IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received May 2, 2020, accepted May 23, 2020, date of publication May 26, 2020, date of current version June 9, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.2997791

A Novel Hybrid Classification Method Based on the Opposition-Based Seagull Optimization Algorithm

HE JIANG, YE YANG, WEIYING PING, AND YAO DONG
School of Statistics, Jiangxi University of Finance and Economics, Nanchang 330013, China
Applied Statistics Research Center, Jiangxi University of Finance and Economics, Nanchang 330013, China

Corresponding author: Ye Yang (yangyejufe2018@126.com)

This work was supported in part by the National Natural Science Foundation of China under Grant 71901109, Grant 71861012, Grant 71761016, and Grant 18ATJ001, in part by the Natural Science Foundation of Jiangxi, China, under Grant 20181BAB211020, in part by the Jiangxi Double Thousand Plan, Scientific Research Fund of Jiangxi Provincial Education Department under Grant GJJ180287 and Grant GJJ190264, in part by the Research Projects for Postdoctoral Researchers of Jiangxi Province under Grant 2018KY08, and in part by the Humanities and Social Sciences Foundation of Jiangxi Province under Grant TJ19202.

ABSTRACT In practice, classification problems have appeared in many scientific fields, including finance, medicine and industry. It is critically important to develop an effective and accurate classification model. Although numerous useful classifiers have been proposed, they are unstable, sensitive to noise and slow in computation. To overcome these drawbacks, the combination of feature selection techniques with traditional machine learning models is of great help. In this paper, a novel feature selection method called the opposition-based seagull optimization algorithm (OSOA) is proposed and studied. The OSOA is constructed based on an SOA whose population is determined by the opposition-based learning (OBL) algorithm. To evaluate its overall classification performance, some measures, including classification accuracy, number of selected features, receiver operating characteristic curve (ROC), and computation time, are adopted. The empirical results indicate that the suggested method exhibits higher or similar accuracy and computational efficiency in comparison with genetic algorithm (GA)-, simulated annealing (SA)-, and Fisher score (FS)-based classification models. The experimental results show that the OSOA is a computationally efficient feature selection technique that has the ability to select relevant variables. Furthermore, it performs well with high-dimensional data whose number of variables exceeds the number of samples. Thus, the OSOA is an effective approach for the enhancement of classification performance.

INDEX TERMS Hybrid method, machine learning, OSOA, OBL, feature selection.

I. INTRODUCTION

With the development of computer and information techniques, large amounts of data are being generated from numerous sources, including economic activities, public administration and other scientific research fields [1]. To make sense of the data, machine learning techniques for the extraction of important patterns and trends from data and the prediction of data properties are employed. Machine learning techniques have been applied to a wide range of fields, including agriculture [2], finance [3], [4], and medicine [5]. Basically, the related techniques can be categorized as supervised or unsupervised learning methods. In supervised learning, the goal is to predict the value of predefined target variables based on independent variables, whereas in unsupervised learning, there are no predefined target variables, and the goal is to describe the relationship and patterns among a set of independent variables [6], [7]. Classification is one of the typical and fundamental tasks of supervised learning. Feature selection is effective for handling high-dimensional data to enhance the overall performance of classification, which has been proven in both theory and practice [8]–[10]. The main goal of classification is to assign the instances in the test datasets to a predefined category based on the information classifiers acquired from the training datasets.

The associate editor coordinating the review of this manuscript and approving it for publication was Zhanyu Ma.

A. LITERATURE REVIEW

Many classification algorithms have been developed thus far. For instance, Logistic regression (LR) is a simple and effective classifier. It has wide applications in fields that require interpreting the relationship between independent variables (features) and dependent variables (classes) or the roles independent variables play in models, such as business [4] and industry [11]. Support vector machines (SVMs) handle classification tasks by constructing a hyperplane in sample space or feature space mapped by a kernel function. The application of a kernel function makes SVM a powerful method [12]. In some specific problems, e.g., prediction of chemical activity [13] and credit risk evaluation [14], researchers have designed new kernel functions to improve the performance of SVM. Least squares support vector machine (LSSVM) is a least squares version of SVM whose constraints are a set of linear equations, while the classical SVMs use a quadratic programming problem. Thus, LSSVM is more computationally efficient and can thus be applied to large-scale problems [15]. An artificial neural network (ANN) is a system with numerous connected neurons that simulates biological neural networks. The topology of ANNs can be categorized into 3 parts: the input layer, hidden layer and output layer. The training procedure of ANNs is to adjust the connection weights between neurons. ANNs are valuable and attractive classification techniques because they are nonlinear, data-driven self-adaptive classifiers and universal functional approximators, which can handle noisy data and do not need many priori assumptions [16]. These distinguishing features have allowed ANNs to enjoy fruitful applications in many fields [17]. A multi-layer perceptron neural network (MLPNN) is constructed of original simple perceptions and trained by a back-propagation algorithm [18], which has received wide applications in time-series problems [19], [20]. The back propagation neural network (BPNN), a typical and classic ANN, can find highly complex and nonlinear solutions to classification problems, which makes BPNN a very popular algorithm in complex nonlinear systems. However, it has problems regarding local optima and poor convergence, especially when it has a large set of neurons [21]. A radial basis function neural network (RBFNN) is a type of feed-forward network based on computational intelligence with a simple structure and high efficiency. Moreover, RBFNN has the ability to perform nonlinear mapping and global optimal approximation [22]. Though classification approaches have achieved great success in various fields, they encounter a serious problem in high-dimensional data, which is known as the ''curse of dimensionality'' [23]. In high-dimensional data, a large number of features increase the size of the feature space, and many of them are irrelevant or redundant, which makes it difficult to recognize patterns for forecasting or classification. In addition, computational complexity is another challenge in processing high-dimensional data. Consequently, it is necessary to reduce the dimensionality of data. Feature selection identifies relevant features from an original feature set by removing these irrelevant and redundant features. It contributes to the reduction of training time, interpretability of the classification results and improvement of the classification performance, especially for high-dimensional cases. From the searching strategies perspective, feature selection methods can be classified into filter or wrapper approaches [24], [25]. Wrapper-type methods select a subset of features by a search algorithm binding with a given classifier. Many intelligent optimization algorithms have been adopted to build wrapper feature selection methods. A genetic algorithm (GA) is a stochastic, global optimization algorithm that can be used to perform feature selection naturally. It converges very slowly due to the unguided mutation operator [26]. Simulated annealing (SA) is another optimization algorithm used in feature selection tasks. However, SA cannot handle problems with large solution spaces well. Particle swarm optimization (PSO), a swarm intelligence optimization method, has the ability to retain and share good solutions with all particles. Moreover, PSO is easy to implement and computationally effective due to its algorithmic simplicity. However, PSO is not stable in high-dimensional search spaces and suffers from early convergence [27].

When features are evaluated by some criteria without classifiers, these approaches are called filter-type methods [28]. Fisher score, a filter method, computes a score for each attribute. The most discriminative features are those with higher scores. Then, a proper number of features can be picked according to their scores. The minimum redundancy maximum relevance (mRMR) filter method is based on mutual information and mainly contains two stages. First, the best individual features correlated to target variables are selected by the maximal relevance method. Then, the redundant features among the features obtained in the first step are removed by the minimal redundancy method [29]. A risk that mRMR suffers is that some uninformative features called irrelevant redundant features may be retained. In addition, mRMR is not suitable for high-dimensional data [30]. ReliefF, derived from the original Relief algorithm, evaluates the usefulness of features according to the feature's weight by searching the nearest neighbor from the same and different classes of randomly selected instances. ReliefF is capable of handling incomplete and noisy data but is still unable to delete redundant features [31].

One can also find a subset of features by minimizing the goodness-of-fit measurement score, such as AIC [32], BIC [33], and Mallow's Cp [34], of the model. However, these approaches are infeasible for a large number of features. To overcome this shortcoming, some regularization methods have been applied as feasible approaches to high-dimensional problems.

Ridge regression [35] with l_2 penalty and LASSO [36] with l_1 penalty are two typical regularization methods. Ridge regression is an effective technique for multicollinearity problems [37], and it can yield a coefficient contraction but never reaches zero. Namely, ridge regression is unable to complete feature selection tasks. In contrast, LASSO has the desirable

quality of shrinking some coefficients of uninformative features to zero.

That is, LASSO achieves the goal of feature selection by compressing some coefficients to zero using the l_1 penalty term. It is well accepted that LASSO has the ability to select the most relevant features from a broad set of candidate variables and enhance the predictive performance. In addition, LASSO behaves consistent statistically as the number of samples increases, and strict assumptions are not required [38]. Importantly, LASSO can be employed for the problem of multicollinearity, which is a very common phenomenon in high-dimensional problems, and it is an effective feature selection technique for high-dimensional data [39], [40]. It has been proven that hybrid models have the ability to overcome the drawbacks of using a single classifier [6].

