Information Sciences 679 (2024) 121099

Contents lists available at ScienceDirect

Information Sciences

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

A novel adaptive neighborhood rough sets based on sparrow search algorithm and feature selection

Caihui Liu [a,*], Bowen Lin [a], Duoqian Miao [b]

[a] Department of Mathematics and Computer Science, Gannan Normal University, Ganzhou, 341000 Jiangxi, China
[b] Department of Computer Science and Technology, Tongji University, 201804 Shanghai, China

ARTICLE INFO

Keywords:
Adaptive neighborhood rough set model
Feature selection
Sparrow search algorithm

ABSTRACT

Neighborhood rough sets-based methods have been widely used for feature selection. However, the existing methods have some problems in neighborhood construction, such as the application of the same neighborhood radius for all samples. Thus, this paper proposed a novel adaptive neighborhood rough set model based on Sparrow Search Algorithm (SSA) to tackle the above problems, and applied the model to feature selection. First, we reconsidered the problems mentioned above from the viewpoint of optimization where the neighborhood radius of the target sample is considered as the solution to the optimization, and the maximum percentage of the label of the neighborhood formed is considered as the target to the optimization. Second, SSA is introduced to design the adaptive neighborhood construction to tackle the optimization problem where all candidate neighborhood radii of the target sample are considered as sparrows, the maximum and minimum distances between the target sample and other samples are considered as the search range, and the maximum label rate defined in this paper is considered as the search target. Then, a novel adaptive neighborhood rough set model is proposed by using the adaptive neighborhood construction. Third, we proposed a feature selection algorithm based on the adaptive neighborhood rough set model. Finally, the experimental results on seventeen datasets demonstrate the effectiveness of our algorithm. The running time of the proposed algorithm is at least one time less than classical algorithms under the condition that the classification performance is better, the accuracy increases 3% and the balanced accuracy increases 4%.

1. Introduction

In the past decades, due to the digitalization of life, the proliferation of various data created by people has rapidly increased the data dimension [1–3]. There are usually redundant or irrelevant features in such high-dimensional data, which may lead to several problems such as high time cost and low model prediction performance [4,5]. Therefore, feature selection has received much attention in recent years as a practical approach for handling high-dimensional data [6–8].

It is generally accepted that the neighborhood rough set model plays a crucial role in feature selection [9–11]. Along with data sources becoming increasingly complex, scholars have investigated various rough set models to enhance the robustness of the theory [12–14]. Hu et al. [15] studied a heterogeneous feature subset selection method by optimizing the distance measure in the η-neighborhood rough set model. Yuan et al. [16] proposed a feature selection method by exploiting the granular level structure in

* Corresponding author.
E-mail addresses: liucaihui@gnnu.edu.cn, liu_caihui@163.com (C. Liu), lin_bw@qq.com (B. Lin), dqmiao@tongji.edu.cn (D. Miao).

https://doi.org/10.1016/j.ins.2024.121099
Received 18 January 2024; Received in revised form 21 June 2024; Accepted 22 June 2024

Available online 27 June 2024

knowledge space and designing a zentropy-based uncertainty measure. Xu et al. [17] designed the k-nearest neighborhood rough set model by considering heterogeneous data and feature interaction, and proposed a feature selection algorithm using information entropy. Ibrahim et al. [2] proposed a new metaheuristic algorithm called the runner-root algorithm by combining rough sets and neighborhood rough sets. Then, the algorithm was applied to feature selection and performs better than the state-of-the-art metaheuristic algorithms. Yuan et al. [18] proposed a feature selection using the variable precision composite measure and the local forward search approach. All experiments showed that their local method is efficient. Wang et al. [19] proposed a feature selection to handle heterogeneous data by combining the η-neighborhood rough set model and the k-nearest neighbor rough set model.

Although the existing methods have achieved great success, three problems limit their applications [20–22]: (1) the recognition of discrete attributes in heterogeneous data; (2) the approach of iteration to select the neighborhood radius with the best performance; (3) the application of the same neighborhood radius for all samples. Fortunately, some attempts to focus on the problems have been reported. Zhang et al. [20] proposed a feature selection approach by inducing the conditional neighborhood combination entropy using the neighborhood rough set model as a framework to handle heterogeneous data. The experimental results showed the effectiveness and superiority of the proposed algorithm. Chen et al. [23] introduced a feature selection approach by composing classical rough set and fuzzy rough set models to perform attribute reduction for heterogeneous data. Yuan et al. [24] studied a feature selection approach based on fuzzy mutual information, which aims to effectively select relevant features from heterogeneous data without decision. Alshami et al. [25] presented a kind of subset neighborhood rough set model by using the inclusion relations between neighborhoods under an arbitrary binary relation. Zhang et al. [26] proposed two attribute reduction methods based on variable radius neighborhood rough set and α-covering neighborhood rough set models to customize the neighborhood radius for each object by considering the different environment of each object and label distribution. Qu et al. [27] proposed a feature selection method by the adaptive neighborhood rough set model considering the label of the nearest sample and using the rough mutual information. The experimental results demonstrated the efficiency of the proposed algorithm.

Even though these approaches have addressed the problems to some extent, improvements remain needed. For instance, their capacity for adaptation may not always be enough to handle more complex data distribution. The introduction of a more powerful search strategy may be helpful to the issues. As is well known, swarm intelligence algorithms have been widely used in parameter optimization due to their excellent performance. Thus far, an increasing number of introductions of swarm intelligence algorithms to rough set theory have been recorded to enhance the performance [28]. Liu et al. [29] presented an attribute reduction algorithm based on the adaptive genetic algorithm that adjusts the crossover probability and mutation probability of each individual according to individual fitness value. Sadiq et al. [30] presented a hybrid approach for solving null value problems by hybridizing rough set theory with the intelligent swarm algorithm. Chen et al. [31] proposed a feature selection method by combining the rough set model and the Fish Swarm Algorithm. The results demonstrate that the algorithm can provide an efficient tool for finding a minimal subset of the features without information loss. Maini et al. [32] proposed a feature selection method by employing the rough dependency measure as the fitness function in Particle Swarm Optimization and Intelligent Dynamic Swarm. The results of the experiments show that with the help of the proposed initialization, the two algorithms are able to select the best set of features with less execution time. Sun et al. [33] proposed a two-stage feature selection by fusing the fuzzy multi-neighborhood rough set with binary whale optimization. Experiments show their algorithm is efficient and can achieve excellent classification efficiency for binary and multiclass imbalanced data. Chen et al. [34] proposed an artificial hummingbird algorithm-based three-way K-means clustering algorithm. The experimental results demonstrate that AHA-3WKM performs well, and enhances the stability and accuracy of clustering results. Therefore, how to introduce the swarm intelligence algorithm to the neighborhood rough set model to tackle the problems above is worth discussion and research.

In this paper, we first introduce the concept of label rate, which is defined via the percentage of the label of x in the neighborhood of x, where x is the target sample. Second, we apply SSA to the search of the adaptive neighborhood radius of x by using the following settings: the neighborhood radius is regarded as a sparrow, the maximum and minimum distances between x and other samples are considered as the search range, and the maximum label rate is regarded as the search target. These settings can solve problems (1)-(3) mentioned above, where the solution to problem (1) is detailed in Example 2, and the solutions to problems (2) and (3) are detailed in Example 1. Third, we construct an SSA-based adaptive neighborhood rough set model by conducting the neighborhood granulation of samples via the adaptive neighborhood radius. Fourth, we propose a feature selection algorithm based on the adaptive neighborhood rough set model. The proposed algorithm is compared with five neighborhood rough set-based feature selection algorithms regarding running time and the attribute group selected. The attribute group returned is verified via the KNN and Decision tree classifiers. Finally, the experimental results show that our algorithm has lower running time, and the attribute group selected by our algorithm has higher accuracy and balanced accuracy.

The main contributions of this paper are given as follows.

(1) We transform the problems into a parameter optimization problem and propose the concept of label rate according to the percentage of the target sample's label in the target sample's neighborhood.

(2) We apply SSA to the search of the adaptive neighborhood radius of the target sample by using the following settings: the candidate neighborhood radii are regarded as sparrows, the maximum and minimum distances between the target sample, and other samples are regarded as the search range, and the maximum label rate is considered as the search target.

(3) We apply the adaptive neighborhood radius to the neighborhood granulation of samples and construct SSA-based adaptive neighborhood rough set model.

(4) We propose a novel feature selection algorithm based on the adaptive neighborhood rough set model. By comparing our algorithm with other algorithms, we demonstrate its effectiveness. That is, it has a lower running time and more robust adaptability to complex environments, and the attribute group selected by it has higher accuracy and balanced accuracy.

The remaining part of this paper is organized as follows. Section 2 reviews the basic concepts of the neighborhood rough sets and SSA. In Section 3, the concept of label rate is defined. SSA is applied to calculate the adaptive neighborhood radius, and SSA-based adaptive neighborhood rough set model is constructed. Moreover, a novel feature selection algorithm based on the model is proposed. In Section 4, the experiments are conducted to prove the effectiveness of our algorithm. Finally, conclusions and future work are given in Section 5.

2. Preliminaries

In this section, the η-neighborhood rough set model and the Sparrow Search Algorithm are reviewed.

2.1. η-Neighborhood rough sets

The neighborhood rough set model is an extension of the classical rough set model, which can effectively handle numerical and heterogeneous data [15].

In rough sets, an information table is a 2-tuple $\langle U, A\rangle$, where $U = \{sp^1, sp^2, \ldots, sp^n\}$ is a set containing all samples, denoted as the universe, A is an attribute set to describe the characteristics of the samples. If $A = C \cup D$, where C is an attribute subset to describe the condition characteristics and D is an attribute subset to describe the decision characteristics, then the information table $\langle U, A\rangle$ is also called a decision table.

For any two samples $sp^i, sp^j \in U$ and any attribute subset $subC \subseteq C$, let η be the neighborhood radius, $MinD^{subC}\left(sp^i, sp^j\right)$ represent the Euclidean distance between sample sp^i and sample sp^j under attribute subset $subC$. The η-neighborhood of sample sp^i under $subC$ is defined as follows:

$$\eta^{subC}\left(sp^i\right) = \left\{ sp^j \in U \,\middle|\, MinD^{subC}\left(sp^i, sp^j\right) < \eta \right\}. \tag{1}$$

In neighborhood rough sets, a neighborhood approximation space is a 2-tuple $\langle U, Nei\rangle$, where $U = \{sp^1, sp^2, \ldots, sp^n\}$ is the universe and Nei is the neighborhood relation.

Let $\langle U, Nei\rangle$ be a neighborhood approximation space, for any target set $X \subseteq U$ and any attribute subset $subC \subseteq C$, the lower and upper approximations of X under $subC$ are defined as follows:

$$\underline{Nei}^{subC} X = \left\{ sp^i \in U \,\middle|\, \eta^{subC}\left(sp^i\right) \subseteq X \right\}; \tag{2}$$

$$\overline{Nei}^{subC} X = \left\{ sp^i \in U \,\middle|\, \eta^{subC}\left(sp^i\right) \cap X \neq \emptyset \right\}, \tag{3}$$

where the lower approximation of target set X under $subC$ is also called the positive region of X under $subC$, denoted as $POS^{subC}(X)$.

2.2. Sparrow search algorithm

SI-MA et al. [35] evaluated 123 swarm intelligence algorithms to find the appropriate algorithm for diverse issues. Based on the results, they claimed that SSA is one of the algorithms suitable for optimization in problems with low and medium dimensions. SSA, proposed by Xue et al. [36], is one of the novel and robust swarm intelligence algorithms for handling optimization problems. The evaluation of SSA is performed with 19 known mathematical functions. This algorithm has shown a remarkable power in addressing complex calculations and solutions [37,38]. SSA [36] is a nature-inspired algorithm inspired by the sparrow population's foraging and anti-predator behaviors. SSA provides an excellent global search capability. The sparrows, in SSA, are divided into producers, scroungers, and warners to simulate the procedure of search optimization, where the percentage of producers and scroungers in sparrows is constant, and the warners account for 20% to 30%. As a particle in the search space, each sparrow represents a feasible solution to the current problem. The behaviors of sparrows are idealized and formulated by the following rules.

(1) The producers have high energy reserves and provide food and guide the direction of food for scroungers. The levels of energy reserves rely on fitness values.

(2) If the sparrows detect the predator, the individuals begin to warn. All sparrows must fly to safe areas once the warning value exceeds the safety threshold set.

(3) The roles of producers and scroungers are sometimes different. Searching for better food sources can make the sparrow become a producer.

(4) Scroungers can find better food sources through producers.

Hence, the mathematical model of SSA can be defined as follows.

In SSA, the position of a producer at the $t{+}1^{th}$ iteration can be defined as follows [36].

$$x_{i,j}^{t+1} = \begin{cases} x_{i,j}^{t} \times \exp\left(\frac{-i}{\alpha * item_{max}}\right), & \text{if} \quad R_2 < ST; \\ x_{i,j}^{t} + Q \times L, & \text{if} \quad R_2 \geq ST. \end{cases} \tag{4}$$

where $x_{i,j}^{t}$ represents the value of the j^{th} dimension of the i^{th} producer at the t^{th} iteration, $\alpha \in (0, 1)$ is a random value, $item_{max}$ represents the maximum number of iterations, Q represents a random value following the normal distribution, L represents a matrix

of $1 \times d$ in which all elements are 1. R_2 represents the warning value (R_2 is a random value uniformly distributed between $[0, 1]$), and $ST \in [0.5, 1]$ represents the safety threshold.

In Equation (5), $R_2 < ST$ indicates that the group is in a safe environment, and the producers enter a global search. On the contrary, $R_2 \geq ST$ indicates that some sparrows have discovered the danger, and all sparrows need to quickly fly to the safe area.

