Comparing time series and ANN models’ forecasting effectiveness on stocks and cryptocurrency

**Name:** Dave Opoku Asiedu

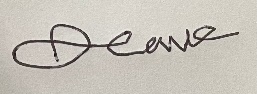
**MMU ID Number:** 20031164

**Date of Submission:** [] June 2023

**Name of Degree:** BSc Economics and Finance

The dissertation is submitted in part fulfilment of the requirements of the BSc Economics and Finance degree in the Department of Economics, Policy and International Business at Manchester Metropolitan University.

# **Declaration**

******This dissertation is an original and authentic piece of work produced in fulfilment of my degree regulations. I have fully acknowledged and referenced all secondary sources. The dissertation has not been submitted in whole or part of an assessment in another unit at this or any other university.

Signature: Date: /06/2023

# **Abstract**

Markets are plagued with uncertainty; investors consistently seek methods to counteract the phenomenon. Forecasting is used in the stock and cryptocurrency markets as the solution to the problem of this uncertainty. This study compares time series, and artificial neural network (ANN) models returns and volatility forecasting performance, utilising selected stocks and cryptocurrency historical close price data between 2018-2023, namely FTSE, S&P 500, Bitcoin and Ethereum. The time series models used within the study are the ARIMA model for forecasting returns; GARCH and EGARCH were also used for volatility forecasting. The long short-term memory (LSTM) model was utilised as the ANN model within the study to forecast returns.

Additionally, a comparison point was formed between linear GARCH and non-linear EGARCH. Due to improved forecasting accuracy, log returns were used as the variable to conduct forecasts for the ARIMA, GARCH and EGARCH models. A similar method was utilised for the LSTM model by scaling data sets.

The results of the models indicate that the ARIMA model vastly outperformed the LSTM model used in the study. Regardless of asset class, the ARIMA model consistently provided returns forecasts with higher accuracy than the LSTM model. Additionally, the non-linear EGARCH model provided slightly more significant forecasts than linear GARCH. Moreover, stocks used within the study, FTSE and S&P 500, provided more significant volatility and returns forecasts than the selected cryptocurrency in all models used within the study. To conclude, the ARIMA model was found to be the better returns forecasting model, EGARCH was shown to provide the most significant volatility forecasts, and stocks provided the most significant forecasts.

# **Acknowledgements**

I would like to thank my dissertation supervisor Lotanna Emediegwu for all the support and advice given in the process of completing this study.

# **Glossary of Abbreviations**

ANN: Artificial Neural Network

EMH: Efficient Market Hypothesis

AI: Artificial Intelligence

NYSE: New York Stock Exchange

WFE: World Federation of Exchanges

RWT: Random Walk Theory

ADF: Augmented Dicky Fuller

IPS: Im, Pesaran and Shin

SUR: Seemingly Unrelated Regression

MADF: Multivariate Augmented Dicky Fuller

LM: Lagrange multiplier

AR: Autoregressive

MR: Moving Average

ARIMA: Autoregressive Integrated Moving Average

ARCH: Autoregressive Conditional Heteroscedasticity

GARCH: Generalized Autoregressive Conditional Heteroscedasticity

EGARCH: Exponential Generalised Autoregressive Conditional Heteroscedasticity

ACF: Autocorrelation Function

PACF: Partial Autocorrelation Function

PM: Performance Metric

RMSE: Root Mean Square Error

MAPE: Mean Absolute Percentage Error

MSE: Mean Square Error

GRU: Gated Recurrent Unit

DR: Deterministic Regressors

MDA: Multivariate Discriminant Analysis

APGARCH: Asymmetric Power Generalised Autoregressive Conditional Heteroscedasticity

DFFNN: Deep Feed Forward Neural Network

LSTM: Long Short-Term Memory

MLE: Maximum Likelihood Estimation

AIC: Akaike Information Criterion

**Table of Contents**

[**Declaration** i](#_Toc139051953)

[**Abstract** ii](#_Toc139051954)

[**Acknowledgements** iii](#_Toc139051955)

[**Glossary of Abbreviations** iv](#_Toc139051956)

[**List of Figures** viii](#_Toc139051957)

[**List of Tables** ix](#_Toc139051958)

[**Chapter 1: Introduction** 1](#_Toc139051959)

[**1.1** **Research Question** 2](#_Toc139051960)

[**1.2** **Research Aim** 2](#_Toc139051961)

[**1.3** **Research Objectives** 2](#_Toc139051962)

[**1.4** **Rationale** 2](#_Toc139051963)

[**1.5** **Structure** 3](#_Toc139051964)

[**Chapter 2: Literature Review** 4](#_Toc139051965)

[**2.1** **Efficient Market Hypothesis and its Implications on Forecasting** 4](#_Toc139051967)

[**2.2** **Time series Forecasting** 6](#_Toc139051968)

[**2.2.1** **Forecasting Stock Prices and Returns** 7](#_Toc139051969)

[**2.2.2** **Forecasting Stock Volatility** 7](#_Toc139051970)

[**2.2.2.1** **Symmetric and Asymmetric Forecasting Volatility Models** 8](#_Toc139051971)

[**2.3** **Artificial Neural Network (ANN) Forecasting** 9](#_Toc139051972)

[**2.3.1** **Forecasting Crypto Prices and Returns** 10](#_Toc139051973)

[**Chapter 3: Methodology** 12](#_Toc139051976)

[**3.1** **Data Collection** 12](#_Toc139051978)

[**3.1.1** **Variables** 12](#_Toc139051979)

[**3.1.2** **Assets** 13](#_Toc139051980)

[**3.2** **Data Limitations** 13](#_Toc139051981)

[**3.3** **Data Cleaning** 14](#_Toc139051982)

[**3.4** **Forecasting Methods** 14](#_Toc139051983)

[**3.4.1** **Time Series Forecasting Models** 14](#_Toc139051984)

[**3.4.1.1** **AutoRegressive Integrated Moving Average (ARIMA)** 14](#_Toc139051985)

[**3.4.1.2** **ARIMA Forecasting Process Flowchart** 15](#_Toc139051986)

[**3.4.1.3** **Generalised AutoRegressive Conditional Heteroskedasticity (GARCH)** 16](#_Toc139051987)

[**3.4.1.4** **Exponential GARCH** 16](#_Toc139051988)

[**3.4.1.5** **GARCH & EGARCH Forecasting Process Flowchart** 18](#_Toc139051989)

[**3.4.2** **Artificial Neural Network Forecasting Models** 19](#_Toc139051990)

[**3.4.2.1** **Lost Short-Term Memory (LSTM)** 19](#_Toc139051991)

[**3.4.2.2** **LSTM Forecasting Process Flowchart** 21](#_Toc139051992)

[**3.5** **Performance Metrics** 23](#_Toc139051993)

[**3.5.1** **Root Mean Square Error (RMSE)** 23](#_Toc139051994)

[**3.5.2** **Mean Absolute Percentage Error (MAPE)** 23](#_Toc139051995)

[**3.5.3** **Mean Absolute Error (MAE)** 24](#_Toc139051996)

[**Chapter 4: Data Analysis** 25](#_Toc139051997)

[**4.1** **Data Cleaning** 25](#_Toc139052002)

[**4.2** **Data Exploration** 25](#_Toc139052003)

[**4.3** **Returns Forecasting Results** 27](#_Toc139052004)

[**4.3.1** **ARIMA** 27](#_Toc139052005)

[**4.3.2** **LSTM** 30](#_Toc139052006)

[**4.3.3** **Model Comparison Discussion** 35](#_Toc139052007)

[**4.4** **Volatility Forecasting Results** 36](#_Toc139052008)

[**4.4.1** **GARCH** 36](#_Toc139052009)

[**4.4.2** **EGARCH** 39](#_Toc139052010)

[**4.4.3** **Model Comparison Discussion** 41](#_Toc139052011)

[**4.5** **Limitations and Future Recommendations** 42](#_Toc139052012)

[**Chapter 5: Conclusion** 43](#_Toc139052013)

[**References** 44](#_Toc139052014)

[**Appendices** 45](#_Toc139052015)

# **List of Figures**

[**Figure 1- ARIMA Forecasting Flowchart** 15](https://d.docs.live.net/15d95714c3d93845/Documents/Dave%20O%20Asiedu%20Dissertation%20Updated%20(002).docx#_Toc139052016)

[**Figure 2- GARCH/EGARCH Forecasting Flowchart** 18](https://d.docs.live.net/15d95714c3d93845/Documents/Dave%20O%20Asiedu%20Dissertation%20Updated%20(002).docx#_Toc139052017)

[**Figure 3- LSTM Forecasting Flowchart** 21](https://d.docs.live.net/15d95714c3d93845/Documents/Dave%20O%20Asiedu%20Dissertation%20Updated%20(002).docx#_Toc139052018)

[**Figure 4- Stocks & Cryptocurrency Closing Prices, 2018 - 2023** 25](#_Toc139052019)

[**Figure 5- Bitcoin ARIMA Forecast** 27](https://d.docs.live.net/15d95714c3d93845/Documents/Dave%20O%20Asiedu%20Dissertation%20Updated%20(002).docx#_Toc139052020)

[**Figure 6- Ethereum ARIMA Forecast** 27](https://d.docs.live.net/15d95714c3d93845/Documents/Dave%20O%20Asiedu%20Dissertation%20Updated%20(002).docx#_Toc139052021)

[**Figure 7- FTSE ARIMA Forecast** 28](https://d.docs.live.net/15d95714c3d93845/Documents/Dave%20O%20Asiedu%20Dissertation%20Updated%20(002).docx#_Toc139052022)

[**Figure 8- S&P 500 ARIMA Forecast** 28](https://d.docs.live.net/15d95714c3d93845/Documents/Dave%20O%20Asiedu%20Dissertation%20Updated%20(002).docx#_Toc139052023)

[**Figure 9- Bitcoin LSTM Testing, Validation, Training and Returns Forecast** 30](#_Toc139052024)

[**Figure 10- Ethereum LSTM Testing, Validation, Training and Returns Forecast** 31](#_Toc139052025)

[**Figure 11- FTSE LSTM Testing, Validation, Training and Returns Forecast** 32](#_Toc139052026)

[**Figure 12- S&P 500 LSTM Testing, Validation, Training and Returns Forecast** 33](#_Toc139052027)

[**Figure 13- Bitcoin GARCH Forecast** 36](https://d.docs.live.net/15d95714c3d93845/Documents/Dave%20O%20Asiedu%20Dissertation%20Updated%20(002).docx#_Toc139052028)

[**Figure 14- Ethereum GARCH Forecast** 36](https://d.docs.live.net/15d95714c3d93845/Documents/Dave%20O%20Asiedu%20Dissertation%20Updated%20(002).docx#_Toc139052029)

[**Figure 15- FTSE GARCH Forecast** 37](https://d.docs.live.net/15d95714c3d93845/Documents/Dave%20O%20Asiedu%20Dissertation%20Updated%20(002).docx#_Toc139052030)

[**Figure 16- S&P 500 GARCH Forecast** 37](https://d.docs.live.net/15d95714c3d93845/Documents/Dave%20O%20Asiedu%20Dissertation%20Updated%20(002).docx#_Toc139052031)

[**Figure 17- Bitcoin EGARCH Forecast** 39](https://d.docs.live.net/15d95714c3d93845/Documents/Dave%20O%20Asiedu%20Dissertation%20Updated%20(002).docx#_Toc139052032)

[**Figure 18- Ethereum EGARCH Forecast** 39](https://d.docs.live.net/15d95714c3d93845/Documents/Dave%20O%20Asiedu%20Dissertation%20Updated%20(002).docx#_Toc139052033)

