Computational Economics (2024) 63:1511–1542
https://doi.org/10.1007/s10614-023-10373-8

LSTM–GARCH Hybrid Model for the Prediction of Volatility in Cryptocurrency Portfolios

Andrés García-Medina[1,2] · **Ester Aguayo-Moreno**[1]

Accepted: 19 February 2023 / Published online: 14 March 2023

Abstract

In the present work, the volatility of the leading cryptocurrencies is predicted through generalised autoregressive conditional heteroskedasticity (GARCH) models, multilayer perceptron (MLP), long short-term memory (LSTM), and hybrid models of the type LSTM and GARCH, where parameters of the GARCH family are included as features of LSTM models. The study period covered the scenario of the World Health Organization pandemic declaration around March 2020 at hourly frequency. We have found that the different variants of deep neural network models outperform those of the GARCH family in the sense of the hetorerocedastic error, and absolute and squared error (HSE). Under the sharpe ratio, the volatility forecasting of a uniform portfolio at long horizons systematically outperforms the stablecoin Tether, which is considered here as the risk-free asset. Also, including transaction volume helps reduce the value at risk or loss probability for the uniform portfolio. Moreover, in a minimum variance portfolio, it is observed that before the pandemic declaration, a large proportion of the capital was allocated to bitcoin (BTC). In contrast, after March 2020, the portfolio is more diversified with short positions for BTC. Moreover, the MLP models give the best predictive results, although not statistically different in accuracy compared to the LSTM and LSTM–GARCH versions under the Diebold–Mariano test. In sum, MLP models outperform most stylised financial models and are less computationally expensive than more complex neural networks. Therefore, simple learning models are suggested in highly non-linear time series volatility forecasts as it is the cryptocurrency market.

Keywords Cryptocurrencies · GARCH–LSTM models · Volatility

✉ Andrés García-Medina
andres.garcia@cimat.mx

Ester Aguayo-Moreno
ester.aguayo@cimat.mx

[1] Unidad Monterrey, Centro de Investigación en Matemáticas, Alianza Centro 502, 66628 Apodaca, Nuevo Leon, Mexico

[2] Consejo Nacional de Ciencia y Tecnología, Insurgentes Sur 1582, 03940 Mexico City, Mexico

1 Introduction

Given the recent technological revolution, virtual currencies have gained tremendous popularity with individual investors and have increasingly positioned themselves as one of the most attractive assets for institutional investors. A cryptocurrency is a digital asset encrypted with blockchain technology (Nakamoto, 2008). Given that the issuance and transaction of these assets are carried out through decentralised networks, they are theoretically immune to interference or manipulation by an authority or government. In particular, this feature makes them difficult to counterfeit or to spend twice.

Cryptocurrencies have an advantage over fiat money in portability since they can be used in international operations with great speed and low cost. In addition, they tend to resist inflation, so they have been considered safe-haven assets. One of their main characteristics is the variability they present over short periods and their atypical behaviour concerning traditional markets. In this study, it is preferred to model their challenging volatility behaviour.

Forecasting the dynamics of a financial instrument is one of the essential ingredients for optimising investment portfolios. Here, we address the scenarios of a uniform portfolio in a univariate and multivariate estimation of variance or portfolio risk. Neither case is necessary to forecast the returns but only the volatility to obtain the portfolio's profit. Likewise, special attention is given to the transaction volume assuming that the number of times an asset is bought or sold affects its price.

The work aims to apply statistical and deep learning techniques to forecast the volatility of nine risky cryptocurrencies, considering the price and transaction volume at high frequencies during the onset of the COVID-19 pandemic. Thus, we want to answer the following question: does building portfolios through the information of statistical, deep learning, and hybrid models, where volume is included in the estimation, represent a significant improvement concerning naive methods in periods of high volatility?

Specifically, an exploratory study of cryptocurrency assets in terms of their volatility in high-frequency periods is sought. In particular, we apply asymmetric generalised autoregressive conditional heteroskedasticity (GARCH) models. The eGARCH exponential model (Nelson, 1991), and the gjrGARCH model (Glosten et al., 1993) are proposed to model the volatility of cryptocurrencies. It is interesting to include volume as an exogenous variable in volatility models and examine the corresponding information. In addition, a relevant point is the estimation of DCC-GARCH multivariate volatility models (Engle, 2002) to forecast the covariance matrix. In the context of deep learning, long short-term memory (LSTM) models are studied on time series of volatilities. In this way, mixed models of the LSTM–GARCH type are compared, adding the parameters of the eGARCH and gjrGARCH models. In the final stage, investment strategies are constructed: (1) uniform portfolios with univariate forecasts and (2) uniform portfolios with multivariate forecasts of the covariance matrix. Finally, the performance of the models is analysed through the variance of both portfolios using statistical, computational, and financial metrics. Specifically, the heteroscedastic absolute error (HAE), the heteroscedastic quadratic error (HSE), the sharpe ratio (SR), and the value at risk (VaR) are considered, and the accuracy of the prediction is measured through the Diebold–Mariano (DM) test. In the case of SR, we have considered the stable-coin Tether as the risk-free asset as the performance benchmark.

On the one hand, an extensive literature has shown evidence that GARCH models capture the heteroskedastic effects of time series of financial returns (Cont, 2001; Taylor, 1994; Ghysels et al., 1996). For their part, the asymmetric models eGARCH and gjrGARCH are a more stylised proposal to model the asymmetric response of volatility to negative impacts (Nelson, 1991; Glosten et al., 1993; Hentschel, 1995; Harvey & Shephard, 1996). In the case of multivariate models, DCC-GARCH has been the most parsimonious proposal to model the covariance matrix due to its small number of parameters to be estimated (Engle, 2002). In the case of neural network models, recurrent networks have been able to capture time dependencies. In contrast, their LSTM successors have solved the stability problem during optimisation. Thus, combining both models will enhance the predictive capabilities of the individual models in forecasting the volatility of cryptocurrencies (Kim & Won, 2018; Makridakis et al., 2018, 2020; Lahmiri & Bekiros, 2019; Kristjanpoller & Minutolo, 2018; D'Amato et al., 2022).

In this work, hybrid LSTM–GARCH models are used to measure the effect of volatility forecasts in the construction of high-frequency cryptocurrency investment portfolios. The hybrid model is proposed to capture short-term dynamics and obtain predictions that meet the needs of this market. In addition, transaction volume has been included as an exogenous variable. Its behaviour and the relationship it presents in terms of the performance of the forecasts are explored. The results are evaluated using metrics that capture the heteroscedasticity, return, and risk of the associated portfolio and the DM test to determine if the models are significantly different in accuracy.

There are several contributions to this study. We forecast high-frequency volatility in cryptocurrency markets using hybrid deep-learning models combined with stylised GARCH models. One main differentiating factor is to discuss the forecasting performance of univariate and multivariate uniform portfolios through the VaR and the SR metrics. Then, we avoid relying only on statistical tests and computational error metrics to discriminate the best models. Another contribution is considering a highly capitalised selection of cryptocurrencies and comparing them against an *stablecoin*. Then, we exemplify the scope of the study on the portfolio allocation problem between risky assets and the cryptocurrency analogous free-rate asset: Tether. We frame our analysis on the turbulent time of the beginning of the COVID-19 pandemic with a highly systemic risk component. Moreover, we have applied a sliding window strategy to avoid biases due to a particularly low or high turbulent period of the cryptocurrency market. Our methodology can be applied in general to highly non-linear time series and has forecasting implications on simple allocation strategies.

In summary, the present work seeks to contrast different univariate and multivariate volatility forecast models through uniform variance portfolios and evaluate their performance through different metrics with a naive model as a reference. We tie to the recent evidence in the literature showing that machine learning models do not always represent the best results when forecasting financial time series (Makridakis et al., 2018). Finally, the analysis is framed into the beginning of the COVID-19 pandemic as it is a period of high volatility and because it represents a systemic risk scenario on international finances.

Our analysis has implications in the management of investment portfolios, where it is crucial to estimate the variance of the assets in a univariate and multivariate manner. Then, to have a better allocation of assets that minimises exposure to risk and the probability of loss. Our results suggest using simple learning models to improve heteroskedastic errors, drastically reducing computation time compared to more elaborate neural network models. Thus, allowing rapid risk management in high-frequency scenarios, such as the case of a hypothetical investment fund that seeks to diversify into cryptocurrencies.

In Sect. 2, a literature review of volatility forecasts and cryptocurrency markets is performed, as well as the recent application of the DM test. Section 3 introduces the deep learning models used in this study. Section 4 presents the models of the GARCH family. Section 5 presents the elements of portfolio theory and its metrics considered in this study. Section 6 describes the data used and performs a preliminary exploratory analysis. The methodology followed to perform the volatility forecasts through the different models is detailed in Sect. 7. Section 8 shows the main results and discusses their implications regarding the selected metrics. Finally, in Sect. 9, the main findings are summarised, and future lines of research are proposed.

2 Literature Review

The GARCH volatility models were initially introduced in Bollerslev (1986). Engle presents an emblematic financial application that predicts volatility through a GARCH(1,1) model for a portfolio composed of the Nasdaq, Dow Jones and the ten-year Treasury Bond (Engle, 2001). The VaR was taken as a reference to evaluate the proposed solution. Likewise, a class of multivariate GARCH models with the dynamic conditional correlation (DCC) property were proposed by the same author (Engle, 2002); these models are denoted by DCC-GARCH and have fewer parameters than other multivariate implementations of the same type. On the other hand, a GARCH model was extended in Lamoureux and Lastrapes (1990) to consider transaction volume as an exogenous variable, which showed that adding this variable improves the explanatory power of the model. This relationship was also explored in the futures market (Najand & Yung, 1991). In addition, the effect of the volatility index (VIX) of the Chicago Board Options Exchange (CBOE) Market and the transaction volume were compared when modelling them as exogenous variables through a GARCH model (Kambouroudis & McMillan, 2016). In this work, as in the studies mentioned above, the family of GARCH(1,1) models is considered. This choice is based on the discussion of Bollerslev (1987), where the author shows evidence that a GARCH(1,1) model is more parsimonious than a higher-order autoregressive conditional heteroskedasticity(ARCH) model. In this sense, we seek to implement a model with few parameters that correctly captures the dynamics of volatility in financial time series.

Notably, in the topic of cryptocurrencies, the study (Chu et al., 2017) is one of the first to use different variants of the GARCH model to model the daily returns of the seven cryptocurrencies with the highest market capitalisation. In the same sense, previous studies have modelled Bitcoin and found evidence of it exhibiting characteristics between gold and the dollar when applying asymmetric GARCH models (Dyhrberg, 2016). However, the authors in Baur et al. (2018) replicate the previous work reaching the opposite conclusion; that is, the dynamics of bitcoin are very different from those of gold and dollars.