In SSA, the scroungers frequently monitor the producers, and as soon as the producers find good food, they immediately go to appropriate locations to compete. If they win, they will obtain the food; otherwise, they will continue to monitor. The position of a scrounger at the $t{+}1^{th}$ iteration can be defined as follows [36].

$$x_{i,j}^{t+1} = \begin{cases} Q \times \exp\left(\frac{x_{worst}^{t} - x_{i,j}^{t}}{i^2}\right), & \text{if} \quad i > \frac{n}{2}; \\ x_p^{t+1} + \left|x_{i,j}^{t} - x_p^{t+1}\right| \times A^{+} \times L, & \text{otherwise}, \end{cases} \tag{5}$$

where x_{worst}^{t} represents the global worst position at the t^{th} iteration, x_p^{t+1} represents the local best position at the $t{+}1^{th}$ iteration, and A^{+} is a matrix of $d \times 1$ obtained by the operation $A^{T}(AA^{T})^{(-1)}$ on a matrix A of $1 \times d$ (each element in A is randomly assigned a value of 1 or -1).

In Equation (6), $i > n/2$ indicates that the i^{th} scrounger with a poor fitness value may become hungry, and it will fly elsewhere to find food.

In SSA, the warners warn of danger to other sparrows so that the producers can lead the sparrows to more secure locations. The warners generally account for 20% to 30% of the entire sparrows. The position of a warner at the $t{+}1^{th}$ iteration can be defined as follows [36].

$$x_{i,j}^{t+1} = \begin{cases} x_{best}^{t} + \beta \times \left|x_{i,j}^{t} - x_{best}^{t}\right|, & if \quad f_i > f_g; \\ x_{i,j}^{t} + k \times \frac{\left|x_{i,j}^{t} - x_{worst}^{t}\right|}{(f_i - f_w) + \epsilon}, & if \quad f_i = f_g, \end{cases} \tag{6}$$

where x_{best}^{t} represents the global best location (the center of the sparrow group with a secure location) at the t^{th} iteration, $\beta \sim N(0, 1)$ is a random value used to control the step of iteration, $k \in [-1, 1]$ is a random value, f_i represents the fitness value of the current sparrow, f_g and f_w represent the global best and worst fitness values respectively, and η is a very small constant that prevents the denominator from being zero.

In Equation (7), $f_i > f_g$ indicates that the current sparrow is at the edge of the group. Moreover, $f_i = f_g$ indicates that the current sparrow is aware of the danger and needs to get closer to other sparrows.

3. SSA-based adaptive neighborhood rough sets and feature selection

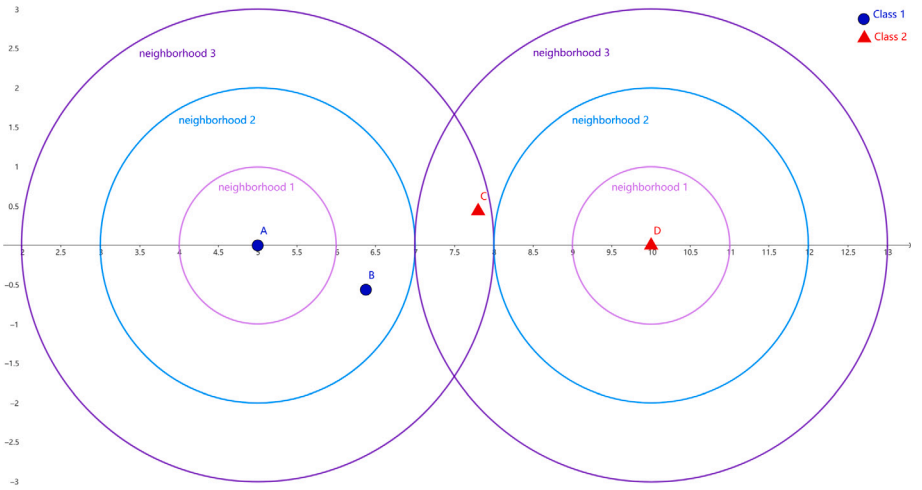
In this section, the problems mentioned above are reconsidered from the optimization viewpoint. SSA is introduced to the neighborhood rough set model to compute the best neighborhood radius of each sample in a given dataset. A novel adaptive neighborhood construction method is designed. Moreover, an adaptive neighborhood rough set model is constructed by the best neighborhood radius and is applied to feature selection.

3.1. The transformation to an optimization problem

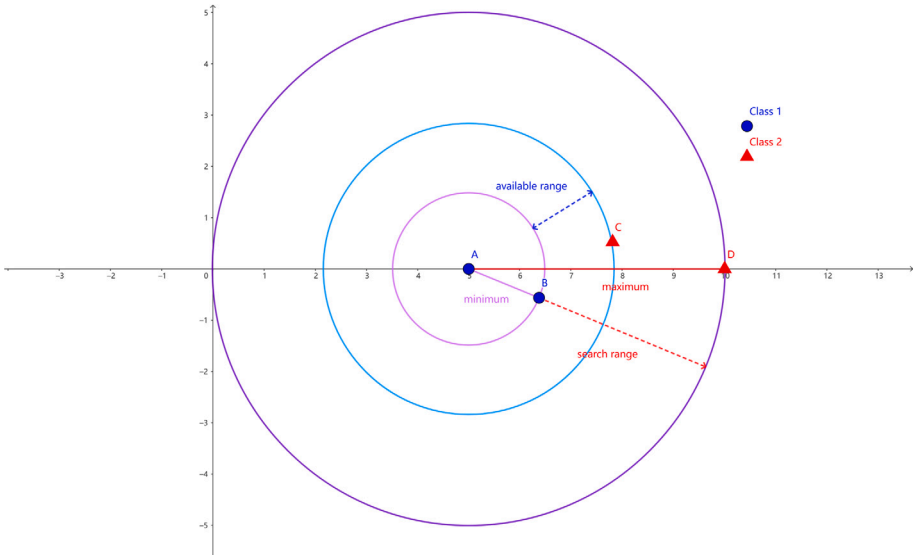
It is known that the neighborhood radius of samples is set empirically in classical neighborhood rough sets [1,39]. However, this kind of setting does not consider the actual situation around the sample during the neighborhood construction. As a result, the performance of a single neighborhood radius may need improvement. To solve this problem, the strategy of iteration [40,15], also known as the step-by-step approach, is adopted, where an initial value and a terminal value are given to iterate the neighborhood radius in a certain step. Unfortunately, it is clear that the strategy is affected seriously by the empirical settings, i.e., the initial value, the terminal value, and the step, which may induce the model's performance to be suboptimal rather than optimal in some complicated situations. Therefore, it is necessary to transform a classical neighborhood construction into an adaptive one to avoid the influence caused by the empirical settings. In this subsection, we shall use Example 1 to show how to transform the problems mentioned above into an optimization problem and how to solve it.

Example 1. Four samples A, B, C and D are given as Fig. 1(a), where A and B belong to Class 1 (blue dot), and C and D belong to Class 2 (red triangle). To show the procedure of a classical neighborhood construction, in the step-by-step approach, we set the initial value as 1, the terminal value as 3 and the step as one step. Therefore, for sample A, neighborhood 1 is constructed via the initial value 1 of the neighborhood radius in the first round. Then the neighborhood radius is turned into 2 by the step value 1 in the next round, and the neighborhood 2 is constructed via the value 2 of the neighborhood radius. Similarly, the neighborhood radius is turned into 3 in the last round, and the neighborhood 3 is constructed. The three neighborhoods of sample D can be gotten in a similar way. As a result, the three neighborhoods of sample A and sample D are formed in three rounds of iterations.

According to the attribute selection approach in classical neighborhood rough sets, we know that the attribute with the largest attribute dependency is preferred. On this basis, the attribute with the largest positive region is preferred. Meanwhile, the size of the positive region is determined by whether the label of the samples in the neighborhood of the target sample x is the same as



(a) The search approach of classical neighborhood rough sets



(b) The search approach of adaptive neighborhood radius

Fig. 1. Example 1.

that of x. Hence, when the labels in the neighborhood formed are uniform, the neighborhood radius is biased towards the selected neighborhood radius. Since the application of neighborhood radius is the same for each sample, the feature selection algorithm outputs the neighborhood radius that makes the labels in the neighborhood formed of all samples as unique as possible.

For sample A, from Fig. 1(a), it can be seen that neighborhood 2 is the preferred neighborhood, but some problems still exist. For example, neighborhood 1 only contains sample A itself. This situation should be avoided as much as possible since the information obtained is very limited. Neighborhood 2 contains sample A and B, which is an ideal neighborhood. For the sample D, it can be seen that neighborhood 3 is the preferred neighborhood, and neighborhood 1 and neighborhood 2 are the situations that should be avoided.

However, when we consider the two samples at the same time, it is apparent that there exist some inevitable situations: (1) when the algorithm outputs neighborhood 1, the information of sample A and sample D can not be mined simultaneously, since the neighborhood of sample A and sample D only contains itself, respectively; (2) when the algorithm outputs neighborhood 2, the information of sample A can be mined, i.e., sample A belongs to the positive region, but the information of sample D can not be mined, because the neighborhood of sample D only contains itself; (3) when the algorithm outputs neighborhood 3, the information of sample A and sample D can be mined, i.e., sample D belongs to the positive region, but sample A does not belong to the positive region. These inevitable situations are caused by two reasons: (1) The empirical setting of neighborhood radius can not make sure that the neighborhood formed contains at least two samples in each round so that the information of certain samples can not be

mined surely; (2) In the step-by-step approach, the calculation of the neighborhood in each round is independent to each other, i.e., it can not guarantee that each sample belongs to the same positive region.

Therefore, for sample A and sample D, there is no chance to obtain the neighborhood preferred concurrently, because the neighborhood calculation in each step is independent of each other, and the empirical setting to neighborhood radius does not take into account the actual situation around each sample. Based on the above analysis, it can be seen that the step-by-step approach should be improved.

In summary, we need a novel method that satisfies the following conditions: (1) the neighborhood contains at least two samples; (2) only one iteration is required; (3) the labels of the neighborhood formed should be as uniform as possible. If we turn to the optimization perspective, the solution will be clearer. We may use condition (3) as the optimization target. However, it is necessary to quantify it. This paper defines the label rate to quantify the optimization target. Moreover, we may use conditions (1) and (2) as the constraints. In terms of condition (1), we consider the maximum and minimum distances between the target sample and other samples as the value range and select the neighborhood radius as the parameter. SSA is introduced to solve the optimization problem. The search approach of adaptive neighborhood radius is shown in Fig. 1(b).

In Fig. 1(b), for sample A, the search range of SSA is the difference between AB and AD, shown as the red dashed line. The available range, the neighborhood making the label rate 1, is the difference between AB and AC, shown as the blue dashed line, owing to sample C being the closest sample with diverse labels. Hence, for sample A, the algorithm would regard the sparrows as the neighborhood radius and search for the radii that make the label rate 1 in the search range. Subsequently, the algorithm would find that the radii in the available range meet the search target, i.e., the maximum label rate is 1. Consequently, the algorithm outputs the maximum label rate of sample A, which is 1.

3.2. SSA-based adaptive neighborhood rough sets

The application of SSA in searching for the adaptive neighborhood radius of each sample in a given dataset is divided into three parts, i.e., the number of sparrows, the search range of sparrows, and the value of fitness. To avoid the adaptive neighborhood radius of each sample searched by SSA being too small to involve any other sample in the neighborhood formed, in this paper, the search range is set to the maximum and minimum distances between the current sample and other samples. Since the number of sparrows is related to the search range, the more extensive the search range of sparrows, the larger the number. Since the search range is not large enough to utilize too many sparrows, in this paper, the number of sparrows is set to 30 empirically (the number of sparrows can be adjusted to other values according to actual situations). Moreover, to ensure the labels of the elements in the neighborhood of the target sample are the same as the sample possible, the maximum label rate is set to the fitness.

Due to the addition of decision attribute, the neighborhood information table can be represented as a neighborhood decision table $\langle U, C \cup D, Nei \rangle$.

Definition 1. Given a neighborhood decision table $\langle U, C \cup D, Nei \rangle$, for any attribute subset $subC \subseteq C$ and any sample $sp^i \in U$, let $\eta_{sp^i}^{subC} = \left\{ \eta_{(1,sp^i)}^{subC}, \eta_{(2,sp^i)}^{subC}, \cdots, \eta_{(t,sp^i)}^{subC} \right\}$ be the set of all candidate neighborhood radii of sp^i searched by SSA under $subC$ during iteration, where $\eta_{(w,sp^i)}^{subC} \in \eta_{sp^i}^{subC}$ represents the w^{th} $(1 \le w \le t)$ candidate neighborhood radius of sp^i under $subC$ during iteration. The label rate of sample sp^i under $subC$ with $\eta_{(w,sp^i)}^{subC}$ as the neighborhood radius can be defined as follows.

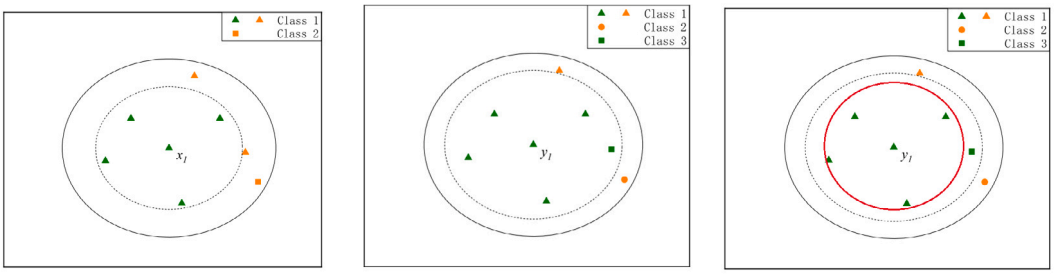
$$Labelrate^{subC}\left(sp^i, \eta_{(w,sp^i)}^{subC}\right) = \frac{Samelabel(\eta_{(w,sp^i)}^{subC}\left(sp^i\right))}{\left|\eta_{(w,sp^i)}^{subC}\left(sp^i\right)\right|}, \tag{7}$$

where $|x|$ represents the cardinality of set x, $\eta_{(w,sp^i)}^{subC}\left(sp^i\right)$ represents the neighborhood of sample sp^i with $\eta_{(w,sp^i)}^{subC}$ as the neighborhood radius, $Samelabel(\eta_{(w,sp^i)}^{subC}\left(sp^i\right))$ represents the number of occurrences of the label of sp^i in the neighborhood of sp^i.