[**Figure 19- FTSE EGARCH Forecast** 40](https://d.docs.live.net/15d95714c3d93845/Documents/Dave%20O%20Asiedu%20Dissertation%20Updated%20(002).docx#_Toc139052034)

[**Figure 20- S&P 500 EGARCH Forecast** 40](https://d.docs.live.net/15d95714c3d93845/Documents/Dave%20O%20Asiedu%20Dissertation%20Updated%20(002).docx#_Toc139052035)

[**Figure 21- Bitcoin ADF Test** 45](https://d.docs.live.net/15d95714c3d93845/Documents/Dave%20O%20Asiedu%20Dissertation%20Updated%20(002).docx#_Toc139052036)

[**Figure 22- Ethereum ADF Test** 45](https://d.docs.live.net/15d95714c3d93845/Documents/Dave%20O%20Asiedu%20Dissertation%20Updated%20(002).docx#_Toc139052037)

[**Figure 23- FTSE ADF Test** 45](https://d.docs.live.net/15d95714c3d93845/Documents/Dave%20O%20Asiedu%20Dissertation%20Updated%20(002).docx#_Toc139052038)

[**Figure 24- S&P 500 ADF Test** 45](https://d.docs.live.net/15d95714c3d93845/Documents/Dave%20O%20Asiedu%20Dissertation%20Updated%20(002).docx#_Toc139052039)

[**Figure 25- Bitcoin ACF & PACF** 46](#_Toc139052040)

[**Figure 26- Ethereum ACF & PACF** 46](#_Toc139052041)

[**Figure 27- FTSE ACF & PACF** 47](#_Toc139052042)

[**Figure 28- S&P 500 ACF & PACF** 47](#_Toc139052043)

# **List of Tables**

[**Table 1 – Stock and Cryptocurrency market cap and listed companies/crypto, October 2022** 1](#_Toc139052044)

[**Table 2– Description of variables used within this study** 12](#_Toc139052045)

[**Table 3- Description of stocks and cryptocurrencies used within the study** 13](#_Toc139052046)

[**Table 4- ARIMA variables descriptions** 14](#_Toc139052047)

[**Table 5- GARCH variables descriptions** 16](#_Toc139052048)

[**Table 6- EGARCH variables descriptions** 16](#_Toc139052049)

[**Table 7- LSTM Input, Forget and Output gate variables descriptions** 19](#_Toc139052050)

[**Table 8- LSTM Cell State Update, Candidate Memory and Hidden State variables descriptions** 20](#_Toc139052051)

[**Table 9- RMSE variables descriptions** 23](#_Toc139052052)

[**Table 10- MAPE variables descriptions** 23](#_Toc139052053)

[**Table 11- MAE variables descriptions** 24](#_Toc139052054)

[**Table 12- ARIMA Model Forecasting Results** 29](#_Toc139052055)

[**Table 13- LSTM Model Cryptocurrencies Forecasting Results** 34](#_Toc139052056)

[**Table 14- LSTM Model Stocks Forecasting Results** 34](#_Toc139052057)

[**Table 15- GARCH Model Forecasting Results** 38](#_Toc139052058)

[**Table 16- EGARCH Model Forecasting Results** 41](#_Toc139052059)

[**Table 17- LSTM Models Used For Stocks & Cryptocurrency** 48](#_Toc139052060)

Word Count: **7,923**

# **Chapter 1: Introduction**

Forecasting has been utilised as an effective method for minimising investment risk exposure, as well as to assist investors and academics in attempting to predict financial market movements (Mallikarjuna and Rao, 2019). Markets are massive, which leads to intense competition; forecasts can essentially serve as a tool for investors to make wiser data-driven choices (Bustos and Pomares-Quimbaya, 2020). These high levels of competition are prevalent in the cryptocurrency market due to its relatively new status as an asset type, non-traditional decentralised nature, and exponential growth (Gandal and Halaburda, 2016). *Table 1* depicts the high degree of competitiveness in the cryptocurrency industry; the NYSE has approximately 20x the global cryptocurrency market cap; nevertheless, there are nearly 4x the number of listed cryptocurrencies than listed stocks on the NYSE.

Due to the popularity surrounding forecasting, countless methods and models have been created, ranging from traditional time series analysis to statistical modelling and AI forecasting in the form of artificial neural networks. The key variables forecasted in the existing literature are; volatility, returns, and prices, all of these being critical when it comes to asset valuation.

**Table 1 – Stock and Cryptocurrency market cap and listed companies/crypto, October 2022**

(Data from: Statistica, 2023; WFE, 2023; CoinMarketCap, 2023)

|  |  |  |
| --- | --- | --- |
|  | **Stock Market** | **Cryptocurrency Market** |
|  | NYSE | Global |
| **Market Capitalisation** | $22.77 trillion | $1.01 trillion |
| **Amount of listed Stocks/Crypto** | 2,496 | 9,310 |

## **Research Question**

To what extent do forecasting models differ in their effectiveness, and do specific asset classes generate more significant forecasts?

## **Research Aim**

This study aims to compare time series and artificial neural network returns and volatility forecasting models to assess their forecasting performance on selected asset classes. In addition, the forecasting performance of stocks and cryptocurrencies will be compared to investigate if particular asset classes generate more significant returns and volatility forecasts than others.

## **Research Objectives**

1. To utilise time-series and artificial neural network models to forecast volatility and returns for stocks and cryptocurrency.
2. To evaluate the accuracy of forecasted values through performance metrics, namely: RMSE, MAPE and MSE tests.
3. To compare the performance of artificial neural networks against time series forecasting models.
4. To compare the performance of symmetric and asymmetric volatility forecasting models.
5. To conclude if particular forecasting methods are more beneficial to utilise and if forecasting significance differs between stocks and cryptocurrencies.

## **Rationale**

This study aims to contribute to the literature by including subjects covered throughout the existing literature and combining them into a research case. The literature available is scarce concerning studies centred around comparing stock market and cryptocurrency forecasting, making this study a valuable contribution in that regard. Moreover, this study will attempt to provide helpful advice concerning investing in stocks or cryptocurrency based on the effectiveness of forecasting methods, going further by investigating which forecasting methods provide the most significant values and should therefore be the preferred method used for forecasting.

## **Structure**

This dissertation will be split into five chapters; chapter 1 contains the study’s introduction, as well as its aims, objectives, research question and rationale. Moving forward, chapter 2 critically discusses the theoretical and empirical literature and findings. Chapter 3 illustrates the variables which will be used in conjunction with empirical models, and the justification for those models being evidenced through previous empirical studies within the literature. Chapter 4 will centre around an analysis of the results of the models used within the study, with chapter 5 concluding all results and objectives laid out in chapter 1.

# **Chapter 2: Literature Review**

This section of the study will discuss relevant topics, namely, **“the random walk theory”** due to its relation to the efficient market hypothesis, which is influential in the forecasting literature. Following, time series and artificial neural network models used to forecast the stock and cryptocurrency market will be discussed. Additionally, all topics will be underpinned with theoretical and empirical critical literature discussions, highlighting points of contention and progression.



## **Efficient Market Hypothesis and its Implications on Forecasting**

Traditionally the efficient market hypothesis was the most acknowledged response towards the question of if investors could beat the market. Beginning with the inception of the hypothesis, the theory of random walks in the stock market was first discussed by (Bachelier, 1964). Essentially, the theory states that stock prices and other financial assets move randomly and unpredictably over time; stock prices can only be determined through the collective actions of investors buying and selling, thus meaning it is not feasible to beat the market consistently by relying on past market trends (Cheng and Deets, 1971).

This theory left the literature divided when it was first discussed and is still divided within the literature now. (Cootner, 1962) argued that the stock market did not follow a random walk, stating that the perfect market scenario on which random walk supporters hedge their beliefs is not practical in the real world. Going further by characterising the perfect market scenario, this being the idea of prices reflecting all available knowledge within the market; thusly, price fluctuations would only occur in the presence of new information, which could occur at random intervals, allowing the stock market to be viewed as a random walk. Nevertheless, (Cootner,1962) presented a critique of the stock market, highlighting its imperfections. One of his arguments was that the transition matrices of price changes within trends lacked the property of zero price change expectations, contradicting the notion of the market behaving as a random walk.

Eventually, (Fama, 1965) coined the term **“efficient market”,** its meaning being similar to the perfect market scenario discussed by Cootner. (Fama, 1970) created the definitive and most cited paper on the Efficient Market Hypothesis, discussing its forms and definition. The mentioned EMH’s forms are divided into; weak, semi-strong, and strong. The weak form relates to stock prices reflecting all historical price data; the semi-strong form states that stock prices reflect all public information within the market, meaning only private information can give investors increased returns; finally, the strong form relates to stock prices reflecting all public and private information within the market, meaning investors cannot beat the market (Fama, 1970).

The presence of the EMH essentially results in all attempts to beat the market through forecasting being futile; even the weakest form of the EMH covers all historical price data, which is the basis of any attempt to forecast stock returns, prices and volatility. However, similar to the RWT, the EMH is also heavily disputed within the literature. (Malkiel, 2003) and (Tóth and Kertész, 2006) argue in favour of the EMH, stating that the market is more efficient than studies against the EMH claim.

Additionally, there have been empirical studies to either prove or disprove the existence of the EMH in real-world markets. (Narayan and Narayan, 2007) analysed monthly stock prices for G7 countries, panel unit root tests were then utilised to determine the stationarity of the data set. A unit root within the panel data is consistent with the random walk process, suggesting that the market is efficient and validifies the EMH. The tests conducted, namely IPS, LM, SUR and MADF, all indicated that stock prices within G7 countries have a unit root and are consistent with the EMH. Note, (Narayan and Narayan, 2007) emphasised the lack of structural breaks within the panel unit root tests, indicating that the tests ran could have been more robust.

Contrastingly, a study by (Narayan and Smyth, 2005) utilised similar panel unit root tests with breaks, and results indicated that markets were efficient. Daily stock prices for 22 OECD countries were used in the study; structural trend breaks unit root tests, t-bar panel unit root tests, and LM panel unit root tests were used to assess the stationarity of the data set. The results showed that only Mexico and New Zealand failed to reject the null hypothesis of the data being random walk, strongly supporting the EMH.

Ultimately, the literature is varied in its opinion of the EMH; both sides provide strong arguments. Within this study, the view of the EMH is consistent with the beliefs of (Timmermann and Granger, 2004). Those being that in the short-run, forecasting may provide investors profits, the reasoning being that innovation drives new and improved forecasting methods to be created, allowing investors to continue to benefit from predicting the market.

## **Time series Forecasting**

Within the existing literature, there are many ways to attempt to forecast market assets. Time series forecasting is a popular and influential method within the literature (Pai and Lin, 2005). Typically, time series models utilise collections of data items collected at successive and identical time intervals; through analysis, the given data can be analysed and used to predict future data values (Mondal et al., 2014). The theoretical basis for these models’ forecasting ability is that stock prices or other financial market variables are not randomly generated values, meaning they are not a random walk; thus, they can be treated as a discrete time series model, and the associated model’s trend can be analysed, allowing future values in the trend to be forecasted (Mondal et al., 2014).

A key and widely used time series forecasting method within the literature is the ARIMA model, which will be utilised in this study. Following the work of Slutsky, Walker, Yaglom and Yule in conceptualising AR and MA models, (Box and Jenkins, 1970) combined their efforts and created the ARIMA, Autoregressive Integrated Moving Average, model. (Ariyo et al., 2015) illustrates the three stages used to forecast a discrete time series, namely, identification, estimation and verification.

