Sinopfbd, 2024; 9(2): 388-420 https://doi.org/10.33484/sinopfbd.1470329

Sinop Üniversitesi Fen Bilimleri Dergisi
Sinop Uni J Nat Sci

E-ISSN: 2564-7873

https://dergipark.org.tr/tr/pub/sinopfbd

An Example of Classification Using a Neural Network Trained by the Zebra Optimization Algorithm

Emine BAŞ[1] and Şaban BAŞ[2]

How to cite: Baş, E., & Baş, Ş. (2024). An example of classification using a neural network trained by the zebra optimization algorithm. *Sinop Üniversitesi Fen Bilimleri Dergisi, 9*(2), 388-420. https://doi.org/10.33484/sinopfbd.1470329

Research Article

Corresponding Author
Emine BAŞ
ebas@ktun.edu.tr

ORCID of the Authors
E.B: 0000-0003-4322-6010
Ş.B: 0000-0002-4142-6580

Received: 18.04.2024
Accepted: 07.08.2024

Abstract

Artificial intelligence techniques are a broad field of research with training, computation and prediction capabilities. Among these techniques, artificial neural networks (ANNs) are widely used as a predictive model. Learning algorithms in ANN classifiers have great importance on the success of ANN. The ANN model generally uses gradient-based learning models. However, due to the disadvantages of gradient-based learning models in local search, they have begun to be replaced by heuristic-based algorithms in recent years. Heuristic algorithms have attracted the attention of many researchers in recent years due to their success in problem solving. In this study, the Zebra Optimization Algorithm (ZOA), which has been proposed recently to train ANN networks, was examined. The main purpose of this study is to train the neural network using ZOA and increase the sensitivity of the perceptron neural network. In this study, a new ANN network integrated with ZOA is proposed. In this study, a detailed parameter analysis was carried out to show the effect of the population size and maximum generation number parameter settings, which form the basis for ZOA, on the ANN network. Then, a parameter analysis was carried out for the number of layers, number of neurons and epoch values, which are important for ANN networks. Such an ideal ANN network has been identified. This ideal ANN model was run on seven different data sets and was successful in predicting accurate data. In addition, three different heuristic algorithms (Gazelle Optimization Algorithm (GOA), Prairie Dogs Optimization (PDO), and Osprey Optimization Algorithm (OOA)) selected from the literature were integrated on the same ANN model and compared with the results of ANN integrated with ZOA operated under similar conditions. The results reveal that the proposed algorithm leads to greater convergence with the neural network coefficient compared to other algorithms. In addition, the proposed method caused the prediction error in the neural network to decrease.

Keywords: ANN, zebra, layer, neuron, network, prediction

Zebra Optimizasyon Algoritması Tarafından Eğitilmiş Bir Sinir Ağının Kullanıldığı Sınıflandırma Örneği

Öz

Yapay zeka teknikleri eğitim, hesaplama ve tahmin yeteneklerine sahip geniş bir araştırma alanıdır. Bu teknikler arasında yapay sinir ağları (YSA) tahmin modeli olarak yaygın olarak kullanılmaktadır. YSA

[1]Konya Technical University, Faculty of Engineering and Nature Sciences, Department of Software Engineering, Konya, Türkiye

[2]Selcuk University, Beyşehir Ali Akkanat Vocational School, Konya, Türkiye

sınıflandırıcılarındaki öğrenme algoritmaları YSA'nın başarısı üzerinde büyük önem taşımaktadır. YSA modeli genellikle gradyan tabanlı öğrenme modellerini kullanır. Ancak yerel aramada gradyan tabanlı öğrenme modellerinin dezavantajları nedeniyle son yıllarda yerini sezgisel tabanlı algoritmalar almaya başlamıştır. Sezgisel algoritmalar problem çözmedeki başarılarından dolayı son yıllarda birçok araştırmacının dikkatini çekmiştir. Bu çalışmada YSA ağlarının eğitimi için son dönemde önerilen Zebra Optimizasyon Algoritması (ZOA) incelenmiştir. Bu çalışmanın temel amacı sinir ağını ZOA kullanarak eğitmek ve algılayıcı sinir ağının duyarlılığını arttırmaktır. Bu çalışmada ZOA ile entegre yeni bir YSA ağı önerilmektedir. Bu çalışmada ZOA'ya temel oluşturan popülasyon büyüklüğü ve maksimum nesil sayısı parametre ayarlarının YSA ağı üzerindeki etkisini göstermek amacıyla detaylı bir parametre analizi yapılmıştır. Daha sonra YSA ağları için önemli olan katman sayısı, nöron sayısı ve çağ değerleri için parametre analizi yapılmıştır. Böyle ideal bir YSA ağı belirlendi. Bu ideal YSA modeli yedi farklı veri seti üzerinde çalıştırılmış ve doğru verileri tahmin etmede başarılı olmuştur. Ayrıca literatürden seçilen üç farklı sezgisel algoritma (Ceylan Optimizasyon Algoritması (GOA), Çayır Köpekleri Optimizasyonu (PDO), and Balıkkartalı Optimizasyon Algoritması (OOA)) aynı YSA modeli üzerine entegre edilmiş ve benzer koşullar altında çalışan ZOA ile entegre edilmiş YSA'nın sonuçları ile karşılaştırılmıştır. Sonuçlar, önerilen algoritmanın diğer algoritmalara göre sinir ağı katsayısı ile daha fazla yakınsamaya yol açtığını ortaya koymaktadır. Ayrıca önerilen yöntem sinir ağındaki tahmin hatasının azalmasına neden olmuştur.

Anahtar Kelimeler: ANN, zebra, katman, nöron, ağ, tahmin

Introduction

Artificial Neural Networks (ANNs), proposed in 1943, are one of the most widely used artificial intelligence approaches in the literature [1]. ANN was inspired by the biological nervous system. ANNs have been used in various problem solutions in recent years (classification, regression, pattern recognition, forecasting and time series problems, etc.) [2-5]. ANNs are a frequently used classifier in the field of data mining. ANNs can be used for supervised and unsupervised learning. Training of ANNs is one of the most important tasks. It has a complex structure. Classification error is minimized by updating the weights during the ANN training process. This means ANNs can recognize patterns and respond to their behavior accordingly [6-8]. There are two types of supervised trainers: deterministic trainers and stochastic trainers. Gradient descent and backpropagation-based methods are well-known deterministic trainers [9]. Learning algorithms used in ANN network training are generally gradient-based learning algorithms. These algorithms have several negative aspects. These algorithms depend on local minima and primary weights. Additionally, it may not show the same performance on all datasets [10, 11]. Therefore, despite their simplicity and fast convergence rates, they are not reliable in practical applications trainers [12]. Stochastic algorithms, on the other hand, start the learning process with stochastic solutions and improve them. Randomness is the most important feature of stochastic trainers. The most important advantage of stochastic trainers is the avoidance of high local minima. However, their most important disadvantage is that they work slower than deterministic algorithms. When the literature is examined, it shows that stochastic trainers are more preferred due to their ability to avoid local minima. Stochastic trainers are divided into two main categories: single solution and multiple solutions. It has been proven in many studies that multi-solution stochastic trainers avoid local optimum traps better than single-solution stochastic trainers [13-19]. Local pitfalls can be avoided by using meta-heuristic algorithms in ANN training. By using heuristic algorithms in ANN training, acceptable solutions are provided in a reasonable time to solve complex problems. Heuristic algorithms are less likely to get stuck in local minima than gradient-based search algorithms. Heuristic algorithms can be used in almost all types of ANNs [11]. The slow convergence and learning ability deficiencies of ANN have been overcome with heuristic algorithms. Determining the weights and bias values of ANN with heuristic algorithms improved the learning process of ANN [20]. Various heuristic algorithms have been used for ANN training in the literature. Some of these are Chimp Optimization Algorithm (COA) [12], Invasive Weed Optimization (IWO) [21], Particle Swarm Optimization (PSO) [22, 23], Firefy algorithms (FA) [22], Genetic Algorithms (GA) [23], Arithmetic Optimization Algorithm (AOA) [24], etc. Khishe and Mosavi [12] developed an ANN trained with Chimpanzee Optimization Algorithm (ChOA) for classification of underwater acoustic dataset Movassagh et al. [21] designed an ANN training model with IWO and demonstrated its success on heart, cancer, and iris datasets in a 5 and 10 layer network structure. Dang et al. [22] developed an ANN model optimized with particle swarm optimization and firefy algorithm to predict the scour depths around circular piers in the equilibrium phase. Jamali et al. [23] proposed an Artificial Neural Network (ANN) model based on PSO-GA optimization algorithm to predict a Solar Space Heating System (SSHS) performance. Khatir et al. [24] proposed IANN-AOA and IANN-BCMO developed with Arithmetic Optimization Algorithm and Composite Motion Optimization (BCMO) and solved the problem of damage measurement. They compared both methods. For damage measurement, IANN-AOA provided more accurate results than IANN-BCMO. The proposed algorithm is compared with Ion Motion Algorithm (IMA), Gray Wolf Optimization (GWO) and a hybrid algorithm. The results prove that the newly proposed algorithm performs better than other benchmark algorithms in most cases. The results obtained were compared with an ANN network trained by the Levenberg-Marquardt (LM) algorithm, which is widely adopted in the literature. It can be seen that the prediction results obtained from the proposed models are better compared to the values obtained from the single ANN model trained by LM. To demonstrate the success of the PSO-GA-ANN model, the results are compared to High Exploration Particle Swarm Optimization (HEPSO) and Team Game Algorithm (TGA). According to the results, the highest R2 and RMSE belong to PSO-GA-ANN. Gurgenc et al. [25] trained the MLP network with the adaptive opposition slime mold algorithm and estimated the reservoir temperature of geothermal resources. The results were compared with MLP-ANNs and basic artificial neural networks trained with the whale optimization algorithm and the antlion algorithm under equal conditions. The results prove that AOSMA-MLP outperforms the baseline MLP and other metaheuristic-based MLPs. Altay and Altay [26] also developed the Gray Wolf Optimizer (GWO) and hybridized the developed new GWO and MLP. As a result, the IMP-GWO-MLP algorithm was proposed and its success was tested on various datasets. The obtained results were proposed in the literature and compared with the commonly used GWO, particle swarm optimization, whale optimization algorithm, antlion algorithm and genetic algorithm-based MLP methods. Experimental results show that the proposed method is superior to other current methods in the literature. Altay et al. [27] hybridized Gray Wolf Optimizer with MLP (GWO-MLP) and used naïve Bayes classifier, K-nearest neighbor, linear discrimination analysis, binary decision tree and support vector machine approaches to predict the reservoir temperature using hydrogeochemical data of different. They used it in geothermal areas in Anatolia [27]. Altay and Gurgenc [28] estimated wear losses using the proposed hybrid golden jackal optimizer-multilayer perceptron (GJO-MLP) method. The performance of GJO-MLP was compared with whale optimization-MLP (WOA-MLP), genetic algorithm-MLP (GA-MLP) and antlion optimization-MLP (ALO-MLP) methods. Cinar [29] trained a feed forward MLP (FF MLP) networks using the Tree Seed Algorithm. Particle swarm optimization, gray wolf optimizer, genetic algorithm, ant colony optimization, evolution strategy, population-based incremental learning, artificial bee colony, biogeography-based optimization were compared with TSA. The results confirmed the superiority of TSA. In this study, the newly proposed Zebra Optimization Algorithm (ZOA) was used in ANN training [30]. The reason why the ZOA algorithm was preferred in this study is because it has been newly proposed in recent years. Herustic algorithms continue to be proposed in recent years. The success of the newly proposed heuristic algorithms is higher than the old algorithms. Due to the success of ZOA in the tests performed in the original paper, it was preferred as the heuristic algorithm for MLP-ANN training in this study. Additionally, when the literature was examined, ZOA had never been used as a training algorithm in MLP-ANN before. The motivation for this study begins at this point. Learning algorithms are of great importance in ANN-based classifier models. In this study, an ANN training model with ZOA is proposed. The local and global search capabilities offered by ZOA have been transferred to the ANN training and learning model. Thus, ANN classification was performed faster. First of all, a detailed parameter analysis was carried out to determine the best parameter values. ANN training was carried out with the ZOA learning model on the zoo dataset for ten different population sizes (10, 20, 30, 40, 50, 60, 70, 80, 90, and 100) and the most appropriate population size was determined as 100. Then, the effect of four different maximum iteration values (20, 50, 75, and 100) on ANN classification is shown. Six different ANN network structures were determined and the effects of ZOA on the ANN learning model were examined. The effect of four different epoch values (500, 1000, 5000, and 10000) on the ANN learning model of ZOA is detailed in this study. In this study, the effect of ZOA on the ANN learning model was demonstrated on six different datasets (somerville happiness survey 2015, iris, breast cancer wisconsin, wine, ecoli, and fertility), apart from the zoo dataset, on the ANN network model determined using the most appropriate parameter values. In addition, the effect of Gazelle Optimization Algorithm (GOA) [31], Prairie Dogs Optimization (PDO) [32], and Osprey Optimization Algorithm (OOA) [33] heuristic algorithms on the ANN learning model is shown and compared with ZOA. The results showed that ZOA can be used as an training model. In this study, the success of ZOA in an ANN training model was demonstrated for the first time. In this respect, this study shows originality. The rest of this work follows: In Section 2, the structure of ZOA, ANN training model, and dataset definitions are explained. In Section 3, parameter analyzes of ZOA determined for ANN training and comparisons of ZOA with different heuristic algorithms are presented. In the last section, the results are explained.

Related Works

Heuristic algorithms are frequently used in the literature in training Multi-Layer Perceptron Artificial Neural Networks (MLP-ANN*)*. Some of these are presented in Table 1.

Table 1. *A comprehensive review of related work on MLP-ANN*

References	Using Heuristic Algorithms	Models of ANN	Recommended Method
[12]	Chimpanzee Optimization Algorithm (COA)	Multilayer perceptron (MLP)	MLP-ChOA
[21]	Invasive Weed Optimization (IWO)	Multilayer perceptron (MLP)	MLP-IWO
[22]	Particle Swarm Optimization (PSO) and Firefy Algorithm (FA)	Multilayer perceptron (MLP)	MLP-PSO MLP-FA
[23]	Particle Swarm Optimization (PSO) and Genetic Algorithm (GA)	Multilayer perceptron (MLP)	PSO-GA-ANN
[24]	Arithmetic Optimization Algorithm (AOA) and Composite Motion Optimization (BCMO)	Multilayer perceptron (MLP)	IANN-AOA IANN-BCMO
[25]	Adaptive Opposition Slime Mold Algorithm (AOSMA)	Multilayer perceptron (MLP)	AOSMA-MLP
[26, 27]	Gray Wolf Optimizer (GWO)	Multilayer perceptron (MLP)	IMP-GWO-MLP
[28]	Golden Jackal Optimizer	Multilayer perceptron (MLP)	GJO-MLP
[29]	Tree Seed Algorithm (TSA)	Multilayer perceptron (MLP)	TSA-MLP

The Main Contribution of the Study

- ZOA is used for training the feed forward (FF) MLP ANN for the first time.
- ZOA is compared and outperformed on 7 different datasets with 3 metaheuristic algorithms (GOA, PDO, and OOA).
- ZOA finds eligible weights and biases of FF MLP ANN.
- In terms of average classification rates, ZOA ranked second in 7 different datasets, except Zoo.

