Implementing high accuracy and model reliability in credit scoring using improved EWOA-BP algorithm

Yanqin Fan[1*] , Haiwu Huang[1], Guolin Wu[1]

[1] *School of Science, Guilin University of Aerospace Technology, Guilin, 541004, Guangxi, China.*
**Corresponding author: Yanqin Fan (E-mail: yqfan@guat.edu.cn).*

Abstract

Credit assessment is a key problem in the field of finance, which can predict whether a user has the risk of delinquency, thereby reducing the loss of bad debts. BP neural network has been widely used in credit evaluation because of its excellent ability of data learning and induction. However, it also has disadvantages, such as slow convergence and being susceptible to local outliers. Swarm-based intelligence algorithms offer advantages such as simplicity of use, rapid convergence, and powerful global optimization capabilities, making them effective optimization algorithms. In order to improve the convergence speed and prediction accuracy of BP neural network in credit evaluation, this paper constructs EWOA-BP model based on improved whale algorithm. Firstly, we propose a multi-strategy Enhanced Whale Algorithm (EWOA) by introducing a new non-linear decrease approach based on cosine function, a unique exploration technique that employs leader-based adaptive tangent travel, and a novel exploitation strategy, which can effectively balance the exploration and exploitation, reduce the risk of local optima trapping, and accelerate the convergence speed while ensuring the accuracy. Secondly, the Enhanced whale algorithm is used to optimize the neural network and construct the EWOA-BP model. Finally, the Enhanced Whale Algorithm (EWOA) is verified and analyzed by using 23 classical benchmark functions, and the performance is excellent. The EWOA-BP model is validated on three credit assessment datasets, and by comparing it with 10 contemporary algorithms, the results show that the EWOA-BP model obtains better performance in personal credit assessment, and comprehensively concludes that EWOA-BP algorithm is effective and more competitive.

Keywords: whale optimization algorithm; BP Neural Network; Credit Scoring; EWOA-BP; Metaheuristics; Model evaluation.

INTRODUCTION

With the increasing scale of credit in China, various bank merchant credit problems have arisen. Therefore, it is crucial to construct an efficient and stable bank merchant credit assessment model. The personal credit assessment problem under study is essentially a binary classification problem based on unbalanced loan data. Personal credit assessment is essentially to construct a credit assessment model using the borrower's loan application information and the default situation after the loan, and then use the credit assessment model to review the loan qualification of the later applicants and issue loans to customers who meet the requirements. In practice, in order for borrowers and lenders to understand the loan rules, it is necessary for the credit assessment model to combine strong interpretability and high classification accuracy.

Foreign research on credit assessment started earlier, and the initial credit assessment model is that banks and other financial institutions first analyze the borrower's provision of various types of loan information, and then use the experience of experts to assess the customer's default risk. 1941 Durand [1] first relied on the borrower's consumer credit behavioral characteristics to establish a mathematical model of personal consumer credit, and from then on opened up the quantitative analysis of personal credit risk control; 1956 Bill Fair and Earl Isaac worked together to develop the personal credit scoring system FICO [2], and from then on opened up the field of personal credit scoring research, and was widely used In 1980, Wiginton [3] established personal credit scoring model through logistic regression for the first time, and concluded that the logistic regression model has a better fitting effect for the assessment of personal risk. Makowski (1985) was among the pioneers in using machine learning techniques to evaluate personal credit risk in the field, and he constructed a decision tree assessment model based on user credit data to classify customers with good results [4].Cover and Hart (1986) proposed the

K-nearest neighbor method, which has been widely used in binary classification and credit assessment problems [5].

Machine learning techniques have become extensively employed in credit assessment in recent times, including decision trees[6-9], random forest algorithms[10-12], logistic regression[13-15], support vector machines[16-17], K-nearest neighbor algorithms[18-19], K-mean algorithms[20-21], AdaBoost algorithms[22-23], neural networks[24-26], Bayesian networks[27-28], and various integrated algorithms. In particular, BP neural networks have been widely used in credit assessment, and various researchers have improved their algorithms based on BP neural networks [29-32], among which Zhengwei Ma et al. [29] trained BP neural networks by adopting the LM algorithm, scaled conjugate gradient and Bayesian regularization , and applied them to credit risk assessment.

As artificial intelligence continues to advance, swarm intelligence optimization algorithms are also widely used in personal credit assessment. The multitude intelligence optimized method is a widely used technique in artificial intelligence. Its main concept is to simulate the optimized processes of different animals or groups of entities in order to identify the best solution to an optimization issue. This type of algorithm transforms the engineering optimization problem into a functional optimization problem, establishes the objective function, and seeks the optimal solution of the objective function. In the iterative process, the swarm intelligence optimization algorithm will constantly use the individual optimal value and the group optimal value to search for the optimal search, and complete the interaction between the individual information and the group information. The swarm-based intellect-optimizing algorithm is a prevalent method in intelligent computation. Its fundamental concept involves replicating an optimized process by different animals or groups of entities to determine the ideal answer to an optimization issue.

Swarm intelligent optimization algorithm has the following characteristics: progressive search for the optimal, through repeated iterations to gradually get the optimal results; reflect the "survival of the fittest, the poorer die out" law of natural selection, with the ability to dynamically adapt to the environment. When using guided stochastic search, the fitness coefficient guides the search path. Additionally, using parallel search lessens the risk of becoming stuck in a local optimum solution. It can achieve more benefits at a lower cost. Several domains extensively use swarm intelligent optimization techniques, such as power system, transportation field, data transmission, regional coordination optimization, machine vision, e-commerce, finance and so on. For example, in the field of finance, they can also be used for data analysis and investment decisions. Particle Swarm Optimization (PSO) [33–34], Ant Colony Optimization (ACO) [35–36], Genetic Algorithm (GA) [37–38], Artificial Bee Colony Optimization Algorithm (ABC) [39], Differential Evolutionary Algorithm (DE) [40–41], Gravitational Search Algorithm (GSA) [42–43], Firefly Algorithm (FA)[44], Cuckoo Optimization Algorithm (COA) [45], Grey Wolf Optimization Algorithm (GWO) [46], etc. In 2016, Mirjalili [47], an Australian scholar, proposed the Whale Optimization Algorithm (WOA) which is a simulation algorithm based on the feeding behavior of whales, and has the advantages of simple operation, stronger ability to jump out of the local optimum, and fewer tuning parameters, etc. However, there are some disadvantages of the algorithm, for example, it may be trapped in the local optimal solution, which leads to unsatisfactory search results. In order to overcome these drawbacks, researchers have improved the algorithm, such as combining adaptive weights and simulated annealing to improve the search performance and convergence speed of the algorithm. The algorithm is often combined with other algorithms to solve optimization problems in computational problems. Yutong Zhang et al. [48] introduced a method known as the multifaceted learning whale optimization approach (MLWOA). The MLWOA employs exploratory processes and a learning approach to global whale population exploration. Ziying Liang et al. [49] introduced a more advanced version of the whale optimization algorithm called the Dynamic Gain-Sharing Whale Optimization Algorithm (DGSWOA). According to the experimental findings, this approach is generally comparable when it comes to both the quality of the solution and the speed of convergence. While the whale optimization method finds extensive applications in artificial intelligence, healthcare, economic administration, and transportation, its use in credit evaluation is rather limited.

In order to avoid the early maturity phenomenon of WOA, we propose a new optimizer called EWOA based on WOA. And then, we construct a credit assessment classification model based on EWOA combined with the BP neural network algorithm (EWOA-BP) by analyzing the credit assessment indexes. The primary contributions of this work may be succinctly described as listed below:

(1) Introducing a novel convergence approach that uses a cosine function to balance both discovery and extraction.

(2) Introducing a leader-based adaptive tangent flight strategy to enhance the search efficiency and solution quality.

(3) A dimensional centroid-based random opposition learning strategy, is introduced to improve the algorithm's global search capabilities and population diversity , to prevent local optima and premature convergence.

(4) To assess the effectiveness of EWOA, we conduct a comparative analysis of various widely recognized algorithms using a set of 23 test procedures that vary in complexity.

(5) The aim is to develop a credit assessment classification model using the EWOA-BP algorithm, which combines the EWOA method with the BP neural network. We confirm the effectiveness of the EWOA-BP algorithm by testing it on three distinct credit score datasets.

We organize the paper as follows: The second part provides a concise description of both the WOA and BP algorithms. Section 3 provides a comprehensive explanation of the proposed EWOA-BP algorithm. In Section 4, we conduct experiments using the EWOA algorithm and thoroughly examine its performance. We next utilize the credit assessment dataset to validate the EWOA-BP model. Section 5 ultimately closes and provides a preview of the complete work.

IMPROVED WHALE OPTIMIZATION ALGORITHM(EWOA)

This part provides a concise overview of the mathematical framework used in the Whale Optimization Algorithm (WOA) and the Extended Whale Optimization Method (EWOA).

Whale optimization algorithm (WOA) [47]

The Whale Optimization Method is a meta-heuristic intelligent swarm optimization method that has the advantages of fewer adjustable variables, straightforward operation, and simple comprehensibility. The method is based on the humpback whale's "spiral bubble net" approach. The algorithm draws inspiration from the whales' "spiral bubble net" technique. It has three primary phases: surrounding the target, assaulting the inflated net, and looking to find the prey. Humpback whales demonstrate their bubble net feeding habit in Figure 1.

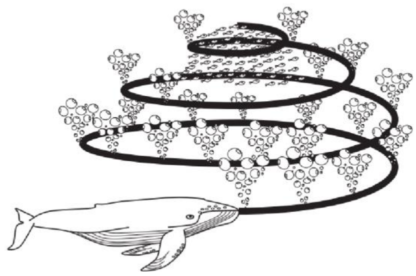


Figure 1 Identifying and attacking prey

Encircling prey

When whales hunt, it is critical to first identify the target's position before surrounding it. Therefore, the whale algorithm functions under the assumption that the optimal target within the current population is the desired prey, and it adjusts the positions of the remaining individual whales in the population according to the position of this target prey. The mathematical model is defined in the following manner:

$$D = \left| C \cdot X^*(t) - X \right| \tag{1}$$

$$X(t+1) = X^*(t) - A \cdot D \tag{2}$$

Where t is the current number of iterations, X^* denotes the position of the optimal whale in the current whale group, X denotes the position of the current whale, $|\cdot|$ is the absolute value. The following algebraic models describe the coefficient vectors A and C:

$$A = 2a \cdot r - a \tag{3}$$

$$C = 2 \cdot r \tag{4}$$

$$a = 2\left(1 - \frac{t}{T_{\max}}\right) \tag{5}$$

Where a is linearly reduced in the interval (0, 2) in the iterative process, and r is a random number between [0,1], and $T_{\max}$ is the maximum number of iterations.