B. CONTRIBUTION

In this research, we propose a novel hybrid classification method based on an OSOA. We borrow the strengths from both the SOA and OBL, which is embedded to determine the population of the SOA. Computationally, we have derived an efficient algorithm to obtain a global minimizer of the method and better classification performance. We have shown the advantages of the proposed method in different datasets, including high-dimensional datasets, via comparison with some state-of-the-art feature selection methods such as Fisher score, simulated annealing and genetic algorithm. In addition, the well-known LASSO method is also compared. To evaluate the proposed method, accuracy, ROC, AUC and computational efficient are adopted, and comprehensive comparisons are made between the proposed method and other popular methods. The rest of this paper is organized as follows. Section I gives the introduction, and the theoretical background is presented in Section II. Section III exhibits the proposed OSOA method, and Section IV shows the experimental results. Finally, a conclusion is drawn in Section V.

II. THEORETICAL BACKGROUND

A. SEAGULL OPTIMIZATION ALGORITHM

The seagull optimization algorithm (SOA) [41] is a recently proposed metaheuristic optimization technique inspired by the natural behaviors of seagulls. Seagulls, scientific named Laridae, are intelligent birds. They can attract fish and earthworms by using breadcrumbs or making a rain-like sound with their feet. Generally, seagulls live in colonies. To find abundant food, they often migrate from one place to another. After arriving at a new place, seagulls attack their prey. he most important thing about seagulls is their migrating and attacking behaviors. Thus, the SOA focuses on these two natural behaviors, and the mathematical models are presented below.

First, seagulls perform migration behavior. During migration, the members of a seagull swarm should avoid colliding with each other. To achieve this purpose, an additional variable A is employed.

$$C = A \times P(t), \tag{1}$$

where $P(t)$ represents the current position of seagulls in the t-th iteration and A depicts the movement behavior of seagulls.

$$A = a - (t \times (a/MAX_{iteration})), \tag{2}$$

where a is a constant and responsible for controlling the frequency of employing variable A, which linearly decreases from a to 0. To find the richest food resources, seagulls move toward the best search agent.

$$M = B \times (P_{bs}(t) - P(t)), \tag{3}$$

where M represents seagull position toward the best search agent (seagull). The coefficient B is a random value responsible for making a trade-off between exploitation and exploration, and is defined as:

$$B = 2 \times A^2 \times rd, \tag{4}$$

where rd is a random number that lies in the interval $[0, 1]$. As seagulls move toward the fittest search agent, they might remain close to each other. Thus, seagulls can update their position according to the following rule:

$$D = |C + M|, \tag{5}$$

where D represents the distance between seagulls and the best search agent.

Second, seagulls attack prey in a spiral shape after arriving at a new place. Their attacking behavior can be formulated as:

$$P(t) = (D \times x \times y \times z) + P_{bs}(t), \tag{6}$$

where $P(t)$ retains the best solution and x, y, z depict the traits of spiral motion.

$$x = r \times \cos(k) \tag{7}$$

$$y = r \times \sin(k) \tag{8}$$

$$z = r \times k \tag{9}$$

$$r = u \times e^{kv}, \tag{10}$$

where u and v are constants, e is the base of the natural logarithm, and k is a random number between 0 and 2π.

1) OPPOSITION-BASED LEARNING

Opposition-based learning (OBL) [42] was first proposed in 2005. Since then, OBL has been widely applied to improve the performance of metaheuristic algorithms, reinforcement learning and other machine intelligence techniques. In this work, we focus on employing OBL to help a metaheuristic optimization algorithm search for the global optimum. In general, a metaheuristic starts with a randomly generated population and iteratively updates the current solutions. By applying OBL, the opposite solution of the current solution is produced. Then, OBL compares the fitness of the current solution with the corresponding opposite solution and

keeps the better one. Therefore, OBL has the potential to accelerate the convergence of the metaheuristic algorithm and obtain optima more easily. Here, we introduce some key concepts related to our work.

Assuming that x is a real number that lies in the interval $[u, l]$, the opposite number of x is defined as:

$$\bar{x} = u + l - x, \tag{11}$$

where u and l are the upper and lower bounds of the problem, respectively. For higher-dimensional problems, let $x = (x_1, x_2, \ldots, x_d) \in R^d$ be a d-dimension vector, where $x_i \in [u_i, l_i]$, $i = 1, 2, \ldots, d$. The opposite vector $\bar{x}$ can be defined as

$$\bar{x} = (\bar{x}_1, \bar{x}_2, \ldots, \bar{x}_d), \tag{12}$$

where $\bar{x}_i = u_i + l_i - x_i$, $i = 1, 2, \ldots, d$. Furthermore, if x is a binary vector, $x = (x_1, x_2, \ldots, x_d) \in \{0, 1\}^d$, then $u_i = 1$, $l_i = 0$. Thus, in binary space, the opposite vector of x is defined as:

$$\bar{x} = (1 - x_1, 1 - x_2, \ldots, 1 - x_d). \tag{13}$$

The details of how to integrate OBL into the metaheuristic (SOA, in this work) will be discussed in the next section.

2) FISHER SCORE

Fisher score (FS) is a type of filter method, based on the Fisher criterion, which has the ability to select the most relevant features. FS indicates that these features with higher Fisher scores should be selected. Given a dataset $\{(x_i, y_i)\}_{i=1}^n$, $x_i \in R^d$ denotes that there are d features in the dataset, and $y_i \in R^k$ denotes the dataset has k classes. Then, the Fisher score of the i-th feature, f_i, is calculated by the following expression:

$$f_{FS}(f_i) = \frac{\sum_{j=1}^{k} n_j (\mu_{ij} - \mu_i)^2}{\sum_{j=1}^{k} n_j \sigma(i, j)^2}, \tag{14}$$

where n_j indicates the number of class j of the sample, the mean value of f_i is denoted by μ_i, and μ_{ij} and $\sigma(i, j)^2$ denote the mean value and variance of f_i corresponding to the j-th class, respectively. In a nutshell, the importance of every feature is measured by FS, and then top features with high scores are selected after ranking.

3) LASSO

Least absolute shrinkage and selection operator (LASSO) was first introduced by Tibshirani [36], which is a constrained version of ordinary least squares [43]. Given a dataset $\{(x_i, y_i)\}_{i=1}^n$, $x_i \in R^d$, $y_i \in R$ the definition of LASSO is given by

$$\min_{\alpha} \frac{1}{2} \|y - x\alpha\|_2^2 \quad \text{s.t. } \|\alpha\|_1 \leq t, \tag{15}$$

where $x = [x_1, x_2 \ldots x_n]^T$ is an $n \times d$ dimensional feature matrix, $y = [y_1, y_2 \ldots y_n]$ is the response vector, $\alpha = [\alpha_1, \alpha_2 \ldots \alpha_d]$ is a regression coefficients vector and $t \geq 0$ is the constraint term. $\|\cdot\|_1$ is l_1-norm and $\|\cdot\|_2$ is l_2-norm. Write the above optimization problem in Lagrangian form

$$\min_{\alpha} \frac{1}{2} \|y - x\alpha\|_2^2 + \lambda \|\alpha\|_1. \tag{16}$$

Here, λ is a tuning parameter that controls the strength of shrinkage. By applying the l_1-norm, coefficients can be shrunk to exactly zero if λ is large enough, and more coefficients will be shrunk to zero as λ increases. Thus, LASSO can be seen as a continuous and stable feature selection method. Moreover, it produces a sparse solution and makes the model easier to interpret by adjusting the parameter λ.

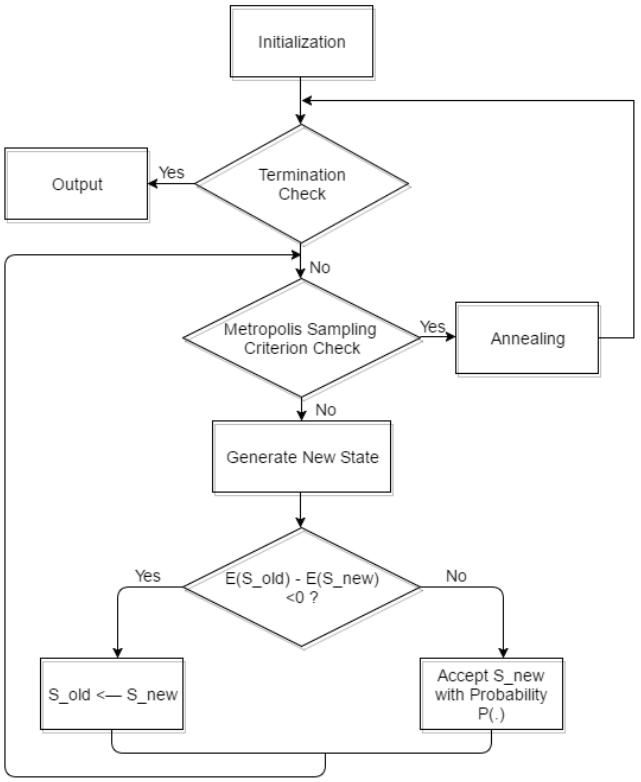


FIGURE 1. Simulated annealing flowchart.

