Assignment 4: Algorithmic Trading

David Smith

March 2019

1 Introduction

This assignment considers the use of a deep learning method to construct an automated trading system. This type of trading, known as algorithmic trading, involves automatically generating orders on an exchange based on certain price ranges or events taking place. Being able to accurately predict these trigger points could allow profitable trades to be made, and is therefore an area of great interest today.

2 Context

This project considers the use of an Artificial Neural Network (ANN) in predicting the daily high and low prices for a given stock. These predictions will be incorporated into trading rules which provide a daily buy, sell or hold recommendation.

ANNs are computational modelling tools which have experienced use in many fields, including the financial sector [1], with a variety of different applications and uses emerging. The use of ANNs in exclusively predicting daily high and low prices is not as ubiquitous, however.

Guresen et. al demonstrate the effectiveness of multi-layer perceptrons (MLP) and dynamic ANNs in predicting daily close prices [2], showing the success of MLPs over other hybrid model configurations. A trading system is developed by Rodríguez-González et al. where ANNs are used to predict the Relative Strength Index (RSI) trading signal of a security, demonstrating accuracy results of 63% [3].

Lin and Lin considered macroeconomic variables such as Gross Domestic Product as inputs to an ANN for predicting average change in the Dow Jones index [4]. Whilst reporting error rates of under 7%, their methodology could not be used for investment purposes. Returning to microeconomics, White applied an ANN to predict the daily returns of IBM stock, where he discovered problems with the non-linearity of time series data [5], suggesting networks be evaluated in terms of profit and loss instead of return values.

This project will pool certain aspects of the aforementioned cases to address an application of ANNs not covered extensively: predicting high and low prices in order to inform on whether to buy or sell a stock. These predictions will be used as the basis of trading rules, along with other performance metrics and indicators. Similar to the projects described above, this project will use ANNs of similar architectures, including multiple hidden layers, involve the use of technical signals such as RSI and consider evaluation of trading based on resultant profit and loss.

3 Data

The project would use market data on Tesla (TSLA) stock, an American automotive and energy company. The data was downloaded from Yahoo Finance and consisted of time series data containing daily open, high, low, closing and adjusted prices, as well as share volume. Using this data, we would extrapolate several technical indicators to include in the project. These were: Exponential Moving Average (EMA), Relative Strength Index (RSI) and the Simple Moving Average (SMA).

We would be looking at Tesla stock performance over a 2 year period, between 2016 and 2018. Summary statistics for the stock can be seen in Figure 1.

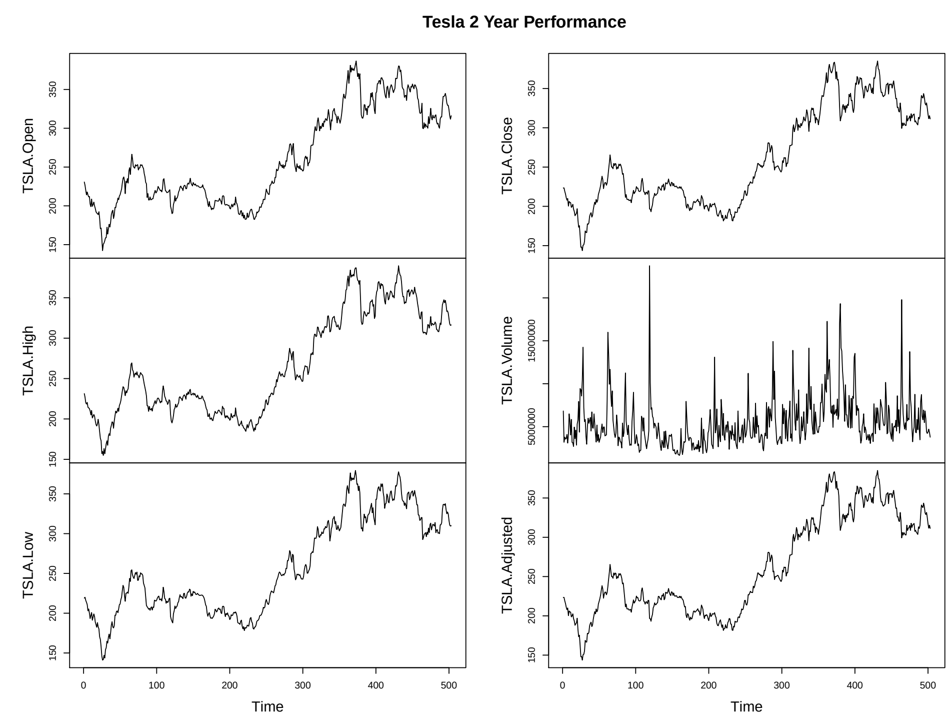


Figure 1: Tesla stock volume level and open, close, high, low and adjusted prices from 01/01/2016 to 01/01/2018.

