



# Forecasting bitcoin volatility: exploring the potential of deep learning

Tiago E. Pratas<sup>1</sup>  · Filipe R. Ramos<sup>2</sup>  · Lihki Rubio<sup>3</sup> 

Received: 25 February 2023 / Revised: 19 May 2023 / Accepted: 23 May 2023 /  
Published online: 14 June 2023  
© The Author(s) 2023

## Abstract

This study aims to evaluate forecasting properties of classic methodologies (ARCH and GARCH models) in comparison with deep learning methodologies (MLP, RNN, and LSTM architectures) for predicting Bitcoin's volatility. As a new asset class with unique characteristics, Bitcoin's high volatility and structural breaks make forecasting challenging. Based on 2753 observations from 08-09-2014 to 01-05-2022, this study focuses on Bitcoin logarithmic returns. Results show that deep learning methodologies have advantages in terms of forecast quality, although significant computational costs are required. Although both MLP and RNN models produce smoother forecasts with less fluctuation, they fail to capture large spikes. The LSTM architecture, on the other hand, reacts strongly to such movements and tries to adjust its forecast accordingly. To compare forecasting accuracy at different horizons MAPE, MAE metrics are used. Diebold–Mariano tests were conducted to compare the forecast, confirming the superiority of deep learning methodologies. Overall, this study suggests that deep learning methodologies could provide a promising tool for forecasting Bitcoin returns (and therefore volatility), especially for short-term horizons.

**Keywords** Cryptocurrencies · Bitcoin · ARCH/GARCH models · Deep learning · Forecasting · Prediction error

---

✉ Filipe R. Ramos  
frramos@fc.ul.pt

✉ Lihki Rubio  
lihkir@uninorte.edu.co

Tiago E. Pratas  
Emanuel\_Teixeira@iscte-iul.pt

<sup>1</sup> Department of Economics, ISCTE-University Institute of Lisbon, 1649-026 Lisbon, Portugal

<sup>2</sup> CEAUL-Centro de Estatística e Aplicações, Faculdade de Ciências, Universidade de Lisboa, Campo Grande 016, 1749-016 Lisbon, Portugal

<sup>3</sup> Department of Mathematics and Statistics, Universidad del Norte, 081007 Barranquilla, Colombia

**JEL Classification** C01 · C02 · C10 · C22 · C45 · C53 · C58 · C60 · G17

## 1 Introduction

Civilization at its present conception would not exist without money. Recent advancements in blockchain technology enable the creation of decentralized monetary systems called cryptocurrencies, where most famous one is Bitcoin, which has become a new asset class. This new type of asset is becoming part of the global financial and economic ecosystem, bringing new and interesting research questions that represent investigation opportunities.

Current macro-economic conditions, with the EUR/USD parity in hand with worldwide high inflation, make it the right time to question the concepts of money, the role of central banks and to better understand what opportunities these alternative systems can bring to the discussion and, ultimately, whether these new ideas can in fact help to improve our societies as whole.

The motivation for this study is to address the need for better understanding and forecasting of Bitcoin volatility, as this new asset class becomes increasingly relevant in the global financial and economic ecosystem. While traditional econometric models have been used to forecast financial assets volatility, the high volatility and unusual market patterns of cryptocurrencies present a challenge for these techniques. As a result, there is a need for more modern and innovative forecasting models that can better capture the nature of these markets. This study compares the prediction results of traditional econometric models, such as ARCH and GARCH, with machine learning models, specifically neural networks, in predicting Bitcoin volatility while doing a review on what might be the causes of this extraordinary volatility. In addition, the additional computational costs associated with machine learning models are justified by the improved forecasting accuracy. Thereby, a new insight into forecasting Bitcoin volatility will be provided and a contribution to the current discussion on the role and potential of cryptocurrencies and machine learning techniques in econometric studies will be made.

Forecasting models are critical decision-making tools for economic agents, investors, and governments, particularly when predicting financial and economic data (Aminian et al., 2006).

Econometric models, such as autoregressive conditional heteroskedasticity (ARCH) and generalized autoregressive conditional heteroskedasticity (GARCH) models, have been extensively used to model the volatility of financial assets. However, the high volatility and unusual patterns and behaviors of cryptocurrency markets make it challenging to apply such models (Franses & Van Dijk, 1996; Pilbeam & Langeland, 2015). To address this challenge, some scholars have proposed the use of modern techniques, such as machine learning/deep learning, to develop models that better explain and predict the nature of cryptocurrency markets (Bezerra & Albuquerque, 2017; Liu, 2019) and help businesses better understand the risks associated with these assets or assist in pricing derivatives. However, most of the time, there are no references to the implicit computational cost.

For example, regarding stock price forecasting, Costa et al. (2019), Lopes et al. (2021) and Ramos et al. (2018) report that some Recurrent Neural Networks (RNN) models, e.g., Long Short-Term Memory networks—LSTM, can be promising for modeling and forecasting time series with structure breaks, or with very irregular behavior (such as time series related to financial markets). However, despite the good forecasting quality, Lopes et al. (2021) and Ramos et al. (2021) note that these neural network architectures have a significant computational cost. Due to the facts mentioned by these authors, further reflection is important, combining the prediction power and computational cost of DNN models.

Thus, in addition to comparing methodologies (classical and deep learning), this work seeks to bring a scientific contribution in two aspects: (i) a comparative analysis between different deep learning methodologies, seeking to understand any differences; (ii) a critical analysis of the implicit computational cost (often omitted in scientific papers). These are aspects that have not been much discussed in the literature, so this work aims to contribute to the scientific debate on the subject.

The results of our study indicate that machine learning models, specifically neural networks, outperform traditional econometric models in forecasting Bitcoin volatility, especially in short-term horizons. Although requiring significant computational costs (specially LSTM models).

This paper is structured as follows: Sect. 2 reviews the relevant literature. The forecasting models are defined formally in Sect. 3, as well the data to be used in the study (including graphics illustrating the volatility to be forecast). Section 4 outlines the methodology employed in the implementation, including the forecasting models, statistical tests, and evaluation metrics. Section 5 presents a descriptive and inferential data analysis, along with visualizations of the forecast obtained by each model and accuracy tables. Finally, Sect. 6 concludes the paper and outlines directions for future research.

## 2 Literature review: bitcoin and volatility

According to the literature, there are conflicting ideas about what may explain the extra-ordinary volatility. Hayes (2017) and Garcia et al. (2014) argue that the main determinants of the Bitcoin price are production costs (electricity costs), and lower electricity prices or reduced mining difficulty will result in negative pressure on the Bitcoin price. Yermack (2015) highlights that since the quantity of new bitcoins is known with certainty by the public, this provides a clear and transparent understanding of the supply of new bitcoins. Gronwald (2019) states that the limited long-term fixed supply of Bitcoin makes it scarce as it is an “exhaustible resource commodity such as crude oil and gold” and analyzes demand shocks. Another important feature is the programmed supply shocks of the production of Bitcoin (halving’s) that result on price volatility as buyers and sellers adjust for an equilibrium price, which however, will become less important over time (Chaim & Laurini, 2018). Pagnotta and Buraschi (2018) model Price-Hash Rate Spirals. Additionally, it is also important to mention the high occurrence of settlement cascades due to the unregulated nature of

most crypto markets which allows the usage of high leverage and market manipulation, contributing to this problem and increase volatility.