4) SIMULATED ANNEALING

Simulated annealing (SA) is a global optimization technique that simulates the annealing phenomenon of metallurgy. Usually, SA starts with a randomly generated solution at a fairly high temperature. To find the global optimal solution, the initial temperature should be as large as possible. Next, the initial solution is updated in a certain way as the temperature decreases until the termination condition is reached. The most used method of temperature decrease is $T_{k+1} = \lambda T_k$, where T_k is the current temperature, T_{k+1} is the updated temperature, and λ is a constant less than 1 but close to 1. Theoretically, the temperature should decrease to 0 or SA will not converge, which is considerably difficult to realize in practice. Some alternative methods, e.g., setting a minimal temperature value or setting a maximal number of iterations

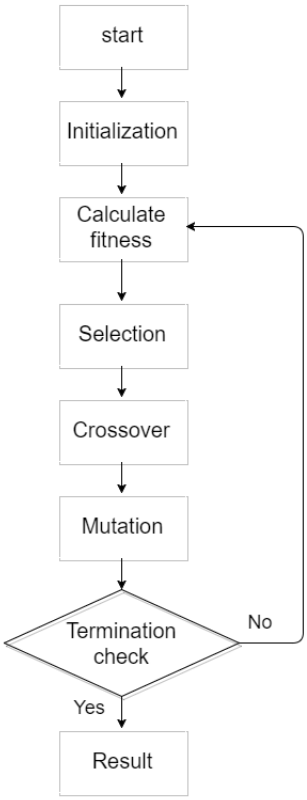


FIGURE 2. GA flowchart.

directly, are often adopted. The flowchart of SA (minimizing problem case) is presented in Fig.1.

At a certain temperature, a new solution, say S_{new}, is generated by a neighbor function, whose specific form depends on the problem domain. Then, the new solution is compared with the current solution, say S_{old}, by an evaluation function. If the new solution is superior to the current solution, then the transposition is: $S_{old} \leftarrow S_{new}$. If not, SA accepts the inferior one with a certain probability $p(\cdot)$ generated by an acceptance function. This is the key step that makes SA jump out of local optima. The probability is associated with the temperature. It is higher at the beginning and tends to lower as the temperature decreases. The Metropolis algorithm is frequently adopted as the acceptance function. These procedures are repeated until the optimal solution is found or some stop criteria are met.

5) GENETIC ALGORITHM

A genetic algorithm (GA), a type of evolutionary algorithm, is inspired by the process of natural selection. The flowchart of a standard GA is presented in Fig.2. It produces a set of solutions, which are completely independent from each other, at the same time. Every solution is encoded as bits, numbers *et al.* in a sequence. The sequence is referred to as a chromosome or individual. Populations consist of chromosomes (individuals), while genes are elements of an encoded solution, which compose chromosomes. In selection, the chromosomes with higher fitness value are more likely to be selected and used for recombination. The Roulette Wheel and Tournament are two commonly used selection operators. The crossover, the pivotal process in GA, refers to two chromosomes exchanging some of their genes with each other according to the crossover probability. The result of crossover is that two new chromosomes are generated. The mutation indicates that the genes in the chromosome are altered with a certain probability. By applying the three genetic operators, the convergence of GA is guaranteed [44]. Moreover, GA has the ability to process large search spaces [45], [46].

B. CLASSIFICATION METHODS

1) SUPPORT VECTOR MACHINE

Support Vector Machine (SVM) was initially introduced for linearly separable classification problems [47]. However, there are numerous datasets in our real life that are nonlinearly separable [48]. To deal with these cases, kernel tricks are adopted. Considering a dataset $\{(x_i, y_i)\}_{i=1}^{m}$, $y_i \in \{-1, +1\}$ SVM has the following form

$$\begin{aligned} &\min_{w,b,\xi} \frac{1}{2}\|w\|^2 + C\sum_{i=1}^{m}\xi_i \\ &s.t.\ y_i(w^T k(x_i) + b) \geq 1 - \xi_i; \\ &\quad \xi_i \geq 0, \quad i = 1, 2, \ldots, m, \end{aligned} \tag{17}$$

where w, b is the weigh vector, b is the bias. ξ_i is slack variables, $k(\cdot)$ is the kernel function that can map the input space into feature space (higher dimension space), and C is a real constant determined by users that balances the margin maximization and training error. According to dual theory, the Lagrangian form of SVM is

$$\max_{\alpha}\left[\sum_{i=1}^{m}\alpha_i - \frac{1}{2}\sum_{i=1}^{m}\sum_{j=1}^{m}\alpha_i\alpha_j y_i y_j k\left(x_i, x_j\right)\right] \tag{18}$$

$$\text{subject to}\sum_{i=1}^{m}\alpha_i y_i = 0 \quad 0 \leq \alpha_i \leq C, \quad i = 1, 2 \ldots m \tag{19}$$

where α_i is the lagrangian multipliers. There are lots of algorithms that can be applied to solve the above optimization problem [49]–[51]. After solving this optimization problem, the decision function is given by

$$y(x) = sign[\sum_{i=1}^{m}\alpha_i y_i k(x, x_i) + b]. \tag{20}$$

Commonly used kernel functions are

- Linear kernel function $k(x_i, x_j) = x_i \cdot x_j$
- Polynomial kernel function $k(x_i, x_j) = (x_i \cdot x_j + 1)^d$
- Radial basis function $k(x_i, x_j) = \exp\left(-\frac{||x_i - x_j||^2}{2\cdot\sigma^2}\right)$, where σ are constant.

2) LEAST-SQUARE SUPPORT VECTOR MACHINE

Least-square support vector machine (LSSVM) is a least-square version of SVM, which is used for classification and

regression analysis. LSSVM uses equality constraints rather than the inequality constraints used in SVM, and it converts the quadratic programming problems into linear equation problems by utilizing a sum square errors cost function instead of the nonnegative errors cost function in the SVM model. Consequently, LSSVM consumes less computational resources [52]–[54]. Given a dataset $\{(x_i, y_i)\}_{i=1}^{n}$, $x_i \in R^d$ indicates that there are d attributes, and $y_i \in \{-1, +1\}$ represents that the output is binary. Assuming that the dataset is linearly inseparable in its attribute space, the input space $x_i \in R^d$ will be mapped into a higher-dimensional space (feature space) by a nonlinear mapping function $\phi(\cdot)$, which is illustrated in Fig.3. Therefore, the optimal decision function can be constructed in the feature space

$$y = \omega^T \cdot \phi(x) + b, \tag{21}$$

where ω, the weight vector, and b, the bias, are two parameters to be estimated. $\phi(x)$ is the nonlinear mapping function. To solve the above regression equation, a constrained optimization problem is constructed according to the structural risk minimization principle [55]:

$$\min_{\omega,\zeta} \left[\frac{1}{2}\|\omega\|^2 + \frac{1}{2} C \sum_{i=1}^{n} \zeta_i^2 \right] \tag{22}$$

$$\text{subject to } y_i = \omega^T \cdot \phi(x_i) + b, \quad i = 1, 2, \ldots, n \tag{23}$$

where C is the penalty factor that controls the trade-off between the complexity and the approximation precision of LSSVM. ζ_i is the error between the prediction value of sample i and its true output value. Due to the difficulty of solving the above optimization problem directly, the Lagrange multiplier method is applied here. The Lagrange multiplier theorem states that, at any local maxima (or minima) of the function evaluated under the equality constraints, if the constraint qualification applies, then the gradient of the function can be expressed as a linear combination of the gradients of the constraints (at that point), with the Lagrange multipliers acting as coefficients. Thus, its corresponding Lagrangian function is built as follows:

$$\begin{aligned} L(\omega, b, \zeta) &= \frac{1}{2} C \sum_{i=1}^{n} \zeta_i^2 + \frac{1}{2}\|\omega\|^2 \\ &\quad - \sum_{i=1}^{n} \left(\alpha_i \left(\omega^T \cdot \phi(x_i) + b + \zeta_i - y_i \right) \right), \end{aligned} \tag{24}$$

where α_i is the Lagrangian multipliers. It is worth noting that the Lagrangian multipliers in LSSVM are positive or negative, whereas they must be positive in SVM [56]. Allowing inequality constraints, the KKT (Karush-Kuhn-Tucker) approach to nonlinear programming generalizes the method of Lagrange multipliers, which allows only equality constraints. Similar to the Lagrange approach, the constrained maximization (minimization) problem is rewritten as

