A Meta-heuristic Feature Selection Algorithm Combining Random Sampling Accelerator and Ensemble Using Data Perturbation

Shuaishuai Zhang[1], Keyu Liu[2], Taihua Xu[1,3†], Xibei Yang[1] and Ao Zhang[1]

[1]School of Computer, Jiangsu University of Science and Technology, Zhenjiang, 212100, Jiangsu China.
[2]School of Computing and Artificial Intelligence, Southwest Jiaotong University, Chengdu, 611756, Sichuan China.
[3]Key Laboratory of Oceanographic Big Data Mining & Application of Zhejiang Province, Zhejiang Ocean University, Zhoushan, 316022, Zhejiang, China.

Contributing authors: 863031433@qq.com; keyu_liu@my.swjtu.edu.cn; xth19890410@163.com; jsjxy_yxb@just.edu.cn; azh@just.edu.cn;
†Corresponding author.

Abstract

Meta-heuristic algorithms have been extensively utilized in feature selection tasks because they can obtain the global optimal solution. However, the meta-heuristic algorithm will take too much time in the face of a large number of samples. Although most of the studies compromise to approximate optimal solutions for avoiding time-consuming problems, a new problem with reduced classification performance, especially classification stability, is then generated. Aiming to above problems, this paper proposes a new feature selection framework. First, this framework exploits a voting ensemble strategy to improve classification stability by reducing the impact of misclassified labels on the overall classification results. Second, the framework uses a data perturbation strategy to enhance classification accuracy. In particular, the data perturbation strategy is able to generate more neighborhood relationships in the dataset, which could reveal the distribution of various features of the samples. A voting ensemble of different feature distributions is capable of extracting more information from the dataset, then the initially misclassified samples are more likely to be returned to the correct classification. Third, the framework takes a random sampling accelerator into account to solve the problem of excessive time consumption by reducing the size of the search sample space. Finally, for the sake of verifying the effectiveness of the proposed framework, four meta-heuristic feature selection methods based on a neighborhood rough set are compared on 20 datasets. The experimental results indicate that our framework could improve classification performance and accelerate feature selection, particularly in confronting large sample sizes.

Keywords: Feature selection; Meta-heuristic; Neighborhood rough set; Data perturbation; Random sampling

1 Introduction

As the main topic in the current research on data preprocessing, feature selection has been broadly employed in the fields of data mining and pattern recognition [1–5]. Feature selection is capable of saving storage space [6, 7] and enhancing the generalization ability of the learned model [8, 9]. Feature selection is generally divided into two strategies, exhaustive and heuristic search, both of which aim to find the most strongly correlated and salient feature subset [10–13]. Exhaustive searches flourished in the early days of feature selection research. For example, methods based on the discriminant matrix [14, 15] represent a type of exhaustive search. However, exhaustive searches are NP-hard problems [16, 17], so the time consumption may become unacceptable as the amount of data rises. To deal with this problem, many researchers have turned to the heuristic search strategy for feature selection with acceptable time consumption. In particular, in order to better find the global optimal feature subset, meta-heuristic algorithms are increasingly adopted in the field of feature selection. For instance, Ritam et al. [18] proposed a novel approach to enhance the efficiency of feature selection by integrating a discrete equilibrium optimizer into the simulated annealing algorithm. Elaziz et al. [19] proposed a new meta-heuristic algorithm, the so-called "atomic orbital search". For feature selection based on opponent learning. There are other meta-heuristic algorithms such as ant colony optimization [20], artificial fish swarm [21], monarch butterfly optimization [22], genetic algorithm [23], and forest optimization algorithm [24]. Overall, meta-heuristic feature selection is currently a popular data classification strategy.

As an effective mathematical tool for feature selection [25–31], rough set theory can be introduced into meta-heuristic algorithms. This approach is therefore able to remove irrelevant and redundant features and preserve the features of the original features, also enhancing the interpretability of feature selection. For instance, Penmatsa et al. [20] combined ant colony optimization with rough set theory to simplify and optimize the features of malware detection systems. Luan et al. [21] proposed a new feature selection algorithm based on an improved artificial fish swarm algorithm and rough sets theory. However, there exist many numerical features in the datasets of practical applications, and classical rough sets are not able to deal with them directly.

Therefore, the theory of rough neighborhood sets, which supports both continuous and discrete properties [32–37], is introduced into meta-heuristic algorithms to address this challenge. Since the combination of rough neighborhood set theory and meta-heuristic algorithms can obtain the most optimal and smallest subset of features from a numerical decision-making system, it has become one of the most important research directions of feature selection in the field of data mining [38–40]. For instance, Zou et al. [38] developed a feature selection algorithm that combined the fish swarm algorithm and the neighborhood rough set theory via an adaptive function to control the vision and step size of artificial fish. Feng et al. [39] proposed a novel feature selection methodology with rough neighborhood sets and improved particle swarm optimization. Ahmed et al. [40] proposed a new feature selection approach via a hybrid whale optimization algorithm for effective recognition of Arabic handwritten optical characters.

In addition, the classification performance of the meta-heuristic algorithm should also be further improved. In the experiments, the iteration times of the meta-heuristic algorithm required for the optimal solution are very high; hence, the general approximate optimal solution is usually chosen as an effective solution to the problem. In this way, the selected features are not stable enough, which leads to a decrease in classification performance. For this problem, the introduction of an ensemble strategy has been revealed to substantially enhance the classification performance of selected features in test samples [41–44]. Specifically, Yang et al. [44] proposed an ensemble selector, which is one of the voting ensemble strategies to compensate for the lack of a meta-heuristic algorithm in classification stability. It effectively employs the discrepancies between the multiple feature selection results to generate more divergent predictions and then votes on those results to produce the final classification results.

However, on the one hand, the ES framework [44] actually uses a part of a dataset that has been combined several times, which has a good effect on greedy algorithms. However, since meta-heuristic algorithms heavily rely on complete data features for each iteration, the performance of the ES framework for meta-heuristic algorithms is reduced. Therefore, to overcome the raised issue here, our framework is obviously more suitable for meta-heuristic algorithms, which produce superior results due to the use of a complete dataset for constructing a complete set of datasets. On the other hand, the ensemble selector essentially relies on more discrimination to improve the classification performance. In order to produce more different feature selection results, this paper adopts a data perturbation strategy. The data perturbation is the reconstruction of the original data structure to better break the coupling between the data, so that more different feature selection results are produced. The data perturbation strategy has been exploited by many researchers to augment the data, revealing more potential relationships between attributes. For instance, Li et al. [45] proposed a data perturbation strategy to classify small datasets with extensive feature information. Wang et al. [46] proposed a data perturbation strategy for coupled feature analysis of numerical data. As a result, the ensemble and perturbation strategies are appropriately introduced to the meta-heuristic algorithm to improve the classification performance, especially the classification stability.

Moreover, such algorithms will face the challenge of excessive time consumption when the number of samples is very large. In this paper, the random sampling accelerator [47, 48] is introduced to solve the above-raised problem. The random sampling accelerator is able to avoid overfitting by randomly grouping samples, which also reduces the scale of the search sample space. In addition, the random sampling accelerator is capable of greatly speeding up the feature selection process because features are selected group by group. Especially, the meaning of "group by group" is that feature selection is performed only on one group, and the result of the feature selection obtained is labeled and added to the next group, which speeds up the feature selection process of the next group.

The main contributions of our research can be summed up in the following.

(1) Combining four meta-heuristics with neighborhood rough sets to configure a new feature selection strategy.
(2) Utilizing the random sample accelerator to speed up the feature selection process of meta-heuristic algorithms.
(3) Performing the data perturbation strategy in the data preparation phase of the ensemble so that the meta-heuristic algorithm produces differential feature selection. Finally, it is effectively exploited to improve the effect of the ensemble classifier, i.e., improving the performance of the algorithm.

The rest of this paper includes the following five aspects. Section 2 briefly introduces the basic concepts of neighborhood rough sets, fitness functions, feature selection, data perturbation and random sampling accelerator. Four meta-heuristic algorithms combining neighborhood rough sets are given in Section 3. Section 4 elaborates a meta-heuristic feature selection algorithm combining random sampling accelerator and ensemble using data perturbation. Section 5 presents experimental results and conducts theoretical analysis. Section 6 summarizes whole work.

2 Preliminaries

In order to design high-performance feature selection methods, several theoretical tools are used in this paper, such as neighborhood rough set, fitness functions, feature selection and data perturbation. It is necessary to briefly introduce them before describing the proposed methods.

2.1 Neighborhood rough set

In general, a decision-making system can be defined as a two-tuple $DS = \langle U, AT \cup \{d\} \rangle$, the universe U is the finite set which contains all samples, AT is the collection of all conditional features and d is the decision feature. For any $x \in U$, $d(x)$ is the decision value of the sample x. Moreover, an equivalence relation over d can be defined as $\mathrm{IND}(U, d) = \{(x, y) \in U \times U : d(x) = d(y)\}$. According to the decision value of all samples, decision classes on the universe U can then be generated by $U/\mathrm{IND}(U, d)=\{X_1, X_2, \cdots, X_n\}$.

Definition 1 [49] Given a decision system DS, let A be a subset of AT. A neighborhood relationship is defined as follows:

$$N_A^\delta = \{(x_i, x_j) \in U \times U : \Delta(x_i, x_j) \leq \delta\}; \tag{1}$$

in which $\Delta(x_i, x_j)$ implies the distance between x_i and x_j over A, δ is the given radius.