Bubble-net attacking method (local search)

The whale algorithm employs two ways to accurately model the properties of huge whale bubble-net predation:

1) Shrinkage encirclement mechanism

The method involves reducing the value of parameter "a" in equation (5). This linear drop of a from 2 to 0 subsequently lowers the fluctuation range of variable A. Specifically, the range of A becomes [-a, a]. When the value of A fluctuates between -1 and 1, it necessitates the presence of whales that have updated their locations, now situated between their initial position and the prey. In other words, all the whales in the population migrate closer to their prey in order to create a barrier.

2) Spiral position updating

The procedure initially computes the distance that exists between the whale and its prey. Next, to mimic the humpback whale's spiral movement, we establish an inverse spiral equation between the two entities. The equation is mathematically illustrated as follows:

$$X(t+1) = D' \cdot e^{bl} \cdot \cos(2\pi l) + X^*(t) \tag{6}$$

Where $D' = \left|X^*(t) - X(t)\right|$ denotes the distance between the optimal whale individual and the current whale individual in the t th iteration, b is used to define the constant to limit the logarithmic spiral and l is a random number between $[-1,1]$.

Since the whale in the predation process, the above two methods at the same time, that is, shrinking the envelope, at the same time to follow the spiral path to approach the prey, in order to simulate this process, the whale optimization algorithm assumes that the probability of the above two methods being selected in the predation process are 0.5, and its mathematical model is:

$$X(t+1) = \begin{cases} X^*(t) - A \cdot D & p < 0.5 \\ D' \cdot e^{bl} \cdot \cos(2\pi l) + X^*(t) & p \geq 0.5 \end{cases} \tag{7}$$

Let p be a randomly generated integer within the range of 0 to 1. Following the inflated netting assault, humpback whales commence a haphazard quest for prey.

Search for prey (global search)

During the iterative phase, we can harness the search for prey to modify vector A. Humpback whales effectively navigate the solution space by exploring it randomly, taking into account other whales' locations. Therefore, the global exploration phase employs A to adjust the whale's position, ensuring it moves away from the present person if the random value exceeds or falls below one. Unlike the localized exploitation phase, the global exploration phase involves upgrading the locations of individual whales using randomly picked whale individuals rather than the best whale individuals discovered thus far. The primary objective of this mechanism is to facilitate exploration. Therefore, the WOA algorithm executes a global exploration operation when $|A| > 1$. In the prey acquisition stage, whales obtain information about prey locations through collective collaboration, as the whereabouts of the prey remain unknown to the whole whale community. The whales use stochastic individual locations across the population as navigational targets to find food, and the mathematical model outlines this process as follows:

$$D = \left| C \cdot X_{rand} - X \right| \tag{8}$$

$$X(t+1) = X_{rand} - A \cdot D \tag{9}$$

Where , X_{rand} represents the spatial coordinates of randomly picked single whales within the whale group.

The proposed EWOA

To address the limitations of WOA, such as its poor accuracy and susceptibility to local optima, we suggest four techniques to improve the efficiency of the initial algorithm.

New adaptive nonlinear convergence strategy based on cosine function

Swarm intelligence systems need to achieve a harmonious equilibrium between conducting a comprehensive search throughout the whole problem space and focusing on exploring smaller, localized areas. The original WOA algorithm suffers from the linearity of the parameters a in the reduced algorithm, which hinders its ability to effectively handle complex nonlinear search issues throughout the optimization process. Consequently, achieving a satisfactory trade-off between searching and exploiting becomes challenging. In order to address this constraint, we suggest an adaptable nonlinear convergence approach that relies on the cosine function. This method takes advantage of the periodic and fluctuating properties of the cosine function to dynamically modify the search range and parameters. As a result, it achieves a harmonious balance between exploration and exploitation. This method enhances its search capability by avoiding premature convergence to local optima. Additionally, it speeds up the convergence procedure and reduces the number of incorrect searches, hence increasing the system's resilience to complicated nonlinear situations. The following formula illustrates the mathematics:

$$a = 2 \times \cos\left[\frac{\pi}{2} \times \left(\frac{t}{T}\right)^3\right] \times e^{-0.05t} \tag{10}$$

$$A = 2a \cdot r - a \tag{11}$$

where, t denotes the present count of iterations, whereas T indicates the highest count of iterations. r represents a random number that falls within the range of 0 to 1, and A refers to the vector of coefficients.

New leader-based adaptive tangent flight strategy

During the development phase, considering the fact that the WOA algorithm is susceptible to becoming stuck at the local optimal point, we propose a leader-based adaptive tangent flight strategy that enhances the search efficiency and solution quality by mimicking the flight patterns of birds in nature. This strategy enables individuals within the algorithm to flexibly adjust their positions based on the leader's position, thereby swiftly approaching the optimal solution. Moreover, its adaptive nature permits the algorithm to dynamically adjust the search strategy in response to the search progress, thereby avoiding local optima. Furthermore, this technique bolsters communication within the algorithm's individuals, facilitating faster convergence and superior solutions when addressing complex optimization problems. The theoretical framework is outlined below.

$$X(t+1) = D' \cdot e^{bl} \cdot \cos(2\pi l) + St \times \tan\left(rand \times \frac{\pi}{2}\right) \times X^*(t) \tag{12}$$

$$St = 0.525 - 0.5 \times \left(\frac{t}{T}\right) \tag{13}$$

Where, $D' = \left| X^*(t) - X(t) \right|$ represents the distance between the humpback whale and the prey, The value of b determines the shape of the exponential spiral, l can be any number within the range of -1 to 1, $X^*(t)$ presents the best position obtained so far, and $rand$ denotes a random number between [0,1]. St represents the adaptive tangent flight step size.

New dimension-by-dimension centroid-based random inverse learning strategy

To address the lack of population diversity in the algorithm's later iterations, We introduce a unique learning strategy known as the dimensional centroid-based random opposition method. This strategy greatly improves the algorithm's ability to explore the entire search space and maintain a diverse population by independently updating each individual through random learning. This strategy effectively prevents the algorithm from becoming stuck in local optima and converging prematurely. This strategy employs centroid calculations to guide the search direction, thereby accelerating the algorithm's convergence process, enabling more effective exploration of complex search spaces for superior solutions. The mathematical model is provided below.

$$X(t+1) = X(t) + rand \times \left(2 \times mean\left(X(t)\right) - X(t)\right) \tag{14}$$

Where, $mean\left(X(t)\right) = \frac{1}{N}\sum_{i=1}^{N} X_i(t)$, $X_i(t)$ indicates the position of the ith individual after the tth iteration.

The smoothing technology

The primary function of smoothing techniques in algorithms is to refine search behaviors and enhance solution quality, particularly when addressing optimization problems characterized by discontinuous or complex objective functions. This strategy enhances the algorithm's ability to navigate the solution space and prevents it from becoming stuck in a suboptimal solution. As a result, it enhances the algorithm's ability to search globally and speeds up convergence. It improves its ability to handle optimization challenges, allowing it to find better solutions. Here is the mathematical framework.

$$X(t+1) = 0.5 \times \left(X_{j-1}(t) + X_{j+1}(t)\right) \tag{15}$$

where $X_{j-1}(t)$ represents the $(j-1)^{th}$ column of the individual, and $X_{j+1}(t)$ indicates the $(j+1)^{th}$ column.

OPTIMIZATION OF BP NEURAL NETWORK PREDICTION MODEL BASED ON EWOA(EWOA-BP)

BP Neural Network

The BP neural network is a type of multi-layer neural network with a feed-forward function that is characterised by the forward propagation of signals and the backward propagation of errors. The BP neural network consists of two primary stages. The initial stage involves the forward transmission of the message, where the data originates from the input layer, moves through the hidden layer, and ultimately reaches the resultant layer. The second stage is the reverse transmission of the error, where the results layer's result is compared with the expected result. The error then propagates from the output layer to the hidden layer and finally to the input layer. We adjust the weights and bias of the concealed layer to the result layer, and then we adjust the bias and weights of the input layer to the hidden layer. The BP neural network possesses the capacity for autonomous learning, translation of nonlinear functions, and high resilience. Figure 2 illustrates the configuration of a three-layer backpropagation neural network.

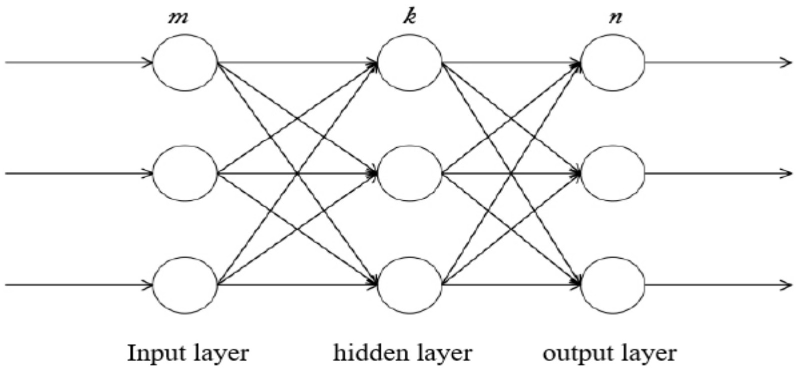


Figure 2 Single hidden layer BP neural network topology

The process of supervised BP neural network algorithm is as below:

1) Begin by initializing the network. Each link weight is assigned an arbitrary number between (-1, 1). Set the error function as *e*. The computational accuracy value is ε , and the maximum learning times are *M*.

2) In an arbitrary manner, select the *k*th input sample and the associated anticipated result report phrase.

3) Compute the input and output values for each neuron in the hidden layer.

4) Compute the partial derivative of the error function, denoted as $\delta_o(k)$, relative to each neuron in the output layer. This calculation should be based on the intended and actual outputs of the neural network.

