Neural Computing and Applications (2023) 35:12565–12581
https://doi.org/10.1007/s00521-023-08354-x

ORIGINAL ARTICLE

Determination of wheat types using optimized extreme learning machine with metaheuristic algorithms

Musa Dogan[1] · Ilker Ali Ozkan[1]

Received: 8 April 2022 / Accepted: 13 February 2023 / Published online: 4 March 2023

Abstract

In order to increase the market value and quality of wheat, it is important to separate different types and determine the amount of foreign matter using the visual properties of durum and bread wheat. In this study, the extreme learning machine (ELM) algorithm, which is often preferred in real-time applications, was used to make classifications using features obtained from images containing the wheat kernel and foreign matter. The feature selection process was applied to remove the irrelevant ones from the obtained 236 features. In addition, the Harris hawks' optimizer (HHO), a novel method in the literature, and the particle swarm optimizer (PSO), one of the well-known algorithms, were used to improve the ELM model. As part of this study, new models called HHO-ELM and PSO-ELM were created and compared with the original ELM model and other artificial neural networks (ANNs) studies published in the literature. As a result, in comparison with other models, the optimized ELM models demonstrated good stability and accuracy, having 99.32% in binary classification and 95.95% in multi-class classification.

Keywords Extreme learning machine · Harris hawks optimizer · Optimization · Particle swarm optimization · Wheat classification

1 Introduction

Wheat, which is produced in most of the world, is an important product in terms of forming the basic food raw material [1]. Wheat production is also increasing due to population growth. It is important that this produced wheat is released in quality in accordance with market demands. The quality of the wheat determines the quality of the flour, which is used to make final products such as bread, pasta and cakes. The amount of protein in wheat is one of the most important factors impacting its quality [2]. Since durum wheat contains higher protein, combining bread wheat with durum wheat reduces protein content [3]. For this reason, wheat goes through various processes to get the quality product label until it is released. For these reasons, the classification of wheat kernels as bread or durum wheat based on their visual features has become more important in recent years [4].

Computer vision systems are being developed to identify and accurately classify product varieties in the field of agriculture [5]. Attributes are determined by image processing techniques using product images obtained with computer vision systems [6]. In many studies, artificial intelligence techniques can create automatic evaluation systems from the field of agriculture [5]. With these developed systems, it is possible to make classification with high accuracy and without the need for human intervention [7].

Different classifier systems have been created that include computerized vision-based artificial intelligence techniques to determine product varieties. Sabanci et al. [3] developed an artificial neural network (ANN)-based system to classify wheat kernels. They created an ANN model using 21 attributes obtained using image processing techniques. 9.8×10^{-6} mean absolute error (MAE) was obtained using seven inputs from the CfsSubsetEval feature

This research is based in part on the Musa Dogan's master's thesis under direction of the Ilker Ali Ozkan.

✉ Ilker Ali Ozkan
ilkerozkan@selcuk.edu.tr

Musa Dogan
musa.dogan@selcuk.edu.tr

[1] Department of Computer Engineering, Faculty of Technology, Selcuk University, 42031 Konya, Turkey

selection algorithm. Pourezza et al. [8] examined various textural property groups of seed images to assess its effectiveness in identifying nine wheat seed varieties. They achieved a 98.15% accuracy rate with 50 features obtained using image processing on the basis of, a novel method in the literature and the particle swarm optimizer (PSO), one textural analysis and stated that the textural features are well identifiers to define wheat varieties. Güneş et al. [9] developed an automatic system to classify wheat varieties in Turkey using image analysis techniques. They used gray-level co-occurrence matrix (GLCM) and linear binary pattern (LBP) for texture analysis and k-nearest neighbors (k-NN) for the classifier and stated the importance of the addition of controlled lighting conditions and color characteristics for the classifier. In another study, wheat and barley kernels were classified. They show that a high classifier accuracy with k-NN can be obtained. It has also been stated that the combination of different features such as morphological, color and texture provides better accuracy than a single feature [10]. Yaşar et al. [11] developed an ANN-based model to classify three different wheat varieties. They show that the model with architecture 7–10-1 was successful in the classification process. Aslan et al. [12] determined wheat kernels using ANN and extreme learning machine (ELM). As a result of the comparison of the models created in the study, ELM showed both faster training and higher accuracy values. ANN, k-NN and support vector machines (SVMs) models are often used in the wheat classification literature [13–16]. In addition, in recent years, various artificial intelligence algorithms and optimization of these algorithms have been carried out in both wheat grain and seed classification in agriculture [5, 17–20].

In recent years, the use of ELM has been proposed to train single-hidden-layer feed-forward neural networks (SLFNs). The output weights are determined analytically with simple generalized reverse operation while input weights and biases, which are learning parameters in ELM, are randomly assigned and do not need to be tuned [21]. Therefore, ELM performs training much faster than traditional algorithms because it does not need iteration [22]. In the field of wheat classification, generally, the features obtained from camera images were used. Especially in this area, ANN algorithms are used very often [3, 23–25]. Instead of the commonly used ANN methods, this research uses an improved ELM classifier design assisted by computer vision to classify wheat kernels as bread or durum, based on their visual properties. The ELM algorithm is important because it is often preferred in real-time hardware applications and does not require heavy computational overhead. Due to successful applications of ELM, researchers are also attracted to utilize it in the field of agriculture such as Prasad et al. [26] designed and tested ELM models for soil moisture (SM). Complete ensemble empirical mode decomposition with adaptive noise (CEEMDAN) and ensemble empirical mode decomposition (EEMD) algorithms were used to create hybrid ELM models, and these models were compared with random forest (RF). ELM outperformed RF in estimating upper and lower layer SM in all different trials. With the study, potentially real-time automated forecasting systems have been developed to estimate the soil moisture of the self-adaptive model hybridized with the ELM algorithm. Kouadio et al. [27] used soil fertility data for the first time to show that ELM is more effective in estimating Robusta coffee yield than RF and multiple linear regression (MLR) algorithms. Sulistyo et al. [28] used deep sparse ELM (DSELM) to distinguish wheat leaves from unwanted images such as soil, weeds, dry leaves, stems and stones. They concluded that unlike other learning algorithms such as backpropagation-based multilayer perceptron (MLP), DSELM learns much faster. At the same time, it has been optimized with genetic algorithm (GA), which is one of the evolutionary algorithms to predict nitrogen status in wheat plants for agricultural automation. Suchithra et al. [29] used ELM with different activation functions to classify and predict soil fertility indices and pH values. They tried to find the most optimal model by determining the number of neurons and activation functions in the hidden layer as meta-parameters. In the study, the authors found that optimizing the ELM parameters contributes to the development of a suitable model for predicting soil fertility indexes. Feng et al. [30] developed a semi-supervised ELM (SS-ELM) framework for land cultivation and classification of agricultural cultivation structure. With this framework, image segmentation, self-labeling and classification processes are performed based on remote data. Results were compared with RF, ELM, SVM and semi-supervised SVM (S-SVM). They achieved an average accuracy of 83.00–92.17% in four different classes using SS-ELM. Mostafaeipour et al. [31] applied ELM and support vector regression (SVR) methods to estimate the output energy of wheat production in the Estahban region, which is a problem of single-output multi-class classification. The ELM method has been shown to have a much smaller error than SVR, provides more accurate predictions and is also faster.

The developed model can give non-optimal results due to the random selection of input weight and the hidden bias in the ELM, which decreases the model's generalization capability. In order to overcome this disadvantage, evolutionary algorithms that derived from natural selection such as differential evolution (DE) [32–34], genetic algorithm (GA) [35], swarm intelligence-based algorithms that mimic the social behavior of swarms such as PSO [36], artificial bee colony (ABC) [37], gray wolf optimization (GWO)

[38, 39] and other optimization algorithms [40, 41] with global search capability have been used. In this study, the recently proposed swarm intelligence-based HHO algorithm inspired by the hunting strategies of the Harris hawks was used. HHO is effective in discovering optimal solutions to multidimensional problems and superior to some other optimization algorithms [42–45]. HHO-optimized ELM has been applied in various fields since its introduction; to the best of the authors' knowledge, this is the first time it has been applied for wheat classification in agriculture. Therefore, this study deals with optimization of the weights and biases of the network with HHO and PSO algorithms (called HHO-ELM and PSO-ELM algorithm) to improve the prediction accuracy and stability of ELM.

When the literature studies are examined, there are studies in which ELM is used as well as other machine learning algorithms in wheat classification and similar classification problems. In addition, studies with different optimization techniques have been carried out in recent years in order to improve ELM performance.

In the production of food products such as bread and pasta, the type of wheat directly impacts the quality of flour. The main objective of this study, using ELM-based neural network models to separate different quality varieties according to visual characteristics of wheat and to successfully separate them from foreign materials. As an innovative approach to the classification of wheat, metaheuristic optimization algorithms will be hybridized with ELM. Therefore, the use of ELM's optimization for the wheat problem is expected by the authors to bring a different perspective to this problem.

The main contributions and motivations of this study are as follows:

(1) Wheat type classification and real-time operation using embedded technologies in the industry are important for production. Therefore, the ELM model outperforms traditional machine learning methods in terms of learning speed.
(2) Individual experiments were carried out, to distinguish the wheat from foreign substances as a binary classification problem and to determine types of wheat as multi-class classification problem.
(3) This paper focuses on ELM's metaheuristic optimization, which consists of PSO and HHO, to improve the performance of the ELM model used in both binary and multi-class classification problems.
(4) Further, the performance of ELM and improved ELM models was compared across all experiments in the study.
(5) Finally, the performance of the models created in this study was compared with another study conducted earlier those proposed solutions for the same problem.

