scientific reports

OPEN

A medical disease assisted diagnosis method based on lightweight fuzzy SZGWO-ELM neural network model

Qiuju Chen✉, Chenglong Zhang, Tianhao Peng, Youshun Pan & Jie Liu

The application of neural network model in intelligent diagnosis usually encounters challenges such as continuous adjustment of network parameters and significant cost in training the network facing numerous complex physiological data. To address this challenge, this paper introduces a fuzzy SZGWO-ELM neural network model for medical disease aid diagnosis with fuzzy membership function and ELM network to refine the improved Gray Wolf optimization algorithm. Firstly, the Z-type membership function is introduced as the inertia weight to get a balance for the grey wolf in seeking the optimal solution globally and locally and ensuring fast convergence. Secondly, the S-type membership function is utilized as the adaptive weight to flexibly adjust the grey wolf search step size to facilitate a quick approximation of the optimal solution. Finally, the improved Gray Wolf optimization algorithm is used to optimize the parameters of the ELM neural network model, termed as SZGWO-ELM. This method can eliminate the need for extensive network parameter adjustments and quickly locate the optimal solution to the problem using a lightweight neural network. The performance of the SZGWO is assessed by using metrics like convergence, mean, and standard deviation. Multiple experiments reveal that this method shows superior performance. Furthermore, five publicly accessible medical disease datasets from UCI were conducted to evaluate the performance of SZGWO-ELM network model comparing with different classify model, and the results in terms of precision, sensitivity, specificity and accuracy can achieve 99.52%, 94.14%, 99.26% and 96.08%, respectively, which illustrate that the proposed SZGWO-ELM neural network significantly enhance the model's accuracy, providing better support for doctors in disease diagnosis.

Keywords Z-type membership function, S-type membership function, Grey wolf optimization, SZGWO-ELM, Disease assisted diagnosis

In recent years, with the high-pressure environments and fast-paced lifestyle, people's unhealthy living habits have increased the risk of sudden cardiac events, potentially leading to fatalities. Heart disease is one of the major causes of mortality rates, with approximately 18 million people annually die to it, as reported by the World Health Organization[1–3]. Heart disease, encompassing conditions like coronary heart disease, atherosclerosis, and congenital heart defects[4], can progress to debilitating conditions such as paralysis, irregular heart rhythms, and heart failure compromising overall bodily functions. However, for asymptomatic patients lacking typical signs of distress, insufficient medical expertise may result in suboptimal diagnoses, impacting optimal treatment and potentially leading to life-threatening consequences[5]. Therefore, achieving accurate diagnoses in the early stages of heart and kidney disease development is critical for human well-being and the extended lifespan of patients[6].

Nevertheless, doctors face the challenge of analyzing large amounts of physiological data to accurately diagnose the disease, which requires them to have sufficient a priori knowledge. Most crucially, for elderly individuals suffering from multiple conditions such as hypertension, obesity, diabetes, hypercholesterolemia, abnormal heart rate, and asymptomatic kidney disease, diagnosing heart disease becomes highly challenging for doctors[7]. Therefore, to effectively analyze the vast and complex data associated with heart disease, many researchers harness the advantages of neural network, applying neural network methods to the field of disease diagnosis to assist doctors in more accurate diagnosis and prediction of various diseases. The application of neural networks for medical assisted diagnosis often encounters challenges like continuous adjustment of network parameters and significant cost in training the network. Therefore, this paper proposes a lightweight

Department of Automation, Moutai Institute, Renhuai, Guizhou, China. ✉email: chjmtxy@163.com

fuzzy SZGWO-ELM neural network model to assist doctors in the diagnosis and prediction of heart disease. The model is composed of an Extreme Learning Machine (ELM) network and an improved Grey Wolf Optimization (GWO) algorithm using fuzzy membership functions. The main purpose of this paper is to utilize the lightweight fuzzy SZGWO-ELM neural network to predict whether a patient is suffering from a disease based on various lesion characteristics in the early stages. This assists doctors in marking timely diagnoses and alerts patients for prompt treatment, thereby reducing the risks associated with late-stage diagnoses. The SZGWO-ELM neural network employs two fuzzy membership functions to improve GWO for overcoming the limitations of getting stuck in local optima and overfitting in neural network solving, and reducing the cost of network parameter adjustments and network training.

The main contributions of this paper are as follows: (1) employing a Z-type membership function (ZMF) to balance the search capability of the GWO algorithm in both global and local searches; (2) dynamically adjusting the search step size of the GWO algorithm using an S-type membership function (SMF) to rapidly obtain optimal solutions; and (3) constructing an SZGWO-ELM neural network model combining the above two fuzzy membership functions improved GWO algorithm for disease diagnosis. The SZGWO-ELM neural network model's effectiveness in diagnosing and predicting heart and kidney diseases is comprehensively validated from various perspectives, including precision, sensitivity, specificity, accuracy, accompanied by ROC curves and AUC values.

The organization of the remaining sections in this paper is as follows: Sect. 2 provides an overview of relevant work on the application of intelligent optimization algorithms in medical disease diagnosis. Section 3 briefly describes the principles of the original GWO, provides a detailed explanation of the proposed fuzzy membership function improved GWO, outlines the steps and algorithmic process, and presents the network model and steps of the improved GWO algorithm combined with the ELM network. Section 4 focuses on setting parameters for the algorithm's test functions, conducting experimental comparisons with 24 test functions, various enhancement strategies, and different optimization algorithms. Section 5 makes a conclusion of this paper.

Related work

Recently, machine learning has been widely used in the medical field for medically assisted diagnosis due to its advantages in handling large amounts of complex physiological data. For example, Alshaikh et al[8]. developed an ML-HDPM model with multi information combination for heart disease prediction and achieved 95.5% in accuracy. Manikandan et al[9]. used logistic classifier for the classification heart disease by extracting the Boruta feature with 88.52% accurate classification. Adler et al[10]. used a decision tree algorithm to establish a machine learning model in heart disease to predict the risk of heart disease-related mortality. Amin et al[11]. introduced a heart disease prediction model that combines different features and seven machine learning classification algorithms, demonstrating an approximately 87.4% prediction accuracy through experiments. Saifudin et al[12]. applied bagging with the random forest algorithm to reduce misclassification predictions for coronary heart disease. Saqlain et al[13]. developed a diagnostic system for detecting clinical coronary heart disease using a support vector machine, achieving an approximately 82% diagnostic accuracy in four UCI heart disease datasets.

Machine learning-based diagnostic models for heart disease can assist doctors in working efficiently and improve the accuracy of disease diagnosis. However, the complexity of certain diseases leads to large-scale datasets, posing challenges when using traditional machine learning models for diagnosis. This can result in significant time consumption and potential model overfitting, especially with high-dimensional disease data. Some researchers focus on dimensionality reduction for high-dimensional data. For example, Zhang et al[14]. applied PCA to reduce data dimensionality and incorporated an autoencoder neural network for breast cancer diagnosis, demonstrating favorable model performance. Rajagopal et al[15]. employed five dimensionality reduction techniques to reduce dimensionality in arrhythmia data and used a probabilistic neural network classifier for data classification with positive outcomes. On the other hand, some researchers utilized increasingly deep neural network (DNN) models to enhance the correct classification rate of data. For instance, Bharti et al[16]. integrated machine learning and deep learning algorithms for the analysis of a heart disease dataset, and the experimental results showed it not only improved the accuracy of heart disease diagnosis but also reduced the diagnosis time. Dun et al[17]. combined deep learning and ensemble learning techniques, fine-tuning hyperparameters for a network model designed for heart disease diagnosis, and experimental validation confirmed its effectiveness. Hamad et al[18]. applied DNN to build a classification and prediction model for heart disease, aiming to diagnose and prevent heart disease in its early stages, reducing the incidence and severity of heart disease. Sharifrazi et al[19]. constructed a deep learning diagnostic model for diseases such as myocarditis. Experimental validation demonstrated a 97.41% diagnostic accuracy. Zeleznik et al[20]. utilized a DNN prediction system for automated coronary artery calcification cardiovascular event prediction. Experimental validation affirmed the strong testing reliability of the prediction system. Shanbhag et al[21]. developed a deep learning model based on a generative adversarial neural network for compensatory research on myocardial perfusion imaging, used for diagnosing coronary artery heart disease. Claux et al[22]. proposed a dual convolutional neural network model based on the U-Net architecture for the segmentation of intracranial arteries and detection of arterial aneurysms. Experimental validation showed that the model effectively improves diagnostic performance. Smith et al[23]. utilized ELM neural network for extracting EEG signals to assist doctors in the early detection of the ADHD and mitigate cognitive impairments and depression associated with the condition. Nahiduzzaman et al[24]. employed ELM neural network to classify diabetic retinopathy, achieving similar outcomes to CNN while utilizing fewer parameters and requiring less training time. Abd et al[25]. improved the Garson algorithm by using ELM for Alzheimer's disease detection, achieving a high accuracy rate of 99.23% in classification.

Facing the classification of high-dimensional disease data, neural network can improve the classification accuracy to a certain extent. However, with the dynamic adjustment of network and the increasing number of number of layers and neuron nodes may occur, which may lead to reduced classification accuracy. To swiftly find optimal solutions in such an extensive network space and tackle the overfitting issue in neural networks, researchers are turning their attention to bio-inspired intelligent optimization algorithms, such as gravitational search algorithm(GSA)[26], Chernobyl Disaster Optimizer (CDO)[27], grey wolf optimization algorithm(GWO)[28], optical microscope algorithm (OMA)[29], sine-cosine optimization algorithm (SCA)[30], wind-driven optimization algorithm (WDO)[31], multiverse optimization algorithm (MVO)[32], Whale optimization algorithm (WOA)[33], Particle swarm optimization (PSO)[34], Genetic Algorithm (GA)[35], and others.

Recently, researchers have also proposed some intelligent optimization with better performance. For instance, Abdullah et al[36]. proposed a novel fitness dependent optimizer (FDO) inspiring by the breeding reproductive progress of bee and achieved the promising results on the CEC 2019 benchmark functions. Mohammed et al[37]. proposed a FOX optimization algorithm inspired by the hunting behavior of foxes in nature, and the FOX algorithm was applied to solve engineering problems, achieving favorable results. Abdulhameed et al[38]. proposed a novel Child Drawing Development Optimization (CDDO) algorithm based on children's learning behaviors and cognitive development stages, and the performance of CDDO algorithm was evaluated on the 19 benchmark functions and verified its outstandingly robust. El-Kenawy et al[39]. inspired by the group flight of greylag goose, proposed the Greylag Goose Optimization (GGO) algorithm and validated its effectiveness using 19 UCI datasets. The algorithm also performs well in engineering problems. Abdollahzadeh et al[40]. presented a hyper-heuristic algorithm, called Puma Optimizer (PO), which has unique and powerful mechanisms to balance the exploration and exploitation phase, and can enhance the performance of against optimization problems.

These intelligent optimization algorithms had been used to solve many applications in different fields, such as image processing, machine learning, fuel and energy, civil engineering, and medical engineering. For instance, Pervaiz et al[41]. conducted an investigation into the usage of PSO methods and various improved versions of PSO for medical disease detection. Khafaga[42]proposed using DNN to extract breast cancer features and applied WOA, GWO, GA, and PSO to optimize classifiers for breast cancer diagnosis. Eid et al[43]. improved Long Short-Term Memory neural networks using PSO and GWO optimization algorithms, constructing a disease model for accurate prediction of monkeypox. Eluri et al[44]. introduced a population-based the Golden Eagle Optimizer, balancing the algorithm with a time-varying flight length. The researcher combined GA with the Firefly Algorithm, proposing an HBFS-GA model that achieved a classification accuracy of 99.51% on a lung cancer dataset[45]. The researcher classified over 100 intelligent optimization algorithms, analyzed their binary variants, and conducted experiments on UCI disease datasets using various neural networks, demonstrating outstanding performance[46]. Bangyal et al[47]. improved PSO by applying pseudorandom sequences and relative inertia weights for population initialization, enhancing diversity and accelerating population search speed. To address the premature convergence issue in search, an adaptive Seagull Optimization Algorithm was proposed to enhance algorithm diversity and convergence factors[48]. Inspired by the spread of the Ebola virus, Oyelade et al[49]. introduced an Ebola Optimization Search Algorithm, constructing a neural network model for breast cancer prediction with a remarkable accuracy of 96%. Nadimi-Shahraki et al[50]. applied various improvements to the WOA algorithm, including convergence, migration, and binary modifications, effectively validating it on medical datasets and achieving high performance on the COVID-19 dataset. Pashaei et al[51]. introduced a Gorilla Optimization Algorithm for the classification of biomedical data, experimentally confirming its effectiveness in feature elimination and improved classification accuracy. Owfek et al[52]. employed binary PSO with WOA to select student learning features for finding students interacting with AI during their learning process.

