Advances in Mathematical Finance
& Applications, 6(4), (2021), 883-894
DOI: 10.22034/AMFA.2021.1903474.1452

Published by IA University of
Arak, Iran
Homepage: www.amfa.iau-arak.ac.ir

Hybrid Multilayer Perceptron Neural Network with Grey Wolf Optimization for Predicting Stock Market Index

Meysam Doaei[a], Seyed Ahmad Mirzaei[b], Mohammad Rafigh[a]*

[a]*Department of Finance, Esfarayen Branch, Islamic Azad University, Esfarayen, Iran*
[b]*Faculty of Management and Accounting, Aliabad Katoul Branch, Islamic Azad University, Aliabad Katoul, Iran*

ARTICLE INFO

Article history:
Received 02 July 2020
Accepted 31 January 2021

Keywords:
Neural Networks
Metaheuristic Algorithms
Stock Market Forecasting

ABSTRACT

Stock market forecasting is a challenging task for investors and researchers in the financial market due to highly noisy, nonparametric, volatile, complex, non-linear, dynamic and chaotic nature of stock price time series. With the development of computationally intelligent method, it is possible to predict stock price time series more accurately. Artificial neural networks (ANNs) are one of the most promising biologically inspired techniques. ANNs have been widely used to make predictions in various research. The performance of ANNs is very dependent on the learning technique utilized to train the weight and bias vectors. The proposed study aims to predict daily Tehran Exchange Dividend Price Index (TEDPIX) via the hybrid multilayer perceptron (MLP) neural networks and metaheuristic algorithms which consist of genetic algorithm (GA), particle swarm optimization (PSO), black hole (BH), grasshopper optimization algorithm (GOA) and grey wolf optimization (GWO). We have extracted 18 technical indicators based on the daily TEDPIX as input parameters. Therefore, the experimental result shows that grey wolf optimization has superior performance to train MLPs for predicting the stock market in metaheuristic-based.

1 Introduction

The stock market is generally considered very significant in the development of the economy of a country. The successful prediction of a stock's future price serves as a guide for investors in relation to their investments [19]. Due to the variations in policies such as economic policy, government policy, political uncertainty and other policies that affect the stock market, stock market prediction is a difficult task. Stock market data is time series data and it has tremendous noise. Prediction of stock market is a crucial task and a prominent research area in the financial domain as investing in the stock market involves high risk. However, with the development of computationally intelligent methods it is possible to reduce most of the risk [15]. ANNs are widely used for predicting nonlinear, noisy and chaotic kinds of data [8, 9, 14, 20]. ANNs are well-known techniques utilized in learning, approximating and investigating various classes of complex problems [2, 7]. Hence, ANNs are accepted as powerful learning methods for pattern recognition, clustering, classification, and regression problems [2]. MLP is known as the most employed class of ANNs in the literature [3]. The main advantages of MLP are the high learning potential, robustness to noise, nonlinearity, fault tolerance, and high capabilities in generalizing tasks [6]. The performance of ANNs is very dependent on the learning technique utilized to train the weight and bias vectors. Gradient-based and metaheuristic-based algorithms are two ways for training MLPs [18]. The first is Gradient-based approaches such as Levenberg-Marquardt (LM), one step secant (OSS),

* Corresponding author. Tel.: *+985837266501*
E-mail address: *doaei@iauesf.ac.ir*

Gradient descent backpropagation (GD), gradient descent with adaptive learning rate (GDA), and gradient descent with momentum (GDM), Gradient descent with momentum and adaptive learning rate backpropagation (GDX). The second is metaheuristic-based approaches which have been utilized by many researchers to train MLPs [5, 11]. GA is traditional metaheuristic algorithms which has been employed for training MLPs [1, 5]. PSO is another technique motivated by the social life of birds [5, 8, 11]. The GOA is one of the recently population-based optimization techniques inspired by the behaviors of grasshoppers in nature [11]. Gradient-based approaches show good performance in local search while metaheuristic-based ones can reveal a high efficiency in avoiding local optima [2].

In this paper, we investigate the hybridization of MLPs with metaheuristic algorithms which have been developed recently and consist of GA, PSO, BH, GOA, and GWO to predict TEDPIX. The current study has two main contributions. Firstly, it aims to develop a new general framework for the prediction of stock market time series, combining metaheuristic approaches with ANNs. The second is to present a comparative study of the performances of different metaheuristics in order to forecast stock prices based on various well-known technical indicators. Finally, the performances of prediction of the proposed approaches are evaluated according to various metrics. The rest of this paper is organized as follows: a brief literature review of MLP is presented in section two and three. The metaheuristic algorithms are presented in "Metaheuristic algorithms" and then research methodology comes in "Research Methodology" and "Data Specification" is assigned to the data description. The experimental results with their discussion are presented in "Experimental Results". Finally, "Conclusions and Future Directions" concludes the paper.

