



Designing an Appropriate Neural Network to Assist in More Accurate Stock Portfolio Predictions

M. A. Alipour^{1,*}, Sh. Parsinejad²

¹ Master's in Computer Engineering, Software Orientation, Islamic Azad University, North Tehran Branch, Iran

² Master's in Information Technology Engineering, E-Commerce Orientation, University of Guilan, Rasht, Iran

ARTICLE INFO	ABSTRACT
<p>Article History: Received 9 July 2018 Received in revised form 2 October 2018 Accepted 24 December 2018 Available online 27 December 2018</p>	<p>The increasing complexity and volatility of financial markets demand sophisticated methodologies for stock portfolio predictions. Recent advancements in deep learning and machine learning have provided new avenues for enhancing prediction accuracy in this domain. This literature review synthesizes current research findings on neural network designs specifically tailored for stock portfolio predictions, identifies knowledge gaps, and suggests potential future research directions. In this study, by designing feedforward artificial neural networks (MLP) and feedback networks (NARX), we examined the behavior of these two artificial neural network models for predicting stock portfolio prices. Subsequently, an Adaptive Neuro-Fuzzy Inference System (ANFIS) was developed and employed to forecast the stock prices of Ford Motor Company. The output of this system was compared with that of the designed artificial neural networks. The results of the simulation indicate that the designed neural network outperforms the ANFIS model in terms of prediction accuracy, exhibiting a lower error rate. The designed artificial neural network was able to predict the next day's closing stock price with minimal error on a daily basis. Additional findings from this project revealed that the ANFIS model performs effectively with limited data, making it suitable for scenarios where comprehensive data collection is either costly or impractical.</p>
<p>Keywords: Artificial Neural Network, Stock Price Prediction, MLP, NARX, Adaptive Neuro-Fuzzy Inference System (ANFIS)</p>	

1. INTRODUCTION

In today's globalized economy, one of the primary concerns for investors is navigating the competitive landscape to identify profitable and reliable markets, such as stock and currency markets, which are inherently complex. Consequently, predicting the return rates of companies' stocks in these capital markets holds significant importance. Various methods are employed for forecasting in financial markets, such as regression, time series analysis, genetic algorithms, and fundamental analysis. Among the precise nonlinear methods applicable in diverse prediction scenarios are Artificial Neural Networks (ANNs). ANNs are considered one of humanity's most recent technological advancements and are widely utilized in various scientific domains.

The adoption of advanced technologies and computational algorithms by investors for predictive purposes has led to increased profitability and enhanced business opportunities. Neural networks are dynamic systems capable of

* Corresponding Author: ali_alipour@live.com

Master's in Computer Engineering, Software Orientation, Islamic Azad University, North Tehran Branch, Iran



processing time series data to extract underlying knowledge or rules and adapt them within the network's structure. This system is based on computational intelligence, mimicking the brain's ability to process complex features.

The objective of this project is to design an appropriate artificial neural network for predicting the stock price of Ford Motor Company in the United States. This study involves the design and performance analysis of two types of artificial neural networks: a feedforward Multilayer Perceptron (MLP) and a feedback Nonlinear Autoregressive Network with Exogenous Inputs (NARX). Additionally, the behavior of the Adaptive Neuro-Fuzzy Inference System (ANFIS) in the context of stock price prediction is also investigated.

1.1. Electronic Trading Networks

One of the most significant transformations in capital markets over the past few decades has been the emergence of modern tools such as advanced trading systems and information technologies, which have exerted an undeniable impact on financial markets. Among these developments is the phenomenon known as electronic trading, which facilitates direct interaction through electronic exchange systems. This advancement reduces transaction costs, increases the speed of trade execution, enhances market fluidity, and ultimately boosts market liquidity.

Innovations in information technology have brought about substantial changes and developments in the structure and functioning of financial markets. By increasing transaction speed and reducing costs, technology has contributed to greater transparency and liquidity in capital markets [1].

1.2. Investment Companies

These institutions or financial intermediaries facilitate the transfer and allocation of funds from sectors with a surplus of capital to those experiencing a deficit, thereby participating in profit-making activities with various commercial enterprises. In other words, they refer to companies and organizations that provide the necessary capital for other firms by purchasing financial assets such as stocks and securities.

Financial intermediaries sell shares to the public and invest the proceeds in a diversified portfolio of securities. Each share sold represents a portion of the securities portfolio managed by the investment company on behalf of its shareholders. It is important to note that the investment objective of the company determines the type of securities it acquires [1,3].

2. REVIEW OF PREVIOUS WORKS

Various studies have highlighted the effectiveness of deep learning methods in financial predictions. Fischer and Krauss (2017) introduced a deep learning model utilizing long short-term memory (LSTM) networks, which are particularly adept at handling high-dimensional time series data. This capability is essential for stock portfolio predictions, as stock prices are inherently time-dependent (Fischer & Krauss, 2017). The model's ability to capture both global patterns and local properties can enhance forecasting accuracy, making it a potentially valuable tool for investors aiming to optimize their portfolios [4].

Incorporating market sentiment into stock predictions has been identified as a critical factor for improving the accuracy of neural networks. Sohangir et al. (2018) explored the role of sentiment analysis within deep learning frameworks, suggesting that integrating sentiment data can provide a more holistic view of market conditions. By analyzing sentiment alongside traditional financial metrics, investors can make more informed decisions regarding asset allocation [5]. Addo et al. (2018) also highlighted the potential of incorporating sentiment data through ensemble neural networks, which can significantly enhance predictive capabilities. This integration allows for a more nuanced understanding of market dynamics and can lead to improved portfolio management strategies [6].

