Forecasting and modelling the VIX using Neural Networks

Nomonde Netshivhambe (NTSNOM029)

Supervisor Name: A/Prof. Chun-Sung Huang (PhD)

University of Cape Town
Minor Dissertation in Finance for completion of Master of Science in Data Science
FTX5003W

April 19, 2022

Contents

Abstract

This study investigates the volatility forecasting ability of neural network models. In particular, we focus on the performance of Multi-layer Perceptron (MLP) and the Long Short Term (LSTM) Neural Networks in predicting the CBOE Volatility Index (VIX). The inputs into these models includes the VIX, GARCH(1,1) fitted values and various financial and macroeconomic explanatory variables, such as the S&P 500 returns and oil price. In addition, this study segments data into two sub-periods, namely a Calm and Crisis Period in the financial market. The segmentation of the periods caters for the changes in the predictive power of the aforementioned models, given the different market conditions. When forecasting the VIX, we show that the best performing model is found in the Calm Period. In addition, we show that the MLP has more predictive power than the LSTM.

List of Figures

Chapter 1

Introduction

Volatility can be defined as the standard deviation of asset returns, and it is often considered to be a good measure of market risk. Hence, volatility forecasting plays an important role in financial risk management, asset management and volatility trading. If the volatility of a security can be forecasted, then the fluctuations or pricing behaviour of the security can be estimated over a short horizon. These estimates can then be used for calculating risk adjusted asset allocations, option pricing formulas and for a more effective implementation of financial instruments to hedge volatility risk.

Financial Market participants are often concerned about future market volatility, which can be represented by the VIX. The VIX is expected to contain some predictive content. The VIX is derived from the prices of a specific basket of S&P 500 options. The value of the basket of S&P 500 options depends importantly on the future level of the S&P 500 volatility. Political events, central bank announcements and other market events can be used to deduce future volatility. If financial market participants are rationally processing the deductions then the VIX can be conceived to reflect a crowd-sourced estimate of market uncertainty (Edwards and Preston, 2017).

In the event of a Black Swan (unpredictable or unforeseen) financial market event, the VIX has often been seen as a "fear gauge". During financial crises we tend to observe significant volatility spikes. When Russia devalued the Ruble and Asian Financial Crisis, 1998, the VIX peaked at 1.8 times the 1998 average. During the subprime crisis in 2007 to 2008, the VIX peaked in 2008 at 80.86. The peak was significantly higher than the average VIX level in 2007 (17.53) and 2008 (32.27), this signaled extreme fear in the market. During the most recent crisis, the COVID pandemic, 2020, the VIX reached a new record high of 82.69 (Kochlin, 2020). The COVID pandemic VIX peak coincided with the record 12-13% drop in the Dow Jones Industrial Average, the NASDAQ Composite, and the S&P 500. It was 5.5 times greater than the average VIX level (15.17) that was observed in the period between 2012 and February

2020, this was a period characterised by low to medium volatility. Elevated VIX levels are often associated with capital market shocks, evidenced by sharp price declines in equities markets. Accurately forecasting future market volatility estimates allows practitioners to hedge portolios against normal and tail risk events. However, the forecasting of market volatility can be a difficult task for most financial market practitioners. This study empirically investigates models that can help accurately forecast the VIX, and hence future volatility.

Practitioners have often looked to models that consider realised and/or implied volatility when forecasting future volatility. Additional considerations have shown that implied volatility models have often been favoured over realised volatility models, which utilise historical volatility (Busch, Christensen and Nielsen, 2011). Implied volatility models have been considered as a more reliable predictor for volatility of the S&P 500 index. The VIX is a measure of implied volatility. Many trading strategies rely on the VIX for hedging and speculative purposes (Ahoniemi, 2007). The VIX is derived from bid/ask quotes of options on the S&P 500 index, it is widely followed by financial market participants and is considered not only to be the expectation of volatility but also reflects investor sentiment and risk aversion (Audrino, Sigrist and Ballinari, 2020; Chung, Tsai, Wang and Weng, 2011).

Several econometric time series models have been developed over the years to forecast volatility. These models include Generalized Autoregressive Conditional Heteroskedasticity (GARCH), stochastic volatility, historical volatility, option implied volatility etc. (Poon Granger, 2005). With the advent of new developments in machine learning, researchers have also tried relaxing assumptions of time series models and made use of neural network and time series hybrid models. This study will explore a hybrid of artificial neural networks and GARCH family models.

GARCH models are one of the many models that are used for time series forecasting in finance and econometrics. Some of the advantages of econometric models is that they can be theoretically explained and are based on sound statistical logic. One of the assumptions of the model is that explanatory variables must be stationary, and a pitfall is when one of the qualitative variables are mixed, the performance of the model deteriorates significantly (Cybenko,1989). Some of the assumptions and pitfalls can be overcome by making use of an artificial neural network when forecasting. An artificial neural network is a type of machine learning algorithm with fewer restrictions and assumptions. It is a non-parametric data driven approach. Since it is a data driven approach it relies heavily on the availability of data. When such a condition has been met, it follows that an artificial neural network is capable of approximating all continuous and finite functions (Cybenko, 1989). Moreover, Artificial Neural Networks are capable of learning underlying patterns in data where other conventional methods may fail (Hiransha, et al., 2018).

There are different types of artificial neural networks (ANN), such as Feed Forward Neural Network, Recurrent Neural Network (RNN), Generative Adversarial Neural Network, Convolutional Neural Network (CNN), among others. The most basic form of an ANN is a Feed Forward Neural Network where data flows from the input layer to the output layer without going backward. Hence, links between the layers are one way. Layers never touch the same node again, thus information does not persist. RNNs allow information to flow back into previous parts of the network thus each model in the layers depends on past events, which allows for information to persists. RNN can use their internal memory to process sequences of input data and are thus suitable to use with time series data. One disadvantage of RNNs is the long-term dependency problem where information gets lost over time because of the exploding and vanishing gradient problem (Qiu, Wang and Zhou, 2020). On the other hand, Neural Networks such as LSTM solve the long-term dependency problem. LSTMs are a special type of RNN that are capable of learning long term dependencies and finding patterns across time such that future forecasts make sense. Finally, the CNN, which are mostly used for computer vision and are able to learn deeper architectures than RNN. CNNs solve RNNs gradient vanishing and exploding problem, since they can be stacked into deeper architectures (SPRH LABS, 2019).

The Multi-Layer Perceptron (MLP) is a feed-forward neural network that creates a latent space where appropriate inputs are mapped from a set of inputs. The MLP has multiple layers where each layer is fully connected to the next layer. An MLP consists of three types of layers: an input layer, an output layer, and one or more hidden layers. If a neural network's architecture has more than one hidden layer, the neural network can be referred to as a deep learning neural network (Yang, 2019). When working with artificial neural networks this paper will focus on MLP and LSTM neural networks.

Several research papers have studied the forecasting power of neural networks in relation to financial time series. The papers have shown that artificial neural networks have the capability to accurately forecast various time series data such as stock price returns (Liu, Zhang and Ma, 2017). In this research, we explore whether the VIX can be more accurately predicted using neural networks.

The objective of this study is to investigate the forecasting power of the MLP and LSTM neural networks on the VIX time series, using GARCH (1,1) fitted values and other macroeconomic variables. We also conduct a comparative analysis of the neural networks. Moreover, we investigate how this forecasting power changes across different volatility regimes, namely Crisis Periods with medium-to-high volatility and Calm Periods where there is low-to-medium volatility. If the volatility fluctuates rapidly over a given period that is considered high volatility and if more stable over a longer period of time, it is considered low volatility. We will expand on this in Chapter 3 where we detail the research methodology.

The remainder of this study is organised as follows. Chapter 2 presents the Literature View, in particular, research on volatility modelling and neural networks. Chapter 3 presents the data used in the modelling process. In addition, it introduces the methodology in the form of the models used to forecast the VIX and the technique employed when segmenting the data into two regimes. Finally, Chapter 4 investigates the forecasting ability of the models, and Chapter 5 concludes the study.

Chapter 2

Literature Review

Numerous research have been dedicated to forecasting volatility and the VIX index. Poon and Granger (2005) looked at 93 studies that conducted tests on volatility forecasting methods. They showed that there was no clear dominant model amongst the time series models investigated. In some instances they ranked the best performing model as historical volatility, followed by GARCH and then stochastic volatility. The results in their study also showed a possible ranking where option implied volatility models outperformed time series models. In practice, the option implied volatility is given by the VIX index levels and has been used as a more reliable predictor of volatility in stock markets.

Liu, Guo and Qiao (2015) proposed a new GARCH approach for forecasting the VIX and variance risk premium. They used the GARCH, GJR-GARCH and Heston Nandi models to forecast the VIX. To assess their models, they used one-day out of sample VIX level. The study showed that the traditional empirical GARCH parameters underestimated the VIX level by 20-30% on average before 22 September 2003 and 10-13% thereafter. It is worth noting that after 22 September 2003 a new methodology had been developed to calculate the VIX by CBOE. Awartani and Corradi (2005) have shown how the predictive ability of both symmetric and asymmetric GARCH models decreases over longer time horizons, so as more data points are used the performance of GARCH models may deteriorate. The deterioration of the GARCH models may also be compounded by observing time series where the stationarity assumption does not hold. A stationary time series exhibits statistical properties such as constant mean, variance, over time.

Stationarity can be considered a restricting assumption. We also observe that GARCH models may struggle to work with huge amounts of data and long-range dependence. Such shortfalls may be remedied through the use of Neural Networks. Neural Networks are data driven approaches which often outperform when there is a large amount of data. Moreover, neural networks such as RNN and LSTM are capable of dealing with long-term dependencies in input sequences. Preliminary analysis into VIX show that the index displays long-range dependency, hence, there exists long memory in both option implied and realised variances (Fernandes, Medeiro, Scharth, 2014). Therefore, Neural Networks may be able to adequately forecast the VIX.

Several research papers have been dedicated to assessing how neural networks perform with financial time series. Liu, Zhang and Ma (2017) used a CNN and LSTM for quantitative strategy analysis in stock markets. They used a CNN for stock selection and LSTM for stock price prediction as timing signals. They found that the CNN with LSTM quantitative strategy significantly outperformed the benchmark, with predictions for stock market prices by LSTM being highly accurate. The overall model was found to be feasible, robust and highly profitable. Hiransha et al. (2018) used four different types of deep learning architectures for the National Stock Exchange of India Limited (NSE) and New York Stock Exchange (NYSE) stock market prediction, namely Multiple Layer Perceptron, Recurrent Neural Networks, Convolutional Neural Network and Long Short-Term Memory. They trained the architectures with one single company from the NSE and predicted for five other different companies from both the NSE and NYSE. The study found that the CNN outperformed the other models. Moreover, although the network was trained on NSE stock price data it was able to predict NYSE stock price data because both stock markets share some common inner dynamics. CNN was more capable of capturing abrupt changes in the stock prices and capturing most of the seasonal patterns.

For forecasting the volatility of a stock price index, Kim and Won (2018) used a hybrid model integrating multiple GARCH type models (GARCH, EGARCH, EWMA). They compared the performance of single plain GARCH type models, single neural network models and then the innovative hybrid models. They built separate models with the KOSPI 200 Index, Gold Price, Oil Price, CB Interest rate and KTB interest rate. The models performed in line with theoretical studies. The EWMA model was suitable for capturing short term changes, the GARCH model captured volatility clustering and leptokurtic distribution, and finally the EGARCH was suitable for leverage effect modelling. The authors combined the different GARCH type models and LSTM to fully capture the advantages of a hybrid model. In particular, they combined GARCH, EGARCH, and LSTM; GARCH, EWMA and LSTM; EGARCH, EWMA and LSTM to predict stock price volatility. They also had varying combinations of single type GARCH models with LSTM, single type GARCH models with a feed forward neural network and a triple hybrid type models with LSTM. They found that triple hybrid models with GARCH, EGARCH, EWMA and LSTM had the best prediction accuracy. However, the triple hybrid type models overlapped certain parameter characteristics with the double hybrid model. Different hybrid models have been tested by several authors to forecasting stock price index volatility. Bildrici and Ersin (2012) combined Markov Switching GARCH models with Neural Networks, specifically, the multiple layer perceptron model, radial basis functions and hybrid MLP models. They used the radial basis function model to train and generate both time series forecasts and certainty factors. The multi-layer perceptron consists of an input, hidden and output layer. The hybrid MLP added an additional layer of input to the multi-layer perceptron. The regime switching GARCH Neural Network allowed the generalization of Markov Switching GARCH type models. They modelled the stock price index volatility of the IMKB100 stock index. Their work showed how combining the asymmetric power of GARCH models and neural networks led to better results when forecasting volatility.

Over the years there has been a significant amount of academic papers focused on using neural networks and hybrid models for financial time series forecasting. Research has shown that there is a huge benefit to combining traditional econometric methods with neural networks, and sometimes neural networks by themselves have significantly delivered performance that was better than conventional econometric methods, because of their deep architectures which give them the capability to learn patterns where conventional methods may fail (Roh, 2007). Research has covered stock price forecasting and stock price index volatility forecasting. There is an opportunity to further extend neural networks to forecast the VIX, which is the index that is mostly used by practitioners for options trading, portfolio management, risk management etc. (Shaikh and Pasha, 2015).