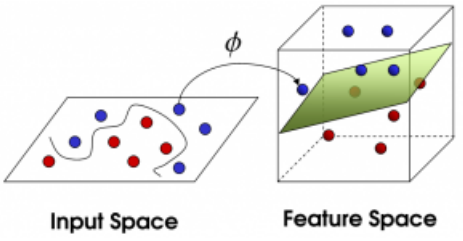


FIGURE 3. Nonlinear mapping.

a Lagrange function whose optimal point is a saddle point. According to KKT conditions, we can get

$$\begin{cases} \dfrac{\partial L}{\partial \omega} = 0 \Rightarrow \omega = \displaystyle\sum_{i=1}^{n} \alpha_i \phi(x_i) \\ \dfrac{\partial L}{\partial b} = 0 \Rightarrow \displaystyle\sum_{i=1}^{n} \alpha_i = 0 \\ \dfrac{\partial L}{\partial \zeta_i} = 0 \Rightarrow \zeta_i = C\alpha_i \\ \dfrac{\partial L}{\partial \alpha_i} = 0 \Rightarrow \omega^T \cdot \phi(x_i) + b + \zeta_i - y_i = 0 \end{cases} \tag{25}$$

Next, applying Mercer's theorem:

$$\phi^T(x_i)\,\phi(x_j) = K(x_i, x_j), \quad i, j = 1, 2, \ldots, n \tag{26}$$

where $K(x_i, x_j)$ is the kernel function. Eliminating ω and ζ_i, a linear equation set is obtained:

$$\begin{pmatrix} 0 & E_n^T \\ E_n & \Phi \end{pmatrix} \begin{pmatrix} b \\ \alpha \end{pmatrix} = \begin{pmatrix} 0 \\ y \end{pmatrix} \tag{27}$$

where E is an n-dimensional unit vector, $\alpha = (\alpha_1, \alpha_2, \ldots, \alpha_n)$ is the parameter of LSSVM, and I_n is an $n \times n$ identity matrix. $\Phi = \Omega + C^{-1} I_n$, where Ω is an $n \times n$ kernel matrix, whose elements are defined as

$$\Omega_{i,j} = \phi^T(x_i)\,\phi(x_j) = K(x_i, x_j), \quad i, j = 1, 2, \ldots, n \tag{28}$$

After solving the above linear equation set, parameter b and α are given by [57]

$$\begin{cases} b = \dfrac{E_n^T \Phi^{-1} y}{E_n^T \Phi^{-1} I_n} \\ \alpha = \Phi^{-1}(y - E_n b). \end{cases} \tag{29}$$

Then, the final model of LSSVM is

$$y(x) = \omega^T \cdot \phi(x) + b = \sum_{i=1}^{n} \alpha_i K(x, x_i) + b. \tag{30}$$

III. PROPOSED METHOD

In this section, the proposed hybrid classification methods are explained. In our work, there are two main stages. First, an opposition-based seagull optimization algorithm (OSOA) is employed to conduct feature selection on the original dataset. Second, classification is performed on the reduced data obtained from the first stage. Details of the proposed hybrid methods are presented below.

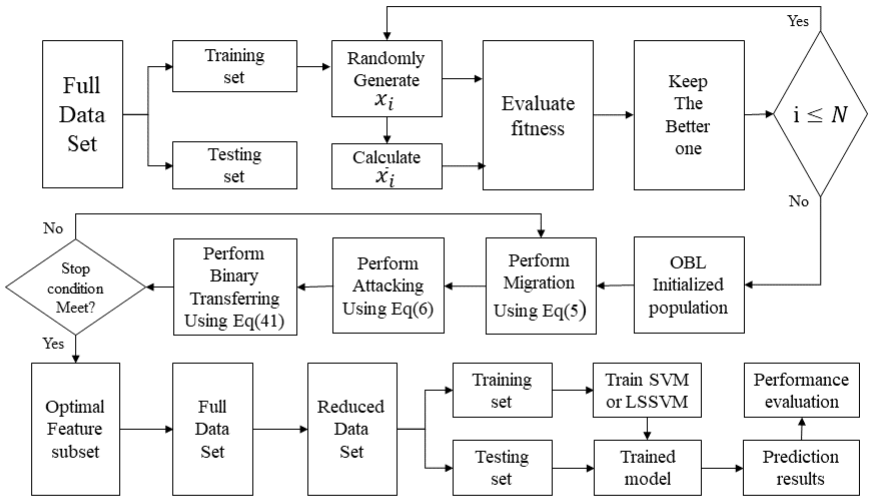


FIGURE 4. The flowchart of the proposed classification method.

A. FEATURE SELECTION STAGE

The OSOA is constructed as a wrapper feature selection model. To represent whether a feature is selected or not, binary values are applied. Specifically, every individual among the population of the SOA is coded as a binary vector. In these binary vectors, "1" indicates that the corresponding feature is selected and "0" indicates that the feature is neglected. A typical binary vector used in the wrapper model could be: $x = (x_1, x_2, \ldots, x_d)$, where $x_i \in \{0, 1\}$, $i = 1, 2, \ldots, d$, and d is the dimension of the original dataset. To evaluate the performance of the selected feature subset, a fitness function is needed. In our work, the fitness function is defined as:

$$F = \beta E(S) + (1 - \beta)\frac{|S|}{|D|}, \tag{31}$$

where $E(S)$ is the error of a classifier on the feature subset S. $|D|$ is the dimension of the original dataset, $|S|$ is the size of the feature subset S, and β is a constant used to balance the feature subset size and the classifier's accuracy. In this work, KNN is employed as the evaluator in the feature selection stage.

1) INITIAL POPULATION

In this step, OBL is applied to initialize the population of the SOA. Generally, the SOA starts with a randomly generated population. By applying OBL, the diversity of the SOA's population is enhanced. The diversified population will improve the convergence and search abilities of the SOA. To initialize the population, the OSOA begins with a predefined population size N. Then, the OSOA randomly generates an individual x and the corresponding opposite individual $\bar{x}$. Next, both x and $\bar{x}$ are evaluated by the fitness function, and the better one is kept. This process is repeated until the predefined population size is satisfied. For clarity, the initialization procedure is presented in Algorithm1.

Algorithm 1 OBL Initialization

```
Input the training dataset, then initialize the population size N and set an empty container P
for i to N do
    randomly generate an individual x_i
    calculate the opposite individual x̄_i using Eq(13)
    evaluate the fitness of x_i and x̄_i using Eq(31)
    if x_i is better than x̄_i then
        P ← x_i
    else P ← x̄_i
    end if
end for
return The initialized population P
```

2) UPDATE POSITION

After initializing the population, the OSOA is applied to update the seagulls' positions. As aforementioned, every seagull is coded as a binary vector, whereas the original SOA was proposed for processing continuous problems. Thus, a binary version of the SOA is needed. To achieve this goal, a transfer function is applied:

$$T_v(x) = \left|erf\left(\frac{\sqrt{\pi}}{2}x\right)\right| = \left|\frac{\sqrt{2}}{\pi}\int_0^{(\sqrt{\pi}/2)x} e^{-t^2}dt\right| \tag{32}$$

To obtain binary values, every seagull is transferred by the above function according to the following formula:

$$P^d(t+1) = \begin{cases} C\left(P^d\right) & \text{if } rd < T_v\left(P^d(t)\right) \\ P^d & \text{if } rd \geq T_v\left(P^d(t)\right), \end{cases} \tag{33}$$

where $P^d(t)$ is the d-th dimension of $P(t)$ obtained from Eq(6), $P(t+1)$ is the updated position, P^d is the value if d-th dimension of $P(t)$, $C\left(P^d\right)$ is the complement of P^d, and rd is a random number between 0 and 1.

After updating positions, every seagull is evaluated by the fitness function. These updating steps are repeated until the maximum number of iterations is reached. Then, the OSOA returns the fittest seagull (best solution). The fittest seagull represents the final selected feature subset. The procedures of the feature selection stage are presented in Algorithm.

Algorithm 2 Opposition-Based Seagull Optimization Algorithm for Feature Selection

Input the training dataset and initialize the parameters of the SOA
Initialize the population $P_i = \{p_1, \ldots, p_{j,\ldots,p_d}\}, i = 1, 2, \ldots, n$ by applying Algorithm1
while $t <$ max iteration **do**
  for i to n **do**
    evaluate fitness of P_i using Eq(31)
    set P_{bs} as the fittest seagull
    perform migration using Eq(5)
    perform attacking using Eq(6)
    perform change on each element of P_i using Eq(33)
  end for
  evaluate fitness of each seagull
  update the fittest seagull
  $t = t + 1$
end while
return P_{bs}

B. CLASSIFICATION STAGE

In this stage, SVM and LSSVM are applied to perform the classification task. In the feature selection stage, KNN is employed to evaluate the quality of the selected feature subset. The main reasons behind choosing different classifiers in the feature selection stage and classification stage are twofold: first, KNN is a computationally efficient model. Usually, wrapper feature selection models area argued that they are expensive at computation. By applying KNN, the OSOA can select the optimal feature subset faster. Second, KNN is a simple model such that the OSOA can avoid overfitting to some extent. The flowchart of the proposed hybrid classification method is given in Figure 4.

