Stock Price Predictions for FAANG Companies Using Machine Learning Models

Authors

Fredrik Fourong

Hugo Dahlquist

A thesis presented for the Degree of
Bachelor of Statistics

Department of Statistics
Lund University School of Economics and Management
Sweden

January 2024

Supervisor: Jonas Wallin

Abstract

The financial industry is one of the highest grossing sectors in the world as it is estimated to represent 24% of the global economy. As most companies want their asset value to increase, it is of high interest to make good investments which will increase in either the short or long run. The main aim of this thesis was to reveal the performance on predictions using two different machine-learning models, namely Random Forest and Artificial Neural Networks. The target variable that our models aimed to predict was the closing prices of stocks for the FAANG companies, all of which are traded on NASDAQ. Our models used data sets dated from 2010 until 2020, that included several different features that often are subject to technical, fundamental and macroeconomic analysis. As we used the year of 2020 as validation data, stocks were highly affected by the Covid-19 pandemic, that caused severe fluctuations in several sectors and of course the financial markets. This might have been the main reason why Artificial Neural Networks was more effective in predicting the closing price, since it took noisy processes into consideration. We believe though that the global pandemic made an impact on our predictions, that did not perform efficiently enough to use in investment decisions. However, a series of results concerning statistical properties of the models are of interest.

Table of Contents

1 Introduction

To foresee the stock market has for a long time been widely considered to be an impossible task. There are countless number of factors, such as financial performances of different companies, political reasons, environmental catastrophes, global pandemics, global economic conditions etc., that affect the outcome. This results in predicting an accurate stock price is deemed to be an elusive task (Li, 2023). However, during the last two decades, predicting the stock market has become a hot topic for researchers. Machine learning methods has been used and developed to better understand the behavior of the stock market. Artificial Neural Networks, Decisions Trees and Random Forest are three methods which are all subsets of supervised machine learning and has been used frequently to predict stock market prices. In this context, our study aims to assess the predictive capabilities of the three prominent machine learning techniques mentioned above.

For our investigation, we will focus on forecasting the closing prices of FAANG stocks – Amazon, Apple, Google, Facebook, and Netflix. These technology giants are all influenced by various factors and predicting the stock price for these companies could therefore be an extremely difficult task. The interplay between macroeconomic, fundamental, and technical data are all very influential in the outcome of each company's closing price. A nuanced analysis and well-developed deep learning model with the right combination of independent variables are therefore required to make a good prediction.

Utilizing 21 carefully selected variables (detailed in Section 3.1, *Explanatory Variables*) spanning from 2010 to 2019 are going to be used to predict the closing price in 2020 of the five companies mentioned above. The independent variable will be lagged by one day to study the closing price on the subsequent day. The variables are divided into three main groups: macroeconomic-, fundamental- and technical data. The main source for the fundamental variable is Bloomberg, considered to be a top provider of financial data. The technical data is supplied by GitHub and The Federal Reserve Bank of St. Louis is used to find valid macroeconomic data. One of many aims of this thesis is to provide a better understanding of the interplay between macroeconomic trends, fundamental indicators, and technical data and their collective impact on stock price predictions.

Two distinct tests will be conducted. In the first test, our models will predict the exact closing price for the following day. The second test, the classification test, a dummy variable will be employed where 1 represents an increase in price and 0 denotes a decrease. We believe that the combination of variable and company selection, coupled with machine learning techniques and the incorporation of these two tests, will address a gap in stock market prediction research that has yet to be explored. Furthermore, this thesis will conduct a comprehensive analysis of the first test, comparing metrics such as mean squared error, mean absolute error, and mean absolute percentage error. The second test will incorporate assessments against the no information rate, Kappa statistic, sensitivity, specificity, positive predicted value, negative predicted value, and balanced accuracy. This multifaceted approach aims to provide a thorough and nuanced assessment of the predictive models. The main target of this thesis is to answer the following question: *How effectively can Random Forest and Artificial Neural Network models predict the closing price for the FAANG companies the coming day?* The findings of this research attempts to contribute to the field, providing insights that may improve the precision of stock price predictions.

1.1 Structure

The thesis will be structured as follows: Chapter two will discuss different theories such as Stock Price Analysis and Effective Market Hypothesis. It will also provide a brief overview of general machine learning concepts. This chapter also aims to equip the reader with a better understanding of the deep learning methods which serves as the thesis's foundation; *Artificial Neural Network, Decision Tree and Random Forrest*. Previous research is also presented in this part of the thesis. Chapter three will present the data that is used in the models. Mainly focusing on the collection of data, data wrangling and explanatory variables. The following chapter, *Empirics* will explain more in to detail how the Artificial Neural Network, Random Forrest and Decision Tree are constructed. A thorough explanation of how the data was trained and how the final models were created will be provided. The results that were found from the models described are presented in chapter six. Finally the results will be summarized in the final chapter.

2 Background

In this section, concepts and theories that are important to consider when forecasting stock prices, such as *Stock Price Analysis* and the *Efficient Market Hypothesis*, are introduced. An overview of machine learning, which is a crucial foundation for this thesis, will also be presented. Finally, earlier studies are analyzed to understand how researchers who have taken on the challenge of predicting the stock market have approached the topic.

2.1 Stock Price Analysis

Financial analysis, or business valuation concerns the collection and compilation of information about a specific company or even an entire market sector. The valuation is then used for several tasks, for both internal and external parties. Reasons to use the information derived from the valuation could be used in a scenario where a specific company is planning to enter a new market sector, through mergers and acquisitions for example. The valuations can also be used for financial reporting in order to comply with financial litigation (Hitchner, 2011). A result from this leads to the wide usage of financial analysis, and is therefore done by several actors in both the corporate as well as the public sector (Ibid).

To understand how valuations are made one should understand how value is defined. Hitchner defines 5 standards of value; fair market value, investment value, intrinsic value, fair value (state rights) and fair value (financial reporting). Fair market value (FMV) is defined according to the International Glossary of Business Valuation Terms as the price that a hypothetical seller and buyer would trade property which is under no form of compulsion, in an unrestricted market and both parties have reasonable knowledge about the asset (Hitchner, 2011). Investment value is the value that a specific investor is willing to pay for an asset, which also reflects more of the risk-adversity of said investor. Intrinsic value is defined as the "true" value of an asset which is based on the fundamentals of the asset, such as company-specific financial performance or the state of the economy. In the case of this paper, financial analysis is mainly focused on the valuation of stock-pricing and the two primary disciplines are fundamental analysis and technical analysis. State rights fair value is defined by the Uniform Business Corporation Act as "The value of the shares immediately before the effectuation of the corporate action to which the dissenter objects, excluding any appreciation or depreciation in anticipation of the corporate action" (Ibid). The last standard is financial reporting fair value and is defined as the price which an asset is sold or paid for in and in a well-managed transaction between market participants at a specific date (Ibid)

Fundamental analysis (FA), the analyst measures assets intrinsic value by gathering information that could affect the price. That information could include anything from both macroeconomic and microeconomic perspectives. The valuation gives the asset a FMV, the valuation is then compared with a current market price. A key idea within fundamental analysis is that the long-run value will reflect the fundamentals of the overall economy, market sector and the company itself (Investopedia, 2023).

Technical analysis (TA), focuses on determining short-run trends in stocks and by using statistical methods and variables such as price and volume are key in this discipline. Previous trading history is used to find patterns in price fluctuations and volume in order to find a trading point which is believed to result in profit. Charles Dow, who introduced this method stated three assumptions; all factors are priced in the asset, price follows a trend and that history tends to repeat (Investopedia, 2022).

2.2 Effective Market Hypothesis

The efficient market hypothesis (EMH) is a theory first developed by Eugene A. Fama and Paul A. Samuelson, independently in the 1960's. The idea says that the price of a stock reflects all available information about the built-in value of the stock. EMH addresses the possibilities of price-prediction of assets (Baldrigde, 2022).

In 1965, Eugene Fama presented comprehensive evidence indicating that the practice of technical analysis lacked the capability to predict future prices. This supported the theory that stock prices are just a random walk. In 1970, Fama developed his famous paper ”Efficient Capital Markets: A Review of Theory and Empirical Work.” Where he defined three forms of efficiency. From weak, to semi-strong to strong. In weak efficiency, future returns cannot be based on past returns or technical analysis indicators. In a semi-strong efficiency, the previous level is included but also indicators in fundamental analysis such as financial reports from companies. The highest level of efficiency defines that even private information is already captured in the price (Ibid). Limitations to the theorem were made by, amongst others, Sanford Grossman and Joseph Stiglitz (1980) who pointed out that market frictions, which includes the costs of analysis and trading, limits the efficiency (Lo, 2017).

2.3 Machine Learning

In the 1950's machine learning was first discovered and has been developed massively since then (Fradkov, 2020). Today it's an effective tool where computers can learn, Without human programming, and a great method to make predictions and optimize data (Bi et al, 2019). There are three main categories; supervised-, unsupervised- and semi-supervised machine learning. The former aims to make predictions on new, unseen data, using labeled data which contains both the target (the response variable) and features (the explanatory variables). The prediction is achieved by training the data until the model executes accurate predictions. The data sets contain input and outputs, where the model with help of the training data, learns the relationship between the two. It adjusts its parameters to minimize prediction errors. The model then uses the patterns it learned from the training data to make predictions on the new, unseen data. Supervised machine learning is an effective tool to predict accurate results, however it requires correctly identified labeled data. Otherwise the supervised algorithms will run the risk of acquiring misinformation. In contrast, unsupervised machine learning operates without labeled data and requires no human intervention. These models operate independently and aims to identify hidden patterns, structures, relationships or groupings within the unlabeled data. Lastly, semi-supervised machine learning contains elements from both supervised and unsupervised. The data set consists of both labeled and unlabeled data. The labeled data is used to train the model and make accurate predictions. Using a semi-supervised machine learning is a solution of not having enough labeled data to perform supervised machine learning (IBM, 2023). This thesis will mostly focus on supervised machine learning.

