STOCK PRICE PREDICTION USING MACHINE LEARNING: AN APPLICATION OF THE RANDOM FOREST MODEL

EMRE AKTÜRK

INTRODUCTION

Forecasting how stock prices will move in the future is a subject that many traders are interested in. In this regard, various strategies can be adopted to choose which stock to invest in. Some common approaches are fundamental analysis which determines the fair value of the company by analyzing its financial statements, technical analysis which examines the statistical trends of the stock and industry analysis which evaluates the industry that the company is operating in to forecast its future stock price movement.

The aim in this project is to employ machine learning (ML) to predict the daily movement of stock prices. The advancements in machine learning together with vast amount of historical data present an exciting opportunity to analyze past movements, identify patterns and thus make predictions regarding the future movement of stock prices.

In this regard, I built a machine learning model that predicts the daily movement of the price of Boeing stock. The model predicts the direction of movement instead of the exact price. This approach is preferred for two reasons. Firstly, many studies reveal that it is very difficult to predict the exact price[1] and secondly it is sufficient to know the direction of movement to be successful in trading.

[1] See Related Literature section for details.

A range of machine learning methods can be adopted to predict how stock prices will move. The model preferred in this project is the Random Forest Model (RF). This model is suitable for this assignment for several reasons. Firstly, RF is able to capture non-linear relationships which is crucial because stock price data usually is non-linear (Vijh et al., 2020). Secondly, this model is less likely to exhibit overfitting. Thirdly, many studies show that RF is more successful than other models in predicting stock prices[2].

In this project, initially a simple model is built using nominal values of Boeings daily opening and closing price, intraday day high and low prices, daily volume, stock splits and dividends as well as the S&P 500 index as the predictors. The model is trained and tested over the last 150 days. However, to increase confidence in the result backtesting is conducted. The result reveals that this initial model has a very low predictability (around 50%). Then, the models predictors, parameters and structure are adjusted with the aim of increasing predictability. In this regard, the rate between the current closing price and rolling averages of closing price, rolling trend and the rate between closing price and S&P 500 index are used as predictors. Moreover, n_estimators is increased and min_sample_split is decreased. Also, the model is adjusted such that it predicts that the stock price will increase if the probability of increasing is found to be at least 65%. The result reveals that the improved model has a much higher predictive capability (precision is around 60%). However, there is the tradeoff that the number of price increase predictions are lower. But this can easily be overcome by applying this model to different stocks and thus obtaining more daily predictions.

[2] See Related Literature section for details.

RELATED LITERATURE

Predicting stock prices using machine learning has been studied extensively in the literature. Due to its chaotic nature and high volatility, it is very difficult to predict the exact value of a stocks future price (Khaidem et al., 2016). Nevertheless, there is evidence that the prediction of the stock price direction is more successful (Sadorsky, 2021). Thus, in this project I predict the direction of the movement of stock prices instead of the exact price.

The model used is the Random Forest Model. Many studies find that RF is a suitable model for stock price prediction. For example, after comparing several methods, Polamuri et al. (2019) find that Decision Tree and Random Forest Regressor are the best models for stock price prediction. Similarly, Sadorsky (2021) finds that Random Forest predictions of stock price direction have a good performance and is more accurate than those obtained from logit models. Moreover, even though RF models have shown a good track record in stock price predictions, Sardosky (2021) finds that they appear to be underutilized in the existing literature, which is also a motivation to use this model.

MODEL

Firstly, yfinance is installed and imported. This package calls Yahoo Finance API to download daily stock information. In this project the stock price direction of Boeing (BA) is predicted because as it is one of the oldest companies in S&P 500 stock exchange it has vast amount of daily historical data which can be utilized. Thus, after installing yfinance the stock information of Boeing is obtained by specifying its ticker symbol, “BA”. The historical data for Boeing is selected to cover the maximum period available, which is from 1962 January 2 until the present day. The information obtained are opening price (“Open”), intraday high (“High”), intraday low

("Low), closing price ("Close"), volume ("Volume"), dividends ("Dividends") and stock splits ("Stock Splits"). Table 1 shows part of this data. After downloading the information of Boeing, the information for S&P 500 index is downloaded using its ticker ^GSPC. However, to be consistent with Boeing data, the historical information obtained is set to start from 1962. For S&P 500, only the closing price variable is needed thus the remaining variables are dropped. Then, the Boeing data and S&P 500 data are concatenated.