Intelligent optimization algorithms come in a variety, each with its distinctive features and areas of application. Among these algorithms, the GWO algorithm stands out for its minimum parameters, simple implementation, and fast convergence, and has been widely used in the field of intelligent medical assisted diagnosis. Suresha et al[53]. used GWO to extracte feature vectors for Alzheimer's disease diagnosis. Saleh et al[54]. used GWO to optimize the features of diseases and selected the most effective features for classification and diagnosis of diseases. Kiliçarslan[55]mixed PSO and GWO to optimize the parameters in the deep learning model for the classification of heart disease. Deep Kusum[56]adopted a random walk strategy based on discrete factors to improve GWO for feature selection in chronic diseases. Chakraborty et al[57]. combined machine learning classification techniques with enhanced GWO algorithm to optimize the feature selection for complex biomedical data analysis.

However, the GWO algorithm tends to get trapped in local optima during the search. Hence, this paper introduces an enhancement GWO through two fuzzy membership functions, which is used the ZMF as the inertia weight to balance the search ability of the global and local area, and employed the SMF as the adaptive weight to dynamically set the search step size for achieving the optimal value rapidly. This modification aims to maintain algorithmic convergence accuracy while preventing it from falling into local optima. Meanwhile, the improved SZGWO algorithm is used to optimize the parameters of the ELM neural network model for enhancing the performance of the model while using fewer parameters and shorter training time.

The improved GWO algorithmic framework

The original GWO algorithm

GWO was designed by Mirjalili et al[28]., which is motivated by the social behaviors of the elitist hierarchy and hunting mechanistic strategy in the natural world. The hierarchical system of the GWO is parted into four grades, namely α, β, δ and ω. where α is the top level in the population, which is a decision maker to manage and leader the wolf pack. The second level is β, which mainly assist α wolves in decision-making and replacing them when necessary. and δ is considered as the third level, which mainly listen to α wolves and β wolves and assist them in managing the pack close to the prey. The remaining individuals are denoted as ω in population, which are responsible for the balance of the relationship within the wolf pack. Individual grey wolves at all levels are in competition during the iteration of the algorithm, the leader wolf needs to be reselected relying on the distance among each individual and its prey, and the wolf pack's location and related parameters are updated according to Eq. (1) to Eq. (7). GWO can get the maximum admissible error by using the weight variables and changing the control the parameter, and has the benefits of simple implementation, fewer parameters and faster convergence[28].

$$u\left(t+1\right)=u_p\left(t\right)-b\cdot p \tag{1}$$

$$p=\left|k\cdot u_p\left(t\right)-u\left(t\right)\right| \tag{2}$$

where t is the current iteration, $u(t+1)$ indicates the grey wolf position, $u_p(t)$ is the prey position, p indicates the distance between the prey and the grey wolf, $\cdot$ represents multiplication. b and k are respectively coefficient vectors obtained by Eqs. (3) and (4).

$$b=2\cdot A\cdot R_1-A \tag{3}$$

$$k=2\cdot R_2 \tag{4}$$

R_1 and R_2 represent two stochastic numbers of the scope [0,1], the value of A is a contraction factor, which is gradually decreased linearly from 2 to 0 as the increasing iteration. $T_{\max}$ express the total amount of iteration.

$$A=2-2t/T_{\max} \tag{5}$$

According to the hunting behavior and the hierarchical system of prey wolf packs, each individual grey wolf move by Eqs. (6) and (7).

$$\begin{cases} p_\alpha=\left|k_1\cdot u_\alpha-u\right| \\ p_\beta=\left|k_2\cdot u_\beta-u\right| \\ p_\delta=\left|k_3\cdot u_\delta-u\right| \end{cases} \tag{6}$$

$$\begin{cases} u_1=u_\alpha-b_1\cdot p_\alpha \\ u_2=u_\beta-b_2\cdot p_\beta \\ u_3=u_\delta-b_3\cdot p_\delta \end{cases} \tag{7}$$

where u means the grey wolves' position before it begin to encircling the prey, u_α, u_β and u_δ is respectively the position of α, β, δ wolf. p_α, p_β, and p_δ is measured by the formula of Euclidean distance and denotes the relationship between the α, β, δ wolf and the ω wolf, respectively. After each search of the grey wolf hunting, the individual location is altered by Eq. (8) to better approach the prey for hunting. The pseudocode algorithm is expressed by the Algorithm 2.

$$u\left(t+1\right)=\frac{u_1+u_2+u_3}{3} \tag{8}$$

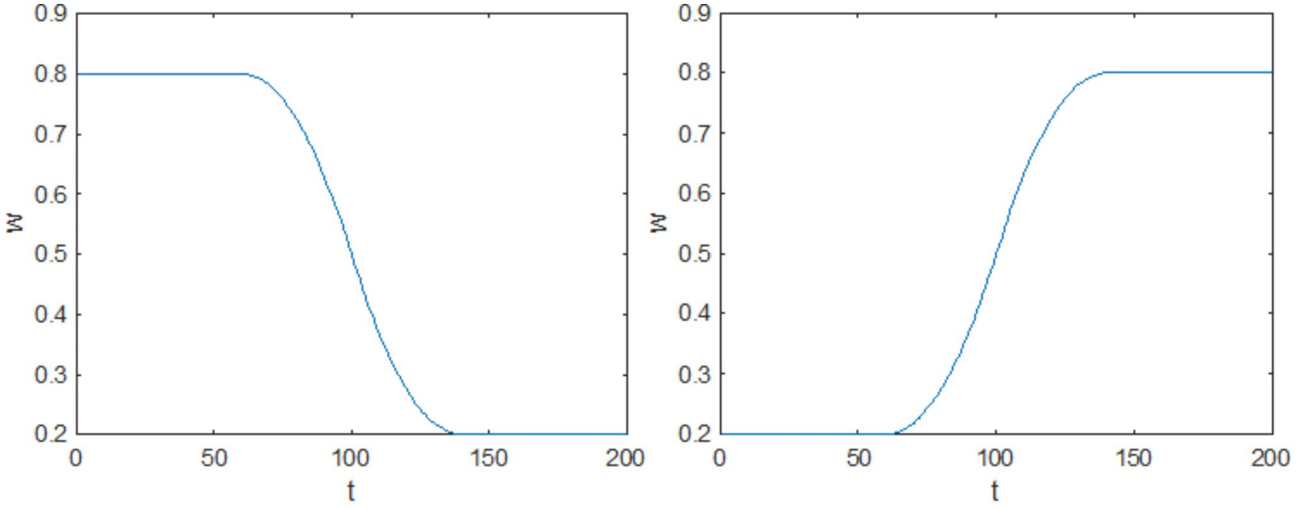
The improved GWO algorithm based on fuzzy membership functions

Aiming to reach a better balance improvement of GWO algorithm on the global search and local exploration, the ZMF[58,59]is adopted as the inertia weight and the SMF[59] is utilized as adaptive weight to improve the grey wolf search process respectively, which is called ZGWO and SGWO. And then, combining the ZMF and the SMF together to enhance the grey wolf search algorithm, named SZGWO, the formula of ZMF is given by Eq. (9).

$$Z\left(x;i,j\right)=\begin{cases} 1\ , & x\leqslant i \\ 1-2\left(\frac{x-i}{j-i}\right)^2, & i\leqslant x\leqslant\frac{i+j}{2} \\ 2\left(\frac{x-j}{j-i}\right)^2, & \frac{i+j}{2}\leqslant x\leqslant j \\ 0\ , & x\geqslant j \end{cases} \tag{9}$$

where i is 40, j is 160, x lies between i and j. The value of i and $j$are determined based on the characteristics of the ZMF[58,59]. $Z(x;\ i,\ j)$ express the inertia weight. Figure 1 describes the inertia weight graph of ZMF. As seen in Fig. 1 (a), the ZMF function initially has a relatively large value, which gradually decreases with the increase in the number of iterations, ultimately approaching a minimum value. This enables the algorithm to conduct expansive searches in the early phases, being favorable for the algorithm to seek the finest value in global area and preventing it from falling into local optima. As the search progresses, the weight is systematically reduced, as well the algorithm search speed is gradually decreased and stabilized in a very small value with the number of iteration increasing, aiding the algorithm to hunt for the best solution locally and obtain the best solution in a smaller optimization space. Through setting the inertia weight of ZMF, the global development and the local exploration of the search for optimal ability are balanced, which not only accelerate the convergence rate but also guarantee the accuracy of the algorithm.

Additionally, it is also very important for grey wolves to set search steps during their search. If a fixed step size is used for searching, it may cause the grey wolf to separate the optimal solution from all solutions and reduce the convergence search speed under large search steps. otherwise, when the search step size is set too small, it may lead to the grey wolf stagnating the local position prematurely and decreasing the accuracy of the solution. If a random search step size is used for searching, it may lead to a smaller step size in the pre-search period of the grey wolf and directly fall into the local extreme point, and a larger step size during the late stage of the search



(**a**) The graph of ZMF function (**b**) The graph of SMF function

Fig. 1. The graph of ZMF and SMF functions.

so that allowing the algorithm to be far away the optimal solution. Therefore, the SMF is applied as an adaptive weight value to flexibly set the search step size for grey wolf search in this paper by using the typical traits of the SMF that the early value increases rapidly and the late value increases slowly, which make the algorithm search quickly by using a bigger step at the previous search stage and quickly approaches the best solution. At the later search period, the step size is decreased to make it gradually approach the optimal solution for precision search. the formula of SMF is shown in Eq. (10):

$$S\left(x;i,j\right)=\begin{cases} 0\ , & x\leqslant i \\ 2\left(\frac{x-i}{j-i}\right)^2, & i\leqslant x\leqslant\frac{i+j}{2} \\ 1-2\left(\frac{x-b}{j-i}\right)^2, & \frac{i+j}{2}\leqslant x\leqslant j \\ 1\ , & x\geqslant j \end{cases} \tag{10}$$

where i is 60, j is 140. x lies between i and j. The value of i and $j$are determined based on the characteristics of the SMF[58]. $S(x; i, j)$ is an adaptive weight value. Figure 1(b) shows the weight graph of SMF.

The position of the individual gray wolf is altered by the ZMF and the SMF together according to Eq. (11). This improved strategy is called SZGWO, and the pseudocode algorithm is expressed by the Algorithm 1.

$$\begin{cases} u_1=S\left(x;i,j\right)\cdot\left[Z\left(x;i,j\right)\cdot u_\alpha-b_1\cdot p_\alpha\right] \\ u_2=S\left(x;i,j\right)\cdot\left[Z\left(x;i,j\right)\cdot u_\beta-b_2\cdot p_\beta\right] \\ u_3=S\left(x;i,j\right)\cdot\left[Z\left(x;i,j\right)\cdot u_\delta-b_3\cdot p_\delta\right] \end{cases} \tag{11}$$

Initialize the population $u_i(i=1,2,\ldots,n)$, parameters b, k, A
Calculate the fitness value of each individual wolf
Select u_α ←the best fitness wolf u_β ←the second fitness wolf u_δ ←the third fitness wolf
While $(t < T_{\max})$
  for each individual wolf
    Update the current grey wolf position by Eq.(8), (9), (10) and (11)
  end for
  Update b, k, A
  Calculate the fitness values of all grey wolves
  Update u_α, u_β, u_δ
  $t \leftarrow t+1$
End while

Algorithm 1. The SZGWO pseudocode algorithm.