A substantial increase in value can be observed over the two year period. An initial investment in January 2016 would yield a 139.4% return by the end of 2017.

There is some volatility in early 2016, with a severe drop in value in early 2016 followed by an immediate sharp rise. This dissipates by late 2016 where a steady rise is observed, increasing to a maximum of $385 per share in May 2017.

This project is aiming to predict daily values for the high and low prices over the final 150 days (approx.) of the period shown. Over this time we can see high volatility and no clear trend, as in the earlier regions of the period.

Along with the performance of the stock attributes shown, the project would also be considering EMA, SMA and RSI indicators. The EMA is a moving average which places a higher importance on more recent prices. This would be useful in predicting each days high and low prices. The EMA was calculated from the adjusted closing price over 10 and 30-day periods. The SMA is a standard moving average of the daily low price over 10 and 30 days. This would inform on more general trends in the prices. The RSI is a momentum indicator which monitors over-bought and over-sold levels within a stock. This would be a useful indicator as these scenarios could heavily influence daily high and low prices.

4 Implementation

4.1 Packages

The project involved using a powerful R neural network package called *neuralnet*. The package provided full customization of the neural network hyperparameters and configuration of different model architectures.

4.2 Representation

Our aim was to achieve accurate predictions for the high and low price of Tesla stock for an unseen period. Therefore our network first had to be trained with known high and low prices. To do this, the data had to be transformed into a shifted observation, or lagged, matrix. This was due to the nature of the task: for a given day, we wanted *that* day's high and low price based on *previous* days information. Therefore it was intuitive to train the network over each known high and low price and their associated previous days data. The network would use this previous performance data to give an estimate for the "current" prices.

The lag matrix was created where each row consisted of one day's high and low price and high, low, open, adjusted, volume, RSI, EMA and SMA values over the three days previous. The current day opening price was also included. It is worth noting that the the indicators RSI, EMA, SMA were calculated over periods of 10 to 30 days, but would have changing daily values.

This lagged matrix was scaled and split into training and testing periods. The high and low price columns were designated as the network outputs, with the lagged data used as inputs. The network would train over 18 months worth of stock performance to learn the relationship between the high and low prices each day and the previous days stock performance. The network could then be tested for a 6 month period to check the quality of the predictions.

4.3 Trading Rules

Trading rules would be established based on the predictions given by the neural network. These can be seen below.

Rule	Condition	Action	Dependencies
1	Price enters margin of 1% around low price prediction.	Buy	2,3,4
2	Low price prediction below the 10-day EMA of previous low prices.	Buy	1,3,4
3	Balance is not zero.	Buy	1,2,4
4	Not the last day of trading.	Buy	1,2,3
5	Price enters margin of 1% around high price prediction.	Sell	5
6	Own stocks.	Sell	4
7	Own stocks on the last day of trading.	Sell	None
8	Sell price greater than buy price from last buy.	Sell	5,6,7
9	No rules satisfied.	Hold	1-7

Table 1: Trading rules with corresponding actions and dependencies.

We can see above the trading rules for buying, selling and holding the stock. Note: at each buy or sell, the entire amount of stock would be exchanged. Rules 1 and 5 have a low margin around the predictions - this was due to expected accuracies of the predictions and was open to change. Rule 2 is based on the concept of the EMA indicating upcoming direction changes. If a stocks current price is lower than its average, it could be assumed it was about to increase and rise again. Therefore it would be advantageous to buy at this point. Rule 3 ensures we buy only when we have capital to spend, ensuring we cannot sell multiple times in a row. Similarly Rule 6 prevents repeated purchases taking place. Rule 7 ensures we do not finish the trading period owning any stocks. Rule 8 ensures sales are only made when they are profitable.