2.4 Previous research

In 2010 Nair, Mohandas and Sakthivel used four different machine learning models to predict trends in the stock market prices between 2003 and 2010. More specifically the study aimed at predicting the Bombay Stock exchange (BSESENSEX). The study mostly focuses on the short-term trends in the stock market, where 21 different technical indicators are used. The authors, with the help of a decision tree, decided which variables are relevant and predicted the trends in BSESENSEX. An artificial neural network and a naive bayes based trend prediction is also applied to predict the stock exchange. Also, rough set based trend Prediction is implemented to handle imperfect or uncertain data. The authors found that the decision tree model could predict the BSESENSEX to an accuracy of 90,22%, and therefore for the best predictor. The rough set based prediction had an accuracy of 88,18%, the artificial neural network of 77,66% and finally the naive bayes based trend prediction of 72,36% (Nair, Mohandas & Sakthivel, 2010).

Tsung-Sheng Chang evaluates the differences in forecasts on the stock market between decision trees and artificial neural networks. The author also adopts a hybrid model to see if it performs better than the two original ones. The study analyses the Taiwanese digital game content stocks between 1 January 2018 and 31 June 2019. The data set consisted of 10 different game stocks, where the closing price for each day was studied. In this particular case the artificial neural network model predicted the stock price most successfully. However, the author also implies weaknesses with this paper. Firstly, the short period of time studied and secondly the period of verifying the prediction model was rather short. Lastly, a high level of correlation between companies' business results might have caused a very high, and maybe false value (Chang Et. al, 2011).

In 2020 Hindrayani, Fahrudin, Aji R and Safitri used a decision tree to determine Indonesian telecommunications stock price. The purpose of this specific study was to see if COVID19 had a negative influence on the impact of fundamental data on stock prices. The fundamental data that is used in this study are the following; total current assets, total liabilities, net income for the period, and Earning Per Share (EPS). Beyond the decision tree, K-Nearest Regression, Mulitple linear regression, Support Vector regression is performed. The study found a correlation between fundamental data and the stock price. The highest correlation is found in the decision tree model (Hindrayani Et. Al, 2020).

In 2017, Rupesh A. Kamble presented a paper with the results he received from stock price predictions using quantitative methods. The goal with the paper was to predict the short-run price trend of a selection of 1000 stocks listed on National Stock Exchange of India and Bombay Stock Exchange. In the study he used several variables and technical oscillators in order to predict the short-run price trend. These were the historical prices of the stocks along with Relative Strength Index, Bollinger band and Moving Average Convergence/Divergence (MACD). However that would not suffice in the long-run prediction, he then used measurements used in fundamental analysis, such as net profits, dividends, promoter holding, debt to equity and prices over earnings. The methods used in the study mainly consisted of random forest decision trees along with J48 algorithms which is a part of the WEKA library in R. An issue that the author stumbled upon was data overfitting which can lead to weaknesses in the model, but can be reduced down by a pruning tree. After conducting the experiment he retrieved positive results where the risk of losing money was about 16% for the short term-run and 7% for the long-run. (Kamble, R, 2017)

Wu, Lin and Lin conducted a study in 2005 with the goal of creating an effective technical method for stock trading. Their chosen method consisted of decision trees and the filter rule with extensions of variables used within fundamental analysis. The decision trees were modeled by the C4.5 algorithm which is an extension of the ID3 algorithm. The filter-rule is a tool in technical analysis where one sets rules for when to buy, hold or sell a stock after it has increased or decreased by k%. Their proposed application of the filter-rule differs from the original method, as it says you should buy whenever an effective buying point appears and sell whenever a selling signal appears. They used WEKA which is a set of algorithms used for data mining. The variables used in order to discover an effective buying point were money supply, inflation rate, revenue of upper stream entities and the price of index futures. Their observations include electronic stocks from Taiwan Stock Market and technology stocks from NASDAQ, and the authors did correct the stock prices in order to adjust for dividends which are included in the price. A part of their findings was that the price of index futures was highly efficient in determining efficient points from the decision tree. As this was a further study of Lin's previous research, they also found that this method did outperform their previous experiment (Wu, Lin & Lin, 2004).

3 Data

The upcoming chapter will discuss the data used in our models to predict the stock prices of the FAANG companies. Firstly, the selection of variables will be presented, followed by an explanation of how the data is collected. Finally, we will discuss how NAs are handled and address any other pre-processing problems that might occur.

3.1 Selection of Variables

As mentioned earlier, the data set consists of technical, fundamental and macroeconomic data. The selection of variables is a crucial aspect in constructing a model for predicting stock prices. The choice of variables that will form the foundation of the model is based on prior research.

Technical Data

Technical data is mainly centered around price and volume and is a very common tool to predict stock prizes. The data is predominantly used as tool to asses investments and pinpoint trading opportunities by analyzing patterns and trends. Studying the historical trading activity and price fluctuations of technical data can offer valuable insights into its future price movements. This, however, relies on the fact that past trading activity and price changes can provide valuable indicators for the stock price's future price movements. Additionally, this approach assumes that prices moves in trends, and history tends to repeat itself. One can also argue for the fact that all known fundamentals, such as revenue and profit margin are already included into the stock. However, critics argue that historical price and volume data may not contain useful information, and some view technical analysis as a self-fulfilling prophecy. Keeping this in mind and unlike some papers presented in previous research, fundamental and macroeconomic data are also going to be included in the data set (Hayes, 2022).

Fundamental Data

Unlike technical data, fundamental data analysis stock prices by taking current industry conditions, market factors, economic conditions and a company's financial circumstances into consideration. The main purpose of using fundamental analysis is to determine whether a stock is overvalued or undervalued. Quantitative fundamentals involve measurable characteristics such as revenue and profit. Qualitative fundamentals, however are less tangible aspects like management quality and brand recognition. Therefore, only quantitative fundamentals is included in the data set. In conclusion, fundamental analysis relies on financial statements and other quantitative and qualitative factors, unlike technical analysis (Segal, 2023).

Macroeconomic Data

Stock prices are not only influenced by fundamental and technical data but also various economic indicators plays a crucial role such as interest rate. Interest rates are a powerful tool for predicting economic trends because they influence borrowing costs, consumer spending, business investment, and various sectors of the economy. Understanding how changes in interest rates can impact these factors helps investors make informed decisions about asset allocation and portfolio management. Increased interest rates often cools down the economic activity which might interrupt an upward trend in the stock markets. Conversely, a decrease in interest rates can trigger an opposite effect (Levitt, 2022).

Attribute	**Interval**	**Type**
Open Price	Daily	Technical
Closing Price	Daily	Technical
Price High	Daily	Technical
Price Low	Daily	Technical
Volume	Daily	Technical
EBITDA margin	Quarterly	Fundamental
Enterprise Value	Quarterly	Fundamental
Debt	Quarterly	Fundamental
Cash Flow	Quarterly	Fundamental
Company market cap	Quarterly	Fundamental
Revenue	Quarterly	Fundamental
EBIT	Quarterly	Fundamental
EBITDA	Quarterly	Fundamental
EPS	Quarterly	Fundamental
Price over Earnings	Quarterly	Fundamental
Price over Shares	Quarterly	Fundamental
EV/EBIT	Quarterly	Fundamental
EV/EBITDA	Quarterly	Fundamental
Real interest rate	Daily	Macro economics
Loan rate	Daily	Macro economics
US interest rate	Daily	Macro economics

Table 3.1: List of variables

3.2 Collection of Data

The collection of data was done by using several databases. The variables which are connected to technical analysis such as daily stock prices and trading volume is found through GitHub where a complete set of the FAANG companies were found. The dataset was created by Aayush Mishra who used the method of web-scraping on the source of Yahoo Finance and dates back to the point where the stocks were first traded on NASDAQ (Kaggle, 2020). Fundamental variables were found through Bloomberg which is the top provider of financial information. The database contains real-time company news and descriptive data of companies, several data dates back several years. It is often used by professional brokers and institutional investors and we therefore deem it a reliable source for our data collection (Hitchner, 2011). Macroeconomic variables were found through The Federal Reserve Bank of St. Louis which supplies a complete economic data source for national, international and private sources (Federal Reserve bank of St. Louis, 2022).

3.3 Pre-processing

Data rarely comes in perfect excel-sheets and there is often some form of pre-processing required in order ensure good data quality that models can handle. The models in this thesis uses information gathered varying sources it was first necessary to compile it into one file for simplicity. Technical, fundamental and macroeconomic variables are each reported at different time frames the first step was to impute values to obtain a proper data alignment. In the data set there were cells containing missing values, these were handled by removing each row that contained one or more *NA's* which also is required to execute the algorithms. However, this can induce some level of bias, but as the number of rows omitted were low this should not affect the modeling and predictions. Due to the nature of our variables there are no direct outliers (Jaberi et al, 2020) As this thesis handles two different response variables which are the *Closing Price* of the company stocks and a *dummy* for the trend prediction. For the first variable, *Closing Price*, there is the issue of data leakage that could occur if other variables that are reported daily were used in the model; because in reality the model would then use information that it does not have access to. Therefore a time lag of n-1 was introduced to counter this, which of course can decrease the accuracy of our models. The other response variable, *dummy*, was created by setting an IF-statement between the closing prices between two days; if today's price were higher than yesterday the observation would take the value 1, if else 0.