Date	Open	High	Low	Close	Volume	Dividends	Stock Splits
1962-01-02 00:00:00-05:00	0.194272	0.194272	0.190931	0.190931	352350	0.0	0.0
1962-01-03 00:00:00-05:00	0.193795	0.197614	0.193795	0.194749	710775	0.0	0.0
1962-01-04 00:00:00-05:00	0.194749	0.198091	0.192840	0.192840	911250	0.0	0.0
1962-01-05 00:00:00-05:00	0.192840	0.193795	0.183771	0.189022	880875	0.0	0.0
1962-01-08 00:00:00-05:00	0.189022	0.192363	0.186635	0.189499	473850	0.0	0.0
...	...	...	...	...	...	...	...
2024-01-08 00:00:00-05:00	228.000000	233.850006	225.789993	229.000000	40730400	0.0	0.0
2024-01-09 00:00:00-05:00	225.660004	228.789993	223.199997	225.759995	20687500	0.0	0.0
2024-01-10 00:00:00-05:00	226.899994	231.610001	226.639999	227.839996	12883700	0.0	0.0
2024-01-11 00:00:00-05:00	228.070007	228.279999	222.619995	222.660004	11830500	0.0	0.0
2024-01-12 00:00:00-05:00	219.970001	222.070007	217.039993	217.699997	11268800	0.0	0.0

Table 1

In this project, the model will predict if daily stock prices will increase or decrease. Thus, it is important to know how the price has evolved historically. Moreover, it is useful to know how the dynamics of the stock's parameters has evolved overtime. Thus, Figure 1 shows the evolution of the closing price, volume, dividends and stock splits of Boeing stock from 1962 until the present day.