4.4 Evaluation

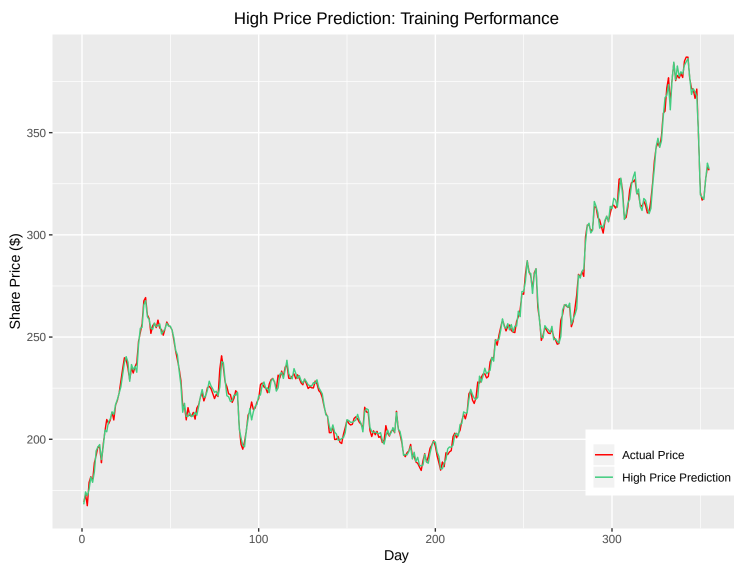
We would be evaluating the performance of the neural network using the root mean squared error (RMSE). This is a suitable metric for regressive prediction and would help us evaluate the training performance of the model, before trading on the testing data. The performance of the trading strategies, based on the network predictions, would be evaluated on how much profit is gained.

This would be done by setting an initial amount of capital and observing on what days during the 6-month test period the trading rules are satisfied. On these days, the resultant action taken would increase or decrease our level of capital. At the end of the 6 month period, we could determine the profit made by subtracting the initial amount from the final balance.

5 Training the Neural Network

This section discusses the performance of the neural network over the training period. The model was given a 3-day lagged matrix of previous stock performance as inputs, with target outputs set as the high and low price of the current day. The architecture of the network was made up of two hidden layers of 15 nodes each. There was no considerable variation in performance when altering this configuration, therefore it was kept constant.

The neural network produced predictions with an overall RMSE of 0.6192. This error was small, considering the average share price of the training period was $221.2. We can visualize the spread of individual low and high price predictions across the period below.



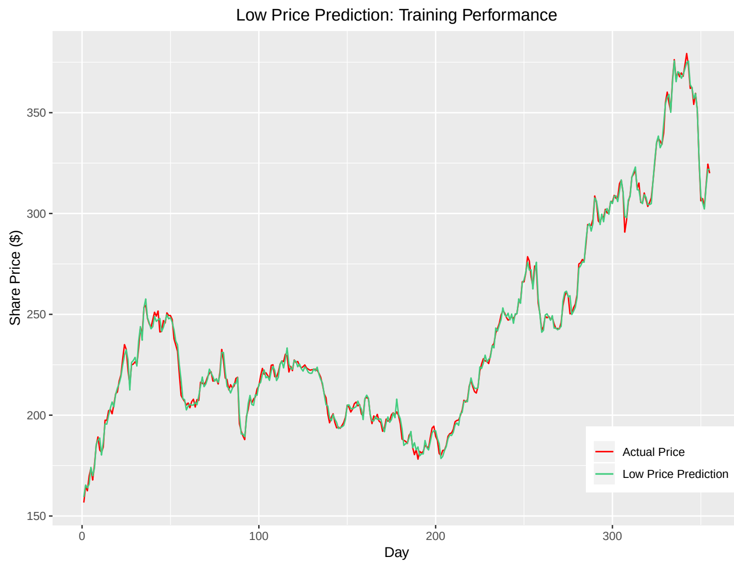


Figure 2: High and low price predictions against the actual stock performance over the training period.

As can be seen in Figure 2, the model fit to the training data to a high degree of accuracy. Whilst this is to be expected during training, as the model is aware of the outputs, caution must be taken to avoid overfitting the model on the training data. This could be detrimental to performance when trying to generalize over the testing period.

6 Performance over Test Data

This section discusses the performance of the neural network over the testing period and the deployment of the trading rules mentioned in Section 4 of this report. We can visualize the performance over this period in Figures 3,4 and 5 below.