In this study, we will explore how deep learning neural network architectures will perform when forecasting the VIX. We will forecast the VIX with MLP and LSTM models, that will include GARCH fitted values as inputs. Given that we will be making use of GARCH fitted values, which will be derived from the GARCH models, these models could be considered as hybrid models, namely MLP-GARCH and LSTM-GARCH. The hybrid model ANN-GARCH will closely follow the literature by Kristjanpoller and Minutolo (2015), where they used ANN-GARCH to model Gold Price Volatility. Whilst we consider the core data in our model to be VIX daily changes and GARCH volatility, we consider other financial and macroeconomic data that has been used to try explain variation in VIX with traditional time series models. Dai et.al (2020) use a combination approach to forecast stock return volatility, they find that the combination of stock market, brent and crude oil volatility helps predict stock return volatility. With stock market volatility and crude oil volatility increasing the predictive power. The authors also added macroeconomic information such as commercial paper rates, treasury spread, default return spread, growth in industrial production, industrial production volatility, inflation volatility, the US housing statistics and the market factor of Fama French three factor models. The study found that adding macroeconomic information increased the predictive power of the regression model when looking at out of sample forecasts. We expect similar behavior when building the Neural Network Models.

The forecasting ability of neural networks and GARCH has been covered in numerous studies. Various authors have investigated the modelling power of neural networks when it comes to stock market price prediction and volatility modelling. Some researchers have gone on to build hybrid time series and neural network models. This study seeks to add to the literature on using neural networks and GARCH for forecasting in the financial market. In particular, it focuses on forecasting the VIX and investigating whether the performance of the models changes across different volatility regimes. Our study also analyses the impact a neural network structure could have on the performance of the model.

Chapter 3

Data and Methodology

3.1 Data

The data used in this study as captured from Bloomberg. We get daily data for the S&P 500 index levels, WTI crude oil price, S&P 500 futures, Gold price, MSCI EAFE index levels and 10-year US Government Bond Yield. We look at data from 01 January 2007 to 21 February 2020. The study makes use of closing prices and the returns/daily changes as inputs in the models. The logged returns presented in Table 3.1 are given by $ln(\frac{P_t}{P_{t-1}})$ and daily changes by $\frac{P_t - P_{t-1}}{Pt-1}$.

Table 3.1: Explanatory Variables- Daily data used in models

Variable	Category	Inputs in Model(s)
S&P 500 Index	Closing Price	Logged Returns and Closing Price
WTI Crude Oil	Closing Price	Logged Returns and Closing Price
S&P 500 futures	Closing Price	Daily Change in Price and Closing Price
Gold	Closing Price	Logged Returns and Closing Price
MSCI EAFE	Closing Price	Logged Returns and Closing Price
10-year US Government Bond Yield	Closing Rate	Daily Change in Rate and Closing Rate
VIX	VIX closing level	Daily Change in levels and Closing level

The macroeconomic factors above were chosen in line with Ahoniemi (2007). The aforementioned paper investigates how the financial and economic indicator data of the factors outlined above could be used to construct a number of explanatory variables for VIX time series models. In particular, the S&P 500 index is found to be a statistically significant explanatory variable in longer sample periods. The VIX often moves in the opposite direction as the S&P 500 index. The explanatory variables have varying risk and return characteristics. In the event of a black swan event we see that there is significant movement in the variables outlined above. The VIX is often seen as a "fear gauge" and Gold is seen as a safe haven in times of uncertainty. We opted for aforementioned variables because there has been evidence of significant linkages between VIX, US Treasury bond yields, Commodity Prices and Developed Market Indices (Vychytilova, 2015).

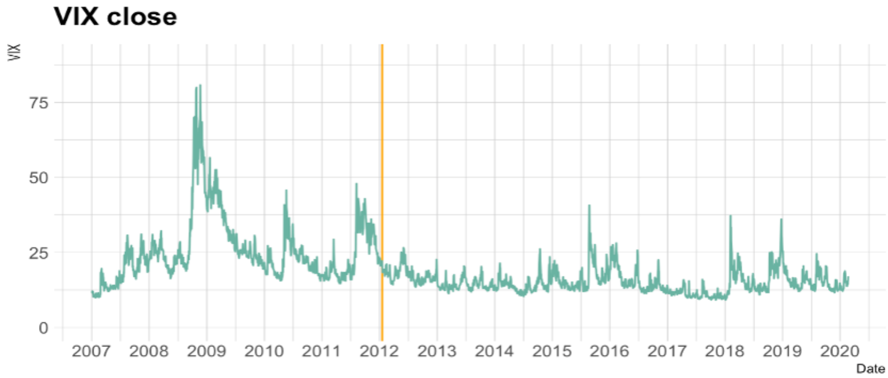
We use lagged versions of the explanatory variables presented in Table 3.1 and GARCH fitted values, to be presented in Chapter 3.2, as inputs for our models. Lagged variables allow for varying amounts of recent history to be fitted into our models. The lagged variables may also allow us to capture the cyclical trends that exist in financial markets. In addition, the dependent variable's present value may depend on it's own and other independent variables' past values. We will further elaborate on how the lagged variables are chosen and included in the models in Chapter 3.2.2 which covers the methodology of the Neural Network models.

The data is segmented into separate time periods in order to assess the performance of the model across different volatility regimes. In particular, we form periods of medium to high volatility regime as well as a low to medium volatility regime. We shall refer our medium to high volatility regime as the Crisis Period and low to medium volatility regime as the Calm Period.

In order to obtain the two different sub-periods (Calm and Crisis Period) we visually inspect the VIX close time series data. In addition, we attempt to detect structural break by implementing a dynamic programming algorithm, which uses the Bellman principle, as implemented in R statistical software, to identify structural breaks empirically (Zeileis et.al, 2003). The Bellman principle is based on the study by Bai and Perron (1998), who in their seminal paper came up with a way to endogenously detect multiple structural breaks in time series.

Following the methods above, we obtain a breakpoint date of 18 January 2012. Figure 3.1 presents a graphical presentation of our breakpoint date in our data. Data in the first sub-period runs from 01 January 2007 to 18 January 2012, this period includes data from the subprime mortgage crisis, which ran from 2007-2010. We consider this Period our medium to high volatility regime, the Crisis period. The rest of the data runs from 19 January 2012 to 21 February 2020, is considered our low to medium volatility regime or the Calm period. The Crisis and Calm periods will each have their data split into in-sample and and out-of-sample data (training, validation and test sets). Our descriptive statistics in Chapter 4.1 concur with the Calm and Crisis periods. The standard deviation of the VIX close in the Calm and Crisis period are 3.76 and 11.10, respectively.

Figure 3.1: VIX close and break point at 2012-01-1



3.2 Methodology

The objective of this study is to empirically investigate whether we can predict the VIX using VIX lagged returns, GARCH volatility and other financial and economic data. The study will analyse the VIX post 2007. Most recent data has shown to have more predictive power so we model the 1 day ahead and 5 days ahead VIX. We use the following lagged values as inputs into our neural networks: GARCH volatility, VIX returns, S&P 500 index closing price, S&P 500 index returns, WTI crude oil price, WTI crude oil returns, S&P 500 futures closing price, S&P 500 futures closing price returns, Gold price, Gold Returns, MSCI EAFE index, MSCI EAFE index returns, 10 year US Government Bond Yield and changes the in the 10 year US Government Bond Yield.

3.2.1 GARCH

GARCH models have become really popular for application to economic and financial data in the past years. Such models have been found to be extremely useful for modelling and forecasting movements in volatilities. Two well-known examples are; pricing financial options, or in the context of risk management. Recent studies have experimented with combining GARCH methods with neural networks (Brooks, Burke and Persand, 2001).

Kristjanpoller and Minutolo (2015) were able to show that when working with ANN-GARCH models, neural networks were able to learn GARCH forecasting errors for Gold Price volatility. To model the GARCH forecasting error they introduced independent variables to the multi-layer perceptron namely; the daily variations of Euro/Dollar and Dollar/Yen exchange rates, the stock market index returns of the Dow Jones Industrial (DJI) and the Financial Time Stock Exchange (FTSE), as well as the daily price variation of oil.

According to the study by Ahoniemi (2007) the GARCH (1,1) specification is able to improve both the point forecasts and direction prediction of the VIX, hence we implement GARCH(1,1) specification in our analysis. We build GARCH(1,1) models for each sub-period separately so as to capture the models applicable for each regime. Roh (2006) mentions that GARCH(1,1) model brings about a similar effect like using the long time lag ARCH model even if it uses a small number of parameters. Therefore, it is desirable to use the GARCH(1,1) model to make a time series which has the characteristics of volatility, clustering and fat tail from the perspective of conditional variances. Based on our results in the following section, Chapter 4, Table 4.1, we see that our S&P 500 time series is characterised by volatility clustering and fat tails, therefore, a GARCH(1,1) model would be suitable.

The GARCH(1,1) models can be written under the following empirical measure as in Lu et.al (2016).

$$ln(\frac{S_t}{S_{t-1}}) = \mu + \varepsilon_t^2, \tag{3.1}$$

$$v_t = \omega + \beta v_{t-1} + u_{t-1}, \tag{3.2}$$

where S_t is the "stock" price at time t, v_t is the conditional variance estimated by GARCH at time t, $\varepsilon = \sqrt{v_t} z_t$, z_t is a standard normal variable or empirical random variable with a mean of zero and a variance of 1. For GARCH(1,1), $u_t = \alpha\varepsilon_t^2$ (Bollerslev, 1986), GJR $u_t = (\alpha + \gamma I\{z_t < 0\})\varepsilon_t^2$

To build the hybrid GARCH and Neural Network model, we extend the different neural network architecture types with results from equation (3.2). We include the GARCH(1,1) fitted values as input values so that the MLP and LSTM can learn the patterns of conditional volatility.

3.2.2 Neural Networks

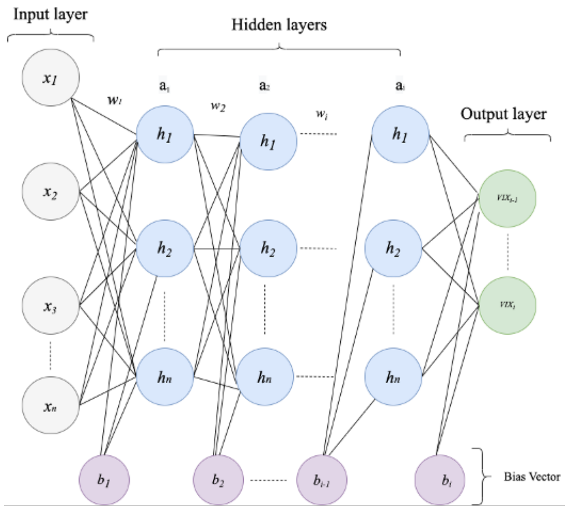
Multi-Layer Perceptron (MLP)

The MLP is a feed-forward neural network that creates a latent space where appropriate outputs are mapped from a set of inputs. It has multiple layers where each layer is fully connected to the next layer. An MLP consists of three types of layers: an input layer, an output layer, and one or more hidden layers. If neural network architecture has more than one hidden layer, the neural network can be considered as a deep learning neural network (Yang, 2019). The representation of the MLP is given by:

$$VIX_t = (\psi_i(W_i \cdot x_i + b_i))_t, \tag{3.3}$$

where i represents the layer, ψ_i represents the element wise function/activation function, b_i represents the bias, W_i represents the hidden-to-hidden weight matrix and x_i includes the GARCH(1,1) fitted values from equation (3.2). A simplified diagram of an MLP is presented in Figure 3.2.

Figure 3.2: MLP architecture



In the diagram presented in Figure 3.2 we can see that information in an MLP flows forward and never backwards this is in contrast with an RNN which allows information to flow back.

Recurrent Neural Networks (RNN)

RNNs allow information to flow back into previous parts of the network thus each model in the layers depends on past events, which allows for information to persist.

A disadvantage of RNNs is the long-term dependency problem where information gets lost over time because of the exploding and vanishing gradient problem. Neural Networks such as LSTM solve the long-term dependency problem. LSTMs are a special type of RNN that are capable of learning long term dependencies and finding patterns across time such that future forecasts make sense. The presentation for RNNs and LSTMs is found in the next section. In the following section we follow the methodology of Qiu, Wang and Zhou (2020) and Ghosh et al. (2019). The representation of RNNs is given by:

$$f_i = (h_{(i-1)}) = \psi(W \cdot h_{(i-1)} + U \cdot (x_i + b)). \tag{3.4}$$

In the conventional RNN model we evolve a hidden state h according to a hidden-to-hidden weight matrix W and an input to hidden weight matrix U. The final hidden states h_t are derived from the layer-wise function, equation (3.4). The prediction function g is then parameterized by matrix-vector multiplication for any h. Finally, V in equation (3.5) is the hidden-to-output matrix and c is the output bias vector.

$$g(h) = V \cdot h + c, \tag{3.5}$$

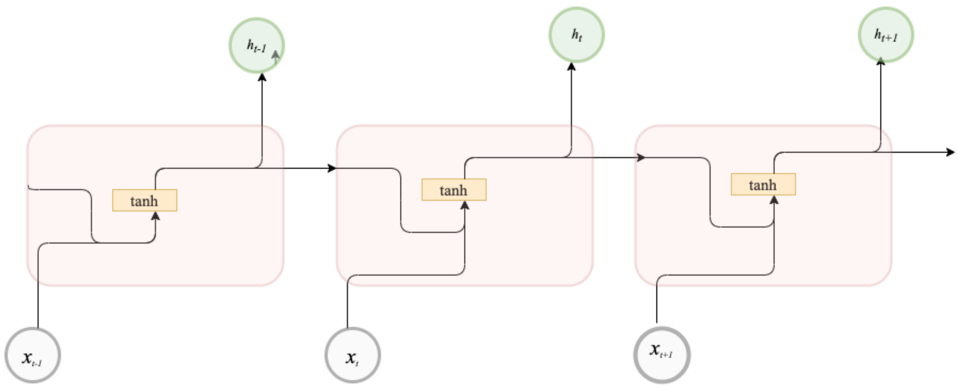
$$VIX_t = (g(h))_t. \tag{3.6}$$

If we were to use the conventional RNN to predict the VIX the output of prediction function g, equation (3.5) at time t, would give us the VIX forecast.