Chang and Liu (2008) employed a Type TSK fuzzy rule for stock price prediction. In their fuzzy TSK model, industrial indices were used as input variables, resulting in a linear combination of these inputs. The model was evaluated using data from the Taiwan Stock Exchange, demonstrating an accuracy rate of approximately 97.6% on the TSE index [7].

In 2010, Akbar Esfahanipour and Voria Aqa Amiri conducted a study on forecasting the Tehran Stock Exchange using an adaptive neuro-fuzzy inference system (ANFIS) based on the indirect Takagi-Sugeno-Kang (TSK) method [8]. The fuzzy TSK model incorporated market indices as input variables and applied a specific fuzzy rule system to predict stock prices. This model was tested through multiple empirical experiments, achieving a high accuracy rate of approximately 97.8%, confirming its effectiveness. An example of the applied rule is as follows:

IF x_1 is A_1 AND x_2 is A_2 AND, ..., AND x_p is A_p

THEN $y = c_0 + c_1 x_1 + c_2 x_2 + \dots + c_p x_p$

In this model, x_1, x_2 represent input variables, A_1, A_2 denote membership functions, y is the output variable, and c_0, c_1, c_2 are consequent parameters. The results indicate that the neuro-fuzzy network architecture can identify the general characteristics of the stock market with higher speed and accuracy [8].

Havandi et al. (2010) proposed a hybrid model combining neural networks and fuzzy genetic algorithms for stock price prediction. They applied this model to forecast stock prices of an IT company and an aerospace company listed on the New York Stock Exchange. The proposed model was compared with the ARIMA model, genetic algorithms, and neural networks. The results demonstrated that the hybrid model outperformed all previous models in prediction accuracy [9].

In reference [10], researchers examined the predictability of stock market returns using adaptive networks based on fuzzy inference systems. The study aimed to determine whether an ANFIS algorithm could achieve high-accuracy predictions of stock market returns. The research incorporated six macroeconomic variables and three indices as input variables to predict stock prices on the Istanbul Stock Exchange. The results indicated an accuracy of 98.3%.

In 2011, Xiang et al. introduced an artificial intelligence-based model incorporating an adaptive neuro-fuzzy inference system (ANFIS) along with reinforcement learning (RL). This system was designed to identify trend changes for investment decision-making. In the presence of dynamic cyclical movements in the moving average, the RL model, in conjunction with ANFIS-RL, determined the correct forecast values. This method, applied to the U.S. stock market, was developed based on optimized ANFIS inference systems. The incorporation of reinforcement learning significantly improved the identification of stock cycles, yielding favorable results [11].

Tsang and Cheng Sheng (2011) analyzed artificial neural networks (ANN), decision trees, and a hybrid model combining both approaches to assess their effectiveness in market analysis. The study utilized 10 different stocks across 320 datasets for prediction. The hybrid model achieved an average accuracy of 13.75%, the ANN model attained 15.31%, and the decision tree model recorded an accuracy of 14.06%. The findings indicated that, compared to the other two methods, ANN demonstrated greater stability in stock price prediction [12].

In reference [13], the direction of the Istanbul Stock Exchange price index was predicted using artificial neural networks and support vector machines (SVM). The study employed daily data over a ten-year period, incorporating ten technical indicators as input variables. The final results revealed that the artificial neural network outperformed the support vector machine model in predictive accuracy.

In 2012, Shahrokh Asadi et al. applied an evolutionary Levenberg-Marquardt neural network to forecast stock portfolios. Their proposed hybrid model combined genetic algorithms with the Levenberg-Marquardt (LM) method, which was used for training feedforward neural networks. The LM algorithm and genetic algorithms were leveraged to optimize the initial weight configuration of the neural network. Additionally, preprocessing techniques, such as data transformation and input variable selection, were employed to enhance model accuracy. The findings demonstrated that integrating multiple methods effectively improved prediction outcomes. The results further indicated that the ANN generated using the PELMNN method consistently outperformed the RW model in all tested scenarios [14].

Mehdi Salehi and Zohreh Aminifard conducted a study on dividend prediction using hybrid models. They utilized data from chemical companies listed on the Tehran Stock Exchange between 2006 and 2010.

The hybrid model framework was based on PSO-SVR and PSO-LARS algorithms. The PSO (Particle Swarm Optimization) algorithm identified the optimal combination of variables that influenced dividend predictions. The selected variable data were then separately fed into the SVR (Support Vector Regression) algorithm, a nonlinear

regression model, and the LARS (Least Angle Regression) algorithm, a linear regression model, for training. The SVR and LARS algorithms were subsequently tested with evaluation data to measure prediction errors. PSO, as a subset of artificial intelligence and an optimization technique inspired by the collective movement of fish and birds in search of food, played a crucial role in this model [15].

By combining the PSO-SVR and PSO-LARS algorithms, the study aimed to identify key factors directly influencing dividends. The results indicated that the PSO-SVR hybrid model exhibited lower prediction error compared to PSO-LARS [15].

In 2013, Jonathan Tikno introduced a novel approach for predicting financial market behavior using a Bayesian regularized artificial neural network to forecast the next-day closing price of a stock portfolio [16].

