$$

$$\forall x_i \in U, Search(x_i) = \bigcap_{j=1}^{b} \bigcup_{r=q(i,j)-1}^{q(i,j)+1} B_{rj} \tag{6}$$

Compared with mapping to one-dimensional space, this method can effectively cope with sample aggregation for normalized samples, and further reduce the sample range that each sample needs to search compared to single-bucket-based search area reduction, because in most cases, the same type of samples will be gathered in the adjacent area of the current calculation sample, and the possibility of different samples in the neighborhood is small. Therefore, this method also reduces the distance calculation due to different class of samples.

Proof Taking the two sample center X_1, X_2, assuming that the research sample object is x_i, for the sample center X_1, the corresponding bucket label number for x_i is B_{a1}, for the sample center X_1, the corresponding bucket label number for x_i is B_{c2}, then we have:

$$\delta(x_i) \subseteq (B_{a1} \cup B_{(a-1)1} \cup B_{(a+1)1}) \tag{7}$$

$$\delta(x_i) \subseteq (B_{c2} \cup B_{(c-1)2} \cup B_{(c+1)2}) \tag{8}$$

Table 2 Demo data

	x_1	x_2	x_3
$\lceil \Delta(x_i, X_1) \rceil$	2	3	1
$\lceil \Delta(x_i, X_2) \rceil$	3	1	1

then we can conclude that:

$$\delta(x_i) \subseteq (B_{a1} \cup B_{(a-1)1} \cup B_{(a+1)1}) \cap (B_{c2} \cup B_{(c-1)2} \cup B_{(c+1)2}) \tag{9}$$

If we choose b sample centers $X = X_1, X_2, \ldots, X_b$, then the proof for the conditions that $b = 2, 3, 4, \ldots$ is similar, finaily we have the result of Eq. (6) □

Assuming that the number of sample centers is b, for each sample center, the number of conditional attributes is m, all samples are evenly divided into h buckets, and different bucket divisions are not correlated, then the search area for a single sample is $3|U|/h^b$, and the positive region of the entire sample set is calculated with the computational times of $O(m|U|^2/h^b)$, h^b is more likely to tend to $|U|$, so this method is theoretically more efficient.

In fact, the number of sample centers b is not as high as possible, selecting too many sample centers will cause the sample search area to be further reduced, while the computational times when initializing the hash bucket is b times of that of selecting a single sample center, and the selection of b will be further tested in subsequent experiments.

3.2 Multi-granularity positive region search algorithm

Liu et al. [13] proposed that the concept of granularity have three parts: parameters, samples and attributes. Most research tended to process samples based on a single granularity, ignoring the use of multiple variable granularities in the algorithm, so there are often certain limitations in their algorithm.

In order to overcome the loss and neglect of information caused by fixed granularity, we hope to obtain more useful information by dynamically changing the granularity of the current calculation when calculating the positive region to simplify the subsequent reduction process.

According to Eq. (6), for the reduction mechanism based on the hash bucket, the search area of the neighborhood of any sample is obtained by the adjacent buckets. For some samples, the search range of the neighborhood content is completely consistent, thus we can omit the processing of those sample and instead process the sample in the same bucket consistently.

Theorem 3 *Assuming that the current research sample is x_i, set the area $E_i = \cap_{j=1}^{b} B_{q(i,j)j}$, $\forall x_j \in E_i$, $Search(x_j) = Search(x_i)$, if it can be judged that all samples in the area $Search(x_i)$ are of the same kind, then we have $\forall x_j \in E_i$, $x_j \in POS_B(D)$.*

Proof We use $D(x_i)$ to represent the class of sample x_i. According to the definition of $Search(x_i)$ and $POS_B(D)$ we can prove that the following equations are ture:

$$\forall x_i \in U, \delta(x_i) \subseteq Search(x_i) \tag{10}$$

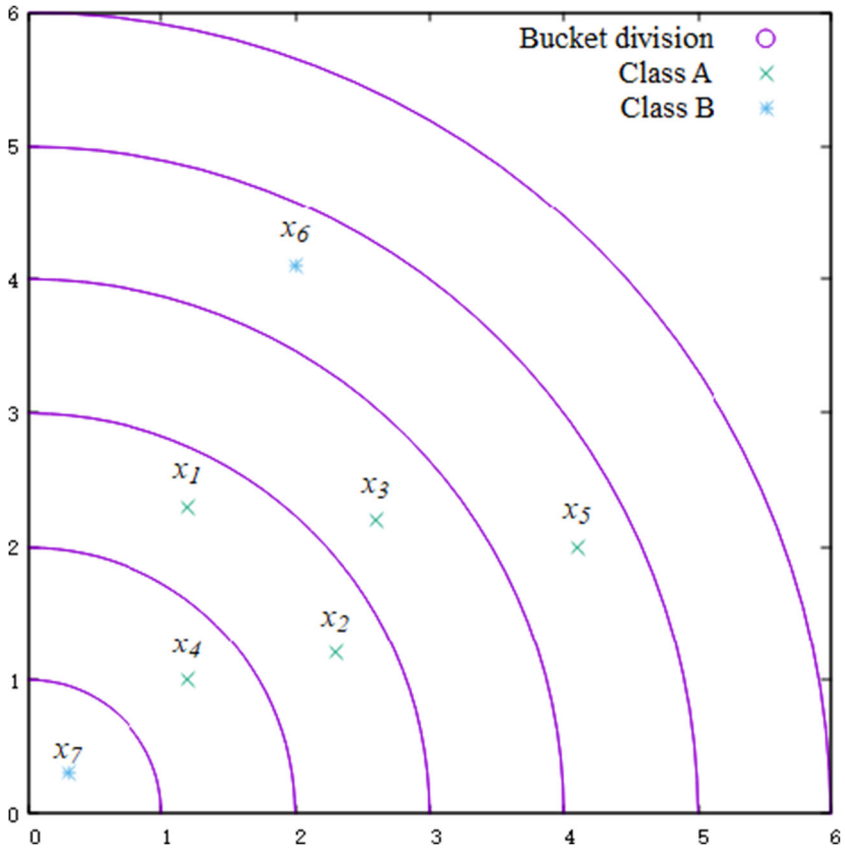


Fig. 5 Reduction mechanism based on multi-granularity

$$\forall x_j \in \delta(x_i), D(x_i) = D(x_j) \rightarrow x_i \in POS_B(D) \tag{11}$$

Assuming a set of samples $E_i = \cap_{j=1}^{b} B_{q(i,j)j}$, all samples in E_i have the same search field, because for any sample center X_k, those samples have the same distance from X_k, so if all the samples in E_i have the same class B' then we can prove that:

$$\forall x_i \in Search(x_i), D(x_i) = B' \rightarrow \forall x_i \in E_i, x_i \in POS_B(D) \tag{12}$$

□

In actual use, samples can be stored by establishing an index list, that is, setting the index list of sample x_i to $\{q^{i1}, \ldots, q^{ib}\}$, then the set of samples with the same index list with the x_i is $E_i = \cap_{j=1}^{b} B_{q(i,j)j}$.