2 Literation Review

In the literature, many researchers have utilized the metaheuristic algorithms to train MLP networks [11]. For instance, Ghasemieh et al. [8] trained MLPs via metaheuristic algorithms such as cuckoo search (CS), GA and PSO to predict prices on a stock exchange. They concluded that PSO has superior performance in training MLPs. Heidari et al. [11] conducted a comparative study of hybrid MLPs with metaheuristic algorithms such as GA, PSO, artificial bee colony (ABC), flower pollination algorithm (FPA), bat algorithm (BAT), firefly algorithm (FF), monarch buttery optimization (MBO), biogeography-based optimization (BBO), grasshopper optimization algorithm (GOA) and they showed that GOA is beneficial in improving the classification rate of MLPs. In [5] examined a hybrid MPLs with GA and PSO to forecast the movement of the Borsa Istanbul (BIST) 100 index. Based on their result, PSO has superior performance in training MPLs. In [12] proposed an efficient hybrid training technique based on the Ant Lion Optimizer (ALO) to be utilized in dealing with MLPs. ALO is a well-regarded swarm-based metaheuristic inspired by the intelligent hunting tricks of antlions in nature [12]. The efficiency of ALO is compared with GA, PSO, differential evolution (DE), and population-based incremental learning (PBIL) in terms of best, worst, average, and median accuracies. They concluded that the ALO outperforms GA, PBIL, DE, and PSO in classifying the majority of datasets and provides improved accuracy results and convergence rates.

Multilayer Perceptron Neural Networks: The MLP is one of the most widely implemented neural network topologies. MLP network has three layers: input layer, hidden layer, and output layer. The architecture of a MLP with a single hidden layer is presented in Fig. 1.

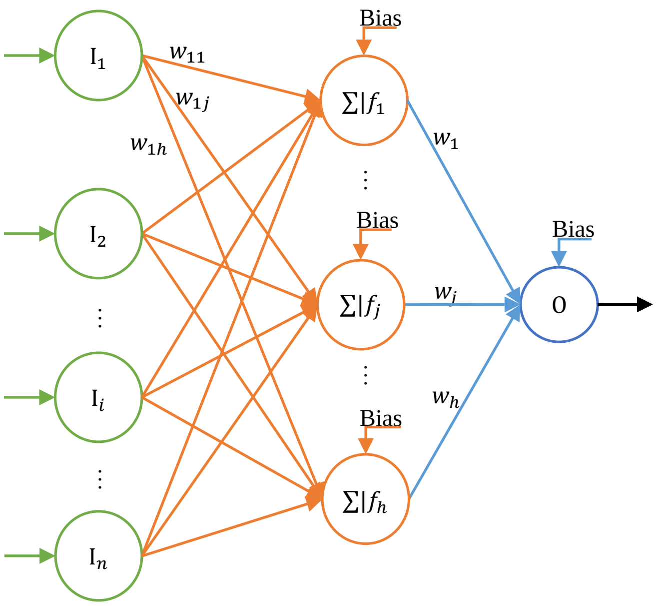

Fig. 1: Structure of MLP Neural Network

The performance of a neuron j can be described based on following equation:

$$y_j = f_j\left(\sum_{i=1}^{n} W_{ij} I_i + b_j\right) \tag{1}$$

Where n is the total number of inputs, I_i is the input variable i, b_j is a bias value, W_{ij} denotes the connection weight, f_j is the activation function, and y_j is the output of the neuron j. various activation functions can be utilized in the MLPs. **Table 1** shows commonly used activation functions in MLPs.

Table 1: Common Activation Functions in MLPs

Transfer functions		Formula
1	Hyperbolic tangent sigmoid transfer function	$tansig(n) = 2/(1+\exp(-2*n)) - 1$
2	Log-sigmoid transfer function	$logsig(n) = 1/(1+\exp(n))$
3	Pure linear transfer function	$purelin(n) = n$
4	Radial basis transfer function	$radbas(n) = \exp((-n)^2)$

Gradient-based and metaheuristic-based algorithms are two ways for training MLPs [18]. Gradient-based has a good performance in local search and metaheuristic-based can reveal a high efficiency in avoiding local optima [2].

4 Metaheuristic Algorithms

In this section, the relevant metaheuristic algorithms are described briefly. GA is presented in the first subsection shortly, then PSO and BH are described. Finally, GOA and GWO are presented respectively.