5) Utilize the activation value $\delta_o(k)$ of every neuron in the output layer and the output value of each neuron in the hidden layer to adjust the link weights $w_{ho}(k)$ accordingly.

6) Fix the relationship between the weights via $\delta_h(k)$ of every neuron within the hidden layer and the input signal of each neuron in the input layer.

7) Determine the total error.

8) Determine if the network's error falls within the range that is appropriate. Terminate the procedure when the mistake attains the predetermined accuracy or when the total number of training iterations exceeds the permitted amount. Alternatively, choose the following set of learning samples and the associated desired outputs, and go back to step 3 to begin the following phase of learning.

Improved EWOA optimization BP neural network (EWOA-BP)

The EWOA-BP model uses the enhanced EWOA algorithm to improve the BP neural network's initial weights and thresholds. It selects the ideal starting weights and thresholds by using the training error of the BP neural network as the individual fitness value. The precise procedure is as depicted in Figure 3.

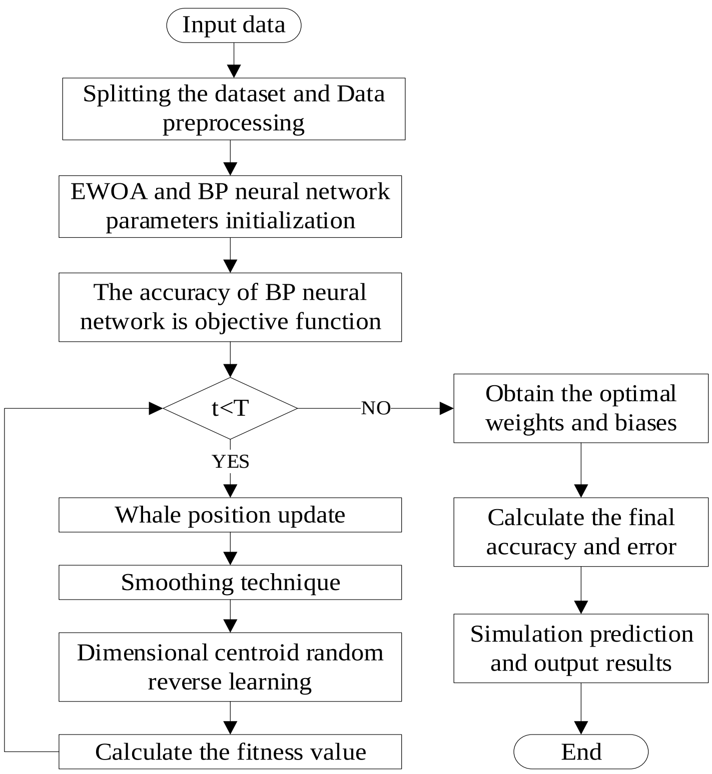


Figure 3 The flow chart of EWOA-BP algorithm

EXPERIMENTAL RESULTS AND ANALYSIS

In this section, we initially performed a test to evaluate the efficiency of the EWOA algorithm; secondly, the constructed EWOA-BP prediction model was compared and validated with other machine learning models and swarm intelligence optimization algorithms on three different credit assessment datasets; and lastly, the model assessment metrics, such as the prediction accuracy rate, ROC curves and AUC values, and confusion matrices were used to analyze and summarize the performance of EWOA-BP in credit assessment. We performed the studies using a notebook computer with a 2.10 GHz CPU and 16 GB of RAM, using the MATLAB 2019b platform.

Improved Whale Algorithm(EWOA) Effectiveness Test

Benchmark test functions

The benchmarking function is critical for algorithm development because it enables the assessment and comparison of different optimization strategies. In order to verify the performance of the improved whale algorithm (EWOA), in this study, we selected 23 international standard test functions for multidimensional testing and set the dimensionality to 30, 50, and 100, respectively. The 23 international standard test functions [50], which contain 7 single-peak functions (F1-F7), 6 multiple-peak functions (F8-F13), and 10 fixed-dimension functions (F14-F23). Among them, the single-peak function can examine the convergence accuracy and effectiveness of the algorithm; while the multi-peak function can effectively test the global search ability of the algorithm. It is worth mentioning that the function F5 is a typical non-convex function, which is also known as valley function or banana function due to the very flat bottom of the function, and it is an important tool to measure the merits of an algorithm. The mathematical expressions for these functions can be found in Table 1.

Table 1 Testing functions

Function	Functional expression	Dimensions	Range	$F_{\min}$
F1	$f_1(x)=\sum_{i=1}^{d} x_i^2$	30/50/100	[-100,100]	0
F2	$f_2(x)=\sum_{i=1}^{d}\lvert x_i\rvert+\prod_{i=1}^{n}\lvert x_i\rvert$	30/50/100	[-10,10]	0
F3	$f_3(x)=\sum_{i=1}^{d}\left(\sum_{j-1}^{i} x_j\right)^2$	30/50/100	[-100,100]	0
F4	$f_4(x)=\max_i\{\lvert x_i\rvert, 1\le i\le n\}$	30/50/100	[-100,100]	0
F5	$f_5(x)=\sum_{i=1}^{d-1}\left(100\left(x_{i+1}-x_i^2\right)+(x_i-1)^2\right)$	30/50/100	[-30,30]	0
F6	$f_6(x)=\sum_{i=1}^{d}([x_i+0.5])^2$	30/50/100	[-100,100]	0
F7	$f_7(x)=\sum_{i=1}^{d} i\, x_i^4+\text{ random }[0,1)$	30/50/100	[-1.28,1.28]	0
F8	$f_8(x)=\sum_{i=1}^{n}-x_i\sin\left(\sqrt{x_i}\right)$	30/50/100	[-500,500]	$-418.9\times Dim$
F9	$f_9(x)=\sum_{i=1}^{d}\left(x_i^2-10\cos(2\pi x_i)+10\right)$	30/50/100	[-5.12,5.12]	0
F10	$f_{10}(x)=-20\exp\left(-0.2\sqrt{\frac{1}{n}\sum_{i=1}^{n}x_i^2}\right)-\exp\left(\frac{1}{n}\sum_{i=1}^{n}\cos(2\pi x_i)\right)+20+e$	30/50/100	[-32,32]	0
F11	$f_{11}(x)=\frac{1}{4000}\left(\sum_{i=1}^{n}(x^2 i)\right)-\left(\prod_{i=1}^{n}\cos\left(\frac{xi}{\sqrt{i}}\right)\right)+1$	30/50/100	[-	0

			600,600]	
F12	$f_{12}(x) = \frac{\pi}{n}\{10\sin(\pi y_1) + \sum_{i=1}^{n-1}(y_i - 1)^2 [1 + 10\sin^2(\pi y_{i+1}] + (y_n + 1)^2\} + \sum_{i=1}^{n} u(x_i, 10, 100, 4)$ $y_i = 1 + \frac{x_i + 1}{4}; u(x_i, a, k, m) = \begin{cases} k(x_i - a)^m & x_i > a \\ 0 & -a < x_i < a \\ k(-x_i - a)^m & x_i < -a \end{cases}$	30/50/100	[-50,50]	0
F13	$f_{13}(x) = 0.1\{\sin^2(3\pi x_1) + \sum_{i=1}^{n}(x_i - 1)^2 [1 + \sin^2(3\pi x_{i+1})] + (x_n - 1)^2[1 + \sin^2(3\pi x_{i+1})]\} + \sum_{i=1}^{n} u(x_i, 5, 100, 4)$	30/50/100	[-50,50]	0
F14	$f_{14} = \left(\frac{1}{500} + \sum_{j=1}^{25} \frac{1}{j + \sum_{i=1}^{2}(x_i - a_{ij})^6}\right)^{-1}$	2	[-65,65]	1
F15	$f_{15} = \sum_{i=1}^{11}\left[a_i - \frac{x_1(b_i^2 + b_1 x_2)}{b_i^2 + b_1 b_3 + b_4}\right]^2$	4	[−5,5]	0.0003
F16	$f_{16} = 4x_1^2 - 2.1x_1^4 + \frac{1}{3}x_1^6 + x_1 x_2 - 4x_2^2 + 4x_2^4$	2	[−5,5]	−1.0316
F17	$f_{17} = \left(x_2 - \frac{5.1}{4\pi^2}x_1^2 + \frac{5}{\pi}x_1 - 6\right)^2 + 10\left(1 - \frac{1}{8\pi}\right)\cos x_1 + 10$	2	[−5,5]	0.398
F18	$f_{18} = [1 + (x_1 + x_2 + 1)^2(19 - 14x_1 + 3x_1^2 + 6x_1 x_2 + 3x_2^2)] \times [30 + (2x_1 - 3x_2)^2] \times (18 - 32x_1 + 12x_1^2 + 48x_2 - 36x_1 x_2 + 27x_2^2))$	2	[−2,2]	3
F19	$f_{19} = -\sum_{i=1}^{4} c_i \exp\left(-\sum_{j=1}^{3} a_{ij}\left(x_{ij} - p_{ij}\right)^2\right)$	3	[1,3]	−3.80
F20	$f_{20} = -\sum_{i=1}^{4} c_i \exp\left(-\sum_{j=1}^{6} a_{ij}\left(x_{ij} - p_{ij}\right)^2\right)$	6	[0,1]	−3.32
F21	$f_{21} = -\sum_{i=1}^{5}[(X - a_i)(X - a_i)^T + c_i]^{-1}$	4	[0,10]	−10.1513
F22	$f_{22} = -\sum_{i=1}^{7}[(X - a_i)(X - a_i)^T + c_i]^{-1}$	4	[0,10]	−10.4028
F23	$f_{23} = -\sum_{i=1}^{10}[(X - a_i)(X - a_i)^T + c_i]^{-1}$	4	[0,10]	−10.5363

Simulation Results and Analysis

To demonstrate the competitiveness of EWOA, we investigate and compare it with six traditional optimization algorithms (PSO, GA, DE, GWO, WOA and EWOA) using standard benchmark functions. The sizes were set to 30,50 and 100 respectively. We set the algorithm parameters uniformly with a maximum number of iterations of 500 and a population size of 50. We executed each method 30 times separately to measure the mean deviation (Std) and average (Ave) and highlight the best results in bold. The experimental results are shown in Tables 2 and 3, where the mean and average deviation of the six optimization algorithms for the test function in 30 and 100 dimensions are given.