Example 4 As shown in Fig. 5 and Table 3, we have $Search(x_1) = \{x_1, x_2, x_3, x_4\}$. At this time, all samples in the range $Search(x_1)$ are of the same kind, $\forall x_i \in Search(x_1)$, $D(x_1) = A$, so according to Eq. (12), $\forall x_i \in Search(x_1)$, $x_i \in POS(D)$, so that the subsequent calculation of the x_2 neighborhood can be ignored, also we have $Search(x_3) = \{x_1, x_2, x_3, x_5, x_6\}$, since the x_6 is a sample with different class at this time, whether the x_6 exists in the x_3 neighborhood, it is impossible to classify all the samples in $Search(x_3)$ into the positive region, it is necessary to determine whether all the samples in the bucket belong to the positive region and whether the current sample belongs to the positive region do not affect each other, so if $\Delta(x_3, x_6) > \delta$, we can still add the x_3 to the positive region.

Table 3 Demo data

	x_1	x_2	x_3	x_4	x_5	x_6	x_7
Bucket	3	3	4	2	5	5	1
$D(x)$	A	A	A	A	A	B	B

Assuming that the probability that the samples with same decision attribute in adjacent hash buckets is p, and when the appropriate conditional attribute set is selected, because the homogeneous aggregation effect of the sample is more obvious, and the samples with different decision attribute are mapped to the space that are far apart from each other, p will dynamically rise, and the corresponding part of the sample only needs to traverse the decision attributes of all the data in the bucket, without the need for distance calculation, in this case, the complexity of the computational times of this part is reduced to $O(|U|)$, this is an exciting conclusion, and the huge amount of computation caused by conditional attribute distance calculations is completely omitted from the positive

region calculation for this part of the sample, which greatly improves the efficiency of the algorithm.

3.3 MMPRSA

Algorithm 1 Multi-hash bucket and Multi-granularity Positive Region Search Algorithm(MMPRSA)

Input: U, C, D, δ, P, R;
Output: $F = \{F_1, F_2, \cdots, F_{|U|}\}$;
find $Search(x_i)$ of $x_i \in U$;
For x_i, if $NonCompute[x_i] \leftarrow 1$, it can be determined whether the sample is in the positive region and no check is needed.;
Initialize $NonCompute$ with all 0s;
for all $x_i \in U$ *and* $NonCompute[x_i] \neq 1$ **do**
  # flag indicates whether the current sample belongs to the positive region
  $flag \leftarrow 0$;
  # Posflag indicates whether all samples in the current hash bucket belong to the positive region
  $PosFlag \leftarrow 1$;
  for all $x_j \in Search(x_i)$ **do**
    if $D(x_i) \neq D(x_j)$ **then**
      $PosFlag \leftarrow 0$;
      if $\Delta(x_i, x_j) \leq \delta$ **then**
        $flag \leftarrow 1$;
        $\#x_j$ is not in positive region;
        $NonCompute[x_j] \leftarrow 1$;
        break;
      end if
    end if
  end for
  if $PosFlag = 1$ **then**
    for all $x_l \in E_i$ **do**
      # x_l is in positive region due to the multi-granularity mechanism;
      $F_l = 1$;
      $NonCompute[x_l] \leftarrow 1$;
    end for
  end if
  if $flag = 1$ **then**;
    $F_i = 1$;
  end if
end for

Combined with the multi-hash bucket and multi-granularity mechanism, the traditional positive region search algorithm is improved, and the pseudocode representation is shown in Algorithm 1. Our algorithm consists of two parts, the first part needs to map all x_i to the corresponding $Search(x_i)$, Since b sample centers are selected, the time complexity of this part of the calculation is $O(b|U|)$. In the second part of our algorithm, all sample points needed to be matched to the samples in $Search(x_i)$ one by one, but we use a multi-hash bucket mechanism, which makes the size of $Search(x_i)$ as $O(|U|/h^b)$, so the time complexity of this part is $O(m|U|^2/h^b)$. Finally, combining the two parts, we propose that the time complexity of the algorithm is $O(m|U|^2/h^b)$. However, it is worth noting that we have adopted a multi-granularity mechanism and there is no need to computation between samples of the same class, which do not change the scale of time complexity, but further increases the speed.

The algorithm adds $Noncompute$ arrays to represent samples that do not require subsequent calculations, including the x_j of excluding positive region samples due to different class with the current sample, and the x_l existing in the positive region can be determined through multi-granularity calculation.

4 WalkNAR

As one of the local search algorithm strategies, the random walk mechanics are simple and effective, and can gradually converge to the local optimal solution with the iterative process. Based on the attribute reduction problem of random walk, the basic solution strategy needs to initialize the reduction attribute set. Based on the guidance of constraints, transfer to the reduction set with better characteristics, so as to gradually update the solution in the current state until the algorithm converges to the local optimal depreciation or reaches the number of iterations given in advance.

In many cases, the better initial solution can reduce the number of trips required for convergence, but the results obtained by using the greedy algorithm for initialization often require a large number of iterations to converge, so obtaining a high-quality attribute set before iteration can make the algorithm converge faster, or even converge to a better attribute set.

In order to further understand how to obtain high-quality initial solutions, it is necessary to study the mechanism of greedy algorithms that generate initial solutions. The FHARA algorithm [38] adopts a positive region search algorithm based on a single hash bucket as shown in Fig. 3. FHARA introduces attributes with the maximum of positive region by measuring the positive region of samples under the reduced attribute set which is based on the idea of greed, gradually generating a reduced attribute set. FHARA is shown in Algorithm 2. Assuming a decision table contains $|U|$ records and k attributes are identified as the reduct from a total of m attributes, with the selection of each attribute typically adding $|U|/k$ samples to the positive region, the computational complexity required to determine the reduct is as Eq. 13

$$m|U| + (m-1)|U|\frac{k-1}{k} + \cdots + (m-k)|U|\frac{1}{k} < \frac{m|U|(1+,\ldots,+k)}{k} = \frac{m|U|(k+1)}{2} \quad (13)$$

Example 5 To further understand Algorithm 2, we provide an example to facilitate the readers' comprehension. We have a dataset $U = \{x_0, x_1, x_2, x_3, x_4, x_5\}$ and conditional attributes, denoted as a_1, a_2, and a_3. We initialize the set of reduced attributes to the empty set, $Red = \emptyset$, and calculate the positive regions for each attribute individually.

1. **First Iteration:**
   - Calculate the positive region for each attribute:

     $$POS(Q, Red \cup \{a_1\}, D, \delta) = \{x_0, x_1\}$$
     $$POS(Q, Red \cup \{a_2\}, D, \delta) = \emptyset$$
     $$POS(Q, Red \cup \{a_3\}, D, \delta) = \emptyset$$

   - Select attribute a_1 to be included in the reduced attribute set, $Red = \{a_1\}$.
   - Update the universal set Q by removing $POS(Q, \{a_1\}, D, \delta)$, resulting in a new set: $U = \{x_2, x_3, x_4\}$.
2. **Second Iteration:**
   - Calculate the positive regions for combinations with a_1:

