Department of Computer Science

BSc (Hons) Business Computing

Academic Year 2019 - 2020

Predicting the next American Stock Market Crash by Evaluating The S&P Index Time-Series Models

Leslie J Sharp
1707077

A report submitted in partial fulfilment of the requirements for the degree of
Bachelor of Science
Total Words: 7640

Brunel University
Department of Computer Science
Uxbridge
Middlesex
UB8 3PH
United Kingdom
T: +44 1895 203397
F: +44 (0) 1895 251686

Abstract

The project analyses the behaviour of the S&P500 index in order to predict the next American crash, which could very well lead to a global recession. By diving into the area of financial markets, we identify and explain how fundamental and technical analysis can gauge the market for investments. As it advances into time series models with the exploration of three different schools of thoughts: No investor can gain an advantage over markets, fundamental analysis driven investing, and artificial neural networks can deliver a high degree of accuracy when predicting prices. The seasonal autoregressive integrated moving average model with external variables was selected, and various orders of SARIMAX were explored and analysed. Furthermore, compared with other models like ARIMAX; its non-seasonal version and auto ARIMA model which picked a SARIMAX model with fewer orders than the best version. The SARIMAX was determined to yield the best results when predicting S&P500 values. However, predicting financial markets is a difficult task. As markets inherently go by changing fundamentals and human psychology which gives rise to uncertainty; leading to volatility. In this paper, we examine how to determine volatility using the GARCH model and explore various orders of the model. Moreover, we discovered that a higher GARCH order outperformed a simple one. Moreover, we discovered the higher GARCH outperformed the simpler GARCH model. However, other papers have shown mathematically that simpler GARCH model is the best but not in this case. Nevertheless, we determined that the higher GARCH model yielded a lower margin of error and outperformed the simpler model, especially during frequencies of higher volatility. The results of the higher GARCH model will set the stage for the next study to determine why it outperformed the simpler model.

GitHub: https://github.com/lesliesharp8/Time-series-analysis-FYP

Table of Contents

1 Introduction

Stock prediction is a challenging task for financial time series prediction. Predicting the next financial crisis using the US economy has a primary measure due to its economic strength; for example, the dollar is a global reserve currency. Major crashes in the past were: The Great Depression of 1929, the 2000-2004 dot.com crash and the 2007-2008 global financial crisis. A range of information determines prices, and in the efficient market hypothesis, a change in information will immediately reflect in the price in the market. Since the first market crash of 1929, economists and analysts have been trying to predict future stock market prices to prevent significant losses in the markets. Due to technological advances, the use of new technologies such as predictive analytics and machine learning to determine future long-term price action; has made it easier and more accurately to predict future outcomes.

Historical observations of time series variables are analysed to develop a model describing the underlying relationship. The developed model is used to extrapolate into the future. The approach is useful when little knowledge is available on the underlying data generating process. Efforts for the development and improvement of time series forecasting models have occurred over the years. (Khashei and Bijari, 2011). However, the question remains; which models are more accurate for precisely predicting market values for the top economies like S&P 500 in USA and Dax30 in Germany. The project studies the two indexes, comparing and contrasting; to determine which models accurately predict current market values with the lowest margin of errors.

Moreover, we shall specifically look at S&P values more as the US market is one of the largest markets in the world. For example, in a paper by Qian and Gao: they explained that ‘ARIMA uses past information of the variable itself and errors in the past; GARCH can capture the variability of variance and helps judge volatility of stock return.’ Volatility is the measure of risk. However, they also note, ‘ARIMA only needs to extract information from price, machine learning models and deep learning models need to take in certain features into model training, which can include features aside from price.’ (Qian and Gao, 2017). Moreover, classical methods like ARIMA out-perform machine learning and deep learning methods for one-step forecasting and multi-step forecasting on univariate datasets. (Chatterjee, 2019)

In financial trading, there is a complicated trading mechanics called options trading; which is comparable with buying insurance. Using a car as an example: when purchasing a new car, usually one gets coverage and pays a premium. Hence, it results in having the right to claim funds if the vehicle is damaged or lost. The insurance company is obligated to pay. Options trading can minimise risk and volatility; helps investors and traders in times of market instability. In a paper by Engle; it shortly looked at the options and data to estimate risk in complex time series to understand enormous skew in index options volatilities by using GARCH models. (Engle, 2002)

In the long run, the best solution would be developing a complex system that takes into act the various economic indicators as a secondary input and the index or asset price as primary inputs. It looks at relationships between changes in risk assets and non-risk assets to gauge the market. For example, during the financial crisis’s precious metals like gold and the Japanese Yen increase in value; as they are safe-haven assets.

We observe the week of 9th March to 13th March 2020, global financial markets crashed due to many economic factors, but most notably was triggered by coronavirus fears. Furthermore, in the US market, the market crash ended its eleven-year bull market. It shows the need to develop a system that could predict and forecast prices and volatility in the market.

1.1 Aims and Objectives

This project aims to identify the events represented by the sequence of observations and determine which forecasting models are the most accurate form of stock market price prediction specifically for S&P500 index with the lowest margin of error by identifying patterns in the observed time series and describing the datasets.

Objectives:

1. Through the analysis of literature relevant to time series analysis about fundamental time series theory, time series models and machine learning methods, which demonstrated and explained the need for maximized accuracy and minimized bias to address the project's aim.
2. Based on the findings from objective one and using appropriate data processing methods that will establish patterns, we can interpret and integrate the datasets and extrapolate the identified trends to predict future events.
3. Using the results of the data processing from objective two, develop and implement various time series models to fit the data set and have a better model.
4. Compare the multiple outputs generated by each model to determine which model produces the most accurate prediction. Models will predict prices which are already known to determine the margin of error in the models. Determine which model has the lowest residual.
5. Project management occurs on GitHub with version control, and the use of a Gantt chart in time management, flowchart to identify the project approach and allocation of resources.
6. Effectively communicate relevant project ideas, strategies, solutions, and evaluation which occurs throughout the various stages of the project.
7. Ethical approval was approved.