RNNs form of a chain of repeating units in the neural network. The chain allows for information to flow back to the cell state, this can be seen in Figure 3.3 (Ghosh et al., 2019). The illustration in Figure 3.3 shows how the modules output hidden state h. The hidden state h is then carried into other modules.

In conventional RNNs, this type of architecture can have a simple structure like a single $tanh$ layer as shown in Figure 3.3. In any RNN structure, there is input and output gates. In contrast, the LSTM has input, forgotten and output gates. This allows the LSTM to deal with the long term dependency problem by filtering information through the gate structure. Consequently, the state of memory cells are maintained and updated, through the forgotten gate.

Figure 3.3: Standard RNN architecture with single layer



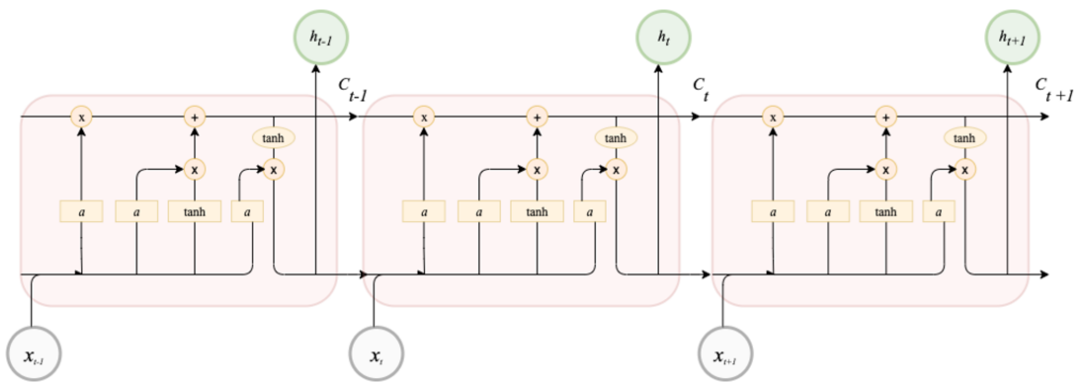
The cell state information that can stay in the LSTM module is determined by the forgotten gate. The main function if the forgotten gate is to determine how much of the previous cell state $C_{(t-1)}$ is reserved in the cell state of the current time C_t. The input gate, on the other hand, determines how much of the current input x_t is reserved into the current cell state C_t. This serves to prevent insignificant content from entering memory cells (Qiu, Wang and Zhou, 2020).

The input gate has two main functions. The first one is to determine the state of the cell that needs to be updated. The value that is to be updated is selected by the sigmoid layer (σ) that is presented in equation (3.7). The sigmoid layers are presented as a in Figure 3.4. The second main function is to update the information that needs to be updated to the cell state. The $tanh$ layer controls how much new information can be added. The adding and updating of information is given in equation (3.8) and (3.9) respectively, where f_t represents the forgotten gate at time t.

$$i_t = \sigma(W_t \cdot [h_{(t-1)}, x_t] + b_i), \quad (3.7)$$
$$\hat{C}_t = tanh(W_c \cdot [h_{(t-1)}, x_t] + b_c), \quad (3.8)$$
$$C_t = f_t \times C_{(t-1)} + i_t \times \hat{C}_t, \quad (3.9)$$
$$O_t = \sigma(W_\sigma \cdot [h_{(t-1)}, x_t] + b_o), \quad (3.10)$$
$$h_t = O_t \times tanh(C_t). \quad (3.11)$$

The final output portion as obtained from the output gate O_t is presented by equation (3.10), and the final output value of the cell is defined by equation (3.11).

Figure 3.4: LSTM architecture containing four interacting layers



Financial and economic data is also included in equation (3.3) and (3.7) as part of the input values given by x.

In the first instance we use all available data in each sub-period to train the model. GARCH fitted values are extracted from the GARCH model and are used in conjunction with the explanatory variables presented in Table 3.1 as features. Our out-of-sample dataset consists 21 days, as that is how many trading days there would typically be in a month. To increase the granularity and strength of our model we lag our explanatory variables. It is necessary to lag the explanatory variables as predictions at time t are built based on the knowledge of the explanatory variables at time $t-1$ or further back in time. We experiment with varying lag-lengths in our models. We opt for the 10 and 66 day lag-length as these result in a better out-of-sample performance. Moreover, with 10 days (two weeks) feature lags and 66 days (quarter) feature lags one can interpolate the one month performance of the models, and one week or less would be too short for our volatility modelling purposes.

A lagged data frame is created for each feature in the model. That is, to forecast 1 day ahead VIX value, we look at the value of each feature for a lag of 10 days and a lag of 66 days, e.g., for a 10 day lag VIX_t = feature1$_{(t-1)}$ + feature1$_{(t-2)}$...feature1$_{(t-10)}$ + feature2$_{(t-1)}$ + feature2$_{(t-2)}$..+feature2$_{(t-10)}$ +... feature$n_{(t-10)}$ and similarly for a 66 day lag VIX_t = feature1$_{(t-1)}$ + feature1$_{(t-2)}$ + ... + feature1$_{(t-66)}$ + feature2$_{(t-1)}$ + feature2$_{(t-2)}$ + ..+ feature2$_{(t-66)}$ +... feature$n_{(t-66)}$, where the features include GARCH fitted values and input variables from the financial and economic data. When we forecast 5 days ahead, the specification looks similar, except we look at the feature$_{(t-5)}$ first, which is the value of the feature 5 days before our forecast date.

In order to come up with a better comparative analysis of the two hybrid models we further segment the two sub-periods. Within the two sub-periods we model and forecast out of sample VIX for 21 days, for 3 windows. Thus for both sub-periods we will have 3 forecast windows each, summing up to six windows (Calm Period end dates: 2020/02/21, 2018/01/11 and 2016/05/10; Crisis Period end dates 2012/01/18, 2011/03/11 and 2011/04/12).

For consistency when comparing the models, we use the same number of observations, because of data scarcity and strict LSTM array shapes we target 920 observations for each model. In each instance we build a separate model with 920 so as to have a robust analysis on how the hybrid models perform in the two volatility regimes. Looking at the LSTM we find that because of shape specifications we can only forecast 20 days ahead, but with MLP we can forecast 21 days as there are no limitations with respect to the shape of the data when considering the MLP. We do not expect there to be any bias with regards to assessing the performance as each forecast is independent of any other forecasts. After taking out the 21 out of sample days, the remaining data is used for training and validation, the split is 80/20. We present results on our out of sample days, which is the test set.

Hyperparameter tuning of neural networks

Neural networks can consist of three kinds of layers namely an input layer, one or more hidden layers and an output layer. If the neural network has more than one hidden layer it could be considered a deep learning neural network.

The multi-layer perceptron has many hyperparameters one can tune, but we concentrate on the following:

1. The number of hidden layers will define the depth of the algorithm and, consequently, how complex computations can the MLP model process.

2. The number of neurons per layer. The number of neurons per layer define the width of the network and the latent space.

Several research papers in the machine learning community concur the Universal Approximation theorem, which states that a neural network with a sufficiently large shallow (two-layer) architecture can approximate any wide set of continuous functions to any desired non-zero level of error (Gori, 2018; Lu and Lu, 2020; Scarselli and Chung Tsoi, 1998). This theory is tested in our study, one challenge is knowing how many neurons and layers are needed for an optimal model. We do a grid search to see how many neurons and hidden layers are needed to approximate the various models. Similar to the MLP, for the LSTM model we analyse how many LSTM layers and the number of neurons it takes to reach the optimal model.

3.2.3 Measuring Forecast Accuracy

To assess the forecasting power of the various models we compare the Mean Absolute Errors (MAE) and the hit ratios.

Absolute error refers to the magnitude of difference between the predicted value and the actual or true value of an observation. MAE takes the average of absolute errors for a specified group of values. It is regularly used as a loss function for regression models and can be used in machine learning algorithms to solve optimisation problems (C3 AI, 2022). The MAE is given by:

$$MAE = \frac{\sum_{t=1}^{n} |VIX_t - VIXclose_t|}{n}, \tag{3.12}$$

where VIX_t is the predicted value and $VIXclose_t$ is the true value.

The hit ratio is determined by the number of correct signs (correct VIX level direction) divided by the total number of predictions. The overall hit ratio is the ratio of correct predictions to all predictions. The hit ratio is given as:

$$Hit\,Ratio = \frac{Number\,of\,correct\,signs}{Total\,number\,of\,predictions} \times 100. \tag{3.13}$$

MAE assesses the accuracy of the point forecast of the VIX and the hit ratio assesses the directional accuracy, that is, whether the VIX is forecasted to move down/up. We build our models to forecast the level of the VIX and not the movement, and so when looking for the overall best performing model the MAE takes precedent over the hit ratio.

The methodology and data detailed in this chapter allow us to carry out the intended research. In the following chapter, we shall present the results and findings from our empirical investigations.

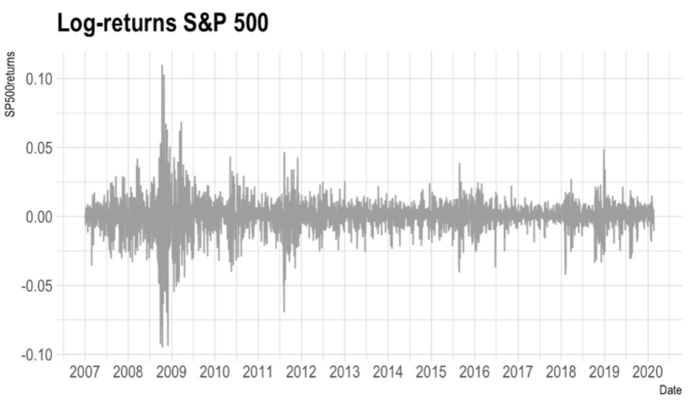
Chapter 4

Empirical Results

4.1 Descriptive Statistics

A summary of our data follows. Table 4.1 presents the descriptive statistics for our input variables. Looking at the sample mean of the VIX close in Table 4.1, we see that the mean of the VIX close in the Crisis period (25.47) is higher than the mean during the Calm period (15.17). Moreover, the sample standard deviation of the VIX close in the Crisis period (11.10) is higher than the one during the Calm period (3.76). Variables such as MSCI EAFE close, S&P 500 futures close and S&P 500 close behave as expected (in line with empirical evidence), with higher sample means in the Calm period and lower minimums in the Crisis Period. We also see Gold closing price as more volatile in the Crisis period, with a higher standard deviation and range than the Calm period.

Figure 4.1: Log-returns S&P 500



Before we proceed to build the GARCH(1,1) model, we analyse the descriptive statistics of the S&P 500 returns, which is the series we will be using when building the GARCH model. The results for kurtosis, skewness and the Jacque-Bera test statistic indicate that the daily S&P 500 returns do not follow a normal distribution. There is also excess kurtosis for the S&P 500 returns and the kurtosis coefficients are greater than 3 in both sub-periods which suggests that the data follows a leptokurtic distribution for both sub-periods. The Jacque-Bera test statistic rejects the null hypothesis of normal distribution in both sub-periods. The summary statistics also show that the S&P 500 return series is negatively skewed. Figure 4.1 represents the daily log returns if the S&P 500.

