Computers in Biology and Medicine 159 (2023) 106930

Contents lists available at ScienceDirect

Computers in Biology and Medicine

journal homepage: www.elsevier.com/locate/compbiomed

Effective detection of Alzheimer's disease by optimizing fuzzy K-nearest neighbors based on salp swarm algorithm

Dongwan Lu [a], Yinggao Yue [b], Zhongyi Hu [a,c,d,*], Minghai Xu [b,**], Yinsheng Tong [a], Hanjie Ma [a]

[a] School of Computer Science and Artificial Intelligence, Wenzhou University, Wenzhou, 325035, China
[b] School of Intelligent Manufacturing and Electronic Engineering, Wenzhou University of Technology, Wenzhou, 325035, China
[c] Intelligent Information Systems Institute, Wenzhou University, Wenzhou, 325035, China
[d] Key Laboratory of Intelligent Image Processing and Analysis, Wenzhou, China

ARTICLE INFO

Keywords:
Feature selection
Salp swarm algorithm
Alzheimer's disease
Medical diagnosis
Swarm intelligence algorithm

ABSTRACT

Alzheimer's disease (AD) is a typical senile degenerative disease that has received increasing attention worldwide. Many artificial intelligence methods have been used in the diagnosis of AD. In this paper, a fuzzy k-nearest neighbor method based on the improved binary salp swarm algorithm (IBSSA-FKNN) is proposed for the early diagnosis of AD, so as to distinguish between patients with mild cognitive impairment (MCI), Alzheimer's disease (AD) and normal controls (NC). First, the performance and feature selection accuracy of the method are validated on 5 different benchmark datasets. Secondly, the paper uses the Structural Magnetic Resolution Imaging (sMRI) dataset, in terms of classification accuracy, sensitivity, specificity, etc., the effectiveness of the method on the AD dataset is verified. The simulation results show that the classification accuracy of this method for AD and MCI, AD and NC, MCI and NC are 95.37%, 100%, and 93.95%, respectively. These accuracies are better than the other five comparison methods. The method proposed in this paper can learn better feature subsets from serial multimodal features, so as to improve the performance of early AD diagnosis. It has a good application prospect and will bring great convenience for clinicians to make better decisions in clinical diagnosis.

1. Introduction

Alzheimer's disease (AD) is a typical senile degenerative disease. Its clinical symptoms include memory loss, mood changes, cognitive decline, difficulty speaking, writing, walking, etc. This disease is one of the important ones that endanger the health of the elderly at present [1]. At present, it affects more than 50 million people worldwide, and is expected to affect 150 million people by 2050. Since the pathogenic mechanism of AD has not been fully elucidated, there is currently no cure for AD in humans. One of the important reasons is that when the disease has not yet been detected, it has already progressed irreversibly, resulting in significant memory loss and neurological decline [2]. Generally speaking, the course of AD is divided into three stages: first, pre-symptomatic AD, then mild cognitive impairment (MCI), and finally, a gradual development into AD. Among them, MCI is often mistaken as a manifestation of normal aging and misses the best time for treatment. Therefore, early diagnosis of AD is crucial to delaying the disease, changing the disease process, or even preventing the disease through early intervention strategies [3].

In recent years, in the field of computer-aided diagnosis, many studies have carried out in-depth explorations of the disease. Most studies have adopted the combination of the dimension reduction method and an effective classifier to further improve the early diagnosis of AD. Y. Gharaibeh M et al. used different pre-training models to extract features: Inception V3 and DenseNet201. The PCA method is used to select features with 0.99 interpretation variance ratio, where the combination of selected features from two pre-training models is fed into the machine learning classifier, and the accuracy of Alzheimer's disease classification is 99.14% [4]. Singh S et al. adopted a hybrid strategy of ant colony optimization (ACO) and feedforward convolution neural network (CNN or ConvNet), achieving an accuracy of 98.67% [5]. Pan D adopts an adaptive interpretable integrated model based on 3D convolution neural network (3DCNN) and genetic algorithm (GA), that is, 3DCNN + EL + GA is used to distinguish AD and MCI, and the discriminative brain regions that are significantly helpful for classification are identified in a data-driven manner [6]. Velliangiri S et al. used

* Corresponding author. School of Computer Science and Artificial Intelligence, Wenzhou University, Wenzhou, 325035, China.
** Corresponding author.
E-mail addresses: hujunyi@163.com (Z. Hu), xmhemail@126.com (M. Xu).

https://doi.org/10.1016/j.compbiomed.2023.106930
Received 23 August 2022; Received in revised form 15 March 2023; Accepted 13 April 2023
Available online 14 April 2023

the depth feature reduction technology and the gradient face optimizer optimized dual support vector machine classifier (TSVM-GDO) to classify AD diseases, which greatly improved the classification accuracy and greatly shortened the execution time [7]. Seo Jungryul et al. used a deep learning model combining multi-layer perceptron, SVM, and RNN and achieved an experimental accuracy rate of 70.97% [8].

The advantages of KNN are that it is user-friendly, easy to understand, interpretable, and has a high accuracy rate. The KNN method equally weighs the selected neighbors without considering their space with specific points [9]. Usually, we use a more advanced form of the KNN method. Keller introduces fuzzy sets to improve KNN and proposes a fuzzy K-nearest neighbor (FKNN). It applies fuzzy logic by assigning a certain degree of membership to groups based on the space of each k-nearest neighbor [10]. Since FKNN was proposed, it has been widely used in various classification task, and has been applied in many fields, such as biological and image data classification [11], face recognition [12], Parkinson's disease diagnosis [13], tracking moving targets in videos [14], etc. Meanwhile, some researchers use meta-heuristics to solve practical problems, such as medical diagnosis [15,16], financial distress prediction [17], parameter extraction of solar cells [18], engineering design problems [19,20], feature selection [21,22], education prediction [23], PID control [24], wind speed prediction [25], rolling bearing fault diagnosis [26], gate resource allocation [27] and scheduling problems [28], etc. When using FKNN to solve practical problems, there are two problems to deal with. On the one hand, proper kernel parameter settings play an important role in designing an effective SVM model. The first parameter, adaptively specified the neighborhood size k, and the second parameter, fuzzy intensity parameter m. On the other hand, choosing the optimal subset of input features also greatly affects the performance of the FKNN model.

Feature selection is a commonly used dimensionality reduction method that refers to selecting a subset of attributes from the original set of attributes. Its main purpose is to identify important features, eliminate the irrelevance of unnecessary features, and build a good learning model. Feature selection greatly reduces the computational time of the induction algorithm and improves the accuracy of the resulting model. Feature selection can be divided into two categories: correlation-based filtered feature selection and search-based heuristic feature selection [29]. In recent years, algorithms inspired by nature have become very popular for solving various optimization problems. However, some meta heuristic algorithms proposed recently, for example, the monarch butterfly optimization (MBO) [30], slime mould algorithm (SMA) [31], moth search algorithm (MSA) [32], hunger games search (HGS) [33], Runge Kutta method (RUN) [34], colony predation algorithm (CPA) [35], weighted mean of vectors (INFO) [36] and Harris hawks optimization (HHO) [37], have also attracted the attention of many scholars. In this paper, the binary salp swarm algorithm(BSSA) is used to optimize the FKNN classifier and perform feature selection at the same time. The salp swarm algorithm (SSA) is a global optimization algorithm based on swarm intelligence that was proposed by Mirjalili et al., in 2017 [38]. The algorithm is simple and effective, and since it was proposed, it has been applied to various optimization tasks.

The choice of dimensionality reduction method and classifier is of great significance for the early diagnosis of AD. The classifier based on FKNN has achieved excellent performance on disease diagnosis problems such as the early diagnosis of AD [39] and thyroid disease diagnosis [40]. In summary, this paper proposes a FKNN feature selection method based on the improved binary salp swarm algorithm. First, the Cubic mapping method is used to initialize the population, so that the initial salp population covers the feasible region space more evenly; Secondly, the variable helix factor is introduced, which makes full use of the individual's opposite solution about the origin, reduces the number of individuals beyond the boundary, and ensures the algorithm has a detailed and flexible search ability. Finally, the best and the worst individuals are selected for the updated individuals to carry out dimensional random difference mutation. In order to further study the role of this method in dealing with practical problems, this paper discretizes it into binary ISSA (IBSSA) and applies it to feature selection with the goal of finding the optimal feature subset. On the one hand, on the datasets of BreastCancer, glass, hepatitisfulldata, Lymphography, and WDBC datasets obtained from the UCI Machine Learning Repository, the effectiveness of this method is tested in terms of classification accuracy, sensitivity, and specificity, and other indicators. On the other hand, in order to verify the effectiveness of this method in the diagnosis of early AD, we used MRI, PET, and CSF multimodal feature data from the international Alzheimer's disease neuroimaging initiative (ADNI) and compared them with other methods, which are swarm intelligence algorithms combined with a FKNN classifier. The experimental results show that the IBSSA-FKNN method can effectively improve the classification performance and the performance of early AD diagnosis. It has a good application prospect and will bring great convenience for clinicians to make better decisions in clinical diagnosis.

The rest of this paper is organized as follows: Section 2 introduces the FKNN classifier; Section 3 introduces the salp swarm algorithm and the binary salp swarm algorithm; Section 4 introduces the classification method proposed in this paper, namely IBSSA-FKNN; Section 5 conducts experiments and results analysis on traditional datasets; and Section 6 conducts experiments and results analysis on sMRI datasets. Finally, Section 7 discusses the conclusions and introduces the prospects for future work.