IV. EXPERIMENTS

A. EXPERIMENTS DESCRIPTION

To validate the superior performance of the hybrid methods, some experiments were performed. In particular, the proposed OSOA feature selection method were compared with four other state-of-the-art feature selection methods, including GA, SA, FS, and Lasso. The selected features will be tested on two classification models, LSSVM and SVM. Then, the hybrid models are established by combining these feature selection models individually with classification models. Firstly, feature selection methods are implemented in these 7 datasets. Second, classification models are applied to each dataset with features selected in the first step. All of the experiments are implemented in R3.6.0. For feature selection methods, FS and Lasso can be implemented using the PredPsych package [58] and glmnet package [59]. GA and SA are obtained in the caret package [60]. For classification models, SVM and LSSVM are available in the kernlab package [61]. There are 7 datasets applied in the experiments; the first five datasets, Hill-Valley, Ionosphere, Heart, Twonorm, and Ringnorm, are taken from the UCI repository [62], and Colon and Prostate are taken from an R package datamicroarray [63], which are high-dimensional datasets. Table.1 shows the details of these datasets. The second column indicates the names of these datasets, and the third column indicates the number of features of each dataset. The training and test samples were divided randomly, which are presented in the fourth and fifth columns, respectively. All 7 datasets are binary classes. The next two columns indicate the labels and the number of instances associated with each label. The last column shows the reference.

B. PARAMETER SETTING

For GA, the population size is 20, and elite is 1 for each generation. The crossover and mutation probability are 0.8 and 0.1, respectively, which are default values. For SA, all of the parameters are defaults. Both GA and SA run 100 iterations. For FS, the threshold was set empirically. For Lasso, all of the parameters are set as defaults.

For classification models including SVM and LSSVM, we mainly tried different kernel functions and parameters and selected the optimum. There are no parameters in Spline and Linear kernel functions. For the other six kernels, all of the scale parameters (sigma in the ANOVA RBF kernel, Polynomial kernel, Bessel kernel, Radial Basis kernel and Laplacian kernel; scale in the Polynomial kernel and Hyperbolic tangent kernel) belong to [0.0001, 1000], and we start from 0.0001 and times 10 for each experiment. For parameter degree (Polynomial kernel, Bessel kernel and ANOVA RBF kernel),it can only be a positive constant. We tuned the parameter from 1 to 20 because the models (SVM, LSSVM) are very sensitive to the parameter. For parameter offset (Polynomial kernel, Hyperbolic tangent kernel), we only tried 1 and 10 since the parameter has little effect on the results.

C. RESULTS

The number of features selected and processing time are presented in Table.1. The second column shows the original features. It easy to see that Colon and Prostate are high-dimensional data, whose numbers of attributes are 2,000 and 12,600, respectively. The number of features selected by GA, SA, FS, and Lasso and their computation time are also presented in this table. It is observed that the proposed OSOA-LSSVM and OSOA-SVM delivered better performance than the alternatives. For instance, for the Twonorm dataset, OSOA-SVM achieved a perfect classification outcome with 100% accuracy, and OSOA-LSSVM achieved an accuracy of 99.32%. For the Hill-Valley dataset, both OSOA-SVM and OSOA-LSSVM achieved 98% accuracy,

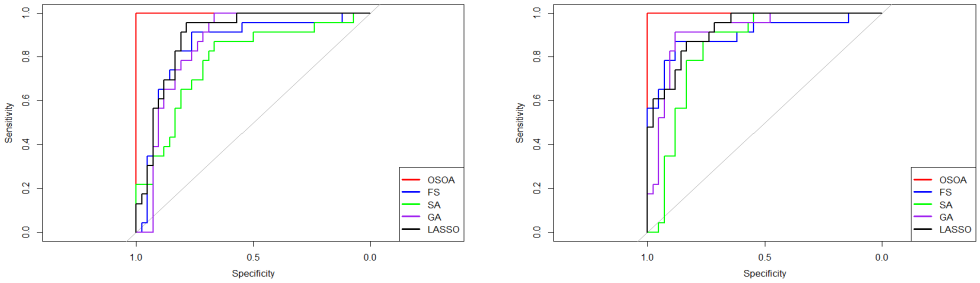
TABLE 1. Description of datasets.

Sources	Datasets	Features	Training instances	Test instances	Labels	Inst./Label	Ref
UCI	Hill-Valley	100	456	150	0/1	305/301	[62]
	Ionosphere	34	267	83	b/g	126/224	
	Heart	13	205	65	1/2	150/120	
	Twonorm	20	5463	1937	0/1	3703/3697	
	Ringnorm	20	5463	1937	0/1	3664/3736	
Microarray	Colon	2000	46	16	Normal/tumor	22/40	[63]
	Prostate	12600	74	28	Tumor/not	52/50	

TABLE 2. Classification accuracy and parameter settings of the hybrid classification methods.

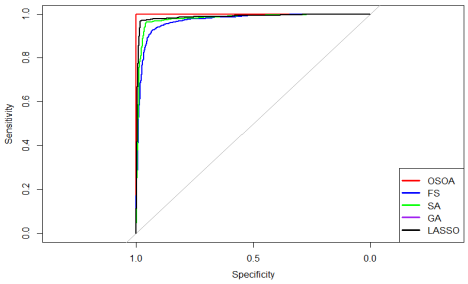
Datasets	Feature selection		Selected	Processing	Classifier	
	methods	parameter	features	time (in seconds)	LSSVM	SVM
Heart	OSOA	k=1 Population Size = 20 iteration=10 beta=2	10	0.66	**1**	**0.9385**
	FS	threshold=0.3	6	0.08	0.8154	0.8308
	SA	iteration=10	8	254.53	0.7385	0.6769
	GA	iteration=10 crossover=0.8 mutation=0.1	9	1.32 hours	0.7077	0.7846
	Lasso	penalty=1	11	0.29	0.6462	0.7692
Twonorm	OSOA	k=1 Population Size = 20 iteration=10 beta=2	15	12.07	**0.9923**	**1**
	FS	threshold=0.4	11	0.22	0.6469	0.9288
	SA	iteration=10	14	3395.08	0.9375	0.9592
	GA	iteration=10 crossover=0.8 mutation=0.1	20	23.84 hours	0.5958	0.9597
	Lasso	penalty=1	20	3.34	0.5958	0.9597
Hill-Valley	OSOA	k=1 Population Size = 20 iteration=10 beta=2	62	1.31	**0.98**	**0.98**
	FS	threshold=0.0008	28	0.33	0.56	0.4467
	SA	iteration=10	43	612.67	0.6333	0.4733
	GA	iteration=10 crossover=0.8 mutation=0.1	37	3.39 hours	0.6333	0.4867
	Lasso	penalty=1	28	12.76	0.5933	0.4533
Ionosphere	OSOA	k=1 Population Size = 20 iteration=10 beta=2	22	0.92	0.9639	**0.988**
	FS	threshold=0.1	9	0.09	0.8675	0.9398
	SA	iteration=10	11	305.69	0.7831	0.9398
	GA	iteration=10 crossover=0.8 mutation=0.1	20	1.62 hours	**0.9759**	0.9759
	Lasso	penalty=1	22	0.89	0.7711	0.9398
Ringnorm	OSOA	k=1 Population Size = 20 iteration=10 beta=2	14	12.48	**0.8544**	**0.9494**
	FS	threshold=0.045	8	2.66	0.8317	0.8921
	SA	iteration=100	16	1643.85	0.7976	0.9458
	GA	iteration=100 crossover=0.8 mutation=0.1	20	14.68 hours	0.7713	0.9076
	Lasso	penalty=1	20	2.26	0.7713	0.9076
Colon	OSOA	k=1 Population Size = 20 iteration=10 beta=2	1500	3.72	**0.875**	**0.9375**
	FS	threshold=0.3	130	0.87	0.6875	0.6875
	SA	iteration=10	640	222.25	0.6875	0.6875
	GA	iteration=10 crossover=0.8 mutation=0.1	1158	1.29 hours	0.6875	0.6875
	Lasso	penalty=1	16	1.04	0.75	0.75
Prostate	OSOA	k=1 Population Size = 20 iteration=10 beta=2	9450	34.29	**0.9286**	**0.9643**
	FS	threshold=0.2	265	8	0.5	0.5
	SA	iteration=10	4861	1560.76	0.5	0.5
	GA	iteration=10 crossover=0.8 mutation=0.1	6403	11.32 hours	0.5	0.5
	Lasso	penalty=1	22	7.27	0.5	0.8571

which is far higher than that of other methods. Figure 4 and Figure 5 show the ROC plot of all of the hybrid classification methods applied in all of the datasets. It is easy to see that OSOA-LSSVM and OSOA-SVM have larger areas, which indicates the advantage of the OSOA in terms of feature selection. From the aspect of number of selected features, the OSOA is comparable to Lasso in the Heart and Ionosphere datasets. The OSOA selected fewer features than