Time series analysis is inherently complex. The backpropagation neural network, widely used in ANN applications that rely on supervised learning, employs gradient descent to minimize error functions.

This method utilizes three main layers: an input layer, a hidden layer, and an output layer, all interconnected through weighted links. These links determine the weight values and the strength of the connection between nodes.

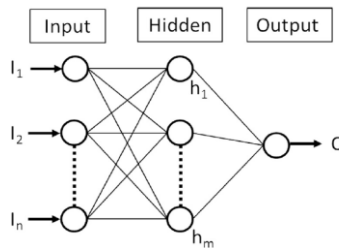


Fig. 1. General Architecture of a Three-Layer Feedforward Neural Network

An important aspect to consider is supervised learning, which is used to adjust and modify the weights in order to minimize errors. One critical issue in this context is overfitting, where excessive training leads to noise in the network and an increase in prediction error.

To mitigate overfitting, a mathematical technique known as Bayesian regularization is employed for transforming nonlinear systems. This technique reduces the sum of squared errors and enhances the objective function value, ultimately improving the model's predictive performance.

$$F = \beta ED + \alpha EW \tag{1}$$

According to Equation (1), where **F** represents the objective function, ED is the sum of squared errors, EW is the sum of squared network weights, and α and β are the parameters of the objective function.

In Bayesian networks, the weights assigned to variables are treated as random variables, and the probability density function is determined based on Bayesian rules as follows:

$$P(w|D, \alpha, \beta, M) = \frac{P(D|w, \beta, M)P(w|\alpha, M)}{P(D|\alpha, \beta, M)} \tag{2}$$

In this formula, W represents the network weight vector, D denotes the data vector, and M refers to the applied neural network model. The findings of this study indicate that overfitting, which negatively impacts the predictive performance of neural networks, is effectively reduced [16].

In 2014, Nouraddin Mosazadeh Abbasi and colleagues developed a stock value and price index prediction system for the Tehran Stock Exchange using a hybrid approach combining genetic algorithms and neural networks. In this method, the initial prediction was conducted using a neural network, after which the output weights of the best-performing network were selected as the standard criterion. To refine the prediction process, genetic algorithms were subsequently applied. In the final stage, the pattern derived from the neural network was further analyzed using fuzzy rules [17]. In the same year, Loukis et al. applied artificial neural networks to process noise in stock market

prediction. The objective of their research was to optimize neural network structures and analyze the causes of overfitting in neural networks [18].

3. METHODOLOGY

To predict stock portfolios using artificial neural networks (ANNs), we can effectively manage financial market fluctuations. This network consists of input and output data, and through these data, the learning process is carried out by ANNs. Compared to other methods, artificial neural networks have demonstrated superior performance and are a useful tool for stock price prediction. This advantage arises from the fact that ANN does not require strong assumptions to construct a predictive model. Furthermore, ANNs have the capability to approximate any nonlinear function without prior assumptions about the properties of the data [16, 17].

The prediction process is implemented using MATLAB software, and the accuracy of the results is evaluated based on performance metrics.

3.1. Multi-Layer Perceptron (MLP) Model

An artificial neural network is a computational model that mimics the learning process of biological neural systems. It is designed to model the way the human brain processes and learns from information.

The Multi-Layer Perceptron (MLP) model is one of the most well-known neural network models. It is an advanced version of the perceptron model initially proposed by Rosenblatt in the 1950s. This model consists of a feedforward layered network, where each neuron in the MLP model includes a nonlinear activation function.

The architecture of the MLP model is illustrated in Figure 2. Each circle represents a neuron, and neurons are organized into layers. The network consists of hidden layers (Layer 1 and Layer 2) and an output layer. Although the inputs are introduced at the bottom as the input layer, no neurons are typically assigned to this layer.

The hidden layers are named as such because their outputs are passed to neurons in the upper layer, making them inaccessible to users who only observe the final output layer. The MLP model enables nonlinear mapping between its inputs and outputs, making it a powerful tool for complex predictive modeling [2, 3, 19].

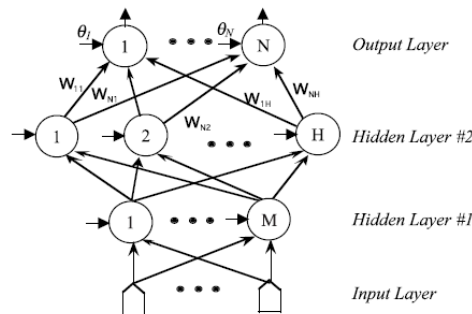


Fig. 2. Multi-Layer Perceptron (MLP) Configuration

3.2. Perceptron Learning Rule Used in the Simulation

3.2.1. Supervised Learning

In this training method, both the network inputs and their corresponding outputs are predefined. During the training process, an input is fed into the neural network, and the network generates an output response. This output is then compared to the desired target output. If a discrepancy exists between the actual and expected output, the network generates an error signal.

This error signal is used to compute the necessary adjustments to the network's weights. In other words, the goal is to minimize the error—or ideally, bring it as close to zero as possible. In this approach, the neural network is trained on a set of samples to achieve optimal learning. Equation (3) represents the sample set used in this training process.

$$\{P_1, t_1\} \{P_2, t_2\} , \dots , \{P_r, t_r\} \tag{3}$$

P_r represents the input to the neural network, and t_r represents the correct or expected target output. When the input is fed into the neural network, the network's outputs are evaluated and compared with the correct target outputs. At this point, the learning rule is applied to adjust the weights and biases of the network so that the network's output becomes closer to the target and the error is minimized as much as possible.

