ERASMUS UNIVERSITY ROTTERDAM
ERASMUS SCHOOL OF ECONOMICS
Bachelor Thesis Economics & Business
Specialization: Financial Economics

Addressing Skepticism: An Evaluation of Standard Volatility Models, Random Forests, and LSTM Neural Networks in Forecasting Realised Volatility

Author: Mara Popescu
Student number: 599307
Thesis supervisor: Dr. Clint Howard
Second reader: Prof. Dr. I. Dittman
Finish date: 30.06.2024

The views stated in this thesis are those of the author and not necessarily those of the supervisor, second reader, Erasmus School of Economics or Erasmus University Rotterdam.

ABSTRACT

Volatility plays a crucial role in asset pricing, hedging or portfolio balancing; hence, gaining insights into its prediction would add substantial value. We propose a comparative study to assess the performance of both machine learning and benchmark econometric models in forecasting the realised volatility of returns of the FTSE 100 index. This research is aimed to test whether machine learning models, such as the long-short-term memory (LSTM) model, outperform popular models, such as the Autoregressive (AR) and Autoregressive-Moving-Average (ARMA) models, the (Generalised) Autoregressive Conditional Heteroskedasticity (ARCH)-type models, the Heterogeneous Autoregressive (HAR) model and another machine learning one, the Random Forest. The weekly realised volatilities of the index are trained over 16 years of data (between 2004 and 2020) and tested pseudo-out-of-sample over two-year periods (2020-2022 and 2022-2024). We interpret the results using the Means Squared Errors (MSE) and the Diebold-Mariano Test. The Autoregressive models (AR, ARMA and HAR-RV) have the lowest prediction errors, outperforming the LSTM. The latter consistently outperforms all ARCH-family models. Additionally, the GARCH(1,1) model measured considerably lower MSEs when its residuals were modelled using the LSTM. These findings showcased that the neural network model is a promising method of predicting realised volatilities; however, it highly depends on its design and chosen (hyper-)parameters.

Keywords: volatility, machine learning, financial time-series prediction

TABLE OF CONTENTS

CHAPTER 1 Introduction

"In finance, volatility is the variation of a financial asset over a certain period of time measured by the standard deviation of returns. It is the risk of change in an asset value" (Costa, 2017, p.1). Due to considerable noise present at the intraday and daily level, the patterns in the volatility series are commonly studied on a weekly basis. In the context of stock indexes, it serves as a substantial tool not only for investors, who use it to measure the overall market attractiveness but also for the policymakers, who judge the economic outlook based on its values. In recent times, financial institutions such as banks and insurers have been grappling with increasing complexity. This stems from various factors, including new regulations, customer interaction across multiple channels, fragmented systems, a wide array of products, and expansion into new geographic areas. As a result, decision-making and risk management have become more opaque and uncertain. This prompts a need to examine the effectiveness of financial models in predicting levels of volatility (Medland, 2015).

In the past, most literature considered conditional (observable) variance and conditional mean to estimate and forecast latent (unobservable) volatility. This line of thought led to the application of Autoregressive (AR) and Autoregressive Moving-Average (ARMA) models to volatility data and, hence, led to the appearance of Autoregressive Conditional Heteroskedasticity (ARCH) models. In studying the volatilities of financial assets, the traditional econometric models mainly include the Generalised Autoregressive Conditional Heteroskedasticity (GARCH) model (Bollerslev, 1986) and its variations (Glosten et al., 1993; Nelson, 1991; Zakoian, 1994). A GARCH model is similar to an Autoregressive Moving Average (ARMA) one by modelling the error variances. One important improvement of the GARCH model relative to previous ones is that it does not allow for the excess kurtosis of returns (Franses & Van Dijk, 1996). The variations of the model (QGARCH and GJR-GARCH) can also account for asymmetries in the magnitudes of negative or positive shocks. As previously stated, these models have consistently been applied to forecast conditional volatility. Nonetheless, Andersen et al. (2001) argue that using high-frequency data to construct estimates of daily volatilities in exchange markets is a more accurate approach than using conditional volatilities due to being (approximately) free of measurement error. The estimate is referred to as realised volatility (RV) and is calculated by summing high-frequency daily returns. Machine learning models have been continuously developed for this purpose. Gu, Kelly and Xiu (2020) performed a comparative analysis of these methods in a cross-sectional study, concluding that Neural Networks are the most efficient. Artificial Neural Networks (ANN) are a type of machine learning often blended with deep learning, which has two or more hidden layers. In finance, this method is able to recognize patterns in time-varying relationships, which might be hidden otherwise. Nelson et al. (2017) found the Long Short-Term Memory (LSTM) model to achieve an average of 55.9% accuracy when predicting the direction of stock price movements. The high performance of the LSTM is explained by the long training periods, which allow it to retain and adapt to the patterns and shocks present in the historical volatility series, and its ability to

capture non-linear data. These characteristics are essential for dealing with the stylised facts of volatility, such as clustering, fat tails and leverage effects.

My goal is to test whether neural network models outperform the traditional econometric models, including the ARCH family, in the context of the expected realised volatility of the UK Financial Times Stock Exchange 100 (FTSE 100) index. It is important to run this comparative study to better grasp the accuracy of traditional linear models in capturing different volatility characteristics, as well as their limitations. We run Autoregressive models, ARCH-, Random Forest and Neural Network models to assess and evaluate their performance in forecasting weekly realised volatility levels. On top of that, we assess the potential improvement of the benchmark GARCH(1,1) model with the integration of the LSTM in predicting its residuals, by implementing a hybrid model. It is unclear whether the results of the above mentioned studies also hold true for the UK stock market, as they only employed US, Chinese or developing countries stock market data. However, the UK market is a highly attractive target for applying modern machine learning techniques, especially in recent years, due to several key factors. Firstly, the UK has a well-developed financial infrastructure, with extensive historical data available on economic indicators. Secondly, the close integration of the UK's financial markets with global counterparts, particularly Europe and the United States, exposes UK stocks to diverse economic and financial shocks, amplifying volatility levels. Therefore, we will explore these ideas by answering the following research question: *"How do the Autoregressive models (AR, ARMA and HAR-RV), ARCH-family models (ARCH, GARCH, GJR-GARCH and GARCH-LSTM) and machine learning models (Random Forest model LSTM model) perform in predicting weekly realised volatilities of the FTSE 100 index over the 2004-2020 training period and the 2020-2022 and 2022-2024 out-of-sample periods, and which one is the most efficient?"*

In this research, we examine the performance of the LTSM model in comparison to the AR(1), ARMA(1,1), HAR-RV, ARCH(1), GARCH(1,1), GJR-GARCH(1,1) and the Random Forest using time series data on the UK stock index FTSE 100 from December 31st, 2003, to December 31st, 2023. The performance of the hybrid GARCH(1,1)-LSTM model will also be analysed in comparison to its counterpart, the GARCH(1,1). The sample is collected from the London Stock Exchange Eikon Data Stream and it contains the daily prices and log returns of the FTSE 100, which will be used to compute the weekly realised return volatilities. The period will be split into three disjoint subperiods, as in Gu et al. (2020): the training period (2004-2020), the validation period (2020-2022) and the testing period (2022-2024). The predictions for the validation and testing periods will be assessed separately using the observed values of the dataset during those years. The results rely on the Mean Square Error (MSE) to measure prediction errors and on the Diebold-Mariano (DM) Test to verify if the difference in model performance is statistically significant. The AR(1), ARMA(1,1), HAR-RV, ARCH(1), GARCH(1,1) and GJR-GARCH(1,1) models are implemented using the STATA software, while the Random Forest and LSTM model using Python. For the set-up of the latter, we use NumPy and Pandas packages as well as the libraries TensorFlow, including the Keras module, and ScikitLearn. The hybrid GARCH(1,1)-LSTM model is analysed using both software.

My hypothesis is that the traditional econometric models will underperform relative to the LTSM model for forecasting the volatility of the chosen sample. Given that the UK has been experiencing a dynamic economic and financial period due to its exit from the European Union in 2020 and the COVID-19 pandemic in early 2020, it is unclear whether traditional models could capture the non-linearities and noise present in the data. These events have likely introduced unprecedented volatility and uncertainty into the market, which could hardly be depicted with the Autoregressive- and ARCH-type models. Surprisingly, the findings suggest that the Autoregressive, Heterogeneous-Autoregressive and Autoregressive- Moving-Average Models outperformed the LSTM, measuring MSEs of 0.3195, 0.3197 and 0.3189 in the validation period and 0.1754, 0.1764, 0.1856 during the test period, while the LSTM measured MSEs of 0.4138 and 0.2092. The model also underperformed in relation to the Random Forest (MSE of 0.3397) in the validation period but overperformed in the testing period (MSE of 0.2279). The ARCH, GARCH and GJR-GARCH models were found to be inferior to the LSTM. Yet, it was found that a hybrid GARCH(1,1)-LSTM model, that predicts residuals using the LSTM, outperformed the standard ARCH-type of models. Although the LSTM demonstrated performance comparable to well-regarded models, it is crucial to carefully evaluate the analysis context to select appropriate input variables, as these can significantly impact the performacne of the method, potentially enhancing or damaging its predictive accuracy and reliability.

This paper is structured as follows: Chapter 2 introduces the theoretical framework of the above-mentioned models in financial literature, emphasising their use for forecasting realised volatility. In this section, the main characteristics of the models are also described, along with stylised facts of volatility. Chapter 3 expands on the selection of dataset, the variables used and their characterisitics. In Chapter 4, the methodology for testing the hypothesis is explained, which includes a more detailed description of the models used and their assumptions. In Chapter 5, the results for each period are presented and discussed, and the research question is answered. The final chapter, Chapter 6, provides a conclusion for this paper, along with its limitations and implications for financial practitioners.

CHAPTER 2 Theoretical Framework

This section briefly introduces the characterisitcs of the selected models and extends upon past literature on the applications of the Autoregressive models, HAR-RV, ARCH models, Neural Networks, specifically LSTM and hybrid models, and Random Forest in asset pricing and forecasting volatility levels. At the end, we also include a review of the stylised facts of volatility.

2.1 Autoregressive Models

An Autoregressive model of order *p*, AR(p), can be expressed as (Brooks, 2019, pp. 340-341):

$$y_t = \mu + \sum_{i=1}^{p} \phi_i y_{t-i} + u_t \quad (1)$$

Where y_t is the dependent variable at time *t,* μ is the constant term, y_{t-i} are the past values of y_t at each time step *t-€,* with € between [1, *p*], while u_t is the error term at time *t.* Poterba and Summers (1986) were one of the first to discover that AR(1) and AR(12) models were effective in forecasting the volatility of index returns, by applying the model to S&P 100 data. More recently, Chen et al. (2018) showed that AR models are more efficient than long-memory models on forecasting high-frequency volatility data for periods of up to one year. Autoregressive Moving-Average models, or ARMA, have frequently been used, as well. For example, Pong et al. (2004) compared short-memory ARMA models to long-memory ARFIMA (Autoregressive Fractionally Integrated Moving Average) and GARCH models, showing that the former is as effective as the long-memory models when forecasting future volatilities. An ARMA(p, q) model is defined as follows (Brooks, 2019, pp. 351-352):

$$y_t = \mu + \sum_{i=1}^{p} \phi_i y_{t-i} + \sum_{i=1}^{q} \theta_i u_{t-i} + u_t \quad (2)$$

Where, y_t is the dependent variable at time *t,* μ is the constant term, y_{t-i} and u_{t-i} are the past values of y_t and u_t at each time step *t-€,* with € between [1, *p*] and [1, *q*], respectively. u_t is the error term at time *t.*

2.2 HAR-RV model

An alternative way of forecasting volatility is by averaging daily realised volatilities (RV_t^d) to reflect weekly or monthly aggregates. The daily realised volatility is calculated as the sum of squared daily returns. The weekly and monthly realised volatilities series are computed as follows:

$$RV_t^w = \frac{1}{5}\sum_{d=1}^{5} r_{t-d}^2 \quad (3)$$

$$RV_t^m = \frac{1}{22}\sum_{d=1}^{22} r_{t-d}^2 \quad (4)$$

where r_t are daily returns during the respective week and month. The idea of incorporating different levels of data in one model was introduced by Muller et al. (1993) with the Heterogeneous Market Hypothesis (HMH). The model was extended by Corsi (2004), who proposed the Heterogeneous Auto-Regressive

Realised Volatility model (HAR-RV), which integrates the effect of trading at a daily, weekly and monthly level on realised volatility. The full model of the one-step ahead forecast is expressed as follows:

$$RV_{t+1} = C + \beta_d \mathrm{RV}_t^{\mathrm{d}} + \beta_w \mathrm{RV}_t^{\mathrm{w}} + \beta_m RV_t^m + u_{t+1} \quad (5)$$

Where *C* is a constant and u_{t+1} is the out-of-sample error term. In the context of realised volatilities of stock indeces, Louzis et al. (2012) showed that the HAR-RV model outperforms ARFIMA models for both in- and out-of-sample forecasts, by using S&P 500 and DIJA five minutes prices over a 10-year period. Additionally, a recent study also found that the HAR-RV can lead to high forecasting accuracy when applied to the volatility of oil prices in the US stock market (Tang et al., 2022).

2.3 ARCH Models

The first model proposed for analysing the volatility of returns was the Autoregressive Conditional Heteroskedasticity model for variance by Engle (1982). The ARCH model is similar to an Autoregressive model, which models the values of the conditional variance in the next time periods using a constant term and the past error terms. An ARCH model of order q, ARCH(q), can be written as (Brooks, 2019, pp. 508-509):

$$y_t = \mu + \beta_0 u_{t-1}$$

$$\sigma_t^2 = \alpha_0 + \sum_{i=1}^{q} \alpha_i u_{t-i}^2 \quad (6)$$

Where σ_t^2 is the conditional variance of the *y* series, α_0 is the constant and u_t is the error term at time *t*. The error term is normally distributed, with mean equal to zero and variance equal to σ_t^2. The model can capture both volatility clustering and fat-tails of the distribution. Yet, due to the large persistence of volatility, an ARCH model often needs a large order. In 1986, Engle and Bollerslev proposed the Generalised Autoregressive Conditional Heteroskedasticity model, which models conditional variance based on the past values of its own and of its error terms, as follows (Bollerslev, 2008; Brooks, 2019, p. 514):

$$\sigma_t^2 = \alpha_0 + \sum_{i=1}^{q} \alpha_i u_{t-i}^2 + \sum_{i=1}^{p} \beta_i \sigma_{t-i}^2 \quad (7)$$

Where $\alpha_0 > 0$, and $\alpha_i, \beta_i \geq 0$, for any i in $[1, q]$ and $[1, p]$, respectively. Taylor (1986) was the earliest to test the predictive power of the GARCH model. Following their studies, the GARCH model became a benchmark in the financial literature. Even so, this model is not robust to negative unexpected returns, which were found by Donaldson and Kamstra (1996) to have a greater impact on the changes in volatilities than positive ones. This is also known as the '*leverage effect'* (Tripathy & Garg, 2013). The leverage effect was earlier observed in the UK stock market by Chelley-Steeley and Steeley (1996). The strength of the leverage effect is inversely related to firm size. Thus, smaller firms exhibit greater volatility when their leverage increases. The asymmetric pattern of volatility spillovers means that the volatility in one portfolio affects volatility in another portfolio. The pattern is asymmetric, meaning it is not equal in both directions.