1.1 Project Approach

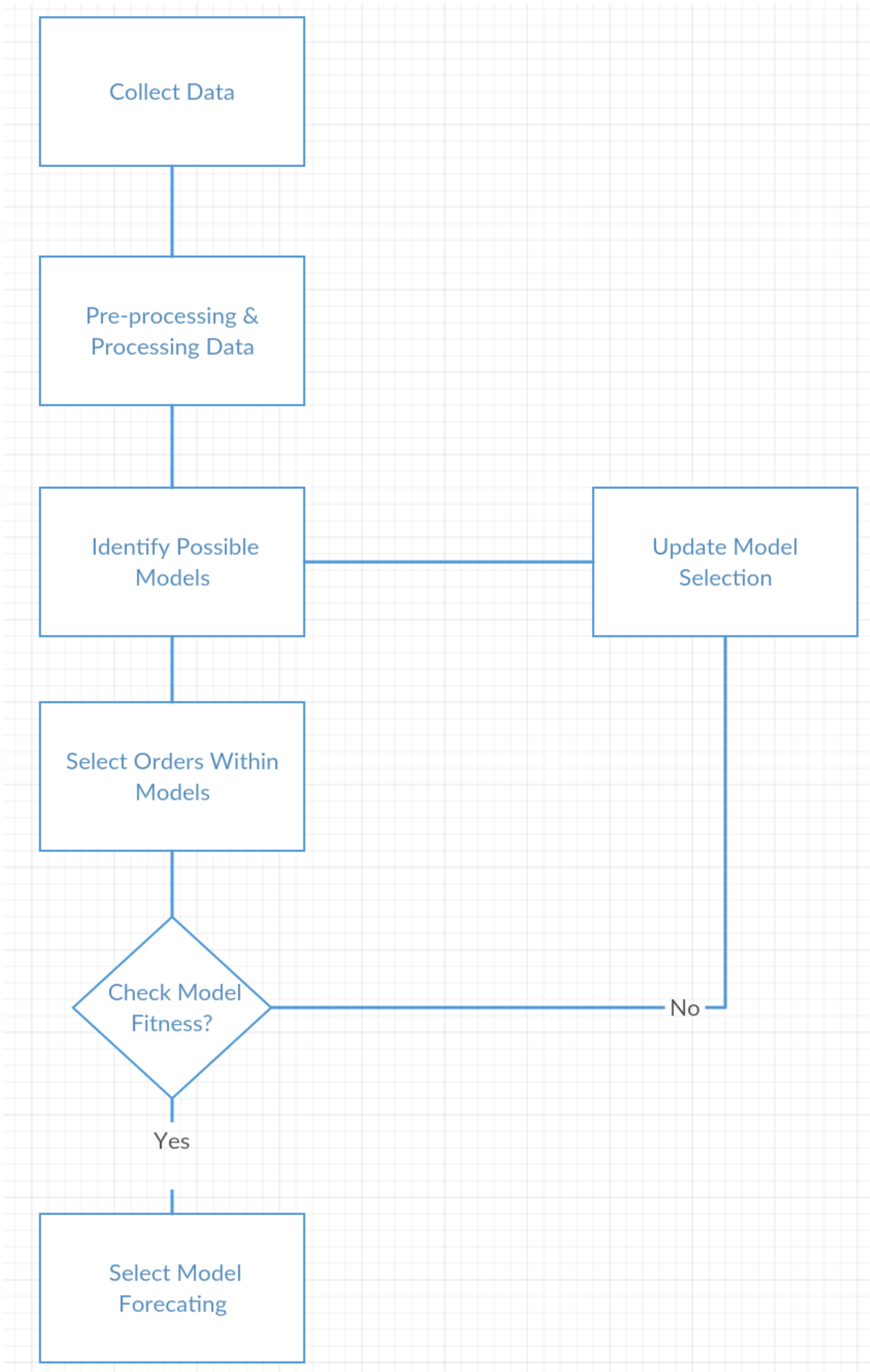


Figure 1 – Project Approach

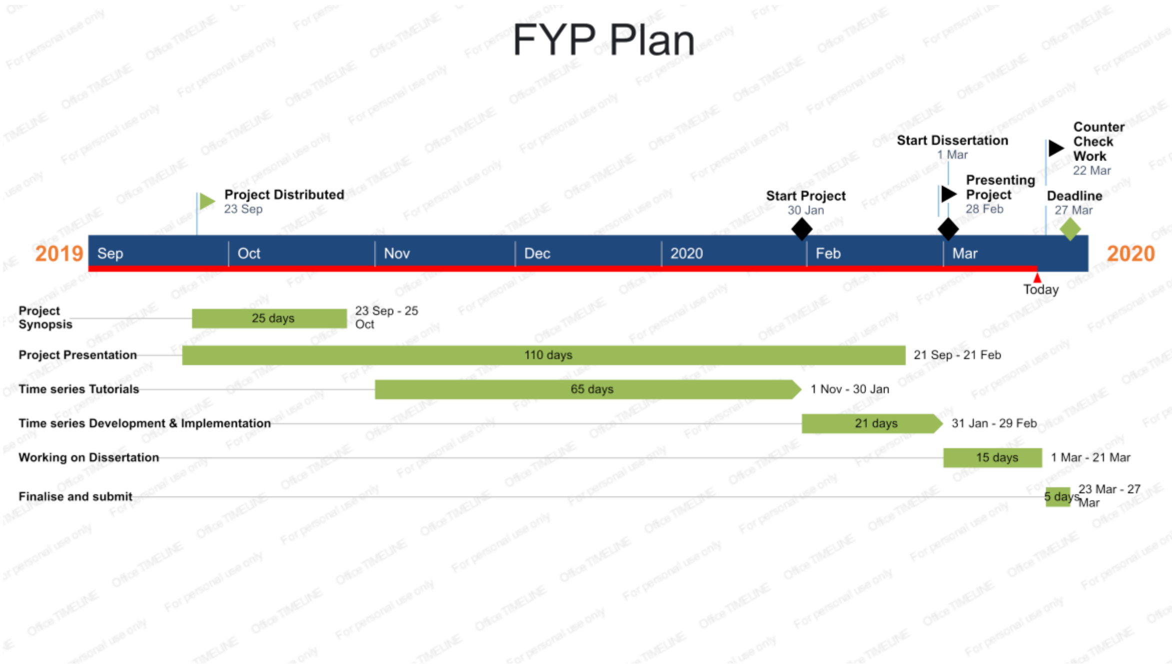


Diagram 1 – Gantt Chart

The first stage, data will be obtained from Yahoo Finance on Dax30 and S&P500; observe the two datasets between the period of 03/01/2000 to 31/01/2020 and pre-process the raw data in Excel. Carry out conditional formatting to only hold records of dates where both prices are available.

The second stage, importing and examining the data on Jupyter Notebook, determine the size of the dataset and identify any missing values. Plotting the data will help visualize how the data behaves and identify any unusual behaviours. Also, compare both graphs and determine data distribution.

The third stage, creating a time-series object by transforming the data frame into a time-series. Set a new index, frequency, and only record business day. Front fill any missing values and spilt the data into two: training data and testing data.

The fourth stage creates a white noise time series and compares the graphs of S&P500 and White Noise. Determine if the data is stationary or non-stationary by using the Dickey-Fuller Test (DF Test). Check for seasonality, both additive and multiplicative. Compute Autocorrelation Function (ACF), to determine the relationship between past and current values. Compute Partial Autocorrelation Function (PACF), to identify only a direct link between the time series and its lagged version. Finally, the project shall look at various simple models like the Autoregressive (AR) and Moving Average (MA) Models and building up the model complexity to Autoregressive Moving Average (ARMA) Model. Following a path of implementing these models: Autoregressive Integrated Moving Average (ARIMA) and Autoregressive Integrated Moving Average with Explanatory Variable (ARIMAX) for modelling non-stationary data. Then, use Generalized Autoregressive Conditional Heteroscedasticity (GARCH) for measuring volatility.

The fifth stage identifies the models that are useful and group them into one file directory. Finish off with forecasting volatility and begin to write up the report.

1.2 Dissertation Outline

Chapter 2 discusses the background literature for the project; identifying key areas of price forecasting like the basic forecasting methods used by trades: fundamental and technical analysis. Moreover, diving into traditional time series modelling, deep learning and other models which hardly get any attention. The chapter tries to argue for the use of conventional models and other traditional models to obtain greater predictive accuracy. The section covers objectives one, two and six.

Chapter 3 explains how the project will develop by exploring and expanding on the tools, steps and detailed approach of the project. The section will give details on the expected results when computing and analysing the datasets. Some critical process shall show how it potentially affect the overall outcomes of the forecasting section. The chapter covers objectives one, two and six.

Chapter 4 shows how the project underwent during the pre-processing data stage. It explores the difficulties encountered during this stage and develops mitigating solutions to overcome the issues that arise. The chapter covers objectives two and six.

Chapter 5 shows how the project was determining which appropriate models delivered the best results. Here different models are looked at; from a simple model to a complex model and explore the benefits and shortcomings of each model; while trying to improve on them. The chapter covers objectives three and six.

Chapter 6 focuses on evaluating the results obtained from the models and forecasting them to compare them to the actual values. It will determine which model predicts closer to the real value and compare the margin of errors of the different models. The chapter covers objectives four and six.

Chapter 7 is the conclusion section which sums up the results of the project and discusses whether the project was successful or unsuccessful. Furthermore, it looks at potential future work in ANNs and explores chaos theory and neural networks. The chapter covers objectives six.

2 Background

Financial forecasting has a long history in the mainstream financial markets. Traditional establishment methods for gauging future prices have been mainly fundamental analysis and technical analysis. Still, a new wave of technological revolutions is changing the way financial forecasting occurs at every level by using machine learning (ML) and traditional time series models.

2.1 Market Crashes

A crash is usually a rapid decline in market prices by a steep drop of double-digit percentage points losses in an index over several days. Corrections are drops of at least 10% in the index from the most recent peak point. Market crashes are different from market corrections; crashes are more sudden than market corrections. Crashes occur because of domestic or international issues; driven by panic and economic factors. Mainly, due to over-leveraging; bubbles happen in the market. Price bubbles translate into inefficient markets. Crashes lead to recessions. (En.wikipedia.org, 2020).

A belief is that the markets run on Dow Theory; which operates on the efficient markets hypothesis (EMH), stating that market movements are similar to a random walk. Meaning, it is tuff to predict the market due to its random nature accurately. (En.wikipedia.org, 2020). But, saying that markets are efficient is incorrect, as the market is involved with human interactions, thus creating inefficiencies as humans are prone to error and irrational thinking.

Furthermore, government intervention causes more significant inefficiencies. However, understanding market inefficiencies and predicting when price bubbles will occur is a difficult task. However, it’s important to note that financial markets crash due to other reasons. And using other economic indicators like housing prices, industrial production, GDP, unemployment rate and CPI for measuring inflation.