- A detailed analysis was made with ZOA for the effects of population sizes and maximum iteration on classification success.
- A detailed analysis was conducted with ZOA for the effects of 6 different MLP-ANN network structures on classification success.
- A detailed analysis was carried out with ZOA for the effects of 4 different epoch values on the classification success of MLP-ANN.

Zebra Optimization Algorithm (ZOA)

Zebras are animals from the horse breed and generally live in eastern and southern Africa. The body feathers of these animals are black and white striped. Their most charismatic features come from this fur structure. Zebras are social living creatures. They exhibit two types of characteristic behaviors in social life. These are: food search and defense behaviors against predators. A zebra leads the zebras in their search for food. Lead zebras are responsible for guiding other zebras in the herd towards food sources. Zebras exhibit two behaviors to escape predators. The first of these is to escape with a zigzag movement pattern. The second is to come together and try to confuse or scare the predator [30]. Zebra Optimization Algorithm (ZOA) was created inspired by the behavior of zebras in social life.

Mathematical model of ZOA:

Initialization: The zebra population in ZOA is defined mathematically as candidate solutions searching the search space. Zebras are initially placed randomly in the search space, that is, on the plain where the food sources are located. The position of each zebra is a matrix of decision variables. The number of decision variables varies depending on the problem size. When the population matrix is first created in ZOA, it is randomly generated according to Equation 1 [30].

$$X = \begin{bmatrix} X_1 \\ \vdots \\ X_i \\ \vdots \\ X_{pop} \end{bmatrix}_{pop\times dim} = \begin{bmatrix} x_{1,1} & \cdots & x_{1,j} & \cdots & x_{1,dim} \\ \vdots & \ddots & \vdots & \iddots & \vdots \\ x_{1,i} & \cdots & x_{i,j} & \cdots & x_{i,dim} \\ \vdots & \iddots & \vdots & \ddots & \vdots \\ x_{pop,i} & \cdots & x_{pop,j} & \cdots & x_{pop,dim} \end{bmatrix}_{pop\times dim} \tag{1}$$

where X is zebra population, X_i is the i^{th} zebra, $x_{i,j}$ is position for the j^{th} dimension of the i^{th} zebra, pop is the population size of the zebra, and dim is the dimension of the problem. Each zebra individual represents one candidate solution. By using the size values of each zebra individual, the function values of the target zebras are calculated. Values from the objective function of the zebra population are stored in a matrix. This matrix structure is shown in Equation 2 [30].

$$Fitness = \begin{bmatrix} Fit_1 \\ \vdots \\ Fit_i \\ \vdots \\ Fit_{pop} \end{bmatrix}_{pop\times 1} = \begin{bmatrix} Fit(X_1) \\ \vdots \\ Fit(X_i) \\ \vdots \\ Fit(X_{pop}) \end{bmatrix}_{pop\times 1} \tag{2}$$

where $Fitness$ is the matris of the objective function values.

The values obtained with the objective function are compared with the individuals in the population and the leader zebra in the best position is determined. Depending on the type of problem, the zebra with the lowest fitness value or the zebra with the highest fitness value is determined as the best leader zebra. In each iteration, the positions of the zebras and their fitness values in their new positions are updated. Two types of behavior of zebras are used when determining new positions of the zebra population [30]. These behaviors are: (a) searching for food and (b) defending against predators.

(a) Foraging Behavior: Zebras spend most of their time eating food. Generally, their food sources are grasses and sedges. One of the zebras is defined as the plains zebra and this zebra leads the population. In ZOA, the best member of the population is considered the lead zebra and leads the other population members towards its position in the search area. Mathematical modeling of this stage is shown in Equations 3 and 4 [30].

$$x_{i,j}^{new1} = x_{i,j} + rand.(Zebra_j^{Best} - I.x_{i,j}) \tag{3}$$

$$X_i = \begin{cases} X_i^{new1}, & Fit_i^{new1} < Fit_i; \\ X_i, & else, \end{cases} \tag{4}$$

where X_i^{new1} is the new position of the i^{th} zebra based on foraging behavior, $x_{i,j}^{new1}$ is the j^{th} dimension position of the i^{th} new zebra, Fit_i^{new1} is the fitness value of the i^{th} new zebra, $Zebra_j^{Best}$ is the pioneer zebra, $rand$ is a random number in interval [0, 1], and I = round (1 + rand) [30].

(b) Defense Strategies Against Predators:

At this stage, the defense strategies of zebras against their enemies were modeled mathematically in order to update their positions in the search space of the zebra population. Zebras' defense strategies vary depending on the type of their enemies. They escape against their main enemies, the lions, in a zigzag pattern and with a random side-turning movement. They act in a confusing and frightening manner towards other enemies. These two defensive strategies are assumed to be similarly likely. In Equation 5, the defense strategy of zebras against lions is modeled in M1, and the defense strategy of zebras against other predators is modeled in M2. The position of the zebras is updated in Equation 6 [30].

$$x_{i,j}^{new2} = \begin{cases} M1: x_{i,j} + R.(2.rand - 1).\left(1 - \frac{Iter}{Iter_{max}}\right).x_{i,j}, & S \leq 0.5; \\ M2: x_{i,j} + rand.\left(Zebra^{Attack} - I.x_{i,j}\right), & else, \end{cases} \tag{5}$$

$$X_i = \begin{cases} X_i^{new2}, & Fit_i^{new2} < Fit_i; \\ X_i, & else, \end{cases} \tag{6}$$

where X_i^{new2} is the new position of the i^{th} zebra based on defense strategies behavior, $x_{i,j}^{new2}$ is the j^{th} dimension position of the i^{th} new zebra, Fit_i^{new2} is the fitness value of the i^{th} new zebra, $Zebra_j^{Attack}$ is the attack zebra, $rand$ is a random number in interval [0, 1], and I = round (1 + rand), $Iter$ is current iteration number, $Iter_{max}$ is maximum iteration number, R is a constant value (R=0.01). S is the probability of choosing one of the defense strategies for randomly generated zebras in the range [0, 1]. Figure 1 shows the flowchart of the ZOA [30].

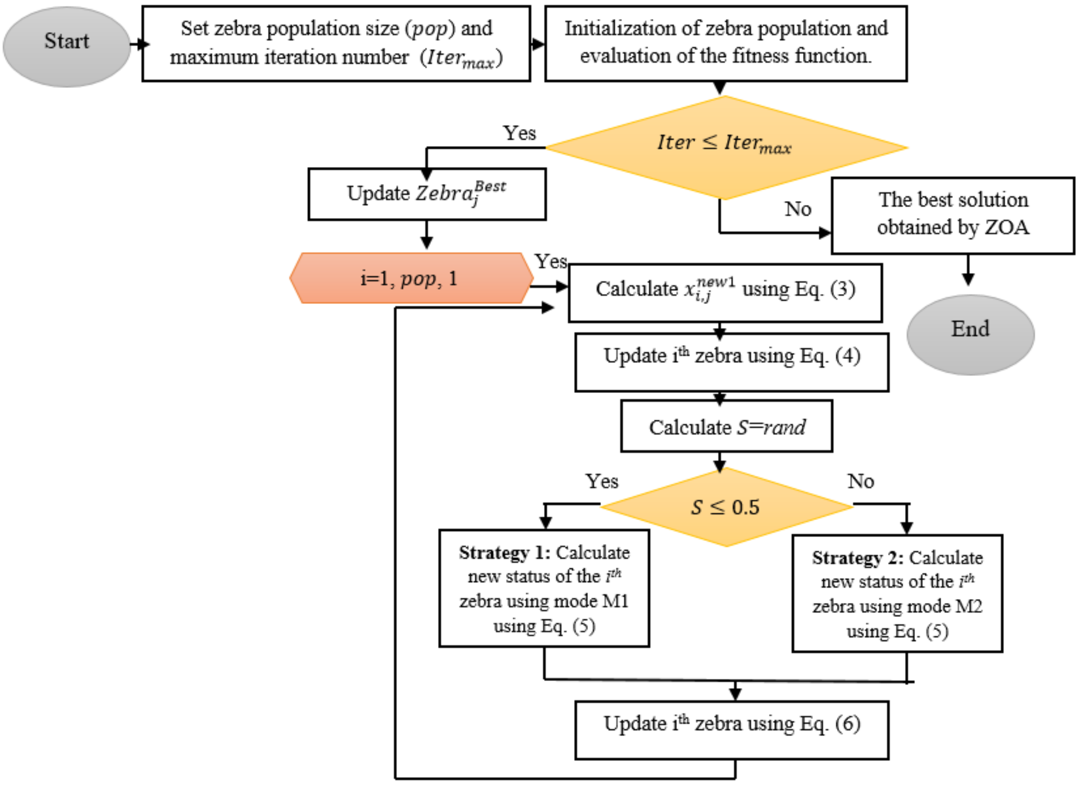

Figure 1. *The flowchart of the ZOA [30]*

Artificial Neural Network (ANN)

Artificial neural network (ANN) is a classification technique inspired by the human brain cell structure (neuron). Generally, ANNs consist of three layers. These are: input layer, hidden layers and output layer. An ANN structure can consist of a single hidden layer or it can consist of many hidden layers. The purpose of an ANN is to find the optimum weight values and make the most appropriate classification in the least possible iterations. There are many types of ANNs in the literature. Some of them are feedforward networks (FNNs) [34], Kohonen self-organizing networks [35], radial basis function (RBF) networks [36], recurrent neural networks [37], convolutional neural networks [38], spiking neural networks [39], etc. Multilayer perceptron (MLP), a special type of feed-forward networks (FNNs), is one of the most widely used models in the literature [34, 40]. In this study, an MLP-ANN structure was analyzed by training it with a metaheuristic algorithm selected from the literature. It is often seen that heuristic algorithms are used as training algorithms in MLP structures.

Multi-Layer Perceptron Artificial Neural Networks (MLP-ANN): Similar to ANNs, MLP works by matching a set of input values to a corresponding set of output values. This mapping is accomplished through a transformation process designed to derive the output. An MLP consists of three layers: the input layer contains n input values; The size of the hidden layer located between the input and output layers varies depending on the type of problem; and there is the output layer, which combines the results of the MLP network [25, 41]. The input layer hosts *n* neurons, the output layer includes *k* neurons, and the hidden layer comprises *m* neurons. Each neuron in the hidden layer performs two critical operations: summation and activation. The sum obtained is subsequently passed through an activation function, as depicted in Equation 7. Here $w_{i,j}$ is the connection weight between the hidden neuron j and the input neuron i. b_j is the bias value. y_j is the output value of neuron j, and f is the sigmoid function. Figure 2 shows a single hidden layer MLP network.

$$Sum_j = \sum_{i=1}^{n} w_{i,j} * in_i + b_j \tag{7}$$

$$y_j = f(Sum_j) \tag{8}$$

$$f(Sum_j) = \frac{1}{1 + e^{-Sum_j}} \tag{9}$$

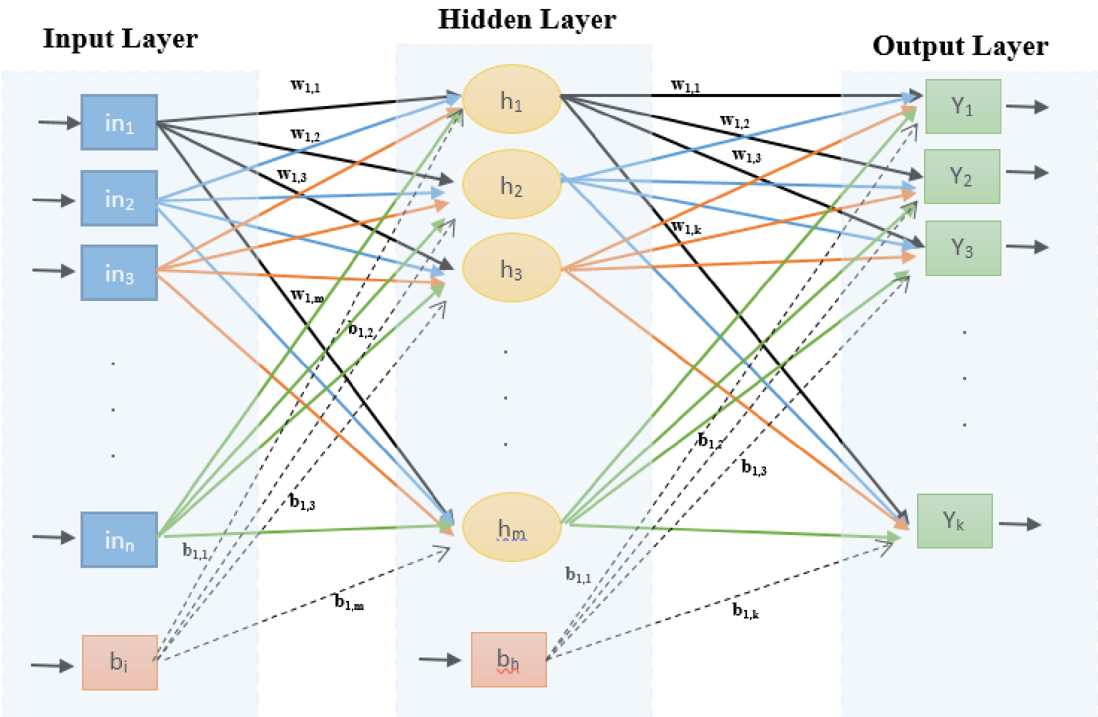

Figure 2. *The single hidden layer MLP network [25]*

Training An ANN and Dataset Definition

In this subsection, the updates of ZOA that can be classified with Multi Layered Perceptron ANN (MLP ANN) are explained. To train the ANN network, weights and bias values, which are ANN components, were placed on the dimensions of each population in the ZOA structure [12, 42]. The dimension value of zebra population individuals was calculated using Equation 10. The problem dimension was calculated using the number of inputs in the ANN (number of features in the datasets) (*m*) and the number of neurons in the hidden layers (*n*) (Equation 11). The ANN network was created by recombining the ANN values (weights and biases) in each dimension value from the zebra population. Classification was made using the created network. Mean Square Error (MSE) was used to evaluate the classification rate. MSE has also been used to evaluate ZOA individuals as a fitness function. MSE calculation is shown in Equation 12 [12, 42]. In this study, the datasets shown in Table 2 obtained from the UCI library were used (https://archive.ics.uci.edu/) [43]. 80% of the datasets used in classification were set as training and 20% as test dataset.

$$Length\ of\ problem\ dimension = (m \times n) + (2 \times n) + 1 \tag{10}$$

$$Zebra_i = \ [weight_1\ weight_2\ weight_3 \ldots bias_1\ bias_2\ bias_3 \ldots] \tag{11}$$

$$Minimization \qquad Fitness\ Function = MSE = \frac{1}{k}\sum_{i=1}^{k}(X_{real} - X_{model})^2 \tag{12}$$

where X_{real} is desired values and X_{model} is evaluated values. k is the number of instances in the training dataset [12, 42].