2. Background materials

2.1. Fuzzy K-nearest neighbors(FKNN)

KNN is one of the simplest classifiers. For the samples to be classified, KNN determines the sample class as the pattern of the neighbor's class according to the k neighbors closest to the sample. However, this method defaults to assuming that each sample has the same weight and has only one class, which is not the case in reality. In order to solve these two problems, Keller introduced fuzzy set theory into KNN and proposed the FKNN algorithm. In FKNN, each sample now belongs to multiple classes with different membership degrees, and no longer belongs to only one class. Furthermore, FKNN assigns different weights to each neighbor according to the distance between samples. Simply put, neighbors with similar distances have greater weight in determining the class than those with farther distances. In FKNN, the fuzzy membership of samples is assigned to different classes according to the following formula:

$$u_i(x) = \frac{\sum_{j=1}^{k} u_{ij}\left(1/\left\|x - x_j\right\|^{2/(m-1)}\right)}{\sum_{j=1}^{k}\left(1/\left\|x - x_j\right\|^{2/(m-1)}\right)} \tag{1}$$

where $i = 1, 2, \ldots, C$, $j = 1, 2, \ldots, k$, the number of classes is C, and the number of nearest neighbors is K. When calculating the contribution of each neighbor to the membership value, the fuzzy intensity parameter m is used to determine the weight of the distance, and its value is usually chosen as $m \in (1, \infty)$. $\left\|x - x_j\right\|$ is the distance between x and its j-th nearest neighbor x_j, Euclidean distance is usually chosen as the distance metric. u_{ij} is the membership degree of pattern x_j from the training set to class i, in the k nearest neighbors of x. In this study, we adopted constrained fuzzy membership, that is, we find the k nearest neighbors of each training pattern (such as x_k), and the membership of x_k in each class is assigned as:

$$u_{ij}(x_k) = \begin{cases} 0.51 + \left(n_j/k\right) \times 0.49, \text{if } j = i \\ \left(n_j/K\right) \times 0.49, \text{if } j \neq i \end{cases} \tag{2}$$

The value n_j is the number of neighbors found belonging to the j-th class. Note that the membership calculated by equation (2) should meet

the following equation:

$$\begin{aligned}&\sum_{i=1}^{C}\mu_{ij}=1, j=1,2,\ldots,n,\\&0<\sum_{j=1}^{n}u_{ij}<n,\\&u_{ij}\in[0,1]\end{aligned}\tag{3}$$

After calculating all memberships of the query sample, assign it to the class with the highest membership value, that is:

$$C(x)=\underset{i=1}{\overset{C}{\arg\max}}(u_i(x))\tag{4}$$

The steps of FKNN are as follows.

1) Calculate the membership degree of all training samples for each category by eq. (2);
2) For the test sample, find its K nearest neighbors by distance measurement, and calculate the membership degree of the test sample for each class by Eq. (1);
3) Get the predicted label by Eq. (4).

2.2. Salp swarm algorithm (SSA)

Salp Swarm Algorithm (SSA) is a global optimization algorithm based on swarm intelligence that was proposed by Mirjalili et al., in 2017 [38]. The salp is a kind of marine creature with body tissue and a movement mode highly similar to jellyfish, and it is a kind of capsule animal that can float freely. In the chain-like group behavior of salps, individuals usually connect head to tail to form a "chain" and move in sequence. In the salp chain, divided into leaders and followers, the leader moves towards the food and guides the movement of the followers who follow them. According to the strict "hierarchy" system, the movement of the followers is affected only by the previous step. Such a movement mode makes the salps chain have a strong ability for global exploration and local development.

Population initialization: Let the search space be the Euclidean space of $D\times N$, D is the space dimension, and N is population number. The position of the salps in space is denoted by $X_n=[X_{n1},X_{n2},X_{n3},\cdots,X_{nD}]^{\mathrm{T}}$, the position of the food is denoted by $F_n=[F_{n1},F_{n2},F_{n3},\cdots,F_{nD}]^{\mathrm{T}}$, $n=1,2,3,\cdots,N$.The upper bound of the search space is $ub=[ub_1,ub_2,ub_3,\cdots,ub_j,\cdots,ub_D]$,and the lower bound is $lb=[lb_1,lb_2,lb_3,\cdots,lb_j,\cdots,lb_D]$, $j=1,2,3,\cdots,D$.

$$X_{D\times N}=rand(D,N)\cdot(ub-lb)+lb\tag{5}$$

Leaders in the population are represented by X_d^l, followers are represented by X_d^i, $i=1,2,3.4,\cdots N$; $d=1,2,3,\cdots D$.

Leader position update: During salp chain movement and foraging, the leader's position update is expressed as:

$$x_d^l=\begin{cases}F_d+c_1((ub-lb)c_2+lb), c_3\geq 0.5\\F_d-c_1((ub-lb)c_2+lb), c_3<0.5\end{cases}\tag{6}$$

where X_d^l and F_d are the position of the first salp (leader) and the position of food in the d-th dimension, respectively; ub and lb are the corresponding upper and lower bounds, respectively. where c_1, c_2, and c_3 are control parameters.

Equation (2) shows that the leader's location update is only related to the location of the food. c_1 is the convergence factor in the optimization algorithm, which plays the role of balancing global exploration and local development, and is the most important control parameter in SSA. The expression of c_1 is:

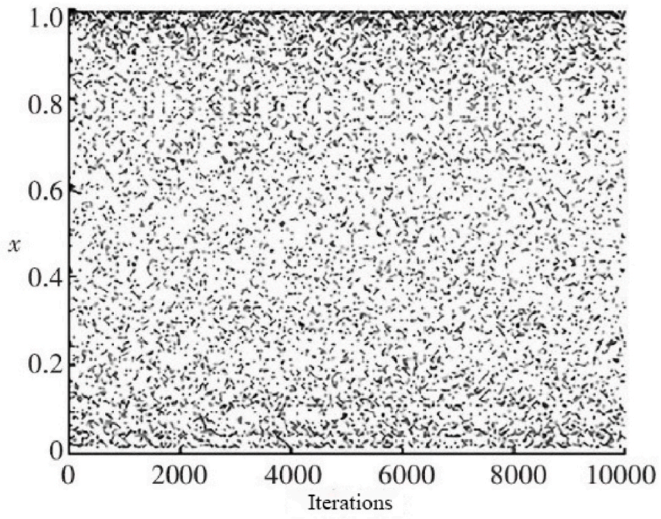

Fig. 1. Cubic mapping when $\rho=2.59$, $x_0=0.3$.

$$c_1=2e^{-\left(\frac{4l}{L}\right)^2}\tag{7}$$

where l is the current iteration number; L is the maximum iteration number. The convergence factor is a decreasing function of 2–0.

The control parameters c_2 and c_3 are random numbers of [0,1], which are used to enhance the randomness of X_d^l and improve the global search and individual diversity of the chain groups.

Followers position update: in the process of movement and foraging in the salp chain, the followers move forward in a chain-like manner through the mutual influence between the front and rear individuals. Their displacement conforms to Newton's law of motion, and their motion displacement is:

$$X=\frac{1}{2}at^2+v_0t\tag{8}$$

where t is the time; a is the acceleration, the formula is $a=(v_{final}-v_0)/t$; v_0 is the initial velocity, and $v_{final}=(X_d^i-X_d^{i-1})/t$.

Considering that t is iterative in the optimization algorithm, let $t=1$ and $v_0=0$ in the iterative process. Then formula (4) can be expressed as:

$$X=\frac{X_d^i-X_d^{i-1}}{2}\tag{9}$$

where $i\geq 2$; X_d^i and X_d^{i-1} are the positions of two salps that are closely connected to each other in the d-th dimension, respectively. Therefore, the position of the follower is expressed as:

$$X_d^{i'}=\frac{X_d^i+X_d^{i-1}}{2}\tag{10}$$

where $X_d^{i'}$ and X_d^i are the updated follower's position and the pre-update follower's position in the d-th dimension, respectively.

The pseudocode of the SSA algorithm is as follows.

Algorithm 1. Salp Swarm Algorithm

Algorithm1:Salp Swarm Algorithm

Input: Engender initial population (x_d^i), where $i = 1, 2, \ldots, N$ randomly

Output: The solution vector F_d

while stopping criterion **do**
  Compute fitness and find the current best salp
  Calculate c_1 using to Eq. (7)
  for each salp (X_j) in salp chain **do**
    if (j == 1) **then**
    Modify position of leader salp using Eq. (6)
    else
    Modify locations of follower salps according to Eq. (10)
    end if
  end for
end while

Print the best solution F_d

3. Improved binary salp swarm algorithm (IBSSA)

3.1. Population initialization strategy based on cubic chaotic map

Chaotic sequences have the advantages of easy implementation, short execution time, and being able to jump out of the local optimal value, so they are widely used in random based optimization algorithms. Lyapunov exponent is often used to judge the dynamic performance of the system. The larger the value, the higher the degree of chaos. FENG et al. analyzed the best chaotic sequences generated by 16 common chaotic maps [41], and the results showed that the running time of cubic chaotic map is short, and the lyapunov exponent is close to the optimal value. In this paper, cubic chaotic map is used to optimize the initial solution and improve the search efficiency.

The expression of the standard Cubic chaotic mapping function is:

$$x_{n-1} = \alpha x_n^3 - \beta x_n \tag{11}$$

where, α and β are chaos influencing factors, and the range of Cubic maps with different α and β values is different. Generally, when $\beta \in (2.3, 3)$, the sequence generated by Cubic mapping is chaotic.

In addition, when $\alpha = 1, x_n \in (-2, 2)$. when $\alpha = 4, x_n \in (-1, 1)$. To make $x_n \in (0, 1)$, the Cubic mapping used in the improved algorithm is in the following form:

$$x_{n+1} = \rho x_n \left(1 - x_n^2\right), x_n \in (0, 1) \tag{12}$$

where, ρ is the control parameter. Chaos of Cubic map is closely related to the value of parameter ρ. Here, take the initial value $x_0 = 0.3$, and the number of iterations is 10000. The simulation results of Cubic mapping are shown in Fig. 1.

It can be seen from the figure that when $\rho = 2.59$, Cubic map is a full map between (0,1) and has the best chaotic ergodicity.