In the area of machine learning, the idea of LSTM has been implemented to predict directional movements in the shares of the S&P 500, and with this information to propose an investment strategy, arguing that it is possible to obtain meaningful information even in markets with statistical noise (Fischer & Krauss, 2018). Likewise, there are competencies where statistical and learning methods are contrasted to evaluate the prognosis of macroeconomic variables, finding that in most cases, the classical statistical methods surpass those of learning (Makridakis et al., 2018). In cryptocurrencies, recent work has implemented deep learning models to predict the direction of Bitcoin by taking into account non-traditional variables and using a non-parametric method of variable selection (García-Medina & Luu Duc Huynh, 2021). Recently, the authors of Banik et al. (2022) built a decision support system (DSS) to forecast the Indian stock market based on the LSTM model. However, none of these studies have explicitly paid attention to volatility.

A line of work that combines the study of volatility through GARCH and learning models has led to the proposal of hybrid models to improve the accuracy of volatility forecasts. In this direction, the study (Kim & Won, 2018) mixes LSTM models with different GARCH models to predict the volatility of the KOSPI 200 index (Korea Composite Stock Price Index). A similar methodology was implemented in Quintero Valencia et al. (2019) to forecast the volatility of the dollar to Colombian peso exchange rate, where a combination of interest rates and commodity prices are added as exogenous variables.

In this work, to compare forecast accuracy is used the DM test. This test was proposed by Diebold and Mariano (1995) with the characteristic that forecast errors can be non-Gaussian, non-zero mean, serially correlated, and contemporaneously correlated. Initially, the authors illustrate the practical use of the tests with an application to exchange rate forecasting between the three-month change in the nominal Dollar/Dutch Guilder. The test was incorporated as a standard practice for testing models in econometrics, to which Diebold criticises in (2015) the applied methodology of practitioners. He emphasises that the DB test should be interpreted as comparing forecasts, not as comparing econometric models. Also, he pointed out the power loss of the pseudo-out-of-sample model comparisons and the limited and unclear benefits of following that path. In the context of forecasting financial series using machine learning, the authors of Gu et al. (2020) modified the DM test. They intended to compare the cross-sectional average of the prediction errors of each model instead of comparing errors between individual returns and avoid violating the condition of weak error dependence. In Mensi et al. (2019) is studied the structural break of price returns of Bitcoin and Ethereum under different models of the GARCH family and compared using the DM test. Similarly, the authors of Catania et al. (2019) study Bitcoin, Litecoin, Ripple, and Ethereum under several multivariate vector autoregressive models with different forms of variation over time, where they also apply the DM test to contrast results. On the other hand, the test has also begun to be used in high-frequency forecasts of cryptocurrencies (Peng et al. 2018) to evaluate hybrid models, as are the support vector regression (SVR) and GARCH type. In the same spirit, it has been decided to use the DM test in this study. F.X. Diebold's discussion (Diebold, 2015) should prevent us from taking the test results as confirmatory rather than indicative of the superiority of one forecast over another.

3 LSTM Deep Learning Models

Machine learning is a field of study in artificial intelligence where algorithms are developed so that computers can learn without being explicitly programmed. Deep learning methods have gained significant importance in recent years, and the number of people or organisations that develop them has increased. In addition, the computational resources to perform this type of processing have become more accessible. In the case of financial series, statistical and econometric methods often present difficulties when modelling non-stationary variables or those with complex dependencies (Dixon et al., 2015). Fortunately, deep learning techniques can identify and deal with these types of complex structures (Fischer & Krauss, 2018).

Artificial neural networks (ANN) are part of machine learning models. Different mathematical disciplines guide this research, and these models can be understood as function approximation machines designed to achieve statistical generalisation (Goodfellow et al., 2016). In general, these models are based on a set of units called neurons that interact with each other, sharing information between them. An artificial neuron or unit receives an input vector x that represents the input or output data of the other neurons connected. This information is weighted by multiplying a vector of weights w that the algorithm estimates during the learning process. This way, an output value or signal $f(x)$ is generated through a transformation function $g(\cdot)$ known as the activation function. This process can be represented in a simplified way through the following expression:

$$f(x) = g(x \cdot w) + b, \tag{1}$$

where b is the trend. MultiLayer perceptrons (MLPs) are the simpler ANNs and are usually used as a vanilla model. The architecture of a MLP consists of an input layer, a hidden layer and an output layer. In this naive model, each node is a neuron.

Recurrent neural networks (RNNs) are a more elaborated class of networks specializing in representing and modelling sequential data. This type of network uses its hidden layers to summarize the historical information of a data sequence. RNNs implement cycles that allow information to persist over time. An RNN unit can be represented by an input x_t that generates a value h_t, where

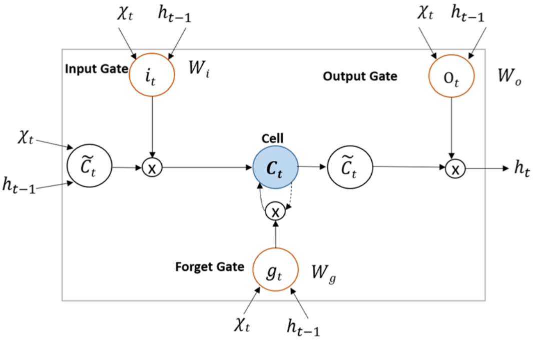


Fig. 1 General structure of an LSTM unit (García-Medina & Luu Duc Huynh, 2021). The memory gate is represented as c_t, the input gate is denoted by i_t, the forget gate is indicated as g_t, and the output gate is as o_t

in its structure, the output of the state at time $t-1$ is the input for the state at time t, and the output of the state at time t is the input for the state at time $t+1$.

The LSTM model was proposed by Hochreiter and Schmidhuber (1997) and is designed to address the problem of long-term dependencies. This model implements gates that specify the amount of information that passes in each time state in such a way that the network discriminates the information that is relevant for the next state. This model solves some difficulties encountered in the RNN training process when calculating the derivatives involved in the optimisation process. Unlike a conventional RNN, each neuron has different layers that help to filter the information. Specifically, an LSTM unit consists of a memory gate (c_t), an input gate (i_t), a forget gate (g_t), and an output gate (o_t). In Fig. 1, the general structure of the mechanism of an LSTM unit is shown. This figure shows at time t, the input x_t, the hidden state h_t, and $\tilde{c}_t$, which determines the amount of accepted information or discarded in the status cell. The internal process of an LSTM unit can be described mathematically through the following expression

$$g_t = \sigma(U_g x_t + W_g h_{t-1} + b_f) \tag{2}$$

$$i_t = \sigma(U_i x_t + W_i h_{t-1} + b_i) \tag{3}$$

$$\tilde{c}_t = tanh(U_x x_t + W_c h_{t-1} + b_c) \tag{4}$$

$$c_t = g_t * c_{t-1} + i_t * \tilde{c}_t \tag{5}$$

$$o_t = \sigma(U_o x_t + W_o h_{t-1} b_o) \tag{6}$$

$$h_t = o_t * tanh(c_t) \tag{7}$$

where U and W are weight matrices, b is the bias term, and the symbol $*$ denotes element-wise multiplication (Kim & Won, 2018).

4 GARCH Models

The GARCH model introduced by Bollerslev (1986) can be understood as a combination of the long-term average value, the volatility information in the previous periods and the adjusted variance of the model. Explicitly, the conditional variance of a GARCH model(q, p) is expressed as

$$\sigma_t^2 = \alpha_0 + \sum_{i=1}^{q} \alpha_i u_{t-i}^2 + \sum_{j=1}^{p} \beta_j \sigma_{t-j}^2, \tag{8}$$

where u_t are the residuals, $\alpha_0 > 0$, $\alpha_i \geq 0$ must be met for $i = 1, \ldots, q$ and $\beta_j \geq 0$ for $j = 1, \ldots, p$.

In the GARCH models, positive and negative values have a symmetrical effect on conditional variance. However, empirical evidence shows that negative returns have a more significant impact on the increase in volatility about positive returns of similar magnitude (Jondeau et al., 2007). The exponential GARCH model or eGARCH was introduced (Nelson, 1991) to improve two aspects of the GARCH model. First, it limits the values of the parameters α and β to ensure a favourable variance. Second, it explicitly includes an asymmetric volatility response to positive and negative news. In the eGARCH model, σ^2 depends on the size and sign of these changes

$$log(\sigma_t^2) = \alpha_0 + \sum_{i=1}^{q} \alpha_i g(z_{t-i}) + \sum_{j=1}^{p} \beta_j log(\sigma_{t-j}^2), \tag{9}$$

where $g(z_t) = \psi z_t + \gamma[|z_t| - \mathbb{E}(|z_t|)]$, in which ψ and γ are real constants. Both z_t, and $|z_t| - \mathbb{E}(|z_t|)$ are independent and identically distributed zero-mean sequences, so $\mathbb{E}[g(z_t)] = 0$. The asymmetry is modelled through the function $g(z_t)$, which has the following form:

$$g(z_t) = \begin{cases} (\psi + \gamma)z_t - \gamma\mathbb{E}(|z_t|) \ \ if\, z_t \geq 0 \\ (\psi - \gamma)z_t - \gamma\mathbb{E}(|z_t|) \ \ if\, z_t < 0 \end{cases}$$

On the other hand, the gjrGARCH model introduced by Glosten et al. (1993) is an extension of the GARCH model that includes an additional term to account for possible asymmetries. This model alternatively captures the empirically observed fact that negative changes at time $t - 1$ have a more substantial impact on the variance at time t than positive changes. The variance in the model gjrGARCH(q, p) is defined by

$$\sigma_t^2 = \alpha_0 + \sum_{i=1}^{q} (\alpha_i + \gamma_i I_{t-1}) u_{t-i}^2 + \sum_{j=1}^{p} \beta_j \sigma_{t-j}^2 \tag{10}$$

for $\alpha_0 > 0$, $\alpha_i \geq 0$, $\alpha_i + \gamma_i \geq 0$ and $\beta_j \geq 0$ for $i = 1, \ldots, q$ and $j = 1, \ldots, p$, where

$$I_t = \begin{cases} 0 \ \ if\, u_t \geq 0 \\ 1 \ \ if\, u_t < 0 \end{cases}$$

On the other hand, GARCH models can be modified to introduce exogenous variables that influence the volatility of returns. In the case of the GARCH(q, p) volatility model, these variables are added by introducing an extra term as follows (Lamoureux & Lastrapes, 1990):

$$\sigma_t^2 = \alpha_0 + \sum_{i=1}^{q} \alpha_i u_{t-i}^2 + \sum_{j=1}^{p} \beta_j \sigma_{t-j}^2 + \delta V_t, \tag{11}$$

where V_t is the exogenous variable, with the condition that $\delta \geq 0$.

