Research on Student Performance Prediction Based on SVM Optimized by Hybrid SSA

Xiaoling Guo, Rui Wang, Xinghua Sun, Qiyue Zhang

Abstract—**A student performance prediction model based on support vector machine optimized by the hybrid sparrow search algorithm is proposed. Firstly, the basic Sparrow Search Algorithm is enhanced through hybrid improvements. During the discovery search process, levy flight and a golden sine function strategy are introduced to expand the search area for discoverers. Additionally, a t-distribution perturbation strategy is incorporated to adjust the position of discoverers, thereby enhancing both flexibility and effectiveness. In the warning search phase, a normally distributed random number is utilized. The variance of this normal distribution decreases progressively with population size, which enhances the algorithm's local development capability. Simultaneously, a random number following Poisson distribution is introduced, its expected value and variance increase with iterations to bolster global search ability. Subsequently, the hybrid improved algorithm is employed to optimize parameters of support vector machine aimed at determining the optimal combination of penalty factor and kernel function. This established student performance prediction model is applied for forecasting student outcomes. Final conclusions indicate that compared to traditional SVM model, Neural Network and SSA-SVM algorithm, the HSSA-SVM algorithm significantly improves accuracy and precision in predicting student performance, providing decision-making basis for teacher teaching and student learning behavior.**

Index Terms—**Sparrow Search Algorithm, Support Vector Machine, Performance prediction, Education and teaching**

I. Introduction

Education data mining has become a crucial area in artificial intelligence applications. Its main goal is to identify the influencing factors for teaching behaviors, providing decision-making guidance for educators and students [1]. By utilizing machine learning techniques, educational data mining can predict students' academic performance in advance. These predictive outcomes enable teachers to intervene proactively, offering essential support for student learning [2].

Academic performance is a key metric for assessing students' overall achievements and the quality of instruction provided by educators. An effective prediction mechanism allows schools to implement timely interventions in students' learning paths based on expected outcomes, delivering precise and tailored instructional guidance suited to diverse student groups. This can significantly enhance teaching quality.

Manuscript received Nov 15, 2024; revised May 19, 2025.

This work was supported in part by the Doctoral Research Fund of Hebei North University under Grant BSJJ202322 and in part by the Research Project on Educational and Teaching Reform of Hebei North University under Grant JG2024044.

Xiaoling Guo is a lecturer of Hebei North University, Zhangjiakou 075000, China. (e-mail: 175666832@qq.com).

Rui Wang is a lecturer of Hebei North University, Zhangjiakou 075000, China. (corresponding author to provide phone:189-313-37970; fax:189-313-37970; e-mail: 33403408@qq.com).

Xinghua Sun is a professor of Hebei North University, Zhangjiakou 075000, China. (e-mail:1030704295@qq.com).

Qiyue Zhang is a undergraduate student of Hebei North University, Zhangjiakou 075000, China. (e-mail:3205735627@qq.com).

II. Related Work

Educational data mining endeavors to identify and address prevalent research challenges in the educational field. It employs a variety of data mining techniques and advanced artificial intelligence deep learning algorithms to analyze educational data, predict students' learning outcomes and future employment prospects, thereby establishing a robust foundation for their successful career paths [3]. Additionally, it serves as an effective tool for validating educational systems and enhancing the quality of education and learning processes [4].

Traditional predictive models in machine learning filed, such as linear regression, neural networks (NN), fuzzy logic theory, and support vector machines (SVM) have been extensively deployed in the educational field [5] [6]. Notably, neural networks require large sample sizes to ensure precise performance predictions; a deficiency in sample quantity can lead to inadequate training and a subsequent decline in prediction accuracy [7]. Consequently, when confronted with small sample sizes related to academic grades, SVM model is typically preferred due to its numerous advantages. SVM is widely recognized as a versatile and highly effective learning approach grounded in the principle of structural risk minimization. This method demonstrates commendable stability and predictability across classification and regression tasks, making it particularly well-suited for binary classification applications [8] [9].

Lv et al. [10] developed performance prediction models employing three machine learning algorithms: perceptron, SVM, and NN. Upon evaluating these models based on metrics such as accuracy, recall, F-value, misclassified samples count, and overall model precision, SVM emerged as the most suitable algorithm for constructing an effective prediction framework. Wang et al. [11] introduced a degree warning model utilizing SVM techniques. This innovative model was constructed with actual grade data from five disciplines at a specific university, including Mathematics and Applied Mathematics, Chinese Language and Literature, Accounting, among others, demonstrating its feasibility and effectiveness through comprehensive random experimental results. Zhang et al. [12], on their part, opted for the SVM method as an instrument for predicting college entrance examination scores. In their simulation experiment, they

harnessed feature scores from six simulated exams gathered from ordinary university students as primary training data to forecast both feature scores and admission batches pertinent to the college entrance examination. The viability of employing SVM in forecasting national college entrance examination scores was confirmed through comparisons with neural networks algorithm.

In the realm of performance classification and prediction, SVM model typically demonstrates exceptional binary classification capabilities, exhibiting elevated accuracy rates and robust generalization potential, particularly when confronted with limited datasets [13] [14]. Nevertheless, ascertaining the optimal synergy between penalty factors and kernel function parameters for SVM presents considerable challenges. When these parameters are directly employed to forecast student grades, the resultant predictive accuracy often remains disappointingly low, coupled with suboptimal operational efficiency [15]. In recent years, swarm optimization algorithms have emerged as a promising avenue to enhance model parameter optimization for performance prediction endeavors. This innovative approach has yielded notable advancements in predictive accuracy, universality, and generalization capacity. To adeptly tackle the parameter-related dilemmas associated with SVM, researchers have harnessed an array of techniques including Genetic Algorithms, Particle Swarm Optimization, Sparrow Search Algorithm [16], Simulated Annealing, Firefly Algorithm [17], among others. These methodologies strive to pinpoint critical parameters that significantly bolster predictive precision.

To further elevate predictive ability of SVM, Jin et al. [18] implemented GA to refine SVM model parameters and developed a GA-SVM framework specifically tailored for predicting college entrance examination scores. By leveraging GA technology to fine-tune both the penalty factor and kernel function parameter of the Radial Basis Function-Support Vector Machine (RBF-SVM) model, initial encoding was conducted meticulously. Subsequently, within the constraints dictated by the objective function, a global search for optimal parameter combinations was executed through processes such as random selection, crossover operations, and mutation strategies, thereby effectively enhancing both accuracy and efficiency in predicting college entrance examination scores. Compared with BP, MLR and SVM, the accuracy and precision of GA-SVM model in predicting college entrance examination scores had been significantly improved, which can provide reference for the direction of college entrance examination review. Zhao et al. [19] introduced an enhanced Artificial Bee Colony (ABC) algorithm aimed at optimizing SVM for the prediction and analysis of mathematical scores. Their innovative approach integrated the Random Forest (RF) algorithm for feature extraction from input variables, thereby refining the conventional ABC methodology by initializing a two-dimensional uniform population and updating food sources based on Euclidean distance, ultimately constructing a robust classification model. Guo et al. [20] devised a sports performance prediction model leveraging Particle Swarm Optimization-Support Vector Machine (PSO-SVM). By fine-tuning SVM parameters through the PSO algorithm, they established a predictive framework for athletic performance. This model was subsequently deployed within cloud computing platforms, showcasing the potential applications of such technologies in educational contexts. The accuracy of sports performance predictions saw significant enhancement through an efficient data processing paradigm, thus fostering further advancements in informatization. Zhang et al. [21] presented a predictive model grounded in SVM optimized via SSA. Initially, SSA was employed to refine SVM parameters, specifically targeting the penalty factor and kernel function parameter through iterative searches for global optimal positions. The resulting optimized SVM classifier was then utilized to forecast student grades effectively. Ultimately, this predictive framework underwent rigorous evaluation using the UCI-Mat dataset. Experimental findings revealed that compared to traditional methodologies such as standard SVM, BP, and RF algorithms, the SSA-SVM approach markedly enhanced prediction accuracy, achieving an impressive success rate of up to 95%.