The neighborhood relationship N_A^δ can be used to judge whether the samples in the universe U are similar. However, according to Eq. (1), the characterization of similarity between any two samples depends greatly on the radius size. A larger radius means that a coarser neighborhood relationship will be generated, that is, the determination of similarity will be looser. A smaller radius will produce a thinner neighborhood relationship, which means that the determination of similarity will be stricter. The calculation process of $U/\mathrm{IND}(U, d)$ is usually considered as information granulation [50]. There are some researchers having developed some other granulation strategies to deal with complex data. For example, cluster-based information granularity [51], fuzzy-based information granularity [52], neighborhood-based information granularity [53], etc.

For each $x_i \in U$, the neighborhood of x_i, $\delta_A(x_i) = \{x_j \in U : (x_i, x_j) \in \delta_A\}$, can be obtained by the neighborhood relationship N_A^δ. Further, a neighborhood rough set model can be defined as Definition 2.

Definition 2 [53] Given a decision system DS, let A be a subset of AT. The neighborhood lower and upper approximation sets induced by A are respectively defined as follows:

$$\underline{N_A^\delta}(X) = \{x \in U : \delta_A(x_i) \subseteq X\}, \tag{2}$$

$$\overline{N_A^\delta}(X) = \{x \in U : \delta_A(x_i) \cap X \neq \phi\}. \tag{3}$$

To characterize the dependency of decision feature d related to conditional feature subset A, Definition 3 is given.

Definition 3 [54] Given a decision system DS and neighborhood radius δ, let A be a subset of AT. The approximate quality of the decision feature d with respect to the conditional feature subset A is defined as follows:

$$\gamma_U(A,d) = \frac{\mid \underline{N_A^{\delta}}(d) \mid}{\mid U \mid}, \tag{4}$$

where $\mid X \mid$ represents the cardinality of the set X.

Obviously, $0 \leq \gamma_U(A,d) \leq 1$ holds.

Based on neighborhood relationship, a fitness function can be defined, as shown in Definition 4. As one of the most commonly used types of heuristic information, the fitness function can guide direction for iterative optimization of meta-heuristic algorithm in the process of feature selection.

Definition 4 [21] Given a decision system DS and neighborhood radius δ, let A be a subset of AT. Then the fitness function based on neighborhood dependence is expressed a as follows:

$$fitness = \lambda\gamma_U(A,d) + (1-\lambda)\frac{\mid A \mid}{\mid AT \mid}, \tag{5}$$

where λ represents the weight of the approximate quality.

2.2 Feature selection

Feature selection is the process used to select the most valuable features for classification results in large-scale data. It is a very important data preprocessing work in machine learning. Without feature selection, data processing would consume a lot of time and CPU performance when the data scale is large. Therefore, one of the most important tasks of feature selection is to select some qualified features from data with a specific criterion. For example, one or more of the most important features can be identified and added into the feature selection candidates after evaluation [55–58]. This paper mainly studies the meta-heuristic based feature selection algorithm in classification problems. A general form of feature selection can be abstracted as Definition 5 [59].

Definition 5 [59] Given a decision system $DS = \langle U, AT \cup \{d\} \rangle$, a constraint related to measure ρ over the universe U is denoted by $\rho(U, AT)$. Let A be a subset of AT, A can be referred to as a $\rho(U, AT)-$qualified feature subset if and only if the following conditions hold:

(1) A meets the constraint $\rho(U, AT)$.

(2) $\forall A' \subseteq A$, A' does not meet the constraint $\rho(U, AT)$.

2.3 Data perturbation

Multiple perturbed data with significant differences can be constructed by data perturbation strategy on the original feature space [60–62]. Although data perturbation strategy can mine more coupled information about the feature space, the usefulness of the altered feature space for classification is not always guaranteed. The ensemble strategy has been used by many scholars to address this issue [41–44]. In this paper, firstly, five feature spaces are generated by using data perturbation. Secondly, five feature selection results are obtained from the five feature spaces. Thirdly, the five feature selection results can produce more differentiated predictions. Finally, the prediction-based votes can be used to induce the final classification results. The process of data perturbation is shown as Fig. 1.

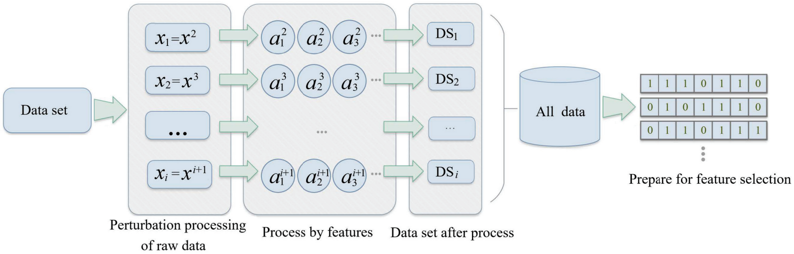

Fig. 1 Data perturbation process. For each $x \in U$, $x_i\,(i = 1, 2, \ldots)$ represents the data after different perturbation processing.

3 Four Meta-heuristic Algorithms Combining Neighborhood Rough Sets

As the ability to search for the optimal solution without a greedy strategy, and produce feature selection results with better classification performance, meta-heuristic algorithms are increasingly employed in the field of feature selection. This paper aims to design four meta-heuristic algorithms based on the fitness function generated by the neighborhood rough set. To this end, four subsequent meta-heuristic algorithms were evaluated for T iterations with population size N for the worst case.

The Monarch Butterfly Optimization (MBO) algorithm was proposed based on the migration behavior of monarch butterflies [22]. In the algorithm, the position of each butterfly represents a feasible solution, and its effectiveness is measured by the fitness function. The higher the fitness value obtained by introducing the feasible solution into the fitness function, the more effective the feasible solution is for classification problems. For each butterfly, two behaviors can be adopted, migration or adaptation to the environment. In this paper, the fitness function used is defined with a neighborhood rough set. As a result, a feature selection algorithm called NMBO based on the MBO and neighborhood rough set is proposed. The main factor contributing to the time consumption of the NMBO is the calculation of the fitness function for each individual in the population. Based on the performed analysis in Ref. [47], the time complexity of computing the fitness value for a single monarch butterfly individual is $\mathcal{O}(\mid U \mid^2)$. For the worst case that all individuals must be evaluated for T iterations with a population size of N, the time complexity of the NMBO can be expressed as $\mathcal{O}(N\cdot \mid U \mid^2 \cdot T)$.

The Genetic Algorithm (GA) [23] is a simulation of the biological evolution process in nature, which searches for the optimal solution to the problem through the genetic mechanism. It has been employed to solve various problems, including feature selection. In the present work, each individual gene represents a feature selection result. Additionally, a “1” or “0” single-gen bit indicates whether a feature is selected or not, respectively. As a result, GA generates new genetic individuals and searches for individuals with suitable fitness values by using the three behaviors of selection (), crossover (), and mutation (). As a result, a feature selection algorithm, the so-called NGA, based on the GA and neighborhood rough set is proposed. As stated in the NGA, the initial consumption time results from the generation of new individuals during gene selection behavior. The overall time complexity of the NGA will be $\mathcal{O}(N\cdot \mid U \mid^2 \cdot T)$.

Fish swarms exhibit hunting, swarming, following, and random swimming behaviors around food. Artificial Fish Swarm Algorithm (AFSA) formulates four behaviors to simulate fish habits according to their survival characteristics [21]. The location of each fish can be considered as its food concentration, which has a fitness value evaluated by the fitness function. During each iteration, each fish fulfills its food needs through the prey (), Swarm (), follow (), and random Swim (). As a result, a feature selection algorithm called NAFSA based on the AFSA and neighborhood rough set is proposed. The consumption time of the NAFSA mainly arises from calculating individual fitness values in Swarm () behavior. Subsequently, the total time complexity of the NAFSA will be $\mathcal{O}(N^2\cdot \mid U \mid^2 \cdot T)$.

Forest Optimization Algorithm (FOA) is a meta-heuristic algorithm proposed for feature selection problems, leading to remarkable results [24]. Each tree in the FOA reflects a subset of features that could be utilized for feature selection problems. As a result, a feature selection algorithm called NFOA based on the FOA and neighborhood rough set is proposed. NFOA’s time consumption is mainly related to the globalSeeding () stage. In the worst case, the computation must be performed in T iterations, then it is $\mathcal{O}(GSC^T \cdot N\cdot \mid U \mid^2 \cdot T)$, where GSCrepresents the number of positions selected in the globalSeedin () stage.

4 Improvements to the above four algorithms

There are two main issues associated with the NMBO, NGA, NAFSA and NFOA. First, these algorithms are time-consuming when the sample size is large, making them difficult to exploit in practice. Second, the feature selection results obtained from these algorithms are often unstable, which negatively affects the classification performance, especially in terms of stability.

To tackle the problem of excessive time consumption, one solution is to utilize a random sampling accelerator. Accelerators are able to speed up the feature selection process by selecting features group by group. This is the motivation to establish a new algorithm, the meta-heuristic feature selection algorithm based on a random sampling accelerator, which is given as Algorithm 1.

In addition, for the sake of improving the classification performance, we follow the following steps. In the first step, the data perturbation strategy is adopted to discover the relationship between samples. Second, a voting strategy is implemented to select robust forecast information, such as prediction labels, for subsequent classification. Finally, these prediction labels are appropriately compiled to produce the final prediction results. Therefore, a meta-heuristic feature selection algorithm combining accelerated random sampling and ensemble based on the data perturbation is proposed, which is provided as Algorithm 2. This algorithm not only performs feature selection (when the sample is large) faster than meta-heuristics but also enhances the classification performance.