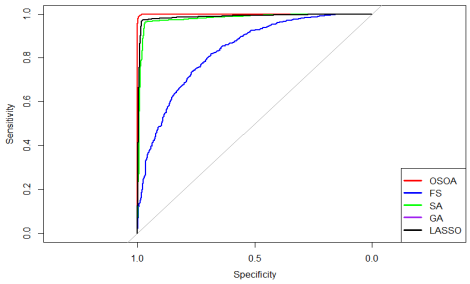


(a) Heart, SVM

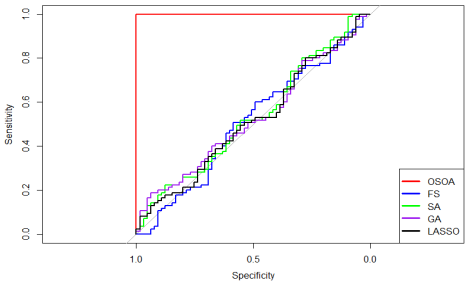
(b) Heart, LSSVM



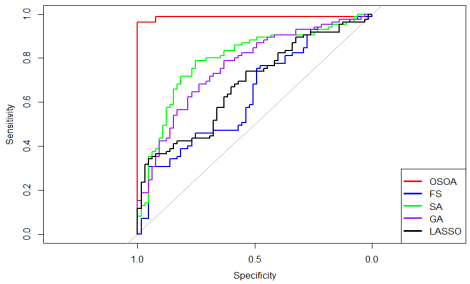
(c) Twonorm, SVM



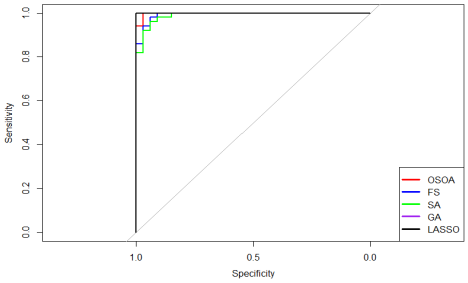
(d) Twonorm, LSSVM



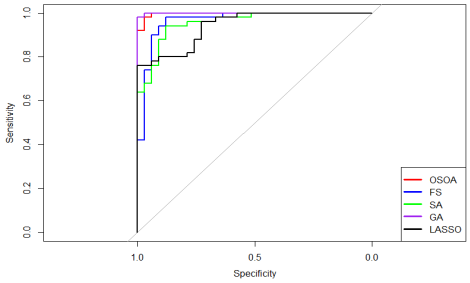
(e) Hill-valley, SVM



(f) Hill-valley, LSSVM



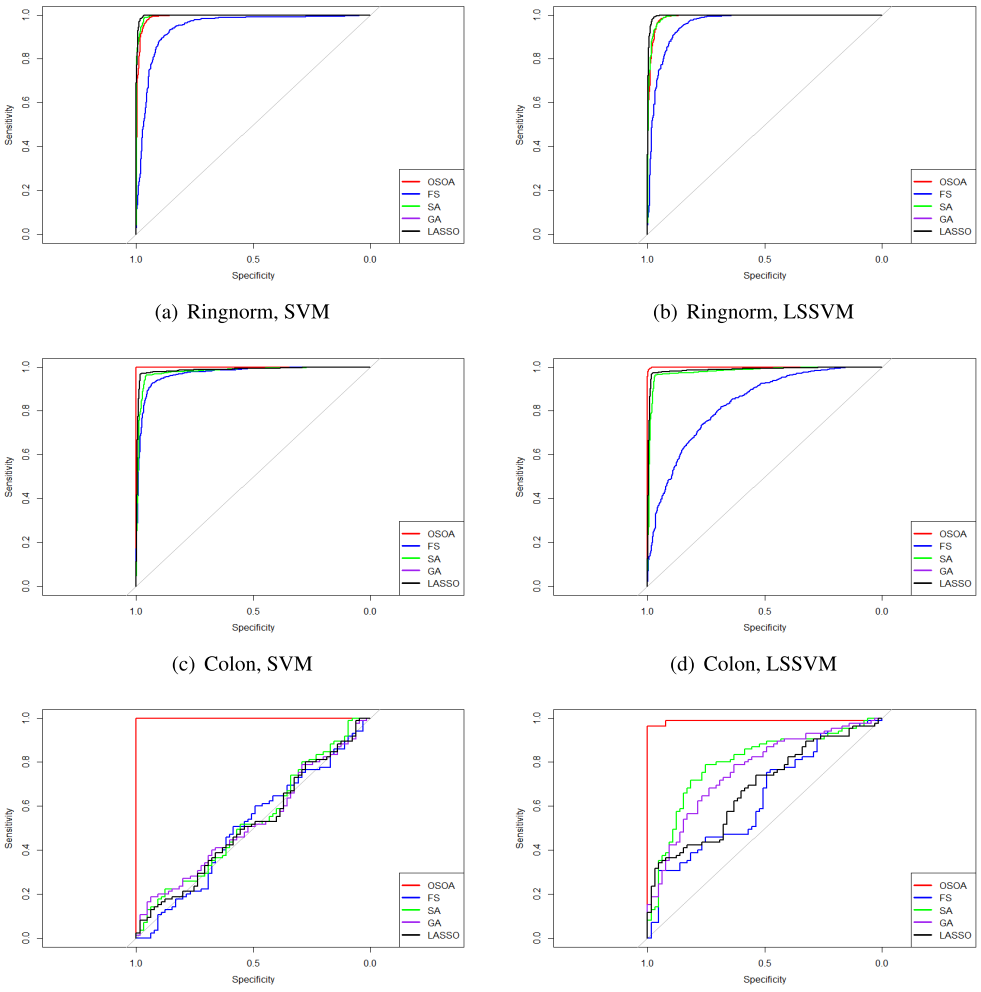
(g) Ionosphere, SVM



(h) Ionosphere, LSSVM

FIGURE 5. **ROC curves of the hybrid methods with the Heart, Twonorm, Hill-Valley, and Ionosphere datasets.**

Lasso did in the Twonorm and Ringnorm datasets. In the colon and prostate datasets, the OSOA selected more features. However, it is believed that the selected features are important since the classification accuracy is boosted to a large extent. Comparing with Lasso-SVM, which achieved 75% accuracy using 16 features, OSOA-SVM obtained 93.75% accuracy using 1,500 features. Furthermore, we find the processing times of the OSOA, FS, SA and Lasso to be comparable, while SA and GA take more computational time. Notably, the computational cost of GA is extremely expensive (14.68 hours) in the Ringnorm dataset. Consequently, OSOA-based hybrid classification methods are the best models in terms of both classification accuracy and computational efficiency.



(a) Ringnorm, SVM (b) Ringnorm, LSSVM

(c) Colon, SVM (d) Colon, LSSVM

(e) Prostate, SVM (f) Prostate, LSSVM

FIGURE 6. ROC curves of the hybrid methods with the Ringnorm, Colon and Prostate datasets.

V. CONCLUSION

In this study, a novel hybrid classification approach is suggested by combining feature selection and machine learning methods. Specifically, the proposed approach is based on an OSOA, which performs feature selection. The OSOA is an effective and computationally efficient feature selection technique. Moreover, the OSOA has the ability to process high-dimensional data as well. In the proposed model, there are two phases: (1) feature selection is done by the OSOA, and (2) the data with selected features are classified. The developed method was tested with seven datasets. Among these datasets, Colon and Prostate are high-dimensional data. Comparisons were made between the proposed method and other popular methods. The experimental results indicate that the overall performance of the proposed method is superior to that of other well-known feature selection approaches.