Specifically, shocks or unexpected changes in the stock prices of larger firm portfolios impact the volatility and average returns of smaller firm portfolios. However, shocks in smaller firm portfolios do not spread on larger firm portfolios (Chelley-Steeley & Steeley, 2005). The GJR-GARCH model of Glosten, Jagannathan and Runkle (1993) incorporates this by adding an indicator function, *I*, to the equation, which takes the value 1 if the shock is negative and 0 otherwise. The GJR-GARCH(1,1) model is, therefore, given by (Brooks, 2019, p. 522):

$$\sigma_t^2 = \alpha_0 + \alpha_1 u_{t-1}^2 + \gamma_1 u_{t-1}^2 I_{t-1} + \beta_1 \sigma_{t-1}^2, \quad (8)$$

$$\text{where } I_{t-1} = \begin{cases} 1, & u_{t-1} < 0 \\ 0, & u_{t-1} \geq 0 \end{cases}$$

The rest of the terms in the equation are defined similarly to the ones in eq. (6) and (7) of the ARCH and GARCH models. Previous research, such as the one of Engle and Ng (1993), tested the predictive power of the GJR model to other ARCH variations by forecasting daily Japanese stock returns. The results confirmed the top performance of the model. In a similar manner, Hansen and Lunde (2005), compared 330 types of ARCH models to forecast out-of-sample both conditional and realised volatility levels. For the former, they used nine years of intraday returns of the IBM stock, finding that the GJR was the most efficient. However, for the latter, they employed data on the spot exchange rates of the German Deutsche Mark to the US Dollar, concluding that the GARCH(1,1) remained the most effective one. They attributed this difference in results to the leverage effects, present in the IBM stock returns. Other studies, such as the one of Franses and Van Dijk (1996), argued that the GJR-GARCH is not a useful tool for forecasting weekly market volatilities of indices. Pagan and Schwert (1990), who introduced the *news impact curve*, raised concerns about the non-stationarity of volatility data, claiming that, even though the GJR may be the most effective model, it may not be applicable to long periods of data.

2.4 Neural Networks

Neural Networks (NN) have gained popularity due to their successful predictive power and application in a broad range of fields, including finance. A neural network is defined as "a collection of connected units, where each connection has a weight associated with it" (Joarder et al., 2006, p. 48). This idea resembles the human nervous system, where the biological neurons are replaced by mathematical functions, i.e. artificial neurons. In practice, the human brain consists of 100 billion neurons and over 100 trillion synaptic connections (Herculano-Houzel, 2009). The neurotransmission takes place through a neuron's dendritic receptors, which capture stimuli from external sources and send them to the cell body. Then, upon receiving enough stimuli, neurons propagate information to one another by action potentials (i.e. nerve impulses). Similarly, artificial neurons process the external inputs in proportion to their strength and type and attribute them a weight accordingly. In the cell body, the weighted average of these inputs is calculated and, if this is great enough, the neuron activates to output 1; otherwise, it remains inactive at 0.

An Artificial Neural Network (ANN) uses three types of layers: the input, which consists of the external data sample, the hidden layer, which carries the infromation from the previous layer along with their weights, and the output. The hidden layers are those layers that contribute to the training of the network. Particularly, these layers consider the outputs of the layers before transforming them. The type of neural connections is essential for operating ANNs, as the excitatory inputs correspond to the operation of addition, while inhibitory inputs to subtraction. Mathematically, the inputs of each layer can be expressed as sums of series $x_1, x_2, x_3 \dots x_n$, each of them having corresponding weights $w_1, w_2, w_3, \dots w_n$ and by adding a bias term b, as in eq. (9) (Hornik et al., 1989).

$$z = w_1 x_1 + w_2 x_2 + \cdots w_n x_n + b \tag{9}$$

We can denote the weight matrix of each layer *l,* $W^{[l]}$ and the bias vector*,* $b^{[l]}$. Then, the output of the layer *l* can be expressed as:

$$a^{[l]} = \sigma(W^{[l]} a^{[l-1]} + b^{[l]}) \tag{10}$$

Where σ is the sigmoid function, its activation function. The sigmoid function is interpreted as the degree to which the inputs of each layer are forgotten. The range of the function is the interval [0,1], 0 meaning that no information from is retained, while 1 means that all information is kept (eq. 11).

$$\sigma(x) = \frac{1}{1 + e^{-x}} \tag{11}$$

Furthermore, to assess the performance of the network, the following cost function is defined:

$$Cost = \frac{1}{n} \sum_{i=n}^{n} \left\| y(x_i) - a^{[l]} x_i \right\|_2^{\,2} \tag{12}$$

Where $y(x_i)$ denotes the desired output of the method, $a^{[l]} x_i$ is the actual output of the layer *l* and x_i is the vector of inputs in each layer, with ϵ between [0, *n*] and *n* is the total number of inputs. The cost function is minimised by *'training'* the model using the '*gradient descent method'*. The gradient is recursively lowered in each layer at the learning rate. The computation is shown in Appendix A. The *learning rate* indicates how much the parameters are adjusted during the step towards the steepest descent. If the learning rate is too high, the model might converge too quickly to a suboptimal solution, while if the learning rate is too low, it can lead to a slow and suboptimal training of the model. The learning process employed by an artificial neural network is called '*back-propagation'*, and it comprises a series of test cases (the training set) that are inputted into the net. The errors between the actual and desired output of the net are propagated backwards to the internal layer(s) such that the weights are adjusted in proportion to the error. Specifically, in the context of volatility forecasting, NNs can learn any arbitrary mapping *f* from a matrix *X* of observations (e.g. the volatility lags of an index $X_t = [r_t, r_{t-1}, r_{t-2}, \dots]^T$) to the output *y:*

$$\widehat{y_{t+1}} = f(X_t)$$

Where $\widehat{y_{t+1}}$ is the forecasted volatility in the next period (Ge et al., 2022).

One type of neural network is the Recurrent Neural Network (RNN), which is particularly designed for capturing time-series data and, hence, is the one applied in this paper. In an RNN network, the output of one layer can return as an input to the previous or the same layer. This also means that such a network cannot differentiate between the three types of layers (i.e., input, hidden and output) (Agatonovic-Kustrin & Beresford, 2000). The RNN builds a hidden state as it recursively parses through the input sequence, retaining useful information from previous elements, often referred to as network memory. However, "learning to store information over extended time intervals by recurrent backpropagation takes a very long time, mostly because of insufficient, decaying error backflow" (Hochreiter & Schmidhuber, 1997, p. 1). This problem is known as '*constant error back propagation*'. A solution to the problem was *The Long Short-Term Memory Model* (LSTM), which uses gates to allow the network to remember, update, and forget information.

2.4.1 LSTM

Studies involving Neural Networks, specifically long-memory (LM) models, have been applied to volatility forecasting relatively late. The earliest paper appeared in 1998 and applied the algorithm to forecast the implied volatility of Black-Scholes (Hwang & Satchell, 1999). However, research on realised volatility, and not implied volatility, begins with Andersen et al. (2001), Vilasuso (2002) and Zumbach (2003). The LSTM algorithm started to gain popularity even later. Here, we will discuss the most notable and relevant studies for volatility forecasting.

Firstly, Fischer and Krauss (2018) found that the LSTM gave more accurate results than the Random Forest, deep neural net, and the logistic regression classifier for the prediction of S&P 500 price-movements. Tamura et al. (2018) blended technical and fundamental analyses to assess stock values in the Japanese market. They employed the LSTM for technical analysis while using a program to gather and organise financial data from Japanese-listed companies for fundamental analysis. Their results showed an improvement of 11.92% when using the LSTM compared to single-factor approaches. Moon and Kim (2019) tested the performance of the model on the prices and volatilities of five stocks, including NASDAQ and S&P 500, over a seven-year period, emphasising its high performance as well. Fister et al. (2019) proposed the LSTM algorithm for automatic stock market trading, which outperformed other trading strategies such as passive, rule-based, and surrogate model strategies. Lastly, Wang et al. (2020) compared the results of LSTM against the support vector machine (SVM) model, Random Forest, DNN, and ARIMA to find out how to form the optimal portfolio. The authors conclude that the LSTM was the most suitable method for forecasting financial time series.

2.4.2 Hybrid GARCH-LSTM Model

In addition to the traditional GARCH(1,1) model, we propose an innovative approach that predicts its residuals using the LSTM method. As the original model cannot capture non-linear correlations, combining it with an RNN would enhance its predictions. Previous studies discussed the improvements brought by the

integration of GARCH-type of models and Neural Networks. To illustrate this, Tseng et al. (2008) estimated the volatility of option prices of a Taiwan index using a hybrid model, integrating the feedforward neural network and GARCH, which outperformed the singular models. Kim and Won (2018) combined the LSTM with three types of GARCH models (GARCH, Exponential GARCH, or EGARCH, and the Exponentially Weighted Moving Average, EWMA) to forecast the realised volatility of the KOSPI 200 index. Their results indicated that the prediction error is remarkably reduced using the new model. Similar results were also obtained by Hajizadeh et al. (2012) and Kristjanpoller et al. (2014) who apply GARCH-ANN models for predicting the volatility of S&P 500 and the volatility in South American regions, respectively.

2.5 Random Forest

Breiman (2001, p. 1) introduced the Random Forest (RF) model as " a combination of tree predictors (…) each tree depends on the values of a random vector sampled independently and with the same distribution for all trees in the forest". As at the basis of the model lays the concept of *'decision trees'*, in the following section, we briefly discuss their mechanism, followed by their implementation in a Random Forest, and, lastly, we discuss their applications in the financial literature.

2.5.1 Decision Trees

A decision tree is a network of nodes and connecting edges, referred to as '*branches*'. The information flows form the highest node, the '*root*,' towards the lowest nodes, the '*leaves*'. Thus, a decision tree has three levels: the *root nodes*, the *inner nodes* and the *leaf nodes*. The inner nodes are those intermediary nodes, between the root and the leaves, at the level of which decisions are made. The decisions are made through the branches, which aim to subdivide the data recursively (Luong & Dokuchaev, 2018; Breiman, 2017). The structure of an example of a decision tree is depicted in Fig. 2.1.

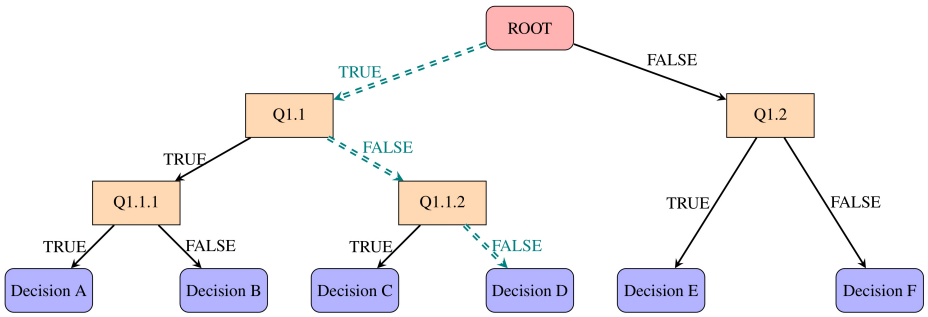


Figure 2.1. The structure of a binary decision tree.

Note: Starting from the root node, a decision is drawn at each leaf.

Source: Luong C, Dokuchaev N. (2018). Forecasting of Realised Volatility with the Random Forests Algorithm (p. 3). *Journal of Risk and Financial Management*.; 11(4):61. https://doi.org/10.3390/jrfm11040061.

Tree-based models have the disadvantage of being unstable and having a high variance, so even slight changes in the data can lead to drastically different results or even to overfitting. To mitigate this issue, the Random Forests algorithm can be employed. Random Forests are an ensemble approach, meaning that they

combine multiple models (or base classifiers) to increase predictive accuracy. As such, Breiman (2001) find that, by using the methods developed by Ho (1995), who develops the *'random selection of features'*, the variation of the decision trees can be controlled. *'Bagging sampling'* (or '*bootstrap aggregation'*) is a method used for reducing the variance of noisy data by averaging the predictions of various regression trees that employ random data sub-samples. Overall, a Random Forest comprises multiple decision trees. For regression problems, bagging sampling is employed, and each tree is constructed using a sample with replacement (i.e., selected subjects are put back into the population before another subject is sampled) from the training set. Each sample was found to have a probability of approximately 70% to be selected. Instances in the sample are called 'in-bag instances', while the rest are 'out-of-bag instances'. The total number of instances is also referred to as the *'Breiman's bagger parameter'*. In this paper, this is equal to three, as we only use the last three lagged values of realised volatility. For classification problems, each decision tree is considered a base classifier for the class label (i.e. dependent variable) of an unlabeled instance. The outcome of each tree is decided using a random sample of the three instances and by aiming to minimise the objective function, the MSE. The classification of the instances uses the '*majority voting'* rule, which means that after each classifier indicates one class label (i.e. a prediction value), the class label with the most votes is decided (Fawagreh, 2016). The process is depicted in Fig. 2.2:

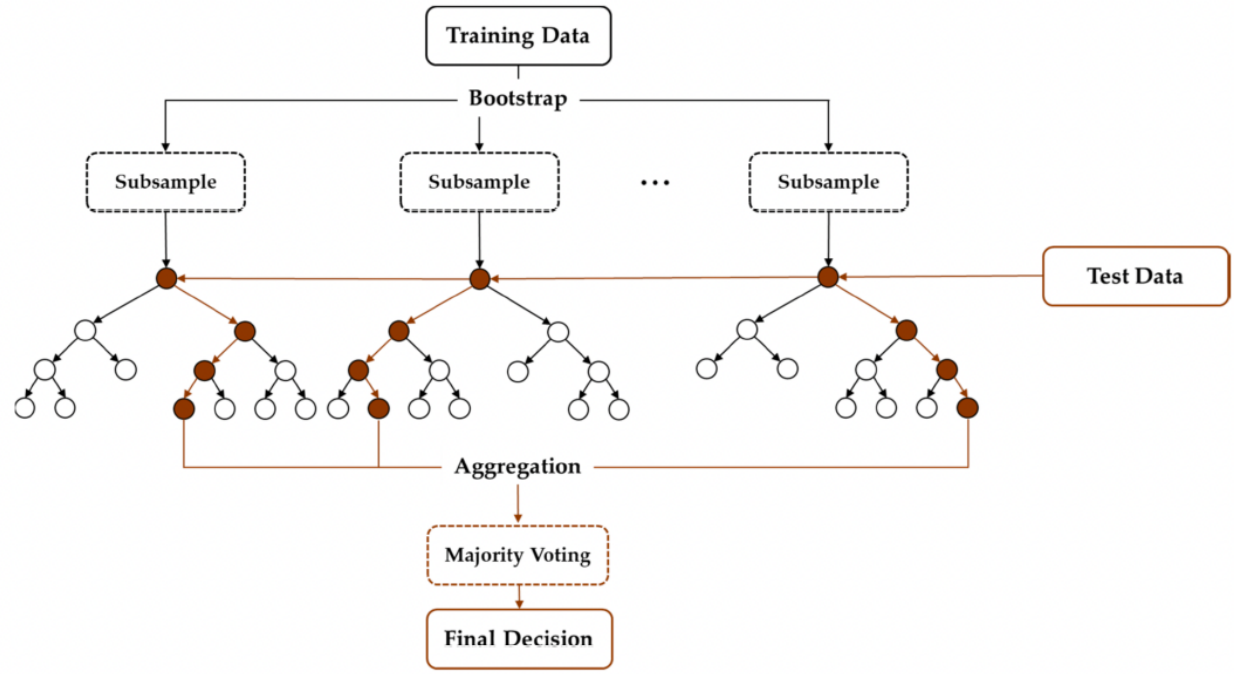


Figure 2.2: Conceptual framework of Random Forest Classifier.

