

Optimization of Artificial Neural Network for Stock Market Price Prediction Using an Enhanced Firefly Algorithm

¹Ameen Ahmed Oloduowo, ²Fashoto Stephen Gbenga, ¹Ogeh Clement,
¹Balogun Abdullateef and ¹Mashwama Petros

¹Department of Computer Science, University of Ilorin, Kwara State, Nigeria

²Department of Computer Science, University of Swaziland, Kwaluseni, Swaziland

Abstract: The aim of this study is to develop a system that predicts the future closing price of daily stocks based on historic data of the stocks. The Artificial Neural Network (ANN) was employed to learn the historic data and make predictions for the next few days while the Firefly Algorithm (FA) was used to optimize the weights of the network for accurate predictions. The daily historic dataset of stock prices for five companies trading the New York Stock Exchange (NYSE) from 13-14-10-2005 was used for the experiment. This study shows that the proposed model made 6.8% better predictions with less average errors when compared with the predictions made with ANN trained with the Genetic Algorithm (GA). The introduction of the reduction scheme employed reduces the randomization parameter of the fireflies in the algorithm and this significantly improves the prediction accuracy of the proposed model.

Key words: Artificial neural network, firefly, optimization, price prediction, stock market, model

INTRODUCTION

The growth of information technology has seen the application of computer science theories and principles in so, many human endeavors. The stock market is not an exception to the great advances in technology as in recent times, computer system have been designed to capture real time trade of the stock market and statistically analyses stock data in order to make future prediction and classification of stocks. Traders, analysts and investors are always in search for techniques to better predict stock price. According to Ting *et al.* (2006), investors and management of companies can apply their creativity and judgment to make better decisions in order to maximize profit, if properly armed with better information.

Data mining is known for the extraction of implicit, previously unknown and useful information in data, hence, stock data mining is possible using a variety of data mining algorithm. Information derive from mining the stock market can help investor or want to be investors to invest in stocks that would yield dividend within the shortest possible time. It can also help the management of a company to make decisions that can increase the growth of the company. However, the mining of data in the stock market is not a trivial issue as the size of data is enormous with so, many variables.

Mining of the stock market involves the discovery of patterns and predictions of stock market prices. Today, most stock market trades are executed electronically which is due to the growth of information technology. The stocks themselves are now in electronic form and not in physical certificates form. This makes it easy for stock data to be collected for the purpose of technical and systematic analysis which in turn has also given rise to a situation where very large amount of data are generated in a short amount of time. The nature and the size of data generated in the stock market poses problems to knowledge discovery. Hence, the difficulty in mining the stock market as the system is noisy, nonlinear and unstable with so, many factors influencing the trend in the market (Das and Uddin, 2013; Kumbhar and Arigiddi, 2015; Mahajan and Kulkarni, 2013). Some of these factors are naturally (earthquake, flood), politically (political systems, election), socially (news) and economically (inflation, interest rates, foreign exchange demand and supply) motivated. It is due to the uncertainties in the stock market that there is the need to mine stock data generated in the stock market with appropriate computer models/algorithms using selected data variables for the purpose of forecasting and increasing revenue generation.

The neural network has been widely applied in various fields for the purpose of classification and predictions. It has its second highest application of 25.4% in the field of

finance which includes the stock market (Das and Uddin, 2013). This is significantly due to the fault tolerant nature of the artificial neural network. Recently, researchers have shown increased interest on the need to improve the performance of the artificial neural network. This need was further emphasized in a recent methodological review by Das and Uddin (2013) that the performance of the artificial neural network can be improved if hybridized with other models. An illustration is the hybrid neuro-genetic algorithm for stock prediction (Brajevic and Tuba, 2013). Which combined the genetic algorithm and artificial neural network to improve the prediction performance of the neural network. The genetic algorithm was used to train the artificial neural network and the model was later used for stock price prediction.

Data mining techniques are used to discover hidden pattern and to predict future trends in the stock market, so as to increase revenue and decision making capability of investors. Some of these techniques are highly dependent on the study of historic data of a company which makes it difficult to make accurate predictions due to the volume of the data used.

This study proposes to apply the firefly algorithm, a nature inspired algorithm to train and optimize the weight of an Artificial Neural Network (ANN) in order to improve the prediction capability of the Artificial Neural Network (ANN). It also adopts a randomization reduction scheme in the firefly algorithm to reduce the errors in the prediction of stock prices. The firefly algorithm has the ability of covering a large search space due to the random movements of fireflies which in turn enables quick convergence of the algorithm. The firefly algorithm is a meta-heuristic optimization algorithm that has outperformed algorithms like the Genetic Algorithm (GA) and the Particle Swarm Optimization (PSO) algorithm in some optimization problem domain (Alweshah, 2014), hence, its adoption for this study. The proposed model has been successfully implemented on time series problems and since, the stock market is a time series problem domain, the model can help investors to better predict the stock market prices when properly implemented.

Literature review: In a similar study, used the multilayered perceptron trained by the back propagation algorithm to predict the stock price of fourteen companies listed under LIX15 of the National Stock Exchange of India (NSE) for 5 years. The result indicated a 51.06% accuracy suggesting a room for improvement in the proposed model. Neenwi *et al.* (2013) also used the multilayered perceptron trained by the back propagation algorithm to predict the stock price of three companies

trading in the Nigerian stock exchange and had the following result: access bank plc. 50%, first bank: 83.3% and UBA 83.33% prediction accuracy but it was indicated that the amount of data used were inadequate which may have affected the result coupled with the drawbacks of the learning algorithm. Different studies suggested that the weights of the algorithm can be further optimized, hence, the use of fast and easy converging nature inspired optimization algorithms.

By Kumbhar and Arigiddi (2015) the neuro-genetic data mining method was proposed to predict stock prices. The algorithm combined the recurrent multilayered perceptron neural network and the genetic algorithm to form a hybrid model. The genetic algorithm was used to optimize the weight of the neural network while training and the trained neural network was used to predict the future trend of the stock market. The study used a ten year dataset of stock market price indices (from 01-01-2004-2014) which was divided into the training and testing dataset. The minimum mean square error was used as the stopping criterion. Hence, reduction in prediction errors was not adequately dealt with the model proved to be better in prediction when compared with the artificial neural network trained by the back propagation algorithm but the genetic algorithm has its drawbacks which includes getting stuck at local optimal if not properly, handled and longtime of convergence (Thomas, 2016).