3.3. Dataset Used

For this prediction, daily data from Apple Inc. stock (from the NASDAQ website, a major stock exchange in the U.S.) was used. NASDAQ is the first electronic stock exchange in the U.S. and, after the New York Stock Exchange, is the second largest stock market globally.

3.4. Input and Output Data

The dataset spans from 2010 to 2016 and includes variables such as trading volume, opening stock price, highest price, and lowest price. The output data refers to the closing stock price of the next day, which is used as the target output for the neural network. Before processing the neural network data, normalization is performed on the data. The following equation (4) is used for normalization:

$$x_n = (x - \min x) / (\max x - \min x) * 2 - 1 \tag{4}$$

3.5. Model Validation Using the Dataset

Model validation is the process through which input and output vectors, not involved in the model training process, are applied to the model to determine its accuracy. One of the challenges in model validation is selecting appropriate validation data. The purpose of using validation data is to assess overfitting in the model. If training is performed incorrectly, the model may perfectly match the training data but fail to generalize properly to unseen data. Validation data helps detect overfitting occurrences.

3.6. Project Overview

For this analysis, Ford Motor Company stock data from the United States was used. The data, collected from NASDAQ (the first internet-based stock exchange globally), was evaluated over a five-year period from 2010 to 2016. The forecasting involved two data stages: one for daily stock prices and another for monthly stock prices. To predict stock prices, a feedforward and feedback neural network was designed, and the results were compared between the two types of networks. The project also included a study on Adaptive Neuro-Fuzzy Inference System (ANFIS), and the daily and monthly stock price predictions using ANFIS were compared with the artificial neural network outcomes.

4. DAILY STOCK PRICE PREDICTION FOR FORD MOTOR COMPANY

4.1. Dataset Used

The daily stock data of Ford Motor Company was sourced from NASDAQ, a leading U.S. stock exchange. NASDAQ is the first electronic stock exchange in the U.S. and the second largest stock market in the world, after the New York Stock Exchange.

4.2. Input and Output Data

The dataset spans from 2010 to 2016 and includes trading volume, opening price, highest price, and lowest price of the stock. The output data represents the closing price of the stock for the following day, which serves as the target for the neural network output. Before processing the neural network data, normalization is performed on the data. The following equation (5) is used for normalization:

$$x_n = (x - \min x) / (\max x - \min x) * 2 - 1 \tag{5}$$

4.3. Design of the Feedback Neural Network

For the design of the neural network, MATLAB was used, with four inputs and one output. A hidden layer consisting of 10 neurons was chosen. The network is a NARX-type feedback network. The Mean Squared Error (MSE) was calculated to be 0.002, and the regression coefficient obtained was 0.99. Figure (3) illustrates the proper training of the feedforward network.

The green dashed lines shown in Figure 4-9 indicate that the intersection point of these lines represents the best point at which the network was trained. This intersection is marked with a circle in the figure. When the lines on the graph pass through this circle, it confirms that the neural network was trained correctly.

As seen in the figure, at Epoch 8, the network reached a mean squared error of 0.00063795, and training was stopped due to this very low error. Therefore, it can be concluded that the best training response is obtained at Epoch 8, based on the system's output, as shown in the figure.

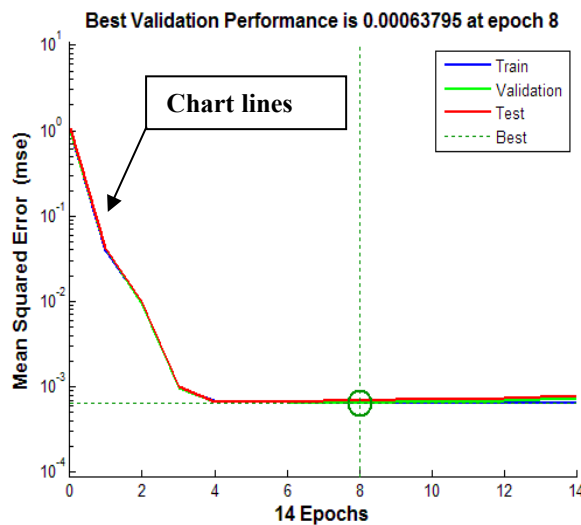


Fig. 3. Mean Squared Error Graph of the Feedforward Network

In Figure (4-10), the regression coefficients calculated by the designed neural network are shown. The closer the regression coefficient is to 1, the stronger the correlation between the output and the input. In other words, the output is more influenced by the input as the regression coefficient approaches 1.