From Definition 1, we can clearly see that the maximum value of label rate is 1 and the minimum value of label rate is larger than 0. Hence, the value range of label rate is $1 \geqslant Labelrate^{subC}\left(sp^i, \eta_{(w,sp^i)}^{subC}\right) > 0$.

For any sample $sp^i \in U$, if the label of the sample sp^i itself in the neighborhood of sp^i is uniform, then the label rate of sp^i takes the maximum value 1. In this case, the neighborhood of sp^i is the best case, without any adjustment. If the number of the label of the sample sp^i itself in the neighborhood of sp^i is only 1, then in this case, the neighborhood of sp^i is the worst case. The neighborhood radius will be adjusted by SSA, in order to maximize the label rate of sp^i, i.e., the label of sp^i in the neighborhood of sp^i takes the maximum value.

To maximize the attribute dependency of the final attribute group, during the process of neighborhood granulation, the label of the neighborhood of each sample should be the same as the sample itself as possible. On this basis, we propose the concept of label rate to calculate the percentage of the label of the sample sp^i in its neighborhood. The larger the label rate, the more elements with the same label as sample sp^i in its neighborhood, and the better the neighborhood radius. Therefore, the label rate is an excellent measure to evaluate the neighborhood radius.



(a) The situation that Zhang's approach can solve [26] (b) The situation that Zhang's approach cannot solve [26] (c) The approach proposed in this paper

Fig. 2. Example 2.

Example 2. To compare our approach with the approach proposed in [26], two figures in [26] (that is, Fig. 2(a) and Fig. 2(b)) are used to illustrate the situations that Zhang's approach can and cannot solve, respectively. Moreover, Fig. 2(c) is used to demonstrate the approach proposed in this paper. In Fig. 2, the solid black line denotes the initialized neighborhood, and the dashed black line denotes the neighborhood obtained by Zhang's approach. In Fig. 2(a), the label rate of the initialized neighborhood of sample x_1 is equal to 0.875, and it is clear from the figure that a reduced neighborhood can increase the label rate. Therefore, Zhang's approach generates a novel neighborhood, which increases the label rate to 1. In this case, the novel neighborhood of sample x_1 is the best. However, in Fig. 2(b), the label rate of the initialized neighborhood of sample y_1 is equal to 0.75, and it is clear from the figure that a reduced neighborhood can increase the label rate. Therefore, Zhang's approach generates another novel neighborhood, which increases the label rate to 0.8333. In this case, the novel neighborhood of sample y_1 is not the best. However, it is clear that continuously reducing the neighborhood can solve the above problem, as shown by the red line in Fig. 2(c). In this paper, the process of continuously reducing the neighborhood is achieved by SSA with the maximum value of label rate as the termination condition.

Definition 2. Given a neighborhood decision table $\langle U, C \cup D, Nei \rangle$, for any attribute subset $subC \subseteq C$ and any sample $sp^i \in U$, let $\eta^{subC}_{sp^i} = \left\{ \eta^{subC}_{(1,sp^i)}, \eta^{subC}_{(2,sp^i)}, \cdots, \eta^{subC}_{(t,sp^i)} \right\}$ be the set of all candidate neighborhood radii of sp^i searched by SSA under $subC$ during iteration, where $\eta^{subC}_{(w,sp^i)} \in \eta^{subC}_{sp^i}$ represents the w^{th} $(1 \le w \le t)$ candidate neighborhood radius of sp^i under $subC$ during iteration. Assume that $Labelrate^{subC}\left(sp^i, \eta^{subC}_{(w,sp^i)}\right)$ represents the label rate of sp^i under $subC$ with $\eta^{subC}_{(w,sp^i)}$ as the neighborhood radius, the adaptive neighborhood radius of sp^i under $subC$ can be defined as follows.

$$\eta^{subC}_{(adap,sp^i)} = arg_{\eta^{subC}_{(w,sp^i)} \in \eta^{subC}_{sp^i}} max\ Labelrate^{subC}\left(sp^i, \eta^{subC}_{(w,sp^i)}\right). \tag{8}$$

The selection of the adaptive neighborhood radius is the process of searching the neighborhood radius with the maximum label rate achieved by SSA. During the iteration, the neighborhood radius with the maximum label rate is selected via SSA until the global optimal neighborhood radius is selected.

Definition 3. Given a neighborhood decision table $\langle U, C \cup D, Nei \rangle$, for any attribute subset $subC \subseteq C$ and any two samples $sp^i, sp^j \in U$, let $\eta^{subC}_{(adap,sp^i)}$ be the adaptive neighborhood radius of sp^i under $subC$ during iteration, and let $MinD^{subC}\left(sp^i, sp^j\right)$ represent the Euclidean distance between sample sp^i and sample sp^j under attribute subset $subC$. The adaptive neighborhood of sp^i under $subC$ is defined as follows.

$$\eta^{subC}_{(adap,sp^i)}\left(sp^i\right) = \left\{ sp^j \in U \,\middle|\, MinD^{subC}\left(sp^i, sp^j\right) < \eta^{subC}_{(adap,sp^i)} \right\}. \tag{9}$$

Definition 4. Given a neighborhood decision table $\langle U, C \cup D, Nei \rangle$, for any attribute subset $subC \subseteq C$ and any sample $sp^i \in U$, let $\eta^{subC}_{(adap,sp^i)}$ be the adaptive neighborhood radius of sp^i under $subC$ during iteration, the lower approximation of D with respect to $subC$ is defined as follows.

$$\underline{AdapN}^{subC} D = \{ sp^i \in U \,|\, Labelrate^{subC}\left(sp^i, \eta^{subC}_{(adap,sp^i)}\right) = 1 \}. \tag{10}$$

$\underline{AdapN}^{subC} D$ is also called the positive region of D with respect to $subC$, denoted as $POS^{subC}(D)$.

Example 3. Table 1 is used to illustrate the solution for discrete attributes. Table 1 shows the four most common cases in datasets. First of all, all attributes are normalized into the interval [0,1], including discrete attributes. The results are shown in Table 1. Sample x^1 is used as an example. In case 1, no matter how the neighborhood radius is searched, the label rate of the formed

Table 1
The four most common cases in datasets (Example 3).

	case1		case2		case3		case4	
sample	attribute	label	attribute	label	attribute	label	attribute	label
x^1	0	1	0	0	0	0	0	0
x^2	0	1	0	0	0	0	0	0
x^3	0	1	0	0	0	0	0	1
x^4	0.5	1	0.5	1	0.5	0	0.5	1
x^5	0.5	1	0.5	1	0.5	1	0.5	1
x^6	0.5	1	0.5	1	0.5	1	0.5	1

neighborhood is always 1. In case 2, if the neighborhood radius η is equal to 0, then $\eta(x^1) = \{x^1, x^2, x^3\}$, and $Labelrate(x^1, \eta) = 1$. If the neighborhood radius η is equal to 0.5, then $\eta(x^1) = \{x^1, x^2, x^3, x^4, x^5, x^6\}$, and $Labelrate(x^1, \eta) = 0.5$. Therefore, in this case, the neighborhood radius η is set to 0 by SSA. In case 3, if the neighborhood radius η is equal to 0, then $\eta(x^1) = \{x^1, x^2, x^3\}$, and $Labelrate(x^1, \eta) = 1$. If the neighborhood radius η is equal to 0.5, then $\eta(x^1) = \{x^1, x^2, x^3, x^4, x^5, x^6\}$, and $Labelrate(x^1, \eta) = 0.6667$. Therefore, in this case, the neighborhood radius η is set to 0 by SSA. In case 4, if the neighborhood radius η is equal to 0, then $\eta(x^1) = \{x^1, x^2, x^3\}$, and $Labelrate(x^1, \eta) = 0.6667$. If the neighborhood radius η is equal to 0.5, then $\eta(x^1) = \{x^1, x^2, x^3, x^4, x^5, x^6\}$, and $Labelrate(x^1, \eta) = 0.3334$. So, in this case, the neighborhood radius η is set to 0 by SSA. Therefore, there exist no possibility to include the elements from other classes into the neighborhood by SSA. Because the distance between any two samples in the same class is equal to 0, if a certain element in another class is included in the neighborhood, the whole class is also included. In most cases, the label rate is worse than before. Therefore, SSA aims at the maximum of label rate to handle discrete attributes without additional settings.

Given a neighborhood decision table $\langle U, C \cup D, Nei \rangle$, for any attribute subset $subC \subseteq C$, the classical definition of the lower approximation of D with respect to $subC$ is based on the inclusion relation, i.e., the neighborhood of the sample should be completely included in a certain decision class. In this paper, the definition of the lower approximation of D with respect to $subC$ is based on the label rate. If the maximum value of the label rate is 1, then the neighborhood of the sample must be included in a certain decision class.

Definition 5. Given a neighborhood decision table $\langle U, C \cup D, Nei \rangle$, for any attribute subset $subC \subseteq C$, the attribute dependency of D on $subC$ is defined as follows.

$$\gamma(D, subC) = \frac{\left|POS^{subC}(D)\right|}{|U|}. \tag{11}$$

Definition 6. Given a neighborhood decision table $\langle U, C \cup D, Nei \rangle$, let $subC \subset C$ be an attribute subset of C, for any $c \in C - subC$, the significance of attribute c with respect to $subC$ and D is defined as follows.

$$Sig(c, subC, D) = \gamma(D, subC \cup \{c\}) - \gamma(D, subC). \tag{12}$$

It is easy to verify that $0 \leqslant Sig(c, subC, D) \leqslant 1$. If $Sig(c, subC, D) = 0$, then the attribute c is meaningless to $subC$, and if $Sig(c, subC, D) > 0$, then the attribute c is meaningful to $subC$. If $Sig(c, subC, D) > Sig(c', subC, D)$, then for $subC$, the attribute c is more important than the attribute c'.

3.3. Algorithm design

In this subsection, we propose a greedy feature selection algorithm using SSA-based adaptive neighborhood rough set model given in Section 3.1.

The time complexity of Algorithm 1 is given as follows. In the worst case, the time complexity of Step 6 is $O(|C| \times |U|^2)$, and the time complexity of Step 9 is $O(r \times |C| \times |U|^2)$, where r is the iterations in SSA. As for the step 15, its time complexity depends on the iterations of the whole algorithm. In the worst case, the algorithm selects all features, which indicates that the time complexity of step 15 is $O(\frac{(1+|C|)\times|C|}{2})$. Therefore, in the worst case, the time complexity of Algorithm 1 is $O(r \times |C| \times |U|^2)$. For comparison, we show that the time complexity of the classical neighborhood rough set model is $O(|C|^2 \times |U| log|U|)$.

4. Experimental analysis

In this section, the effectiveness of the algorithm proposed in this paper is verified by experiments.

4.1. The settings of experiments

In this subsection, the settings of experiments are introduced. We conducted the comparative experiments on seventeen datasets, where the datasets were taken from the UCI Machine Learning Repository (https://archive.ics.uci.edu/ml/index.php) and (https://

Algorithm 1 Feature selection algorithm using the adaptive neighborhood rough set model.

Input: A neighborhood decision table $\langle U, C \cup D, Nei \rangle$
Output: A reduct $subC$
1. $subC \leftarrow \emptyset, V \leftarrow U, A \leftarrow C$;
2. **For** $k = 1 : |A|$ **do**
3. $POS(D) \leftarrow \emptyset, subC' \leftarrow subC \cup \{a_k\}$;
4. **For** $i = 1 : |V|$ **do**
5. **For** $j = 1 : |U|$ **do**
6. Compute the distance between sample sp^i and sample sp^j;
7. Record the maximum and minimum distances $[min(sp^i), max(sp^i)]$;
8. **End For**
9. Use SSA to compute the adaptive neighborhood radius of sample sp^i;
10. Compute the adaptive neighborhood of sample sp^i;
11. **If** $max\ Labelrate^{subC'}\left(sp^i, \eta^{subC'}_{(w,sp^i)}\right) = 1$ **do**
12. $POS(D) \leftarrow POS(D) \cup \{sp^i\}$;
13. **End if**
14. **End for**
15. Compute $Sig\left(subC', subC, D\right) = \gamma\left(D, subC'\right) - \gamma\left(D, subC\right)$;
16. **End for**
17. Select $a' = \underset{a \in A - subC}{arg} \max Sig\left(a, subC, D\right)$;
18. **If** $Sig\left(a, subC, D\right) \leqslant 0$ **do**
19. **Return** $subC$;
20. **Else**
21. $A \leftarrow A - \{a'\}, subC \leftarrow subC \cup \{a'\}$;
22. Return to Step 2.

csse.szu.edu.cn/staff/zhuzx/datasets.html) [41]. The properties of the above datasets are shown in Table 2. All condition attributes were normalized in each dataset into the interval [0,1]. In terms of SSA, the number of iterations was set to 30. We compared the algorithm proposed in this paper (denoted by SSAANRS) with five existing algorithms. A brief description of the five algorithms is given as follows.

(1) Hu et al. [15] proposed a novel approach to handle heterogeneous data based on η-neighborhood rough sets, and applied it to attribute reduction. This algorithm (denoted by ENUM) used Euclidean distance as the distance measure and set the parameter δ to vary from 0.01 to 0.1 in steps of 0.01.