Next, the detailed explanations of Algorithm 1 and 2 are provided.

4.1 Meta-heuristic Feature Selection Algorithm Based on Random Sampling Accelerator

Algorithm 1 aims to address the issue of excessive time consumption, which combines meta-heuristic algorithms and random sampling accelerators. The main steps of Algorithm 1 are as follows:

First, the entire sample space is randomly divided into several groups to speed up the execution of the algorithm. In the second step, a meta-heuristic algorithm is employed to determine the feature selection results from the samples of the first group. During this process, the algorithm evaluates the selected feature to ensure that it meets certain constraints, such as approximate quality thresholds. Third, the first group of samples is re-evaluated after adding a new sample group. If the selected features still satisfy the constraints, Algorithm 1 adds these sample groups directly without running the meta-heuristic algorithm. Otherwise, the algorithm performs meta-heuristic feature selection using the previous feature selection results. Finally, the algorithm repeats the above process until all sample groups are suitably traversed.

In Algorithm 1, $\rho(U^{'}, A_i)$ is the measure of the feature selection set A_i under the sample set $U^{'}$ (the approximate quality is used as the metric). Following the procedure of Algorithm 1, the worst case consists of two aspects: (1) each iteration adds at least one feature to the selected features; (2) all features in AT are required for the final feature selection set A. Therefore, the worst-case total time complexity of Algorithm 1 is: $T(n) = \mathcal{O}(\sum_{i=1}^{m} n_i)$, where $n_i = (\frac{i}{m} \cdot \mid U \mid \cdot T_i)^2 (i = 1, 2, \ldots, m)$ is the time consumption in the ith loop by using Algorithm 1. Let T represent the number of iterations required for the

Algorithm 1: Meta-heuristic Feature Selection Algorithm Based on Random Sampling Accelerator.

Input: A Decision system DS, a constraint $\rho(U, AT)$.
Output: A feature selection set A.

```
Initialization, A = ∅.
The dataset U is randomly divided into m groups U_1, ..., U_m.
Use U_1 to solve the feature selection set A_1 using one of the above four meta-heuristics.
For i = 2 : m do
    //Continue feature selection guided by the last feature selection set.
    Let U' = Σ_{k=1}^{i} U_k, A_i = A_{i-1}.
    ρ = ρ(U', A_i).
    While ρ ≤ ρ(U', AT) do
        Use meta-heuristic algorithm to continue selecting feature group b from the remaining features.
        Evaluate feature group b by computing ρ(U', A ∪ b).
        A_i = A_{i-1} ∪ {b}.
    End While
End For
A = A_m.
Return A;
```

meta-heuristic algorithm used under the entire sample set, and T_i is the number of iterations used for each grouping. Obviously, $T = T_1 + T_2 + \cdots + T_m$. And by calculating $T(n)$, $T(n) \leq \mathcal{O}(N \cdot \mid U \mid^2 \cdot T)$ holds.

Compared to the time complexities of NMBO, NGA, NAFSA and NFOA, Algorithm 1 can accelerate them. For the NAFSA and NFOA, the acceleration is obviously produced. In theory, NMBO and NGA are less time consuming than NFOA. However, the acceleration effect can still be obtained. Therefore, Algorithm 1 is able to produce a speed-up effect. A simple example is given below to help understand the main principles of our proposed algorithm.

Table 1 An example of decision system.

	a_1	a_2	a_3	a_4	L
x_1	0.8000	0.8000	0.5678	1.0000	1
x_2	0.5621	0.5621	1.0000	0.5399	1
x_3	0.7173	1.0000	0.8168	0.5043	2
x_4	1.0000	0.7586	0.8864	0.1334	2

Example 1 Given a decision system $DS =< U, AT \cup \{d\} >$, where $U = \{x_1, x_2, x_3, x_4\}, AT = \{a_1, a_2, a_3, a_4\}$, $L = \{1, 1, 2, 2\}$ is the value of decision feature d by each sample. Table 1 is the specific description of the decision system.

By the decision system in Table 1, the steps of Algorithm 1 are :

(1) The samples are randomly divided into 2 groups: $\{x_1, x_2\}$ and $\{x_3, x_4\}$.
(2) NMBO (one meta-heuristic algorithm) is used to solve the feature selection for the first group, and the result set $\{a_1\}$ of the first feature selection is obtained.
(3) The result set $\{a_1\}$ is measured. It is found that the requirements are not met. Then, the second sample group $\{x_3, x_4\}$ is added to the first group $\{x_1, x_2\}$. And the meta-heuristic algorithm is used again for feature selection (immediately use the previous feature selection result $\{a_1\}$) to obtain the new result set $\{a_1, a_2, a_4\}$.
(4) Measure the result obtained in step 3 and find that it meets the requirements. Therefore, Algorithm 1 ends the loop.

4.2 A Meta-heuristic Feature Selection Algorithm Combining Random Sampling Accelerator and Ensemble Using Data Perturbation

In recent years, to improve the performance of feature selection, many researchers have incorporated ensemble strategies into classification problems. These strategies can be classified into two types: (i) introducing an ensemble mechanism in the feature selection problem to help the search process for better evaluation of the candidate features and selecting more robust features for subsequent learning; (ii) reusing a particular search or using various search processes to find multiple selected features that could provide a basic integration tool for subsequent learning. The latter is more flexible in constructing the framework and is able to greatly enhance learner performance.

To implement this approach, we first transform the original neighborhood decision system into five neighborhood decision systems via the perturbation strategy. Then, Algorithm 1 is employed to generate feature selection results on five perturbed datasets. Then, multiple classifiers are applied to these feature selection results to generate five sets of predicted labels. Finally, a predicted label set is generated using majority voting, and the classification result is appropriately obtained.

In this section, we propose a new framework "REP" to improve the performance of Algorithm 1 through ensemble strategy and data perturbation, which is provided in Algorithm 2. Here, "R", "E", and "P" stand for random sampling accelerator, ensemble strategy, and perturbation strategy, respectively. Then, the meta-heuristic feature selection algorithm based on the random sampling, accelerator, and ensemble using data disturbance, respectively, the so-called REP-NMBO, REP-NGA, REP-NAFSA, and REP-NFOA, are constructed.

Following the process of Algorithm 2, its time consumption is mainly generated by Algorithm 1 in k perturbed data sets. Therefore, its main time

Algorithm 2: A Meta-heuristic Feature Selection Algorithm Combining Random Sampling Accelerator and Ensemble Using Data Perturbation.

Input:A Decision system DS, a constraint $\rho(U, AT)$.
Output:A feature selection set A.
//SA represents a group of feature selection sets
Initialization, $A = \emptyset$, $SA = \emptyset$.
Set $k = 5$. //k is the number of times for perturbation data process.
For $i = 1 :$ k **do**
    The data set U generates U_i through a perturbation strategy.
    Use Algorithm 1 for U_i to get a feature selection A_i.
    $SA = SA \cup A_i$.
End For
//Use the voting strategy for the SA result set.
Let A=vote(SA).
Return A;

consumption depends on Algorithm 1. After the analysis of Algorithm 1, it can be concluded that the time complexity of Algorithm 2 in the worst case is: $T(n) = k \cdot \mathcal{O}(\sum_{i=1}^{m} n_i)$, where $n_i = (\frac{i}{m} \cdot \mid U \mid \cdot T_i)^2 (i = 1, 2, \ldots, m)$ is the time consumption in the ith loop by using Algorithm 1. Let T represent the number of iterations required for the meta-heuristic algorithm used under the entire sample set, and T_i is the number of iterations used for each grouping. Obviously, $T = T_1 + T_2 + \cdots + T_m$. And by calculating $T(n)$, $T(n) \leq \mathcal{O}(N \cdot \mid U \mid^2 \cdot T)$ holds when the value of $\mid U \mid$ is large. Therefore, the algorithm is able to produce a speed-up effect. Furthermore, it should be noticed that the performance is not perfectly linear in terms of the value of k. The authors employ the perturbation approach to establish more diversity in the neighborhood relationship of the sample. However, when the value of k is larger than 5, the change of the sample neighborhood is not obvious enough, and the consumption time rises after the number of groups increases; therefore, the appropriate value of k is set as 5. And due to the ensemble based on voting strategy, it outperforms the original algorithm in classification performance.

The following Fig. 2 is provided to illustrate the process of Algorithm 2.

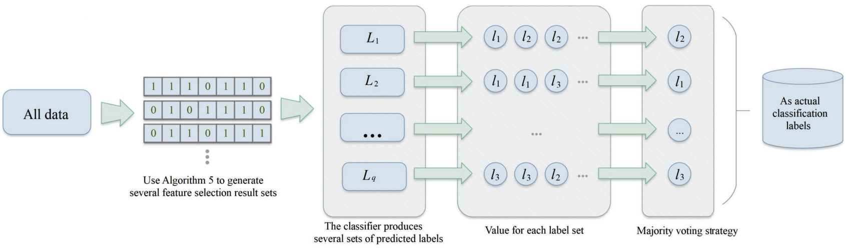

Fig. 2 The process of a Meta-heuristic Feature Selection Algorithm Combining Random Sampling Accelerator and Ensemble Using Data Perturbation.

5 Experimental analyses

5.1 Datasets

In this part, 20 datasets from the UCI standard datasets are selected for experiments to prove the effectiveness of our algorithms. The specific information of the data is shown in Table 2 below. All of the experiments were carried out on a personal computer with Windows 10, Intel Core (TM) i5-7300HQ (2.50 GHz), and 8.00 GB memory. The experimental platform was MATLAB R2018a.

