



# STOCK PRICE FORECASTING USING TIME SERIES ANALYSIS

Higher Diploma in Science in Data Analytics

**Pawel Jankowski**  
**Student No: 10510292**  
**Email: [10510292@mydbs.ie](mailto:10510292@mydbs.ie)**  
**June 2020**

## Contents

Introduction .....	2
Aims and Objectives.....	2
Aim .....	3
Objective .....	3
Methods .....	3
Methodology.....	3
Data Selection .....	4
Data Exploration .....	4
Candlesticks .....	5
Stock Volume .....	6
Simple Moving Average .....	7
Predictive modelling .....	9
ARIMA .....	9
PROPHET .....	9
Methodology.....	10
Smoothing.....	10
Cross-Validation .....	10
Comparing performance.....	13
Forecasting.....	15
Limitations .....	16
Conclusion.....	17
References .....	18
List of Figures .....	18

# Introduction

The Stock Market refers to the collection of markets and exchanges in which the purchase, sale and issue of shares of publicly held companies take place regularly. The share of the stock is the share of the company's equity interest. The investor buys company shares anticipating portion of the profits in the form of dividends or benefiting from the rises in its share price, or both. (Chen, 2020)

The stock market is among essential components of the economic system since it provides businesses with access to capital by enabling investors to purchase ownership stakes in a company. Commodity prices are not entirely random values and could be treated as a discrete time-series model based on a set of numeric values collected successively at regular time intervals.

Forecasting stock exchange data includes the assumption that the information currently available to the public has some predictive relationship to future stock returns. Stock trend forecasting is one of the most complicated things to achieve in the financial market, given the difficulty in the highly intricate world of the stock market. The trends of stock prices are non-linear and non-stationary time-series data. (Reddy, 2018)

Time Series analysis consists of techniques for evaluating historical data to extract meaningful statistics and data characteristics; and predict future values by identifying patterns, seasonality and cyclical behaviour. Many businesses work on time series data to measure the number of sales, website traffic and number of calls received: such data can then be used for forecasting purposes.

Components of a Time Series:

- Trend: A general direction in which something develops or changes. In the stock market, the uptrend is an overall price increase, while the downtrend is marked by a price decrease.
- Seasonality: Predictable change or pattern that is repeated at a regular time interval. For example, a time-series data that repeats its pattern every 12 months.

## Aims and Objectives

The primary purpose of this project is to implement, compare and facilitate the use of statistical models. Such models should not be treated as oracles, but rather as an aid in the analysis of movements in stock prices.

Time series forecasting involves the creation and use of a predictive model by analysing historical data that has a relationship between observations. The decision taken based on the models directly impacts every step of the project, including the evaluation of the forecast models on the fundamental difficulties of the forecast research. Stock price forecast to attempt to estimate future values for the stock of any company.

## Aim

The main goal of this study is to build a user-friendly tool that helps predict the future stock prices of the chosen company. The objective is to make it easier for experts and non-experts to make high-quality stock-value forecasts which help in understanding the market and make a wise decision to invest in the stock market.

To achieve it, I decided to develop an interactive dashboard consisting of user input controls and responsive visualisations.

## Objective

Project focus:

- comparison between several forecasting models
- model evaluation (accuracy assessment)
- presentation of various aspects of stock time series
- accessible ways to fine-tune desired parameters
- application of the models in forecasting future values

## Methods

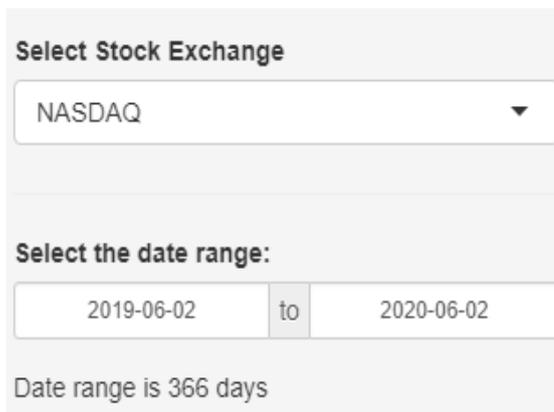
- Simple Exponential Smoothing (SES)
- AutoRegressive Integrated Moving Average (ARIMA) methods
- Automatic Forecasting Procedure (PROPHET)

## Methodology

- Data is collected from Yahoo Finance using the Quantitative Financial Modeling Framework (Quantmod). Data obtained in eXtensible-Time-Series format is being used for data exploration.
- Time series models – SES, ARIMA and PROPHET as stock price prediction methods.
- Data split into training and testing sets for model validation.
- Model performance gauged using Root Mean Square Error metrics.
- Trained models applied to derive future data points (forecasting) with 80% and 95% confidence intervals.
- For deployment purposes and user interface, I chose Shiny App. Shiny is an R package that makes it easy to build interactive web apps straight from R. The app can be hosted locally, on a dedicated private server or publicly available, cloud-based virtual server.

## Data Selection

I selected three major stock exchanges, AMEX, NASDAQ and NYSE, to serve as a source stock selection for further analysis. Shiny quantmod package provides methods to acquire not only most recent but a wide range of historical records. In this project, user has an option to define data range (Figure 1).