Taleb (2021) disagrees with the cost models discussed above and states that any price should be zero, arguing that Bitcoin does not exhibit inflation hedging properties and has failed as a payment network due to high transaction costs and volatility in value.

Volatility plays an important role to measure and access potential risks and by getting a better understanding and knowledge of how it can be predicted, may support decision-making regarding future expectations. Due to cryptocurrencies high volatility, classical methodologies may face some difficulties. Kim and Won (2018) state that volatility plays crucial roles in financial markets, such as in derivative pricing, portfolio risk management, and hedging strategies. Black and Scholes (1973) would corroborate this importance due to their work and research on option pricing models. Markowitz (1952) argues that volatility is one of the key indicators to measure risk and uncertainty implying that the higher the volatility, the higher the risk of the asset or portfolio of assets. Hang (2019) highlighted the importance of forecasting, stating that it is an important tool to help companies create competitive advantage.

Several authors have applied the most diverse techniques to forecast volatility. Some of the most important models for forecasting volatility across the literature include ARCH by Engle (1982) and GARCH by Bollerslev (1986). Some authors studied their properties on crypto assets (Bergsli et al., 2022; Gronwald, 2019; Klose, 2022). Kim and Won (2018) agree on the advantages of such, since volatility clustering, heteroscedasticity and leptokurtosis can be captured. On the other hand, Klose (2022), uses GARCH models to forecast volatility of crypto assets and gold. In addition, he studies similarities and differences based on important factors related to liquidity premia, volatility and pronounced responses.

Classical machine learning tools, such as random forest (RF) and support vector machine (SVM) models, have been used to forecast volatility. SVM model, for example, have been used to forecast volatility of the S&P 500 index, taking advantage of its tolerance to high-dimensional inputs (Gavrishchaka & Banerjee, 2006). On the other hand, some authors have opted to use hybridization strategies mixing SVM with other models such as GARCH, ARIMA and wavelet transform to improve forecasting performance, for example, in the forecast of real stock market data, daily changes of the pound sterling, the New York Stock Exchange composite index and major stocks in Colombia (Chen et al., 2010; Rubio & Alba, 2022; Tang et al., 2009). In addition, RF model is widely used in volatility forecasting, e.g. for high-frequency historical data, crude oil and electricity market volatility, obtaining in each case competitive forecasting in terms of error for different forecast horizons (Luong & Dokuchaev, 2018; Wang et al., 2022).

Regarding Bitcoin volatility, it should be noted that, historically, cryptocurrencies exhibit higher volatility than other traditional asset classes and their returns exhibit a set of structural anomalies and breaks that could generate forecasting problems for the mentioned models. Ramos (2021) argues that although simple in application, classic linear methodologies have some difficulties in dealing with events that have out-of-the-ordinary patterns, as Pesaran and Timmermann (2004) and Chatfield

(2016). Contagion spill overs are also a phenomenon in cryptocurrencies, particularly in Bitcoin, which exhibit strong interdependence across different exchange markets. Giudici and Pagnottoni (2019, 2020) have shown that this interdependence persists both at high and low frequencies.

Due to the challenges, over the past decade, it has been possible to see different Artificial Intelligence techniques, such as artificial neural networks (ANN)/deep neural networks (DNN) have been pointed out in the scientific literature as a promising alternative (Sezer et al., 2020; Tealab, 2020; Tkáč & Verner, 2016). Research on nonlinear methodologies based on neural networks, extensively discussed in the nineties and abandoned due to computational limitations (Bengio et al., 1994) reappear in recent works. Therefore, the scientific research along with the computational progress seen in recent years—due to the use of graphic process units (GPUs)—has assumed a fundamental role in the adoption of ANN to a larger audience. This is seen in simpler DNN structures (e.g. multilayer perceptron (MLP)) or more complex DNN structures (e.g. recurrent neural networks (RNN) and long short-term memory (LSTM) (Ramos et al., 2022).

In fact, many applications of DNN have appeared in the scientific literature in solving some problems related to the modeling and forecasting of time series, referring to its success (Balcilar et al., 2017; Kristjanpoller & Minutolo, 2018; Lahmiri & Bekiros, 2019; Mallqui & Fernandes, 2019; Pichl et al., 2017; Ramos et al., 2022). Part of these works points into that direction when forecasting volatility and prices of cryptocurrencies and/or financial time series using methodologies such as deep learning and hybrid models with both classical and neural network techniques. These techniques have shown significant improvements over classical approaches (Smyl, 2020).

However, the lack of interpretability in DNN models, commonly referred to as the "black box" problem, is a major challenge in adopting these models, particularly in finance where interpretability is crucial for regulatory compliance, risk management, and stakeholder communication. Previous studies, such as Bracke et al. (2019), have applied Shapley values to compare the explainability of neural network-based models with logistic regression models for default risk analysis. The Shapley values provided a useful tool to interpret the neural network model, highlighting the importance of individual input variables in predicting the model output. In a recent study, Giudici and Raffinetti (2021) proposed a novel approach, called Shapley-Lorenz explainable artificial intelligence (SLXAI), which combines Shapley values and Lorenz curves to provide a more nuanced measure of model explainability. The effectiveness of their approach in explaining the predictions of a random forest model for credit rating was demonstrated. On the other hand, there are other methodologies such as Recurrent Neural Networks (RNN) with Temporal Attention and Bayesian Neural Networks (BNN). Each of these methodologies allows assigning weights in the recurrent neural networks based on relevance and probability distribution, thus solving problems of interpretability, overfitting, and low data (Mirikitani & Nikolaev, 2010; Qin et al., 2017).

Despite these advances, the trade-off between model performance and interpretability remains an open question, and further research is needed to develop more effective approaches to model explainability.

This highlights the importance of this article to make literature contributions that generates awareness of such methodologies to researchers in business and financial markets, so that these tools are used on a daily basis in research.

### 3 Methodology and data

#### 3.1 ARCH and GARCH models

An ARCH/GARCH model for the daily return  $y_t$  is given by Eq. (1)

$$y_t - \mu_t = \xi_t = \sigma_t Z_t \tag{1}$$

where  $Z_t$  is a random variable that is an *i.i.d.* process such that,  $E(Z_t) = 0$  and  $Var(Z_t) = 1$ . The  $\mu_t$  and  $\sigma_t$  represent measurable functions related to a  $\sigma$ -field  $\Sigma_{t-1}$  produced by historical returns  $y_{t-k}, k \geq 1$ .