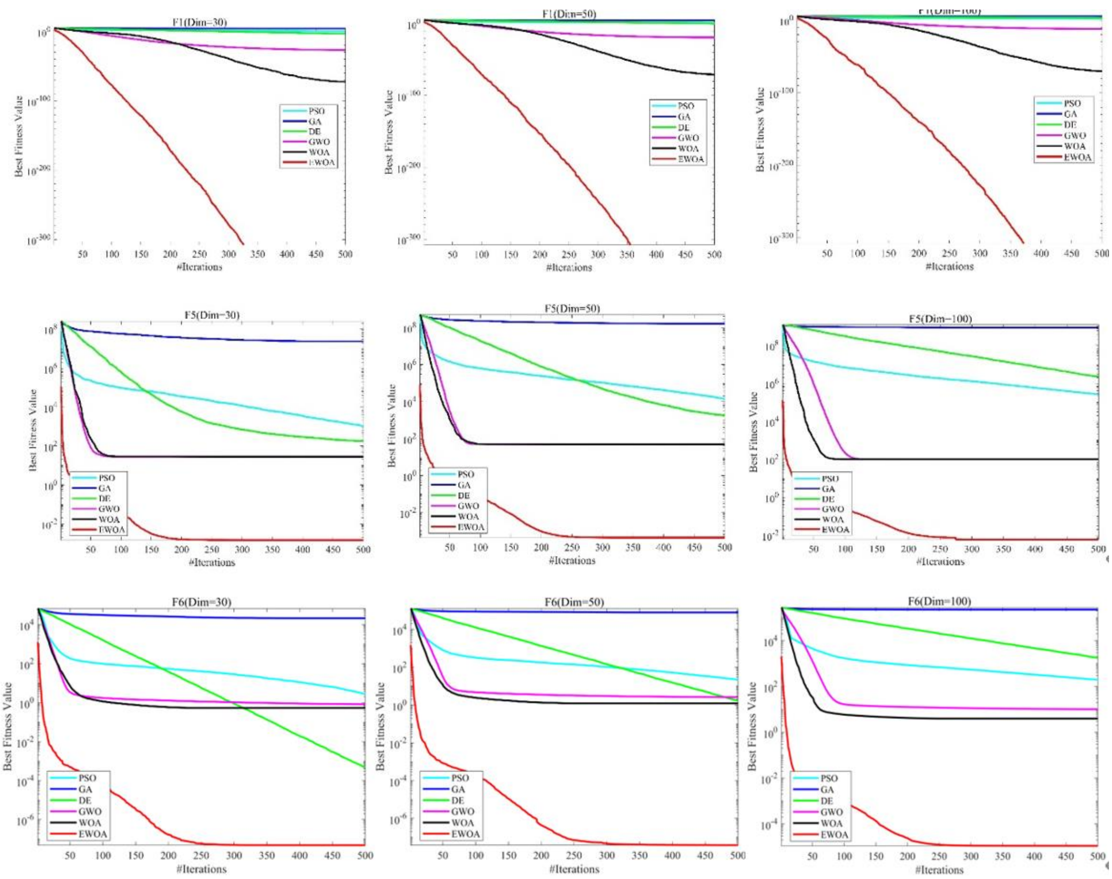
From the experimental data in Table 2, it can be seen that the EWOA algorithm is the optimal among the 6 algorithms. Compared with the other 5 types of algorithms, among the 23 functions tested, the EWOA algorithm is the closest to the theoretical optimum in 20 test functions, has the smallest standard deviation and the highest stability in 18 test functions, and performs the best in both the optimum and stability in 17 test functions. Specifically, among the first 13 multidimensional functions, the EWOA algorithm directly converges to the theoretical minimal value of 0 in 4 functions, including F1, F3, F9, and F11. Compared with the other five algorithms, the EWOA algorithm performs the best in terms of optimal search ability and stability for the 12 multidimensional functions. For function F2, although the WOA algorithm slightly outperforms the EWOA algorithm in finding the theoretical optimum, the EWOA algorithm is also closer to the theoretical optimum and outperforms the WOA algorithm in stability and ability to jump out of the minima. In the last 10 function tests with fixed dimensions, the EWOA algorithm is closest to the theoretical minima on 8 functions, including F15, F16, F17, F18, F19, F21, F22, and F23. Compared to the other five algorithms, the EWOA algorithm has the best optimization ability and stability in the five fixed-dimensional test functions. For the F16 function, the five algorithms PSO, DE, GWO, WOA, and EWOA have the same optimal value, which is very close to the theoretical optimal value of -1.0316, but GWO and WOA perform poorly in terms of stability, and EWOA has the smallest average deviation and the best stability.

From the experimental data in Table 3, it can be seen that the EWOA algorithm is still the best performer among the six algorithms, and compared with the other five algorithms, the EWOA algorithm performs the best in terms of optimal value and stability for the 13 tested functions. Specifically, the EWOA algorithm converges directly to the theoretical minima 0 in four functions F1, F3, F9 and F11.

Meanwhile, the test function F5 serves as an important tool to measure the merits of the algorithm, and it is easy to see that the EWOA algorithm obtains the best mean and standard deviation in both dimensions. The analysis of Tables 2 and 3 shows that with the increase of dimensions, the EWOA algorithm is the best performer among the six algorithms in terms of the optimization seeking ability and stability of the benchmark test function, which reflects the superiority and stability of the EWOA algorithm.

Figure 4 illustrates the convergent curves of various competitive techniques for every dimension. The results of our study suggest that as the number of dimensions rises, the level of difficulty in optimisation also increases. Additionally, other competitive methods tend to become stuck at the optimal location. Nevertheless, the average convergence curve of EWOA has a consistent downward trajectory, indicating a higher probability of attaining ideal values. Overall, the EWOA algorithm exhibits exceptional resilience, precision, and rapid convergence, positioning it as the most formidable algorithm in terms of competition. See Figure 4 for details.

In summary, the EWOA algorithm is the most competitive algorithm with better robustness, accuracy and convergence speed.



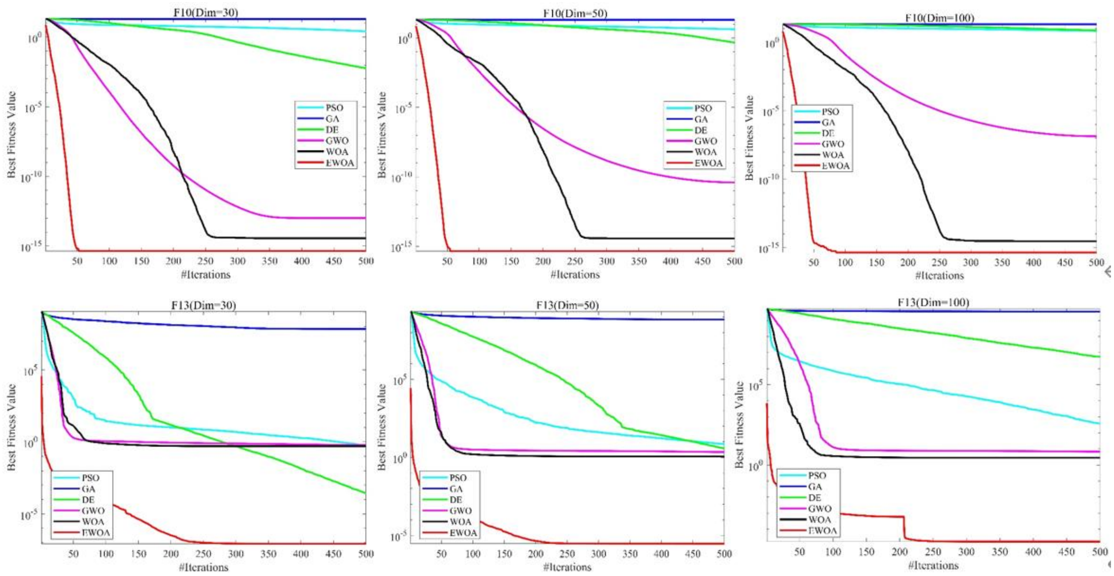


Figure 4 Comparative analysis of the convergence curves of six algorithms on benchmark

Table 2 Comparative analysis of results on classical benchmark functions (Dim = 30)