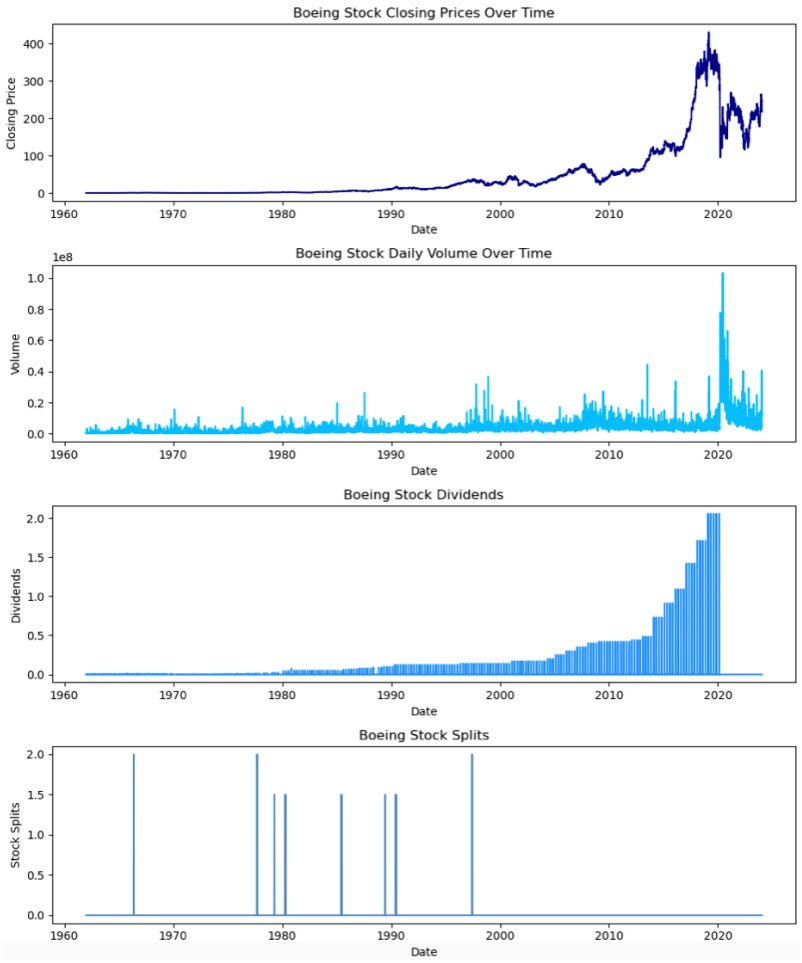

Figure 1

Next, the data must be cleaned and adjusted such that it can have a format that would enable us to build the desired machine learning model. Firstly, as the model will predict if the stock price goes up or down at a daily basis, it is important to have the information for each day on whether the price increases or decreases the next day. To have this information two new columns are created. The first is called “Tomorrow” which gives tomorrows closing price for each day. The second new column is called “Direction” which compares today’s closing price with tomorrows closing price and returns Rise if tomorrow is higher (so if the stock price increases) and Fall if it is lower (so if the stock price decreases). However, for convenience in

coding the Direction variable is converted to numeric binary such that “Rise” is 1 and “Fall” is 0. Table 2 shows part of the data with the changes in the variables.

Date	Open	High	Low	Close	Volume	Dividends	Stock Splits	Close (sp500)	Tomorrow	Direction
1962-01-02 00:00:00-05:00	0.194272	0.194272	0.190931	0.190931	352350	0.0	0.0	70.959999	0.194750	1
1962-01-03 00:00:00-05:00	0.193795	0.197614	0.193795	0.194750	710775	0.0	0.0	71.129997	0.192840	0
1962-01-04 00:00:00-05:00	0.194750	0.198091	0.192840	0.192840	911250	0.0	0.0	70.639999	0.189022	0
1962-01-05 00:00:00-05:00	0.192840	0.193795	0.183771	0.189022	880875	0.0	0.0	69.660004	0.189499	1
1962-01-08 00:00:00-05:00	0.189022	0.192363	0.186635	0.189499	473850	0.0	0.0	69.120003	0.189976	1
...	...	...	...	...	...	...	...	...	...	...
2024-01-05 00:00:00-05:00	245.039993	250.190002	245.039993	249.000000	3846200	0.0	0.0	4697.240234	229.000000	0
2024-01-08 00:00:00-05:00	228.000000	233.850006	225.789993	229.000000	40730400	0.0	0.0	4763.540039	225.759995	0
2024-01-09 00:00:00-05:00	225.660004	228.789993	223.199997	225.759995	20687500	0.0	0.0	4756.500000	227.839996	1
2024-01-10 00:00:00-05:00	226.899994	231.610001	226.639999	227.839996	12883700	0.0	0.0	4783.450195	222.660004	0
2024-01-11 00:00:00-05:00	228.070007	228.279999	222.619995	222.660004	11830500	0.0	0.0	4780.240234	217.699997	0

Table 2

With the adjustments made in the variables, the data is ready to form the machine learning model. The model used in this project is the Random Forest Model. The reason for choosing this model is its ability to capture non-linear relationships and its resistance to overfitting. Moreover, many studies reveal the success of RF over other machine learning models in predicting stock prices. First a simple model will be built, then the model will be improved.

In the initial model, n_estimators[3] (the number of individual decision tree’s trained) and min_sample_splits[4] (minimum number of samples required to split a node in a decision tree) are chosen to be 100 and the Random State is set to 42. Random State is chosen to be 42 (setting it to any specific number would be sufficient) so that when the same model is run twice, the numbers

[3] Higher n_estimators will give higher accuracy but can cause overfitting.
[4] Higher min_sample_split can protect against overfitting but can reduce accuracy.

generated will be in a predicted sequence. This is crucial because when the model is updated and improved we will be certain that the change is not random.

After setting the model parameters, the training and test sets and the predictor and target variables are determined. Initially, the training set is chosen to be all observations excluding the last 150 while the test set is chosen to be the last 150 observations. Later, this will be improved and backtesting will be conducted. The predictors of the model are chosen to be the rows "Open", "Close", "Volume", "High", "Low", "Dividends", "Stock Splits" and "Close (sp500)" while the target variable is chosen as the row "Direction".

Then, the model is trained and tested. Table 3 shows the predicted direction, actual direction and whether the prediction was correct for part of the predictions. Figure 2 shows the actual direction values in green and predicted direction values in blue.