III. Hybrid Sparrow Search Algorithm

A. Hybrid Sparrow Search Algorithm (HSSA)

SSA is a novel optimization algorithm known for its robust global search capabilities, high stability, and rapid convergence. However, it can get trapped in local optima. To address this issue, we propose HSSA, a hybrid improved algorithm designed to enhance global search capacity.

The levy flight strategy represents a stochastic behavior approach, where both the step size and direction governing the search process adhere to a specific probability distribution. This characteristic significantly enhances the algorithm's ability for global exploration. Additionally, we incorporate the relationship between the sine function and the unit circle, enabling comprehensive traversal of all positions on this circle. Furthermore, by introducing a golden ratio coefficient, we can effectively reduce the solution space while accelerating search speed. The formula for discoverers at $R_2 < ST$ is shown in Eq. (1)-(2), which integrates levy flight with the golden sine mechanism to expand the search area for discoverers.

$$X_i^{t+1} = X_i^t \cdot \left|\sin(r_1)\right| + \gamma \cdot levy(s) \oplus dis \tag{1}$$

$$\gamma = r_2 \cdot \sin(r_1) \cdot \exp(\frac{-i}{\alpha \cdot T_{\max}}) \tag{2}$$

r_1 is a random number of $[0, 2\pi]$. r_2 is a random number of $[0, \pi]$. α is a random number of $[0,1]$. Levy flight is shown in Eq. (3)-(6). β generally is 1.5.

$$levy(s) \sim \left|s\right|^{-1-\beta} \tag{3}$$

$$s = \frac{u}{\left|v\right|^{1/\beta}} \tag{4}$$

$$u \sim N(0, \sigma_u^2), v \sim N(0, \sigma_v^2) \tag{5}$$

$$\sigma_u = \left\{ \frac{\Gamma(1+\beta)\sin(\pi\beta/2)}{\Gamma[(1+\beta)/2]2^{(\beta-1)/2}} \right\}^{1/\beta}, \sigma_v = 1 \tag{6}$$

dis is shown in Eq. (7)-(10). X_{best}^t is the best position. X_i^t

is the current position.

$$dis = \left|\theta_1 \cdot X_{best}^t - \theta_2 \cdot X_i^t\right| \tag{7}$$

$$\theta_1 = -\pi + 2\pi \cdot (1-\tau) \tag{8}$$

$$\theta_2 = -\pi + 2\pi \cdot \tau \tag{9}$$

$$\tau = (\sqrt{5}-1)/2 \tag{10}$$

The t-distribution perturbation strategy is used to perturb the position of the discoverers. This approach can enhance the flexibility and solving effectiveness of the algorithm. For the parameter associated with this distribution, $t(n \to \infty) \to N(0,1)$, $t(n=1) = C(0,1)$. The two boundaries of t-distribution are Gaussian distribution and Cauchy distribution respectively. Here we utilizes this perturbation strategy when $R_2 \geq ST$, the improved formula is provided in the Eq. (11):

$$X_i^{t+1} = X_i^t + t\text{-}distribution(n) * X_i^t \tag{11}$$

In this context, the current iteration number *t* is used as freedom degree parameter of *t*-distribution. This approach enhances global search capabilities during early iterations while simultaneously improving local development potential in later stages.

The update process for a discoverer's position follows the Eq. (12):

$$X_i^{t+1} = \begin{cases} X_i^t \cdot \left|\sin(r_1)\right| + \gamma \cdot levy(s) \oplus dis, R_2 < ST \\ X_i^t + t\text{-}distribution(n) * X_i^t, R_2 \geq ST \end{cases} \tag{12}$$

$R_2 \in [0,1]$ and $ST \in [0.5,1]$, is respectively warning value and safety value.

Positions of followers are updated according to Eq. (13):

$$X_i^{t+1} = \begin{cases} Q \bullet \exp(\frac{X_{worst}^t - X_i^t}{i^2}), i > \frac{M}{2} \\ X_p^{t+1} + \left|X_i^t - X_p^{t+1}\right| \bullet A^+ \bullet L, i \leq \frac{M}{2} \end{cases} \tag{13}$$

Q follows a normal distribution. X_{worst}^t is the global worst position. M is the population number. X_p^{t+1} represents the optimal position in the $t+1$ generation. A is a matrix of $1*d$.

The warner search is the last step of SSA, as it has many parameters. Good improvements can greatly enhance population diversity and address the problem of the population converging to local extrema.

The step size control parameters β and k of the warner search play an important role in balancing global search capability and local development capability. Directly using random numbers β and k may not satisfy the exploration of the algorithm in the solution space and may lead to falling into local optima. They are refined by Eq. (14)-(17).

$$B = normrnd(0, (1-m/M), M, D) \tag{14}$$

$$\beta' = B(m,d) \tag{15}$$

β' is a random number which obeys Normal distribution with variance 1-*m*/*M*. *m* denotes the population size, *M* is the total population size. The variance's decreases with population changes will promote local development.

$$K = poissrnd(i, M, D) \tag{16}$$

$$k' = K(i,d) \tag{17}$$

k' is defined as a random number which obeys Poisson distribution. The expected value and variance are both i . i express current iteration. As i increases, global exploration ability enhances.

The position update for Warner's algorithm is given by the Eq. (18):

$$X_i^{t+1} = \begin{cases} X_{best}^t + \beta' \bullet \left|X_i^t - X_{best}^t\right|, f_i \neq f_g \\ X_i^t + k' \bullet (\frac{\left|X_i^t - X_{worst}^t\right|}{(f_i - f_w) + \varepsilon}), f_i = f_g \end{cases} \tag{18}$$

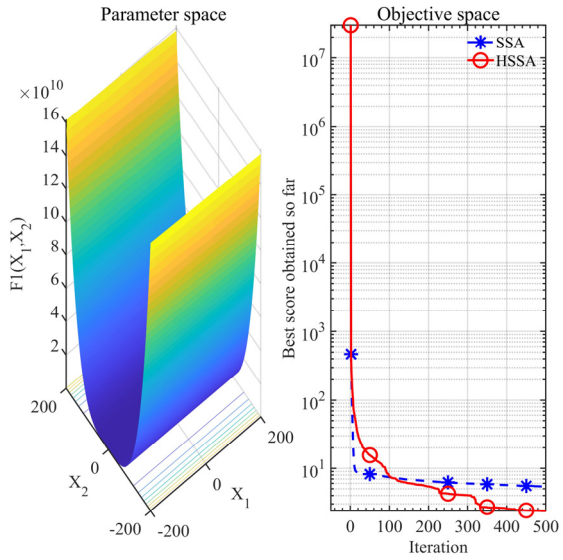
X_{best}^t is the global optimal position at the t iteration. f_i , f_g and f_w is respectively the current, global optimal, and the worst fitness value.