Table 4.1: Descriptive Statistics of dataset

	Crisis Period, medium to high volatility (01 January 2007 to 18 January 2012)								
	mean	**sd**	**median**	**min**	**max**	**range**	**skew**	**kurtosis**	**se**
VIX close	25.47	11.1	22.9	9.89	80.86	70.97	1.82	4.12	0.31
VIX changes	0	0.08	-0.01	-0.35	0.5	0.85	0.78	4.13	0
MSCI EAFE close	1694.4	360.82	1622.13	911.39	2388.74	1477.35	0.2	-0.98	10.21
MSCI EAFE returns	0	0.02	0	-0.09	0.08	0.17	-0.21	4.97	0
S&P 500 futures close	1211.93	205.95	1229.88	676	1576.25	900.25	-0.32	-0.68	5.83
S&P 500 futures returns	0	0.02	0	-0.1	0.13	0.24	-0.04	9.01	0
S&P 500 close	1212.13	203.65	1230.57	676.53	1565.15	888.62	-0.35	-0.67	5.76
S&P 500 returns	0	0.02	0	-0.09	0.11	0.2	-0.25	6.83	0
US 10 year Bond close	3.5	0.77	3.52	1.72	5.3	3.58	-0.02	-0.42	0.02
US 10 year Bond changes	0	0.02	0	-0.17	0.1	0.28	-0.35	3.48	0
WTI Crude Oil close	81.97	21.1	80.47	33.87	145.29	111.42	0.33	0.2	0.6
WTI Crude Oil returns	0	0.03	0	-0.13	0.16	0.29	0.1	4.38	0
Gold Close	1065.34	315.66	954.94	607.4	1882.96	1275.56	0.62	-0.57	8.94
Gold Returns	0	0.01	0	-0.07	0.1	0.17	-0.05	4.68	0
	Calm Period, low to medium volatility (19 January 2012 to 21 February 2020)								
	mean	**sd**	**median**	**min**	**max**	**range**	**skew**	**kurtosis**	**se**
VIX close	15.17	3.76	14.3	9.14	40.74	31.6	1.49	3.79	0.08
VIX changes	0	0.08	0	-0.3	0.77	1.07	1.14	7.39	0
MSCI EAFE close	1796.62	167.58	1826.15	1308.01	2186.65	878.64	-0.49	-0.35	3.72
MSCI EAFE returns	0	0.01	0	-0.07	0.04	0.11	-0.66	5.94	0
S&P 500 futures close	2175.23	519.94	2090	1273	3387.25	2114.25	0.19	-0.9	11.55
S& P500 futures returns	0	0.01	0	-0.06	0.05	0.11	-0.65	5.11	0
S&P 500 close	2177.63	518.03	2093.25	1278.05	3386.15	2108.1	0.18	-0.89	11.51
S&P 500 returns	0	0.01	0	-0.04	0.05	0.09	-0.45	3.22	0
US 10 year Bond close	2.24	0.44	2.25	1.36	3.24	1.88	0.12	-0.97	0.01
US 10 year Bond changes	0	0.02	0	-0.11	0.1	0.22	0.14	1.48	0
WTI Crude Oil close	68.48	22.29	60.2	26.21	110.53	84.32	0.38	-1.31	0.5
WTI Crude Oil returns	0	0.02	0	-0.11	0.14	0.24	0.14	3.82	0
Gold Close	1337.84	167.2	1289.76	1051.1	1790.15	739.05	0.99	0.11	3.72
Gold Returns	0	0.01	0	-0.1	0.05	0.14	-0.68	8.67	0

To determine whether a GARCH model would be suitable for the data we conduct further statistical tests on the segmented data. Presented in Table 4.2 are results from the Augmented Dickey-Fuller (ADF), Phillips-Perron Unit Root (PP), Jacque-Bera and ARCH tests. These are to test for the underlying assumptions of the GARCH model. The unit root tests ADF and PP, test for the stationarity of the dataset and the ARCH LM-test tests for the ARCH effect which determines whether or not a GARCH model is suitable (Jebran, Chen, Ullah and Mirza, 2017). We run an ARCH LM-test to test for ARCH effects and from the results we find that the test statistic is significant for both the sub-periods at the 1% level. The ADF and PP tests also give us significant test statistics indicating our data is stationary at level for both sub periods. The statistical tests indicate that the GARCH statistical conditions have been met, hence we can proceed to build a GARCH model.

Following the GARCH model building we extract the GARCH fitted values and use those as inputs in our neural networks. Inputs into our neural networks the following lagged values: GARCH volatility, VIX returns, S&P 500 index closing price, S&P 500 index returns, WTI crude oil price, WTI crude oil returns, S&P 500 futures closing price, S&P 500 futures closing price returns, Gold price, Gold Returns, MSCI EAFE index, MSCI EAFE index returns, 10-year US Government Bond Yield and changes the in the 10 year US Government Bond Yield.

Table 4.2: Statistical Tests for S&P 500 Log-Returns

Statistical Tests S&P 500 Log-Returns	Crisis Period	Calm Period
ARCH LM-test	204,58***	166,69***
Augmented Dickey-Fuller Test	10,769***	-13,225***
Phillips-Perron Unit Root Test	-1353,2***	-1918,9***
Jarque Bera Test	2448,5***	946,9***
Box-Pierce Test	62,061***	98,221***

Notes on Tables: The asterisks ***, **, * represent significance at 1, 5 and 10% respectively.

Once we have determined that we can fit an ARCH model with the statistical tests performed above we proceed to fit a GARCH(1,1) model. Given that the model does not follow a normal distribution we fit the model using a Student-t distribution. The Student-t distribution, may be of greater fit because of the high peaks and fat tails, as presented in the Descriptive Statistics section. Table 4.3 presents the results from the GARCH(1,1) fit. Results indicate that the coefficients μ, ω, α and β are statistically significant in both sub-periods. When we analyse the results we find that as we look into the Akaike Information Criterion (AIC) we find that the GARCH(1,1) model might be more suited for the Calm period where we get an AIC figure -7.07 lower than the Crisis period -5.89. The Calm Period also has a higher Log Likelihood of 7159 compared to the Crisis Period where the Log Likelihood 3682.

Table 4.3: GARCH results

Error Analysis Crisis Period				
	Estimate	Std. Error	t value	Pr(>|t|)
μ	1.020e-03	2.722e-04	3.746	0.00018 ***
ω	2.166e-06	1.025e-06	2.113	0.03458 *
α	1.314e-01	2.275e-02	5.773	7.78e-09 ***
β	8.776e-01	1.747e-02	50.235	< 2e-16 ***
	Log Likelihood			
	3681.663 normalized: 2.95005			
	Information Criterion Statistics:			
	AIC	BIC	SIC	HQIC
	-5.892087	-5.871537	-5.892119	-5.884361
	Error Analysis Calm Period:			
	Estimate	Std. Error	t value	Pr(>|t|)
μ	8.349e-04	1.280e-04	6.524	6.86e-11 ***
ω	3.396e-06	8.063e-07	4.212	2.53e-05 ***
α	1.948e-01	2.961e-02	6.580	4.71e-11 ***
β	7.728e-01	2.956e-02	26.145	< 2e-16 ***
	Log Likelihood:			
	7159.222 normalized: 3.537165			
	Information Criterion Statistics:			
	AIC	BIC	SIC	HQIC
	-7.069389	-7.055523	-7.069401	-7.064301
Signif. codes: 0 *** 0.001 ** 0.01 * 0.05 . 0.1 1				

Following the GARCH model building we extract the GARCH fitted values and use those as inputs in our neural networks. Inputs into our neural networks the following lagged values: GARCH volatility, VIX returns, S&P 500 index closing price, S&P 500 index returns, WTI crude oil price, WTI crude oil returns, S&P 500 futures closing price, S&P 500 futures closing price returns, Gold price, Gold Returns, MSCI EAFE index, MSCI EAFE index returns, 10-year US Government Bond Yield and changes the in the 10 year US Government Bond Yield.

4.2 Neural Network Results

Neural Network models can overfit the data when optimising for performance. We account for overfitting in our model building by setting aside an out-of-sample dataset (test set) and splitting our in-sample dataset between the training set (80%) and validation set (20%). In addition, we use an L1 regularisation technique, which is also referred to as Lasso Regression. Lasso Regression adds penalty terms when optimising the models for performance, it shrinks the less important features' coefficients to zero. This means it can remove features with no predictive content and thus aiding with feature selection.

The neural network modelling results from the out-of-sample data are presented as follows, first we assess the forecasting power of the neural networks given all the data points. This allows us to focus on how well the model is able to predict the VIX given all the available data points within the scope of this study, i.e., post 2007, we still segment our data between the two sub-periods and build those models separately, so we are able to track the performance of the models across the two regimes. Following this assessment, we proceed to analyse the architectures and performances of the top performing models. The objective of the first assessment is to meet the first aim of the study, which is to empirically investigate the performance of the neural network in forecasting the VIX. The second aim is to investigate whether the performance of the neural network varies across the different regimes, particularly across the Crisis and Calm Period. Subsequently, as mentioned in the Methodology section, we assess the performance of the models given the three forecast windows in each sub-period. We expect the performance of the results based on MAE to deteriorate given we are working with less observations in these forecast windows. Despite the deterioration of the results, we expect to better assess how the performance varies across the two sub-periods.

Based on the first set of results we see that indeed the VIX can be forecast with neural networks and the models can achieve promising levels of accuracy. We also find that the models are capable of forecasting not just a one day ahead forecast but also a 5 day ahead forecast. We do a deviation comparison of the models, first based on MAE and then finally the hit ratio. As the objective of the research is to investigate point accuracy when forecasting the VIX, the MAE takes precedent in terms of assessing the overall top performing model. Performance results are presented in Table 4.4.

To begin, the results indicate that the MAE ranges from 1.03 to 6.48 and the hit ratio ranges from 37% to 84%. The MLP in the Calm Period with 10 day lag that forecasts 5 days ahead has the lowest MAE at 1.03 indicating that it is the top performing model, this model is followed by the MLP in the Calm Period with 66 day lag that forecast 1 day ahead with an MAE of 1.06.

The best performing models are highlighted in green.

Table 4.4: Performance results (MAE and Hit ratio)

Performance of models based on Mean Absolute Error					
	Forecast horizon	**10 day lags**	**66 day lags**	**10 day lags**	**66 day lags**
		Crisis Period ended 2012/01/18		Calm Period ended 2020/02/21	
MLP	**1**	5.18	4.44	2.15	1.06
	5	1.50	2.69	1.03	1.29
LSTM	**1**	4.99	1.77	2.61	6.44
	5	6.48	2.37	1.52	4.42
Performance of models based on Hit Ratio					
	Forecast horizon	**10 day lags**	**66 day lags**	**10 day lags**	**66 day lags**
		Crisis Period ended 2012/01/18		Calm Period ended 2020/02/21	
MLP	**1**	58%	84%	45%	60%
	5	75%	55%	45%	55%
LSTM	**1**	53%	42%	58%	47%
	5	37%	53%	63%	47%

The results for the models are presented in Figures 4.2 and 4.3, predicted_10 represents predicted values for features with 10 day lag, predicted_66 represents predicted values for features with 66 day lag and VIX close is the actual VIX value. Figures 4.2 and 4.3 are for the Calm Period ended 21/02/2020.

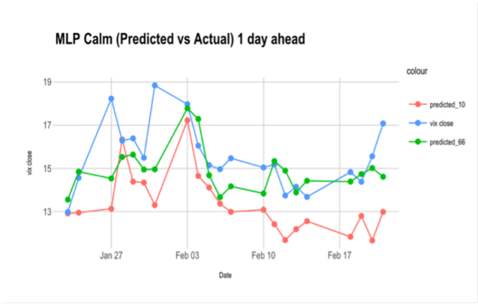


Figure 4.2: MLP Calm Period ended 2020/02/21 1 day ahead

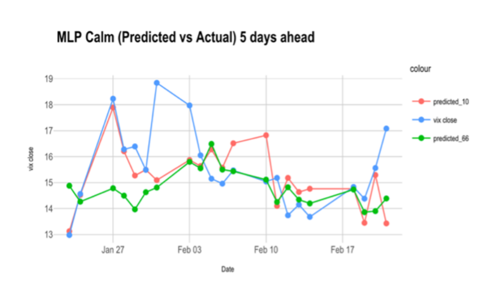


Figure 4.3: MLP Calm Period ended 2020/02/21 5 days ahead

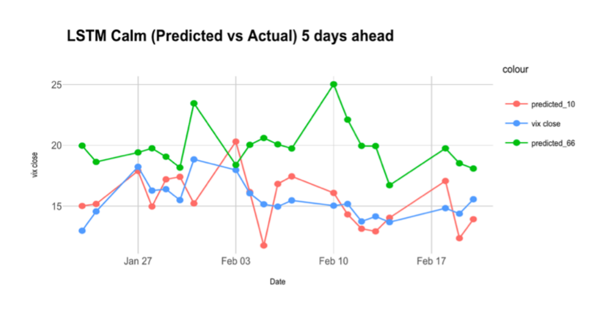
Relating to the hit ratio, which indicates the percentage of times that the model predicts the direction correctly, we see that the MLP in the Calm Period with 66 day lag that forecast 1 day ahead outperforms the MLP in the Calm Period with 10 day lag that forecasts 5 days ahead, with hit ratios of 60% and 45%, respectively. The highest hit ratio in the Calm Period is 63%, which is for the LSTM with 10 day lag forecasting 5 days ahead. The predicted values are presented in Figure 4.4. The MAE for this model is 1.52, the model has the fifth lowest MAE out of the 16 models built in this instance.

We can also draw a direct comparison between MLP and LSTM. Given that the 5-days ahead forecast produced the lowest MAE for both models (1,03 and 1,52), MLP outperformed LSTM in this regard. However, when viewing from the perspective of hit ratio, LSTM 5-day ahead forecast produced better results than the MLP instead (with 63% vs 45%).

A further direct comparison into the MLP versus the LSTM in the Calm Period follows. Referencing the MAE, we see that the MLP definitively outperforms the LSTM, where all MAE values for MLP are lower than LSTM. The hit ratio, however, yields mixed results. We see the MLP, with a 66 day lag, results in hit ratios of 60% (1 day ahead) and 50% (5 days ahead), thus outperforming the LSTM with a hit ratio of 47% for both 1 day and 5 day ahead forecast. Conversely, on a 10 day lag, we see higher hit ratios 58% (1 day ahead), 63% (5 days ahead) resulting from the LSTM, thus outperforming performance the MLP, with a hit ratio of 45% for both 1 day and 5 days ahead forecasts. Therefore, we can see that the lag period has a significant impact on the hit ratios of both models in the Calm Period. It is evident that over the shorter lag period (10 days) the LSTM outperforms the MLP in terms of the hit ratio. Whereas over the longer lag period (66 days) the converse is true (MLP hit ratio higher than LSTM).