3.2. Binary mechanism

In practice, according to the different types of solutions, it can be divided into continuous solution space and discrete solution space, while the standard Salp Swarm Algorithm can only use the position vector in the continuous domain to move around the search space. The transformation between the continuous solution space and the discrete solution space can be discretized through a specific transformation function, generally using a sigmoid transformation function. At the same time, the position of the salps may stay at some local points and remain unchanged when the value is large. To avoid this weakness, the sigmoid transformation function is used here [42]. Use the particle's velocity probability to change the position of an element.

$$S\left(x_j^i(t)\right) = \frac{1}{1 + \exp\left(-x_j^i(t)\right)} \tag{13}$$

where $x_j^i(t)$ is the velocity of the i-th individual in the j-th dimension at time t; $S(x_j^i(t))$ is the transformation probability that the position $x_j^i(t)$ takes 1 or 0.

After calculating the transition probability, the following equation (10) needs to be used to update the position of the salps:

$$x_j^i(t) = \begin{cases} 1, rand \geq S\left(x_j^i(t)\right) \\ 0, rand < S\left(x_j^i(t)\right) \end{cases} \tag{14}$$

where $x_j^i(t)$ is the position of the i-th salps in the j-th dimension at time t. $X_{\max}$ is selected as the maximum position value to limit the range of $x_j^i(t)$, that is, $x_j^i(t) \in [-X_{\max}, X_{\max}]$, and also limit the probability that the position $x_j^i(t)$ is converted to 1 or 0.

3.3. Follower position updating strategy based on the variable helix mechanism

The location update of the i-th follower of the thallus group algorithm is determined by the location coordinates of the i-th and i-1 thallus, and this location update rule is only determined by the positions

of the previous individual and the current individual in the thallus chain, Therefore, the updated thimbles are highly dependent on the leader individuals of the previous update, which easily limits the global search ability and local search speed of the algorithm. To solve the above problems, the variable helix factor is introduced, which makes full use of the individual's opposite solution about the origin, reduces the number of individuals beyond the boundary, and ensures the algorithm has a detailed and flexible search ability.

The variable helix factor is calculated as follows:

$$H = a \cdot \cos(k \cdot l \cdot \pi) \tag{15}$$

$$a = \begin{cases} 1, t < \dfrac{M}{2} \\ e^{5 \cdot l}, otherwise \end{cases} \tag{16}$$

$$l = 1 - 2 \cdot \frac{t}{M} \tag{17}$$

where, H represents variable helix factor; a is the parameter used to control the spiral. The early iteration value is close to 1, and the later iteration value gradually decreases; k is the parameter representing the spiral cycle, and the value is $M/10$; l is a parameter that decreases linearly from 1 to - 1 as the number of iterations increases.

The improvement of the follower's extensive search enables the follower to make full use of the entire search space, more easily get rid of the attraction of the local optimal solution, strengthen the search of the entire space, maintain the diversity of the population, enhance the early algorithm exploration ability, and improve the later algorithm development ability. Based on this, the follower formula is updated as follows:

$$x_d^{i'} = \begin{cases} \dfrac{1}{2} \cdot \cos(a \cdot l \cdot \pi) \cdot \left(x_d^i + x_d^{i-1}\right), t < \dfrac{M}{2} \\ \dfrac{1}{2} \cdot e^{5 \cdot l} \cdot \cos(a \cdot l \cdot \pi) \cdot \left(x_d^i + x_d^{i-1}\right), t > \dfrac{M}{2} \end{cases} \tag{18}$$

3.4. Dimensional random difference mutation

Use random difference mutation to carry out dimensional mutation, and obtain a new individual dimension through this mutation. The specific formula is as follows.

$$x_j^i = r_1 \times \left(F_j - x_j^i\right) + r_2 \times \left(x_j^{'} - x_j^i\right) \tag{19}$$

Among them, x_j^i is the j-th dimension of the i-th individual in the salps group; F_j is the j-th dimension of food source location; $x_j^{'}$ is the j-th dimension of a random individual in the population; r_1 and r_2 are random numbers of [0,1]. After the population location update is completed, use the dimension-by-dimension random differential mutation to mutate each dimension of the individual, and evaluate a certain dimension after it mutates. If it is excellent, retain the solution after the mutation. If the evaluation result after the mutation becomes poor, discard the poor dimension information, reduce the interference between each dimension, and increase the search scope. Due to the blindness of mutation operation, the search efficiency of the algorithm will be reduced and the calculation amount will be greatly increased if all individuals are subjected to dimensional random differential mutation. Therefore, only the best and worst individuals in the population are selected for mutation. The best individual mutation can improve the search efficiency, and the worst individual mutation can improve the search range and jump out of the local optimal solution.

4. Proposed IBSSA-FKNN model

On the one hand, in FKNN, the distance weights of k neighbors are calculated based on distance measures, without distinguishing the importance of features and without taking into account the impact of

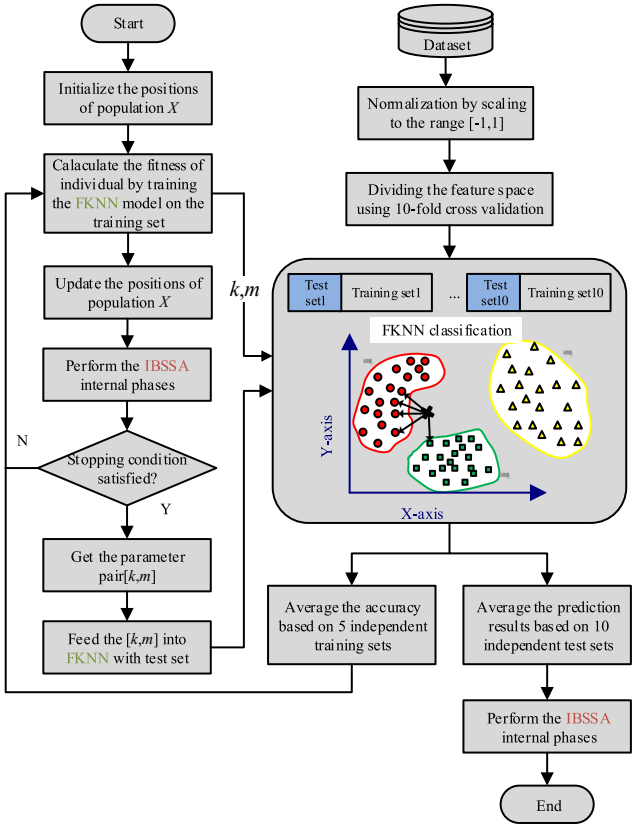

Fig. 2. Flowchart of the proposed IBSSA-FKNN diagnostic system.

different distances from different neighbors to the center of the sample class. FKNN needs to get the distance from all training set samples to get its k neighbors, resulting in a large amount of computation. In view of the problems existing in FKNN, many scholars have studied and improved it, mainly aiming at some parameter selection or optimization problems involved in this algorithm. These improved methods have improved accuracy and reliability to a certain extent, but still have shortcomings. On the other hand, SSA has strong optimization ability and high optimization accuracy. But for complex problems, they will also fall into local extremum. Therefore, first, the Cubic mapping method is used to initialize the population, so that the initial salp population covers the feasible region space more evenly; Secondly, the variable helix factor is introduced, which makes full use of the individual's opposite solution about the origin, reduces the number of individuals beyond the boundary, and ensures the algorithm has a detailed and flexible search ability. Finally, the best and the worst individuals are selected for the updated individuals to carry out dimensional random difference mutation. In order to further study the role of this method in dealing with practical problems, this paper discretizes it into binary ISSA (IBSSA) and applies it to feature selection with the goal of finding the optimal feature subset.

In this section, we use the IBSSA algorithm for feature selection to the original FKNN and create a model called IBSSA-FKNN. The main goal of this model is to optimize the FKNN classifier: (1) determine the number of the nearest neighbors k and the fuzzy strength parameter m;(2) identify the best subset of discriminative features and feature selection. The appropriate feature subset obtained is used as input to the optimized FKNN model for classification. The IBSSA-FKNN method takes diagnostic accuracy as the fitness of feature selection. The IBSSA-FKNN flowchart of the overall architecture of the proposed model is shown in Fig. 2.

A flag vector for feature selection is shown in Fig. 3. The vector

consisting of a series of binary values of 0 and 1 represents a subset of features, that is, an actual feature vector, which has been normalized. For a problem with D dimensions, there are D bits in the vector. The i-th feature is selected if the value of the i-th bit equals one; otherwise, this feature will not be selected ($i = 1, 2, \ldots, D$). The size of a feature subset is the number of bits, whose values are one in the vector. The pseudocode of the IBSSA algorithm is presented as shown in Algorithm 2.

Algorithm 2. Pseudo-code for feature selection procedure

Algorithm2:Pseudo-code for feature selection procedure

Input: Engender initial population (x_d^l) according to Eq. (12)

Output: Selected feature List F

Convert initial population into binary using threshold δ

while stopping criterion **do**

Train FKNN model with the randomly chosen features by using 10-fold CV

Compute fitness and find the current best salp

Calculate c_1 using to Eq. (7)

for each salp (X_j) in salp chain **do**

if (j == 1) **then**

Modify position of leader salp using Eq. (6) and convert them into binary using threshold δ

else

Update locations of follower salps according to Eq. (18) and convert them into binary using threshold δ

end if

end for

end while

Print the optimal features

After the parameter pair and feature subset were obtained, the FKNN model began to perform the classification tasks. At first, the FKNN trained on reduced the training feature space using the parameter pair to evolve an optimal model, and then the optimal FKNN model was employed to predict the new samples on the reduced testing feature space. The whole process was done via the 10-fold CV analysis, and finally the average results over 10 folds were computed. The detailed pseudo-code for the classification phase is as follows.

Algorithm 3. Pseudo-code f or the classification procedure

Algorithm3:Pseudo-code f or the classification procedure

/*performance estimation buy using k-fold CV where k=10*/

Begin:

for j=1:k

Reduced training set=k-1 subsets;

Reduced testing set = remaining subsets;

Train the FKNN model on the reduced training feature space using the obtained optimal parameter combination;

Test it on the reduced testing feature space and save the mean CV accuracy;

end for

Return the average classification accuracy of FKNN over j test set.