Construction of the SZGWO-ELM network model

ELM[60], a single hidden layer feedforward neural network, is composed of three layers including the data input layer, the hidden layer serves as a space for random mapping to process intermediate data from the input layer, and the output layer outputs the classification categories. The ELM network employs random input layer weights and biases, and solves for the weights connecting the hidden layer and the output layer by minimizing the approximate square error, and utilizes the Moore-Penrose pseudo-inverse to compute the output weights, and finally computes the network's output to accomplish data classification. Meanwhile, the ELM network has many advantages, such as fewer training parameters, fast learning speed, and strong generalization capability, and being widely applied in data classification and prediction.

Input: UCI disease datasets
Output:classification accuracy

Data pre-processing: normalization, sample partition(90% as train data, 10% as test data)
Initialize ELM network parameters: the number of hidden layer nodes.
Initialize the population u_i(i=1,2,…,n), $T_{max,}$ the search upper and lower bounds, b, k, A, the position of α, β, and δ wolf, the positions of search wolves.
While ($t < T_{max}$)
  for i=1: u_i
    the position of wolf > search upper bound
    adjusting the position of wolf to the upper bound
    the position of wolf < search lower bound
    Adjusting the position of wolf to the lower bound
   Calculate the fitness value of each individual wolf
    Select u_α ←the best fitness wolf
      u_β ←the second fitness wolf
      u_δ ←the third fitness wolf
Update the current grey wolf position by using ZMF and SMF base on Eq.(8), (9), (10) and (11)
    end if
  end for
   Update b, k, A
   Calculate the fitness values of all grey wolves
   Update u_α, u_β, u_δ
    The gbest solution = the position of u_α
    The gbest_fitness = the fitness of u_α
   Using the gbest solution as the optimal weights and biases to train network
   $t \leftarrow t$+1
End while
    The best solution = the position of u_α
    The best_fitness = the fitness of u_α
Using the best solution as the optimized network model

Algorithm 2. The SZGWO-ELM model pseudocode.

Suppose there are N samples of heart disease (y_i,t_i), y_i represents n-dimensional features, and t_i is the category of heart disease, where 1 indicates the presence of heart disease. The ELM[60] neural network can be represented as Eq. (12).

$$\sum_{\mathrm{i}=1}^{L} \beta_i g\left(\omega_i \cdot y_j + b_i\right) = t_j \qquad j = 1, 2, \cdots n \tag{12}$$

Where g(y) is activation function, i denotes the hidden layer, w_i represents the input weights, β_i indicates the output weights, and b_i is the bias, · denotes the dot product. Since ELM network randomly selects input weights and biases, the inability to adjust the hidden layer of the network leads to instability. Continuously increasing hidden layer neurons to improve training accuracy may cause overfitting, reducing the model's generalization ability. Therefore, this paper utilizes an improved SZGWO algorithm to search for the optimal solution for the randomly selected parameters in ELM, thereby constructing the SZGWO-ELM network model. The objective of this network model can be formulated as Eq. (13).

$$\sum_{j=1}^{m}\left\|t_j-\hat{t}_i\right\|=0 \tag{13}$$

Where t_j denotes the predicted value of the network output, y_i represents the actual value, and m indicates the number of samples being tested. The Eq. (12) can be represented as $H\beta = T$, H stands for the output of the hidden layer nodes, β represents the output weight, and T signifies the desired output, the best input weights w_i and hidden layer bias b_i can be obtained via the SZGWO algorithm to establish H. Consequently, the output weight β can be represented as $\hat{\beta} = H^{\dagger}T$, $H^{\dagger}$ is the g Moore-Penrose pseudo-inverse matrix of H. According to the above description, the SZGWO algorithm is employed to optimize parameters of the ELM network model and the specific processes of the SZGWO-ELM network model are given as algorithm 2.

Experiment result and analysis

Classical benchmark test functions

In this segment, 24 various classical benchmark test functions selected form IEEE CEC 2022 and literature[61], encompassing both unimodal functions and multimodal functions, are chosen to verify the proposed improvement SZGWO algorithm. In addition, the parameter of D is set under low dimension, normal dimension and high dimension, which are set as 20, 200, and 500, respectively, and the grey wolf population size is fixed at 20, and T_{max} is 200. The relevant details are provided in Table 1.

As shown in Table 1, there are a total of 12 unimodal functions, which are respectively f_1-f_5, f_9, f_{10}, f_{12}, f_{14}, f_{16}, f_{17}, and f_{23}, while the rest 12 functions are multi-modal functions.

Experimental organization and arrangement

In order to validate the effectiveness of the proposed SZGWO in this paper, several experiments were organized as follows: (1) A comparative analysis with ablative experiments was conducted to validate the effectiveness of SZGWO by utilizing fuzzy functions (ZMF and SMF) on the 24 test functions. (2) Nine different intelligent optimization algorithms were used to compare the superiority of SZGWO algorithm. (3) Comparison with optimized- improvement algorithms on fuzzy functions. (4) Four different weight modification strategies for GWO were compared with the SZGWO to validate its effectiveness. (5) Experiment on the selection of parameters for the ELM network is conducted. (6) The comparative analysis between the SZGWO-ELM model and the GWO-ELM model on the five medical disease datasets. (7) Performance comparison on different classify model.

Data preprocessing

Five UCI datasets were conducted to experiment in our paper, including three heart datasets (ceveland, heart prediction, and heart statlog) and two kidney datasets (kidney and KID new). Where the kind of heart datasets are made up of a total of 13 features and 1 target variable, the kidney dataset is composed by 24 features and 1 target variable, and the KID new is consist of a number of 13 features and 1 target variable. In terms of data volume, the three heart datasets contain 303 rows, 270rows, and 270 rows, respectively. The two kidney datasets have 397 rows and 400 rows, respectively. To ensure the validity of the experimental result, we imputed the mean in place of the null values for missing values in the raw data. For example, in the cleveland dataset, the “ca” variable has 4 missing values. We use the average of that column’s features to fill in these 4 missing values, as this approach can increase the variance of the dataset and provide more accurate results. If we were to remove these 4 rows instead, it might introduce additional bias in subsequent experiments.Additionally, categorical variables such as “sex” and “cp” were converted into numerical values for ease of computation. The gender “male” is converted to 1, while “female” is represented as 0. All features were then normalized by Eq. (14) to a range of 0 to 1 to ensure consistent computation across datasets. Meanwhile, using this normalization method can prevent issues of vanishing or exploding gradients during the network training process.

$$x^{'}=\frac{x-x_{min}}{x_{max}-x_{min}} \tag{14}$$

where x_{min} and x_{max} are the minimun and maximun values of the feature, respectively.

Analysis and evaluation of the experimental results

Assessing the improved performance on fuzzy functions

In order to validate the efficiency of the improvement algorithms on fuzzy functions, this paper conducted an ablative experiment to pinpoint the most effective improvement strategy, which utilized the mean and standard deviation to test the original GWO, the improved algorithm ZGWO with only ZMF, the improved algorithm SGWO with only SMF, and improved algorithm SZGWO combing ZMF with SMF together, respectively. The experiment was organized three kinds according to the low dimension(D = 20), normal dimension(D = 200), and high dimension(D = 500), each kind of dimension was executed 30 trials.

Table 2 presented the mean and standard deviation of 24 test functions in low dimension(D = 20) under the original GWO, ZGWO, SGWO and SZGWO algorithms, respectively. It can be discovered form Table 2 that the SZGWO algorithm consistently achieves optimal solutions of 0 for both mean and standard deviation across the majority of functions. Exceptions include f_2, f_4, and f_5, where the mean and standard deviation of

Function	Benchmark	Function	Benchmark
f_1	$f_1(x) = \sum_{i=1}^{n} {x_i}^2$	f_{13}	$f_{13} = \sum_{i=1}^{n} x_i^2 + \left(\sum_{i=1}^{n} 0.5ix_i\right)^2 + \left(\sum_{i=1}^{n} 0.5ix_i\right)^4$
f_2	$f_2(x) = \sum_{i=1}^{n} \lvert x_i \rvert + \prod_{i=1}^{n} \lvert x_i \rvert$	f_{14}	$f_{14} = (x_1 - 1)^2 + \sum_{i=2}^{n} i\left(2x_i^2 - x_{i-1}\right)^2$
f_3	$f_3(x) = \sum_{i=1}^{n} \left(\sum_{j-1}^{i} x_j\right)^2$	f_{15}	$f_{15} = \sum_{i=1}^{n/4} \begin{bmatrix} (x_{4i-3} + 10x_{4i-2})^2 + 5(x_{4i-1} - 4x_i)^2 + \\ (x_{4i-2} - 2x_{4i-1})^4 + 10(x_{4i-3} - x_{4i})^4 \end{bmatrix}$
f_4	$f_4(x) = \max_i \{\lvert x_i \rvert, 1 \leqslant i \leqslant n\}$	f_{16}	$f_{16} = x_1^2 + 10^6 \sum_{i=2}^{n} x_i^2$
f_5	$f_5(x) = \sum_{i=1}^{n} ix_i^4 + random[0, 1)$	f_{17}	$f_{17} = 0.5 + \frac{\sin^2\left(x_1^2 + x_2^2\right) - 0.5}{\left[1 + 0.001\left(x_i^2 + x_{i+1}^2\right)\right]^2}$
f_6	$f_6(x) = \sum_{i=1}^{n} \left[x_1^2 - 10\cos(2\pi x_i) + 10\right]$	f_{18}	$f_{18} = x_1^2 + 2x_2^2 - 0.3\cos(3\pi x_1) \cdot 0.4\cos(4\pi x_2) + 0.3$
f_7	$f_7(x) = -20\exp\left(-0.2\sqrt{\frac{1}{n}\sum_{i=1}^{n} x_i^2}\right) - \exp\left(\frac{1}{n}\sum_{i=1}^{n}\cos(2\pi x_i)\right) + 20 + e$	f_{19}	$f_{19} = x_1^2 + 2x_2^2 - 0.3\cos(3\pi x_1) - 0.4\cos(4\pi x_2) + 0.7$
f_8	$f_8(x) = \frac{1}{4000}\sum_{i=1}^{n} x_i^2 - \prod_{i=1}^{n}\cos\left(\frac{x_i}{\sqrt{i}}\right) + 1$	f_{20}	$f_{20} = x_1^2 + x_2^2 + 25\left(\sin^2(x_1) + \sin^2(x_2)\right)$
f_9	$f_8(x) = \frac{1}{4000}\sum_{i=1}^{n} x_i^2 - \prod_{i=1}^{n}\cos\left(\frac{x_i}{\sqrt{i}}\right) + 1$	f_{21}	$f_{21} = \left(2x_1^3x_2 - x_2{}^3\right)^2 + \left(6x_1 - x_2^2 + x_2\right)^2$
f_{10}	$f_{10} = \sum_{i=1}^{n} ix_i^2$	f_{22}	$f_{22} = 1 - \cos\left(2\pi\sqrt{\sum_{i=1}^{n} x_i^2}\right) + 0.1\sqrt{\sum_{i=1}^{n} x_i^2}$
f_{11}	$f_{11} = 0.1\left\{\sin^2(3\pi x_1) + \sum_{i=1}^{n}(x_i - 1)^2\left[1 + \sin^2(3\pi x_i + 1)\right] \cdot (x_n - 1)^2\left[1 + \sin^2(2\pi x_n)\right]\right\} + \sum_{i=1}^{n} u(x_i, 5, 100, 4)$	f_{23}	$f_{23} = 7x_1^2 - 6\sqrt{3}x_1x_2 + 13x_2^2$
f_{12}	$f_{12} = 0.26(x_1^2 + x_2^2) - 0.48x_1x_2$	f_{24}	$f_{24} = 100\sqrt{\Vert x_2 - 0.01x_1^2 \Vert} + 0.01\Vert x_1 + 10 \Vert$

Table 1. 24 classical benchmark test functions.

the SZGWO algorithm reach the minimum values rather than the optimal solutions. On f_6, f_8, f_{17}, f_{18}, and f_{19}, both the ZGWO and the SGWO algorithm also reach optimal solutions, but across most other functions, the ZGWO and the SGWO display mean and standard deviation values inferior to those of the SZGWO. The largest discrepancy between ZGWO and SZGWO is observed in mean, reaching a maximum of 6.38E-04, and in standard deviation, reaching 4.31. Meanwhile, the difference between SGWO and SZGWO is 0.1 in mean and 0.4 in standard deviation. In comparative terms, ZGWO outperforms SGWO, as ZGWO maintains a balance between the algorithm's global and local optimization capabilities, while SGWO rapidly adjusts the search step size to reach optimal solutions. The original GWO, without any enhancement, consistently demonstrates the poorest performance in both mean and standard deviation. Additionally, the experimental results in dimensions of 200 and 500 are similar to those in Table 2, further indicating that using ZMF as inertia weights and SMF as adaptive weights can balance the outstanding search capabilities of global exploitation and local exploration, reduce blindness, enhance the algorithm's search speed, avoid the algorithm getting stuck in the optimal stagnant state during local search, and improve the algorithm's solution accuracy.