Finally, we can compare the MLP versus the LSTM in the Crisis Period. When analysing the results based on MAE we see that the LSTM outperforms the MLP in all cases, except one, which is the 10 day lag and 5 day ahead forecast (MAE LSTM 6.48 versus MAE MLP 1.50). On the other hand, the results for hit ratio indicate that the MLP outperforms the LSTM. We can see that in each instance, the MLP has a higher hit ratio. For example, MLP 10 day lag 1 day ahead forecast has a hit ratio of 58% which is greater than the LSTM 10 day lag 1 day ahead forecast with a hit ratio of 53%. Another example is where MLP 66 day lag, 1 day ahead forecast has a hit ratio of 84% which is greater than the LSTM 66 day lag 1 day ahead with a hit ratio of 42%.

Figure 4.4: LSTM Calm Period ended 2020/02/21 5 days ahead



Following the most prior analysis, we further investigate the architectures of the best performing models (the MLP in the Calm Period with 66 day lag that forecasts 1 day ahead and the MLP in the Calm Period with 10 day lag that forecasts 5 days ahead). When building the neural network we searched across various model architectures to obtain optimal models, we varied the number of neurons and the number of hidden layers. The best model was selected based on the lowest mean residual deviance or mean squared error in the validation dataset. The various models built for the best performing models presented above are given in Table 4.5 and Table 4.6.

Table 4.5: Calm Period MLP 1 day ahead 66 day lag

Hidden Layers & Neurons	Mean Residual Deviance
(100, 100, 100, 100)	0.428
(100, 150, 200)	0.440
(200, 200, 200)	0.446
(100, 100, 100)	0.451
(100, 100, 150, 150)	0.479
(200)	0.534
(150)	0.597
(30, 40)	0.638
(80)	0.640
(20, 30, 50)	0.642
(20, 20, 20)	0.667
(30, 30)	0.670
(100)	0.675
(50)	0.682
(40)	0.828
(30)	0.840
(20, 20)	0.895
(20, 30)	0.900
(20)	1.113

Table 4.6: MLP 5 days ahead with 10 day lag respectively

Hidden Layers & Neurons	**Mean Residual Deviance**
(100, 150, 200)	0.184
(100, 100, 100)	0.186
(200, 200, 200)	0.193
(100, 100, 150, 150)	0.199
(100, 100, 100, 100)	0.205
(30, 30)	0.206
(150)	0.209
(200)	0.211
(100)	0.213
(80)	0.214
(20, 20, 20)	0.218
(20, 30)	0.219
(20, 30, 50)	0.219
(30, 40)	0.226
(50)	0.229
(20, 20)	0.230
(30)	0.237
(40)	0.238
(20)	0.256

For Table 4.6 the model is predicting 5 days ahead whilst in Table 4.5 the model is predicting 1 day ahead, additionally, in Table 4.5 the model considers 66 day lag and in Table 4.6 the model considers 10 day lag Initially, in Table 4.5, we see that increasing the number of hidden layers from three to four hidden layers decreases the mean residual deviance by 0.012, in Table 4.6 we see that building the model past three hidden layers does not result in an improvement in the mean residual deviance, which indicates that the three hidden layers could be sufficient. Models with three hidden layers can be considered to be deep learning neural networks. The range of the Mean Residual Deviance in Table 4.5 is 0.428 to 1.113, and in Table 4.6 the range is 0.184 to 0.256, the marginal improvements in the mean residual deviance is Table 4.6 across the various architectures are lower than the marginal improvements in Table 4.5. The structure of the inputs and the output affects how long it takes the model to reach the lowest RMSE/MSE, in terms of epochs. In supervised learning, the neural network passes through the input data is from the neurons in one layer to the neurons in the next layer. This process is repeated many times and one iteration is called an epoch (Georgevici and Terblanche, 2019). The best epoch is one where learning is optimised, this is where the RMSE is the lowest. The scoring histories of the above two models is captured in Figures 9 and 10.

The scoring histories of the best performing models are presented in Figures 4.5 and 4.6. We see a huge difference in the epochs that are run through to reach the optimal models. For MLP in the Calm Period with 66 day lag that forecasts 1 day ahead the best epoch is 190 and the best epoch for the MLP in the Calm Period with 10 day lag that forecasts 5 days ahead is 1810. The epochs required to reach the optimal models in all our models are presented in Table 4.7. The results show that there is not a perfect number of epochs that can satisfy the models across the various models. When analysing the results in Table 4.7, particularly the MLP, we see that as we increase the features i.e. from 10 day lag to 66 day lag, the models require significantly less epochs. This implies that the network has to run through less iterations, decreasing the amount of time it takes to reach the lowest RMSE for the given architecture.

Figure 4.5: Scoring history Calm Period MLP 1 day ahead 66 day lag best epoch 190

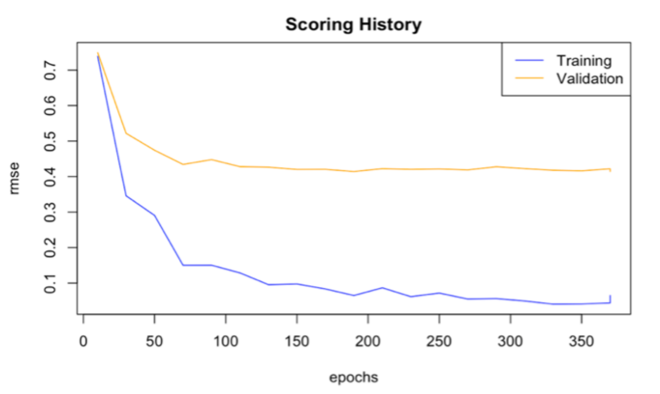


Figure 4.6: Scoring history Calm Period MLP 5 days ahead 10 day lag that best epoch 1810

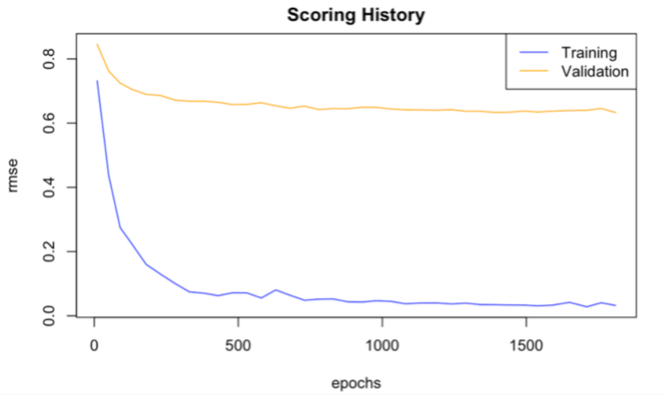


Table 4.7: Epochs to reach optimal models

Epochs to reach optimal models					
	Forecast horizon	**10 day lag**	**66 day lag**	**10 day lag**	**66 day lag**
		Crisis Period ended 2012/01/18		**Calm Period ended 2020/02/21**	
MLP	**1**	1900	990	820	190
	5	2680	110	1810	130
LSTM	**1**	210	300	100	100
	5	130	150	70	90

The models we have presented so far showcase the forecasting power of the models with respect to the VIX, one other objective of the study was to see how the performance of the model is affected in different volatility regimes, just analysing results from two periods might not be enough to conclude that the models perform better in the calm period. Accordingly, we proceed with this study by extending our modelling to more forecast windows within both sub-periods. One disadvantage of this is we have less observations to work with, as we shift our forecast windows to dates within the periods. For consistency when comparing the models we 900 observations are used to train data for each forecast window. We have six periods that we look at in total when assessing the performance in different volatility regimes (Calm Period end dates: 2020/02/21, 2018/01/11 and 2016/05/10; Crisis Period end dates 2012/01/18, 2011/03/11 and 2011/04/12). The MAE results are presented in Table 4.8. The first thing we notice is that the forecasting power of the model with respect to previously best performing model, MLP with Calm Period ended 2020/02/21, is significantly reduced, this could be because we are now just using 900 observations and previously we were working with over 2000 observations.

Figure 4.7: MLP Calm Period ended 2020/02/21 1 day ahead

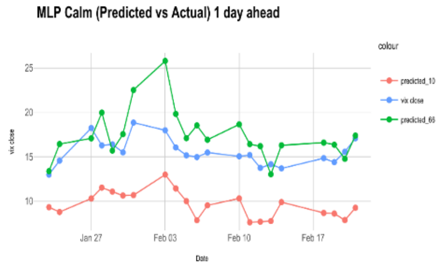


Figure 4.8: MLP Calm Period ended 2020/02/21 5 days ahead

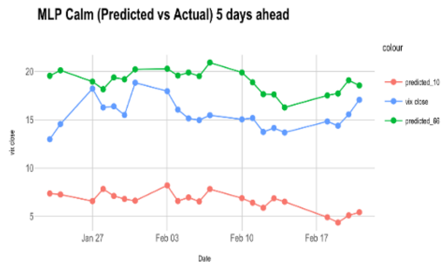


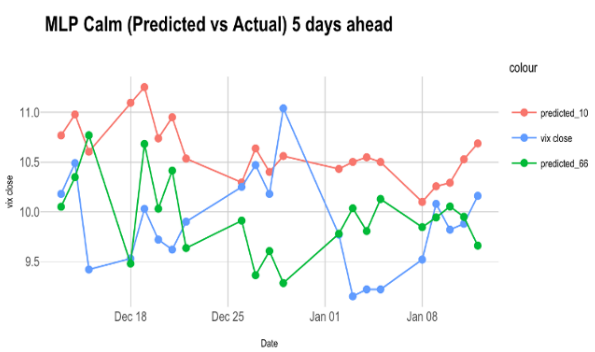
Table 4.8: Performance results 2

Performance of models based on Mean Absolute Error					
	Forecast horizon	**10 day lags**	**66 day lags**	**10 day lags**	**66 day lags**
		Crisis Period ended 2012/01/18		Calm Period ended 2020/02/21	
MLP	**1**	4.54	5.52	5.92	2.28
	5	4.10	1.98	8.96	3.48
LSTM	**1**	5.06	7.45	5.00	3.67
	5	2.33	4.83	5.58	4.38
		Crisis Period ended 2011/03/11		Calm Period ended 2018/01/11	
MLP	**1**	2.32	2.20	3.06	0.96
	5	4.13	2.01	0.76	0.53
LSTM	**1**	4.03	1.97	3.56	9.30
	5	7.81	3.88	5.96	13.69
		Crisis Period ended 2011/04/12		Calm Period ended 2016/05/10	
MLP	**1**	2.58	2.69	2.21	0.83
	5	3.54	2.87	5.55	2.67
LSTM	**1**	6.29	3.91	1.75	2.06
	5	4.25	3.57	4.39	3.99

Investigating the performance of the models based on the Mean Absolute Error, the results indicate that on average the MLP outperforms the LSTM. For the Crisis Period ended 2012/01/18, we see that in all cases the MLP outperforms the LSTM, except in the case where we are forecasting 5 days ahead with a 10 day lag. We see a similar situation in the Crisis Period ended 2011/03/11, where the MLP outperforms the LSTM in all cases, except where the models are forecasting 1 day ahead with 66 day lag. In the final Crisis Period ended 2011/04/12 we see that MLP outperforms LSTM invariably. This is similar to the Calm Period ended 2018/01/11, where the MLP also outperforms LSTM invariably. In the other Calm Periods we see that there are instances where the LSTM beats the MLP. This is in the Calm Periods ended 2016/05/10 and 2020/02/21 we see that the LSTM outperforms MLP in cases where we use 10 day lag. Specifically, the LSTM forecasting both 1 day and 5 days ahead with 10 day lag. Overall, it is evident that in most cases the MLP outperforms LSTM. In the Crisis Period MLP outperforms LSTM in 10 out of 12 cases. In the Calm Period the MLP outperforms the LSTM in 8 out of 12 cases. Hence, 75% of the time the MLP outperforms the LSTM in predicting the value of the VIX.

The top performing model is the MLP for the Calm Period ended 2018/01/11, where the model is forecasting 5 days ahead, the MAE for the 66 days lag is 0.53 and the MAE for the 10 day lag is 0.76. Results show that in this instance more features equates to an improved MAE. This assertion however does not hold when looking at the hit ratio in Table 4.9. We see in this table that the 10 day lag have a significantly higher hit ratio, it predicts the movement of the correctly 70% of the time, whilst with the 66 day lag the model only predicts the direction correctly 25% of the time.

Figure 4.9: MLP Calm Period ended 2018/01/11 5 days ahead



Moreover, when investigating the models in terms of how often they predict direction correctly we find that the MLP for the Crisis Period ended 2012/01/18 predicts the direction of the VIX correctly 80% of the time, the model is forecasting 5 days ahead with 10 day lag, although this model is able to capture the shape of the VIX, when assessing the model relative to the other models it does not capture the point forecast of the VIX well, the MAE of this model is 4.09. The plot in Figure 4.10 has predictions of this model, predictions are given by predicted_10 and the actual VIX is given by VIX close lines.