B. Test Analysis of HSSA

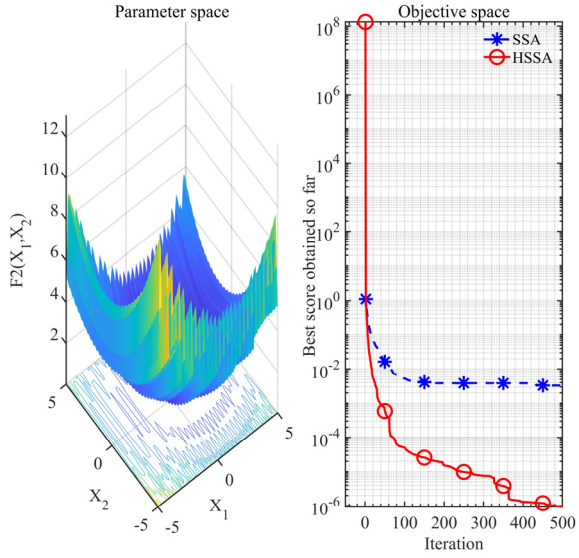
To evaluate HSSA's effectiveness, we tested it on four functions: one unimodal and three multimodal (see Table I). The population size is set at 15 with a maximum of 500 iterations. Each experiment is run 30 times to obtain average results. The outcomes are shown in Fig. 1(a)-(d), highlighting HSSA's superior convergence speed and accuracy. The HSSA algorithm can converge to optimal values faster on three multimodal functions. Its ability to jump out of local optima is excellent.

TABLE I
TEST FUNCTIONS AND PARAMETERS

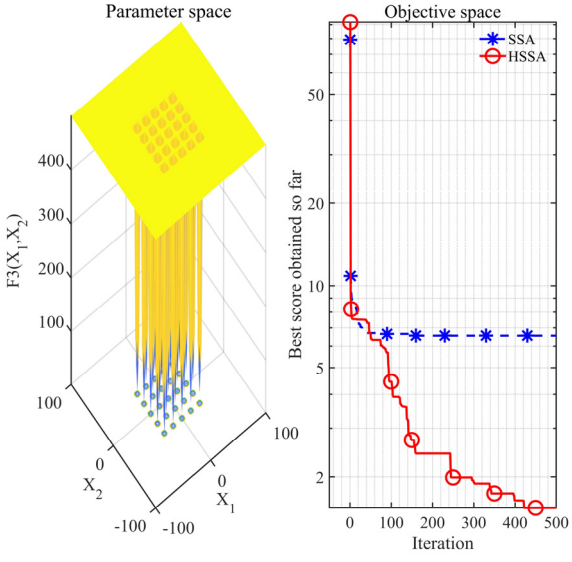
Test Function
$F_1(x) = \sum_{i=1}^{n-1}[100(x_{i+1} - x_i^2)^2 + (x_i - 1)^2]$, d=30, [-30,30], 0
$F_2(x) = 0.1\{\sin^2(3\pi x_1) + \sum_{i=1}^{n}(x_i - 1)^2[1 + \sin^2(3\pi x_i + 1)] + (x_n - 1)^2[1 + \sin^2(2\pi x_n)]\} + \sum_{i=1}^{n} u(x_i, 5, 100, 4)$, d=30, [-50,50], 0
$F_3(x) = \left(\frac{1}{500} + \sum_{j=1}^{25}\frac{1}{j + \sum_{i=1}^{2}(x_i - a_{ij})^6}\right)^{-1}$, d=2, [-65,65], 1
$F_4(x) = -\sum_{i=1}^{5}[(X - a_i)(X - a_i)^T + c_i]^{-1}$, d=4, [0,10], -1



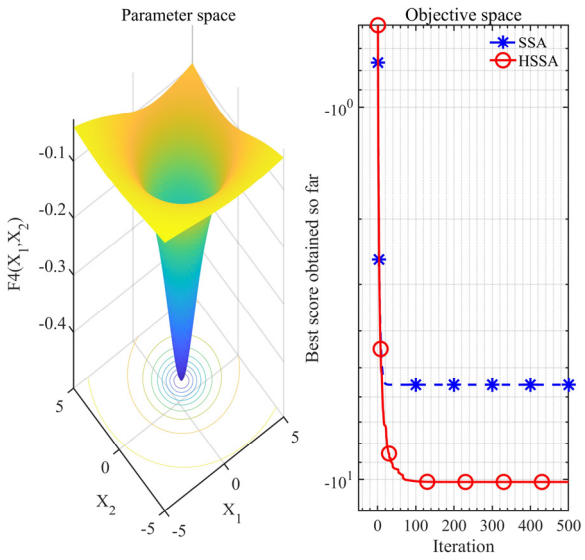
(a)



(b)



(c)



(d)

Fig. 1. The convergence curve of test function. (a) shows the $F_1(x)$. (b) shows the $F_2(x)$. (c) shows the $F_3(x)$. (d) shows the $F_4(x)$.

IV. HSSA-SVM Performance Prediction Model

A. SVM (Support Vector Machine)

SVM is adept at handling both linearly separable and non-linearly separable data while constructing optimal decision boundaries within high-dimensional spaces. Its core idea is to find a hyperplane which separates sample points of different categories as much as possible and maximizes the gap between the two categories. Its schematic diagram is depicted in Fig. 2.

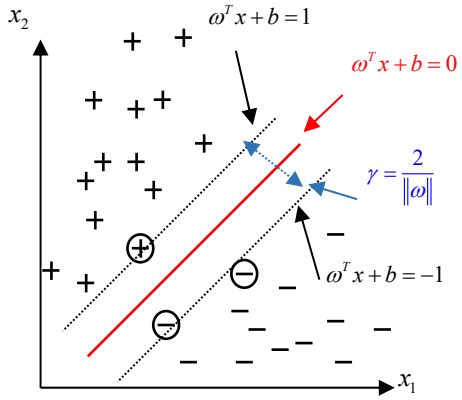


Fig. 2. Support Vector Machine

This hyperplane is referred to as the maximum margin hyperplane and performs classification predictions effectively. Specifically, SVM maps samples into a high-dimensional feature space where linear separability can be achieved; if no linear hyperplane exists within the original input space for data separation purposes, SVM employs kernel function to transfer computational complexity from this high-dimensional feature space back into the original input domain. The commonly employed kernel functions in SVM include the linear kernel, polynomial kernel, and Gaussian kernel. The training process of SVM is framed as a convex optimization problem, aimed at minimizing the

model's structural risk. During this optimization process, SVM primarily concentrates on samples that are situated near the decision boundary, which are referred to as support vectors.

The SVM linear regression model can be represented mathematically as shown in Eq. (19).

$$f(x) = \omega^T x + b \tag{19}$$

Transform it into a convex quadratic function programming problem, with its optimization expression detailed in Eq. (20).

$$\begin{aligned} &\min_{w,\xi} : \frac{1}{2}\omega^T \omega + C\sum_{i=1}^{n} \xi_i \\ &s.t. \begin{cases} y_i(\omega^T \Phi(x_i) + b) \geq 1 - \xi_i \\ \xi_i \geq 0 \\ i = 1, 2, ..., n \end{cases} \end{aligned} \tag{20}$$

By introducing Lagrange multipliers, we derive the dual problem associated with SVM nonlinear regression as presented in Eq. (21).

$$\begin{aligned} &\max_{\omega,b}(\sum_{i=1}^{n} y_i(\hat{\alpha}_i + \alpha_i) - \varepsilon(\hat{\alpha}_i + \alpha_i)) \\ &-\frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n}(\hat{\alpha}_i - \alpha_i)(\hat{\alpha}_j - \alpha_j)x_i x_j \\ &s.t.\sum_{i=1}^{n}(\hat{\alpha}_i - \alpha_i) = 0, 0 \leq \hat{\alpha}_i, \alpha_i \leq C \end{aligned} \tag{21}$$

Incorporating kernel technique, the expression for SVM nonlinear regression model is shown in Eq. (22).

$$\begin{aligned} &f(x) = \sum_{i=1}^{n}(\hat{\alpha}_i - \alpha_i)K(x, x_i) + b \\ &K(x_i, x_j) = \Phi(x_i)\Phi(x_j) \end{aligned} \tag{22}$$

A commonly utilized representation of the Gaussian radial basis kernel function is provided by Eq. (23).

$$K(x_i, x_{i'}) = \exp[-\gamma\sum_{j=1}^{n}(x_{ij} - x_{ij'})^2] \tag{23}$$

SVM is widely considered to be very suitable for binary classification applications by determining the combination of penalty factors and kernel function parameters, and can effectively predict whether students' course grades meet the standards.