The theoretical basis underpinning the model is as follows:

* The AR component looks at the financial variables’ relation to past values,
* Integration deals with making the data set stationary, which is needed to forecast future values,
* Moreover, MA looks at how the financial variables are related to past forecasting errors (Siami-Namini et al., 2018) (Stellwagen and Tashman, 2013).

After the model is built, future values can be forecasted.

Although the ARIMA model is extensively used within the literature, it has its critiques. (Zhang, 2003) discusses the model’s inability to account for non-linear processes. The ARIMA model’s basis is linear, meaning a direct relationship between past and future values within a time series is assumed. The real world is complex in nature, and non-linear processes such as volatile social media sentiment can influence the relationship between past and future values, causing trend breaks and reducing the ARIMA models forecasting accuracy as a result (Khashei and Bijari, 2011).

### **Forecasting Stock Prices and Returns**

Within the literature, there have been various empirical studies on the topic of the forecasting ability of ARIMA models. (Banerjee, 2014) used an ARIMA (1,0,1) model to forecast S&P BSE Sensex stock market index future values based on close price data. The (1,0,1) model was chosen after a series of tests; ACF and PACF tests were used to determine which order of lags would best fit the Sensex stock data, as well as RMSE, MAPE and MAE performance metrics to judge model efficiency. The p-value of the model was significant at the 1% level, indicating that the forecasted market index values were predicted accurately, according to its historical values. (Banerjee, 2014) did note that similar to the discussion above, government fiscal or monetary policy changes would reduce the accuracy of the ARIMA model. A potential workaround discussed was utilising a non-linear forecasting method, for example, ANN or fuzzy time series. Further on, (Karakoyun and Cibikdiken, 2018) attempted to forecast Bitcoin cryptocurrency prices 30 days into the future; LSTM and ARIMA were used to allow time series and ANN model comparisons, and both models were successful in their forecasts. According to the PM used within the study, MAPE, LSTM performed better than the ARIMA model, with a lower score of 1.40% to ARIMA’s 11.86%. The results of this study validify (Banerjee, 2014) discussion relating to ANNs potentially proving more accurate forecasting.

Similarly, (Yamak et al., 2019) forecasted Bitcoin close prices using ARIMA and LSTM, with GRU, another ANN, used in conjunction. The MAPE scores indicated that ARIMA was the most accurate model, with a score of 2.76%, followed by GRU at 3.97% and LSTM with the highest score of 6.80%. These results contradict the previous studies mentioned. However, (Yamak et al., 2019) did mention that RNNs, a form of ANN, tend to perform better on large sets of data; the 1639 rows of data used within the study would be considered a short data set; thus, ARIMA outperforming LSTM and GRU could have been because they were disadvantaged.

### **Forecasting Stock Volatility**

Although the ARIMA model is a popular model within the literature for forecasting stock and cryptocurrency prices and returns, other models are used more intensively concerning volatility forecasting, namely, the GARCH model and its alternatives. Initially, (Engle, 1982) created the ARCH model, improving on its predecessors by taking into consideration the idea of non-constant levels of volatility and forming a link between historical financial data and potential future values. Ultimately, uncertainty plays a crucial role in financial theory, and forecasting can reduce the speculative nature of prices; thus, the ARCH model’s ability to forecast future values rapidly led to it becoming a popular model within the literature (Bollerslev et al., 1992). However, although the model was well received, it still had its critiques. In their study, (Chand et al., 2012) addressed the concern of the high-order problem in the ARCH model, where an increase in the number of lags leads to a significant rise in computational time. This arises because the restricted variance depends on the lag term of the squared error in the previous period. Consequently, the model requires constant verification of the preceding lag to proceed, thereby increasing the overall time required for model completion.

These issues were addressed by the new GARCH model proposed by (Bollerslev, 1986). The model utilised a flexible lag structure and extended memory, thus countering the higher order problem with more processing power and efficiency, thereby speeding up the time required to check previous lag terms.

#### **Symmetric and Asymmetric Forecasting Volatility Models**

A common issue is present with both models, namely their symmetric nature. The issue with symmetric models is that a positive shock to volatility is assumed to have the same effect as a negative shock, of the same magnitude, in terms of their impact on future volatility (Amiri, 2013). (Amiri, 2013) postulates the issue with this assumption. Bad news typically tends to have more significant negative shocks than positive shocks created through good news. This results in a leverage effect that symmetric models cannot capture in their forecast, causing forecasting accuracy to fall. As a result, researchers have developed various asymmetric variations of GARCH models specifically designed to address the issues above, such as the leverage effect (Chong et al., 1999). One influential model in this regard is EGARCH, introduced by (Nelson, 1991). It is worth mentioning that, within the literature, the terms linear, non-linear, symmetric, and asymmetric are sometimes used interchangeably.

Within the literature, there have been considerable amounts of studies attempting to forecast volatility and compare the forecasting ability between symmetric and asymmetric volatility forecasting models. (Gokcan, 2000) conducted a comparative forecasting ability study between symmetric, GARCH, and asymmetric EGARCH, utilising monthly returns data for seven emerging stock markets: Argentina, Brazil, Colombia, Malaysia, Mexico, Philippines and Taiwan. The forecasting ability between the two models was deduced through mean squared error terms, with the lower number of error terms indicating the more efficient model. For all countries except Brazil and Taiwan, the linear/symmetric GARCH model generated fewer error terms and was the more efficient model.

On the other hand, (Lin, 2018) similarly compared the forecasting ability of GARCH and EGARCH; TARCH was also included as another non-linear/asymmetric model within the study. Daily SSE composite index closing prices were used within the research, converting the price data into compounded returns. An ADF test was used to assess stationarity within the data set; the null hypothesis of a unit root being present was rejected, allowing GARCH (1,1), EGARCH(1-1), and TGARCH(1,1) models to be used to forecast values within the data set. All three models performed reasonable estimations; however, EGARCH was the superior model. Both asymmetric models, EGARCH and TARCH, were able to capture the impact of positive and negative shocks, the leverage effect, which the symmetric GARCH model failed to capture. The ARIMA model was also briefly used as a comparison point between itself and GARCH models. The results indicated that GARCH models are more suitable for forecasting the SSE composite index.

## **Artificial Neural Network (ANN) Forecasting**

Another somewhat newer forecasting method used within the literature is ANNs; some of their benefits are already discussed in the previous section above, namely their ability to capture non-linear processes and locate hidden data patterns, thereby improving forecasting model accuracy (Khan et al., 2011). To elaborate, ANNs are composed of layered structures consisting of artificial neurons, including an input layer, a hidden layer, and an output layer. The flow of data follows a sequential path through these layers, entering via the input layer and exiting through the output layer (Vui et al., 2014).

Initially, the LSTM model was created by (Hochreiter and Schmidhuber, 1997) to address the vanishing gradient problem present within the RNN model. (Hochreiter, 1998) conducted a study to discuss the issue. Within the learning phase present in the RNN model, gradients used to update weights within the network become small, thus resulting in the network’s learning ability diminishing and the forecasting ability of the RNN model subsequently falling in tandem. The LSTM model counteracted this issue by using memory cells that selectively decide to either remember or delete information over time; this method efficiently manages gradients, ensuring they do not become too large or small, thus correcting the vanishing gradient problem (Hochreiter and Schmidhuber, 1997).

### **Forecasting Crypto Prices and Returns**

The LSTM model has been utilised in the literature to forecast stock and cryptocurrency returns and volatility. (Yildiz and Yildiz, 2020) conducted a study comparing portfolio strategies, utilising BIST30 Index stocks closing price data to forecast stock prices, which are then converted to returns. Strategies were split into three categories; LSTM-based forecasting strategies, market capitalisation risk-based weighted forecasting methods and BIST30 index funds. The primary method used to compare portfolio strategies were percentage returns and risk. The results indicated that LSTM-based forecasting strategies had the highest average return of 25.42%, with the BIST30 index funds having the lowest of 6.75%. However, the LSTM-based strategies also had the highest level of risk at 19.88%. To note, (Yildiz and Yildiz, 2020) did mention that although the LSTM-based methods had higher annual risk, the returns were significantly higher than other categories, resulting in the risk being worth taking.

Additionally, (Bhandari et al., 2022) utilised the S&P 500 index closing price data in conjunction with macroeconomic features, for example, the interest rate and unemployment rate, to forecast future stock index values. The study also compared LSTM models with a single layer and multiple layers; RMSE, MAPE and r correlation coefficients were used as the performance metrics. A welch test was conducted to determine whether LSTM models with a single layer outperformed multiple layers. The p-value of the test was close to zero and thus significant. Therefore, the null hypothesis of multiple layers being the superior model was rejected, and an LSTM model consisting of a single layer with 150 neurons was determined to have the best forecasting performance. It is important to note that all models within the study did predict the S&P 500 index stock closing prices with some level of accuracy; LSTM models with single layers simply performed the best within the study.

On the other hand, relating to cryptocurrency forecasts. (Lahmiri and Bekiros, 2019) conducted a study comparing the forecasting ability of deep learning models, LSTM, and GRNN models, which are generalised RNNs. Daily Bitcoin, Digital Cash and Ripple close prices were used within the study, with the RMSE performance matrix used to compare the forecasting performance of both models. The study’s results indicated that the LSTM deep learning models had lower RMSE scores for all cryptocurrencies used within the research, the most significant values being 0.0499 for LSTM and 0.3115 for the GRNN model concerning Ripple. Furthermore, despite the GRNN models exhibiting lower forecasting accuracy, they could complete the training phase in seconds, whereas the LSTM model required up to 10 minutes to compute its learning phase. The time taken acted as a cost for utilising the model.

Contrastingly, in a study by (Hamayel and Owda, 2021), the LSTM model and a variant reported the lowest forecasting accuracy. LSTM, bi-LSTM and GRU models were used in the study, along with RMSE and MAPE as the performance measures. Bitcoin, Litecoin and Ethereum datasets containing open, high, low, and close crypto data were used in the study. The forecasting models utilised the closing price data. The variant LSTM model & bi-LSTM were used within the study; the bi-LSTM consisted of running two linked LSTM models in parallel: one operating input in a forward direction while the other in a reverse direction. The study’s findings revealed that the bi-LSTM model displayed the weakest overall forecasting capability among all the cryptocurrencies examined, as indicated by the lowest RMSE and MAPE scores. Conversely, the GRU model exhibited the highest forecasting ability.



# **Chapter 3: Methodology**

# 

## **Data Collection**

This study primarily focuses on employing time series and ANN forecasting models, using time series data to attempt to forecast future stock and cryptocurrency returns and volatility. Thus, closing price data for the following stocks and cryptocurrencies were obtained from Yahoo Finance between 2018-2023; FTSE 100, S&P 500, Bitcoin, and Ethereum.

### **Variables**

**Table 2– Description of variables used within this study**

|  |  |  |  |
| --- | --- | --- | --- |
| **Variable** | **Description** | **Sources** | **Justification** |
| Close | The last price of the stock that is transacted before the market closes for normal trading. | Yahoo Finance | (Banerjee, 2014), (Karakoyun and Cibikdiken, 2018), (Patel et al., 2020), (Yildiz and Yildiz, 2020), (Amirshahi and Lahmiri, 2023), (Hamayel and Owda, 2021), (Lahmiri and Bekiros, 2019) |
| Return(%) | The daily change in the price of an asset over time. | Yahoo Finance | (Bildirici and Ersin, 2009), (Chong et al., 1999), (Gokcan, 2000), (Yildiz and Yildiz, 2020), |
| Log Returns | The log of the daily change in the price of an asset over time. | Yahoo Finance | (Vo and Ślepaczuk, 2022), (Chudziak, 2023) and (Lu et al., 2016) |

Closing prices represent the final price of a stock or cryptocurrency traded on a given day; thus, they are used as an indicator of the performance of an asset throughout the day (Vijh et al., 2020). Therefore, closing prices have been used consistently within forecasting literature; the reasoning being that if forecasting models are given historical data based on stocks or cryptocurrencies performance, future performance can be forecasted. Examples of studies within the literature that have utilised close and returns historical data for forecasting have been included in **Table 2**.