Table 2. *Dataset descriptions*

ID	Dataset	Number of features	Number of instances	Number of classes	Missing values	Type
1	Zoo	17	101	7	No	Life
2	Somerville Happiness Survey 2015	7	143	2	No	Health and Medicine
3	Iris	5	150	3	No	Biology
4	Breast Cancer Wisconsin	31	569	2	No	Health and Medicine
5	Wine	14	178	3	No	Physics and Chemistry
6	Ecoli	8	336	8	No	Biology
7	Fertility	10	100	2	No	Health and Medicine

Results and Discussion

In this subsection, classification was made by training an ANN with the ZOA algorithm. All applications were carried out with a machine with the features used in Table 2. The success of ZOA's parameter settings in ANN training is analyzed in detail in this subsection. Analysis of parameter settings was performed on the zoo dataset.

Table 3. *PC specifications*

Name	Detailed settings
Hardware	
CPU	Core i5
Frequency	1.19 GHz
RAM	12 GB
Software	
Operating system	Windows 10 (64-bit)
Language	MATLAB R2014A

Parameter Analyzes

a- The Analyses of the Population Size: The success of ten different population values on ZOA was analyzed for ANN. The parameter settings used in the population analysis are shown in Table 4. The results are shown in Table 5. The best results are marked in bold. According to the results, the population size is directly proportional to the success of ZOA in ANN training. The most successful population size relative to the average is 100, 80 and 90, respectively. The least successful population values are 10, 20 and 30 respectively. According to the best value, the best population size is 60. According to the standard deviation, the best population size is 90. According to the time value, the fastest working population size is 10. Figure 3 shows the convergence chart of the population size analysis for ZOA on ANN. Figure 4 shows the boxplot of the population size analysis for ZOA on ANN. Figure 5 shows the graphics of the results from ANN trained with ZOA on zoo train data (for pop=100) and Figure 6 shows the graphics of the results from ANN trained with ZOA on zoo test data (for pop=100). In the graphs, it can be seen that as the population size increases, ZOA's success in ANN training increases. At the same time, for the value of 100, which is the most successful population amount, the actual values in both the training and test data sets in the zoo data set and the training and test results of the ANN trained with ZOA were compared graphically. In this study, the population size was selected as 20 in classifying other datasets with ANN. In Figure 3, the x axes value shows the MSE value and the y axes value shows the iteration number. According to Figure 3, the fastest convergences were obtained at pop=90 and pop=100 values. The slowest convergent value was pop=10. In Figure 4, the x axes value shows the MSE value and the y axes value shows the population sizes. According to Figure 4, the average values vary in almost all population values. In this case, as the population size changes, the similarities between the results also differ. In Figure 5, the x axes value shows the training data set class values, and the y axes value shows the number of training data set samples. Figure 5 shows the error amounts between the actual training dataset values and the predicted training datasets. According to Figure 5, there is not much difference between the predicted target value and the actual target values for the training data set. Close values have been estimated. In Figure 6, the x axes value shows the test data set class values, and the y axes value shows the number of test data set samples. Figure 6 shows the error amounts between the actual test dataset values and the predicted test datasets. According to Figure 6, there is not much difference between the predicted target value and the actual target values for the test data set.

Table 4. *Parameter settings*

Parameters	Values
Population size (pop)	10, 20, 30, 40, 50, 60, 70, 80, 90, 100
The maximum number of iterations ($Iter_{max}$)	20
Dimension	16
R value	0.1
The number of run	20
Training data rate	80% (81 instance for zoo dataset)
Test data rate	20% (20 instance for zoo dataset)
Search space boundary	[-1,1]
Hidden Layer number	1
Neuron number	5
Epochs (for ANN)	500
Transfer function (for ANN)	Tansig

Table 5. *The results of ZOA for population size analysis on zoo dataset*

MSE	pop =10	pop =20	pop =30	pop =40	pop =50	pop =60	pop =70	pop =80	pop =90	pop =100
Best	0.3472	0.4256	0.1805	0.3788	0.3042	**0.1652**	0.2770	0.1806	0.1675	0.1919
Worst	3.3380	2.7869	2.1350	1.7901	1.3299	1.0884	0.8718	0.6493	0.6810	0.7707
Median	1.5175	0.9517	0.5818	0.5720	0.6019	0.4783	0.5049	0.3849	0.3875	0.3565
Mean	1.7167	1.0625	0.7272	0.6809	0.6367	0.5228	0.5027	0.3894	0.3915	**0.3782**
SD	0.8521	0.5393	0.4969	0.3344	0.2702	0.2319	0.1609	0.1356	**0.1262**	0.1354
Time	**7.3397**	13.9212	20.7881	36.1899	36.7285	43.6268	54.3813	60.8385	69.7256	85.3010
Rank (According Mean)	10	9	8	7	6	5	4	2	3	**1**

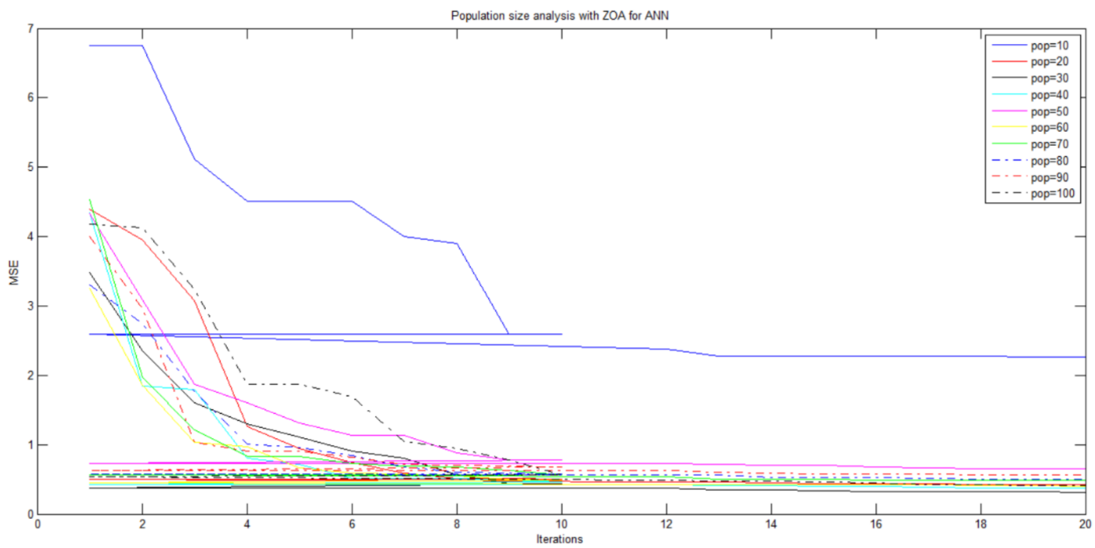

Figure 3. *The convergence chart of the population size analysis for ZOA on ANN*

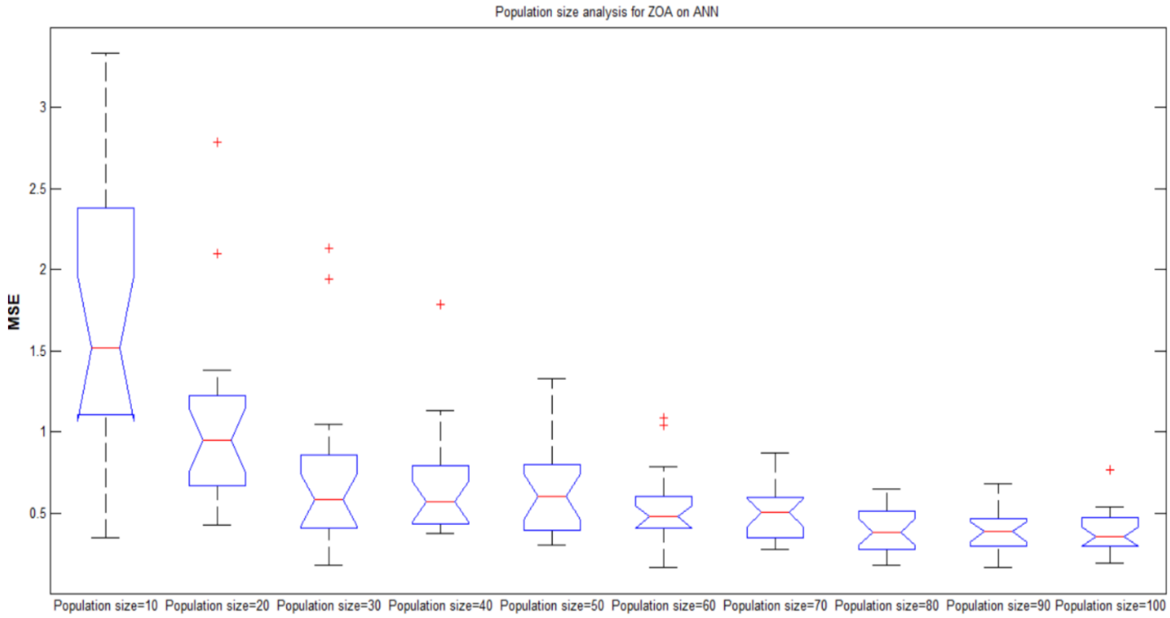

Figure 4. *The boxplot of the population size analysis for ZOA on ANN*

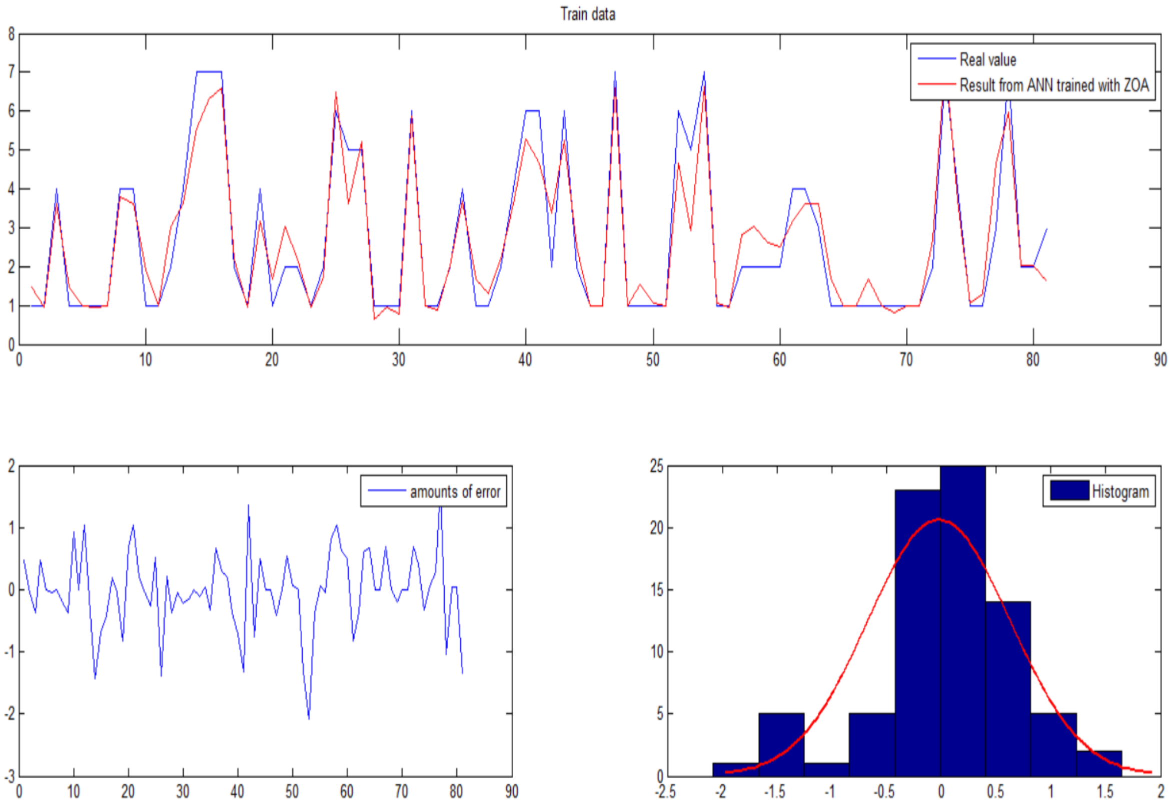

Figure 5. *The graphics of the results from ANN* trained *with ZOA on zoo train data (for pop=100).*