(2) Hu et al. [40] applied the K-nearest neighbor relation to neighborhood rough sets and proposed a novel attribute reduction algorithm. This algorithm (denoted by TNN) used Euclidean distance as the distance measure and set the parameter K to vary from 0.01 to 0.1 in steps of 0.01.

(3) Wang et al. [39] proposed the concept of maximal neighborhood discernibility and applied it to neighborhood rough sets. They proposed a feature selection algorithm based on maximal neighborhood discernibility. This algorithm (denoted by HARCD) used Chebyshev distance as the distance measure and set the parameter ε to vary from 0.01 to 0.1 in steps of 0.01.

(4) Wang et al. [19] proposed the K-nearest neighborhood rough set model based on δ-neighborhood rough sets and K-nearest neighbor relation, and applied it to attribute reduction. This algorithm (denoted by KNNRS) used Euclidean distance as the distance measure and set the parameter K to vary from 0.01N to 0.1N in steps of 0.01N, where N represents the number of samples.

(5) Qu et al. [27] proposed the adaptive neighborhood rough set model based on the label distribution near the sample, and used it to design a maximum relevance minimum redundancy-based feature selection algorithm. This algorithm (denoted by FSMRI) used Euclidean distance as the distance measure.

To evaluate the performance of different feature selection algorithms, the following two parts are adopted in the experiments: (1) evaluate the feature selection process; (2) evaluate the attribute groups selected. For Part (1), the running time is recorded for comparison. For Part (2), the KNN and Decision Tree algorithms in Matlab are used to evaluate the accuracies and balanced accuracies of the attribute groups selected by different feature selection algorithms. The parameter K in KNN is set to vary from 1 to 10, and the Gini index is used as the partition measure of the decision tree. The ten-fold cross validation is applied to each dataset when evaluating the accuracy and balanced accuracy of different attribute groups. The experiments are run in MATLAB2023a, and the hardware environment is as follows: Intel(R) Core(TM) i9-13900HX CPU @ 2.20 GHz with 32 GB of RAM.

4.2. The analysis of experiments

According to Table 3, the SSAANRS ranks third among the six algorithms compared. Regarding running time, the SSAANRS is less than TNN about two times, ENUM about four times, and HARCD about three times. From Table 4 and Table 5, it can be seen that SSAANRS is less than KNNRS and more than the other algorithms in the number of attributes selected, but there is not much diversity. Therefore, in the process of feature selection, SSAANRS shows better performance.

Fig. 3 describes the trends in the performance of the diverse algorithms on two classifiers KNN and Decision Tree. It is clear that SSAANRS outperforms other algorithms in most datasets, particularly in the case of the Decision Tree algorithm. Moreover, more details are perceived in Table 6 to Table 11. In terms of the accuracy and balanced accuracy in the case of the KNN classifier with diverse K, the SSAANRS proves to be the best, which outperforms in most datasets compared with other algorithms. For the decision tree classifier, both the accuracy and balanced accuracy, the SSAANRS has the best experiment results. In addition, from Fig. 4, the

Table 2
The properties of seventeen datasets.

No.	Datasets	Abbreviation	Samples	Features	Classes
1	plrx	plrx	182	12	2
2	ionosphere	ionosphere	351	33	2
3	parkinsons	parkinsons	195	21	2
4	heart	heart	270	13	2
5	wpbc	wpbc	194	21	2
6	audit	audit	776	26	2
7	breast cancer coimbra	bcc	116	9	2
8	Climate Model Simulation Crashes	cmsc	540	18	2
9	messidor feature	mf	1151	19	2
10	glass1	glass1	214	9	2
11	wine	wine	178	13	3
12	speaker accent recognition	sar	329	12	6
13	hill	hill	606	100	2
14	gait classification	gc	48	321	15
15	wdbc	wdbc	569	23	2
16	Toxicity	tc	171	1203	2
17	colon	colon	62	2000	2

Table 3
The running time of different algorithms.

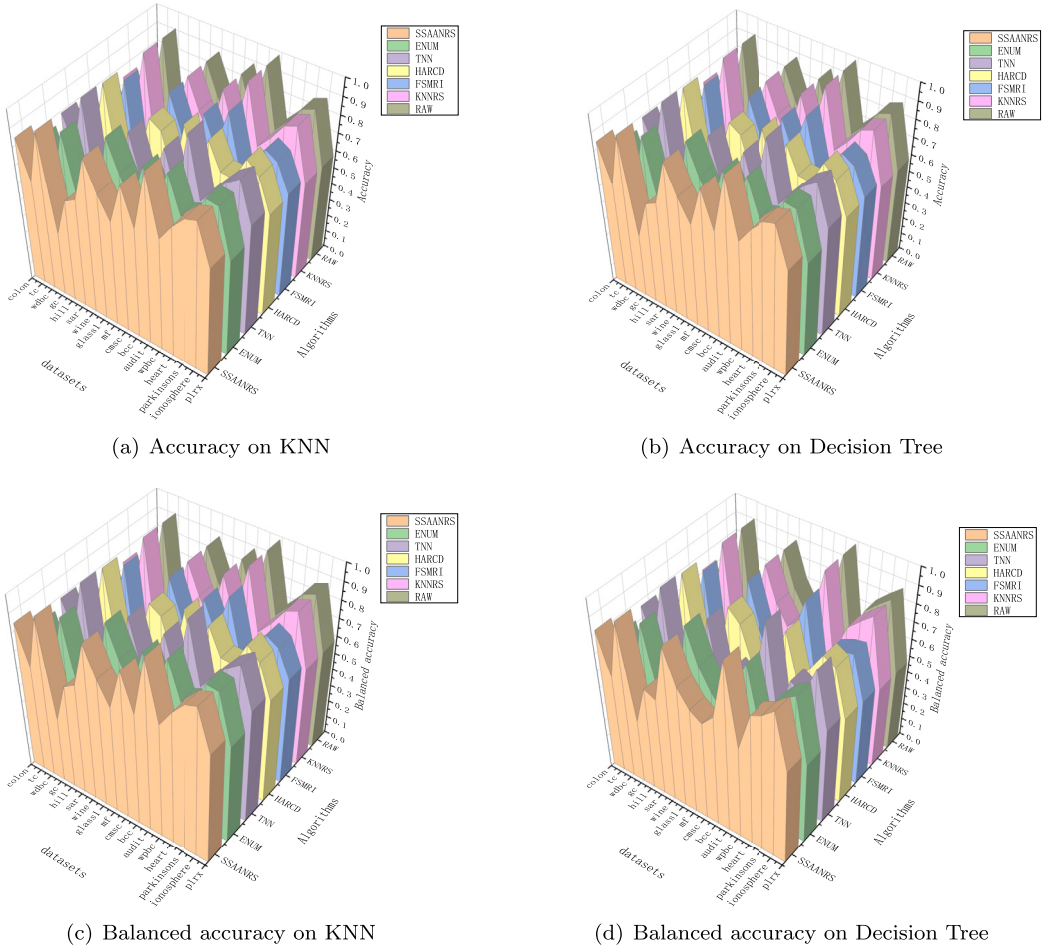
no	datasets	SSAANRS	TNN	ENUM	HARCD	FSMRI	KNNRS
1	plrx	2	15	27	16	**1**	18
2	ionosphere	133	899	1036	564	**1**	359
3	parkinsons	20	74	99	71	**1**	52
4	heart	20	68	112	45	**1**	43
5	wpbc	29	176	255	146	**1**	105
6	audit	658	1387	4060	2919	**2**	1370
7	bcc	**1**	6	7	3	**1**	5
8	cmsc	133	591	541	365	**1**	47
9	mf	556	1996	9049	4974	**2**	1296
10	glass1	4	20	53	29	**1**	8
11	wine	6	33	33	18	**1**	17
12	sar	21	88	113	57	**1**	19
13	hill	8991	13154	26912	18982	**11**	5801
14	gc	190	1228	1585	1193	**2**	392
15	wdbc	31	11	13	17	**1**	17
16	tc	519	3587	1365	3103	**204**	696
17	colon	362	803	1513	1767	**48**	227
	average	707.1	1458.3	2828.8	2031.4	**14.5**	640.3

Table 4
The number of attributes selected by different algorithms.

no	datasets	SSAANRS	TNN	ENUM	HARCD	FSMRI	KNNRS	RAW
1	plrx	**1**	**1**	3	2	**1**	12	12
2	ionosphere	4	3	3	2	**1**	33	33
3	parkinsons	5	**1**	3	5	2	22	22
4	heart	8	3	5	3	**2**	13	13
5	wpbc	3	**1**	2	2	2	33	33
6	audit	16	**1**	3	3	3	22	26
7	bcc	3	3	**2**	4	**2**	9	9
8	cmsc	5	2	2	3	**1**	18	18
9	mf	3	3	7	4	**2**	19	19
10	glass1	2	**1**	4	4	2	9	9
11	wine	5	**2**	3	4	**2**	13	13
12	sar	8	3	3	4	**1**	12	12
13	hill	10	**2**	3	7	7	100	100
14	gc	3	**1**	3	5	2	321	321
15	wdbc	5	3	3	6	**1**	23	23
16	tc	3	**2**	**2**	**2**	6	1206	1206
17	colon	4	4	6	2	**1**	2000	2000
	average	5.4	**1.9**	3.3	3.7	2.1	45.4	45.7

Table 5
The attributes selected by different algorithms.

no	datasets	SSAANRS	TNN	ENUM	HARCD	FSMRI	KNNRS	RAW
1	plrx	12	12	3,6,12	1,9	1	all	12
2	ionosphere	3,6,8,26	4,5,17	2,31,32	1,7	33	all	33
3	parkinsons	1,18,19,20,21	5	1,19,20	1,7,16,19,20	5,6	all	22
4	heart	1,2,3,5,9,11,12,13	3,10,13	3,4,9,10,11	7,10,12	8,13	all	13
5	wpbc	5,16,31	23	1,15	1,12	11,28	all	33
6	audit	11,12,13,14,15,16,17,18,19,20,21,22,23,24,25,26	26	14,19,22	2,4,14	4,14,18	Except for 2,4,13,25	26
7	bcc	1,4,8	1,4,8	1,3	1,2,3,7	1,3	all	9
8	cmsc	1,2,14,15,16	1, 3	2 ,14	1,2,6	3	all	18
9	mf	3,7,10	3,7,16	3,6,9,10,11,15,18	1,7,8,16	11,18	all	19
10	glass1	1,4	8	1,3,4,9	1,3,4,6	5,6	all	9
11	wine	1,4,5,7,11	7,11	7,10,13	1,6,8,13	7,10	all	13
12	sar	1,3,4,5,6,7,10,11	5,6,12	1,3,10	1,2,8,10	3	all	12
13	hill	1,2,5,16,18,44,52,72,89,100	25,73	2,14,15	22,43,58,72,74,83,89	32,42,55,64,65,67,71	all	100
14	gc	149,259,267	302	100,221,255	163,192,223,281,291	94,110	all	321
15	wdbc	14,23,24,25,28	23,26,29	22,23,30	3,4,21,22,23,24	8	all	23
16	tc	430,758,1104	430,442	517,571	41,464	67,204,264,343,798,1035	all	1206
17	colon	95,258,1047,1411	95,258,1047,1411	1293,1673	72,781,1187,1231,1241,1539	1909	all	2000



(a) Accuracy on KNN

(b) Accuracy on Decision Tree

(c) Balanced accuracy on KNN

(d) Balanced accuracy on Decision Tree

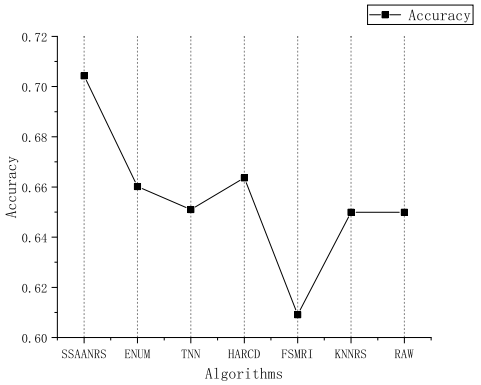
Fig. 3. The performance of different algorithms on two classifiers.

SSAANRS owns the best average performance. Hence, we can obtain that the SSAANRS performs stably and robustly. Interestingly, in some datasets, while the SSAANRS ranks not first in accuracy, in balanced accuracy it does. In addition, this phenomenon occurs in both of the classifiers. By contrast, other algorithms perform well in accuracy but not in balanced accuracy. This indicates that our method performs well in all decision classes, and the adaptive approach is necessary in some situations with complex data distribution.

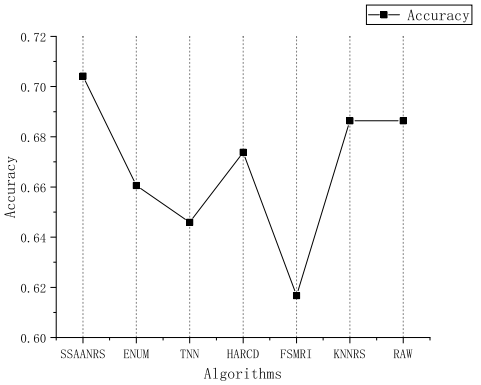
The Wilcoxon pairwise test is applied to compare the experimental results and test whether a significant diversity exists between the SSAANRS and other algorithms in classification performance [1]. Here, we set the threshold as 0.05. From Table 12, in light of the decision tree classifier, we can see that all the P-values are lower than 0.05 except for ENUM and HARCD in the accuracy measure. Besides, all P-values are lower than 0.05 in balanced accuracy except for HARCD, KNNRS, and RAW. Hence, for the decision tree classifier, we can obtain that the SSAANRS is significantly diverse from TNN, FSMRI, KNNRS, and RAW in accuracy and ENUM, TNN, and FSMRI in balanced accuracy. In light of the KNN classifier, it can be seen that all the P-values are lower than 0.05 except for ENUM, TNN, and HARCD in the accuracy measure. In addition, all P-values are lower than 0.05 in balanced accuracy except for ENUM, TNN, and HARCD. Therefore, for the KNN classifier, the results show that the SSAANRS is significantly different from FSMRI, KNNRS and RAW in both accuracy and balanced accuracy. Consequently, the statistical test shows the SSAANRS has a good classification performance.