The “fitcsvm” function provided by MATLAB can be used to train SVM classifiers, and we can optimize the performance of SVM classifiers by adjusting the parameters of this function. These parameters include BoxConstraint, KernelFunction and KernelScale. BoxConstraint is used to adjust the penalty factor C of SVM, KernelFunction is used to specify different kernel functions, and KernelScale is used to adjust the parameters of kernel functions. The BoxConstraint and KernelScale provided by MATLAB can be adjusted within the range of 10^{-5}-10^{+5}. Finding the optimal combination of these parameters can optimize the performance of the SVM classifier.

B. HSSA-SVM Prediction Model

Fig. 3 shows the flowchart of student performance prediction based on the HSSA-SVM model.

The specific process is as follows:

(1) Preprocess the sample data, including converting non-numeric data into numeric data, removing abnormal values or invalid data, and completing missing values in the sample data.

(2) Divide the sample data into a training sample set and a testing sample set at a specific ratio.

(3) Label the output classification attribute data in both the training and testing samples, and convert the output classification attribute data into positive or negative sample labels according to the classification criteria.

(4) Separate the training samples into a feature matrix and label vector, while similarly dividing the test samples into a feature matrix and label vector.

(5) Through “OptimizeHyperparameters” of the “fitcsvm” function, and using the training sample feature matrix and label vector to train the SVM model, the initial values of “BoxConstraint” and “KernelScale” are obtained, which is the initialization of SVM parameters.

(6) HSSA iteratively searches for the optimal combination of “BoxConstraint” and “KernelScale”, which are the combination parameters of the SVM model's “penalty factor” and “kernel function” parameters.

(7) The SVM model trained with the optimal parameter combination of “BoxConstraint” and “KernelScale” is used to classify and predict the data in the test sample feature matrix.

(8) Finally, obtain classification prediction results and compute prediction accuracy.

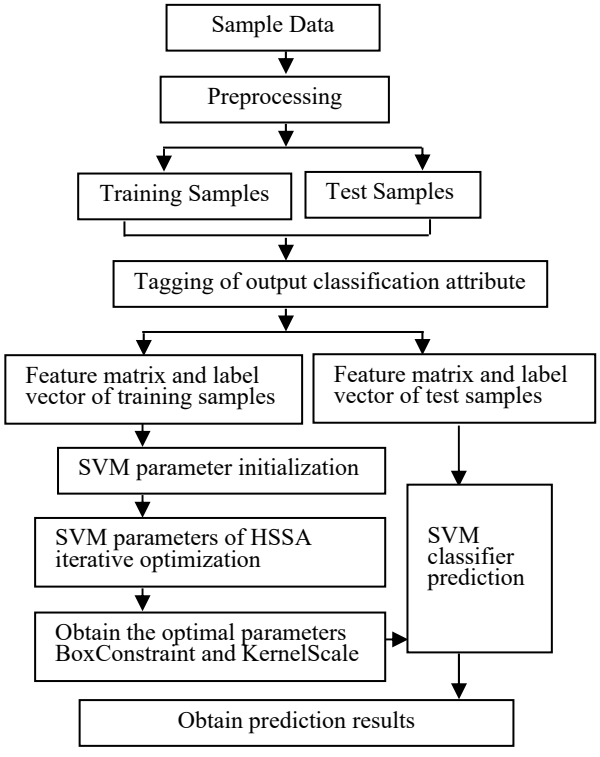


Fig. 3. HSSA-SVM Prediction Model

V. Experiment Simulation And Analysis

A. Dataset and Preprocessing

This study collected the performance data of 300 college students studying computer network courses in 8 classes of different majors as the experimental simulation dataset, as shown in Table II. The dataset includes 10 feature attributes. The feature attributes in the sample dataset include the student's major, semester, class hours, learning scores of video materials, average scores of chapter tests, learning scores of chapter content, participation scores in course discussions, average homework scores, and offline classroom check-in scores. The final exam score serves as the output classification attribute.

First, convert the feature attribute data “Specialty” into numerical values. Subsequently, utilize the “cvpartition” function to randomly partition the sample data into a training set and a testing set at an 80:20 ratio. In this way, there are 240 training samples and 60 test samples. Next, label the “Score” attribute in both training and test samples. Samples with a “Score” value below 60 are classified as negative samples and assigned a label of -1; conversely, samples with a “Score” value equal to or greater than 60 are classified as positive samples and labeled +1. Finally, separate the training data into a feature matrix and a label vector; similarly divide the test data into a feature matrix and a label vector for test samples.

Table II
Dataset Properties and Description

Property	Description
Speciality	Student's major
Semester	Course opening semester
Period	Learning hours
Video	Video materials average score
Test	Chapter test average score
Chapter	Chapter content learning average score
Discussion	Score for participating in course discussions
Homework	Average homework score
Attendance	Offline classroom check-in score
Score	Final exam scores

B. Evaluation Indicators

To evaluate the predictive model's performance, we employed confusion matrix analysis as an evaluation tool, referred to Table III.

Table III
The confusion matrix

	Actual Positive(1)	Actual Negative(-1)
Predicted Positive(1)	TP	FP
Predicted Negative(-1)	FN	TN

In student performance prediction research, seven primary evaluation metrics are utilized. They are respectively: Accuracy, Error Rate, Sensitivity, Specificity, Precision, Recall and F1-Score.

Accuracy is defined as the ratio of correctly predicted instances to all instances within the dataset; it reflects how effectively a classifier or model can accurately assess overall sample classifications. A higher accuracy indicates superior performance by the classification model, as expressed in the Eq. (24).

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{24}$$

Error Rate refers to the ratio of all incorrectly predicted samples to all samples examined. The sum of Error Rate and Accuracy equals 1. The Error Rate can be articulated through the Eq. (25).

$$Error\ Rate = \frac{FP + FN}{TP + TN + FP + FN} \tag{25}$$

The Sensitivity refers to the proportion of correctly predicted positive samples to the actual positive samples, which can measure the classifier's recognition ability for positive samples, as expressed in the Eq. (26).

$$Sensitivity = \frac{TP}{TP + FN} \tag{26}$$

The Specificity refers to the proportion of correctly predicted negative samples to the actual negative samples, which can measure the classifier's recognition ability for negative samples, as illustrated in the Eq. (27).

$$Specificity = \frac{TN}{FP + TN} \tag{27}$$

Precision refers to the ratio of correctly predicted positive samples to all predicted positive samples, or the ratio of correctly predicted negative samples to all predicted negative samples. It reflects the classifier's ability to predict accurately. The higher the precision value, the more accurate the classifier's prediction, as shown in the Eq. (28)-(29).

$$Precision_{positive} = \frac{TP}{TP + FP} \tag{28}$$

$$Precision_{negative} = \frac{TN}{FN + TN} \tag{29}$$

Recall is a measure of the completeness of the search results. The recall rate of positive samples is the Sensitivity, and the recall rate of negative samples is the Specificity, as illustrated in Eq. (30)-(31).

$$Recall_{positive} = \frac{TP}{TP + FN} = Sensitivity \tag{30}$$

$$Recall_{negative} = \frac{TN}{FP + TN} = Specificity \tag{31}$$

The F1-Score serves as a comprehensive evaluation metric, representing the harmonic mean between precision and recall. It is particularly well-suited for assessing the performance of classification models in scenarios characterized by imbalanced positive and negative sample categories. A higher

F1-Score indicates greater model stability, as illustrated in the Eq. (32)-(33).

$$F1\text{-}Score_{positive} = \frac{2 * Recall_{positive} * Precision_{positive}}{Recall_{positive} + Precision_{positive}} \quad (32)$$

$$F1\text{-}Score_{negative} = \frac{2 * Recall_{negative} * Precision_{negative}}{Recall_{negative} + Precision_{negative}} \quad (33)$$

C. Results and Analysis

To demonstrate the validity and feasibility of HSSA-SVM classification prediction, we compared its performance with three other algorithms: SVM, Neural Network (NN), and SSA-SVM.