According to Maharana *et al.* (2015) and Nhu *et al.* (2013), the fuzzy logic, neural network and the firefly algorithm are combined to form a hybrid intelligent artificial neural network model for stock market index prediction. The fuzzy logic was used to generate rules that explain the relationship between outputs and inputs of the neural network while the firefly algorithm trained the neural network for stock price predictions. The model showed better performance when compared with other models like the artificial neural network. The model used the Nokia data set with four parameter of the data set placed under consideration. Recommendations were made for a larger data set as the data set used may have affected the efficiency of the model.

MATERIALS AND METHODS

Overview of the method: This study seeks to use a hybridization algorithm which comprise of an artificial neural network trained by the firefly algorithm in order to predict the closing price of stocks in the stock market. It also includes the use of a randomization reduction scheme to improve the predictive ability of the model. The proposed method is in two stages:

- The training stage
- The testing stage

The training stage is further divided into two stages which are the creation of the artificial neural network and the training of the neural network with the historic data of stock prices. The weight of the neural network is updated by the firefly algorithm in order to predict stock prices with minimum error. The dataset is divided into three parts. The training dataset for training, the validation dataset used to prevent overfitting of the neural network and the testing dataset for testing the model. A randomization reduction scheme is also applied in the firefly algorithm, so, as to gradually reduce the random movements of the fireflies in a geometric progression in order to improve stock price predictions. If the desired price output is found, then the firefly algorithm stops else, it continues until the maximum iteration is reached. Once the solution is obtained the neural network saves the weight and uses it for future prediction which is the second stage of the process. In the testing stage, the optimized weights obtained in the training stage are set to the neural network and tested for the desired output which is predicting the future price of stocks using the testing data.

Dataset: A ten years daily historic stock price data from 13-10-2005-2015 of five airline companies trading on the New York Stock Exchange (NYSE) was used for the training, validating and testing the model. The names of the companies are Alaska Air Group Inc. (Alk), Southwest Airline Co (Luv), LATAM Airlines Group SA (Lfl), Gol Linhas Aereas Inteligents SA (Gol) and China Southern Airline Co ltd (Zhn).

The data were collected from the Google finance website and they were divided in this order. About 80% as the training dataset and 20% as the testing data. The parameters were selected due to availability and their regular use in stock prediction studies.

The artificial neural network: The artificial neural network used in this study is the feed-forward neural network. It has majorly three layers of nodes with architecture [4, 9, 1] which are:

The input layer: The nodes in the input layer accept the input signals (stock data) from the environment (the stock market). The inputs include the closing price of at least 4 previous days. In other words, the input layer has four nodes. The nodes of the input layer can be increased but it will in turn, affect the architecture (complexity) of the

neural network by increasing the number of nodes in the hidden layer which may cause poor generalization and performance of the neural network a situation known as overfitting.

The hidden layer: The node of the hidden layer which is in between the input and the output layer does the processing and it can have more than one layer of nodes. Although, the multilayer neural network has the capability of approximating any non-linear and continuous function there are no theories that stipulate the number of hidden layer or nodes of the hidden layer the neural network needs to have for optimal performance Brabazon and O'Neill (2006). In other to prevent overfitting, it is advisable to use more training dataset. The calculation in the hidden layer is express mathematically in Eq. 1 as follows:

$$H = \sum x_i w_i + b \tag{1}$$

Where:

- X_i = A vector of inputs. That is $X_i = x_1, x_2, \dots, x_n$
- W_i = The vector of weights. That is $W_i = w_1, w_2, \dots, w_n$
- b = Bias, the bias is usually multiplied by an input of 1. Hence, the bias does not change

The weighted sum of input is passed into an activation function in this case the sigmoid function which transforms the input signal into an output signal.

The output layer: The node (s) output layer presents the outputs of the whole process. The output of the hidden layer are sent to the output layer as inputs and in turn the weighted sum is calculated and passed to the activation function, the process is mathematically explained in Eq. 2:

$$O = \sum H_i W_i + b \tag{2}$$

Where:

- H = The output from the hidden layer
- W = The vector of weights in the output layer
- b = Bias

The input to the Artificial Neural Network (ANN) for this study is the closing price of stock for some number of days as specified by the user hence the structure of the Artificial Neural Network (ANN) is determined by the user but the hidden layer has nine nodes and the output layer has one node. The training algorithm (in this case the firefly algorithm) initializes a random population of weights (w) (Fireflies) normally from the close interval [0, 1] for each input parameter and the bias. Hence, given n input parameters, the neural network would have $n+1$

weight. The weights (w) of each input is multiplied with the inputs and passed into the activation function which transforms it into a set of input signals for other nodes of the neural network. The output from the neural network is the closing price which is later compared to the target closing price for accurate prediction or to determine the error in the predictions. The weights are then adjusted based on the errors.

As opined by Brabazon and O'Neill (2006), the multilayered neural network if properly implemented has a high chance of overtraining occurring, this is a situation where by the model does not generalise well and memorising disturbance (noise) in the data. Overtraining leads to poor performance of data mining models. In a bid to reduce the chances of the occurrence of overtraining, the early stopping method was applied in the implementation of the model. The early stopping method detects on time the increase in error which is referred to as the start of overfitting (Teodorescu *et al.*, 1998).

Formation of the firefly algorithm: The two important issues in the formation of the firefly algorithm is the variation of the light Intensity (I) and the attractiveness (β) of the fireflies. It is assumed that the attractiveness (β) of a firefly is determined by its brightness which is related to the objective function.