Date	Direction	Predicted Direction	Comparison
2023-06-08 00:00:00-04:00	0	0	Correct
2023-06-09 00:00:00-04:00	1	0	Wrong
2023-06-12 00:00:00-04:00	0	0	Correct
2023-06-13 00:00:00-04:00	0	0	Correct
2023-06-14 00:00:00-04:00	1	0	Wrong
...	...	...	...
2024-01-05 00:00:00-05:00	0	0	Correct
2024-01-08 00:00:00-05:00	0	0	Correct
2024-01-09 00:00:00-05:00	1	0	Wrong
2024-01-10 00:00:00-05:00	0	0	Correct

Table 3

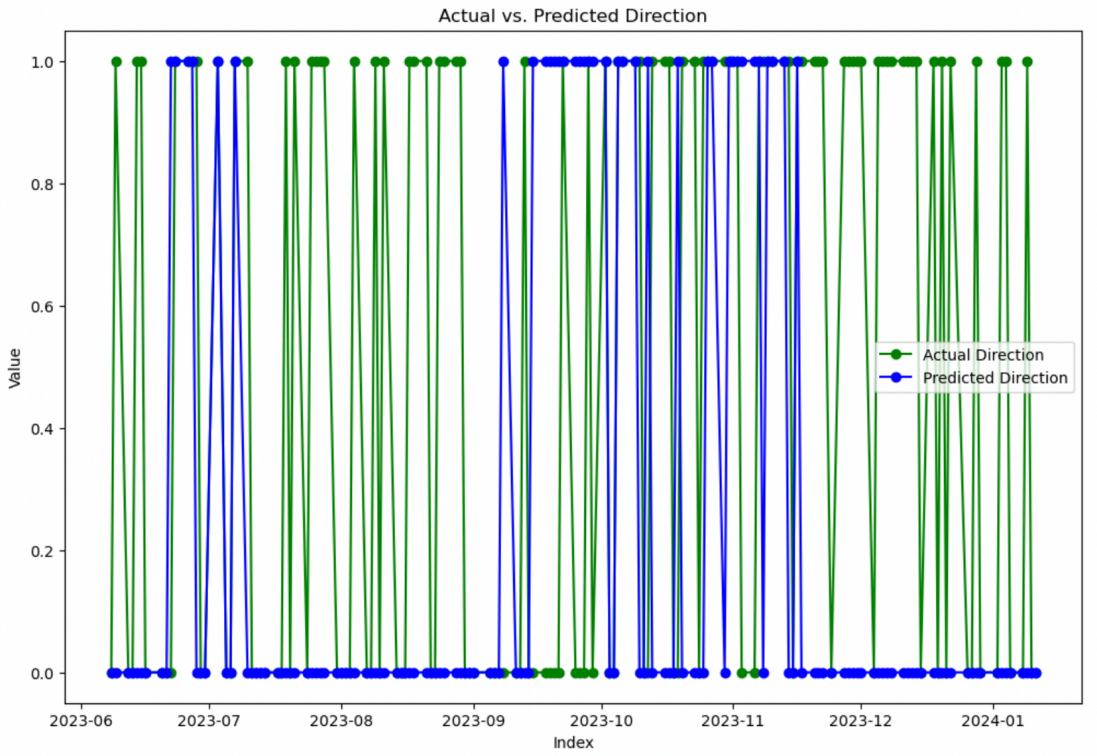

Figure 2

Next, the predictability and accuracy of the model must be assessed. This is done by examining the precision and rate of correct predictions. Firstly, precision measures the accuracy of positive predictions made by the model. Specifically, it reveals the rate of instances in which the price actually rose when the model forecasted an increase. The precision obtained by this model is only around 0.55 which is not very high. Secondly, we evaluate the rate of correct predictions. This rate is around 50%, which is not high. However, the model is currently only tested against the last 150 days, which, given the large dataset, is very limited and it prevents us from being confident about the results. Thus, a more robust testing algorithm is built which tests across multiple years of data. This practice, which is popular in financial modelling, is called backtesting.

The backtesting function will iteratively train and test the model on rolling windows of data. In this regard, the function will take the first 5 years of data and predict values for the 6th year, then take the values of the first 6 years of data and make predictions for the 7th year and so on.