The data was also independently processed in order to fit the different models, as they have different requirements and these steps will be further discussed in section 4.3.

4 Empirics

The following section will present the main methods used in this thesis: *Artificial Neural Networks*, *Decision Trees*, and *Random Forest*. It will also provide a comprehensive discussion of how the methods are implemented. Finally, performance metrics will be presented to evaluate the results.

4.1 Artificial Neural network

Artificial Neural Network (ANN) is considered to be a fundamental component in machine- and deep learning. ANN is inspired and modeled after the human brain, mirroring the communication between organic neurons. This type of machine learning consists of multiple neurons and layers. The layers that are most often included are input-, hidden- and output layers (IBM, 2023).

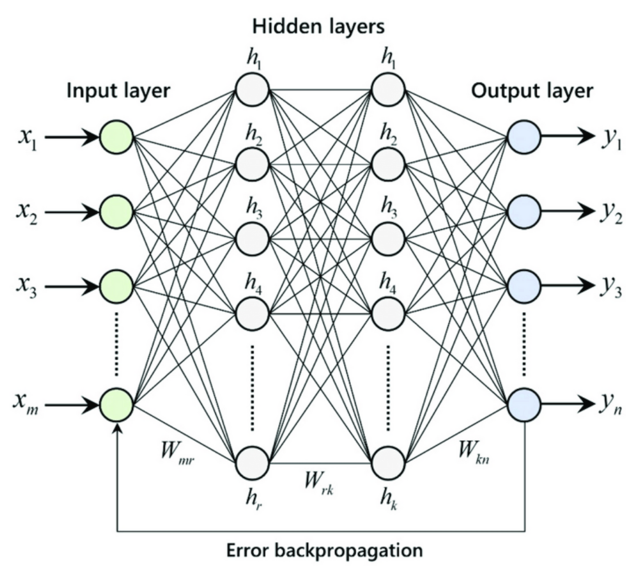

Figure 4.1: Artificial Neural Network (Esfe et. Al, 2021)

As one can tell by the name, the input layer receives the initial data that works as an input to the network. Then, neurons that process incoming data and extract patterns are present in the hidden layers. Lastly, the output layer creates the ultimate outcome or forecasts for the network. The data set is weighted which makes the more contributing variables have a greater impact on the output. Each input is multiplied by a corresponding weight, which is adjusted during the training process to optimize the neuron's prediction. This model is a version of supervised machine learning and therefore relies on training data which results in a more accurate prediction. Any node whose output exceeds the defined threshold value is activated and begins providing data towards the network's uppermost layer. Once the algorithms are trained and effective instruments it's a very effective tool in computer science (IBM, 2023).

4.2 Decision Tree & Random Forest

Just like ANN decision trees are considered to be supervised machine learning and are common machine learning algorithms for classification and regression tasks. By moving through the tree's branches from the root to the leaf nodes, its structure resembles a tree that makes judgements based on input data. First of all, the structure of decision trees are the following; root-, internal- and leaf nodes. The entire data set is first represented by the root node and this is where the first decision is made. The decision is then made in the internal nodes. For each question presented, the data is divided in to new nodes, depending on the value. The nodes that are the final ones are the leaf nodes, and represent the outcome of a decision.

The decision is based on a question which can be answered with either yes and no or lower and higher. The answer is determined by a criterion from the data set. Based on the decision from the result a branch evolves from an internal node. During this process, the training of the decision tree, the algorithm seeks to identify the decision points and criteria which minimizes the classification errors. The model can then be used to predict the outcome for new, untrained data. This kind of machine learning comes with the risk of overfitting the data. However, it's great at handling numerical and categorical data and quite easy to interpret (S. B. Kotsiantis, 2013)

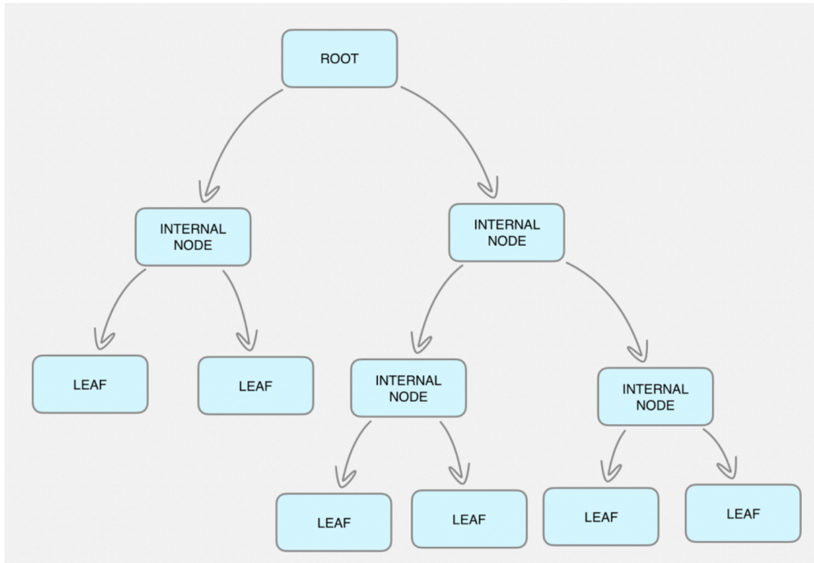

Figure 4.2: Decision Tree (Peña, 2018)

Although decision trees has been and is a widely used algorithm within machine learning, there has been new varieties implemented in the past years. One of these are the Random Forest algorithm which is trademarked by Leo Breiman and Adele Cutler. It is often used in finance and healthcare, and has been a part of several business decisions. The algorithm is used to solve for both classification and regressions problems and consists of three main hyperparameters which are node size, number of trees and number of branch variables.

It creates a number of decision trees and each tree draws a so called bootstrap sample. As the bootstrap samples do not contain all observations, it indirectly creates a so called Out-of-bag, or OOB for short. Since the observations in the OOB sample isn't used, the algorithm uses this OOB sample to perform its own cross validation. The algorithm also uses bagging to introduce a higher level of randomnesss, which is meant to decrease correlation between trees. As mentioned earlier, random forest can be used for classification and regression problems and will slightly differ in the next step. For classification problems the predicted class will be equal to the most frequent categorical variable, whilst for regression problems it will average the decision trees (IBM, 2023).

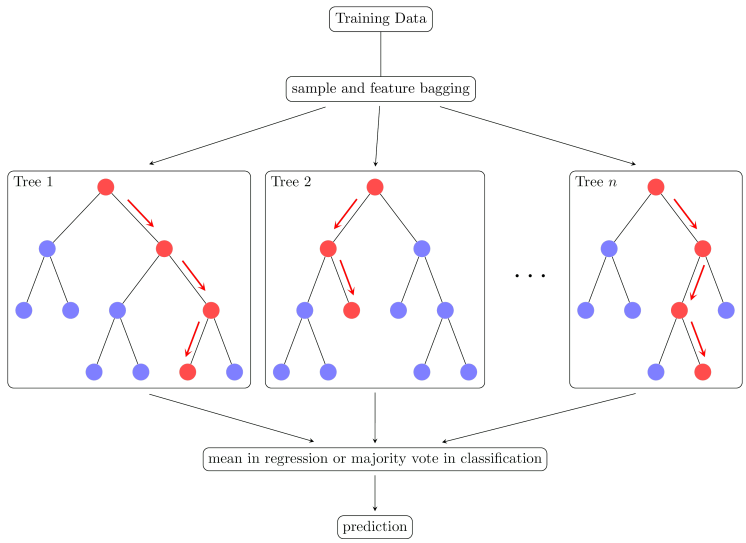

Figure 4.3: Random Forest (Riebesell, 2021)

4.3 Implementation

Artificial Neural Network

The package Keras, which is a high level application program interface of TenserFlow, is used to create the model and enables deep learning tasks in R. This allows one to define and configure a deep learning model using a simple and intuitive syntax.

The ANN is defined layer by layer, which can be added one after the other using a sequential model. Selecting the right amount of layers is a crucial part of finding the right architecture of the ANN. A relatively large amount of layers might result in overfitting, whereas underfitting may occur with too few layers. One or two hidden layers is suitable for a model where the data is complex and has few amount of dimensions. In models with large dimensions 3-5 amount of hidden layers should be used to find an accurate prediction (Adil et. al, 2020). Taking this into consideration, the model contained 5 layers in total, one input, three hidden and one output. As mentioned earlier, too many layers might result in a prediction that corresponds to closely to the training data. Therefore the decision of having 5 layers introduced the challenge of overfitting. Considering this issue, an L2 was introduced between the layers, which is a form of regularization. This is a method employed in machine learning to manage the complexity of a model by adding a penalty term into the loss function. It decreases the magnitudes of the model's parameters, preventing the model from closely adapting itself to the training data (Pykes, 2023). Dropout layers were also introduced to simplify the model and decrease the over reliance on specific neurons or feature combinations. This approach ensures that the neural network maintains a more generalized learning, preventing it from overfitting (Srivastava Et. al, 2014).