The brightness of a firefly which is the same as its light Intensity (I) at a certain location x is directly proportional to the objective function $f_{(x)}$ in Eq. 3:

$$I_{(x)} = f_{(x)} \tag{3}$$

The attractiveness β or light intensity I of a firefly i as seen by another firefly j at distance r_{ij} , reduces with distance from the light source and it varies with a degree of absorption γ . From the inverse square law, the light intensity $I(r)$ can be represented in relation to distance (r) as defined in Eq. 4:

$$I(r) = \frac{I_s}{r^2} \tag{4}$$

Where:

I_s = Light intensity at the source

R = Distance

For a fixed absorption coefficient γ , the light Intensity I varies with distance r in Eq. 5: (Neenwi *et al.*, 2013)

$$I(r) = I_0 e^{-\gamma r} \tag{5}$$

Where:

I_0 = Original light intensity. In order to prevent singularity

$r = 0$, the effect of distance r and the absorption coefficient

γ = Combined using the Gaussian form in Eq. 6

$$I(r) = I_0 e^{-\gamma r^2} \tag{6}$$

The function may sometimes be needed to decrease at a slow rate monotonically to improve the algorithm (Neenwi *et al.*, 2013). Hence, the light intensity can also be represented as in Eq. 7:

$$I(r) = \frac{1}{1 + \gamma r^2} \tag{7}$$

For the purpose of this study, the algorithm starts by generating randomly an initial population. The quality/fitness of each individual in the population is then calculated by using the attractiveness function and the best solution is obtained (Alweshah, 2014). The attractiveness function is shown in Eq. 8:

$$\beta(r) = \beta_0 e^{-\gamma r^2} \tag{8}$$

Which can also be represented in Eq. 9:

$$\beta(r) = \frac{1}{1 + \gamma r^2} \tag{9}$$

Where:

r = Distance between two fireflies

β_0 = Initial attractiveness

$r = 0$ (which is the original attractiveness)

γ = The absorption coefficient which determines the decrease of the light intensity

The Cartesian or Euclidean distance is used to calculate the distance between two fireflies. If there are two fireflies i and j at position x_i and x_j then the distance between the two fireflies is shown in Eq. 10:

$$r_{ij} = \|x_i - x_j\| = \sqrt{\sum_{k=1}^d (x_{i,k} - x_{j,k})^2} \tag{10}$$

where d = The problem domain dimensionality, hence, in a two dimensional domain, the distance is represented in Eq. 11:

$$r_{ij} = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2} \tag{11}$$

The movement of a firefly i attracted by another firefly j is represented thus:

$$x_i = x_i + \beta_0 e^{-\gamma r_{ij}^2} (x_j - x_i) + \alpha \left(\text{rand} - \frac{1}{2} \right) \tag{12}$$

In Eq. 12, the first term is the current position of the firefly, the second term is responsible for the attraction of a firefly toward the light intensity of a neighboring firefly. The third term is used for the random movement of the firefly when there are no brighter fireflies. The parameter α is a randomization parameter which is problem determined and rand is a random number generator distributed uniformly in the closed interval [0, 1] (Yang, 2008). If there are no brighter fireflies, the best individual in the population will be moved randomly by Eq. 13:

$$x_i = x_i + \alpha \left(\text{rand} - \frac{1}{2} \right) \tag{13}$$

As proposed by Yang (2009) the solutions of the firefly algorithm can be improved by the gradual geometric reduction of the randomization parameter α . The convergence of the firefly algorithm can also be improved if the randomization parameter α is varied, so that, it decreases gradually as optimal solution is approaching (Costa *et al.*, 2014). Hence, the adoption of a reduction scheme for this study as used in (Brajevic and Tuba, 2013). The reduction scheme experimented is mathematically represented in Eq. 14:

$$\alpha = \alpha_0 (x)^t \tag{14}$$

where t is the current number of iteration, α_0 is the initial randomization parameter which is usually 1 and x is a real number in the in the range $0 < x < 1$.

The algorithm: The optimization process of the firefly algorithm greatly depends on the brightness (light intensity) of the fireflies and the random movement of fireflies towards brighter counterparts. Each firefly is attracted to other fireflies depending on the brightness which is due to the first assumption in the design of the algorithm that all artificial fireflies are unisex. The pseudocode of the algorithm as illustrated in (Yang, 2008) is given below:

Define and initialize the function $f(x_i)$, $x = (x_1, \dots, x_n)$; Generate initial firefly population x_i ($i = 1, 2, \dots, n$); Determine light intensity for each firefly x_i by calculating $f(x)$; Define light absorption coefficient γ .

Firefly algorithm:

```

While t < Maximum Generation
Make a copy of the generated firefly population for move
function
For i = 1: n all n fireflies
For j = 1: i all n fireflies
If (Ij > Ii)
Move fireflies i and j according to attractiveness
Evaluating new solutions and updating light intensity for
next iteration
End if
End for j
End for i
Sorting the fireflies to find the best
End while
    
```

Begin post process on best result obtained. The algorithm starts by initializing a population of fireflies and every firefly is distinct. The distinction in the fireflies is the light intensity of each firefly. The brightness determines the movement of the fireflies in the swarm.

During the iterative process, the brightness of one firefly is compared with others in the swarm. The difference in brightness result to the movement of the fireflies. The distance travelled by the fireflies depends on the attractiveness between the fireflies. The best solution is continuously updated during the iterative process and it continues until the stopping conditions are met or satisfied. After the iterative process ends, the best solution is determined and the post process is initiated to obtain the result. The parameter and definition of the notation for the firefly algorithm shows in Table 1.

Objective function for the proposed model: In this study, the aim of the Firefly Algorithm (FA) is to optimize the weight of the Artificial Neural Network (ANN) in order for the Artificial Neural Network (ANN) to predict stock prices with minimum or zero error. In other words, the lesser the error, the greater the light intensity of the firefly. It can be mathematically expressed as in Eq. 15:

$$I \propto \frac{1}{E} \tag{15}$$

Table 1: Parameter and definition of notation for the firefly algorithm

Parameters	Notations in algorithm
Brightness	Objective or fitness function
Beta (β)	Attractiveness
Alpha (α)	Randomness parameter
Gamma (γ)	Absorption coefficient
Number of generations (epochs)	Iterations
Number of fireflies	Population to be initialized
Dimensions (d)	Problem dimension
R	Radius, time interval (application dependent)

where I is the light intensity and E is the error. Most predictive models use the Mean Square Error (MSE) or the Mean Absolute Percentage Error (MAPE) but in this study the Root Mean Square Error (RMSE) was used to measure the performance of the proposed model. That is, the Root Mean Square Error (RMSE) was used as the fitness function which also calculates the error in the prediction. Mathematically the root mean square error is expressed in Eq. 16:

$$F_{(x)} = RMSE = \sqrt{\frac{1}{n} \sum^n (Y_i - T_i)^2} \quad (16)$$

Where:

- N = Number of forecast periods
- Y_i and T_i = The predicted and target closing price, respectively

The Artificial Neural Network (ANN) trained with the Firefly Algorithm (FA) for stock market price prediction: The proposed hybrid model used the Firefly Algorithm (FA) to train the weights of an Artificial Neural Network (ANN) which is represented with the acronym FA-ANN. The model is an interactive model in the sense that the user is allowed to alter the structure of the Multilayer Perceptron (MLP) used for the prediction by specifying the number of input nodes used for the prediction. The hidden/middle layer has nine nodes and the output layer has one node.