Table 2 Datasets description.

ID	Datasets	#Samples	#Features	#Labels
1	Cardiotocography	2126	21	10
2	SPECTF Heart	267	44	2
3	Statlog (Vehicle Silhouettes)	846	18	4
4	Wdbc	569	30	2
5	Dermatology	366	34	6
6	Forest Type Mapping	523	27	4
7	Wine	178	13	3
8	Urban Land Cover	675	147	9
9	German	1000	24	2
10	QSAR Biodegradation	1055	41	2
11	Synthetic Control Chart Time Series	600	60	6
12	Climate Model Simulation Crashes	540	20	2
13	Statlog (Heart)	270	13	2
14	Ionosphere	351	34	2
15	waveform	5000	21	3
16	Wall-Following Robot Navigation	5456	24	4
17	Pen-Based Recognition of Handwritten Digits	10992	17	10
18	MAGIC Gamma Telescope	19020	11	2
19	twonorm	7400	20	2
20	Crowdsourced Mapping	10845	29	6

5.2 Experiments-part

In this section, the experiments implement NMBO, NGA, NAFSA, NFOA, an ensemble framework [44] abbreviated as ES, and our framework is represented by REP. The first comparison made on the time consumption of each algorithm is to verify the effectiveness of the random sampling accelerator by employing the REP. Then, the classification performance of each algorithm is appropriately compared to verify the effectiveness of the ensemble strategy in the presence of the data perturbation. Finally, the lengths of the feature selection result set for each algorithm are compared. No significant change is observed during the set of the feature selection results of the REP, NMBO, NGA, NAFSA, NFOA and ES. Overall, the experiments reveal that the REP exhibits better classification performance compared to NMBO, NGA, NAFSA, NFOA and ES, whereas it is able to perform feature selection faster with a large number of samples.

According to the experimental results, when the number of random sampling groups is set equal to 5, a better acceleration effect can be achieved. The experiment also creates five new data sets by perturbing the original data. These five datasets are then employed to produce five feature selection results for classifying the test samples. The achieved results indicate that the ensemble improves the classification performance compared to the original meta-heuristic algorithms (NMBO, NGA, NAFSA and NFOA). Following neighborhood rough set, we have used 20 different radii, such as 0.02, 0.04, . . . , 0.40. And under these 20 different radii, the average value, classification stability and classification accuracy of the time consumption required for the reduction solution are obtained. It should be noted that the algorithms shown above were tested using 10-fold cross-validation. In addition, the length of each individual in the initial population generated by the meta-heuristic algorithm is set to $10\% \times \mid AT \mid$.

5.2.1 Comparisons of elapsed time

The consumption time of NMBO, NGA, NAFSA, NFOA, ES and REP are calculated and their corresponding consumption times have been listed in Table 3. In the given table, the first column signifies the number of the dataset (ID), whereas the other columns are divided into four parts. For each part: (1) The first column represents the time consumption data of the meta-heuristic algorithm based on a rough set (one of NMBO, NGA, NAFSA and NFOA); (2) The second column denotes the time consumption data of the algorithm using the ES ensemble strategy. (3) The third column signifies the REP's time consumption data. According to Table 3, the following results can be expressed:

According to Table 3, the following conclusions can be outlined.

(1) For the Algorithm 2, the random sampling accelerator is used to reduce the time consumption of feature selection. Algorithm 2 takes much less time than ES. Take data "Wdbc" (ID:4) as an example, When using ES to solve the feature selection problem, the time consumption is 2.5765 seconds (ES-NMBO), 102.9289 seconds (ES-NGA), 64.3960 (ES-NAFSA) seconds and 70.5950 seconds (ES-NFOA), respectively, while the time required by Algorithm 2 which uses random sampling is 1.3710 seconds (REP-NMBO), 34.0406 seconds (REP-NGA), 17.7254 seconds (REP-NAFSA) and 16.5258 seconds (REP-NFOA), respectively. In the ensemble case, the number of data sets used by Algorithm 2 and ES is same. Therefore, it can be seen that the introduction of random sampling accelerator can indeed speed up the process of feature selection.

Table 3 Time consumptions of Algorithms for feature selection (Seconds).

ID	NMBO	ES-NMBO	REP-NMBO	NGA	ES-NGA	REP-NGA	NAFSA	ES-NAFSA	REP-NAFSA	NFOA	ES-NFOA	REP-NFOA
1	47.9497	253.9560	**8.2515**	136.1728	716.8000	**90.9692**	212.2761	1069.9139	**49.7526**	304.8315	1608.6263	**114.8214**
2	**0.1042**	0.5490	0.1131	1.2168	6.2261	**1.0003**	4.9257	25.5569	**0.8921**	2.1948	11.2884	**0.4109**
3	4.3831	22.1658	**1.3257**	40.6925	207.2552	**13.3648**	14.3639	74.9855	**4.7697**	51.6666	264.2128	**9.4115**
4	**0.5040**	2.5765	1.3710	**20.2095**	102.9289	34.0406	**12.2268**	64.3960	17.7254	**13.7486**	70.5950	16.5258
5	**0.1201**	0.6114	0.2844	**0.4513**	2.3071	6.5563	**0.2672**	1.3571	4.1367	**0.5253**	2.7144	1.8853
6	2.0721	10.9357	**0.5521**	57.0385	291.7061	**14.1218**	15.1733	76.8530	**6.3251**	28.4386	145.6626	**7.9157**
7	**0.0405**	0.2209	0.2158	**0.2770**	1.3978	2.6217	**0.1680**	0.8459	0.9258	**0.2177**	1.1212	1.0531
8	**0.2791**	1.4534	0.4503	1.6414	8.2940	**1.5702**	0.6731	3.4060	**0.6458**	3.6413	18.9342	**1.7580**
9	0.8446	4.2713	**0.8368**	5.3340	27.5003	**10.0480**	14.0919	71.0319	**9.0919**	13.9799	71.9142	**5.0426**
10	6.8049	34.1967	**3.8049**	211.8428	1089.1899	**53.2515**	580.2302	2923.1996	**59.8269**	12.2091	62.5409	**30.8624**
11	0.1398	**0.7521**	0.8305	**1.0032**	5.1942	2.4632	**0.2366**	1.2534	3.0369	**1.6109**	8.3149	6.2625
12	**0.0793**	0.4083	0.1402	**0.7359**	3.6898	0.9724	**0.1400**	0.7367	0.4859	0.4668	2.3827	**0.3900**
13	**0.0581**	0.3070	0.1764	**1.3016**	6.6825	1.6135	**0.6021**	3.0944	0.7517	0.8506	4.4071	**0.7751**
14	**0.0398**	0.2088	0.2031	**0.3978**	2.0586	1.4608	1.1996	6.0332	**1.1261**	0.6321	3.2428	**0.4615**
15	36.8335	190.4366	**17.7251**	222.3042	1112.9437	**108.2044**	1079.4309	5465.6982	**136.7422**	3471.1192	17709.6503	**308.9361**
16	49.8467	255.9628	**26.7037**	1234.5499	6338.3027	**269.7378**	2897.5100	14794.3963	**373.5053**	1188.5205	6077.3808	**268.3441**
17	253.0885	1326.1837	**199.1138**	2335.7810	11959.1990	**543.2049**	1081.7891	5575.5410	**457.3345**	3121.2945	15824.3389	**876.7681**
18	1325.1548	6825.0366	**548.4141**	3244.6915	16288.3512	**1005.9624**	3398.1025	17373.1390	**809.0720**	4100.7325	25457.4364	**1658.3990**
19	294.4798	1480.2029	**45.0275**	746.3525	3798.6881	**175.6124**	772.4650	3949.1503	**177.5782**	1101.5132	5706.5316	**337.9912**
20	109.6315	550.3501	**84.7585**	967.2155	4988.3308	**367.3605**	942.1020	4743.0127	**314.0340**	2056.6406	10587.2031	**685.5469**
average	106.6227	548.0393	**47.0149**	461.4605	2347.8523	**135.2068**	551.3987	2811.1800	**121.3929**	773.7417	4181.9249	**216.6781**

(2) For the Algorithm 2, when the number of samples is greater than 2000, it can also be found that Algorithm 2 is faster than NMBO, NGA, NAFSA and NFOA. Take data "Cardiotocography" (ID:1) as an example, the number of samples is 2126, and the time consumed by NMBO, NGA, NAFSA and NFOA is 47.9497 seconds, 136.1728 seconds, 212.2761 seconds and 304.8315 seconds, respectively. The time consumed by Algorithm 2 is only 8.2515 seconds (REP-NMBO), 90.9692 seconds (REP-NGA), 49.7526 seconds (REP-NAFSA) and 114.8214 seconds (REP-NFOA). Take data "MAGIC Gamma Telescope" (ID:18) as another example, the number of samples is 19020, and the time consumed by NMBO, NGA, NAFSA and NFOA is 1325.1548 seconds, 3244.6915 seconds, 3398.1025 seconds and 4100.7325 seconds, respectively. The time consumed by Algorithm 2 is only 548.4141 seconds (REP-NMBO), 1005.9624 seconds (REP-NGA), 809.0720 seconds (REP-NAFSA) and 1658.3990 seconds (REP-NFOA).