ID		PSO	GA	DE	GWO	WOA	EWOA
F1	Ave	2.3540E+00	2.1774E+04	5.2203E-04	2.7356E-27	2.0500E-73	**0.0000E+00**
	Std	8.9850E-01	7.3095E+03	1.5782E-04	6.2642E-27	8.8704E-73	**0.0000E+00**
F2	Ave	4.8050E+00	5.7267E+01	2.2446E-03	1.2568E-16	**3.9681E-51**	3.5017E-27
	Std	1.2727E+00	8.0554E+00	4.3058E-04	1.0774E-16	1.4042E-50	**0.0000E+00**
F3	Ave	1.9343E+02	5.3472E+04	3.2401E+04	7.7569E-06	4.3199E+04	**0.0000E+00**
	Std	6.8938E+01	1.8011E+04	5.3745E+03	1.1275E-05	1.3161E+04	**0.0000E+00**
F4	Ave	2.0019E+00	7.2886E+01	1.2733E+01	7.9918E-07	5.6736E+01	**7.5532E-21**
	Std	2.7450E-01	9.0300E+00	1.1769E+00	9.4913E-07	2.7962E+01	**0.0000E+00**
F5	Ave	1.0331E+03	2.3314E+07	1.7903E+02	2.6894E+01	2.8105E+01	**1.4713E-03**
	Std	5.1639E+02	2.0536E+07	4.7066E+01	6.7624E-01	4.6197E-01	**3.6270E-03**
F6	Ave	2.6628E+00	2.1265E+04	4.5606E-04	8.2745E-01	5.2610E-01	**4.8340E-08**
	Std	1.1885E+00	8.4451E+03	1.7984E-04	2.9728E-01	3.2508E-01	**2.6467E-07**
F7	Ave	1.6107E+01	1.2942E+01	5.5343E-02	1.8440E-03	2.8354E-03	**9.8199E-05**
	Std	1.4951E+01	7.7157E+00	1.4357E-02	8.2557E-04	2.4794E-03	**8.0215E-05**
F8	Ave	-5.8357E+03	-2.1015E+03	-9.9643E+03	-6.0966E+03	-1.0603E+04	**-1.2565E+04**
	Std	1.5832E+03	4.5368E+02	6.0269E+02	7.3720E+02	1.7676E+03	**1.4712E+01**
F9	Ave	1.6696E+02	2.6831E+02	8.9184E+01	2.7989E+00	**0.0000E+00**	**0.0000E+00**
	Std	3.2922E+01	5.4592E+01	7.6590E+00	2.9866E+00	**0.0000E+00**	**0.0000E+00**
F10	Ave	2.5755E+00	1.9918E+01	5.7183E-03	1.0193E-13	3.6415E-15	**4.4409E-16**
	Std	4.2463E-01	3.6829E-01	1.4226E-03	1.8512E-14	2.3511E-15	**0.0000E+00**
F11	Ave	1.3933E-01	1.9331E+02	1.0930E-02	5.3743E-03	2.5290E-02	**0.0000E+00**
	Std	8.6736E-02	6.5660E+01	1.1322E-02	1.2782E-02	5.8241E-02	**0.0000E+00**
F12	Ave	6.5842E-02	1.9069E+07	6.0818E-05	5.0220E-02	2.9130E-02	**3.1771E-08**
	Std	8.8014E-02	1.4697E+07	4.4392E-05	3.0554E-02	1.8413E-02	**1.6357E-07**
F13	Ave	5.3638E-01	7.0116E+07	2.7496E-04	5.9829E-01	5.0846E-01	**8.5305E-08**
	Std	2.5691E-01	6.1373E+07	1.2543E-04	2.8494E-01	2.2577E-01	**3.0532E-07**
F14	Ave	2.3823E+00	1.1828E+00	**9.9800E-01**	4.4211E+00	2.5699E+00	1.3871E+00
	Std	1.7840E+00	4.9319E-01	**0.0000E+00**	4.4350E+00	2.9090E+00	2.1311E+00
F15	Ave	8.9204E-04	1.4667E-02	7.6302E-04	4.4018E-03	6.8511E-04	**3.1466E-04**
	Std	1.2835E-04	1.1322E-02	1.7060E-04	8.1189E-03	3.9964E-04	**3.4458E-06**
F16	Ave	**-1.0316E+00**	-9.3575E-01	**-1.0316E+00**	**-1.0316E+00**	**-1.0316E+00**	**-1.0316E+00**
	Std	4.7908E-16	1.1351E-01	6.7752E-16	2.1017E-08	1.3991E-09	**4.3441E-16**
F17	Ave	**3.9789E-01**	6.7558E+01	**3.9789E-01**	**3.9789E-01**	**3.9789E-01**	**3.9789E-01**
	Std	**0.0000E+00**	5.6898E+00	**0.0000E+00**	5.3761E-06	1.2708E-05	4.9476E-15

F18	Ave	**3.0000E+00**	1.1361E+01	**3.0000E+00**	**3.0000E+00**	**3.0001E+00**	**3.0000E+00**
	Std	5.4433E-15	1.2264E+01	**1.3117E-15**	2.9464E-05	3.5919E-04	3.3548E-15
F19	Ave	**-3.8628E+00**	-3.4030E+00	-3.8628E+00	-3.8618E+00	-3.8586E+00	**-3.8605E+00**
	Std	1.6349E-14	3.0378E-01	**2.7101E-15**	2.0560E-03	3.8953E-03	2.3044E-03
F20	Ave	-3.2784E+00	-1.5099E+00	**-3.3191E+00**	-3.2448E+00	-3.2238E+00	-3.0787E+00
	Std	5.8273E-02	4.7617E-01	**1.1279E-02**	7.1178E-02	9.8360E-02	1.3217E-02
F21	Ave	-7.5531E+00	-7.9895E-01	-9.8463E+00	-8.9726E+00	-8.2732E+00	**-1.0153E+01**
	Std	3.1192E+00	4.1533E-01	8.6165E-01	2.1735E+00	2.4906E+00	**9.0248E-09**
F22	Ave	-8.3791E+00	-1.0696E+00	-1.0187E+01	-1.0147E+01	-7.0393E+00	**-1.0403E+01**
	Std	3.2014E+00	6.4743E-01	1.1060E+00	1.3940E+00	3.0345E+00	**2.4713E-06**
F23	Ave	-9.7656E+00	-1.2464E+00	-1.0531E+01	-1.0084E+01	-6.2452E+00	**-1.0536E+01**
	Std	2.3520E+00	4.8144E-01	2.2727E-02	1.7514E+00	3.0078E+00	**1.2158E-06**

Table3 Comparative analysis of results on classical benchmark functions (Dim = 100)

ID		PSO	GA	DE	GWO	WOA	EWOA
F1	Ave	1.8753E+02	2.3276E+05	1.7950E+03	1.4534E-12	6.3817E-71	**0.0000E+00**
	Std	2.3570E+01	2.6509E+04	2.7185E+02	7.9596E-13	2.7005E-70	**0.0000E+00**
F2	Ave	2.2311E+02	2.6480E+02	2.8921E+01	4.6201E-08	9.3174E-50	**1.6254E-25**
	Std	5.5596E+01	2.7873E+01	3.0962E+00	1.6652E-08	4.0562E-49	**0.0000E+00**
F3	Ave	2.0172E+04	6.7224E+05	4.3273E+05	5.5597E+02	1.1065E+06	**0.0000E+00**
	Std	4.5038E+03	2.3453E+05	3.7152E+04	5.3249E+02	2.7451E+05	**0.0000E+00**
F4	Ave	1.2922E+01	9.4161E+01	8.8673E+01	7.2285E-01	7.5516E+01	**3.4405E-13**
	Std	1.5203E+00	2.5211E+00	2.5378E+00	6.7512E-01	2.7418E+01	**1.8845E-13**
F5	Ave	2.5827E+05	8.4934E+08	2.2539E+06	9.8027E+01	9.8226E+01	**5.8052E-03**
	Std	5.8964E+04	1.3986E+08	6.0653E+05	5.3483E-01	2.0632E-01	**6.4125E-03**
F6	Ave	1.9527E+02	2.2593E+05	1.7448E+03	1.0130E+01	3.9659E+00	**1.0928E-05**
	Std	3.1851E+01	1.9440E+04	2.6474E+02	1.0759E+00	1.3143E+00	**1.7722E-05**
F7	Ave	1.5137E+03	1.3506E+03	3.6131E+00	5.8173E-03	3.4682E-03	**1.9228E-04**
	Std	2.6145E+02	2.4600E+02	5.1625E-01	2.4808E-03	4.1338E-03	**2.0340E-04**
F8	Ave	-1.9432E+04	-4.0630E+03	-1.6358E+04	-1.6902E+04	-3.6806E+04	**-4.1849E+04**
	Std	4.8825E+03	9.4166E+02	5.8403E+02	1.4079E+03	4.8867E+03	**1.1910E+02**
F9	Ave	1.2073E+03	1.5317E+03	8.0061E+02	9.4238E+00	2.2737E-14	**0.0000E+00**
	Std	1.1226E+02	5.6020E+01	2.5419E+01	7.3519E+00	6.9378E-14	**0.0000E+00**
F10	Ave	6.8083E+00	2.0795E+01	7.4341E+00	1.3683E-07	2.9310E-15	**4.4409E-16**
	Std	3.0787E-01	1.2157E-01	4.2447E-01	6.4909E-08	2.1173E-15	**0.0000E+00**
F11	Ave	1.0413E+00	2.0402E+03	1.6858E+01	3.9075E-03	**0.0000E+00**	**0.0000E+00**
	Std	2.0883E-02	1.9323E+02	2.3564E+00	9.1374E-03	**0.0000E+00**	**0.0000E+00**
F12	Ave	1.4080E+01	1.8329E+09	1.5480E+06	2.9468E-01	4.0051E-02	**3.2983E-07**
	Std	6.3356E+00	3.5744E+08	7.2910E+05	7.0323E-02	1.8701E-02	**8.4035E-07**
F13	Ave	3.7350E+02	3.6308E+09	5.2644E+06	6.7823E+00	2.8496E+00	**1.6727E-05**
	Std	3.9159E+02	6.2042E+08	2.0359E+06	4.1669E-01	1.0567E+00	**4.7654E-05**

Credit scoring experimental analysis of BP neural network method based on EWOA

Credit datasets

To validate the proposed model, we utilize three widely used real-world credit datasets from the UCI machine learning library [51]. The Australian, Japanese, and German datasets are included. Table 3 provides detailed information about these datasets. The German dataset comprises an overall sample of 1000, with 700 classified as non-defaulting (an excellent sample) and 300 as defaulting (a poor sample). Each sample consists of 20 characteristic feature parameters, without including labels for classes. These dimensions include seven numerical attributes and 13 category characteristics. The Australian database comprises a total of 690 samples, with 307 labelled as non-defaulting (representing good samples) versus 383 classed as defaulting (representing bad samples). Each sample encompasses 14 attribute feature dimensions excluding class labels, comprising of 8 numerical features and 6 categorical features. Similarly to the Australian dataset, the Japanese dataset also contains an equal number of samples; it also has 307 good samples and 383 bad samples. In each sample of this dataset there are15 attribute feature dimensions (excluding class labels), including 6numerical features and 9 categorical features. See Table 4 for details.

Table 4 Description of the three

Dataset	Total samples	Good samples	Bad samples	Numerical features	Categorical features	Total features
German	1000	700	300	7	13	20
Australian	690	307	383	8	6	14
Japanese	690	307	383	6	9	15

Data preprocessing and parameter setting

First of all, check the number and proportion of missing values in each column of the data, and for numeric variables, delete those with more than 30% of missing values, and fill in the median for those with less than 30% of missing values.

Secondly, the data type is checked and coded, and the category variables are coded as numerical values before modeling, and the label coding method is used in the paper, that is, each value of the category variable is assigned an integer, but no new columns are generated.

Next, data standardization, in the process of neural network modeling, data standardization can effectively prevent gradient explosion. In addition, data standardization has the effect of unifying the scale, reducing the training time, and balancing the feature contribution. Since the neural network contains a large number of sigmoid activation functions and tanh activation functions, the use of [0,1] values is prone to gradient disappearance, so this paper chooses to use z-score standardization.