In this study, the previous day's historic closing price for a company is used as input for the input node of the Artificial Neural Network (ANN). This is because the study intends to predict the closing price of stock for a particular day based on previous closing price of stocks for a number of days as specified by the user. In other words, if the nodes of the Artificial Neural Network (ANN) are four, it means the closing prices of 4 previous days will be used to predict the closing price for the 5th day; This view is as supported by 19 in a study of the use of an adaptive neuro-fuzzy system for stock prediction. The model predicted the closing price of day x (t+1) using the series of closing prices for days x (t), x (t-1), x (t-2) and x (t-3).

Aside the nodes of the artificial neural network there are other parameters to be specified by the user such as the number of fireflies, number of iterations (also known as epochs) and the file name of the dataset to be used in training and testing phases of the experiments. However, precautions should be taken in the setting up of the parameter due to the fact that the problem is a non-deterministic polynomial time (NP hard) problem as an increase in the number of parameter will increase exponentially the running time of the algorithm.

Table 2: Parameter settings for the FA-ANN

Notations	Symbols
Population of fireflies	N
Attractiveness	β_0
Absorption coefficient	Γ
Randomization parameter	α_0
Random number generator	rand
Number of iterations (Epoch)	T
Dimension (number of nodes)	D

Table 3: Parameter settings for the GA ANN

Notations	Values
Populations size	50
Single point cross over rate	0.08
Uniform mutation rate	0.05
Generation gap	0.9
Iteration	10,000

The ANN-FA starts by asking the user to specify the parameters for the algorithm. After the parameters are set, the training start automatically and as the training is going on there is a progress report showing on a console of the current set of weights and errors with each iteration. The testing phase starts when the training phase is done and the results are presented in a CSV format. The system does not use the Graphic User Interface (GUI) as a mode of interaction. It rather uses the console. This is due to the consideration of the loading time of the algorithm, as Graphic User Interface (GUI) libraries may increase the loading time of the algorithm.

Parameters settings for the FA-ANN experiments: The initialization of the notations in the FA-ANN shows in Table 2. Parameters settings for the GA ANN experiments. The initialization of the notations in the GA ANN shows in Table 3.

Summary of data used for the experiments: The summary of the dataset of the 5 company used for the experiments is as shown in Table 4:

Result of the experiments: The performance of the experiments was calculated using the objective function in this case, the Root Mean Square Error (RMSE). In the first stage of the experiments which is the training phase, the results are a set of optimized weights and error values with iteration intervals. The best set of weights are selected and used in the second stage of the experiment for training. The result from the second stage of the experiment, the testing stage of the model is the predicted closing price as presented on from Table 5-10 and Fig. 1-10.

The randomization reduction scheme on the FA-ANN prediction results. Table 11 shows the effect of the reduction scheme of the randomization parameter of the

Table 4: Summary of the dataset used for experiments

Name	Period		Number of records		
	From	To	Training	Validation	Testing
Alaska Air Group (ALK)	13-Oct-05	14-Oct-15	1511	504	504
Southwest Airline (LUV)	17-Oct-05	13-Oct-15	1509	502	502
LATAM Airlines (LFL)	14-Oct-05	13-Oct-15	1509	502	502
Gol Linhas Aereas (Gol)	14-Oct-05	13-Oct-15	1510	503	503
China Southern (ZHN)	20-Oct-05	13-Oct-15	1505	504	504

Table 5: The result for ALK company

ALK

t0	t1	t2	t3	FA-ANN predicted	GA-ANN predicted	Actual price	FA-ANN error	GA-ANN error
18.28	18.41	18.55	18.73	18.89	18.81	18.88	0.01	0.07
19.08	18.88	19.04	19.11	19.28	19.25	19.18	0.01	0.07
19.08	19.14	19.12	19.05	19.65	19.57	19.52	0.13	0.05
20.06	20.73	20.22	20.58	20.74	20.69	20.34	0.04	0.35
20.67	20.66	20.26	20.44	20.62	20.64	20.36	0.26	0.28
20.76	20.91	20.09	21.24	21.41	21.33	21.24	0.17	0.09
21.14	21.28	21.03	21.38	21.57	21.53	21.25	0.32	0.28
21.03	21.24	21.26	21.25	21.45	21.44	21.28	0.17	0.16
21.72	21.56	21.67	21.58	21.79	21.79	21.99	0.02	0.02
22.18	22.04	22.46	22.21	22.44	22.44	22.08	0.36	0.36
21.74	21.63	21.36	21.54	21.73	21.74	21.97	0.24	0.23

Results for ALK company

Table 6: The result for Gol company

Gol

t0	t1	t2	t3	FA-ANN predicted	GA-ANN predicted	Actual	FA-ANN error	GA-ANN error
5.29	5.16	5.34	5.35	5.35	5.36	5.25	0.01	0.11
5.42	5.37	5.11	5.17	5.17	5.16	4.98	0.19	0.18
4.99	4.94	4.85	5.21	5.21	5.01	5.32	0.11	0.31
5.24	5.15	5.09	4.97	4.97	5.06	4.84	0.13	0.22
4.86	4.83	4.63	4.63	4.63	4.65	4.77	0.14	0.12
4.54	4.69	4.53	4.58	4.58	4.57	4.66	0.08	0.09
4.85	4.81	4.82	4.81	4.81	4.83	4.87	0.06	0.04
4.92	5.08	5.01	4.99	4.99	5.07	5	0.01	0.07
4.83	5.16	5.17	5.23	5.22	5.21	5.51	0.29	0.03
5.61	5.43	5.83	6.36	6.35	6.05	6.24	0.11	0.19
6.29	6.56	6.42	6.46	6.46	6.46	6.36	0.01	0.01
6.41	6.05	6.58	6.68	6.67	6.64	6.85	0.18	0.21
6.83	6.91	7.09	7.25	7.24	7.17	7.25	0.01	0.08