In the multivariate case, we seek to estimate a covariance matrix (Jondeau et al., 2007; Bauwens et al., 2006). Let us consider a vector of random variables $x_t = (x_{1,t}, x_{2,t}, \dots, x_{n,t})'$ whose joint distribution is given by:

$$x_t = \mu_t(\theta) + u_t \tag{12}$$

$$u_t = \Sigma_t^{1/2}(\theta) z_t, \tag{13}$$

where $\mu_t(\theta)$ denotes the $n \times 1$ conditional mean vector, $\Sigma_t(\theta)$ is the $n \times n$ conditional covariance matrix of the error term u_t, and θ is the vector of unknown parameters. Let D_t be a diagonal matrix of size $n \times n$ with conditional variances σ_i^2 on the diagonal. Thus, the conditional correlation matrix of u_t can be represented as

$$R_t = D_t^{-1/2} \Sigma_t D_t^{-1/2} = \{\rho_t\}_{ij}. \tag{14}$$

To estimate R_t, it is introduced in Engle (2002) the DCC-GARCH model. In this model, the dynamic correlation matrix R_t is represented as follows:

$$R_t = diag(Q_t)^{-1/2} Q_t diag(Q_t)^{-1/2}, \tag{15}$$

$$Q_t = (1 - \delta_1 - \delta_2)\bar{Q} + \delta_1 (v_t v_t^{\mathsf{T}}) + \delta_2 Q_{t-1}, \tag{16}$$

where $\bar{Q}$ is the unconditional variance matrix of $v_t = \{z_{i,t}/\sigma_{i,t}\}_{i=1,\dots,n}$ (Silvennoinen & Teräsvirta, 2009) and $diag(Q_t)$ is the diagonal matrix of Q_t. The matrix $\bar{Q}$ can be estimated through the sample $\frac{1}{T}\sum_{t=1}^{T} \hat{v}_t \hat{v}_t'$, while the parameters δ_1 and δ_2 satisfy $0 \leq \delta_1, \delta_2 \leq 1$ and $\delta_2 + \delta_2 \leq 1$.

5 Investment Portfolio

In 1952, Markowitz introduced the concept of the efficient frontier. This was the first mathematical formulation of portfolio optimisation. The idea is that investors see an expected return as desirable, while the variance of the returns is undesirable. What Markowitz demonstrated is the existence of a set of optimal portfolios that maximise the expected return given a risk value. This was what he called the optimal frontier (Roncalli, 2013).

Let the vector of returns be $\mathbf{r} = (r_1, \dots, r_p)$, where each element of the vector is the logarithmic return of the p instruments in the portfolio. Additionally, let $\mathbf{w} = (w_1, \dots, w_p)$ be the weights of each asset in the investment portfolio. The return of the portfolio is expressed as

$$R = \sum_{i=1}^{p} w_i r_i, \tag{17}$$

On the other hand, portfolio risk is traditionally defined as the variance of returns.

$$\sigma^2 = \mathbf{w}' \boldsymbol{\Sigma} \mathbf{w}, \tag{18}$$

where Σ is the covariance matrix of the returns.

In the case of equally weighted portfolios or uniform portfolios, the weights are estimated using the formula

$$w_i^* = \frac{1}{p}. \tag{19}$$

Likewise, for the case of a portfolio of minimum variance, the weights are chosen through the expression

$$w^* = \mathrm{argmin}_w \{ w' \Sigma w \} \tag{20}$$

In the scenario in which the weights take both positive and negative values, the following analytical solution can be obtained:

$$w^* = \frac{\Sigma^{-1} 1}{1^\top \Sigma^{-1} 1}. \tag{21}$$

There are variants of this solution based on whether leverage (negative weights) or higher or lower weight restrictions are considered. Here, we only consider the budget restriction: $\sum_{i=1}^{n} w_i = 1$. In both scenarios, uniform portfolio and minimum variance portfolio, once the portfolio weights are determined, the variance of the portfolio is obtained through Eq. 18. This last quantity is modelled and predicted in the following sections.

On the other hand, the SR was developed by the Nobel Prize winner William F. Sharpe and is used to compare a portfolio's return, the risk-free rate and its volatility (Sharpe, 1994). The SR is defined as

$$SR = \frac{R - R_f}{\sigma^2}, \tag{22}$$

where R_f is the risk-free rate. The higher this value, the better trade-off between return and risk associated with the investment.

The VaR is a probabilistic method that measures the potential loss of the portfolio value given a period of time. Specifically, the VaR is the percentage loss of the portfolio value that the actual loss is equal to or greater than only α percent of the time. That is, there is a probability of α percent that the loss in the portfolio's value is equal to or greater than the VaR measure. In our case, we follow the parametric approximation for the calculation of the VaR. In general, the VaR is given by

$$\mathrm{VaR}^{\alpha} = -\sigma_t \Phi^{-1}(\alpha), \tag{23}$$

where σ_t is the square root of the estimated portfolio's variance and $\Phi^{-1}(\alpha)$ is the inverse probability function of α.

6 Data

The prices and transaction volume of the $p = 10$ leading cryptocurrencies in terms of capitalisation (at the date of capture) are obtained through the application programming interface (API) of `CoinMarketCap`.[1] The analysis period is from 01/01/2020 to 06/30/2020, with a frequency of 5 min. The cryptocurrencies considered are listed in Table 1.

The data are transformed to obtain the logarithmic returns of the ten price and volume series. In the case of missing data, linear interpolation is applied. Next, the average returns per hour and the realized variance through the variance of the intrahour data are obtained. In this way, time series of average returns are obtained and volatility series for both the price and the transaction volume, giving a total of $n = 4,368$ observations at hourly frequency. In Fig. 2a, the average hourly returns of the prices and transaction volume are plotted for each of the currencies studied. At the beginning of 2020, the currencies presented low volatility, while in mid-March, there was a structural change in the returns, which can be inferred by their increase in volatility. We can observe a similar behaviour for the volatility returns shown in Fig. 2b.

In Tables 2 and 3, the main statistics for performance concerning price and volume are presented, respectively. The mean (μ), standard deviation (σ), skew (s), kurtosis (τ), and Phillips Perrón (PP) statistics are used to verify the existence of unit roots. In addition, the Ljung-Box (LB) and ARCH LM (LM) statistics have been included in the case of price returns to test the fit of a heteroscedastic model.

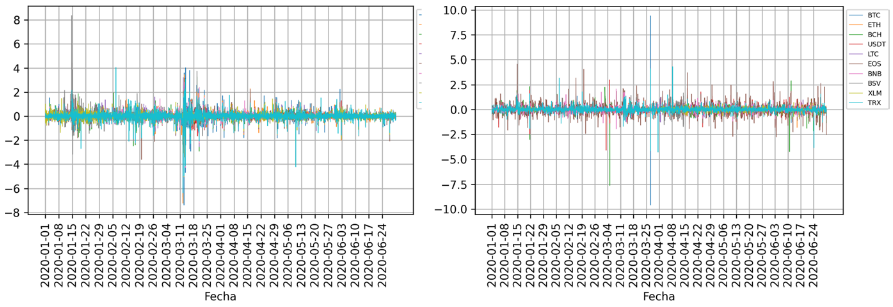
In the PP test, the null hypothesis has a unit root at $\alpha = 0.01$ in all the time series considered. The LB and LM tests evaluate the dependence of the second moment with a time lag. These have as a null hypothesis that an ARCH process adequately fits the data. The test does not reject H_0 for any cryptocurrency.

As a complementary part of the exploratory analysis, the partial autocorrelation function (PACF) is measured to determine the relevance of applying the GARCH heteroscedastic models. In Fig. 3, the corresponding correlogram is plotted for the square of the price returns. It can be observed that the values do not stabilise for any of the cryptocurrencies considered in this study. In sum, this exploratory data analysis shows preliminary evidence of the heteroscedasticity of the returns of cryptocurrencies. This allows us to proceed with the methodology described in the following section.

[1] https://coinmarketcap.com/api/.

Table 1 Name, symbol, release year, and description of the analysed cryptocurrencies

Name	Symbol	Release	Description
Bitcoin	BTC	2009	First virtual currency to use blockchain and distributed computing as a transaction verification method
Ethereum	ETH	2015	It is a digital platform that adopts blockchain technology, its native cryptocurrency is the second largest in the market
Bitcoin cash	BCH	2017	Fork of Bitcoin.
Tether	USDT	2014	Stable currency that replicates the behaviour of the dollar
Litecoin	LTC	2011	Designed to provide fast, secure and low-cost payments
Eos	EOS	2018	Blockchain architecture designed to enable decentralised application programs
Binance coin	BNB	2017	Token of the Binance exchange
Bitcoin SV	BSV	2018	Fork of Bitcoin Cash to improve stability and scalability
Stellar	XLM	2014	Designed to be an intermediary in currency conversion
Tron	TRX	2017	Decentralised platform that focuses on content sharing and entertainment



(a) (b)

Fig. 2 **a** Hourly average logarithmic return of price. **b** Hourly average logarithmic return of volume. In both cases, the colours for each cryptocurrency is described in the sidebar

Table 2 Descriptive statistics of logarithmic returns of prices: mean (μ), standard deviation (σ), skew (s), kurtosis (τ)

	μ	σ	s	τ	PP	LB	LM
Bitcoin	-0.0220×10^{-2}	0.3732	−3.3164	84.8543	−62.4678*	0.3935	0.4327
Ethereum	-0.000110×10^{-2}	0.3595	−2.7697	62.0278	−64.1589*	0.5192	0.6115
Bitcoin Cash	0.0188×10^{-2}	0.3337	−1.9304	37.2363	−61.8218*	0.9741	1.1063
Tether	-0.0224×10^{-2}	0.1455	−0.0979	22.6456	−109.492*	1.6730	0.4826
Litecoin	-0.0412×10^{-2}	0.3336	−2.016	45.093	−62.9980*	0.4633	0.6237
Eos	0.0337×10^{-2}	0.3231	−1.9347	44.6825	−64.6345*	0.8532	0.4467
Binance Coin	0.0633×10^{-2}	0.3000	−2.9582	63.9365	−67.3297*	0.1749	0.3892
Bitcoin SV	0.0952×10^{-2}	0.3255	3.9356	124.63	−59.8099*	1.7390	0.6918
Stellar	0.0262×10^{-2}	0.2897	−1.0965	17.5313	−66.2977*	0.0584	0.9873
Tron	0.1040×10^{-2}	0.3574	−2.5815	52.4063	−63.9707*	0.5209	0.5260

Column PP represent the Phillips–Perron test, LB the Ljung–Box test, and LM the ARCH LM test

The symbol (*) denote the significance at the 1% level

7 Methodology

The models described in Table 4 are implemented. In each scenario, a window sliding of 504 hours is performed, with a window shift of 24 hours. In this way, each model is applied to $M = 161$ windows to forecast the volatility in the hour horizon $h = \{1, 2, \dots, 24\}$. For each value of h, the forecasts of the M windows are generated. Therefore, an average volatility prediction over times $T + 1$, ..., $T + 24$ under different market conditions is computed. The average prediction avoids bias in selecting a specific time window. Instead, it is intended to show average results that capture the periods of high and low volatility as a whole. The specific methodology followed to implement each model (see Table 4) is described below.