Functions	GWO		ZGWO		SGWO		SZGWO	
	Mean	Std	Mean	Std	Mean	Std	Mean	Std
f_1	1.05E-05	1.43E-05	2.17E-148	3.03E-148	9.65E-119	1.66E-118	**3.58E-256**	**0**
f_2	2.58E-04	1.15E-04	4.23E-76	2.65E-76	2.72E-58	4.70E-58	**2.11E-128**	**2.93E-128**
f_3	5.8405	6.5937	4.15E-139	4.56E-139	1.64E-118	2.84E-118	**0**	**0**
f_4	1.41E-02	6.11E-03	8.11E-74	9.06E-74	3.22E-60	5.36E-60	**7.32E-128**	**1.22E-127**
f_5	1.01E-02	4.54E-03	6.29E-04	5.27E-04	5.73E-04	3.69E-04	**2.61E-04**	**2.08E-04**
f_6	1.39E+01	5.2303	**0**	**0**	**0**	**0**	**0**	**0**
f_7	9.01E-04	6.03E-04	1.13E-15	4.10E-16	3.26E-15	1.23E-15	**8.88E-16**	**0**
f_8	5.17E-02	3.35E-02	**0**	**0**	**0**	**0**	**0**	**0**
f_9	2.61E-23	4.46E-23	4.61E-211	**0**	1.50E-155	2.59E-155	**0**	**0**
f_{10}	1.09E-06	1.03E-06	1.56E-149	1.68E-149	3.20E-119	5.29E-119	**0**	**0**
f_{11}	9.52E+01	7.23E+01	3.22	6.92	1.67E+01	3.01	**0**	**2.61**
f_{12}	1.75E-20	2.73E-20	2.45E-168	**0**	2.58E-131	4.24E-131	**0**	**0**
f_{13}	1.05E+01	7.89	2.77E-133	4.66E-133	7.98E-124	1.38E-123	**0**	**0**
f_{14}	7.12E-01	7.74E-02	6.67E-01	**9.58E-06**	9.47E-01	5.36E-02	**0**	7.22E-02
f_{15}	1.61E-04	1.23E-04	3.72E-139	6.42E-139	1.23E-120	2.12E-120	**0**	**0**
f_{16}	4.25E-02	3.86E-02	9.80E-145	1.46E-144	6.33E-115	1.09E-114	**0**	**0**
f_{17}	3.26E-04	5.64E-04	**0**	**0**	**0**	**0**	**0**	**0**
f_{18}	9.99E-17	5.77E-17	**0**	**0**	**0**	**0**	**0**	**0**
f_{19}	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
f_{20}	1.38E-37	2.40E-37	1.10E-194	0	2.23E-137	3.87E-137	**0**	**0**
f_{21}	3.80E-08	3.64E-08	1.01E-20	1.75E-20	2.73E-24	4.73E-24	**0**	**3.67E-31**
f_{22}	3.47E-01	6.93E-02	2.00E-02	2.31E-02	7.17E-02	5.30E-02	**0**	**6.61E-73**
f_{23}	2.85E-27	4.92E-27	6.18E-187	0	3.12E-132	5.40E-132	**0**	**0**
f_{24}	3.46E-01	2.54E-01	2.55E-01	2.68E-01	1.00E-01	**1.70E-17**	**0**	**1.70E-17**

Table 2. The results of different algorithm improvement strategies.

Convergence comparison of different optimization algorithms

To comprehensively assess the efficiency of the SZGWO algorithm presented in this study, nine kinds intelligent optimization algorithms, such as GSA, CDO, GWO, OMA, WDO, MVO, FDO, CDDO and FOX are applied to compare the convergence under 24 test functions in Table 1and on CEC 2019 benchmark functions[36] with three dimensions, and the comparison graphs of average convergence curves are shown at Fig. 2, where the *x*-axis indicates the quantity of iterations, and the *y*-axis denotes the average fitness value of the objective algorithm. The parameters of the nine optimization algorithms are given in Table 3. The population value was set from 5 to 50 and the number of iterations were set from 20 to 400 to get the best value in experiment. According to experiment results, we selected the population and iteration number value shown in Table 3.

Figure 2 illustrates the comparison of average convergence curves for three unimodal (f_1, f_{17}, and f_{23}) and three multimodal functions (f_6, f_8, and f_{18}) in low dimension (D = 20), it is evident from Fig. 2 that the SZGWO algorithm finds the optimal value at the 13th iteration in f_{17}, while the GWO algorithm achieves this at the 31st iteration. For multimodal function 6, 8 and 18, the SZGWO algorithm discovers the optimal value at the 24th, 28th and 19th iteration in turn. The CDO algorithm reaches the best value at the 137th iteration in f_8. In function 18, the GSA and OMA algorithms achieve the optimal solution at the 146th and 130th iterations, respectively. While the optimal solution is not found within the given 200 iterations for the remaining functions. Figure 2 indicate that whether for unimodal or multimodal functions, the SZGWO algorithm exhibits the best convergence performance and the fastest convergence speed. Particularly in the case of multimodal functions, the SZGWO algorithm has the most excellent convergence performance under low dimension. In addition, the SZGWO algorithm has the best performance on the convergence curve under the 200 and 500 dimensions. Regardless of whether the dimension is low or high, it can search for the optimal solution in a short time, followed by CDO and GWO. However, the other four algorithms find it difficult to find the optimal solution in a given 200 iterations and gradually fall into local optima, This once again proves that the SZGWO algorithm has a significant improvement in search speed and solution accuracy under both unimodal and multimodal functions.

To better validate the effectiveness of the SZGWO algorithm, CEC 2019 benchmark test functions[36] were used to conduct comparative experiments between the new algorithms FDO, FOX, CDDO, and the SZGWO algorithm. The parameter settings of these algorithms are shown in Table 3. The performance of the four optimization algorithms are presented in Table 4.

As shown in Table 4, apart from CEC-f_2, CEC-f_3, and CEC-f_6, the mean value and standard deviation of the SZGWO algorithm are the best, followed by FOX algorithm, which performs similarly to SZGWO algorithm. The performance of the FDO and CDDO algorithms is slightly inferior to SZGWO and FOX algorithms. Notably, on CEC-f_7 and CEC-f_8, the performance of all four algorithms is identical. In terms of runtime complexity, the execution times of SZGWO, FOX, and CDDO are 0.611s, 0.709s, and 0.897s, respectively, with little difference

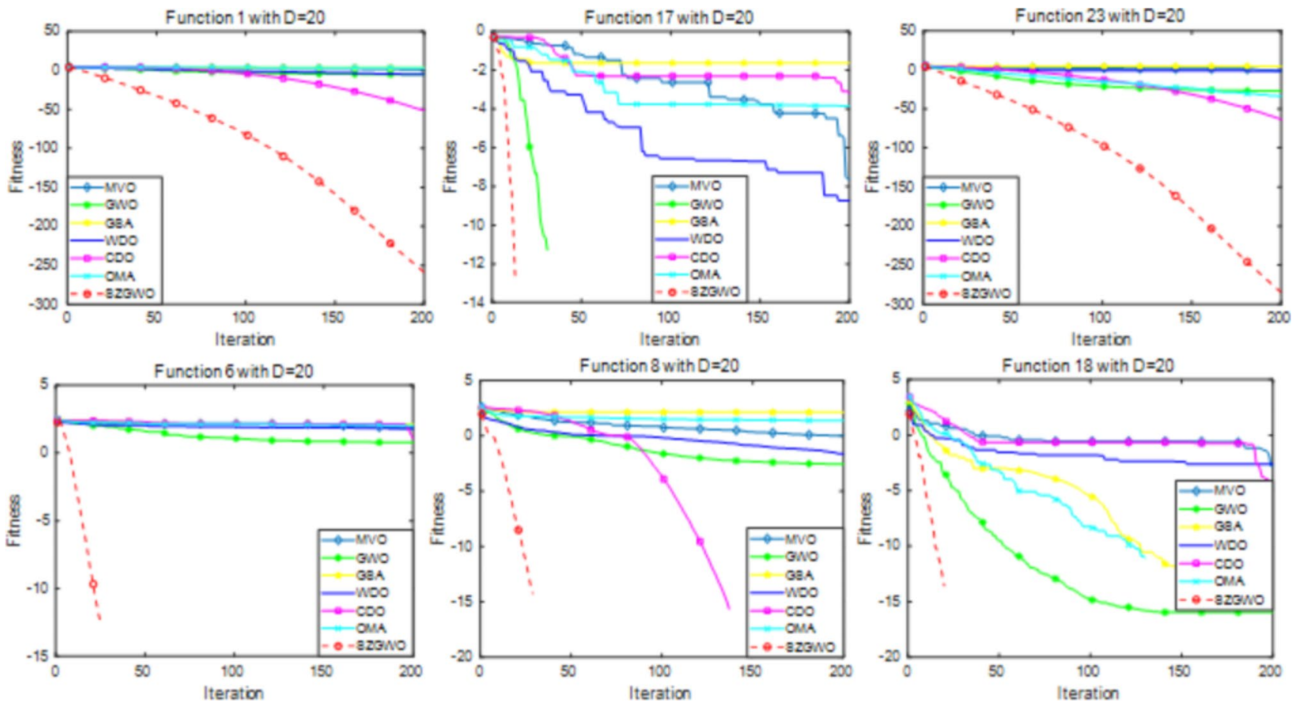


Fig. 2. Average convergence curves on test functions with D = 20.

Algorithm	Parameter configuration
GSA	Population = 20, T_{max}=200, *alfa* = 20,G0 = 100, *rand*=[0,1],*final-per* = 2,
CDO	Population = 20, T_{max}=200, *loop conter* = 0
OMA	Population = 20, T_{max}=200, *r*=[0,1]
WDO	Population = 20, T_{max}=200, *g* = 0.2, *c* = 0.4, *alpha* = 0.4, *R**T* = 3
MVO	Population = 20, T_{max}=200, WEP_{min}=0.2, WEP_{max}=1
GWO	Population = 20, T_{max}=200, *A*=[2,0], *R*=[0,1]
SZGWO	Population = 20, T_{max}=200, *A*=[2,0], *R*=[0,1]
FDO	Population = 20, T_{max}=200, *weightFactor* = 0, *pace* = 0,
CDDO	Population = 20, T_{max}=200, *LR* = 0.01, *SR* = 0.9, *PS* = 10, *CR* = 0.1, *ncount* = 0,
FOX	Population = 20, T_{max}=200, c_1 = 0.18, c_2 = 0.82, *jump* = 0, *l* = 0

Table 3. The parameters of different optimization algorithm.