End

5. Traditional data set experiment

5.1. Dataset description

In order to verify the effectiveness of the proposed method, this section conducts experiments on the SSA-FKNN method on 5 classification datasets, which are BreastCancer, glass, hepatitisfulldata, Lymphography, WDBC. The datasets are from the UCI Machine Learning Repository (http://archive.ics.uci.edu/ml/datasets). Among them, the BreastCancer dataset has 699 data, including 9 features and 2 categories; the grass dataset has 214 data, including 10 features and 2 categories; the hepatitisfulldata dataset has 155 data, including 20 features and 2 categories; Lymphography dataset has 148 data, including 18 features and 4 categories; WDBC dataset has 569 data, including 30 features and 2 categories. The specific description information of the dataset is shown in Table 1.

Before the experiment, the data needs to be preprocessed first. Since the BreastCancer data set has the missing features, in order to ensure the integrity of the sample data, the average value of these records is processed in this experiment. At the same time, in order to reduce the difference between the eigenvalues and prevent the larger eigenvalues

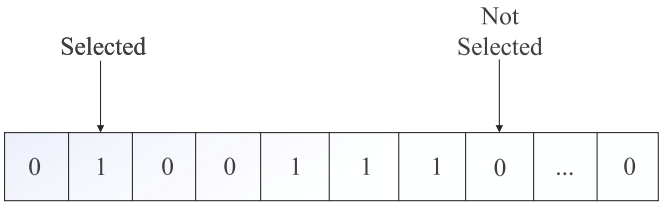

Fig. 3. A flag vector for feature selection.

from excessively affecting the smaller eigenvalues, we normalize each eigenvalue to the [-1,1] interval. The normalized calculation formula is:

$$x^{'} = \left(\frac{x - \min_a}{\max_a - \min_a}\right) * 2 - 1 \tag{20}$$

where x is the original value of the data, $x^{'}$ is the normalized value, $\max_a$ is the maximum value in feature a, and $\min_a$ is the minimum value in feature a.

5.2. Experimental setup and description

The proposed IBSSA-FKNN method is implemented on the MATLAB2018b platform. This experiment is performed on an NVIDIA GeForce GTX 1660 with Windows 10 as the operating system. The detailed parameters of IBSSA-FKNN are set as follows: the number of populations is 20, and the maximum number of iterations is set to 1000. In order to verify the effectiveness of the improved IBSSA algorithm in feature selection, a total of 5 comparison algorithms are set up for comparison, which are Binary Bat Algorithm (BBA) [43], Binary Moth Flame Optimizer (BMFO) [44], Quantum Gaussian Dragonfly Algorithm (QGDA) [45], Binary Quantum Grey Wolf Optimization Algorithm (BQGWO) [46], Binary Spread Strategy with the Chaotic Local Search Grey Wolf Optimization (BSCGWO) [47]. The parameter settings of the contrast group intelligent optimization algorithm involved in this paper are shown in Table 3.

The experiments are mainly carried out by using the wrapped feature selection method. During the experiments, the IBSSA algorithm is used to generate feature subsets, and the resulting feature subsets are evaluated using the results obtained by the FKNN classifier. In the feature selection process, the IBSSA algorithm realizes the search through a ten-fold cross-validation strategy and applies it to practical problems

Table 1
Detailed description of the dataset.

NO.	dataset	number of categories	number of samples	number of features	is there a missing value
1	BreastCancer	2	699	9	yes
2	glass	2	214	10	no
3	hepatitisfulldata	2	155	20	no
4	Lymphography	4	148	18	no
5	WDBC	2	569	30	no

Table 2
Parameter setting of swarm intelligence optimization algorithm.

Algorithm	Parameters
BBA	$[f_{\min}, f_{\max}] = [0,2]; A = 0.5; r = 0.5; \alpha = 0.95; \gamma = 0.05$
BMFO	$b = 1$
BQGWO	$a = 2 - FEs \times (2/MaxFEs); r_1 = r_2 = rand(0,1); A = 2 \times a \times r_1; C = 2 \times r_2; \beta = \omega = 10$
BSCGWO	$a = 2 - FEs \times (2/MaxFEs); r_1 = r_2 = rand(0,1); A = 2 \times a \times r_1; C = 2 \times r_2; \beta = a * rand(0,1)$
IBSSA	$c_1 = 2 \times e^{-\left(\frac{4 \times FEs}{MaxFEs}\right)^2}; c_2 = c_3 = rand(0,1); \rho = 2.59; x_0 = 0.3$

Table 3
The detailed results of the IBSSA-FKNN model on the BreastCancer dataset.

Runs of 10-fold CV	ACC	SEN	SPE	PRE	No.of selected feature
#1	1	1	1	1	4
#2	1	1	1	1	4
#3	1	1	1	1	4
#4	0.98571	1	1	0.97872	4
#5	1	1	0.95833	1	4
#6	1	1	1	1	4
#7	1	1	1	1	3
#8	0.98571	0.97826	1	1	3
#9	0.98551	0.97778	1	1	4
#10	0.97183	0.95652	1	1	5
Mean	0.9928	0.9913	0.9958	0.9979	3.9

through the KNN model. K-fold cross-validation is mainly used to obtain an unbiased estimate of generalization accuracy. If K is set to 10, the data set is divided into 10 subsets, one of which is taken as the test set, and the remaining part is taken as the training set. Then, the average error of all 10 tests is calculated. During the implementation of the K-fold cross-validation strategy, all test sets are independent, and relatively stable and reliable results can be obtained. In addition, this section uses the IBSSA algorithm to generate the optimal feature subset on the training set and then uses the validation dataset filtered by the optimal feature subset to classify using the FKNN classifier to obtain the final result. In subsequent experiments, the best results of the evaluation indicators have been bolded in the table.

5.3. Evaluation criteria

Evaluate the classification performance of this method, which are classification Accuracy (ACC), Sensitivity (SEN), Specificity (SPE), Precision(PRE), F-measure. Defined as follows:

Accuracy is the proportion of the total number of correct predictions. Use the following methods to determine:

$$ACC = \frac{TP + TN}{TP + TN + FN + FP} \times 100\% \tag{21}$$

Sensitivity is an index used to measure the classifier's recognition of abnormal records, and is also often expressed as the TP rate.

$$SEN = \frac{TP}{TP + FN} \times 100\% \tag{22}$$

Specificity is often used to estimate the ability of a classification model to identify normal examples, which is also often expressed as the TN rate.

$$SPE = \frac{TN}{TN + FP} \times 100\% \tag{23}$$

Precision is the correct proportion of positive instances of prediction, as calculated using:

Table 4
The detailed results of the IBSSA-FKNN model on the glass dataset.

Runs of 10-fold CV	ACC	SEN	SPE	PRE	No.of selected feature
#1	0.95238	0	0	0.95238	3
#2	0.85	0	0	0.85	3
#3	0.90476	0	0	0.90476	4
#4	0.80952	0	0	0.80952	3
#5	0.90909	0	0	0.90909	3
#6	0.95	0	0	0.95	4
#7	0.86957	0	0	0.86957	0
#8	0.90909	0	0	0.90909	0
#9	0.78261	0	0	0.78261	4
#10	0.90476	0	0	0.90476	6
Mean	0.8842	0	0	0.8842	3

Table 5
The detailed results of the IBSSA-FKNN model on the hepatitisfulldata dataset.

Runs of 10-fold CV	ACC	SEN	SPE	PRE	No.of selected feature
#1	1	1	1	1	2
#2	1	1	1	1	5
#3	1	1	1	1	4
#4	1	1	1	1	2
#5	1	1	1	1	2
#6	1	1	1	1	3
#7	1	1	1	1	3
#8	1	1	1	1	5
#9	1	1	1	1	4
#10	1	1	1	1	1
Mean	1	1	1	1	3.1

Table 6
The detailed results of the IBSSA-FKNN model on the Lymphography dataset.

Runs of 10-fold CV	ACC	SEN	SPE	PRE	No.of selected feature
#1	1	0	0	1	4
#2	0.9375	0	0	0.9375	3
#3	1	0	0	1	3
#4	1	0	0	1	7
#5	1	0	0	1	7
#6	1	0	0	1	7
#7	1	0	0	1	3
#8	1	0	0	1	6
#9	0.92857	0	0	0.92857	3
#10	1	0	0	1	8
Mean	0.9866	1	1	0.9866	5.1

Table 7
The detailed results of the IBSSA-FKNN model on the WDBC dataset.

Runs of 10-fold CV	ACC	SEN	SPE	PRE	No.of selected feature
#1	0.98276	0.95455	1	1	13
#2	1	1	1	1	5
#3	1	1	1	1	9
#4	1	1	1	1	2
#5	0.98246	1	1	1	2
#6	1	1	1	1	6
#7	1	1	1	1	8
#8	1	1	1	1	3
#9	1	1	1	1	4
#10	1	1	1	1	3
Mean	0.9965	0.9955	1	1	5.5

Table 8
Experimental results of six methods on the BreastCancer dataset.

Algorithm	Features' size	ACC (%)	SEN (%)	SPE (%)	PRE (%)	F-measure (%)
IBSSA-FKNN	**3.3**	**0.9929**	**0.9913**	**0.9958**	**0.9979**	**0.9945**
BBA-FKNN	3.8	0.9411	0.9453	0.9333	0.9650	0.9543
BMFO-FKNN	3.6	0.9871	0.9847	0.9917	0.9957	0.9901
QGDA-FKNN	3.6	0.9872	0.9847	0.9920	0.9957	0.9901
BQGWO-FKNN	4.1	0.9885	0.9913	0.9833	0.9914	0.9913
BSCGWO-FKNN	3.4	0.9857	0.9869	0.9833	0.9914	0.9890

$$PRE = \frac{TP}{TP + FP} \times 100\% \tag{24}$$

Among them, TP (True Positive), FP (False Positive), TN (True Negative) and FN (False Negative) represent true positive, false positive, true negative and false positive, respectively.

Table 9
Experimental results of six methods on the glass dataset.