Engle (1982) selected the following representation for  $\sigma_t^2$

$$\sigma_t^2 = \gamma + \sum_{j=1}^p \alpha_j \xi_{t-j}^2 \tag{2}$$

where  $\gamma, \alpha_j, j = 1, \dots, p$  are positive real values. The Eq. (2) is frequently known as the ARCH( $p$ ) model. The strength of the model defined by Eq. (2) resides in how it handles positive serial correlation  $\xi_t^2$ , that is, large (small)  $\xi_t^2$  values is followed with large (small)  $\xi_{t+1}^2$  values.

Bollerslev (1986) extended the ARCH( $p$ ) method to introduce the GARCH( $p, q$ ) defined by Eq. (3)

$$\sigma_t^2 = \gamma + \sum_{i=1}^q \beta_i \sigma_{t-i}^2 + \sum_{j=1}^p \alpha_j \xi_{t-j}^2 \tag{3}$$

allowing an improved expression for  $\sigma_t^2$  based on lagging  $\xi_t^2$  values (constants  $\gamma, \beta_i, i = 1, \dots, q, \alpha_j, j = 1, \dots, p$ , are each positive). The seasonal and non-market impact are integrated to GARCH models by treating  $\gamma$  as function of the time.

#### 3.2 Deep neural networks models

The MLP architectures is nowadays one of the most widely used network structures for classification and regression (Bishop, 1995). MLP model is defined by Eq. (4)

$$\hat{y} = \alpha_0 + \sum_{j=1}^N \alpha_j f_A(\omega_j^T y') \tag{4}$$

where  $y'$  represent de vector of inputs,  $y' = (1, y^T)^T$ ,  $\omega_j$  is the weight vector,  $\alpha_0, \alpha_1, \dots, \alpha_N$  are the output weights and  $\hat{y}$  is the network output. Function  $f_A$  is

the hidden node output, and is expressed as a squashing function, e.g. the logistic function.

From a data set of predefined outputs, neural networks can rapidly auto-learn and adapt themselves, allowing them to model and forecast non-linear and highly complex structures. RNNs are a group of neural networks that, because of more than one connection(s) among neurons, create cycles. The RNN cycles save and transmit information between neurons, building an inner memory which permits learning sequential information. In this way RNNs differ from standard neural networks since memory allows them to detect sequential correlations.

RNNs may be trained via backpropagation through time (BPTT) algorithm (Pineda, 1987). To calculate outputs in the hidden layer units, the following procedure shall be followed

$$h_t = f_A(\mathcal{M}_{\nearrow} y_t + \mathcal{M}_{\nearrow} h_{t-1}) \quad (5)$$

where  $f_A$  is named the activation function for the occult layer,  $y_t$  the entry corresponding to the preceding layer,  $\mathcal{M}_{\nearrow}$  is the binding weights in the prior layer,  $h_{t-1}$  is a return output determined from the previous step and  $\mathcal{M}_{\nearrow}$  its weight (Hopfield, 1982; Rumelhart et al., 1986). Different researchers demonstrated that RNNs can collect only limited data, causing long-term dependency issues. To address this problem, RNN frameworks as the LSTM architectures are available (Hochreiter & Schmidhuber, 1997; Malhotra et al., 2015).

The LSTM model pioneered by Hochreiter and Schmidhuber (1997) is probably the preferred deep learning method for natural language processing problems as it can handle long term dependencies inherent in the data and overcome gradient vanishing issues. Equations for calculating outputs and state values for the LSTM module are given by

$$f_t = f_A(\mathcal{M}_{\nearrow} (y_t, \mathcal{C}_{t-1}, h_{t-1}) + b_f) \quad (6)$$

$$i_t = f_A(\mathcal{M}_i \cdot (y_t, \mathcal{C}_{t-1}, h_{t-1}) + b_i) \quad (7)$$

$$o_t = f_A(\mathcal{M}_o \cdot (y_t, \mathcal{C}_{t-1}, h_{t-1}) + b_o) \quad (8)$$

$$C_t = C_{t-1} + f_t + \tilde{C} \cdot i_t \quad (9)$$

where  $f_A$  represents the activation function,  $y_t$  the input data,  $h_{t-1}$  the prior output,  $\mathcal{M}_{\nearrow}, \mathcal{M}_i, \mathcal{M}_o$  and  $b_f, b_i, b_o$  represent weights and input, forget and output gate biases (Chung et al., 2014; Hochreiter & Schmidhuber, 1997).

### 3.3 Cross-validator and performance metrics

The methodology proposed by Hodrick and Prescott (1997) was used to remove the cycle and trend components, and the CUSUM algorithm by Duarte and Watanabe

(2018) to study structural breaks. For inferential analysis, several hypothesis tests were applied, to study normality (Jarque–Bera, Skewness and Kurtosis) and BDS test to study data independence.

To systematically assess the quality of forecasting models, error metrics are used. The most common performance/error metrics are the following: mean absolute error (MAE) and mean absolute percentage error (MAPE) (Willmott & Matsuura, 2005). Considering the time series  $\{y_t\}_{t \in T}$  and the past observations from period  $1, \dots, t$ , and being  $y_{t+h}$  an unknown value in the future  $t + h$  and  $\hat{y}_{t+h}$  its forecast, the prediction error corresponds to the difference of these two values, that is,

$$e_{t+h} = y_{t+h} - \hat{y}_{t+h} \tag{10}$$

where MAE and MAPE are defined, respectively, by

$$\text{MAE} = \frac{\sum_{i=1}^s |e_{t+i}|}{s} \tag{11}$$

$$\text{MAPE} = \frac{\sum_{i=1}^s \left| \frac{y_{t+i} - \hat{y}_{t+i}}{y_{t+i}} \right|}{s} \times 100 \tag{12}$$

where  $s$  corresponds to the number of observations in the forecasting samples (forecasting window).

In addition, a Diebold–Mariano test (Diebold & Mariano, 2002) was performed with the most efficient model for each category (ARCH/GARCH vs Neural Networks). The Diebold–Mariano test is in fact the most used instrument to estimate significance differences for forecasting precision. This is a  $z$ -test for the statistical hypothesis for the loss differential series mean defined by Eq. (13)

$$\delta_k^{A,B} = \mathcal{L}(\epsilon_k^A) - \mathcal{L}(\epsilon_k^B) \tag{13}$$

where  $\epsilon_k^Z = y_k - \tilde{y}_k$ , is the prediction error for the  $Z$  model at timestep  $k$  and  $\mathcal{L}$ , is the function of loss. To provide forecast at  $k$ , loss function is defined as  $\mathcal{L}(\epsilon_k^Z) = |\epsilon_k^Z|^p$ , for  $p = 1, 2$ .