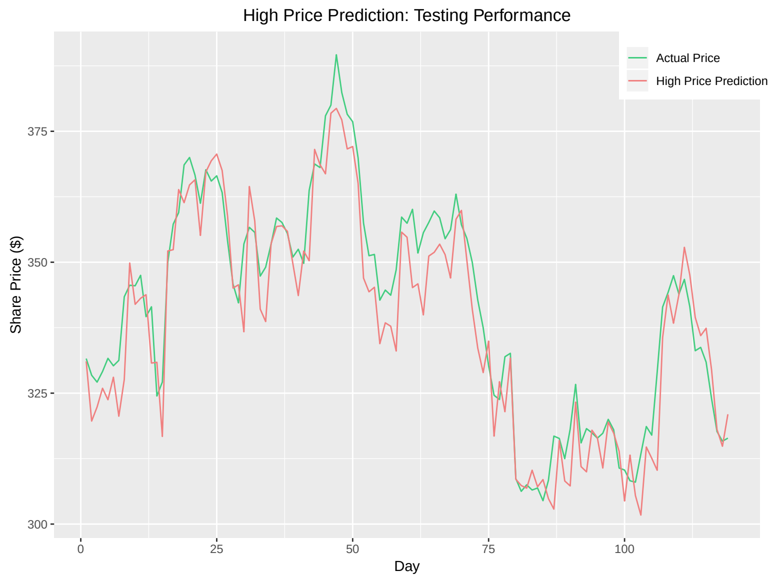


Figure 3: High price predictions against the actual stock performance over the unseen testing period.

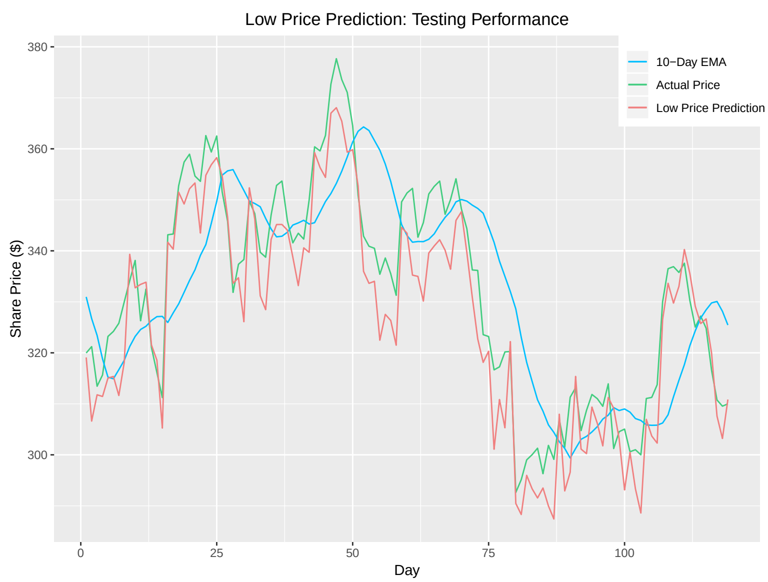


Figure 4: High price predictions against the actual stock performance over the unseen testing period.

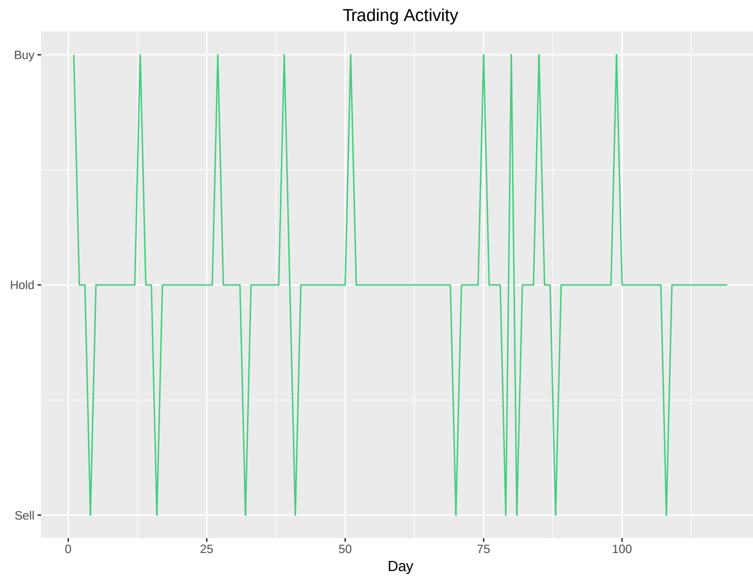


Figure 5: High price predictions against the actual stock performance over the unseen testing period.