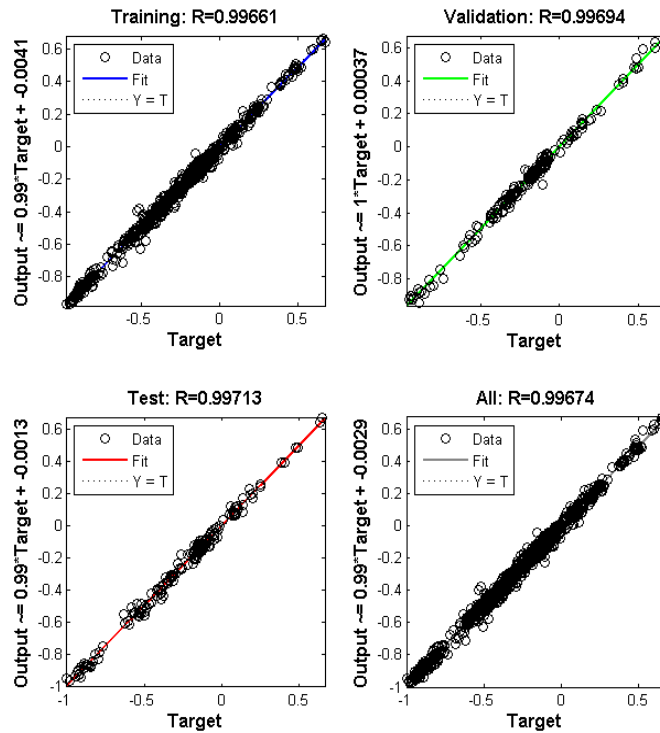


Fig. 4. Obtained Regression Graphs

In Figure (4), Y represents the output of the artificial neural network during the training process, and T represents the network's target. The best result obtained from training the neural network is represented by the alignment of the fit line (dashed line) with the line $Y = T$. This indicates optimal model performance, where the output closely matches the target.

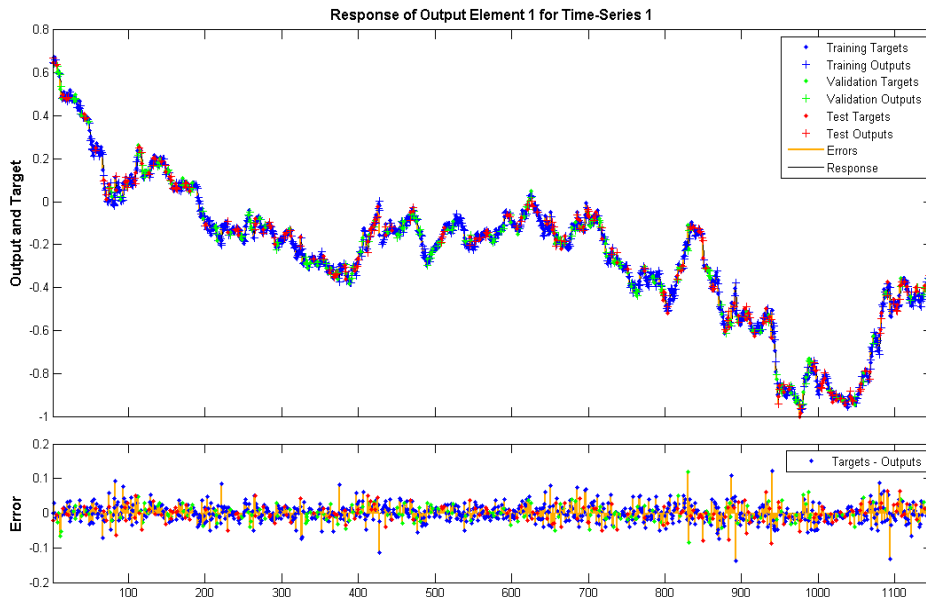


Fig. 4. Network Response to Training and Testing Data

In Figure (4), the results for both the training data and the testing data are presented along with their respective outputs and target responses. This visualization helps to compare the network's performance across both datasets, demonstrating how well the network has learned from the training set and how effectively it generalizes to unseen testing data.

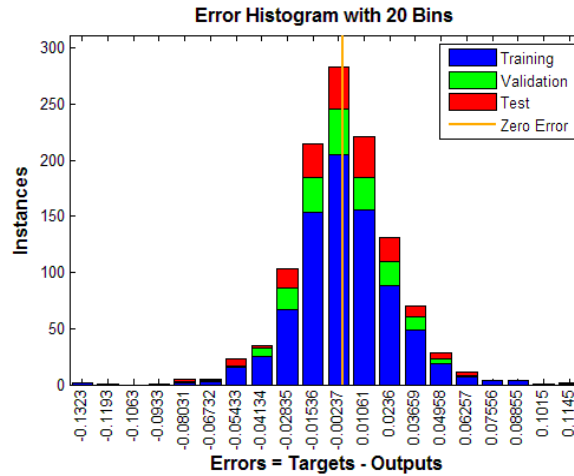


Fig. 5. Error Histogram

Figure (5) illustrates the error histogram. As shown, with the increase in test data, the bars become taller and shift towards the orange line, which represents zero error. The closer the bars are to this line, the better the artificial neural network is at reducing error and providing accurate responses.

4.4. Design of the Feedforward Neural Network

In this design, the network's inputs, outputs, hidden layer, and the number of neurons in the hidden layer have been selected similarly to the feedback neural network. The feedforward network is designed as an MLP (Multilayer Perceptron), and the results indicate that the behavior of the feedforward neural network is very similar to that of the feedback neural network. Figure (6) illustrates the training process of the feedforward neural network.