The rest of the sections of this article is organized as follows. In Sect. 2, ELM, HHO and PSO are briefly introduced. Section 3 provides a detailed description of the proposed HHO-ELM and PSO-ELM algorithms for wheat classification. The experiments and results are given and discussed in Sect. 4. Finally, conclusions and future works are given in Sect. 5.

2 Preliminary

2.1 Extreme learning machine

Extreme learning machine (ELM) is a feed-forward learning algorithm with a single hidden layer that stands out for its performance in both regression and classification problems [46]. Just as in the studies [47, 48], neuron weights in the hidden layer are determined randomly and do not need to be tuned. The neurons in the hidden layer are randomly determined and do not need to be tuned. Activation functions such as sigmoid, sin and Gaussian are used for neurons in the hidden layer, and a linear activation function is used for neurons in the output layer. As shown in Fig. 1, the ELM structure consists of the input, hidden and output layers. In the following, ELM will be briefly introduced.

When compared to traditional neural network-based methods, ELM has the advantages of fast learning and high performance. A mathematical representation of a model with M neurons in a hidden layer and N samples in an ELM network is given in Eq. (1).

$$o_j = \sum_{i=1}^{M} \beta_i g\left(w_i \cdot x_j + b_i\right),\ j = 1, 2, \ldots, N. \tag{1}$$

where o_j is output vector of SLFN, β_i is the output weights, $g(\cdot)$ is the activation function, w_i is the input weights and b_i is the hidden bias value. If the hidden layer output matrix in

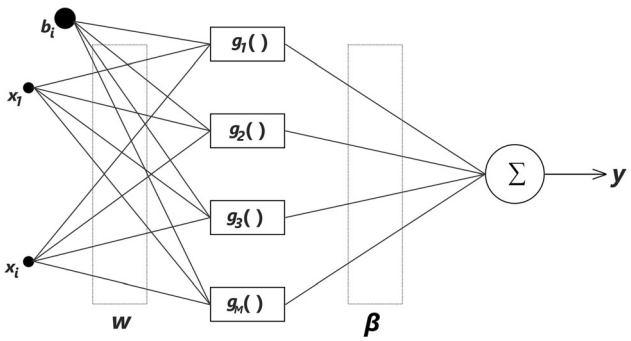

Fig. 1 Structure of basic ELM model

the previous equation is written as $H = g(w_i \cdot x_j + b_i)$, we can express Eq. (1) as follows:

$$H\beta = y \tag{2}$$

The explicit matrix H is given in Eq. (3), and the values β and y are given in Eq. (4).

$$\begin{bmatrix} g(w_1 \cdot x_1 + b_1) & \cdots & g(w_M \cdot x_1 + b_M) \\ \vdots & \cdots & \vdots \\ g(w_1 \cdot x_N + b_1) & \cdots & g(w_M \cdot x_N + b_M) \end{bmatrix}_{NxM}, \tag{3}$$

$$\beta = \begin{bmatrix} \beta_1 \\ \vdots \\ \beta_M \end{bmatrix}_{Mx1}, y = \begin{bmatrix} y_1 \\ \vdots \\ y_N \end{bmatrix}_{Nx1} \tag{4}$$

The target t_i^k is defined as

$$t_i^k = \begin{cases} 1 \text{ if } c_i = k \\ -1 \text{ otherwise}, k = 1, 2, \ldots, C \end{cases}^T \tag{5}$$

where c_i is the class label for x_i.

In the ELM approach, w_i and b_i values are randomly assigned, and output weights are calculated by Eq. (6).

$$\hat{\beta} = H^{\dagger} y \tag{6}$$

where $\widehat{\beta}$ is the approximate output weight and $H^{\dagger}$ is the generalized inverse Moore–Penrose (MP) [46] matrix of the H matrix used to determine the output weight.

The original ELM algorithm can be split into three steps:

Step 1: Define the number of hidden layer neurons and randomly assign input parameters (w_i and b_i).

Step 2: Calculate the H hidden layer output matrix.

Step 3: Calculate the output weight $\widehat{\beta}$ according to Eq. (6).

2.2 Harris Hawks optimization

The Harris hawks optimization (HHO) algorithm, based on the swarm intelligence inspired by the collaboration of Harris hawks with other hawks, goes after their prey was developed by Heidari et al. [42]. It is a population-based algorithm that uses an individual population to investigate promising regions in a search area. This algorithm, which is new in the literature, has been noticed to be a powerful aspects such as fast convergence, bypass local optima and has been used in different applications. [42, 49, 50]. In this algorithm, hawks cooperatively try to surprise and attack their prey in different ways. In addition, Harris hawks have the ability to choose the type of chase based on the prey's escape patterns [43]. Harris hawks trap their prey and act through the phases of exploration and exploitation.

2.2.1 Exploration phase

In HHO, each hawk is the candidate solution, and the best candidate solution at every step is considered targeted prey or convergence to optimum. Hawks perch and wait in random positions, trying to detect their prey with two different strategies. The first of these strategies (q) is to be able to stand in a certain distance to both family members and the prey in the event of an attack on any prey; this strategy is indicated as $q < 0.5$. The other strategy is the perching of hawks to attack on random tall trees within nest range, which is indicated as $q \geq 0.5$. These two strategies are given in Eq. (7).

$$X(i+1) = \begin{cases} X_{\text{rand}}(i) - r_1 |X_{\text{rand}}(i) - 2r_2 X(i)| & q \geq 0.5 \\ (X_{\text{rabbit}}(i) - X_m(i)) - r_3(LB + r_4(UB - LB)) & q < 0.5 \end{cases} \tag{7}$$

where i is the current iteration, X_{rand} is the random selected hawk, X_{rabbit} is the location of the rabbit, r_1, r_2, r_3 and r_4 random values between 0 and 1, X_m indicates the average position of the hawks and LB and UB show the upper and lower bounds of variables. The average position of hawks can be simply calculated as in Eq. (8).

$$X_m(i) = \frac{1}{N} \sum_{i=1}^{N} X_i(i) \tag{8}$$

where the position of the hawks is indicated by X_i, while N shows the total number of hawks.

2.2.2 Transition phase

In the HHO algorithm, the hawks will go into operation after their exploration. When they go into operation, they can engage in different operational behaviors depending on the escape energy of the prey. During the escape, the escape energy of the prey decreases. This context is modeled by Eq. (9).

$$E = 2E_0 \left(1 - \frac{i}{T}\right) \tag{9}$$

where E is the scape energy of the prey, T maximum number of iterations and E_0 indicates the initial energy. E_0 changes in the range $[-1, 1]$ for each iteration. While $|E| \geq 1$, hawks tend to explore different locations, and hence HHO operates the exploration phase. In contrast, when $|E| < 1$, hawks will go into the exploitation phase within the area (neighborhood) and seek solutions. In short, it processes the exploration phase $|E| \geq 1$, and the exploitation phase when $|E| < 1$.

2.2.3 Exploitation phase

At this stage, the Harris hawks make surprise attacks on the prey, which they have detected in the previous stage. However, the prey will try to escape dangerous situations too. In real situations, this causes the hawks to follow different tracking strategies. In the exploitation phase of the HHO algorithm, there are also 4 different attack models to suit this situation:

(1) Soft besiege.

$$X(i+1) = \Delta X(i) - E|JX_{\text{rabbit}}(i) - X(i)| \quad (10)$$

$$\Delta X(i) = X_{\text{rabbit}}(i) - X(i) \quad (11)$$

where ΔX is the difference between the position vector of the prey in the current iteration and J refers to the random value that mimics the jumping power of the prey during the escape process.

(2) Hard besiege.

$$X(i+1) = X_{\text{rabbit}}(i) - E|\Delta X(i)| \quad (12)$$

(3) Soft besiege with progressive rapid dives.

$$X(i+1) = \begin{cases} Y & \text{if } F(Y) < F(X(i)) \\ Z & \text{if } F(Z) < F(X(i)) \end{cases} \quad (13)$$

$$Y = X_{\text{rabbit}}(i) - E|JX_{\text{rabbit}}(i) - X(i)| \quad (14)$$

$$Z = Y + S \times LF(D) \quad (15)$$

where S is the one-dimensional random vector, D is the size of the problem and LF refers to the levy flight function, which is calculated as Eq. 16

$$LF(x) = 0.01 \times \frac{u \times \sigma}{|v|^{\frac{1}{\beta}}}, \sigma = \left(\frac{\Gamma(1+\beta) \times \sin\left(\frac{\pi\beta}{2}\right)}{\Gamma\left(\frac{1+\beta}{2}\right) \times \beta \times 2^{\left(\frac{\beta-1}{2}\right)}} \right)^{\frac{1}{\beta}} \quad (16)$$

where β is the constant value and u and v refer to randomly given values in the range [0, 1].

(D) Hard besiege with progressive rapid dives.where $F(.)$ is the objective function.

$$X(i+1) = \begin{cases} Y & \text{if } F(Y) < F(X(i)) \\ Z & \text{if } F(Z) < F(X(i)) \end{cases} \quad (17)$$

$$Y = X_{\text{rabbit}}(i) - E|JX_{\text{rabbit}}(i) - X_m(i)| \quad (18)$$

$$Z = Y + S \times LF(D) \quad (19)$$

In Algorithm 1, the HHO algorithm's pseudocode is presented [42].