Select Stock Exchange

NASDAQ

Select the date range:

2019-06-02 to 2020-06-02

Date range is 366 days

Figure 1: Selection of Stock Exchange (data source)



Search: GO

Symbol	Company Name
GOODN	Gladstone Commercial Corporation
GOOG	Alphabet Inc.
GOOGL	Alphabet Inc.
GOSS	Gossamer Bio, Inc.
GPAQ	Gordon Pointe Acquisition Corp.
GPAQU	Gordon Pointe Acquisition Corp.
GPAQW	Gordon Pointe Acquisition Corp.
GPRO	GoPro, Inc.
GSHD	Goosehead Insurance, Inc.
GT	The Goodyear Tire & Rubber Company

Showing 31 to 40 of 62 entries (filtered from 3,601 total entries)

Figure 2: Selection of stock of interest

The selected Exchange contains a large number of stocks. Each stock is assigned a unique symbol, making selection of one desired stock quite a difficult task. The dashboard includes a data-table which provides search, filter and sorting capabilities - significantly improving user experience (Figure 2).

## Data Exploration

The first section shows necessary information about the selected company - last sale price, market capitalisation, sector, and industry, along with the stock time series plot giving us an idea of the general trend and stock prices over the selected timeframe (Figure 3).

last.sale.price	market.cap	ipo.year	sector	industry
1431.82	\$977.39B	2004.00	Technology	Computer Software: Programming, Data Processing



Figure 3: Basic stock pieces of information and close price chart

More advanced plots, which illustrate the acquired data in more detail can be seen in the next section. These graphs are useful for short-term analysis and help to interpret current market trends.

## Candlesticks

The first one is the "candlestick" graph. It originated in Japan more than 100 years before the Western world developed similar charts. In the 1700s, a Japanese man named Homma discovered that, while there was a link between price and supply and demand of rice, the markets were greatly influenced by the emotions of merchants. (Northcott, 2009). The Candlestick graph shows these emotions by visually representing the size of price movements in different colours. Traders can use this graph to make trading decisions based on regularly occurring patterns that help predict the short-term direction of prices. (Mitchell, 2010).

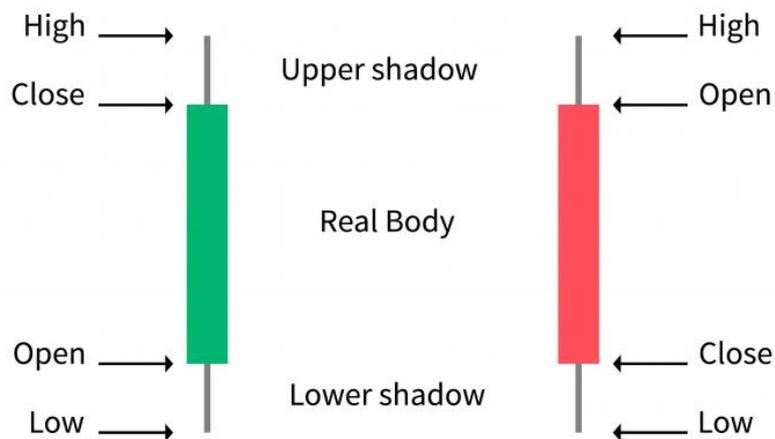


Figure 4: Example of candlestick chart components

The Candlestick chart shows the market's open, close, low and high prices for the day. The candlestick consists of a large part, which is called the "real body," which represents the price range between the opening and closing prices for that trading day. When the real body is filled-in or red, it means that the closing price was lower than the opening price. If the real body is empty or green, it means that the closing price was higher than the opening price.

The "shadows" or "wicks" are above and below the real body. They represent high and low prices of that day's trading. If the upper shadow on the red candle is short, it indicates that the opening price was near the top for that day. A short upper shadow on a green candle dictates that the closing price was near the day's top. The relationship between day's open, high, low, and close determines the appearance of the daily candlestick. Real bodies can be long or short, red, or green. Shadows may be long or short.



Figure 5: Candlestick graph

Candlesticks are created by up and down movements in the price. While these price movements sometimes appear random, at other times they form patterns that traders use for analysis or trading purposes.

## Stock Volume

Volume is the amount of an asset or security that changes hands over a period of time, often in the course of a day. For example, the volume of stock trading would refer to the number of security shares traded between its open and close daily price. Volume trading and changes in volume over time are essential inputs for technical traders. (Hayes, 2020)

Volume is used to measure the relative importance of any market move. If the market moves by a substantial number over a given time period, then the strength of that movement either boosts credibility or is treated with scepticism based on the volume for that period. The higher the volume during the price shift, the more significant it is considered in this form of analysis. Alternatively, if the volume is low, the movement is viewed with far less relevance.