### 3.4 Data

Data used on this study was obtained from the *Yahoo Finance* public API by calling the ticker “BTC-USD” from 07-09-2014 to 01-05-2022 and obtaining the “Close Price” values expressed in U.S. Dollars. Using this time series, daily logarithmic returns were calculated given by the expression

$$y_t := \log(P_t/P_{t-1}) = \log(P_t) - \log(P_{t-1}) \tag{14}$$

where,  $P_t$  denote the close price at time  $t$ . For time series forecasting there is a precedent for transforming non-iid returns to a closer approximation using log normalization (or the Fisher Transform) for the prediction process. The inverse transform



is then performed on the output to restore the original distribution ready to use predicted returns that allows for calculation of the predicted volatility. With the goal of achieving the research purpose of this study, two-time series variables are defined: BTC-USD that represents Bitcoin's Daily Closing Prices (Fig. 1) and BTC-USD-RET that represents Bitcoin's Daily Returns (Fig. 2).

## 4 Empirical findings

Initial steps in this study involved performing statistical calculations to better describe the BTC-USD-RET series. Results presented in Table 1, showed a strongly positively skewed time series with an extremely high positive leptokurtic kurtosis, and non-normal distribution confirmed by rejecting all the null hypothesis for normality. In addition, both the Augmented Dickey–Fuller (ADF) and the Kwiatkowski–Phillips–Schmidt–Shin (KPPS) tests confirmed the series' stationarity and independence and therefore the data is i.i.d. highlighting the importance of implementing nonlinear models for forecasting time series.

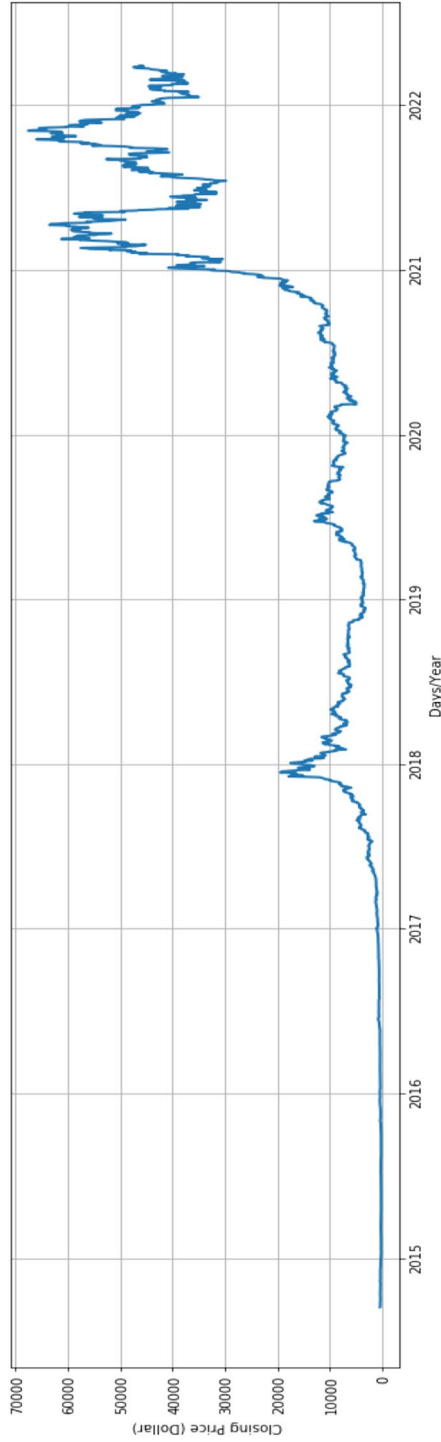
Several structural break situations were identified by applying the CUSUM algorithm, where shifts along the time series were observed in several regimes, as can be seen in Pratas (2022) for the same data set. As pointed out, the high number of structural breaks might represent forecasting difficulties for the classical econometric models and an advantage for the deep learning methodologies.

The subsequent step in the analysis involves model implementation, with the stationary nature of the time series allowing for the use of autocorrelation and partial autocorrelation test functions to determine the optimal number of lags for the models. It was found that the ARCH(4) and GARCH(4, 2) models had the best expected generalization properties, with both AIC and BIC showing lower values for the given parameters. This finding is contrary to the literature's preference for GARCH(1, 1) but in conformity in volatility forecasting for Bitcoin, as noted by Senarathne (2019).

In terms of forecasting itself, the forecasting out-of-sample for the ARCH model (Fig. 3) and the GARG model (Fig. 4) do not seem to be well-adjusted to the real data, as the forecast line does not follow the real data line.

For the neural network study, prior to training, the entire BTC-USD-RET dataset underwent an exponential smoothing pre-processing procedure. Three to five hidden layers were employed in conjunction with architecture-specific hyper parameters and the ADAM optimization algorithm, as recommended Brownlee (2018) and Kingma and Ba (2015) respectively. Cross-validation was performed using the Forward Chaining methodology, as suggested by Ramos (2021). Model performance was assessed using the Mean Absolute Error (MAE) and Mean Absolute Percentage Error (MAPE), and forecasts were generated for one, three, and seven-day horizons. The resulting models were evaluated, and the forecasts were plotted (see Fig. 5 for DNN Models: (A) MLP model, (B) RNN model, (C) LSTM model).

Upon examination of the DNN models, it was found that all models exhibited some degree of forecasting ability. However, the MLP model performed better on shorter time horizons (one-day and three-days), while the RNN model had lower