Algorithm 1 The pseudo-code of the HHO Algorithm

```
//Input: The population size N and the number of iterations T
//Output: The fitness value of the best hawk
Initialize a population X_i (i = 1, 2, ..., N)
while (stopping condition is not met) do
    Calculate the fitness values of hawks
    Set X_rabbit as the location of rabbit (best location)
    for (each hawk (X_i)) do
        Update the initial energy E_0 and jump strength J
        E_0 = 2rand() − 1, J = 2(1 − rand())
        Update E using Eq. (9)
        if (|E| ≥ 1) then
            Update the location vector using Eq. (7)
            if (|E| < 1) then
                if (r ≥ 0.5 and |E| ≥ 0.5) then
                    Update the location vector using Eq. (10)
                else if (r ≥ 0.5 and |E| < 0.5) then
                    Update the location vector using Eq. (12)
                else if (r < 0.5 and |E| ≥ 0.5) then
                    Update the location vector using Eq. (13)
                else if (r < 0.5 and |E| < 0.5) then
                    Update the location vector using Eq. (17)
Return X_rabbit
```

2.3 Particle swarm optimization

Particle swarm optimization (PSO) is a population-based optimization algorithm based on swarm intelligence [51]. It was inspired by the social and physical behavior of animals such as ants, birds, bees and fish that move in flocks in nature. PSO and its variants have been successfully applied to a variety of real-world applications due to its efficacy in solving challenging optimization issues and ease of implementation with fast convergence to a satisfactory solution [52–54].

PSO is an intelligent system in which each individual in the swarm is attributed to a particle. The velocity and position information are kept for each particle. Each particle is initially placed in problem space at a random location. These particles adjust their velocity according to the best position (*pbest*) found in each iteration, and when the iteration ends, the global best position (*gbest*) value is found. Particles move according to the equations below during the search process.

$$v_i = wv_x + r_1c_1(pbest_i - x_i) + r_2c_2 * (gbest_x - x_i) \quad (20)$$

$$x_i = x_i + v_i \quad (21)$$

where w denotes inertia weight included in the adaptive PSO (APSO) [54], x_i is the current position of the particle, v_i is current velocity of the particle and r_1, r_2 are random numbers. In the process of velocity update, the values of parameters such as c_1 and c_2 must be determined in advance. The optimization process is repeated until the stop criterion is reached. In Algorithm 2, the pseudocode of the PSO is presented.

Algorithm 2 The pseudo-code of the PSO Algorithm

```
Initialize a population of particles with random positions X and velocities V
while (stopping condition is not met) do
    for (each particle (X_i)) do
        Calculate the fitness value of particle
        if (fitness < p_best ) then
            p_best = fitness
        if (fitness < g_best) then
            g_best = fitness
    for (each particle (X_i)) do
        Update particle velocity using Eq. (20)
        Update particle position using Eq. (21)
```

3 Proposed approach

As mentioned before, the purpose of this method is to use ELM in wheat classification and to maximize the generalization capability of ELM. In the first stage, the method of selecting features was applied by editing the dataset with pre-processing techniques on the high-dimensional dataset. In the second stage, classification was performed on the selected features by the original ELM algorithm. In the third stage, ELM's parameters are optimized with the metaheuristic HHO and PSO algorithms to obtain an optimal model.

Initialization of population, fitness function and other optimization parameters determination of the proposed method in this study are given below.

3.1 Design issues

The individual vector in the population to be used in optimization consists of a combination of a number of input weights (w) and hidden biases (b). This is illustrated in Fig. 2. In the given problem, HHO and PSO find the best input weights and biases to increase the efficiency of the ELM classifier on the validation set. The best solution is searched in the $(n \times H + H)$ dimensional search space. H is hidden neuron number and w_i and b_i values in the population are randomly initialized in the range $(-1, 1)$ [32, 36, 55, 56]. The corresponding output weights for each individual obtained are calculated. Then, the fitness of each individual is evaluated. In this study, the evaluation of fitness is calculated by the mean square error (MSE) function given in Eq. (22).

$$MSE = \frac{1}{n}\sum_{k=1}^{n}(a_k - p_k)^2 \quad (22)$$

where a_k is the actual values, p_k is the predicted values and n is the size of the validation set. Generally, the fitness function is evaluated on the whole training set. However, this may cause the network to overfit. In addition, since output weight is already the minimum norm least square solution of the training dataset, according to the different individual, the training error should be very similar. Therefore, to save time, only the MSE value of the validation set was used as fitness function instead of the whole training set used.

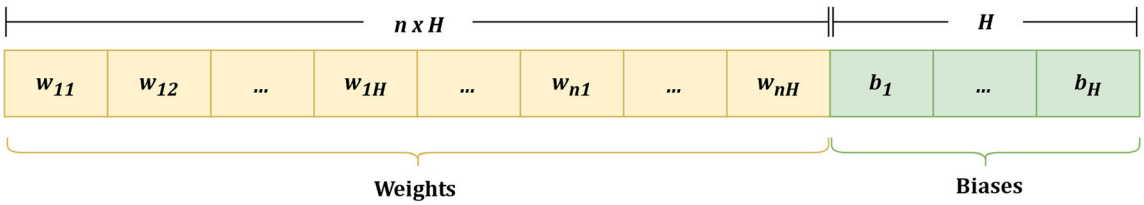

Fig. 2 Representation of population individual structure used in proposed models

3.2 Proposed ELM Models

Since ELM randomly selects input weights and hidden bias values, the model saves time. However, since output weights are calculated according to predetermined input weights and hidden biases, there may be several non-optimal or unnecessary input weights and hidden biases [32, 33]. ELM may require more hidden neurons in some applications than traditional tuning-based learning algorithms and this may cause ELM to respond slowly to unknown test data. In this study, detailed representation of the proposed PSO-ELM and HHO-ELM models as a solution to these problems is given in Fig. 3.

After calculating the fitness of population, the optimization process given below is repeated until the goal is met or the maximum iteration is reached.

With the HHO-ELM model presented in this study, the most appropriate input weights and hidden bias values were selected to improve ELM's generalization performance. The optimization process can be summarized in the following steps.

1. Set the population size and maximum number of iterations and the number of neurons in the hidden layer.
2. A random population is generated in the range [− 1.1], where the position of Harris hawks consists of the hidden bias value and input weights.
3. The output weights corresponding to each hawk consisting of input weights and hidden bias values are calculated with generalized MP inverse in Eq. (6) without recursion with MSE. The best location is set to the location of the rabbit. The fitness value of the position of each hawk is evaluated according to Eq. (22).
4. Hawks' positions are updated with two strategies in exploration phase. These two strategies given in Eq. (7) are updated by perching compared to other family members ($q<0.5$) and update ($q\geq 0.5$) by perching on random tall trees within the nest range (the position of the rabbit) of the hawks that will attack.
5. The position of the hawks is updated with 4 different strategies given in the exploitation phase that comes after the exploration phase. These are soft besiege methods given in Eq. (10), hard besiege given in Eq. (12) and soft and hard besiege methods with progressive rapid dives given in Eq. (13–19).
6. Except the first step, all other steps are repeated until the maximum number of iterations has been reached and the optimum output weights from optimization operations are applied to the test dataset to measure the generalization performance of the model.

In the PSO-ELM model, the initial population consisting of weight and biases is generated. MSE was chosen as the fitness value for fair comparison with the HHO model. Then velocity and position vectors are updated through iterations [57]. The best population obtained from the PSO-ELM model is updated as the parameters of the ELM.

3.3 Evaluation measures

The performance of all classifier models developed in this study is evaluated by accuracy, sensitivity, specificity and F-score. The equations for the abovementioned metrics are given below at Eqs. (23)–(26) [19, 58].

Accuracy is frequently used in classification studies. It denotes the ratio between true predictions and total samples.

$$ACC = \frac{T_P + T_N}{(T_P + T_N + F_P + F_N)} * 100 \tag{23}$$

Sensitivity also known as recall and denotes the percentage of true positives that are correctly identified.

$$SEN = \frac{T_P}{T_P + F_N} * 100 \tag{24}$$

Specificity denotes the percentage of true negatives that are correctly identified.

$$SPE = \frac{T_N}{T_N + F_P} * 100 \tag{25}$$

F-score indicates whether the model is valid with datasets that are not equally distributed.

$$F - Score = \frac{2 * T_P}{2 * T_P + F_P + F_N} * 100 \tag{26}$$

where T_P denotes the true positives, F_N denotes the false negatives, T_N denotes the true negatives and F_P denotes the