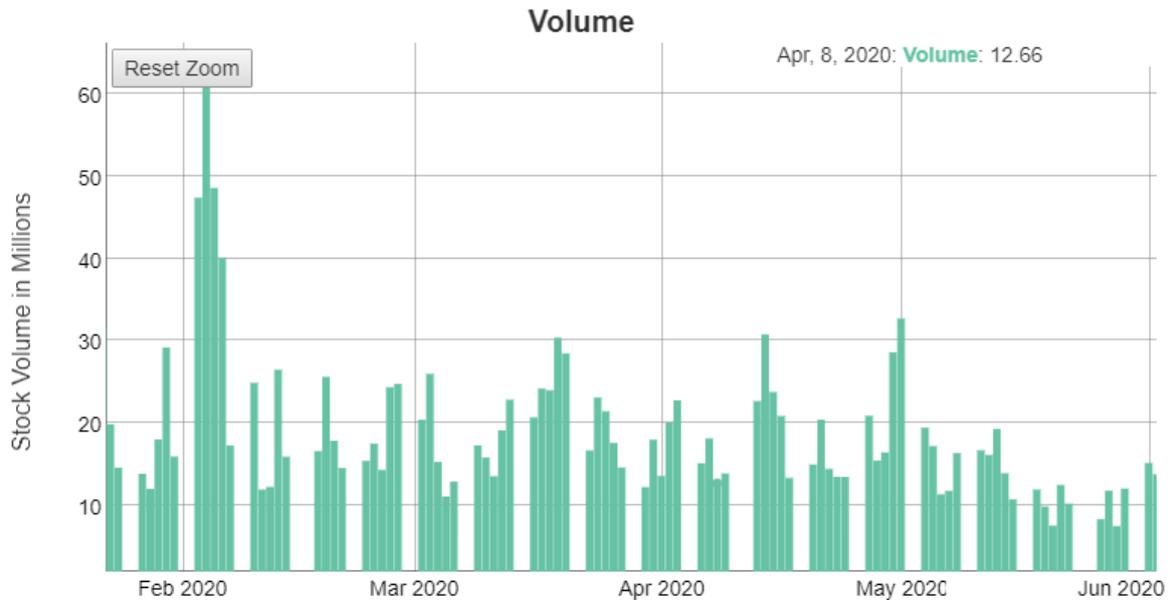


Figure 6: Stock Volume chart

### Simple Moving Average

The purest form of a moving average, known as a simple moving average, is calculated using the arithmetic mean of a given set of values. In other words, a set of stock prices is added together and then divided by the number of prices in the set. (Hayes, 2020)

The formula for calculating the simple moving average of security is as follows:

$$SMA = \frac{A_1 + A_2 + \dots + A_n}{n}$$

Figure 7: Simple Moving Average formula

Where:

$A_n$  = the price of an asset at period  $n$

$n$  = the number of total periods

A 10-day moving average is a mean across closing prices for 10 days before the data point. In simple terms, the result if SMA is a new collection of data points, where first ten are used to calculate first point of new set. Second data point is calculated using points 2 to 11 from original set. This way resulting dataset contains less datapoints than the original, and they values are much smoother.

Likewise, a 50-day moving average would accumulate 50 consecutive days of data and calculate the mean on a rolling basis.

Moving averages are an important analytical method used to identify current market patterns and the potential for a shift in the current trend. The primary use of an SMA in an analysis is to quickly

determine whether the security is in uptrend or downtrend. Another common, although slightly more complex, analytical usage is to compare a pair of simple moving averages with each of different time frames. If the shorter-term simple moving average is higher than the longer-term average, an uptrend is expected. On the other hand, if the long-term average is above the shorter-term average, the expected result could be a downtrend.



Figure 8: Two SMA trends imposed on stock close price plot

Two popular trading patterns that use simple moving averages include the Death Cross and the Gold Cross. A death cross occurs when the 50-day SMA passes below the 200-day SMA, known to be a bearish indication that more losses are anticipated. The golden cross happens when a short-term SMA breaks over a long-term SMA, indicating a bullish trend. (Hayes, 2020)

The graph in our dashboard allows the users to adjust the desired SMA timeframes to fine-tune the desired strategy. The additional feature of the graph is changing the shade according to the relative values of the averages - when the shorter-term SMA is above the shade is pink, when the positions are reversed, the shade changes to green.

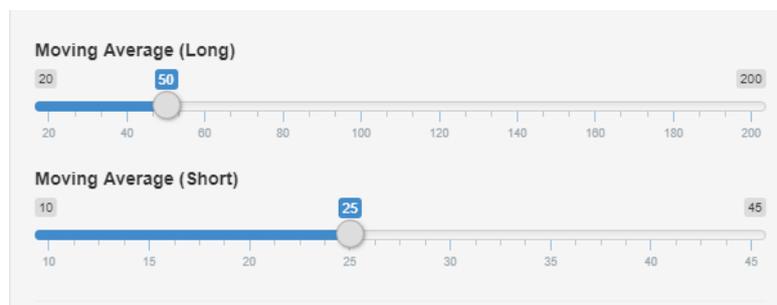


Figure 9: User interface sliders for SMA adjustment

# Predictive modelling

There are many algorithms which can be used to create prediction models, but for this project, I have chosen the two popular models – ARIMA and PROPHET. These two methods are widely used in time series forecasting. While ARIMA is a popular model for detection of trends and seasonality, PROPHET (by Facebook) uses far more advanced additive regression. It also handles well trend and seasonality detection but also tolerates reasonable number of missing observations, large outliers and historical trend changes.

## ARIMA

Autoregressive Integrated Moving Average (ARIMA) is a generalisation of the ARMA model. The model's "AR" part refers to the regression of its own lagged values while the "MA" part indicates that the regression error is a linear combination of values that occurred in the past at different times. The differentiation may be performed more than once. The purpose of each of those features is to make to model best fitting the data while remaining as general as possible (Chen, 2019)

For forecasting using ARIMA model, we need to assume that the time series is stationary, the residuals are uncorrelated and normally distributed.