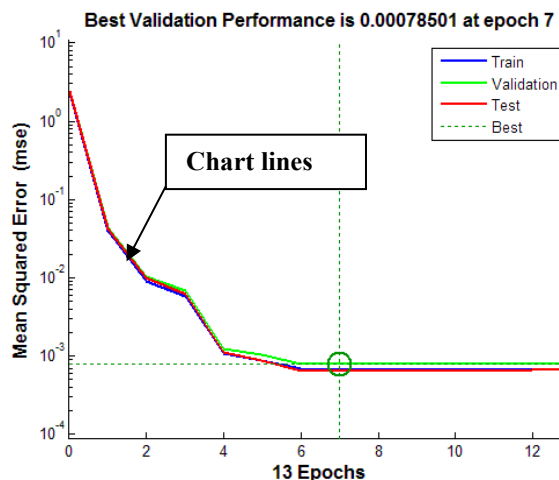


Fig. 6. Error Mean Square Plot of the Feedforward Neural Network

In Figure (6), the mean square error (MSE) of the feedforward MLP network is shown. As mentioned in previous explanations, the green dashed points indicate the best point where the network was trained, and the crossing of the

graph lines through the green circle signifies the correctness of the artificial neural network training. As observed in Figure (6), the network reaches a mean square error of 0.00078501 after the 7th epoch, and the training stops with this very low error.

4.5. Monthly Stock Prediction for Ford

For the monthly stock prediction, the monthly stock data of Ford from NASDAQ is used. Similar to the daily prediction, the same type of input and output data is employed. The only difference is that the monthly data contains fewer data points within the same five-year period compared to the daily data, and the output of the artificial neural network is the closing stock price for the next month. The reduced amount of input data increases the system's error and decreases the prediction accuracy compared to the daily forecast.

4.5.1. Monthly Prediction Using NARX Feedback Network

The design of this artificial neural network uses five neurons in a hidden layer.

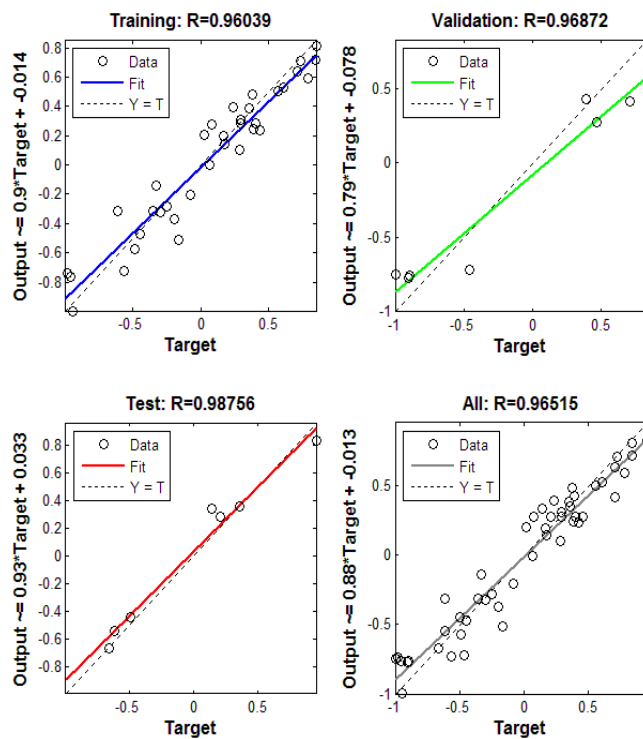


Fig. 7. Regression Plots

Figure (7) illustrates that the fit line does not completely align with the dashed line $Y=T$, indicating an increase in the neural network error during learning. The data points are shown as circles in the figure. In the monthly prediction, the reduction in data is clearly visible in the figure, and this reduction negatively impacts the network's learning process.

In this simulation, four inputs are provided to the system, and one output, which is the closing stock price for the next day, is obtained from the system. The tests conducted on the network showed that the mean square error eventually reached 0.02. Figure (8) shows the increase in error.

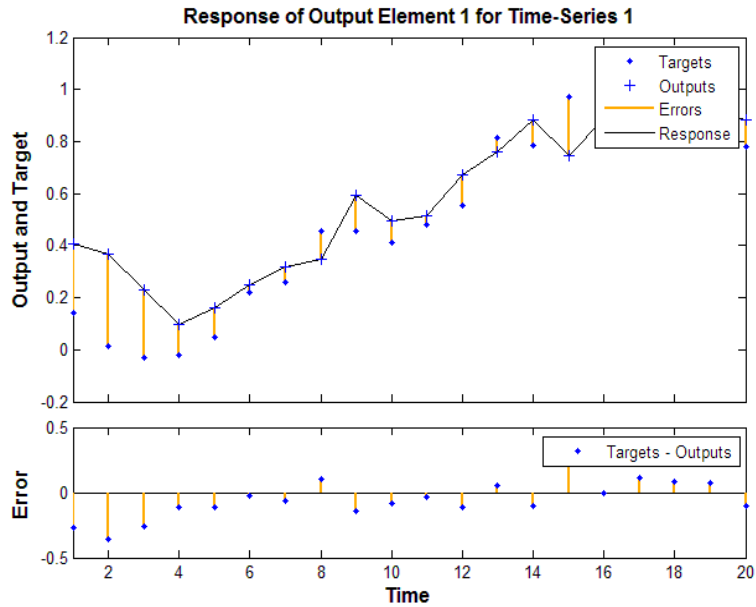


Fig. 8. Output Error

In Figure (8), the blue points represent the target of the network, while the output of the network is denoted by the blue plus sign symbol. The difference between them is the error, which is shown in orange.

4.5.2. Monthly Prediction with MLP Feedforward Network

This network is designed with one hidden layer consisting of five neurons, four inputs, and one output. The Mean Squared Error (MSE) obtained from this network was found to be similar to the NARX feedback network.