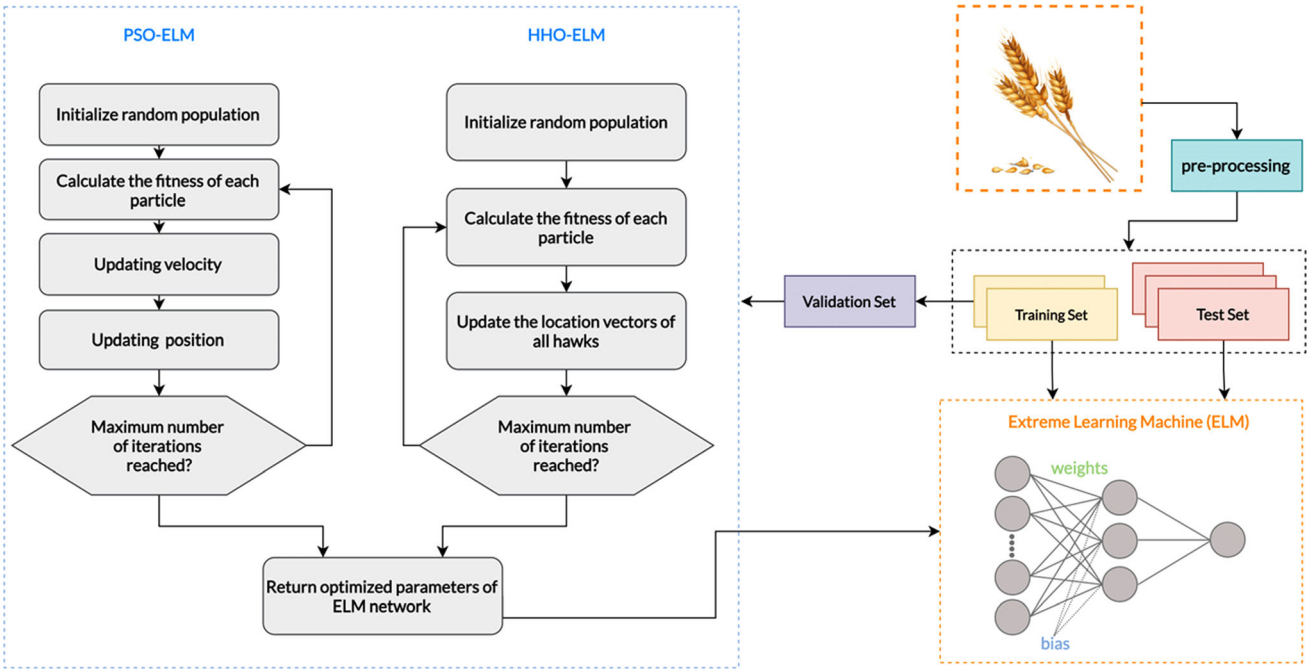

Fig. 3 Schematic representation of the proposed PSO-ELM and HHO-ELM models

false positives. The area under the ROC curve [59], also known as the AUC, is one of the most commonly used metrics for evaluating binary classifiers. The bigger the value of the AUC, the better the performance of the classifier [60].

4 Experiments and analysis

4.1 Experimental setup

The following models were developed to test the performance of the proposed model, and comparisons were made separately with and without feature selection: original HHO-based ELM (HHO-ELM), PSO-based ELM (PSO-ELM). All experiments are conducted with Python 3.8 (64-bit) programming language on Ubuntu 20.04 LTS platform running on a computer with Intel i7-6700HQ 2.60 GHz CPU and 16 GB RAM. The ELM classifier's behavior varies depending on the number of neurons in the hidden layer. For this reason, the datasets to be used in the study were run with the number of neurons in the hidden layer from 10 to 100 with an increase of 10.

For a fair comparison, implementations of optimized models were made under the same conditions. To ensure that the results are not misleading, 30 trials were performed in all models and average results were obtained. In each trial, the particles were randomly initialized and randomly distributed in the train, test and validation sets. Because the population size in the proposed algorithms is significantly dependent on the complexity of the given real-world implementation, it is left as user-defined parameters. The number of particles was set to 20, and the maximum number of iterations was set to 100 in both algorithms. The algorithm-specific parameters of PSO, both the cognition learning factor c_1 and the social learning factor c_2, were set to 2. In addition, the inertia weight w was set to between 0.2 and 0.9.

4.2 Data description and pre-processing

In this study, an open access dataset containing vitreous and starchy type wheat kernels and contaminants contained in the kernels was used [25, 61]. The durum wheat dataset is created from real-time moving images under controlled light with a CMOS-type camera [25]. Figure 4. contains images of wheat types and foreign matters.

The obtained images are converted to binary mode and segmented with image processing. Morphological and color features were obtained for each segmented object. Finally, the extracted features are given in Table 1.

A total of 236 features were obtained with the specified morphological and color characteristics, statistical functions, wavelet and gaborlet functions [25].

In the experimental study, all attributes were normalized to the range $[-1, 1]$ with the min–max normalization process given in Eq. (27). There are 9000 samples in the dataset. In each trial, training set and test set are divided into 80% and 20%, respectively. For all algorithms, 50% of the training set is used as a validation set.

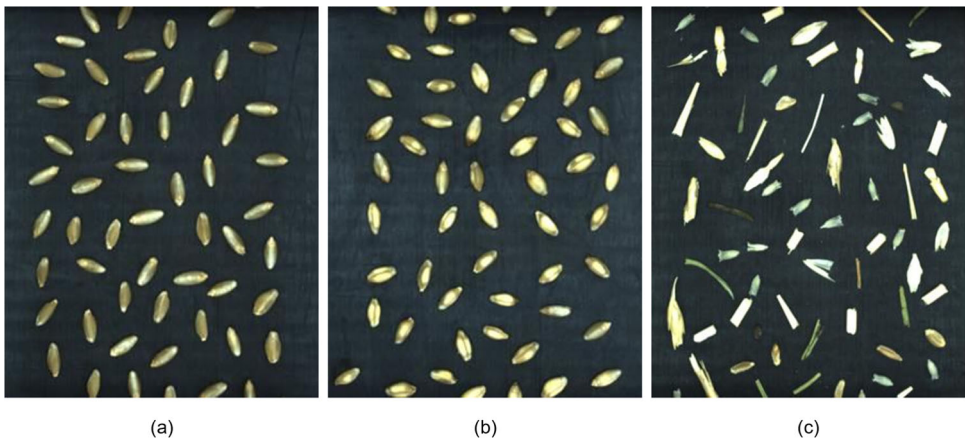

Fig. 4 Frame samples from videos **a** vitreous kernels, **b** starchy kernels and **c** foreign matters [25]

Table 1 Morphological and color features with definitions [25]

Type	Features	Definition
Morphological	Area	Number of pixels in a region
	Prime axis	The length of the prime axis of the ellipse (in pixels)
	Secondary axis	The length of the secondary axis of the ellipse (in pixels)
	Perimeter	The length of the boundary perimeter of the region (in pixels)
	Equivalent diameter	Diameter of the circle with the same area as the region (in pixels)
	Eccentricity	Ratio of the distance between the focal points of the ellipse to the prime axis length
	Roundness	4.Area/π (Prime Axis)^2
	Shape factor	4.π. Area/Perimeter^2
	Compactness	Sqrt (4. Area/π)/ Prime Axis
	Extent	Ratio of pixels in the region to the number of pixels in the bounding box
Color	R, G, B	Red, Green, Blue
	Y, Cb, Cr	Brightness, Blue-difference chroma, Red-difference chroma
	H, S, V	Hue, Saturation, Value
	X, Y, Z	Red, Luminance, Blue
	L, a*, b*	Lghtness, Green–Red, Blue–Yellow

$$x' = \left(\frac{x - min_a}{max_a - min_a}\right) * 2 - 1 \tag{27}$$

where x is the original value, x' is the scaled value, max_a is the maximum value and min_a is the minimum value of feature a.

4.3 ANOVA f-test for feature selection

The ANOVA (variance analysis) f-test statistic is used widely when dealing with problems involving a large number of inputs. The ANOVA f-test ratio does not indicate how much each class is separated from each other, but how much separation there is between the classes [62]. The process of selecting features using the ANOVA method was performed to optimize inputs in the wheat