2.2 Fundamental Analysis

Fundamental analysis tries through the analysis of economic and political data or data from companies to determine future price actions within the market. It determines intrinsic value & studies the cause of price movements; this type of analysis has the goal of 'why' or the cause of the change. It is usually best for medium to long term forecasting (Ganti and Segal, 2019). However, it is limited to trend predicting; by only assisting in long term forecasting instead of short term (Qian and Gao, 2017). Yet, fundamental analysis can be too simple with an over-reliance on macroeconomic factors which can be easily manipulated by any party or source party to make the information look useful to their benefit.

2.3 Technical Analysis

Technical Analysis is a methodology for determining future price action through the study of past market values, usually price (primary input) and volume (secondary input). It Illustrates market sentiment and momentum for short to medium periods; it studies the effects on the market (Kirkpatrick II and Dahlquist, 2011). Usually, both technical and fundamental analysis are used together. These two are the traditional approach for predicting prices. Technical Analysis is like the point where forecasting starts to gain complexity and more accuracy in predictions. Charles Dow first introduced technical analysis and the Dow Theory in the late 1800s. He developed two assumptions that have formed the framework for technical analysis, (Murphy, 1999) which are: Markets are efficient with values representing factors that influence the price. However, price action is hardly random but moves in identifiable patterns and trends that tend to repeat over time. (CFA Institute Research Foundation, 2016). The second assumption is more important as we shall see later on that a critical appropriation of time series data: patterns observed or found in the past persist in the future. Hence, trying to predict the future by analysing recorded values.

2.4 Traditional Time Series Models

A time series is a sequence of information or numerical data points which attaches a time-frequency to each value. 'Time series forecasting uses information regarding historical values and associated patterns to predict future activity. Most often, this relates to trend analysis, cyclical fluctuation analysis, and issues of seasonality. And, not all forecasting tools lead to success.' (Kenton, 2019). AR relies on past period values and prior periods, only to predict current period values. It is a linear model, where current period values are a sum of previous outcomes multiplied by a numeric factor. The primary difference between AR and MA is that AR uses the value of the variable, while MA relies on the residual or some error term (Losada, n.d.).

ARIMA model has three components and seen as the best predictor in time series analysis. Its expressed in two forms: non-seasonal models where the model is in the form (p,d,q). P is the autoregressive model order, d is the order of differencing, and q is the moving average order. AR models are similar to a regression model but the regressor in this case is the same dependent variable with a specific lag. Differencing (I) from ARIMA; the model needs stationary data to predict better, meaning the mean and variance are constant over the dataset. Hence, differencing transforms the data into stationary data. (Qian and Gao, 2017).

An author stated that 'some major disadvantage of ARIMA was the identification techniques for identifying correct models were difficult and depended on the skill of the forecaster. Also, just like other forecasting methods, are 'backwards-looking'. Such that, the long term forecast eventually beings to straighten the line and is poor at predicting series with turning points.' (Libres.uncg.edu, n.d.). Mitigating the first challenge is easy. The second point on turning points, however not stated I assume the author was talking about seasonality or unpredicted events. In that case, a seasonal version of the model would implement well. The AR model needs time to adjust from unexpected shocks. But the MA is a self-correcting model with more errors examined, the more the model adapts. Thus, the MA component gives the model prediction a much higher chance to move in a similar direction to the values it is trying to predict.

2.5 Deep Learning

A deep learning artificial recurrent neural network which I came across was the long short-term memory (LSTM) which is said to model univariate and multivariate time series problems. (Brownlee, 2018). 'It can divide the dataset into smaller batches and train into multiple stages. LSTM is said to perform better than traditional predictive models. However, research shows ARIMA could focus on univariate data with linear relationships, fixed and manually diagnosed temporal dependence. ARIMA yields better results in forecasting short term, whereas LSTM yields better results for long term modelling. It is a more complex model which sometimes underperforms compared to ARIMA model.' (Chatterjee, 2019).
I came across articles looking at artificial neural networks (ANNs), which are becoming popular because they have self-learning capabilities that enable better results with more data, learn to detect intricate patterns in data and provide a bridge to fundamental analysis. Neural networks can learn mathematical relationships between a series of inputs and corresponding output variables and 'have successfully improved numerous forecasting applications, but with several issues in ANN model building still unsolved.' (Zhang et al., 1998). ANNs often suffer overfitting problems; they fit training data very well but forecast poorly on the test set.

2.6 Schools of Thoughts

I discovered three schools of thoughts regarding prediction. 'The first school believes that no investor can achieve above-average trading advantages based on historical and present data, the major theories of which contain Random Walk Hypothesis and Efficient Market Hypothesis (Peters, 1996), which was defeated by the persuasive evidence in the document (Taylor, 1986). The second view is that fundamental analysis which encourages the study of macro-economic factors and looking at financial conditions. And, the results of the industry to discover the extent of correlation that may exist with the changes in prices (Ritanjali Majhi & Panda, 2007). The third view is that most ANN-based models use historical and present data to predict future prices. ANNs have increasing gained their popularity due to their inherent capabilities to approximate any nonlinear function to a high degree of accuracy (Han, 2006).'

The first school is wrong because it suggests that significant theories contain random walk hypothesis; which states that market prices move like a random walk and thus unpredictable. (En.wikipedia.org, 2020). A random walk is a type of time series, where values tend to persist over time, and the differences between periods are white noise. But it is easy to see why the first school was defeated as there are always arbitrage opportunities; meaning when investors manage to buy and sell commodities and still make a safe profit while price adjusts. If such opportunities exist within a market, investors are bound to take advantage which would eventually lead to matching prices with the expected one; as a result, prices adjust accordingly. Hence, the expectation is that most markets to be more or less efficient.

The other school, as discussed in 2.1, can be easily manipulated by the leading information source provider. Though through various regulations and governing authorities that control information to ensure accuracy; 'fake' information on financial institutions is difficult to go public. However, fake news is also advancing with technology by using AI and machine learning. The extent to which fundamental analysis used to obtain correlations in the price changes has to link up with technical analysis for greater predictive accuracy. Technical analysis would take advantage of more advance tools for getting relationships within prices.

The last school, as discussed in 2.4, ANN is seen as the new framework that is applicable in a wide range of predicting issues and yield a higher degree of accuracy. But as more research on ANNs surfaces; new problems are discovered like outputting mixed results when modelling linear aspects. But solutions are being developed to counter some issues like the overfitting effect by using cross-validation based approach; dividing available data into three parts: training, validation, and test sets. The training and validation sets are used for ANN model building while the test set for genuine out of sample evaluation. ANN is an exciting topic to cover and would benefit the field of forecasting financial data with more research done.

2.7 Other Models

Other models which hardly have any recent papers or attention on are ARCH and GARCH. Jason Brownle; PhD in Artificial Intelligence stated, 'A change in the variance or volatility over time can cause problems when modelling time series with classical methods like ARIMA.' This statement stands out because the papers referenced in the project does not discuss the potential use of ARCH or GARCH models to predict values better. Authors did not establish an argument for using ARCH; which focuses on predicting turbulence in the data, regardless of an increase or decrease in the values. Neither the use of GARCH; which is a generalized version that includes both past variables and past errors; allowing it to support changes in the time-dependent volatility. (Brownlee, 2018). Financial institutions use GARCH to estimate the volatility of returns.

PennState: Eberly College of Science identify the ARCH's purpose to be advantageous when determining series with periods of increased or decreased variance, which can be a property of residuals after fitting the data on an ARIMA model. (PennState: Statistics Online Courses, 2020). Moreover, advocates for ANN point out that ARIMA models in time series forecasting cannot model stochastic non-constant volatility. But fail to cite Robert Engle's work which solved the issue by creating ARCH. The model can model volatility clustering, stochastic properties of volatility, mean reversion, and many more features of financial instability. (Engle, 1982). Recent papers were scarce comparing the use of ANNs to ARCH and GARCH models.

3 Approach

The following chapter explores and expands on the tools, steps and detailed approach of the project. It will explain how data handling throughout the study occurred to arrive at the various model solutions.