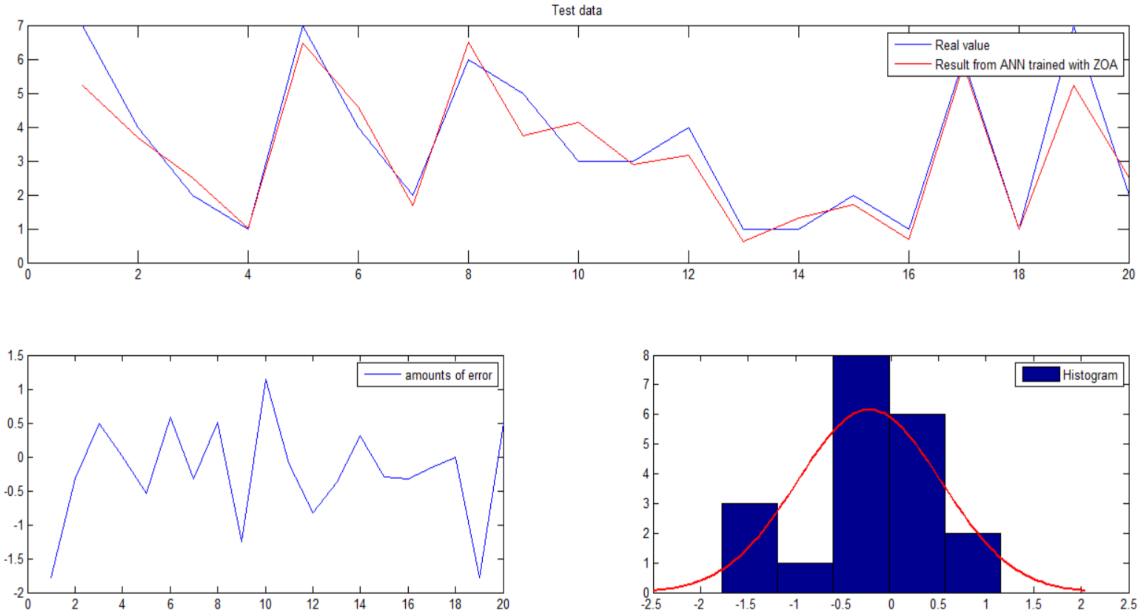

Figure 6. The graphics of the results from ANN trained *with ZOA on zoo test data (for pop=100)*

b- The Analyses of the Maximum Iteration: The success of four different maximum number of iterations on ZOA was analyzed for ANN. The parameter settings used in the maximum number of iterations are shown in Table 6. The results are shown in Table 7. The best results are marked in bold. According to the results, the maximum number of iterations is directly proportional to the success of ZOA in ANN training. The most successful maximum number of iterations relative to the average is 100 and 75, respectively. The least successful maximum number of iterations are 20 and 50, respectively. According to the best value, the best maximum number of iteration is 75. According to the standard deviation, the best maximum number of iteration is 75. According to the time value, the fastest working maximum number of iteration is 20. Figure 7 shows the convergence chart of the maximum number of iteration analysis for ZOA on ANN. Figure 8 shows the boxplot of the maximum number of iteration analysis for ZOA on ANN. Figure 9 shows the graphics of the results from ANN trained with ZOA on zoo train data (for $Iter_{max}$=100) and Figure 10 shows the graphics of the results from ANN trained with ZOA on zoo test data (for $Iter_{max}$=100). In the graphs, it can be seen that as the maximum number of iteration increases, ZOA's success in ANN training increases. At the same time, for the value of 100, which is the most successful number of the maximum iteration, the actual values in both the training and test data sets in the zoo data set and the training and test results of the ANN trained with ZOA were compared graphically. In this study, the maximum number of iteration was selected as 50 in classifying other datasets with ANN. In Figure 7, the x axes value shows the MSE value and the y axes value shows the iteration number. According to Figure 7, the fastest convergences were obtained at maximum iteration=75 and maximum iteration=100 values. The slowest convergent value was maximum iteration=20. In Figure 8, the x axes value shows the MSE value and the y axes value shows the number of the maximum iteration. According to Figure 8, the average values vary in almost all numbers of the maximum iterations. In this case, as the number of the maximum iterations changes, the similarities between the results also differ. In Figure 9, the x axes value shows the training data set class values, and the y axes value shows the number of training data set samples. Figure 9 shows the error amounts between the actual training dataset values and the predicted training datasets. According to Figure 9, there is not much difference between the predicted target value and the actual target values for the training data set. Close values have been estimated. In Figure 10, the x axes value shows the test data set class values, and the y axes value shows the number of test data set samples. Figure 10 shows the error amounts between the actual test dataset values and the predicted test datasets. According to Figure 10, there is not much difference between the predicted target value and the actual target values for the test data set.

Table 6. *Parameter settings*

Parameters	**Values**
Population size (pop)	20
The maximum number of iterations ($Iter_{max}$)	20, 50, 75, 100
Dimension	16
R value	0.1
The number of run	20
Training data rate	80% (81 instance for zoo dataset)
Test data rate	20% (20 instance for zoo dataset)
Search space boundary	[-1,1]
Hidden Layer number	1
Neuron number	5
Epochs (for ANN)	500
Transfer function (for ANN)	Tansig

Table 7. *The results of ZOA for population size analysis on zoo dataset*

MSE	$Iter_{max}$**=20**	$Iter_{max}$**=50**	$Iter_{max}$**=75**	$Iter_{max}$**=100**
Best	0.4256	0.2503	**0.1516**	0.1572
Worst	2.7869	0.9374	0.8963	0.9777
Median	0.9517	0.5579	0.3111	0.3998
Mean	1.0625	0.5568	0.3630	**0.4440**
SD	0.5393	0.1955	**0.1756**	0.1898
Time	**13.9212**	36.3699	55.5139	77.3042
Rank (According Mean)	4	3	2	**1**

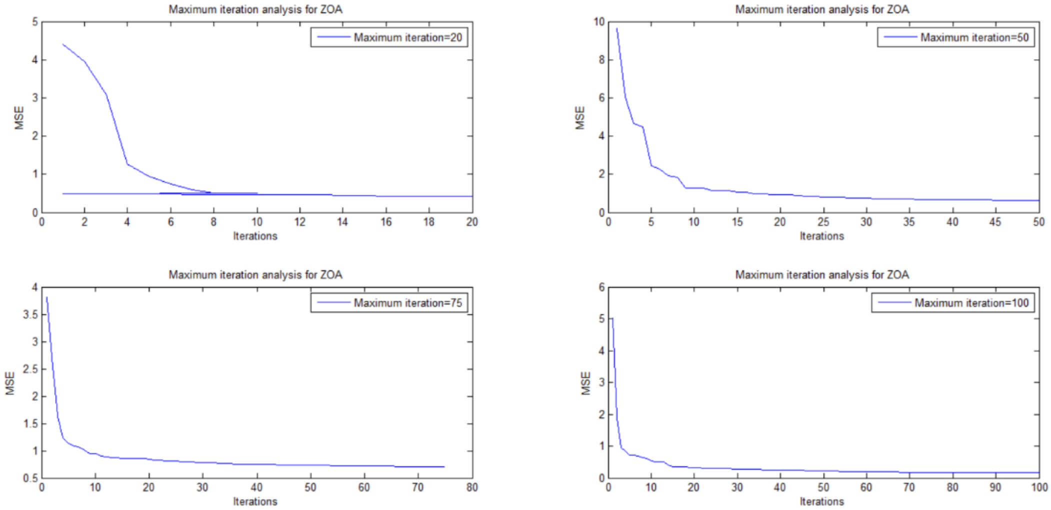

Figure 7. *The convergence chart of the maximum iteration analysis for ZOA on ANN.*

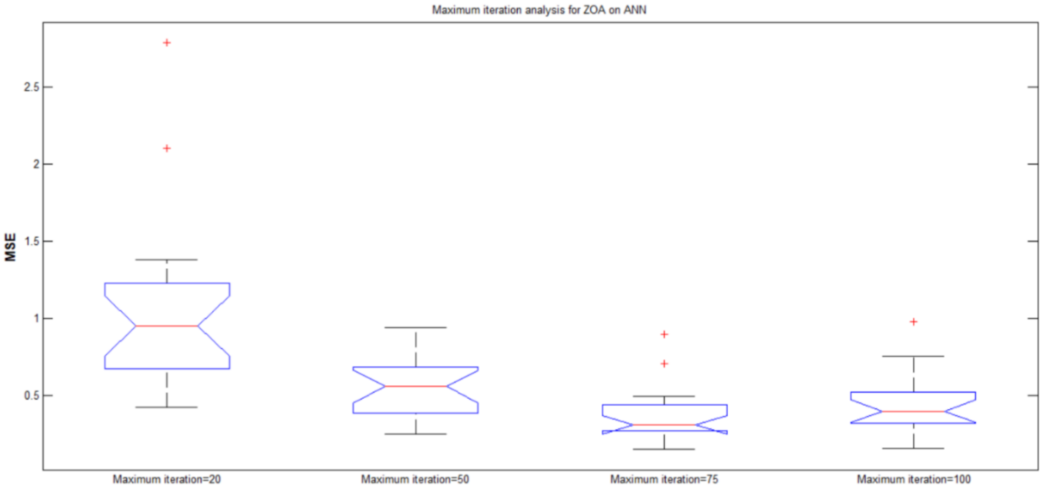

Figure 8. *The boxplot of the the maximum iteration analysis for ZOA on ANN.*

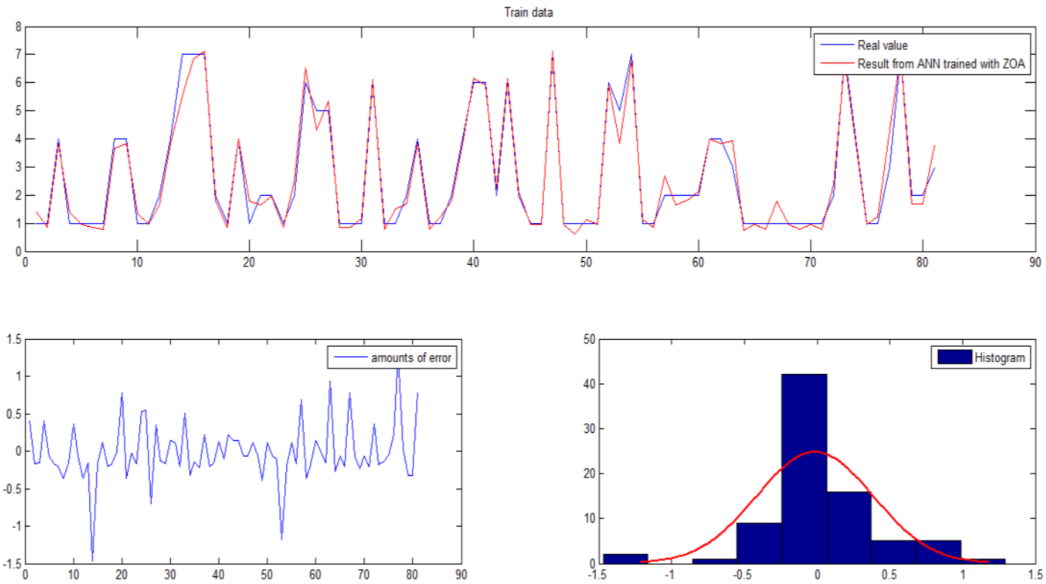

Figure 9. *The graphics of the results from ANN trained with ZOA on zoo train data (for $Iter_{max}$=100)*

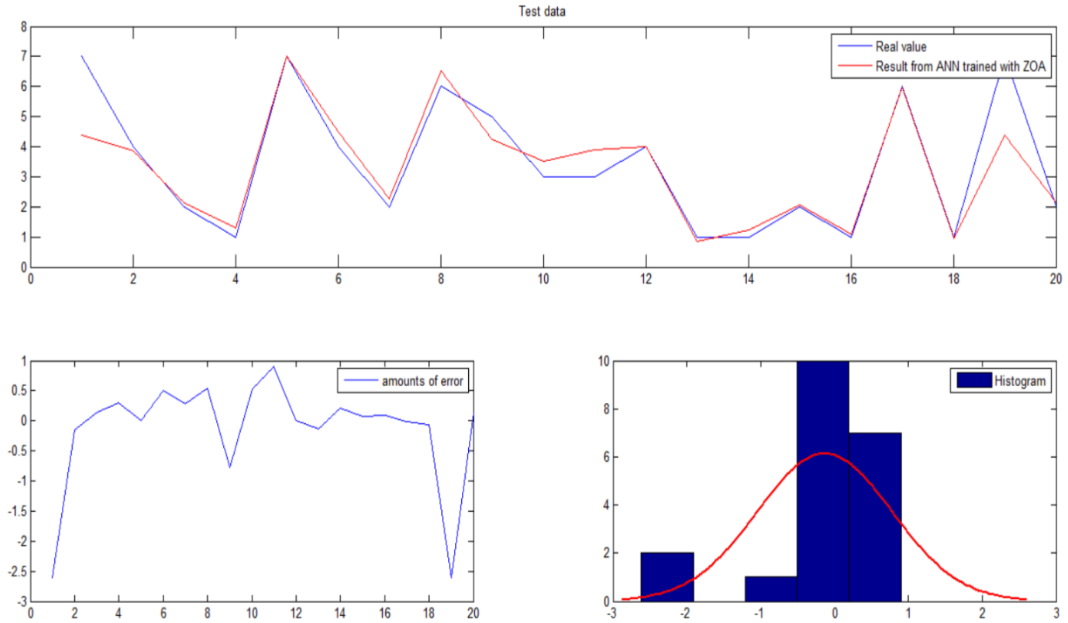

Figure 10. *The graphics of the results from ANN trained with ZOA on zoo test data (for $Iter_{max}$=100)*

c- The Analyses of the Layer and Neuron Number: The success of three different layer number and six different neuron number on ZOA was analyzed for ANN. The parameter settings used in the number of the layer and neuron are shown in Table 8. The results are shown in Table 9. The best results are marked in bold. According to average values, the most successful network design is a two-layer network design with five and ten neurons each (Network4 = {5,10}). Figure 11 shows the convergence chart of the different ANN networks analysis for ZOA.

Table 8. *Parameter settings*

Parameters	Values
Population size (pop)	20
The maximum number of iterations ($Iter_{max}$)	50
Dimension	16
R value	0.1
The number of run	20
Training data rate	80% (81 instance for zoo dataset)
Test data rate	20% (20 instance for zoo dataset)
Search space boundary	[-1,1]
Hidden Layer number	{1, 2, 3}
Neuron number	{5},{10},{5, 5},{5, 10},{5, 5, 5}, {10, 10, 10}
Epochs (for ANN)	500
Transfer function (for ANN)	Tansig, purelin

Table 9. *The results of ZOA for layer and neuron number analysis on zoo dataset*

MSE	**Network1={5}**	**Network2={10}**	**Network3={5,5}**	**Network4={5,10}**	**Network5={5, 5, 5}**	**Network6={10, 10, 10}**
Best	0.2503	0.2596	0.1826	**0.1595**	0.2419	0.2285
Worst	0.9374	0.9411	1.6367	0.8175	1.7184	1.0251
Median	0.5579	0.3776	0.6605	0.3692	0.8875	0.3316
Mean	0.5568	0.4198	0.6827	**0.4168**	0.8804	0.4331
SD	0.1955	**0.1597**	0.3655	0.1789	0.3617	0.2034
Time	36.3699	25.7894	27.7151	**24.6374**	47.5577	46.2427
Rank (According Mean)	4	2	5	**1**	6	3

Figure 12 shows the boxplot of the different ANN networks analysis for ZOA. Figure 13 shows the graphics of the results from ANN trained with ZOA on zoo train data (for Network6={=10 10 10}) and Figure 14 shows the graphics of the results from ANN trained with ZOA on zoo test data (for Network6={=10 10 10}). In this study, the number of layers and neurons was selected as Network4 = {5,10} in the classification of other data sets with ANN. In Figure 11, the x axes value shows the MSE value and the y axes value shows the iteration number. According to Figure 11, the fastest convergences were obtained at Network4 and Network3. The slowest convergent value was Network5. In Figure 12, the x axes value shows the MSE value and the y axes value shows the number of the network. According to Figure 12, the average values vary in almost all networks (except Network2 and Network4). In this case, as the networks changes, the similarities between the results also differ. In Figure 13, the x axes value shows the training data set class values, and the y axes value shows the number of training data set samples. Figure 13 shows the error amounts between the actual training dataset values and the predicted training datasets. According to Figure 13, there is not much difference between the predicted target value and the actual target values for the training data set. Close values have been estimated. In Figure 14, the x axes value shows the test data set class values, and the y axes value shows the number of test data set samples. Figure 14 shows the error amounts between the actual test dataset values and the predicted test datasets. According to Figure 14, there is much difference between the predicted target value and the actual target values for the test data set.