Note: The Classifier firstly uses boostrap aggregation to split the training data into subsamples. Each subsample selects one class label, based on a random subsample of the initial instances (i.e., in Fig. 2.2, the decisions taken along the red-path). Then, the labels are aggregated and decided upon, through majority voting.

Source: Jeong, D. H., Kim, S. E., Choi, W. H., & Ahn, S. H. (2022). A comparative study on the influence of undersampling and oversampling techniques for the classification of physical activities using an imbalanced accelerometer dataset. In *Healthcare* (Vol. 10, No. 7, p. 1255). MDPI.

The main advantage of the model is that it can solve both classification and regression problems with high accuracy, even though other advantages, such as simplicity and insightfulness through revelaing internal estimates are also present (Breiman, 2001). In more recent years, researchers have tried improving the RF model by addressing issues with feature selection (Robnik-Šikonja, 2004; Amaratunga et al., 2008) and voting (Robnik-Šikonja, 2004; Tsymbal et al., 2006).

Similarly to Neural Networks, research on realised volatility forecasting using RF is limited, even though its popularity has grown significantly (Diane & Brijlal, 2024). Ballings et al. (2015) compared a diveristy of ensemble methods, including the Random Forest, to Neural Networks, Logistic Regression, SVM and K-Nearest Neighbor, showing that Random Forest is the top-performing method for European stock price direction prediction. Khaidem et al. (2016) found that for all the datasets used (Apple, Microsoft and Samsung stock prices), the accuracy of RF for long-term prediction was 85-95%, beating all other models considered (SVM, NN and I Bayesian Classifier). Sharma and Juneja (2017) tested the same model for the short term using historical data from the Indian stock markets of NIFT 50 and S&P Bombay Stock Exchange. The paper concluded that RF outperformed the Support Vector Regression (SVR). In 2018, Luong and Dokuchaev applied RF to the existing HAR model framework in forecasting the direction of volatility of S&P 200 prices. The model attained 80% accuracy compared to the 57% of the benchmark.

2.6 Stylised Facts

In order to estimate and forecast the volatility of financial time series, it is important to assess the particular characteristics of this type of data. Therefore, we consider the following stylised facts for the volatility of returns in choosing model specifications further:

Volatility clustering refers to the fact that whenever there are big shocks in the market, the volatility increases and stays high for consecutive periods, until the effects of the news are fully propagated.

Fat tails, or *excess kurtosis*, means that the the volatility of financial assets does not typically follow a normal distribution, depincting rather high probabilities of extreme profits or losses.

Leverage effects, already defined, refer to the idea that smaller firms can face higher volatility levels when their leverage increases. Additionally, in the presence of negative shocks, smaller firms are more prone to suffer larger decreases in their average returns than larger firms.

Long memory refers to the idea that volatility is more persistent in high-frequency data than in longer time intervals, which might have unit root behaviour. This led to the development of two processes for modelling the volatility, depending on the time level at which it is being measured: ARCH models, for unit roots, and

stochastic volatility models, for long-memory processes (Satchell & Knight, 2011). The long memory characteristic of volatility is also equivalent to the phenomenon of 'mean-reversion'.

2.7 Contribution and Hypotheses

This research aims to assess the performance of the LSTM algorithm in forecasting the realised volatility of the UK stock index FTSE 100 in comparison to the Autoregressive models, the ARCH family, and the Random Forest model. In particular, the AR(1), ARMA(1,1), ARCH(1), GARCH(1,1) and GJR-GARCH(1,1) are the most popular ones in the literature and serve as a benchmark in this study. The performance of the LSTM is also compared with that of other machine learning algorithms: the Random Forest and the hybrid GARCH-LSTM. As previously stated, this model was found to entail high levels of accuracy, and it is designed to manage volatilities in noisy data. Hence we expect it to outperform other models, particularly when applying it to weekly realised volatilities of the UK's stock index FTSE 100. Additionally, this study assesses the potential improvement of the traditional GARCH(1,1) by integrating it with the LSTM. The studied hypotheses are the following:

H1: The LSTM model outperforms the Autoregressive models (AR, ARMA, HAR-RV), ARCH-family models, the Random Forest and the hybrid GARCH-LSTM in predicting weekly realised volatilities of the FTSE 100 index during 2020-2022.

H2: The LSTM model outperforms the Autoregressive models (AR, ARMA, HAR-RV), ARCH-family models, the Random Forest and the hybrid GARCH-LSTM in predicting weekly realised volatilities of the FTSE 100 index during 2022-2024.

H3: The hybrid GARCH-LSTM model outperforms the GARCH(1,1) in predicting weekly realised volatilities of the FTSE 100 index during 2020-2022.

H4: The hybrid GARCH-LSTM model outperforms the GARCH(1,1) in predicting weekly realised volatilities of the FTSE 100 index during 2022-2024.

The aim of testing these hypotheses is to gain more insights into how to achieve more accurate forecasts of volatility, specifically in the UK. The findings would be helpful in better understanding the performance of different econometric models for modelling financial data, an important question for academic practitioners, policymakers, risk managers and bankers. For instance, activities such as stress testing in risk management, hedging and developing arbitrage opportunities, or developing effective economic policies are guided by volatility forecasts (Engle & Patton, 2007). Even though these practitioners might be interested in finding a parsimonious model that reaches high levels of efficiency and measure low levels of errors, it is important to understand and evaluate the complexity of these models in order to apply and design them. The findings of this study go beyond predicting the realised volatility of returns, as it sheds light on the limitations and biases faced when using machine learning methods and their complex architecture.

CHAPTER 3 Data

3.1 Sample and Data Collection Method

The data collected comprises the daily frequencies of the FTSE 100 index price from December 31st, 2003, to December 31st, 2023, totalling to 5,218 observations. The period will be split in three disjoint subperiods: the training period (2004- 2020), the validation period (2020-2022) and the testing period (2022-2024). The source of the dataset is the London Stock Exchange Eikon Data Stream. The choice of these periods ensures a balanced approach to develop the model. The training period, spanning 16 years of weekly data points, is long enough to capture a diversity of market conditions, while the validation period, includuing two recent years, is used to adjust the parameters and configuration of the models by evaluating different variations and choosing the one that performs best on this dataset. The testing period, comprising the most recent two years of data, is used to evaluate the accuracy of the models. As the stock market is not active during the weekend and certain holidays, observations taking place from Saturday to Sunday are excluded such that the distributional characteristics of the volatility will not be obscured by their deterministic calendar effects.

3.2 Variables

The variables used include the weekly realised return volatility of the FTSE 100 over the period 2004 to 2024. We use RV_t as a notation for the weekly realised volatility at time *t.* The weekly realised volatility is computed as the square root of the average of the sum of the squared daily differences in returns during that week, as in Kim and Won (2018):

$$RV_t^w = \sqrt{\frac{1}{5}\sum(r_t - \overline{r_t})^2} \tag{13}$$

The returns used are continuously compounded returns over the period from *(t-1)* to *t* using the logarithm of the price, and multiplied by 100 to express percentages:

$$r_t = 100 \times (\ln(P_t) - \ln(P_{t-1})),$$

And $\overline{r_t}$ is the average of the log returns. P_t is the closing price of the index at time *t*. For this variable there are 5,217 observations due to the missing value of the first observation. Monthly volatilities are calculated similarly as the average of the 22-sums of differences in returns realised in that month:

$$RV_t^m = \sqrt{\frac{1}{22}\sum(r_t - \overline{r_t})^2} \tag{14}$$

3.3 Summary Statistics

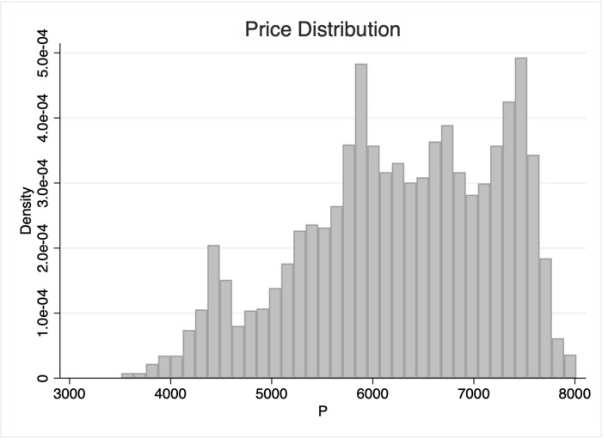
Descriptive statistics for all four variables are included in Table 1, on the next page. The prices and weekly realised volatilities are expressed in real terms, while the returns variable is expressed in percentages.

Table 1. Descriptive statistics of weekly realised volatility, daily prices and daily returns of the FTSE 100, 2004-2024

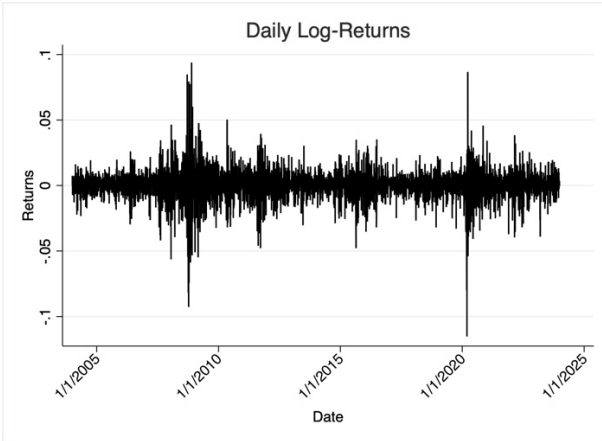
	N	Mean	Std. dev.	Min	Max
Price_t	5,218	6,265.97	970.11	3,512.09	8,014.31
Return_t	5,217	0.01	1.09	-11.51	9.38
RV_t^w	1,044	0.80	0.60	0.05	6.42

The closing price of the index measures an average of 6,265.97 and a standard deviation of 970.11 (Table 1). The time-series of daily closing prices of the FTSE 100 between 2004 and 2024, along with its distribution, are depicted in Fig. 3 (a) and (b). The series registered several dramatic falls, specifically in 2008 and 2020, which were followed by abrupt increases. The distribution is non-normal, being rather skewed to the left. Nonetheless, the log-returns of prices seem to exhibit a normal one (Fig. 3 (d)). The stationarity of returns was also verified using the Dickey-Fuller test, whose hypothesis that the series is non-stationary was rejected at a 1% significance level (Table 5.4 in Appendix A).

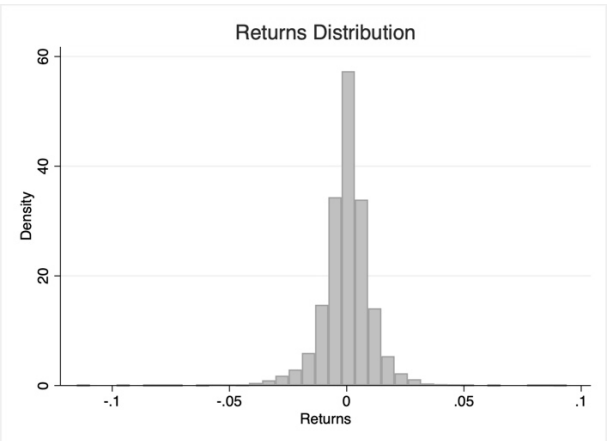
(a)



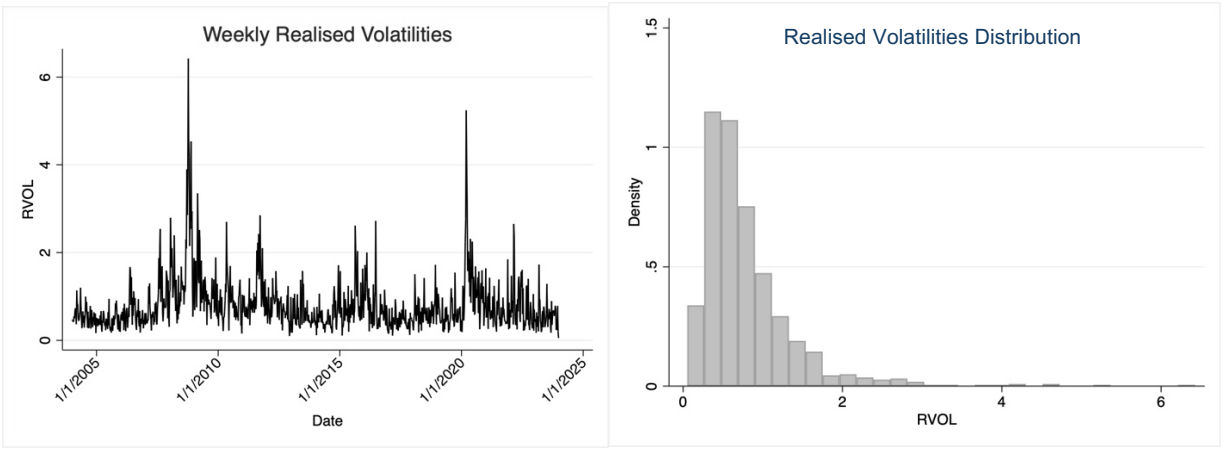
(b)



(c)



(d)



(e) (f)

Figure 3: *The evolution and distribution of FTSE 100 index price, daily returns and realised volatility, 2004-2024*. *Note*: The subplots show as follows: (a) the evolution of the index price on a daily level; (b) the distribution of the daily index prices; (c) the evolution of daily log-price-returns; (d) the distribution of daily log-price-returns; (e) the evolution of the weekly realised volatility of price returns; (f) the distribution of realised volatility of price returns.

Fig. 3 (e) shows the way FTSE 100 realised volatility changes throughout the period. The mean of the series is 0.80, with high peaks during 2008 and at the end of 2020. The peak in 2008, which is also the highest of the series (6.42), is associated with the global financial crisis, characterised by investors' increased uncertainty and risk aversion, the spillover effects from other major economies, particularly the US, where the crisis originated. Furthermore, the realised volatility in December 2020, measuring 5.24, can be attributed to the COVID-19 pandemic. Some factors encountered at that time, and which may explain this value are the global supply chain disruptions, increased uncertainty among both managers and investors or rapid changes in the interest rate levels. Additionally, December 2020 brought about the negotiations between the UK and the EU regarding Brexit. This played a role in the volatility of the UK stock market, as investors repositioned their portfolios to hedge against potential risks of a 'no-deal' or 'deal' scenario. Fig. 3 (f) depicts a right-skewed distribution of realised volatilities, with few values above three, which correspond to the 2008 and 2020 peaks. The Dickey-Fuller Test was also rejected at 5% significance level, concluding that the series is stationary (Table 5.4 in Appendix A).

CHAPTER 4 Method

To evaluate the performance of the LSTM model in forecasting volatility, we implement three Autoregressive models: AR(1) ARMA(1,1) and HAR-RV, three ARCH family models: ARCH(1,1), GARCH(1,1), GJR-GARCH(1,1), the Random Forest model and a hybrid GARCH-LSTM model. The specifications of these models are further described in this section.

4.1 Autoregressive Models

Adapting the equations in the theoretical framework to the context of this research, the AR(1) forecasts realised volatility (RV_t) as follows (eq. 15.1):

$$\mathrm{RV_t} = \mu + \phi_1 \mathrm{RV_{t-1}} + u_t \tag{15.1}$$

the out-of-sample one-step-ahead forecast being (eq. 15.2):

$$\widehat{\mathrm{RV}}_{t+1} = \hat{\mu} + \hat{\phi}_1 \mathrm{RV_t} \tag{15.2}$$

Where we use the notation $\widehat{RV_t}$ as the forecasted value of RV_t, μ is the constant term in the AR(1) model and ϕ_1 is the coefficient of the Autoregressive term. u_t in eq. (15.1) is the error term at time *t*.