Here SSA-SVM refers to optimizing SVM using the original SSA algorithm, while HSSA-SVM refers to optimizing SVM using a hybrid improved SSA algorithm. Neural network (NN) algorithm is also classic algorithms in the field of performance prediction.

The final prediction results are illustrated in the Fig. 4. The cross sign represents the real sample, and the circle represents the predicted result.The coincidence of cross signs and circles indicates correct predictions, while other situations indicate incorrect predictions.

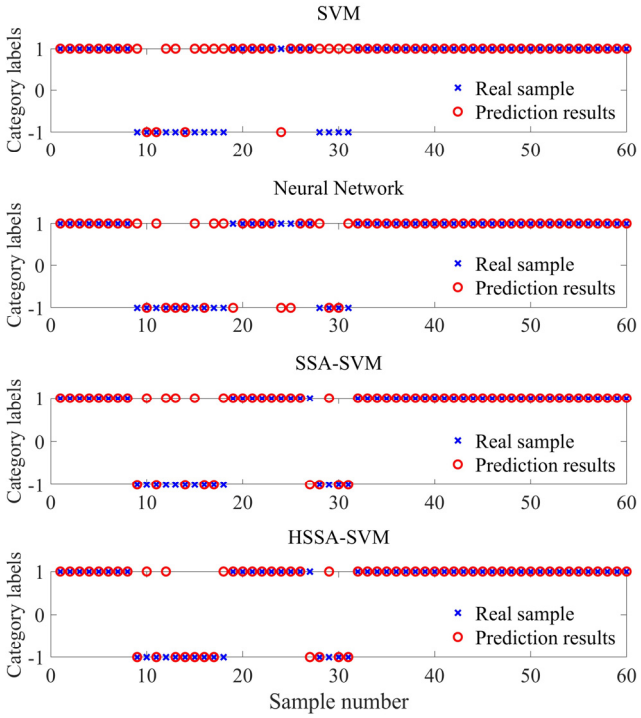


Fig. 4. Performance classification prediction

From the Fig. 4, we can obtain the values of the confusion matrix value for the four algorithms, as shown in Table IV.

TABLE IV
THE CONFUSION MATRIX VALUE OF FOUR ALGORITHMS

	TP	FP	FN	TN
SVM	45	11	1	3
NN	43	7	3	7
SSA-SVM	45	6	1	8
HSSA-SVM	45	4	1	10

Among the 60 tested samples, there were actually 46 positive samples and 14 negative samples. As shown in Fig. 4, when using SVM directly for classification prediction, one positive sample (1) was incorrectly predicted as a negative sample (-1), while eleven negative samples (-1) were mistakenly classified as positive samples (1). In total, SVM made twelve incorrect predictions, resulting in a relatively high Error Rate. In contrast, when employing NN for classification prediction, three positive samples (1) were misclassified as negative samples (-1), along with seven negative samples (-1) being erroneously predicted as positive ones (1). There were a total of 10 prediction errors in NN, and the Error Rate was also relatively high. The SSA-SVM yielded one misclassification of a positive sample (1) as a negative sample (-1), alongside six instances where negative category (-1) samples were inaccurately identified as positive category (1). Consequently, SSA-SVM recorded seven total prediction errors, with a decrease in Error Rate and a certain improvement in Accuracy. Utilizing the hybrid improved HSSA-SVM for classification predictions led to one instance where a positive class sample (1) was misclassified as a negative class sample (-1), along with four cases of negative class samples (-1) being wrongly classified as positives (1). Overall, HSSA-SVM misclassified five samples. The prediction accuracy was significantly improved.

Error Rate(SVM)=12/60=0.2;

Accuracy(SVM)=48/60=0.8;

Error Rate(NN)=10/60=0.167;

Accuracy(NN)=50/60=0.833;

Error Rate(SSA-SVM)=7/60=0.117;

Accuracy(SSA-SVM)=53/60=0.883;

Error Rate(HSSA-SVM)=5/60=0.083;

Accuracy(HSSA-SVM)=55/60=0.917;

Table V presents the Error Rate and Accuracy values of SVM, NN, SSA-SVM, and HSSA-SVM algorithms. The sum of Error Rate and Accuracy equals 1.

TABLE V
ERROR RATE AND ACCURACY VALUE

	SVM	NN	SSA-SVM	HSSA-SVM
Error Rate	0.2	0.167	0.117	0.083
Accuracy	0.8	0.833	0.883	0.917

Fig. 5 depicts the accuracy line graph for the SVM, NN, SSA-SVM, and HSSA-SVM algorithms. It is evident from the graph that HSSA-SVM achieves the highest accuracy at 91.7%. This superior performance can be attributed to the utilization of a hybrid improved HSSA algorithm, which efficiently identifies the optimal portfolio values for BoxConstraint and KernelScale.

The study further compared the fitness function (Error Rate) changes of SSA-SVM and HSSA-SVM and found that HSSA-SVM can converge quickly, and the final fitness function value is relatively low. Fig. 6 illustrates the fitness value of both SSA-SVM and HSSA-SVM models. The prediction model HSSA-SVM not only converges more rapidly but also exhibits lower Error Rate alongside higher accuracy level. This improvement stems from employing a hybrid enhanced HSSA algorithm to determine the optimal portfolio of BoxConstraint and KernelScale which affecting SVM classification effect globally. This also means

HSSA-SVM has higher work efficiency.