**Fig. 1** BTC-USD: bitcoin's daily closing prices Source: Own author

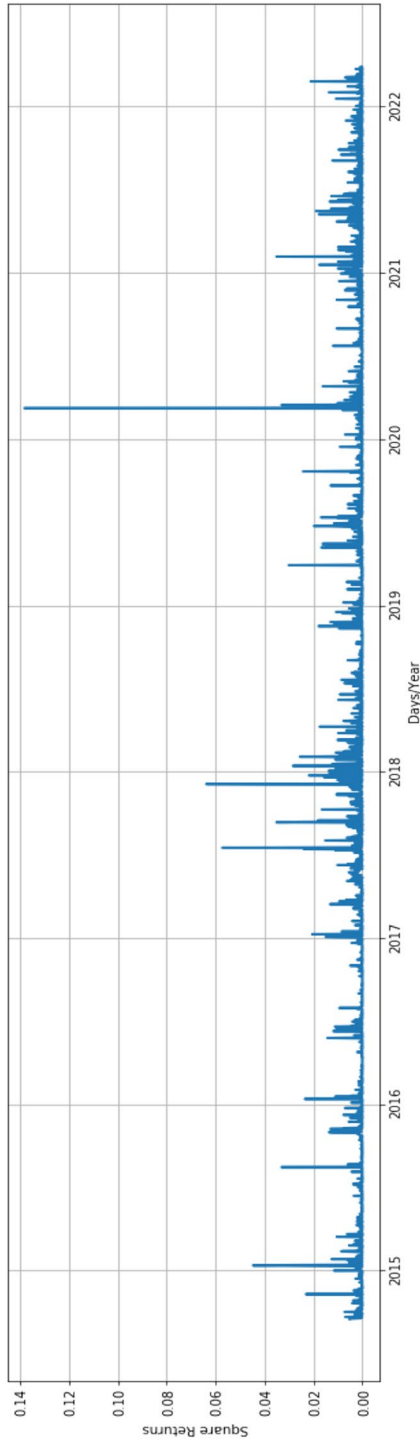


Fig. 2 BTC-USD-RET: bitcoin's daily returns Source: Own author

**Table 1** BTC-USD-RET: main statistics, normality, stationarity and independence tests

Count	Mean					Std
2753	0.001513					0.004576
Min	Q1	Q2	Q3	Max	Kurtosis	Skew
0.000000	0.000038	0.000251	0.001196	0.138157	319.950	13.469
Statistic	Normality tests			Unit root tests		Independency tests
	Kurtosis	Skewness	Jarque–Bera	ADF	KPPS	BDS (Dim.2–Dim. 6)
Statistic	36.4657	61.5091	11,825,693.6472	−15.1772	0.2025	0.2417–0.4654
<i>p</i> value*	0.0000*	0.0000*	0.0000*	0.0000*	–	0.8090–0.6416

\* $H_0$  is rejected for the significance levels of 1%, 5% and 10%

prediction errors on the seven-day horizon. Interestingly, the LSTM model, despite its complexity, performed the worst in terms of accuracy. This was anticipated, as LSTM models tend to underperform when forecasting stationary time series, as pointed out by Ramos et al. (2022). Moreover, this type of neural network is the one that requires the most computational cost and time, making it highly inefficient to use it on our forecast. Nonetheless, both the MLP and RNN models produced smoother forecasts with less fluctuation, but failed to capture large volatility spikes, such as the one that occurred on day two. In contrast, the LSTM model reacted strongly to such movements and attempted to adjust its forecast accordingly, due to its long-term memory properties that allow the model to “remember” that past volatility spikes may lead to high volatility spikes in the future, known as volatility clustering.

To compare performance of ARCH/GARCH models and DNN models, MAE and MAPE values for their forecasting out-of-sample were calculated for three different time horizons. For the DNN models, the parameters of the neural network (weights and bias) benefited from a pseudo-random initialization instead of using a fixed seed (Glorot & Bengio, 2010). To ensure the reliability of the results and avoid outliers, the forecasting was conducted in a loop of 200 runs, and the 5% worst and best results were excluded (according Ramos, 2021). The range of MAPE values, with the lower and upper bounds trimmed by 5%, are presented in Table 2, along with the MAE values for models with intermediate forecast quality chosen from each architecture (as shown in Fig. 5).

Once all the models have been estimated, metrics calculated, and results presented, it was deemed useful to conclude with a visual representation comparing five models (see Fig. 6). Results indicate that ARCH (4) and GARCH (4.2) models are superior, with ARCH (4) being the best model in terms of forecast accuracy as measured by mean absolute percentage error (MAPE).

Regarding deep learning approach, MLP model demonstrated superior performance for shorter time horizons (1-day and 3-days), while the RNN model showed lower prediction errors for seven-day horizon. This finding can be attributed to the basic memory capabilities of the RNN model, which produce a