We can see above the high and low price predictions against the actual Tesla stock performance. In both figures, the jagged trajectory of the predictions indicate a degree of overfitting within the model. This can be observed between Day 25 and 40 and also Day 80 to 100 of the period. Overall, the predictions are fairly close to the actual stock levels. The main trends of the stock performance are mirrored in the predictions: a sharp drop in both prices around Day 17 going up to a peak around Day 50 and dropping to just above $300 around Day 80.

With RMSEs of 6.80 and 7.61 for the high and low price respectively, the overall error for the predictions is low. This error level helped to inform the creation of the previously mentioned trading rules. If we knew the actual price would be close to the predictions then the margin around the prediction, which would trigger buying and selling, could be small. Therefore the margins were set at +/- 1%. If the price of the stock on the day fell within this margin, an action would be triggered (referring to Rules 1 and 5 which depend on prerequisite rules being satisfied).

Figure 5 shows the timeline of the trading decisions made over the testing period. Over the period, 18 trades were made: 9 buy and 9 sell, to produce a profit of $5683, or 56.8%.

The rules enforced a fairly complex trading system. Several conditions had to be met before a buy or sell could take place. The buy action prerequisite depending on a current price lower than the 10-day EMA had a large effect. We can see this effect in the lower frequency of trades taking place in the first 25 days, where the actual price is undercut by the EMA. This is indicative of an upcoming downwards trend and would not be an optimum time to purchase during. The system does make some trades, buying whenever the actual price dips below the EMA and crosses the predicted prices, and sells instantly once it rises above.

This is reversed in the period between Day 50 and Day 80. Here, the EMA skirts mainly above the actual price. Also, the predicted and actual prices frequently overlap during this period. Therefore several buy-sell trades occur within quick succession. The predictions align, a buy occurs, and as the EMA is around $20 above, the stock is sold whenever the predictions align at a point of higher value than the entry point.

7 Comparison

To validate the performance of the neural network, a comparative method was carried out. As an alternative to using deep learning to provide predictions, the predictions would be formed from the 10-day average of the previous high and low prices. These would then be used as the basis of the same trading rules as used with the neural network predictions. The performance of this method can be visualized below.

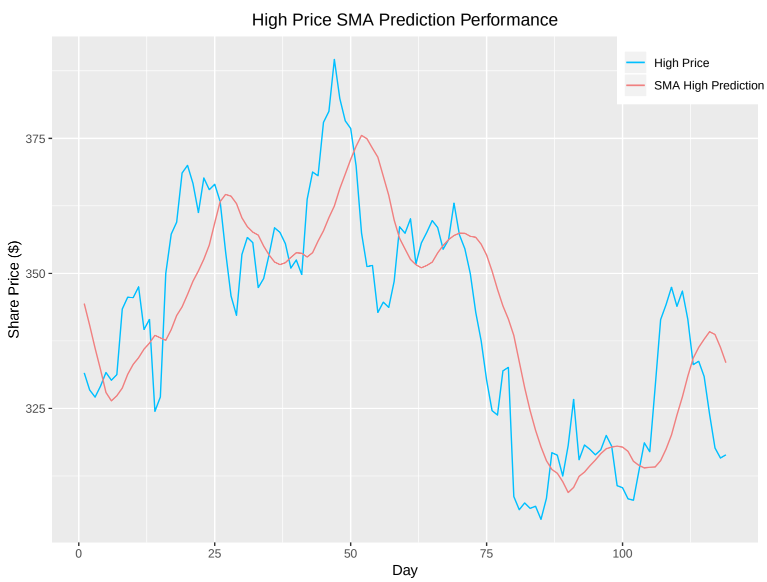


Figure 6: High price predictions against the actual stock performance over the unseen testing period.

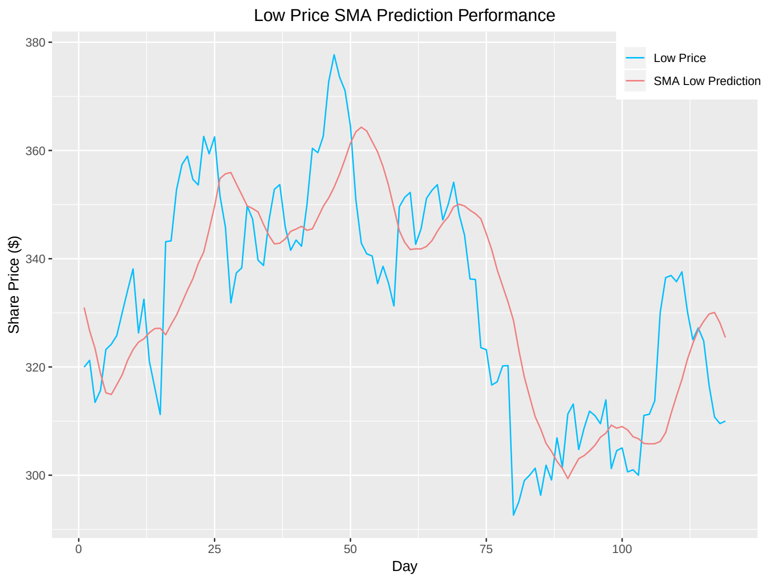


Figure 7: High price predictions against the actual stock performance over the unseen testing period.