4.1 Genetic Algorithm

Genetic algorithm is an optimization tool that is inspired by Darwin's theory of natural evolution and proposed by Holland [5, 12, 13]. GA reflects the process of natural selection where the fittest individuals are selected for reproduction in order to produce offspring of the next generation. The procedure of GA is a simulation of the biological evolution behavior. Crossover and mutation are introduced as main GA operators.

4.2 Particle Swarm Optimization

PSO was developed in 1995, by the authors Eberhart and Kennedy [4, 5, 12, 22]. PSO has been applied to numerous areas in optimization. The two main equations in PSO as follows:

$$v_i^{t+1} = wv_i^t + c_1 r_1 \left(xBest_i^t - x_i^t\right) + c_2 r_2 (gBest_i^t - x_i^t) \tag{2}$$
$$x_i^{t+1} = x_i^t + v_i^{t+1} \tag{3}$$

where x_i^t is the position of the *i*th particle and *xBest* and *gBest* denote the best particle position and best group position and the parameters ω, c_1, c_2, r_1 and r_2 are respectively inertia weight, two positive constants and two random parameters within [0, 1]. v_i^t and v_i^{t+1} are the velocities of the *i*th particle at time t and (t+1) in the population, respectively.

4.3 Black Hole

Black hole algorithm was developed in 2013, by Hatamlou [10]. BH is a population-based metaheuristic inspired by the black hole phenomenon. The two main equations in BH algorithm as follows:

$$x_i^{t+1} = x_i^t + r(xBH - x_i^t) \tag{4}$$
$$R = \frac{f_{BH}}{\sum_{i=1}^{N} f_i} \tag{5}$$

Where x_i^t and x_i^{t+1} are the locations of ith star at iterations t and t+1, respectively; xBH is the location of the BH in the search space; r is a random number in the interval [0, 1]; f_{BH} is the fitness value of the BH and f_i is the fitness value of the *i*th star; *N* is the number of stars.

4.4 Grasshopper Optimization Algorithm

Grasshopper optimization algorithm is a new metaheuristic algorithm inspired by the swarming behavior of grasshoppers [21]. The mathematical model employed to simulate the swarming behavior of grasshoppers is presented as follows:

$$X_i = r_1 S_i + r_2 G_i + r_3 A_i \tag{6}$$

Where X_i defines the position of the *i*th grasshopper; S_i is the social interaction; G_i is the gravity force on the *i*th grasshopper, and A_i shows the wind advection. r_1 , r_2 and r_3 are random numbers in the interval [0, 1].

4.5 Grey Wolf Optimization

Grey wolf optimization algorithm is a new metaheuristic optimization technology [16], designed by Mirjalili which is imitating the searching and hunting process of grey wolves. The fittest solution is called the alpha (α), the second best is beta (β), and consequently, the third best is named the delta (δ). The rest of the candidate solutions are all assumed to be omegas (ω). All of the omegas would be guided by these three grey wolves during the searching and hunting. When a prey is found, the iteration begins. Thereafter, the alpha, beta, and the delta wolves would lead the omegas to pursue and eventually encircle the prey. Three coefficients$\vec{A}$, $\vec{C}$ and $\overrightarrow{D}$ are proposed to describe the encircling behavior:

$$\overrightarrow{D_\alpha} = \left|\overrightarrow{C_1}.\overrightarrow{X_\alpha} - \vec{X}(t)\right|, \overrightarrow{D_\beta} = \left|\overrightarrow{C_2}.\overrightarrow{X_\beta} - \vec{X}(t)\right|, \overrightarrow{D_\delta} = \left|\overrightarrow{C_3}.\overrightarrow{X_\delta} - \vec{X}(t)\right| \tag{7}$$

$$\overrightarrow{X_1} = \overrightarrow{X_\alpha} - \overrightarrow{A_1}.\left(\overrightarrow{D_\alpha}\right), \overrightarrow{X_2} = \overrightarrow{X_\beta} - \overrightarrow{A_2}.\left(\overrightarrow{D_\beta}\right), \overrightarrow{X_3} = \overrightarrow{X_\delta} - \overrightarrow{A_3}.\left(\overrightarrow{D_\delta}\right) \tag{8}$$

$$\vec{X}(t) = \frac{\overrightarrow{X_1} + \overrightarrow{X_2} + \overrightarrow{X_3}}{3} \tag{9}$$

Where $\vec{X}$ is the position vector of the grey wolf, and $\overrightarrow{X_1}, \overrightarrow{X_2}$ and $\overrightarrow{X_3}$ are the position vectors of the alpha, beta, and delta wolves. The vectors $\vec{A}$ and $\vec{C}$ are calculated as follows:

$$\vec{A} = 2\vec{a}.\overrightarrow{r_1} - \vec{a} \tag{10}$$

$$\vec{C} = 2.\overrightarrow{r_2} \tag{11}$$

Where components of $\vec{a}$ are linearly decreased from 2 to 0 over the course of iterations and $\overrightarrow{r_1}, \overrightarrow{r_2}$ are random vectors in the interval [0, 1].