Results for Gol company

Table 7: The result for LFL company

LFL

t0	t1	t2	t3	FA-ANN predicted	GA-ANN predicted	Actual	FA-ANN error	GA-ANN error
25.43	25.74	25.75	26.05	26.08	25.72	26.01	0.02	0.38
25.87	25.44	25.35	24.58	24.61	25.53	24.07	0.09	0.83
24.72	24.76	24.79	24.84	24.87	24.81	24.06	0.27	0.21
24.07	24.87	24.89	24.78	24.81	24.84	24.51	0.03	0.33
24.27	23.78	23.31	23.03	23.32	23.09	22.96	0.36	0.94
23.17	23.01	22.09	22.99	23.01	23.11	22.08	0.21	0.31
22.48	22.01	22.45	22.15	22.17	22.33	22.18	0.01	0.15
22.18	22.27	22.44	22.36	22.38	22.31	22.37	0.01	0.06
22.69	23.04	23.03	23.37	23.39	23.16	23.35	0.04	0.19
23.33	23.38	23.36	23.05	23.07	23.36	23.26	0.19	0.01
23.41	23.05	23.39	23.56	23.58	23.51	23.44	0.14	0.07
23.45	23.84	23.89	23.73	23.75	23.73	24.05	0.03	0.32
24.07	24.13	24.17	24.19	24.21	24.17	24.57	0.36	0.04

Result for LFL company

Table 8: The result for LUV company

LUV									
t0	t1	t2	t3	FA-ANN predicted	GA-ANN predicted	Actual	FA-ANN error	GA-ANN error	
8.84	8.89	8.88	8.95	8.95	8.96	8.98	0.03	0.02	
8.88	8.83	8.79	8.75	8.08	8.08	8.83	0.03	0.03	
8.74	8.82	9.05	9.03	9.01	9.01	9.04	0.03	0.03	
9.15	9.01	9.18	9.06	9.01	9.09	9.07	0.03	0.02	
9.03	8.76	8.84	8.93	8.96	8.95	9.01	0.14	0.15	
9.11	9.23	9.37	9.04	9.38	9.38	9.29	0.09	0.09	
9.43	9.05	9.53	9.44	9.47	9.47	9.63	0.16	0.16	
9.79	9.88	9.81	9.84	9.86	9.87	10.12	0.26	0.25	
10.02	10.14	10.19	10.36	10.32	10.33	10.05	0.18	0.17	
10.56	10.54	10.46	10.43	10.48	10.48	10.31	0.17	0.17	
10.26	10.16	10.24	10.47	10.44	10.44	10.72	0.28	0.28	
10.88	10.91	11.03	11	11.01	11.01	11.07	0.06	0.06	
11.05	11.02	11.08	11.28	11.26	11.26	11.04	0.14	0.14	
11.26	11.48	11.36	11.45	11.45	11.46	11.44	0.01	0.02	

Results for LUV company

Table 9: The result table for ZNH company

ZNH									
t0	t1	t2	t3	FA-ANN predicted	GA-ANN predicted	Actual	FA-ANN error	GA-ANN error	
23.88	24.38	24.22	24.44	24.36	24.33	23.93	0.43	0.04	
23.35	23.28	23.15	23.99	23.62	23.42	24.02	0.04	0.06	
23.46	23.23	24	23.95	23.75	23.61	24.24	0.49	0.63	
23.64	23.07	22.32	22.05	22.68	22.07	22.05	0.18	0.02	
22.05	21.73	21.04	21.83	21.74	21.65	22.45	0.71	0.08	
21.91	22.08	22.84	22.61	22.48	22.41	22.15	0.33	0.26	
22.35	22.11	22.16	22.17	22.17	22.12	22.08	0.09	0.04	
22.75	22.88	23.26	24.08	23.54	23.29	23.44	0.01	0.15	
24.14	24.05	24.46	25.25	24.73	24.47	25.02	0.47	0.73	
25.49	26.07	25.75	25.07	25.49	25.69	24.97	0.52	0.72	
24.91	24.79	25.85	26.07	25.63	25.04	25.37	0.26	0.03	
25.35	25.72	25.99	26.52	26.15	25.99	28.47	2.32	2.48	
28.11	28.21	28.44	28.51	28.04	28.33	28.18	0.22	0.15	
28.83	29.48	29.01	29.24	29.26	29.29	29.56	0.03	0.27	

Result table for ZNH company

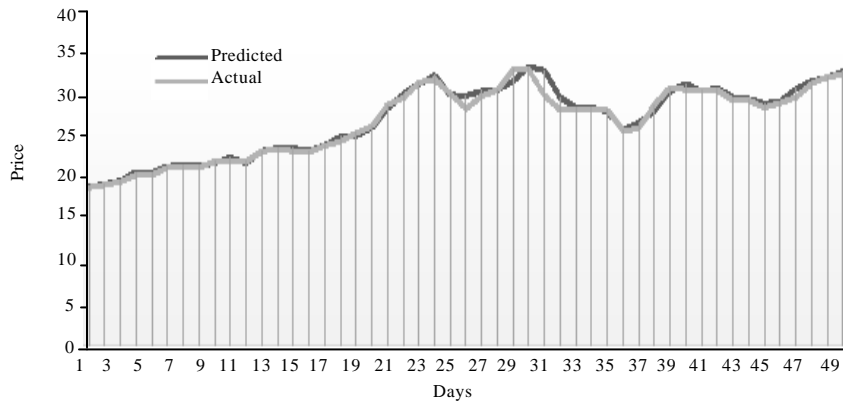


Fig. 1: Line chart comparing the FA-ANN predicted and actual closing price for ALK company

FA on the prediction result of the FA-ANN. The randomization parameter was reduced gradually by 0.01. The effect on prediction is evaluated by the average errors of the total prediction for each step of the reduction scheme.

From Table 11 the best results were derived where the variable x in the scheme is at 0.5 for companies ALK and LUV, 0.9 for Gol and LFL and 0.1 for ZNH. From the experiments, at the various best average errors, the FA-ANN showed significant improvements when the iteration was increased.