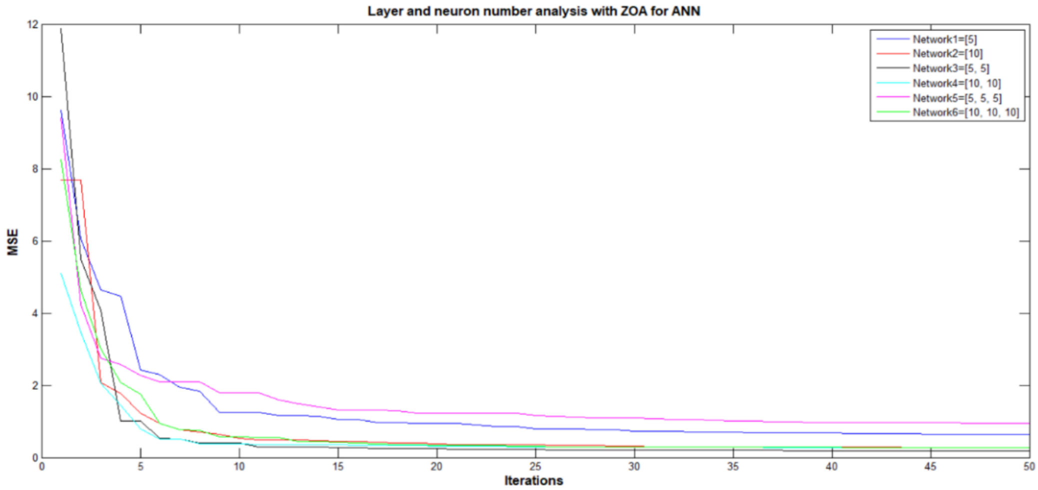

Figure 11. *The convergence chart of the layer and neuron number analysis for ZOA on ANN*

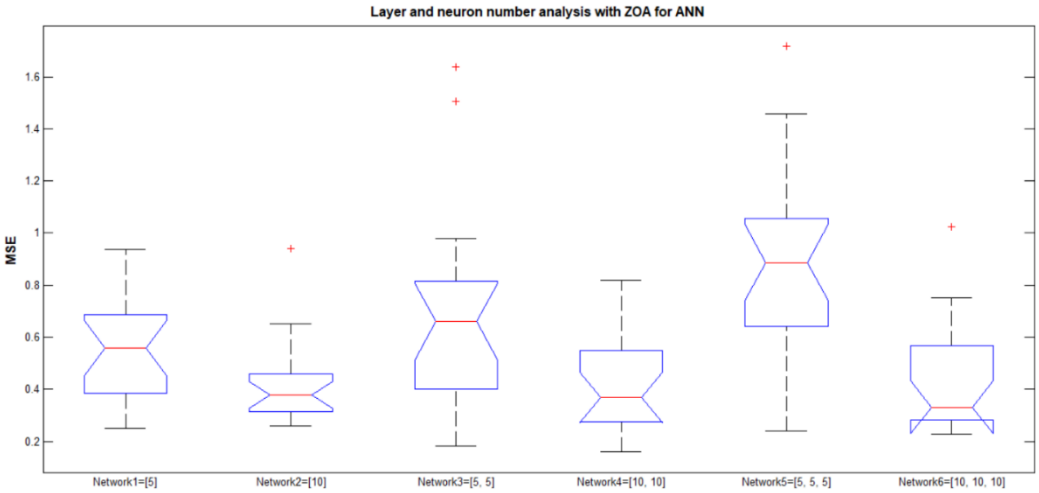

Figure 12. *The boxplot of the the layer and neuron number analysis for ZOA on ANN.*

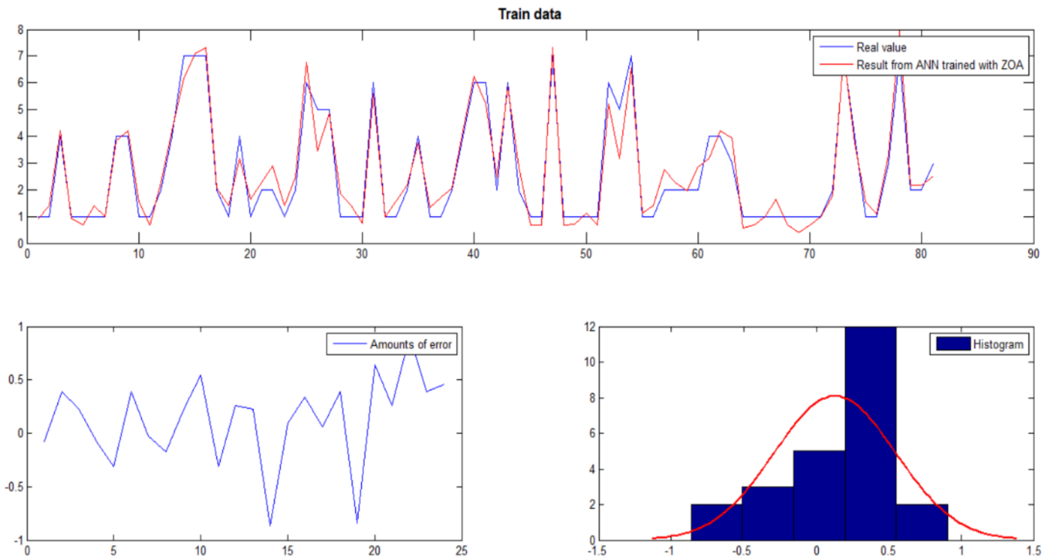

Figure 13. *The graphics of the results from ANN* trained *with ZOA on zoo train data (for Network6={10, 10, 10}).*

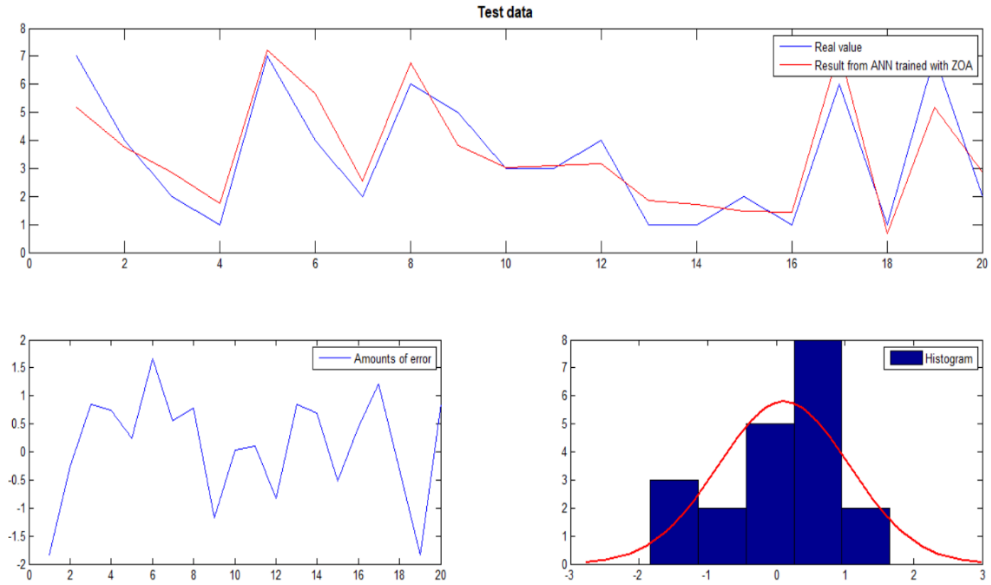

Figure 14. *The graphics of the results from ANN* trained *with ZOA on zoo test data (for Network6={10, 10, 10}).*

d- The Analyses of the Number of Epoch for ANN: The success of four different epochs values on ZOA was analyzed for ANN. The parameter settings used in the epochs values (500, 1000, 5000, and 10000) are shown in Table 10. The results are shown in Table 11.

Table 10. *Parameter settings*

Parameters	**Values**
Population size (pop)	20
The maximum number of iterations ($Iter_{max}$)	50
Dimension	16
R value	0.1
The number of run	20
Training data rate	80% (81 instance for zoo dataset)
Test data rate	20% (20 instance for zoo dataset)
Search space boundary	[-1,1]
Hidden Layer number	2
Neuron number	{5, 10}
Epochs (for ANN)	500, 1000, 5000, 10000
Transfer function (for ANN)	Tansig, purelin

Table 11. *The results of ZOA for number of epochs analysis on zoo dataset*

MSE	**Epochs=500**	**Epochs=1000**	**Epochs=5000**	**Epochs=10000**
Best	0.1595	0.1843	0.2835	0.2054
Worst	0.8175	1.0438	0.8614	1.0442
Median	0.3692	0.4306	0.5215	0.5908
Mean	0.4168	0.4796	0.5464	0.5692
SD	0.1789	0.2362	0.1719	0.2324
Time	24.6374	41.8842	33.4862	25.1088
Rank (According Mean)	**1**	2	2	3

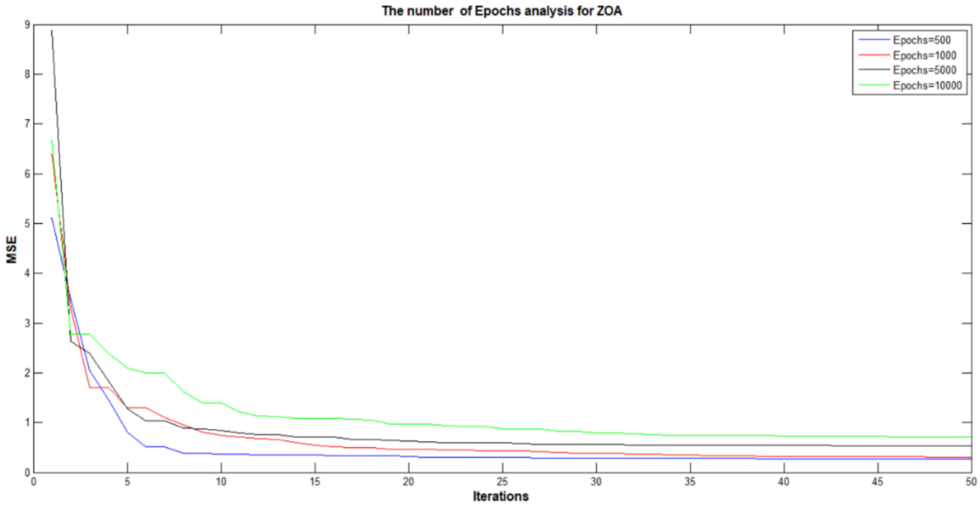

Figure 15. *The convergence chart of the number of epochs analysis for ZOA on ANN.*

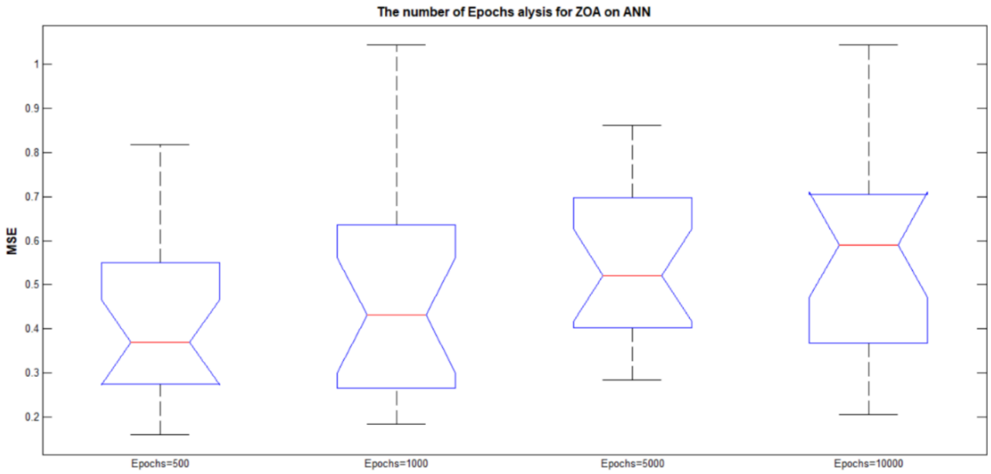

Figure 16. *The boxplot of the number of epochs analysis for ZOA on ANN*

The best results are marked in bold. According to average values, the most successful epoch value is 500. Figure 15 shows the convergence chart of the different epoch values analysis for ZOA. Figure 16 shows the boxplot of the different epoch values analysis for ZOA. Figure 17 shows the graphics of the results from ANN trained with ZOA on zoo train data (for Epoch value=1000) and Figure 18 shows the graphics of the results from ANN trained with ZOA on zoo test data (for Epoch value=1000). In this study, the epoch value was selected as 500 in the classification of other data sets with ANN. In Figure 15, the x-axis value shows the MSE value and the y-axis value shows the number of iterations. According to Figure 15, the fastest convergence was achieved at Epochs=500 and Epochs=1000. The slowest convergent value was Epochs=10000. In this case, the ANN network faced overfitting at high epoch values. In Figure 16, the x-axis value shows the MSE value and the y-axis value shows the number of epochs. According to Figure 16, the average values differ in almost all epochs. In this case, as the number of epoch change, the similarities between the results also differ. In Figure 17, the x-axis value shows the training data set class values, and the y-axis value shows the number of training data set samples. Figure 17 shows the error amounts between the actual training data set values and the predicted training data sets. According to Figure 17, there is not much difference between the predicted target value and the actual target values for the training data set. Close values have been estimated. In Figure 18, the x-axis value shows the test data set class values, and the y-axis value shows the number of test data set samples. Figure 18 shows the error amounts between the actual test data set values and the predicted test data sets. According to Figure 18, there is not a lot of difference between the predicted target value and the actual target values for the test data set.