Table 3 Descriptive statistics of logarithmic returns of volumes: mean (μ), standard deviation (σ), skew (s), and kurtosis (τ)

	μ	σ	s	τ	PP
Bitcoin	0.1976×10^{-2}	0.3065	−0.3516	424.4616	−67.8452*
Ethereum	0.0651×10^{-2}	0.1692	0.2030	10.4398	−58.9243*
Bitcoin Cash	0.0294×10^{-2}	0.2926	−4.4658	126.9292	−51.7996*
Tether	0.1969×10^{-2}	0.3077	−0.2800	16.8439	−46.8008*
Litecoin	0.0375×10^{-2}	0.1401	−0.0564	15.9906	−62.9605*
Eos	0.2207×10^{-2}	0.4393	0.8287	11.4172	−42.8717*
Binance Coin	0.0997×10^{-2}	0.2391	0.7383	35.1063	−60.2495*
Bitcoin SV	0.0383×10^{-2}	0.1699	0.8485	23.8815	−65.7187*
Stellar	-0.0120×10^{-2}	0.0949	−0.1625	21.3002	−85.9246*
Tron	0.2275×10^{-2}	0.2621	−0.0908	80.7404	−61.8854*

Column PP represent the Phillips–Perron test

The symbol (*) denote the significance at the 1% level

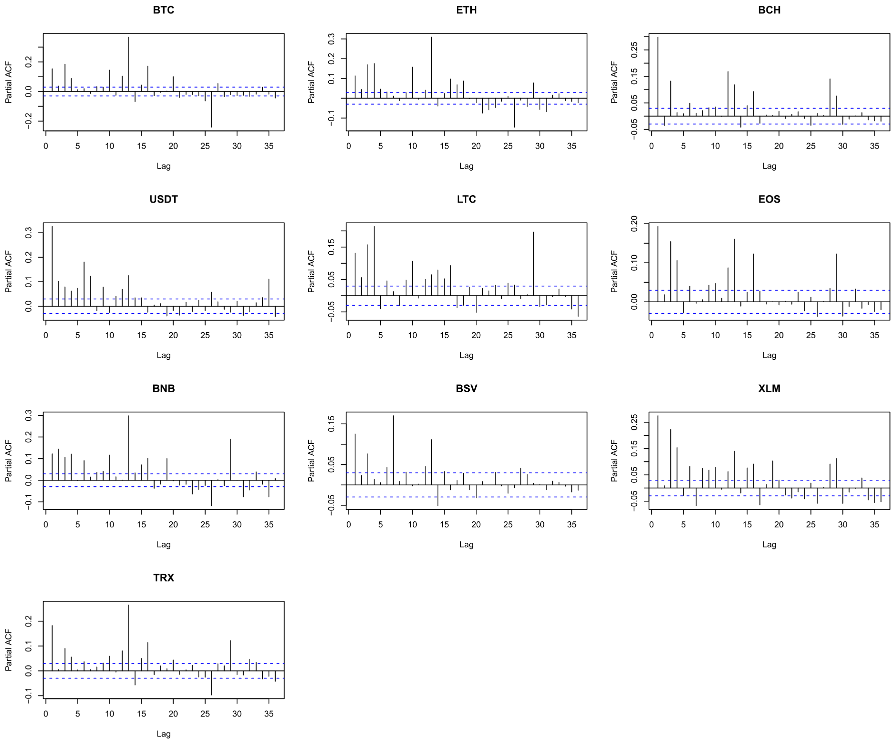


Fig. 3 PACF of squared logarithmic returns of prices from one to 36 lags

Table 4 Identifier (ID), type of model (Model), and specification of the implementation (description)

ID	Model	Description
E1	Naive	The mean of the historical realized variance is considered the forecast in an equally weighted portfolio
E2	eGARCH	eGARCH(1,1) model in an equally weighted portfolio
E3	eGARCH-Vol	eGARCH(1,1) model with volume as an exogenous variable in an equally weighted portfolio
E4	gjrGARCH	gjrGARCH(1,1) model in an equally weighted portfolio
E5	gjrGARCH-Vol	gjrGARCH(1,1) model with volume as an exogenous variable in an equally weighted portfolio
E6	DCC-eGARCH	DCC-eGARCH(1,1) model in a minimum variance portfolio
E7	DCC-eGARCH-Vol	DCC-eGARCH(1,1) model with volume as an exogenous variable in an minimum variance portfolio
E8	MLP	MLP model over realized variance
E9	LSTM	LSTM model over realized variance
E10	LSTM-Vol	LSTM model over realized variance with volume as an additional feature
E11	LSTM-eGARCH	LSTM model over realized variance with the estimated parameters of eGARCH(1,1) as additional features
E12	LSTM-gjrGARCH	LSTM model over realized variance with the estimated parameters of gjrGARCH(1,1) as additional features

7.1 Naive Model

An estimate of the future volatility is constructed using the average of the realized variance during the time windows. In particular, with this procedure, future estimates for the portfolio of equal weights are the same for all prediction horizons $h = 1, \ldots, 24$.

7.2 GARCH Models

In the case of the eGARCH, eGARCH-Vol, gjrGARCH, and gjrGARCH-Vol models, the volatility is predicted directly in terms of the portfolio's performance of cryptocurrencies with uniform weights. In the case of the models that include volume, the volume returns index is constructed by taking the average volume returns of the individual variables. This index represents the volume return of the uniform portfolio and is included as an exogenous variable in the corresponding models.

In the case of the DCC-eGARCH and DCC-eGARCH-Vol models, it is necessary to estimate the portfolio's volatility in two steps. First, the covariance matrix is estimated, where an eGARCH model is specified for each variable, and the corresponding DCC model is estimated with it. In the case of the model with volume, the volume return is included in each individual specification of the eGARCH model for all variables. In the second step, the variance of the portfolio is computed through Eq. 18 considering uniform weights.

The models are estimated for the order (1,1) in all cases. In addition, Student's t-distribution is assumed for the residuals, and the model of the mean is specified as constant. For the estimation of the parameters, the libraries `rugarch` and `rmgarch` in the software `R` are used.

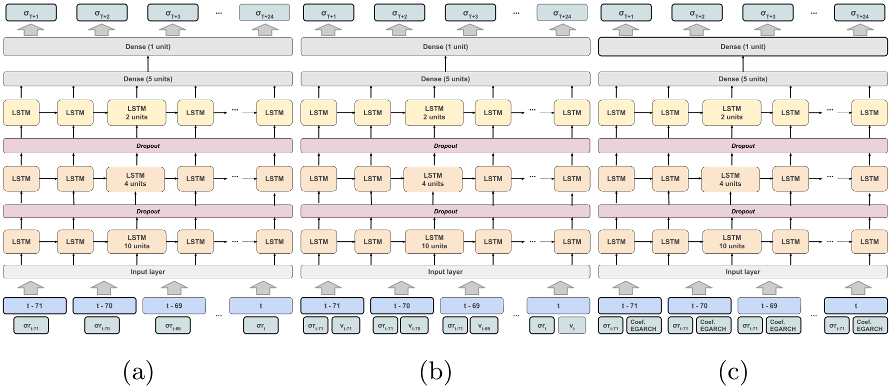
7.3 Deep Learning Models

The MLP and LSTM deep learning models are trained with the variance of each asset. In addition, the volatility data are structured in a supervised learning format. Thus, given the series of volatilities $x_1, \ldots, x_n$, we proceed to create a matrix X that serves as an independent variable of the model and a variable y as the dependent variable, that is, the variable to be predicted. Let n be the size of each of the realized variance series and m be the size of the samples in X. In this way, to obtain the forecast of volatilities $h = \{1, \ldots, 24\}$ steps forwards, the following matrices are constructed:

$$X = \begin{bmatrix} x_1 & x_2 & \cdots & x_m \\ x_2 & x_3 & \cdots & x_{m+1} \\ \vdots & & & \\ x_{n-m+1} & x_{n-m+2} & \cdots & x_n \end{bmatrix} \quad y = \begin{bmatrix} x_{m+1} & \cdots & x_{m+24} \\ x_{m+2} & & x_{m+25} \\ \vdots & & \\ x_{n+1} & \cdots & x_{n+24}, \end{bmatrix} \tag{24}$$

Table 5 Asymmetric GARCH model coefficients added as features of the LSTM learning model (see section 4 for further details)

Parameter	Description
α_0	Trend
α_1	Residual coefficient
β_1	Conditional volatility coefficient
γ	Asymmetry coefficient



(a) (b) (c)

Fig. 4 Architectures used to predict volatility over time $T+1,\ldots,T+24$. Here, σ_{r_t} represent the realized variance at time t, and σ_{T+h} represent the predicted volatility at $T+h$. **a** LSTM models. **b** LSTM-Vol models. **c** LSTM–GARCH models. From bottom to top the flow diagram is the same in all cases. First the input layer, then several hidden layers, and at the top the output layer

where samples have size $m = 72$. With the $n = 504$ observations, the matrices X and y are divided into training, validation and testing subsets of sizes 383, 48, and 1, respectively.

Likewise, the input data are scaled in [0, 1] to streamline the learning process. In the training and validation, the hyperparameters are optimised in a mesh. Specifically, the set of values is considered for the batch sizes [24, 48, 72], with the sigmoid activation function. The combination of parameters that decreases the mean square error is used during the validation. These optimal parameters are used in the test set, and 10 model runs are performed to avoid biases due to the stochastic nature of the model. Finally, the average of the nine predictions is calculated, and this average is considered the estimate of the volatility in each step of the horizon $h = 1,\ldots,24$.