Once a fitting amount of layers and regularization has been established, the fit function is used to train data and layer_dense is applied to connect the layers. Just like the layers, the right amount of units had to be decided. Units refers to the amount of "nodes" in each label and a learnable parameter that contributes to the transformation of input data. One can follow the process of finding the right amount of units in table 4.1. In the test where the aim is to predict the exact closing price on the subsequent day model is set up via the compile function. The optimizer "adam", loss function "mean_squared_loss" and metrics "accuracy" are used in order to train the model well. The main focus while training the model is to adjust the weights of the neural network to minimize the mean square error. "Adam", or *Adaptive Moment Estimation*, is widely considered to be efficient and a good performer across a wide range of deep learning tasks. It is eminent for its simplicity, computational efficiency, low memory requirements, and suitability for large-scale problems with substantial data. It is also effective for non-stationary objectives and situations with noisy processes. The algorithm's hyper-parameters have intuitive interpretations and typically require minimal tuning (Kingma & Ba, 2015). In the test where the objective is to predict the exact closing price, the computation is identical, with the only difference being the substitution of "mean_squared_loss" with "accuracy". Another difference is that the target variable is standardized ensuring data reliability and accuracy. These adjustments enables the analysis of how successful the model was to predicted the correct closing price.

After model computation, a broad range of epochs and batch sizes were explored using the "fit" function to identify an optimal model. The architecture plays a pivotal role in creating an efficient Artificial Neural Network (ANN). To determine an effective combination of layers, units, epochs, and batch sizes, an extensive array of different configurations was tested. The accuracy and the loss function were analyzed in order to find the combination that would result in an efficient model. Table 4.1 showcases nine distinct combinations of units, layer dropout, epochs, and batch size. In the quest for an accurate predictive model, the goal was to achieve high accuracy and low loss. It's essential to note that numerous tests were conducted beyond those presented in Table 4.1. The configuration featuring a layer dropout of 0.3, 300 epochs, and a batch size of 400 yielded the highest accuracy and the second-lowest loss on the training data. Therefore, our artificial neural network is configured as follows: There are 5 layers with 400 nodes in the input layer, 300, 200 and 100 nodes in the hidden layers, finally one node in the output layer.

Layers	Units	Drop.	Epochs	Batch	Acc.	Loss
5	150, 100, 50, 25, 1	0.3	300	250	0.605	0.245
5	200, 150, 100, 50, 1	0.3	300	250	0.603	0.245
5	200, 150, 100, 50, 1	0.3	200	250	0.590	0.244
5	200, 150, 100, 50, 1	0.3	200	350	0.592	0.246
5	200, 150, 100, 50, 1	0.3	300	400	0.605	0.245
5	200, 150, 100, 50, 1	0.3	200	650	0.555	0.249
5	300, 200, 100, 50, 1	0.3	300	400	0.611	0.245
5	300, 200, 100, 50, 1	0.4	300	500	0.604	0.246
5	400, 300, 200, 100, 1	0.4	300	600	0.597	0.246

Table 4.1: Training ANN

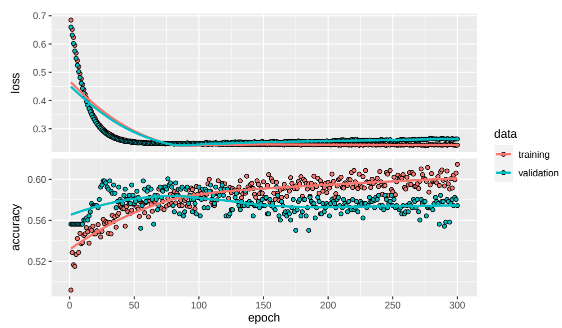

Figure 4.4: ANN - Acuracy

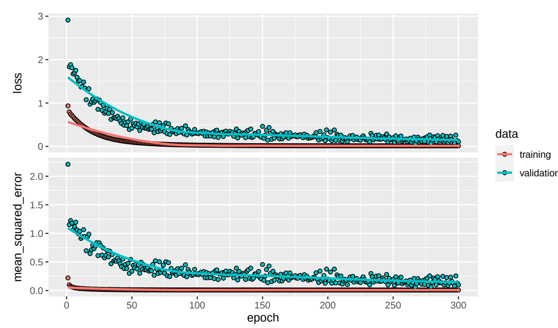

Figure 4.5: ANN - Mean squared loss

As evident from Figures 4.2 and 4.3, the mean square loss exhibits a decrease for both the training and validation data sets. Given the minimal difference observed between the training and validation sets, the likelihood of overfitting can be interpreted as very small. Throughout the training process, the mean squared error decreases and signifying improved accuracy in predictions with an increase in the number of epochs.

Company	Accuracy	Loss
Apple	0.611	0.245
Amazon	0.620	0.222
Google	0.634	0.228
Facebook	0.613	0.240
Netflix	0.610	0.238

Company	Mean Square Errors	Loss
Apple	0.0320	0.0359
Amazon	0.0153	0.0208
Google	0.0053	0.0124
Facebook	0.0053	0.0172
Netflix	0.0082	0.0137

Table 4.2: Accuracy, Loss & Mean Square Errors

From table 4.2 one can interpret the accuracy, loss and mean square errors on the training set.

Random Forest

The Random Forest model used in the thesis was built using the *random Forest*-package which is a downloadable package in R. The package contains several different features that can be helpful for this type of machine learning. In addition to this package, a package called *caret* was used which calculates the importance of variables used in the model.

Setting up a Random Forest is relatively straight forward as it's built like an ordinary linear model. The model is versatile as it can handle both numerical and categorical variables, and it doesn't require scaling or standardization. The main pre-processing needed to perform a Random Forest is to either select your response-variable as a factor or a numeric. If the response variable is a factor the model will automatically perform a classification task, and if the response variable is a numeric it will perform a regression. As Random Forest is an ensemble model made from several decision trees which then are averaged it naturally adapts to prevent overfitting. One must however choose how to tune the models hyperparameters which were mentioned earlier in section 4. The model will, if nothing is changed, use the original setting for all hyperparameters. With help of the *caret*-package it was also possible to determine the importance of the explanatory variables. As one can see in the example output for Facebook below, the key variables for the model were technical variables.

Although the Random Forest algorithm perform the OOB-prediction, we decided to use cross-validation to decide the number of variables included at each split. We first performed a cross-validation with 10 folds for each company. *Mtry* is the hyper parameter that determines the number of variables included at each split. As one can see in the output summary below, the overall metrics becomes significantly improved as number of *mtry* increases from 2 to 13, and a lighter decrease from 13 to 24. Testing was also performed on the individual Random Forest models which also found that the mean absolute error did not heavily decrease as we included more than 13, beginning with the six variables which exhibited a higher importance than 10 in modelling.

Mtry	RMSE	R-Squared	MAE	RMSESD	R-squared Sd.	MAEsd
2	2.4399	0.9982	1.7518	0.6291	0.0008	0.4111
13	0.9959	0.9997	0.6786	0.1219	0.00006	0.0536
24	0.9948	0.9997	0.6746	0.1436	0.00008	0.0613

Table 4.3: Cross validation on Facebook

As the *Random Forest*-algorithm is generally robust against overfitting, a too high number of decision trees can introduce some level overfitting. Finding the correct number of decision trees executed by the algorithm can be a difficult task, and also that it does not necessarily improve the statistical significance for the model as you increase the number of trees (Oshiro et al, 2012). Therefore a decision was made to use 2000 trees in order to let the model train extensively while allowing a lower CPU level. To summarize, a decision to use 2000 decision trees and 20 variables randomly sampled as candidates at each split was made.

4.4 Performance metrics

To perform a proper evaluation of the two different machine learning models that we are investigating in this thesis, it is important to first determine which metrics are relevant to the different tasks. In this section we will first present the different metrics which we will form a basis for the evaluation of the closing prices, namely the performance metrics regression. After that we will then introduce the numerous metrics used for evaluating the dummy variable, that is classification.

4.4.1 Regression metrics

MSE

The Mean Squared Error (MSE) is given by the formula:

$$\text{MSE} = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2$$

It calculates the squared difference between the actual and the predicted values and then divided by the amount of observation. MSE is a common performance metrics to evaluate how accurate predictions are. The vector of observed values is represented by y , while $\hat{y}$ represents the corresponding predicted values and n corresponds to the amount of observations (Hodson Et. Al, 2021).

MAE

The Mean Absolute Error (MAE) is given by the formula:

$$\text{MAE} = \frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_i|$$

MAE calculates the difference between the actual value and the predicted. It is widely considered to be more intuitive than RMSE, since it exhibits linear changes. Unlike RMSE and MSE, all errors are treated equally and larger ones don't have a disproportionately large impact on the metric. Just as for MSE, y is the observed values, $\hat{y}$ represents the corresponding predicted values and n is given by the amount of observations (Wang Et. Al, 2018).

MAPE

The Mean Absolute Percentage Error (MAPE) is given by the formula:

$$\text{MAPE} = \frac{1}{n}\sum_{i=1}^{n}\left|\frac{y_i - \hat{y}_i}{y_i}\right| \times 100$$

Just like MAE, MAPE is easy to interpret as it provides the average percentage difference between predicted and actual values. It takes the average of the absolute percentage differences across all observations. Small values indicate precise predictions, whereas large values implies less precise. The vector of observed values are represented by y , $\hat{y}$ are the predicted values and n is equal to the amount of observations (Khair Et. al, 2017).