The ARMA(1,1) model is expressed as (eq. 16.1):

$$\mathrm{RV_t} = \mu + \phi_1 \mathrm{RV_{t-1}} + \theta_1 u_{t-1} + u_t \tag{16.1}$$

With an out-of-sample one-step-ahead forecast as (eq. 16.2):

$$\widehat{RV}_{t+1} = \hat{\mu} + \hat{\phi}_1 RV_t + \hat{\theta}_1 u_t \tag{16.2}$$

Where μ is the constant term in the ARMA(1,1) model, ϕ_1 is the coefficient of the Autoregressive term and θ_1 is the coefficient of the Moving Average term. The HAR-RV model takes the form already stated in eq. (3), (4) and (5), being shaped as an Ordinary Least Squares regression, containing the first lag of RV_t and its corresponding monthly aggregate. The stationarity assumption of the autoregressive models is satisfied for the series.

4.2 ARCH Models

The adapted ARCH(1) model from eq. (6), is defined as in eq.(17.1):

$$r_t = \beta_0 + \varepsilon_t$$

$$\sigma_t^2 = \alpha_0 + \alpha_1 \varepsilon_{t-1}^2 \tag{17.1}$$

Where r_t is the log-return of the FTSE 100 at time *t*, β_0 is their mean value and ε_t are the normally distributed error terms at time step *t*. In the second equation, σ_t^2 is the conditional variance of returns and α_0 is a constant. ARCH(1) simultaneously estimates the two regression equations and chooses the parameters that give the best description of the data, along with estimating $\widehat{\sigma_t^2}$. Note that in this case the

model forecasts the conditional variance of the FTSE 100 log-returns σ_t^2, and not the realised volatilities, RV_t. Therefore, its predicted values are compared with the realised variances of the returns. The one-step-ahead forecast using the ARCH(1) is (eq. 17.2):

$$\widehat{\sigma_{t+1}^2} = \alpha_0 + \widehat{\alpha_1}\varepsilon_t^2. \tag{17.2}$$

The GARCH(1,1) model is similarly defined using the following set of equations (eq. 18.1):

$$r_t = \beta_0 + \varepsilon_t$$

$$\sigma_t^2 = \alpha_0 + \alpha_1\varepsilon_{t-1}^2 + \beta_1\sigma_{t-1}^2 \tag{18.1}$$

Where $\sigma_{t-1}^2$are the lagged values of conditional variance. The performance of the model is also assessed by comparing the observed realised variances with its one-step-ahead predicted values (eq. 18.2):

$$\widehat{\sigma_{t+1}^2} = \alpha_0 + \widehat{\alpha_1}\varepsilon_t^2 + \widehat{\beta_1}\sigma_t^2 \tag{18.2}$$

The GJR-GARCH(1,1) estimates the conditional variance of the FTSE 100 log-returns as (eq. 19.1):

$$r_t = \beta_0 + \varepsilon_t$$

$$\sigma_t^2 = \alpha_0 + \alpha_1\varepsilon_{t-1}^2 + \gamma_1\varepsilon_{t-1}^2 I(\varepsilon_{t-1} < 0) + \beta_1\sigma_{t-1}^2 \tag{19.1}$$

Where α_0 is a constant term, α_1 is the coefficient for the lagged squared error term and the impact of the positive shock, while γ_1 measures the impact of the negative shock in the previous period, if it exists. The existence of the shock is given by the function *I,* which equals 1 if ε_{t-1} is negative and 0 otherwise. β_1 is the coefficient for the lagged variance term. The one-step-ahead forecast of the variance of returns is given by (eq. 19.2):

$$\widehat{\sigma_{t+1}^2} = \omega + \widehat{\alpha_1}\varepsilon_t^2 + \widehat{\gamma_1}\varepsilon_t^2 I(\varepsilon_t < 0) + \widehat{\beta_1}\sigma_t^2 \tag{19.2}$$

The ARCH-type models assume both stationarity and homoscedasticity, which are satisfied in the case of the FTSE 100 log-returns.

4.3 The LSTM: Model Specifications

As discussed in Section 2.4.1, the LSTM is a recurrent neural network, whose process is illustrated in Fig. 4.1. The main aspects of the network are the memory unit C_t, the hidden state $h_t$and the three gates. The process starts from the left bottom of the figure, with an input vector x_t and the hidden state h_{t-1}. The forget, input and output gates are then activated through the sigmoid function. The process ends by finding the next memory unit and the hidden state (Nosratabadi et al., 2020).

As Fischer and Krauss (2018) indicate, each gate has two weight matrices, one for each source of information: either the input vector x_t or the previous hidden state $h_{t-1,}$ which are denoted as $W_{f,x}$, $W_{f,h}$, $W_{i,x}$, $W_{i,h}$, $W_{o,x}$ and $W_{o,h}$. Additionally, we also consider the weight matrices of the candidates of the cell states

$\widetilde{Ct}$, $W_{C,x}$ and $W_{C,h}$. The candidates represent the new information that needs to be added at time t, considering the previous hidden state h_{t-1} and the current input x_t:

$$\widetilde{Ct} = \tanh\left(W_{C,x}[h_{t-1}, x_t] + b_c\right) \quad (20)$$

Where b_c is the bias term.

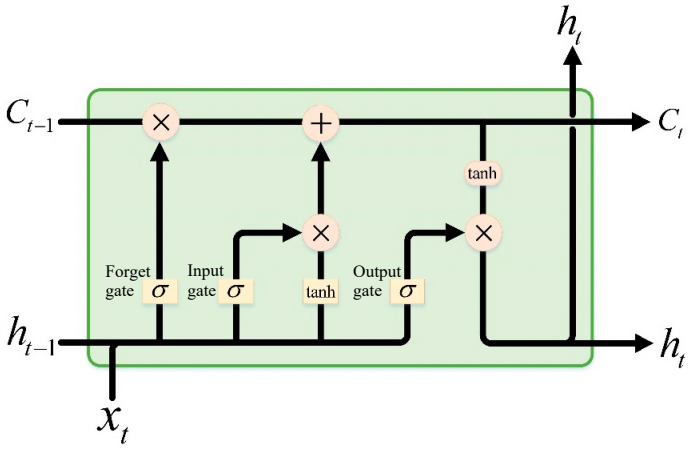


Figure 4.1: The structure of a long-short term memory cell.

Adapted source: Nosratabadi, S., Mosavi, A., Duan, P., Ghamisi, P., Filip, F., Band, S. S. & Gandomi, A. H. (2020). Data science in economics: comprehensive review of advanced machine learning and deep learning methods. *Mathematics*, *8*(10), 1799.

The hyperbolic tangent function below (eq. 21) is a function that scales down the values within the cell, to [-1,1] to prevent exploding or vanishing gradients:

$$\tanh(x) = \frac{\exp(x)-\exp(-x)}{\exp(x)+\exp(-x)} \quad (21)$$

The sigmoid function is defined as in eq. (11). The gates are the input gate i_t, the output gate g_t and the forget gate f_t, for every time t. The input gate uses the sigmoid function of sending the necessary data in the next state C_t:

$$i_t = \sigma\left(W_{i,x}[h_{t-1}, x_t] + b_i\right) \quad (22)$$

The forget gate filters the information and deletes what is not relevant for the next state:

$$f_t = \sigma\left(W_{f,x}[h_{t-1}, x_t] + b_f\right) \quad (23)$$

Lastly, the output gate has the role to send the final information to the next state (Fischer & Krauss, 2018; Nosratabadi et al., 2020):

$$o_t = \sigma\left(W_{o,x}[h_{t-1}, x_t] + b_o\right) \quad (24)$$

Following this, both the cell state (eq. 25) and the hidden state (eq. 26) need to be updated using the three gates:

$$C_t = f_t C_{t-1} + i_t \widetilde{C_t} \quad (25)$$

and

$$h_t = o_t \tanh(C_t)\,. \tag{26}$$

In this paper, the LSTM network assumes only one hidden layer containing 64 neurons and two dense layers. The activation function used is the rectifier linear unit (ReLU). The loss function that the model tries to optimise is the MSE, using the Adaptive Moment Estimation (Adam) optimiser. The input layer includes the weekly realised volatilities with their first three lagged values. During training, there are assumed to be 100 epochs, with a batch size of 64 and a learning rate of 0.001. The dropout ratio is 0.2, which indicates the rate of the random removal of connections between hidden layers. This method induces the model to depend on a varied range of nodes, discouraging overfitting. The training data (December 31st, 2003, to December 31st, 2019) includes 80% of the whole sample, taken in ascendent order of the dates. The validation data (January 1st, 2020, to December 31st, 2021) is the next 10%, while the testing data (January 1st, 2022, to December 31st, 2023) is the remaining 10% of the sample.

4.4 The Hybrid GARCH-LSTM

The model first predicts the values of conditional variance of the FTSE 100 returns using GARCH(1,1) during 2020-2022 and 2022-2024, respectively. The residuals are computed as the difference between the observed values of weekly realised variances and the predicted conditional ones from 2004 until 2020, and from 2004 until 2022. Next, these residuals are modelled using the same LSTM configuration as described above. The final output of the model is created by adding the fitted values of the predicted conditional variance of the GARCH and the forecasts of the residuals using the LSTM.

4.5 Random Forest: Algorithm and Model Specifications

Forecasting realised volatility using the Random Forest included two main steps: building decision trees and building the Random Forest. The Classification and Regression Trees (CART) algorithm is used for the former, as in Luong and Dokuchaev (2018) and explained in Appendix A (Table 4.1 and Table 4.2). The second, third and fourth lags of RV_t are used as features. Hence, Breiman's bagger parameter equals three. The selection was made by assessing the correlation of the last ten lags with the first one and choosing the three with the highest values. Then, different samples containing combinations of the realised volatility data are created and divided using random samples of the three lags into samples of in-bag- and out-of-bag-instances. The first out-of-sample forecast, the validation period (2020-2022), uses only data between 2004-2020, while the test period (2022-2024) also integrates the last two years (2004-2024). The objective function used is the MSE. At each node, the predictor variables that leads to a lower MSE is used for splitting the sample until all nodes are grown. This algorithm is presented in Appendix A, as well.

4.6 Forecast Evaluation

Poon and Granger (2003) reviewed different evaluation methods for volatility forecasts. West et al. (1993) use a utility-based criterion for rankings model performance in predicting the volatilities of exchange rates.

However, this requires several assumptions to be made for the form of the utility function, as, in reality, this cannot be known. Other statistical measures for forecast errors include the Mean Error (ME), the Mean Square Error (MSE) or the Mean Logarithm of Absolute Errors (MLAE). In this paper, we focus on the most popular metric, the Mean Squared Error. The aim of all candidate models is to minimise it. The formula for the MSE is given below (eq. 28):

$$MSE = \frac{1}{n}\sum_{i=0}^{n}(actual\ output - predicted\ output)^2 \tag{28}$$

It can be noticed that loss functions are symmetric functions. In some cases, this could be problematic, as positive and negative forecast errors might not be symmetric. Therefore, it is useful to determine whether the differences in the MSEs are accurately reflected, or merely a result of the particular dataset employed. Hence, the Diebold-Mariano Test is employed. The hypothesis of the test is that the predictive accuracy of two models are equal, while the alternative hypothesis is that they differ. In case the hypothesis is not rejected at 10% significance level, the MSEs are regarded as the main measurement to compare model accuracy. We apply the DM test to each pair of models for both out-of-sample periods. The test statistics can be computed using the following (eq. 29):

$$DM\ statisitc = \frac{\bar{d}}{\sqrt{Var(\bar{d})}}, \tag{29}$$

Where $\bar{d}$ is the average of the loss differentials of the two models that are compared and $Var(\bar{d})$ is the standard deviation.

CHAPTER 5 Results and Discussion

In this section, we present the results entailed by the model for the periods 2020-2022 and 2022-2024, followed by a discussion of their overall performance.

5.1 Prediction Results for 2020-2022

The period between 2020 and 2022 was regarded as the validation dataset for training the LSTM. The results of the model for this period are compared with the ones of the others. Table 5.1 shows the MSE scores of all candidate models in this study for predicting the realised volatility of the FTSE 100, sorted in alphabetic order.

Table 5.1 Mean-square-error scores of models' predictions between 2020 and 2022 and between 2022 and 2024.

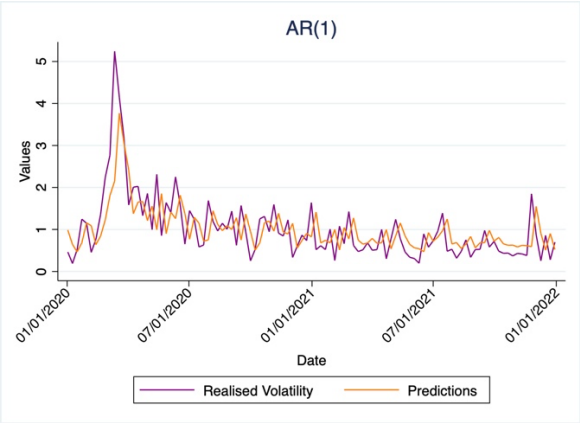
Model	**MSE 2020-2022**	**MSE 2022-2024**
AR(1)	**0.3195**	**0.1754**
ARCH(1)	2.0787	0.9377
ARMA(1,1)	0.3128	0.1856
GARCH-LSTM	1.3875	0.3465
GARCH(1,1)	2.0425	0.9373
GJR-GARCH(1,1)	2.0425	0.9296
HAR-RV.	0.3197	0.1764
LSTM	0.4138	0.2092
Random Forest	0.3397	0.2279

Note: The rows are sorted in alphabetical order. The first column states the characteristics of the models compared, while the second and third columns reveal the MSE scores for each of them, during the validation period (2020-2022) and the test period (2022-2024). The lowest MSEs are in bold.

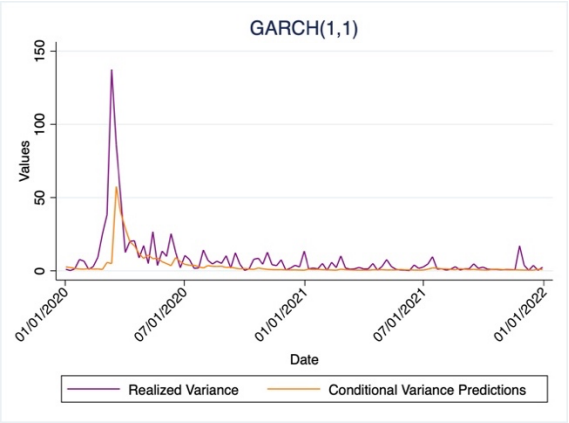
The results of the Diebold-Mariano Test are presented in Table 5.2 in Appendix A. The hypothesis that two models have equal forecasting accuracy was rejected for the following pairs of models at 10% significance level: AR(1) and LSTM, HAR-RV and GJR-GARCH(1,1), HAR-RV and LSTM, GARCH(1,1) and GJR-GARCH(1,1), GARCH(1,1) and RF, GARCH-LSTM and LSTM and for GARCH(1,1) and LSTM. For the rest of the pairs, there is insufficient information to conclude that their predictive accuracy differs at a significance level of 10%. This means that another performance measurement, such as the MSE, should be employed, as it can still provide valuable insights. The MSE shows how far away from the observed values the predicted ones are situated. This means that the model with the lowest MSE has the most accurate forecast. The best performing one was ARMA(1,1), with an MSE of 0.3128, followed by AR(1), with an

MSE of 0.3195, and HAR-RV, with an MSE of 0.3197. The ARCH-family models were found to have the poorest performance of all, while the machine learning models, the Random Forest (MSE of 0.3397) and the LSTM (MSE of 0.4138), had a standard performance. Fig. 5.1 (h) reveals the performance of the LSTM over the period. In comparison with the ARMA(1,1) and AR(1) (Fig. 5.1 (a) and Fig. 5.1 (b)), the predictions of the model depend on the training sample, with data from 2004 and 2020. The training period is characterized by many shocks, particularly between 2008 and 2012, which might have influenced the predictions of the LSTM over the validation period (2020-2022). In contrast, AR(1) and ARMA(1,1) models tend to be more robust to changes in market regimes because they focus on short-term dependencies and reflect the complex patterns of the long historical training period.