4.5.3. Model Learning and Inference via ANFIS

Adaptive Neuro-Fuzzy Inference System (ANFIS) operates similarly to neural networks. Neuro-adaptive learning methods provide an approach for creating a fuzzy modeling framework to learn information from a dataset [20]. ANFIS extracts a Fuzzy Inference System (FIS) using input and output data [21]. The parameters of the membership functions of this system are adjusted through the backpropagation algorithm or by combining it with the least squares method. This adjustment process enables the fuzzy system to learn its structure from the dataset. It is worth noting that ANFIS has a simpler architecture compared to artificial neural networks [22-23].

4.5.4. Daily Stock Prediction for Ford with ANFIS

In this prediction, the same five-year dataset used for the neural network design was employed, consisting of four inputs and one output. As shown in Figure (9), each input includes three membership functions, and the blue circles represent the rules generated by the system. The number of rules extracted by the network reached 86.

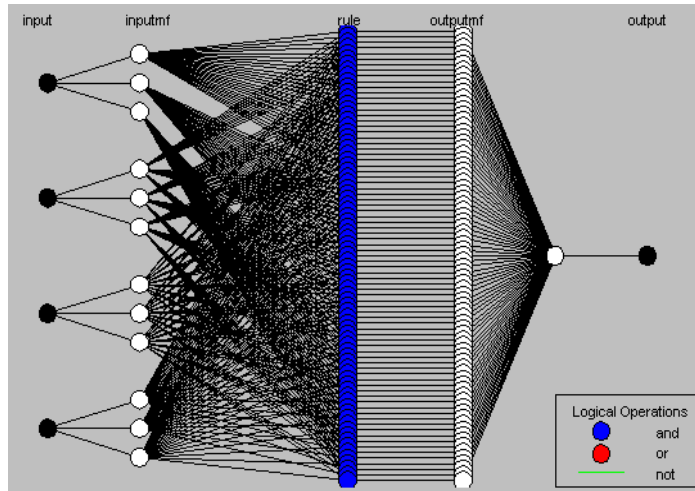


Fig. 9. Input and Output of the Adaptive Neuro-Fuzzy Inference System

In Figure (10), this network is trained, and the parameters are adjusted during the training process. The calculations of these parameters are facilitated by the gradient, which serves as a measure for evaluating the adequacy of the fuzzy inference system’s model parameters. After calculating the gradient vector, other optimization routines can be used to optimize the parameters and reduce the error. Typically, the accuracy of the network is assessed by calculating the sum of the squared errors. In the adaptive neuro-fuzzy system, the backpropagation method, or a combination of it with the least squares estimation, is used for estimating the parameters of the membership functions.

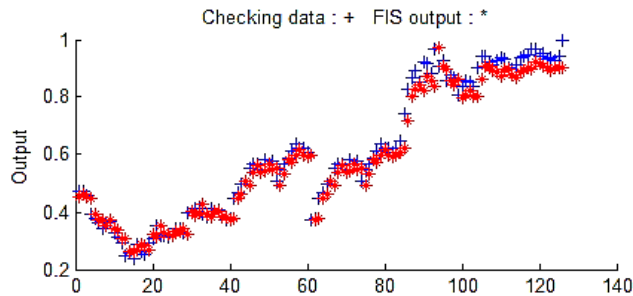


Fig. 10. Training of the ANFIS System

In Figure (10), the blue points represent the trained data, while the red points correspond to the test data. In this simulation, the resulting MSE is 0.02.

4.5.5. Monthly Stock Prediction of Ford with ANFIS

The monthly prediction, using a structure similar to the adaptive neuro-fuzzy inference system (ANFIS) applied in daily predictions, was carried out. The only difference was that the data set was smaller, and the resulting MSE was calculated by the system to be 0.16.

5. CONCLUSION

In this paper, an artificial neural network of the feedforward MLP and feedback NARX types was designed to examine the behavior of these two neural networks for predicting stock portfolio prices.

Subsequently, the adaptive neuro-fuzzy inference system (ANFIS) was designed and used for predicting the stock price of Ford. The output of this system was compared with the designed artificial neural networks.

The results of this simulation indicate that the designed neural network performs better than the adaptive neuro-fuzzy inference system in terms of prediction error. The designed artificial neural network was able to predict the closing stock price for the next day with minimal error on a daily basis.

Other results from this project showed that the ANFIS system, with a smaller data set, performs well in situations where collecting all the data for prediction is costly or impossible.

Another finding of this project is that an ANFIS system can only produce a single output, whereas artificial neural networks have no such limitation on the number of outputs, as long as the hardware system can handle the processing.

Transparency Statement

The data supporting this study are available upon reasonable request to the corresponding author, subject to ethical and confidentiality considerations.

Acknowledgments

We would like to express our gratitude to all individuals who contributed to this project.

Declaration of Interest

The authors declare that they have no competing interests.