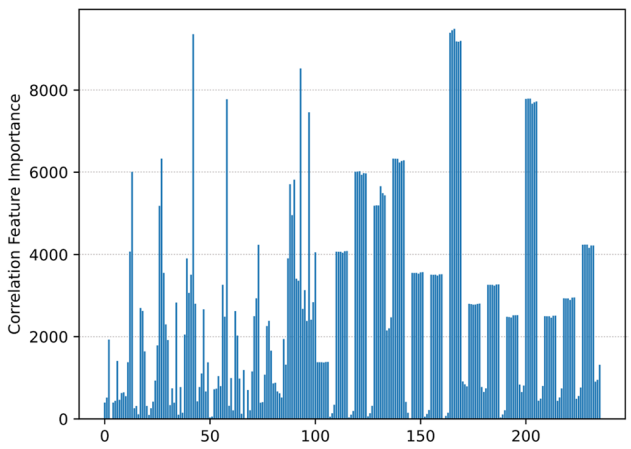

Fig. 5 Score values of input feature in wheat dataset

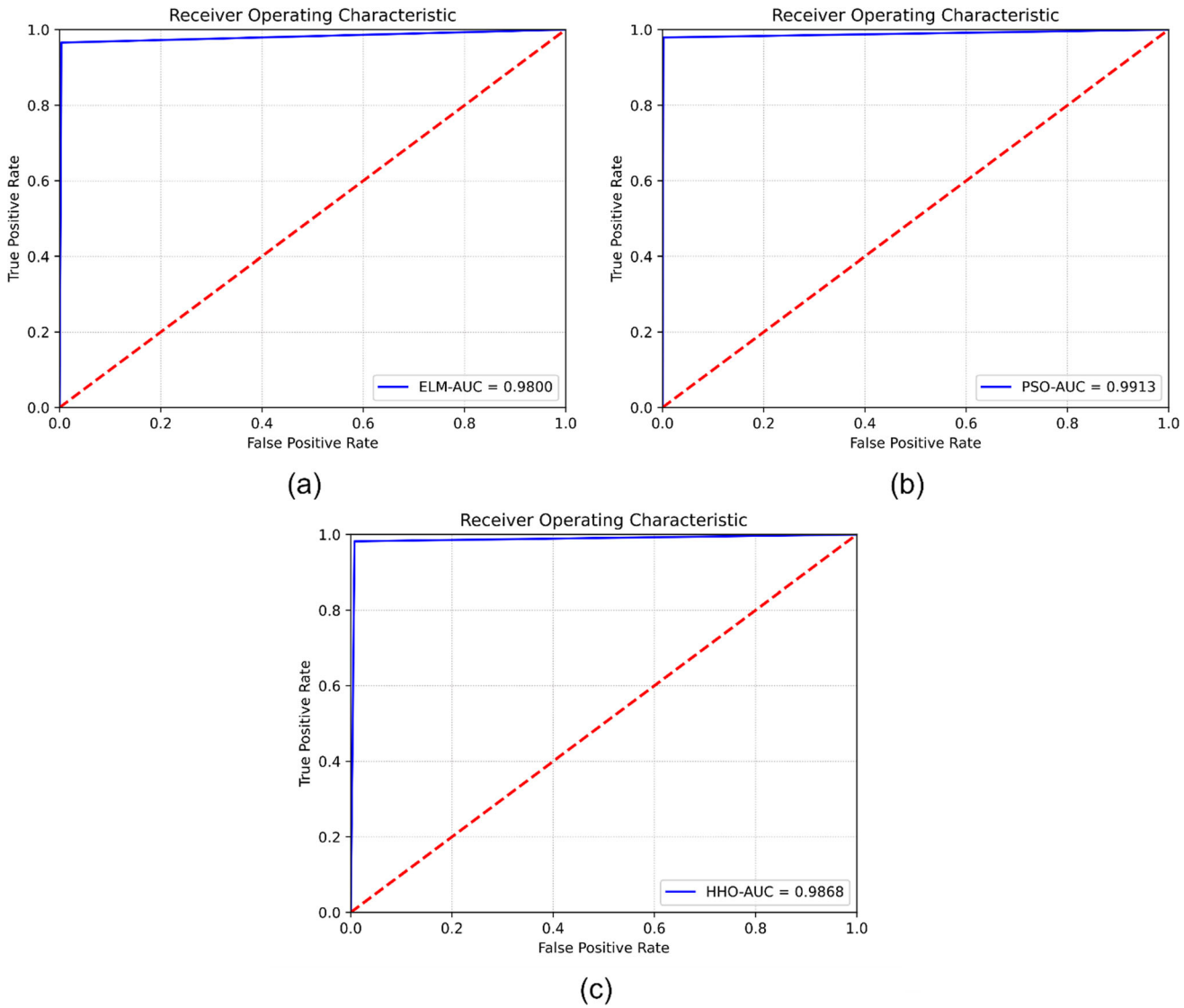

Fig. 6 ROC curves for **a** ELM, **b** PSO-ELM and **c** HHO-ELM for binary classification

Table 2 The confusion matrix of models by performing mean of trials for binary classification

Model	Class	Wheat	Non-wheat
HHO-ELM	Wheat	1200	4
	Non-wheat	14	582
PSO-ELM	Wheat	1197	7
	Non-wheat	7	589
ELM	Wheat	1200	4
	Non-wheat	22	574

classification process since it allows for comparison with studies in the literature. For each input feature, a bar graph of feature importance points has been generated. Importance scores for the 236 features in the dataset used in the study are given in Fig. 5.

In this study, determination wheat types and separation of foreign matters were experimented in as multi-class classification and detection of foreign matters as binary classification problem.

4.4 Results obtained binary classification and comparison other models

This experiment has been considered as a binary classification problem developed for the separation of wheat from foreign matters. HHO-ELM, PSO-ELM and ELM models are used in this problem solution. The ROC curve and AUC value of the developed models are given in Fig. 6.

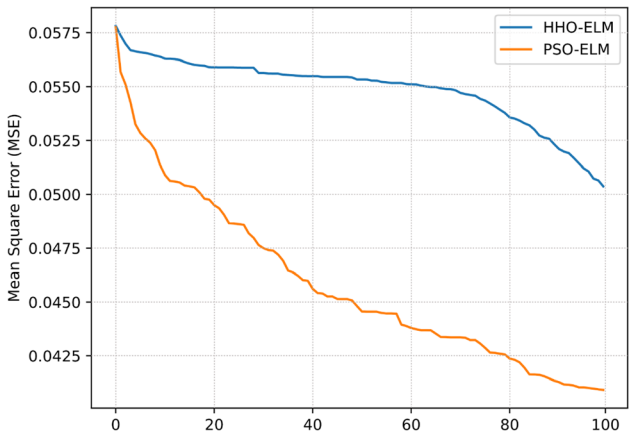

Fig. 7 Convergence curves for HHO-ELM and PSO-ELM for binary classification

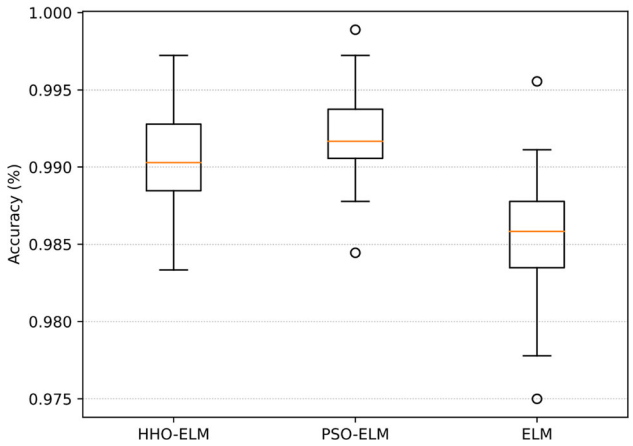

Fig. 8 Box plots for HHO-ELM, PSO-ELM and ELM for binary classification

Table 2 shows the classification confusion matrix of all models for this experiment. As we can see, optimization models used in the study have given a better result than original ELM.

Figure 7 shows the convergence graphs for the proposed HHO-ELM and PSO-ELM models. According to these graphs, it can be noticed that PSO has a faster convergence than HHO. We can say that HHO's convergence rate has increased from approximately 70.

While the models were being developed, 30 runs were performed for each model. The accuracy ranges of the created models are given in Fig. 8. Accordingly, it is seen that the PSO-ELM is more compact than other models.

Table 3 provides the average, best and worst accuracy values and standard deviation values of all models used in this experiment depending on the number of neurons in the hidden layer. It is observed that the accuracy values increase as the number of neurons in the hidden layer of all models increases. According to Table 3, it is seen that the standard deviation values of optimization models are generally less than the original ELM classifier.

Table 3 The mean accuracy (%), std, best and worst accuracy values of models used in this study for binary classification

Hidden nodes	Methods	Mean	Std	Best	Worst
10	HHO-ELM	97.0215	0.6003	98.3889	95.7778
	PSO-ELM	97.4609	0.3848	98.2778	96.7222
	ELM	94.8730	1.6077	97.3889	91.6111
20	HHO-ELM	97.7051	0.5121	99.0000	96.6667
	PSO-ELM	97.8027	0.4973	99.0000	96.7222
	ELM	96.6309	0.8435	97.8889	94.5000
30	HHO-ELM	97.8516	0.4678	98.7778	97.1111
	PSO-ELM	98.1445	0.4983	99.1667	97.1667
	ELM	97.1680	0.6703	98.3333	96.0000
40	HHO-ELM	98.0957	0.3800	98.6667	97.2778
	PSO-ELM	98.3887	0.4803	99.2222	97.0556
	ELM	97.6074	0.4819	98.4444	96.6111
50	HHO-ELM	98.3887	0.2930	98.9444	97.5556
	PSO-ELM	98.5840	0.4669	99.5000	97.7222
	ELM	97.8027	0.4062	98.6667	96.8333
60	HHO-ELM	98.4863	0.4062	99.3889	97.5000
	PSO-ELM	98.7305	0.4098	99.5556	97.8333
	ELM	97.9004	0.4055	98.8333	97.2778
70	HHO-ELM	98.7305	0.3096	99.1667	98.0000
	PSO-ELM	98.9258	0.2385	99.3889	98.4444
	ELM	98.1445	0.4203	98.7778	97.2778
80	HHO-ELM	98.7305	0.3165	99.3333	98.0556
	PSO-ELM	99.0723	0.3207	99.7222	98.4444
	ELM	98.3398	0.3609	99.0000	97.2778
90	HHO-ELM	98.8770	0.3692	99.5000	98.0000
	PSO-ELM	99.1699	0.3797	99.7778	98.0556
	ELM	98.4863	0.4003	99.2222	97.7222
100	HHO-ELM	**99.0234**	0.3332	99.7222	98.3333
	PSO-ELM	**99.2188**	0.3106	99.8889	98.4444
	ELM	**98.5352**	0.3974	99.5556	97.5000

The best results are given in bold

As mentioned earlier, experiments were carried out in this study to observe the effects on the performance of models in different feature numbers. Model inputs were created according to the best rank values obtained as a result of the feature selection process. For all developed models, performance values were observed in different feature numbers and in the number of 100 neurons that work best in the hidden layer. Sensitivity, specificity, f-score and accuracy performance metrics of these models are given for comparison on Table 4.