3.1 Data Pre-processing

We obtained data from Yahoo Finance on Dax30 (Dax) and S&P500 (spx); the initial plan was to get additional data on FTSE100 and NIKKEI225. To have different models for different regional markets. However, there were no FTSE values past 2012, and many Nikkei values were missing. It was critical to use data from the same data source; it was possible to find FTSE values on uk.investing.com however, formatting and matching the dates column with data from Yahoo provided challenging. Thus, the project observed the two datasets between the period of 03/01/2000 to 31/01/2020. The goal for this range was to train data to handle market crash values when developing models. The data was processed in Excel, using conditional formatting on the dates column to determine and match records where both prices are available. Dates with no values were deleted and saved in a CSV file. Somehow there were thirty-six missing values in the DAX column, discovered when examining the data in Jupyter Notebook. Thus, suggesting the method used in Excel was not efficient enough.

3.2 Time Series Development

Furthermore, we move on to creating a time series object by transforming a data frame into a time series and turning the date column into a date type. The date values are currently a text representation. Changing the string values imported into date type values would help refer to specific values by the dates recorded. Each value should correspond to a period. The date type would be the new index, and each date would be a separate period.

Setting the desired frequency; in this case, it will be daily, since our data is daily closing prices. Setting the frequency would most likely generate new periods with missing values associated with the period, due to these dates not included in the original set. However, adding more parameter ‘business days’ should remove some missing values. Python will expect missing values when dates fall between weekends. The missing values will be front filling (assigns the value of the previous period).

To conduct successful machine learning, we split the data into two sets: a training set and a testing set. The goal is to have the option of feeding new information into the model and comparing its predictions to actual real market values. The closer the forecasts are, the better the model. The testing and training sets would have to be uninterrupted sequences of values. Therefore, training would be from the beginning up to some cut-off point (80%), and testing would be from the cut-off point until the end (20%). Both sets are to be data frame objects and saved. But we have to make sure there are no overlapping values, by comparing the last elements of the training and the first elements of the testing set.

After the previous steps, we create a white noise time series. White noise is a sequence of random data, where every value has a period associated with it—generation of the white noise done from S&P values. With the mean normally distributed around the mean of S&P. The reason why we create white noise time series is to compare with the S&P; a type of time series where data does not follow a pattern. We would be able to see what kind of time series that is unpredictable and has no patterns. Furthermore, we would check for stationary in both using the Dickey-Fuller Test (DF Test). Stationarity means taking consecutive samples of data with the same size should have identical co-variances regardless of the starting point.

The results should be S&P values are non-stationary and white noise are stationary because there is no clear relationship between past and present values. Later, checking for decomposition seasonality in the data, using both additive and multiplicative. Decomposition splits into three effects: trend (pattern), seasonal (cyclical effects), and residual (error of prediction or the difference between the actual data and model fitted). Additive assumes that for any period, the observed value is the sum of the trend, seasonal and residual for that period. Multiplicative assumes the original series is a product of the trend, seasonal and residual values.

Next, computing and comparing the autocorrelation of both S&P and white noise. Autocorrelation simply meaning the correlation between a sequence and itself. In this case, it is a correlation between the values of the time series and a lagged version of itself. Autocorrelation Function (ACF) captures direct and indirect ways in which the lagged series affects the original one. Thus, to determine only the direct relationship between the time series and its lagged version, we need to compute the Partial Autocorrelation Function (PACF). The expectation for the white noise in both tests should be no autocorrelation and values for all periods to be insignificant.

3.3 Modelling and Forecasting

Finally, the project shall look at various simple models like the Autoregressive (AR) and Moving Average (MA) Models and building up the model complexity to Autoregressive Moving Average (ARMA) Model. AR relies on past period values and prior periods, only to predict current period values. We would examine the ACF and PACF; to determine the appropriate number of lags needed to incorporate into the model. We would fit a simple model first, then decide whether a more complex model makes better predictions.

However, keeping in mind that we can overfit the model due to an increase in complexity. The goal is to develop AR and MA models for both Dax and S&P values and compare how good they are. However, some point the prices would have to change into returns. Returns are the percentage change between the values for two consecutive periods. Again, examine ACF and PACF for returns and fit the AR model. Later, we would normalize the yields and examine the residuals (errors) of the AR model. To adjust for unexpected shocks from preceding periods, we would use the MA model and carry out the same process of the AR model. ARMA is the marriage between the two models of AR and MA. It solves the issues of both models.

For the other, more advance models like ARIMA, ARCH and GARCH; a similar approach will determine the most appropriate model to use for prediction. By this stage, we will have identified the ARIMAX, GARCH, SARIMAX and Auto ARIMA as the best models to move forward. The files used were then grouped into one file directory and arranged in the order: data processing, ARIMAX & SARIMAX, GARCH, Auto ARIMA and predictions (five files). And, finish off the project by forecasting our values and comparing the results. Other market indexes like FTSE and Nikkei will be exogenous variables in the auto ARIMA and prediction stage.

4 Data Processing

Taking advantage of the imported phyton packages like pandas, NumPy, and matplot to carry out the processing. The data was imported and examined on Jupyter Notebook. As mention before, missing Dax data was discovered, and the size of the dataset was 4990 rows by three columns. Both spx and dax values were plotted; this helped visualise how the data behaves and identify any unusual behaviours. Observation is that both graphs have a similar trend; thus, they look identical. The data contains the two market bubbles of the dotcom crash of 2000 and the 2008 stock market crash. Price fluctuations observed by periods of booming growth followed by sharp falls. X-axis: these are the indices associated with each value (represent the periods the values come from); Period 0 will express the first date of the dataset, which is 03/01/2000 because the index for each period is numeric values rather than dates .Y-axis: Index market values (prices).

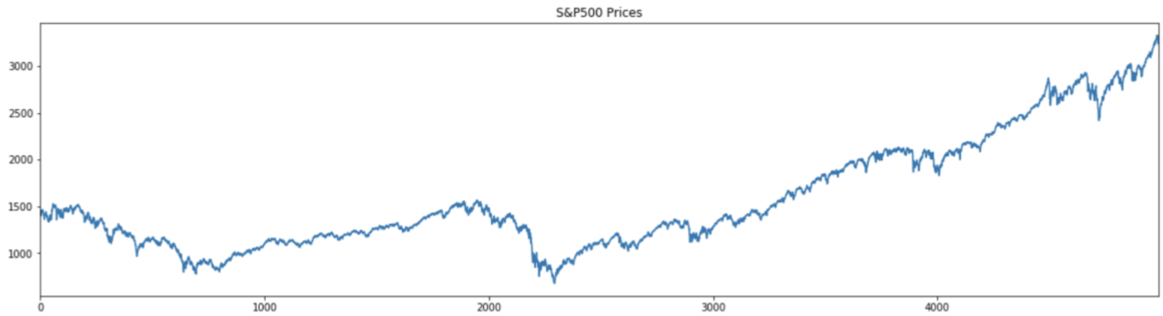


Figure 2 – Plotted graph of S&P500 values

Figure 3 – Plotted graph of Dax30 values

In figure four, the S&P is in blue, and Dax is in orange. Dax values are much more significant than S&P values. The difference between them; S&P looks flatter than Dax, which would suggest the S&P is much more stable. However, it is misleading; the graph of the S&P looks more compact because of the vastly different magnitudes between both indexes.

Figure 4 – Plotted graph of both S&P500 & Dax30 values

In figure 5 and 6; we look at the distribution of observations on raw observations. Comparing the results of the histogram and the density plot, we can see to the extent that both results behave similarly. S&P500 has a higher density than DAX30; S&P is more skewed than Dax.