5 Research Methodology

In this section, we explain the framework that is undertaken in this paper, which has two main steps. In the first step, we have extracted 18 technical indicators based on TEDPIX as input parameters and contain day open index, day low index, day high index and day close index. In technical analysis, a technical indicator is a mathematical calculation based on historic price, volume, or open interest information that aims to forecast financial market direction [17].

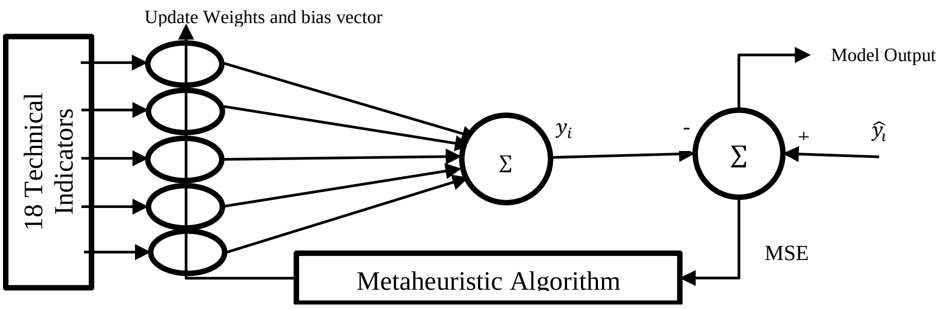

Fig. 2: Hybrid MLP with Metaheuristic Algorithm

5.1 Main Information

The mean squared error (MSE) is employed as fitness function for assessing the fitness of the MLP network. The objective is to minimize the value of the MSE as much as possible. Table 2 shows 18 technical indicators and their formulas used to build data inputs for MLP network. In the second step, metaheuristic algorithms have been employed for training the MLP network. We have used metaheuristic approach to train ANNs that recently developed. The forecasting model with parallel inputs as shown in Fig. 2. The mean squared error (MSE) is employed as fitness function for assessing the fitness of the MLP network. The objective is to minimize the value of the MSE as much as possible.

Table 2: Technical Indicators and its Formulas Used to Build Variables Set

Technical indicators	Calculation	Number of days
Today's close price	C_t	0
Previous close price	C_t	1
Previous highest price	H_t	1
Previous lowest price	L_t	1
Previous open price	O_t	1
simple moving average (SMA)	$(C_t + C_{t-1} + \cdots + C_{t-n+1})/n$	20
exponential moving average (EMA)	$(C_t - EMA(n)_{t-1}) * (2/(n+1)) + EMA(n)_{t-1}$	20
triangular moving average (TMA)	$(SMA_t + SMA_{t-1} + \cdots + SMA_{t-n+1})/n$	20
moving average convergence/ divergence	$EMA(n) - EMA(M)$	[12,26]
Momentum open price	$(O_t/O_{t-1}) * 100$	28
Momentum highest price	$(H_t/H_{t-1}) * 100$	28
Momentum lowest price	$(L_t/L_{t-1}) * 100$	28
Momentum close price	$(C_t/C_{t-1}) * 100$	28
Stochastic	$((C_t - L_t(n))/(H_t(n) - L_t(n))) * 100$	28
Relative strength index (RSI)	$RSI = 100 - 100/(1 + RS)$ $RS = Avg(Gain)/Avg(Loss)$	14
Williams R	$((H_t(n) - C_t)/(H_t(n) - L_t(n))) * -100$	14
Commodity Channel Index (CCI)	$(((H + L + C)/3) - SMA)/(0.015 * MeanDeviation)$	14
Price rate of change (ROC)	$((C_t - C_t(n))/C_t(n)) * 100$	12

Table 3: Structure of MLP Network Used for Proposed Approach

MLP network parameters	Explanation
Number of layers	4
Number of hidden layers	2
Number of neurons	5
Transfer function for hidden layers	tansig
Transfer function for output layer	purelin
Pre-processing	Transfer data into the range [-1 1]
Percentage of training and test set	70 , 30
Number of inputs	18

Table 3 shows the structure of MLP network that used in this work. It is noted that the configurations of this network are derived by trial and error. In this study, the maximum number of iterations is set to be 100 and the number of populations is set be 40 for all metaheuristic algorithms. The specific input parameters of all metaheuristic algorithms are specified in Table 4.