REFERENCES

[1] C. L. Philip Chen and C.-Y. Zhang, "Data-intensive applications, challenges, techniques and technologies: A survey on big data," *Inf. Sci.*, vol. 275, pp. 314–347, Aug. 2014. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0020025514000346

[2] M. A. Ebrahimi, M. H. Khoshtaghaza, S. Minaei, and B. Jamshidi, "Vision-based pest detection based on SVM classification method," *Comput. Electron. Agricult.*, vol. 137, pp. 52–58, May 2017. [Online]. Available: http://linkinghub.elsevier.com/retrieve/pii/S016816991631136X

[3] J. Patel, S. Shah, P. Thakkar, and K. Kotecha, "Predicting stock market index using fusion of machine learning techniques," *Expert Syst. Appl.*, vol. 42, no. 4, pp. 2162–2172, Mar. 2015. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0957417414006551

[4] G. Nie, W. Rowe, L. Zhang, Y. Tian, and Y. Shi, "Credit card churn forecasting by logistic regression and decision tree," *Expert Syst. Appl.*, vol. 38, no. 12, pp. 15273–15285, Nov. 2011. [Online]. Available: http://linkinghub.elsevier.com/retrieve/pii/S0957417411009237

[5] E. Yılmaz, "An expert system based on Fisher score and LS-SVM for cardiac arrhythmia diagnosis," *Comput. Math. Methods Med.*, vol. 2013, pp. 1–6, 2013. [Online]. Available: http://www.hindawi.com/journals/cmmm/2013/849674/

[6] J. Hur and J. W. Kim, "A hybrid classification method using error pattern modeling," *Expert Syst. Appl.*, vol. 34, no. 1, pp. 231–241, Jan. 2008. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0957417406002727

[7] J. H. Friedman, R. Tibshirani, and T. Hastie, *The Elements of Statistical Learning*. Berlin, Germany: Springer, 2009.

[8] A. L. Blum and P. Langley, "Selection of relevant features and examples in machine learning," *Artif. Intell.*, vol. 97, nos. 1–2, pp. 245–271, Dec. 1997. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0004370297000635

[9] H. Motoda and H. Liu, *Feature Selection for Knowledge Discovery and Data Mining*. Berlin, Germany: Springer, 1998.

[10] I. Guyon and A. Elisseeff, "An introduction to variable and feature selection," *J. Mach. Learn. Res.*, vol. 3, pp. 1157–1182, Jan. 2003. [Online]. Available: https://ci.nii.ac.jp/naid/80016398667/en/

[11] J. Phillips, E. Cripps, J. W. Lau, and M. R. Hodkiewicz, "Classifying machinery condition using oil samples and binary logistic regression," *Mech. Syst. Signal Process.*, vols. 60–61, pp. 316–325, Aug. 2015. [Online]. Available: http://linkinghub.elsevier.com/retrieve/pii/S0888327014005093

[12] F. Jamil, M. Abid, M. Adil, I. Haq, A. Q. Khan, and S. F. Khan, "Kernel approaches for fault detection and classification in PARR-2," *J. Process Control*, vol. 64, pp. 1–6, Apr. 2018. [Online]. Available: http://linkinghub.elsevier.com/retrieve/pii/S0959152418300039

[13] A. H. A. El-Atta and A. E. Hassanien, "Two-class support vector machine with new kernel function based on paths of features for predicting chemical activity," *Inf. Sci.*, vols. 403–404, pp. 42–54, Sep. 2017. [Online]. Available: http://linkinghub.elsevier.com/retrieve/pii/S0020025517306448

[14] H. Jiang, W.-K. Ching, K. F. C. Yiu, and Y. Qiu, "Stationary mahalanobis kernel SVM for credit risk evaluation," *Appl. Soft Comput.*, vol. 71, pp. 407–417, Oct. 2018. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1568494618303892

[15] H. Wang and D. Hu, "Comparison of SVM and LS-SVM for regression," in *Proc. Int. Conf. Neural Netw. Brain*, vol. 1, Oct. 2005, pp. 279–283.

[16] G. Zhang, B. E. Patuwo, and M. Y. Hu, "Forecasting with artificial neural networks: The state of the art," *Int. J. Forecasting*, vol. 14, pp. 35–62, Mar. 1998. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0169207097000447

[17] M. Khashei, A. Zeinal Hamadani, and M. Bijari, "A novel hybrid classification model of artificial neural networks and multiple linear regression models," *Expert Syst. Appl.*, vol. 39, no. 3, pp. 2606–2620, Feb. 2012. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0957417411012474

[18] X. Fan, L. Wang, and S. Li, "Predicting chaotic coal prices using a multi-layer perceptron network model," *Resour. Policy*, vol. 50, pp. 86–92, Dec. 2016. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0301420716301799

[19] F. A. de Oliveira, C. N. Nobre, and L. E. Zárate, "Applying artificial neural networks to prediction of stock price and improvement of the directional prediction index–case study of PETR4, petrobras, Brazil," *Expert Syst. Appl.*, vol. 40, no. 18, pp. 7596–7606, Dec. 2013. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0957417413004703

[20] S. Mabu, M. Obayashi, and T. Kuremoto, "Ensemble learning of rule-based evolutionary algorithm using multi-layer perceptron for supporting decisions in stock trading problems," *Appl. Soft Comput.*, vol. 36, pp. 357–367, Nov. 2015. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1568494615004603

[21] C. H. Li and S. C. Park, "Combination of modified BPNN algorithms and an efficient feature selection method for text categorization," *Inf. Process. Manage.*, vol. 45, no. 3, pp. 329–340, May 2009. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0306457308000897

[22] H. Liang, T. Yongqiang, and L. Xiang, "Research on drilling kick and loss monitoring method based on Bayesian classification," *Pakistan J. Statist.*, vol. 30, pp. 1251–1266, Dec. 2014. [Online]. Available: http://search.ebscohost.com/login.aspx?direct=true&db=aph&AN=102250832&lang=zh-cn&site=ehost-live

[23] M. Radovanović, A. Nanopoulos, and M. Ivanović, "Hubs in space: Popular nearest neighbors in high-dimensional data," *J. Mach. Learn. Res.*, vol. 11, pp. 2487–2531, 2010. [Online]. Available: http://dl.acm.org/citation.cfm?id=1756006.1953015

[24] M. Dash, K. Choi, P. Scheuermann, and H. Liu, "Feature selection for clustering–a filter solution," in *Proc. IEEE Int. Conf. Data Mining*, Dec. 2002, pp. 115–122.

[25] R. Caruana and D. Freitag, "Greedy attribute selection," in *Machine Learning Proceedings 1994*, W. W. Cohen and H. Hirsh, Eds. San Francisco, CA, USA: Morgan Kaufmann, 1994, pp. 28–36. [Online]. Available: http://www.sciencedirect.com/science/article/pii/B978155860335650012X

[26] E. Pashaei and N. Aydin, "Binary black hole algorithm for feature selection and classification on biological data," *Appl. Soft Comput.*, vol. 56, pp. 94–106, Jul. 2017. [Online]. Available: http://linkinghub.elsevier.com/retrieve/pii/S1568494617301242

[27] C. Qi, Z. Zhou, Y. Sun, H. Song, L. Hu, and Q. Wang, "Feature selection and multiple kernel boosting framework based on PSO with mutation mechanism for hyperspectral classification," *Neurocomputing*, vol. 220, pp. 181–190, Jan. 2017. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0925231216309158

[28] P. Shunmugapriya and S. Kanmani, "A hybrid algorithm using ant and bee colony optimization for feature selection and classification (AC-ABC Hybrid)," *Swarm Evol. Comput.*, vol. 36, pp. 27–36, Oct. 2017. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S2210650217302687

[29] P. Drotár, J. Gazda, and Z. Smékal, "An experimental comparison of feature selection methods on two-class biomedical datasets," *Comput. Biol. Med.*, vol. 66, pp. 1–10, Nov. 2015. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0010482515002917

[30] V. Bolón-Canedo, N. Sánchez-Maroño, and A. Alonso-Betanzos, "Recent advances and emerging challenges of feature selection in the context of big data," *Knowl.-Based Syst.*, vol. 86, pp. 33–45, Sep. 2015. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0950705115002002

[31] Z. Chen, C. Wu, Y. Zhang, Z. Huang, B. Ran, M. Zhong, and N. Lyu, "Feature selection with redundancy-complementariness dispersion," *Knowl.-Based Syst.*, vol. 89, pp. 203–217, Nov. 2015. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0950705115002567

[32] H. Akaike, "A new look at the statistical model identification," *IEEE Trans. Autom. Control*, vol. 19, no. 6, pp. 716–723, Dec. 1974.

[33] G. Schwarz, "Estimating the dimension of a model," *Ann. Statist.*, vol. 6, no. 2, pp. 461–464, Mar. 1978, doi: 10.1214/aos/1176344136.