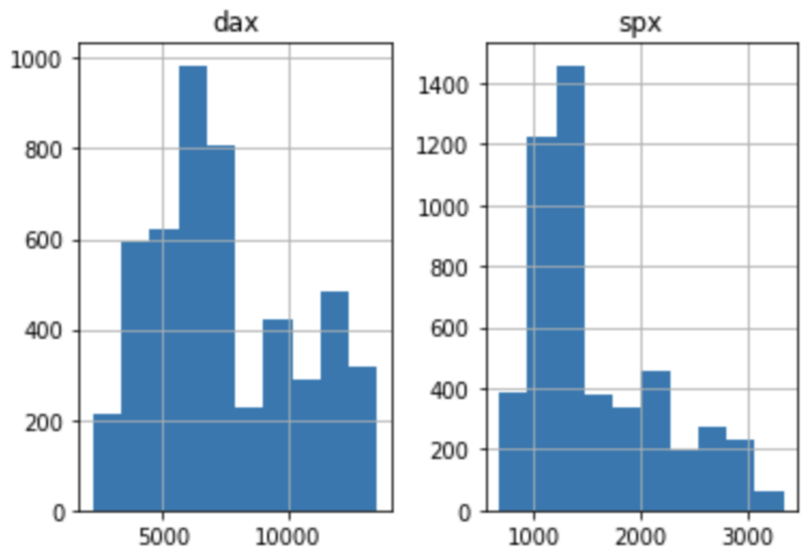


Figure 5 – S&P500 & DAX30 Histogram

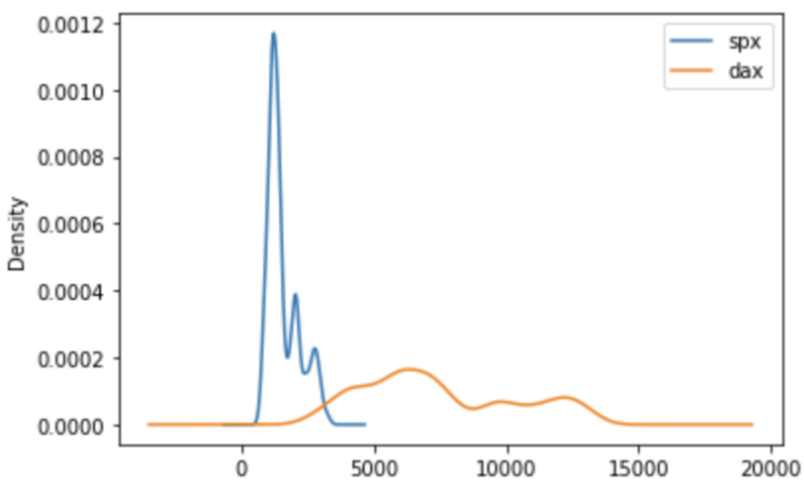


Figure 6 – S&P500 & DAX30 Density Plot

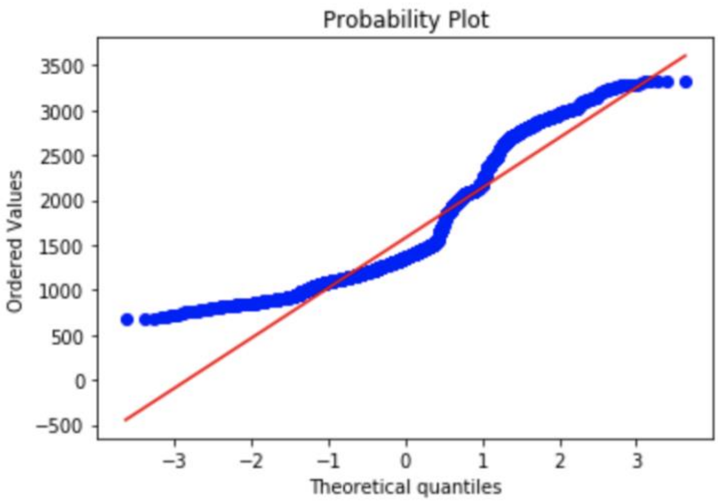


Figure 7– The QQ-Plot of S&P500

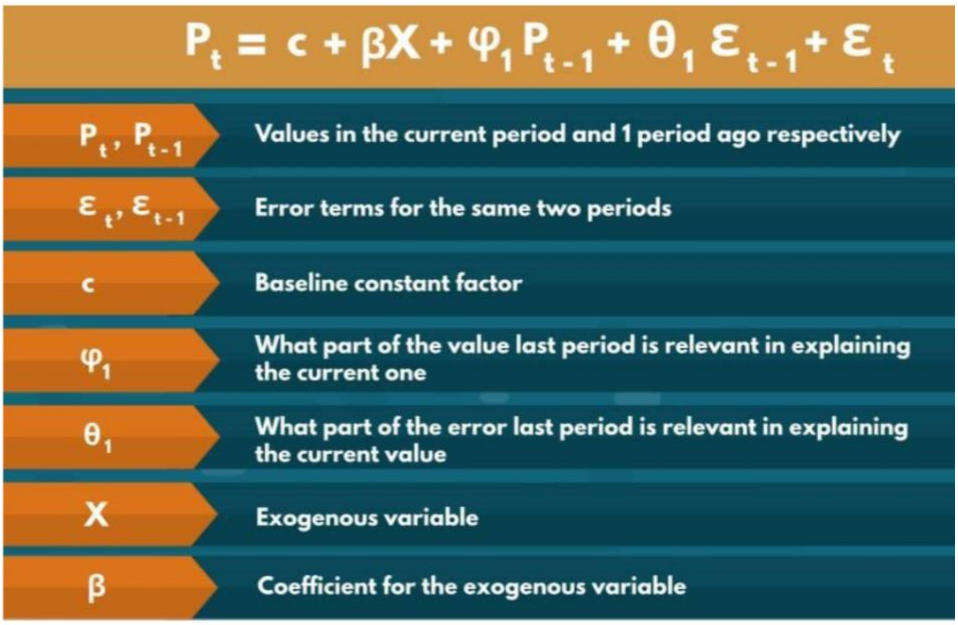
The Quantile-Quantile Plot; takes all the values a variable can take and arranges them in ascending order. X-axis: represents the theoretical quantiles of the data set; meaning how many standard deviations away from the mean the values are and the Y-axis shows the prices. The red diagonal line represents what the data points should follow if normally distributed. As observed there more values between 500 and 1000, thus suggesting the data is not normally distributed, but is expected from time-series data. The idea was to have data containing at least two periods of crashes. Later, I realised that the models would benefit more from having more data containing more crashes in order to identify the beginning and endings of these periods accurately.

5 Modelling & Models

This section we will look at the models which have a chance to produce significance. I looked at basic models like AR and MA but found that the three models below were the ones that stood out to be the best. As the results will show, SARIMAX was the best model because even the auto ARIMA model picked it. As stated earlier GARCH will determine the volatility in the index.

5.1 ARIMAX & SARIMAX

ARIMAX takes into account past prices, past residuals and outside factors which we can add another market value like Dax or even other market indexes. Introducing another index helps determine stability. ARIMAX is a multivariate model which surpasses ARIMA. Using SARIMAX helps identify when seasonality occurs when specific patterns are not consistent but appear periodically. It supports univariate time series data with a seasonal component. It uses differencing at a lag equal to the number of seasons to remove additive seasonal effects. Model selection is by higher loglikelihood and lower Akaike information criterion (AIC). MAX models can be great when analysing data but poor when forecasting.



(Time Series Analysis in Python, 2020)

Figure 8– ARIMAX Formula

ARIMAX		
	Log Likelihood	AIC
Model (1,0,1)	-16212.504	32435.007
Model (1,1,1)	-17248.042	34506.085

Table 1 – ARIMAX Model Selection

SARIMAX		
	Log Likelihood	AIC
Model(3,0,2,5)(3,0,4)	-16216.25	32460.501
Model(2,0,1,5)(1,0,1)	-16216.081	32446.163
Model(1,0,1,5)(2,0,2)	-16215.235	32446.47

Table 2 – SARIMAX Model Selection

5.2 Auto ARIMA

Auto ARIMA is a type of model that automatically picks the best model which will yield the best results. Some benefits are it saves time when trying to choose a model, removes ambiguity, and reduces the risk of human error; which is very likely with lack of experience. However, some issues that arise are the over-reliance on one criterion and never get to see how well other models perform. The model selected SARIMAX (2, 0, 2)(1, 0, 1, 5). However, we saw from the previous model of SARIMAX based on the rule of higher loglikelihood and lower AIC. The best SARIMAX was (3,0,4)(3,0,2,5); as seen in table 2. Examining the p-values generated by the auto ARIMA: SARIMAX (2, 0, 2)(1, 0, 1, 5) in table 3; the algorithm picked AR (2) and MA (2) which is not significant at the 5% (0.05) level.