In the case of the LSTM-Vol models, the procedure is similar to that described above. The radical difference is how the volume returns index is included as a feature of the model. Explicitly, this is done by extending the number of columns of the matrix X in Eq. 24. In both models, LSTM and LSTM-Vol, the procedure is carried out through the libraries `TensorFlow` and `keras` in `Python`. In Fig. 4a, b, the architecture followed for the LSTM and LSTM-Vol models is presented, where how the volume index is introduced to the deep learning model can be explicitly seen. The MLP model is used as a vanilla approach, where it is set simply by an input layer, a hidden dense layer of 10 neurons, and an output layer for the predictions.

7.4 LSTM–GARCH Models

In the case of the LSTM-eGARCH and LSTM-gjrGARCH hybrid models, the methodology is similar to that described for the LSTM models. The main difference is that the parameters estimated in the GARCH models for a uniform portfolio are now included as additional features, that is, the parameters of the eGARCH and gjrGARCH models. Specifically, the coefficients added to the hybrid models are described in Table 5. The model includes these parameters by extending the number of columns of matrix X in Eq. 24. In this case, the GARCH parameters are not scaled as is done in the preprocessing of the returns. To estimate the models, the libraries of `R` and `Python` mentioned in the previous sections are used. In Fig. 4c, the architecture followed for the LSTM-eGARCH and LSTM-eGARCH-Vol models is presented, where the procedure to include the GARCH parameters is explicitly shown. In practical terms, the input layer receives the four characteristics given in Table 5 and the value of the historical volatility as in the other simple LSTM models.

7.5 Metrics

To compare the performance of each of the models presented in the previous section, the metrics HAE, HSE, SR, and VaR are evaluated, and the DM test is applied to determine if there are significant differences in the prediction accuracies of the different models. The HAE and HSE are defined at the prediction horizon h and window T by the following expressions:

$$HAE(h,T) = \| 1 - \frac{\sigma^2_{predicted}(T+h)}{\sigma^2_{realized}(T+h)} \|, \tag{25}$$

$$HSE(h,T) = \left(1 - \frac{\sigma^2_{predicted}(T+h)}{\sigma^2_{realized}(T+h)} \right)^2 \tag{26}$$

where $\sigma^2_{predicted}(T+h)$ is the estimated variance over time $T+h$, and $\sigma_{realized}(T+h)$ is the realized variance at time $T+h$. Different values of T represent different time windows or financial scenarios. The HAE and HSE are metrics designed to capture the heteroscedastic effect of the time series. Therefore, it is more natural and fair to use these versions since our object of study is the volatility forecast.

The SR and VaR are defined in the portfolio section. Defined in terms of the prediction horizon h and window T take the following forms:

$$SR(h,T) = \frac{R(h,T) - R_f(h,T)}{\sigma^2_{predicted}(h,T)} \tag{27}$$

$$VaR^{\alpha}(h,T) = -\sigma_{predicted}(h,T)\Phi^{-1}(\alpha), \tag{28}$$

where $\sigma_{predicted}$ is computed by Eq. 18 in the particular case of multivariate DCC-GARCH type models. These financial metrics are included to compare the performance of the forecasts in a possible application, such as the allocation of capital in investment portfolios.

On the other hand, suppose that we have two forecasts $y^{(1)}_{T+h}, y^{(2)}_{T+h}$ at horizon h. We want to know which prognosis is better concerning predictive accuracy. The objective is to select the forecast with the smallest error measure and determine if this difference is significant or simply due to the specific choice of data in the sample. Given the actual time series y_{T+h} and the estimation $\hat{y}_{T+h}$, the prediction error is

$$e^{(i)}_{T+h} = \hat{y}^{(i)}_{T+h} - y_{T+h}, \quad i = 1, 2 \tag{29}$$

In this way, the loss associated with the forecasts is a function associated with the forecast error denoted by $g(e^{(i)}_{T+h})$. The loss differential between two forecasts is defined as

$$d_{T+h} = g(e^{(1)}_{T+h}) - g(e^{(2)}_{T+h}). \tag{30}$$

Thus, two forecasts have the same precision at horizon h if and only if the expected value of d_{T+h} is zero for all T. This condition is the null hypothesis in the DM statistic, which is expressed at horizon h by the formula (Diebold & Mariano, 1995)

$$DM(h) = \frac{\bar{d}}{\sqrt{2\pi\hat{f}_d(0)/M}}, \tag{31}$$

where $\bar{d} = \frac{1}{M}\sum_{T=1}^{M}(g(e^{(1)}_{T+h}) - g(e^{(2)}_{T+h}))$ and $\hat{f}_d(0)$ is a consistent estimator of $f_d(0)$, which denotes the spectral density of the loss differential at frequency 0.

8 Results

This section describes the results obtained in terms of the metrics specified in the methodology. The forecasts of each of the models proposed in Table 4 are compared with the variance of a uniform portfolio, denoted as the realized variance. The uniform portfolio is built with the cryptocurrencies described in Table 1 except Tether, which is considered the risk-free rate R_f in the SR metric. For the calculation of the VaR, the inverse function is estimated based on the 5% quantile. In the DM test, the null hypothesis is that there is no difference in the accuracy of the volatility prediction between one model and another.

The GARCH family models last 5854.06 s (approximately 1 h and 37 min) on an Intel® Core$^{\text{TM}}$ i5-8250U CPU @ 1.60 GHz × four cores. In the case of the deep learning models, it has taken 228,910 s (2 days, 15 h, 35 min and 10 s) to run all 24,150 configurations (5 models x 161 windows x 3 batch values x 10 realizations) in parallel using an Intel® Xeon(R) Gold 6254 CPU @ 3.10GHz × eight cores. Figure 5 shows two typical learning curves for the MLP and LSTM-gjrGARCH models at window 161 and batch size of 72. It can be seen that the

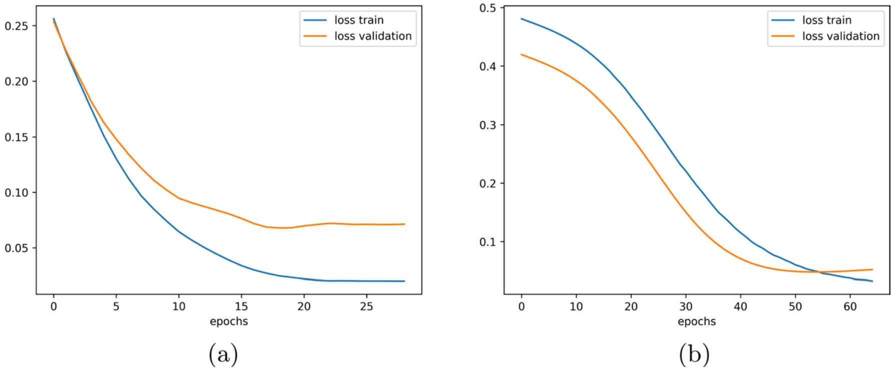


Fig. 5 A typical learning curves at window 161 and batch size 72 for **a** MLP. **b** LSTM-gjrGARCH

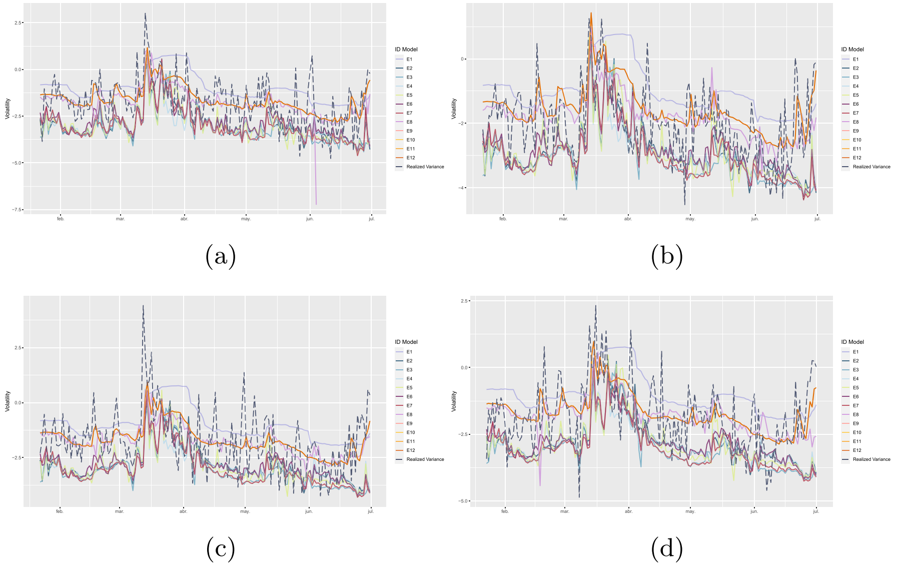
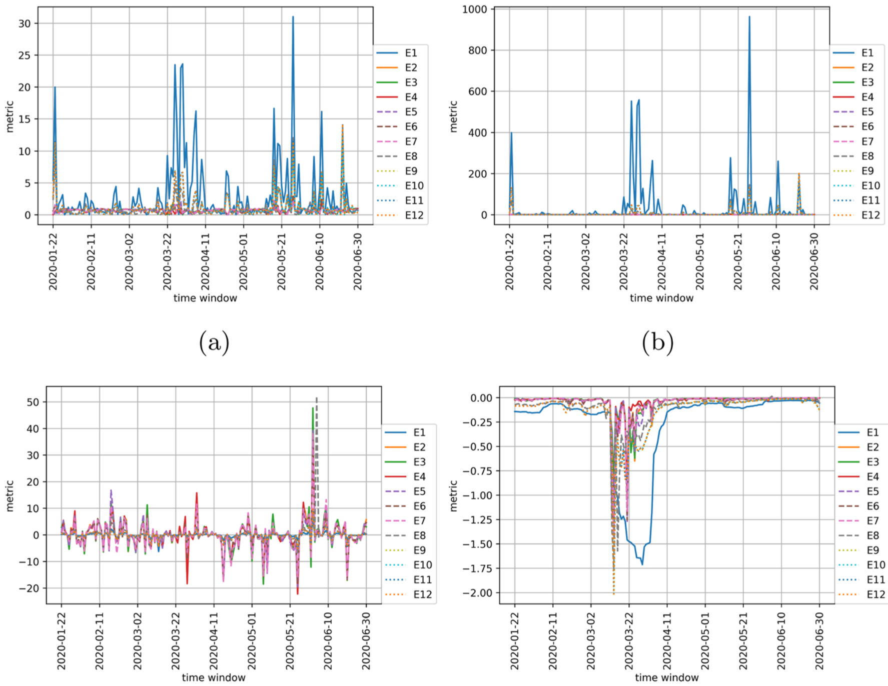


Fig. 6 Prediction of volatility for each time window. **a** horizon $T+1$. **b** horizon $T+6$. **c** horizon $T+12$. **d** horizon $T+24$. The ID Model of the sidebar is described in Table 4. The realized variance is represented by a gray dashed line

number of epochs is different since we have used the criterion of stopping the learning process once a local minimum is reached to avoid overfitting problems.