Table 4 Sensitivity, specificity, f-score and accuracy performance metrics for HHO-ELM, PSO-ELM and ELM for binary classification

Feature #	Methods	Sensitivity	Specificity	F-score	Accuracy (%)
30	HHO-ELM	**0.9576**	0.9979	**0.9774**	**98.4375**
	PSO-ELM	0.9550	0.9980	0.9760	98.3887
	ELM	0.9518	**0.9983**	0.9745	98.2910
180	HHO-ELM	0.9782	0.9971	0.9875	99.0723
	PSO-ELM	**0.9897**	0.9950	**0.9923**	**99.3164**
	ELM	0.9624	**0.9975**	0.9796	98.5840
236	HHO-ELM	0.9770	**0.9966**	0.9867	99.0234
	PSO-ELM	**0.9889**	0.9937	**0.9913**	**99.2188**
	ELM	0.9635	**0.9966**	0.9797	98.5352

The best results are given in bold

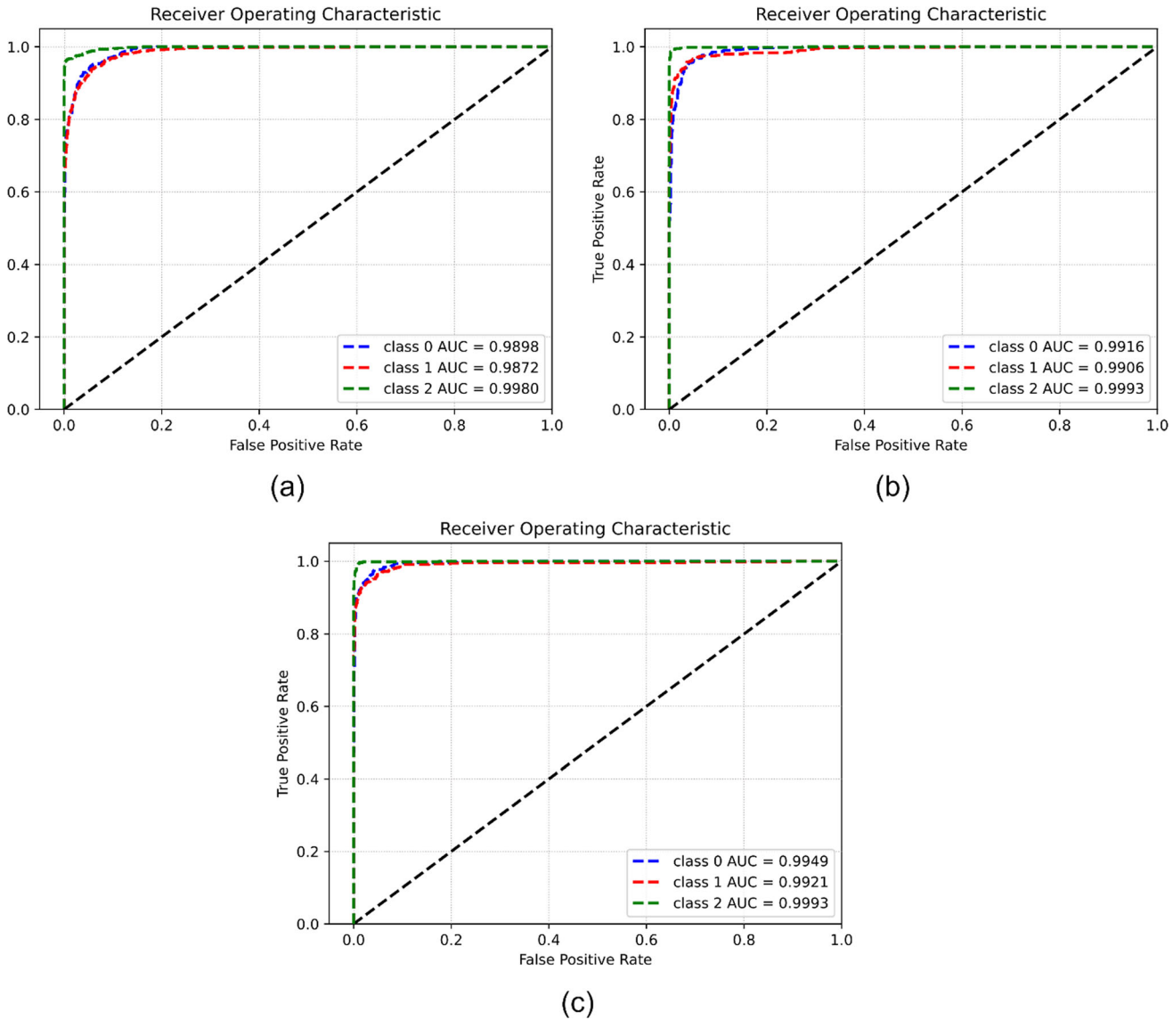

Fig. 9 ROC curves for **a** ELM, **b** PSO-ELM and **c** HHO-ELM for multi-class classification

Table 5 The confusion matrix of models by performing mean of trials

Model	Class	Vitreous	Starchy	Foreign
HHO-ELM	Vitreous	576	21	2
	Starchy	34	561	1
	Foreign	10	5	590
PSO-ELM	Vitreous	566	31	2
	Starchy	42	553	1
	Foreign	9	6	590
ELM	Vitreous	569	28	2
	Starchy	48	547	1
	Foreign	12	9	584

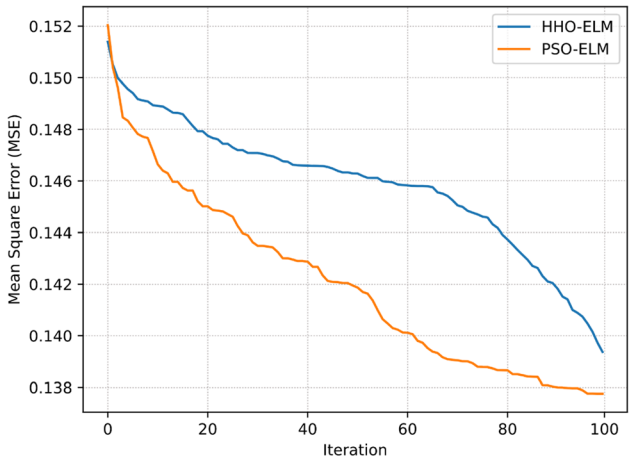

Fig. 10 Convergence curves for HHO-ELM and PSO-ELM for multi-class classification

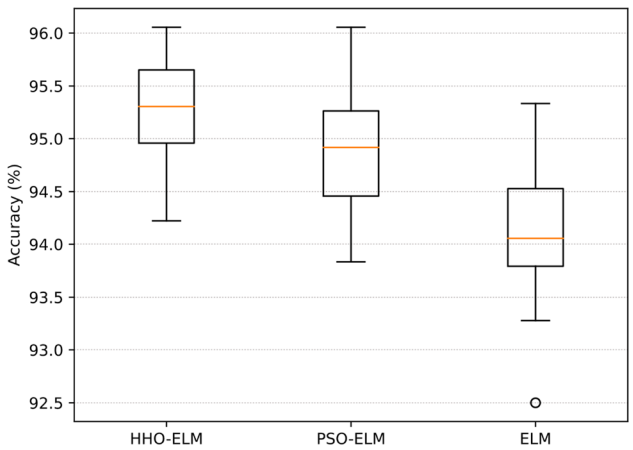

Fig. 11 Box plots for HHO-ELM, PSO-ELM and ELM for multi-class classification

Table 6 The mean accuracy (%), std, best and worst accuracy values of models used for multi-class classification

Hidden nodes	Methods	Mean	Std	Best	Worst
10	HHO-ELM	89.1602	1.7675	92.7222	84.7778
	PSO-ELM	89.1602	1.2009	91.6111	86.2222
	ELM	81.0059	3.1656	88.5000	73.4444
20	HHO-ELM	91.6016	1.3377	94.3889	88.5556
	PSO-ELM	91.3086	1.0202	93.8333	89.6111
	ELM	86.5723	2.2700	90.3333	80.2778
30	HHO-ELM	92.7246	0.6823	94.0556	91.5556
	PSO-ELM	91.8457	0.9538	93.8889	90.2778
	ELM	89.0625	1.6599	91.7778	85.0556
40	HHO-ELM	93.5059	0.6772	95.0000	92.3333
	PSO-ELM	92.8223	0.9547	94.6111	90.6667
	ELM	90.9180	1.1768	93.6111	88.7778
50	HHO-ELM	94.0918	0.6726	95.9444	93.0000
	PSO-ELM	93.6035	0.9256	95.8333	91.9444
	ELM	92.1387	0.8037	93.7778	90.6111
60	HHO-ELM	94.6289	0.6481	96.2222	93.0000
	PSO-ELM	93.9453	0.8692	95.5000	91.4444
	ELM	92.8711	0.9327	94.6667	90.8889
70	HHO-ELM	94.8730	0.8526	96.4444	93.1111
	PSO-ELM	94.4336	0.9421	96.3889	92.5556
	ELM	93.3105	0.9557	95.6111	91.1111
80	HHO-ELM	95.2148	0.7079	96.7222	93.7778
	PSO-ELM	94.7754	0.7060	96.2222	93.0556
	ELM	93.6035	0.8062	95.3889	91.9444
90	HHO-ELM	**95.9473**	0.5590	96.8889	94.6667
	PSO-ELM	94.9707	0.8057	96.6667	93.5000
	ELM	94.4336	0.7536	95.7778	92.3333
100	HHO-ELM	95.8984	0.6654	97.0000	94.3333
	PSO-ELM	**95.5566**	0.6473	96.7778	94.0556
	ELM	**94.5801**	0.6766	96.1111	93.6111

The best results are given in bold

4.5 Results obtained multi-class classification and comparison other models

This experiment is considered to test the overall performance of models in the multi-class classification of the wheat dataset (vitreous, starchy, non-wheat). The ROC curve and AUC values for the proposed models are given in Fig. 9. In fact, the AUC is a binary classification metric, but in this study, ROC curves were drawn separately for each class with the one vs all technique and the AUC value was calculated.