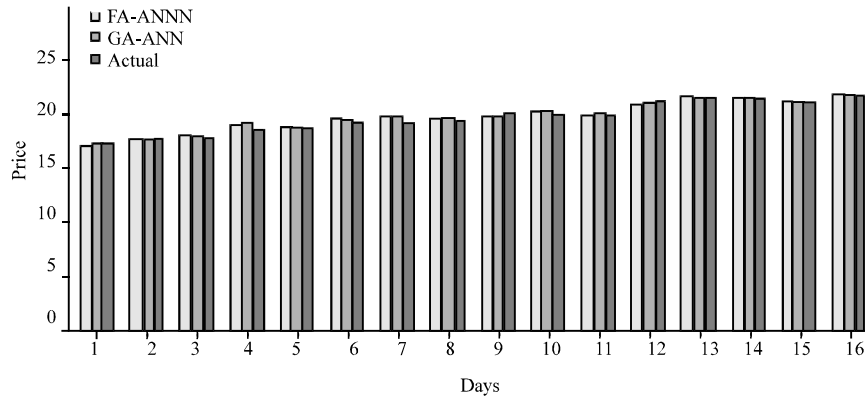


Fig. 2: Bar chart comparing the FA-ANN and GA-ANN predicted prices with the actual price for ALK

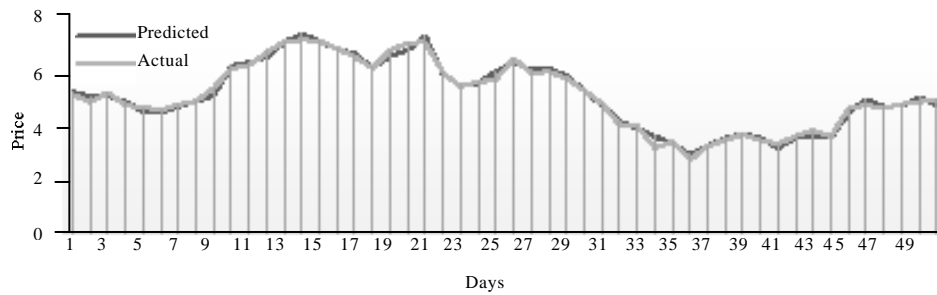


Fig. 3: Line chart comparing the FA-ANN predicted and actual closing price for Gol company

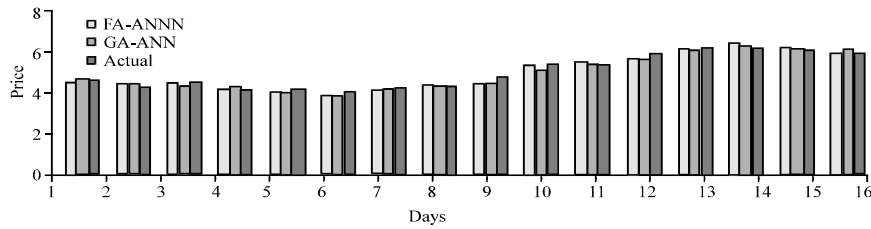


Fig. 4: Bar chart comparing the FA-ANN and GA-ANN predicted prices with the actual price for Gol

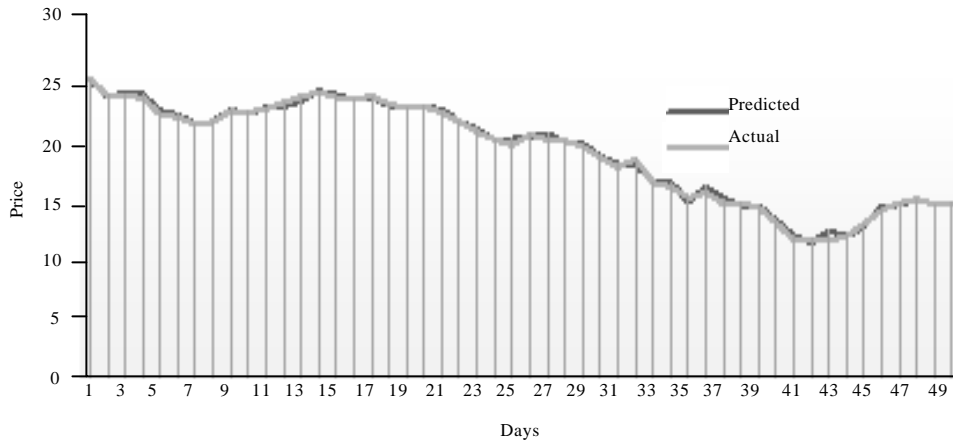


Fig. 5: Line chart comparing the FA-ANN predicted and actual closing price for LFL company

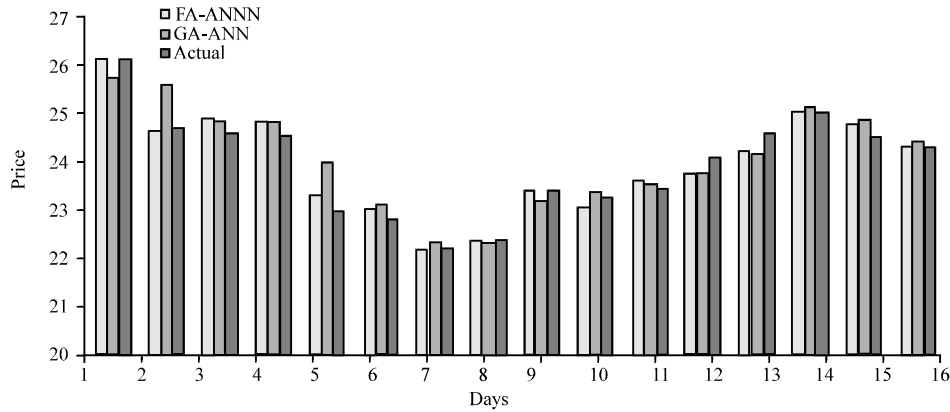


Fig. 6: Bar chart comparing the FA-ANN and GA-ANN predicted prices with the actual price for LFL