Finally, the data division, this paper in accordance with the ratio of 7:3 split the data set into training data set and test data set, in which the training data set is mainly used to allow the model to learn autonomously, while the data set used for testing is used to test the specific effect performance of the learned model. Considering that the German credit data shows some imbalance, in order to solve the sample imbalance problem, SMOTE oversampling technique is used before dividing the dataset.

The structure of the neural network and the setting of the parameters are more dependent on the experience of experts, a three-layer BP neural network can fit a function of arbitrary complexity with arbitrary accuracy, because having a hidden layer is enough to build a model of personal credit assessment. The input and output layers of the BP neural network are determined by the size of the input and output vectors, and the number of inputs is the characteristic dimension of the data. The number of inputs of the model in this study is determined by the number of attribute indicators of credit data in Germany, Australia and Japan. The personal credit assessment model is to predict possible defaults after inputting multidimensional feature data, so the output of the model is 1 or 0, and the output dimension is 1. The number of neurons in the hidden layer is determined by the empirical formula $n=\sqrt{n_1+n_2}+a$, where n_1 is the number of nodes in the input layer, n_2 is the number of nodes in the output layer1 , and a is a constant from 1 to 10. The maximum number of iterations is set to 1000, the learning rate is 0.01, the minimum training objective error is 0.00001, the momentum factor is 0.01, the minimum performance gradient is set to 1e-6, and the maximum number of failures is 6. The number of input nodes of the EWOA-BP model on the three credit datasets of German, Australian, and Japanese credits are 20,14,15, and the optimal hidden layer number of nodes are 7,6,6.

Evaluation indicators

Multiple evaluation indicators exist for machine-learning models. Commonly employed metrics include accuracy, precision, recall, F-score, area under the curve (AUC), and receiver operating characteristic (ROC) curves. The evaluation metrics used include the algorithm's overall accuracy, its ability to accurately forecast both positive and negative samples, the trade-off between accuracy and recall, and the area under the ROC curve.

When judging artificial intelligence models, precision and recall are very important factors, especially when datasets aren't balanced or when performance in finding specific classes is important. Precision is a metric that quantifies the accuracy of a model's positive class predictions, while recall measures the model's ability to identify positive class samples out of all the examples that are really positive. Precision and memory have a trade-off relationship. Recall and accuracy have an inverse relationship, meaning that as recall increases, precision decreases, and vice versa. This occurs because while attempting to identify additional positive class samples to increase recall, there is a higher likelihood of incorrectly identifying negative class samples as positive, resulting in a decrease in the precision rate. On the other hand, if we exercise caution in selecting only highly certain positive class examples (enhancing precision), we may overlook some genuinely positive class samples, resulting in a decrease in recall.

The F1 score, a commonly used metric, combines precision and recall by calculating their reconciled average. The F1 score is a concise metric that evaluates a model's ability to balance precision and recall in its performance. For this study, we have selected accuracy, AUC, F-score, and ROC curves as the evaluation metrics for the classification model. We choose these metrics by combining several assessment indices of the model. We can use TP, TN, FP, and FN values to determine accuracy, precision, recall, F-score, and AUC.

TP represents the instances where the model correctly predicts positive values, TN represents the instances where the model correctly predicts negative values, FP represents the instances where the model incorrectly predicts positive values, and FN represents the instances where the model incorrectly predicts negative values. The following are mathematical expressions for various model evaluation indicators:

$$Accuracy = \frac{TP + TN}{TP + FP + TN + FN} \tag{16}$$

$$Precision = \frac{TP}{TP + FP} \tag{17}$$

$$Recall = \frac{TP}{TP + FN} \tag{18}$$

$$F_{score} = \frac{2 \times Precision \times Recall}{Precision + Recall} \tag{19}$$

Experimental results and comparative analysis

Cross-validation is a commonly used method for model evaluation, cross validation divides the dataset into multiple subsets, one of which is used as the test set and the others as the training set, and then repeats the process several times, each time a different subset is selected as the test set, and finally averages the results of the multiple tests. In this paper, we adopt the ten-times cross-validation method, and get the final evaluation results by calculating the average performance index of the 10-times cross-validation, which avoids the dependence on a single test set and improves the model's generalization ability, reliability and stability.

In order to evaluate the efficacy of the proposed EWOA-BP model for credit assessment, a comprehensive comparison was conducted. The EWOA-BP model was pitted against both prevalent machine learning classification models and swarm intelligent optimization algorithms. This evaluation encompassed three datasets and six key performance indicators. The models under comparison encompassed SVM, ELM, RF, RBF, BP, as well as various swarm intelligence optimized versions of BP, namely PSO-BP, GA-BP, DE-BP, GGO-BP, WOA-BP, and our proposed EWOA-BP. The outcomes of this evaluation are presented in Table 5 and Table 6, where the top performers for each evaluation criterion are emphasized in bold.

Table 4 gives the three credit datasets studied in this paper, and Tables 5 and 6 give the performance of each model on the three credit datasets in Germany, Australia and Japan. In order to better reflect the performance of the EWOA-BP model proposed in the paper, we compare the EWOA-BP model with five machine learning models, namely, SVM, ELM, RF, RBF, and BP, respectively, and the results are detailed in Table5; meanwhile, we also compare it with the swarm intelligent optimization algorithms, PSO-BP, GA-BP, DE-BP, GO-BP and WOA-BP, and the results are detailed in Table 6.

In particular, the F1 score is often used as a statistical metric to assess the accuracy of binary classification algorithms. This metric balances the precision and recall of the classification model and provides a balanced measure of the model's performance in the classification task. It is often used to assess the effectiveness of classification models. the higher the F1 score, the more effective the experimental approach.

The results in Table5 show that the EWOA-BP model has the highest accuracy on all three datasets, the largest AUC values on all three datasets, and the best F1 values on two occasions when compared to the five machine learning models, SVM, ELM, RF, RBF, and BP. The results in Table 6 show that the EWOA-BP model has the highest accuracy on all three datasets, the largest AUC value on all three datasets and obtains the best F1 value twice when compared with the five swarm intelligent optimization algorithms PSO-BP, GA-BP, DE-BP, GWO-BP and WOA-BP.

Moreover, it is easy to see that the EWOA-BP model outperforms the German credit dataset in both comparisons of the Australian and Japanese credit datasets, especially in the Australian dataset where the best performance is obtained for both. Further analysis of the dataset shows that the German credit dataset is unbalanced with a ratio of good to bad credit of 7:3. In order to obtain better experimental results, the German credit data was balanced by performing the SMOTE stochastic oversampling process. Figure 5 gives a comparative analysis of the 11 algorithms before and after data balancing in terms of three important evaluation metrics: accuracy, AUC take and precision. After the SMOTE random oversampling data balancing process, the accuracy, AUC value and precision rate of the 11 algorithms are improved to some extent.

The accuracy rate is easily affected by the sample imbalance, and if a certain category has a majority of samples, the model may tend to categorize all samples into this category. After the data is balanced, the accuracy rate increases, meaning that the model does a better job overall. The AUC value reflects the model's ability to categorize positive and negative samples under different thresholds, and an increase in the AUC value indicates that the model's ability to differentiate between positive and negative samples improves, and the false positive rate decreases, which indirectly influences the accuracy rate The increase of AUC value indicates that the model's ability to distinguish between positive and negative samples is improved, and the false alarm rate is reduced, which indirectly affects the improvement of accuracy rate. Accuracy rate is a measure of the proportion of samples predicted by the model to be positive that are actually positive, which reflects the accuracy of the model's prediction results. In credit evaluation, we pay more attention to defaulted loans as positive examples and non-defaulted loans as negative examples. When the accuracy ratio is higher, it means that we screen out the defaulted loans more accurately and reduce the possibility of misjudgment, which will effectively reduce unnecessary economic losses. Also the comparison results reflect the importance and necessity of data preprocessing. In order to solve the challenges of the dataset and reduce the negative impact on the experimental results, data preprocessing is performed in this study through 4 steps.

ROC and AUC are widely used as important evaluation methods for model performance metrics. Among them, the ROC curve describes how the classifier performance varies with the classifier threshold. The ROC curve plots sensitivity on the vertical axis and 1-perpendicularity on the horizontal plane. Subsequently, a region is created below the curve, bounded by a straight line at a 45-degree angle, called the Area Under the Curve (AUC). The size of the AUC is directly related to the quality of the judgments, with a larger AUC indicating higher quality judgments. The closer the value of this area is to 1, the better the recognition ability. Generally speaking, the AUC value of a suitable model is not less than 0.5.

Table 5 and Table 6 gives the results of the EWOA-BP model on a total of three credit data in Germany, Australia, and Japan, respectively, with the five models of machine learning (SVM, ELM, RF, RBF, and BP), the five models of swarm intelligent optimization algorithm (PSO-BP, GA-BP, DE-BP, GGO-BP, WOA- BP, WOA-BP) with comparative analysis of ROC curves. It is easy to see that on the Australian credit and Japanese credit datasets, the AUC values of the selected comparison models are more than 0.8, and the EWOA-BP model performs even better, with the AUC values of the two datasets exceeding 0.9, which reflects that the EWOA-BP model has a high classification ability. However, on the German credit dataset, the EWOA-BP model still performs better, but the overall performance of the models is average, with some correlation with the data itself. The combined excellent performance of the EWOA-BP model on the three datasets demonstrates the effectiveness and robustness of the model.

Although the EWOA-BP model performs well on all three classical credit datasets, the experimental results are somewhat dependent on the datasets, and further optimization of the model and good feature engineering of the experimental data are needed to better enhance the classification effect and reflect the robustness of the model. Especially when encountering larger data sample sizes and complex datasets in the future, this is the work we need to further explore in the future.

Taken together, our EWOA-BP model outperforms the other standard classifiers on most of the assessment metrics on all three credit assessment datasets. An exhaustive survey establishes that the EWOA-BP model is efficient and superior for credit assessment. In addition, the performance of the Random Forest (RF) model and the GA-BP model is also impressive.