In general, future values in the ARIMA model are derived from a linear combination of past values and errors. Non-seasonal model denoted ARIMA(p,d,q) with parameters

- p – order (number of time lags) of the autoregressive model.
- d – degree of differencing – number of times where the values have had the past values subtracted)
- q – order of moving average model

## PROPHET

Prophet is Facebook's open source project available for Python and R (Sean J. Taylor, 2017). It is optimized for large business forecast tasks, which would typically have any of the following characteristics:

- temporal observations (hourly, daily, weekly) with at least few months (preferably a year) of observations
- strong seasonality
- important holidays of regular events known in advance (sports world series, etc.)
- reasonably small number of missing observations and large outliers
- historical trend changes
- linear and non-linear trends (curves)

Prophet is very robust and straightforward. It produces as accurate forecasts with default settings compared to those created by skilled analysts, using much less effort. The procedure is completely automated and can be modified and tweaked if the results are not satisfactory.

The core of Prophet is an additive regression model with few main components:

- Linear or logistic growth curve trend detection, selecting changepoints from data

- Yearly/Monthly seasonal components using Fourier series
- Weekly seasonal components using random variables
- User-provided list of important holidays

## Methodology

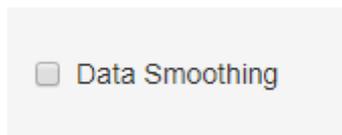
The goal of this project is to deliver an easy to use tool, which can be used as an aid in creating forecasts, and not to be treated as an oracle. Since it will be used on such inherently random time series data it is not possible to create one model to fit all possible inputs.

In case of ARIMA model I decided to use Auto-ARIMA procedure, which automatically checks and detects optimal parameters for given time series. Results produced by this method are very close to optimal forecasts based on prior feature detection. Since each dataset (stock) presents unique trends and features, it is impossible to select optimal parameters in easy manner. Here is where Auto-ARIMA shows its potential.

PROPHET's default settings are very robust and perform very well with many tasks: it is possible to tweak them. It is beyond the scope of this project, but could be considered as potential for implementation in the future.

## Smoothing

In some cases the dataset may include a lot of noise, which could make forecasts less accurate. In order to mitigate the tool contains data smoothing parameter.



*Figure 10: Data smoothing checkbox*

Smoothing is implemented using R's SMA function, which calculates moving averages of predefined order. In this case it uses last two days to calculate moving average value (SMA) for each data point. This reduces data noise while keeping trend consistent. It also improves results in some cases while reducing required computation power for generating forecasts.

## Cross-Validation

In order to select best model/method that will need specific requirements, we should be able to gauge the model's performance in specific scenarios. Depending on the investment strategy the user might consider different initial parameters such as source dataset period and desired forecast

length. Short-term or day-trading is focusing on very recent input data to make rapid predictions. Long-term investments may take much longer input period, but be more focused on forecasting much further into the future.

Cross-validation section of the project allows for easy and dynamic adjustment of such parameters. While it does not yet produce any real forecasts, it uses known data to gauge the performance of each model and given parameters.

The input dataset is split into two parts – Train-Set and Validation-Set. The model is then optimized using training data, generates the predicted values and compares them on the actual withheld data.



Figure 11: Short term forecast of TSLA stock using Auto ARIMA

The results are presented to user in form of charts and numerical scores. The example chart (Figure 10) presents a typical visualisation of the validation process. The TSLA stock has been split into train and test datasets. The ARIMA model was used to produce the forecast (blue line on the chart), with low and high predictions based on 80% and 95% confidence intervals (orange and yellow ribbons respectively). The red line represents hold-out data, which is then superimposed on top of the predicted values. As we can see they differ slightly. This difference can then be used to calculate error score for the prediction using RMSE (root mean square error) formula

$$RMSE = \sqrt{(f - o)^2}$$

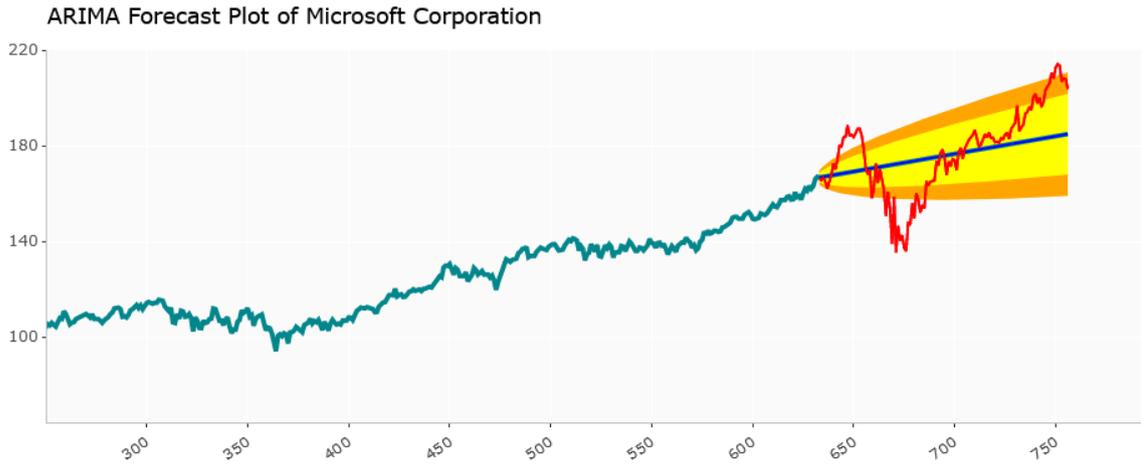
Where:

f – forecasts (expected values of unknown results)

o – observed values (known results)

This formula calculates the sum of absolute point-differences in expected and actual values. Better predictions should have lower RMSE score, but very rarely this will be close to zero.