Furthermore, log returns are used in this study’s ARIMA and GARCH/EGARCH models. The reasoning is that converting normal returns to log returns is a simple process. Additionally, log returns follow a normal distribution; this is beneficial as variables which follow normal distributions can be forecasted with higher degrees of accuracy, as there are no negative values (Vo and Ślepaczuk, 2022).

### **Assets**

**Table 3- Description of stocks and cryptocurrencies used within the study**

|  |  |  |  |
| --- | --- | --- | --- |
| **Asset Class** | **Ticker** | **Name** | **Source** |
| **Stock** | ^FTSE | FTSE 100 | Yahoo Finance |
| ^GSPC | S&P 500 | Yahoo Finance |
| **Cryptocurrency** | BTC-USD | Bitcoin | Yahoo Finance |
| ETH-USD | Ethereum | Yahoo Finance |

In this study, stock indexes were chosen as the selected stocks. This decision was based on the understanding that stock indexes exhibit lower volatility than individual stocks. This is because stock indexes represent a combination of multiple stocks rather than focusing on a single stock. (Bhowmik and Wang, 2020). Moreover, stocks with higher volatility generally result in more volatile stock return forecasts (Ellahie and Peng, 2021). Therefore, stocks with lower volatility may facilitate improved forecasting ability. **Table 3** displays all the stocks and cryptocurrencies used within the study and their data source.

## **Data Limitations**

The forecasting ability of the models used within this research depends on the historical data inputted into the model. Within this study, the data period analysed spans from 2018 to 2023. The rationale behind this choice is that the earliest data available for cryptocurrencies, specifically Ethereum, starts in 2017. Conversely, data for all the stock market indexes used in the study has been available since 2007. Furthermore, considering the high level of volatility observed in Ethereum during 2017, the starting point for the data set was set in 2018. Therefore, model forecasted values accuracy would be impacted by omitting several years’ worth of data points relating to the stock indexes and cryptocurrencies.

Similar to (Banerjee, 2014), macroeconomic data was not used in this study. It is worth noting that macroeconomic variables are commonly believed to influence stock volatility, and incorporating macroeconomic data is known to enhance forecasting accuracy, as highlighted in the research by (Jabeen et al., 2022). Consequently, this study’s absence of macroeconomic data may result in lower forecasting accuracy.

## **Data Cleaning**

Missing values within datasets can lead to decreased model forecasting accuracy and introduce bias by limiting data points (Ivanovski and Hailemariam, 2023) (Jeris et al., 2022). Therefore, similar to (Lin, 2018), an ADF test will be conducted to determine the stationarity of the data set. Suppose the null hypothesis of a unit root being present is accepted. In that case, methods such as differencing the data set will be applied to remove the unit root present and bring stationarity to the data set (Pesaran, 2012).

## **Forecasting Methods**

### **Time Series Forecasting Models**

This section includes descriptions of all time series models used to forecast within this study.

#### **AutoRegressive Integrated Moving Average (ARIMA)**

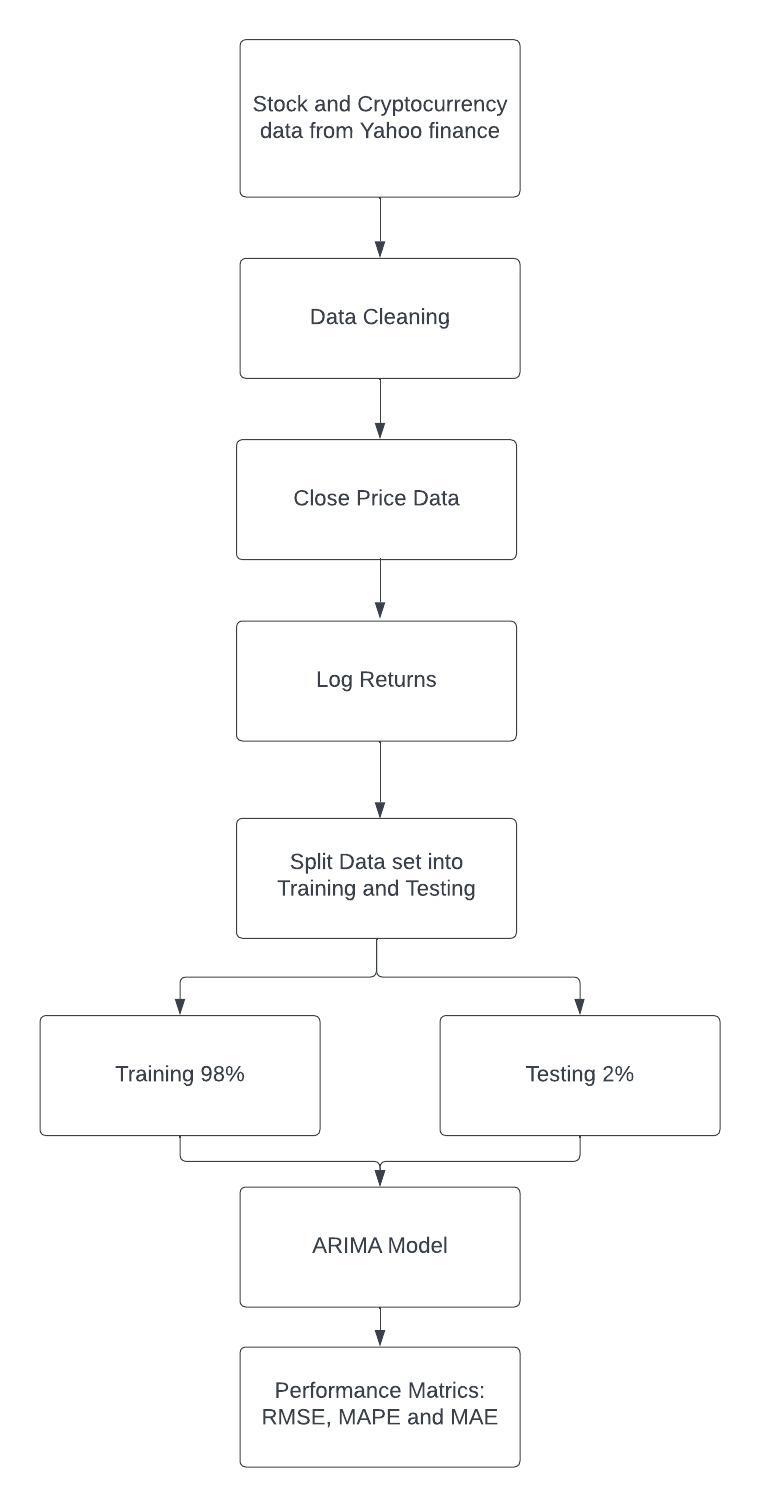
|  |  |
| --- | --- |
|  |  |

The ARIMA(p,d,q) model is split into three sections; p order of AR term, d number of differencing, and q order of MA term.

**Table 4- ARIMA variables descriptions**

|  |  |
| --- | --- |
|  | Asset price, the t subscript denotes the day. |
|  | The constant term within the equation. |
|  | The linear combinations of lags on Y up to the max of p lag. |
|  | The linear combination of lags on forecast errors, X, up to the max of q lag. |
|  | The current error term at time t. |

#### **ARIMA Forecasting Process Flowchart**



**Figure 1- ARIMA Forecasting Flowchart**

The ARIMA(p, d, q) model, shown in **Equation 1.1** and **Table 4**, is configured utilising the auto.arima function within r, this is based on the (Hyndman and Khandakar, 2008) study, which combines unit root tests with AIC and MLE performance evaluators and minimisation to obtain the best ARIMA model for the data set. The forecast package is employed to perform ARIMA forecasting using the model derived from the auto.arima function. The training set data is utilised for training the model and assessing its accuracy by comparing the forecasted testing values against the original and performance metric values. **Figure 1** displays the process flowchart.

#### **Generalised AutoRegressive Conditional Heteroskedasticity (GARCH)**

|  |  |
| --- | --- |
| , |  |

**Table 5- GARCH variables descriptions**

|  |  |
| --- | --- |
|  | Value of the time series at time t. |
|  | is the conditional standard deviation of the time series at time t. is the random error term at time t. |
|  | The conditional variance of an asset’s returns at time t. |
|  | The constant term within the equation. |
|  | The number of lagged conditional variances up to max p lag. |
|  | The number of lagged squared residuals up to max p lag. |

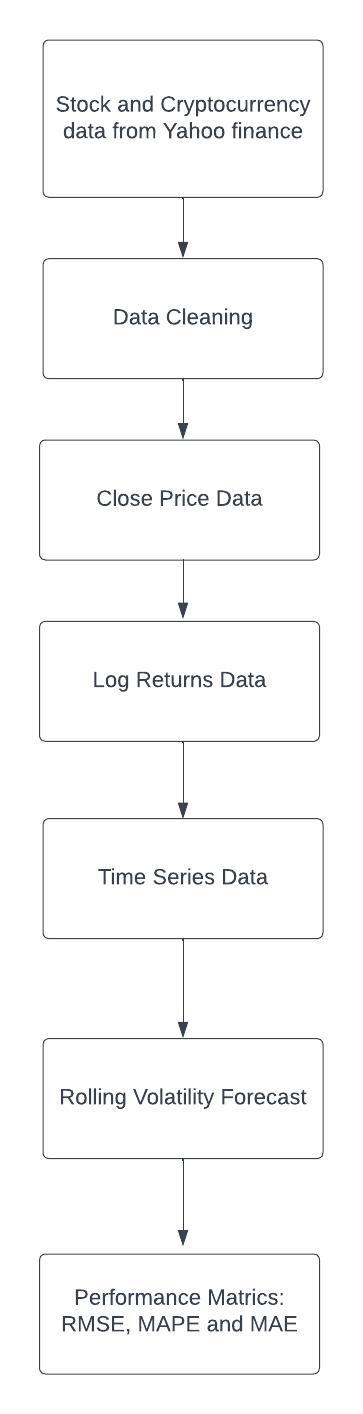
#### **Exponential GARCH**

|  |  |
| --- | --- |
|  |  |

**Table 6- EGARCH variables descriptions**

|  |  |
| --- | --- |
|  | The natural logarithm of the conditional variance. |
|  | The constant term within the equation. |
|  | Represents the sum of max p lagged conditional log variances. are the coefficients of the lagged log conditional variances. is the natural logarithm of lagged conditional variance, representing the log of squared volatility at time t-i. |
|  | Represents the sum of max q lagged error terms. is the coefficient of the lagged error terms. represents the lagged error term at time t-j. |
|  | Represents the sum of max r lagged absolute error terms. is the coefficient of the lagged absolute error terms. is the absolute value of lagged error terms at time t-k. |

#### **GARCH & EGARCH Forecasting Process Flowchart**



**Figure 2- GARCH/EGARCH Forecasting Flowchart**

Within this study, similar to (Verma, 2021), a rolling forecast is used in forecasting volatility. Unlike the abovementioned study, we utilise four months for the rolling forecast instead of the 14 and 28 used within the study. The rolling forecast takes a fixed data window within the data set and regularly updates forecasts without incorporating new data points (Brooks, 1998). GARCH and EGARCH followed the same process shown in **Figure 2**, except the forecasting model used within the rolling forecast was changed to EGARCH from GARCH when conducting the EGARCH forecasts. Additionally, (1, 1) is used as the order for lagged square errors and autoregressive terms as it has been extensively tested within the literature and deemed efficient enough to be used for most intentions (Hayek and Naimy, 2018). **Tables 5** and **6** also contain variable descriptions for GARCH **Equation 2.1** and **EGARCH 3.1**.