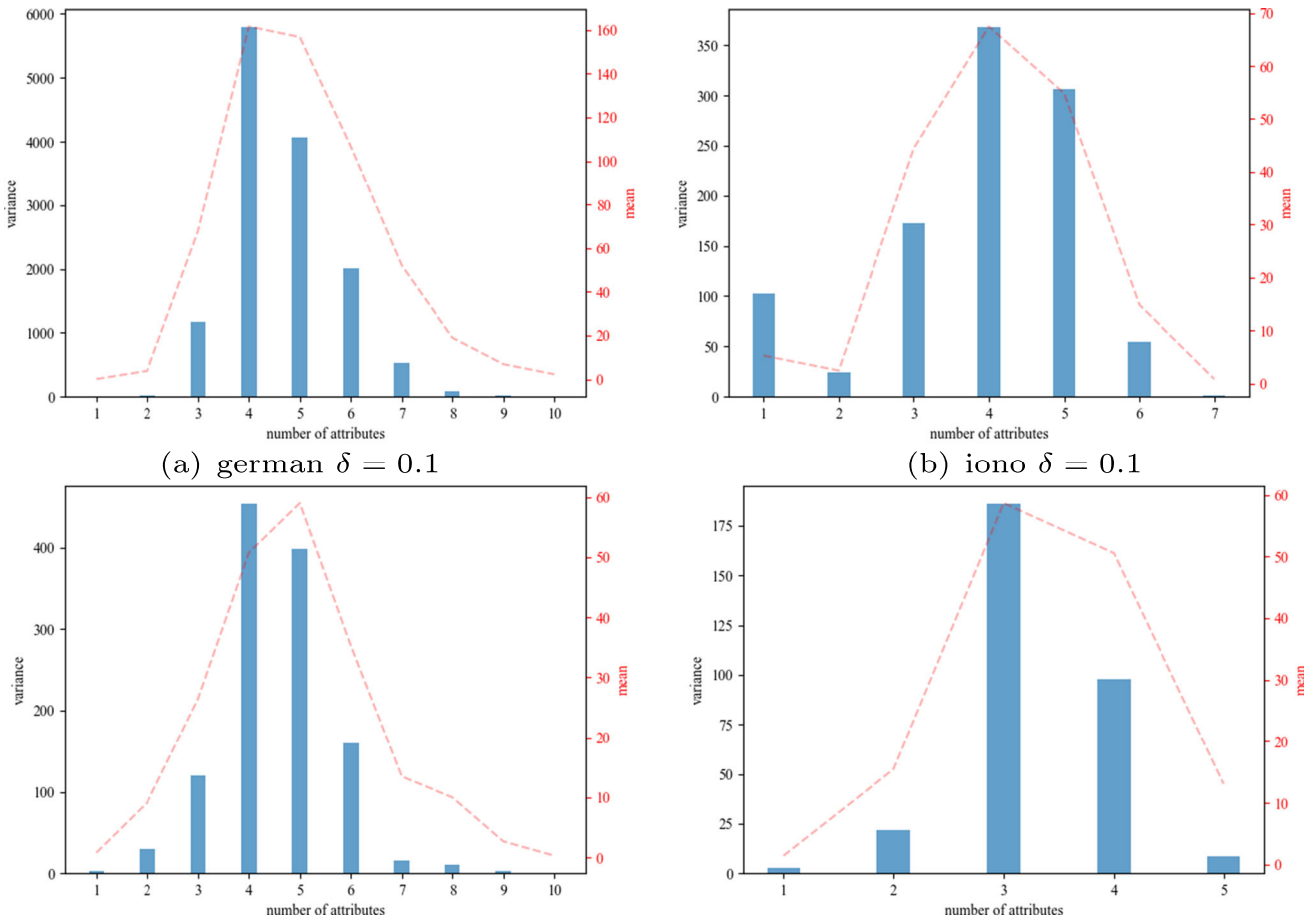
     $$POS(Q, Red \cup \{a_2\}, D, \delta) = \{x_2, x_3\}$$
     $$POS(Q, Red \cup \{a_3\}, D, \delta) = \emptyset$$

   - Choose attribute a_2 to be included in the reduced attribute set, updating Red to $\{a_1, a_2\}$.
   - Update the universal set U to a single sample: $U = \{x_4, x_5\}$.
3. **Third Iteration:**
   - Attempt to include a_3 into the reduced attribute set:

     $$POS(Q, Red \cup \{a_3\}, D, \delta) = \emptyset$$

   - Since the addition of a_3 does not increase the positive region, the algorithm terminates.

The final reduced attribute set is $Red = \{a_1, a_2\}$, and the total size of the positive region is 4, covering all samples of $\{x_0, x_1, x_2, x_3\}$.



(a) german $\delta = 0.1$ (b) iono $\delta = 0.1$

(c) horse $\delta = 0.1$ (d) sonar $\delta = 0.1$

Fig. 6 The variance and mean of the gain in the positive region

In this paper, the four datasets are selected to track and record the mean and variance of the positive region gain obtained by traversing each reduction attribute when it was added during the running of the FHARA algorithm. As shown in Fig. 6, when solving the problem, the greedy algorithm lacks consideration of the overall situation. For each greedy choice, if the selected optimal attribute has a more obvious advantage than other attributes, then the use of the greedy algorithm is completely feasible. On the contrary, if the attribute selected at each step is not clearly advantageous compared with other attributes, the algorithm lacks the interaction between attributes resulting in getting average or even poor results.

Analyzing the variance and mean of the gain in the positive region in Fig. 6, we find that the first-added and last-added attributes have the lowest positive region gain, and the positive region gain corresponding to the first and last judged attributes has a small variance, suggesting that the decision based on the positive region gain alone is blind when making decisions about these two types of attributes. Therefore, there are redundant attributes results obtained by FHARA, and the attributes in it need to be filtered and verified again.

Algorithm 2 Fast Hash Attribute Reduct Algorithm(FHARA)

Input: U, C, D, δ;
Output: Red;
1: $Red \leftarrow \emptyset, Q \leftarrow U$;
2: **while** $Q \neq \emptyset$ **do**
3: **for all** $a \in C - Red$ **do**
4: $F = F - POS(Q, Red \cup \{a\}, D, \delta)$;
5: $Pos = \{x_i \mid F_i = 1\}$;
6: Select a with max $|Pos|$ and add it to Red until max $|Pos| = 0$;
7: **end for**
8: exclude the element with the max Pos from Q;
9: **end while**

In order to improve the overall quality of the reduced set, we proposed an neighborhood rough set-based attribute reduction approach using random walk (WalkNAR), as shown in Algorithm 3, this approach changes the positive region search function in the FHARA algorithm to the MMPRSA algorithm proposed in this paper, as a method to generate the initial reduction set (i.e. MMARA algorithm), and then uses two random walk update strategies to test the reduction set for many times, as shown in Fig. 7:

1. Strategy A: random deletion of single attribute within the reduction set
2. Strategy B: replace two attributes in the reduction set with other attributes

Both strategies can gradually reduce the attributes in the reduced set, and the conditions produced by strategy A are easier to achieve, so they take precedence over strategy B in practical use and in order to prevent the decline of the representationability of the reduced set.

The algorithm will perform T trials, with each trial involving a fine-tuning of the attribute set before executing the MMPRSA algorithm. Consequently, the time complexity of the algorithm is primarily determined by the MMPRSA. Assuming that the execution of the MMPRSA algorithm involves an average of k attributes, the time complexity of the algorithm is denoted as $O(kT|U|^2/h^b)$.