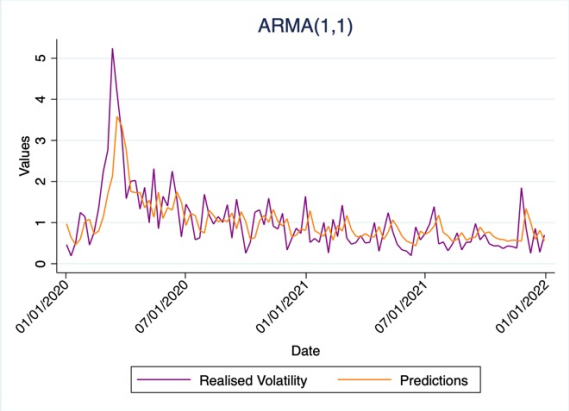
The HAR-RV considered the past values at both the weekly and monthly levels, which entailed a good performance for this specific period. The fluctuations are rather low, apart from the beginning of 2020, when the spike attributed to the COVID-19 pandemic can be observed (Fig. 5.1 (f)). The formal declaration of the pandemic was on March 20th, 2020, which corresponds to the peak in the dataset. Moreover, previous studies have shown that the realised volatility of indexes exhibits long memory, meaning that repercussions of past events can be felt much later after these have ended (Koopman et al., 2005; Bandi & Perron, 2006). As Baker et al. (2020) found that COVID-19 led to an unprecentedly high level of volatility in stock markets, it is expected that the effects of this shock will extend over a long period of time. HAR-RV is specifically designed to capture the heterogeneity of volatility series, and hence, it can seize specific stylised facts of volatility, such as mean-reverting behaviour. Over the period, it can easily be noticed that LSTM's prediction line does not constantly follow the trend of the observed one. Even though the model seems to follow the direction of the realised volatilities, it underestimates the magnitude of these changes, predicting much lower shocks. Given the contrasting results, Hypothesis 1, which stated that the LSTM outperforms all the other eight models employed for forecasting the 2020-2022 realised volatilities of the stock index, is only partially supported.



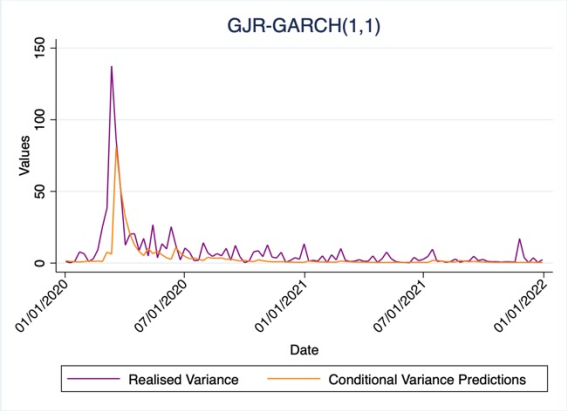
(a)



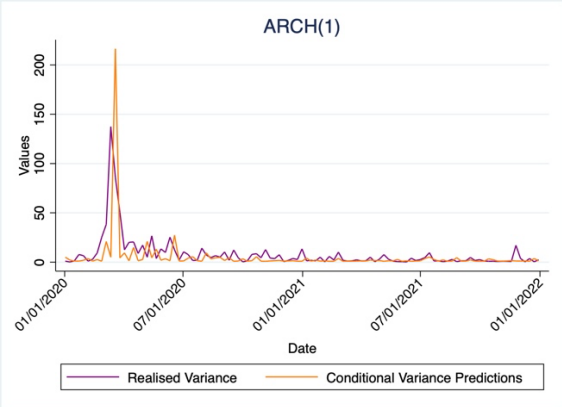
(d)



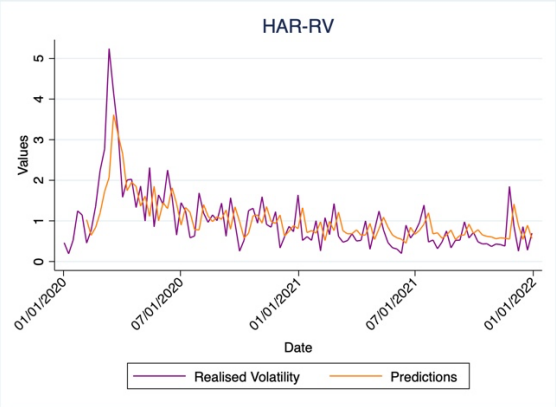
(b)



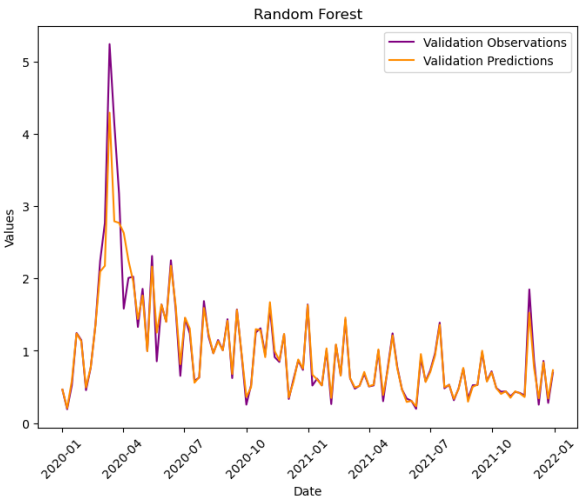
(e)



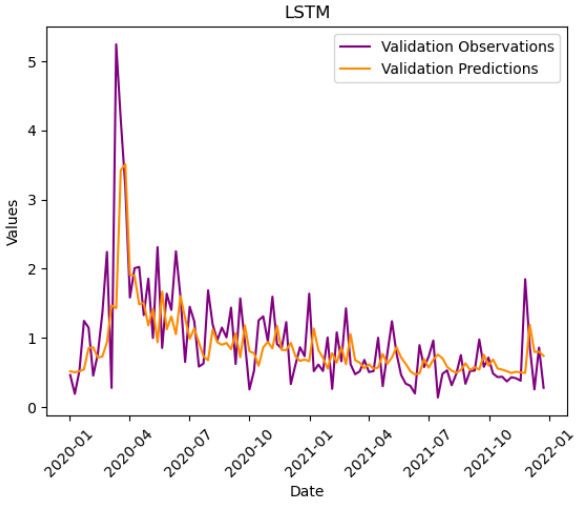
(c)



(f)



(g)



(h)

Figure 5.1 Realised volatility predictions between 2020 and 2022.

Note: The subplots are generated using the (a) AR(1) model, (b) ARMA(1,1) model, (c) ARCH(1) model, (d) GARCH(1,1) model, (e) GJR-GARCH(1,1), (f) HAR-RV, (g) Random Forest, (h) LSTM model. For ARCH(1), GARCH(1,1) and GJR-GARCH(1,1) models, the values reflect conditional variance forecasts. The horizontal axis includes all dates between December 31st, 2019, and December 31st, 2021, on a weekly basis. The vertical axis measures the range of values of the realised volatility in the subplots (a), (b), (f), (g), (h), and the range of values of the conditional and realised variance in the subplots (c), (d), (e).

Even though the LSTM is superior to the hybrid GARCH-LSTM in terms of MSE during this period, modelling the residuals of the GARCH(1,1) using this RNN drastically improved the performance of GARCH(1,1), from an MSE of 2.0425 to an MSE of 1.3875 (by 32.07%). Therefore, Hypothesis 3, which assumed that the hybrid GARCH-LSTM during the same period, is sustained. The predictions of the residuals using the LSTM during 2020-2022 seem to diverge from the observed values, especially after a negative shock and, similarly to the singular LSTM model, the magnitudes of the shocks are underestimated in the following timesteps (Fig. 5.2). This can be explained by the fact that the GARCH(1,1) is not a long-memory model and that the training sample for this model is computed by subtracting the raw values of weekly realised volatility from the fitted values of the GARCH(1,1) between 2004-2020. Hence, the prediction accuracy over the validation period might have been distorted by the effect of using a short-memory model, such as the GARCH, for longer periods of time.

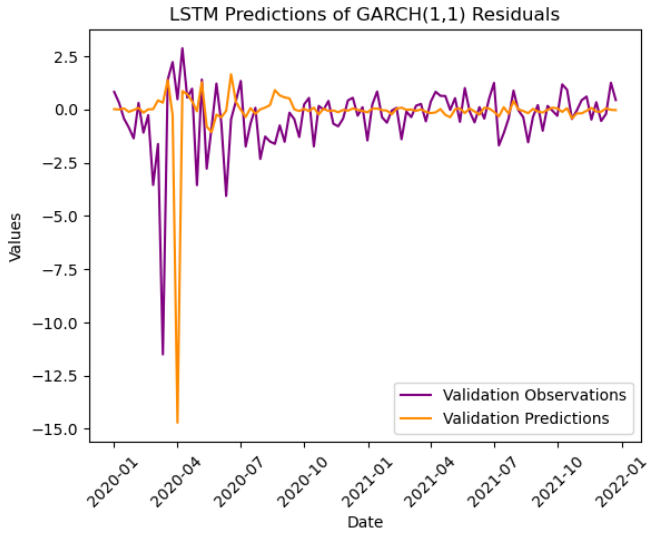


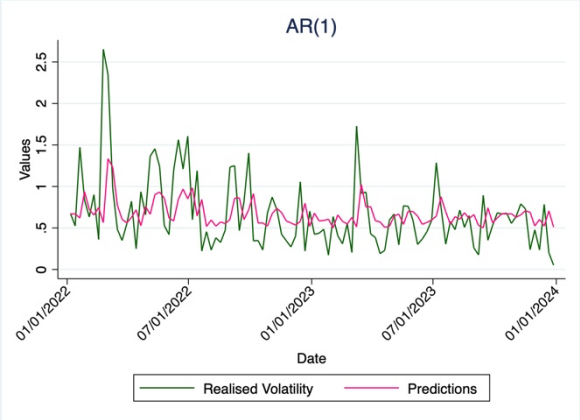
Figure 5.2 GARCH(1,1) residuals predictions generated using the LSTM model during the validation period, 2020-2022.

Note: The horizontal axis includes all dates between December 31st, 2019, and December 31st, 2021, on a weekly basis. The vertical axis measures the range of values of residuals. The residuals were computed as the difference between the raw values of weekly realised volatility of the FTSE 100 and the fitted values of the GARCH(1,1) on the dataset between 2004-2020, and forecasted out-of-sample between 2020-2022.

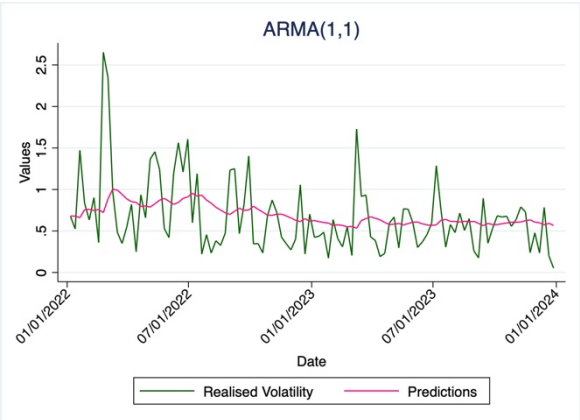
5.2 Prediction Results for 2022-2024

The performance of the models for the next period is described in the second column of Table 5.1. The results of the Diebold-Mariano Test are shown in Table 5.3 in Appendix A. The hypothesis that the forecasts of the models is equivalent was only rejected between RF and GJR-GARCH(1,1) and between RF and LSTM, at the 10% level. For the rest of the pairs there is insufficient evidence to support this conclusion. Similarly, we further use the MSEs to illustrate their relative performance. The period between 2022 and 2024 is the testing period in both Random Forest and LSTM networks. The best-performing model is the AR(1), with an MSE of 0.1754, followed by the HAR-RV (MSE of 0.1764) and ARMA(1,1) (MSE of

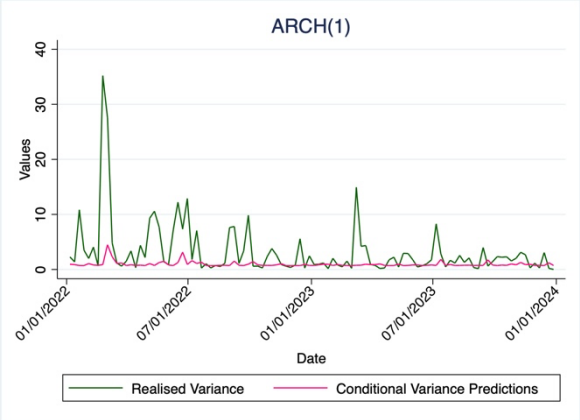
0.1856). Therefore, these three models consistently remain the most efficient ones, as in the validation period, outperforming the LSTM. Fig. 5.3 (a), (b) show that even though AR(1) and ARMA(1,1) models achieved good results, their prediction was not able to capture the full magnitude of the shocks. This problem can be caused by the non-normal distribution of residuals and the peaks throughout the dataset, which inflated AR's estimates.