### **Artificial Neural Network Forecasting Models**

This section includes all of the ANN models which will be used to forecast within this study.

#### **Lost Short-Term Memory (LSTM)**

|  |  |  |
| --- | --- | --- |
| **Input Gate** |  |  |
| **Forget Gate** |  |  |
| **Output Gate** |  |  |
| **Cell State Update** |  |  |
| **Hidden State** |  |  |
| **Candidate Memory** |  |  |

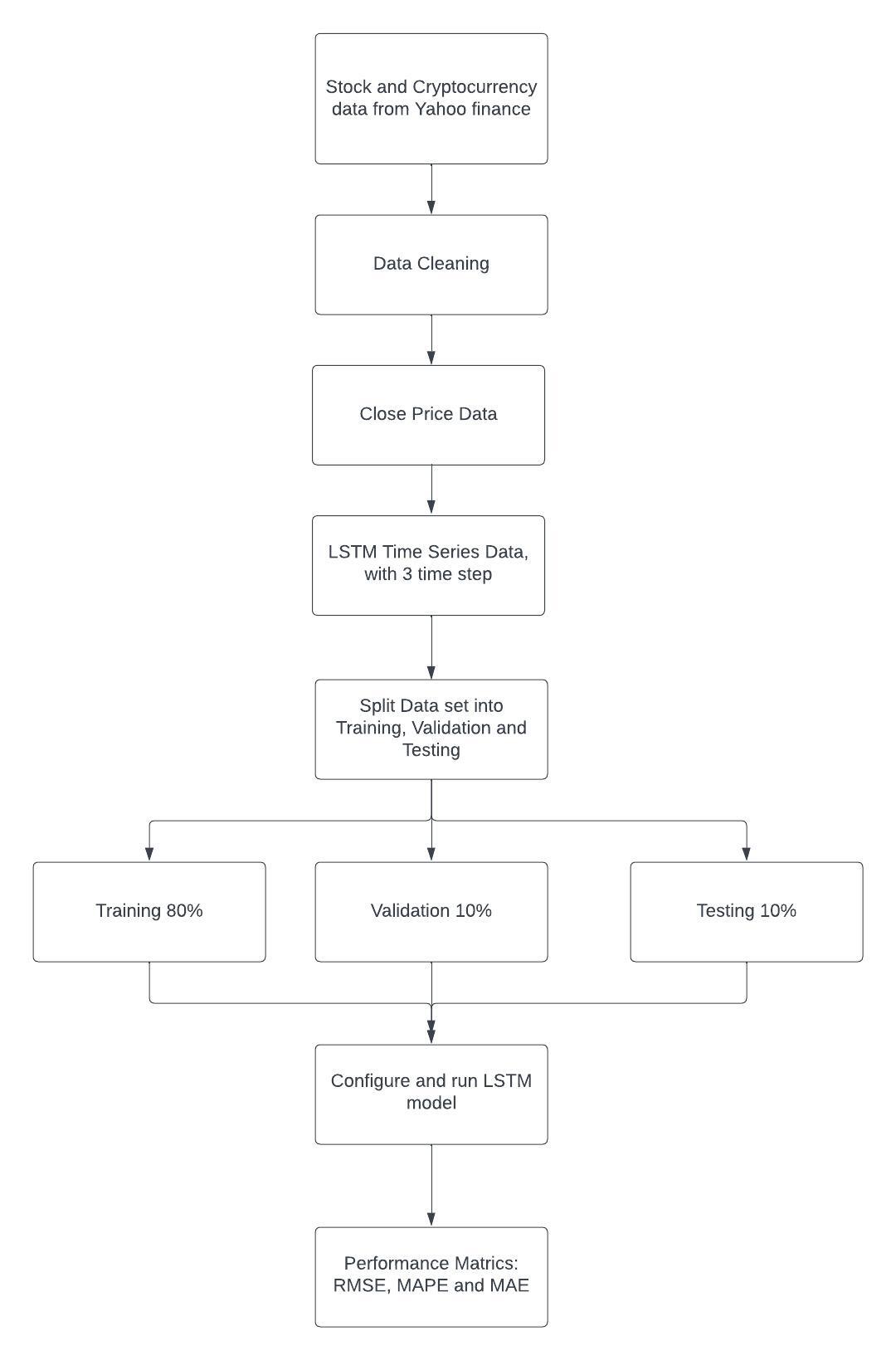
**Table 7- LSTM Input, Forget and Output gate variables descriptions**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Input Gate** | | **Forget Gate** | | **Output Gate** | |
|  | The activation of the input gate at time t. |  | The activation of the forget gate at time t. |  | The activation of the output gate at time t. |
|  | Represents the sigmoid activation function. |  | Represents the sigmoid activation function. |  | Represents the sigmoid activation function. |
|  | Represents the linear transformation of the current input of using the input gate weight matrix. |  | Represents the linear transformation of the current input using the forget gate weight matrix. |  | Represents the linear transformation of the current input using the output gate weight matrix. |
|  | Represents the linear transformation of the previous hidden state using the input gate weight matrix and previous hidden state . |  | Represents the linear transformation of the previous hidden state using the forget gate weight matrix and previous hidden state . |  | Represents the linear transformation of the previous hidden state using the output gate weight matrix and previous hidden state . |
|  | Represents the linear transformation of the previous cell states using the input gate weight matrix and the previous cell state . |  | Represents the linear transformation of the previous cell states using the forget gate weight matrix and the previous cell state . |  | Represents the linear transformation of the previous cell states using the output gate weight matrix and the current cell state . |
|  | The bias term that is related to the input gate. |  | The bias term that is related to the forget gate. |  | The bias term that is related to the output gate. |

**Table 8- LSTM Cell State Update, Candidate Memory and Hidden State variables descriptions**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Cell State Update** | | **Candidate Memory** | | **Hidden State** | |
|  | Represents the current cell state at time t. |  | The candidate cell state; represents new information which can be added to a cell state. |  | Represents the hidden state or the output of the LSTM cell at time t. |
|  | The forget gate; controls how much of the previous cell state should be kept. |  | The hyperbolic tangent activation function. |  | The output gate; controls how much of the current cell state should be outputted. |
|  | The previous cell state at time t-1. |  | Represents the linear transformation of the current input using the candidate memory weight matrix. |  | Represents the application of the hyperbolic tangent activation function on the cell state. |
|  | The input gate; controls how much of the current cell state should be included in the new cell state. |  | Represents the linear transformation of the previous hidden state using the candidate memory weight matrix and previous hidden state . |  |  |
|  | The candidate cell state; represents new information which can be added to a cell state. |  | The bias term that is related to the current cell state. |  |  |
|  |  |  |  |  |  |

#### **LSTM Forecasting Process Flowchart**



**Figure 3- LSTM Forecasting Flowchart**

Similar to (Md et al., 2023), this study uses a sequential LSTM model and follows a similar process, shown in **Figure 3**. In the beforementioned study, the original data set consisting of Samsung close prices was split into training, validation, and testing splits. This allowed the sequential LSTM model to be trained utilising the training split data. The model was subsequently employed to forecast the data in the validation dataset, enabling out-of-sample forecasting. It should be noted that the model was initially trained using the training split data, not the validation data. Finally, the model is used to forecast the supposed values within the testing data set, thus allowing the forecasted testing data and original testing data to be compared to assess the forecasting accuracy of the sequential LSTM model. This process will be implemented utilising the close prices of the stocks and cryptocurrencies within this study.

Additionally, before the sequential LSTM model can be run, it will pass through 3 processes; Data Normalisation, Reshape Data and LSTM model building. Initially, Data Normalisation will consist of scaling the data set used for the LSTM model into a range of 0 to 1. This allows the LSTM model to experience improved forecasting accuracy by mitigating potentially significant gradient error values. Secondly, Reshape Data will convert the dataset into one suitable for the LSTM model, meaning a time step is implemented. In essence, each value in the dataset is accompanied by past values determined by the chosen time step. In the study by (Md et al., 2023), a time step of 100 is used.

However, this study chooses a time step of 3 to reduce the time required for the LSTM model to complete its forecasting process. Lastly, when constructing the LSTM model, various components such as layers, epoch size, optimiser, and loss functions are considered, similar to the approach followed by (Md et al., 2023). The epoch size refers to how many model iterations should be computed before releasing the final forecast.

In this study, the Adam optimiser is employed, and the configuration of layers and epoch size is adjusted for each stock and cryptocurrency, aiming to achieve the most accurate LSTM forecast based on the MSE metric used as the loss function. Furthermore, dropouts will be used to mitigate overfitting within the model, which is common with LSTM models and occurs when the training data has much higher accuracy than the validation data (Zhou et al., 2022). The data sets will be split into 80% training, 10% validation and 10% testing, similar to (Shah et al., 2018), which concluded with their LSTM model providing significant daily forecasts. The LSTM model components are shown within **Equation 4.1**, with variable descriptions in **Tables 7** and **8**.

## **Performance Metrics**

This section will include all the performance metrics that will be used to assess the accuracy and performance of forecasting models used within the study. Within this study, similar to (Lim and Sek, 2013), the performance metrics used are RMSE, MAPE and MSE. The reasoning is that they are metrics extensively used within the active literature and have become the most favoured. The performance metrics models are shown in **Equations 5.1 – 7.1** and **Tables 9 -11** variable descriptions.

### **Root Mean Square Error (RMSE)**

|  |  |
| --- | --- |
|  |  |

**Table 9- RMSE variables descriptions**

|  |  |
| --- | --- |
|  | The predicted value |
|  | The actual value |
|  | The total sample size |
|  | The summation of all values within the data series, beginning with the first value up to |

### **Mean Absolute Percentage Error (MAPE)**

|  |  |
| --- | --- |
|  |  |

**Table 10- MAPE variables descriptions**

|  |  |
| --- | --- |
|  | The actual value |
|  | The predicted value |
|  | The summation of all values within the data series, beginning with the first value up to |
|  | The division of the absolute percentage difference, calculated through and the total observations within the data set obtains the mean percentage error. |
|  | Converts the calculated mean error number into a percentage. |

### **Mean Absolute Error (MAE)**

|  |  |
| --- | --- |
|  |  |

**Table 11- MAE variables descriptions**

|  |  |
| --- | --- |
|  | The actual value |
|  | The predicted value |
|  | The summation of all values within the data series, beginning with the first value up to |
|  | The division of the absolute sum difference, calculated through and the total observations within the data set obtains the mean absolute error. |

# **Chapter 4: Data Analysis**



## **Data Cleaning**

The data sets were checked to ensure no gaps or unidentifiable values were present. Following this, an ADF test was run to ensure stationarity was present at a significant level within the stock and cryptocurrency data. The results of the ADF tests, shown in **Figures 21-24** in the **Appendix**, indicated that the Bitcoin, Ethereum, FTSE and S&P 500 data sets were all stationary at the 1% level, indicating the presence of no unit roots.