Table 5 Performance of machine learning classifiers on test sets with different datasets

Dataset	Classifier	Accuracy	AUC	F-score
German	SVM	0.75	0.7625	0.8314
	ELM	0.7438	0.7732	0.8222
	RF	0.7793	0.7812	0.8519
	RBF	0.7242	0.7323	0.8067
	BP	0.7623	0.7752	0.8451
	EWOA-BP	**0.7836**	**0.7858**	**0.8521**
Australian	SVM	0.8443	0.9139	0.8229
	ELM	0.8325	0.8862	0.8368
	RF	0.8596	0.9185	0.8558
	RBF	0.8283	0.8453	0.8310
	BP	0.8462	0.9127	0.8323
	EWOA-BP	**0.8731**	**0.9265**	**0.8718**
Japanese	SVM	0.8399	0.8918	0.8529
	ELM	0.8215	0.8851	0.8342
	RF	0.8645	0.9186	**0.8781**
	RBF	0.8102	0.8766	08203
	BP	0.8428	0.9014	0.8547
	EWOA-BP	**0.8726**	**0.9218**	0.8667

Table6 Performance of intelligent optimization classifiers on test sets with different datasets

Dataset	Classifier	Accuracy	AUC	F-score
German	PSO-BP	0.7548	0.7822	0.8282
	GA-BP	0.7808	0.7847	**0.8589**
	DE-BP	0.7352	0.7413	0.8127
	GWO-BP	0.761	0.7725	0.8374
	WOA-BP	0.7733	0.7842	0.8531
	EWOA-BP	**0.7836**	**0.7858**	0.8521
Australian	PSO-BP	0.8553	0.9229	0.8289
	GA-BP	0.8706	0.9235	0.8618
	DE-BP	0.8393	0.8543	0.837
	GWO-BP	0.8435	0.8952	0.8428
	WOA-BP	0.8572	0.9217	0.8383
	EWOA-BP	**0.8731**	**0.9265**	**0.8718**
Japanese	PSO-BP	0.8325	0.8941	0.8402
	GA-BP	0.8725	0.9206	0.8841

	DE-BP	0.8212	0.8856	0.8263
	GWO-BP	0.8509	0.9008	0.8589
	WOA-BP	0.8538	0.9104	0.8607
	EWOA-BP	**0.8726**	**0.9218**	**0.8667**

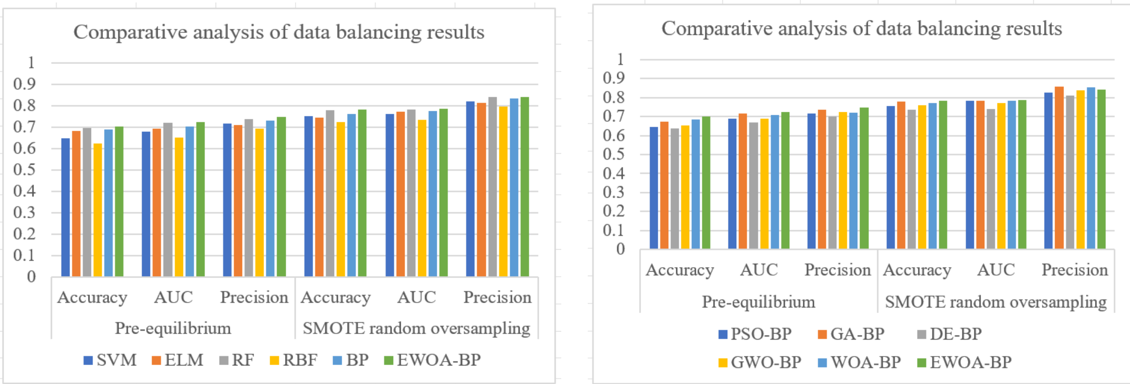


Figure 5 Comparison analysis of data balancing results on German dataset

Figure 6 provides a comparative study of the accuracy of each classification model on both the training and test sets in order to delve deeper into their correctness. See Figure 6 for details.

When comparing the five classes of machine learning models (SVM, ELM, RF, RBF, and BP) with EWOA-BP, it is not surprising that RF and EWOA-BP outperform the others on both the training and test sets. However, EWOA-BP performs particularly well on the test set and places more emphasis on accuracy in this regard. The swarm intelligence optimization algorithms PSO-BP, GA-BP, DE-BP, GGO-BP, and WOA-BP were tested on three credit assessment datasets. GA-BP and EWOA-BP did better on both the learning set and the testing set. However, EWOA-BP outperforms the others specifically on the test set, indicating its effectiveness and robustness.

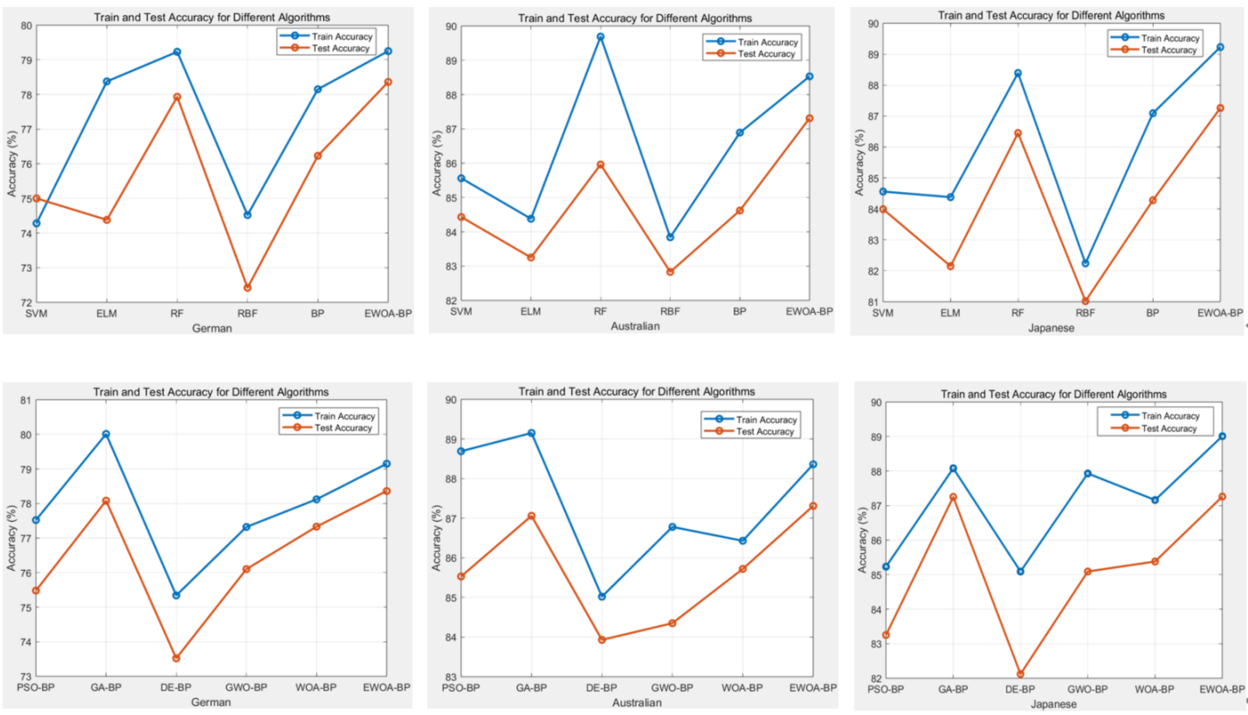


Figure 6 Comparison of accuracy of different classifiers on different datasets

CONCLUSION AND PROSPECTS

This study introduces a personal credit assessment model (EWOA-BP), which is based on the enhanced whale algorithm optimization and uses a BP neural network. We initially present a multi-strategy Improved Whale Algorithm (EWOA), and to achieve a balance between discovery and extraction, we incorporate a novel nonlinear reduction technique utilizing the function of cosine. Moreover, the method employs a unique exploration strategy that harnesses the leader's adaptive tangential flight to reduce the risk of stumbling into the ideal local location. Additionally, the program employs an innovative exploitation strategy to accelerate the rate of convergence while maintaining precision. To test the proposed whale optimization algorithm, we first verify the effectiveness of the derived improved whale algorithm (EWOA) using 23 classical benchmark functions. Then, we use the improved whale algorithm to optimize the neural network and construct the EWOA-BP model. Finally, we experimentally validated the model on three credit assessment datasets: Germany, Australia, and Japan. When comparing different machine learning models like SVM, ELM, RF, RBF, and BP for 5-class classification and Swarm Intelligence Optimization Algorithm models like PSO-BP, GA-BP, DE-BP, GWO-BP, and WOA-BP for the same classification task, the EWOA-BP model does better at judging people's credit. Therefore, we can conclude that the EWOA-BP algorithm is both effective and highly competitive.

However, the research content of this paper still has some room for expansion, and further research can be carried out in the following directions in the future. Firstly, EWOA algorithm has good optimization ability and stability, in order to better play the performance of the algorithm, can consider the EWOA algorithm to extend to multi-objective optimization problems, in order to solve more complex practical applications, such as: how to use the algorithm to solve the actual engineering problems is the future direction of research. Secondly, At the same time, EWOA algorithm can be combined with other intelligent technologies, such as neural networks, deep learning, etc., to improve the adaptability and learning ability of the algorithm. For example, in the future, we can consider using population initialization and boundary control techniques to enhance the effectiveness of EWOA, and using EWOA-BP model to cope with various challenges in the real world, so as to achieve better classification results.

DATA AVAILABILITY STATEMENT

Data will be made available on request. Available online: http://archive.ics.uci.edu/ml.

ACKNOWLEDGMENTS

This work was supported by the National Natural Science Foundation of China (12261027), the National Natural Science Foundation of China (12361031), Guangxi Special Project of Science and Technology Base and Talent Development (GuikeAD23026016), the Basic Ability Improvement Project of young and middle-aged teachers in universities of Guangxi (2021KY0796), “Research on Personal Credit Evaluation Based on Machine Learning”.

AUTHOR CONTRIBUTIONS

Conceptualization: Yanqin Fan, Haiwu Huang.

Data curation: Yanqin Fan, Guolin Wu

Formal analysis: Yanqin Fan.

Funding acquisition: Yanqin Fan, Haiwu Huang.

Investigation: Yanqin Fan.

Methodology: Yanqin Fan, Haiwu Huang.

Project administration: Yanqin Fan.