Figure 4.10: MLP Crisis Period ended 2012/01/18 1 day ahead

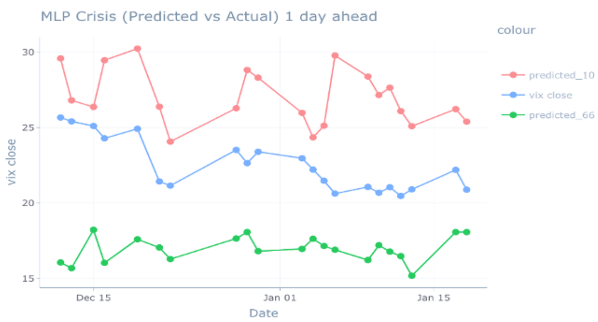


Table 4.9: Performance results 3

Performance of models based on Hit Ratio					
	2[2]*Forecast horizon	**10 day lag**	**66 day lag**	**10 day lag**	**66 day lag**
		Crisis Period ended: 2012/01/18		Calm Period ended: 2/21/20	
MLP	**1**	65%	55%	50%	45%
	5	80%	30%	55%	65%
LSTM	**1**	53%	63%	53%	53%
	5	63%	47%	47%	53%
		3/11/11		1/11/18	
MLP	**1**	40%	60%	50%	45%
	5	45%	50%	70%	25%
LSTM	**1**	47%	21%	37%	47%
	5	47%	47%	52%	48%
		4/12/11		5/10/16	
MLP	**1**	40%	50%	52%	48%
	5	65%	30%	55%	70%
LSTM	**1**	37%	68%	50%	49%
	5	42%	53%	68%	53%

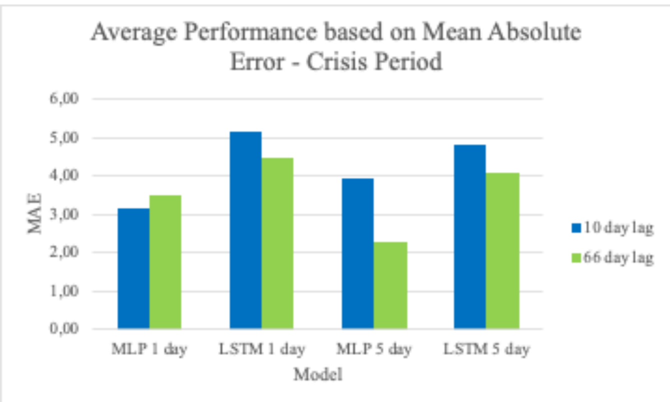
Multiple forecast windows allow for a better deviation comparison when analysing the differences in the performance across the two sub-periods that is the Crisis and Calm Period. Since we cannot directly compare two periods as per the previous analysis where we only had one forecast window per regime, we make the contrast based on average MAE for both MLP and LSTM in both the Crisis and Calm Period.

Table 4.10: Performance results 4

Average Performance based on Mean Absolute Error				
	Crisis Period		Calm Period	
MLP	**10 day lags**	**66 day lags**	**10 day lags**	**66 day lags**
1	3.15	3.47	3.73	1.36
5	3.92	2.29	5.09	2.23
LSTM	**10 day lags**	**66 day lags**	**10 day lags**	**66 day lags**
1	5.13	4.44	3.44	5.01
5	4.80	4.10	5.31	7.35

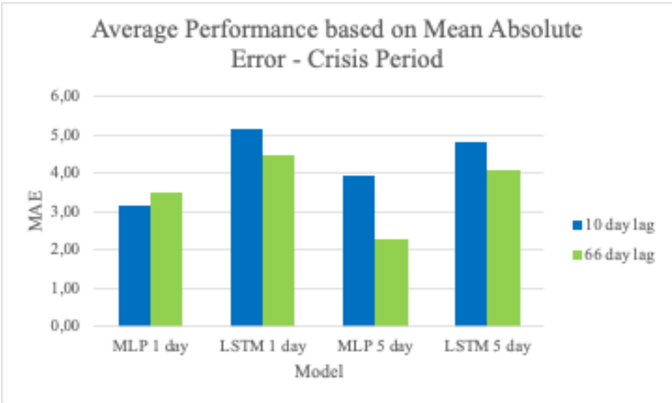
Taking into consideration the average MAE for the two periods, as presented in Table 4.10, we find that the top 3 best performing MLPs (Calm) 1 day ahead 66 day lag > MLPs (Calm) 5 days ahead 66 day lag > MLPs (Crisis) 5 days ahead Crisis 66 day lag. Analysing the overall results we see that although the top two models are in the Calm Period, there are instances where models in the Crisis Period outperform the Calm Period. For example, the MLPs in the Crisis Period forecasting 1 day ahead with 10 day lag, average MAE 3.14, outperforms the MLPs in the Calm Period forecasting 1 day ahead with 10 day lag. Another example is where the LSTMs in the Crisis Period forecasting 5 days ahead with 66 day lag, average MAE 4.10, outperformed LSTMs forecasting 5 days ahead with an average MAE 7.35. Given our analysis there is no clear dominant regime. The top three models , Table 10, are highlighted in green. We further analyse the different neural networks with Figure 4.11 and 4.12.

Figure 4.11: Average Performance based on Mean Absolute Error - Crisis Period



From Figure 4.11, we see that the MLP MAE is lower than the LSTM MAE invariably. Hence, the MLP is the dominant model in the Crisis Period in terms of its ability to forecast the value of the VIX index.

Figure 4.12: Average Performance based on Mean Absolute Error - Calm Period



From Figure 4.12, we can see that the MLP MAE is lower than the LSTM MAE in all cases, except in the case when we are forecasting 1 day ahead with a 10 day lag. Hence, the MLP is the dominant model in the Calm Period in terms of its ability to forecast the value of the VIX index.

Chapter 5

Conclusion

The VIX is seen as a forward-looking estimate of volatility. Thus asset managers, risk managers and traders rely on the VIX for their investment decisions. This study seeks to add to the existing literature that focuses on forecasting the VIX. We estimate 1 day ahead and 5 days ahead point forecasts of the VIX using machine learning techniques.

The main objective of this study is to investigate whether the VIX is forecastable with the use of neural networks. In particular, we investigate the forecasting power of both the LSTM and MLP models across varying structures. In addition, we investigate how the models behave across different volatility regimes, namely a period of low-to-medium volatility (Calm Period) and a period of medium-to-high volatility (Crisis Period). Our results indicate that the MLP Neural Network outperforms the LSTM. A number of models are built with varying number of features, with some forecasting 1 day ahead and others 5 days ahead. The best performing model given both the Mean Absolute Error and the Hit ratio is the MLP in the Calm Period. It forecasts 5 days ahead, with 10 day lag in the features, producing a low MAE and high Hit ratio overall. While the Hit ratio gives the percentage of times which the model predicts the direction of the VIX correctly, the MAE shows prediction errors which is the absolute difference between the true value and predicted value.

The average levels of error, also show that although the best performing model is found in the Calm Period, there are instances where the levels of error in the Crisis Period are lower than those in the Calm Period. Compellingly, our results show that neural network model forecasting power does not always deteriorate when there is more volatility in the financial market. This is well in line with studies on the neural networks Universal Approximation theorem, which states that a neural network with a sufficiently large shallow (two-layer) architecture can approximate any wide set of continuous functions to any desired non-zero level of error (Gori, 2018; Lu and Lu, 2020; Scarselli and Chung Tsoi, 1998).

When analysing the best performing model across the varying structures, we find that an increase in the number of hidden layers up to three hidden layers leads to an improvement in the level of error. Additionally, increasing the number of neurons has a similar effect. Interestingly, we also find that going past three hidden layers does not significantly improve the level of error. Model builders then have to decide whether the decrease in the level of error warrants the complexity of the model and the extra time required.

This study adds to the literature that covers the VIX and the forecasting power of neural networks on time series data. Specifically, we focus on deriving the point forecasts of the VIX. We find that there are varying levels of predictability in different forecast windows, and given a change in the input data, the architectures needed to reach an optimal model changes. While predicting the VIX is not an easy task, predictability of the VIX using neural networks is possible.

Trading strategies were not within the scope of this research. How the predictability of the VIX could be used for trading strategies by traders to potentially profit from this predictability is left for further research. Future research could also potentially focus on predicting the direction of the VIX using neural networks.

References

Ahoniemi, K. (2007). Modelling and Forecasting the VIX Index. *SSRN Electronic Journal*.

Alexander, C., 2011. *Practical Financial Econometrics*. Chichester: Wiley.

Audrino, F., Sigrist, F. and Ballinari, D., 2020. The impact of sentiment and attention measures on stock market volatility. *International Journal of Forecasting*, 36(2), pp.334-357.

Awartani, B. and Corradi, V. (2005). Predicting the volatility of the S&P-500 stock index via GARCH models: the role of asymmetries. *International Journal of Forecasting*, 21(1), pp.167-183.w

Bai, J. and Perron, P., 1998. Estimating and Testing Linear Models with Multiple Structural Changes. Econometrica, 66(1), p.47.

Bildirici, M. and Ersin, Ö. (2009). Improving forecasts of GARCH family models with the artificial neural networks: An application to the daily returns in Istanbul Stock Exchange. *Expert Systems with Applications*, 36(4), pp.7355-7362.

Bildirici, M. and Ersin, O. (2012). Markov Switching Artificial Neural Networks and Volatility Modelling with an Application to a Turkish Stock Index. *SSRN Electronic Journal*.

Bildirici, M. and Ersin, O. (2012). Modelling Markov Switching ARMA-GARCH Neural Networks Models and an Application to Forecasting Stock Returns. *SSRN Electronic Journal*.

Brooks, C., Burke, S. and Persand, G., 2001. Benchmarks and the accuracy of GARCH model estimation. *International Journal of Forecasting*, 17(1), pp.45-56.

Busch, T., Christensen, B. and Nielsen, M., 2011. The role of implied volatility in forecasting future realized volatility and jumps in foreign exchange, stock, and bond markets. *Journal of Econometrics*, 160(1), pp.48-57.

C3 AI. 2022. Mean Absolute Error - C3 AI. [online] Available at: <https://c3.ai/glossary/data-science/mean-absolute-error/>

Caterini, A. (2017). *A Novel Mathematical Framework for the Analysis of Neural Networks*. Masters in Mathematics. University of Waterloo.

Chung, S., Tsai, W., Wang, Y. and Weng, P., 2011. The information content of the S&P 500 index and VIX options on the dynamics of the S&P 500 index. Journal of Futures Markets, 31(12), pp.1170-1201.

Cybenko, G. (1989). Approximation by superpositions of a sigmoidal function. *Mathematics of Control, Signals, and Systems*, 2(4), pp.303-314.

Dai, Z., Zhou, H., Dong, X. and Kang, J., 2020. Forecasting Stock Market Volatility: A Combination Approach. *Discrete Dynamics in Nature and Society*, 2020, pp.1-9.

Edwards, T. and Preston, H., 2017. *A Practitioner's Guide to Reading VIX*. [online] Spglobal.com. Available at: <https://www.spglobal.com/spdji/en/education-a-practitioners-guide-to-reading-vix.pdf> [Accessed 30 January 2022].

Fernandes, M., Medeiros, M. and Scharth, M. (2014). Modelling and predicting the CBOE market volatility index. *Journal of Banking & Finance*, 40, pp.1-10.

Georgevici, A. and Terblanche, M., 2019. Neural networks and deep learning: a brief introduction. *Intensive Care Medicine*, 45(5), pp.712-714.

Ghosh, A., Maji, G., Bose, S., Debnath, N. and Sen, S., 2019. Stock Price Prediction Using LSTM on Indian Share Market. EPiC Series in Computing, Proceedings of 32nd International Conference on Computer Applications in Industry and Engineering,.

Glosten, L., Jagannathan, R. and Runkle, D., 1993. On the Relation between the Expected Value and the Volatility of the Nominal Excess Return on Stocks. *The Journal of Finance*, 48(5), pp.1779-1801.

Gori, M., 2018. Deep Architectures. Machine Learning, pp.236-338. ISBN 9780081006597, https://doi.org/10.1016/B978-0-08-100659-7.00005-1.

Hiransha, M., Gopalakrishnan, E., Vijay Krishna, M. and Soman, K. (2018). NSE Stock Market Prediction Using Deep-Learning Models. *Procedia Computer Science*, 132, pp.1351-1362.

Jebran, K., Chen, S., Ullah, I. and Mirza, S., 2017. Does volatility spillover among stock markets varies from normal to turbulent periods? Evidence from emerging markets of Asia. *The Journal of Finance and Data Science*, 3(1-4), pp.20-30.

Kim, H. and Won, C. (2018). Forecasting the volatility of stock price index: A hybrid model integrating LSTM with multiple GARCH-type models. *Expert Systems with Applications*, 103, pp.25-37.

Kochlin, K., 2020. *The VIX's Wild Ride - SIFMA*. [online] SIFMA. Available at: <https://www.sifma.org/resources/research/the-vixs-wild-ride/> [Accessed 30 January 2022].

Kristjanpoller, W. and Minutolo, M. (2015). Gold price volatility: A forecasting approach using the Artificial Neural Network–GARCH model. *Expert Systems with Applications*, 42(20), pp.7245-7251.