We can now compare both models and decide which suits best for our requirements.

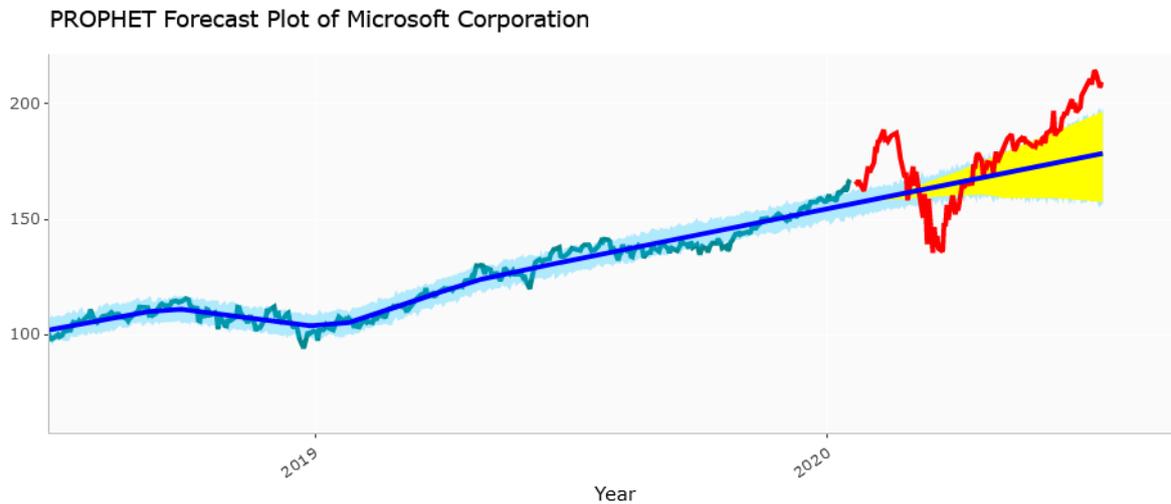


RMSE: 14.87

Figure 12: ARIMA cross validation and RMSE score

In this example I have selected MSFT (Microsoft Corporation) stock with 180 days cross validation period. It can be noticed that the general trend is aligned with the prediction, but due to recent COVID-19 outbreak, the financial markets are highly volatile and the predictions may be inaccurate. The model's performance is moderately accurate, and actual data falls mostly within the confidence interval boundaries.

Note the RMSE resulted to 14.87, which can be compared to the next model's score.



RMSE: 17.257

Figure 13: PROPHET Cross-validation with RMSE score

The same stock and validation period used with PROPHET model results in similar results. The trend detected by regression (thick blue line) is very clear and fitting the dataset very closely. Light blue ribbon representing 80% confidence interval is covering most of data distribution. The algorithm is

very confident with the forecast. Forecasted values are aligned with the actual values (red line) with 95% confidence interval (yellow area) not spreading too much. Due to COVID-19 crisis the recent data used for validation causes high error rate, but again the algorithm still performs reasonably well. RMSE score is slightly lower than before, at 17.257.

The tool allows selection input data time frame and validation period, which might be helpful in tweaking the parameters to user's liking.

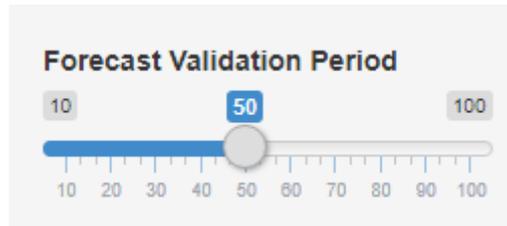


Figure 14: Selection of validation period using the slider

It is possible to include more controls for various model parameters, but this is outside of the scope in this project, but could be implemented in the future revisions.

## Comparing performance

After trying various combination of stocks and validation periods we can note that the models perform slightly differently for certain parameters. Since we can gauge the accuracy using RMSE score, I implemented a Monte-Carlo test comparing performance of both models for selected validation periods. This test will run multiple forecasts using defined period, increasing the cross-validation by small steps. Each result is then collected and displayed on a graph.

The application includes the parameter controls for easier adjustment of validation periods used in RMSE test.