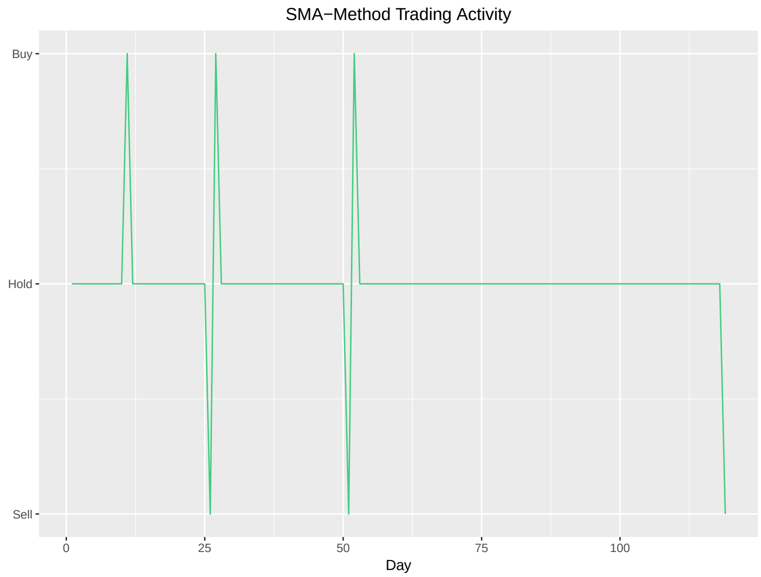


Figure 8: High price predictions against the actual stock performance over the unseen testing period.

Figures 6 and 7 show the high and low prices compared with the 10-day SMA predictions. As expected, the SMA prediction generally follows the trends of the stock. However, the SMA predictions do not pick up any change that is more than moderate on a given day. There are very few occurrences of overlap between the actual price and the SMA prediction.

Due to this, the trading rules are barely satisfied. The prediction derived from the SMA rarely gets within 1% of the actual price. Furthermore, not all of the overlaps that do occur actually allow for a profitable trade to be made (Rule 8). Therefore a small amount of trades were carried out. At the end of the trading period, 6 trades were carried out: 3 buys and 3 sells. Only 2 of these buy/sell pair trades were profitable, with the final sell causing a loss of $443.5 (Rule 7 forces a sell if stocks still held on last day).

The timeline of these exchanges can be seen in Figure 8. We can see the trades occurring when the SMA predictions enter the action margin around the actual prices, assuming a profit can be made. Day 51 induces a buy action as the SMA cross the actual values at around $360. However, no sell can be carried out as whenever the SMA next enters the margin, the sell price is lower than the buy price. This remains until a sell is forced on the final day of trading. The trading strategies fail as a result of the poor predictions from the mean method.

References

[1] A. Bahrammirzaee, "A comparative survey of artificial intelligence applications in finance: artificial neural networks, expert system and hybrid intelligent systems," *Neural Computing and Applications*, vol. 19, no. 8, pp. 1165–1195, 2010.

[2] E. Guresen, G. Kayakutlu, and T. U. Daim, "Using artificial neural network models in stock market index prediction," *Expert Systems with Applications*, vol. 38, no. 8, pp. 10 389–10 397, 2011.

[3] A. Rodríguez-González, Á. García-Crespo, R. Colomo-Palacios, F. G. Iglesias, and J. M. Gómez-Berbís, "Cast: Using neural networks to improve trading systems based on technical analysis by means of the rsi financial indicator," *Expert systems with Applications*, vol. 38, no. 9, pp. 11 489–11 500, 2011.

[4] M. Lam, "Neural network techniques for financial performance prediction: integrating fundamental and technical analysis," *Decision support systems*, vol. 37, no. 4, pp. 567–581, 2004.

[5] H. White, "Economic prediction using neural networks: The case of ibm daily stock returns," 1988.