5. Conclusions and future work

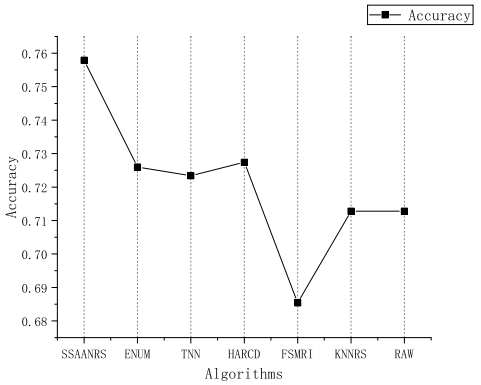
This paper creatively transforms three problems in the classical neighborhood construction approach into an optimization problem. In order to introduce SSA to solve the optimization problem, the label rate is designed and regarded as the fitness function. Driven by SSA, the adaptive neighborhood construction approach ignores the drawbacks of the classical approach and searches the surroundings of the sample, aiming to achieve the target of maxing the label rate. The adaptive neighborhood rough set model is constructed and a novel feature selection algorithm is proposed. The experiments indicate that the proposed algorithm can select the feature subsets with the optimal performance, providing a novel attempt to address the parameter problem in rough set theory from the optimization perspective. However, the high running time also reveals that the adaptive granulation to all samples remains in need of improvement.



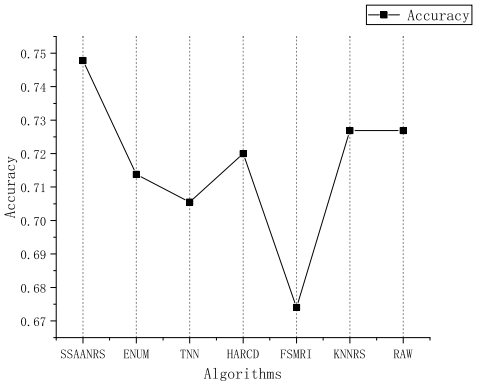
(a) Balanced accuracy on KNN



(b) Balanced accuracy on Decision Tree



(c) Accuracy on KNN



(d) Accuracy on Decision Tree

Fig. 4. The averaged performance of different algorithms on two classifiers.

Table 6
The accuracy of different algorithm on KNN classifier.

no	datasets	algorithm	K=1	K=2	K=3	K=4	K=5	K=6	K=7	K=8	K=9	K=10	average
		SSAANRS	0.6758	0.6533	0.6593	0.6863	0.6918	0.6995	0.7016	0.7049	0.7143	0.6940	**0.6881**
		ENUM	0.5824	0.6374	0.6016	0.6577	0.6412	0.6676	0.6401	0.6731	0.6731	0.6824	0.6457
		TNN	0.6643	0.6577	0.6385	0.6786	0.6962	0.7027	0.6973	0.7104	0.7187	0.6912	0.6855
1	plrx	HARCD	0.6297	0.6725	0.6473	0.6615	0.6456	0.6560	0.6478	0.6808	0.6868	0.6940	0.6622
		FSMRI	0.5571	0.6319	0.5901	0.6505	0.6330	0.6610	0.6577	0.6659	0.6725	0.6874	0.6407
		KNNRS	0.6407	0.6676	0.6505	0.6692	0.6632	0.6549	0.6445	0.6714	0.6692	0.6808	0.6612
		RAW	0.6407	0.6676	0.6505	0.6692	0.6632	0.6549	0.6445	0.6714	0.6692	0.6808	0.6612
		SSAANRS	0.8855	0.8678	0.8781	0.8789	0.8772	0.8801	0.8726	0.8718	0.8595	0.8556	0.8727
		ENUM	0.8812	0.8601	0.8638	0.8413	0.8259	0.8302	0.8376	0.8328	0.8293	0.8308	0.8433
		TNN	0.8880	0.8875	0.9205	0.9242	0.9211	0.9188	0.9137	0.9128	0.9085	0.8966	**0.9092**
2	ionosphere	HARCD	0.7738	0.7818	0.8125	0.8085	0.8251	0.8197	0.8293	0.8265	0.8299	0.8288	0.8136
		FSMRI	0.7396	0.7328	0.8123	0.8137	0.8425	0.8416	0.8444	0.8427	0.8433	0.8399	0.8153
		KNNRS	0.8641	0.8689	0.8450	0.8459	0.8422	0.8373	0.8311	0.8319	0.8311	0.8382	0.8436
		RAW	0.8641	0.8689	0.8450	0.8459	0.8422	0.8373	0.8311	0.8319	0.8311	0.8382	0.8436
		SSAANRS	0.9031	0.8923	0.8821	0.8872	0.8923	0.8744	0.8923	0.8718	0.8744	0.8923	**0.8862**
		ENUM	0.8846	0.8805	0.8733	0.8687	0.8733	0.8585	0.8723	0.8667	0.8733	0.8708	0.8722
		TNN	0.8359	0.8226	0.8149	0.7769	0.7728	0.7990	0.8359	0.8359	0.8359	0.8359	0.8166
3	parkinsons	HARCD	0.8667	0.8667	0.8964	0.8887	0.8944	0.8846	0.8923	0.8872	0.8872	0.8918	0.8856
		FSMRI	0.6979	0.6949	0.7554	0.7492	0.7415	0.7477	0.7410	0.7503	0.7533	0.7513	0.7383
		KNNRS	0.8467	0.8221	0.8477	0.8318	0.8533	0.8410	0.8385	0.8190	0.8267	0.8128	0.8339
		RAW	0.8467	0.8221	0.8477	0.8318	0.8533	0.8410	0.8385	0.8190	0.8267	0.8128	0.8339

(*continued on next page*)

Table 6 (*continued*)

no	datasets	algorithm	K=1	K=2	K=3	K=4	K=5	K=6	K=7	K=8	K=9	K=10	average
		SSAANRS	0.6537	0.6507	0.6689	0.6819	0.6930	0.6881	0.6844	0.6485	0.6496	0.6278	0.6647
		ENUM	0.6689	0.6837	0.7122	0.7004	0.7148	0.7037	0.7111	0.6981	0.6993	0.6952	0.6987
		TNN	0.7300	0.7541	0.7641	0.7741	0.7556	0.7493	0.7626	0.7767	0.7752	0.7785	**0.7620**
4	heart	HARCD	0.7137	0.6893	0.7641	0.7300	0.7493	0.7467	0.7496	0.7519	0.7574	0.7559	0.7408
		FSMRI	0.6526	0.6311	0.7293	0.7274	0.7441	0.7300	0.7400	0.7348	0.7385	0.7319	0.7160
		KNNRS	0.5815	0.6185	0.6530	0.6578	0.6611	0.6522	0.6700	0.6648	0.6596	0.6622	0.6481
		RAW	0.5815	0.6185	0.6530	0.6578	0.6611	0.6522	0.6700	0.6648	0.6596	0.6622	0.6481
		SSAANRS	0.6392	0.6586	0.7113	0.7165	0.7216	0.7113	0.7423	0.7680	0.7629	0.7629	0.7195
		ENUM	0.6758	0.6892	0.7309	0.7253	0.7345	0.7278	0.7309	0.7263	0.7392	0.7402	**0.7220**
		TNN	0.6763	0.6665	0.7052	0.7165	0.7438	0.7356	0.7299	0.7216	0.7479	0.7490	0.7192
5	wpbc	HARCD	0.7108	0.6686	0.6892	0.6881	0.7067	0.6866	0.7253	0.7309	0.7546	0.7418	0.7103
		FSMRI	0.6912	0.6990	0.7046	0.6814	0.7098	0.7211	0.7443	0.7464	0.7526	0.7454	0.7196
		KNNRS	0.6876	0.6670	0.7160	0.6995	0.7108	0.7010	0.7320	0.7330	0.7387	0.7438	0.7129
		RAW	0.6876	0.6670	0.7160	0.6995	0.7108	0.7010	0.7320	0.7330	0.7387	0.7438	0.7129
		SSAANRS	0.9907	0.9910	0.9919	0.9906	0.9905	0.9885	0.9885	0.9881	0.9871	0.9867	0.9894
		ENUM	0.8988	0.9053	0.8974	0.8957	0.8996	0.8921	0.8863	0.8872	0.8836	0.8823	0.8929
		TNN	0.9999	0.9991	0.9987	0.9985	0.9986	0.9973	0.9982	0.9981	0.9972	0.9974	**0.9983**
6	audit	HARCD	0.7869	0.7848	0.8335	0.8231	0.8282	0.8296	0.8456	0.8365	0.8369	0.8259	0.8231
		FSMRI	0.9353	0.9289	0.9367	0.9339	0.9356	0.9304	0.9291	0.9281	0.9276	0.9280	0.9314
		KNNRS	0.9698	0.9682	0.9684	0.9678	0.9719	0.9631	0.9624	0.9575	0.9522	0.9488	0.9630
		RAW	0.9698	0.9682	0.9684	0.9678	0.9719	0.9631	0.9624	0.9575	0.9522	0.9488	0.9630
		SSAANRS	0.6897	0.7328	0.7586	0.7328	0.7500	0.7328	0.7129	0.7155	0.7328	0.7336	0.7292
		ENUM	0.7095	0.7129	0.7069	0.6940	0.7034	0.6871	0.6957	0.7095	0.7310	0.7224	0.7072
		TNN	0.6819	0.6819	0.7138	0.7017	0.7086	0.7138	0.7336	0.7181	0.7293	0.7345	0.7117
7	bcc	HARCD	0.7147	0.7009	0.7509	0.7371	0.7776	0.7828	0.7733	0.7595	0.7733	0.7612	**0.7531**
		FSMRI	0.7103	0.7000	0.7164	0.7052	0.7095	0.6922	0.6948	0.7034	0.7328	0.7284	0.7093
		KNNRS	0.5922	0.5922	0.5388	0.5103	0.5414	0.5172	0.5121	0.5043	0.4853	0.4966	0.5291
		RAW	0.5922	0.5922	0.5388	0.5103	0.5414	0.5172	0.5121	0.5043	0.4853	0.4966	0.5291
		SSAANRS	0.9239	0.9187	0.9370	0.9335	0.9315	0.9269	0.9330	0.9302	0.9250	0.9287	**0.9288**
		ENUM	0.8691	0.8581	0.8835	0.8774	0.9022	0.9004	0.9089	0.9069	0.9104	0.9074	0.8924
		TNN	0.8565	0.8528	0.8985	0.9013	0.9070	0.9050	0.9093	0.9107	0.9146	0.9141	0.8970
8	cmsc	HARCD	0.8894	0.8889	0.9093	0.9085	0.9083	0.9120	0.9143	0.9137	0.9174	0.9141	0.9076
		FSMRI	0.8465	0.8335	0.8980	0.8978	0.9089	0.9102	0.9148	0.9148	0.9148	0.9148	0.8954
		KNNRS	0.8876	0.8874	0.9213	0.9196	0.9293	0.9306	0.9315	0.9280	0.9224	0.9209	0.9179
		RAW	0.8876	0.8874	0.9213	0.9196	0.9293	0.9306	0.9315	0.9280	0.9224	0.9209	0.9179
		SSAANRS	0.6334	0.6247	0.6342	0.6360	0.6481	0.6681	0.6681	0.6620	0.6881	0.6820	**0.6545**
		ENUM	0.6382	0.6375	0.6273	0.6377	0.6507	0.6622	0.6615	0.6685	0.6648	0.6669	0.6515
		TNN	0.6221	0.6419	0.6430	0.6568	0.6493	0.6566	0.6535	0.6606	0.6599	0.6705	0.6514
9	mf	HARCD	0.5703	0.5581	0.5662	0.5600	0.5680	0.5683	0.5692	0.5648	0.5698	0.5620	0.5657
		FSMRI	0.5362	0.5146	0.5271	0.5184	0.5259	0.5107	0.5148	0.4959	0.5203	0.5067	0.5171
		KNNRS	0.6203	0.6282	0.6380	0.6446	0.6461	0.6513	0.6658	0.6692	0.6638	0.6616	0.6489
		RAW	0.6203	0.6282	0.6380	0.6446	0.6461	0.6513	0.6658	0.6692	0.6638	0.6616	0.6489
		SSAANRS	0.6346	0.6220	0.6864	0.6706	0.7023	0.6869	0.6850	0.6935	0.6925	0.6977	0.6771
		ENUM	0.7706	0.7388	0.7668	0.7607	0.7724	0.7804	0.7879	0.7804	0.7832	0.7864	0.7728
		TNN	0.6201	0.6294	0.6299	0.6364	0.6299	0.6355	0.6379	0.6430	0.6449	0.6449	0.6352
10	glass1	HARCD	0.8005	0.7888	0.7855	0.7944	0.7972	0.7893	0.7850	0.7879	0.7827	0.7846	0.7896
		FSMRI	0.6164	0.6350	0.6808	0.6710	0.6930	0.6911	0.6935	0.6874	0.6841	0.6794	0.6732
		KNNRS	0.8561	0.8248	0.8028	0.7925	0.8051	0.7995	0.7953	0.7893	0.7776	0.7799	**0.8023**
		RAW	0.8561	0.8248	0.8028	0.7925	0.8051	0.7995	0.7953	0.7893	0.7776	0.7799	**0.8023**

In the future, we will explore more efficient search strategies to improve the performance of the adaptive approach further. We will also discuss the expansion of the adaptive approach to multi-granulation knowledge discovery to achieve the no-parameterization.