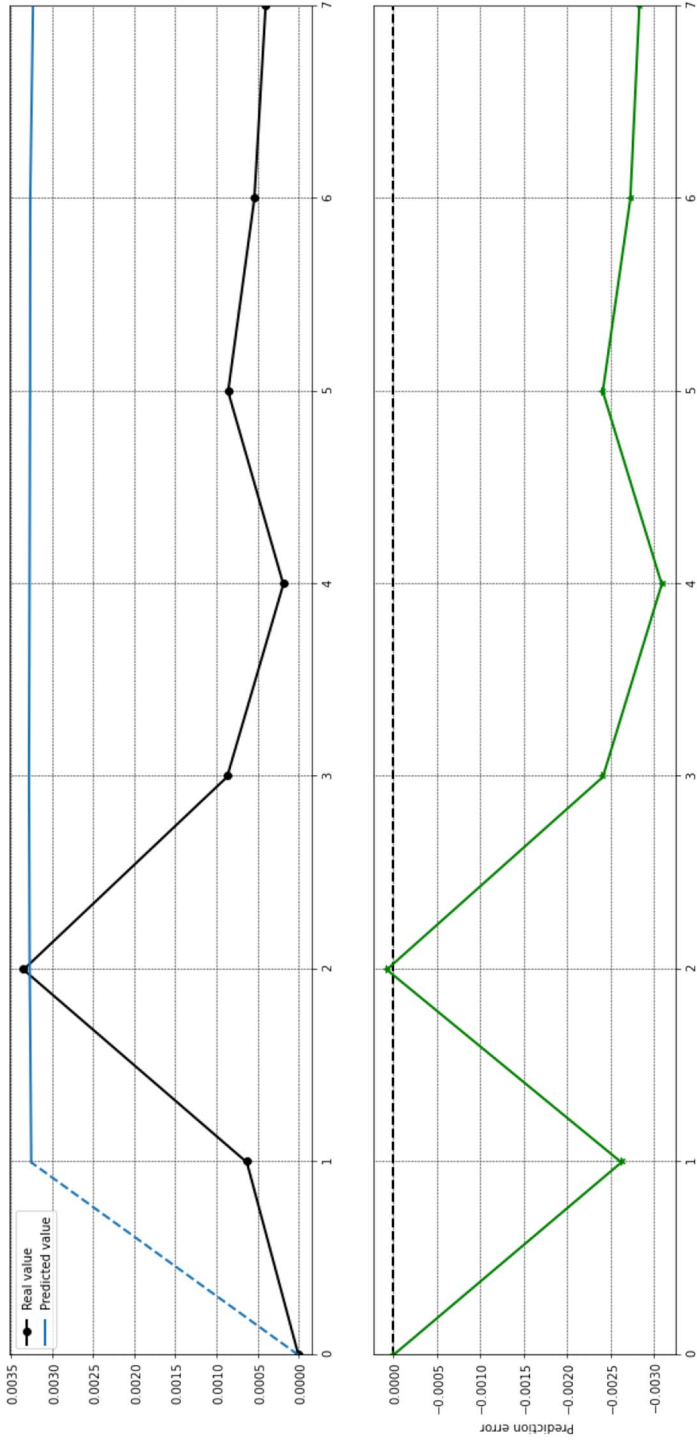


Fig. 3 Forecasting out-of-sample and prediction error comparison: ARCH(4) model

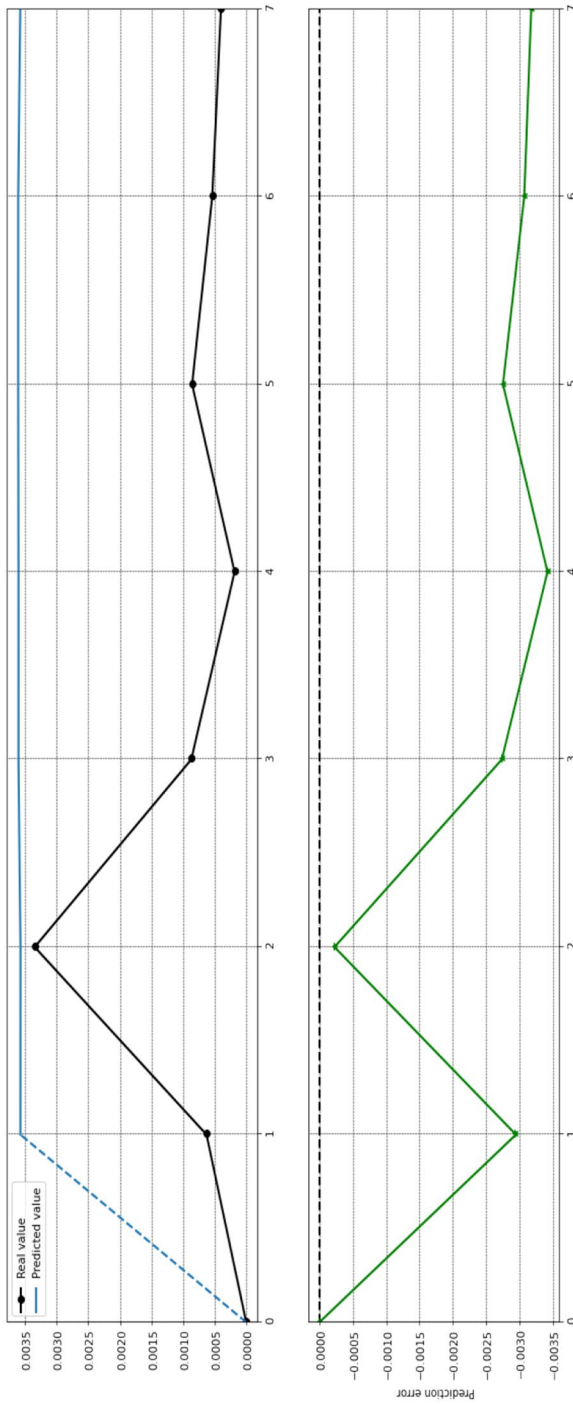
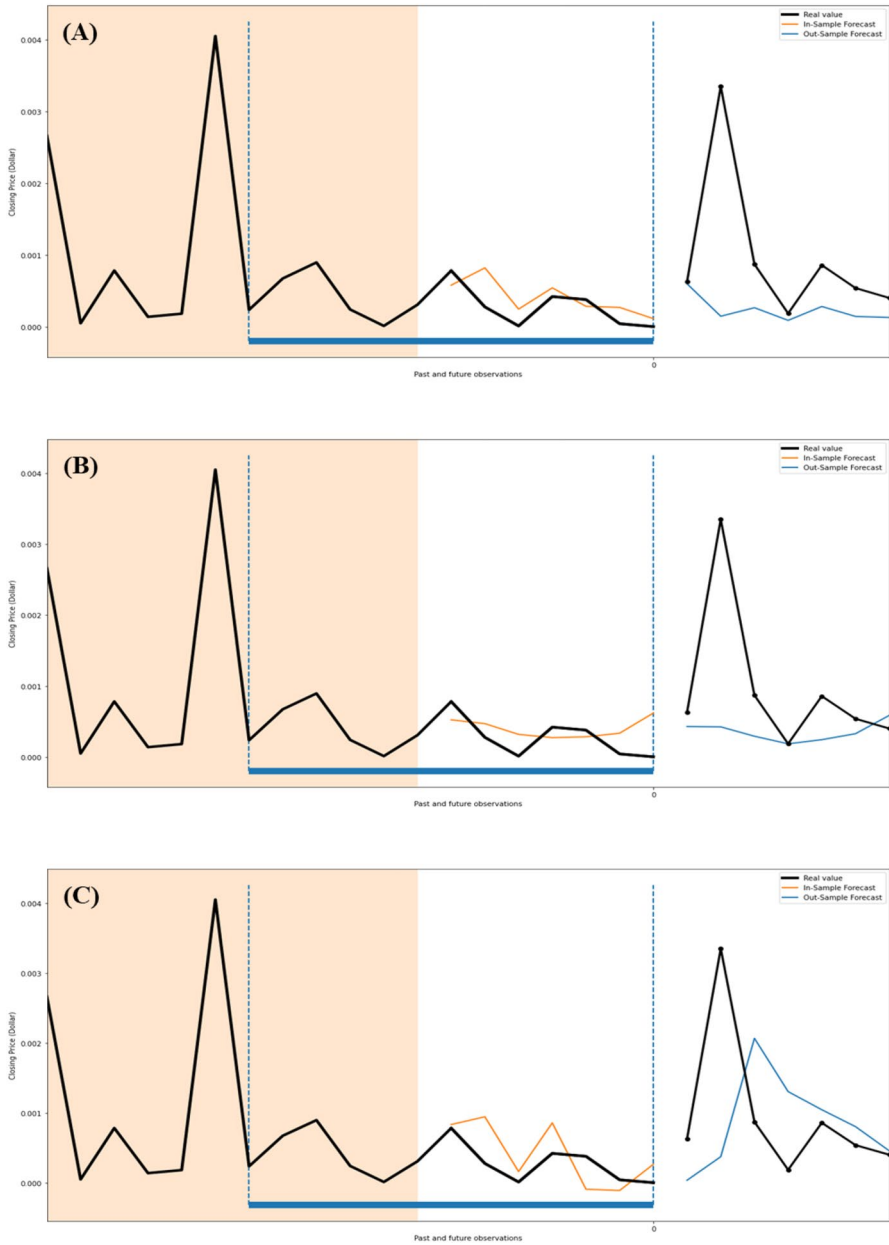


Fig. 4 Forecasting out-of-sample and prediction error comparison: GARCH(4, 2) model



**Fig. 5** Fitting and forecasting DNN models: **A** MLP; **B** RNN; **C** LSTM

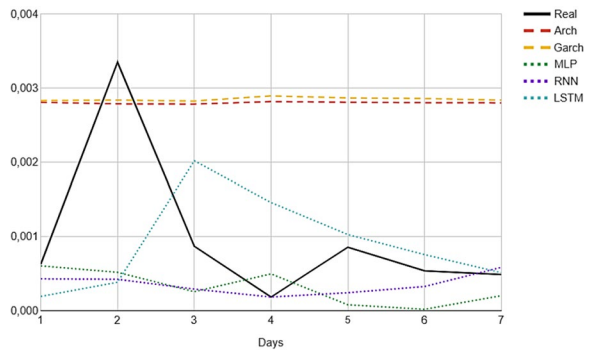
dynamic model with information storage, whereas MLP model produces a static technique of the data. While LSTM model was the most complex, was the lowest performer among the deep learning methodologies. These results are consistent with literature, which suggests that LSTM architecture may underperform when

