A HYBRID ARTIFICIAL NEURAL NETWORK BASED ON INTEGRATE-AND-FIRE NEURON MODEL AND EXPONENTIAL SMOOTHING METHOD

Eren Bas – Erol Egrioglu

Abstract

Artificial neural network methods are particularly useful for modelling non-linear time series. Determining the number of hidden layer units of the network is one of the main problems of many artificial neural network models. Integrate-and-fire neuron model artificial neural network is an artificial neural network without hidden layer unit number problem. The simple exponential smoothing method, which is one of the classical forecasting methods, is a method used in the forecasting of linear time series. This study proposes a hybrid artificial neural network method that combines exponential smoothing, a classical forecasting method, and integrate-and-fire neuron model artificial neural network, an artificial neural network method. The proposed new hybrid method can analyse time series with complex structures, since it is based on both an artificial neural network method and a classical forecasting method. This new hybrid neural network method is trained using differential evolution algorithm. The forecasting performance of the proposed method is evaluated by comparing the time series of an Istanbul Stock Exchange with various classical forecasting methods and artificial neural network methods.

Key words: artificial neural network, integrate-and-fire neuron model, simple exponential smoothing, differential evolution algorithm, forecasting.

JEL Code: C53, C61

Introduction

The fact that artificial neural network (ANN) methods are self-adaptive to data and do not require any assumptions makes them one of the most frequently used analysis methods in forecasting problems. One of the artificial neural network methods frequently used in time series forecasting problems is the multi-layer perceptron artificial neural network (MLP-ANN) proposed by Rumelhart et al. (1986). This method has the input, hidden, and output layers characteristic of a classic artificial neural network. Although a single-input, hidden, and output layer MLP-ANN method appears to be a simple neural network model, the number of units in the hidden layer is one of the main parameters affecting the complexity of this network. Even an MLP-ANN method with a single hidden layer and a dense hidden layer has a complex network structure, and it is inevitable that an MLP-ANN method with a dense hidden layer will have a highly complex network structure. Since the complex network structure of the MLP-ANN makes its use challenging, Yadav et al. (2007a) proposed the single multiplicative neuron model artificial neural network (SMNM-ANN), an artificial neural network model that does not require the determination of the number of hidden layer units. SMNM-ANN is a network where each input has a weight and a bias value. The output of this network is calculated by multiplying the linear combinations of the inputs, weights and bias values. SMNM-ANN is a commonly used artificial neural network in the forecasting problem. Cui et al. (2015) used the improved glowworm swarm optimisation algorithm in the training of SMNM-ANN. The sine cosine optimisation algorithm was used in the training of SMNM-ANN in the study by Kolay (2019). Pan et al. (2021) used the gradient descent algorithm in the training of SMNM-ANN. Egrioglu et al. (2023) proposed the use of a new genetic algorithm method in the training of SMNM-ANN. Isık and Akkan (2025) compared the SMNM-ANN method with MLP-ANN and Pi-Sigma ANN (PS-ANN) for water quality forecasting. Yadav et al. (2007b) proposed an ANN based on a different multiplicative neuron model, the integrate and fire neuron model (IFNM-ANN). The net input value of the IFNM-ANN is given by Equation (1), and the network's output (y) is given by Equation (2).

$$net = \prod_{i=1}^{p}[a_i \, log_e(b_i x_i) + c_i] \qquad (1)$$

$$y = f\,(net) \qquad (2)$$

Equation (1), x_i represents the inputs, and $a_i, b_i$and c_i represent the adjustable parameters of the network. Equation (2), f is the activation function. In IFNM-ANN, unlike SMNM-ANN, there are three different adjustable parameters. The total number of parameters in the network is calculated based on the number of inputs and these parameters. Additionally, IFNM-ANN is based on logarithmic calculations, unlike SMNM-ANN. In the IFNM-ANN, when calculating the net value of the network, the network input is multiplied by one of the adjustable parameters, and this product is subjected to a logarithmic transformation. The net value is then calculated by combining this logarithmic transformation with the linear combination of the other adjustable parameters. Joshi and Yadav (2024) demonstrated that IFNM-ANN produced more successful forecasting results than SNM-ANN.

The simple exponential smoothing (SES) method, which is one of the classical time series forecasting methods used in the analysis of univariate time series, is a method that can obtain forecasts from the past values of the forecasts or the weighted sums of the lagged variables of the time series. In a simple exponential smoothing method, forecasts are obtained by Equation (3).