4.4.2 Classification metrics

No information rate

The No Information Rate (NIR) refers to the accuracy a classification model would achieve if only predicting the majority class and establishes a baseline for comparison of the other metrics. If the difference between NIR and other metrics it could imply that the model doesn't provide meaningful predictions. In addition, if the NIR value is higher than 0.5 it indicates that the data set was balanced (Bicego & Menci, 2023).

$$\text{NIR} = \frac{\text{Number of instances in the majority class}}{\text{Total instances}}$$

Kappa

The Kappa value represents a measure of agreement between predicted and actual values, considering the chance of pure coincidence. This value ranges between -1 and 1, where 1 indicates perfect agreement, 0 indicates that the accuracy is equivalent to chance, and -1 suggests worse than chance agreement. In the equation below Po is the observed agreement and Pe is the expected agreement by chance (Yilmaz & Demirhan, 2023).

$$\text{Kappa} = \frac{P_o - P_e}{1 - P_e}$$

Sensitivity

Sensitivity which is also called true positive rate and calculates the number of true positives amongst the actual number of true positives. This value ranges between 0 and 1, where 0 indicates that the model lacks the capability of predicting positives, whilst 1 indicates that it can perfectly predict positive examples (De Diego Et. Al, 2022).

$$\text{Sensitivity} = \frac{\text{True Positives}}{\text{True Positives} + \text{False Negatives}}$$

Specificity

Specificity, on the other hand, refers to the true negative rate and calculates the proportion of true negatives among the total number of actual negatives. It also ranges from 0 to 1, where a high value indicates a low risk of making false positive predictions (De Diego Et. Al, 2022).

$$\text{Specificity} = \frac{\text{True Negatives}}{\text{True Negatives} + \text{False Positives}}$$

Positive predicted value

Positive predictive value measures the true positives divided by both true positives and false positives calculated by the model. A high positive predictive value indicates that when the model predicts a positive value, it is likely to be correct. In this case, forecasting that the price is decreasing correctly (De Diego Et. Al, 2022).

$$\text{PPV} = \frac{\text{True Positives}}{\text{True Positives} + \text{False Positives}}$$

Negative predictive value

Negative predictive value calculates the proportion of true negatives over the number of true negatives and false negatives. This value also ranges between 0 and 1, where a high negative predictive value suggests that when the model predicts a negative outcome, it is likely to be correctly specified. In this case predicting accurately that the price is going down correctly. In this thesis this is represented by predicting that the stock price increases correctly (De Diego Et. Al, 2022).

$$\text{NPV} = \frac{\text{True Negatives}}{\text{True Negatives} + \text{False Negatives}}$$

Balanced accuracy

The balanced accuracy provides a more reliable assessment of a model's overall accuracy by taking into account the imbalances in the distribution of classes. It is particularly relevant when the classes in the data set have unequal sizes, as it ensures that the evaluation metric does not overly favor the majority class. This makes it a valuable tool in scenarios where certain outcomes are more rare but equally important to detect or predict (De Diego Et. Al, 2022).

$$\text{Balanced Accuracy} = \frac{\text{Sensitivity} + \text{Specificity}}{2} = \frac{\frac{\text{True Positives}}{\text{True Positives} + \text{False Negatives}} + \frac{\text{True Negatives}}{\text{True Negatives} + \text{False Positives}}}{2}$$

5 Results

The upcoming chapter will present the results yielded by the models. Firstly, the results for Artificial Neural Networks (ANN) will be presented, followed by the presentation of the Random Forest results.

5.1 Artificial Neural Network, Classification

Company	Apple	Amazon	Google	Facebook	**Netflix**
Accuracy	0.6036	0.5223	0.6095	0.5849	**0.5946**
95% CI	0.5256, 0.6778	0.4412, 0.6025	0.5315, 0.6835	0.5042, 0.6624	**0.5109,0.6744**
No Information Rate	0.568	0.5796	0.5858	0.5786	**0.5068**
P-Value [Acc >NIR]	0.1968	0.9372	0.2935	0.4697	**0.0197**
Kappa	0.1827	-0.0386	0.1177	0.1522	**0.1905**
Sensitivity	0.4932	0.2273	0.2286	0.5224	**0.6575**
Specificity	0.6875	0.7363	0.8788	0.6304	**0.5333**
Pos Pred Value	0.5455	0.3846	0.5714	0.5072	**0.5783**
Neg Pred Value	0.6408	0.5678	0.6170	0.6444	**0.6154**
Balanced Accuracy	0.5903	0.4818	0.5537	0.5764	**0.5954**

Table 5.1: *Artificial Neural Network*, Classification

Apple

Roughly around 60% of the dummy variables was correctly predicted when the model was forecasting the price movements of the Apple stock. However, it is noticeable that the NIR rate was relatively high as well, 56.8%. As a result, it cannot be conclusively asserted that the accuracy was statistically significant higher than the NIR. Although the p-value is not below 0.05 the Kappa value of 0.1827 indicates that there were some predictions beyond chance. The model's sensitivity is 49.32%. This means that the model correctly identified approximately 49.32% of the actual positive instances. The specificity, on the other hand indicates that the model correctly identified approximately 68.75% of the actual negative instances. The model achieves a positive prediction value of 54.55%, signifying its capability to correctly forecast instances when stock prices were expected to decrease. The negative prediction value was observed to be 64.08%, reflecting the model's accuracy in predicting instances when stock prices were anticipated to raise. The Balanced Accuracy, calculated at 59.03%, was slightly lower than the standard accuracy.

Amazon

The accuracy in predicting the correct price movement for Amazon is 52.23%, which contrasts with a NIR of 57.96%. The p-value for testing whether the accuracy is significantly greater than the NIR is 0.9372, indicating that the accuracy is not statistically significantly larger than the NIR. The calculated kappa value, at -0.0386, shows that the model's performance is even below what would be expected by random chance. Sensitivity and specificity are calculated at 22.73% and 73.63%, respectively. The Positive Predictive Value is 38.46%, while the Negative Predictive Value is 56.78%. Lastly, the balanced accuracy was calculated to 48.18%.

Google

In the evaluation of the model's performance for Google, an accuracy of 60.95% was observed, surpassing the NIR of 58.58%. However, statistical significance was not achieved, as the p-value exceeded 0.05 when testing whether the accuracy significantly surpassed the NIR. The calculated kappa value was 0.1177. Sensitivity was found to be 22.86%, while specificity reached 87.88%. The Positive Predictive Value (PPV) stood at 57.14% and the Negative Predictive Value (NPV) was calculated at 61.7%. The Balanced Accuracy was calculated at 55.37% and provides a nuanced assessment considering class imbalances.

Facebook

For Facebook, the model achieved an accuracy of 58.49%, slightly surpassing the 57.86% NIR. However, statistical significance was not achieved. The kappa value was 0.1522, sensitivity was 52.24%, specificity was 53.04%, PPV was 50.72%, NPV was 64.44%, and the balanced accuracy was 57.64%.

Netflix

In the evaluation for Netflix, the model demonstrated a notable accuracy of 59.46%, significantly surpassing the 50.68% NIR. This difference was statistically significant, as evidenced by a calculated p-value of 0.01969. The kappa value was observed to be 0.1905, sensitivity was 65.75%, specificity was 53.33%, PPV was 57.83%, and NPV was 61.54%. Lastly, the balanced accuracy was determined to be 59.54%.

5.2 Artificial Neural Network, Closing Price

Company	**Apple**	Amazon	Facebook	Google	Netflix
MSE	**0.0026**	0.0058	0.0045	0.0083	0.0028
MAE	**29.611**	999.359	127.868	721.119	100.421
MAPE	**35.67**%	43.53%	60.58%	52.65%	25.90%

Table 5.2: *Artificial Neural Network*, Closing Price

The results of the MSE values reveals distinctive performance measures for the FAANG companies. Specifically, the MSE values were calculated as 0.0261 for Apple, 0.0058 for Amazon, 0.0045 for Facebook, 0.0083 for Google, and 0.0028 for Netflix. Despite the overall similarity in MSE among the companies, the mean absolute error (MAE) exhibits substantial variations. The MAE values differ significantly across companies, with Apple recording 29.611, Amazon at 999.359, Facebook at 127.888, Google at 721.120, and Netflix at 100.421. Furthermore, the analysis extends to the mean absolute percentage error (MAPE), providing additional insights into the forecasting performance. The MAPE values are reported as 35.66% for Apple, 43.53% for Amazon, 60.58% for Facebook, 52.65% for Google, and 25.90% for Netflix.

5.3 Random Forest, Variable Importance

Variable	Importance
Price High	3 770 344
Price Low	2 228 510
Open Price	104 599
Volume	162.7183
USD Exchange rate	136.2492
Real interest rate	42.0861
EBITDA Margin	14.6893
Enterprise Value	14.302
Company market cap	13.9404
Loan rate	11.3250
EBIT	10.5038
EV/EBIT	8.5286
EPS	8.2954
Debt	6.4673
Price over Earnings	6.4591
Revenue	6.276
EBITDA	6.0972
Price over Sales	6.0392
EV/EBITDA	5.551
Cash Flow	0.8095

Table 5.3: Variable importance for Random Forest

In the table above we find an example of the function *varImp* which is included in the Random Forest-package. The score is calculated based on a Random Forest fitted model and tells how much a certain feature contributes to the overall model performance. As we can see in the table the key features are the ones that are connected to daily trading patterns, namely the technical features. The predictors that contributed the least was the fundamental features, such as *Cash flow* and *EV/EBITDA*.