Algorithm	Features' size	ACC (%)	SEN (%)	SPE (%)	PRE (%)	F-measure (%)
IBSSA-FKNN	**3**	**0.8842**	**0**	**0**	**0.8842**	**0**
BBA-FKNN	4.2	0.6856	0	0	0.6856	0
BMFO-FKNN	3.7	0.8788	0	0	0.8788	0
QGDA-FKNN	4.3	0.8773	0	0	0.8773	0
BQGWO-FKNN	3.9	0.8595	0	0	0.8595	0
BSCGWO-FKNN	3.7	0.8744	0	0	0.8744	0

Table 10
Experimental results of six methods on the hepatitisfulldata dataset.

Algorithm	Features' size	ACC (%)	SEN (%)	SPE (%)	PRE (%)	F-measure (%)
IBSSA-FKNN	**3.1**	**1**	**1**	**1**	**1**	**1**
BBA-FKNN	8.2	0.8410	0.6417	0.8949	0.6683	0.6309
BMFO-FKNN	7.6	1	1	1	1	1
QGDA-FKNN	3.6	1	1	1	1	1
BQGWO-FKNN	3.9	0.9875	0.9750	0.9923	0.9750	0.9714
BSCGWO-FKNN	3.7	0.9933	0.9667	1	1	0.9800

Table 11
Experimental results of six methods on the Lymphography dataset.

Algorithm	Features' size	ACC (%)	SEN (%)	SPE (%)	PRE (%)	F-measure (%)
IBSSA-FKNN	**5.1**	**0.9866**	**0**	**0**	**0.9866**	**0**
BBA-FKNN	7.1	0.8268	0	0	0.8268	0
BMFO-FKNN	6.7	0.9749	0	0	0.9749	0
QGDA-FKNN	5.7	0.9799	0	0	0.9799	0
BQGWO-FKNN	4.5	0.9804	0	0	0.9804	0
BSCGWO-FKNN	**4**	0.9518	0	0	0.9518	0

Table 12
Experimental results of six methods on the WDBC dataset.

Algorithm	Features' size	ACC (%)	SEN (%)	SPE (%)	PRE (%)	F-measure (%)
IBSSA-FKNN	**5.5**	**0.9965**	**0.9955**	**1**	**1**	**0.9953**
BBA-FKNN	10	0.9474	0.9288	0.9583	0.9353	0.9297
BMFO-FKNN	12.5	0.9965	0.9952	0.9972	0.9955	0.9952
QGDA-FKNN	5.7	0.9982	0.9952	0.9972	0.9955	**0.9977**
BQGWO-FKNN	**4.1**	0.9948	0.9907	0.9972	0.9952	0.9930
BSCGWO-FKNN	4.3	0.9947	0.9859	1	1	0.9928

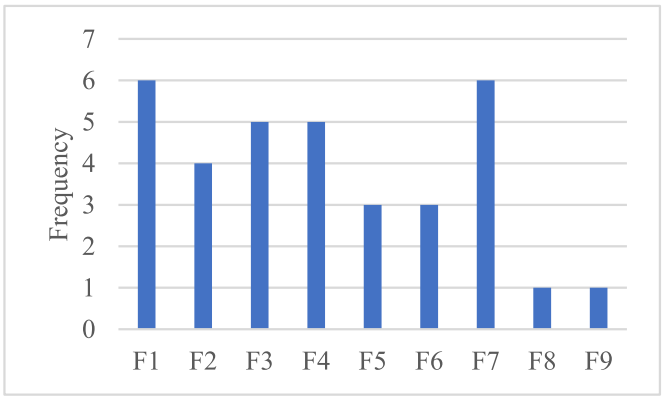

Fig. 4. The frequency of selected features in 10-fold CV on the BreastCancer data set.

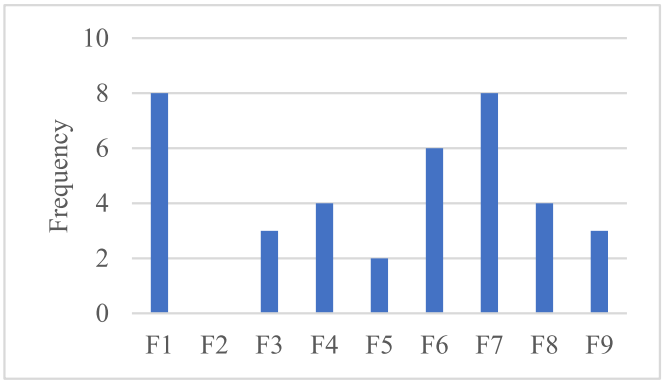

Fig. 5. The frequency of selected features in 10-fold CV on the glass data set.

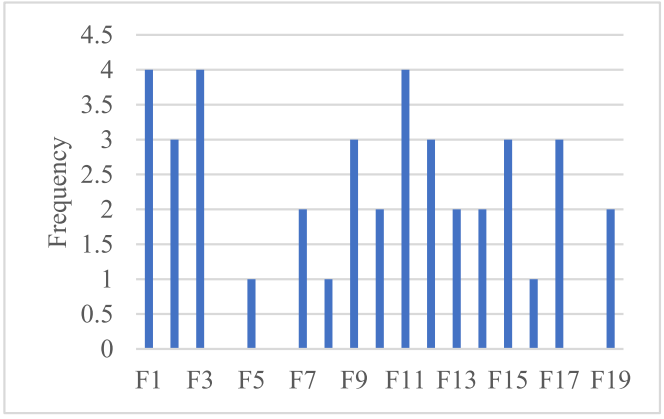

Fig. 6. The frequency of selected features in 10-fold CV on the hepatitisfulldata data set.

Lewis and Gale proposed the F-measure in 1994, which is defined as follows:

$$F - = \frac{(\beta^2 + 1) * \Pr ecision * Sensitivity}{\beta^2 * \Pr ecision + Sensitivity} \tag{25}$$

In Equation (25) above, there is a value from 0 to infinity to control the weights assigned to the precision and sensitivity. If all positive instances are classified incorrectly, any classifier evaluated using the above will have a metric of 0. In this experiment, the β value was set to 1.

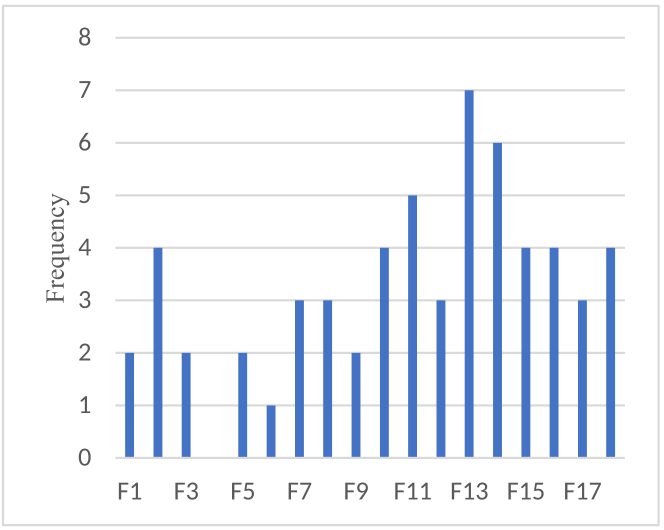

Fig. 7. The frequency of selected features in 10-fold CV on the Lymphography data set.

5.4. Experimental results

Table 3, 4, 5, 6, and 7 show the comparison of the results of the IBSSA-FKNN algorithm performing 10 times of 10-fold cross-validation on 5 datasets, respectively. The performance evaluation criteria include training classification Accuracy (ACC), Sensitivity (SEN), Specificity (SPE), precision(PRE) and number of feature selection. As can be seen from Table 3, the experimental results of the IBSSA-FKNN method on the BreastCancer dataset are recorded in detail. During the 10-fold operation, the average values of the five evaluation indicators are 0.9928, 0.9913, 0.9958, 0.9979 and 3.9, respectively. Similarly, it can be seen from Table 4 that the experimental results of the IBSSA-FKNN method on the glass dataset are recorded in detail. During the 10-fold operation, the average values of the five evaluation indicators are 0.8842, 0, 0, 0.8842 and 3, respectively. It can be seen from Table 5 that the experimental results of the IBSSA-FKNN method on the hepatitisfuldata dataset are recorded in detail. During the 10-fold operation, the average values of the five evaluation indicators are 1, 1, 1, 1 and 3.1, respectively. It can be seen from Table 6 that the experimental results of the IBSSA-FKNN method on the Lymphagraphy data set are recorded in detail. During the 10-fold operation, the average values of the five evaluation indicators are 0.9866, 1, 1, 0.9866 and 5.1, respectively. It can be seen from Table 7 that the experimental results of the IBSSA-FKNN method on WDBC dataset are recorded in detail. During the 10-fold operation, the average values of the five evaluation indicators are 0.9965, 0.9955, 1, 1 and 5.5, respectively.

This section compares the IBSSA algorithm with other 5

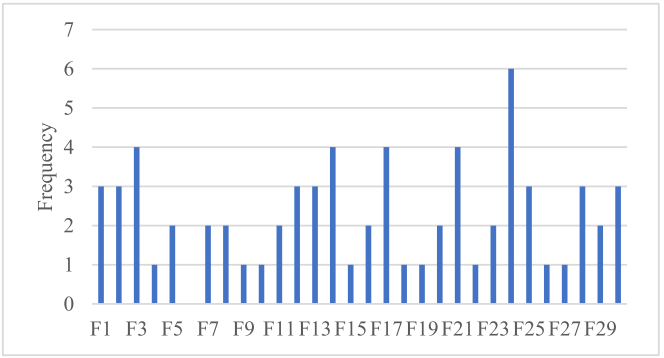

Fig. 8. The frequency of selected features in 10-fold CV on the WDBC dataset.