Figure 6 shows the volatility estimate of each of the models considered throughout the time windows analysed. Due to space, the results are offered only for the representative horizons, where $h = \{1, 6, 12, 24\}$. The forecasts are averages over the $M = 161$ time windows. For comparison terms, the y-axis is logarithmically scaled, and the realized variance of a uniform portfolio is included. It can be observed that



(a) (b) (c) (d)

Fig. 7 Metrics through the $M = 161$ time windows at $T + 1$. **a** HAE. **b** HSE (c) SR. **d** VaR

the deep learning models recover the dynamics of the realized variance. However, the original values present higher orders of magnitudes than the results obtained by the volatility models. This can be observed in the predictions for the horizon $T + 24$, shown in Fig. 6d. A peak is observed in the prognosis of the deep learning models that differ significantly from the GARCH family models. Likewise, a natural grouping can be observed in the models according to their type. For example, the naive model usually exhibits smooth behaviour over time but is of greater magnitude than the other models. This model does not recover all the changes in volatility, which is expected due to its nature. On the other hand, the univariate models of the GARCH family present a similar dynamic but with values lower than the realized variance in most of the windows. Similarly, multivariate GARCH models generally show a common dynamic, although of an even smaller magnitude. This occurs because they are associated with a portfolio of minimum variance. Note that the models that include both pure and hybrid learning are the closest to the realized variance in magnitude. In general, most models recover the dynamics globally; when there is high or low variance, the models capture it, although usually to a lesser extent.

Figure 7 shows the behaviour of the metrics through the $M = 161$ time windows at the horizon $T + 1$. We can see that in specific periods the metrics explode, mainly at times of highest volatility. For example, atypical values are observed around the financial turbulence of March 2020 derived from the announcement of a pandemic

Table 6 MHAE

Model/prediction	$T+1$	$T+6$	$T+12$	$T+24$
Naive	0.9923	1.5329	1.3407	1.2643
eGARCH	0.7181	0.6572	0.7051	0.7193
eGARCH-Vol	0.762	0.715	0.7616	0.7373
gjrGARCH	0.7047	0.6519	0.6895	0.6719
gjrGARCH-Vol	0.7078	0.6636	0.7174	0.7021
DCC-eGARCH	0.7102	0.6425	0.6976	**0.6617**
DCC-eGARCH-Vol	0.7518	0.6861	0.7359	0.7247
MLP	**0.5632**	**0.5977**	**0.6272**	0.6695
LSTM	0.5667	0.7008	0.6575	0.7047
LSTM-Vol	0.5692	0.7018	0.6648	0.7018
LSTM-eGARCH	0.5675	0.7017	0.6582	0.6993
LSTM-gjrGARCH	0.5691	0.7007	0.6554	0.7051

The best scores for each selected forecast horizon are highlighted

Table 7 MHSE

Model/Prediction	$T+1$	$T+6$	$T+12$	$T+24$
Naive	0.9847	2.3496	1.7974	1.5984
eGARCH	0.5157	0.4319	0.4972	0.5174
eGARCH-Vol	0.5807	0.5113	0.58	0.5436
gjrGARCH	0.4966	0.4249	0.4754	0.4515
gjrGARCH-Vol	0.5009	0.4403	0.5147	0.4929
DCC-eGARCH	0.5044	0.4128	0.4867	0.4378
DCC-eGARCH-Vol	0.5651	0.4707	0.5415	0.5251
MLP	**0.3171**	**0.3573**	**0.3934**	**0.4482**
LSTM	0.3211	0.4911	0.4323	0.4967
LSTM-Vol	0.324	0.4926	0.4419	0.4926
LSTM-eGARCH	0.322	0.4923	0.4333	0.489
LSTM-gjrGARCH	0.3239	0.491	0.4295	0.4972

The best scores for each selected forecast horizon are highlighted

by the World Health Organization(WHO). Due to these outliers, instead of averaging the metrics over all the windows, we have taken their median with the value that characterizes the performance of the models over time. Thus we have denoted by MHAE, MHSE, MSR, and MVaR, the median over the M=161 time windows of the HAE, HSE, SR, and VaR metrics, respectively.

In Table 6, the results for the MHAE in the selected prediction horizons are shown. It can be observed that the results in the naive case are the least favourable. For the other models, errors of the same magnitude are presented, although the results obtained by the MLP model show the lowest errors at $T+1$, $T+6$ and $T+12$, while the DCC-eGARCH model present the lowest MHAE at $T+24$. On the other hand, notably, there is no significant increase in errors when increasing the prediction horizon for any model. In Table7, the results corresponding to the MHSE

Table 8 MSR

Model/prediction	$T+1$	$T+6$	$T+12$	$T+24$
Naive	**−0.0093**	**−0.056**	0.0216	0.0872
eGARCH	−0.0312	−0.3141	0.1727	0.5287
eGARCH-Vol	−0.0328	−0.358	0.1816	0.5482
gjrGARCH	−0.0261	−0.3053	0.1918	0.5378
gjrGARCH-Vol	−0.0196	−0.226	0.1444	0.4408
DCC-eGARCH	−0.0198	−0.2331	0.1763	0.4797
DCC-eGARCH-Vol	−0.0261	−0.2778	**0.1934**	**0.5596**
MLP	−0.0204	−0.1222	0.0519	0.2381
LSTM	−0.0208	−0.0872	0.0442	0.1912
LSTM-Vol	−0.0208	−0.0873	0.0444	0.1911
LSTM-eGARCH	−0.0207	−0.0873	0.0441	0.1913
LSTM-gjrGARCH	−0.0208	−0.0872	0.0444	0.1908

The best scores for each selected forecast horizon are highlighted

are presented. Given this metric, the naive model again presents the worst performance by one order of magnitude. On the other hand, the models that involve deep learning exhibit a performance with lower errors than that of the GARCH models at $T+1, T+6$ and $T+12$. In all prediction horizons, the MLP model yields the best results, yet comparable to the other deep learning models.

The performance of each of the portfolios with respect to the MSR is shown in Table 8. In this case, at $T+1$ and $T+6$ the Naive model present the highest SR, which is desirable in this metric. Nevertheless, the values at these horizons of predictions are negative. On the contrary, at horizon $T+12$ and $T+24$ the best result is obtained by the DCC-eGARCH-Vol model with positive values. Hence, the results imply that an investment strategy with a longer time horizon generates a higher Sharpe ratio most of the time. The results also imply that the uniform portfolio performance is worse or better than only investing in Tether, depending on the expected variance at different forecasting horizons. Then, a potential investor should trade on the selected cryptocurrencies at long and Tether at short horizons, considering our high-frequency frame.

Similarly, the results for the MVaR are shown in Table 9. Note that all values are negative by definition. In this case, the best results are consistently shown by the univariate eGARCH-Vol models for the horizons considered. They are an order of magnitude higher than the naive model and considerably lower than the deep learning models. The results are remarkable since we can infer that the transaction volume reduces the portfolio's losses at a tolerance of 5%. In other words, it is essential to include the volume information in the volatility forecast to better management of a hypothetical uniform allocated portfolio.

Next, the comparison of the residuals according to the DM statistic is presented. The null hypothesis of this test is that Model 2 (M2) has greater precision than Model 1 (M1), and the alternative is that Model 1 (M1) has greater accuracy than Model 2 (M2). The null hypothesis is rejected if the p-value is less than

Table 9 MVaR

Model/prediction	$T+1$	$T+6$	$T+12$	$T+24$
Naive	−0.102	−0.102	−0.102	−0.102
eGARCH	−0.0136	−0.0155	−0.0159	−0.0166
eGARCH-Vol	**−0.0121**	**−0.0121**	**−0.0123**	**−0.0125**
gjrGARCH	−0.0139	−0.0162	−0.0167	−0.0171
gjrGARCH-Vol	−0.0134	−0.0149	−0.015	−0.0157
DCC-eGARCH	−0.0168	−0.0173	−0.0177	−0.0188
DCC-eGARCH-Vol	−0.0134	−0.0137	−0.0139	−0.014
MLP	−0.0557	−0.0563	−0.0558	−0.0573
LSTM	−0.0565	−0.0596	−0.0567	−0.0566
LSTM-Vol	−0.0566	−0.0599	−0.0567	−0.0564
LSTM-eGARCH	−0.0565	−0.0595	−0.0565	−0.057
LSTM-gjrGARCH	−0.0567	−0.0594	−0.0567	−0.0564

The best scores for each selected forecast horizon are highlighted

Table 10 Comparison of residuals of the different models with the Diebold–Mariano test at horizon $T+1$

M1 / M2	E1	E2	E3	E4	E5	E6	E7	E8	E9	E10	E11	E12
E1	1	0	0	0	0	0	0	0	0	0	0	0
E2	1	1	1	1	1	0	0	0	0	0	0	0
E3	1	0	1	0	0	0	0	0	0	0	0	0
E4	1	0	1	1	0	0	0	0	0	0	0	0
E5	1	0	1	1	1	0	0	0	0	0	0	0
E6	1	1	1	1	1	1	1	0	0	0	0	0
E7	1	1	1	1	1	0	1	0	0	0	0	0
E8	1	0	0	0	0	0	0	1	0	0	0	0
E9	1	0	0	0	0	0	0	0	1	0	0	0
E10	1	0	0	0	0	0	0	0	0	1	0	0
E11	1	0	0	0	0	0	0	0	0	0	1	0
E12	1	0	0	0	0	0	0	0	0	0	0	1

The cell value is 1 if Model 1 (M1) has greater forecasting accuracy than Model 2 (M2), and zero on the contrary

$\alpha = 0.1$. In Tables 10, 11, 12, and 13, the results to horizons $T+1, T+6, T+12$, and $T+24$ are shown, respectively.