Table 5 shows the classification confusion matrix of all models for this experiment. As we can see, all models used in the study can distinguish the foreign (non-wheat) class

Table 7 Overall accuracy, sensitivity and specificity values for multi-class classification

Features	Methods	Hidden neuron number	Vitreous		Starchy		Non-wheat		Overall accuracy (%)
			Sen.	Spe.	Sen.	Spe.	Sen.	Spe.	
30	HHO-ELM	90	0.9263	**0.9326**	**0.8735**	**0.9507**	0.9595	0.9970	**92.04**
	PSO-ELM	90	0.9200	0.9316	0.8716	0.9478	**0.9604**	0.9970	91.75
	ELM	90	**0.9277**	0.9297	0.8680	0.9490	0.9556	0.9976	91.70
	ANN (Kaya and Saritas, [25])	100	0.8862	0.8543	0.8006	0.9256	0.9220	0.9795	86.43
210	HHO-ELM	100	**0.9595**	**0.9590**	**0.9307**	**0.9775**	**0.9775**	**0.9976**	**95.56**
	PSO-ELM	100	0.9497	**0.9590**	0.9287	0.9717	0.9766	0.9966	95.17
	ELM	100	0.9546	0.9517	0.9175	0.9740	0.9736	**0.9976**	94.83
	ANN (Kaya and Saritas, [25])	100	0.9531	0.9372	0.8936	0.9732	0.9554	0.9788	93.46
236	HHO-ELM	90	**0.9620**	**0.9634**	**0.9404**	**0.9780**	0.9750	**0.9976**	**95.95**
	PSO-ELM	100	0.9556	0.9620	0.9340	0.9746	**0.9770**	0.9970	95.56
	ELM	100	0.9540	0.9480	0.9146	0.9736	0.9690	0.9970	94.58
	ANN (Kaya and Saritas, [25])	100	0.9565	0.8973	0.8506	0.9709	0.9331	0.9848	91.95

The best results are given in bold

better than other classes and incorrectly predicted classes (misclassification number) are less in the proposed model than in other models.

Figure 10 provides convergence graphs of the proposed HHO-ELM and compared PSO-ELM models. According to these graphs, it appears that HHO has shown a descent with a sudden change according to PSO. These sudden changes mean that HHO performs better at exploring the search area. We can also observe that 100 iterations are sufficient for the convergence of algorithms.

While the models were being developed, 30 runs were performed for each model. The accuracy ranges of the resulting models are given in Fig. 11. Accordingly, it is seen that the HHO-ELM is more compact than other models.

Table 6 provides the average, best and worst accuracy values and standard deviation values of all models used in this experiment depending on the number of neurons in the hidden layer. It is observed that the accuracy values increase as the number of neurons in the hidden layer of all models increases. Additionally, Table 6 shows that optimization models has lower standard deviations than the original ELM classifier.

The developed models achieved the best performance in the hidden layer in the number of 90 neurons for HHO-ELM and 100 neurons for PSO-ELM and ELM models. The sensitivity, specificity, accuracy performance metrics of these models, as well as the results of the ANN method study in the literature for the same problem are given for comparison purposes on Table 7.

5 Conclusion

In recent years, the process of separating wheat seeds with different qualities and economic values according to visual characteristics has been gaining importance. In this context, ELM-based models with fast learning capability that are easy to use in embedded systems are presented. It is seen that the original ELM model in this study performs well compared to the similar studies in the literature. In addition, HHO, one of the swarm-based metaheuristic optimization algorithms, has been used to optimize the performance of ELM in the problems of wheat type determination and distinguish from foreign matters. In other words, modified ELM model has been used to tune the weight parameter instead of the traditional random selection. Also, the well-known algorithm PSO was used for comparison purposes. To demonstrate the performance of the proposed method, developed models applied in the binary and multi-class classifications of wheat. At the same time, these models were compared with the methods used in other studies in the literature. Compared with previous study [25], the proposed models had a significant effect on classification performance.

Through experiments on binary and multi-class classification durum wheat dataset, the proposed HHO-ELM algorithm in this paper achieves 99.21% and 95.95% classification accuracy, respectively. Compared with the traditional neural network classifier, HHO-ELM provides about 2%—6% improvement in classification accuracy of all experiments performed. In both binary and multi-class classification experiments, HHO-ELM gives better classification accuracy than original ELM and optimized ELM.

The experiments were carried out using different number of features. In these experiments, it is proved that the proposed model has strong performance in all the wheat classification accuracy, sensitivity and specificity performance metrics.

The most important issue in the models developed is that the appropriate number of neurons in the hidden layer was determined by the trial-and-error method and this process takes time. In addition, determining the lower and upper limits of the search space is a challenging task, as it will affect the convergence of models. The number of parameters that must be set in the HHO algorithm is not a major challenge. However, the randomness of HHO's transition between the phases of exploration and exploitation in the optimization process can lead to slow convergence and fall into local optima. Furthermore, using the optimum weight values obtained by the optimization process and applying them to ELM, much more efficient results can be achieved in future studies carried out as an embedded system in this area. It is seen that pre-optimizing the original ELM algorithm is of great benefit in solving similar problems.

As future work, studies can be done in order to increase HHO's searching ability and to prevent local optima stagnation. Also, it is planned to test the multi-objective optimization and performance of ELM with other types of interest (such as WELM, KELM and regularized ELM) in the literature developed for classification and regression.

Author contributions MD contributed to conceptualization, methodology, software, validation, formal analysis, visualization, writing—original draft, writing review and editing. IAO contributed to methodology, formal analysis, investigation, writing—original draft, writing—review and editing, supervision, project administration.

Data availability The experimental dataset are open source and available via references.

Declarations

Conflict of interest The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

1. Igrejas G, Branlard G (2020) The Importance of Wheat. In: Igrejas G, Ikeda TM, Guzmán C (eds) Wheat quality for improving processing and human health. Springer International Publishing, Cham, pp 1–7
2. Huebner FR, Bietz JA, Nelsen T et al (1999) Soft wheat quality as related to protein composition. Cereal Chem 76:650–655. https://doi.org/10.1094/CCHEM.1999.76.5.650
3. Sabanci K, Kayabasi A, Toktas A (2017) Computer vision-based method for classification of wheat grains using artificial neural network. J Sci Food Agric 97:2588–2593. https://doi.org/10.1002/jsfa.8080
4. Bao Y, Mi C, Wu N et al (2019) Rapid classification of wheat grain varieties using hyperspectral imaging and chemometrics. Appl Sci 9:4119. https://doi.org/10.3390/app9194119
5. Tian H, Wang T, Liu Y et al (2020) Computer vision technology in agricultural automation —a review. Inf Process Agric 7:1–19. https://doi.org/10.1016/j.inpa.2019.09.006
6. Mollazade K, Omid M, Arefi A (2012) Comparing data mining classifiers for grading raisins based on visual features. Comput Electron Agric 84:124–131. https://doi.org/10.1016/j.compag.2012.03.004
7. Patrício DI, Rieder R (2018) Computer vision and artificial intelligence in precision agriculture for grain crops: a systematic review. Comput Electron Agric 153:69–81. https://doi.org/10.1016/j.compag.2018.08.001
8. Pourreza A, Pourreza H, Abbaspour-Fard M-H, Sadrnia H (2012) Identification of nine Iranian wheat seed varieties by textural analysis with image processing. Comput Electron Agric 83:102–108. https://doi.org/10.1016/j.compag.2012.02.005
9. Gunes EO, Aygun S, Kirci M, et al (2014) Determination of the varieties and characteristics of wheat seeds grown in Turkey using image processing techniques. In: 2014 The Third International Conference on Agro-Geoinformatics. IEEE, pp 1–4
10. Guevara-Hernandez F, Gomez-Gil J (2011) Sistema de visión artificial para la clasificación de granos de trigo y cebada. Spanish J Agric Res 9:672–680. https://doi.org/10.5424/sjar/20110903-140-10
11. Yasar A, Kaya E, Saritas I (2016) Classification of wheat types by artificial neural network. Int J Intell Syst Appl Eng 4:12–15. https://doi.org/10.18201/ijisae.64198
12. Aslan MF, Sabancı K, Durdu A (2017) Different wheat species classifier application of ANN and ELM. J Multidiscip Eng Sci Technol 4:2458–9403
13. Ebrahimi E, Mollazade K, Babaei S (2014) Toward an automatic wheat purity measuring device: a machine vision-based neural networks-assisted imperialist competitive algorithm approach. Measurement 55:196–205. https://doi.org/10.1016/j.measurement.2014.05.003
14. Olgun M, Onarcan AO, Özkan K et al (2016) Wheat grain classification by using dense SIFT features with SVM classifier. Comput Electron Agric 122:185–190. https://doi.org/10.1016/j.compag.2016.01.033
15. Shrestha BL, Kang Y-M, Yu D, Baik O-D (2016) A two-camera machine vision approach to separating and identifying laboratory sprouted wheat kernels. Biosyst Eng 147:265–273. https://doi.org/10.1016/j.biosystemseng.2016.04.008
16. Singh P, Nayyar A, Singh S, Kaur A (2020) Classification of wheat seeds using image processing and fuzzy clustered random forest. Int J Agric Resour Gov Ecol 16:123–156. https://doi.org/10.1504/IJARGE.2020.109048
17. JayaBrindha G, Subbu ESG (2018) Ant colony technique for optimizing the order of cascaded SVM classifier for sunflower seed classification. IEEE Trans Emerg Top Comput Intell 2:78–88. https://doi.org/10.1109/TETCI.2017.2772918
18. Kayabasi A (2018) An application of ANN trained by ABC algorithm for classification of wheat grains. Int J Intell Syst Appl Eng 6:85–91. https://doi.org/10.18201/ijisae.2018637936
19. Koklu M, Ozkan IA (2020) Multiclass classification of dry beans using computer vision and machine learning techniques. Comput Electron Agric 174:105507. https://doi.org/10.1016/j.compag.2020.105507