5 Experiments

Our algorithm proposed in this paper selects the L_2-norm function as the distance measurement function. All tests are run in the same hardware environment, which is specified as follows: CPU: Intel(R) Core(TM) i5-11300H CPU @ 2.30GHz; RAM: 16.0 GB; Operating System: Windows 10;

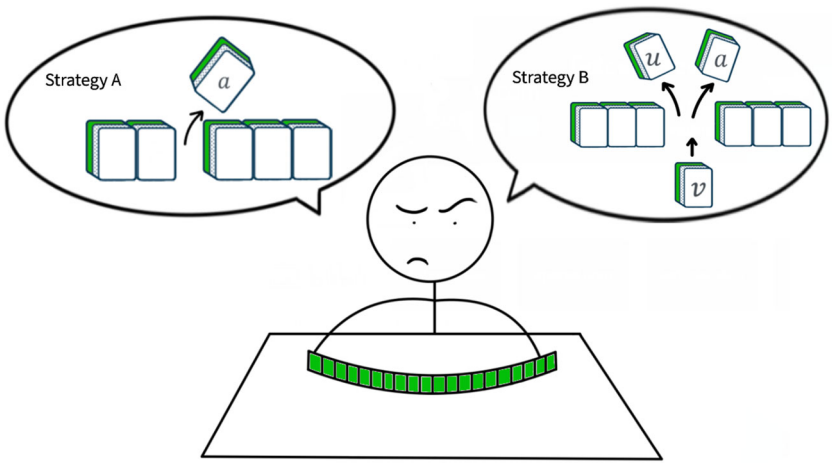


Fig. 7 Schematic diagram of the random walk strategy

Table 4 Datasets

Datasets	samples	attribution	class
abalone	4177	8	3
german	1000	21	2
glass	214	9	6
horse	368	24	2
imgseg	2310	19	7
iono	351	35	2
letter	20000	17	2
sonar	208	61	2
wdbc	569	31	2
shuttle	58000	9	5

Algorithm 3 WalkNAR

Input: U, C, D, T, δ;
Output: Red;
1: $Red \leftarrow MMARA(Q, Red \cup \{a\}, D, \delta), t \leftarrow 0$;
2: **while** $t < T$ **do**
3: remove an attribute a from Red randomly;
4: **if** $MMPRSA_{Red\setminus\{a\}}\{D\} = MMPRSA_{Red}\{D\}$ **then**
5: $Red \leftarrow Red\setminus\{a\}$;
6: **else**
7: select randomly the dealing attribute $u \in Red \setminus \{a\}$and the adding attribute $v \in C \setminus Red$;
8: **if** $MMPRSA_{(Red\setminus\{a,u\})\cup\{v\}}\{D\} \geq MMPRSA_{Red}\{D\}$ **then**
9: $Red \leftarrow (Red\setminus\{a, u\}) \cup \{v\}$;
10: **end if**
11: **end if**
12: **end while**

Python Version: 3.7. Considering that a single experiment has a certain degree of chance, this article runs all the data in the same environment 10 times and selects the average value as the presentation result. We also compared our algorithm with EasiFFRA. EasiFFRA was proposed by Wang et al. [30] and it has many good properties, such as the symmetry of neighborhood relations and the decision value filtering strategy. The average reduction time of EasiFFRA on 12 datasets is only 24.45% of that of the comparison method FHARA.

In order to prove the universality of the algorithm, 10 datasets with a large span of sample size and attribute number are selected for testing, and the details of the dataset are shown in Table 4:

5.1 Algorithm correctness verification

In order to verify the correctness of our algorithm proposed in this paper, EasiFFRA and MMARA were used to perform attribute reduction operations on the same data set. In the experiment, the values of parameter b of MMARA are 2, 4, 6, and 8. The same results were obtained for different values of b and the attribute reduction results are shown in Table 5:

It can be found that the results of the algorithm EasiFFRA and MMARA are consistent, because the attribute reduction idea is consistent with the forward search strategy.

5.2 Comparative experiment of efficiency

5.2.1 Comparison of the running time

In order to verify the high efficiency of MMARA, our experiment calculated the reduction speed of the two in the comparison process. In order to simulate the real situation, we did not change the number of sample centers during the single complete reduction, the number of sample centers selected in this experiment ranges from 2 to 8, and the data displayed are the data with the most suitable parameter selection.

As shown in Fig. 8, in order to conveniently and intuitively reflect the efficiency of our algorithm, the reduction time of the two algorithms is displayed to show the average ability of the reduction. If not mentioned in the subsequent experiments, the neighborhood radius is 0.1, The algorithm proposed in this paper can have better reduction speed on various types of data, and because our algorithm provides

Table 5 Algorithm correctness verification

Datasets	Radii	EasiFFRA	MMARA
abalone	0.18	[3, 6]	[3, 6]
german	0.13	[2, 5, 6, 13, 4, 1, 11, 12, 7, 9, 14, 8]	[2, 5, 6, 13, 4, 1, 11, 12, 7, 9, 14, 8]
glass	0.20	[8, 3, 1, 2, 9, 4, 5, 6, 7]	[8, 3, 1, 2, 9, 4, 5, 6, 7]
horse	0.12	[5, 15, 17, 8, 10, 11, 12, 23, 22]	[5, 15, 17, 8, 10, 11, 12, 23, 22]
imgseg	0.04	[19, 2, 11, 1, 18, 16, 14]	[19, 2, 11, 1, 18, 16, 14]
iono	0.09	[3, 31, 24, 14, 20]	[3, 31, 24, 14, 20]
letter	0.05	[10, 6, 12, 15, 13, 8, 2, 9, 7, 1]	[10, 6, 12, 15, 13, 8, 2, 9, 7, 1]
sonar	0.11	[44, 36, 22, 28, 7, 1]	[44, 36, 22, 28, 7, 1]
wdbc	0.12	[23, 28, 22, 12, 25, 10, 19, 1]	[23, 28, 22, 12, 25, 10, 19, 1]
shuttle	0.10	[1, 7, 3, 9, 5, 8, 2, 6, 4]	[1, 7, 3, 9, 5, 8, 2, 6, 4]