## **Data Exploration**

**Figure 4- Stocks & Cryptocurrency Closing Prices, 2018 - 2023**

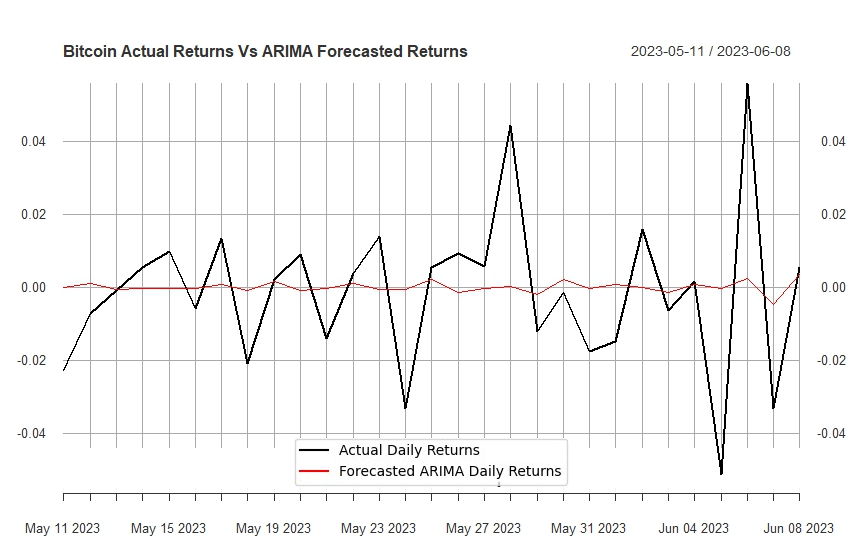
|  |  |
| --- | --- |
|  |  |
|  |  |

**Figure 4** displays the trend of daily closing prices for all stocks and cryptocurrencies used within the study within the 2018-2023 period. The trends seem to correspond to their asset classes. The closing prices of FTSE and S&P 500 stocks exhibit similar trends, while the cryptocurrencies also display a noticeably similar pattern. Moreover, the decline in stock closing prices observed in 2020 aligns with a study conducted by (Huang and Liu, 2021), which explored the impact of COVID-19 on stock prices. The study concluded that stock prices were more susceptible to crashing during the COVID-19 period.

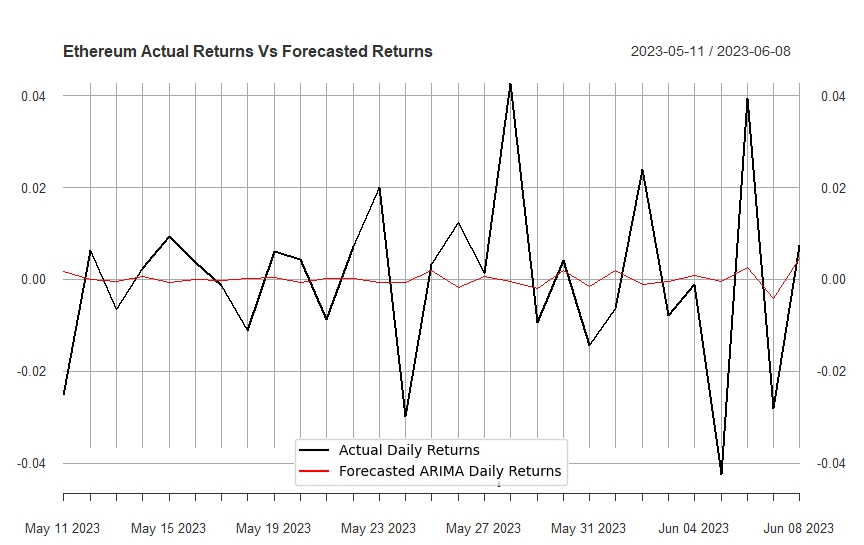
Furthermore, the trend of cryptocurrency closing prices falling at the beginning of 2020 and subsequently rising seems to be consistent with a study by (Demir et al., 2020). The study noted that at the beginning of 2020, cryptos were treated similarly to traditional assets, and hence prices fell. However, due to the uncertainty generated by COVID-19, cryptos were used to hedge against said uncertainty, thus, leading to the steady increase in prices following the initial fall, displayed in **Figure 4**.

## **Returns Forecasting Results**

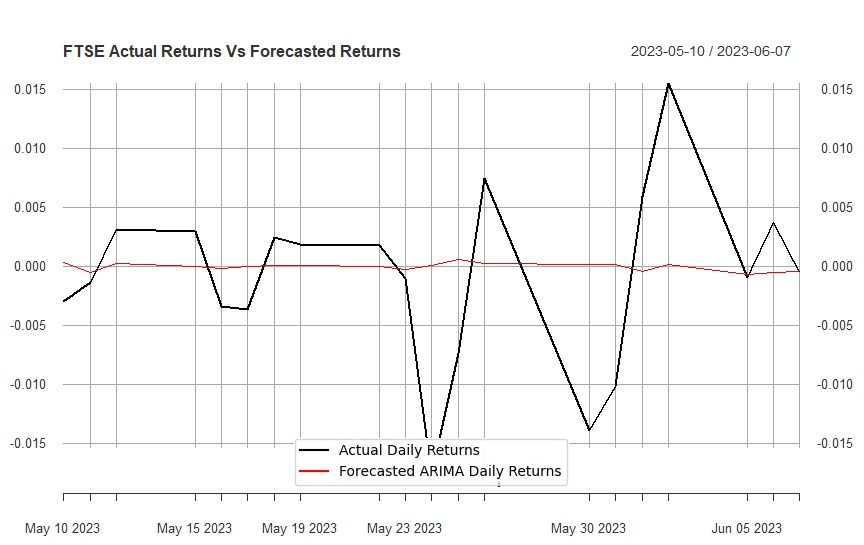
### **ARIMA**



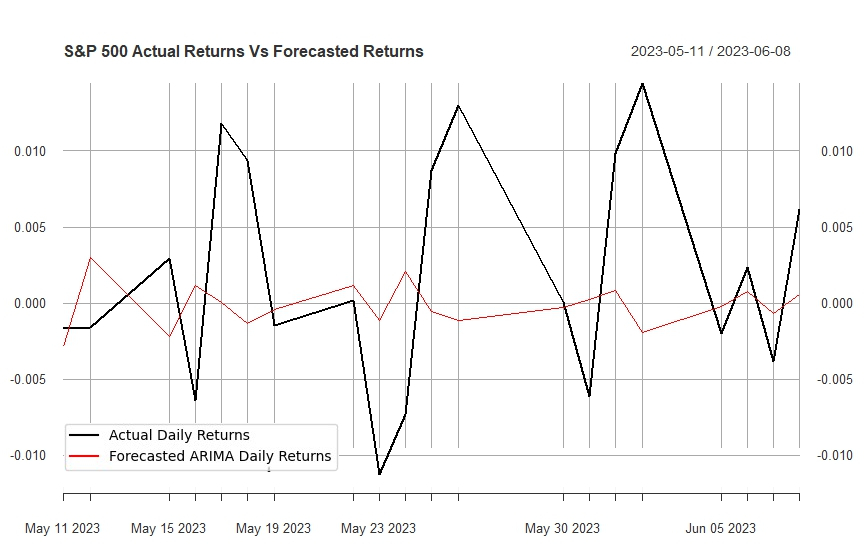
**Figure 5- Bitcoin ARIMA Forecast**



**Figure 6- Ethereum ARIMA Forecast**



**Figure 7- FTSE ARIMA Forecast**



**Figure 8- S&P 500 ARIMA Forecast**

**Table 12- ARIMA Model Forecasting Results**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **Bitcoin** | **Ethereum** | **FTSE** | **S&P 500** |
| **ARIMA Model** | (0, 0, 2) | (1, 0, 1) | (4, 0, 1) | (4, 0, 2) |
| **MAE** | 0.024 | 0.033 | 0.008 | 0.009 |
| **MAPE** | 116.494 | 195.572 | Inf | 182.935 |
| **RMSE** | 0.037 | 0.0480 | 0.012 | 0.013 |

**Figures 5-8** display the ARIMA forecasted graphs for Bitcoin, Ethereum, FTSE and S&P 500. Additionally, **Figures 25-28** in **Appendix** display the ACF and PACF plots for each stock and cryptocurrency. Similar to the (Banerjee, 2014) study, these were used to determine the most appropriate ARIMA model order for forecasting.

As shown in **Table 12**, Bitcoin provided the most significant ARIMA forecast overall according to the performance metrics used, followed by S&P 500, Ethereum and FTSE. The Inf value indicates a large number for FTSE. Overall, stocks provided more significant forecasts than cryptocurrencies - the only metric in cryptos’ favour being the MAPE value. The significant ARIMA forecasting results derived are consistent with the studies of (Banerjee, 2014) and (Poongodi et al., 2020), which conducted ARIMA model forecasts on stocks and cryptocurrency, respectively.

### **LSTM**

**Figure 9- Bitcoin LSTM Testing, Validation, Training and Returns Forecast**

|  |  |
| --- | --- |
| A picture containing text, diagram, plot, screenshot  Description automatically generated | A picture containing text, diagram, plot, line  Description automatically generated |
| A picture containing text, screenshot, plot, diagram  Description automatically generated | A picture containing text, screenshot, line, plot  Description automatically generated |

**Figure 10- Ethereum LSTM Testing, Validation, Training and Returns Forecast**

|  |  |
| --- | --- |
| **A picture containing text, screenshot, plot, font  Description automatically generated** | A picture containing text, screenshot, plot, font  Description automatically generated |
| A picture containing text, screenshot, plot, diagram  Description automatically generated | A picture containing text, screenshot, plot, line  Description automatically generated |

**Figure 11- FTSE LSTM Testing, Validation, Training and Returns Forecast**

|  |  |
| --- | --- |
| A picture containing text, diagram, plot, font  Description automatically generated | A picture containing text, diagram, plot, line  Description automatically generated |
| A picture containing text, screenshot, plot, line  Description automatically generated | A picture containing text, screenshot, plot, diagram  Description automatically generated |

**Figure 12- S&P 500 LSTM Testing, Validation, Training and Returns Forecast**

|  |  |
| --- | --- |
| A picture containing text, plot, diagram, line  Description automatically generated | A picture containing text, diagram, plot, line  Description automatically generated |
| A graph with red and black lines  Description automatically generated with low confidence | A picture containing text, screenshot, plot, font  Description automatically generated |

**Table 13- LSTM Model Cryptocurrencies Forecasting Results**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | **Bitcoin** | | | **Ethereum** | | |
|  | **Training** | **Validation** | **Testing** | **Training** | **Validation** | **Testing** |
| **MAE** | 5836.894 | **8357.512** | 8379.501 | 773.195 | **140.221** | 129.667 |
| **MAPE** | 0.213 | **0.408** | 0.346 | 2.097 | **0.096** | 0.085 |
| **RMSE** | 8300.512 | **10161.288** | 9755.635 | 895.282 | **180.577** | 169.529 |

**Table 14- LSTM Model Stocks Forecasting Results**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | **FTSE** | | | **S&P 500** | | |
|  | **Training** | **Validation** | **Testing** | **Training** | **Validation** | **Testing** |
| **MAE** | 61.653 | **693.870** | 1180.675 | 53.054 | **528.432** | 523.419 |
| **MAPE** | 0.009 | **0.097** | 0.154 | 0.016 | **0.138** | 0.132 |
| **RMSE** | 83.044 | **858.426** | 1280.573 | 71.922 | **630.277** | 671.083 |

**Figures 9-12** display the LSTM forecasted training, validation and testing data graphs for Bitcoin, Ethereum, FTSE and S&P 500 and the forecasted returns. **Table 17** in **Appendix** displays the LSTM models used by the stocks and cryptocurrencies within the study; these were calculated through a series of continual alterations and in choosing the model which provides the most significant forecast based on the performance metrics used.