Table 4: Parameter Description

Metaheuristic algorithm	Parameter	Value
Common	Number of Population	40
	Number of Iterations	100
GA	Crossover probability	0.7
	Mutation probability	0.15
PSO	w	0.9
	c_1, c_2	2

To evaluate our proposed model MSE, root mean squared error (RMSE), mean absolute error (MAE), and regression coefficient (R) are computed as follows:

$$\mathrm{MSE} = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2 \tag{12}$$

$$\mathrm{RMSE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2} \tag{13}$$

$$\mathrm{MAE} = \frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_i| \tag{14}$$

$$\mathrm{R} = \frac{\sum_{i=1}^{n}(y_i - \bar{y}_i)(\hat{y}_i - \bar{\hat{y}}_i)}{\sqrt{\sum_{i=1}^{n}(y_i - \bar{y}_i)^2 \sum_{i=1}^{n}(\hat{y}_i - \bar{\hat{y}}_i)^2}} \tag{15}$$

Where y indicates the actual value, $\hat{y}$ shows the predicted one, and n denotes the total number of instances.

5.2 Data Specification

In this paper, TEDPIX is collected from http://www.tsetmc.com. The variables contain day open index, day low index, day high index and day close index from October 2009 to November 2019. We used linear normalization and all data values are normalized in range [-1, +1]. The linear normalization formula in range [a, b] is computed by the following formula:

$$x_{norm} = (b - a) * \left(\frac{x - x_{min}}{x_{max} - x_{min}}\right) + a \tag{16}$$

Where x is variable, x_{max} is its maximum value, and x_{min} is its minimum value.

6 Experimental Results

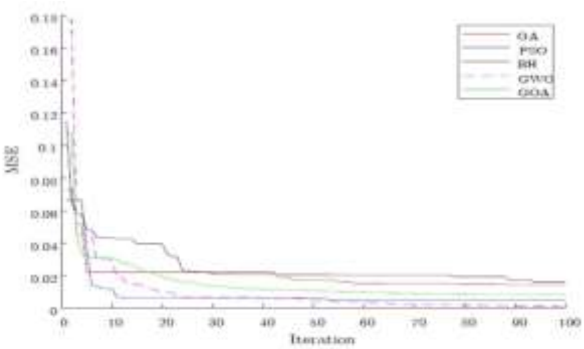

Fig. 3: MSE Convergence Curve of Metaheuristic Algorithms

Fig. 3 illustrates MSE convergence curve of metaheuristic algorithms. As shown in **Fig. 3**, the MSE values gained from various metaheuristic algorithms go down while the iterations of training increase. Among the early iterations, PSO had the best performance and its MSE is calculated to be 0.0062. However, in the recent iterations, GWO has the best performance and its MSE is 0.0011 followed by

PSO and GOA at second and third places, with MSE values of 0.0049 and 0.0081, respectively. The diagrams of performance of the regression coefficient (R) based on various approaches for test data are shown in **Fig. 4**. Based on the results, the GWO algorithm has a far better performance than other approaches. A comparison of the performance of the proposed approaches are presented in **Table 5**. Based on the result the GWO algorithm has a far better performance than other approaches in all data types.