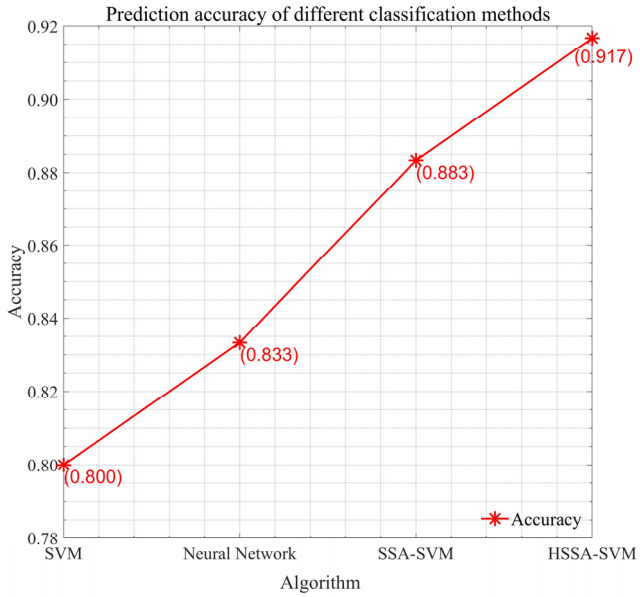


Fig. 5. Accuracy of performance prediction

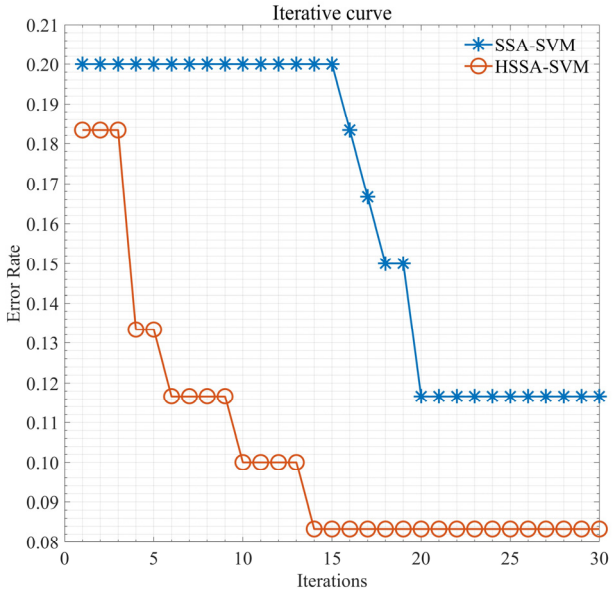


Fig. 6. Iterative curve chart

The number of negative samples (-1) in student performance prediction is significantly lower than the number of positive samples (+1), indicating an imbalance between positive and negative samples. The Accuracy cannot fully reflect the performance of the classification model. In addition, teachers are more concerned about the predictive performance of negative samples (failing students) in the teaching process. Predicting the grades of students who do not meet the standards in advance can help develop teaching strategies, improve teaching methods, and ultimately enhance teaching effectiveness.

Specificity can measure the classifier's ability to recognize negative samples, that is, the proportion of correctly classified negative samples among all negative samples.

Table VI provides a detailed calculation process and compares the Sensitivity and Specificity value of four algorithms. Here, Sensitivity refers to the recall rate of positive samples, while Specificity refers to the recall rate of negative samples.

TABLE VI
SENSITIVE AND SPECIFICITY VALUE

	SVM	NN	SSA-SVM	HSSA-SVM
Sensitivity	0.978 (45/46)	0.935 （43/46）	0.978 (45/46)	0.978 (45/46)
Specificity	0.214 (3/14)	0.5000 (7/14)	0.571 (8/14)	0.714 (10/14)

The Fig. 7 is the line graph of the Sensitivity and Specification of SVM, NN, SSA-SVM, and HSSA-SVM algorithms. From the figure, it can be seen that the Sensitivity of NN is the lowest, while the other three algorithms are the same and all high, reaching 97.8%. HSSA-SVM has not improved Sensitivity value. HSSA-SVM has the highest specificity among the four algorithms, which is 3.5 times that of SVM and 25% higher than SSA-SVM. HSSA-SVM mainly improves the classification accuracy of negative samples. Because the positive sample recognition rate of SVM algorithm is already very high.

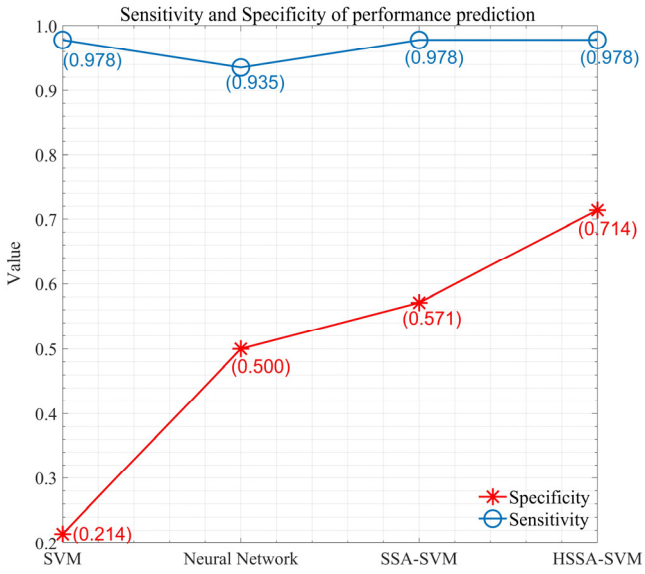


Fig. 7. Sensitivity and Specificity of performance prediction

Table VII displays predictive outcomes for SVM, NN, SSA-SVM, and HSSA-SVM algorithms concerning precision, recall, and F1-Score for both positive and negative samples. It can be observed that the Precision of HSSA-SVM stands at 0.918 for positive samples and 0.909 for negative samples, both are the highest values among all listed algorithms. This achievement results from utilizing HSSA algorithm to identify optimal parameters BoxConstraint and KernelScale specifically tailored for enhancing SVM classification performance in both sample categories effectively. The Precision of NN is higher than SVM on positive samples, but lower on negative samples, indicating that NN has a lower accuracy in predicting fewer sample categories when there is an imbalance between positive and negative samples.

TABLE VII
PRECISION, RECALL, AND F1-SCORE VALUE

	SVM	NN	SSA-SVM	HSSA-SVM
Precision (1: pass)	0.804	0.860	0.882	0.918
Precision (-1: fail)	0.750	0.700	0.889	0.909
Recall (1: pass)	0.978	0.935	0.978	0.978
Recall (-1: fail)	0.214	0.5000	0.571	0.714
F1-Score (1: pass)	0.882	0.896	0.928	0.947
F1-Score (-1: fail)	0.333	0.583	0.696	0.800

Fig.8 illustrates the line graph of the positive sample prediction of SVM, NN, SSA-SVM and HSSA-SVM in Precision, Recall, and F1-Score. Fig.9 shows the line graph of the negative sample prediction of these metrics.

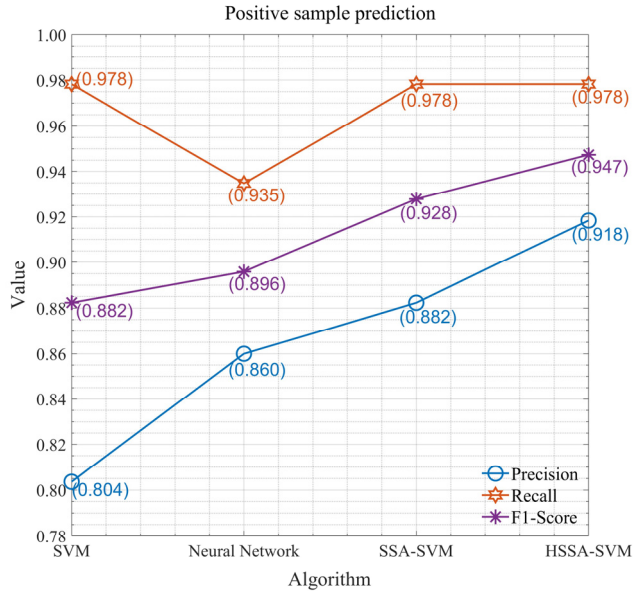


Fig. 8. Precision, Recall, and F1-Score of positive sample prediction

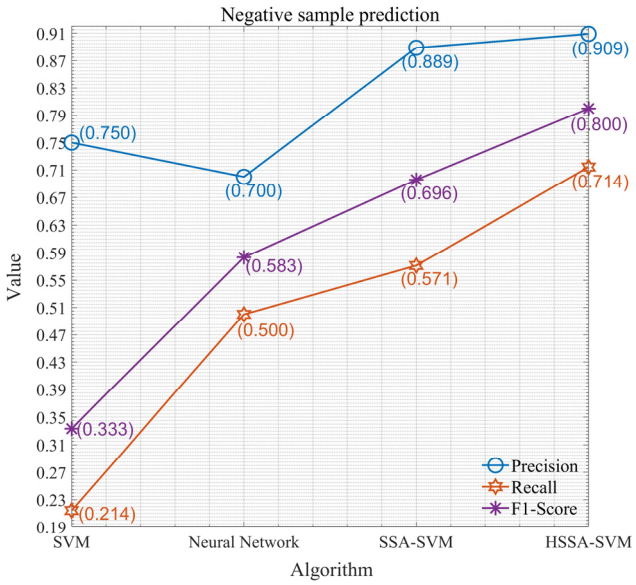


Fig. 9. Precision, Recall, and F1-Score of Negative sample prediction

Recall and Precision are inherently contradictory measures, presenting a trade-off between them. Enhancing Precision typically necessitates a reduction in the number of false positives, which often results in a decrease in true positives and consequently leads to diminished Recall. However, while improving the precision of positive samples, HSSA-SVM maintains consistency with Recall. And HSSA-SVM demonstrates superior performance compared to other algorithms on negative samples, achieving the highest values for both Recall and Precision. Although NN exhibit higher Precision than SVM, it has the lowest Recall. Conversely, SVM outperforms NN in terms of Precision on negative class samples but has lower Recall. In contrast, NN achieves higher Recall than SVM at the expense of its Precision.