REFERENCES

- [1] Torabi, A., Pashapur, S., & Neshat, N. (2013). A novel approach in artificial neural network modeling based on fuzzy regression logic and its application in forecasting. *Journal of Industrial Engineering*, 47, 15-24.
- [2] Soltani, S., Sardari, S., & Sheikhpour, M. (2010). *Artificial neural networks* (1st ed.). Nas.
- [3] Hanfizadeh, P., & Jafari, A. (2010). A hybrid model of feedforward artificial neural networks and self-organizing Kohonen map for stock price prediction. *Quarterly Scientific-Research Journal of Industrial Studies*, 8(19), 165-187.
- [4] Fischer, T. G., & Krauss, C. (2017). Deep learning with long short-term memory networks for financial market predictions. *European Journal of Operational Research*, 270, 654-669. <https://doi.org/10.1016/j.ejor.2017.11.054>
- [5] Sohangir, S., Wang, D., Pomeranets, A., & Khoshgoftaar, T. (2018). Big data: Deep learning for financial sentiment analysis. *Journal of Big Data*, 5, 1-25. <https://doi.org/10.1186/s40537-017-0111-6>
- [6] Addo, P., Guégan, D., & Hassani, B. K. (2018). Credit risk analysis using machine and deep learning models. *ERN: Other Econometrics: Econometric & Statistical Methods - Special Topics*. <https://doi.org/10.2139/ssrn.3155047>
- [7] Chang, P. C., & Liu, C. H. (2008). A TSK type fuzzy rule-based system for stock price prediction. *Expert*

Systems with Applications, 34, 135-144. <https://doi.org/10.1016/j.eswa.2006.08.020>

- [8] Esfahanipour, A., & Aghamiri, W. (2010). Adapted neuro-fuzzy inference system on indirect approach TSK fuzzy rule base for stock market analysis. *Expert Systems with Applications*, 37, 4742-4748. <https://doi.org/10.1016/j.eswa.2009.11.020>
- [9] Hadavandi, E., Shavandi, H., & Ghanbari, A. (2010). Integration of genetic fuzzy systems and artificial neural networks for stock price forecasting. *Knowledge-Based Systems*, 23, 800-808. <https://doi.org/10.1016/j.knosys.2010.05.004>
- [10] Acar Boyacioglu, M., & Avci, D. (2010). An adaptive network-based fuzzy inference system (ANFIS) for the prediction of stock market return: The case of the Istanbul Stock Exchange. *Expert Systems with Applications*, 37, 7908-7912. <https://doi.org/10.1016/j.eswa.2010.04.045>
- [11] Tan, Z., Quek, C., & Cheng, P. Y. K. (2011). Stock trading with cycles: A financial application of ANFIS and reinforcement learning. *Expert Systems with Applications*, 38, 4741-4755. <https://doi.org/10.1016/j.eswa.2010.09.001>
- [12] Chang, T. S. (2011). A comparative study of artificial neural networks, and decision trees for digital game content stocks price prediction. *Expert Systems with Applications*, 38, 14846-14851. <https://doi.org/10.1016/j.eswa.2011.05.063>
- [13] Kara, Y., Acar Boyacioglu, M., & Baykan, M. K. (2011). Predicting direction of stock price index movement using artificial neural networks and support vector machines: The sample of the Istanbul Stock Exchange. *Expert Systems with Applications*, 38, 5311-5319. <https://doi.org/10.1016/j.eswa.2010.10.027>
- [14] Asadi, S., Hadavandi, E., Mehmanpazir, F., & Nakhostin, M. (2012). Hybridization of evolutionary Levenberg–Marquardt neural networks and data pre-processing for stock. *Knowledge-Based Systems*. <https://doi.org/10.1016/j.knosys.2012.05.003>
- [15] Salehi, M., & Amini Fard, Z. (2012). Investigating the factors influencing dividend earnings prediction using hybrid models: Case study of the chemical industry. *Empirical Accounting Research*, 111-130.
- [16] Ticknor, L. J. (2013). A Bayesian regularized artificial neural network for stock market forecasting. *Expert Systems with Applications*, 40, 5501-5506. <https://doi.org/10.1016/j.eswa.2013.04.013>
- [17] Mousazadeh Abbassi, N., Aghaei, M. A., & Moradzadeh Fard, M. (2014). An integrated system based on fuzzy genetic algorithm and neural networks for stock price forecasting: Case study of price index of Tehran Stock. *International Journal of Quality & Reliability Management*, 31(3), 281-292. <https://doi.org/10.1108/IJQRM-06-2012-0085>
- [18] Xi, L., Muzhou, H., Ho Lee, C., Li, J., Wei, C., Hai, H., & Wu, Y. (2014). A new constructive neural network method for noise processing and its application on stock market prediction. *Applied Soft Computing*, 15, 57-66. <https://doi.org/10.1016/j.asoc.2013.10.013>
- [19] Hagan, M. T., Demuth, H. B., & Beale, M. H. (2002). *Neural network design*. Martin Hagan.
- [20] Yen Chen, M., Ren Chen, D., Hsuan Fan, M., & Ying Huang, T. (2013). International transmission of stock market movements: An adaptive neuro-fuzzy inference system for analysis of TAIEX forecasting. *Neural Computing and Applications*. <https://doi.org/10.1007/s00521-013-1461-4>
- [21] Mohandes, M., Rehman, S., & Rahman, S. M. (2011). Estimation of wind speed profile using adaptive neuro-fuzzy inference system (ANFIS). *Applied Energy*, 88, 4024-4032. <https://doi.org/10.1016/j.apenergy.2011.04.015>

- [22] Wei, L. Y. (2013). A hybrid model based on ANFIS and adaptive expectation genetic algorithm to forecast TAIEX. *Economic Modelling*, 33, 893-899. <https://doi.org/10.1016/j.econmod.2013.06.009>
- [23] Yen Chen, M., Ren Chen, D., Hsuan Fan, M., & Ying Huang, T. (2013). International transmission of stock market movements: An adaptive neuro-fuzzy inference system for analysis of TAIEX forecasting. *Neural Computing and Applications*, 23, 369-378. <https://doi.org/10.1007/s00521-013-1461-4>