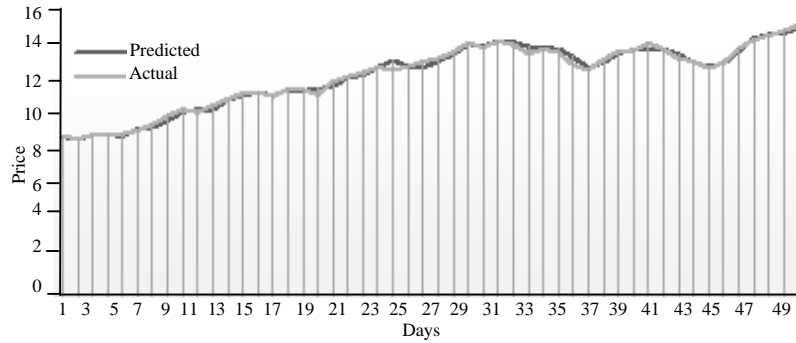


Fig. 7: Line chart comparing the FA-ANN predicted and actual closing price for LUV company

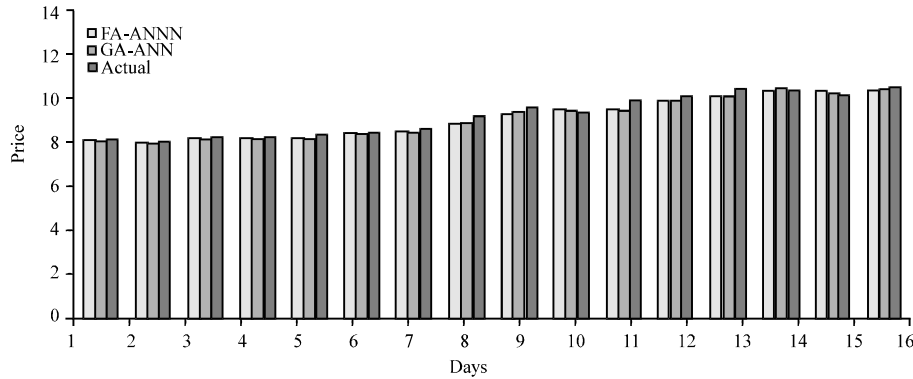


Fig. 8: Bar chart comparing the FA-ANN and GA-ANN predicted prices with the actual price for LUV

Table 10: The result of the randomization reduction scheme on the FA-ANN

X	ALK	Gol	LFL	LUV	ZNH
0.99	0.4310	0.1176	0.2206	0.1458	0.5222
0.98	0.4310	0.1178	0.2204	0.1458	0.5220
0.09	0.4310	0.1176	0.2204	0.1460	0.5220
0.08	0.4330	0.1186	0.2312	0.1490	0.5104
0.07	0.4340	0.1188	0.2382	0.1534	0.5080
0.06	0.4370	0.1282	0.2630	0.1520	0.4994
0.05	0.4298	0.1200	0.2526	0.1420	0.4966
0.04	0.4344	0.1296	0.2604	0.1540	0.5136
0.03	0.4402	0.1512	0.2770	0.1544	0.4976
0.02	0.4470	0.1554	0.3056	0.1504	0.4872
0.01	0.4394	0.1638	0.2802	0.1504	0.4826

Table 11: Summary of the average prediction error for the FA-ANN and GA-ANN

Company	FA-ANN aver error	GA-ANN aver error
ALK	0.4298	0.4308
Gol	0.1176	0.1672
LFL	0.2204	0.4158
LUV	0.1420	0.1408
ZNH	0.4826	0.4968

However, the values at some intervals between the reduction points do not affect the prediction results.

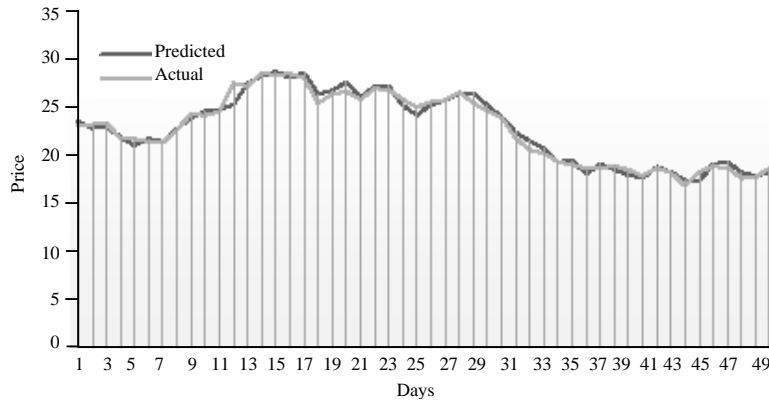


Fig. 9: Line chart comparing the FA-ANN predicted and actual closing price for ZHN company

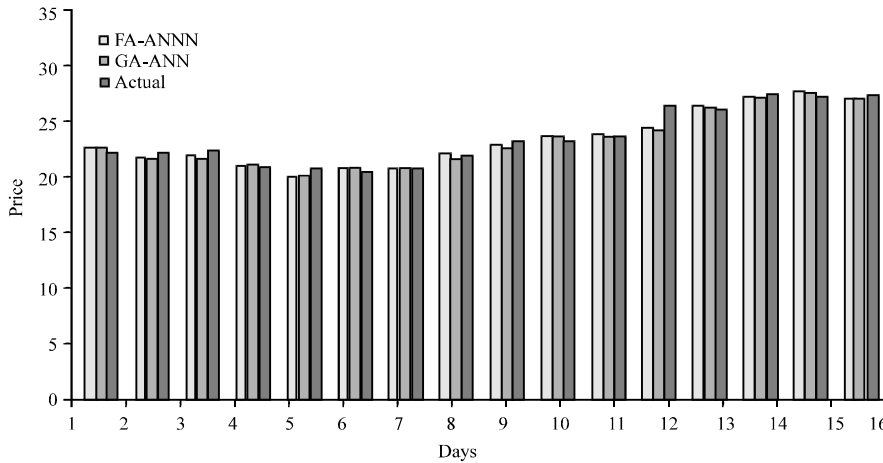


Fig. 10: Bar chart comparing the FA-ANN and GA-ANN predicted prices with the actual price for ZNH

RESULTS AND DISCUSSION

From the Table 5-10 and Fig. 1-10 used in the presentation of results shows that the FA-ANN made better predictions than the GA-ANN. Although in some cases the GA-ANN performed better. The difference is not conspicuous at first but with more iterations and data points, the FA-ANN proved its superiority over the GA-ANN. The average error (error here refers to the difference between the predicted value and the actual value) of each company using both methods showed the FA-ANN had less average error for four of the companies as shown in Table 12. The line chart is usually used to show the behavior and fluctuations of the stock market because the market has so many data points. The line chart is used in this study to show the closeness of the FA-ANN predicted value and the actual value for the closing price while the bar chart is used to compare the FA-ANN predicted, GA-ANN predicted and the actual closing price of the stocks for the companies.