As shown in **Tables 13 & 14**, the LSTM model struggled to provide significant forecasting accuracy according to the performance metrics used; the only metric which provided significant results was MAPE. The LSTM model struggled to learn from the training data and implement its results through the forecasted validation and testing data, evidenced by the consistently low validation accuracy values for nearly all stocks and cryptocurrencies. This meant overfitting was prevalent. Interestingly, Ethereum demonstrated high training accuracy but low validation and testing accuracy, defying the expectations of overfitting and underfitting scenarios. As a result, the forecast for Ethereum remains uncertain and difficult to determine. These results contradict a study by (Shah et al., 2018), which similarly utilised dropouts and stopped the training portion of the model when training and validation loss were stable. Their results indicated that by utilising these strategies, overfitting could be avoided. However, our results are consistent with (Baek and Kim, 2018) study, which also experienced overfitting in one of their LSTM models due to a lack of a prevention module.

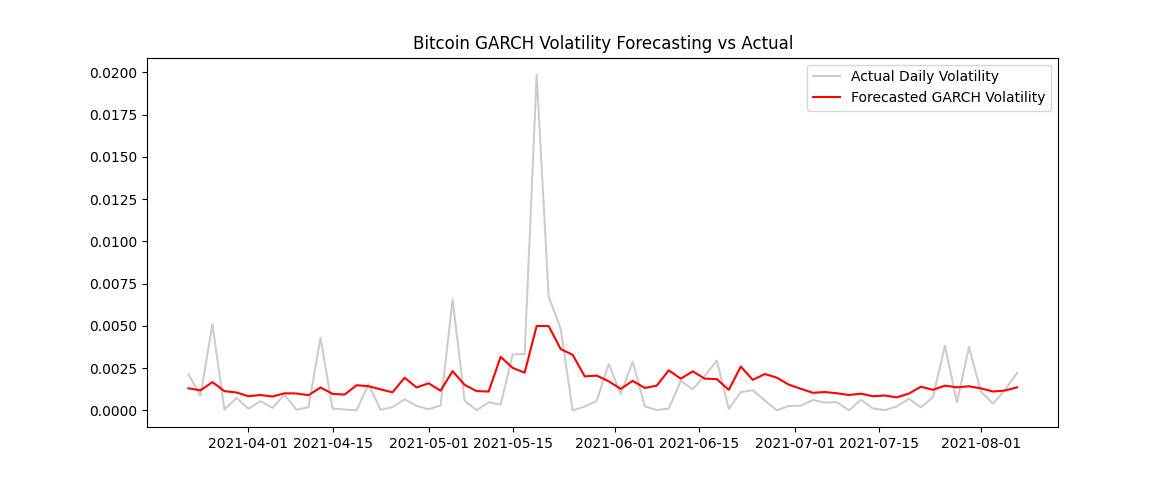
### **Model Comparison Discussion**

Within this study, the ARIMA model provided more significant forecasts than the LSTM model in terms of forecasting stock and cryptocurrency returns. However, the MAPE value was consistently more accurate with the LSTM model. These results contradict (Karakoyun and Cibikdiken, 2018), which found that the LSTM model provided more accurate forecasting results than the ARIMA model. Additionally, the findings of (Siami-Namini et al., 2018) consolidate the previous studies’ results, concluding that the LSTM model improved forecast accuracy by 85% on average compared to the ARIMA model. Similarly, both (Sáenz et al., 2023) and (Alzheev and Kochkarov, 2020) reach the same conclusion.

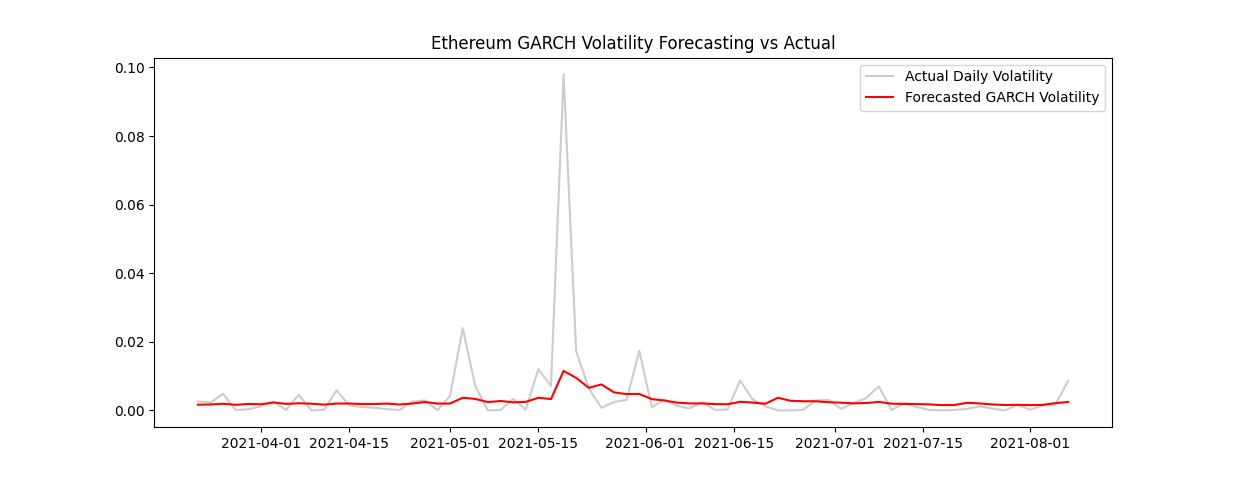
However, a study by (Yamak et al., 2019) concluded that the ARIMA model performed better than LSTM regarding forecasting Bitcoin, which supports the results we received within this study. This is consistent with a study by (Kobiela et al., 2022), which found that through utilising historical price data of selected NASDAQ-listed companies and conducting forecasts through ARIMA and LSTM, the ARIMA model provided vastly more significant forecasts. To note, ARIMA forecasting performance compared to LSTM tended to dip the longer the forecasting period; the ARIMA model performed 3.4 times better than LSTM in 30-day forecasts and only 2.1 times better over nine months.

## **Volatility Forecasting Results**

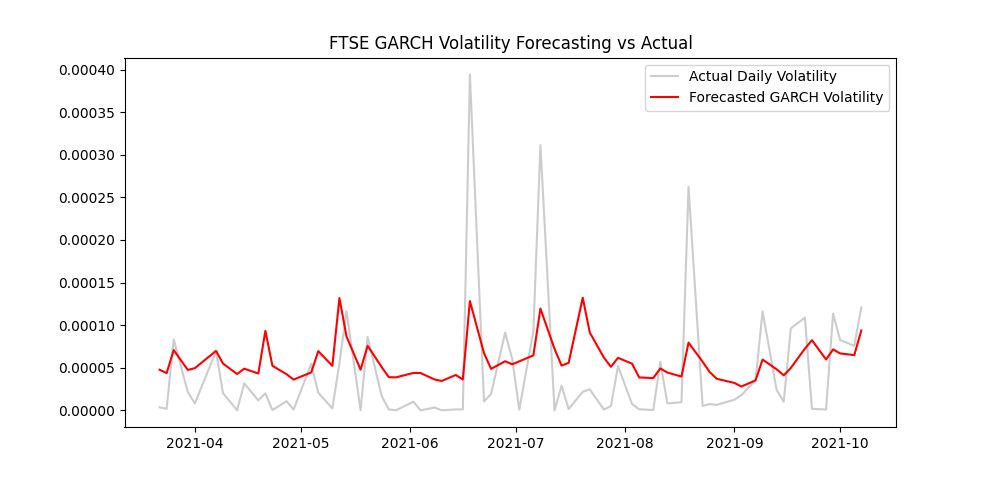
### **GARCH**



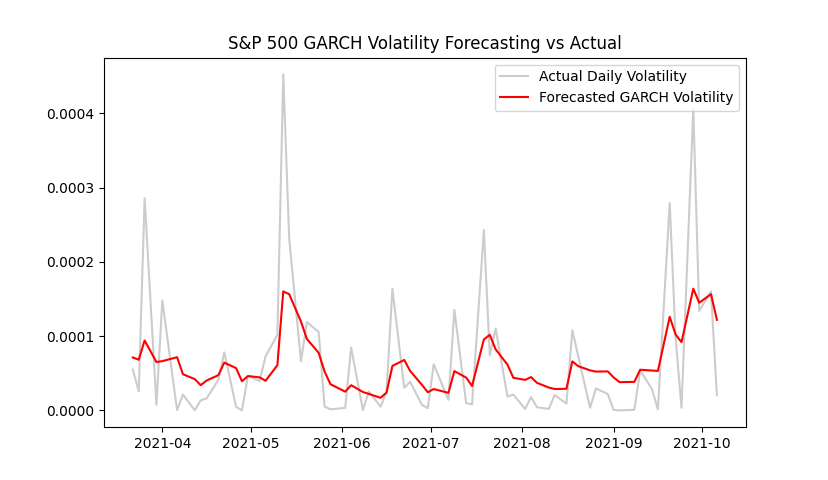
**Figure 13- Bitcoin GARCH Forecast**



**Figure 14- Ethereum GARCH Forecast**



**Figure 15- FTSE GARCH Forecast**



**Figure 16- S&P 500 GARCH Forecast**

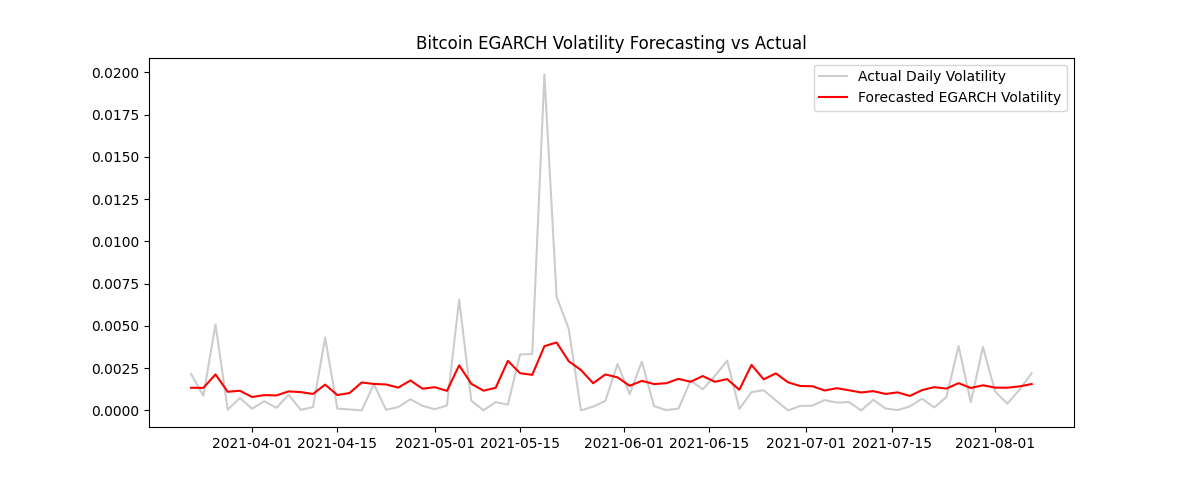
**Table 15- GARCH Model Forecasting Results**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **Bitcoin** | **Ethereum** | **FTSE** | **S&P 500** |
| **GARCH Model** | (1, 1) | (1, 1) | (1, 1) | (1, 1) |
| **MAE** | 0.00131 | 0.004 | 0.00005 | 0.0000458 |
| **MAPE** | 138.39497 | 24.771 | 1084.81409 | 118.1598181 |
| **RMSE** | 0.00225 | 0.011 | 0.00006 | 0.0000692 |

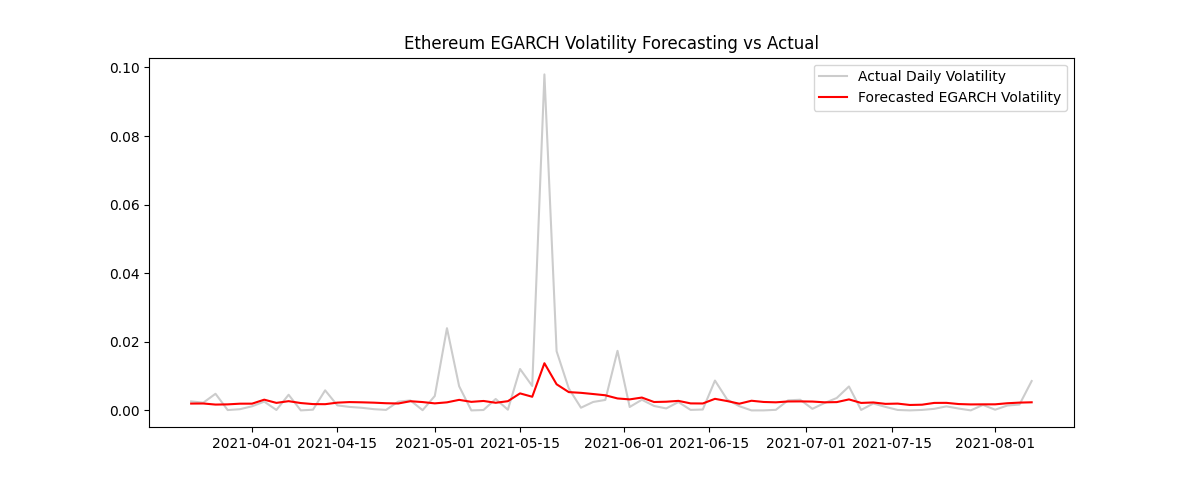
**Figures 13-16** display the GARCH forecasted graphs for Bitcoin, Ethereum, FTSE and S&P 500.