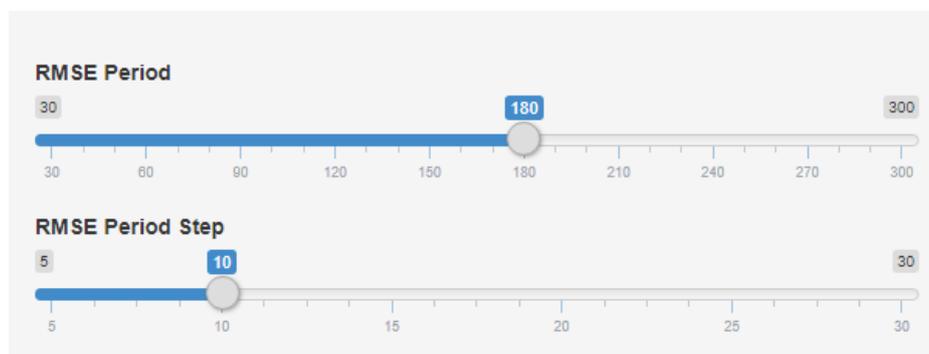


Figure 15: Parameter adjustment controls for RMSE test

“RMSE period slider” controls the total length of validation period. “RMSE Period Step” defines the number of days for increasing the validation period from 10 to defined maximum.

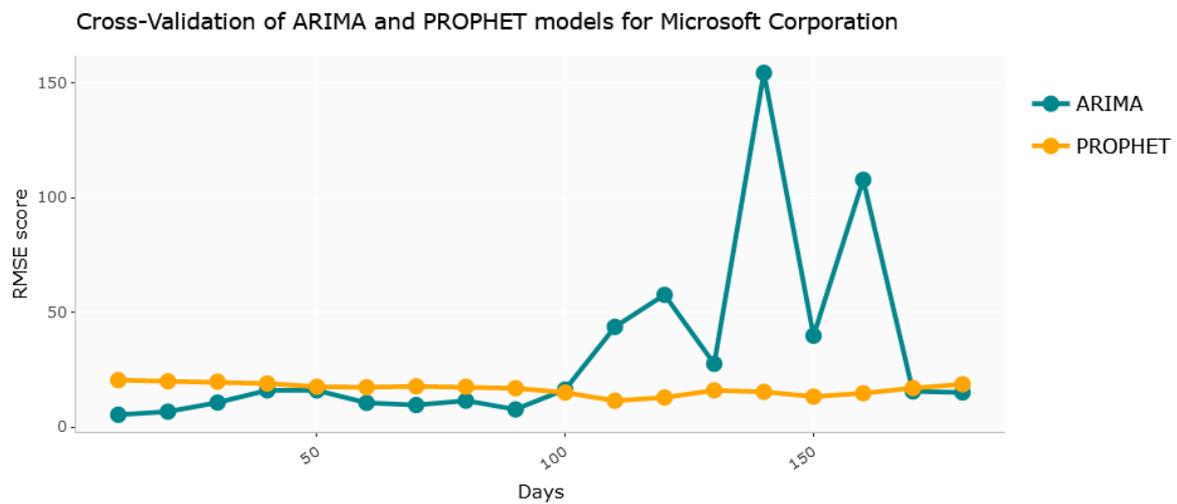


Figure 16: RMSE comparison for MSFT stock

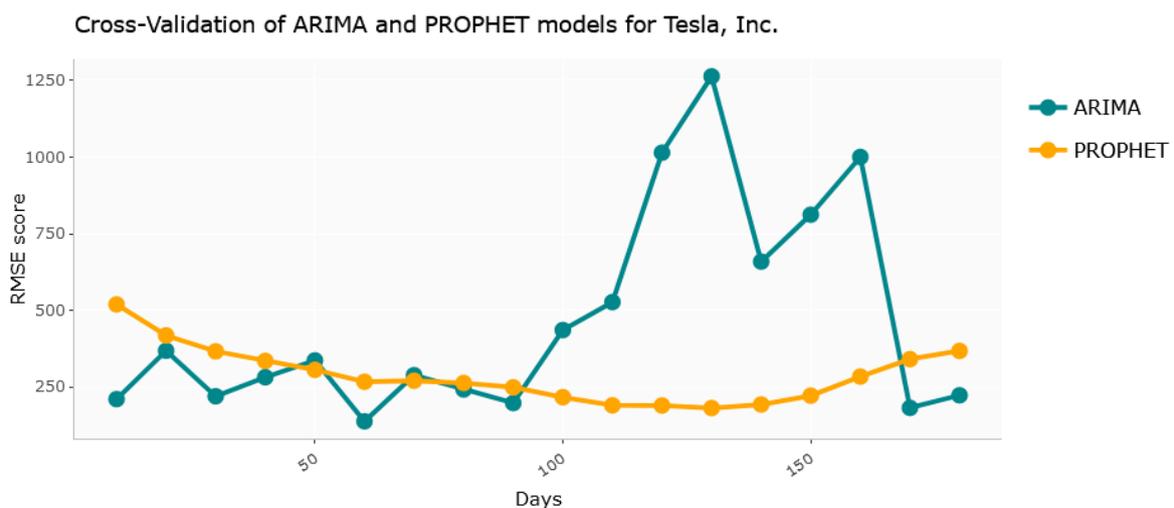


Figure 17: RMSE comparison for TSLA stock

The graphs above show how RMSE score changes for different parameters. Y-axis represents the score value (lower is better), and X-axis represent cross-validation periods (10 to 180 days, incrementing by 10). The graphs are a great illustration of each model’s performance in terms of forecast length. On both graphs we can see that they are comparable for 10-90 days period, where ARIMA is performing slightly better, but is not very consistent. Beyond 100 days the forecasts generated by ARIMA fail to compete with PROPHET. It also shows that PROPHET is very consistently producing accurate forecasts for the entire validation period.