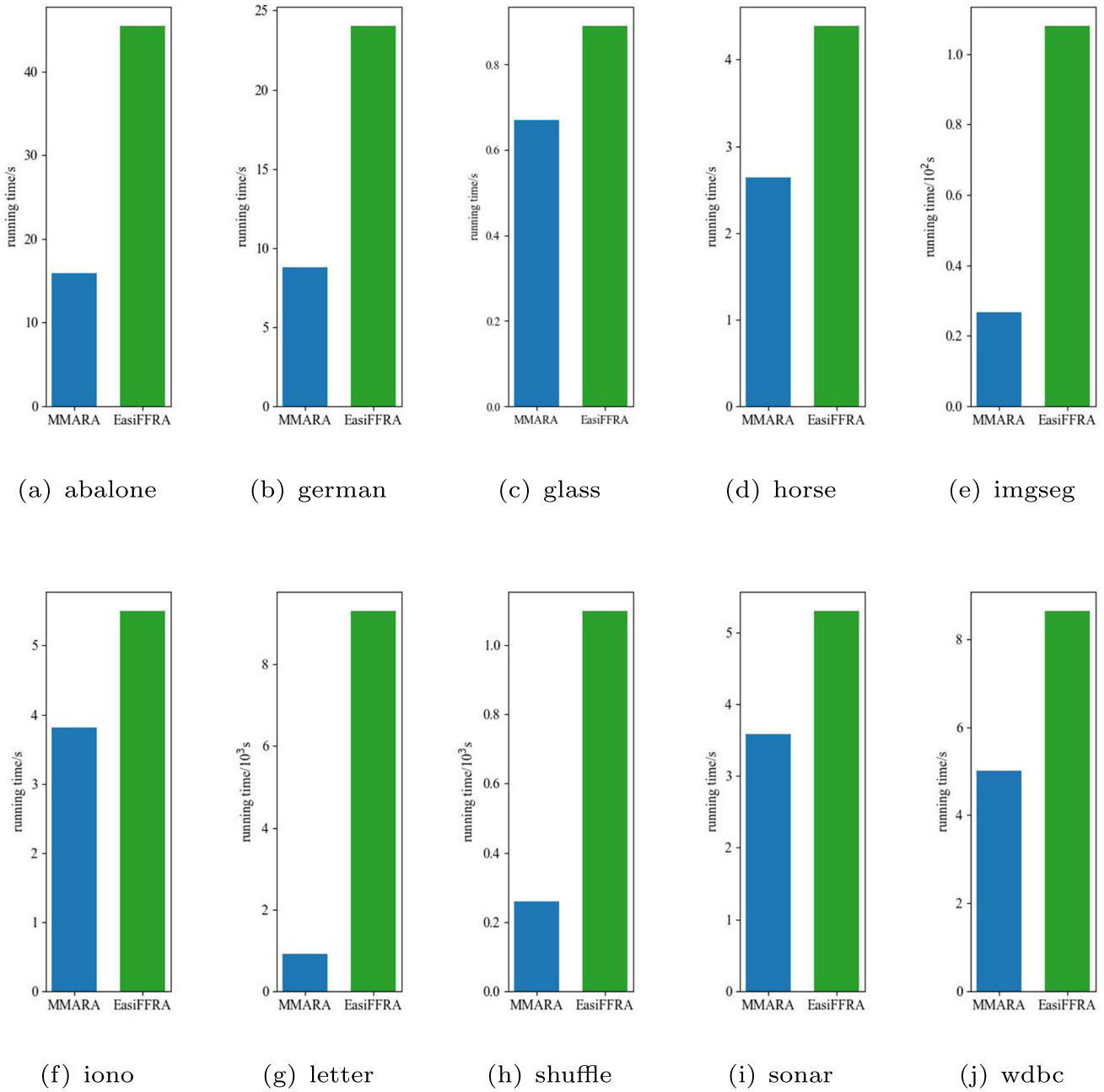


Fig. 8 Results of running time on different datasets

optional parameters b, it can be more adapted to the data with various size by modifying the parameters.

Figure 9 shows the ratio of the time used by our algorithm and the comparison algorithm on each data set. Compared with the EasiFFRA, the average time used by the algorithm proposed in this paper on the selected dataset is 45.98% of EasiFFRA. For the datasets with a data volume greater than 1000, the reduction time required by our algorithm is less than 37% of the comparison algorithm, and the time spent by MMARA on the letter dataset is only 9.87% of the comparison algorithm. This shows that the algorithm in this paper can achieve better results on large datasets.

5.2.2 Comparison of the times of MMARA distance calculations

In this section, we chose to select the number of sample centers ranging from 2 to 8, and fixed the number of sample centers in the process of single reduction, respectively

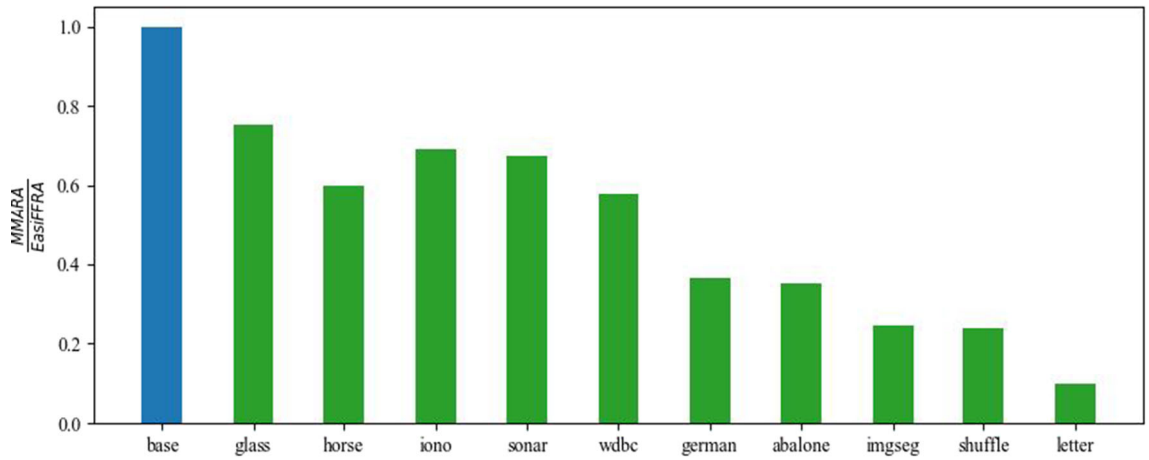


Fig. 9 Runtime comparison

counted the sum of the number of sample distances calculated by the algorithm and EasiFFRA algorithm on different data sets for comparative analysis.

As shown in Table 6, the data presented in the table is rounded, and the number of calculated distances is proportional to the running time of the algorithm, in the MMARA algorithm, unnecessary distance calculations can be effectively avoided, so that the calculation time of the positive region can be greatly improved. In the data set with the sample size greater than 1000, as shown in the data in the second row of the table, the algorithm proposed in this paper can reduce the number of calculated distances for samples to about $\frac{1}{3}$ to $\frac{1}{10}$ of the EasiFFRA algorithm in the process of attribute reduction.

5.2.3 Influence of sample size on MMPRSA

In order to study the influence of sample size on the algorithm MMARA proposed in this paper, we divided the sample center number b into four groups for testing, and in order to further explore the change of sample size, we replicated the dataset with a sample size of less than 20000, we selected the first 20,000 samples from the expanded dataset.