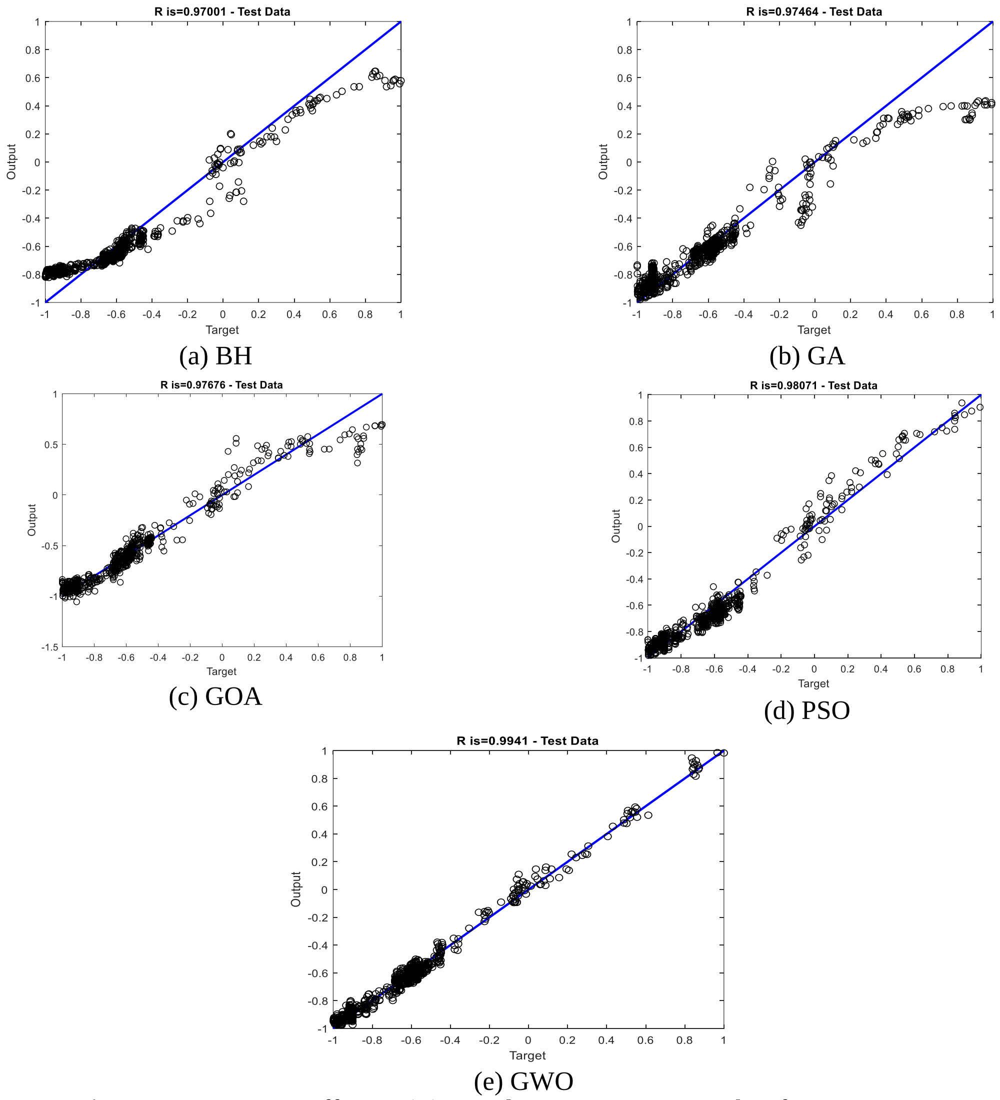

Fig. 4: Regression Coefficient (R) Based on Various Approaches for Test Data

In Table 6, we calculate score of all data types for all metaheuristic algorithms. For example, the first row of the aforementioned table is related to the value and rank of the train data of GA algorithm. The mean of these ranks (10, 10, 10, and 13) is computed as train score of GA algorithm shown in Table 6. Additionally, the final rank is achieved from the final score, computed by the average rank of algorithm in different costs. It reveals that the result obtained by GWO is better than other approaches. Among the other four approaches, PSO goes to rank 2 out of 5 approaches. From Table 6, BH approach has the worst performance compared to other approaches. We also compared GWO results with Gradient-based approaches. In Table 7, we showed results briefly.

Table 5: Summary of MSE, RMSE, MAE, and R for ANN Based on Metaheuristic Algorithms

Metaheuristic Algorithms	Type of Data	MSE		RMSE		MAE		R	
		Value	Rank	Value	Rank	Value	Rank	Value	Rank
GA	Train	0.01418	10	0.11908	10	0.09543	10	0.96174	13
	Test	0.01602	13	0.12658	13	0.10028	13	0.96043	15
	Total	0.01473	11	0.12138	11	0.09689	11	0.96128	14
BH	Train	0.01559	12	0.12488	12	0.09921	12	0.98105	9
	Test	0.0175	15	0.13231	15	0.10356	15	0.98395	7
	Total	0.01616	14	0.12715	14	0.10052	14	0.98197	8
PSO	Train	0.00495	4	0.07037	4	0.05842	4	0.98406	6
	Test	0.00497	6	0.0705	6	0.05864	6	0.9843	4
	Total	0.00495	4	0.07041	5	0.05848	5	0.98413	5
GOA	Train	0.00815	7	0.09029	7	0.06596	7	0.97123	12
	Test	0.00866	9	0.09306	9	0.06657	9	0.97427	10
	Total	0.0083	8	0.09113	8	0.06615	8	0.97223	11
GWO	Train	0.00114	1	0.03386	1	0.02616	1	0.99618	1
	Test	0.00125	3	0.03536	3	0.02642	3	0.99588	3
	Total	0.00117	2	0.03432	2	0.02623	2	0.99609	2

Table 6: The Rank of Algorithms Obtained from Different Data Types

Metaheuristic Algorithms	Train score	Test score	Total score	Final score	Final rank
GA	10.75	13.5	11.75	12	4
BH	11.25	13	12.5	12.25	5
PSO	4.5	5.5	4.75	4.91	2
GOA	8.25	9.25	8.75	8.75	3
GWO	1	3	2	2	1

Table 7: Comparison of of MSE, RMSE, MAE, and R for GWO with Gradient-Based Approaches