	FDO		CDDO		FOX		SZGWO	
Function	Mean	Std	Mean	Std	Mean	Std	Mean	Std
CEC-f_1	1.0473	0.0719	1179.0673	2473.0306	**1**	**0**	**1**	**0**
CEC-f_2	5.2216	0.1972	5	**0**	5	**0**	**1**	0.0141
CEC-f_3	12.7584	**0.4511**	7.2673	0.9725	9.6520	1.2825	**1**	0.9673
CEC-f_4	**1**	4.37E-08	**1**	7.17E-05	**1**	**0**	**1**	**0**
CEC-f_5	**1**	7.31E-07	1.0081	0.0152	**1**	6.12E-09	**1**	**0**
CEC-f_6	1.0006	0.0004	1.0088	0.0165	**1.0001**	**0.0002**	1.0009	0.0005
CEC-f_7	**1**	**0**	**1**	**0**	**1**	**0**	**1**	**0**
CEC-f_8	**1**	**0**	**1**	**0**	**1**	**0**	**1**	**0**
CEC-f_9	**1**	**0**	1.0198	0.0768	**1**	**0**	**1**	**0**
CEC-f_{10}	1.0003	0.0002	**1**	**0**	**1**	**0**	**1**	**0**

Table 4. The performance of different optimization algorithm tested on CEC benchmark 2019.

among them, whereas FDO takes 15.436s, requiring significantly more computational resources. The results in Table 4 indicate that the proposed SZGWO algorithm demonstrates good performance compared to these newer algorithms.

Comparison with optimized- improvement algorithms on fuzzy functions
Additionally, for further validation of the superiority of the SZGWO algorithm proposed in this paper, the ZMF and SMF were applied to optimize the MVO, GSA, WDO, and CDO algorithms, and their performance was compared with the proposed SZGWO. Mean and standard deviation were chosen as the comparison metrics. As shown in Table 5, the SZGWO algorithm exhibits the best performance in both mean and standard deviation. While the SZGSA algorithm and the SZCDO algorithm achieve optimal values of 0 in f_6, f_8, f_{17}, f_{18}, and f_{19}, as well as a mean value of 0.1 in f_{24}, their performance is lower than SZGWO in other functions. In contrast, both the SZMVO algorithm and the SZWDO algorithm show inferior performance compared to the SZGWO algorithm. This experimental result once again emphasizes the exceptional performance of the SZGWO algorithm proposed in this paper, highlighting the positive impact of incorporating ZMF and SMF for GWO enhancement.

Comparison of improvement performance of different weighting strategies
The average value and standard deviation are taken to better assess the improvement performance with the different weighting strategies under the population search dimensions of 20, 200 and 500 dimensions, respectively. The improved effect of this paper is compared with several different weight-improved strategies of the DGWO[62], the ERGWO[63], the IGWO[64], and the FIGWO[65], and the experiment results can be found in Table 6.

Table 6 presented the above four different improvement strategies on the GWO under the 200 dimension testing on the 24 functions. From Table 6, the SZGWO algorithm obtained the highest average value and standard deviation on most test functions, except testing on the f_{11} and f_{14}. On these two functions, the mean of the SZGWO is better than the ERGWO, but the standard deviation of SZGWO is a little lower compared with the ERGWO, which is 1.663 on f_{11} and 0.06E-04 on f_{14}. Apart from these two improvement strategies, the mean and standard deviation of the remaining algorithm improvement strategies are poor than the former. The difference of the mean value is very large particularly on the function 2, which is 1.66E + 48 in IGWO and 2.7E-10 in SZGWO, so as to the standard deviation. The SZGWO outperforms the other four weight improvement strategies in both 20 dimensions and 500 dimensions. This highlights the robust and stable nature of SZGWO, showcasing its ability to achieve favorable outcomes in low, normal, and high-dimensional conditions.

Function	SZMVO		SZGSA		SZWDO		SZCDO		SZGWO	
	Mean	Std	Mean	Std	Mean	Std	Mean	Std	Mean	Std
f_1	3.6283	0.7337	4.46E-17	3.38E-17	1.44E-22	1.34E-22	1.80E-119	3.10E-119	**3.03E-260**	**0**
f_2	1.8238	1.1476	1.91E-08	1.22E-08	4.21E-12	7.30E-13	9.98E-60	1.43E-59	**4.29E-128**	**5.85E-128**
f_3	888.5218	254.1790	1.48E-16	1.16E-16	1.79E-22	2.94E-22	1.20E-86	2.08E-86	**1.98E-257**	**0**
f_4	0.4478	0.2510	3.69E-09	3.24E-09	4.96E-13	2.78E-13	6.98E-61	6.85E-61	**7.79E-129**	**6.92E-129**
f_5	0.2479	0.1243	0.0007	0.0009	0.0051	0.0045	0.0003	0.0001	**0.0001**	**9.48E-05**
f_6	87.1132	15.6925	**0**	**0**	89.6568	16.9785	**0**	**0**	**0**	**0**
f_7	3.0120	0.5603	7.10E-09	8.15E-10	2.50E-12	1.73E-12	3.26E-15	2.05E-15	8.88E-16	0
f_8	0.0138	0.0073	**0**	**0**	0.2959	0.5125	**0**	**0**	**0**	**0**
f_9	2.12E-06	1.27E-06	5.91E-38	1.00E-37	1.66E-16	2.81E-16	7.35E-142	9.82E-142	**8.92E-320**	**0**
f_{10}	0.9290	0.4647	3.30E-16	4.52E-16	8.70E-22	1.28E-21	1.35E-120	1.27E-120	**3.54E-262**	**0**
f_{11}	2.85E + 21	3.78E + 21	**14.7422**	2.6188	-232.4822	181.7163	0.3620	2.5257	18.1445	**0.1139**
f_{12}	2.91E-07	2.87E-07	6.84E-33	7.22E-33	1.12E-13	1.84E-13	2.97E-134	4.72E-134	**1.79E-285**	**0**
f_{13}	30.5916	5.6169	4.77E-16	2.83E-16	5.54E-23	4.71E-23	8.71E-105	1.51E-104	**1.26E-259**	**0**
f_{14}	13.6380	12.0360	0.8358	**0.0593**	0.9460	0.0115	**0.7779**	0.1852	0.9209	0.1362
f_{15}	1.8881	1.2577	1.42E-16	1.81E-16	1.49E-24	1.41E-24	3.73E-122	5.77E-122	**1.26E-262**	**0**
f_{16}	79.4557	58.1403	2.40E-16	1.95E-16	3.43E-22	4.14E-22	6.94E-122	9.27E-122	**2.88E-260**	**0**
f_{17}	1.25E-06	9.89E-07	**0**	**0**	0.0016	0.0028	**0**	**0**	**0**	**0**
f_{18}	0.0073	0.0020	**0**	**0**	0.0754	0.1307	**0**	**0**	**0**	**0**
f_{19}	0.0090	0.0088	**0**	**0**	0.1443	0.2330	**0**	**0**	**0**	**0**
f_{20}	3.1627	5.4780	2.12E-36	3.52E-36	3.56E-10	6.01E-10	2.23E-133	3.64E-133	**8.50E-291**	**0**
f_{21}	1.78E + 08	3.09E + 08	1.27E-32	2.20E-32	0.0493	0.0853	1.65E-130	1.58E-130	**3.89E-221**	**0**
f_{22}	1.1666	0.2517	7.21E-10	4.26E-10	0.2667	0.1529	0.0685	0.0594	**1.88E-73**	**3.26E-73**
f_{23}	0.1324	0.2089	6.02E-36	9.97E-36	1.34E-07	2.32E-07	1.89E-129	3.22E-129	**7.64E-284**	**0**
f_{24}	1.1631	0.6430	**0.1**	1.63E-09	0.2973	0.1614	**0.1**	1.39E-17	**0.1**	**1.70E-17**

Table 5. Performance of different intelligence optimization improved by two fuzzy functions.

Function	DGWO[62]		ERGWO [63]		IGWO[64]		FIGWO[65]		SZGWO	
	Mean	Std	Mean	Std	Mean	Std	Mean	Std	Mean	Std
f_1	2.25E+05	2.05E+04	1.12E-10	4.51E-11	2.30E+05	1.87E+04	6.08E+05	2.17E+04	**1.30E-19**	**6.25E-20**
f_2	5.13E+10	8.88E+10	9.00E-06	1.58E-06	1.66E+48	2.84E+48	1.35E+10	2.33E+10	**2.70E-10**	**1.48E-10**
f_3	1.06E+07	7.73E+06	2.28E-08	3.24E-08	1.53E+06	3.09E+05	4.83E+06	1.11E+06	**4.34E-18**	**6.98E-18**
f_4	9.88E+00	5.15E-02	4.31E-06	3.25E-06	8.16E+00	3.81E-01	9.83E+00	2.17E-02	**1.30E-11**	**6.58E-12**
f_5	8.83E+03	5.48E+02	4.01E-03	1.72E-03	1.84E+03	1.26E+02	8.32E+03	2.63E+02	**2.69E-03**	**2.07E-03**
f_6	3.41E+03	6.13E+01	1.36E-10	7.24E-11	2.44E+03	2.35E+01	3.47E+03	3.33E+01	**0**	**0**
f_7	2.08E+01	3.90E-02	1.01E-06	1.74E-07	1.92E+01	4.24E-01	2.05E+01	3.51E-02	**2.19E-11**	**6.07E-12**
f_8	3.03E+02	1.82E+01	4.67E-11	3.24E-11	2.17E+03	6.36E+01	5.30E+03	1.57E+02	**0**	**0**
f_9	2.24E+00	5.22E-01	4.93E-21	8.52E-21	4.45E-02	2.40E-02	2.27E+00	9.99E-01	**0**	**0**
f_{10}	6.23E+05	2.00E+04	1.32E-10	6.52E-11	1.93E+05	7.17E+03	5.81E+05	2.90E+03	**0**	**0**
f_{11}	4.41E+04	4.12E+04	1.99E+02	**9.37E-01**	2.88E+04	2.78E+03	7.09E+04	4.92E+03	**0**	2.60
f_{12}	6.87E-01	3.27E-01	3.85E-20	4.36E-20	1.22E-03	1.75E-03	1.68E-04	2.90E-04	**0**	**0**
f_{13}	5.61E+11	9.69E+11	8.20E-01	1.42E+00	4.17E+03	9.66E+02	1.42E+06	1.87E+06	**0**	**0**
f_{14}	1.24E+08	1.32E+07	1.00E+00	**1.04E-04**	2.79E+07	6.39E+06	1.34E+08	4.75E+06	**0**	2.10E-04
f_{15}	4.12E+05	4.96E+04	5.31E-12	2.14E-12	7.10E+04	1.52E+04	4.51E+05	6.02E+04	**0**	**0**
f_{16}	6.13E+09	2.41E+08	8.18E-07	3.27E-07	2.37E+09	3.53E+08	6.15E+09	8.53E+07	**0**	**0**
f_{17}	3.12E-01	2.26E-01	0.00E+00	0.00E+00	1.19E-01	1.97E-01	4.97E-01	4.14E-03	**0**	**0**
f_{18}	9.38E+01	1.30E+02	7.40E-17	3.20E-17	8.74E-03	8.73E-04	2.48E+03	3.59E+03	**0**	**0**
f_{19}	1.92E+02	1.69E+02	2.22E-16	2.22E-16	5.83E-01	1.44E-01	6.55E+02	7.99E+02	**0**	**0**
f_{20}	1.17E+01	8.40E+00	2.07E-18	3.22E-18	1.06E-01	1.79E-01	6.66E+00	8.29E+00	**0**	**0**
f_{21}	5.42E+07	8.61E+07	6.13E-14	1.06E-13	8.58E+01	1.35E+02	2.11E+16	3.56E+16	**0**	**0**
f_{22}	4.74E+01	4.84E+00	1.31E-01	2.67E-02	4.91E+01	1.65E+00	7.78E+01	1.83E+00	**0**	**4.68E-59**
f_{23}	4.44E+02	4.01E+02	9.62E-17	6.18E-17	9.47E+00	9.04E+00	5.29E+04	8.99E+04	**0**	**0**
f_{24}	6.53E+01	5.30E+01	1.38E-01	6.59E-02	4.56E-01	7.08E-02	3.25E+01	4.31E+01	**0**	**0**

Table 6. Performance comparison of different weight improvement strategies with D = 200.