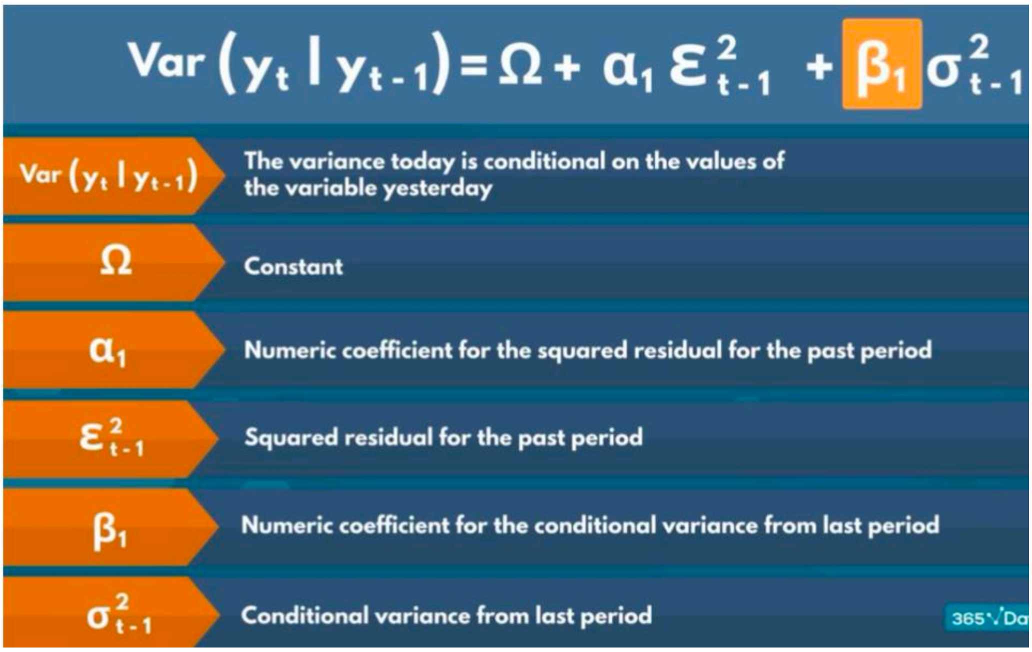
Moreover, the seasonality elements in the p-value coefficients are also not substantial. We see one of the issues mentioned earlier; how was this model selected. When a better was available with more iterations of AR and MA with a higher seasonality order.

	coef	std err	z	P>|z|	[0.025	0.975]
intercept	-0.0180	0.017	-1.083	0.279	-0.050	0.015
drift	1.648e-06	2e-06	0.823	0.411	-2.28e-06	5.57e-06
ret_ftse	0.1656	0.011	14.420	0.000	0.143	0.188
ret_dax	0.3880	0.009	44.166	0.000	0.371	0.405
ret_nikkei	-0.0307	0.006	-4.960	0.000	-0.043	-0.019
ar.L1	0.7439	0.307	2.422	0.015	0.142	1.346
ar.L2	-0.0149	0.044	-0.342	0.732	-0.100	0.070
ma.L1	-1.0683	0.307	-3.479	0.001	-1.670	-0.467
ma.L2	0.2375	0.136	1.751	0.080	-0.028	0.503
ar.S.L5	-0.0048	0.398	-0.012	0.990	-0.786	0.776
ma.S.L5	-0.0181	0.398	-0.046	0.964	-0.798	0.762
sigma2	0.7947	0.008	97.570	0.000	0.779	0.811

Table 3 – Auto ARIMA

5.3 GARCH

Heteroscedasticity is when the variability of a variable is not equal across the range of the second variable values that it uses for predictions—also expressed as a non-constant size of expected residuals of a time series, where the residual depends on the size of the independent variable. The model identifies stochastic processes for the residual and predicts the average size of residuals when fitted to empirical data. (Engle, 2001). Also, described as the squared ARMA model for the error terms of the mean equation. Thus, adding a single past variance gives more predictive power than squared residuals; including previous values as a form of baseline provides much higher accuracy. Hence the model becomes the best for the time being to measure volatility. Benefits of using ARCH models are that it takes care of clustered errors and non-linearities, simple and easy to handle.



(Time Series Analysis in Python, 2020)

Figure 10– Garch Formula

The GARCH model had two potential models (2,1) and (3,1), using the rule of higher LLR and lower AIC. To best determine which model is best to use, we will use both and compare them to see which has the best volatility prediction.

GARCH		
	Log Likelihood	AIC
Garch(1,1)	-5926.6	11861.2
Garch(1,2)	-5926.6	11863.2
Garch(1,3)	-5926.6	11865.2
Garch(2,1)	-5917.2	11844.4
Garch(3,1)	-5917.2	11846.4

Table 4 – GARCH Model Selection

However, observing the p-values of GARCH (2,1) and GARCH (3,1) some values are not significant at the 5% significant level. So, picking a model using the rule of the non-significant p-value for the highest lag coefficients. Examining the tables below: Table 5 has non-significant p-value in alpha and beta, and it has a lower AIC compared to the rest in table 4 but does not have a higher log-likelihood compared to GARCH (2,1) in table 4. Table 7 fails the significance test in beta 2 and 3. P = 1; means full multicollinearity (aka perfect collinearity).

The variance from the last period already captures the meaning, all the explanatory power of the conditional variance two periods ago. Table 6 looked promising. However, proven by math that no higher-order GARCH models outperform the GARCH (1,1) when it comes to the variance of market returns. Due to the recursive nature in which the past conditional variance is computed, including one not only makes it redundant to include past squared residuals since it already captures the effect. All the effects of the conditional variance two days ago will be contained in the conditional variance yesterday, hence no need to include more than 1 GARCH component. The safest model would be the GARCH (1,1).

Volatility Model

	coef	std err	t	P>|t|	95.0% Conf. Int.
omega	0.0169	4.479e-03	3.781	1.564e-04	[8.155e-03,2.571e-02]
alpha[1]	0.0884	1.087e-02	8.134	4.145e-16	[6.713e-02, 0.110]
beta[1]	0.8990	1.158e-02	77.623	0.000	[0.876, 0.922]

Table 5 – GARCH (1,1)

Volatility Model

	coef	std err	t	P>|t|	95.0% Conf. Int.
omega	0.0229	6.143e-03	3.725	1.951e-04	[1.084e-02,3.492e-02]
alpha[1]	0.0300	1.975e-02	1.518	0.129	[-8.727e-03,6.868e-02]
alpha[2]	0.0807	2.583e-02	3.124	1.784e-03	[3.007e-02, 0.131]
beta[1]	0.8725	1.699e-02	51.354	0.000	[0.839, 0.906]

Table 6 – GARCH (2,1)

Volatility Model

	coef	std err	t	P>|t|	95.0% Conf. Int.
omega	0.0169	4.418e-03	3.833	1.267e-04	[8.274e-03,2.559e-02]
alpha[1]	0.0884	1.514e-02	5.841	5.187e-09	[5.876e-02, 0.118]
beta[1]	0.8990	0.190	4.733	2.217e-06	[0.527, 1.271]
beta[2]	3.3460e-11	0.312	1.073e-10	1.000	[-0.611, 0.611]
beta[3]	0.0000	0.226	0.000	1.000	[-0.443, 0.443]

Table 7 – GARCH (1,3)

6 Evaluation and Forecasting

This section we will look at the models which have performed well in predicting our test data and forecasted the volatility. I compared previous basic models again; found like in the last section, the three models were the best. The 'manual' SARIMAX was the best, not to be confused with the SARIMAX picked by auto ARIMA.

6.1 Predictions

All the models that perform well rely on exogenous variables; meaning the MAX models rely on outside data much more than past values (AR) and past errors (MA). However, exogenous variables are hardly available in the long run. For all model figures from 11 to 13; the testing data is blue, and the prediction or model data is red. The ARIMAX model was the worst performer in the set even though other indexes used the exogenous variables. Figure 11; shows how the model performs compared with the test data. Suggesting, ARIMAX greatly underpredicts prices during sharp market gains and overpredicts during severe market declines. However, overpredicting only occurred on four occasions, and one out of four was a sharp decline. The model struggles when determining positive price actions, especially during sharp gains.

Moreover, the model either struggles or accurately predicts in advance stable price action movements—focusing attention between December 2016 and January 2017. We see the red line (model) predict the next small gain in the market as it declines down (where red and blue meet signalling a coming price decline). However, the model may have missed the previous price action and is just using past data to adjust itself; thus, the observation could be an outlier.

ARIMAX

Figure 11– ARIMAX

Figure 12; shows the SARIMAX which performs very similarly to the ARIMAX model. However, it has a more precise prediction during sharp movements to an extent but at a minimalistic level reducing the gap between the prediction and actual results. In figure 14; we see that the ARIMAX (in green) is hard see. However, due to the lack of apparent change, it raises the question; why is SARIMAX even used, especially by the Auto ARIMA model.

SARIMAX

Figure 12– SARIMAX

Figure 13; shows the Auto ARIMAX which compared with the previous models is very out of sync with the test data. The question remains why was SARIMAX used for Auto ARIMA when it hardly is in sync during stable price movements. The model overshoots small price changes but undershoots more massive price changes with some accurate predictions. This model lacks stability; hence an investor would be unlikely to use such for trading or investments.