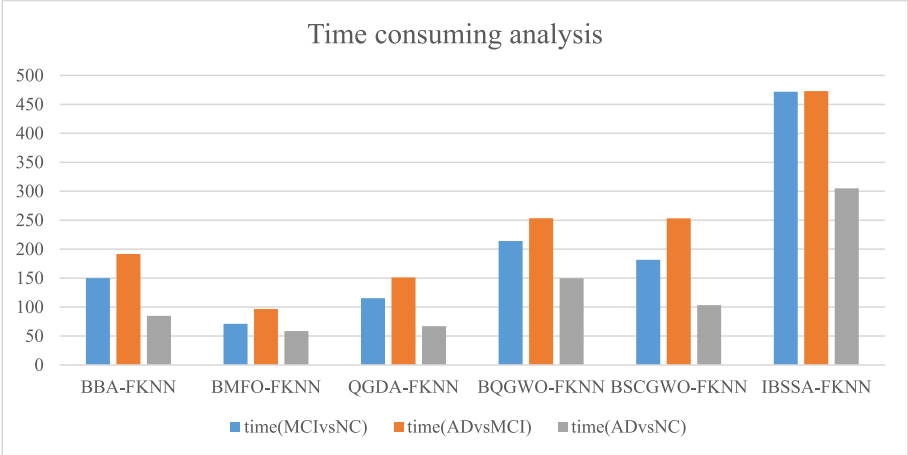

Fig. 9. Comparison of time consumption of different methods.

metaheuristic optimization algorithms on 5 different datasets to test its performance on feature selection problems. Tables 8–12 record the mean values of the selected feature length, classification accuracy, sensitivity, specificity, precision, and F-measure obtained by the BMFO, BBA, QGDA, BQGWO, BSCGWO, and IBSSA algorithms under the experiment of 10-fold crossover.

It can be seen from the experimental results in Tables 8–12 that for the IBSSA algorithm, only in the BreastCancer dataset, the sensitivity index is slightly inferior to other algorithms. On the four datasets, including grass, the algorithm achieves the best selected feature length, classification accuracy, sensitivity, specificity, precision, and F-measure. For example, on the BreastCancer dataset, the IBSSA algorithm obtained the optimal average number of feature selection 3.3, the optimal average classification accuracy 99.29%, the optimal average sensitivity 99.13%, the optimal average specificity 99.58%, the optimal average precision 99.79%, and the optimal average F-measure 99.45%. Experimental results show that the IBSSA algorithm improves the classification accuracy, sensitivity, specificity, precision, and F-measure of feature subsets to a certain extent. It is worth noting that although this algorithm does not perform very well in improving classification accuracy, they have a better performance in reducing the data dimension.

In order to explore how many and which features are selected in the feature selection process, we further conduct experiments on 5 datasets to investigate the details of the feature selection mechanism of the salp swarm optimization algorithm. Figs. 3–7 show the statistical diagram of the number of times each feature value is selected in the 10-fold cross-validation experiment of the IBSSA-FKNN method. From these figures, we can find that some features are selected more times, while some features are selected less times.

The 10-fold selection features in the BreastCancer dataset are shown in Fig. 4. The original number of features in the BreastCancer dataset is 9. After feature selection, not all features are selected for classification. The average number of selected features for the IBSSA-FKNN method is 3.3, and its most important features are F1, F3, F4, F7, i.e. bundle thickness, cell shape uniformity, edge adhesion, and chromatin, which have been selected more than 5 times, so we think that these features can be used as a reference for distinguishing breast cancer, which can be found in the selection feature's 10-fold CV.

The 10-fold selection features in the glass dataset are shown in Fig. 5. The original number of features in the glass dataset is 9. After feature selection, not all features are selected for classification. The average number of selected features of the IBSSA-FKNN method is 3.6, and its most important features are F1, F4, F6, F7, and F8, all of which have been selected more than 4 times or more, so we think that these features can be used as a reference for distinguishing. It can be found in the selection feature's 10-fold CV.

The 10-fold selection features in the hepatitisfulldata dataset are shown in Fig. 6. The original feature number in the hepatitisfulldata dataset is 19. After feature selection, not all features are selected for classification. The average number of selected features of the IBSSA-FKNN method is 3.3, and its most important features are F1, F2, F3, and F11, all of which have been selected more than 3 times or more, so we think that these features can be used as a reference for distinguishing, which can be found in the selection feature's 10-fold CV.

The 10-fold selection features in the Lymphography dataset are shown in Fig. 7. The original number of features in the Lymphography dataset is 17. After feature selection, not all features are selected for classification. The average number of selected features of the IBSSA-FKNN method is 3.6, and its most important features are F2, F7, F11, F13, and F14, all of which have been selected more than 4 times or more, so we think these features can be used as a reference for distinguishing. It can be found in the selection feature's 10-fold CV.

The 10-fold selection features on the WDBC dataset are shown in Fig. 8. The original number of features on the Wdbc dataset is 29. After feature selection, not all features are selected for classification. The average number of selected features of the IBSSA-FKNN method is 4.3, and its most important features are F1, F2, F3, F14, F17, F21, F24, which are all selected more than 3 times or more, so we think these features can be used as a reference for distinguishing. It can be found in the selection feature's10-fold CV.

Table 13
Subject information(mean ± std).

category	Number of subjects	Age	Years of Education	MMSE	ASAS-Cog
AD	51	75.2 ± 7.4	14.7 ± 3.6	23.8 ± 2.0	18.3 ± 6.0
NC	52	75.3 ± 5.2	15.8 ± 3.2	29.0 ± 1.2	7.4 ± 3.2
MCI-C	43	75.8 ± 6.8	16.1 ± 2.6	26.6 ± 1.7	12.9 ± 3.9
MCI-NC	56	74.7 ± 7.7	16.1 ± 3.0	27.5 ± 1.5	10.2 ± 4.3

6. sMRI dataset experiment

6.1. Dataset description

The experimental data are obtained from the international Alzheimer's disease neuroimaging initiative (ADNI) database (http://adni.loni.usc.edu/). ADNI was established in 2003 by the National Institute on Aging (NIA), the National Institute of Biomedical Imaging and Bioengineering (NIBIB), the Food and Drug Administration (FDA),

Table 14
Different methods classify AD/MCI, AD/NC, MCI/NC on multimodal data.

Algorithm	Features' size	ACC (%)	SEN (%)	SPE (%)	PRE (%)	F-measure (%)
AD vs. MCI						
IBSSA-FKNN	**11.5**	**0.9537**	**0.99**	**0.9233**	**0.9627**	**0.9657**
BBA-FKNN	73.2	0.5608	0.6656	0.36	0.6696	0.6638
BMFO-FKNN	117	0.8803	0.9289	0.79	0.8999	0.9109
QGDA-FKNN	44.7	0.9533	0.98	0.86	0.9409	0.9574
BQGWO-FKNN	8.5	**0.9667**	0.9889	0.92	0.9642	**0.9761**
BSCGWO-FKNN	**5.6**	0.9667	0.97	**0.94**	**0.9727**	0.9747
AD vs. NC						
IBSSA-FKNN	**11.4**	**1**	**1**	**1**	**1**	**1**
BBA-FKNN	76.5	0.8145	0.8	0.83	0.8223	0.8037
BMFO-FKNN	103.6	0.96	0.94	0.98	0.98	0.9578
QGDA-FKNN	32.3	0.97	0.98	0.96	0.9667	0.9707
BQGWO-FKNN	20.8	0.9909	0.98	1	1	0.9889
BSCGWO-FKNN	**2.6**	0.99	0.98	1	0.9817	0.9889
MCI vs. NC						
IBSSA-FKNN	**27.3**	**0.9395**	**0.9789**	**0.94**	**0.9718**	**0.9555**
BBA-FKNN	76.5	0.7432	0.8	0.6367	0.8100	0.8024
BMFO-FKNN	103.6	0.8686	0.9089	0.79	0.8994	0.8970
QGDA-FKNN	39.3	0.9252	0.9478	0.88	0.9436	0.9437
BQGWO-FKNN	21.6	0.9354	0.9589	0.8567	0.9292	0.9527
BSCGWO-FKNN	**5.8**	0.9137	0.92	0.9067	0.9496	0.9314

private pharmaceutical enterprises, and non-profit organizations. Its main goal is to test whether the progress of MCI and early AD can be measured by combining MRI, PET, other biomarkers, and clinical neuropsychological evaluation. The database contains data modalities, including MRI image data based on time series, PET image data, and other types of biomarker values, such as CSF, and some clinical neuropsychological assessment scores, such as the mini-mental state examination (MMSE) and the Alzheimer's disease assessment scale-cognitive (ADAS-Cog). These data categories are mainly: patients with early AD, patients with mild cognitive impairment (MCI), and the cognitive normal control group (NC). While mild cognitive impairment (MCI) is usually considered an early stage of AD, which is a transition state from normal control (NC) to AD, especially late-stage MCI is likely to develop into AD. Therefore, MCI is generally divided into MCI converted to AD (MCI patients who will convert to AD, MCIc) and MCI not converted to AD (MCI patients who will not convert to AD, MCInc). The subjects of the ADNI database were recruited from 50 websites across the United States and Canada. Their initial goal was to recruit 800 adult volunteers, ranging in age from 55 to 90 years old. Among them, 200 were elderly people with normal cognition in the follow-up test for three consecutive years, 400 were patients with mild cognitive impairment in the follow-up test for three consecutive years, and 200 were patients with AD in the follow-up test for two consecutive years. The personal basic information of these subjects can be obtained from the official website of ADNI.

In this paper, the sample data of subjects with MRI, PET, and CSF modalities are selected for the experiment, and only the data collected at the benchmark time point of these subjects are selected. In the International AD Database, 202 subjects have the above three modalities at the same time. Table 13 lists the demographic information of these subjects.

6.2. Experimental setup and description

This paper adopts a 10-fold cross-validation strategy to evaluate the classification performance of the proposed method. Specifically, the sample set is divided into 10 pieces on average, one of which is selected one by one as the test set, and the remaining 9 pieces are used as the training set. Calculate the features' size, average accuracy, sensitivity, specificity, and F-measure of these 10 experiments as the experimental results of one division. Then randomly exchange the order of the samples, divide the 10-fold cross validation once more, and calculate the features' size, average accuracy, sensitivity, specificity, and F-measure. Repeat the division 10 times, and calculate the features' size, average accuracy, sensitivity, specificity, and F-measure for these 10 divisions. The experiment adopts the two-class method (AD/MCI, AD/NC, and MCI/NC) to fully verify the influence of different classifications on the experimental results.