Next, the predictiveness of the model is once again evaluated using backtesting. Now, we obtain daily predictions for 1966 onwards. We do not have predictions for earlier dates since the start value is set to 1250[5]. We once again calculate the accuracy and rate of correct predictions. The precision is lower now, it is around 0.5 and rate of correct predictions is also around 50%. Thus, we can say that this initial simple model has a low predictability. Moreover, now that we have backtested we can be confident about these results. Next, the model is improved by adding new predictors, changing some parameters and updating the structure of the prediction function with the aim of increasing its predictability.

Firstly, a shortfall of the initial model predictors was that predictions relied only on features the Open, High, Low, Close, Volume, Dividends, Stock Splits and Close (sp500) which were not sufficient to make successful predictions. To overcome this issue and achieve a more advanced model, new predictors can be derived using existing variables (Vijh et al., 2020). The new variables will be such that they will not just look at the individual nominal values of past data, instead, they will look at various rolling time periods and rates. More specifically, rolling averages for closing prices and rolling trends will be created and the rate between the current days closing price over the rolling average, the rolling trend and the rate between Boeings closing price and S&P 500's closing price will be used as predictors in the advanced model. The horizons for these rolling values will be 2 days, 5 days (a trading week), 20 days (a trading

[5] A trading year is around 250 days. Thus as the starting date is 5 years, the start value is set to 1250.

month), 60 days (a trading quarter), 250 days (a trading year), 500 days (two trading years) and 1000 days (four trading years). With these new variables, the first observation starts from 1965 December 21. This is because the rolling averages and rolling trends require a certain number of days (1000 for the highest one) to be able to be calculated. Thus, as the rows with NaN are deleted, to not encounter any missing values, the observations start from December 1965. Figure 3 visualizes the values for close ratios overtime, Figure 4 visualizes the values for trends overtime and Figure 5 visualizes the rate between Boeings closing price and S&P 500's closing price overtime.