5.2.2 Comparisons of classification performances.

The classification performance of NMBO, NGA, NAFSA, NFOA, ES and REP are also suitably compared. It should be noted that 20 radii are used to obtain the selection of features and the results below represent the average classification accuracy and stability of the classification in relation to the 20 relevant feature selections.

Two classifiers, namely, KNN (K=3) and CART (with default parameters), are utilized to demonstrate the classification performance. The main reason for using the KNN classifier is that it is simple and does not require parameter estimation, which is especially suitable for multi-classification problems. The reason for using the CART classifier is that its classification rules are easy to understand and have high accuracy. Table 4 and Table 5 present the average KNN and CART classification stability values obtained by different algorithms in the test set. Table 6 and Table 7 show the mean values of KNN and CART classification accuracy obtained by different algorithms in the test sets.

Table 4-7 shows that Algorithm 2 performs better than NMBO, NGA, NAFSA, NFOA and ES by using KNN and CART classifiers. In particular, the use of Algorithm 2 significantly improves the stability of the classification. This indicates that Algorithm 2 can generate more satisfactory feature subsets. Then, Table 4-7 combined with the time consumption comparison of Table 3 shows:

(1) Algorithm 2 can also improve the classification performance when compared to NMBO, NGA, NAFSA and NFOA. Consider the data "waveform" (ID:15). Compared with NMBO, NGA, NAFSA and NFOA, Algorithm 2 is improved by 18.84%, 14.44%, 15.51% and 11.15% respectively on KNN classification stability (Table 4). The classification stability of CART (Table 5) was increased by 18.63%, 13.63%, 12.49% and 10.21% respectively. The classification accuracy of KNN (Table 6) is also improved by 13.25%, 5.77%, 7.00% and 5.95% respectively. The accuracy of CART classification (Table 7) was

Table 4 Classification stabilities based on KNN classifiers.

ID	NMBO	ES-NMBO	REP-NMBO	NGA	ES-NGA	REP-NGA	NAFSA	ES-NAFSA	REP-NAFSA	NFOA	ES-NFOA	REP-NFOA
1	0.8443	0.8755	**0.9020**	0.8498	0.8597	**0.8779**	0.8295	0.8430	**0.8763**	0.8591	0.8705	**0.9044**
2	0.7698	0.8331	**0.8547**	0.7303	0.7673	**0.8041**	0.7616	0.7950	**0.8244**	0.7866	0.8187	**0.8447**
3	0.8273	0.8409	**0.8778**	0.8073	0.8233	**0.8509**	0.8040	0.8180	**0.8384**	0.8082	0.8206	**0.8472**
4	0.9703	0.9853	**0.9834**	0.9656	0.9679	**0.9753**	0.9622	0.9724	**0.9729**	0.9644	0.9757	**0.9910**
5	0.9063	0.9455	**0.9756**	0.8876	0.9108	**0.9606**	0.8045	0.8609	**0.9181**	0.8630	0.8979	**0.9339**
6	0.9235	0.9366	**0.9520**	0.9327	0.9391	**0.9485**	0.9230	0.9441	**0.9550**	0.9108	0.9305	**0.9576**
7	0.9229	0.9344	**0.9610**	0.9330	0.9448	**0.9556**	0.8860	0.9142	**0.9539**	0.9109	0.9319	**0.9660**
8	0.8576	0.8600	**0.8913**	0.8593	0.8788	**0.9033**	0.8513	0.8703	**0.9066**	0.8267	0.8457	**0.8934**
9	0.7382	0.7983	**0.8303**	0.7294	0.7659	**0.8162**	0.7187	0.7706	**0.8128**	0.7330	0.7901	**0.8305**
10	0.9038	0.9192	**0.9323**	0.9219	0.9258	**0.9400**	0.9033	0.9138	**0.9223**	0.9023	0.9134	**0.9235**
11	0.9060	0.9140	**0.9442**	0.9104	0.9121	**0.9671**	0.8951	0.9246	**0.9572**	0.8909	0.9289	**0.9664**
12	0.9407	0.9523	**0.9829**	0.9561	0.9706	**0.9984**	0.9223	0.9421	**0.9876**	0.9397	0.9529	**0.9947**
13	0.7563	0.7609	**0.8370**	0.8510	0.8762	**0.9031**	0.8345	0.8642	**0.9054**	0.8374	0.8443	**0.9092**
14	0.8861	0.8977	**0.9133**	0.9376	0.9531	**0.9716**	0.9074	0.9374	**0.9539**	0.8750	0.8934	**0.9392**
15	0.5870	0.6301	**0.6976**	0.7602	0.8395	**0.8700**	0.7414	0.8034	**0.8564**	0.7811	0.8002	**0.8682**
16	0.5542	0.6202	**0.6707**	0.8839	0.8840	**0.9246**	0.8983	0.9131	**0.9308**	0.8610	0.8723	**0.9136**
17	0.9227	0.9562	**0.9933**	0.9378	0.9531	**0.9865**	0.9351	0.9449	**0.9754**	0.9372	0.9479	**0.9782**
18	0.8721	0.9009	**0.9221**	0.8869	0.8987	**0.9369**	0.8918	0.9119	**0.9531**	0.9309	0.9578	**0.9763**
19	0.9067	0.9353	**0.9743**	0.8935	0.9111	**0.9610**	0.8736	0.9335	**0.9855**	0.8956	0.9273	**0.9710**
20	0.9394	0.9581	**0.9773**	0.9447	0.9623	**0.9854**	0.9331	0.9544	**0.9746**	0.9407	0.9685	**0.9838**
average	0.8468	0.8727	**0.9037**	0.8789	0.8972	**0.9269**	0.8638	0.8916	**0.9230**	0.8727	0.8944	**0.9296**

Table 5 Classification stabilities based on CART classifier.

ID	NMBO	ES-NMBO	REP-NMBO	NGA	ES-NGA	REP-NGA	NAFSA	ES-NAFSA	REP-NAFSA	NFOA	ES-NFOA	REP-NFOA
1	0.8112	0.8363	**0.8636**	0.8060	0.8260	**0.8643**	0.7959	0.8275	**0.8647**	0.7909	0.8221	**0.8585**
2	0.7190	0.7639	**0.8034**	0.7159	0.7750	**0.8117**	0.7565	0.8265	**0.8665**	0.7365	0.7717	**0.8272**
3	0.7125	0.7385	**0.7451**	0.6899	0.7037	**0.7454**	0.7310	0.7745	**0.8089**	0.6862	0.7020	**0.7588**
4	0.9260	0.9346	**0.9378**	0.8997	0.9023	**0.9153**	0.8979	0.9208	**0.9268**	0.8850	0.9144	**0.9299**
5	0.8944	0.9231	**0.9669**	0.8696	0.9145	**0.9456**	0.8127	0.8519	**0.9041**	0.8365	0.8580	**0.9294**
6	0.8617	0.8619	**0.8723**	0.8655	0.8721	**0.8869**	0.8839	0.9167	**0.9261**	0.8694	0.8917	**0.9222**
7	0.8301	0.8683	**0.8996**	0.8272	0.8542	**0.9031**	0.8783	0.9084	**0.9487**	0.8421	0.8763	**0.9277**
8	0.7756	0.8030	**0.8401**	0.7965	0.8031	**0.8396**	0.7653	0.8212	**0.8608**	0.7463	0.7927	**0.8418**
9	0.6980	0.7169	**0.7590**	0.6860	0.7286	**0.7599**	0.6919	0.7315	**0.7728**	0.6868	0.7509	**0.7825**
10	0.8051	0.8364	**0.8592**	0.8068	0.8252	**0.8594**	0.8132	0.8642	**0.8823**	0.8024	0.8430	**0.8827**
11	0.8060	0.8567	**0.8929**	0.8172	0.8739	**0.9034**	0.8060	0.8832	**0.9178**	0.8069	0.8690	**0.9162**
12	0.8474	0.9041	**0.9330**	0.8858	0.9268	**0.9771**	0.8371	0.9048	**0.9571**	0.8749	0.8931	**0.9764**
13	0.7080	0.7545	**0.7763**	0.7541	0.7779	**0.8157**	0.7819	0.8291	**0.8555**	0.7560	0.7825	**0.8316**
14	0.8803	0.9054	**0.9260**	0.8706	0.9120	**0.9437**	0.8733	0.8914	**0.9333**	0.8105	0.8695	**0.9036**
15	0.6102	0.6832	**0.7239**	0.6883	0.7326	**0.7821**	0.6691	0.7001	**0.7527**	0.6974	0.7164	**0.7686**
16	0.7903	0.8079	**0.9142**	0.9798	0.9868	**0.9952**	0.9748	0.9800	**0.9845**	0.9396	0.9646	**0.9843**
17	0.9174	0.9184	**0.9421**	0.9001	0.9226	**0.9406**	0.9017	0.9164	**0.9596**	0.9131	0.9378	**0.9640**
18	0.8123	0.8299	**0.8509**	0.8310	0.8500	**0.8832**	0.8298	0.8556	**0.8985**	0.7966	0.8482	**0.8768**
19	0.7785	0.8159	**0.8473**	0.7584	0.7879	**0.8107**	0.7492	0.8023	**0.8458**	0.7505	0.8206	**0.8600**
20	0.8446	0.8847	**0.9199**	0.8455	0.8688	**0.8927**	0.8389	0.8770	**0.8965**	0.8246	0.8834	**0.9105**
average	0.8014	0.8322	**0.8637**	0.8147	0.8422	**0.8738**	0.8144	0.8542	**0.8881**	0.8026	0.8404	**0.8826**

Table 6 Classification accuracies over KNN classifier.