Liu, Q., Guo, S. and Qiao, G., 2015. VIX forecasting and variance risk premium: A new GARCH approach. *The North American Journal of Economics and Finance*, 34, pp.314-322.

Liu, Q. and Qiao, G. (2012). Variance Risk Premium and VIX Pricing: A Simple GARCH Approach. *SSRN Electronic Journal*.

Liu, S., Zhang, C. and Ma, J. (2017). CNN-LSTM Neural Network Model for Quantitative Strategy Analysis in Stock Markets. *Peking University*.

Lu, X., Que, D. and Cao, G. (2016). Volatility Forecast Based on the Hybrid Artificial Neural Network and GARCH-type Models. *Procedia Computer Science*, 91, pp.1044-1049.

Lu, Y. and Lu, J., 2020. Universal Approximation Theorem of Deep Neural Networks for Expressing Probability Distributions. In: 34th Conference on Neural Information Processing Systems (NeurIPS 2020). [online] Vancouver, Canada

Majmudar, U. and Banerjee, A. (2004). VIX Forecasting. *SSRN Electronic Journal*.
Poon, S. and Granger, C. (2005). Practical Issues in Forecasting Volatility. *CFA Digest*, 35(2), pp.70-71.

Qiu, J., Wang, B. and Zhou, C., 2020. Forecasting stock prices with long-short term memory neural network based on attention mechanism. PLOS ONE, 15(1), p.e0227222.

Roh, T., 2007. Forecasting the Volatility of Stock Price Index. Advanced Data Mining and Applications, pp.424-435.

Scarselli, F. and Chung Tsoi, A., 1998. Universal Approximation Using Feedforward Neural Networks: A Survey of Some Existing Methods, and Some New Results. Neural Networks, 11(1), pp.15-37.

Shaikh, I. and Pasha, P., 2015. The implied volatility index: Is 'investor fear gauge' or 'forward-looking'?. *Borsa Istanbul Review*, 15(1), pp.44-52.

SPRH LABS (2019). *Understanding Deep Learning: DNN, RNN, LSTM, CNN and R-CNN.* [online] Medium. Available at: https://medium.com/@sprhlabs/understanding-deep-learning-dnn-rnn-lstm-cnn-and-r-cnn-6602ed94dbff [Accessed 20 Jan. 2020].

Thavaneswaran, A., Appadoo, S. and Peiris, S. (2005). Forecasting volatility. *Statistics & Probability Letters*, 75(1), pp.1-10.

Tsaih, R., Hsu, Y. and Lai, C., 1998. Forecasting S&P 500 stock index futures with a hybrid AI system. Decision Support Systems, 23(2), pp.161-174[Accessed 20 Jan. 2020].

Vychytilova, J., 2015. Linkages among U.S. Treasury Bond Yields, Commodity Futures and Stock Market Implied Volatility: New Nonparametric Evidence. *Journal of Competitiveness*, 7(3), pp.143-158.

Wang, Y., n.d. A New Concept Using LSTM Neural Networks For Dynamic System Identification. [ebook] p.Yale University, Department of Electrical Engineering. Available at: <https://cpb-us-w2.wpmucdn.com/campuspress.yale.edu/dist/7/677/files/2016/09/paper_LSTM_2017ACC-1sz2yir.pdf> [Accessed 4 Dec. 2020].

Yang, X., 2019. Neural networks and deep learning. Introduction to Algorithms for Data Mining and Machine Learning, Academic Press, pp.139-161. [Accessed 02 Dec. 2020].

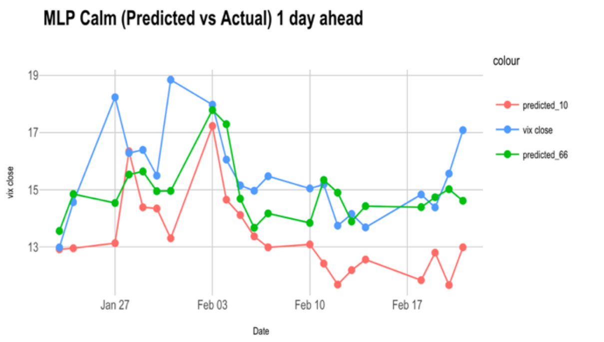
Zeileis, A., Kleiber, C., Krämer, W. and Hornik, K., 2003. Testing and dating of structural changes in practice. Computational Statistics & Data Analysis, 44(1-2), pp.109-123.