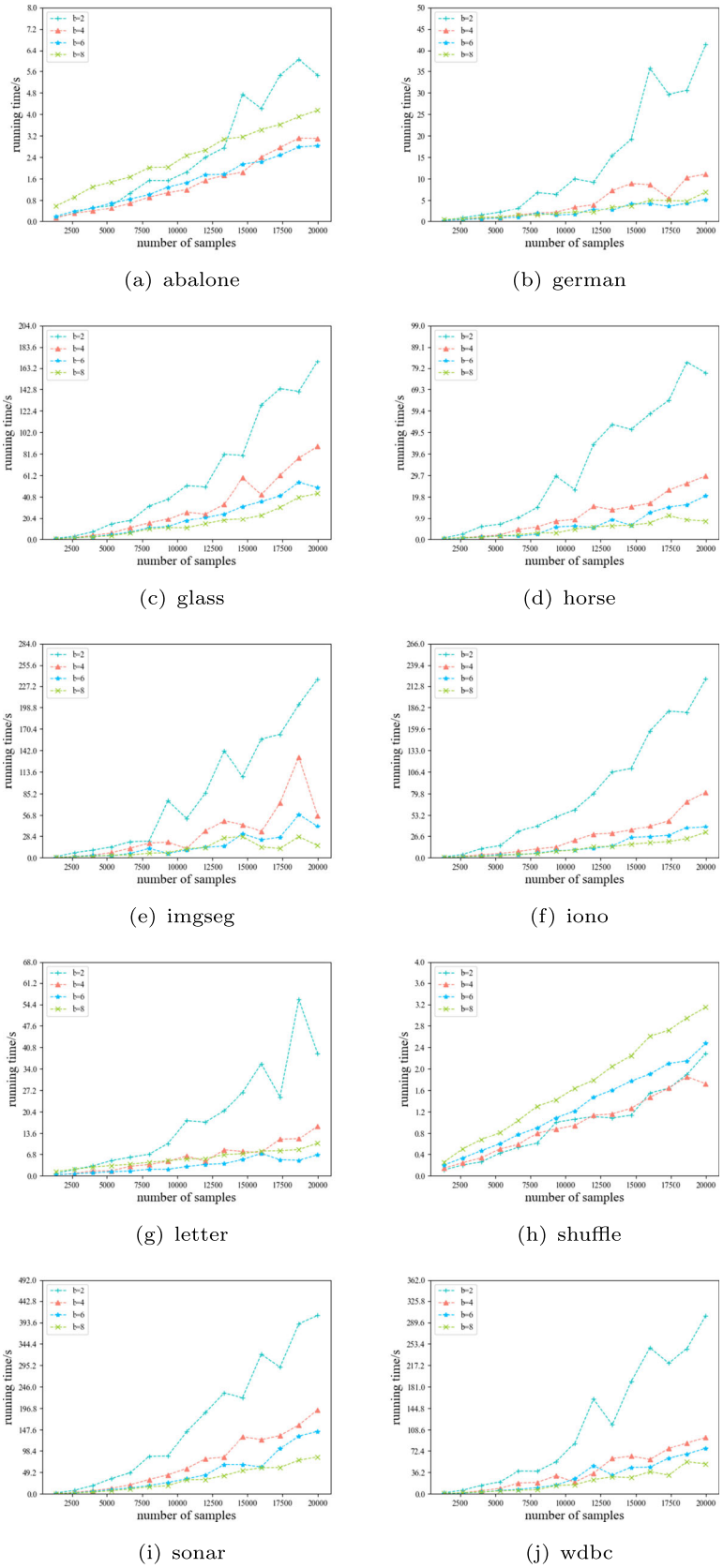
The replication and enrichment operation will change the distribution of the dataset, resulting in the significant aggregation of the same type of samples in the dataset with a small amount of data, while sample distribution properties will be retained with a larger amount of samples. Therefore, this analysis mainly based on the original sample size of data sets for a comprehensive discussion. Our experiments repeated 10 times and the average is taken as the final data display, the test results on the impact of sample size on the algorithm are shown in Fig. 10.

For most data sets, a significant phenomenon has emerged. With the increase of sample size, the algorithm proposed in this paper will prefer a larger value for the selection of parameter b. And in the case of the small original samples number, this phenomenon will be more obvious. For smaller data sets, when the sample size is greater than 14000, it is more inclined to choose $b = 8$ as the optimal parameter, and for medium-sized datasets, such as german, abalone, when the sample size reaches 16000, $b = 6$ is still used as the optimal parameter, and for the dataset that centerally has a large sample size, it mainly has different parameter selection preferences according to the nature of the dataset itself.

For datasets other than the shuttle dataset, when $b = 2$, the efficiency of the algorithm has a very serious decrease with the growth of the sample size, and when the value of b is larger, it has better operating efficiency. To explain the anomalies on the shuttle dataset, we make a specific exploration. In fact, the running speed of the algorithm in this paper on the shuttle dataset is significantly improved com-

Table 6 Comparison of the times of distance calculations

Datasets	sonar	iono	horse	glass	wdbc
$MMARA/EasiFFRA$	40.14%	69.05%	34.42%	64.00%	31.21%
$EasiFFRA$	275721	261103	223109	43727	446641
Datasets	german	abalone	imgseg	letter	shuttle
$MMARA/EasiFFRA$	20.78%	19.93%	22.80%	9.53%	35.57%
$EasiFFRA$	1295856	2058812	5850263	489112533	21630168



(a) abalone (b) german

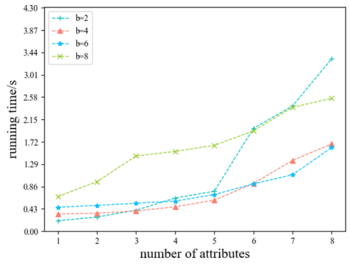
(c) glass (d) horse

(e) imgseg (f) iono

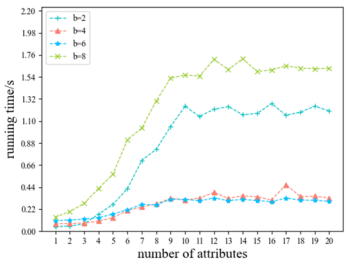
(g) letter (h) shuffle

(i) sonar (j) wdbc

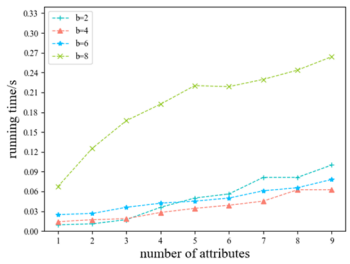
Fig. 10 Running time of MMPRSA at different sample numbers



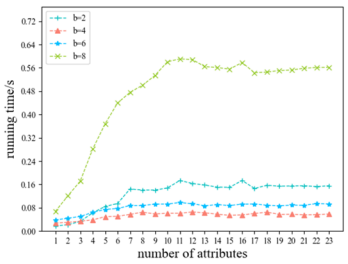
(a) abalone



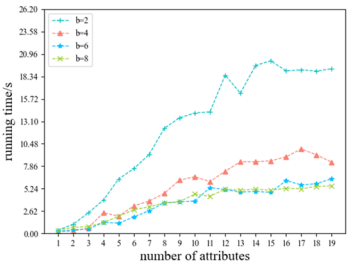
(b) german



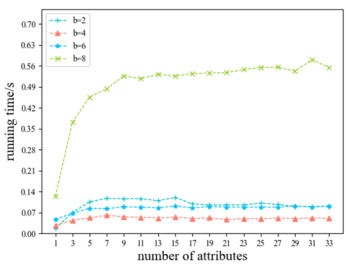
(c) glass



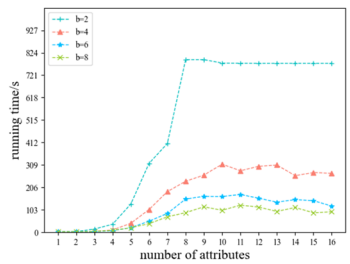
(d) horse



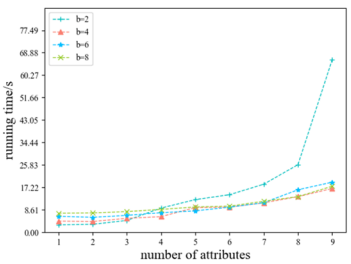
(e) imgseg



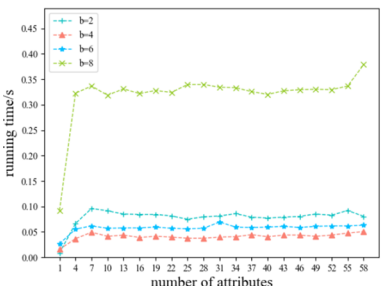
(f) iono



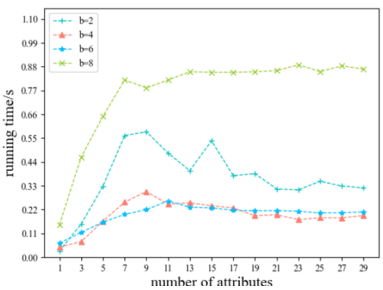
(g) letter



(h) shuffle



(i) sonar



(j) wdbc

Fig. 11 Running time of MMPRSA at different attribute sizes

Table 7 Parameter recommendation

samples	attribution	b
< 300	< 4	2
< 300	> 4	4
< 1000	any	4
≤ 5000	any	6
> 5000	< 5	6
> 5000	> 5	8

pared with other data sets. The small attribute size, and the multiple categories of samples resulting in more samples of distinct classes in the neighborhood and the early end of the traversal process, and this phenomenon will decrease with the increase of the number of attributes, which is consistent with the conclusion of Section 5.2.4.