ID	NMBO	ES-NMBO	REP-NMBO	NGA	ES-NGA	REP-NGA	NAFSA	ES-NAFSA	REP-NAFSA	NFOA	ES-NFOA	REP-NFOA
1	0.7401	0.7434	**0.7478**	0.7426	0.7433	**0.7503**	0.7346	0.7441	**0.7469**	0.7758	0.7838	**0.7896**
2	0.7719	0.7790	**0.8067**	0.7554	0.7615	**0.7863**	0.7565	0.7644	**0.7837**	0.7287	0.7361	**0.7485**
3	0.6562	0.6634	**0.6731**	0.6343	0.6447	**0.6501**	0.6570	0.6644	**0.6724**	0.6504	0.6604	**0.6650**
4	0.9696	0.9726	**0.9705**	0.9550	0.9571	**0.9606**	0.9374	0.9464	**0.9525**	0.9664	0.9790	**0.9744**
5	0.9311	0.9400	**0.9675**	0.9300	0.9395	**0.9633**	0.8444	0.8490	**0.9345**	0.8973	0.9125	**0.9400**
6	0.8704	0.8772	**0.8855**	0.8664	0.8686	**0.8876**	0.8432	0.8483	**0.8655**	0.8626	0.8732	**0.8816**
7	0.9433	0.9456	**0.9644**	0.9333	0.9404	**0.9461**	0.9120	0.9268	**0.9429**	0.9203	0.9354	**0.9466**
8	0.8012	0.8042	**0.8219**	0.7427	0.7428	**0.7570**	0.7674	0.7740	**0.7902**	0.7646	0.7728	**0.7863**
9	0.6658	0.6667	**0.6901**	0.6924	0.7024	**0.7126**	0.6855	0.6923	**0.7102**	0.7036	0.7137	**0.7263**
10	0.8411	0.8471	**0.8505**	0.8530	0.8547	**0.8598**	0.8745	0.8890	**0.8897**	0.8311	0.8410	**0.8421**
11	0.9458	0.9576	**0.9613**	0.9453	0.9609	**0.9661**	0.9233	0.9311	**0.9580**	0.9234	0.9354	**0.9680**
12	0.9065	0.9166	**0.9254**	0.8819	0.8924	**0.8982**	0.9373	0.9522	**0.9638**	0.8943	0.9035	**0.9078**
13	0.7248	0.7313	**0.7476**	0.8137	0.8142	**0.8176**	0.8087	0.8216	**0.8280**	0.7874	0.8098	**0.8163**
14	0.8356	0.8403	**0.8461**	0.8709	0.8726	**0.8753**	0.8800	0.8890	**0.9083**	0.8377	0.8493	**0.8636**
15	0.6526	0.6566	**0.7391**	0.7727	0.7858	**0.8173**	0.7631	0.7716	**0.8165**	0.7643	0.7733	**0.8098**
16	0.5559	0.5579	**0.6625**	0.8620	0.8726	**0.8683**	0.8561	0.8693	**0.8614**	0.8634	0.8770	**0.8797**
17	0.9547	0.9579	**0.9887**	0.9461	0.9473	**0.9831**	0.9377	0.9518	**0.9727**	0.9225	0.9375	**0.9833**
18	0.7779	0.7809	**0.8189**	0.7921	0.8052	**0.8501**	0.7936	0.8065	**0.8348**	0.7624	0.8016	**0.8341**
19	0.9386	0.9506	**0.9681**	0.9235	0.9281	**0.9561**	0.8859	0.8921	**0.9279**	0.8856	0.8996	**0.9389**
20	0.9156	0.9204	**0.9587**	0.9083	0.9162	**0.9373**	0.8854	0.8903	**0.9290**	0.9051	0.9154	**0.9550**
average	0.8199	0.8255	**0.8497**	0.8411	0.8475	**0.8622**	0.8342	0.8437	**0.8645**	0.8323	0.8455	**0.8628**

Table 7 Classification accuracies over CART classifier.

ID	NMBO	ES-NMBO	REP-NMBO	NGA	ES-NGA	REP-NGA	NAFSA	ES-NAFSA	REP-NAFSA	NFOA	ES-NFOA	REP-NFOA
1	0.7959	0.8049	**0.8290**	0.7830	0.7999	**0.8135**	0.7661	0.7834	**0.8056**	0.7984	0.8094	**0.8373**
2	0.7581	0.7659	**0.7945**	0.7586	0.7754	**0.8068**	0.7653	0.7769	**0.8137**	0.7165	0.7239	**0.7569**
3	0.7005	0.7092	**0.7251**	0.6444	0.6599	**0.6736**	0.6994	0.7098	**0.7395**	0.6609	0.6718	**0.6936**
4	0.9290	0.9470	**0.9385**	0.9212	0.9316	**0.9302**	0.9229	0.9381	**0.9390**	0.9172	0.9387	**0.9332**
5	0.9191	0.9295	**0.9736**	0.9070	0.9173	**0.9564**	0.8470	0.8626	**0.9100**	0.8801	0.8933	**0.9312**
6	0.8427	0.8595	**0.8621**	0.8277	0.8382	**0.8542**	0.8310	0.8451	**0.8508**	0.8631	0.8751	**0.8861**
7	0.8651	0.8741	**0.8848**	0.8965	0.9063	**0.9529**	0.8892	0.9031	**0.9283**	0.8733	0.8903	**0.9203**
8	0.7531	0.7712	**0.8004**	0.7611	0.7690	**0.8016**	0.7440	0.7520	**0.8019**	0.7422	0.7577	**0.8051**
9	0.6488	0.6626	**0.6712**	0.6987	0.7074	**0.7306**	0.6747	0.6905	**0.7089**	0.6766	0.6903	**0.7188**
10	0.8069	0.8229	**0.8430**	0.8001	0.8158	**0.8323**	0.8224	0.8340	**0.8581**	0.7998	0.8148	**0.8316**
11	0.8671	0.8878	**0.9211**	0.8769	0.8969	**0.9382**	0.8758	0.8853	**0.9468**	0.8589	0.8781	**0.9351**
12	0.8771	0.8923	**0.9274**	0.8634	0.8773	**0.8941**	0.8968	0.9069	**0.9538**	0.8817	0.8997	**0.9152**
13	0.7062	0.7215	**0.7353**	0.7806	0.7937	**0.8095**	0.7786	0.7902	**0.8045**	0.7540	0.7672	**0.7754**
14	0.8930	0.9066	**0.9177**	0.8943	0.9162	**0.9327**	0.8938	0.9104	**0.9297**	0.8435	0.8540	**0.8850**
15	0.6809	0.6954	**0.7637**	0.7334	0.7494	**0.7915**	0.7258	0.7395	**0.7817**	0.7306	0.7433	**0.7806**
16	0.8493	0.8674	**0.9320**	0.9836	0.9877	**0.9881**	0.9818	0.9860	**0.9906**	0.9672	0.9779	**0.9800**
17	0.9271	0.9486	**0.9561**	0.9193	0.9404	**0.9577**	0.9062	0.9251	**0.9371**	0.8971	0.9174	**0.9506**
18	0.7765	0.7906	**0.8237**	0.7639	0.7736	**0.8220**	0.7550	0.7720	**0.8241**	0.7772	0.7938	**0.8263**
19	0.8484	0.8625	**0.9006**	0.8345	0.8447	**0.8907**	0.7630	0.7790	**0.8170**	0.7841	0.7959	**0.8339**
20	0.8566	0.8670	**0.8969**	0.8427	0.8552	**0.8918**	0.8517	0.8703	**0.8971**	0.8509	0.8687	**0.9187**
average	0.8151	0.8293	**0.8548**	0.8245	0.8378	**0.8634**	0.8195	0.8330	**0.8619**	0.8137	0.8281	**0.8558**

Table 8 The length of feature selection result.