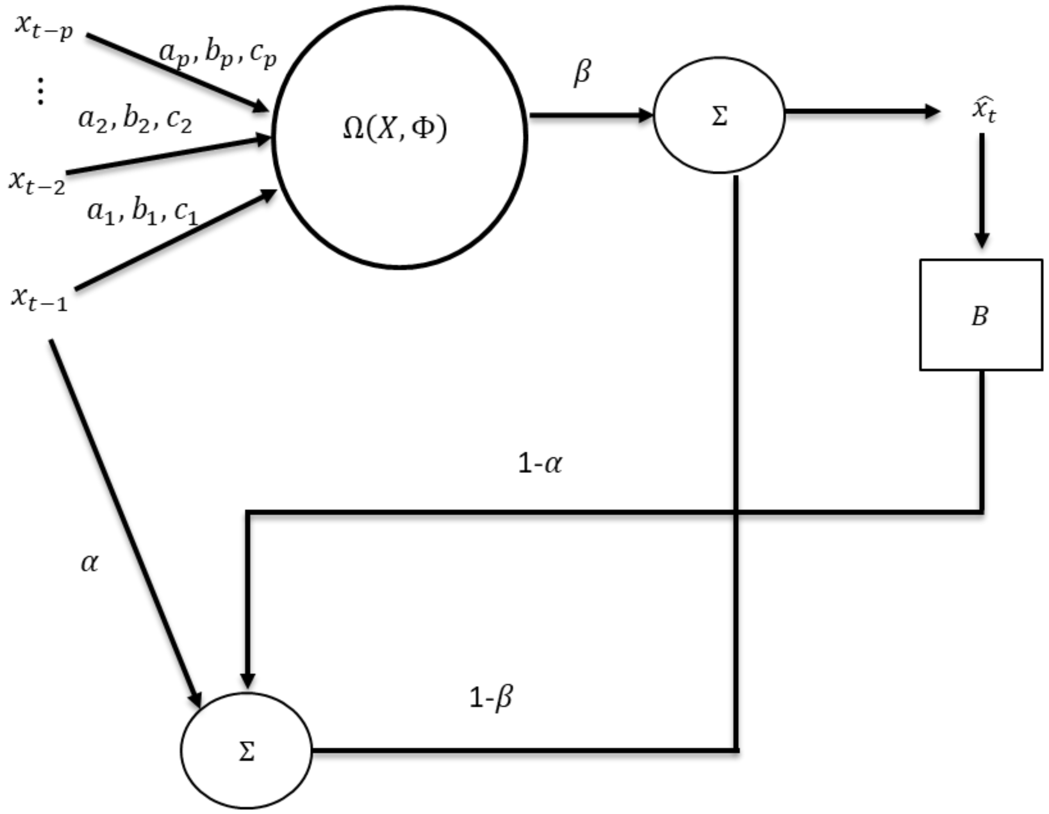
$$\hat{x}_{t+1} = \alpha x_t + (1-\alpha)\hat{x}_t, 0 \leq \alpha \leq 1 \qquad (3)$$

In Equation (3), α is expressed as the smoothing parameter. Although the α parameter is known as a weight assigned to the last observation, choosing a low α parameter means that a high weight value will be assigned to the last observation, while choosing a low α parameter means that a low weight value will be assigned to the last observation, that is, more weight will be assigned to older observations. Although this α parameter takes a value between zero and one, there is no precise method to determine this value. In this study, a new hybrid artificial neural network based on IFNM-ANN and SES method (SES-IFNM-ANN) is proposed. Differential evolution algorithm (DEA) proposed by Storn and Price (1997) is used in the training of this new hybrid artificial neural network. Both the weight values of IFNM-ANN, the α parameter in the SES method and the weight value used in the combination of these two methods are optimised simultaneously with this algorithm. In the performance evaluation of the proposed method, Istanbul Stock Exchange time series is used. This time series is a time series observed daily between 2014 and 2018 and a time series with different start and end dates is randomly selected from this time series. This time series is analysed with different artificial neural network methods other than SES-IFNM-ANN method. The other sections of the study are as follows. In the first part of the paper, the proposed hybrid artificial neural network method is introduced. The second section of the paper presents the application results. The third and last section of the paper is the discussion and conclusion.

1 The Proposed Hybrid Artificial Neural Network Method

The proposed SES-IFNM-ANN method is a new hybrid artificial neural network method that uses SES and IFNM-ANN methods together. The proposed SES-IFNM-ANN method has a common network architecture that combines the IFNM-ANN, an artificial neural network method, and the SES method, a classical forecasting method. This network architecture is shown in Fig. 1. The proposed SES-IFNM-ANN method allows the weight value used in the combination of these two methods to be transformed into the SES method by taking the weight value as zero and into the IFNR method by taking it as one. DEA is used in the training of this proposed new hybrid artificial neural network method.