5.4 Random Forest, Classification

Company	**Amazon**	Apple	Facebook	Google	Netflix
Accuracy	**0.5605**	0.4438	0.5786	0.509	0.5068
95% CI	**(0.4792, 0.6395)**	(0.3675, 0.5221)	(0.4979, 0.6564)	(0.4306, 0.587)	(0.4234, 0.5898)
No Information Rate	**0.5669**	0.5562	0.5786	0.5808	0.5068
P-Value [Acc >NIR]	**0.5967**	0.9987	0.5337	0.9745	0.5329
Kappa	**0.1671**	0	0.0362	0.0415	0.0167
Sensitivity	**0.8088**	1.0000	0.1194	0.3000	0.6301
Specificity	**0.3708**	0.0000	0.9130	0.6598	0.3867
Pos Pred Value	**0.4955**	0.4438	0.5000	0.3889	0.5000
Neg Pred Value	**0.7174**	NAn	0.58741	0.5664	0.5179
Balanced Accuracy	**0.5898**	0.5000	0.5162	0.4799	0.5084

Table 5.4: Random Forest, Classification

Amazon

As the table above shows the NIR of 56.69% the classification was performed on a balanced data set. Achieving a Kappa value of 0.1671 indicates that the predictions was almost equal to random. Sensitivity of 80.88%. Specificity turned out to be 37.08%. The PPV of 49.55% suggests that the model could predict a decrease in the stock price value, with a probability of said percentage. NPV was calculated to 71.74% which indicates the models ability to predict true negative cases. The balanced accuracy for Amazon 58.98%

Apple

For Apple we received a NIR of 55.62% which then also indicates a balanced data set. The Kappa value for Apple was determined to be 0, which indicates that the predictions were equal to random. Sensitivity was calculated to 100% and specificity was calculated to 0. PPV was calculated to 44.38% and NPV couldn't be computed as there was no true negatives. Balanced accuracy turned out to be 50%.

Facebook

Facebook received an accuracy of 57.86%. And a NIR of 57.86%. The calculated Kappa for Facebook 0.0362 which similarly to Apple is close to random. Sensitivity turned out to be 11.94% . Specificity was calculated to 91.3%. PPV had a value of 50% whilst NPV received a value of 58.74%. Balanced accuracy was calculated to 51.62%.

Google

Google had a NIR of 57.86%. The model received a Kappa value of 0.0415 which also is in accordance to earlier models, and close to random. Sensitivity was calculated to 30% and specificity received a value of 65.98% . PPV was calculated to 38.89% and NPV to 56.64%. Lastly the balanced accuracy was 47.99%.

Netflix

NIR for Netflix was 50.68%, which is on the line for being an imbalanced data set. The calculated Kappa was 0.0167. Sensitivity was calculated to 63.01%. Specificity turned out to be 38.67%. PPV for Netflix was 50% whilst NPV 51.79%. Lastly the balanced accuracy for Netflix was 50.84%.

5.5 Random Forest, Closing Price

Company	Amazon	Apple	Facebook	Google	Netflix
MSE	4 170 883	5 431.942	323.4144	1 345 567	81 476.74
MAE	1 993.744	71.7858	11.8927	1 151.8790	279.9341
MAPE	83.90%	85.25%	5.05%	82.52%	68.20%

Table 5.5: *Random Forest*, Closing Price

As we can see in the table below the Random Forest-predictions gave widely different results for all companies, as the asset values differs in range as there were no standardization. For Amazon we received a MSE of 4170883, Apple was 5431.942, Facebook was 323.4144, Google 1345567 and lastly Netflix with 81476.74. The mean absolute error for Amazon was calculated to 1993.744, Apple was 71.78576, Facebook became 11.89268, Google at 1151.879 and Netflix at 279.9341. MAPE was for Amazon calculated to 83.89503%, Apple at 85.2473%, Facebook at 5.0546%, Facebook at 82.7154% and Google at 68.2032%.

6 Analysis

As addressed in the introduction, the primary aim of this thesis was to answer the question: *How effectively can Random Forest and Artificial Neural Network models predict the closing price for the FAANG companies the following day?* In the upcoming chapter, the tests will be discussed and analyzed individually. The main focus is the analysis of each model and to evaluate their effectiveness. We also explore potential improvements and considerations for future researchers when forecasting stock prices.

6.1 Classification Test

In the previous chapter the results from our models were presented and it can be stated that ANN had a higher overall performance for predicting whether the closing would increase or decrease the upcoming day, compared with random forest. It had higher accuracy and also lower p-values. However only Netflix had an accuracy which was statistically significantly higher than the NIR. The positive Kappa value of 0.1905 further supports the conclusion that the model's performance exceeds what would be expected by random chance, indicating a higher level of agreement. Sensitivity stands at a noteworthy 65.75%, highlighting the model's effectiveness in capturing instances when the stock price for Netflix is expected to decrease. In addition to these metrics, the model exhibits a specificity of 53.33%, showcasing moderate success in correctly identifying instances when the stock price is expected to increase. The Positive Predictive Value (PPV) of 57.83% indicates that the model's predictions of positive movements are correct approximately 57.83% of the time. The Negative Predictive Value (NPV) of 61.54% reflects the model's accuracy in predicting instances of negative movements. These comprehensive metrics collectively underscore the model's strength in both positive and negative predictions. The overall findings suggest that the model for Netflix not only performs well in terms of overall accuracy, statistical significance, and balanced accuracy but also exhibits a nuanced ability to correctly identify both positive and negative instances, showcasing a robust and effective predictive capability.

Although Apple, and Facebook didn't have an accuracy that was statistically significantly higher than the NIR, a quite high Kappa value for the companies were yielded. The high values suggests a predictive capacity beyond random chance. For Apple, the model's sensitivity of 49.32% suggests an ability to correctly identify instances of of the price going down. The specificity of 68.75% further indicates success in recognizing increasing closing price the upcoming day. The Balanced Accuracy, accounting for class imbalances, stands at 59.03%, providing a nuanced assessment of the model's performance. Similarly, for Facebook the model's sensitivity of 52.24% demonstrates effectiveness in capturing positive instances. The specificity of 53.04% reflects success in identifying negative instances. The Balanced Accuracy for Facebook is calculated at 57.64%. In summary, these results suggest that the models for both Apple and Facebook have the ability to capture essential patterns in closing price movements, as evidenced by high Sensitivity and Specificity values. However the ANN did not predict the price fluctuations for Apple and Facebook as well as it did for Netflix.

Just like Apple and Facebook, Google did not get a statistical higher accuracy than the NIR. However, the KAPPA was not as high as it was in the case for Facebook and Apple. One can therefore conclude that there is a modest agreement in predictions beyond random chance. The sensitivity was found to be 22.86%, suggesting a relatively lower ability to capture positive instances, the specificity reached a notably high 87.88%, indicating a strong ability to correctly identify negative instances. The PPV stood at 57.14%, showcasing the precision of positive predictions, while the NPV was calculated at 61.7%, reflecting the accuracy in negative predictions. The Balanced Accuracy, accounting for class imbalances, was calculated at 55.37%. This suggests a relatively balanced performance. In summary, the model for Google exhibits a favorable accuracy, particularly in correctly identifying negative instances. While statistical significance was not achieved, the Balanced Accuracy and other metrics provide a nuanced understanding of the model's overall performance, highlighting its strengths and areas for improvement.

For Amazon, however The negative Kappa value (-0.0386) indicates performance below random chance. This challenges the model's reliability, further emphasized by sensitivity and specificity values of 22.73% and 73.63%, respectively. This suggests that the model struggles to correctly identify instances of the positive class, resulting in a lower ability to capture a decrease in the closing price the following day. The Positive Predictive Value (PPV) and Negative Predictive Value (NPV) were calculated at 38.46% and 56.78%, respectively. The Balanced Accuracy, considering class imbalances, was observed at 48.18%. In this case, the model of Amazon faces challenges, as reflected in the negative Kappa value and low sensitivity, while the specificity is relatively high. The overall performance suggests limitations in effectively capturing positive movements in closing prices. The model's performance for Amazon appears less robust.

Looking at the prediction outputs for Random Forest we find that the accuracy level for all companies were relatively similar to each other with Amazon and Facebook standing out at a 56% and 57% respectively. The poorest accuracy performance was made by the model based on Apple data with 44.38%. However, we don't find any statistically significance in any of the models accuracy as P-value [Acc >NIR] were all significantly higher than the 5% level. But there are other important performance metrics to look at which are of great interest. There is no model that clearly outperforms the other models, as it often is a weigh-off between different metrics such as sensitivity and specificity. Apple did have however have some very interesting results, as the Kappa value was exactly null rounded at four decimals, this indicates that the models predictive performance could be purely due to chance or randomness. And as we can see in the P-value we received 0.9987 which indicates that there are almost no significance at all. It also exhibited a sensitivity value of 100% and specificity of 0%. The Apple-model did also receive a negative predictive value of NAn since the model did not predict any true negative values.

In general the models had a quite low Kappa value, however Amazon with the highest achieved a value of 0.1671. This indicated that are some predictions made beyond random chance, whilst the other circled positively over 0. But if we see to the sensitivity and specificity of Amazon and Facebook, we find that the Amazon model did quite well in sensitivity achieving 80.88%. Facebook on the other hand, had quite low sensitivity of 11.94%. As the importance of specificity and sensitivity depends on ones specific goals one must do a weigh-off.