Learning Algorithm	Type of Data	MSE		RMSE		MAE		R	
		Value	Rank	Value	Rank	Value	Rank	Value	Rank
GWO	Train	0.00114	10	0.03386	10	0.02616	13	0.99618	10
	Test	0.00125	12	0.03536	12	0.02642	15	0.99588	12
	Total	0.00117	11	0.03432	11	0.02623	14	0.99609	11
GD	Train	0.00658	16	0.08112	16	0.06463	16	0.97753	16
	Test	0.00733	18	0.08565	18	0.06782	18	0.97652	18
	Total	0.0068	17	0.0825	17	0.06559	17	0.97719	17
GDX	Train	0.0004	7	0.02024	7	0.01314	7	0.99861	7
	Test	0.00049	9	0.02218	9	0.0134	9	0.99852	9
	Total	0.00043	8	0.02084	8	0.01322	8	0.99857	8
GDA	Train	0.00131	15	0.03627	15	0.02447	12	0.99581	13
	Test	0.00128	13	0.03589	13	0.02371	10	0.99549	15
	Total	0.0013	14	0.03616	14	0.02424	11	0.99566	14
GDM	Train	0.01198	21	0.10949	21	0.08622	21	0.96044	21
	Test	0.01093	19	0.10456	19	0.08207	19	0.96161	19
	Total	0.01167	20	0.10803	20	0.08498	20	0.96075	20
OSS	Train	0.00005	3	0.00733	4	0.00381	5	0.99982	4
	Test	0.00004	2	0.00669	2	0.0037	3	0.99985	2
	Total	0.000051	5	0.00714	3	0.00378	4	0.99983	3
LM	Train	0.00001	1	0.00329	1	0.00201	1	0.99997	1
	Test	0.00015	6	0.01241	6	0.00384	6	0.99944	6
	Total	0.00005	3	0.00733	4	0.00255	2	0.99982	4

Finally, in Table 8, we calculate score of all data types for GWO and other Gradient-based approaches. From Table 8, we find out LM has the best performance. OSS has the second place and GWO is located in the third place. GDA has the worst performance compared to other approaches.

Table 8: The Rank of GWO with other Gradient-based Approaches

Learning Algorithm	Train score	Test score	Total score	Final score	Final rank
GWO	10.75	12.75	11.75	11.75	3
GD	12.75	11.75	16	13.5	4
GDX	11.75	16	18	15.25	6
GDA	16	18	17	17	7
GDM	18	17	7	14	5
OSS	17	7	9	11	2
LM	7	9	8	8	1

8 Conclusions

In this paper, we investigated the hybridization of MLPs with metaheuristic algorithms which consist of GA, PSO, BH, GOA, and GWO to predict a stock market index. Among all of these optimization algorithms, GWO provides much better performance. We also compared GWO results with Gradient-based approaches and results show that GWO is a capable performance in training MPLs. For future works, the focus will be on the hybridization of MLPs network with other recently developed metaheuristic algorithms to predict a stock market index. In addition, metaheuristic algorithms can be applied to other types of ANNs like long short-term memory (LSTM).

Acknowledgments

We gratefully acknowledge the research center of Islamic Azad University (Esfarayen Branch). In addition, the authors wish to express their gratitude to the reviewers; they found their comments to be very helpful.

References

[1] Alba, E., Chicano, J.F., *Training neural networks with GA hybrid algorithms*, In Genetic and Evolutionary Computation Conference, Springer, Berlin, Heidelberg, 2004, P.852-863. Doi: 10.1007/978-3-540-24854-5_87

[2] Aljarah, I., Faris, H., Mirjalili, S., *Optimizing connection weights in neural networks using the whale optimization algorithm*, Soft Computing, 2018, **22**(1), P.1-5. Doi: 10.1007/s00500-016-2442-1

[3] Chen, J.F., Do, Q.H., Hsieh, H.N., *Training artificial neural networks by a hybrid PSO-CS algorithm*, Algorithms, 2015, **8**(2), P.292-308. Doi: 10.3390/a8020292

[4] Dibachi, H., Behzadi, M.H., Izadikhah, M., *Stochastic Modified MAJ Model for Measuring the Efficiency and Ranking of DMUs*, Indian Journal of Science and Technology, 2015, **8**(8), P.1-7.
Doi: 10.17485/ijst/2015/v8iS8/71505

[5] Eberhart, R., Kennedy, J., *A new optimizer using particle swarm theory*, InMHS'95, Proceedings of the Sixth International Symposium on Micro Machine and Human Science, 1995, P.39-43.
Doi: 10.1109/MHS.1995.494215

[6] Ecer, F., Ardabili, S., Band, S.S., Mosavi, A., *Training Multilayer Perceptron with Genetic Algorithms and Particle Swarm Optimization for Modeling Stock Price Index Prediction*, Entropy, 2020, **22**(11), P.1239.
Doi: 10.3390/e22111239