The F1-Score takes into account both Precision and Recall of the model. It is particularly suitable for evaluating classification models under conditions of imbalanced categories since it more accurately reflects comprehensive performance across both positive and negative class samples. Its value closer to 1 indicates better predictive capability of the model. Unlike Accuracy, which does not consider FP and FN equally well when addressing issues related to class imbalance or cost sensitivity, F1-Score provides a more precise reflection of model performance under such circumstances. Fig. 8 and 9 illustrate that HSSA-SVM excels in F1-Score for both positive and negative samples, attaining the highest values among all listed algorithms.

Fig. 10 presents the ROC (Receiver Operating Characteristic) curve of SVM and Fig. 11 illustrates the ROC curve for HSSA-SVM. An ROC curve that approaches (0,1) point signifies superior classification prediction outcomes. The Area Under Curve (AUC) quantifies this area beneath the ROC curve and serves primarily as an indicator of a model's generalization capability. The closer the AUC value is to 1, the greater the effectiveness of the model. The AUC for SVM in Fig. 10 is 0.7500, whereas that for HSSA-SVM is 0.8898 in Fig. 11. This indicates a difference of 0.1398, suggesting that HSSA-SVM exhibits superior generalization performance. The ROC curve of HSSA-SVM lies above that of SVM and approaches the (0,1) point more closely, signifying better classification prediction results.

Typically, ISO accuracy lines can be employed to identify optimal points on the ROC curve. The ISO precision line consists of a series of straight lines characterized by a fixed slope but an uncertain intercept. Let us denote the equation for the ISO precision line as follows: $y = ax + b$, where "a" represents the slope and "b" denotes the intercept, specifically, "a" can be defined as:

$$a = NEG / POS$$

Here, NEG refers to the number of negative samples in the dataset while POS signifies positive samples.

In Fig. 10 and Fig. 11, L1, L2, and L3 represent three distinct ISO accuracy lines. L1 serves as initial ISO accuracy line with an intercept set at 0. In Fig. 10, move L1 to the upper left corner (0,1) to obtain the ISO accuracy line L2 that intersects with the SVM ROC curve and is closest to (0,1). L2 intersects with the SVM ROC curve at two points "A" and "B", with intersection point "A" being closest to (0,1), which corresponds to the SVM model with the best classification performance. In Fig. 11, move L1 to the upper left corner to obtain the ISO accuracy line L3 that intersects with the ROC

curve of HSSA-SVM and is closest to (0,1). L3 has only one intersection point “C” with the ROC curve of HSSA-SVM, which corresponds to the best classification performance point of the HSSA-SVM model. Comparing the “C” and “A” of HSSA-SVM and SVM, the “C” distance (0,1) is closer, indicating that HSSA-SVM has better classification performance.

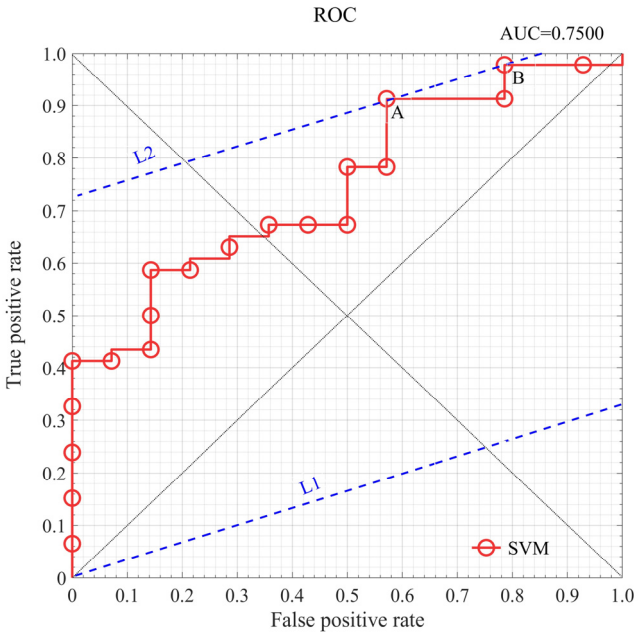


Fig. 10. ROC curve of SVM

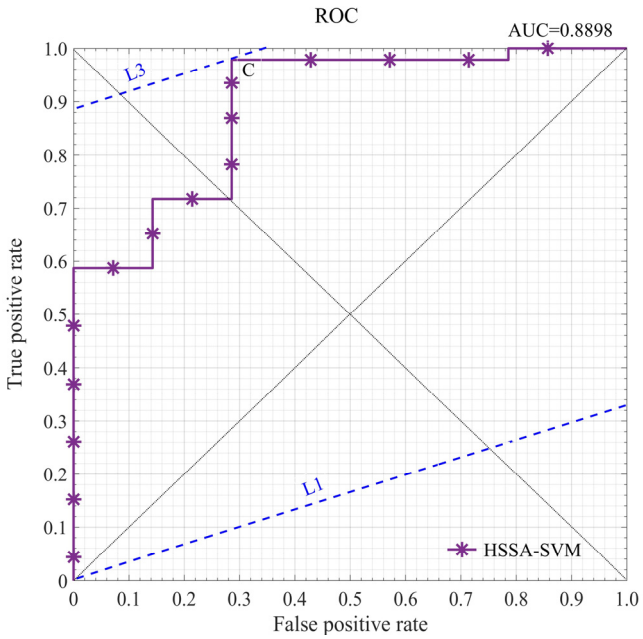


Fig. 11. ROC curve of HSSA-SVM

The DET (Detection Error Tradeoff) curve is one of the indicators for evaluating the performance of pattern recognition classifiers. The lower the curve is towards the origin, the better the prediction performance. As can be seen from Fig. 12 and Fig. 13, HSSA-SVM lies below SVM. HSSA-SVM has a lower EER (Equal Error Rate) than SVM, indicating superior performance.