**Table 2** Model prediction errors by accuracy metrics (MAE and MAPE)

Model	Days	MAE	MAPE (%)
ARCH (4)	1	0.00216	343.67
	3	0.00155	194.45
	7	0.00198	463.81
GARCH (4, 2)	1	0.00244	388.74
	3	0.00164	217.58
	7	0.00219	519.65
MLP	1	0.00003	4.91–5.73 <sup>a</sup>
	3	0.00123	58.28–62.73 <sup>a</sup>
	7	0.00074	52.45–68.63 <sup>a</sup>
RNN	1	0.0002	28.11–32.85 <sup>a</sup>
	3	0.00128	59.91–64.70 <sup>a</sup>
	7	0.00067	45.01–53.39 <sup>a</sup>
LSTM	1	0.00043	62.46–73.42 <sup>a</sup>
	3	0.001517	92.32–103.77 <sup>a</sup>
	7	0.00089	137.38–154.80 <sup>a</sup>

<sup>a</sup>Minimum and maximum values (5% trimmed) obtain from the 200 runs

**Fig. 6** Forecasting out-of-sample of all models



the time series is stationary. However, MLP neural network model demonstrated highest forecasting accuracy and lowest prediction error among all five models investigated in this study, while exhibiting lowest computational costs among the deep learning methodologies.

Deep learning methodologies seem to show advantages over classical methodologies in terms of forecast quality, since nonlinear dependencies of the data are better captured. However, it is noteworthy that these models are associated with considerably higher computational costs and greater implementation complexity compared to classical techniques (corroborating with the scientific literature—Lopes et al., 2021 and Ramos et al., 2021). Despite these limitations, implementation of deep learning models in the present study yielded a substantial reduction in prediction errors. As such, it can be inferred that increased computational



costs associated with deep learning model implementation is justifiable, particularly when considering MLP model—which is the least complex model in terms of computational requirements—provided highest forecast accuracy for the time series studied.

Finally, to infer whether forecast accuracy of these two models is the same, Diebold–Mariano test, with modification suggested by Harvey et al. (1997) was implemented (see Table 3).

With this, for a significance level of 5%, there is statistically significant evidence to suggest that forecasts do not have the same precision and one is significantly better than the other. According to previous information, the MLP model has better forecast accuracy.

These facts are consistent with previous research findings and highlight the importance of these new methodologies and how researchers must be equipped with knowledge about how these models can help to understand economic reality.

## 5 Conclusion

In recent years, Bitcoin has received significant attention from scholars due to its distinctive patterns and characteristics, including high volatility, multiple structural breaks, and unusual probability distributions. However, academic literature has noted a lack of research on this topic. This study contributes to understanding factors underlying Bitcoin volatility by examining price of production (electricity costs), programmed scarcity, programmed supply shocks (halvings), demand shocks (price-hash rate spirals), hash rate, network trust, and liquidation cascades.

Our findings suggest that ARCH(4) and GARCH(4, 2) models are the most effective to forecasting Bitcoin returns. ARCH(4) model performed best in terms of the MAPE metric. Among deep learning approaches, MLP model showed the best performance on shorter time horizons (one-day and three-days), while RNN model had the lowest prediction errors on seven-day horizon. LSTM model, being the most complex, performed weakly among the deep learning methods. Deep learning models have advantages over classical methods in terms of forecast quality, providing an effective capability to capture nonlinear dependencies in the data. However, higher computational costs and implementation difficulties are also involved. Nonetheless, the improvement in prediction errors justifies their implementation, especially considering that the MLP model used in this study is not the most complex or computationally expensive. Our results are consistent with prior research and underscore the significance of these new methodologies for understanding economic reality.

**Table 3** Diebold–Mariano test

Diebold–Mariano	
Statistic	− 3.2724894
<i>p</i> value	0.03345

Although this study has made valuable contributions to understand Bitcoin's returns and volatility factors is also important to recognize its limitations. One limitation is that it focuses on internal mechanisms of protocols as drivers of volatility, with less attention given to market dynamics specific to the cryptocurrency market such as low liquidity, market microstructure, high leverage, and market manipulation. Another limitation is the limited range of ARCH/GARCH models, which may not be the most advanced or effective for forecasting. In addition, models in this study used only one variable and did not consider external factors, which could be important in financial time series with nonlinear properties. Future research could consider a multi-variable perspective that considers derivatives data, on-chain data, and market sentiment data, as well as the use of hybrid models to better understand Bitcoin volatility.

In addition to this, this work also makes a reflective contribution to scientific literature by comparing classical methodologies (ARCH and GARCH models) and deep learning methodologies (DNN models) for returns and volatility forecasting. According to the scientific literature, classical methodologies are still the most used by professionals in economic, financial, and business fields (Wilson & Spralls, 2018). As expected, the results of the analysis show that DNN models have better forecast quality. However, it is important to highlight not only the potential of deep learning methodologies, but also the significant difference in forecast quality. In the economic and financial field, it is noteworthy that professionals often deal with high error rates. Therefore, in an increasingly competitive economic environment, those who use robust tools to support decision-making have an advantage. Therefore, it is important to encourage the training and awareness of these professionals, particularly investors in the cryptocurrency market, to use more accurate methodologies (e.g., deep learning). This is a challenge that this work aims to highlight.

In conclusion, this study aims to provide a valuable contribution to the understanding of Bitcoin's daily returns and the potential of deep learning methodologies. While many researchers have traditionally used classical approaches to volatility models, the recent advancements in computational power suggest that deep learning methodologies may offer a promising option for improving forecast quality. It is important for researchers to consider the use of these advanced methodologies, not only in the study of crypto assets but in other areas as well.

**Acknowledgements** This work is partially financed by national funds through FCT—Fundação para a Ciência e a Tecnologia under the project UIDB/00006/2020. The paper has also benefited from discussions at the 42nd EBES Conference, in Lisbon.

**Author contributions** All authors contributed to the study conception. All authors read and approved the final manuscript.

**Funding** Open access funding provided by FCTIFCCN (b-on). This work is partially financed by national funds through FCT—Fundação para a Ciência e a Tecnologia under the project UIDB/00006/2020.