[7] Faris, H., Aljarah, I., Al-Madi, N., Mirjalili, S., *Optimizing the learning process of feedforward neural networks using lightning search algorithm*, International Journal on Artificial Intelligence Tools, 2016, **25**(06), P.1650033. Doi: 10.1142/S0218213016500330

[8] Faris, H., Aljarah, I., Mirjalili, S., *Training feedforward neural networks using multi-verse optimizer for binary classification problems,* Applied Intelligence, 2016, **45**(2), P.322-32. Doi: 10.1007/s10489-016-0767-1

[9] Ghasemiyeh, R., Moghdani, R., Sana, S.S., *A hybrid artificial neural network with metaheuristic algorithms for predicting stock price,* Cybernetics and Systems, 2017, **48**(4), P.365-92.
Doi: 10.1080/01969722.2017.1285162

[10] Ghasemzadeha, M., Mohammad-Karimi, N., Ansari-Samani, H., *Machine learning algorithms for time series in financial markets,* Advances in Mathematical Finance and Applications, 2020, **5**(4), P.479-490.
Doi: 10.22034/amfa.2020.674946

[11] Hatamlou, A., *Black hole: A new heuristic optimization approach for data clustering*, Information sciences, 2013, **222**, P.175-84. Doi: 10.1016/j.ins.2012.08.023

[12] Heidari, A.A., Faris, H., Aljarah, I., Mirjalili, S., *An efficient hybrid multilayer perceptron neural network with grasshopper optimization*, Soft Computing, 2019, **23**(17), P.7941-58.
Doi: 10.1007/s00500-018-3424-2

[13] Heidari, A.A., Faris, H., Mirjalili, S., Aljarah, I., Mafarja, M., *Ant lion optimizer: theory, literature review, and application in multi-layer perceptron neural networks*, Nature-Inspired Optimizers, 2020, P.23-46.
Doi: 10.1007/978-3-030-12127-3_3

[14] Holland, J. H, *Adaptation in natural and artificial systems,* University of Michigan press. Ann arbor, MI, 1975, **1**(97), 5.

[15] Khodayari, M.A., Yaghobnezhad, A., Khalili Eraghi, K. E., *A Neural-Network Approach to the Modeling of the Impact of Market Volatility on Investment,* Advances in Mathematical Finance and Applications, 2020, **5**(4), P.569-581. Doi: 10.22034/amfa.2020.674953

[16] Kumar, G., Jain, S., Singh, U.P., *Stock Market Forecasting Using Computational Intelligence: A Survey*, Archives of Computational Methods in Engineering, 2020, P.1-33. Doi: 10.1007/s11831-020-09413-5

[17] Mirjalili, S., Mirjalili, S.M., Lewis, A., *Grey wolf optimizer*, Advances in engineering software, 2014, 69, P.46-61. Doi: 10.1016/j.advengsoft.2013.12.007

[18] Murphy, J.J., *Technical analysis of the financial markets: A comprehensive guide to trading methods and applications,* Penguin, 1999.

[19] Ojha, V.K., Abraham, A., Snášel, V., *Metaheuristic design of feedforward neural networks: A review of two decades of research,* Engineering Applications of Artificial Intelligence, 2017, 60, P.97-116.
Doi: 10.1016/j.engappai.2017.01.013

[20] Pillay, B.J., Ezugwu, A.E., *Metaheuristics optimized feedforward neural networks for efficient stock price prediction*, arXiv preprint arXiv, 2019, 1906.10121.

[21] Rezaei, N., Javaheri, M., *The Predictability Power of Neural Network and Genetic Algorithm from Firms' Financial crisis,* Advances in Mathematical Finance and Applications, 2020, **5**(2), P.1-17.
Doi: 10.22034/amfa.2019.1863963.1195

[22] Saremi, S., Mirjalili, S., Lewis, A., *Grasshopper optimisation algorithm: theory and application*, Advances in Engineering Software, 2017,**105**, P.30-47. Doi: 10.1016/j.advengsoft.2017.01.004

[23] Tone, K., Toloo, M., Izadikhah, M., *A modified slacks-based measure of efficiency in data envelopment analysis*, European Journal of Operational Research, 2020, **287** (2), P. 560-571, Doi: 10.1016/j.ejor.2020.04.019.

[24] Thakkar, A., Chaudhari, K., *A Comprehensive Survey on Portfolio Optimization, Stock Price and Trend Prediction Using Particle Swarm Optimization*, Archives of Computational Methods in Engineering, 2020, P.1-32. Doi: 10.1007/s11831-020-09448-8