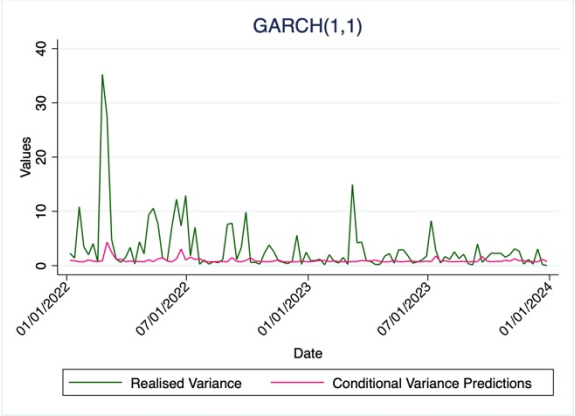
(a)



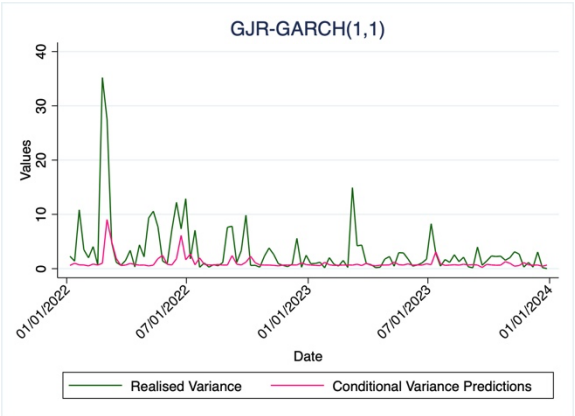
(b)



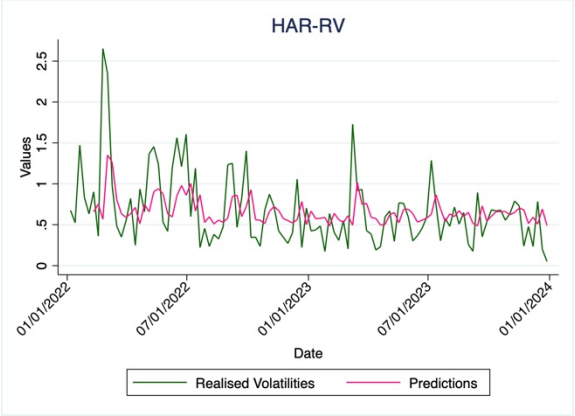
(c)



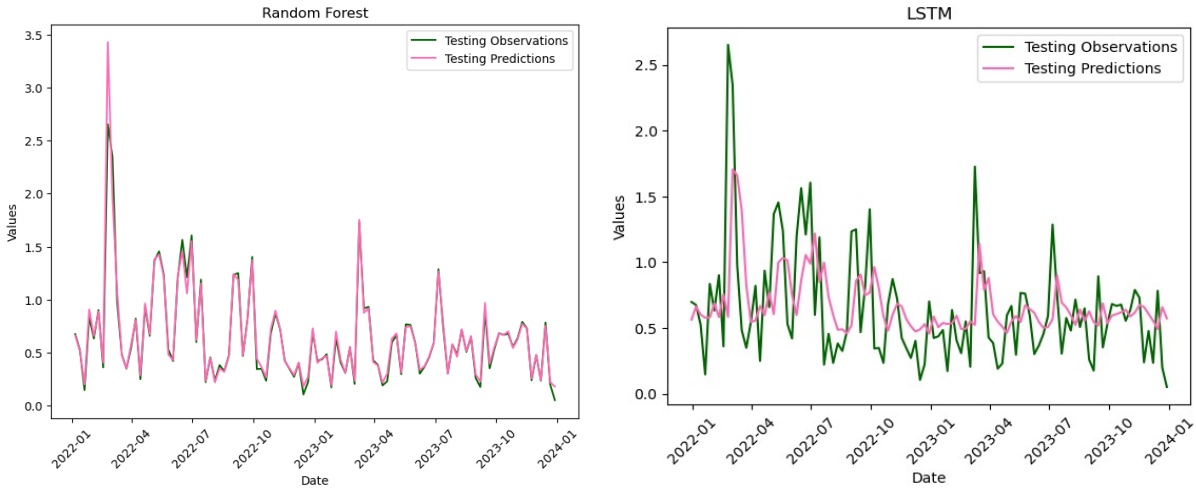
(d)



(e)



(f)



(g) (h)

Figure 5.3 Realised volatility predictions between 2022 and 2024.

Note: The subplots are generated using the (a) AR(1) model, (b) ARMA(1,1) model, (c) ARCH(1) model, (d) GARCH(1,1) model, (e) GJR-GARCH(1,1), (f) HAR-RV, (g) Random Forest, (h) LSTM model. For ARCH(1), GARCH(1,1) and GJR-GARCH(1,1) models, the values reflect conditional variance forecasts. The horizontal axis includes all dates between December 31st, 2021, and December 31st, 2023, on a weekly basis. The vertical axis measures the range of values of the realised volatility in the subplots (a), (b), (f), (g), (h), and the range of values of the conditional and realised variance in the subplots (c), (d), (e).

Between 2022 and 2024, the LSTM performed better than the Random Forest, which was not the case in the previous period. The Random Forest had an MSE of 0.2279, slightly lower than the one measured in the validation period. This can be explained by capturing specific patterns in the data and improving their predictions over the next two years, as depicted in Fig. 5.3 (g). Likewise, the LSTM boosted its performance during the testing period, having an MSE of 0.2092, less than half of the one measured before (Fig. 5.3 (h)). The market conditions in the two time frames include many volatility shocks, specifically with a spike at the beginning of each period. As the LSTM is designed to capture temporal dependencies and structures, the remarkably similar patterns over the validation and test periods can explain the considerable improvement in its performance.

The figure above (Fig. 5.3 (h)) depicts the time series of the testing period and its predictions using the LSTM. An important result of the model is that there is no overfitting, which can be reflected in a higher MSE in the testing period relative to the other ones, which was not the case. Even so, the LSTM did not react as expected when dealing with sudden shifts in the data, predicting them later than it was supposed to (i.e. it embraces a '*delayed'* aspect). Specifically, a shock at time *t* takes place in the next period, *t+1*.

Graphically, both prediction lines in Fig. 5.1 (h) and Fig 5.3 (h) capture the patterns in the observed data, but the former one is shifted to the right relative to the real values.

In addition, the worst performing models are the GJR-GARCH(1) (MSE of 0.9296), GARCH(1,1) (MSE of 0.9373) and ARCH(1) (MSE of 0.9377), occupying the same ranks as in the previous period (Fig. 5.3 (e), (d) and (c)). A notable difference between these three models and the rest is that they forecast conditional variance. This might not perfectly predict realised variance and hence lead to higher errors. Moreover, even though the ARCH-type models account for volatility clustering and asymmetries, such as leverage effects in the case of the GJR-GARCH, they do not capture the non-linearities present in the data. However, the GARCH-LSTM model does so by addressing the remaining noise. This improves the prediction error of the GARCH (1,1) by 63.03%, the model achieving an MSE of 0.3465. The performance of the LSTM to predict the noise in the residuals of the GARCH, between 2022 and 2024, is shown in Fig. 5.4, below. In contrast to the validation period, the performance of the model almost triples, even if the LSTM consistently underestimated the shocks over the period. The efficiency of the hybrid model might be due the predictive performance of the GARCH, as the MSE of the latter also drastically improved, from 2.0425 in the validation, to 0.9373, in the testing period.

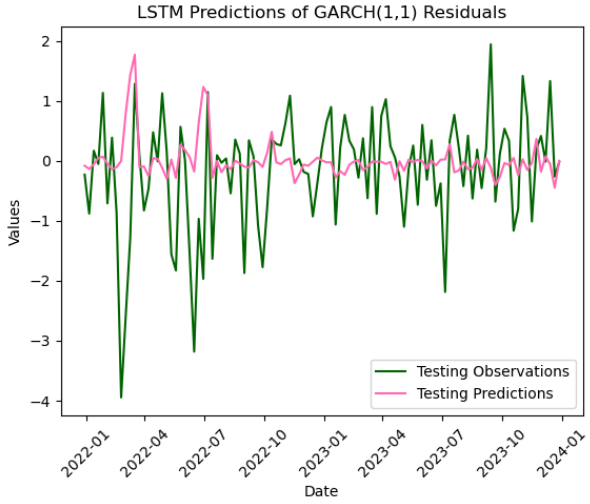


Figure 5.4 GARCH(1,1) residuals predictions generated using the LSTM model during the testing period, 2022-2024
Note: The horizontal axis includes all dates between December 31st, 2021, and December 31st, 2023, on a weekly basis. The vertical axis measures the range of values of residuals. The residuals were computed as the difference between the raw values of weekly realised volatility of the FTSE 100 and the fitted values of the GARCH(1,1) on the dataset between 2004-2022, and forecasted out-of-sample between 2022-2024.

Subsequently, we conclude that Hypothesis 2, affirming that the performance of the LSTM is superior to that of the other models in the test period, is not accepted only in the case of Autoregressive models. Hypothesis 4, that the benchmark GARCH(1,1) is inferior to the GARCH-LSTM model when forecasting 2022-2024 FTSE 100 realised volatilities is supported.

5.3 Discussion

The results of our study show the opposite of what was initially expected, that the LSTM could outperform the benchmark models at forecasting index return volatility, particularly at forecasting the realised volatility of the FTSE 100 index between 2020 and 2022, and between 2022 and 2024. Even though the number of financial studies using Neural Networks has been growing recently, not many are applied to the realised volatility of indexes. For instance, Yao, Luo and Peng (2018) used LSTMs to predict the short-term price movement of stocks selected from CSI 300. The precision, recall rate and critical error of the LSTM outperformed the ones of the other methods involved, such as the SVM, ANN, and Hidden Markov Model. As mentioned in Section 2.4.1, Fischer and Krauss (2018) found clear results that LSTM networks the classic models for predicting price movements of the S&P 500 index. However, Neural Networks are characterised by high complexity in their architecture, which can substantially impact their results. Aspects such as the choice of its features, batch size, dropout ratio or activation function directly influence the performance of the model, but no optimal design exists. Through this study, the selection of hyperparameters was by trial and error in order to find the best-fitting model, but changes in the number of layers, batch sizes or dropout ratios might lead to improvements in the results (Gal & Ghahramani, 2016).

Furthermore, other important insights should also be considered in this study, such as the high performance of the AR, HAR-RV, and ARMA models. The high performance of the HAR-RV model is in line with Corsi's findings (2004) that, in the foreign exchange market, the HAR-RV model successfully reproduces the main empirical features present in the volatility of financial data, steadily outperforming AR models. Other studies revealed a good performance of the AR models in forecasting return volatility. For instance, Pong et al. (2004) found that the ARMA model is superior to ARFIMA and GARCH in predicting realised volatilities of exchange rates, while Poterba and Summers (1986) successfully predicted the S&P 100 index between 1960 and 1985 using AR models. Additionally, the models ARCH, GARCH, and GJR-GARCH had an unexpectedly poor performance, as many studies employing these models had contrary results. For instance, Samouilhan and Shannon (2008) achieved great forecasts of volatility in the JSE 40 using the ARCH model, for a two-year period. Hansen and Lunde (2005) and Engle and Ng (1993) also showed the effectiveness of the GJR-GARCH(1,1) in comparison to other econometric ARCH models in predicting exchange rates and stock returns. In another study, the results revealed that the GARCH model led to better forecasts than the AR or NN models when forecasting volatility of six Asian stock indices (Hossain & Nasser, 2008).

On the other hand, there are also studies which sustain our results: Donaldson and Kamstra (1997), who tested the performance of an ANN model in comparison to GARCH variations on four stock indices (S&P 500, NIKKEI, FTSE and TSEC), found that the ANN outperformed. In addition, Chaudhuri and Ghosh (2016) showed that NNs lead to better forecasts than the GARCH and GJR-GARCH models, which supports the results as well. D'Ecclesia and Clementi (2021) reached the same conclusion when applying both the ANN and the GJR-GARCH model to the implied volatility of indices in China, Australia, Japan,

Italy, Germany, the UK and the USA. Furthermore, this research also introduced an innovative approach of improving the GARCH (1,1), by modelling its residual using the LSTM. This GARCH-LSTM hybrid model increased the performance of the benchmark by 32.07% and 63.03% for the validation and test periods, respectively. Kristjanpoller and Minutolo (2015; 2016) also emphasised the improvements in the predictive power of the GARCH by using this model for forecasting the volatility of gold and oil prices. These results are similar to the ones of Tseng et al. (2008) and Hajizadeh et al. (2012), who showed that by first removing the volatility structure using a GARCH model, the RNN model learns the more subtle, complex patterns in the residuals. Hence, the models complete each other's weaknesses leading to better predictive performance.

Finally, the results of the two simulations show contradicting results regarding the performance of the Random Forest relative to the LSTM. During 2020-2022, the Random Forest slightly outperformed the LSTM, while in the second period, the LSTM had substantially better results. Few papers examine the relative performance of the Neural Networks relative to the Random Forest. Ahmad et al. (2017) found that in the energy sector, the ANN outperformed, while in the field of landslide susceptibility, Sevgen et al. (2019) found similar results. In asset pricing, Sharma and Juneja (2017), Luong and Dokuchaev (2018), and Khaidem et al. (2016) compared the RF to various machine learning techniques, all concluding that it was the best-performing for forecasting financial data. These results highlight the idea that no consensus has been reached so far, as there are no straightforward instructions on how model specifications should be chosen.

The specific data sample used played an important role in shaping these results. In particular, the dataset comprises a single index, collecting the 100 largest UK companies by value. The largest sectors within the index are the energy, financial and consumer staples, while another index may attribute them different weights. Additionally, the index is a market-capitalisation-weighted index, meaning that companies with larger market capitalisation have a greater influence on its performance. This is aspect relates to the different up- or downturns in the value of the constituent stocks, due to either innovations or negative shocks in the market. Moreover, the returns of the FTSE 100 are highly sensitive to the specific market movements in the UK, which has suffered many economic shocks during the chosen period. Therefore, the results of the long-memory models, which used training data on 16 years of a UK index, cannot be generalisable for a different geographic location. Similarly, nor can they be for a different time period. Even though the models indicated roughly similar results between 2020-2022 and 2022-2024, it is essential to note that these periods were marked by many economic shocks and unprecedently highly volatile periods.

CHAPTER 6 Conclusion

In this thesis, we have assessed the performance of eight econometric models, specifically AR(1), ARMA(1,1), ARCH(1), GARCH(1,1), GJR-GARCH(1,1), HAR-RV, the Random Forest and the LSTM, forecasting the realised volatility of FTSE 100 index returns over the periods between 2020 and 2022 and between 2022 and 2024, respectively. Emerging research has indicated that applying machine learning methods to financial datasets for prediction purposes leads to efficient results, outperforming the standard models. One example of these is Neural Networks, such as the Long-Short-Term Memory Model, which many scholars pursue due to their ability to capture non-linearities. Hence, the research question of this study was *"How do the Autoregressive models (AR, ARMA and HAR-RV), ARCH-family models (ARCH, GARCH, GJR-GARCH and GARCH-LSTM) and machine learning models (Random Forest model LSTM model) perform in predicting weekly realised volatilities of the FTSE 100 index over the 2004-2020 training period and the 2020-2022 and 2022-2024 out-of-sample periods, and which one is the most efficient?"*

To answer this research question, the weekly realised volatilities were computed from FTSE 100 daily data between 2004 and 2024. The data was split into three disjoint periods: training (2004-2020), validation (2020-2022) and test (2022-2024) periods. Then, the performance of all models was compared for both the validation and the test data. Both analyses showed that AR, ARMA and the HAR-RV outperformed the LSTM, while the Random Forest outperformed it only during the validation period. Additionally, the LSTM led to substantially lower measurement errors (MSE) than the ARCH, GARCH and GJR-GARCH in both periods. Therefore, the first two hypotheses, which assumed that the LSTM could achieve lower prediction errors than all other models in both out-of-sample periods, are only partially sustained. However, it was also shown that, by joining the characteristics of the traditional GARCH(1,1) model with the ones of the LSTM, the hybrid model outperforms in both periods. Consequently, the last two hypothesis, namely Hypothesis 3 and Hypothesis 4, are confirmed.

Following these results, the performance of the LSTM for forecasting FTSE 100 realised volatilities is still open to question, as the design of its architecture could lead to better or poorer results, depending on the hyperparameter selections. However, this research sheds light on the effectiveness of the LSTM, specifically with the selected configuration, which outperformed almost half of the regarded standard models and, by modelling residuals, particularly improved the GARCH model. This shows the great potential of using this method in volatility forecasting and risk management.

6.1 Limitations

Although these findings seem promising for implementing machine learning in forecasting realised volatility, further emphasis should be placed on the network architecture and input variables. The input selection, data frequency, and data samples can heavily affect the performance of neural network models. In this research, the data frequency used was weekly. However, if the frequency increases (e.g., using intraday data to calculate daily realised volatilities), the model's performance will change and probably

decrease. Moreover, as we only used the past three lags as input variables in the training sample, other potential ones were omitted. Hence, the method could be improved using sensitivity analysis to assess the relevance of all variables and exclude the least important ones. Not only this, but the selected number of inputs is also an essential factor that might limit the results.