20. Koklu M, Cinar I, Taspinar YS (2021) Classification of rice varieties with deep learning methods. Comput Electron Agric 187:106285. https://doi.org/10.1016/j.compag.2021.106285
21. Ding S, Xu X, Nie R (2014) Extreme learning machine and its applications. Neural Comput Appl 25:549–556. https://doi.org/10.1007/s00521-013-1522-8
22. Huang G, Huang G-B, Song S, You K (2015) Trends in extreme learning machines: a review. Neural Netw 61:32–48. https://doi.org/10.1016/j.neunet.2014.10.001
23. Murthy CS, Raju PV, Badrinath KVS (2003) Classification of wheat crop with multi-temporal images: performance of maximum likelihood and artificial neural networks. Int J Remote Sens 24:4871–4890. https://doi.org/10.1080/0143116031000070490
24. Mahesh S, Manickavasagan A, Jayas DS et al (2008) Feasibility of near-infrared hyperspectral imaging to differentiate Canadian wheat classes. Biosyst Eng 101:50–57. https://doi.org/10.1016/j.biosystemseng.2008.05.017
25. Kaya E, Saritas I (2019) Towards a real-time sorting system: Identification of vitreous durum wheat kernels using ANN based on their morphological, colour, wavelet and gaborlet features. Comput Electron Agric 166:105016. https://doi.org/10.1016/j.compag.2019.105016
26. Prasad R, Deo RC, Li Y, Maraseni T (2018) Soil moisture forecasting by a hybrid machine learning technique: ELM integrated with ensemble empirical mode decomposition. Geoderma 330:136–161. https://doi.org/10.1016/j.geoderma.2018.05.035
27. Kouadio L, Deo RC, Byrareddy V et al (2018) Artificial intelligence approach for the prediction of Robusta coffee yield using soil fertility properties. Comput Electron Agric 155:324–338. https://doi.org/10.1016/j.compag.2018.10.014
28. Sulistyo SB, Wu D, Woo WL et al (2018) Computational deep intelligence vision sensing for nutrient content estimation in agricultural automation. IEEE Trans Autom Sci Eng 15:1243–1257. https://doi.org/10.1109/TASE.2017.2770170
29. Suchithra MS, Pai ML (2020) Improving the prediction accuracy of soil nutrient classification by optimizing extreme learning machine parameters. Inf Process Agric 7:72–82. https://doi.org/10.1016/j.inpa.2019.05.003
30. Feng Z, Huang G, Chi D (2020) Classification of the complex agricultural planting structure with a semi-supervised extreme learning machine framework. Remote Sens 12:1–18. https://doi.org/10.3390/rs12223708
31. Mostafaeipour A, Fakhrzad MB, Gharaat S et al (2020) Machine learning for prediction of energy in wheat production. Agric 10:1–18. https://doi.org/10.3390/agriculture10110517
32. Zhu Q-Y, Qin AK, Suganthan PN, Huang G-B (2005) Evolutionary extreme learning machine. Pattern Recognit 38:1759–1763. https://doi.org/10.1016/j.patcog.2005.03.028
33. Cao J, Lin Z, Huang G-B (2012) Self-adaptive evolutionary extreme learning machine. Neural Process Lett 36:285–305. https://doi.org/10.1007/s11063-012-9236-y
34. Dash R, Dash PK, Bisoi R (2014) A self adaptive differential harmony search based Optimized extreme learning machine for financial time series prediction. Swarm Evol Comput. https://doi.org/10.1016/j.swevo.2014.07.003
35. Yang H, Yi J, Zhao J, Dong Z (2013) Extreme learning machine based genetic algorithm and its application in power system economic dispatch. Neurocomput 102:154–162. https://doi.org/10.1016/j.neucom.2011.12.054
36. Han F, Yao H-F, Ling Q-H (2013) An improved evolutionary extreme learning machine based on particle swarm optimization. Neurocomputing 116:87–93. https://doi.org/10.1016/j.neucom.2011.12.062
37. Li G, Niu P, Ma Y et al (2014) Tuning extreme learning machine by an improved artificial bee colony to model and optimize the boiler efficiency. Knowledge-Based Syst 67:278–289. https://doi.org/10.1016/j.knosys.2014.04.042
38. Wang M, Chen H, Li H et al (2017) Grey wolf optimization evolving kernel extreme learning machine: application to bankruptcy prediction. Eng Appl Artif Intell 63:54–68. https://doi.org/10.1016/j.engappai.2017.05.003
39. Shariati M, Mafipour MS, Ghahremani B et al (2022) A novel hybrid extreme learning machine–grey wolf optimizer (ELM-GWO) model to predict compressive strength of concrete with partial replacements for cement. Eng Comput 38:757–779. https://doi.org/10.1007/s00366-020-01081-0
40. Eshtay M, Faris H, Obeid N (2019) Metaheuristic-based extreme learning machines: a review of design formulations and applications. Int J Mach Learn Cybern 10:1543–1561. https://doi.org/10.1007/s13042-018-0833-6
41. Wang J, Lu S, Wang S-H, Zhang Y-D (2022) A review on extreme learning machine. Multimed Tools Appl 81:41611–41660. https://doi.org/10.1007/s11042-021-11007-7
42. Heidari AA, Mirjalili S, Faris H et al (2019) Harris hawks optimization: algorithm and applications. Futur Gener Comput Syst. https://doi.org/10.1016/j.future.2019.02.028
43. Moayedi H, Osouli A, Nguyen H, Rashid ASA (2021) A novel Harris hawks' optimization and k-fold cross-validation predicting slope stability. Eng Comput 37:369–379. https://doi.org/10.1007/s00366-019-00828-8
44. Houssein EH, Hosney ME, Elhoseny M et al (2020) Hybrid Harris hawks optimization with cuckoo search for drug design and discovery in chemoinformatics. Sci Rep 10:14439. https://doi.org/10.1038/s41598-020-71502-z
45. Wei Y, Lv H, Chen M et al (2020) Predicting entrepreneurial intention of students: an extreme learning machine with Gaussian Barebone Harris Hawks optimizer. IEEE Access 8:76841–76855. https://doi.org/10.1109/ACCESS.2020.2982796
46. Huang G-B, Zhu Q-Y, Siew C-K (2006) Extreme learning machine: theory and applications. Neurocomputing 70:489–501. https://doi.org/10.1016/j.neucom.2005.12.126
47. Schmidt WF, Kraaijveld MA, Duin RPW (1992) Feedforward neural networks with random weights. In: Proceedings, 11th IAPR International Conference on Pattern Recognition. Vol.II. Conference B: Pattern Recognition Methodology and Systems. IEEE Comput. Soc. Press, pp 1–4
48. Pao Y-H, Park G-H, Sobajic DJ (1994) Learning and generalization characteristics of the random vector functional-link net. Neurocomputing 6:163–180. https://doi.org/10.1016/0925-2312(94)90053-1
49. Essa FA, Abd Elaziz M, Elsheikh AH (2020) An enhanced productivity prediction model of active solar still using artificial neural network and Harris Hawks optimizer. Appl Therm Eng 170:115020. https://doi.org/10.1016/j.applthermaleng.2020.115020
50. Shehab M, Mashal I, Momani Z et al (2022) Harris Hawks optimization algorithm: variants and applications. Arch Comput Methods Eng 29:5579–5603. https://doi.org/10.1007/s11831-022-09780-1
51. Kennedy J, Eberhart R (1995) Particle swarm optimization. In: Proceedings of ICNN'95 - International Conference on Neural Networks. IEEE, pp 1942–1948
52. Kennedy J, Eberhart RC (1997) A discrete binary version of the particle swarm algorithm. In: 1997 IEEE International Conference on Systems, Man, and Cybernetics. Computational Cybernetics and Simulation. pp 4104–4108 vol.5
53. Shi Y, Eberhart R (1998) A modified particle swarm optimizer. In: 1998 IEEE International Conference on Evolutionary Computation Proceedings. IEEE World Congress on Computational Intelligence (Cat. No.98TH8360). pp 69–73

54. Zhan Z, Zhang J, Li Y, Chung HS (2009) Adaptive particle swarm optimization. IEEE Trans Syst Man Cybern Part B 39:1362–1381. https://doi.org/10.1109/TSMCB.2009.2015956
55. Eshtay M, Faris H, Obeid N (2018) Improving extreme learning machine by competitive swarm optimization and its application for medical diagnosis problems. Expert Syst Appl 104:134–152. https://doi.org/10.1016/j.eswa.2018.03.024
56. Niu P, Ma Y, Li M et al (2016) A kind of parameters self-adjusting extreme learning machine. Neural Process Lett 44:813–830. https://doi.org/10.1007/s11063-016-9496-z
57. Kaloop MR, Kumar D, Samui P et al (2019) Particle swarm optimization algorithm-extreme learning machine (PSO-ELM) model for predicting resilient modulus of stabilized aggregate bases. Appl Sci 9:3221. https://doi.org/10.3390/app9163221
58. Naser MZ, Alavi AH (2021) Error metrics and performance fitness indicators for artificial intelligence and machine learning in engineering and sciences. Archit Struct Constr. https://doi.org/10.1007/s44150-021-00015-8
59. Ferri C, Hernández-Orallo J, Modroiu R (2009) An experimental comparison of performance measures for classification. Pattern Recognit Lett 30:27–38. https://doi.org/10.1016/j.patrec.2008.08.010
60. Bui DT, Ngo PTT, Pham TD et al (2019) A novel hybrid approach based on a swarm intelligence optimized extreme learning machine for flash flood susceptibility mapping. CATENA 179:184–196. https://doi.org/10.1016/j.catena.2019.04.009
61. Kaya E, Saritas I (2019) Durum Wheat Dataset. In: Towar. a real-time sorting Syst. Identif. Vitr. durum wheat kernels using ANN based their Morphol. colour, wavelet gaborlet Featur. https://www.kaggle.com/datasets/muratkokludataset/durum-wheat-dataset
62. Johnson KJ, Synovec RE (2002) Pattern recognition of jet fuels: comprehensive GC×GC with ANOVA-based feature selection and principal component analysis. Chemom Intell Lab Syst 60:225–237. https://doi.org/10.1016/S0169-7439(01)00198-8