If we see to the different PPV and NPV of the different companies we find a greater variation in the models accuracy to predict true positives and true negative best could predict true positives where Facebook and Netflix performed better with both a PPV of 50.00%, and Amazon close by with a PPV of 49.55%. If we look to the various accuracy of predicting true negatives we find that Amazon performed best with a NPV of 71.74% and Apple that performed worst since it did not predict any true negatives.

The last metric for comparison between companies we have is balanced accuracy. Here we find a relatively even result between all companies ranging from 47.99% to 58.98%, with Google and Amazon at the bottom and top respectively. One can interpret this to that Amazon had a better overall accuracy if one takes imbalances between classes into account.

To summarize, it can be difficult to determine which Random Forest that had the best performance as it is up to ones specific goals. However, we would claim that either Amazon or Netflix had the best overall performance with a relatively even spread across the different metrics discussed in this section. Since Amazon, is the only one with a predictive power higher than 0.0 one could argue that the other models was purely based on chance and therefore we would claim that the Random Forest was best at predicting based on Amazon data.

6.2 Closing Price

In the pursuit of predicting the exact closing prices through the ANN models, the evaluation metrics provide valuable insights into the models' performance.

The MSE shows that Apple yielded the lowest at 0.0026, indicating a relatively accurate prediction of closing prices. Amazon and Netflix also exhibit favorable MSE values, suggesting reasonable predictive capabilities. However, Facebook and Google display higher MSE values, indicating a potential room for improvement in the models' accuracy for these companies.

The ANN yielded the lowest MAE for Apple, just as for the MSE. Notably, the MAE for Amazon is unexpectedly high, signaling potential challenges in accurately forecasting closing prices for this particular company. Facebook and Google also show relatively high MAE values, suggesting a need for refinement in predicting the closing prices for these entities.

Netflix stands out with the lowest MAPE, indicating a more precise prediction compared to other companies. Apple and Amazon display moderate MAPE values, while Facebook and Google exhibit higher values, indicating areas where the models may benefit from improvement in predictive accuracy.

In summary, while the ANN models demonstrate competence in predicting closing prices for certain companies, variations in performance among the individual companies highlight the nuanced challenges inherent in stock price prediction. The unexpected high MAE for Amazon warrants further investigation into factors influencing its predictability. The divergent performance across companies emphasizes the need for tailored approaches and continuous refinement in model architecture and feature selection.

Random Forest also saw some difficulties in predicting the closing price of the stocks. As the different regression metrics showed high errors in relation to the value range of the variables.

As what comes to MSE for Random Forest we find that the predictions for Facebooks closing price were the lowest with a value of 323.4144, which indicates a relatively poor prediction accuracy for the closing price. However, we received even higher for Amazon with a value of 4170883 which alarms for some concern regarding the prediction capability of Random Forest with the used data set.

The prediction that received the lowest MAE was also Facebook with a value of 11.89 which also indicates a relatively poor performance if one sees to the range of Facebooks closing price, 17.73 to 268.44. And due to the formula of MAE it comes to no surprise that Amazon received the highest MAE with a value of 1993.744.

MAPE will however shine some insight to the accuracy, as a percentage-based metric. Here we find some interesting results where Facebook received a MAPE of 5.05% which indicates a relatively good prediction. While the other companies exhibited greater signs of a poor capability to predict the closing prices, as for example Apple received a MAPE of 85%.

In summary of the Random Forest capability to predict the closing price of the given companies. We have found only one out five cases where, the prediction could be seen as relatively good and that was for Facebook. However the different metrics indicates that the model had difficulties in predicting the price of the stock in the year of 2020. In this test it is crucial to take the range of each companies stock value into account, as the closing prices were not standardized.

In summarizing this test to determine which models outperform others in predicting the closing price, providing a definitive answer is challenging, unlike the classification test. The main reason for this lies in the fact that the ANN used standardized variables while the Random Forest did not, yielding much lower results from the ANN.

6.3 Improvements and Previous Research

We can conclude from the previous chapter that there were some fluctuations regarding the performance of our models. However, for most parts the accuracy of our predictions were quite low, and this can come from various reasons. After examining our main response variables *Closing Price* we find that over time it has a what seems to be a linear trend, but after a certain point in time it develops random walks that never were seen in the earlier years. Why this could be an issue for our models performance is due to how we split our training and validation data, as our models were trained on data which almost never expressed the same trends that the validation data contained. When predicting the "unseen" data of 2020, our models haven't learned how to adapt or pattern this type of trend which then affects our predictions. What also supports that the split of data set could be a reason for the performance, at least for Random Forest, is that before the 2010-2019/2020 split we first sampled training and validation data sets using random sampling by computational algorithms. At this test the Random Forest performed generally well on the training data but also on the validation data, which could explain that the algorithm in this attempt learned to pattern the trends from the period of 2010-2020 and therefore could make relatively accurate predictions. As the year of 2020 was widely influenced by the Covid-19 pandemic, which started in the early 2020 and most sectors were affected by it. This could also have affected everything from institutional investors to private investors in their decisions, which could lead the discussion into behavioral finance.

In comparison to the previous research presented in section 2.2 our models did not perform as well as it did for Nair, Mohandas and Sakthivel. In their study the artificial neural network had an accuracy of 77,66% and the decision tree model had an accuracy of 90.22%, when predicting the Bombay Stock exchange. As shown in the variable importance the technical variables were the key features to contribute in making the decision tree splits. Our data set only contained five types of technical variables, while Nair, Mohandas and Sakthivel used 21. The discrepancy in variable richness raises questions about the adequacy of our feature set. The few number of technical variables in our dataset could be a limiting factor, potentially contributing to the models' relative underperformance compared to the study by Nair, Mohandas, and Sakthivel.

This study finds quite similar results as the paper Tsung-Sheng Chang published. When studying the closing price for Taiwanese digital game content stocks between 1 January 2018 and 31 June 2019 he found that the ANN outperforms the random forest, just like our study. However, the stocks were only studied between 1 January 2018 until 31 June 2019, which is fairly short period of time studied. This might have caused a quite high and false accuracy. Our study didn't quite yield as high accuracy, which might is caused by the studied period, which spans from 2010 to 2020.

7 Conclusion

The presented models, Artificial Neural Network (ANN) and Random Forest, demonstrated lower-than-expected accuracy, particularly in comparison to benchmarks from other studies. The disruptive impact of the COVID-19 shock notably affected the predictive capabilities of the Random Forest, highlighting its sensitivity to abrupt events. In contrast, the ANN model, designed for non-stationary objectives and noisy processes, showcased greater resilience during the possibly challenging conditions induced by the pandemic, emphasizing the significance of model adaptability in navigating unforeseen disruptions. Other contributing factors to the lower accuracy include insights from the study by Nair, Mohandas, and Sakthivel, indicating that a higher number of technical variables may result in more effective and accurate predictions. This suggests a consideration for future studies to incorporate a more extensive set of technical variables. Additionally, examining a shorter time period, as demonstrated in the study by Tsung-Sheng Chang, may lead to higher accuracy. Despite falling short of the anticipated accuracy, our study contributes to the field by moving beyond a sole focus on accuracy. Unlike previous studies, which predominantly examined accuracy, our analysis delves into a more nuanced evaluation of the models' statistical performance. This approach provides a comprehensive understanding of how well our models performed in various aspects, contributing valuable insights to the broader field of financial forecasting.

In conclusion, addressing the research question - *"How effectively can Random Forest and Artificial Neural Network models predict the closing price for the FAANG companies the coming day?"* - our models yielded diverse outcomes. Some models demonstrated promising results with high Kappa values, low MSE values, and one even exhibited statistically significant accuracy lower than the no information rate. However, a notable number of models did not achieve high accuracy and performed inadequately across various performance metrics. As a result, we cannot assert that our models were robust predictors.

Incorporating more technical variables, considering the impact of significant global events, such as pandemics, and potentially studying the stock market over shorter time frames might enhance the accuracy of predictions. These factors should be considered in future research to refine predictive models for closing stock prices.

8 References

Al-Jabery, K, Obafemi-Ajayi, T, Olbricht, G & Wunsch II, D (2020). 2 - Data Preprocessing. https://www.sciencedirect.com/science/article/pii/B9780128144824000024?ref =pdf-download&fr=RR-2&rr=83d207765ce195f6 [Accessed 2023-10-27]

Ayfer Ezgi Yilmaz, Haydar Demirhan (2023), "Weighted kappa measures for ordinal multiclass classification performance". *Department of Statistics, Hacettepe University, Ankara, Turkey & Mathematical Sciences Discipline, School of Science, RMIT University, Melbourne, Australia*, Vol 134.

Baldbridge, Rebecca (2022). "What is the efficient market hypothesis?" . https://www.forbes.com /advisor/investing/ efficient-market-hypothesis/ [Accessed: 2023-10-23]

Binoy.B.Nair, V.P Mohandas, N. R. Sakthivel (2010). "A Decision Tree- Rough Set Hybrid System for Stock Market Trend Prediction", *International Journal of Computer Applications*, vol 6, No. 9

Chiao-Yi Chang (2011). "The relationship between the 52-week high of an individual stock and stock market index level: Evidence from Taiwan", *Department of Insurance and Finance, National Taichung Institute of Technology*, Vol. 21, No. 129, page 14-27.