CRediT authorship contribution statement

Caihui Liu: Writing – review & editing, Validation, Supervision, Methodology, Conceptualization. **Bowen Lin:** Writing – original draft, Validation, Software, Methodology, Data curation, Conceptualization. **Duoqian Miao:** Writing – review & editing, Supervision.

Declaration of competing interest

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

Table 7
The accuracy of different algorithm on KNN classifier (Continued).

no	datasets	algorithm	K=1	K=2	K=3	K=4	K=5	K=6	K=7	K=8	K=9	K=10	average
		SSAANRS	0.9388	0.9253	0.9309	0.9326	0.9365	0.9264	0.9303	0.9287	0.9298	0.9281	**0.9307**
		ENUM	0.7663	0.7298	0.7365	0.7236	0.7146	0.7056	0.7017	0.7051	0.6753	0.6736	0.7132
		TNN	0.8854	0.8640	0.8787	0.8410	0.8393	0.8393	0.8438	0.8427	0.8393	0.8444	0.8518
11	wine	HARCD	0.7140	0.7140	0.6983	0.7051	0.6899	0.6736	0.6708	0.6916	0.6685	0.6770	0.6903
		FSMRI	0.9292	0.9287	0.9326	0.9140	0.9079	0.9169	0.9135	0.9219	0.9169	0.9230	0.9204
		KNNRS	0.7551	0.7579	0.7303	0.7135	0.7096	0.6927	0.7090	0.7169	0.7174	0.7140	0.7216
		RAW	0.7551	0.7579	0.7303	0.7135	0.7096	0.6927	0.7090	0.7169	0.7174	0.7140	0.7216
		SSAANRS	0.7304	0.7176	0.7429	0.7307	0.7252	0.7343	0.7438	0.7210	0.7271	0.7301	0.7303
		ENUM	0.5514	0.5617	0.5708	0.5818	0.5778	0.5717	0.5687	0.5705	0.5857	0.5906	0.5731
		TNN	0.4809	0.5255	0.5608	0.5337	0.5441	0.5307	0.5392	0.5334	0.5389	0.5416	0.5329
12	sar	HARCD	0.6018	0.5778	0.6158	0.6240	0.6343	0.6271	0.6319	0.6286	0.6304	0.6255	0.6197
		FSMRI	0.3422	0.3301	0.3720	0.4046	0.4195	0.4173	0.4210	0.4119	0.4043	0.4298	0.3953
		KNNRS	0.8055	0.7538	0.7903	0.8128	0.8170	0.8188	0.8155	0.8100	0.7991	0.7897	**0.8013**
		RAW	0.8055	0.7538	0.7903	0.8128	0.8170	0.8188	0.8155	0.8100	0.7991	0.7897	**0.8013**
		SSAANRS	0.6700	0.5914	0.5955	0.5853	0.6038	0.5848	0.5698	0.5635	0.5645	0.5545	**0.5883**
		ENUM	0.5581	0.5490	0.5535	0.5559	0.5604	0.5475	0.5611	0.5556	0.5611	0.5540	0.5556
		TNN	0.5703	0.5589	0.5774	0.5525	0.5686	0.5559	0.5551	0.5462	0.5300	0.5442	0.5559
13	hill	HARCD	0.6248	0.5686	0.5746	0.5596	0.5652	0.5441	0.5348	0.5322	0.5315	0.5272	0.5563
		FSMRI	0.5356	0.5370	0.5360	0.5368	0.5348	0.5360	0.5348	0.5462	0.5480	0.5404	0.5386
		KNNRS	0.5944	0.5616	0.5462	0.5513	0.5612	0.5535	0.5536	0.5375	0.5396	0.5414	0.5540
		RAW	0.5944	0.5616	0.5462	0.5513	0.5612	0.5535	0.5536	0.5375	0.5396	0.5414	0.5540
		SSAANRS	0.4688	0.5625	0.5500	0.4833	0.4438	0.4417	0.4292	0.3979	0.3688	0.4354	0.4581
		ENUM	0.7000	0.8104	0.7229	0.6604	0.6021	0.5854	0.5646	0.5625	0.5021	0.4313	0.6142
		TNN	0.1333	0.2708	0.2396	0.1313	0.1417	0.1375	0.1792	0.1813	0.1750	0.1583	0.1748
14	gc	HARCD	0.6958	0.9021	0.7438	0.7167	0.6833	0.6646	0.5313	0.5208	0.4250	0.3563	**0.6240**
		FSMRI	0.0854	0.1667	0.2271	0.1917	0.2375	0.1583	0.1542	0.1792	0.2229	0.2521	0.1875
		KNNRS	0.2292	0.2958	0.3375	0.2604	0.1875	0.1708	0.1583	0.1417	0.1458	0.1625	0.2090
		RAW	0.2292	0.2958	0.3375	0.2604	0.1875	0.1708	0.1583	0.1417	0.1458	0.1625	0.2090
		SSAANRS	0.8840	0.8907	0.8974	0.9000	0.9042	0.9054	0.8996	0.9070	0.9107	0.9148	0.9014
		ENUM	0.8213	0.8353	0.8606	0.8634	0.8735	0.8714	0.8796	0.8731	0.8677	0.8664	0.8612
		TNN	0.9190	0.9132	0.9327	0.9332	0.9394	0.9371	0.9411	0.9411	0.9413	0.9394	**0.9337**
15	wdbc	HARCD	0.9105	0.9139	0.9237	0.9271	0.9304	0.9276	0.9306	0.9311	0.9311	0.9299	0.9256
		FSMRI	0.8434	0.8710	0.8910	0.8917	0.9070	0.9011	0.9130	0.9067	0.9086	0.9070	0.8941
		KNNRS	0.9156	0.9223	0.9251	0.9236	0.9327	0.9299	0.9318	0.9293	0.9329	0.9308	0.9274
		RAW	0.9156	0.9223	0.9251	0.9236	0.9327	0.9299	0.9318	0.9293	0.9329	0.9308	0.9274
		SSAANRS	0.6901	0.6316	0.6000	0.5930	0.5936	0.6374	0.6433	0.6257	0.6257	0.6491	0.6289
		ENUM	0.5450	0.5526	0.5889	0.5515	0.6064	0.5930	0.6275	0.6310	0.6503	0.6409	0.5987
		TNN	0.6374	0.6140	0.6140	0.6199	0.6199	0.6363	0.6257	0.6292	0.6433	0.6433	0.6283
16	tc	HARCD	0.6316	0.6023	0.6023	0.6082	0.6199	0.6199	0.6404	0.6398	0.6906	0.6737	0.6329
		FSMRI	0.5965	0.5556	0.6374	0.6374	0.6608	0.6608	0.6719	0.6766	0.6433	0.6374	**0.6378**
		KNNRS	0.4901	0.4719	0.5205	0.5532	0.5620	0.5550	0.5760	0.5836	0.5667	0.5766	0.5456
		RAW	0.4901	0.4719	0.5205	0.5532	0.5620	0.5550	0.5760	0.5836	0.5667	0.5766	0.5456
		SSAANRS	0.9000	0.8226	0.8161	0.8032	0.8484	0.8452	0.8387	0.8242	0.8323	0.8387	**0.8369**
		ENUM	0.7339	0.7548	0.7016	0.7403	0.7371	0.7290	0.7177	0.7242	0.7065	0.7274	0.7273
		TNN	0.9032	0.8290	0.8097	0.7968	0.8371	0.8355	0.8387	0.8226	0.8371	0.8274	0.8337
17	colon	HARCD	0.7161	0.7129	0.7000	0.6613	0.6484	0.6694	0.6581	0.6516	0.6016	0.6306	0.6650
		FSMRI	0.6452	0.6855	0.7000	0.7097	0.7419	0.7371	0.7548	0.7371	0.7565	0.7548	0.7223
		KNNRS	0.7984	0.8258	0.8500	0.7887	0.8387	0.7984	0.7742	0.7742	0.7726	0.7613	0.7982
		RAW	0.7984	0.8258	0.8500	0.7887	0.8387	0.7984	0.7742	0.7742	0.7726	0.7613	0.7982

Table 8
The balanced accuracy of different algorithm on KNN classifier.

no	datasets	algorithm	K=1	K=2	K=3	K=4	K=5	K=6	K=7	K=8	K=9	K=10	average
		SSAANRS	0.6063	0.5287	0.5087	0.5319	0.5656	0.5329	0.5325	0.5294	0.5225	0.4946	0.5353
		ENUM	0.4948	0.4869	0.4733	0.4977	0.4890	0.4848	0.4679	0.4846	0.4794	0.4887	0.4847
		TNN	0.5960	0.5275	0.5133	0.5325	0.5719	0.5344	0.5358	0.5240	0.5279	0.4937	**0.5357**
1	plrx	HARCD	0.5427	0.5308	0.5448	0.5471	0.5238	0.5040	0.5160	0.5065	0.5231	0.5202	0.5259
		FSMRI	0.4706	0.4850	0.4565	0.4762	0.4696	0.4885	0.4898	0.4900	0.4827	0.4852	0.4794
		KNNRS	0.5423	0.5202	0.4862	0.4925	0.4898	0.4829	0.4556	0.4787	0.4783	0.4802	0.4907
		RAW	0.5423	0.5202	0.4862	0.4925	0.4898	0.4829	0.4556	0.4787	0.4783	0.4802	0.4907

(*continued on next page*)

Table 8 (*continued*)