[34] C. L. Mallows, "Some comments on C_p," *Technometrics*, vol. 15, no. 4, pp. 661–675, 1973. [Online]. Available: http://www.jstor.org/stable/1267380

[35] A. E. Hoerl and R. W. Kennard, "Ridge regression: Biased estimation for nonorthogonal problems," *Technometrics*, vol. 12, no. 1, pp. 55–67, Feb. 1970, [Online]. Available: https://www.tandfonline.com/doi/abs/10.1080/00401706.1970.10488634

[36] R. Tibshirani, "Regression shrinkage and selection via the lasso," *J. Roy. Stat. Soc., B (Methodol.)*, vol. 58, no. 1, pp. 267–288, Jan. 1996. [Online]. Available: http://www.jstor.org/stable/2346178

[37] E. Walker and J. B. Birch, "Influence measures in ridge regression," *Technometrics*, vol. 30, no. 2, pp. 221–227, May 1988. [Online]. Available: https://www.tandfonline.com/doi/abs/10.1080/00401706.1988.10488370

[38] S. A. van de Geer, "High-dimensional generalized linear models and the lasso," *Ann. Statist.*, vol. 36, no. 2, pp. 614–645, Apr. 2008, doi: 10.1214/009053607000000929.

[39] A. Nazemi and F. J. Fabozzi, "Macroeconomic variable selection for creditor recovery rates," *J. Banking Finance*, vol. 89, pp. 14–25, Apr. 2018. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S037842661830013X

[40] H.-D. Li, Y.-Z. Liang, Q.-S. Xu, and D.-S. Cao, "Model population analysis for variable selection," *J. Chemometrics*, vol. 24, nos. 7–8, pp. 418–423, Jul. 2010, [Online]. Available: https://onlinelibrary.wiley.com/doi/abs/10.1002/cem.1300

[41] G. Dhiman and V. Kumar, "Seagull optimization algorithm: Theory and its applications for large-scale industrial engineering problems," *Knowl.-Based Syst.*, vol. 165, pp. 169–196, Feb. 2019. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0950705118305768

[42] H. R. Tizhoosh, "Opposition-based learning: A new scheme for machine intelligence," in *Proc. Int. Conf. Comput. Intell. Modeling, Control Autom. Int. Conf. Intell. Agents, Web Technol. Internet Commerce (CIMCA-IAWTIC)*, vol. 1, 2005, pp. 695–701.

[43] E. Iturbide, J. Cerda, and M. Graff, "A comparison between LARS and LASSO for initialising the time-series forecasting auto-regressive equations," *Procedia Technol.*, vol. 7, pp. 282–288, 2013. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S2212017313000364

[44] M. Zhao, C. Fu, L. Ji, K. Tang, and M. Zhou, "Feature selection and parameter optimization for support vector machines: A new approach based on genetic algorithm with feature chromosomes," *Expert Syst. Appl.*, vol. 38, no. 5, pp. 5197–5204, May 2011. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0957417410011851

[45] P. G. Espejo, S. Ventura, and F. Herrera, "A survey on the application of genetic programming to classification," *IEEE Trans. Syst., Man, Cybern., C Appl. Rev.*, vol. 40, no. 2, pp. 121–144, Mar. 2010.

[46] D. E. Goldberg, *Genetic Algorithms in Search, Optimization, and Machine Learning*, 1st ed. Boston, MA, USA: Addison-Wesley, 1989.

[47] H. Wang, B. Zheng, S. W. Yoon, and H. S. Ko, "A support vector machine-based ensemble algorithm for breast cancer diagnosis," *Eur. J. Oper. Res.*, vol. 267, no. 2, pp. 687–699, Jun. 2018. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0377221717310810
[48] A. A. Aburomman and M. B. Ibne Reaz, "A novel SVM-kNN-PSO ensemble method for intrusion detection system," *Appl. Soft Comput.*, vol. 38, pp. 360–372, Jan. 2016. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1568494615006328
[49] J. K. Anlauf and M. Biehl, "The adatron: An adaptive perceptron algorithm," *Europhys. Lett.*, vol. 10, no. 7, pp. 687–692, 1989. [Online]. Available: https://ci.nii.ac.jp/naid/80004981619/en/
[50] J. C. Platt, "Fast training of support vector machines using sequential minimal optimization," in *Advances in Kernel Methods*, B. Schölkopf, C. J. C. Burges, and A. J. Smola, Eds. Cambridge, MA, USA: MIT Press, 1999, pp. 185–208. [Online]. Available: http://dl.acm.org/citation.cfm?id=299094.299105
[51] P. E. Gill, W. Murray, and M. H. Wright, *Practical Optimization*. New York, NY, USA: Academic, 1981. [Online]. Available: http://ci.nii.ac.jp/ncid/BA07243977
[52] Y. Gu, W. Zhao, and Z. Wu, "Online adaptive least squares support vector machine and its application in utility boiler combustion optimization systems," *J. Process Control*, vol. 21, no. 7, pp. 1040–1048, Aug. 2011. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0959152411001090
[53] X. Yuan, C. Chen, Y. Yuan, Y. Huang, and Q. Tan, "Short-term wind power prediction based on LSSVM-GSA model," *Energy Convers. Manage.*, vol. 101, pp. 393–401, Sep. 2015. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0196890415005300
[54] W. Sun and J. Sun, "Daily PM 2.5 concentration prediction based on principal component analysis and LSSVM optimized by cuckoo search algorithm," *J. Environ. Manage.*, vol. 188, pp. 144–152, Mar. 2017. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0301479716309835
[55] J. A. K. Suykens, J. Vandewalle, and B. De Moor, "Optimal control by least squares support vector machines," *Neural Netw.*, vol. 14, no. 1, pp. 23–35, Jan. 2001. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0893608000000770
[56] E. Çomak, K. Polat, S. Güneş, and A. Arslan, "A new medical decision making system: Least square support vector machine (LSSVM) with fuzzy weighting pre-processing," *Expert Syst. Appl.*, vol. 32, no. 2, pp. 409–414, Feb. 2007. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0957417405003507
[57] X. Yuan, Q. Tan, X. Lei, Y. Yuan, and X. Wu, "Wind power prediction using hybrid autoregressive fractionally integrated moving average and least square support vector machine," *Energy*, vol. 129, pp. 122–137, Jun. 2017. [Online]. Available: http://www.3wsciencedirect.com/science/article/pii/S0360544217306606
[58] A. Koul. (2017). *PredPsych: Predictive Approaches Psychology*. [Online]. Available: https://CRAN.R-project.org/package=PredPsych
[59] J. Friedman, T. Hastie, and R. Tibshirani. (2017). *Regularization Paths for Generalized Linear Models via Coordinate Descent*. [Online]. Available: http://www.jstatsoft.org/v33/i01/
[60] M. Kuhn, J. Wing, S. Weston, A. Williams, C. Keefer, A. Engelhardt, T. Cooper, Z. Mayer, B. Kenkel, T. R. C. Team, M. Benesty, R. Lescarbeau, A. Ziem, L. Scrucca, Y. Tang, and C. Candan. (2016). *Caret: Classification Regression Training*. [Online]. Available: https://CRAN.R-project.org/package=caret
[61] A. Karatzoglou, A. Smola, K. Hornik, and A. Zeileis, "KernlabAnS4Package for kernel methods inR," *J. Stat. Softw.*, vol. 11, no. 9, pp. 1–20, 2004. [Online]. Available: http://www.jstatsoft.org/v11/i09/
[62] D. Dheeru and E. K. Taniskidou. (2017). *UCI Machine Learning Repository*. [Online]. Available: http://archive.ics.uci.edu/ml
[63] J. A. Ramey. (2016). *Datamicroarray: Collection Data Sets for Classification*. [Online]. Available: https://github.com/ramhiser/datamicroarray and http://ramhiser.com

HE JIANG received the bachelor's degree in applied mathematics and the first master's degree in probability and mathematical statistics from Lanzhou University, in 2009 and 2012, respectively, the second master's degree in statistics from Florida State University, in 2014, and the Ph.D. degree from Florida State University, USA. He is currently an Associate Professor of statistics with the Jiangxi University of Finance and Economics. His main research interests are focused on machine learning, variable selection, and data mining techniques related to big data.

YE YANG received the bachelor's degree from the Changsha University of Science and Technology, in 2016. He is currently pursuing the master's degree in probability and mathematical statistics with the Jiangxi University of Finance and Economics. His main research interests are variable selection and data mining.

WEIYING PING received the bachelor's degree in statistics, in 2002, the master's degree in statistics from the Zhongnan University of Economics and Law, China, in 2005, and the Ph.D. degree from the Zhongnan University of Economics and Law. She is currently a Professor of statistics with the Jiangxi University of Finance and Economics. Her main research interests are economic statistics and data mining.

YAO DONG received the Ph.D. degree from Lanzhou University, China, in 2015. She was a Visiting Student with Florida State University for 16 months. She is currently a Lecturer with the Jiangxi University of Finance and Economics, Nanchang, China. She has published more than 20 academic articles in SCI retrieval, and most of them have been published in top international journals. Her research interests include time-series analysis, applied statistics, artificial intelligence, high-dimensional data analysis, and energy prediction theory and methods.