In summary, when the sample size is large, the algorithm in this paper will prefer the larger value of b, at this time the algorithm in this paper with the growth of the sample size closer and linear growth, the selection of smaller b will lead to poor and unstable performance of the algorithm in this paper, because when the value of b is small, the sample centers are randomly initialized, the efficiency of the algorithm will be closely related to the position of a single sample center, and the selection of more sample center can reduce the dependence of the algorithm on the center of a single sample. This makes the algorithm performance more stable.

5.2.4 Running time of MMPRSA at different attribute sizes

In order to verify the relationship between the algorithm parameters and the number of attributes currently participating in the calculation, we dynamically changed the size of attributes and compared the total calculation time under each parameter, so as to give a parameter recommendation scheme for reducing the running MMPRSA algorithm.

As shown in Fig. 11, the results show that when the number of selected attributes is greater than 8, the growth of attributes will no longer significantly affect the efficiency on the dataset, while the positive region search time will show a different increase when the number of attributes is less than 8. When we selected less sample center, our algorithm does better in the case of a smaller sample size and a smaller number of attributes, and when the sample size is larger and we select more attributes, the algorithm will be more inclined to select more sample centers. In addition, when the number of sample centers is selected from 2 to 6, the positive region search time gradually decreases with the increase of the number of selected sample centers, and when the number of sample centers is 8, due to the nature of the data set, the reduction efficiency usually varies greatly.

In order to comprehensively consider the influence of sample size and attribute number on MMPRSA algorithm, we presented a better parameter selection. As shown in Table 7, we preferentially selects the parameter selection with better performance in multiple attribute sets, that is, $b = 4 or 6$, while for smaller data sets and large data sets, the influence of attributes on calculation time is discussed, our algorithm tends to favor small value of b for small sample sizes. As the sample size increases, the choice of parameter b also tends to favor larger values, so as to achieve optimal dynamic parameter selection in the process of MMARA algorithm reduction.

5.3 Quality comparison experiment

In order to test the update effect of the random walk process on the reduction set, we conducted a reduction set quality comparison experiment, using KNN, support vec-

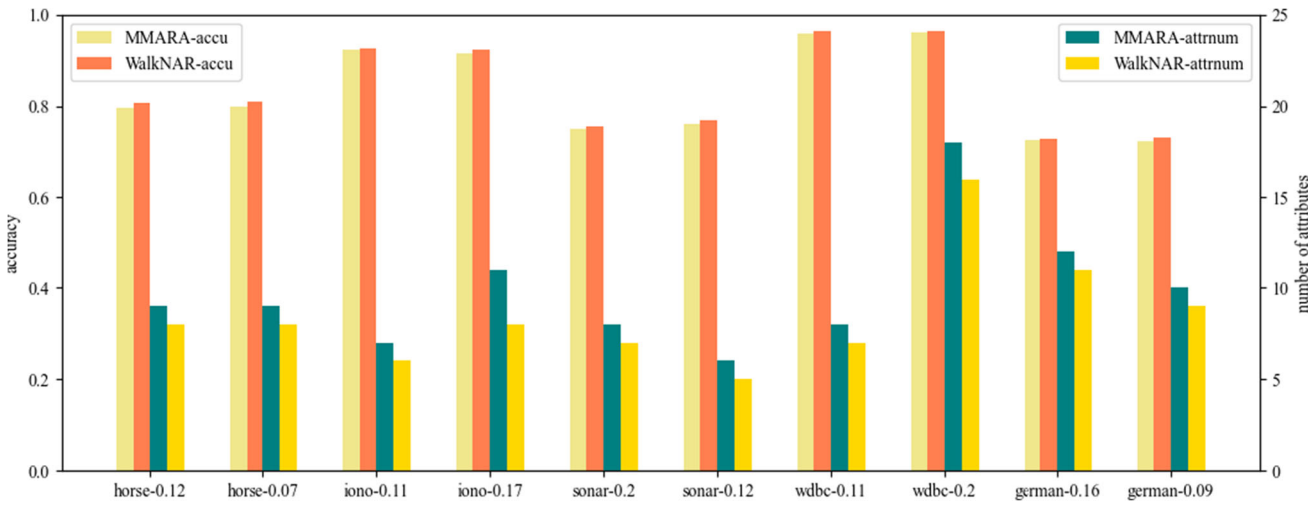


Fig. 12 Accuracy and number of reduction attributes comparison

tor machine classifier, random forest to perform a ten-fold cross-check test on the reduction set obtained by MMARA and the reduced set obtained by WalkNAR. The average classification accuracy of the three classifiers is taken as the final result. In order to facilitate further reduction, we selected datasets with a dataset attribute ≥ 20 for experiments, and mainly compares the results of MMARA and WalkNAR.

As shown in Fig. 12, the data subscript is indicated as the dataset-neighborhood radius. WalkNAR can compress the reduced attribute set to 37.49% of the original attributes, which is 4.96% higher than that of MMARA, and the reduced set after the attribute reduction approach proposed in this paper has the same or even better classification accuracy than the reduced set directly obtained by MMARA. In fact, the classification accuracy of the attribute set obtained by the Narwalk algorithm on each classifier is 84.0476%, which is 0.6667% higher than that of MMARA. The reduced set processed WalkNAR can delete or replace the attributes with poor classification ability in the reduction set so as to achieve the purpose of further reduction.

6 Conclusion

This paper proposed a positive region search algorithm MMPRSA based on multi-hash bucket and multi-granularity mechanism, which solved the problem of locating samples in the neighborhood through information synthesis between multiple sample centers for a large number of redundant calculations in the positive region search process. In addition, the algorithm used multi-granularity to transform the positive region judgment problem of samples into the positive region judgment problem of all samples in a hash bucket, thereby reducing the subsequent traversal operation.

In order to test the characteristics of the algorithm, we proposed an improved algorithm MMARA and an attribute reduction approach WalkNAR, and the comparative experiments show that the proposed algorithm can greatly accelerate the process of attribute reduction, and obtain a better set of reductions. The algorithm gave different parameter choices compared to the comparison algorithm, so it can adapt to data sets with different characteristics.

Acknowledgements The authors thank the support of College of Artificial Intelligence, Nanjing Agricultural University, China. The research was supported by the startup foundation of new doctoral at Nanjing Agricultural University (Grant No. 106/804002), and the National Natural Science Foundation of China(Grant No.62072247).