In these tables, the rows represent Model 1, and the columns represent Model 2. When making the row/column comparison, the cell is highlighted if Model 1 has greater forecasting accuracy than Model 2. We can observe in Table 10 that in the estimation of $T+1$, the DCC-eGARCH (E6) and all the deep learning models (E8-E12) do not reject the null hypothesis when compared to any other model. On the other hand, Table 11 show at horizon T+6 that the MLP, LSTM-Vol, and

Table 11 Comparison of residuals of the different models with the Diebold–Mariano test at horizon $T + 6$

M1 / M2	E1	E2	E3	E4	E5	E6	E7	E8	E9	E10	E11	E12
E1	1	0	0	0	0	0	0	0	0	0	0	0
E2	0	1	0	0	0	0	0	0	0	0	0	0
E3	0	1	1	1	0	0	0	0	0	0	0	0
E4	0	1	0	1	0	0	0	0	0	0	0	0
E5	0	1	1	1	1	0	0	0	0	0	0	0
E6	0	1	1	1	0	1	0	0	0	0	0	0
E7	0	1	1	1	1	1	1	0	0	0	0	0
E8	1	1	1	1	1	1	1	1	0	0	0	0
E9	1	1	1	1	1	1	1	0	1	0	0	0
E10	1	1	1	1	1	1	1	0	1	1	1	0
E11	1	1	1	1	1	1	1	0	0	0	1	0
E12	1	1	1	1	1	1	1	0	0	0	1	1

The cell value is 1 if Model 1 (M1) has greater forecasting accuracy than Model 2 (M2), and zero on the contrary

Table 12 Comparison of residuals of the different models with the Diebold–Mariano test at horizon $T + 12$

M1 / M2	E1	E2	E3	E4	E5	E6	E7	E8	E9	E10	E11	E12
E1	1	0	0	0	0	0	0	0	0	0	0	0
E2	0	1	1	0	0	0	0	0	0	0	0	0
E3	0	0	1	0	0	0	0	0	0	0	0	0
E4	0	1	1	1	1	1	1	0	0	0	0	0
E5	0	1	1	0	1	0	1	0	0	0	0	0
E6	0	1	1	0	1	1	1	0	0	0	0	0
E7	0	0	1	0	0	0	1	0	0	0	0	0
E8	1	1	1	1	1	1	1	1	0	0	0	0
E9	1	1	1	1	1	1	1	0	1	0	0	0
E10	1	1	1	1	1	1	1	0	0	1	0	0
E11	1	1	1	1	1	1	1	0	0	0	1	0
E12	1	1	1	1	1	1	1	0	0	0	0	1

The cell value is 1 if Model 1 (M1) has greater forecasting accuracy than Model 2 (M2), and zero on the contrary

LSTM-gjrGARCH are the only ones for which the null hypothesis is not rejected concerning all the other models. For the case $T + 12$, Table 12 show that there is not significative evidence that any model surpasses those in the class of deep learning models (E8-E12) according to the DM test. A similar behaviour is found at horizon $T + 24$ (see Table 13). Note that the naive model rejects the null hypothesis of better accuracy about the GARCH family models at this horizon. In general, it is found

Table 13 Comparison of residuals of the different models with the Diebold–Mariano test at horizon $T + 24$

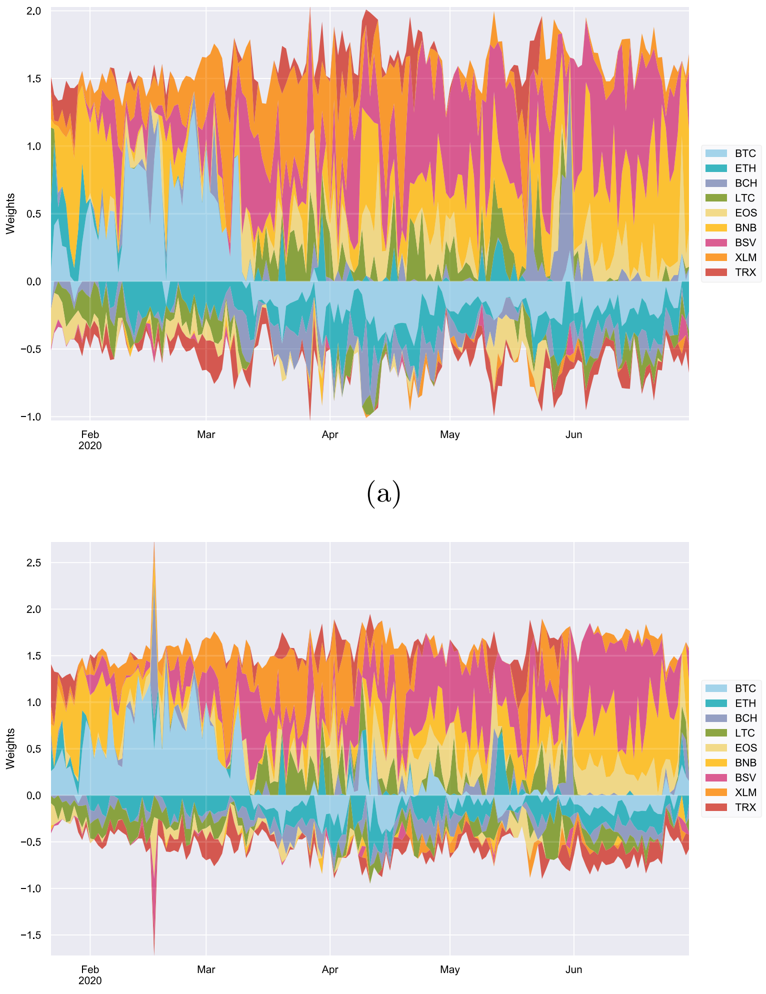
M1 / M2	E1	E2	E3	E4	E5	E6	E7	E8	E9	E10	E11	E12
E1	1	1	1	1	1	1	1	0	0	0	0	0
E2	0	1	1	1	0	0	1	0	0	0	0	0
E3	0	0	1	0	0	0	0	0	0	0	0	0
E4	0	0	1	1	0	0	1	0	0	0	0	0
E5	0	1	1	1	1	0	1	0	0	0	0	0
E6	0	1	1	1	1	1	1	0	0	0	0	0
E7	0	0	1	0	0	0	1	0	0	0	0	0
E8	0	1	1	1	1	1	1	1	0	0	0	0
E9	1	1	1	1	1	1	1	0	1	0	0	0
E10	1	1	1	1	1	1	1	0	0	1	0	0
E11	1	1	1	1	1	1	1	0	0	0	1	0
E12	1	1	1	1	1	1	1	0	0	0	0	1

The cell value is 1 if Model 1 (M1) has greater forecasting accuracy than Model 2 (M2), and zero on the contrary

that the models of the GARCH family are not significantly more accurate than the deep learning models. Even among the latter, we have no evidence to lean towards one or the other; that is, they are statistically equivalent in accuracy.

One possible application of variance forecasting is in the context of asset allocation in an investment portfolio. For this purpose, we construct a portfolio of minimum variance using Eq. 21 and the forecasts of the covariance matrix of the DCC-GARCH models. Figure 8 shows the change in portfolio allocations for each time window at the forecast horizon $T + 1$. Note that the only restriction considered in the allocation of weights is their sum must be equal to one. Therefore, it is expected that there are allocations with negative weights in the portfolios associated with the DCC-GARCH and DCC-GARCH-Vol models. Our results reveal the transition from the long to the short position of BTC, which coincides with the declaration of a pandemic by WHO. Also, BTC allocated the highest proportion of capital before March 2020; after that, there was a more diversified allocation of assets in the portfolio. Hence, we observe investors' typical tendency to diversify in times of high volatility and financial turbulence.

The performance of the forecasts is in line with what is found in Kim and Won (2018) for the volatility forecast of the KOSPI 200 stock index. However, their models are neither compared to vanilla MLP models nor in the context of cryptocurrencies. On the other hand, the recent competition (Makridakis et al., 2020) is favourable to hybrid models, but they do not compare with cryptocurrencies nor price volatility. In fact, in its previous version (Makridakis et al., 2018), it had been concluded that statistical models outperformed learning models in macroeconomic time series. Also, the work (Lahmiri & Bekiros, 2019) predicts the price of Bitcoin, Digital Cash and Ripple using chaotic neural networks. However, they do not compare with traditional and naive methods or analyse volatility. Further, the work



(a)

(b)

Fig. 8 Allocation weights per instrument on the minimum variance portfolio at horizon $T + 1$ through the analysed windows. **a** DCC-eGARH predictions. **b** DCC-eGARCH-Vol predictions. The budget restriction $\sum_{i=1}^{n} w_i = 1$, is considered in all cases

(Kristjanpoller & Minutolo, 2018) is more in line with our object of study. They analyse hybrid models to forecast the volatility of bitcoin and find favourable results for Artificial Neural Network-Generalised AutoRegressive Conditional Heteroskedasticity (ANN-GARCH). Although the approach of the hybrid models is slightly different, and in our case, we analyse a portfolio instead, the results coincide in favouring the deep learning models but not the vanilla MLP models as in our case. Moreover, in D'Amato et al. (2022), the authors propose the Jordan Neural Network model to forecast the volatility of Bitcoin, Ripple and Ethereum. Their work found positive results concerning simpler models, although they do not consider the vanilla MLP version. To summarise, it can be inferred that the performance of the models is closely linked to the object of study. Cryptocurrencies are much more volatile than traditional markets, which may be why hybrid models do not work as expected.

9 Conclusions

We explored different models for volatility forecasting in the cryptocurrency market. The study period covered the financial turbulence of March 2020, when the World Health Organization (WHO) declared that the sanitary situation derived from the COVID-19 virus was a pandemic. In this sense, the cryptocurrency market, like traditional markets, experienced high volatility, representing an atypical period of systemic risk and, therefore, an increase in the complexity of the dependencies to be modelled.

Surprisingly, the median of the heteroscedastic metrics HAE and HSE reveal better performance for vanilla MLP models, yet comparable to the simple and hybrid deep learning models of LSTM and LSTM–GARCH. Particularly at T+1, a clear distinction in performance can be seen between the models of the GARCH family and the deep learning models. In general, the latter outperforms the former under these metrics.

Further, the performance of forecasts is evaluated in the context of univariate and multivariate uniform variance portfolios through the SR and VaR. Here we consider as a free-rate asset the stablecoin Tether and set a level of loss of $\alpha = 0.5$. Interestingly, we have found that the median of SR is negative for horizons T+1 and T+6 and positive for horizons T+12 and T+24. Also, we find that the naive models give better results for T+1 and T+6, while the DCC-eGARCH-Vol model maximises the SR for T+12 and T+24. Hence, the results imply that an investment strategy with a longer time horizon generates a higher Sharpe ratio. On the contrary, investors should allocate the capital to the risk-free asset Tether in the short horizon.

In the case of VaR, the lowest losses were found when we forecasted with the eGARCH-Vol models at the selected horizons. Therefore, including transaction volume information helps reduce losses in the portfolio. Nevertheless, we must be cautious with these two metrics since we are using data within the sample, and comparisons between models are only helpful to evaluate which reproduces the best-desired characteristics.