Table 12: The percentage accuracy and error of the FA-ANN and GA-ANN

Variables	Accuracy (%)		Error (%)	
	FA-ANN	GA-ANN	FA-ANN	GA-ANN
ALK	42.00	40.00	0.84	0.71
Gol	86.00	66.00	0.18	0.24
LFL	50.00	32.00	0.45	1.00
LUV	78.00	78.00	0.11	0.08
ZNH	20.00	26.00	0.24	0.14
Average	55.20	48.40	0.36	0.43

Prediction of future events is not a trivial task: The choice of a forecasting model depends on how accurate it is with respect to the measure of errors (Sengupta, 2016). As suggested by a prediction with an error of 0.2 can be considered as accurate, depending on the nature of the dataset used. Table 13 shows the percentage accuracy and error for each company’s dataset using the FA-ANN and GA-ANN, respectively. The results were rounded up to two decimal places because the datasets are in two decimal places.

CONCLUSION

Error reduction is an important to predictive model, hence, the use of the randomization reduction scheme in this study, to improve the performance of the firefly algorithm. Optimization algorithm like the Firefly Algorithm (FA) will be a continuous area of research for researchers in the development of hybrid algorithms for prediction and classification purposes, especially for stock market predictions as investors search for secure investment options. This study applied the Firefly Algorithm (FA) an optimization algorithm to optimize the weights and train the Artificial Neural Network (ANN) for the purpose of stock market price prediction and it outclassed the Genetic Algorithm (GA) used for the same purpose. A randomization reduction scheme was adopted in the FA to gradually reduce the movements of firefly in the FA. Hence, the FA-ANN has the potential to predict accurately with less error, if parameters are properly set and trained with the appropriate amount of data.

The FA-ANN has been used in this study to predict the closing price for a day based on the previous closing prices for a particular number of days. However, the convergence rate of the firefly algorithm used in the model can be improved by making use of other meta-heuristic algorithms in order to make real time prediction of stock market prices which may include the opening prices, high prices, low prices, average prices and the volume of stock sold in the stock market.

REFERENCES

- Alweshah, M., 2014. Firefly algorithm with artificial neural network for time series problems. *Res. J. Applied Sci. Eng. Technol.*, 7: 3978-3982.
- Brabazon, A. and M. O'Neill, 2006. *Biologically Inspired Algorithms for Financial Modelling*. Springer, Berlin, Germany, ISBN:13-978-3-540-26252-7, Pages: 274.
- Brajevic, I. and M. Tuba, 2013. Training Feed-forward neural networks using firefly algorithm. *Proceedings of the 12th International Conference on Artificial Intelligence, Knowledge Engineering and Data Bases (AIKED'13)*, February 20-22, 2013, Clare College, Cambridge, UK., ISBN:978-1-61804-162-3, pp: 156-161.
- Costa, M.F.P., A.M.A. Rocha, R.B. Francisco and E.M. Fernandes, 2014. Heuristic-based firefly algorithm for bound constrained nonlinear binary optimization. *Adv. Oper. Res.*, 2014: 1-12.
- Das, D. and M.S. Uddin, 2013. Data mining and neural network techniques in stock market prediction: A methodological review. *Intl. J. Artif. Intell. Appl.*, 4: 117-127.
- Kumbhar, G.V. and R.V. Argiddi, 2015. Stock market index prediction by hybrid Neuro-genetic data mining technique. *Intl. J. Adv. Res. Comput. Commun. Eng.*, 4: 318-322.
- Mahajan, K.S. and R.V. Kulkarni, 2013. A review: Application of datamining tools for stock market. *Intl. J. Comput. Technol. Appl.*, 4: 19-27.
- Maharana, I., S. Das and M.R. Senapati, 2015. A hybrid intelligent artificial neural network model for stock market index prediction. *Intl. J. Comput. Appl.*, 2: 27-31.
- Neenwi, S., P.O. Asagba and L.G. Kabari, 2013. Predicting the Nigerian stock market using artificial neural network. *Eur. J. Comput. Sci. Inf.*, 1: 30-39.
- Nhu, H.N., S. Nitsuwat and M. Sodanil, 2013. Prediction of stock price using an adaptive Neuro-fuzzy inference system trained by firefly algorithm. *Proceedings of the 2013 International Conference on Computer Science and Engineering (ICSEC)*, September 4-6, 2013, IEEE, Nakorn Pathom, Thailand, ISBN: 978-1-4673-5324-3, pp: 302-307.
- Sengupta, S., 2016. What are the acceptable values for mean squared percentage error in a demand forecasting model?. Quora, Mountain View, California, USA.
- Teodorescu, H., A. Kandel and L.C. Jain, 1998. *Fuzzy and Neuro-Fuzzy Systems in Medicine*. CRC Press, New York, USA., ISBN:9780849398063, Pages: 401.
- Thomas, D., 2016. What are the disadvantage of genetic algorithm?. Quora, Mountain View, California, USA.
- Ting, J., T.C. Fu and F.L. Chung, 2006. Mining of stock data: Intra- and inter-stock pattern associative classification. *Threshold*, 5: 1-7.
- Yang, X.S., 2008. *Nature-Inspired Metaheuristic Algorithms*. Luniver Press, Beckington, UK., ISBN:13-978-1-905986-10-1, Pages: 116.
- Yang, X.S., 2009. Firefly algorithms for multimodal optimization. *Proceedings of the International Symposium on Algorithms Stochastic Algorithms: Foundations and Applications*, October 26-28, 2009, Springer, Berlin, Germany, ISBN:978-3-642-04943-9, pp: 169-178.