As shown in **Table 15**, Ethereum provided the most significant GARCH forecast overall according to the performance metrics used, followed by S&P 500, Bitcoin and FTSE. Furthermore, overall stocks provided more significant forecasts than cryptocurrencies, the only metric in cryptos’ favour being the MAPE value. The significant stocks GARCH forecasts derived align with the results of (Lin, 2018) and (Gokcan, 2000). Furthermore, the MAE and RMSE values for stocks and cryptocurrencies were statistically significant at the 1% level. This observation aligns with the assertion made by (Andersen and Bollerslev, 1998), suggesting that GARCH models can accurately forecast out of sample.

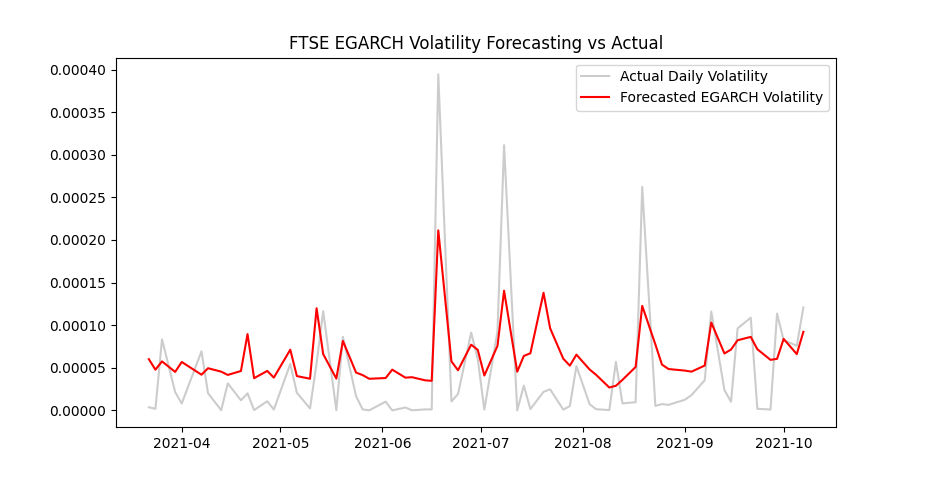
### **EGARCH**



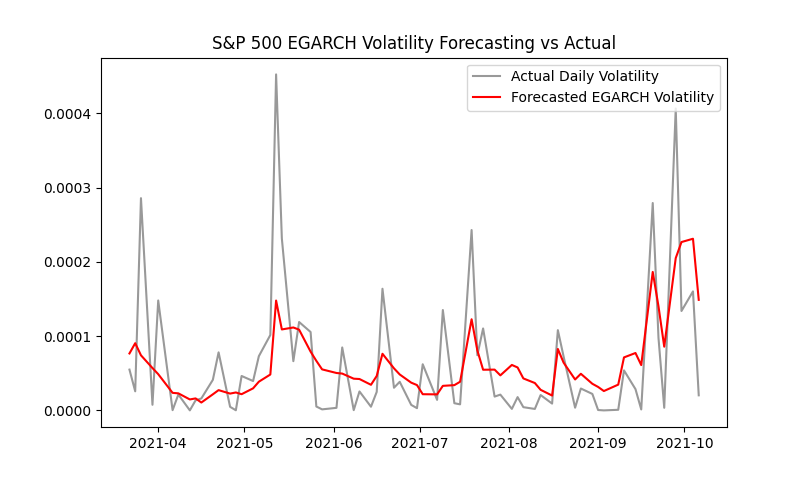
**Figure 17- Bitcoin EGARCH Forecast**



**Figure 18- Ethereum EGARCH Forecast**



**Figure 19- FTSE EGARCH Forecast**



**Figure 20- S&P 500 EGARCH Forecast**

**Table 16- EGARCH Model Forecasting Results**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **Bitcoin** | **Ethereum** | **FTSE** | **S&P 500** |
| **EGARCH Model** | (1, 1) | (1, 1) | (1, 1) | (1, 1) |
| **MAE** | 0.00135 | 0.00357 | 0.00004 | 0.00005 |
| **MAPE** | 115.15216 | 23.70015 | 1165.52457 | 78.13837 |
| **RMSE** | 0.00234 | 0.01081 | 0.00005 | 0.00007 |

**Figures 17-20** display the EGARCH forecasted graphs for Bitcoin, Ethereum, FTSE and S&P 500.

Forecasting results were similar to GARCH, except with slightly higher forecasting accuracy. As shown in **Table 16**, Ethereum provided the most significant EGARCH forecast overall according to the performance metrics used, followed by S&P 500, Bitcoin and FTSE. Furthermore, overall stocks provided more significant forecasts than cryptocurrencies, the only metric in cryptos’ favour being the MAPE value. Additionally, (Wei, 2002) found that in terms of non-linear GARCH models, the EGARCH model is tedious due to not displaying rapid convergence, resulting in poor forecasting performance. This goes against the results of this study in which the EGARCH model operated efficiently and provided significant forecasting performance.

### **Model Comparison Discussion**

Within this study, the EGARCH model provided marginally more significant forecasts than the traditional GARCH model. These results align with the (Lin, 2018) study, which also concluded that the non-linear EGARCH model had greater forecasting power than linear GARCH. Additionally, this is consistent with the findings of (Bergsli et al., 2022), which were that the non-linear models used within the study EGARCH, APGARCH and others typically provided superior forecasting performance over linear GARCH.

However, the results differ from the (Gokcan, 2000) study, which found that for all countries used within the study except two, the linear GARCH model performed better in terms of forecasting performance when compared to EGARCH. This is also consistent with a study by (Lee, 2009), which discusses the notion that asymmetric/non-linear GARCH models, such as EGARCH, are not superior to a linear GARCH(1, 1) model.

## **Limitations and Future Recommendations**

A limitation of this study is the amount of data used within the data sets. Although forecasting models had significant accuracy, others, such as LSTM, suffered low accuracy. As mentioned beforehand, (Yamak et al., 2019) discuss the significance of a large data set concerning forecasting with ANN models. In general, LSTMs and ANN models struggle with smaller data sets. Therefore, a future recommendation would be to choose stocks and cryptocurrencies with similar extensive historical data available, thus potentially mitigating the data constraints ANNs experience and perhaps increasing forecasting accuracy for all models used within the study.

Another limitation is the absence of macroeconomic data, which has been shown to improve forecasting model accuracy within the literature. (Abounoori and Tazehabadi, 2009) utilised ARIMA, ANN and hybrid models to forecast Tehran Stock Index prices, finding that the forecasting models’ accuracy could be improved by including macroeconomic variables. Hence, a future recommendation would be to include macroeconomic data within the forecasting models to improve accuracy.

# **Chapter 5: Conclusion**

To conclude, this study investigated the effectiveness of time series and artificial neural network forecasting models between asset classes, namely stocks and cryptocurrency. A separate point of contention discussed and investigated was the comparison of symmetric and asymmetric time series volatility forecasting models. In terms of time series forecasting models, ARIMA, GARCH and EGARCH were used in this study. ARIMA for forecasting returns and GARCH/EGARCH for forecasting volatility. Furthermore, the LSTM model was utilised in this study in terms of artificial neural networks.

Additionally, log returns were used over normal returns due to their beneficial normal distributions, thus playing a vital role in the study as it facilitated improvements in the forecasting ability of these models. A similar process was applied to the LSTM model through scaling datasets to hold values within a range of 0 to 1, effectively improving the model’s forecasting ability. Lastly, performance metrics such as RMSE, MAPE, and MSE were employed to evaluate the performance of all forecasting models. These metrics served as benchmarks for comparison and enabled the measurement of forecasting accuracy.

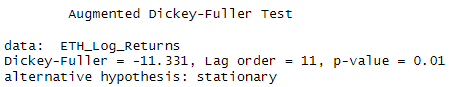
The results of this study illustrate that the ARIMA forecasting model provides significantly more accurate stock and cryptocurrency results than the LSTM model, these results falling in line with (Yamak et al., 2019) and (Kobiela et al., 2022) while invalidating (Karakoyun and Cibikdiken, 2018) and (Sáenz et al., 2023). Additionally, relating to volatility forecasting, the non-linear EGARCH model generated slightly more significant values than its linear GARCH counterpart, thus confirming the (Lin, 2018) study and countering (Gokcan, 2000) and (Lee, 2009). Furthermore, it was found that for all forecasting models used, stocks typically provided more significant forecasts than cryptocurrencies.

# **References**

# **Appendices**



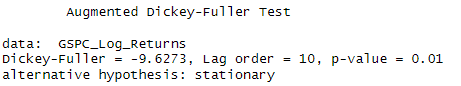
**Figure 21- Bitcoin ADF Test**



**Figure 22- Ethereum ADF Test**



**Figure 23- FTSE ADF Test**



**Figure 24- S&P 500 ADF Test**

**Figure 25- Bitcoin ACF & PACF**

|  |  |
| --- | --- |
|  |  |

**Figure 26- Ethereum ACF & PACF**

|  |  |
| --- | --- |
|  |  |

**Figure 27- FTSE ACF & PACF**

|  |  |
| --- | --- |
|  |  |

**Figure 28- S&P 500 ACF & PACF**

|  |  |
| --- | --- |
|  |  |

**Table 17- LSTM Models Used For Stocks & Cryptocurrency**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **Bitcoin** | **Ethereum** | **FTSE** | **S&P 500** |
| **Layers** | **LSTM(32)**  **Dropout(0.4)**  **Dense(16)**  **Dropout(0.4)**  **Dense(16)**  **Dense(1)** | **LSTM(4)**  **Dropout(0.9)**  **Dense(2)**  **Dense(2)**  **Dense(1)** | **LSTM(32)**  **Dropout(0.6)**  **Dense(32)**  **Dense(32)**  **Dense(1)** | **LSTM(8)**  **Dropout(0.2)**  **Dense(4)**  **Dense(4)**  **Dense(1)** |
| **Epoch** | **200** | **300** | **598** | **500** |