Parameters of ELM neural network

In this paper, the GWO optimization algorithm is employed to select the optimal parameters for the ELM[60] network model. The iteration is varied from 0 to 500, the number of neurons in the hidden layer is explored from 3 to 20, the grey wolf population ranges from 5 to 20, and the algorithm is executed 30 times to explore the parameter space of the ELM network model. Figure 3 illustrates the objective function values for each run of the algorithm, and the optimal value is found at the 74th run, where the optimal function value is 0.1153. The best parameters are as follow: the number of neurons in the hidden layer is 10, and the grey wolf population is 9.

Comparison of the prediction performance on five medical disease

In an effort to confirm the introduced SZGWO-ELM model in the real-life applications, five UCI datasets were conducted to experiment, including three heart datasets (ceveland, heart prediction, and heart statlog) and two kidney datasets (kidney and KID new). These datasets are adopted to test for evaluating the prediction effectiveness of the SZGWO-ELM model on the real applications. The model can predict whether a person will suffer from medical disease based on the given features. Each kind of datasets is split into two parts, which one part is the training set randomly selected 90% of the dataset and the other one part is the remaining 10% as testing set in the experiment. The prediction performance is evaluated based on four conventional test metrics, which are precision, sensitivity, specificity and accuracy, respectively.

Figure 4 illustrates the four characteristics performance of the above five datasets under the SZGWO-ELM model and GWO-ELM model. It can be clearly seen from Fig. 4(a) that the efficiency of the SZGWO-ELM model has a significant advantage compared with the GWO-ELM model on test data, especially when testing on the former three heart disease datasets. In the Cleveland dataset, for the accuracy attribute, the SGZWO-ELM model achieved a value of 96.08%, which is 7.5% higher than that of the GWO-ELM model. The SGZWO-ELM model is 10.4%, 7.56% better than the GWO-ELM model on the sensitivity and precision attribute. The specificity value for the SGZWO-ELM model is 99.26%, surpassing the GWO-ELM model's 93.96%. When testing on the later two kidney diseases datasets, the SZGWO-ELM model outperforms the GWO-ELM model slightly, showing an improvement of around 7.5% in performance. From the experimental comparison Fig. 4(b) on the train data, it is evident that SZGWO-ELM model performs slightly better than GWO-ELM model when testing on the two kidney datasets. In contrast, the SZGWO-ELM model outperformed the GWO-ELM model on the Cleveland dataset for the four attributes: precision by 7.86%, sensitivity by 13.88%, specificity by 14.05%, and accuracy by 9.03%. Overall, the SZGWO-ELM model shows relatively favorable performance across these five medical disease datasets.

For the purpose of further evaluating effectiveness about the proposed SZGWO-ELM model, Fig. 5 showed the ROC curve and AUC value of ceveland dataset under SZGWO-ELM model and GWO-ELM model. As

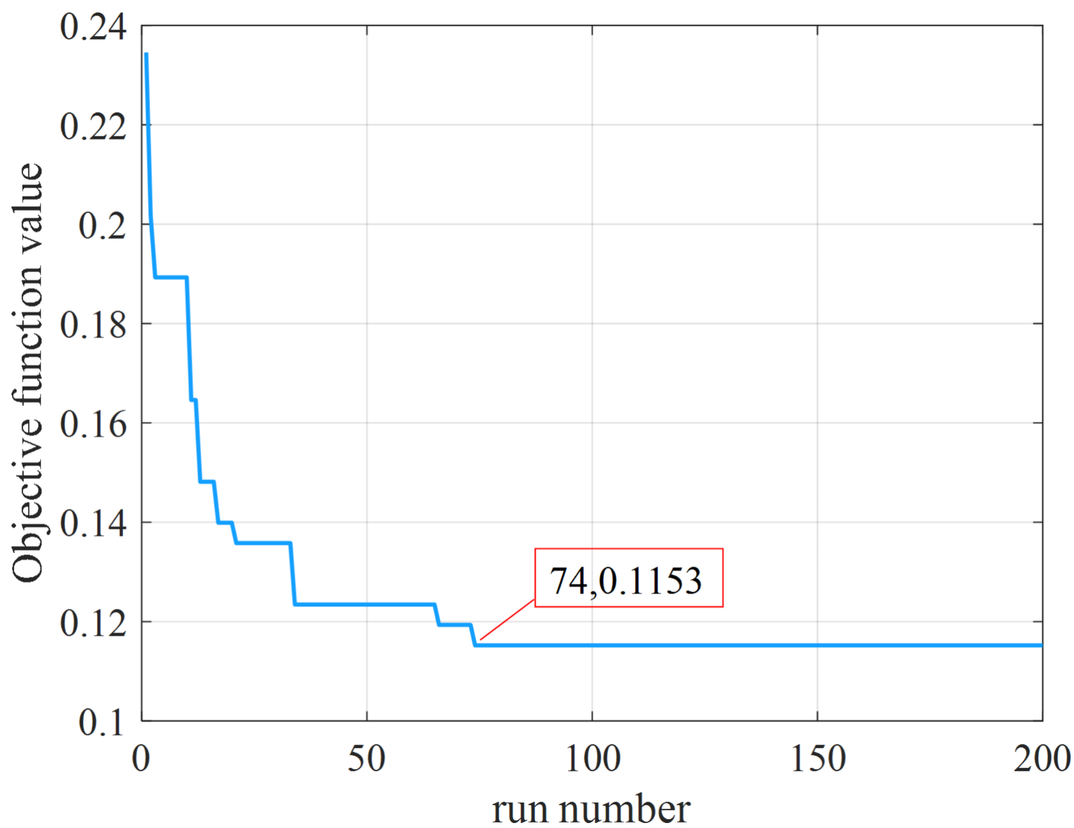
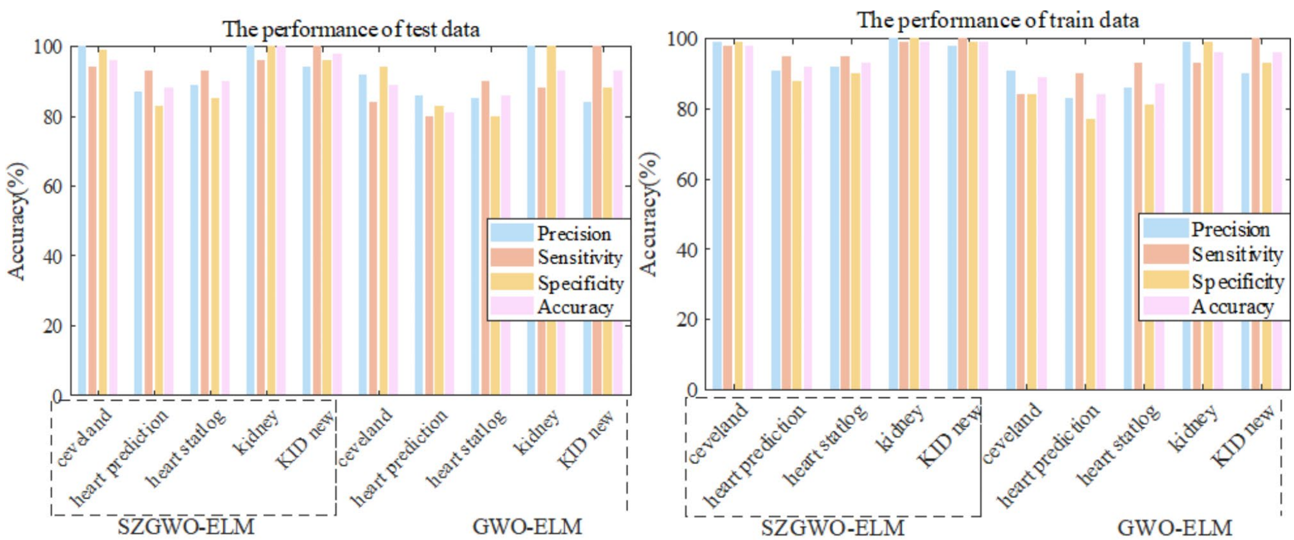


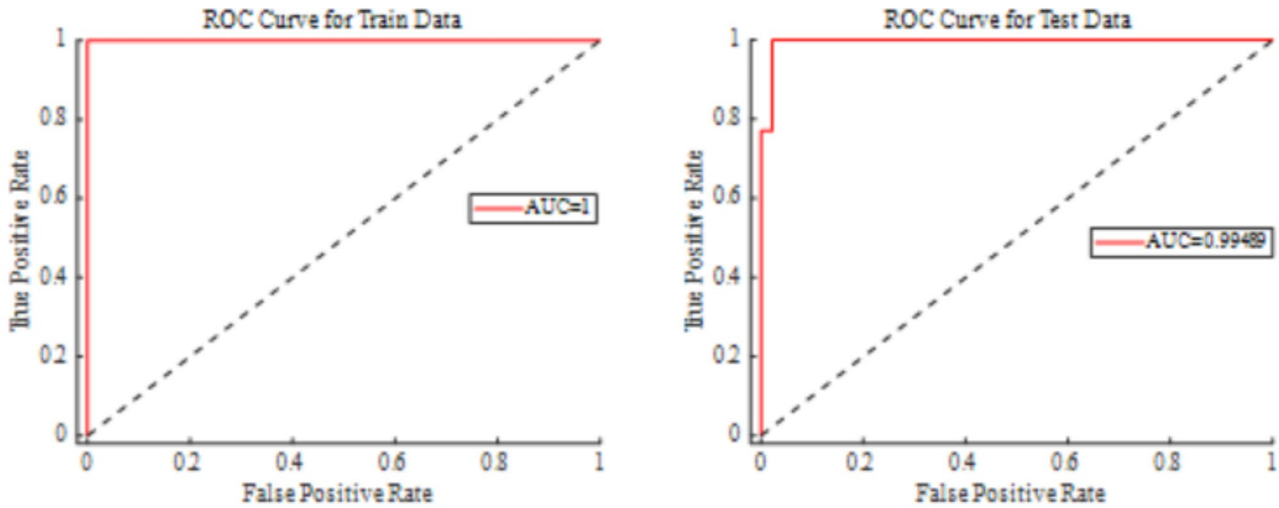
Fig. 3. The objective function value of ELM neural network with the increase run number .



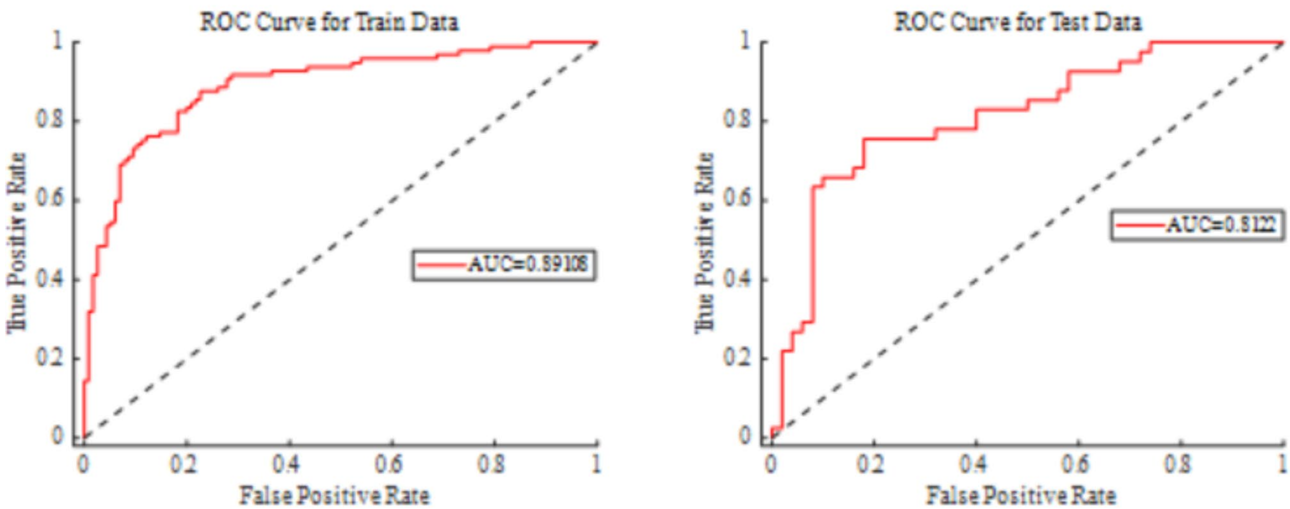
(**a**) Classification accuracy on test data (**b**) Classification accuracy on train data

Fig. 4. The classification accuracy using five datasets on SZGWO-ELM and GWO-ELM model.

shown in Fig. 5, the AUC value of SZGWO-ELM model is 10.8% better than the GWO-ELM model when testing on the train datasets, in which the AUC value is 1 in SZGWO-ELM model and 0.891 in GWO-ELM model on ceveland datasets, and the AUC value is 0.967, 0.948, 0.999, 1 in SZGWO-ELM model and 0.934, 0.914, 0.992, 1 respectively in GWO-ELM model on the other datasets in turn. But there is a great difference when testing on the test datasets, especially on the former three heart datasets, in which the difference of AUC value between the two model is 0.006, 0.033, 0.034 in sequence, thus once again point out that the proposed SZGWO-ELM model has an excellent classification performance and can accurately classify.