In order to verify the performance of the method proposed in this paper for the diagnosis of early AD, it is compared with five classification methods that are also the same as the swarm intelligence optimization algorithm combined with the FKNN classifier.

6.3. Experimental results and discussion

In order to verify the performance of the IBSSA-FKNN method proposed in this paper for early AD diagnosis, it is compared with other methods of swarm intelligence optimization combined with classifiers. The five classification methods are: a feature selection method based on

Table 15
Sample size and classification results of AD prediction and diagnosis methods.

references	methods	sample size	Accuracy(%)
Literature [49]	SVM with Gaussian kernel	Baseline MRI:198 AD,409MCI(pMCI and sMCI), 231 NC	Baseline MRI: AD vs. NC 87.9% pMCI vs. NC 83.2% pMCI vs. sMCI 70.4%
Literature [50]	Bagging algorithm and SVM	56 AD, 60 MCI,60 NC	AD vs. NC 89%% MCI vs. NC 72%
Literature [51]	deep full link network and stacked self-encoder	65 AD,67 cMCI, 102 ncMCI,77HC	AD vs. HC 87.76%% MCI vs. HC 76.92%
Literature [52]	multi-instance learning techniques of graph	198 AD,238 sMCI, 167 pMCI , 234 NC	AD vs. NC 88.8%% pMCI vs. sMCI 69.6%
Literature [53]	Independent Component Analysis (ICA) and SVM	202 AD, 410 MCI, 236 NC	75% of data in training set: AD vs. NC 78.4% MCI vs. NC 71.2% 90% of data in training set: AD vs. NC 85.7%% MCI vs. NC 79.2%

a binary bat algorithm combined with a fuzzy k-nearest neighbor classifier; Feature selection method based on a binary Moth-Flame Optimization combined with a fuzzy k-nearest neighbor classifier; Feature selection method based on binary Gaussian discriminant analysis algorithm combined with fuzzy k-nearest neighbor classifier; And two feature selection methods based on binary improved grey wolf optimization algorithm combined with a fuzzy k-nearest neighbor classifier.

Table 14 shows the experimental results of the performance comparison between the IBSSA-FKNN method and the other five methods on the concatenated multimodal data, respectively, for classifying AD/NC, AD/MCI, and MCI/NC. In Table 14, IBSSA-FKNN indicates that the binary salp swarm algorithm is first used for feature selection, and then the FKNN classification model is used for classification experiments. Other methods are the same. Among them, all the experimental results listed in Table 14 are the average value of each index divided by 10 times of 10-fold cross-validation.

The experimental results in Table 14 show that employing the feature selection step can improve the performance of the classification model in diagnosing early AD. The classification accuracy of IBSSA-FKNN method for AD and MCI, AD and NC, MCI and NC is 95.37%, 100%, and 93.95%, respectively. From the six indicators of the three groups of classification results, the methods proposed in this paper are better than the other five methods. At the same time, the advantages of BQGWO-FKNN and BSCGWO-FKNN methods are also obvious, second only to IBSSA-FKNN method. In AD/NC and MCI/NC classification experiments, BSCGWO-FKNN method is superior to IBSSA-FKNN in selecting the number of feature subsets, but ranks first in other indicators. In the AD/MCI classification experiment, IBSSA-FKNN method is slightly lower than BQGWO-FKNN or BSCGWO-FKNN method in a certain index. The experimental results show that IBSSA-FKNN is still better than other methods in the 3 groups of classification experiments. Based on the experimental analysis results in Table 2 above, the following conclusions can be drawn: The FKNN feature selection method based on the Salp Swarm Algorithm proposed in this paper can significantly improve the classification performance of only using the FKNN classifier. Compared with other swarm intelligence methods combined with FKNN classifier, the method proposed in this paper improves various indicators such as classification accuracy, sensitivity and specificity. Among them, the improvement of AD/MCI classification performance is particularly significant, and the combination of FKNN classifier can achieve higher classification performance, so the IBSSA-FKNN method proposed in this paper can be well applied to the diagnosis of early AD.

From another point, this paper analyzes the time consumption of different algorithms on AD classification, as shown in Fig. 9. It can be seen from the figure that the classification method proposed in this question takes a relatively long time. From the results of the above indicators, this method is superior to other methods in classification accuracy and other indicators. Next, I will continue to explore how to ensure accuracy while saving time. It takes longer than other classification techniques because of improvement approach 4, namely, the dimensional random difference mutation. Since this technique requires mutating each aspect of the individual and then judging and screening the outcomes. The blindness of the mutation process is certain to result in a decrease in the algorithm's search efficiency and a considerable increase in the quantity of computation. This method, however, can cause the algorithm to deviate from the local optimum solution, boosting the accuracy of AD classification.

In recent years, scholars have proposed many diagnostic algorithms for AD. Since these algorithms use different databases and different preprocessing methods, it is difficult to directly conduct comparative experiments. Therefore, relevant algorithms that perform well in different sample numbers are selected for comparison. Table 15 lists the sample size and classification results of each algorithm. Janoušová E et al. combined penalty regression data resampling to extract features and classify data by using SVM with Gaussian kernel [48]. Batmanghelich N et al. used Bagging algorithm and SVM for AD/NC classification, and logistic regression model using Boosting algorithm was used for MCI/NC classification [49]. Liu S et al. realized the diagnosis and prediction of AD by using deep full link network and stacked self-encoder [50]. Tong T et al. used multi-instance learning techniques of graph to classify samples by extracting local density blocks as features [51]. Yang W et al. used Independent Component Analysis (ICA) for feature extraction and combines it with SVM algorithm for AD prediction [52].

7. Conclusion and future work

In this study, we propose a FKNN feature selection method based on the binary salp swarm algorithm and apply this method to the early diagnosis of AD. First, on the datasets of BreastCancer, glass, hepatitisfulldata, Lymphography, and WDBC obtained from the UCI Machine Learning Repository, the effectiveness of this method is tested in terms of classification accuracy, sensitivity, and specificity and other aspects. Second, in order to verify the effectiveness of this method in the diagnosis of early AD, the multimodal feature data of MRI, PET and CSF from the international AD neuroimaging initiative (ADNI) were used and compared with other swarm intelligence algorithms combined with the FKNN classifier. The experimental results show that the proposed IBSSA-FKNN method is superior to the other five FKNN models based on swarm intelligence algorithms in various performance indicators and that can effectively improve the classification performance and the performance of early AD diagnosis. The promising application prospect will bring great convenience for clinicians to make better decisions in clinical diagnosis.

On the one hand, future research will expand the suggested strategy to considerably bigger datasets. Second, the approach suggested in this study may be developed further to improve the AD classification impact. Next, I'll aim to integrate deep learning with a swarm intelligence optimization method and apply it to the early detection of Alzheimer's disease in order to accomplish AD classification. On the other hand, this work only focuses on a small quantity of labeled training data, although there is a large amount of unlabeled multimodal data accessible in clinic. Also, there are a considerable amount of incomplete multimodal data in clinic. Making full use of this incomplete multimodal labeled data may not only enhance the quantity of training samples, but also build learning methods for incomplete multimodal data, which can improve the model's promotion performance. In a nutshell, the experiment gives a useful research idea and algorithm for the study of Alzheimer's disease, and it demonstrates that the swarm intelligence optimization algorithm has a positive influence on the early detection of Alzheimer's disease.

Data availability

The data used to support the findings of this study are included in the article.

Declaration of competing interest

The authors declare that they have no conflicts of interest.

Acknowledgments

Dongwan Lu and Yinggao Yue contributed equally to this work and should be considered as co-first authors. This work was supported in part by the Natural Science Foundation of Zhejiang Province under Grant LY23F010002, in part by Wenzhou basic scientific research project under Grant R20210030 and Service science and technology innovation project of Wenzhou Science and Technology Association under Grant kjfw36, the general scientific research projects of Zhejiang Provincial Department of Education under Grant Y202250103, in part by Major

scientific and technological innovation projects of Wenzhou Science and Technology Plan under Grant ZG2021021, School Level Scientific Research Projects of Wenzhou University of Technology under grants ky202201 and ky202209, the Teaching Reform Research Project of Wenzhou University of Technology under grant 2022JG12, Major Project of Zhejiang Natural Science Foundation under Grant LD21F020001, Grant LSZ19F020001, and the National Natural Science Foundation of China under Grant U1809209, Wenzhou Intelligent Image Processing and Analysis Key Laboratory Construction Project under Grant 2021HZSY007105.