Supervision: Guolin Wu.

Visualization: Yanqin Fan.

Writing– original draft: Yanqin Fan, Guolin Wu.

Writing– review & editing: Yanqin Fan.

CONFLICTS OF INTEREST

The authors affirm that they do not possess any identifiable conflicting financial interests or personal ties that could have potentially influenced the findings presented in this paper.

REFERENCES

[1] Durand, David. Appendix B: Application of the Method of Discriminant Functions to the Good-and Bad-Loan Samples. Risk Elements in Consumer Instalment Financing, Technical Edition. NBER, 1941, 125-142.

[2] Myers J H, Forgy E W. The development of numerical credit evaluation systems. Journal of the American Statistical association, 1963, 58(303): 799-806.

[3] John C. Wiginton. A Note on the Comparison of Logit and Discriminant Models of Consumer Credit Behavior. Journal of Financial and Quantitative Analysis, 1980, 15(3):757-770.

[4] Makowski P. Credit scoring branches out. The Credit World, 1985, 75(1): 30-37.

[5] Enas G G, Choi S C. Choice of the smoothing parameter and efficiency of k-nearest neighbor classification. Statistical methods of discrimination and classification. Pergamon, 1986, 235-244.

[6] Mushava J, Murray M. Flexible loss functions for binary classification in gradient-boosted decision trees: An application to credit scoring. Expert Systems with Applications, 2024, 238: 121876.

[7] Bastos J A. Predicting credit scores with boosted decision trees. Forecasting, 2022, 4(4): 925-935.

[8] Zou Y, Gao C , Xia M, Pang C . Credit scoring based on a Bagging-cascading boosted decision tree. Intelligent Data Analysis, 2022, 26(6): 1557-1578.

[9] Xia Y, Guo X, Li Y, He L, Chen X. Deep learning meets decision trees: An application of a heterogeneous deep forest approach in credit scoring for online consumer lending. Journal of Forecasting, 2022, 41(8): 1669-1690.

[10] Zhou Ying, Shen Long, Ballester Laura. A two-stage credit scoring Model based on random forest: Evidence from Chinese small firms. International Review of Financial Analysis, 2023, 89: 102755.

[11] Kuyoro A O, Ogunyolu O A, Ayanwola T G, Ayankoya F Y. Dynamic effectiveness of random forest algorithm in financial credit risk management for improving output accuracy andloan classification prediction. Ingénierie des Systèmes d'Information, 2022, 27(5): 815-821.

[12]Boughaci D, Alkhawaldeh K A A , Jaber J J , Nawaf Hamadneh N. Classification with segmentation for credit scoring and bankruptcy prediction . Empirical Economics, 2020, 61: 1-29.

[13] Ling S K , Jamaian S S , Mansur S , et al. Modeling Tenant's Credit Scoring Using Logistic Regression. SAGE Open, 2023, 13(3).

[14] D'Amato A , Mastrolia E. Linear discriminant analysis and logistic regression for default probability prediction: the case of an Italian local bank. International Journal of Managerial and Financial Accounting, 2022, 14(4): 323-343.

[15] Elena D, Sullivan H, Christophe H , Sessi T. Machine learning for credit scoring: Improving logistic regression with non-linear decision-tree effects. European Journal of Operational Research, 2022, 297(3): 1178-1192.

[16]Shen F, Yang Z, Zhao X, Lan D. Reject inference in credit scoring using a three-way decision and safe semi-supervised support vector machine. Information Sciences, 2022, 606: 614-627.

[17]Sebastián M, Julio L, Carla V .Time-weighted Fuzzy Support Vector Machines for classification in changing environments. Information Sciences, 2021, 559: 97-110.

[18] Rahul H , Kumar S S ,Uwe A . Optimised deep k-nearest neighbour's based diabetic retinopathy

diagnosis(ODeep-NN) using retinal images. Health Information Science and Systems, 2024, 12(1): 23-23.

[19]María A C P, Isaac R S, Pedro F M G .K-nearest neighbour and K-fold cross-validation used in wind turbines for false alarm detection. Sustainable Futures, 2023, 6: 100132 .

[20] Chuah C, He W, Huang D. G Mean-a semi-supervised GRU and K-mean model for predicting the TF binding site. Scientific reports, 2024, 14: 2539.

[21] Pankaj J, Saurabh G. Enhancing blood flow prediction in multi-exposure laser speckle contrast imaging through ensemble learning with K-mean clustering. Biomedical physics engineering express, 2023, 10(2).

[22] Li G , Yuan N , Jiang M , Yan S, Lou M. Study on the spatial and temporal evolution of industrial carbon emission efficiency and influencing factors based on improved AdaBoost regression algorithm. Applied Mathematics and Nonlinear Sciences, 2024, 9(1): 1-15.

[23] Li Y, Chen W. A Comparative Performance Assessment of Ensemble Learning for Credit Scoring. Mathematics, 2020, 8(10):1756.

[24] Wu S, Xie L, Xian J. Finite-time output feedback trans-media tracking control of a slender body trans-media vehicle via neural network extended state observer. Transactions of the Institute of Measurement and Control, 2024, 46(7): 1397-1409.

[25] Li Z, Zhang X, Deng F, Zhang Y. Integrating deep neural network with logic rules for credit scoring. Intelligent Data Analysis, 2023, 27(2):483–500.

[26] He, X, Li, S, He, X, Wang, W, Zhang, X, Wang, B. A Novel Ensemble Learning Model Combined XGBoost With Deep Neural Network for Credit Scoring. Journal of Information Technology Research, 2022,15: 1-18.

[27] Aydogmus O, Boztas G. Implementation of singularity-free inverse kinematics for humanoid robotic arm using Bayesian optimized deep neural network. Measurement, 2024, 229: 114471.

[28] Koichi I , Masanori A ,Tomoki H .Bayesian Network Oriented Transfer Learning Method for Credit Scoring Model. IEEJ Transactions on Electrical and Electronic Engineering, 2021, 16(9): 1195-1202.

[29]] Ma Z , Hou W , Zhang D . A credit risk assessment model of borrowers in P2P lending based on BP neural network. PLOS ONE, 2021, 16.

[30] Li Z , Liu L , Zhu L ,et al. Parallel double-layer prediction model construction and empirical analysis for enterprise credit assessment. Intell. Data Anal. 2022, 26: 1007-1022.

[31] Networks S C . Retracted: Application of Business Intelligence Based on the Deep Neural Network in Credit Scoring. Security and Communication Networks, 2023.

[32] Maina L K , Kasamani B S .A Centralized Credit Scoring Prototype for Microlending Institutions Using Neural Networks. The Proceedings of the International Conference on Smart City Applications. Springer, Cham, 2022.

[33] Sun B, Liu H, Tang J, Rong S, Liu Y, Jiang W. Optimization of heat treatment deformation control process parameters for face-hobbed hypoid gear using FEA-PSO-BP method. Journal of Manufacturing Processes, 2024, 117:40-58.

[34] Cui C, Wu H, Jiang X , Jing L. Short- and medium-term forecasting of distributed PV output in plateau regions based on a hybrid MLP-FGWO-PSO approach. Energy Reports, 2024, 11: 2685-2691.

[35] Ren J .A Reliable Night Vision Image De-Noising Based on Optimized ACO-ICA Algorithm. IAENG International Journal of Computer Science, 2024, 51(3): 169-177.

[36] Huo F, Zhu S, Dong H, Ren W. A new approach to smooth path planning of Ackerman mobile robot based on improved ACO algorithm and B-spline curve. Robotics and Autonomous Systems, 2024, 175: 104655.

[37] Rajapackiyam E , Devi A ,Reddy S M , et al. An efficient computation offloading in edge environment using genetic algorithm with directed search techniques for IoT applications. Future Generation Computer Systems, 2024, 158: 378-390.

[38] Olivo J ,Cucuzza R ,Bertagnoli G , Domaneschi M. Optimal design of steel exoskeleton for the retrofitting of RC buildings via genetic algorithm. Computers and Structures, 2024, 299: 107396.

[39] Karaboga D. Basturk B. A powerful and efficient algorithm for numerical function optimization: artificial bee colony (ABC) algorithm, J. Global Optim, 39 (2007) :459–471.

[40] Zhang F, Huang Y, Chen B, et al. The Calibration of theta-phi Fiber Positioners Based on the Differential Evolution Algorithm. The Astronomical Journal, 2024, 167(3).

[41] Salgotra R, Gandomi A H. A novel multi-hybrid differential evolution algorithm for optimization of frame structures. Scientific Reports 14, 2024:4877-4877.

[42] Rashedi E, Nezamabadi-pour H, Saryazdi S. GSA: A gravitational search algorithm, Information Sciences, 2009, 179(13): 2232-2248.

[43] Lin Z , Xiao J .Reconfiguration of distributed power distribution networks based on improved gravitational search algorithms. Journal of Physics: Conference Series, 2024, 2741(1): 012054.

[44] Wu J , Wang Y ,Burrage K , et al. An improved firefly algorithm for global continuous optimization problems. Expert Systems With Applications, 2020, 149: 113340.

[45] Pierezan J, Coelho L D S. Coyote optimization algorithm: a new metaheuristic for global optimization problems, 2018 IEEE congress on evolutionary computation (CEC). IEEE, 2018: 1-8.

[46] Mirjalili S, Mirjalili S M, Lewis A. Grey wolf optimizer. Advances in engineering software, 2014, 69: 46-61.

[47] Mirjalili S, Lewis A. The whale optimization algorithm. Advances in engineering software, 2016, 95: 51-67.

[48] Zhang Y, Li H, Wang Z, Wang H. A Multi-Objective Learning Whale Optimization Algorithm for Open Vehicle Routing Problem with Two-Dimensional Loading Constraints. Mathematics, 2024, 12(5):731.

[49] Liang Z, Shu T, Ding Z. A Novel Improved Whale Optimization Algorithm for Global Optimization and Engineering Applications. Mathematics, 2024, 12(5): 636.

[50] Yao X, Liu Y, Lin G. Evolutionary programming made faster. IEEE Transactions on Evolutionary computation, 1999, 3(2): 82-102.

[51] Dua, D.; Graff, C. UCI Machine Learning Repository. 2019. Available online: http://archive.ics.uci.edu/ml.