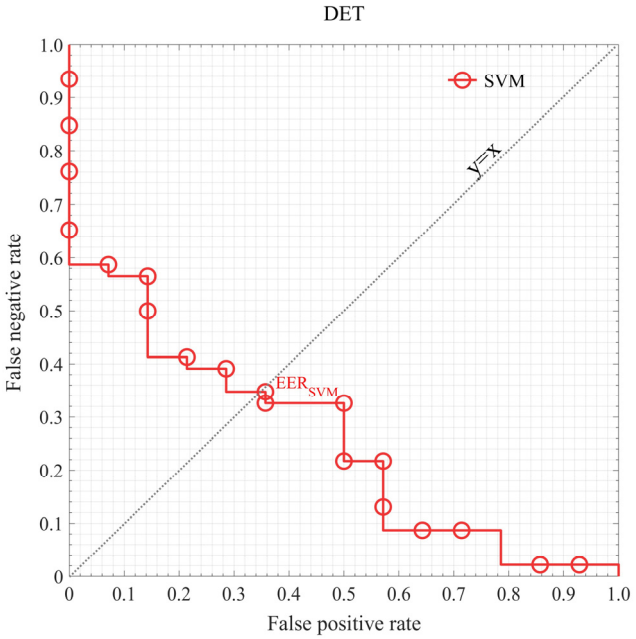


Fig. 12. DEC curve of SVM

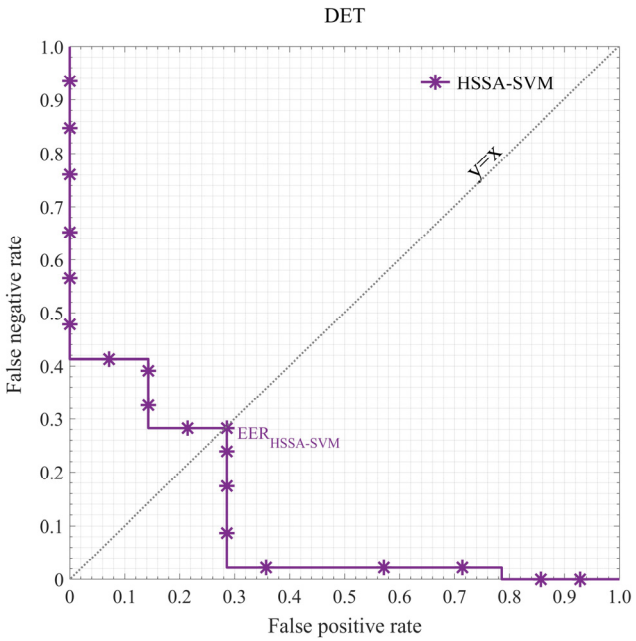


Fig. 13. DEC curve of HSSA-SVM

VI. Conclusion

In order to enhance the prediction of student grades and facilitate timely interventions, as well as assist educators in adjusting their teaching methodologies in a targeted manner, we propose a student grade prediction model algorithm based on support vector machine optimized by the hybrid sparrow search algorithm. Firstly, by incorporating levy flight and t-distribution perturbation strategies into the discovery search phase of the sparrow search algorithm, we expand the search area available for discoverers. Next, adaptive normal distribution random number and poisson distribution random number are introduced to bolster both local development and global exploration capabilities during the warning search phase. Subsequently, the HSSA algorithm is employed to iteratively identify the optimal combination of SVM's penalty factor and kernel function parameters. Finally, we utilize the

established HSSA-SVM student performance prediction model to forecast students' academic achievements. The results demonstrate that the HSSA-SVM algorithm significantly enhances both accuracy and precision in predicting student performance when compared to traditional SVM, Neural Network, and SSA-SVM algorithm; moreover, this model exhibits commendable stability. The HSSA-SVM algorithm offers effective methods and decision-making references for future educational data mining endeavors. In our next steps, we will continue exploring the generalization capability of this algorithm across various datasets.

REFERENCES

[1] L. R. Pelima, Y. Sukmana and Y. Rosmansyah, "Predicting University Student Graduation Using Academic Performance and Machine Learning: A Systematic Literature Review," *IEEE Access*, vol. 12, pp. 23451-23465, 2024.

[2] Chien-Chang. Lin, Anna Y. Q. Huang and Owen H. T. Lu, "Artificial intelligence in intelligent tutoring systems toward sustainable education: a systematic review," *Smart Learning Environments*, vol. 10, pp. 41, 2023.

[3] J. H. Lee, M. Kim, D. Kim and Joon-Min. Gil, "Evaluation of Predictive Models for Early Identificationof Dropout Students," *Journal of Information Processing Systems*, vol. 17, no. 3, pp. 630-644, 2021.

[4] Shyam R. Sihare, "Student Dropout Analysis in Higher Education and Retention by Artificial Intelligence and Machine Learning," *SN Computer Science*, vol. 5, pp. 202, 2024.

[5] P. C. Jiao, F. OY, Q. Y. Zhang and Amir H. Alavi, "Artificial intelligence-enabled prediction model of student academic performance in online engineering education," *Artificial Intelligence Review*, vol. 55, pp. 6321-6344, 2022.

[6] J. Qu, "Research on graduate employment prediction based on LSSVM integrated model," *Journal of Yunnan Minzu University*, vol. 32, no. 3, pp. 382-387, 2023.

[7] L. Al-Alawi, J. Al Shaqsi, A. Tarhini and Adil S. Al-Busaidi, "Using machine learning to predict factors affecting academic performance: the case of college students on academic probation," *Education and Information Technologies*, vol. 28, pp. 12407-12432, 2023.

[8] X. Han, W. Z. Chen and C. J. Zhou, "Musical Genre Classification Based on Deep Residual Auto-Encoder and Support Vector Machine," *Journal of Information Processing Systems*, vol. 20, no. 1, pp. 13-23, 2024.

[9] Z. Quan and L. X. Pu, "An improved accurate classification method for online education resources based on support vector machine (SVM): Algorithm and experiment," *Education and Information Technologies*, vol. 28, pp. 8097-8111, 2023.

[10] P. Lv, W. B. Yu, X. Wang and C. L. Ji, "Students' Performances Prediction and Teaching Cogitation Based on Machine Learning," *Computer Technology and Development*, vol. 29, no. 4, pp. 200-203, 2019.

[11] N. Wang, J. S. Li, Z. Y. Pan and M. H. Yao, "Research on Early Warning of Degree Based on Support Vector Machine," *Journal of Jilin University*, vol. 41, no. 5, pp. 903-907, 2023.

[12] L. Zhang, X. N. Lu, C. L. Lu, B. J. Wang and F. C. Li, "National Matriculation Test Prediction based on Support Vector Machines," *Journal of University of Science and Technology of China*, vol. 47, no. 1, pp. 1-9, 2017.

[13] V. Blanco, A. Japón and J. Puerto, "Multiclass optimal classification trees with SVM-splits," *Machine Learning*, vol. 112, pp. 4905-4928, 2023.

[14] H. TH. Duong, L. TM. Tran, H. Q. To and K. Van Nguyen, "Academic performance warning system based on data driven for higher education," *Neural Computing and Applications*, vol. 35, pp. 5819-5837, 2023.

[15] J. Ding and Y. Su, "A teaching management system for physical education in colleges and universities using the theory of multiple intelligences and SVM," *Soft Computing*, vol. 28, pp. 685-701, 2024.

[16] X. Y. Zhang, K. Q. Zhou, P. C. Li, Y. H. Xiang, A. M. Zain and A. Sarkheyli-Hägele, "An Improved Chaos Sparrow Search Optimization Algorithm Using Adaptive Weight Modification and Hybrid Strategies," *IEEE Access*, vol. 10, pp. 96159-96179, 2022.

[17] Y. L. Fan, Z. Q. Zheng and W. Zheng, "Prediction of College Students' Athletic Performance Based on Improved SSA-LSSVM Model," *Computer Simulation*, vol. 40, no. 1, pp. 283-289, 2023.

[18] X. L. Jin and Y. R. Zhuo, "Prediction of college entrance examination results based on GA-SVM," *Journal of Henan University of Engineering*, vol. 32, no. 2, pp. 62-65, 2020.

[19] N. Zhao, Z. X. Duan and H. Wang, "Math scores prediction model based on improved support vector machine," *Journal of Baicheng Normal University*, vol. 37, no. 2, pp. 21-27, 2023.

[20] H. W. Guo and H. J. Yang, "Sports performance prediction model based on PSO-SVM," *Electronic Measurement Technology*, vol. 43, no. 17, pp. 87-91, 2020.

[21] G. H. Zhang and X. Zu, "Research on Student Performance Prediction Based on Sparrow Search Algorithm and SVM," *Journal of Chizhou University*, vol. 37, no. 3, pp. 11-15, 2023.