Fig. 1: The architecture of SES-IFNM-ANN



The output of the IFNM-ANN $\left(\hat{x}_t^{IFNM-ANN}\right)$ part of the proposed hybrid artificial neural network method is obtained using Equation (4), while the output of the SES $\left(\hat{x}_t^{SES}\right)$ part is obtained using Equation (5). Finally, the output of the proposed hybrid SES-IFNM-ANN method is obtained as the weighted sum of these two methods, as shown in Equation (6).

$$\hat{x}_t^{IFNM-ANN} = f\left(\prod_{i=1}^{p}[a_i \log_e(b_i x_{t-i}) + c_i]\right) \quad (4)$$

$$\hat{x}_t^{SES}=(\alpha\hat{x}_{t-1} + (1-\alpha)x_{t-1}) \quad (5)$$

$$\hat{x}_t = \left(\beta\hat{x}_t^{IFNM-ANN} + (1-\beta)\hat{x}_t^{SES}\right) \quad (6)$$

1.1 Training of SES-IFNM-ANN method with DEA

Algorithm 1 provides a step-by-step explanation of the training algorithm for the proposed SES-IFNM-ANN with DEA.

Algorithm 1. The training of SES-IFNM-ANN method with DEA

Step 1. Determining the algorithm parameters

p: input number,

cn: the number of chromosomes in the population,

co: crossover rate,

$nval$: length of the validation set,

$ntest$: length of the test set

Step 2. Creating the initial population

Each chromosome in the population consists of a total of $3p + 2$ genes. The first $3p$ genes represent the weight and bias values of IFNM-ANN, the $(3p + 1)$ genes represent the α parameter in the SES method, and the $(3p + 2)$ genes represent the combination parameter (β).

Step 3. Partitioning the data set

The data set is divided into three parts: training, validation, and test sets.

Step 4. Calculating the fitness value of each chromosome

The fitness value of each chromosome in the population is calculated using the training data. The fitness value of each chromosome is obtained by calculating the root mean square error given by Equation (7).

$$RMSE_j = \sqrt{\frac{1}{ntrn}\sum_{i=1}^{ntrn}(x_t - \hat{x}_t)^2} \; ; j = 1,2,\dots,cn \qquad (7)$$

Step 5. Application of DEA operators

The DEA operators, mutation and crossover operators, are applied to each chromosome in the population.

Step 5.1. Application of the de/rand/1 mutation operator Three different chromosomes are randomly selected except for the relevant chromosome, and the total vector is obtained.

Step 5.2. Application of the crossover operator

The total vector is crossed over with the relevant chromosome according to the crossover rate, and a candidate chromosome is obtained. The candidate chromosome is compared with the related chromosome in terms of fitness value, and the chromosome with the lowest RMSE value is included in the population.

Step 6. Calculation of fitness values for the validation set and determination of optimal parameters

Step 7. Calculation of fitness values for the test set

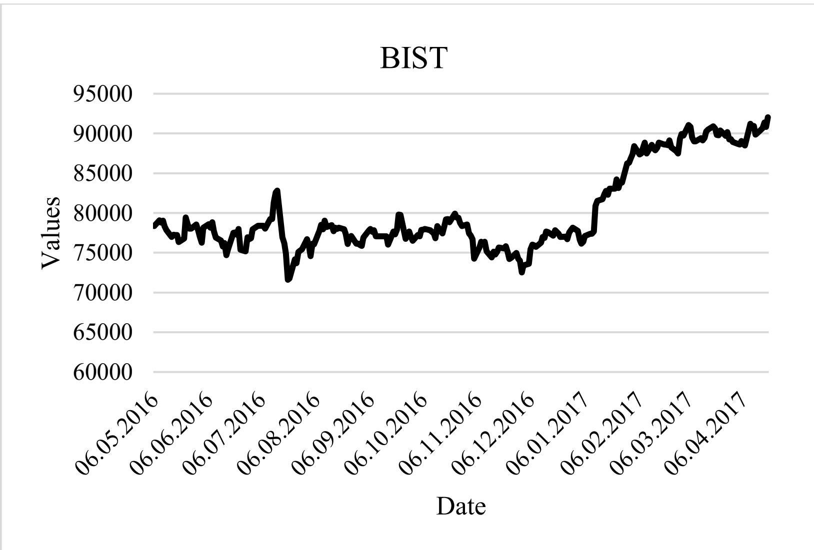
2 Applications

In the application part of the paper, a time series (BIST) with different start and end dates are obtained from the BIST time series observed daily between 2014 and 2018. The characteristics of the BIST time series are presented in Tab. 1, and the graph of BIST time series is shown in Fig. 2.