In conclusion – for short term forecasts both models can be used with reasonable confidence, but for long-term forecasts PROPHET produces far better results.

## Forecasting

Finally, after we have tried different parameters we can run actual forecast of the future unknown values. The tool generates both forecast, but it is up to the user to select one which is better suited for desired purpose.

I generated longer term forecast for Tesla Inc stock using three years of historical data. As we can see the stock value was mostly stable and noted several surges and drops in recent times. Right now it is on a steadily increasing trend.

ARIMA takes it as a very positive signal and produces very generous forecast, surpassing value of \$3000 in next 180 days, with end margin between \$2200 and \$4000 with 80% confidence.

ARIMA Forecast Plot of Tesla, Inc.

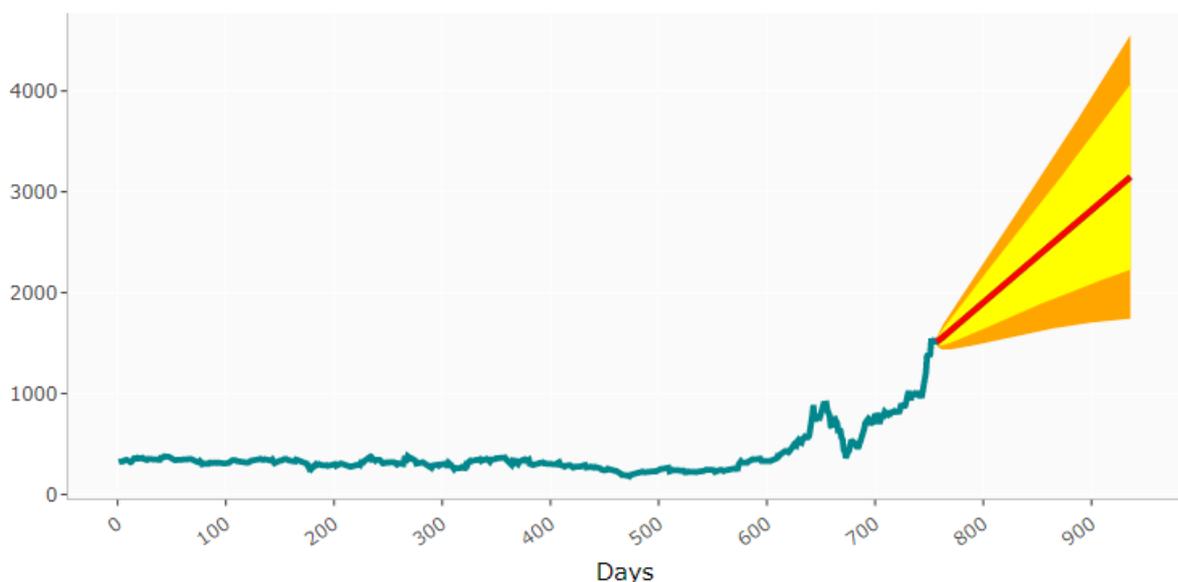


Figure 18: ARIMA forecast for TSLA (180 days)

PROPHET produces more realistic forecast, and isn't easily triggered by recent fluctuations. The main trend predicting value of \$1660 with very narrow margin ranging from \$1580 and \$1730. This shows the algorithm has a rather high confidence in this matter. It is impossible to tell at this time which algorithm is right. This can only be verified later, especially taking into account very volatile market situation in recent months. It is worth noting, however, that RMSE test for this stock is greatly favouring PROPHET for longer term predictions.

## PROPHET Forecast Plot of Tesla, Inc.

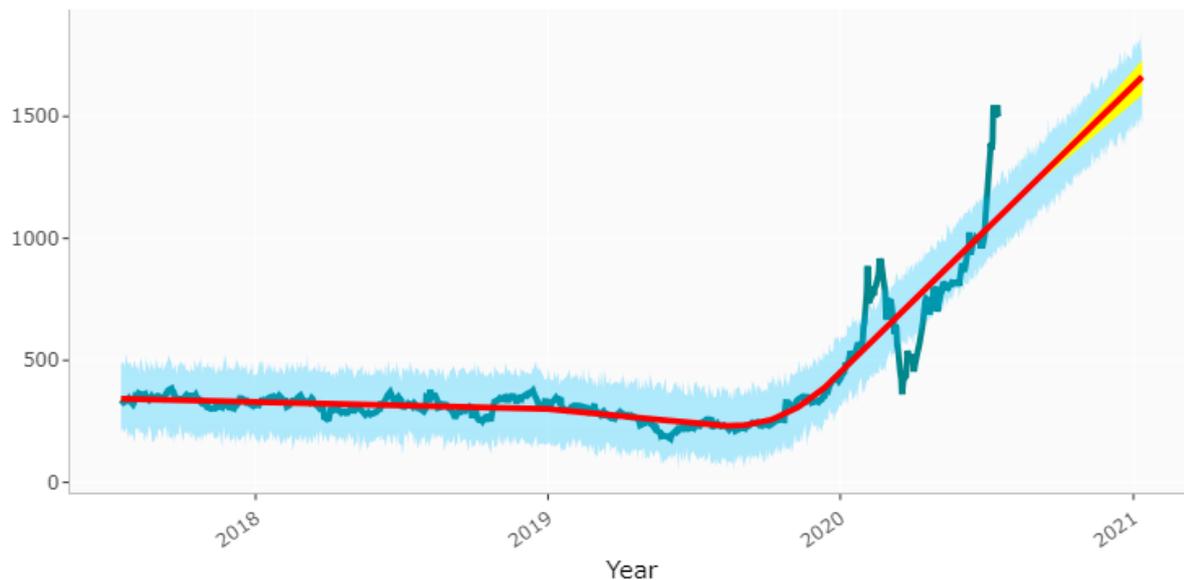


Figure 19: PROPHET forecast for TSLA (180 days)