(a) The ROC curve and AUC value test on SZGWO-ELM model



(b) The ROC curve and AUC value test on GWO-ELM model

Fig. 5. The ROC curve and AUC value testing on ceveland dataset under SZGWO-ELM and GWO-ELM model.

Performance comparison on different classify model

Moreover, the proposed SZGWO-ELM model are compared with the other six classify model to further verify the classify performance. The parameters of the different classify model are configured by Table 7.

Figure 6(a) presents the experimental comparison results of six classification models: FDO-MLP, GWO-MLP, SZGWO-MLP, FDO-ELM, GWO-ELM, and SZGWO-ELM. It can be seen that FDO-MLP, SZGWO-MLP, and SZGWO-ELM perform quite well, with SZGWO-ELM showing only 3.5% and 7.72% improvements over SZGWO-MLP and FDO-MLP, respectively. The next best is FDO-ELM, achieving a classification accuracy of 88.54%. In contrast, the GWO algorithm, whether applied to MLP or ELM, yields the poorest results, with an accuracy of only around 84.34%. This indicates that the proposed algorithm demonstrates better classification performance.

Additionally, we further evaluated the performance of the proposed method using five characteristics (sensitivity, specificity, precision, accuracy, and AUC) across five classification models (DNN, KNN, SUM, RF, and DT). The comparison result is shown in Fig. 6(b). It can be seen from Fig. 6(b) that the five characteristics results of SZGWO-ELM model are significantly better than the later four model, the precision of the SZGWO-ELM model is respectively 19%, 12%, 15% and 21% higher than the rest four conventional model, the difference of sensitivity is 19%, 12%, 15% and 21% in sequence between the SZGWO-ELM model and the other four conventional model, the specificity of SZGWO-ELM model is 16%, 33%, 13% and 18% larger than the others in turn, the gap of the accuracy between the SZGWO-ELM model and the other four conventional model is 19%, 25%, 19% and 21% in order, and the AUC value of the five different model is respectively 99.9%, 77.7%, 80.9%,

Classify model	Parameter configuration
ELM	The number of neurons in hidden layer = 10, 'Sigmoid', max iteration = 200
DNN	epoch = 1000, learning rate = 0.05, hidden neural node = 20, hidden layer = 5, goal error = 10^{-9},momentum = 0.9,
KNN	P = 2, n-neighbors = 5, 'minkowski'
SVM	Degree = 3, *coef0* = 10, C = 1, gamma = 0.1,'RBF'
RF	n-estimators = 1000,'Gini', max-feature = sqrt(feature)
DT	min-leaf = 1, min-split = 2, max-feature = log2(feature), splitter='best'
MLP	Runno = 5, pop = 20, max iteration = 200, Ino = 13, Hno = 27, Ono = 1, dim = 406, *ub* = 3, *lb*=-3

Table 7. The parameters of five different classify model.

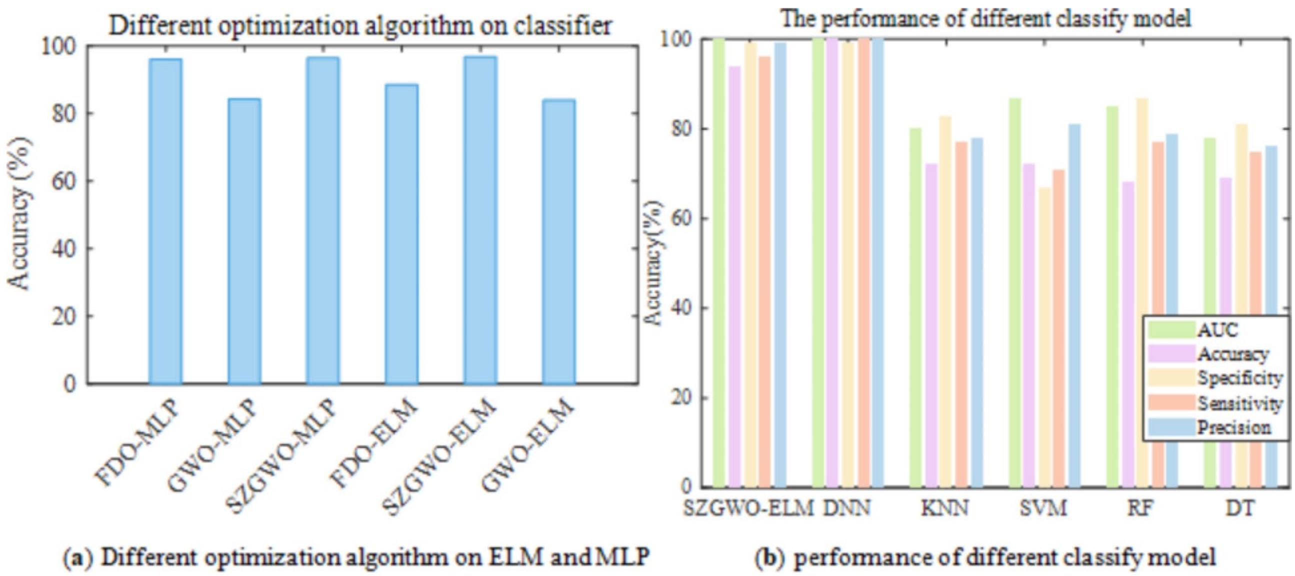


Fig. 6. Performance of different optimization algorithm on different classify model.

78.7% and 75.6%. The performance of the SZGWO-ELM is sightly less than the DNN, however, the SZGWO-ELM network employs fewer parameters and layers, resulting in a considerable reduction in training time, which reveals that the lightweight model also exhibits excellent performance among the five classification model.

Summary and prospect

In addressing the problem of neural networks tending to fall into local optima, continuous adjustment of network parameters and significant cost in training the network when dealing with large amounts of complex medical disease data, this paper proposes a medical assisted diagnosis method based on lightweight fuzzy SZGWO-ELM neural network model. Firstly, two fuzzy functions (ZMF and SMF) are used to improve GWO algorithm, where the ZMF is employed to balance the search capability of the GWO for seeking optimal solutions globally and locally, while the SMF function is used to dynamically adjust the search step of the grey wolves to get the best solution quickly. Then, the SZGWO is used to optimize the parameters of the ELM neural network model for enhancing the performance and reducing the training time. Finally, the performance of the SZGWO is evaluated on 24 test functions from IEEE CEC 2022 on three dimensions, nine intelligent optimization algorithms, four weight improvement strategies, and two GWO variants, and the result reveals the SZGWO has fast convergence ability, the minimum mean value and standard deviation. Additionally, the performance of SZGWO-ELM neural network is assessed by experimenting on five publicly disease datasets from UCI comparing with traditional classify model, and the results demonstrate that the SZGWO-ELM neural network model has outstanding performance, which the accuracy and precision of the SZGWO-ELM model can achieve 96.08% and 99.52%, respectively. Overall, the SZGWO-ELM model proposed in this paper exhibits high accuracy and excellent performance, has fewer parameters and fast training time, and can provide valuable assistance to doctors in enhancing the diagnosis of medical conditions.

However, the SZGWO-ELM neural network model has certain limitations. When faced with extremely large data volumes, this lightweight network requires significant computational resources for weight calculations, making it inefficient for training. Its generalization performance is poor when addressing nonlinear problems, resulting in performance that is far inferior to that of deep neural networks. Therefore, in future research, we will focus on the depth of the network in response to the gradually increasing volume of big data. We aim to explore ways to combine this network with other deep neural networks to optimize it for efficient computation of large-scale data and improve the network's generalization performance. Additionally, to better test the model proposed in this paper in real medical environments, we need to establish close collaborative relationships with medical institutions to obtain more medical disease data. we intend to enrich the dataset with additional features, such as patients' examination records, medication history, and changes in related characteristics after medication. We will then conduct more experiments and comparative studies using updated optimization algorithms and deep neural networks to enhance and refine the performance of the network proposed in this paper. Furthermore, we will explore integrating other internet of things technologies to present real-time data on patients' conditions and promptly inform them about their health information, achieving personalized medical recommendations by building an intelligent warming model. We will explore blockchain technology for patients' disease data, utilizing its decentralization, access control, and encryption features to ensure the confidentiality of storage and transmission.Through these endeavors, it can gain a more thorough understanding of the performance features of the SZGWO-ELM model, thereby offering more robust support for its practical utilization in the medical domain.

Data availability
The original data can be acquired from UCI datasets, where three heart disease respectively derived from https://archive.ics.uci.edu/ml/datasets/Heart+Disease, https://archive.ics.uci.edu/ml/datasets/statlog+(heart), and https://www.kaggle.com/datasets/redwankarimsony/heart-disease-data, and two kidney disease came from https://www.kaggle.com/datasets/abhia1999/chronic-kidney-disease and https://archive.ics.uci.edu/ml/datasets/Chronic_Kidney_Disease.

Code availability
The code for this paper is available on GitHub at the following URL: https://github.com/qiujchen/Swarm-intelligence.git.

Received: 6 May 2024; Accepted: 8 November 2024
Published online: 11 November 2024