no	datasets	algorithm	K=1	K=2	K=3	K=4	K=5	K=6	K=7	K=8	K=9	K=10	average
		SSAANRS	0.8717	0.8576	0.8509	0.8522	0.8463	0.8462	0.8414	0.8290	0.8217	0.8148	0.8432
		ENUM	0.8770	0.8547	0.8552	0.8286	0.8135	0.8082	0.8040	0.8040	0.7929	0.7920	0.8230
		TNN	0.8811	0.8840	0.9064	0.9055	0.9067	0.9019	0.8965	0.8884	0.8811	0.8690	**0.8921**
2	ionosphere	HARCD	0.7501	0.7685	0.7932	0.7915	0.7965	0.8000	0.8069	0.7996	0.7956	0.8013	0.7903
		FSMRI	0.7303	0.7160	0.7833	0.7829	0.8045	0.8047	0.8113	0.8063	0.8082	0.8048	0.7852
		KNNRS	0.8208	0.8218	0.7931	0.7989	0.7869	0.7937	0.7754	0.7756	0.7746	0.7800	0.7921
		RAW	0.8208	0.8218	0.7931	0.7989	0.7869	0.7937	0.7754	0.7756	0.7746	0.7800	0.7921
		SSAANRS	0.8607	0.8522	0.8096	0.8436	0.8165	0.8180	0.7935	0.7935	0.7968	0.8083	0.8193
		ENUM	0.8303	0.8504	0.8130	0.8319	0.7933	0.8067	0.7891	0.8024	0.7923	0.8039	0.8113
		TNN	0.7007	0.6972	0.6996	0.6991	0.6779	0.6928	0.7017	0.7017	0.7017	0.7017	0.6974
3	parkinsons	HARCD	0.8334	0.8677	0.8528	0.8595	0.8314	0.8281	0.8365	0.8445	0.8266	0.8233	**0.8404**
		FSMRI	0.5478	0.6227	0.6360	0.6577	0.6077	0.6678	0.6417	0.6669	0.6444	0.6753	0.6368
		KNNRS	0.8000	0.8047	0.7786	0.7712	0.7675	0.7518	0.7287	0.7386	0.7061	0.6966	0.7544
		RAW	0.8000	0.8047	0.7786	0.7712	0.7675	0.7518	0.7287	0.7386	0.7061	0.6966	0.7544
		SSAANRS	0.6544	0.6334	0.6611	0.6621	0.6876	0.6767	0.6710	0.6276	0.6360	0.5948	0.6505
		ENUM	0.6591	0.6581	0.6997	0.6813	0.7097	0.6838	0.6962	0.6736	0.7008	0.6783	0.6841
		TNN	0.7158	0.7338	0.7493	0.7633	0.7433	0.7346	0.7583	0.7621	0.7652	0.7665	**0.7492**
4	heart	HARCD	0.6986	0.6677	0.7512	0.7133	0.7498	0.7276	0.7369	0.7354	0.7445	0.7350	0.7260
		FSMRI	0.6298	0.6048	0.7246	0.7111	0.7403	0.7235	0.7418	0.7224	0.7349	0.7213	0.7054
		KNNRS	0.5703	0.5920	0.6523	0.6363	0.6590	0.6365	0.6593	0.6419	0.6440	0.6376	0.6329
		RAW	0.5703	0.5920	0.6523	0.6363	0.6590	0.6365	0.6593	0.6419	0.6440	0.6376	0.6329
		SSAANRS	0.5126	0.5257	0.5214	0.5305	0.4923	0.4823	0.4955	0.5000	0.5049	0.5035	0.5069
		ENUM	0.5575	0.5733	0.5947	0.5938	0.5636	0.5783	0.5608	0.5568	0.5475	0.5626	**0.5689**
		TNN	0.5272	0.5656	0.5466	0.5584	0.5515	0.5270	0.4963	0.4907	0.4990	0.4947	0.5257
5	wpbc	HARCD	0.6088	0.5843	0.5462	0.5572	0.5018	0.5081	0.5144	0.5336	0.5599	0.5501	0.5464
		FSMRI	0.5519	0.6017	0.5017	0.4874	0.4873	0.4829	0.4935	0.4914	0.4926	0.4845	0.5075
		KNNRS	0.6009	0.5639	0.5693	0.5495	0.5290	0.5145	0.4964	0.5038	0.4918	0.4904	0.5310
		RAW	0.6009	0.5639	0.5693	0.5495	0.5290	0.5145	0.4964	0.5038	0.4918	0.4904	0.5310
		SSAANRS	0.9891	0.9890	0.9899	0.9876	0.9881	0.9871	0.9855	0.9850	0.9845	0.9831	0.9869
		ENUM	0.9011	0.8995	0.9032	0.9073	0.9110	0.9067	0.9030	0.8997	0.8977	0.8983	0.9028
		TNN	0.9995	0.9995	0.9982	0.9984	0.9977	0.9980	0.9972	0.9969	0.9966	0.9962	**0.9978**
6	audit	HARCD	0.7802	0.7860	0.8284	0.8079	0.8166	0.8182	0.8244	0.8189	0.8185	0.8060	0.8105
		FSMRI	0.9329	0.9249	0.9393	0.9369	0.9365	0.9355	0.9333	0.9337	0.9319	0.9344	0.9339
		KNNRS	0.9648	0.9615	0.9603	0.9601	0.9621	0.9584	0.9531	0.9457	0.9378	0.9363	0.9540
		RAW	0.9648	0.9615	0.9603	0.9601	0.9621	0.9584	0.9531	0.9457	0.9378	0.9363	0.9540
		SSAANRS	0.6725	0.7119	0.7013	0.6933	0.7073	0.7047	0.7225	0.7199	0.7252	0.7209	0.7080
		ENUM	0.6996	0.7159	0.7088	0.6853	0.6866	0.6745	0.6796	0.7012	0.7275	0.7206	0.7000
		TNN	0.6681	0.6934	0.6977	0.6974	0.7036	0.7033	0.7136	0.7100	0.7329	0.7338	0.7054
7	bcc	HARCD	0.7247	0.6984	0.7474	0.7469	0.7725	0.7607	0.7666	0.7526	0.7600	0.7665	**0.7496**
		FSMRI	0.7121	0.7328	0.7102	0.6880	0.7017	0.6903	0.6889	0.7127	0.7231	0.7157	0.7076
		KNNRS	0.5776	0.5883	0.5267	0.5150	0.4830	0.5142	0.5031	0.4859	0.4457	0.4811	0.5121
		RAW	0.5776	0.5883	0.5267	0.5150	0.4830	0.5142	0.5031	0.4859	0.4457	0.4811	0.5121
		SSAANRS	0.7214	0.7607	0.7497	0.7504	0.7079	0.7150	0.6919	0.6763	0.6489	0.6648	**0.7087**
		ENUM	0.5520	0.5697	0.5120	0.5145	0.5180	0.5195	0.5113	0.5118	0.5057	0.5081	0.5223
		TNN	0.5503	0.5402	0.5218	0.5295	0.5149	0.5244	0.5016	0.5122	0.4999	0.4996	0.5194
8	cmsc	HARCD	0.6275	0.6786	0.6332	0.6595	0.5944	0.6305	0.5997	0.6081	0.5830	0.5924	0.6207
		FSMRI	0.4878	0.5084	0.4990	0.5022	0.4970	0.4976	0.5000	0.5000	0.5000	0.5000	0.4992
		KNNRS	0.6043	0.6443	0.6018	0.6247	0.6158	0.6098	0.6012	0.5731	0.5631	0.5446	0.5983
		RAW	0.6043	0.6443	0.6018	0.6247	0.6158	0.6098	0.6012	0.5731	0.5631	0.5446	0.5983
		SSAANRS	0.6196	0.6234	0.6157	0.6386	0.6387	0.6554	0.6553	0.6669	0.6685	0.6728	0.6455
		ENUM	0.6413	0.6462	0.6386	0.6536	0.6590	0.6678	0.6620	0.6744	0.6667	0.6769	**0.6587**
		TNN	0.6207	0.6498	0.6486	0.6652	0.6485	0.6676	0.6597	0.6675	0.6684	0.6767	0.6573
9	mf	HARCD	0.5669	0.5639	0.5620	0.5665	0.5704	0.5799	0.5676	0.5684	0.5636	0.5653	0.5674
		FSMRI	0.5337	0.5268	0.5300	0.5235	0.5220	0.5172	0.5144	0.4947	0.5113	0.5025	0.5176
		KNNRS	0.6213	0.6414	0.6376	0.6515	0.6512	0.6536	0.6642	0.6705	0.6676	0.6669	0.6526
		RAW	0.6213	0.6414	0.6376	0.6515	0.6512	0.6536	0.6642	0.6705	0.6676	0.6669	0.6526
		SSAANRS	0.6016	0.6064	0.6566	0.6420	0.6548	0.6450	0.6386	0.6374	0.6369	0.6407	0.6360
		ENUM	0.7472	0.7131	0.7262	0.7333	0.7477	0.7568	0.7620	0.7611	0.7465	0.7552	0.7449
		TNN	0.4793	0.4871	0.4879	0.4960	0.4898	0.4950	0.4960	0.4982	0.5000	0.5000	0.4929
10	glass1	HARCD	0.7767	0.7798	0.7519	0.7513	0.7665	0.7528	0.7507	0.7508	0.7566	0.7497	0.7587
		FSMRI	0.5782	0.6118	0.6158	0.6301	0.6326	0.6346	0.6192	0.6198	0.6166	0.6218	0.6181
		KNNRS	0.8263	0.8022	0.7653	0.7664	0.7698	0.7764	0.7520	0.7487	0.7337	0.7339	**0.7675**
		RAW	0.8263	0.8022	0.7653	0.7664	0.7698	0.7764	0.7520	0.7487	0.7337	0.7339	**0.7675**

Table 9
The balanced accuracy of different algorithm on KNN classifier (Continued).

no	datasets	algorithm	K=1	K=2	K=3	K=4	K=5	K=6	K=7	K=8	K=9	K=10	average
		SSAANRS	0.9503	0.9409	0.9410	0.9429	0.9474	0.9371	0.9425	0.9359	0.9408	0.9418	**0.9421**
		ENUM	0.7410	0.7372	0.7231	0.7196	0.6953	0.6822	0.6854	0.6931	0.6688	0.6689	0.7014
		TNN	0.8993	0.8826	0.8950	0.8653	0.8594	0.8674	0.8657	0.8648	0.8595	0.8660	0.8725
11	wine	HARCD	0.6932	0.7097	0.6724	0.6916	0.6549	0.6587	0.6592	0.6766	0.6686	0.6832	0.6768
		FSMRI	0.9356	0.9354	0.9450	0.9263	0.9194	0.9317	0.9218	0.9308	0.9268	0.9334	0.9306
		KNNRS	0.7542	0.7460	0.7222	0.7250	0.7010	0.6938	0.7000	0.7205	0.7079	0.7075	0.7178
		RAW	0.7542	0.7460	0.7222	0.7250	0.7010	0.6938	0.7000	0.7205	0.7079	0.7075	0.7178
		SSAANRS	0.7084	0.6829	0.6893	0.6915	0.6669	0.6637	0.6820	0.6597	0.6511	0.6594	0.6755
		ENUM	0.4506	0.4731	0.4445	0.4402	0.4427	0.4165	0.3994	0.4021	0.4018	0.4183	0.4289
		TNN	0.3652	0.4058	0.4136	0.3950	0.3935	0.3847	0.3775	0.3836	0.3666	0.3669	0.3852
12	sar	HARCD	0.5087	0.4897	0.4920	0.4948	0.4951	0.5013	0.4944	0.4799	0.4913	0.4869	0.4934
		FSMRI	0.2007	0.1992	0.1984	0.2209	0.2149	0.2121	0.2031	0.1724	0.1688	0.1623	0.1953
		KNNRS	0.7933	0.7682	0.7651	0.7819	0.7806	0.7722	0.7552	0.7637	0.7479	0.7219	**0.7650**
		RAW	0.7933	0.7682	0.7651	0.7819	0.7806	0.7722	0.7552	0.7637	0.7479	0.7219	**0.7650**
		SSAANRS	0.6592	0.5997	0.5872	0.5833	0.5925	0.5778	0.5652	0.5630	0.5656	0.5570	**0.5851**
		ENUM	0.5614	0.5471	0.5478	0.5483	0.5617	0.5477	0.5563	0.5524	0.5616	0.5548	0.5539
		TNN	0.5668	0.5538	0.5771	0.5603	0.5611	0.5596	0.5496	0.5451	0.5399	0.5422	0.5556
13	hill	HARCD	0.6194	0.5618	0.5720	0.5607	0.5620	0.5456	0.5360	0.5276	0.5209	0.5318	0.5538
		FSMRI	0.5326	0.5335	0.5408	0.5391	0.5278	0.5411	0.5379	0.5382	0.5444	0.5325	0.5368
		KNNRS	0.5953	0.5600	0.5467	0.5620	0.5703	0.5546	0.5508	0.5448	0.5343	0.5386	0.5558
		RAW	0.5953	0.5600	0.5467	0.5620	0.5703	0.5546	0.5508	0.5448	0.5343	0.5386	0.5558
		SSAANRS	0.4583	0.5708	0.5396	0.4813	0.4542	0.4500	0.4500	0.3833	0.3896	0.4292	0.4606
		ENUM	0.6958	0.8021	0.7292	0.6500	0.6021	0.5729	0.5625	0.5667	0.5125	0.4167	0.6110
		TNN	0.1313	0.2813	0.2458	0.1417	0.1333	0.1250	0.1750	0.1646	0.1729	0.1563	0.1727
14	gc	HARCD	0.7063	0.9021	0.7458	0.7167	0.6917	0.6563	0.5354	0.5042	0.4188	0.3500	**0.6227**
		FSMRI	0.0896	0.1667	0.2167	0.1813	0.2479	0.1646	0.1354	0.1875	0.2250	0.2583	0.1873
		KNNRS	0.2292	0.2958	0.3396	0.2521	0.1875	0.1604	0.1333	0.1542	0.1542	0.1708	0.2077
		RAW	0.2292	0.2958	0.3396	0.2521	0.1875	0.1604	0.1333	0.1542	0.1542	0.1708	0.2077
		SSAANRS	0.8751	0.8753	0.8846	0.8860	0.8909	0.8877	0.8851	0.8908	0.8937	0.9002	0.8869
		ENUM	0.8029	0.8090	0.8433	0.8400	0.8586	0.8595	0.8614	0.8527	0.8484	0.8493	0.8425
		TNN	0.9171	0.9030	0.9235	0.9232	0.9332	0.9305	0.9324	0.9328	0.9362	0.9341	**0.9266**
15	wdbc	HARCD	0.9001	0.8946	0.9133	0.9159	0.9222	0.9159	0.9208	0.9166	0.9205	0.9188	0.9139
		FSMRI	0.8331	0.8585	0.8843	0.8810	0.8992	0.8950	0.9061	0.9019	0.9045	0.9018	0.8865
		KNNRS	0.9084	0.9062	0.9169	0.9168	0.9230	0.9192	0.9236	0.9218	0.9227	0.9210	0.9180
		RAW	0.9084	0.9062	0.9169	0.9168	0.9230	0.9192	0.9236	0.9218	0.9227	0.9210	0.9180
		SSAANRS	0.6502	0.6203	0.6113	0.5717	0.4940	0.5245	0.5154	0.5429	0.5245	0.5339	**0.5589**
		ENUM	0.4716	0.5061	0.4882	0.4787	0.4971	0.4977	0.5169	0.5209	0.5302	0.5265	0.5034
		TNN	0.6042	0.5983	0.5571	0.5530	0.5605	0.5619	0.5518	0.5365	0.5221	0.5348	0.5580
16	tc	HARCD	0.4809	0.4770	0.4815	0.5060	0.4905	0.4960	0.5057	0.5346	0.5689	0.5563	0.5098
		FSMRI	0.5399	0.5054	0.5518	0.5607	0.5731	0.5733	0.5797	0.5700	0.5577	0.5528	0.5564
		KNNRS	0.4390	0.4329	0.4305	0.4910	0.4659	0.4744	0.4617	0.4838	0.4438	0.4580	0.4581
		RAW	0.4390	0.4329	0.4305	0.4910	0.4659	0.4744	0.4617	0.4838	0.4438	0.4580	0.4581
		SSAANRS	0.8885	0.8010	0.7978	0.7765	0.8419	0.8467	0.8427	0.8065	0.8209	0.8108	**0.8233**
		ENUM	0.7180	0.7270	0.6936	0.6802	0.6774	0.6848	0.6590	0.6520	0.6585	0.6733	0.6824
		TNN	0.8915	0.8005	0.8034	0.7843	0.8318	0.8247	0.8361	0.7949	0.8327	0.8302	0.8230
17	colon	HARCD	0.6606	0.6188	0.5878	0.5430	0.5450	0.5765	0.5855	0.5490	0.5415	0.5415	0.5749
		FSMRI	0.6191	0.6066	0.6627	0.6663	0.6897	0.6701	0.7184	0.6931	0.7070	0.6786	0.6712
		KNNRS	0.7548	0.7734	0.8008	0.7553	0.7909	0.7318	0.7045	0.7103	0.6967	0.6856	0.7404
		RAW	0.7548	0.7734	0.8008	0.7553	0.7909	0.7318	0.7045	0.7103	0.6967	0.6856	0.7404

Data availability

Data will be made available on request.

Acknowledgements

The research is supported by the National Natural Science Foundation of China under Grant Nos. 62166001, Graduate Innovation Funding Program of Gannan Normal University, China under Grant No. YCX22A025. Jiangxi Province Natural Science Foundation of China under Grant Nos. 20232ACB202013.