Tab. 1: The characteristics of BIST time series

Series	The number of Observations	Start Date	End Date
BIST	250	27/01/2016	10/01/2017

Fig. 2: The graph of BIST time series



BIST time series is analysed using SNM-ANN, PS-ANN, and MLP-ANN in addition to the proposed SES-IFNM-ANN method. In the analysis phase, $nval$ and $ntest$ are set to 30. The number of hidden layer units (m) in PS-ANN and MLP-ANN is increased by 1 to 5. co is varied from 0.1 to 0.7 in increments of 0.1, p is increased from 1 to 5 in increments of 1, and the number of chromosomes is increased from 30 to 100 in increments of 10. Experiments are conducted using different parameter combinations. As a result of the experiments, the parameter set with the lowest RMSE value is obtained as the optimal parameters. Since the analysis methods are artificial neural network methods, they will be affected by different initial conditions. Therefore, each artificial neural network method is rerun with ten different initial conditions. The mean, median, standard deviation, interquartile range, minimum, and maximum statistics of the different RMSE values obtained from each method are calculated, and the methods are compared based on these statistics. The RMSE statistics of each method for BIST time series is given in Tab.2.

Tab. 2: The RMSE statistics each method for BIST time series

Methods	Mean	Median	Standard Deviation	Interquartile Range	Minimum	Maximum	p	m
MLP-ANN	694,9756	687,6317	27,9601	27,6912	663,8632	753,5890	4	5
SMNM-ANN	665,8045	637,0609	90,7664	1,9823	636,1874	924,1204	2	-
PS-ANN	715,2646	724,0175	78,4243	105,2232	582,2653	876,7315	4	5
SES-IFNM-ANN	666,2007	667,7772	27,9560	49,0300	624,0619	709,7853	3	-

The results in Tab. 2 show that the new SES-IFNM-ANN method is better than most other methods, except for the minimum, median, and interquartile range statistics. However, the lowest maximum and standard deviation statistics are shown by the SES-IFNM-ANN method.

3 Conclusions and Discussion

Experts use the SES and IFNM-ANN methods to analyse time series forecasting problems. In this study, a new feedback hybrid artificial neural network called the SES-IFNM-ANN method is proposed, which combines both methods. The proposed SES-IFNM-ANN method is a hybrid artificial neural network that combines the SES and IFNM-ANN methods, and the proposed SES-IFNM-ANN method has the potential to revert to either of these two methods. DEA is employed to train the proposed SES-IFNM-ANN method. Following data analysis, it is concluded that the SES-IFNM-ANN method is the most successful analysis method. In future studies, researchers can train the SES-IFNM-ANN method with different artificial intelligence optimisation algorithms or convert the SES-IFNM-ANN method into a robust neural network.

Acknowledgment

This study is funded by the Turkish Scientific and Technological Research Council (TÜBİTAK) 2224-A - Grant Program for Participation in Scientific Meetings Abroad.

References

Cui, H., Feng, J., Guo, J., & Wang, T. (2015). A novel single multiplicative neuron model trained by an improved glowworm swarm optimization algorithm for time series prediction. Knowledge-Based Systems, 88, 195-209.

Egrioglu, E., Grosan, C., & Bas, E. (2023). A new genetic algorithm method based on statistical-based replacement for the training of multiplicative neuron model artificial neural networks. The Journal of Supercomputing, 79(7), 7286-7304.

Isık, H., & Akkan, T. (2025). Water quality assessment with artificial neural network models: Performance comparison between SMN, MLP and PS-ANN methodologies. Arabian Journal for Science and Engineering, 50(1), 369-387.

Joshi, H., & Yadav, A. (2024). Multiplicative neuron models for very short-term load forecasting. Electrical Engineering, 106, pages 7967–7972, (2024)

Kolay, E. (2019). A novel multiplicative neuron model based on sine cosine algorithm for time series prediction. Eskişehir Technical University Journal of Science and Technology A-Applied Sciences and Engineering, 20(2), 153-160.

Pan, W., Zhang, L., & Shen, C. (2021). Data-driven time series prediction based on multiplicative neuron model artificial neuron network. Applied Soft Computing, 104, 107179.

Rumelhart, D. E., Hinton, G. E., & Williams, R. J. (1986). Learning representations by back-propagating errors. *Nature*, 323(6088), 533-536.

Storn, R., & Price, K. (1997). Differential evolution–a simple and efficient heuristic for global optimization over continuous spaces. *Journal of Global Optimization*, 11, 341-359.

Yadav, A., Mishra, D., Yadav, R. N., Ray, S., & Kalra, P. K. (2007b). Time-series prediction with single integrate-and-fire neuron. Applied Soft Computing, 7(3), 739-745.

Yadav, R. N., Kalra, P. K., & John, J. (2007a). Time series prediction with single multiplicative neuron model. *Applied Soft Computing*, 7(4), 1157-1163.

Contact

Eren Bas

Giresun University

Department of Statistics, Faculty of Arts and Science, Giresun University, 28200, Giresun, Türkiye

eren.bas@giresun.edu.tr

Erol Egrioglu

Giresun University

Department of Statistics, Faculty of Arts and Science, Giresun University, 28200, Giresun, Türkiye

erol.egrioglu@giresun.edu.tr