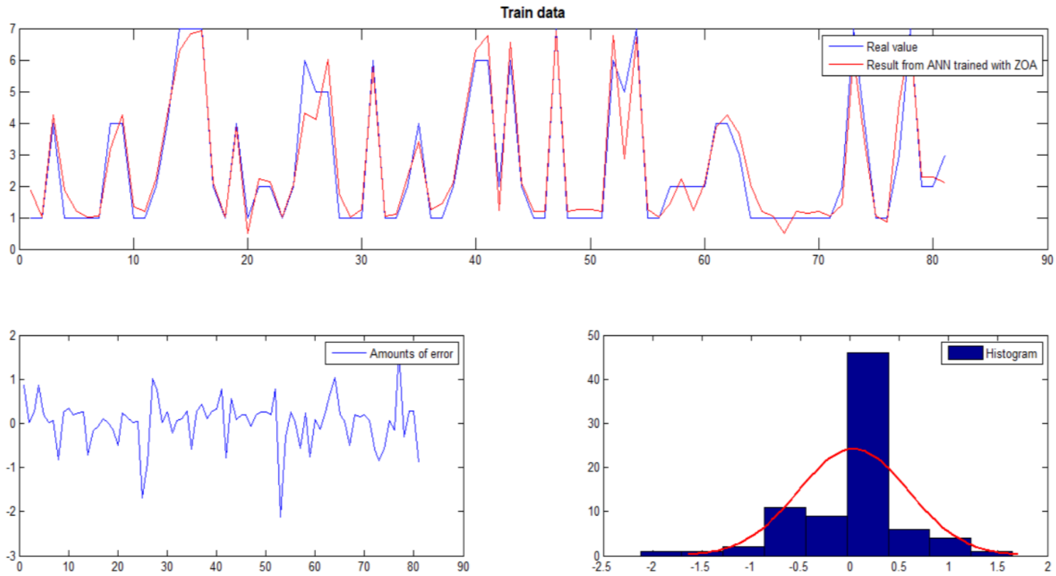

Figure 17. *The graphics of the results from ANN* trained *with ZOA on zoo train data (for Epochs=1000).*

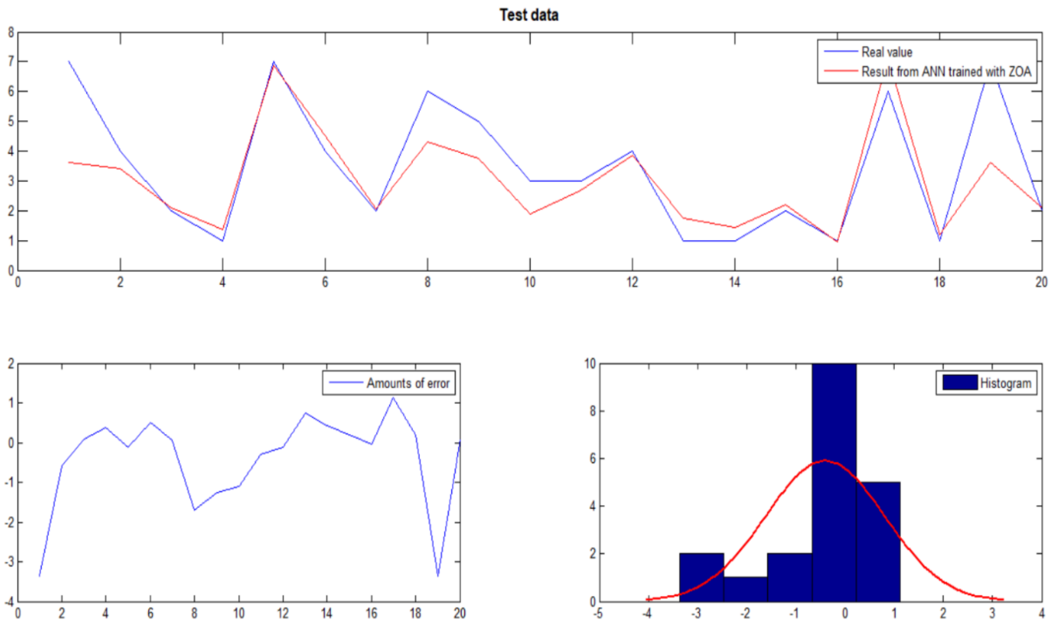

Figure 18. *The graphics of the results from ANN* trained *with ZOA on zoo test data (for Epochs=1000).*

Evaluation of the Success of the ZOA Algorithm and Other Algorithms on Different Data Sets: In this subsection, ZOA is compared with three different heuristic algorithms that have been proposed in recent years and selected from the literature. These heuristic algorithms are Gazelle Optimization Algorithm (GOA) [31], Prairie Dogs Optimization (PDO) [32], and Osprey Optimization Algorithm (OOA) [33]. GOA was created inspired by the behavior of gazelles. PDO was inspired by the social lifestyle of Prairie dogs. OOA was proposed inspired by the lifestyle of osprey creatures. The parameter settings used in the population analysis are shown in Table 12. The results are shown in Table 13.

Table 12. *Parameter settings*

Parameters	Values
Population size (pop)	20
The maximum number of iterations ($Iter_{max}$)	50
Dimension	16
R value for ZOA	0.1
The number of run	20
Training data rate	80%
Test data rate	20%
Search space boundary	[-1,1]
Hidden Layer number	2
Neuron number	{5, 10}
Epochs (for ANN)	500
Transfer function (for ANN)	Tansig, purelin
Fixed parameters for PDO	rho=0.005; epsPD=0.1
Fixed parameters for GOA	PSRs=0.34; S=0.88;

The best results are marked in bold. A detailed comparison analysis was performed on seven different datasets (zoo, somerville happiness survey 2015, iris, breast cancer wisconsin, wine, ecoli, and fertility). The details of these datasets are shown in Table 1. Total mean, standard deviation (SD), and time results are shown in Figure 19, Figure 20, and Figure 21, respectively. Figure 22 and Figure 23 show the convergence and box plots of the comparison algorithms on ANN training. Figure 24, Figure 25, and Figure 26 compare the results from the ANN trained and tested with ZOA, GOA, PDO and OOA on the zoo, wine, and iris datasets, respectively, with their real values. When Table 13 is examined, according to the average results, GOA algorithm is ranked first and ZOA algorithm is ranked second in almost every dataset. According to the total average results, ZOA, GOA, PDO and OOA are listed respectively. According to the total average results, the best heuristic algorithm that trained the ANN network was ZOA, and the worst heuristic algorithm was OOA. Figure 19 proves this situation. The best total standard deviation results belong to GOA, while the worst total standard deviation results belong to OOA. Figure 20 proves this situation. According to Figure 21, the fastest running heuristic algorithm was PDO, while the slowest running heuristic algorithm was OOA. The best working speeds of heuristic algorithms on ANN are listed as PDO, GOA, ZOA and OOA, respectively. Figure 22 shows the convergence of the comparison algorithms on each data set while training the ANN. In general, OOA converged slowly to optimum results, while ZOA converged faster. It can be seen that the ANN training success of ZOA on the iris data set is not very good. In Figure 23, the success of the comparison algorithms in ANN training on three different data sets is shown as box plots. On the Zoo data set, the results of ZOA, GOA and PDO algorithms, except OOA, are close to each other. There is consistency among the results obtained. The GOA algorithm obtained the most consistent results on the Iris data set. It is also seen that ZOA, PDO and OOA algorithms do not achieve very good results. The most successful ANN training on the Wine dataset belongs to ZOA and GOA. In this case, the box plots are close to each other and it can be said that the results are consistent. The heuristic algorithms that performed the least successful ANN training on the Wine dataset were PDO and OOA. In Figure 24, the predictions made by the comparison algorithms during ANN training and testing on the zoo dataset are compared with the real prediction values. According to the results, the best predictive heuristics were ZOA and GOA, while PDO and OOA were ranked lower. In Figure 25, the predictions made by the comparison algorithms during ANN training and testing on the wine dataset are compared with the real prediction values. A similar situation shown by the algorithms in Figure 24 can also be seen in Figure 26. In Figure 26, the predictions made by the comparison algorithms during ANN training and testing on the iris dataset are compared with the real prediction values. It is seen that a better ANN training is performed on the iris dataset with PDO and GOA heuristic algorithms.

Table 13. *The comparison results of ZOA and other algorithms on different data sets*

Datasets	MSE	ZOA	GOA	PDO	OOA
Zoo	***Best***	**0.1595**	0.5798	0.7316	4.2066
	Worst	0.8175	1.5346	2.9024	25.5013
	Median	0.3692	0.9183	1.7671	11.0427
	Mean	**0.4168**	0.9257	1.7519	12.5194
	SD	**0.1789**	0.2847	0.5680	6.7819
	Time	24.6374	24.4081	**18.1180**	36.6146
Rank		1	2	3	4
Somerville Happiness Survey 2015	***Best***	**0.1795**	0.1824	0.2054	0.1947
	Worst	0.2207	0.2028	0.2355	0.2279
	Median	0.2030	0.1951	0.2147	0.2081
	Mean	0.2011	**0.1941**	0.2164	0.2090
	SD	0.0110	**0.0050**	0.0081	0.0070
	Time	32.3040	25.6380	**17.2326**	38.0155
Rank		2	1	4	3
Iris	***Best***	0.0435	0.0392	**0.0344**	0.0379
	Worst	0.1573	0.0498	0.1005	0.1215
	Median	0.0684	0.0420	0.0539	0.0855
	Mean	0.0784	**0.0425**	0.0625	0.0847
	SD	0.0318	**0.0029**	0.0206	0.0243
	Time	60.3356	24.9541	**14.5459**	40.1815
Rank		3	1	2	4

Table 13 continued…

Breast Cancer Wisconsin	***Best***	0.0480	0.0467	**0.0376**	0.0680
	Worst	0.0931	0.0878	0.1030	0.2274
	Median	0.0583	0.0580	0.0644	0.1276
	Mean	0.0645	**0.0601**	0.0688	0.1411
	SD	0.0130	**0.0107**	0.0200	0.0437
	Time	31.8206	31.4186	**21.6767**	44.8578
Rank		2	1	3	4
Wine	***Best***	**0.0728**	0.0750	0.0775	0.1071
	Worst	0.1476	0.1198	0.2783	0.3567
	Median	0.1038	0.0997	0.1454	0.1909
	Mean	0.1065	**0.0988**	0.1577	0.1907
	SD	0.0219	**0.0136**	0.0528	0.0613
	Time	38.8456	27.5143	**20.0673**	40.0668
Rank		2	**1**	3	4
Ecoli	***Best***	0.8076	0.8019	0.8735	1.1652
	Worst	1.5705	1.0730	1.4427	2.3248
	Median	1.0107	0.9376	0.9918	1.4828
	Mean	1.0462	**0.9343**	1.0792	1.5353
	SD	0.1802	**0.0652**	0.1878	0.2944
	Time	28.8783	24.5822	**14.7665**	122.8615
Rank		2	**1**	3	4
Fertility	***Best***	**0.0872**	0.0934	0.1064	0.1028
	Worst	0.1090	0.1049	0.1232	0.1117
	Median	0.1006	0.1002	0.1103	0.1076
	Mean	0.1001	**0.0996**	0.1125	0.1073
	SD	0.0051	0.0034	0.0047	**0.0022**
	Time	38.9971	24.3091	**22.4168**	26.3487
Rank		2	**1**	4	3
Total mean		2.0136	2.3551	3.449	14.7875
Rank		**1**	2	3	4