ID	NMBO	ES-NMBO	REP-NMBO	NGA	ES-NGA	REP-NGA	NAFSA	ES-NAFSA	REP-NAFSA	NFOA	ES-NFOA	REP-NFOA
1	**16.2100**	16.8900	17.1900	**15.7200**	16.3400	16.2200	**14.2300**	15.3300	15.0200	**15.7400**	16.3000	16.6000
2	24.1800	25.4700	**22.9900**	24.4100	25.9300	**22.3700**	17.8900	19.1900	**14.3900**	20.9200	22.2200	**14.7400**
3	13.8100	14.1000	**13.6500**	13.6300	14.3800	**12.8300**	12.6800	13.6200	**11.3900**	13.3200	14.3200	**12.6700**
4	**18.9300**	19.4500	19.8300	18.8100	19.4300	**18.7700**	**14.9800**	15.3700	15.5600	**17.0000**	17.8200	18.2000
5	**17.1900**	18.0100	19.4100	**17.2100**	18.0600	18.2100	**11.0400**	11.5000	13.3400	**13.0000**	13.7100	14.7500
6	20.6300	21.0500	**18.2800**	20.1000	20.7200	**17.2800**	17.4900	18.3000	**14.2700**	19.2100	20.3100	**16.2800**
7	**7.9100**	8.2300	9.1000	**7.5200**	7.9900	8.2900	**6.0600**	6.3100	7.0200	**6.6100**	6.9500	7.1900
8	**75.4100**	78.6600	76.1600	74.8600	76.9000	**73.5500**	**44.4000**	46.3300	44.9600	**44.6300**	46.3800	44.7100
9	**14.0200**	14.4600	16.2500	**13.1100**	13.9100	14.1800	**12.0400**	12.6200	13.0900	**15.1800**	15.5100	15.2000
10	29.6600	31.5700	**27.1100**	27.6300	28.9600	**25.7100**	25.2400	27.2300	**20.5900**	27.5000	28.4100	**25.1800**
11	**30.5600**	31.8900	32.8000	**30.3400**	31.2600	31.1300	**18.3400**	18.8300	21.8200	**19.0900**	19.7200	23.5400
12	**10.1200**	10.8200	10.7700	**10.0500**	10.4000	10.0900	**6.5100**	6.8000	6.6500	6.7700	7.2100	**6.3900**
13	9.6400	9.9300	**9.3400**	8.9800	9.2500	**7.9200**	8.2500	8.7800	**6.9700**	9.1200	9.8100	**7.5200**
14	**17.5400**	18.0400	18.2200	17.0800	17.8000	**17.0500**	11.7900	12.3100	**11.5000**	13.6600	14.1700	**11.8400**
15	**12.5400**	12.7900	15.3300	**14.4200**	15.3900	16.2200	**13.1200**	13.5300	14.9900	**15.1200**	15.5800	15.8100
16	**13.0500**	13.6800	16.7200	**19.0700**	20.1300	19.8000	**18.6500**	19.9900	19.5300	**14.7100**	15.2800	17.3800
17	10.2700	10.5500	**9.8600**	**11.2800**	12.0500	11.4200	10.5200	10.9500	**9.4600**	12.0200	12.7400	**11.5700**
18	9.7200	10.0600	**9.5600**	**9.6200**	9.8400	9.6500	9.2000	9.9100	**8.4700**	**9.6300**	10.4000	9.6900
19	14.4400	14.7300	**13.2900**	14.0200	14.4000	**13.9000**	**13.3600**	14.1800	13.5100	14.0100	14.8400	**13.6100**
20	19.6800	21.0300	**19.1800**	16.5300	17.5500	**15.5300**	16.2400	17.3600	**14.3500**	16.6900	17.5500	**15.1400**
average	**19.2755**	20.0705	19.7520	19.2195	20.0345	**19.0060**	15.1015	15.9220	**14.8440**	16.1965	16.9615	**15.9005**

improved by 11.93%, 7.92%, 7.70% and 6.84% respectively. Similarly, compared with Algorithm 1, the classification performance of Algorithm 2 has been greatly improved.

(2) Algorithm 2 can also improve the classification performance when compared to ES. Consider the data "MAGIC Gamma Telescope" (ID:18). Compared with ES, Algorithm 2 is improved by 2.35%, 4.25%, 4.52% and 1.93% respectively on KNN classification stability (Table 4). The classification stability of CART (Table 5) was increased by 2.53%, 3.91%, 5.01% and 3.37% respectively. The classification accuracy of KNN (Table 6) is also improved by 4.87%, 5.58%, 3.51% and 4.05% respectively. The accuracy of CART classification (Table 7) was improved by 4.19%, 6.26%, 4.09% and 6.84% respectively.

(3) Algorithm 2 has an acceptable time consumption. Consider the datasets (ID: 1, 3, 6, 9, 10, 15, 16, 17, 18, 19, 20). Algorithm 2 takes less time than NMBO, NGA, NAFSA, NFOA and ES, and its classification performance is also higher than these algorithms.

Overall, the performance of Algorithm 2 is better than that of NMBO, NGA, NAFSA, NFOA and ES.

5.2.3 The length of feature selection result.

In this part, the length of the feature selection results is compared and displayed. The shorter the length, the better it can maintain classification performance in the face of data loss. Moreover, the classifier can also classify faster with a shorter feature length. The length of feature selection result obtained by Algorithm 2 is the mean value of several feature selection results in its ensemble. Five perturbations were used in this experiment, which means the length is the mean value of the five feature selection results. Table 8 shows the length of feature selection results obtained by NMBO, NGA, NAFSA, NFOA, ES and REP. The following can be drawn:

(1) Algorithm 2's performance is comparable to NMBO, NGA, NAFSA and NFOA. In the comparison of the length of feature selection results between REP-NMBO (Algorithm 2) and NMBO, NMBO has gained some advantages. And, the comparison of their mean values also shows this. Specifically, the average length of NMBO is 19.2755, and the average length of REP-MBO is 19.7520. However, Algorithm 2 also gained some advantage in the next three comparisons. The length of NGA, NAFSA and NFOA is 19.2195, 15.1015 and 16.1965, respectively, while the length of Algorithm 2 is only 19.0060 (REP-NGA), 14.8440 (REP-NAFSA) and 15.9005 (REP-NFOA), respectively.

(2) Algorithm 2's performance is also comparable to ES. In the comparison of the length of feature selection results between REP and ES, REP has gained some advantages. Specifically, the average length of ES is 20.0705 (ES-NMBO), 20.0345 (ES-NGA), 15.9220 (ES-NAFSA) and 16.9615 (ES-NFOA), respectively, while the length of REP is only 19.7520 (REP-NMBO), 19.0060 (REP-NGA), 14.8440 (REP-NAFSA) and 15.9005 (REP-NFOA), respectively.

Therefore, it is concluded that Algorithm 2 can provide feature selection results with reasonable length and high classification ability under acceptable time consumption.

6 Conclusions and future work

In this paper, we developed a novel framework for improving meta-heuristic algorithms. In the first step, the framework employs a voting ensemble strategy to enhance classification stability. Second, the framework adopts a data perturbation strategy to increase the classification accuracy of the voting ensemble strategy. Thirdly, the framework takes a random sampling accelerator into account to solve the problem of excessive time consumption. Finally, the experimentally achieved results on 20 UCI datasets show that the proposed framework could not only noticeably enhance the classification performance of meta-heuristic algorithms, but also substantially reduce the consumption time of these algorithms.

There may be many research fields that may need to be focused on in the future, and some of the following fields are considered worth exploring.

(1) The proposed method actually discusses the accelerated meta-heuristic feature selection problem at the sample level, which can be further studied from both the feature and sample levels in the future.
(2) The strategy of Algorithm 2 can be implemented not only by neighborhood rough sets, but also by other rough sets, such as fuzzy rough sets.
(3) Algorithm 2 is an algorithm framework, which can not only be applied to the four meta-heuristic algorithms mentioned in this paper, but also improve the experimental results of other algorithms.
(4) The impact of data perturbation strategy on ensemble classification deserves to be explored deeply.

CRediT author statement

Shuaishuai Zhang: Conceptualization, Methodology, Formal analysis, Investigation, Writing - Original Draft, Writing - Review & Editing, Project administration. Keyu Liu: Resources, Data Curation. Taihua Xu: Investigation, Resources, Data Curation, Writing - Review & Editing. Xibei Yang: Investigation, Resources. Ao Zhang: Data Curation.

Declaration of competing interest

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

Data availability

The authors do not have permission to share data.

Acknowledgment

This work is supported by the National Natural Science Foundation of China (Grant Nos. 62006099, 62076111), and the Key Laboratory of Oceanographic Big Data Mining & Application of Zhejiang Province (No. OBDMA202104).

References

[1] Ding W P, Nayak J, Naik B, Pelusi D, Mishara M (2021) Fuzzy and real-coded chemical reaction optimization for intrusion detection in industrial big data environment. IEEE Trans Ind Inf 17(6):4298-4307

[2] Dong L J, Wang R H, Chen D G (2023) Incremental feature selection with fuzzy rough sets for dynamic data sets. Fuzzy Sets Syst 467:108503

[3] Zhang X, Mei C L, Li J H, Yang Y Y, Qian T (2023) Instance and feature selection using fuzzy rough sets: a bi-selection approach for data reduction. IEEE Trans Fuzzy Syst 31(6):1981-1994

[4] Chen J K, Lin Y J, Mi J S, Li S Z, Ding W P (2022) A spectral feature selection approach with kernelized fuzzy rough sets. IEEE Trans Fuzzy Syst 30(8):2886-2901

[5] Liu K Y, Li T R, Yang X B, Yang X, Liu D (2022) Neighborhood rough set based ensemble feature selection with cross-class sample granulation. Appl Soft Comput 131: 109747

[6] Ismail A, Sandell M (2022) A low-complexity endurance modulation for flash memory. IEEE Transactions on Circuits and Systems II: Express Briefs 69(2):424-428

[7] Tang Y J, Zhang X M (2022) Low-complexity resource-shareable parallel generalized integrated interleaved encoder. IEEE Trans Circuits Syst I Regul Pap 69(2):694-706

[8] Li Z J, Kamnitsas K, Glocker B (2021) Analyzing overfitting under class imbalance in neural networks for image segmentation. IEEE Trans Med Imaging 40(3):1065-1077

[9] Park Y B, Ho J C (2021) Tackling overfitting in boosting for noisy healthcare data. IEEE Trans Knowl Data Eng 33(7):2995-3006

[10] Baisantry M, Sao A K, Shukla D P (2022) Discriminative spectral spatial feature extraction-based band selection for hyper spectral image classification. IEEE Trans Geosci Remote Sens 60:1-14

[11] Ding W P, Triguero I, Lin C T (2021) Coevolutionary fuzzy at tribute order reduction with complete attribute-value space tree. IEEE Trans Emerging Top Comput Intell 5(1):29-41

[12] Momeni N, Valdés A A, Rodrigues J, Sandi C, Atienza D (2022) CAFS: Cost-Aware Features Selection Method for Multimodal Stress Monitoring on Wearable Devices. IEEE Trans Biomed Eng 69(3):1072-1084