Diederik P. Kingma & Jimmy Lei Ba (2015), "ADAM: A METHOD FOR STOCHASTIC OPTIMIZATION", *ICLR 2015*

Esfe, M.H., Eftekhari, S.A., Hekmatifar, M. et al. "A well-trained artificial neural network for predicting the rheological behavior of MWCNT" –Al2O3 (30–70%)/oil SAE40 hybrid nanofluid. Sci Rep 11, 17696 (2021). https://doi.org/10.1038/s41598-021-96808-4

Federal Reserve Bank of St Louis (2023). https://fredhelp.stlouisfed.org/fred/about/about-fred/what-is-fred/ [Accessed 2023-10-04]

Fradkov (2020), "Early History of Machine Learning" *Institute for Problems in Machine Engineering, Russian Academy of Sciences, Saint-Petersburg.*

Hayes, Adam (2022): Technical Analysis. https://www.investopedia.com/terms/t/ technicalanalysis.asp [Accessed 2023-10-08]

Hitchner, James (2011). Financial Valuation: Applications and models. Hoboken: Wiley.

IBM. (2023), "What is Machine Learning?" [Electronic] Available: https://www.ibm.com /topics/machine-learning, Accessed: 2023-12-29.

IBM. (2023), "What is a neural network?" [Electronic] Available: https://www.ibm.com /topics/neural-networks, Accessed: 2023-12-29.

Isaac Martín De Diego, Ana R. Redondo1, Rubén R. Fernández1, Jorge Navarro, Javier M. Moguerza (2022), "General Performance Score for classifcation problems", *Applied Intelegence*, Vol. 52.

Kartika Maulida Hindrayani, Tresna Maulana Fahrudin, Tresna Maulana Fahrudin, Eristya Maya Safitri, 2020. " Indonesian Stock Price Prediction including Covid19 Era Using Decision Tree Regression", *IEEE*, vol 1, No. 9, Page 344-347.

Levitt, Aaron (2023): Macroeconomic Indicators That Affect the US Stock Market. https://www.investopedia.com/articles/investing /031413/economic-indicatiors-affect-us-stock-market.asp [Accessed 2023-10-08]

Lo, A.W. (2008). Efficient Markets Hypothesis. In: The New Palgrave Dictionary of Economics. Palgrave Macmillan, London. https://doi.org/10.1057/978-1-349-95121-5_42-2

M. Bicego, A. Mensi, 2023. " Null/No Information Rate (NIR): a statistical test to assess if a classification accuracy is significant for a given problem", *University of Verona*

Mario Pisa Peña (2018). "Decision Tree For Trading Using Python". https://blog.quantinsti .com /decision-tree/ [Accessed 2024-01-05]

Katherine (Yi) Li, Neptune.ai. (2023). "Predicting Stock Prices Using Machine Learning" [Electronic] Available: https://neptune.ai/blog/predicting-stock-prices -using-machine-learning, Accessed: 2023-12-29

Kurtis Pykes, Neptune.ai. (2023). "Fighting Overfitting With L1 or L2 Regularization: Which One Is Better?" [Electronic] Avilable: https://neptune.ai/blog/fighting-overfitting-with-l1-or-l2-regularization, Accessed: 2023-12-29

Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky , Ilya Sutskever, Ruslan Salakhutdinov (2014)."Dropout: A Simple Way to Prevent Neural Networks from Overfitting", *Department of Computer Science, University of Toronto.*

Oshiro, T.M., Perez, P.S., Baranauskas, J.A. (2012). How Many Trees in a Random Forest?. In: Perner, P. (eds) Machine Learning and Data Mining in Pattern Recognition. MLDM 2012. Lecture Notes in Computer Science(), vol 7376. Springer, Berlin, Heidelberg. https://doi.org/10.1007/978-3-642-31537-4-13

Qifang Bi, Katherine E. Goodman, Joshua Kaminsky, and Justin Lessler. 2019. "What is Machine Learning? A Primer for the Epidemiologist", *Oxford University press*, vol 188, No. 12.

Riebesell, Janosh (2021). "Random Forest". https://tikz.net/random-forest/ [Accessed 2024-01-05]

S. B. Kotsiantis (2013), " Decision trees: a recent overview", *Educational Software Development Laboratory, Department of Mathematics, University of Patras.*

Timothy O. Hodson, Thomas M. Over, Sydney S. Foks, 2021. "Mean Square Error, Deconstructed." *Journal of Advances in Modeling Earth System*, Vol 13, Vol 12.

Rupesh A. Kamble, 2017, “Short and long term stock trend prediction using decision tree *International Conference on Intelligent Computing and Control Systems*, Vol 1, No.1.

Ummul Khair, Hasanul Fahmi, Sarudin Al Hakim, Robbi Rahim, 2017. “Forecasting Error Calculation with Mean Absolute Deviation and Mean Absolute Percentage Error”, *Journal of Physics: Conference Series*, ser. 930.

Wu Muh-Cherng, Lin Sheng-Yu, Lin Chia-Hsin, 2006. “An effective application of decision tree to stock trading”, vol 31, No 1.

Weijie Wang, Yanmin Lu, 2018. ” Analysis of the Mean Absolute Error (MAE) and the Root Mean Square Error (RMSE) in Assessing Rounding Model”, *CFETS Information Technology (Shanghai) Co., Ltd. China Foreign Exchange Trade System, Shanghai, China*, vol. 1, No.1.

9 Appendix

Code for ANN, Classification

```
model <- keras_model_sequential()

model %>%
  layer_dense(units = 400, activation = "relu",
  input_shape = ncol(scaled_predictors_train)) %>%
  layer_dropout(rate = 0.3) %>%
  layer_dense(units = 300, activation = "relu",
  kernel_regularizer = regularizer_l2(0.001)) %>%
  layer_dropout(rate = 0.3) %>%
  layer_dense(units = 200, activation = "relu",
  kernel_regularizer = regularizer_l2(0.001)) %>%
  layer_dropout(rate = 0.3) %>%
  layer_dense(units = 100, activation = "relu",
  kernel_regularizer = regularizer_l2(0.001)) %>%
  layer_dropout(rate = 0.3) %>%
  layer_dense(units = 1, activation = "sigmoid",
  kernel_regularizer = regularizer_l2(0.001))

  model %>% compile(
  optimizer = "adam",
  loss = "mean_squared_error",
  metrics = c('accuracy')
)

predictor_matrix_train <- as.matrix(scaled_predictors_train)
```

```
history <- model %>% fit(
  x = predictor_matrix_train,
  y = y_train,
  epochs = 300,
  batch_size = 400,
  validation_split = 0.2
)

plot(history)
```

Code for ANN, Closing Price

```
model <- keras_model_sequential()

model %>%
  layer_dense(units = 400, activation = "relu",
  input_shape = ncol(scaled_predictors_train)) %>%
  layer_dropout(rate = 0.3) %>%
  layer_dense(units = 300, activation = "relu",
  kernel_regularizer = regularizer_l2(0.001)) %>%
  layer_dropout(rate = 0.3) %>%
  layer_dense(units = 200, activation = "relu",
  kernel_regularizer = regularizer_l2(0.001)) %>%
  layer_dropout(rate = 0.3) %>%
  layer_dense(units = 100, activation = "relu",
  kernel_regularizer = regularizer_l2(0.001)) %>%
  layer_dropout(rate = 0.3) %>%
  layer_dense(units = 1, activation = "linear",
  kernel_regularizer = regularizer_l2(0.001))


model %>% compile(
```

```
  optimizer = "adam",
  loss = "mean_squared_error",
  metrics = list("mean_squared_error")
)

summary(model)

predictor_matrix_train <-as.matrix(scaled_predictors_train)

history <- model %>% fit(
  x = predictor_matrix_train,
  y = scaled_y_train,
  epochs = 300,
  batch_size = 400,
  validation_split = 0.2
)

plot(history)
```

Code for Random Forest, classification

```
Skog_classification_meta<-randomForest(‘Dummy_variable‘~‘Price_High‘+
‘Price_Low‘+‘Open_Price‘+‘Volume‘+‘EBITDA_Margin‘+
‘Enterprise_Value‘+‘Debt‘+‘Cash_flow‘+
‘Company_market_cap‘+‘Revenue‘+‘EBIT‘+‘EBITDA‘+
‘EPS‘+‘Realränta‘+‘Låneränta‘+‘EV_EBITDA‘+
‘EV_EBIT‘+‘USD_Växelkurs‘+‘P_E‘+‘P_S‘,
                   data = data_train,
                   ntree = 2000,
                   mtry=20,
                   na.action=na.omit)
```

Code for Random Forest Regression

```
Skog_classification_meta<-randomForest(‘Dummy_variable‘~‘Price_High‘+
‘Price_Low’+ ‘Open_Price‘+‘Volume‘+‘EBITDA_Margin‘+
```

```
‘Enterprise_Value‘+‘Debt‘+‘Cash_flow‘+
‘Company_market_cap‘+‘Revenue‘+‘EBIT‘+‘EBITDA‘+
‘EPS‘+‘Realränta‘+‘Låneränta‘+‘EV_EBITDA‘+
‘EV_EBIT‘+‘USD_Växelkurs‘+‘P_E‘+‘P_S‘,
                   data = data_train,
                   ntree = 2000,
                   mtry=20,
                   na.action=na.omit)
```