Auto ARIMA

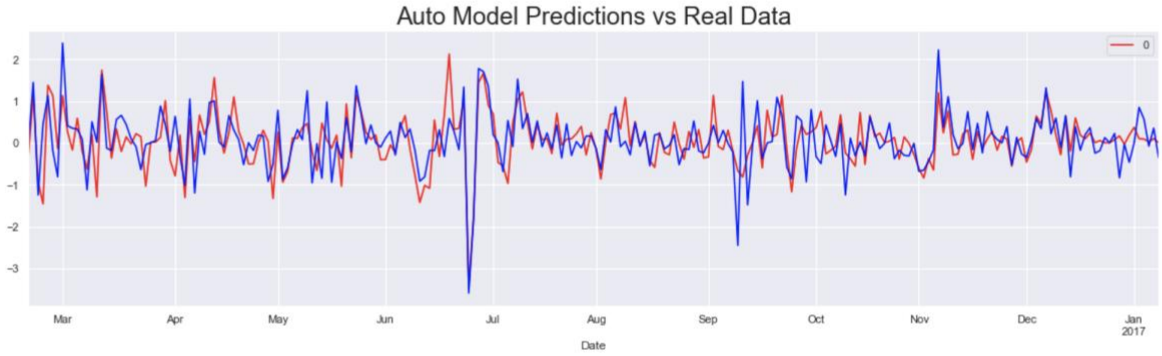


Figure 13– Auto ARIMAX

In figure 14; we get the overall performance of the two models. As stated earlier, SARIMAX does a better job than ARIMAX (observing the small green line at the highs).

All Models

Figure 14– Model Comparison

6.2 Volatility Forecasting

For Figure 15 to 17; Blue = actual and Red = Predictions. Figure 15 and 17; look very similar to the model behaves in the same way. The two models bring confusion as GARCH (1,1) is said to be mathematical better than any higher order of GARCH. If that is the case, why is GARCH (1,3) behaving like the simpler GARCH (1,1) model? Concerning the GARCH (2,1), the GARCH (1,1) underperforms greatly. The gap between the two graphs may seem minimal; however, for example: If an investor wants to buy into the market, they would need a model with the least gap or margin of error. If they use GARCH (1,1) during 2019, they would have lost more money as the model failed to predict the rise of volatility (approximately failed to predict level 4.25 to 5). Now, GARCH (2,1) during the same period only failed to capture approximately 4.78 to 5; which is a much better result. (Using the scale: y-axis 0 to 1 is about 1.4cm).

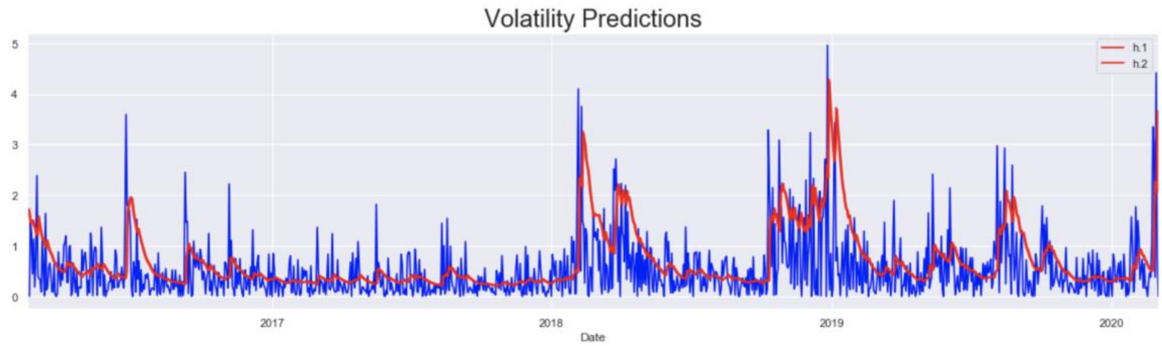


Figure 15– Garch (1,1)

In GARCH (2,1) only fails to predict one sharp rise in volatility but accurately predicts an even more definite increase in volatility; which is good as investors would be able to use the model for higher volatility. However, during market stability the model struggles predicting periods of low volatility. The model falls behind the actual market; as it tries to adjust from past movements; for example: from late 2018 to early 2019. The model attempted to predict the high volatility, but the market volatility reduced slightly but later went up. However, it maintained the higher volatility rate and 'waited' for the market volatility; which the market reached. Overall, based on the observation, the SARIMAX is the best.

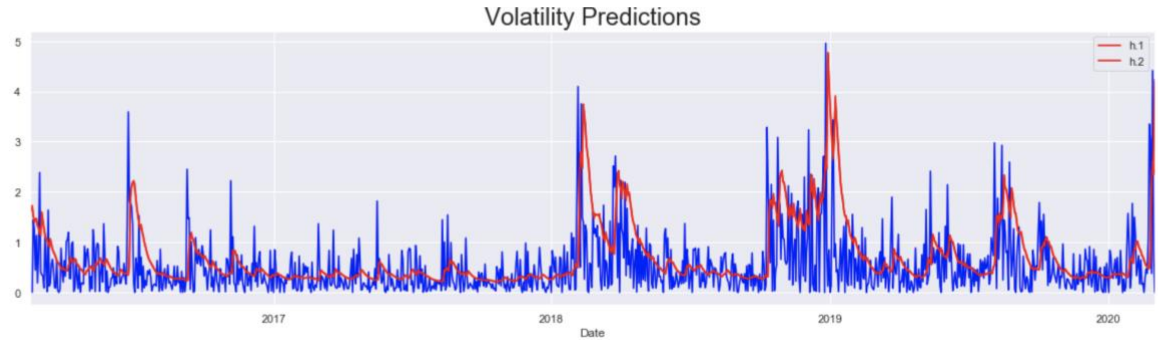


Figure 16– Garch (2,1)

Figure 17 was just here to show the similarities with GARCH (1,1). However, despite the mathematical prove, GARCH (2,1) is the best forecaster of volatility.

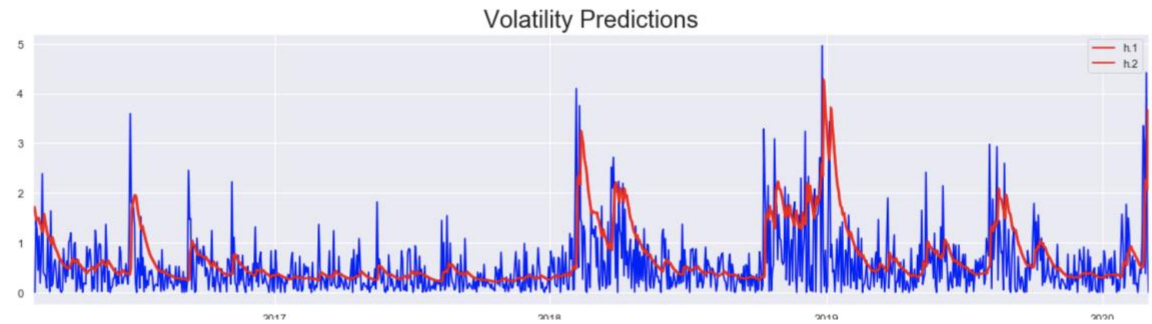


Figure 17– Garch (1,3)

7 Conclusions

The project investigated how to predict the next American stock market crash through the project investigated how to predict the next American stock market crash by looking at S&P500 closing prices and determining its volatility. The reason financial modelling and volatility modelling exists is due to uncertainties created in the market; thus, discovering these difficulties is the main reason for all modelling. Financial time series makes volatility essential predictable. However, due to the stochastic nature of volatility, there is no amount of past data that could enter into models that will fully capture the current or future behaviour of volatility. Moreover, predicting market prices are also tricky as it requires more data; specifically (as mentioned in the introduction) housing prices, industrial production output, GDP, unemployment rate and CPI for measuring inflation. Realistically speaking these measures are used in the real world to gauge the health of the markets. As seen, the models can approximate market behaviour of prices and volatility during the analysed period of training and forecasting time of testing sets.

The models behave differently when predicting different types of data like when we have seen the difference in returns and prices as the model complexities raised the models gained higher levels of accuracy and lower margins of errors. However, this complexity was met with a false sense of security. Regardless of the methodology, we have recently seen the failure of not knowing how financial markets will move during the trading week of 9th March to 13th March 2020, where global markets lost approximately $6 trillion in just six days. Thus, market price prediction and most especially volatility forecasting is a critical tool for investors seeking suitable investments and implementing risk management strategies.