Author Contributions Haibo Li: Conceptualization, Methodology, Writing original draft, Software. Wuyang Xiong: Data processing, Validation. Yanbin Li: Conceptualization, Writing-review and editing. Xiaojun Xie: Conceptualization, Methodology, Writing-review and editing.

Data Availability The data that support the findings of this study are available from the corresponding author upon reasonable request.

Declarations

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

References

1. Abdolrazzagh-Nezhad M, Radgohar H, Salimian SN (2020) Enhanced cultural algorithm to solve multi-objective attribute reduction based on rough set theory. Math Comput Simul 170:332–350
2. An S, Guo X, Wang C et al (2023) A soft neighborhood rough set model and its applications. Inf Sci 624:185–199
3. Atef M, Khalil AM, Azzam A et al (2021) Comparison of twelve types of rough approximations based on j-neighborhood space and j-adhesion neighborhood space. Soft Comput 26(1):215–236
4. Cha B, Li Z (2020) A dynamic framework for updating neighborhood multigranulation approximations with the variation of objects. Inf Sci 519:382–406
5. Chu X, Sun B, Li X et al (2020) Neighborhood rough set-based three-way clustering considering attribute correlations: An approach to classification of potential gout groups. Inf Sci 535:28–41
6. Gao C, Zhou J, Miao D et al (2021) Granular-conditional-entropy-based attribute reduction for partially labeled data with proxy labels. Inf Sci 580:111–128
7. Guo Y, Hu M, Wang X et al (2022) A robust approach to attribute reduction based on double fuzzy consistency measure. Knowl-Based Syst 253(109):585
8. Hu Q, Yu D, Liu J et al (2008) Neighborhood rough set based heterogeneous feature subset selection. Inf Sci 178(18):3577–3594
9. Jiang Z, Liu K, Yang X et al (2020) Accelerator for supervised neighborhood based attribute reduction. Int J Approximate Reasoning 119:122–150
10. Ju H, Ding W, Shi Z et al (2022) Attribute reduction with personalized information granularity of nearest mutual neighbors. Inf Sci 613:114–138
11. Kang Y, Dai J (2023) Attribute reduction in inconsistent grey decision systems based on variable precision grey multigranulation rough set model. Appl Soft Comput 133(109):928
12. Li Y, Cai M, Zhou J, et al (2022) Accelerated multi-granularity reduction based on neighborhood rough sets. Applied Intelligence pp 1–16
13. Liu K, Yang X, Fujita H et al (2019) An efficient selector for multi-granularity attribute reduction. Inf Sci 505:457–472
14. Liu K, Li T, Yang X et al (2022) Hierarchical neighborhood entropy based multi-granularity attribute reduction with application to gene prioritization. Int J Approximate Reasoning 148:57–67
15. Pawlak Z (1982) Rough sets. International journal of computer & information sciences 11:341–356
16. Pawlak Z, Skowron A (2007) Rudiments of rough sets. Inf Sci 177(1):3–27. Zdzislaw Pawlak life and work (1926–2006)

17. Peng X, Wang P, Xia S et al (2022) Vpgb: A granular-ball based model for attribute reduction and classification with label noise. Inf Sci 611:504–521
18. Peng X, Wang P, Xia S et al (2022) FNC: A fast neighborhood calculation framework. Knowl-Based Syst 252:109394
19. Qing H (2008) Efficient symbolic and numerical attribute reduction with neighborhood rough sets. Pattern Recognition and Artificial Intelligence
20. Sanabria J, Rojo K, Abad F (2023) A new approach of soft rough sets and a medical application for the diagnosis of coronavirus disease. AIMS Mathematics 8(2):2686–2707
21. Srirekha B, Sathish S, Narmada Devi R et al (2023) Attributes reduction on se-isi concept lattice for an incomplete context using object ranking. Mathematics 11(7):1585
22. Su J, Wang Y, Li J (2023a) A novel fuzzy covering rough set model based on generalized overlap functions and its application in mcdm. Symmetry 15(3)
23. Su L, Yu F, Li J et al (2023b) Incremental updating reduction for relation decision systems with dynamic conditional relation sets. Information Sciences
24. Sun B, Tong S, Ma W, et al (2021) An approach to mcgdm based on multi-granulation pythagorean fuzzy rough set over two universes and its application to medical decision problem. Artificial Intelligence Review
25. Tallón-Ballesteros A (2020) Neighborhood based multi-granularity attribute reduction: An acceleration approach. Fuzzy Systems and Data Mining 331:234
26. Wang C, Huang Y, Shao M et al (2019) Feature selection based on neighborhood self-information. IEEE Transactions on Cybernetics 50(9):4031–4042
27. Wang C, Wang Y, Shao M et al (2019) Fuzzy rough attribute reduction for categorical data. IEEE Trans Fuzzy Syst 28(5):818–830
28. Wang C, Huang Y, Ding W et al (2021) Attribute reduction with fuzzy rough self-information measures. Inf Sci 549:68–86
29. Wang C, Qian Y, Ding W et al (2021) Feature selection with fuzzy-rough minimum classification error criterion. IEEE Trans Fuzzy Syst 30(8):2930–2942
30. Wang N, Peng Z, Cui L (2019) EasiFFRA: a fast feature reduction algorithm based on neighborhood rough set(in Chinese). Journal of Computer Research and Development 56(12):2578–2588
31. Xia S, Zhang H, Li W et al (2020) GBNRS: A novel rough set algorithm for fast adaptive attribute reduction in classification. IEEE Trans Knowl Data Eng 34(3):1231–1242
32. Yang X, Li T, Liu D et al (2020) A multilevel neighborhood sequential decision approach of three-way granular computing. Inf Sci 538:119–141
33. Yang X, Li M, Fujita H et al (2022) Incremental rough reduction with stable attribute group. Inf Sci 589:283–299
34. Yang X, Yang Y, Luo J et al (2022) A unified incremental updating framework of attribute reduction for two-dimensionally time-evolving data. Inf Sci 601:287–305
35. Yao W, Zhang G, Zhou CJ (2023) Real-valued hemimetric-based fuzzy rough sets and an application to contour extraction of digital surfaces. Fuzzy Sets Syst 459:201–219
36. Ye J, Zhan J, Ding W et al (2021) A novel fuzzy rough set model with fuzzy neighborhood operators. Inf Sci 544:266–297
37. Yin T, Chen H, Yuan Z et al (2023) Noise-resistant multilabel fuzzy neighborhood rough sets for feature subset selection. Inf Sci 621:200–226
38. Yong L, Wenliang H, Yunliang J et al (2014) Quick attribute reduct algorithm for neighborhood rough set model. Inf Sci 271:65–81
39. Zhao DS, Song JJ, Xu TH, et al (2021) Accelerator on multi-granularity attribute reduction for continuous parameters. In: 2021 International Conference on Machine Learning and Cybernetics (ICMLC), IEEE, pp 1–6
40. Zhe D, Jianhui L (2015) A positive region-based dimensionality reduction from high dimensional data. In: 2015 8th International Conference on Biomedical Engineering and Informatics (BMEI), IEEE, pp 624–628
41. Zou L, Ren S, Li H et al (2021) An optimization of master s-n curve fitting method based on improved neighborhood rough set. IEEE Access 9:8404–8420
42. Zou L, Ren S, Sun Y et al (2023) Attribute reduction algorithm of neighborhood rough set based on supervised granulation and its application. Soft Comput 27(3):1565–1582