References
1. Ali, F. et al. A smart healthcare monitoring system for heart disease prediction based on ensemble deep learning and feature fusion. *J. Inform. Fusion*. **63**, 208–222 (2020).
2. Tsao, C. W. et al. Heart disease and stroke statistics—2022 update: a report from the American Heart Association. *J. Circulation*. **145** (8), 153–639 (2022).
3. Petersen, K. S. & Kris-Etherton, P. M. Diet quality assessment and the relationship between diet quality and cardiovascular disease risk. *J. Nutrients*. **13** (12), 4305 (2021).
4. Correction Naghavi, M. et al. Global, regional, and national age-sex specific all-cause and cause-specific mortality for 240 causes of death, 1990–2013: a systematic analysis for the global burden of Disease Study 2013. *J. Lancet*. **385**, 117–171 (2015).
5. Gansevoort, R. T. et al. Chronic kidney disease and cardiovascular risk: epidemiology, mechanisms, and prevention. *J. Lancet*. **382** (9889), 339–352 (2013).
6. Sengupta, P. P. et al. The future of valvular heart disease assessment and therapy. *J. Lancet*. **403**, 1590–1602 (2024).
7. Vamsi, B., Doppala, B. P., Thirupathi Rao, N. & Bhattacharyya, D. Comparative analysis of prevalent disease by preprocessing techniques using big data and machine learning: An extensive review. *In Proceedings of ICMISC 2020*. 27–38 (2021).
8. Al-Alshaikh, H. A. et al. Comprehensive evaluation and performance analysis of machine learning in heart disease prediction. *J. Sci. Rep.* **14** (1), 7819 (2024).
9. Manikandan, G. et al. Classification models combined with Boruta feature selection for heart disease prediction. *J. Inf. Med. Unlocked*. **44**, 101442 (2024).
10. Adler, E. D. et al. Improving risk prediction in heart failure using machine learning. *J. Eur. J. Heart Fail.* **22** (1), 139–147 (2020).
11. Amin, M. S., Chiam, Y. K. & Varathan, K. D. Identification of significant features and data mining techniques in predicting heart disease. *J. Telematics Inf.* **36**, 82–93 (2019).
12. Saifudin, A., Nabillah, U. U. & Desyani, T. Bagging technique to reduce misclassification in coronary heart disease prediction based on random forest. *J. Physics: Conf. Ser.* **1477(3)**, 032009. 2020 (March).
13. Saqlain, S. M. et al. Fisher score and Matthews correlation coefficient-based feature subset selection for heart disease diagnosis using support vector machines. *J. Knowl. Inform. Syst.* **58**, 139–167 (2019).
14. Zhang, D., Zou, L., Zhou, X. & He, F. Integrating feature selection and feature extraction methods with deep learning to predict clinical outcome of breast cancer. *J. Ieee Access.* **6**, 28936–28944 (2018).
15. Rajagopal, R. & Ranganathan, V. Evaluation of effect of unsupervised dimensionality reduction techniques on automated arrhythmia classification. *J. Biomedical Signal. Process. Control*. **34**, 1–8 (2017).
16. Bharti, R. et al. Prediction of heart disease using a combination of machine learning and deep learning. *J. Computational intelligence and neuroscience*. 1–11 (2021). (2021).
17. Dun, B., Wang, E. & Majumder, S. Heart disease diagnosis on medical data using ensemble learning. *J. Comput. Sci.* **1**, 1–5 (2016).
18. Hamad, A. & Jasim, A. Heart disease diagnosis based on deep learning network. *J. Open. J. Sci. Technol.* **4** (1), 1–9 (2021).
19. Sharifrazi, D. et al. CNN-KCL: automatic myocarditis diagnosis using convolutional neural network combined with k-means clustering. *J. Math. Biosci. Eng.* **19** (3), 2381–2402 (2022).
20. Zeleznik, R. et al. Deep convolutional neural networks to predict cardiovascular risk from computed tomography. *J. Nat. Commun.* **12** (1), 715 (2021).
21. Shanbhag, A. D. et al. Deep learning–based attenuation correction improves diagnostic accuracy of Cardiac SPECT. *J. J. Nuclear Med.* **64** (3), 472–478 (2023).
22. Claux, F., Baudouin, M., Bogey, C. & Rouchaud, A. Dense, deep learning-based intracranial aneurysm detection on TOF MRI using two-stage regularized U-Net. *J. J. Neuroradiol.* **50** (1), 9–15 (2023).
23. Khare, S. K., Gaikwad, N. B. & Bajaj, V. V. H. E. R. S. A novel variational mode decomposition and hilbert transform-based EEG rhythm separation forautomatic ADHD detection. *J. IEEE Trans. Instrum. Meas.* **71**, 1–10 (2022).
24. Nahiduzzaman, M. et al. Diabetic retinopathy identification using parallel convolutional neural network based feature extractor and ELM classifier. *J. Expert Syst. Appl.* **217**, 119557 (2023).
25. Abd Algani, Y. M. et al. Innovative Method for Alzheimer Disease Prediction using GP-ELM-RNN. *In Proceedings of ICAAIC*, 723–728 (2023).

26. Rashedi, E., Nezamabadi-Pour, H. & Saryazdi, S. GSA: a gravitational search algorithm. *J. Inform. Sci.* **179**, 13, 2232–2248 (2009).
27. Shehadeh, H. A. Chernobyl disaster optimizer (CDO): a novel meta-heuristic method for global optimization. *J. Neural Comput. Appl.* **35**, 15, 10733–10749 (2023).
28. Mirjalili, S., Mirjalili, S. M. & Lewis, A. Grey wolf optimizer. *J. Adv. Eng. Softw.* **69**, 46–61 (2014).
29. Cheng, M. Y. & Sholeh, M. N. Optical microscope algorithm: a new metaheuristic inspired by microscope magnification for solving engineering optimization problems. *J. Knowledge-based Syst.* **279**, 110939 (2023).
30. Mirjalili, S. SCA: a sine cosine algorithm for solving optimization problems. *J. Knowledge-based Syst.* **96**, 120–133 (2016).
31. Bayraktar, Z., Komurcu, M. & Werner, D. H. Wind Driven Optimization (WDO): A novel nature-inspired optimization algorithm and its application to electromagnetics. *IEEE Antennas Propag. Soc. Int. Symp.* 1–4 (2010).
32. Mirjalili, S., Mirjalili, S. M. & Hatamlou, A. Multi-verse optimizer: a nature-inspired algorithm for global optimization. *J. Neural Comput. Appl.* **27**, 495–513 (2016).
33. Mirjalili, S. & Lewis, A. The whale optimization algorithm. *J. Adv. Eng. Softw.* **95**, 51–67 (2016).
34. Holland, J. H. *Adaptation in Natural and Artificial Systems: An Introductory Analysis with Applications to Biology, Control, and Artificial Intelligence* (MIT Press, 1992).
35. Mirjalili, S. Evolutionary algorithms and neural networks. *J. Springer*. **780**, 43–55 (2019).
36. Abdullah, J. M. & Ahmed, T. Fitness dependent optimizer: inspired by the bee swarming reproductive process. *J. IEEE Access.* **7**, 43473–43486 (2019).
37. Mohammed, H. & &Rashid, T. F. O. X. A FOX-inspired optimization algorithm. *J. Appl. Intell.* **53** (1), 1030–1050 (2023).
38. Abdulhameed, S. & Rashid, T. A. Child drawing development optimization algorithm based on child's cognitive development. *J. Arab. J. Sci. Eng.* **47** (2), 1337–1351 (2022).
39. El-Kenawy, E. S. M. et al. Greylag goose optimization: nature-inspired optimization algorithm. *J. Expert Syst. Appl.* **238**, 122147 (2024).
40. Abdollahzadeh, B. et al. Puma optimizer (PO): a novel metaheuristic optimization algorithm and its application in machine learning. *J. Cluster Comput.* **1**, 49 (2024).
41. Pervaiz, S., Ul-Qayyum, Z., Bangyal, W. H., Gao, L. & Ahmad, J. A systematic literature review on particle swarm optimization techniques for medical diseases detection. *J. Computational and Mathematical Methods in Medicine*. (2021). (2021).
42. Khafaga, D. Meta-heuristics for feature selection and classification in diagnostic breast cancer. *J. Computers Mater. Continua*. **73** (1), 749–765 (2022).
43. Eid, M. M. et al. Meta-heuristic optimization of LSTM-based deep network for boosting the prediction of monkeypox cases. *J. Math.* **10** (20), 3845 (2022).
44. Eluri, R. K. & Devarakonda, N. Binary golden eagle optimizer with time-varying flight length for feature selection. *J. Knowledge-Based Syst.* **247**, 108771 (2022).
45. Eluri, R. K. & Devarakonda, N. Feature selection with a binary Flamingo Search Algorithm and a genetic algorithm. *J. Multimedia Tools Appl.* **15**, 1–52 (2023).
46. Eluri, R. K. & Devarakonda, N. A. Concise Survey on Solving Feature Selection Problems with Metaheuristic Algorithms. *In International Conference on Advances in Electrical and Computer Technologies*. 207–224 (2021).
47. Bangyal, W. H. et al. An Improved particle swarm optimization algorithm for data classification. *J. Appl. Sci.* **13** (1), 283 (2022).
48. Bangyal, W. H., Shakir, R., Rehman, N. U., Ashraf, A. & Ahmad, J. An Improved Seagull Algorithm for Numerical Optimization Problem. *In International Conference on Swarm Intelligence*. 297–308 (2023).
49. Oyelade, O. N., Ezugwu, A. E. S., Mohamed, T. I. & Abualigah, L. Ebola optimization search algorithm: a new nature-inspired metaheuristic optimization algorithm. *J. IEEE Access.* **10**, 16150–16177 (2022).
50. Nadimi-Shahraki, M. H., Zamani, H. & Mirjalili, S. Enhanced whale optimization algorithm for medical feature selection: a COVID-19 case study. *J. Computers Biology Med.* **148**, 105858 (2022).
51. Pashaei, E. & Pashaei, E. An efficient binary chimp optimization algorithm for feature selection in biomedical data classification. *J. Neural Comput. Appl.* **34** (8), 6427–6451 (2022).
52. owfek, S. K. et al. AI in Higher Education: insights from student surveys and Predictive Analytics using PSO-Guided WOA and Linear regression. *J. J. Artif. Intell. Eng. Pract.* **1** (1), 1–17 (2024).
53. Suresha, H. S. & Parthasarathy, S. S. Detection of Alzheimer's disease using grey wolf optimization based clustering algorithm and deep neural network from magnetic resonance images. *J. Distrib. Parallel Databases*. **40** (4), 627–655 (2022).
54. Saleh, A. I. & Hussien, S. A. Disease diagnosis based on Improved Gray Wolf Optimization (IGWO) and ensemble classification. *J. Annals Biomedical Eng.* **51** (11), 2579–2605 (2023).
55. Kiliçarslan, S. PSO + GWO: a hybrid particle swarm optimization and Grey Wolf optimization based Algorithm for fine-tuning hyper-parameters of convolutional neural networks for Cardiovascular Disease Detection. *J. J. Ambient Intell. Humaniz. Comput.* **14** (1), 87–97 (2023).
56. Deep, K. A random walk Grey wolf optimizer based on dispersion factor for feature selection on chronic disease prediction. *J. Expert Syst. Appl.* **206**, 117864 (2022).
57. Chakraborty, C., Kishor, A. & Rodrigues, J. J. P. C. Novel enhanced-Grey Wolf optimization hybrid machine learning technique for biomedical data computation. *J. Computers Electr. Eng.* **99**, 107778 (2022).
58. Agarwal, P., Arya, A. & Suryaprasad, J. Comparative study of Gaussian and Z-shaped curve membership function for fuzzy classification Approach. *J. Int. J. Comput. Sci. Eng.* **9** (9), 573–579 (2017).
59. Khayatzadeh, R. & Yelten, M. B. A novel multiple membership function generator for fuzzy logic systems. *In Proceedings of SMACD.* 101–104 (2018).
60. Huang, G. B., Zhu, Q. Y. & Siew, C. K. Extreme learning machine: theory and applications [J]. *Neurocomputing*. **70** (1), 489–501 (2006).
61. Jamil, M. & Yang, X. S. A literature survey of benchmark functions for global optimisation problems. *J. Int. J. Math. Modelling Numer. Optimisation*. **4** (2), 150–194 (2013).
62. Hou, Y., Gao, H., Wang, Z. & Du, C. Improved Grey Wolf Optimization Algorithm and application. *J. Sens.* **22**, 3810 (2022).
63. Long, W., Cai, S., Jiao, J. & Tang, M. An efficient and robust grey wolf optimizer algorithm for large-scale numerical optimization. *J. Soft Comput.* **24** (2), 997–1026 (2020).
64. Li, Y., Lin, X. & Liu, J. An Improved Gray Wolf Optimization Algorithm to Solve Engineering problems. *J. Sustain.* **13**, 3208 (2021).
65. Shen, W., Xiao, M., Wang, Z. & Song, X. Rolling Bearing Fault diagnosis based on support Vector Machine optimized by Improved Grey Wolf Algorithm. *J. Sens.* **23** (14), 6645 (2023).

Author contributions

Conceptualization and writing, Chenglong Zhang and Tianhao Peng; Data Analysis, You shun Pan and Jie Liu, methodology and supervision, Qiuju Chen.

Funding

Please add: This research was funded by Zunyi Technology and Big Data Bureau, Moutai Institute Joint Science and Technology Research and Devel-opment Project (No. ZSKHHZ [2023] 122, No. ZSKHHZ [2022] 172), Science and Technology Foundation of Guizhou Province (No. QKHZK[2023]YB450) and Youth Science and Technology Talents Development Project of Guizhou Education Department (No. Qianjiaohe KY [2020] 240).

Declarations

Competing interests

The authors declare no competing interests.

Additional information

Correspondence and requests for materials should be addressed to Q.C.

Reprints and permissions information is available at www.nature.com/reprints.