Objective one was met by analysing essential literature background surrounding the topic and project; which helped maximised accuracy and minimised bias. Objective two was met in section four where data processing occurred, patterns established and interpreted the data for predicting. Objective three was achieved in part five, where models were developed and implemented to fit the data set to find a suitable model. Objective four was met in section six have we compared the outputs generated by the models, determine volatility and the model's margin of error. Objective five was achieved by using GitHub and using a Gantt chart for planning the project. Objective six was met throughout the project in every section. Objective seven was met through ethical approval which will be available under appendixes.

The aim was to identify the event represented by the sequence of observations; the event being identified was the price action of S&P500 and its volatility. We determined the appropriated models, which were ARIMAX and SARIMAX for S&P prices, respectively. And, GARCH for determining volatility for the data. Raising of the complexity level of the models showed a positive change in lowering the margin of error; which improved the model results, especially for the GARCH model. Thus, meeting the overall aim.

In summary, the question of predicting the US stock market crash through market index evaluation was successful to some extent, but there were some weaknesses in the approach used. Some of those weaknesses came about as the project progressed and became difficult at specific levels to turn back or correct. Examples: the data set was too short, and I only discovered how to import and update the data from Yahoo Finance which helped with the difficulty of obtaining past data for other market indexes. Likely human error occurred during the model selection. The results of the best GARCH model to use is different from want had been explained in tutorials. Therefore, this solution was successful in its goal but created a puzzling situation that would need to be solved to determine why the GARCH (1,1) model was not the best even though mathematically it has been proven.

7.1 Future Work

Firstly, the next course of action would be to investigate the mathematical prove of the GARCH model and determine why the GARCH (2,1) yields more accuracy than GARCH (1,1). Another GARCH based study would be how to determine the price of options ('the insurance premium') when the market follows the results of a GARCH model. This topic is explicitly of great interest because of investors and traders like using options due to its flexible nature; as they can generate alternative strategies. It is also cost-efficient and reduces risk. Combining the GARCH and options trading; there is potential to identify volatility with GARCH and use options or hedges to guide investors and traders.

The use of ANNs for forecasting asset prices is an ever-growing field of study with researchers and investors try to outperform the markets to gain more profits. Neural networks can train to learn different types of data; like stated before I believe the best system would be one that takes various economic indicators as the second input and what is being predicted as the primary input. The best part is that the neural networks can prevent overtraining and memorizing the data, thus, reducing the likelihood of overfitting. Also, it becomes easy to remove redundant input nodes which speed up training and potentially identify negative feedback loops in the market; reducing one of ANNs disadvantages.

Furthermore, neural networks support the belief that EMH does not work and markets are inefficient. ANNs also outperforms statistical, and regression models in forecasting, leading to the next study of ANNs in conjunction with chaotic systems as markets are dynamic, chaotic systems; which ANNs can discover patterns in both nonlinear and chaotic systems. However, ANNs are not perfect predictors due to the factors that cause price actions in the stock market are dynamic and too involved to understand in the long term viewpoint of an investor.

References

Brownlee, J. (2018). How to Develop LSTM Models for Time Series Forecasting. [online] Machine Learning Mastery. Available at: https://machinelearningmastery.com/how-to-develop-lstm-models-for-time-series-forecasting/ [Accessed 18 Feb. 2020].

Brownlee, J. (2018). How to Model Volatility with ARCH and GARCH for Time Series Forecasting in Python. [online] Machine Learning Mastery. Available at: https://machinelearningmastery.com/develop-arch-and-garch-models-for-time-series-forecasting-in-python/ [Accessed 19 Feb. 2020].

CFA Institute Research Foundation. "TECHNICAL ANALYSIS: Modern Perspectives," Page 6. Accessed Nov. 12, 2019.

Chatterjee, S. (2019). ARIMA/SARIMA vs LSTM with Ensemble learning Insights for Time Series Data. [online] Medium. Available at: https://towardsdatascience.com/arima-sarima-vs-lstm-with-ensemble-learning-insights-for-time-series-data-509a5d87f20a [Accessed 18 Feb. 2020].

Engle, R. (1982). Autoregressive Conditional Heteroscedasticity with Estimates of the Variance of United Kingdom Inflation. Econometrica, [online] 50(4), p.987. Available at: https://www.jstor.org/stable/1912773?seq=1&cid=pdf-reference#references_tab_contents [Accessed 19 Feb. 2020].

Engle, R. (2001), "Garch 101: The Use of ARCH/GARCH Models in Applied Econometrics", Journal of Economic Perspectives, vol.15, no.4, pp.157-168.

Engle, R., 2002. New frontiers for arch models. Journal of Applied Econometrics, 17(5), pp.425-446.

En.wikipedia.org. (2020). Dow theory. [online] Available at: https://en.wikipedia.org/wiki/Dow_theory [Accessed 8 Mar. 2020].

En.wikipedia.org. (2020). Random walk hypothesis. [online] Available at: https://en.wikipedia.org/wiki/Random_walk_hypothesis [Accessed 18 Feb. 2020].

En.wikipedia.org. (2020). Stock market crash. [online] Available at: https://en.wikipedia.org/wiki/Stock_market_crash [Accessed 8 Mar. 2020].

Ganti, A. and Segal, T. (2019). Fundamental Analysis. [online] Investopedia. Available at: https://www.investopedia.com/terms/f/fundamentalanalysis.asp [Accessed 16 Feb. 2020].

Han, L. Q. (2006). Tutorial of artificial neural network. Beijing Posts & Telecommunication Press (in Chinese).

Kenton, W. (2019). Understanding Time Series. [online] Investopedia. Available at: https://www.investopedia.com/terms/t/timeseries.asp [Accessed 17 Feb. 2020].

Khashei, M. and Bijari, M. (2011). A novel hybridization of artificial neural networks and ARIMA models for time series forecasting. Applied Soft Computing, [online] 11(2), pp.2664-2675. Available at: https://www.sciencedirect.com/science/article/pii/S1568494610002759#fig0005.

Kirkpatrick II, C. and Dahlquist, J. (2011). Technical analysis: the complete resource for financial market technicians. Choice Reviews Online, [online] 48(09), pp.48-5201-48-5201. Available at: http://1.droppdf.com/files/wWSus/technical-analysis-the-complete-resource-for-financial-market-technicians-2011.pdf.

Libres.uncg.edu. (n.d.). [online] Available at: https://libres.uncg.edu/ir/uncw/f/zhai2005-2.pdf [Accessed 17 Feb. 2020].

Losada, L. (n.d.). Time Series Analysis with Auto.Arima in R. [online] Medium. Available at: https://towardsdatascience.com/time-series-analysis-with-auto-arima-in-r-2b220b20e8ab [Accessed 17 Feb. 2020].

Mrao.cam.ac.uk. (2011). [online] Available at: http://www.mrao.cam.ac.uk/~mph/Technical_Analysis.pdf [Accessed 16 Feb. 2020].

Murphy, J. (1999) Technical Analysis of the Financial Markets: A Comprehensive Guide to Trading Methods and Applications, Pages 23–24, 33, & 87–88. New York Institute of Finance. 1999.

PennState: Statistics Online Courses. (2020). 11.1 ARCH/GARCH Models | STAT 510. [online] Available at: https://online.stat.psu.edu/stat510/lesson/11/11.1 [Accessed 19 Feb. 2020].

Peters, E. (1996). Chaos & order in capital markets: A new view of cycles, prices, market volatility. John Wiley & Sons Inc.

Qian, X. and Gao, S. (2017). Financial Series Prediction: Comparison Between Precision of Time Series Models and Machine Learning Methods. [online] arXiv.org. Available at: https://arxiv.org/abs/1706.00948 [Accessed 17 Oct. 2019].

Taylor, S. (1986). Modeling financial time series. John Wiley & Sons, Inc.

Ritanjali Majhi, & Panda, G., (2007). Stock market prediction of S&P 500 and DJIA using bacterial foraging optimization technique. In 2007 IEEE congress on evolutionary computation (CEC 2007) (pp. 2569–2579).

Udemy. 2020. Time Series Analysis in Python. [online] Available at: <https://www.udemy.com/course/time-series-analysis-in-python/> [Accessed 12 March 2020].

Zhang, G., Patuwo, B.E, Hu, M.Y., 1998. Forecasting with artificial neural networks: The state of the art. International Journal of Forecasting 14, 35±62.