References

[1] J. Weller, A. Budson, Current understanding of Alzheimer's disease diagnosis and treatment[J], F1000Research 7 (7) (2018) 1–9.
[2] J. Rasmussen, H. Langerman, Alzheimer's disease–why we need early diagnosis[J], Degener. Neurol. Neuromuscul. Dis. 9 (12) (2019) 123–130.
[3] D. Lu, K. Popuri, G.W. Ding, et al., Multimodal and multiscale deep neural networks for the early diagnosis of Alzheimer's disease using structural MR and FDG-PET images[J], Sci. Rep. 8 (1) (2018) 1–13.
[4] M. Gharaibeh, M. Almahmoud, M.Z. Ali, et al., Early diagnosis of alzheimer's disease using cerebral catheter angiogram neuroimaging: a novel model based on deep learning approaches[J], Big Data and Cognitive Computing 6 (1) (2022) 2.
[5] S. Singh, R.R. Janghel, Early diagnosis of alzheimer's disease using aco optimized deep cnn classifier[C]//Ubiquitous Intelligent Systems, in: Proceedings of ICUIS 2021, Springer, Singapore, 2022, pp. 15–31.
[6] D. Pan, G. Luo, A. Zeng, et al., Adaptive 3DCNN-Based Interpretable Ensemble Model for Early Diagnosis of Alzheimer's Disease[J], IEEE Transactions on Computational Social Systems, 2022.
[7] S. Velliangiri, S. Pandiaraj, S. Muthubalaji, Multiclass recognition of AD neurological diseases using a bag of deep reduced features coupled with gradient descent optimized twin support vector machine classifier for early diagnosis[J], Concurrency Comput. Pract. Ex. 34 (21) (2022), e7099.
[8] J. Seo, T.H. Laine, G. Oh, et al., EEG-based emotion classification for Alzheimer's disease patients using conventional machine learning and recurrent neural network models[J], Sensors 20 (24) (2020) 7212–7225.
[9] T. Zheng, Y. Yu, H. Lei, et al., Compositionally graded KNN-based multilayer composite with excellent piezoelectric temperature stability[J], Adv. Mater. 34 (8) (2022), 2109175.
[10] A. Shokrzade, M. Ramezani, F.A. Tab, et al., A novel extreme learning machine based kNN classification method for dealing with big data[J], Expert Syst. Appl. 183 (11) (2021), 115293.
[11] Y. Huang, Y. Li, Prediction of protein subcellular locations using fuzzy k-NN method[J], Bioinformatics 20 (1) (2004) 21–28.
[12] K.C. Kwak, W. Pedrycz, Face recognition using a fuzzy fisherface classifier[J], Pattern Recogn. 38 (10) (2005) 1717–1732.
[13] H.L. Chen, C.C. Huang, X.G. Yu, et al., An efficient diagnosis system for detection of Parkinson's disease using fuzzy k-nearest neighbor approach[J], Expert Syst. Appl. 40 (1) (2013) 263–271.
[14] A. Mondal, S. Ghosh, A. Ghosh, Efficient silhouette-based contour tracking using local information[J], Soft Comput. 20 (2) (2016) 785–805.
[15] W. Shan, Z. Qiao, A.A. Heidari, et al., Double adaptive weights for stabilization of moth flame optimizer: balance analysis, engineering cases, and medical diagnosis [J], Knowl. Base Syst. 214 (2) (2021), 106728.
[16] S. Wang, Y. Zhao, J. Li, et al., Neurostructural correlates of hope: dispositional hope mediates the impact of the SMA gray matter volume on subjective well-being in late adolescence[J], Soc. Cognit. Affect Neurosci. 15 (4) (2020) 395–404.
[17] Y. Zhang, R. Liu, A.A. Heidari, et al., Towards augmented kernel extreme learning models for bankruptcy prediction: algorithmic behavior and comprehensive analysis[J], Neurocomputing 430 (3) (2021) 185–212.
[18] S. Jiao, G. Chong, C. Huang, et al., Orthogonally adapted Harris hawks optimization for parameter estimation of photovoltaic models[J], Energy 203 (7) (2020), 117804.
[19] J. Tu, H. Chen, J. Liu, et al., Evolutionary biogeography-based whale optimization methods with communication structure: towards measuring the balance[J], Knowl. Base Syst. 212 (1) (2021), 106642.
[20] S. Song, P. Wang, A.A. Heidari, et al., Dimension decided Harris hawks optimization with Gaussian mutation: balance analysis and diversity patterns[J], Knowl. Base Syst. 215 (3) (2021), 106425.
[21] Y. Fan, P. Wang, M. Mafarja, et al., A bioinformatic variant fruit fly optimizer for tackling optimization problems[J], Knowl. Base Syst. 213 (2) (2021), 106704.
[22] X. Zhang, Y. Xu, C. Yu, et al., Gaussian mutational chaotic fruit fly-built optimization and feature selection[J], Expert Syst. Appl. 141 (3) (2020), 112976.
[23] W. Zhu, C. Ma, X. Zhao, et al., Evaluation of sino foreign cooperative education project using orthogonal sine cosine optimized kernel extreme learning machine [J], IEEE Access 8 (3) (2020) 61107–61123.
[24] G.Q. Zeng, X.Q. Xie, M.R. Chen, et al., Adaptive population extremal optimization-based PID neural network for multivariable nonlinear control systems[J], Swarm Evol. Comput. 44 (2) (2019) 320–334.
[25] M.R. Chen, G.Q. Zeng, K.D. Lu, et al., A two-layer nonlinear combination method for short-term wind speed prediction based on ELM, ENN, and LSTM[J], IEEE Internet Things J. 6 (4) (2019) 6997–7010.
[26] W. Deng, H. Liu, J. Xu, et al., An improved quantum-inspired differential evolution algorithm for deep belief network[J], IEEE Trans. Instrum. Meas. 69 (10) (2020) 7319–7327.
[27] W. Deng, J. Xu, H. Zhao, et al., A novel gate resource allocation method using improved PSO-based QEA[J], IEEE Trans. Intell. Transport. Syst. 23 (3) (2020) 1737–1745.
[28] J. Pang, H. Zhou, Y.C. Tsai, et al., A scatter simulated annealing algorithm for the bi-objective scheduling problem for the wet station of semiconductor manufacturing[J], Comput. Ind. Eng. 123 (9) (2018) 54–66.
[29] B. Venkatesh, J. Anuradha, A review of feature selection and its methods[J], Cybern. Inf. Technol. 19 (1) (2019) 3–26.
[30] G.G. Wang, S. Deb, Z. Cui, Monarch butterfly optimization[J], Neural Comput. Appl. 31 (7) (2019) 1995–2014.
[31] S. Li, H. Chen, M. Wang, et al., Slime mould algorithm: a new method for stochastic optimization[J], Future Generat. Comput. Syst. 111 (2020) 300–323.
[32] G.G. Wang, Moth search algorithm: a bio-inspired metaheuristic algorithm for global optimization problems[J], Memetic Computing 10 (2) (2018) 151–164.
[33] Y. Yang, H. Chen, A.A. Heidari, et al., Hunger games search: visions, conception, implementation, deep analysis, perspectives, and towards performance shifts[J], Expert Syst. Appl. 177 (2021), 114864.
[34] J.C. Butcher, A history of Runge-Kutta methods[J], Appl. Numer. Math. 20 (3) (1996) 247–260.
[35] J. Tu, H. Chen, M. Wang, et al., The colony predation algorithm[J], J. Bionic Eng. 18 (3) (2021) 674–710.
[36] I. Ahmadianfar, A.A. Heidari, S. Noshadian, et al., INFO: an efficient optimization algorithm based on weighted mean of vectors[J], Expert Syst. Appl. 195 (2022), 116516.
[37] A.A. Heidari, S. Mirjalili, H. Faris, et al., Harris hawks optimization: algorithm and applications[J], Future Generat. Comput. Syst. 97 (2019) 849–872.
[38] S. Mirjalili, A.H. Gandomi, S.Z. Mirjalili, et al., Salp Swarm Algorithm: a bio-inspired optimizer for engineering design problems[J], Adv. Eng. Software 114 (12) (2017) 163–191.
[39] M. Emmanuel, J. Jabez, An enhanced fuzzy based KNN classification method for Alzheimer's disease identification from SMRI images[J], JOURNAL OF ALGEBRAIC STATISTICS 13 (3) (2022) 89–103.
[40] H. Abbad Ur Rehman, C.Y. Lin, Z. Mushtaq, et al., Performance analysis of machine learning algorithms for thyroid disease[J], Arabian J. Sci. Eng. 46 (10) (2021) 9437–9449.
[41] J. Feng, J. Zhang, X. Zhu, et al., A novel chaos optimization algorithm[J], Multimed. Tool. Appl. 76 (16) (2017) 17405–17436.
[42] S.A. Mirjalili, S.Z.M. Hashim, BMOA: binary magnetic optimization algorithm[J], International Journal of Machine Learning and Computing 2 (3) (2012) 204.
[43] R.Y.M. Nakamura, L.A.M. Pereira, K.A. Costa, et al., BBA: a binary bat algorithm for feature selection[C], in: 2012 25th SIBGRAPI Conference on Graphics, Patterns and Images, IEEE, 2012, pp. 291–297.
[44] A. Patil, G. Soni, A. Prakash, A BMFO-KNN based intelligent fault detection approach for reciprocating compressor[J], International Journal of System Assurance Engineering and Management 13 (2) (2022) 797–809.
[45] C. Yu, Z. Cai, X. Ye, et al., Quantum-like mutation-induced dragonfly-inspired optimization approach[J], Math. Comput. Simulat. 178 (2020) 259–289.
[46] E. Emary, H.M. Zawbaa, A.E. Hassanien, Binary grey wolf optimization approaches for feature selection[J], Neurocomputing 172 (2016) 371–381.
[47] J. Hu, A.A. Heidari, L. Zhang, et al., Chaotic diffusion-limited aggregation enhanced grey wolf optimizer: insights, analysis, binarization, and feature selection[J], Int. J. Intell. Syst. 37 (8) (2022) 4864–4927.
[48] E. Janoušová, M. Vounou, R. Wolz, et al., Biomarker discovery for sparse classification of brain images in Alzheimer's disease[J], Annals of the BMVA (2) (2012) 1–11.
[49] N. Batmanghelich, B. Taskar, C. Davatzikos, A general and unifying framework for feature construction, in: Image-Based Pattern classification[C]//International Conference on Information Processing in Medical Imaging, vol. 5636, Springer, Berlin, Heidelberg, 2009, pp. 423–434.
[50] S. Liu, S. Liu, W. Cai, et al., Early diagnosis of Alzheimer's disease with deep learning[C], in: 2014 IEEE 11th International Symposium on Biomedical Imaging (ISBI), IEEE, 2014, pp. 1015–1018, 7.
[51] T. Tong, R. Wolz, Q. Gao, et al., Multiple instance learning for classification of dementia in brain MRI[J], Med. Image Anal. 18 (5) (2014) 808–818.
[52] W. Yang, R.L.M. Lui, J.H. Gao, et al., Independent component analysis-based classification of Alzheimer's disease MRI data[J], J. Alzheim. Dis. 24 (4) (2011) 775–783.