[13] Yan W W, Ba J, Xu T H, Yu H L, Shi J L, Han B (2022) Beam-Influenced Attribute Selector for Producing Stable Reduct. Mathematics 10(4):553

[14] Wei W, Wu X Y, Liang J Y, Cui J B, Sun Y J (2018) Discernibility matrix based incremental attribute reduction for dynamic data. Knowl Based Syst 140:142-157

[15] Wei W, Cui J B, Liang J Y, Wang J H (2016) Fuzzy rough approximations for set-valued data. Inf Sci 360:181-201

[16] Etesami O, Haemers W (2020) On NP-hard graph properties characterized by the spectrum. Discret Appl Math 285:526-529

[17] Zhang A, Chen Y, Chen L, Chen G T (2018) On the NP-hardness of scheduling with time restrictions. Discret Optim 28:54-62

[18] Guha R, Ghosh K K, Bera S K, Sarkar R, Mirjalili S (2023) Discrete equilibrium optimizer combined with simulated annealing for feature selection. J Comput Sci 67:1877-7503

[19] Elaziz M A, Ouadfel S, El-Latif A A A, Ali Ibrahim R (2022) Feature Selection Based on Modified Bio-inspired Atomic Orbital Search Using Arithmetic Optimization and Opposite-Based Learning. Cognit Comput 14(6):2274-2295

[20] Penmatsa R K V, Kalidindi A, Mallidi S K R (2020) Feature reduction and optimization of malware detection system using ant colony optimization and rough sets. Int J Inf Secur Priv 14(3):95-114

[21] Luan X Y, Li Z P, Liu T Z (2016) A novel attribute reduction algorithm based on rough set and improved artificial fish swarm algorithm. Neurocomputing 174:522-529

[22] Wang G G, Deb S, Cui Z H (2019) Monarch butterfly optimization. Neural Comput Appl 31(7):1995-2014

[23] Shreem S S, Turabieh H, Azwari S A, Baothman F (2022) Enhanced binary genetic algorithm as a feature selection to predict student performance. Soft Comput, 26(4):1811-1823

[24] Ghaemi M, Feizi-Derakhshi M-R (2016) Feature selection using Forest Optimization Algorithm. Pattern Recognit 60:121-129

[25] Campagner A, Ciucci D, Hüllermeier E (2021) Rough set-based feature selection for weakly labeled data. Int J Approx Reason 136:150-167

[26] Pawlak Z (2002) Rough sets and intelligent data analysis. Inf Sci 147(1-4):1-12

[27] Tawhid M A, Ibrahim A M (2020) Feature selection based on rough set approach, wrapper approach, and binary whale optimization algorithm. Int. J. Mach. Learn. Cybern. 11(3):573-602

[28] Xu T H, Wang G Y, Yang J (2020) Finding strongly connected components of simple digraphs based on granulation strategy. Int J Approx Reason 118:64-78

[29] Fujita H, Gaeta A, Loia V, Orciuoli F (2020) Hypotheses analysis and assessment in counterterrorism activities: a method based on OWA and fuzzy probabilistic rough sets. IEEE Trans Fuzzy Syst 28(5): 831-845

[30] Zhang C, Li D Y, Liang J Y (2020) Multi-granularity three-way decisions with adjustable hesitant fuzzy linguistic multigranulation decision-theoretic rough sets over two universes. Inf Sci 507:665-683

[31] Qian J, Han X, Yu Y, Liu C H (2023) Multi-granularity decision-theoretic rough sets based on the fuzzy T-equivalence relation with new strategies. J Intell Fuzzy Syst 44(4): 5617-5631

[32] Yang X B, Liang S C, Yu H L (2019) Pseudo-label neighborhood rough set: Measures and attribute reductions. Int J Approx Reason 105:112-129

[33] Hu Q H, Yu D R, Liu J F, Wu C X (2008) Neighborhood rough set based heterogeneous feature subset selection. Inf Sci 178(18):3577-3594

[34] Zhang K, Zhan J M, Wu W Z (2021) On multi-criteria decision-making method based on a fuzzy rough set model with fuzzy α-neighborhoods. IEEE Trans Fuzzy Syst 29(9): 2491-2505

[35] Yao Y Y (1998) Relational interpretations of neighborhood operators and rough set approximation operators. Inf Sci 111:239–259

[36] An S, Guo X Y, Wang C Z, Guo G, Dai J H (2023) A soft neighborhood rough set model and its applications. Inf Sci 624:185-199

[37] Yang L, Qin K Y, Sang B B, Xu W H (2021) Dynamic fuzzy neighborhood rough set approach for interval-valued information systems with fuzzy decision. Appl Soft Comput 111:107679

[38] Zou L, Li H X, Jiang W, Yang X H (2019) An Improved Fish Swarm Algorithm for Neighborhood Rough Set Reduction and its Application. IEEE Access 7:90277-90288

[39] Feng J D, Gong Z T (2022) A Novel Feature Selection Method With Neighborhood Rough Set and Improved Particle Swarm Optimization. IEEE Access 10:33301-33312

[40] Sahlol A T, Elaziz M A, Al-Qaness M A A, Kim S (2020) Handwritten Arabic Optical Character Recognition Approach Based on Hybrid Whale Optimization Algorithm With Neighborhood Rough Set. IEEE Access 8:23011-23021

[41] Zhang Y D, Mao Z D, Li J T, Tian Q (2014) Salient region detection for complex background images using integrated features. Inf Sci 281:586-600

[42] Kanna P R, Santhi P (2021) Unified Deep Learning approach for Efficient Intrusion Detection System using Integrated Spatial-Temporal Features. Knowl Based Syst 226:107132

[43] Gong Z C, Liu Y X, Xu T H, Wang P X, Yang X B (2022) Unsupervised attribute reduction: improving effectiveness and efficiency. Int J Mach Learn Cybern 13(11):3645-3662

[44] Yang X B, Yao Y Y (2018) Ensemble selector for attribute reduction. Appl Soft Comput 70:1-11

[45] Li D C, Liu C W (2012) Extending attribute information for small data set classification. IEEE Trans Knowl Data Eng 24(3):452-464

[46] Wang C, She Z, Cao L B (2013) Coupled attribute analysis on numerical data. In: International Joint Conference on Artificial Intelligence (IJCAI 2013), OPUS, pp 1736-1742

[47] Chen Z, Liu K Y, Yang X B, Fujita H (2022) Random sampling accelerator for attribute reduction. Int J Approx Reason 140:75-91

[48] Chen Q, Xu T H, Chen J J (2022) Attribute Reduction Based on Lift and Random Sampling. Symmetry 14(9):1828

[49] Chen H M, Li T R, Fan X, Luo C (2019) Feature selection for imbalanced data based on neighborhood rough sets. Inf Sci 483:1-20

[50] Chen Y, Wang P X, Yang X B, Mi J S, Liu D (2021) Granular ball guided selector for attribute reduction. Knowl Based Syst 229:107326

[51] Jia X Y, Rao Y, Shang L, Li T J (2020) Similarity-based attribute reduction in rough set theory: A clustering perspective. Int J Mach Learn Cybern 11(5):1047-1060

[52] Hu Q H, Zhang L, Chen D G, Pedrycz W, Yu D R (2010) Gaussian kernel based fuzzy rough sets: Model uncertainty measures and applications. Int J Approx Reason 51(4):453-471

[53] Hu Q H, Yu D R, Xie Z X (2008) Neighborhood classifiers. Expert Syst Appl 34(2):866-876

[54] Hu Q H, Pedrycz W, Yu D R, Lang J (2009) Selecting Discrete and Continuous Features Based on Neighborhood Decision Error Minimization. IEEE Trans Syst Man Cybern B 40(1):137-150

[55] Li W T, Zhou H X, Xu W H, Wang X Z, Pedrycz W (2022) Interval dominance-based feature selection for interval-valued ordered data. IEEE Trans Neural Netw Learn Syst 1-15. http://dx.doi.org/10.1109/TNNLS.2022.3184120

[56] Li W T, Zhai S C, Xu W H, Pedrycz W, Qian Y H, Ding W P, Zhan T (2022) Feature selection approach based on improved Fuzzy C-Means with principle of refined justifiable granularity. IEEE Trans Fuzzy Syst 1-15. http://dx.doi.org/10.1109/TFUZZ.2022.3217377.

[57] Rao X S, Yang X B, Yang X, Chen X J, Liu D, Qian Y H (2020) Quickly calculating reduct: an attribute relationship based approach. Knowl Based Syst 200:106014

[58] Liu K Y, Yang X B, Fujita H, Liu D, Yang X, Qian Y H (2019) An efficient selector for multi-granularity attribute reduction. Inf Sci 505:457-472

[59] Yao Y Y, Zhang Y, Wang J (2008) On reduct construction algorithms. Trans. Comput. Sci. II 5150:100-117

[60] Chapman-Rounds M, Bhatt U, Pazos E, Schulz M-A, Georgatzis K (2021) FIMAP: Feature Importance by Minimal Adversarial Perturbation. In: Association for the Advancement of Artificial Intelligence (AAAI 2021), pp 11433-11441

[61] Inkawhich N, Wen W, Li H, Chen Y R (2019) Feature Space Perturbations Yield More Transferable Adversarial Examples. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR 2019), IEEE, pp 7066-7074

[62] Aksakalli V, Malekipirbazari M (2016) Feature selection via binary simultaneous perturbation stochastic approximation. Pattern Recognit Lett 75:41-47