Additionally, the choice of other parameters and hyperparameters should be explored further. One such choice would be the ReLU activation function, which we used, and which can lead to the 'dying ReLU problem' (Lu et al., 2020). This happens when the output of the ReLU layer is constant for all inputs. Other activation functions (Leaky ReLU and ELU) could be used instead. Another limitation of the study stems from the fact that for the Autoregressive and ARCH-family of models, only the benchmark configurations were analysed, which include a single lag for their prediction. Similarly, the Random Forest only uses three predictor variables, while the LSTM uses four. However, more information from previous lags can be used to forecast volatilities. Other variables could be included as well, such as macroeconomic ones, treating the analysis as a cross-section.

6.2 Implications for Practitioners

Forecasting the realised volatility of an index could be a valuable tool for various specialists. One of the most common applications might be portfolio management, which is used to achieve desired risk and return levels. In this way, forecasts can be used to assess better the risk associated with each portfolio and hence to increase or decrease the exposure to the risky asset by implementing hedging strategies. Even though they might underestimate the magnitudes of the shocks, the study results show that, in the case of FTSE 100, the best-performing models were autoregressive ones, and, therefore, should be considered when evaluating realised volatility forecasts. Additionally, if the baseline model GARCH(1,1) is employed, practitioners should be mindful that this model cannot capture non-linearities in the dataset and that it uses conditional volatilities. According to the results, one improvement may be to use machine learning techniques, such as LSTM, to predict the model's residuals. Another application of the study is in the area of quantitative research, as professionals can use the results to back test trading strategies or develop trading algorithms. For instance, one could use the model's forecasts to create realistic scenarios to evaluate how strategies would have performed under different levels of volatility.

REFERENCES

Agatonovic-Kustrin, S., & Beresford, R. (2000). Basic concepts of artificial neural network (ANN) modeling and its application in pharmaceutical research. *Journal of pharmaceutical and biomedical analysis*, *22*(5), 717-727.

Ahmad, M.W., Mourshed, M., Rezgui, Y. (2017), Trees vs Neurons: Comparison between random Forest and ANN for high-resolution prediction of building energy consumption. Energy and Buildings, 147, 77-89.

Amaratunga, D., Cabrera, J., & Lee, Y.-S. (2008). Enriched Random Forests. *Bioinformatics*, 24(18), 2010–2014. doi: 10.1093/bioinformatics/btn356

Andersen, T. G., Bollerslev, T., Diebold, F. X., & Labys, P. (2001). The distribution of realised exchange rate volatility. *Journal of the American statistical association*, *96*(453), 42-55.

Baker, S. R., Bloom, N., Davis, S. J., Kost, K., Sammon, M., & Viratyosin, T. (2020). The unprecedented stock market reaction to COVID-19. *The review of asset pricing studies*, *10*(4), 742-758.

Ballings, Michel & Van den Poel, Dirk & Hespeels, Nathalie & Gryp, Ruben. (2015). Evaluating multiple classifiers for stock price direction prediction. *Expert Systems with Applications*. 42. 10.1016/j.eswa.2015.05.013.

Bandi, F. M., & Perron, B. (2006). Long memory and the relation between implied and realized volatility. *Journal of Financial Econometrics*, *4*(4), 636-670.

Bollerslev, T. (1986). Generalised Autoregressive conditional heteroscedasticity. *Journal of Econometrics*, 31(3), 307–327.

Bollerslev, T. (2008). Glossary to arch (garch). *CREATES Research paper*, *49*.

Breiman, L. (2001). Random Forests. *Machine Learning*, 45(1), 5–32. doi: 10.1023/A:1010933404324

Breiman, L. (2017). *Classification and regression trees*. Routledge.

Brooks, C. (2019). *Introductory econometrics for finance (*pp. 340-341, 351-352, 508-509, 514, 522*)*. Cambridge university press.

Chaudhuri, T. D., & Ghosh, I. (2016). Artificial neural network and time series modelling based approach to forecasting the exchange rate in a multivariate framework. *arXiv preprint arXiv:1607.02093*.

Chelley-Steeley, P. L., & Steeley, J. M. (1996). Volatility, leverage and firm size: the UK evidence. *The Manchester School*, *64*(S1), 83-103

Chelley-Steeley, P. L., & Steeley, J. M. (2005). The leverage effect in the UK stock market. *Applied Financial Economics*, *15*(6), 409–423. doi: 10.1080/0960310052000337669

Chen, Y., Han, Q., & Niu, L. (2018). Forecasting the term structure of option implied volatility: The power of an adaptive method. *Journal of Empirical Finance*, *49*, 157-177

Corsi, F. (2004). A simple long memory model of realised volatility. *Available at SSRN 626064*.

Costa, F. J. M. (2017). *Forecasting volatility using GARCH models* (p. 1).(Master's thesis, Universidade do Minho (Portugal)).

D'Ecclesia, R. L., & Clementi, D. (2021). Volatility in the stock market: ANN versus parametric models. *Annals of Operations Research*, *299*(1), 1101-1127

Diane, L., & Brijlal, P. (2024). Forecasting stock market realized volatility using random forest and artificial neural network in South Africa. *International Journal of Economics and Financial Issues*, *14*(2), 5-14.

Donaldson, R. G., & Kamstra, M. (1996). Forecast combining with neural networks. *Journal of Forecasting*, *15*(1), 49-61.

Donaldson, R. G., & Kamstra, M. (1997). An artificial neural network-GARCH model for international stock return volatility. *Journal of Empirical Finance*, *4*(1), 17-46.

Engle, R. F. (1982). Autoregressive conditional heteroscedasticity with estimates of the variance of United Kingdom inflation. *Econometrica: Journal of the econometric society*, 987-1007

Engle, R. F., & Bollerslev, T. (1986). Modelling the persistence of conditional variances. *Econometric reviews*, *5*(1), 1-50.

Engle, R. F., & Ng, V. K. (1993). Measuring and testing the impact of news on volatility. *The journal of finance*, *48*(5), 1749-1778.

Engle, R. F., & Patton, A. J. (2007). What good is a volatility model?. *In Forecasting volatility in the financial markets* (pp. 47-63). Butterworth-Heinemann.

Fawagreh, K. (2016). *On pruning and feature engineering in Random Forests* (Doctoral dissertation).

Fischer, T., & Krauss, C. (2018). Deep learning with long short-term memory networks for financial market predictions. *European journal of operational research*, *270*(2), 654-669.

Fister, D., Mun, J. C., Jagrič, V., & Jagrič, T. (2019). Deep learning for stock market trading: a superior trading strategy?. *Neural Network World, (3).*

Franses, P. H., & Van Dijk, D. (1996). Forecasting stock market volatility using (non-linear) Garch models. *Journal of forecasting*, *15*(3), 229-235.

Gal, Y., & Ghahramani, Z. (2016). Dropout as a bayesian approximation: Representing model uncertainty in deep learning. In *international conference on machine learning* (pp. 1050-1059). PMLR.

Ge,W., Lalbakhsh,P., Isai,L., Lenskiy,A. & Suominen, A. (2022). Neural Network–Based Financial Volatility Forecasting: A Systematic Review. *ACM Comput*. Surv. 55, 1, Article 14 (January 2023), 30 pages. https://doi.org/10.1145/3483596

Glosten, R., Jagannathan, R., & Runkle, E. (1993). On the relation between the expected value and the volatility of the nominal excess return on stocks. *Journal of Econometrics*, 48(5), 1779–1801.

Gu, S., Kelly, B., Xiu, D. (2020). Empirical Asset Pricing via Machine Learning, *The Review of Financial Studies*, Volume 33, Issue 5, May 2020, Pages 2223–2273, https://doi.org/10.1093/rfs/hhaa009

Hajizadeh, E., Seifi, A., Zarandi, M. F., & Turksen, I. B. (2012). A hybrid modeling approach for forecasting the volatility of S&P 500 index return. *Expert Systems with Applications*, *39*(1), 431-436.

Hansen, P. R., & Lunde, A. (2005). A forecast comparison of volatility models: does anything beat a GARCH(1, 1)?. *Journal of applied econometrics*, *20*(7), 873-889.

Herculano-Houzel, S. (2009) The human brain in numbers: a linearly scaled-up primate brain. Front Hum Neurosci, 9;3:31. doi: 10.3389/neuro.09.031.2009. PMID: 19915731; PMCID: PMC2776484.

Ho, T. K. (1995). Random decision Forests. In *Document analysis and recognition, 1995, Proceedings of the third international conference, Montreal, Quebec, Canada* (Vol. 1, pp. 278–282). New York City, NY: IEEE.

Hochreiter, S., and Schmidhuber, J. (1997). Long short-term memory (p. 1). *Neural Computation* 9, 8 (Nov. 1997),1735-1780. https://doi.org/10.1162/neco.1997.9.8.1735

Hornik, K., Stinchcombe, M. and White, H. (1989). Multilayer Feedfor- ward Networks are Universal Approximators. *Neural Networks*, 2(5):359–366.

Hossain, A., & Nasser, M. (2008). Comparison of GARCH and neural network methods in financial time series prediction. In *2008 11th International Conference on Computer and Information Technology* (pp. 729-734). IEEE.

Hwang, S., & Satchell, S. E. (1999). Modelling emerging market risk premia using higher moments. *International Journal of Finance & Economics*, *4*(4), 271-296.

Jeong*, D. H., Kim, S. E., Choi, W. H., & Ahn, S. H. (2022).* A comparative study on the influence of undersampling and oversampling techniques for the classification of physical activities using an imbalanced accelerometer dataset*. In Healthcare (Vol. 10, No. 7, p. 1255). MDPI*

Joarder, K., Rezaul, K., & Ruhul, A. (2006). Artificial neural networks in finance and manufacturing (p. 48). *Paris: Idea Group Inc.*

Kathuria, A. (2018). *Intro to optimization in deep learning: Gradient Descent.* Paperspace*.* Retrieved from: *https://blog.paperspace.com/intro-to-optimization-in-deep-learning-gradient-descent/*

Khaidem, L., Saha, S., & Dey, S. R. (2016). Predicting the direction of stock market prices using Random Forest. *arXiv preprint arXiv:1605.00003*.

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

Koopman, S. J., Jungbacker, B., & Hol, E. (2005). Forecasting daily variability of the S&P 100 stock index using historical, realised and implied volatility measurements. *Journal of Empirical Finance*, *12*(3), 445-475.

Kristjanpoller, W., Fadic, A., & Minutolo, M. C. (2014). Volatility forecast using hybrid neural network models. *Expert Systems with Applications*, *41*(5), 2437-2442.

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

Kristjanpoller, W., & Minutolo, M. C. (2016). Forecasting volatility of oil price using an artificial neural network-GARCH model. *Expert Systems with Applications*, *65*, 233-241.

Louzis, D. P., Xanthopoulos-Sisinis, S., & Refenes, A. P. (2012). Stock index realized volatility forecasting in the presence of heterogeneous leverage effects and long range dependence in the volatility of realized volatility. *Applied Economics*, *44*(27), 3533-3550.

Lu, L., Shin, Y., Su, Y., & Karniadakis, G. E. (2019). Dying relu and initialization: Theory and numerical examples. *arXiv preprint arXiv:1903.06733*.

Luong C, Dokuchaev N. (2018). Forecasting of Realised Volatility with the Random Forests Algorithm. *Journal of Risk and Financial Management*.; 11(4):61. https://doi.org/10.3390/jrfm11040061

Medland, A. (2015, Jan. 15) *Financial Services Is Drowning In Its Own Complexity*. Forbes. Retrieved from: https://www.forbes.com/sites/dinamedland/2015/01/21/financial-services-is-drowning-in-its-own-complexity/

Moon, K.S., & Kim, H. (2019) Performance of deep learning in prediction of stock market volatility. *Econ. Comput. Econ. Cybern. Stud. Res*., *53*, 77–92.

Muller, U., Dacorogna, M., Dav, R., Pictet, O., Olsen, R. & Ward, J., (1993). Fractals and Intrinsic Time - A Challenge To Econometricians. XXXIXth International AEA Conference on Real Time Econometrics, Luxembourg.

Nelson, D. (1991). Conditional Heteroskedasticity in asset returns: A new approach. *Econometrica*, 59(2), 347–370. https://doi.org/10.2307/2938260

Nelson, D. M. Q., Pereira, A. C. M., & de Oliveira, R. A. (2017). Stock market's price movement prediction with LSTM neural networks. Paper presented at the 2017 International Joint Conference on Neural Networks (IJCNN), 1419–1426.

Nosratabadi, S., Mosavi, A., Duan, P., Ghamisi, P., Filip, F., Band, S. S. & Gandomi, A. H. (2020). Data science in economics: comprehensive review of advanced machine learning and deep learning methods. *Mathematics*, *8*(10), 1799.

Pagan, A. R., & Schwert, G. W. (1990). Alternative models for conditional stock volatility. *Journal of econometrics*, *45*(1-2), 267-290.

Pong, S., Shackleton, M. B., Taylor, S. J., & Xu, X. (2004). Forecasting currency volatility: A comparison of implied volatilities and AR(FI) MA models. *Journal of Banking & Finance*, *28*(10), 2541-2563.

Poon, S. H., & Granger, C. W. J. (2003). Forecasting volatility in financial markets: A review. *Journal of economic literature*, *41*(2), 478-539.

Poterba, J. M., & Summers, L. H. (1986). The Persistence of Volatility and Stock Market Fluctuations. *The American Economic Review*, *76*(5), 1142–1151. http://www.jstor.org/stable/1816476

Robnik-Šikonja, M. (2004). Improving Random Forests. In J.-F. Boulicaut, F. Esposito, F. Giannotti, & D. Pedreschi (Eds.), *Machine Learning: ECML 2004, 15th European Conference on Machine Learning, Pisa, Italy, 2004 Proceedings*, Lecture Notes in Computer Science (pp. 359–370). Berlin: Springer.

Samouilhan, N. L., & Shannon, G. (2008). Forecasting volatility on the JSE. *Investment Analysts Journal*, *37*(67), 19-28.

Satchell, S., & Knight, J. (2011). *Forecasting volatility in the financial markets*. Elsevier.

Sevgen, E., Kocaman, S., Nefeslioglu, H.A., Gokceoglu, C. (2019), A novel performance assessment approach using photogrammetric techniques for landslide susceptibility mapping with logistic regression, ANN and Random Forest. *Sensors*, *19*(18), 3940.

Sharma, N. & Juneja, A. (2017), Combining Random Forest Estimates Using LSboost for Stock Market Index Prediction. In: 2017 2nd International Conference for Convergence in Technology (I2CT). United States: IEEE, p1199-1202.

Tamura, K., Uenoyama, K., Iitsuka, S., & Matsuo, Y. (2018). Model for evaluation of stock values by ensemble model using deep learning.

Tang, Y., Ma, F., Zhang, Y., & Wei, Y. (2022). Forecasting the oil price realized volatility: A multivariate heterogeneous autoregressive model. *International Journal of Finance & Economics*, *27*(4), 4770-4783.

Taylor, S. J. (1986). Modelling Financial Time Series. Wiley, Chichester.

Tripathy, N., & Garg, A. (2013). Forecasting stock market volatility: Evidence from six emerging markets. *Journal of International Business and Economy*, *14*(2), 69-93.

Tseng, C. H., Cheng, S. T., Wang, Y. H., & Peng, J. T. (2008). Artificial neural network model of the hybrid EGARCH volatility of the Taiwan stock index option prices. *Physica A: Statistical Mechanics and its Applications*, *387*(13), 3192-3200.

Tsymbal, A., Pechenizkiy, M., & Cunningham, P. (2006). Dynamic integration with Random Forests. In J. Fürnkranz, T. Scheffer, & M. Spiliopoulou (Eds.), *Machine Learning: ECML 2006, 17th European Conference on Machine Learning, Berlin, Germany, 2006 Proceedings*, Lecture Notes in Computer Science (pp. 801–808). Berlin: Springer

Vilasuso, J. (2002). Forecasting exchange rate volatility. *Economics Letters*, *76*(1), 59-64.