## Limitations

Selected models are using only a single input variable. While this method might be useful, there are more advanced algorithms. Algorithms such as Neural Networks can make use of many more, seemingly uncorrelated variables, for example, Stock Indexes, Oil and commodity markets, Real-estate markets, or even textual data from articles and social media. While those methods would generate more accurate predictions, they require far more data processing and computational power.

Alternatively, it might be possible to create an ensemble model, utilizing strengths of multiple models to further improve the accuracy and performance.

## Conclusion

Creating accurate forecast for such volatile data as stock markets takes a lot of skill, experience and time. Unfortunately, each time, and even each day, this forecast may become outdated and unusable. Fortunately, using modern statistical methods and having high computational power at our disposal, it became possible to create fast and fairly well performing models. Such models might still require advanced programming knowledge of Python, R or other statistical environments. This was the main reason to implement easy-to-use interface, which could be accessible and friendly for an average user with no programming experience. As we learned in this document, the results vary depending on many factors, but with help of easily adjustable parameters they can be tweaked to optimize forecasting accuracy.

During development of this app I gained a lot of valuable experience not only in analyzing stock markets, but also in understanding the logic behind forecasting, testing and validation. Most importantly, I gained an invaluable experience of running, understanding and programming in R and Shiny. These platforms combined with wide collections of libraries allow rapid development of user friendly and powerful applications.

The apps can be run locally, but R-Studio provides very easy way of deployment of the app on dedicated servers. This is a very important feature as it gives the developers great way of delivering the app to a large user base across internal company network or even public internet servers.

The app developed in course of the project is available live under following address:

<https://p4xel.shinyapps.io/DBS-Forecasting/>

Source code is also included as a separate file for review.

## References

- Chen, J., 2019. *Investopedia - ARIMA Technical Analysis*. [Online]  
Available at: <https://www.investopedia.com/terms/a/autoregressive-integrated-moving-average-arima.asp>  
[Accessed 04 06 2020].
- Chen, J., 2020. *Stock Market Definition - Investopedia.* [Online]  
Available at: <https://www.investopedia.com/terms/s/stockmarket.asp>  
[Accessed 03 06 2020].
- Hayes, A., 2020. *Investopedia - Simple Moving Average (SMA)*. [Online]  
Available at: <https://www.investopedia.com/terms/s/sma.asp>  
[Accessed 3 6 2020].
- Hayes, A., 2020. *Investopedia - Volume*. [Online]  
Available at: <https://www.investopedia.com/terms/v/volume.asp>  
[Accessed 03 06 2020].
- Mitchell, C., 2010. *Investopedia - Understanding Basic Candlestick Charts*. [Online]  
Available at: <https://www.investopedia.com/trading/candlestick-charting-what-is-it>  
[Accessed 03 06 2020].
- Northcott, A., 2009. *The Complete Guide to Using Candlestick Charting: How to Earn High Rates of Return - Safely*. s.l.:Atlantic Publishing Group.
- Reddy, N. V. G. S., 2018. Stock market prediction using time series analysis. *International Journal of Statistics and Applied Mathematics*, Issue 2456-1452.
- Sean J. Taylor, B. L., 2017. *Prophet: forecasting at scale*. [Online]  
Available at: <https://research.fb.com/prophet-forecasting-at-scale/>  
[Accessed 16 07 2020].

## List of Figures

Figure 1: Selection of Stock Exchange (data source) .....	4
Figure 2: Selection of stock of interest .....	4
Figure 3: Basic stock pieces of information and close price chart .....	5
Figure 4: Example of candlestick chart components .....	5
Figure 5: Candlestick graph.....	6
Figure 6: Stock Volume chart .....	7
Figure 7: Simple Moving Average formula.....	7
Figure 8: Two SMA trends imposed on stock close price plot .....	8
Figure 9: User interface sliders for SMA adjustment.....	8
Figure 10: Data smoothing checkbox.....	10
Figure 11: Short term forecast of TSLA stock using Auto ARIMA .....	11
Figure 12: ARIMA cross validation and RMSE score .....	12
Figure 13: PROPHET Cross-validation with RMSE score .....	12
Figure 14: Selection of validation period using the slider .....	13

Figure 15: Parameter adjustment controls for RMSE test.....	13
Figure 16: RMSE comparison for MSFT stock .....	14
Figure 17: RMSE comparison for TSLA stock.....	14
Figure 18: ARIMA forecast for TSLA (180 days) .....	15
Figure 19: PROPHET forecast for TSLA (180 days) .....	16