Table 10
The accuracy of different algorithm on Decision Tree classifier.

no	datasets	SSAANRS	ENUM	TNN	HARCD	FSMRI	KNNRS	RAW
1	plrx	0.6588	0.5819	**0.6670**	0.6170	0.6302	0.5868	0.5868
2	ionosphere	0.8795	0.8467	**0.8858**	0.7974	0.7772	0.8764	0.8764
3	parkinsons	0.8549	0.8544	0.8359	**0.8610**	0.8164	0.8492	0.8492
4	heart	**0.7785**	0.7193	0.7544	0.7237	0.7181	0.7544	0.7544
5	wpbc	0.6768	0.6933	0.6670	**0.7294**	0.6644	0.6778	0.6778
6	audit	**0.9999**	0.8934	0.9997	0.8715	0.9273	**0.9999**	**0.9999**
7	bcc	0.6621	**0.7103**	0.6319	0.6466	0.7043	0.7017	0.7017
8	cmsc	0.8654	0.8626	0.8639	0.8950	0.8622	**0.9044**	**0.9044**
9	mf	0.6294	0.6202	**0.6540**	0.5913	0.5217	0.6104	0.6104
10	glass1	0.7514	**0.7607**	0.6215	0.8014	0.7079	0.7355	0.7355
11	wine	0.9124	**0.9247**	0.8601	0.8416	0.8910	0.8944	0.8944
12	sar	0.6526	0.5140	0.5064	0.5836	0.3766	**0.6608**	**0.6608**
13	hill	**0.6114**	0.5503	0.5449	0.5617	0.5165	0.5865	0.5865
14	gc	0.3833	**0.4833**	0.1500	0.4917	0.1896	0.2938	0.2938
15	wdbc	**0.9295**	0.8332	0.9281	0.9204	0.8531	0.9223	0.9223
16	tc	0.6275	0.5392	**0.6333**	0.6012	0.5632	0.5673	0.5673
17	colon	**0.8387**	0.7468	0.7903	0.7048	0.7387	0.7355	0.7355
	Average	**0.7478**	0.7138	0.7055	0.7200	0.6740	0.7269	0.7269

Table 11
The balanced accuracy of different algorithm on Decision Tree classifier.

no	datasets	SSAANRS	ENUM	TNN	HARCD	FSMRI	KNNRS	RAW
1	plrx	**0.5685**	0.4852	0.5669	0.5213	0.5133	0.4787	0.4787
2	ionosphere	0.8672	0.8366	**0.8758**	0.7842	0.7580	0.8688	0.8688
3	parkinsons	0.8076	0.7926	0.7017	**0.8201**	0.7456	0.8126	0.8126
4	heart	**0.7728**	0.7151	0.7464	0.7150	0.7148	0.7493	0.7493
5	wpbc	0.5517	0.5638	0.5038	**0.6151**	0.5284	0.5656	0.5656
6	audit	**0.9998**	0.8818	0.9997	0.8593	0.9301	**0.9998**	**0.9998**
7	bcc	0.6581	**0.7076**	0.6291	0.6462	0.7023	0.6989	0.6989
8	cmsc	0.5845	0.5345	0.5974	0.6469	0.5038	**0.6751**	**0.6751**
9	mf	0.6286	0.6199	**0.6548**	0.5922	0.5209	0.6095	0.6095
10	glass1	0.7218	0.7373	0.4822	**0.7780**	0.6793	0.7142	0.7142
11	wine	0.9130	**0.9260**	0.8676	0.8406	0.8960	0.8986	0.8986
12	sar	**0.5807**	0.3914	0.3835	0.4786	0.2169	0.5756	0.5756
13	hill	**0.6109**	0.5502	0.5445	0.5612	0.5165	0.5856	0.5856
14	gc	0.3833	0.4833	0.1500	**0.4917**	0.1896	0.2938	0.2938
15	wdbc	0.9228	0.8210	**0.9236**	0.9140	0.8434	0.9176	0.9176
16	tc	0.5829	0.4714	**0.5869**	0.5280	0.5153	0.5111	0.5111
17	colon	**0.8177**	0.7127	0.7669	0.6618	0.7095	0.7142	0.7142
	Average	**0.7042**	0.6606	0.6459	0.6738	0.6167	0.6864	0.6864

Table 12
The wilcoxon test results of SSAANRS and other algorithm.

Classifier	Measure	ENUM	TNN	HARCD	FSMRI	KNNRS	RAW
DT	Accuracy	0.0684	0.0148	0.084	0.0008	0.0437	0.0437
	Balanced accuracy	0.0245	0.0129	0.1626	0.0006	0.0879	0.0879
KNN	Accuracy	0.0552	0.0929	0.2097	0.0036	0.0312	0.0312
	Balanced accuracy	0.1359	0.2274	0.2097	0.0023	0.0442	0.0442

References

[1] Weihua Xu, Kehua Yuan, Wentao Li, Weiping Ding, An emerging fuzzy feature selection method using composite entropy-based uncertainty measure and data distribution, IEEE Trans. Emerg. Top. Comput. Intell. 7 (1) (2023) 76–88.

[2] Rehab Ali Ibrahim, Mohamed Abd Elaziz, Diego Oliva, Songfeng Lu, An improved runner-root algorithm for solving feature selection problems based on rough sets and neighborhood rough sets, Appl. Soft Comput. 97 (2020) 105517.

[3] Doudou Guo, Weihua Xu, Yuhua Qian, Weiping Ding, M-fccl: memory-based concept-cognitive learning for dynamic fuzzy data classification and knowledge fusion, Inf. Fusion 100 (2023) 101962.

[4] Weihua Xu, Doudou Guo, Yuhua Qian, Weiping Ding, Two-way concept-cognitive learning method: a fuzzy-based progressive learning, IEEE Trans. Fuzzy Syst. 31 (6) (2023) 1885–1899.

[5] Weihua Xu, Doudou Guo, Jusheng Mi, Yuhua Qian, Keyin Zheng, Weiping Ding, Two-way concept-cognitive learning via concept movement viewpoint, IEEE Trans. Neural Netw. Learn. Syst. 34 (10) (2023) 6798–6812.

[6] Changzhong Wang, Yang Huang, Mingwen Shao, Qinghua Hu, Degang Chen, Feature selection based on neighborhood self-information, IEEE Trans. Cybern. 50 (9) (2020) 4031–4042.

[7] Tengyu Yin, Hongmei Chen, Zhong Yuan, Jihong Wan, Keyu Liu, Shi-Jinn Horng, Tianrui Li, A robust multilabel feature selection approach based on graph structure considering fuzzy dependency and feature interaction, IEEE Trans. Fuzzy Syst. 31 (12) (2023) 4516–4528.
[8] Doudou Guo, Weihua Xu, Weiping Ding, Yiyu Yao, Xizhao Wang, Witold Pedrycz, Yuhua Qian, Concept-cognitive learning survey: mining and fusing knowledge from data, Inf. Fusion 109 (2024) 102426.
[9] Tengyu Yin, Hongmei Chen, Jihong Wan, Pengfei Zhang, Shi-Jinn Horng, Tianrui Li, Exploiting feature multi-correlations for multilabel feature selection in robust multi-neighborhood fuzzy β covering space, Inf. Fusion 104 (2024) 102150.
[10] Doudou Guo, Weihua Xu, Fuzzy-based concept-cognitive learning: an investigation of novel approach to tumor diagnosis analysis, Inf. Sci. 639 (2023) 118998.
[11] Doudou Guo, Weihua Xu, Yuhua Qian, Weiping Ding, Fuzzy-granular concept-cognitive learning via three-way decision: performance evaluation on dynamic knowledge discovery, IEEE Trans. Fuzzy Syst. 32 (3) (2024) 1409–1423.
[12] Yun Kang, Jianhua Dai, Attribute reduction in inconsistent grey decision systems based on variable precision grey multigranulation rough set model, Appl. Soft Comput. 133 (2023) 109928.
[13] Wentao Li, Haoxiang Zhou, Weihua Xu, Xi-Zhao Wang, Witold Pedrycz, Interval dominance-based feature selection for interval-valued ordered data, IEEE Trans. Neural Netw. Learn. Syst. 34 (10) (2023) 6898–6912.
[14] Weihua Xu, Man Huang, Zongying Jiang, Yuhua Qian, Graph-based unsupervised feature selection for interval-valued information system, IEEE Trans. Neural Netw. Learn. Syst. (2023) 1–14.
[15] Qinghua Hu, Daren Yu, Jinfu Liu, Congxin Wu, Neighborhood rough set based heterogeneous feature subset selection, Inf. Sci. 178 (18) (2008) 3577–3594.
[16] Kehua Yuan, Duoqian Miao, Yiyu Yao, Hongyun Zhang, Xuerong Zhao, Feature selection using zentropy-based uncertainty measure, IEEE Trans. Fuzzy Syst. 32 (4) (2024) 2246–2260.
[17] Weihua Xu, Ziting Yuan, Zheng Liu, Feature selection for unbalanced distribution hybrid data based on *k*-nearest neighborhood rough set, IEEE Trans. Artif. Intell. 5 (1) (2024) 229–243.
[18] Kehua Yuan, Weihua Xu, Duoqian Miao, A local rough set method for feature selection by variable precision composite measure, Appl. Soft Comput. 155 (2024) 111450.
[19] Changzhong Wang, Yunpeng Shi, Xiaodong Fan, Mingwen Shao, Attribute reduction based on k-nearest neighborhood rough sets, Int. J. Approx. Reason. 106 (2019) 18–31.
[20] Pengfei Zhang, Tianrui Li, Zhong Yuan, Chuan Luo, Keyu Liu, Xiaoling Yang, Heterogeneous feature selection based on neighborhood combination entropy, IEEE Trans. Neural Netw. Learn. Syst. 35 (3) (2024) 3514–3527.
[21] Xiaoling Yang, Hongmei Chen, Tianrui Li, Jihong Wan, Binbin Sang, Neighborhood rough sets with distance metric learning for feature selection, Knowl.-Based Syst. 224 (2021) 107076.
[22] Hongmei Chen, Tianrui Li, Xin Fan, Chuan Luo, Feature selection for imbalanced data based on neighborhood rough sets, Inf. Sci. 483 (2019) 1–20.
[23] Degang Chen, Yanyan Yang, Attribute reduction for heterogeneous data based on the combination of classical and fuzzy rough set models, IEEE Trans. Fuzzy Syst. 22 (5) (2014) 1325–1334.
[24] Zhong Yuan, Hongmei Chen, Pengfei Zhang, Jihong Wan, Tianrui Li, A novel unsupervised approach to heterogeneous feature selection based on fuzzy mutual information, IEEE Trans. Fuzzy Syst. 30 (9) (2022) 3395–3409.
[25] Tareq M. Al-shami, Davide Ciucci, Subset neighborhood rough sets, Knowl.-Based Syst. 237 (2022) 107868.
[26] Di Zhang, Ping Zhu, Variable radius neighborhood rough sets and attribute reduction, Int. J. Approx. Reason. 150 (2022) 98–121.
[27] Kanglin Qu, Jiucheng Xu, Ziqin Han, Shihui Xu, Maximum relevance minimum redundancy-based feature selection using rough mutual information in adaptive neighborhood rough sets, Appl. Intell. 53 (14) (2023) 17727–17746.
[28] Abdel-Fattah Attia, Mohamed Abd Elaziz, Aboul Ella Hassanien, Ragab A. El-Sehiemy, Prediction of solar activity using hybrid artificial bee colony with neighborhood rough sets, IEEE Trans. Comput. Soc. Syst. 7 (5) (2020) 1123–1130.
[29] BingXiang Liu, Feng Liu, Xiang Cheng, An adaptive genetic algorithm based on rough set attribute reduction, in: 2010 3rd International Conference on Biomedical Engineering and Informatics, volume 7, 2010, pp. 2880–2883.
[30] Ahmed Tariq Sadiq, Mehdi Gzar Duaimi, Samir Adil Shaker, Data missing solution using rough set theory and swarm intelligence, in: 2012 International Conference on Advanced Computer Science Applications and Technologies (ACSAT), 2012, pp. 173–180.
[31] Yumin Chen, Qingxin Zhu, Huarong Xu, Finding rough set reducts with fish swarm algorithm, Knowl.-Based Syst. 81 (2015) 22–29.
[32] Tarun Maini, Abhishek Kumar, RakeshKumar Misra, Devender Singh, Rough set based feature selection usi ng swarm intelligence with distributed sampled initialisation, in: 2017 6th International Conference on Computer Applications in Electrical Engineering-Recent Advances (CERA), 2017, pp. 92–97.
[33] Lin Sun, Shanshan Si, Weiping Ding, Xinya Wang, Jiucheng Xu, Tfsfb: two-stage feature selection via fusing fuzzy multi-neighborhood rough set with binary whale optimization for imbalanced data, Inf. Fusion 95 (2023) 91–108.
[34] Xiying Chen, Caihui Liu, Bowen Lin, Jianying Lai, Duoqian Miao, Aha-3wkm: the optimization of k-means with three-way clustering and artificial hummingbird algorithm, Inf. Sci. 672 (2024) 120661.
[35] Shen SI-MA, Hanming Liu, Hongxiang Zhan, Gang Guo, Cong Yu, Pengcheng Hu, Swarm intelligence algorithms evaluation, arXiv e-prints, 2023.
[36] Jiankai Xue, Bo Shen, A novel swarm intelligence optimization approach: sparrow search algorithm, Syst. Sci. Control Eng. 8 (1) (2020) 22–34.
[37] Farhad Soleimanian Gharehchopogh, Mohammad Namazi, Laya Ebrahimi, Benyamin Abdollahzadeh, Advances in sparrow search algorithm: a comprehensive survey, Arch. Comput. Methods Eng. 30 (2023) 427–455.
[38] S. Udhaya Kumar, H. Hannah Inbarani, Pso-based feature selection and neighborhood rough set-based classification for bci multiclass motor imagery task, Neural Comput. Appl. 28 (2017) 3239–3258.
[39] Changzhong Wang, Qiang He, Mingwen Shao, Qinghua Hu, Feature selection based on maximal neighborhood discernibility, Int. J. Mach. Learn. Cybern. 9 (2018) 1929–1940.
[40] Qinghua Hu, Jinfu Liu, Daren Yu, Mixed feature selection based on granulation and approximation, Knowl.-Based Syst. 21 (4) (2008) 294–304.
[41] Zexuan Zhu, Yew-Soon Ong, Manoranjan Dash, Markov blanket-embedded genetic algorithm for gene selection, Pattern Recognit. 40 (11) (2007) 3236–3248.