Wang, W., Li, W., Zhang, N., & Liu, K. (2020). Portfolio formation with preselection using deep learning from long-term financial data. *Expert Systems with Applications, 143, 113042.*

West, K. D., Edison, H. J., & Cho, D. *(*1993*).* A utility-based comparison of some models of exchange rate volatility. *Journal of international economics, 35(*1-2*), 23-45.*

Yao, S., Luo, L., & Peng, H. (2018). High-frequency stock trend forecast using LSTM model. In *2018 13th International Conference on Computer Science & Education (ICCSE)* (pp. 1-4). IEEE.

Zakoian, J. (1994). Threshold heteroskedastic models. *Journal of Economic Dynamics & Control*, 18(5), 931–955. https://doi.org/10.1016/0165-1889(94)90039-6

Zumbach, G. (2003). Volatility processes and volatility forecast with long memory. *Quantitative Finance*, *4*(1), 70.

Appendix A

1. Gradient descent optimisation

To compute the gradient of the cost function, we define the following Taylor expansion:

$$Cost(\mathrm{p} + \Delta\,\mathrm{p}) \approx \mathrm{Cost(p)} + \sum_{r=1}^{s} \frac{\partial Cost(p)}{\partial p_r} \Delta\,\mathrm{p_r} \tag{12.1}$$

From which we write:

$$\nabla\big(Cost(p)\big)_r = \frac{\partial Cost(p)}{\partial p_r} \tag{12.2}$$

where $\nabla\big(Cost(p)\big)_r$ is the vector of partial derivatives, the *gradient*, which shows the direction of the steepest increase (ascent) of the cost function. We can write eq. (12.1) using (12.2):

$$Cost(\mathrm{p} + \Delta\,\mathrm{p}) \approx \mathrm{Cost(p)} + \nabla\big(Cost(p)\big)^{\mathrm{T}} \Delta\,\mathrm{p_r} \tag{12.3}$$

Where $\nabla\big(Cost(p)\big)^{\mathrm{T}}$ must be as small as possible to minimise the cost function. By applying the *Cauchy-Schwarz Theorem*, it can be shown that Δ p must be chosen in the direction of -$\nabla\big(Cost(p)\big)^{\mathrm{T}}$, the opposite direction of the ascent (e.g. descent). Then the value *p* is recursively updated as:

$$p \leftarrow p - \eta\nabla\big(Cost(p)\big) \tag{12.4}$$

Where $\eta > 0$ is the *learning rate,* which inidcates the magnitude of the step towards the steepest descent. The process is illustrated in Figure 2.3, where even though there exists a global minimum, the cost function might only reach the local one.

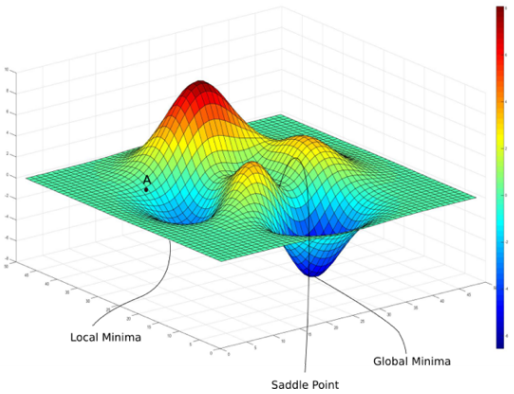


Figure 2.3 Gradient descent optimisation.

Source: Kathuria, A. (2018). *Intro to optimization in deep learning: Gradient Descent.* Paperspace. https://blog.paperspace.com/intro-to-optimization-in-deep-learning-gradient-descent/

2. Random Forest Algorithms

The algorithms used follow the pseudocode provided by Luong and Dukuchaev (2018) and included below for clarification.

Table 4.1 CART algorithm for building decision trees

Algorithm 1: CART
1: Let N be the root node with all available data.
2: Find the feature F and threshold value T that split the samples assigned to N into subsets $ITRUE$ and $IFALSE$, to maximise the label purity within these subsets.
3: Assign the pair (F, T) to N.
4: If $I(s)$ is too small to be split, attach a 'child' leaf node to $LTRUE$ and $LFALSE$ to N and assign the leaves with the most present label in $ITRUE$ and $IFALSE$, respectively. If subset $I(s)$ is large enough to be split, attach child nodes $NTRUE$ and $NFALSE$ to N, and then assign $I(s)$ to them, respectively.
5: Repeat steps 2–4 for the new nodes N=$NTRUE$ and N=$NFALSE$ until the new subsets can no longer be split.

Source: Luong, C., & Dokuchaev, N. (2018). Forecasting of realised volatility with the random forests algorithm (p. 3). *Journal of Risk and Financial Management*, *11*(4), 61.

Table 4.2 Random Forests algorithm

Algorithm 2: Random Forests
1: Draw a number of bootstrap samples from the original data ($ntree$) to be grown.
2: Sample N cases at random with replacement to create a subset of the data. The subset is then split into in-bag and out-of-bag samples at a selected ratio (i.e., 7:3).
3: At each node, for a preselected number m, m predictor variables ($mtry$) are chosen at random from all the predictor variables.
4: The predictor variable that provides the best split, according to some objective function, is used to build a binary split on that node.
5: At the next node, choose another m variables at random from all predictor variables.
6: Repeat 3–5 until all nodes are grown.

Source: Luong, C., & Dokuchaev, N. (2018). Forecasting of realised volatility with the random forests algorithm (p. 4). *Journal of Risk and Financial Management*, *11*(4), 61.

3. Diebold-Mariano Test Results

Table 5.2 Diebold-Mariano Test Results for Test Period Predictions using MSE, 2022-2024

Model	AR(1)	ARMA(1,1)	HAR-RV	ARCH(1)	GARCH(1,1)	GJR-GARCH(1,1)	GARCH(1,1)-LSTM	RF	LSTM
AR(1)		1.488	1.823	1.456	1.823	1.823	1.822	1.823	1.823*
ARMA(1,1)			7.159	-1.553	7.158	7.159	7.141	7.134	7.161
HAR-RV				6.754	-2.481	1.983*	4.444	1.956	2.197**
ARCH(1)					6.754	6.754	6.740	6.756	6.756
GARCH(1,1)						-3.163***	-5.026	-2.723**	-2.913***
GJR-GARCH(1,1)							-4.205	-1.461	-1.417
GARCH(1,1)-LSTM								6.188	3.887***
RF									1.057

Note: The values provided are the DM-Test statistics. *Significant at the 10% level, **Significant at the 5% level, ***Significant at the 1% level.

Table 5.3 Diebold-Mariano Test Results for Test Period Predictions using MSE, 2022-2024

Model	AR(1)	ARMA(1,1)	HAR-RV	ARCH(1)	GARCH(1,1)	GJR-GARCH(1,1)	GARCH(1,1)-LSTM	RF	LSTM
AR(1)		-0.292	1.158	-1.433	1.158	1.158	1.158	1.158	1.158
ARMA(1,1)			1.134	-1.141	1.134	1.134	1.134	1.134	1.134
HAR-RV				-1.268	-1.466	-0.930	-4.443	3.047	0.524
ARCH(1)					1.268	1.268	1.268	1.268	1.268
GARCH(1,1)						0.309	-4.173	5.906	1.165
GJR-GARCH(1,1)							-4.490	2.834**	1.207
GARCH(1,1)-LSTM								-5.045	-4.496
RF									2.056*

Note: The values provided are the DM-Test statistics. *Significant at the 10% level, **Significant at the 5% level, ***Significant at the 1% level.

4. Dickey-Fuller Test Results

Table 5.4 Dickey-Fuller Test Results for the returns and realised volatilities of the FTSE 100, 2004-2024

Variable	DF-Statistic
Return	-32.436***
RV	-14.583***

Note: The values provided are the DF Test Statistics with constant for the log-returns and realised volatility of FTSE 100 between 2004 and 2024. DF critical values are: -3.430 at 1% significance levels, -2.860 at 5% and -2.570 at 10%. The hypothesis that the series are stationary is rejected if the DF-Statistic is below these thresholds. *Significant at the 10% level, **Significant at the 5% level, ***Significant at the 1% level.

5. Models' predictions between 2004-2024

An overview of the prediction of the LSTM over all three sets (training, validation and testing) is depicted in Fig. 5.5, which also includes the recursive prediction over the 2020-2024 out-of-sample timeframe. For completeness, we include the predications of the short-memory models (AR(1), ARMA(1,1), ARCH(1), GARCH(1,1) and GJR-GARCH(1,1)) and of HAR-RV over the whole sampled period (Fig. 5.6).

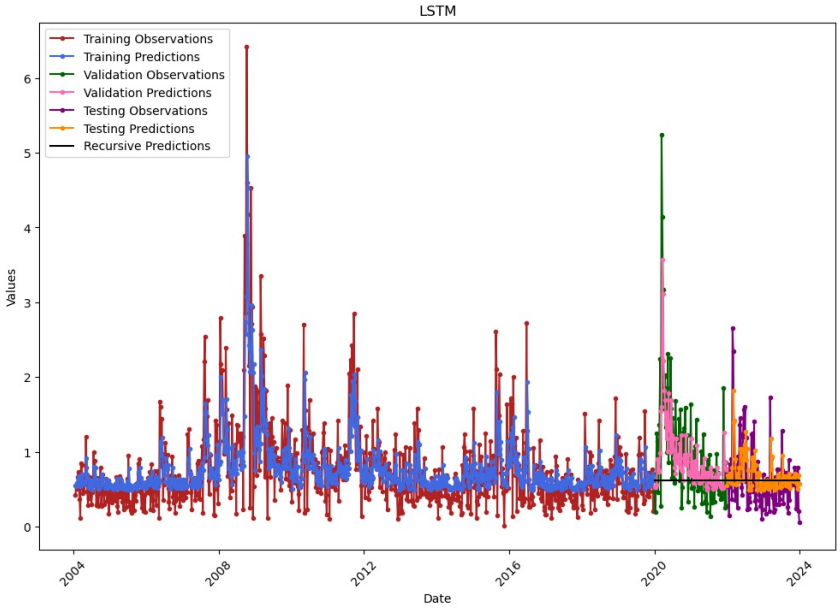
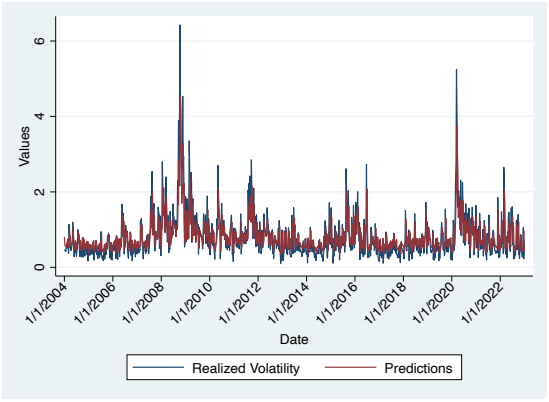
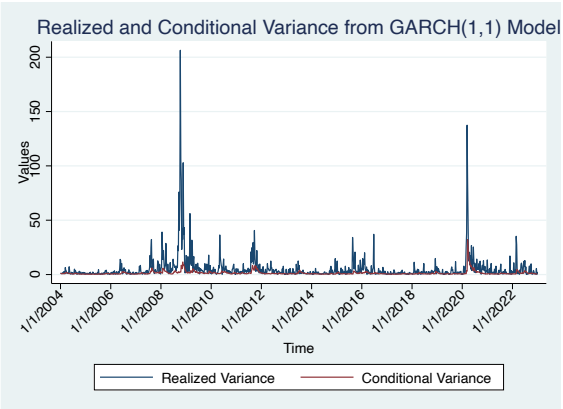


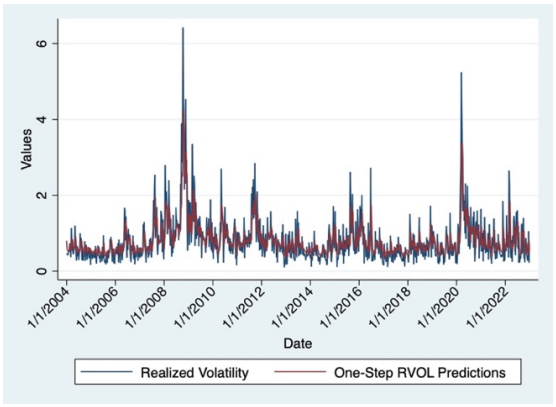
Figure 5.5 FTSE 100 Realised volatility predictions between 2004 and 2024, generated using the LSTM model.



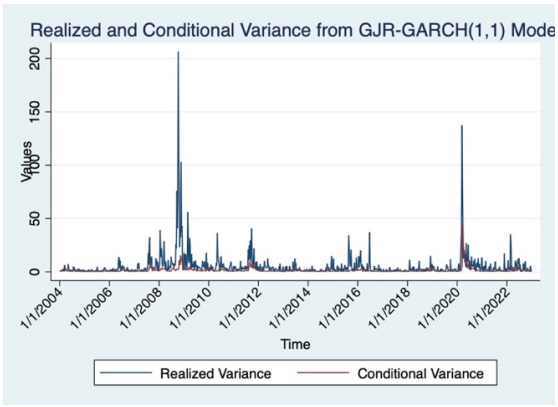
(a)



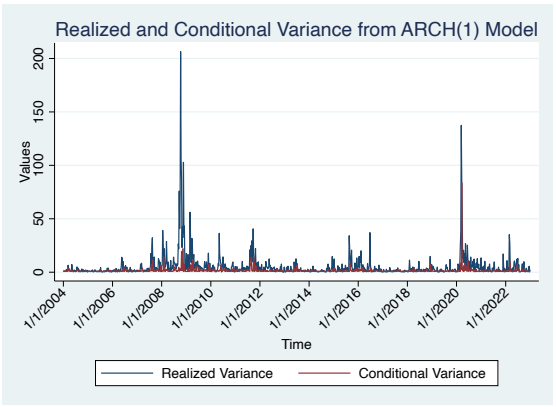
(d)



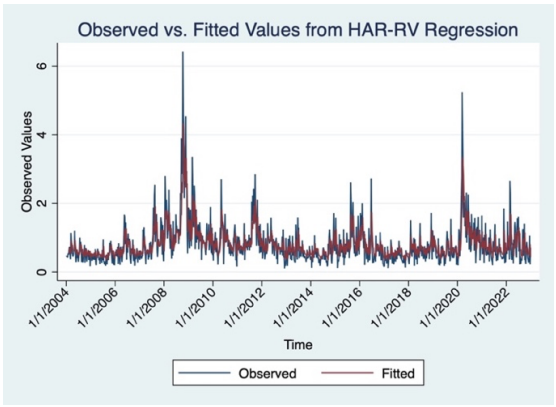
(b)



(e)



(c)



(f)

Figure 5.6 FTSE 100 Realised volatility predictions for the whole period between 2004 and 2024.

Note: The subplots are generated using the (a) AR(1) model, (b) ARMA(1,1) model, (c) ARCH(1) model, (d) GARCH(1,1) model, (e) GJR-GARCH(1,1), (f) HAR-RV. For ARCH(1), GARCH(1,1) and GJR-GARCH(1,1) models, the values reflect conditional variance forecasts. The horizontal axis includes all dates between December 31st, 2003, and December 31st, 2023, on a weekly basis. The vertical axis measures the range of values of the realised volatility in the subplots (a), (b), (f) and the range of values of the conditional and realised variance in the subplots (c), (d), (e).