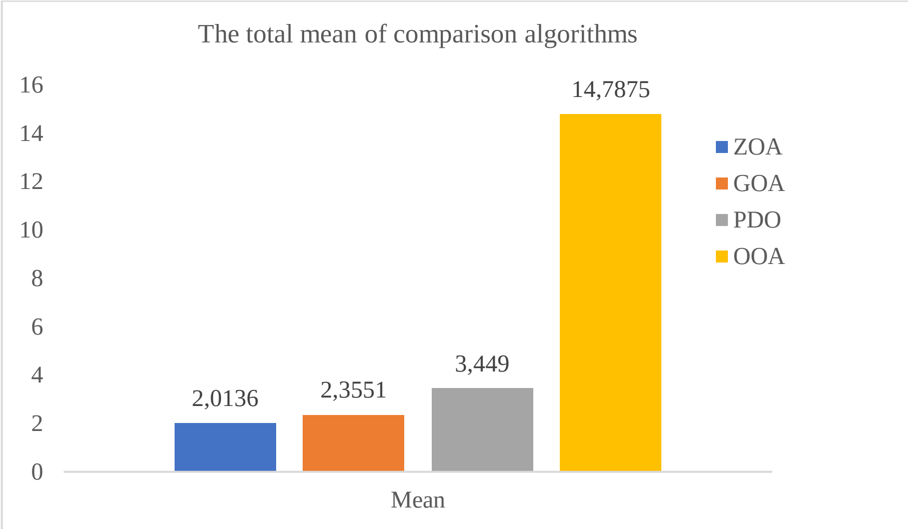

Figure 19. *The graphic of total mean of ZOA and comparison algorithms*

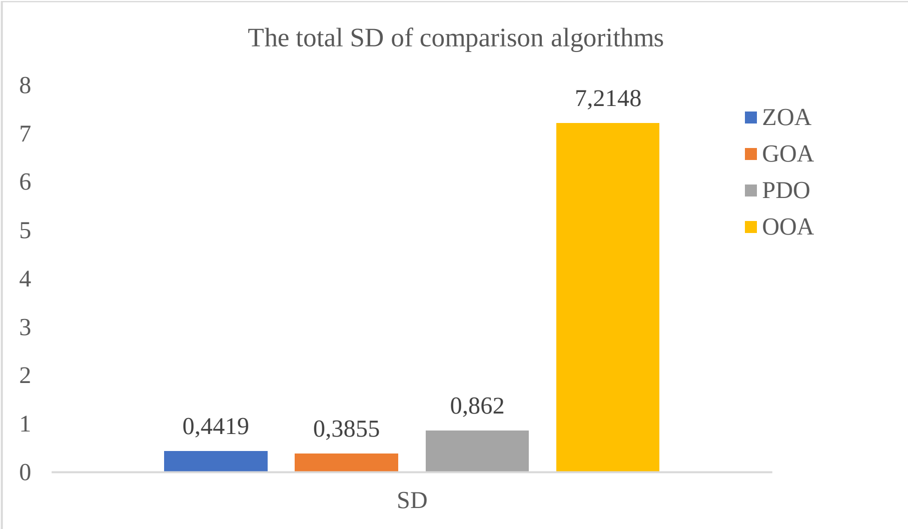

Figure 20. *The graphic of total SD of ZOA and comparison algorithms*

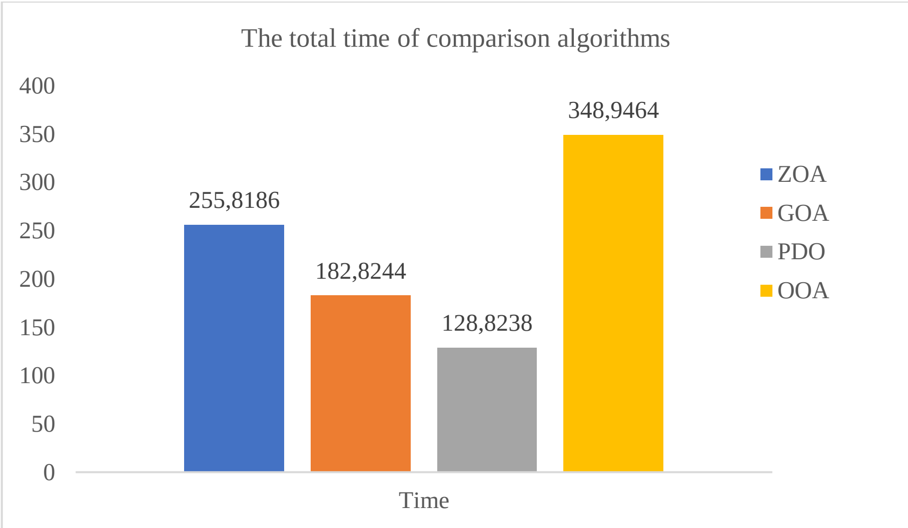

Figure 21. *The graphic of total time of ZOA and comparison algorithms*

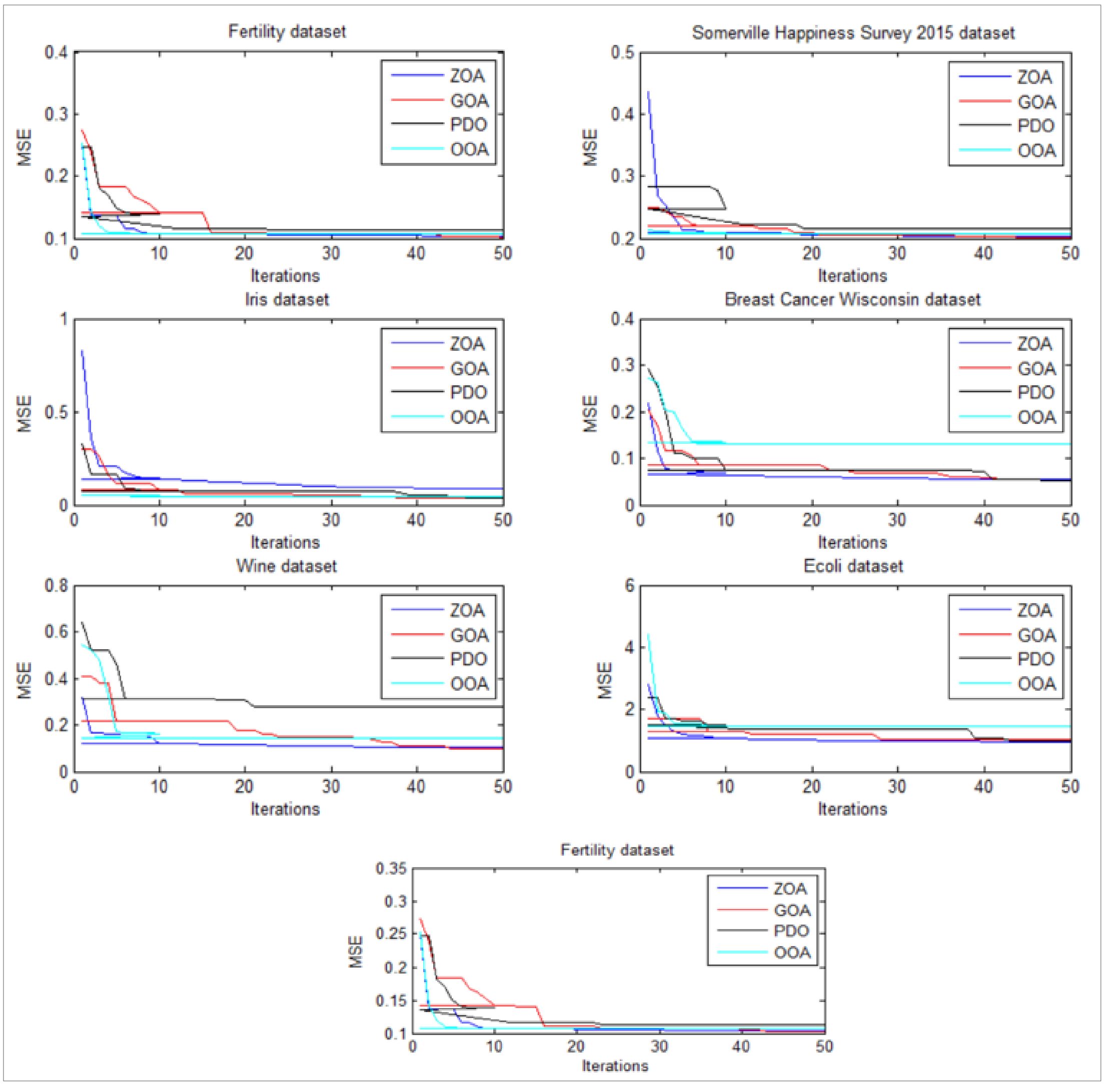

Figure 22. *The convergence chart of ZOA and comparison algorithms on ANN*

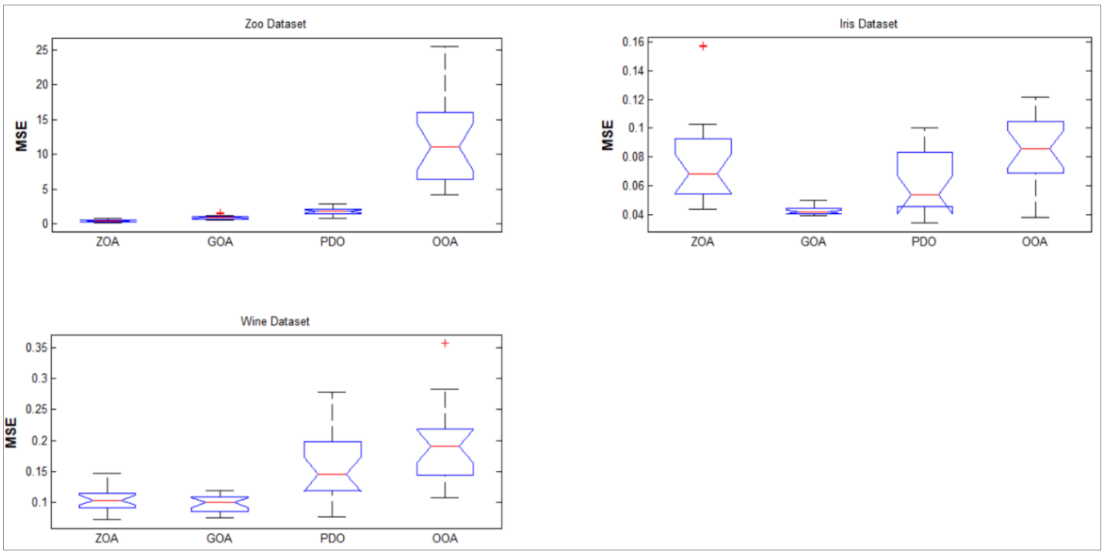

Figure 23. *The boxplot chart of ZOA and comparison algorithms on Zoo, Iris, and Wine datasets*

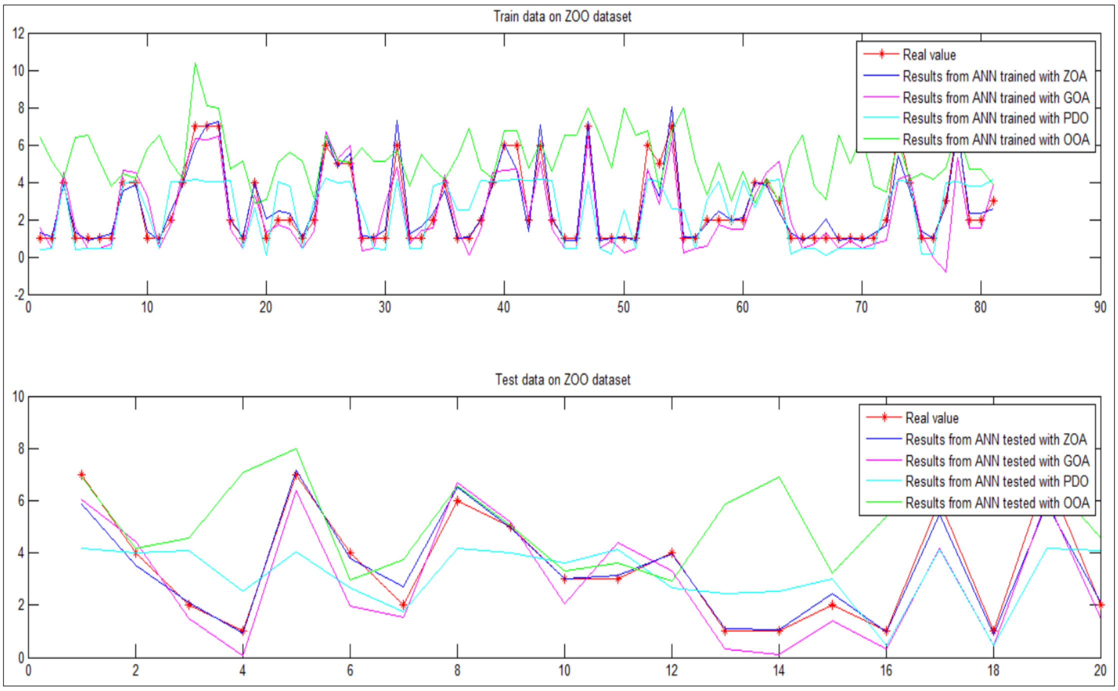

Figure 24. *The graphics of the results from ANN* trained *and tested with ZOA, GOA, PDO, OOA on zoo dataset*

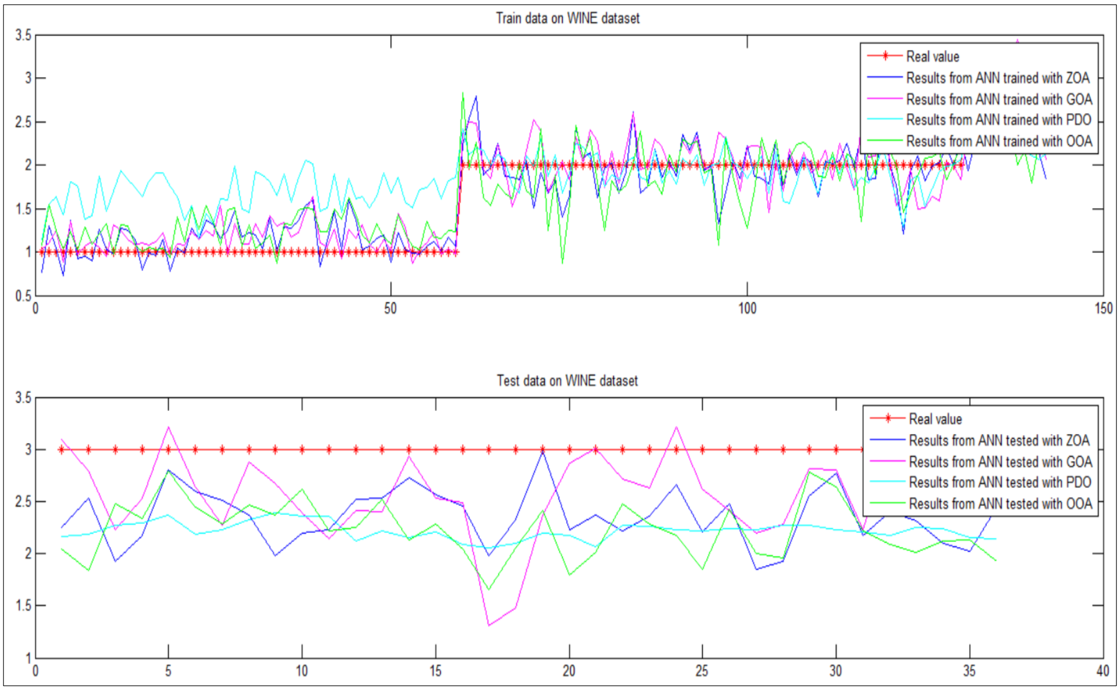

Figure 25. *The graphics of the results from ANN* trained *and tested with ZOA, GOA, PDO, OOA on wine dataset*

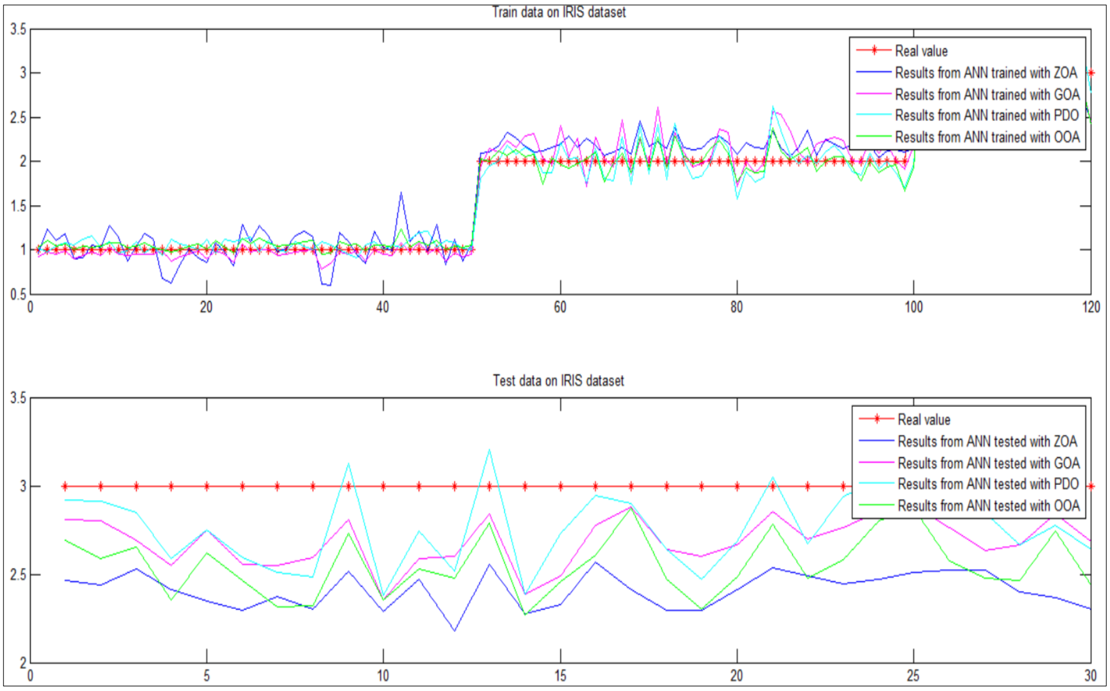

Figure 26. *The graphics of the results from ANN* trained *and tested with ZOA, GOA, PDO, OOA on iris dataset*

Conclusions

In this study, an Artificial Neural Network (ANN) model with predictive ability was designed on seven different data sets obtained from the UCI data set. In this designed ANN model, weight values were determined by the Zebra Optimization Algorithm (ZOA), a heuristic algorithm. Due to the success of heuristic algorithms in estimating weight values, classical gradient-based algorithms were abandoned in this study and replaced by heuristic-based algorithms. First, a detailed parameter analysis was carried out for parameter settings that are important for ZOA and ANN. Thus, the effects of population size and maximum number of iterations on ANN training are shown. Additionally, the effects of neuron, layer, and epoch values on the success of ANN are explained in detail. As a result, a suitable ANN network was designed with a ZOA with the most appropriate parameter values. Predictions were made on seven different data sets with this ANN model with ZOA. The results obtained were compared with the ANN model created with three different state-of-the-art heuristic algorithms (Gazelle Optimization Algorithm (GOA), Prairie Dogs Optimization (PDO), and Osprey Optimization Algorithm (OOA)) with similar network structures selected from the literature. The results show that the ANN model trained with ZOA is more successful in predicting layer weights. While faster convergence was achieved, the amount of error in predicting the classes of the data decreased. The predictive analytics performed by the proposed model can also improve ANN performance with proper convergence. While the ANN model integrated and trained with ZOA can find better coefficients, other algorithms used in this paper except GOA cannot obtain suitable outputs. ZOA's capabilities in local and global search are also reflected in the training of the ANN network. In the future, we plan to further increase the success of estimating ANN weights by improving ZOA's local and global search capabilities with the help of chaotic maps.