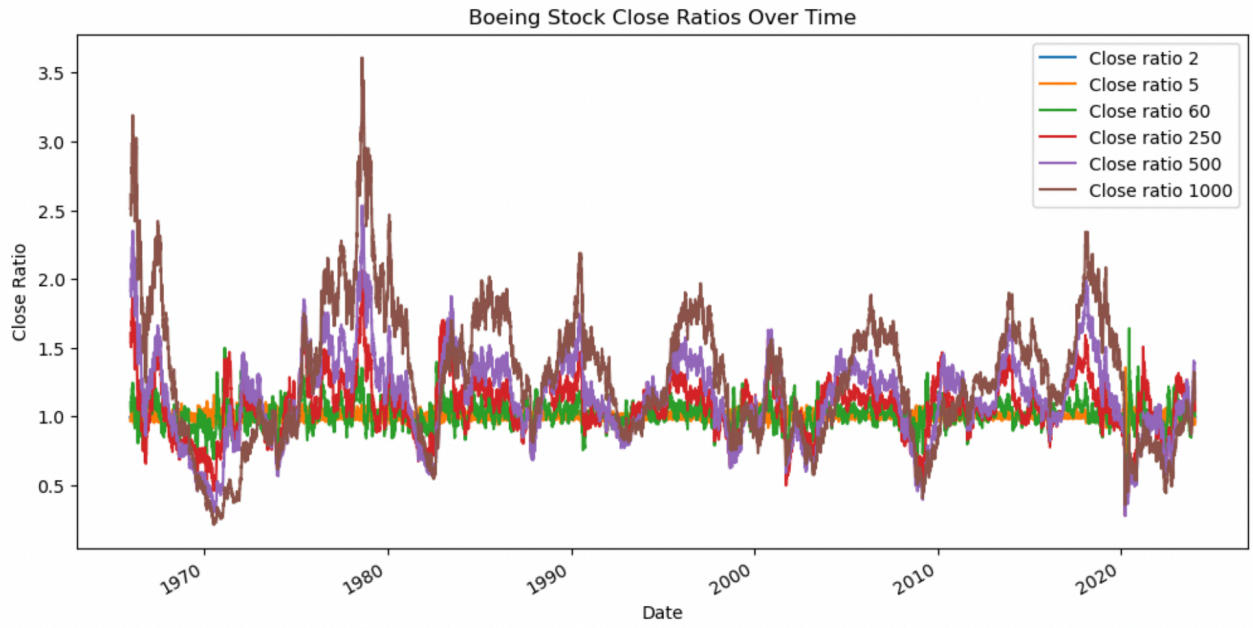

Figure 3

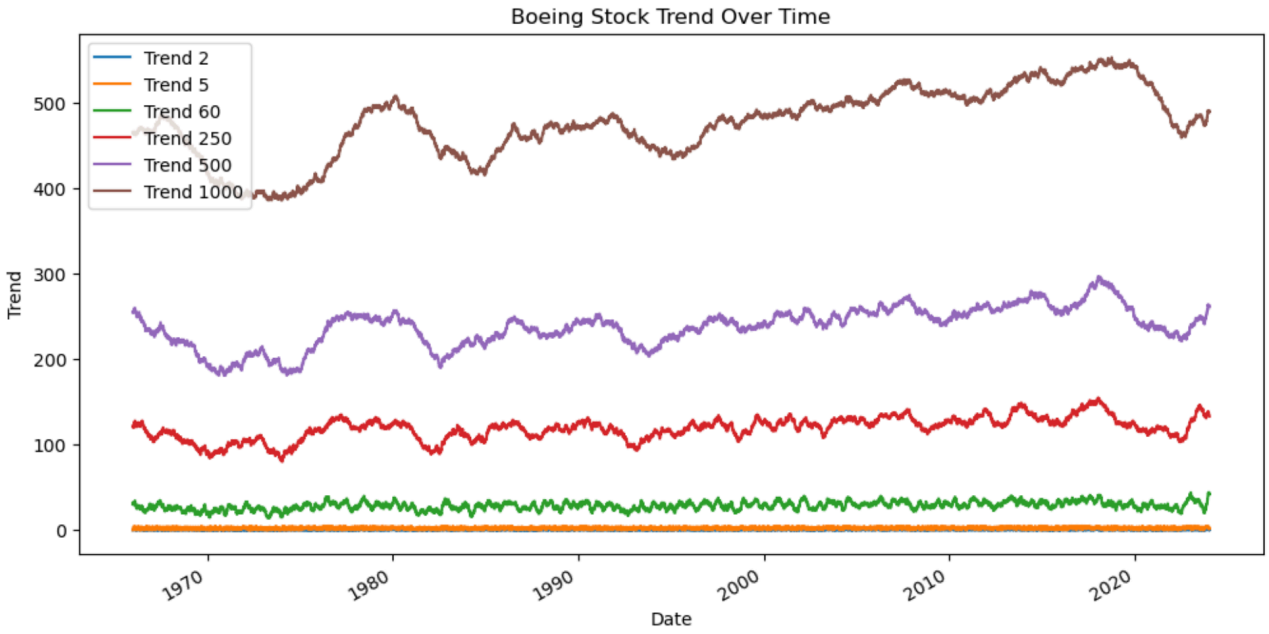

Figure 4

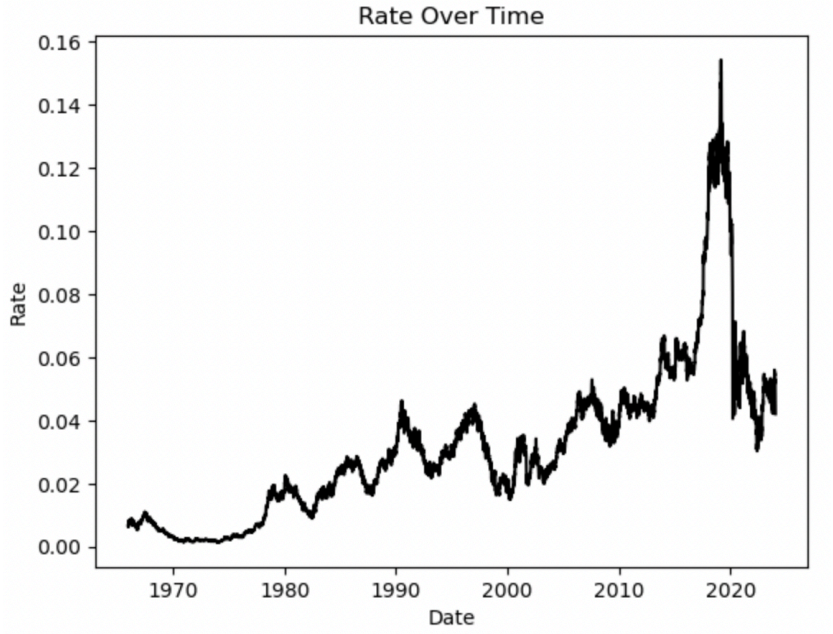

Figure 5

To see more explicitly how the rolling values are calculated, for the calculation of Close Ratio, consider Close Ratio 2 in 1965-12-22. This value comes from (Closing Price of Boeing in