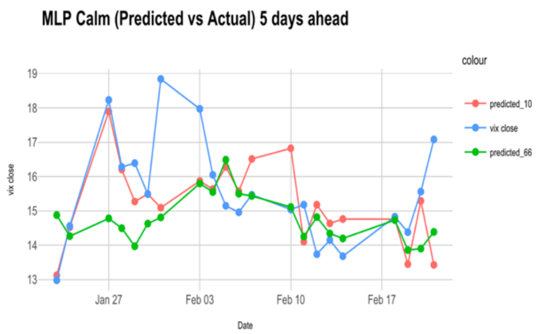
APPENDIX

Performance results 1

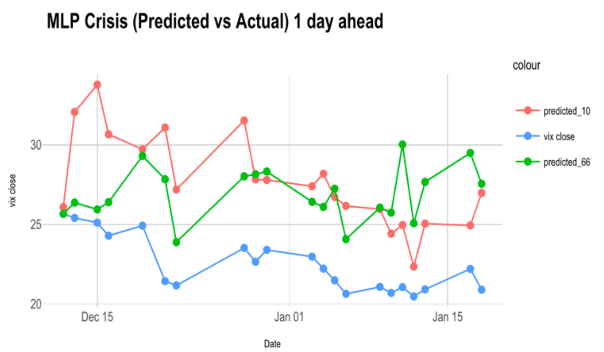
MLP

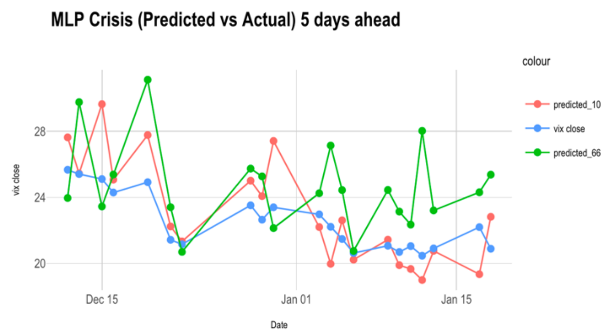
Calm Period ended 2020/02/21





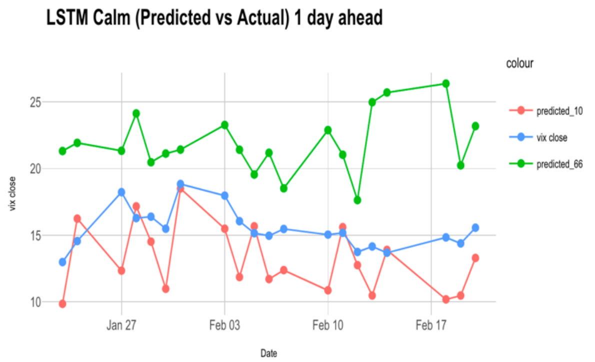
Crisis Period ended 2012/01/18

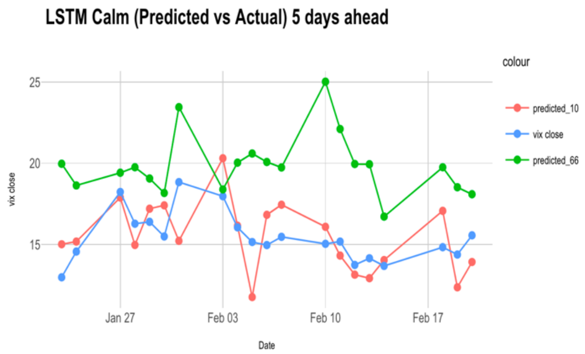




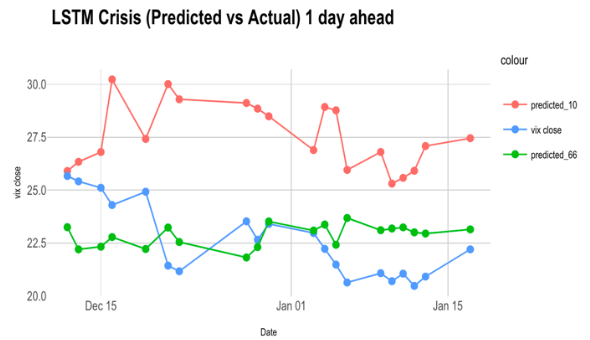
LSTM

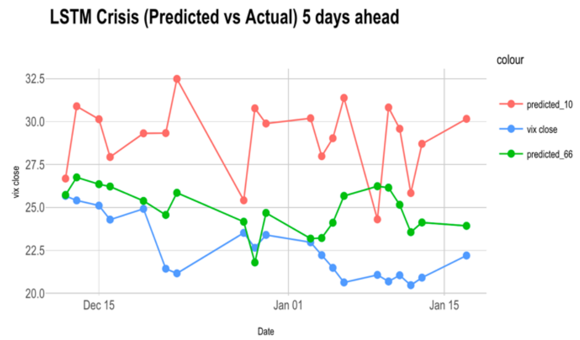
Calm Period ended 2020/02/21





Crisis Period ended 2012/01/18

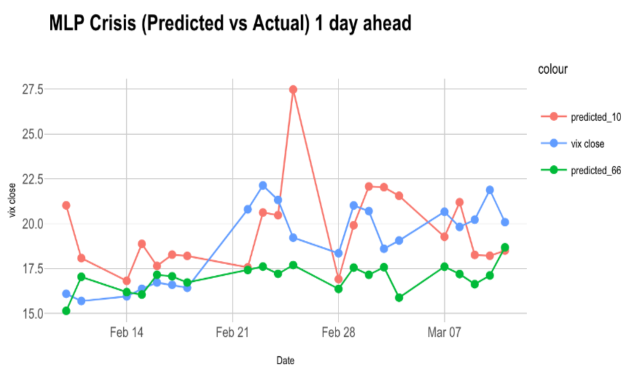


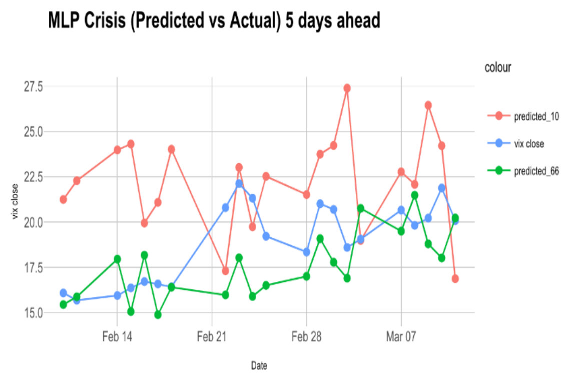


Performance results 2

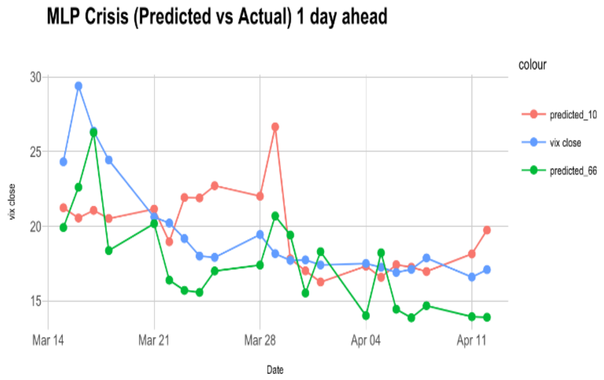
MLP

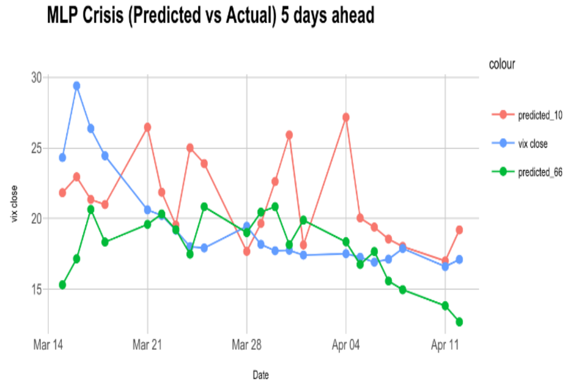
Crisis Period ended 2011/03/11



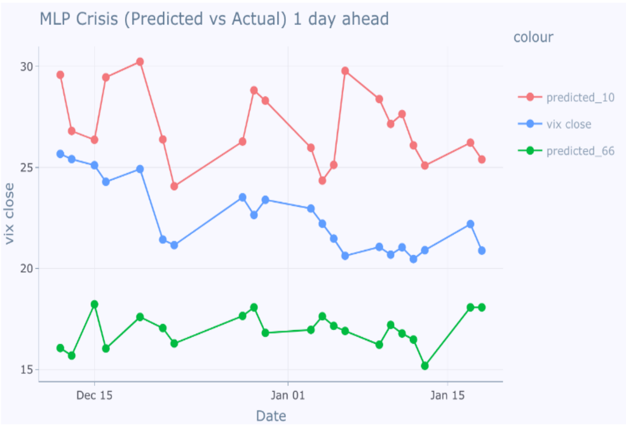


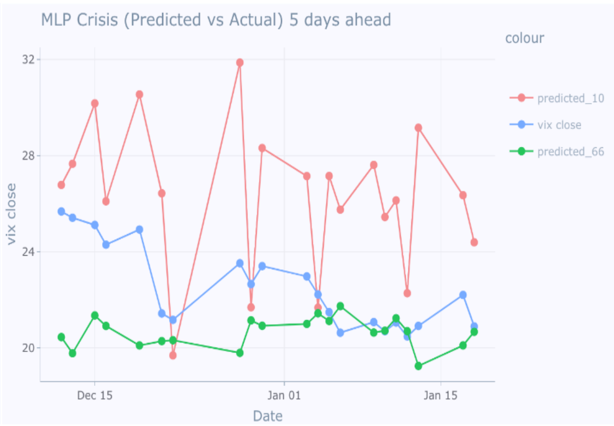
Crisis Period ended 2011/04/12



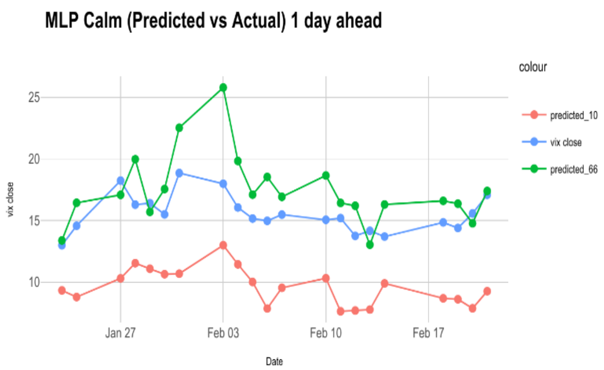


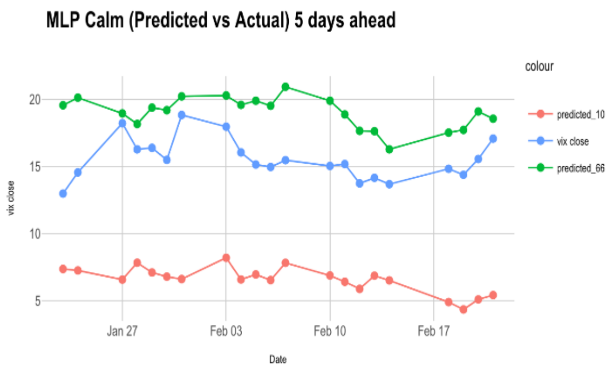
Crisis Period ended 2012/01/18



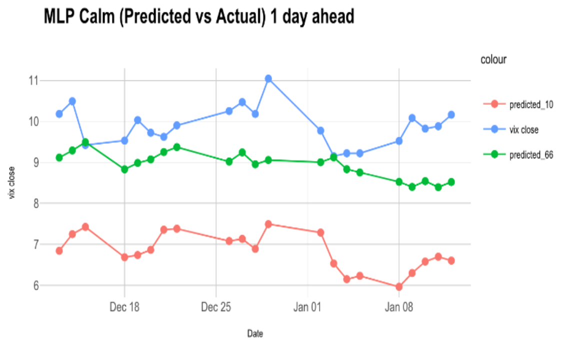


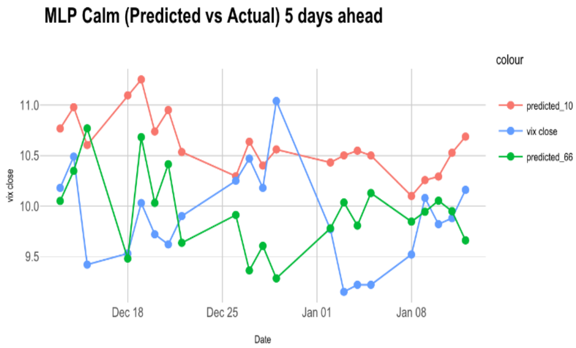
Calm Period ended 2020/02/21



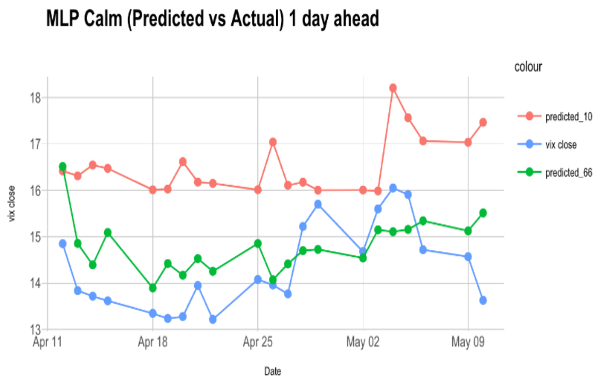


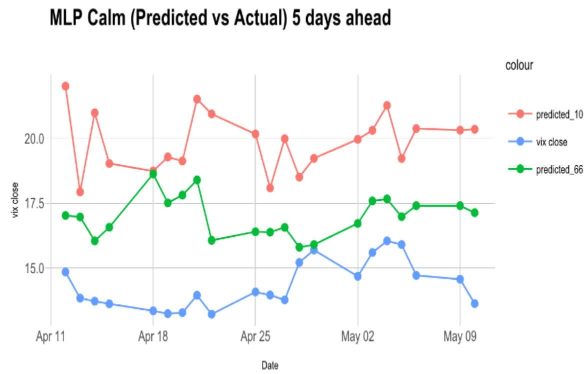
Calm Period ended 2018/01/11





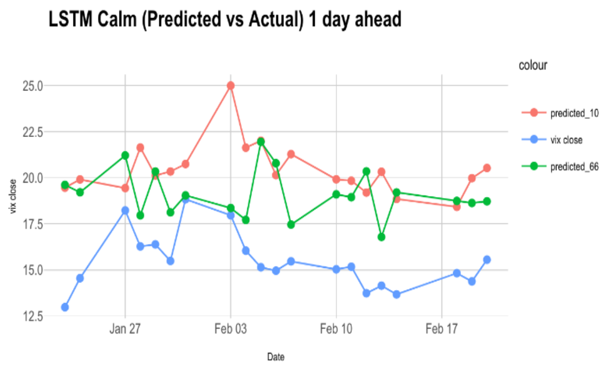
Calm Period ended 2016/05/10

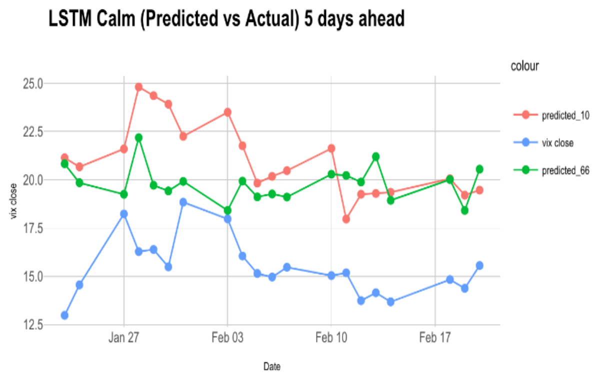




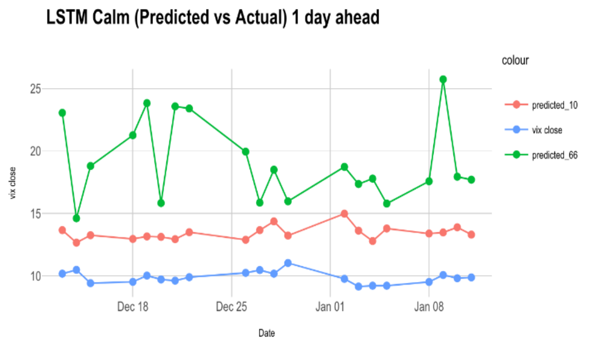
LSTM

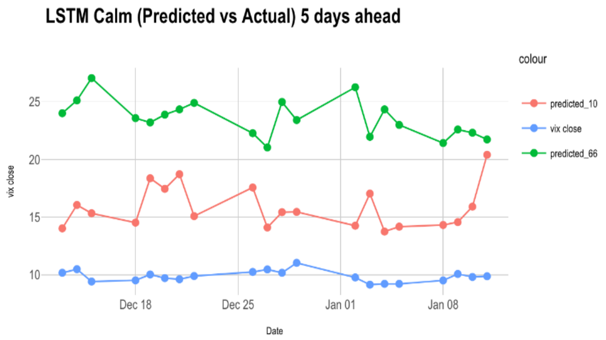
Calm Period ended 2020/02/21



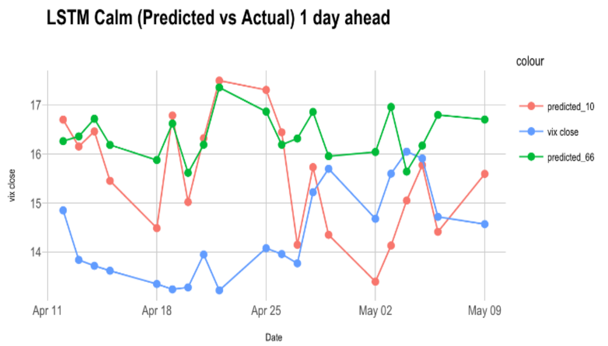


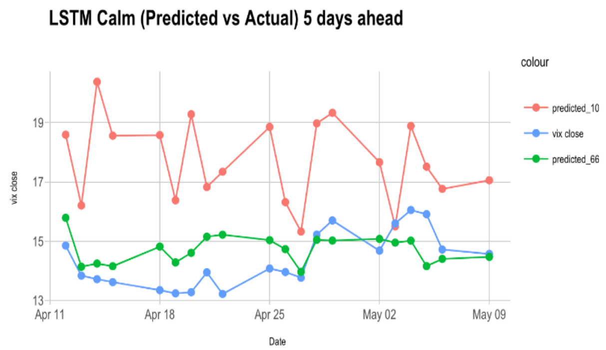
Calm Period ended 2018/01/11



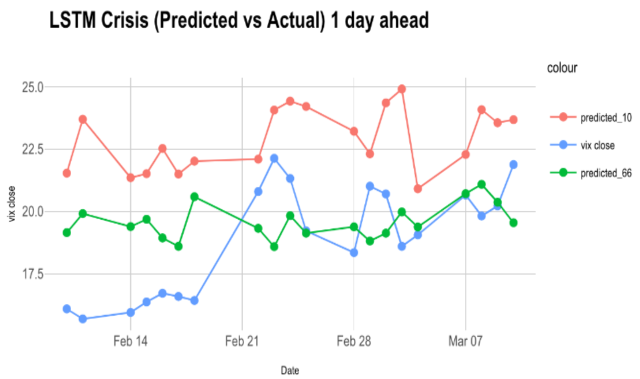


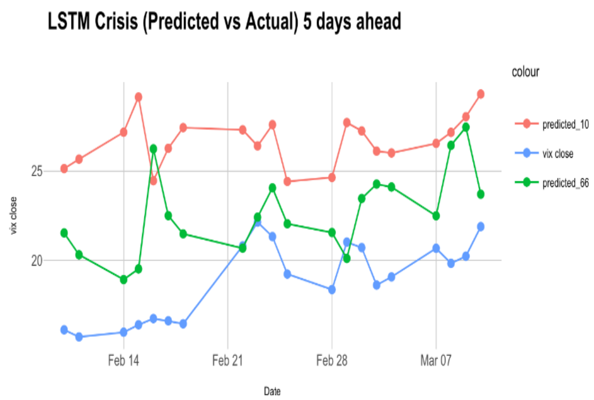
Calm Period ended 2016/05/10



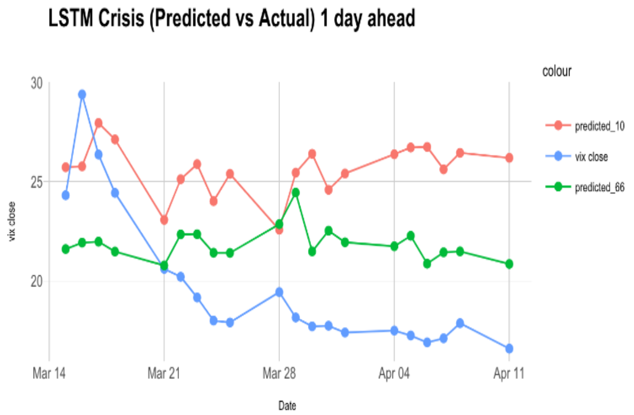


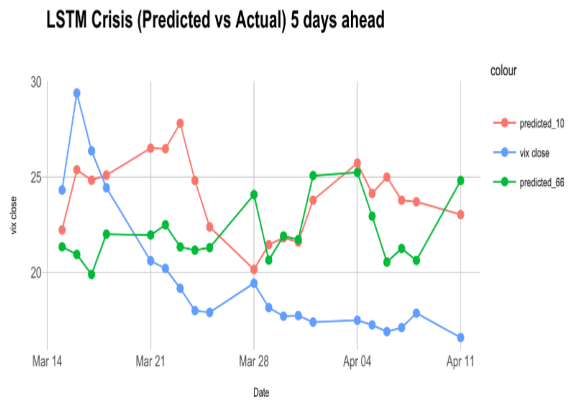
Crisis Period ended 2011/03/11





Crisis Period ended 2011/04/12





Crisis Period ended 2012/01/18