Acknowledgements The Zebra Optimization Algorithm (ZOA) code used in this study is available on MATLAB Central File Exchange: Zebra Optimization Algorithm (ZOA). Furthermore, the datasets for instances referenced in this study can be accessed at https://archive.ics.uci.edu/datasets. We would like to express our gratitude to these sources for providing valuable data for our research.

Funding/Financial Disclosure The author (s) has not received any financial support for the research. authorship or publication of this study.

Ethics Committee Approval and Permissions The study does not require ethics committee permission or any special permission.

Conflict of Interests No conflict of interest or common interest has been declared by the authors.

Authors Contribution The authors contributed equally to the study. The authors read and approved the final manuscript.

References

[1] McCulloch, W. S., & Pitts, W. (1943). A logical calculus of the ideas immanent in nervous activity. *The Bulletin of Mathematical Biophysics, 5*(4), 115-133. https://doi.org/10.1007/BF02478259

[2] Feng, Z. K., & Niu, W. J. (2021). Hybrid artificial neural network and cooperation search algorithm for nonlinear river flow time series forecasting in humid and semi-humid regions. *Knowledge-Based Systems*, *211*, 106580. https://doi.org/10.1016/j.knosys.2020.106580

[3] Fuqua, D., & Razzaghi, T. (2020). A cost-sensitive convolution neural network learning for control chart pattern recognition. *Expert Systems with Applications*, *150*, 113275. https://doi.org/10.1016/j.eswa.2020.113275

[4] Chatterjee, S., Sarkar, S., Hore, S. Dey, N., Ashour, A. S., & Balas, V. E. (2017). Particle swarm optimization trained neural network for structural failure prediction of multistoried RC buildings. *Neural Computing and Applications, 28*, 2005-2016. https://doi.org/10.1007/s00521-016-2190-2

[5] Ulas, M., Altay, O., Gurgenc, T., & Ozel, C. (2020). A new approach for prediction of the wear loss of PTA surface coatings using artificial neural network and basic, kernel-based, and weighted extreme learning machine. *Friction*, *8*, 1102-1116. https://doi.org/ 10.1007/s40544-017-0340-0

[6] Vosniakos, G. C., & Benardos, P. G. (2007). Optimizing feedforward artifcial neural network architecture. *Engineering Applications of Artificial İntelligence*, *20*(3), 365-382. https://doi.org/10.1016/j.engappai.2006.06.005

[7] Mosavi, M. R., Khishe, M., & Ghamgosar, A. (2016). Classification of sonar data set using neural network trained by gray wolf optimization. *Neural Network World*, *26*(4), 393-415. https://doi.org/ 10.14311/NNW.2016.26.023

[8] Mosavi, M. R., Khishe, M., Parvizi, G. R., Naseri, M. J., & Ayat, M. (2019). Training multi-layer perceptron utilizing adaptive best-mass gravitational search algorithm to classify sonar dataset. *Archives of Acoustics*, *44*(1), 137-51. https://doi.org/10.24425/aoa.2019.126360

[9] Mosavi, M. R., & Khishe, M. (2017). Training a feed-forward neural network using particle swarm optimizer with autonomous groups for sonar target classification. J. Circuits. *Journal of Circuits, Systems and Computers, 26*(11), 1-20. https://doi.org/10.1142/S0218126617501857

[10] Yaghini, M., Khoshraftar, M. M., & Fallahi, M. (2011). *HIOPGA: a new hybrid metaheuristic algorithm to train feedforward neural networks for prediction* [Conference presentation]. In Proceedings of the International Conference on Data Science (ICDATA). The Steering Committee of The World Congress in Computer Science, Computer Engineering and Applied Computing (WorldComp). Las Vegas, NV, USA. http://world-comp.org

[11] Kiranyaz, S., Ince, T., Yildirim, A., & Gabbouj, M. (2009). Evolutionary artifcial neural networks by multi-dimensional particle swarm optimization. *Neural Networks*, *22*(10), 1448-1462. https://doi.org/10.1016/j.neunet.2009.05.013

[12] Khishe, M., & Mosavi, M. R. (2020). Classification of underwater acoustical dataset using neural network trained by Chimp Optimization Algorithm. *Applied Acoustics*, *157*, 107005. https://doi.org/10.1016/j.apacoust.2019.107005

[13] Afrakhteh, S., Mosavi, M., Khishe, M., & Ayatollahi, A. (2020). Accurate classification of EEG signals using neural networks trained by hybrid population-physic-based algorithm. *International Journal of Automation and Computing*, *17*, 108-122. https://doi.org/10.1007/s11633-018-1158-3

[14] Khishe, M., Mosavi, M. R., & Kaveh, M. (2017). Improved migration models of biogeography based optimization for sonar data set classification using neural network. *Applied Acoustics, 118*, 15-29. https://doi.org/10.1016/j.apacoust.2016.11.012

[15] Khishe, M., Mosavi, M. R., & Moridi, A. (2018). Chaotic fractal walk trainer for sonar data set classification using multi-layer perceptron neural network and its hardware implementation. *Applied Acoustics*, *137*, 121-39. https://doi.org/10.1016/j.apacoust.2018.03.012

[16] Ravakhah, S., Khishe, M., Aghababaee, M., & Hashemzadeh, E. (2017). Sonar false alarm rate suppression using classification methods based on interior search algorithm. *International Journal of Computer Science and Network Security, 17*(7), 58-65.

[17] Mosavi, M. R., Khishe, M., & Akbarisani, M. (2017). Neural network trained by biogeographybased optimizer with chaos for sonar data set classification. *Wireless Personal Communications, 95*(4), 4623-4642. https://doi.org/10.1007/s11277-017-4110-x

[18] Kaveh, M., Khishe, M., & Mosavi, M. R. (2019). Design and implementation of a neighborhood search biogeography-based optimization trainer for classifying sonar dataset using multi-layer perceptron neural network. *Analog Integrated Circuits Signal Process*, *100*, 405-428. https://doi.org/10.1007/s10470-018-1366-3

[19] Mosavi, M. R., Khishe, M., Hatam Khani, Y., & Shabani, M. (2017). Training radial basis function neural network using stochastic fractal search algorithm to classify sonar dataset. *Iranian Journal of Electrical & Electronic Engineering, 13*(1), 100-111.

[20] Dangare, C. S., & Apte, S. S. (2012). Improved study of heart disease prediction system using data mining classifcation techniques. *International Journal of Computer Applications, 47*(10), 44-48.

[21] Movassagh, A. A., Alzubi, J. A., Gheisari, M., Rahimi, M., Mohan, S., Abbasi, A. A., & Nabipour, N. (2023). Artificial neural networks training algorithm integrating invasive weed optimization with differential evolutionary model. *Journal of Ambient Intelligence and Humanized Computing*, *14*, 6017-6025. https://doi.org/10.1007/s12652-020-02623-6

[22] Dang, N. M., Tran Anh, D., & Dang, T. D. (2021). ANN optimized by PSO and Firefly algorithms for predicting scour depths around bridge piers. *Engineering with Computers*, *37*, 293-303. https://doi.org/10.1007/s00366-019-00824-y

[23] Jamali, B., Rasekh, M., Jamadi, F., Gandomkar, R., & Makiabadi, F. (2019). Using PSO-GA algorithm for training artificial neural network to forecast solar space heating system parameters. *Applied Thermal Engineering*, *147*, 647-660. https://doi.org/10.1016/j.applthermaleng.2018.10.070

[24] Khatir, S., Tiachacht, S., Le Thanh, C., Ghandourah, E., Mirjalili, S., & Wahab, M. A. (2021). An improved Artificial Neural Network using Arithmetic Optimization Algorithm for damage assessment in FGM composite plates. *Composite Structures*, *273*, 114287. https://doi.org/10.1016/j.compstruct.2021.114287

[25] Gurgenc, E, Altay, O, & Altay, E. V. (2024). AOSMA-MLP: A novel method for hybrid metaheuristics artificial neural networks and a new approach for prediction of geothermal reservoir temperature. *Applied Sciences*, *14*(8), 3534. https://doi.org/10.3390/app14083534

[26] Altay, O., & Altay, E. V. (2023). A novel hybrid multilayer perceptron neural network with improved grey wolf optimizer. *Neural Computing and Applications*, *35*, 529–556. https://doi.org/10.1007/s00521-022-07775-4

[27] Altay, E. V., Gurgenc, E., Altay, O., & Dikici, A. (2022). Hybrid artificial neural network based on a metaheuristic optimization algorithm for the prediction of reservoir temperature using hydrogeochemical data of different geothermal areas in Anatolia (Turkey). *Geothermics*, *104*, 102476. https://doi.org/10.1016/j.geothermics.2022.102476

[28] Altay, O., & Gurgenc, T. (2024). GJO-MLP: a novel method for hybrid metaheuristics multi-layer perceptron and a new approach for prediction of wear loss of AZ91D magnesium alloy worn at dry, oil, and H-BN nanoadditive oil. *Surface Review and Letters (SRL)*, *31*, 06, 1-16. https://doi.org/10.1142/S0218625X24500483

[29] Cinar, A. C. (2020). Training feed-forward multi-layer perceptron artificial neural networks with a tree-seed algorithm. *Arabian Journal for Science and Engineering*, *45*, 10915–10938. https://doi.org/10.1007/s13369-020- 04872-1

[30] Trojovská, E., Dehghani, M., & Trojovský, P. (2022). Zebra optimization algorithm: A new bio-inspired optimization algorithm for solving optimization algorithm. *IEEE Access*, *10*, 49445-49473. https://doi.org/10.1109/ACCESS.2022.3172789

[31] Agushaka, J. O., Ezugwu, A. E. & Abualigah, L. (2023). Gazelle optimization algorithm: a novel nature-inspired metaheuristic optimizer. *Neural Computing and Applications*, *35*, 4099–4131. https://doi.org/10.1007/s00521-022-07854-6

[32] Ezugwu, A. E., Agushaka, J. O., Abualigah, L., Mirjalili, S., & Gandomi, A. H. (2022). Prairie dog optimization algorithm. *Neural Computing and Applications*, *34*, 20017–20065. https://doi.org/10.1007/s00521-022-07530-9

[33] Dehghani, M., & Trojovský, P. (2023). Osprey optimization algorithm: A new bio-inspired metaheuristic algorithm for solving engineering optimization problems. *Frontiers in Mechanical Engineering*, *8*, 1126450. https://doi.org/10.3389/fmech.2022.1126450

[34] Yang, J., & Ma, J. (2019). Feed-forward neural network training using sparse representation. *Expert Systems with Applications*, *116*, 255–264. https://doi.org/10.1016/j.eswa.2018.08.038

[35] Siemon, H. P., & Ultsch, A. (1990, July 9–13). *Kohonen networks on transputers: implementation and animation* [Conference presentation]. In: International Neural Network Conference (INNC), Springer, Dordrecht. https://doi.org/10.1007/978-94-009-0643-3_31

[36] Orr, M. (1996). Introduction to radial basis function networks. Technical Report, center for cognitive science. The University of Edinburgh.

[37] Kousik, N., Natarajan, Y., Raja, R. A., Kallam, S., Patan, R., Gandomi, A. H. (2021). Improved salient object detection using hybrid convolution recurrent neural network. *Expert Systems with Applications*, *166*, 114064. https://doi.org/10.1016/j.eswa.2020.114064

[38] Winoto, A. S., Kristianus, M., & Premachandra, C. (2020). Small and slim deep convolutional neural network for mobile device. *IEEE Access*, *8*, 125210-125222. https://doi.org/10.1109/ACCESS. 2020.3005161

[39] Ghosh-Dastidar, S., & Adeli, H. (2009). Spiking neural networks. *International journal of neural systems*, *19*, 295-308. https://doi.org/10.1142/ S0129065709002002

[40] Fekri-Ershad, S. (2020). Bark texture classification using improved local ternary patterns and multilayer neural network. *Expert Systems with Applications*, *158*, 113509. https://doi.org/10.1016/j.eswa.2020.113509

[41] Ren, H., Ma, Z., Lin, W., Wang, S., & Li, W. (2019). Optimal design and size of a desiccant cooling system with onsite energy generation and thermal storage using a multilayer perceptron neural network and a genetic algorithm. *Energy Conversion and Management*, *180*, 598-608. https://doi.org/10.1016/j.enconman.2018.11.020

[42] Aljarah, I., Faris, H., & Mirjalili, S. (2018). Optimizing connection weights in neural networks using the whale optimization algorithm. *Soft Computing*, *22*(1), 1-15. https://doi.org/10.1007/s00500-016-2442-1

[43] Bache, K., & Lichman, M. (2024, December 5). UCI Machine learning repository. http:// archive.ics.uci.edu/ml.