We have analysed the error residuals according to the DM statistics for the selected horizons. In general, GARCH family models are not found to be significantly more accurate than deep learning models. Even among the latter, we have no evidence to lean towards one or the other; that is, they are statistically equivalent in accuracy. In sum, the DM test favours the predictions obtained by the LSTM models for most of the prediction horizons.

One of the applications of this work is in the context of the allocation of assets in a portfolio of minimum variance or minimum investment risk. We illustrate the T+1 case, where we find a large proportion of capital assigned to BTC before the pandemic declaration. While from March, the portfolio becomes more diversified in the predictions of both DCC-eGARH and DCC-eGARCH-Vol. In short, there is a clear phase transition from a period highly dominated by BTC to a period where investors prefer diversifying their portfolios. The radical change from long to short positions in BTC is observed. So this type of forecast could serve as a thermometer in the face of financial turmoil in general. The allocation change of BTC is an unexpected finding and has exciting risk management implications that should be studied more carefully.

The evidence of our work leans towards a vanilla learning model as the preferred to model the volatility of high-frequency cryptocurrencies. The MLP outperforms the more stylised statistical models of the GARCH type. Even more surprising, LSTM and hybrid models do not show significant improvement over the vanilla neural network model. Further, including transaction volume showed improvements by reducing VaR. The unexpected result suggests that the increased parameters estimated by the more elaborated deep learning models destabilise the solutions and increase the prediction error. Then, using simple learning models in the volatility forecast of highly non-linear time series is suggested. On the one hand, MLP outperforms the most stylised models as the family GARCH. At the same time, MLP is not computationally as expensive as the LSTM models while keeping the same level of accuracy.

Future work can be extended in different aspects. Within finance, it would be interesting to analyse portfolio optimisation techniques by including additional constraints in a multistage manner and exploring risk metrics that consider more general distributions of returns. In the line of time series, the models of the GARCH family can be explored exhaustively using other specifications. In terms of machine learning, there is a wide field of work. Regarding the model selection, it would be desirable to perform tests with different architectures and hyperparameters, considering a more significant amount of information, both for the size of the input sequence and the dataset in general. It would be interesting to integrate data preprocessing, feature selection, and classification strategies as proposed in Pustokhina et al. (2021) to improve the methodology of our study. Even more, implementing hybrid models, including dynamic parameters of the GARCH models, is of great relevance to tackling real applications. Finally, we are interested in exploring more flexible models of deep learning to model the covariance matrix in the multivariate context.

Acknowledgements The authors are grateful for the discussions with Graciela Gonzalez Farías and Norberto A. Hernández Leandro.

Author Contributions EAM: Methodology, software, validation, formal analysis, investigation, data curation, writing—original draft, writing—review and editing, visualization. AGM: Conceptualization, methodology, validation, investigation, resources, writing—original draft, writing—review and editing, project administration, funding acquisition.

Funding This study received financial support from the National Council of Science and Technology (Consejo Nacional de Ciencia y Tecnología - CONACYT) through FOSEC SEP-INVESTIGACION BASICA (Grant No. A1-S-43514).

Availability of Data and Materials The data that support the findings of this study are available from the corresponding author upon request.

Code Availability Deep learning codes are available at: https://github.com/agarciam/lstmgarch_project

Declarations

Conflict of interest The authors declare no conflict of interest. The funders had no role in the design of the study; in the collection, analyses, or interpretation of data; in the writing of the manuscript, or in the decision to publish the results.

Ethics Approval Not applicable

Consent to Participate Not applicable

Consent for Publication Not applicable

References

Banik, S., Sharma, N., Mangla, M., Mohanty, S. N., & Shitharth, S. (2022). Lstm based decision support system for swing trading in stock market. *Knowledge-Based Systems, 239*, 107994.

Baur, D. G., Dimpfl, T., & Kuck, K. (2018). Bitcoin, gold and the us dollar-a replication and extension. *Finance Research Letters, 25*, 103–110.

Bauwens, L., Laurent, S., & Rombouts, J. V. (2006). Multivariate GARCH models: a survey. *Journal of Applied Econometrics, 21*(1), 79–109.

Bollerslev, T. (1986). Generalized autoregressive conditional heteroskedasticity. *Journal of Econometrics, 31*(3), 307–327.

Bollerslev, T. (1987). A conditionally heteroskedastic time series model for speculative prices and rates of return. *The Review of Economics and Statistics* (pp. 542–547).

Catania, L., Grassi, S., & Ravazzolo, F. (2019). Forecasting cryptocurrencies under model and parameter instability. *International Journal of Forecasting, 35*(2), 485–501.

Chu, J., Chan, S., Nadarajah, S., & Osterrieder, J. (2017). GARCH modelling of cryptocurrencies. *Journal of Risk and Financial Management, 10*(4), 17.

Cont, R. (2001). Empirical properties of asset returns: Stylized facts and statistical issues. *Quantitative Finance, 1*, 223–236.

Diebold, F. X. (2015). Comparing predictive accuracy, twenty years later: A personal perspective on the use and abuse of Diebold–Mariano tests. *Journal of Business & Economic Statistics, 33*(1), 1–1.

Diebold, F. X., & Mariano, R. S. (1995). Comparing predictive accuracy. *Journal of Business and Economic Statistics, 13*(3), 253–263.

Dixon, M., Klabjan, D., & Bang, J.H. (2015). Implementing deep neural networks for financial market prediction on the intel xeon phi. In *Proceedings of the 8th workshop on high performance computational finance* (pp. 1–6).

Dyhrberg, A. H. (2016). Bitcoin, gold and the dollar-a GARCH volatility analysis. *Finance Research Letters, 16*, 85–92.

D'Amato, V., Levantesi, S., & Piscopo, G. (2022). Deep learning in predicting cryptocurrency volatility. *Physica A: Statistical Mechanics and its Applications, 596*, 127158.

Engle, R. (2001). GARCH 101: The use of arch/GARCH models in applied econometrics. *Journal of Economic Perspectives, 15*(4), 157–168.

Engle, R. (2002). Dynamic conditional correlation: A simple class of multivariate generalized autoregressive conditional heteroskedasticity models. *Journal of Business & Economic Statistics, 20*(3), 339–350.

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.

García-Medina, A., & Luu Duc Huynh, T. (2021). What drives bitcoin? An approach from continuous local transfer entropy and deep learning classification models. *Entropy, 23*(12), 1582.

Ghysels, E., Harvey, A. C., & Renault, E. (1996). 5 stochastic volatility. *Statistical Methods in Finance. Handbook of Statistics* (Vol. 14, pp. 119–191). The Netherlands: Elsevier.

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

Goodfellow, I., Bengio, Y., Courville, A., & Bengio, Y. (2016). *Deep learning* (Vol. 1). Cambridge: MIT Press.

Gu, S., Kelly, B., & Xiu, D. (2020). Empirical asset pricing via machine learning. *The Review of Financial Studies, 33*(5), 2223–2273.

Harvey, A. C., & Shephard, N. (1996). Estimation of an asymmetric stochastic volatility model for asset returns. *Journal of Business & Economic Statistics, 14*(4), 429–434.

Hentschel, L. (1995). All in the family nesting symmetric and asymmetric GARCH models. *Journal of Financial Economics, 39*(1), 71–104.

Hochreiter, S., & Schmidhuber, J. (1997). Long short-term memory. *Neural Computation, 9*(8), 1735–1780.

Jondeau, E., Poon, S.-H., & Rockinger, M. (2007). *Financial modeling under non-Gaussian distributions*. New York: Springer.

Kambouroudis, D. S., & McMillan, D. G. (2016). Does VIX or volume improve GARCH volatility forecasts? *Applied Economics, 48*(13), 1210–1228.

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.

Kristjanpoller, W., & Minutolo, M. C. (2018). A hybrid volatility forecasting framework integrating GARCH, artificial neural network, technical analysis and principal components analysis. *Expert Systems with Applications, 109*, 1–11.

Lahmiri, S., & Bekiros, S. (2019). Cryptocurrency forecasting with deep learning chaotic neural networks. *Chaos, Solitons & Fractals, 118*, 35–40.

Lamoureux, C. G., & Lastrapes, W. D. (1990). Heteroskedasticity in stock return data: Volume versus GARCH effects. *The Journal of Finance, 45*(1), 221–229.

Makridakis, S., Spiliotis, E., & Assimakopoulos, V. (2018). Statistical and machine learning forecasting methods: Concerns and ways forward. *PloS One, 13*(3), 0194889.

Makridakis, S., Spiliotis, E., & Assimakopoulos, V. (2020). The m4 competition: 100,000 time series and 61 forecasting methods. *International Journal of Forecasting, 36*(1), 54–74.

Mensi, W., Al-Yahyaee, K. H., & Kang, S. H. (2019). Structural breaks and double long memory of cryptocurrency prices: A comparative analysis from bitcoin and Ethereum. *Finance Research Letters, 29*, 222–230.

Najand, M., & Yung, K. (1991). A GARCH examination of the relationship between volume and price variability in futures markets. *The Journal of Futures Markets (1986–1998), 11*(5), 613.

Nakamoto, S. (2008). Bitcoin: A peer-to-peer electronic cash system. Available at https://bitcoin.org/bitcoin.pdf.

Nelson, D. B. (1991). Conditional heteroskedasticity in asset returns: A new approach. *Econometrica: Journal of the Econometric Society*, *59*(2), 347–370.

Peng, Y., Albuquerque, P. H. M., de Sá, J. M. C., Padula, A. J. A., & Montenegro, M. R. (2018). The best of two worlds: Forecasting high frequency volatility for cryptocurrencies and traditional currencies with support vector regression. *Expert Systems with Applications, 97*, 177–192.

Pustokhina, I. V., Pustokhin, D. A., Mohanty, S. N., García, P. A. G., & García-Díaz, V. (2021). Artificial intelligence assisted internet of things based financial crisis prediction in fintech environment. *Annals of Operations Research*. https://doi.org/10.1007/s10479-021-04311-w.

Quintero Valencia, D.E. *et al.* (2019). Pronóstico de volatilidad de la trm mediante un modelo híbrido LSTM–GARCH. PhD thesis, Universidad del Rosario.

Roncalli, T. (2013). *Introduction to risk parity and budgeting*. New York: CRC Press.

Sharpe, W. F. (1994). The sharpe ratio. *Journal of Portfolio Management, 21*(1), 49–58.

Silvennoinen, A., & Teräsvirta, T. (2009). Multivariate GARCH models. In: Handbook of Financial Time Series, pp. 201–229. Springer, London.

Taylor, S. J. (1994). Modelling stochastic volatility: A review and comparative study. *Mathematical Finance, 4*, 183–204.

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.