1965-12-22)/((Closing Price of Boeing in 1965-12-21 + Closing Price of Boeing in 1965-12-22)/2) . For Trend calculations consider Trend 2 in 1965-12-23. This value comes from (Direction in 1965-12-21) + (Direction in 1965-12-22). For Rate, consider Rate in 1965-12-22. This value comes from (Closing price of Boeing in 1965-12-22)/(Closing price of S&P 500 in 1965-12-22).

It is important to note that in Close Ratio the value of the current day is included in the calculation, but this is not the case for Trend calculations. The reason for this is that Trend is based on the comparison of today and tomorrow's price. Thus, if we include the current day of Trend calculation and then use Trend as a predictor this would cause leakage.

Another improvement to the model is made through changing the models parameters. In order to improve the accuracy of the model n_estimators is increased to 300 and min_samples_split is reduced 50, which were chosen after extensive trails.

Another change to the model is done regarding the prediction function. In the initial prediction function .predict was used. With .predict only a 1 or 0 indicating if the stock price went up or down was obtained. Now, instead of .predict, .predict_proba is used. With .predict_proba, instead of getting a 1 or 0, a probability of the stock price movement is obtained. This adds flexibility to the model as the threshold percentage can be adjusted. In this regard, in the adjusted prediction function, .predict_proba is chosen to be 0.65 which indicates that the model will return that the price will go up if the probability of going up is at least 65%. With this specification it is expected that the total number of days predicted that the price will go up will be reduced but the accuracy that the price goes up in those days will increase.

Next, the predictiveness of the improved model is evaluated. In this regard, a rolling backtest is performed using the new model and new predictors. The result reveals that while the

number of days the model predicted that the stock price will increase is much lower (around 60), the precision is much higher (around 0.6). This finding was what was expected. Thus, with the changes made to the model, the models predictive capability has increased. It must be noted that even though 60 predictions may seem low given the long horizon, this model used only one stock (Boeing). By applying the same model to multiple stocks, the number of price increase predictions obtained will rise.

CONCLUSION

In this assignment I built a Random Forest Model that predicts the direction of the movement of the stock price of Boeing. Random Forest was chosen because it can capture non-linear relationships and it is less likely to overfit. Moreover, many previous studies found that RF is more successful than other models in predicting future price movements.

Initially a simple model was built which takes individual past nominal variables as predictors. Then the model was improved such that new variables were derived out of the existing ones and these were used as predictors. Moreover, the models parameters and structure were adjusted with the aim of increasing predictability.

The finding revealed that even though the initial simple model had a very low predictability, the improved version had considerable predictability. The improved model had a precision of around 0.6. Even though the number of price increases seems low in the final model, this study only considers one stock. By applying this model to multiple stocks, the number of predictions will increase. Therefore, as there are thousands of stocks all around the world, this model has the potential to guide traders in choosing stocks.

REFERENCES

Khaidem, L., Saha, S., & Dey, S. R. (2016). Predicting the direction of stock market prices using random forest. arXiv (Cornell University). https://arxiv.org/pdf/1605.00003.pdf

Polamuri, S. R., Srinivas, K., & Mohan, A. K. (2019). Stock Market Prices Prediction using Random Forest and Extra Tree Regression. International Journal of Recent Technology and Engineering, 8(3), 1224–1228. https://doi.org/10.35940/ijrte.c4314.098319

Sadorsky, P. (2021). A random forests approach to predicting clean energy stock prices. Journal of Risk and Financial Management, 14(2), 48. https://doi.org/10.3390/jrfm14020048

Vijh, M., Chandola, D., Tikkiwal, V. A., & Kumar, A. (2020). Stock Closing Price Prediction using Machine Learning Techniques. Procedia Computer Science, 167, 599–606. https://doi.org/10.1016/j.procs.2020.03.326