## Declarations

**Conflict of interest** There is no conflict of interest to disclose.

**Open Access** This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as

you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

## References

- Aminian, F., Suarez, E. D., Aminian, M., & Walz, D. T. (2006). Forecasting economic data with neural networks. *Computational Economics*, 28(1), 71–88.
- Balcilar, M., Bouri, E., Gupta, R., & Roubaud, D. (2017). Can volume predict Bitcoin returns and volatility? A quantiles-based approach. *Economic Modelling*, 64, 74–81.
- Bengio, Y., Simard, P., & Frasconi, P. (1994). Learning long-term dependencies with gradient descent is difficult. *IEEE Transactions on Neural Networks*, 5(2), 157–166.
- Bergsli, L. Ø., Lind, A. F., Molnár, P., & Polasik, M. (2022). Forecasting volatility of Bitcoin. *Research in International Business and Finance*, 59, 101540.
- Bezerra, P. C. S., & Albuquerque, P. H. M. (2017). Volatility forecasting via SVR–GARCH with mixture of Gaussian kernels. *Computational Management Science*, 14(2), 179–196.
- Bishop, C. M. (1995). *Neural networks for pattern recognition*. Oxford University Press.
- Black, F., & Scholes, M. (1973). The pricing of options and corporate liabilities. *Journal of Political Economy*, 81(3), 637–654.
- Bollerslev, T. (1986). Generalized autoregressive conditional heteroskedasticity. *Journal of Econometrics*, 31(3), 307–327.
- Bracke, P., Datta, A., Jung, C., & Sen, S. (2019). *Machine learning explainability in finance: An application to default risk analysis*. Staff Working Paper No. 816, Bank of England. <https://doi.org/10.2139/SSRN.3435104>.
- Brownlee, J. (2018). *Deep learning for time series forecasting: Predict the future with MLPs, CNNs and LSTMs in Python*. Machine Learning Mastery.
- Chaim, P., & Laurini, M. P. (2018). Volatility and return jumps in Bitcoin. *Economics Letters*, 173, 158–163.
- Chatfield, C. (2016). *The analysis of time series: An introduction* (6th ed.). Chapman and Hall/CRC.
- Chen, S., Härdle, W. K., & Jeong, K. (2010). Forecasting volatility with support vector machine-based GARCH model. *Journal of Forecasting*, 29(4), 406–433.
- Chung, J., Gulcehre, C., Cho, K., & Bengio, Y. (2014). Empirical evaluation of gated recurrent neural networks on sequence modeling. ArXiv preprint arXiv:1412.3555.
- Costa, A., Ramos, F. R., Mendes, D., & Mendes, V. (2019). Forecasting financial time series using deep learning techniques. In *IO 2019—XX Congresso da APDIO 2019*. Instituto Politécnico de Tomar - Tomar.
- Diebold, F. X., & Mariano, R. S. (2002). Comparing predictive accuracy. *Journal of Business & Economic Statistics*, 20(1), 134–144.
- Duarte, M., & Watanabe, R. N. (2018). *Notes on scientific computing for biomechanics and motor control*. GitHub. Retrieved from <https://github.com/BMClab/BMC>
- Engle, R. F. (1982). Autoregressive conditional heteroscedasticity with estimates of the variance of United Kingdom inflation. *Econometrica*, 50(4), 987.
- Franses, P. H., & Van Dijk, D. (1996). Forecasting stock market volatility using (non-linear) Garch models. *Journal of Forecasting*, 15(3), 229–235.
- Garcia, D., Tessone, C. J., Mavrodiev, P., & Perony, N. (2014). The digital traces of bubbles: Feedback cycles between socio-economic signals in the bitcoin economy. *Journal of the Royal Society Interface*, 11(99), 20140623.
- Gavrishchaka, V. V., & Banerjee, S. (2006). Support vector machine as an efficient framework for stock market volatility forecasting. *Computational Management Science*, 3(2), 147–160.
- Giudici, P., & Pagnottoni, P. (2019). High frequency price change spillovers in Bitcoin markets. *Risks*, 7(4), 111.

- Giudici, P., & Pagnottoni, P. (2020). Vector error correction models to measure connectedness of bitcoin exchange markets. *Applied Stochastic Models in Business and Industry*, 36(1), 95–109.
- Giudici, P., & Raffinetti, E. (2021). Shapley-Lorenz explainable artificial intelligence. *Expert Systems with Applications*, 167, 114104.
- Glorot, X., & Bengio, Y. (2010). Understanding the difficulty of training deep feedforward neural networks. In *JMLR workshop and conference proceedings*. Retrieved from <http://www.ijro.umontreal>.
- Gronwald, M. (2019). Is Bitcoin a commodity? On price jumps, demand shocks, and certainty of supply. *Journal of International Money and Finance*, 97, 86–92.
- Hang, N. T. (2019). Research on a number of applicable forecasting techniques in economic analysis, supporting enterprises to decide management. *World Scientific News*, 119, 52–67.
- Harvey, D., Leybourne, S., & Newbold, P. (1997). Testing the equality of prediction mean squared errors. *International Journal of Forecasting*, 13(2), 281–291.
- Hayes, A. S. (2017). Cryptocurrency value formation: An empirical study leading to a cost of production model for valuing bitcoin. *Telematics and Informatics*, 34(7), 1308–1321.
- Hochreiter, S., & Schmidhuber, J. (1997). Long short-term memory. *Neural Computation*, 9(8), 1735–1780.
- Hodrick, R. J., & Prescott, E. C. (1997). Postwar U.S. business cycles: An empirical investigation. *Journal of Money, Credit and Banking*, 29(1), 1–16.
- Hopfield, J. J. (1982). Neural networks and physical systems with emergent collective computational abilities. *Proceedings of the National Academy of Sciences of the United States of America*, 79(8), 2554–2558.
- 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.
- Kingma, D. P., & Ba, J. L. (2015). Adam: A method for stochastic optimization. In *3rd International conference on learning representations, ICLR 2015—Conference track proceedings*. International conference on learning representations, ICLR.
- Klose, J. (2022). Comparing cryptocurrencies and gold - a system-GARCH-approach. *Eurasian Economic Review*, 12(4), 653–679.
- 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.
- Liu, Y. (2019). Novel volatility forecasting using deep learning—Long short term memory recurrent neural networks. *Expert Systems with Applications*, 132, 99–109.
- Lopes, D. R., Ramos, F. R., Costa, A., & Mendes, D. (2021). Forecasting models for time-series: A comparative study between classical methodologies and Deep Learning. In *SPE 2021 – XXV Congresso da Sociedade Portuguesa de Estatística*. Évora - Portugal.
- Luong, C., & Dokuchaev, N. (2018). Forecasting of realised volatility with the random forests algorithm. *Journal of Risk and Financial Management*, 11(4), 61.
- Malhotra, P., Vig, L., Shroff, G., & Agarwal, P. (2015). Long short term memory networks for anomaly detection in time series. In *The European symposium on artificial neural networks*.
- Mallqui, D. C. A., & Fernandes, R. A. S. (2019). Predicting the direction, maximum, minimum and closing prices of daily Bitcoin exchange rate using machine learning techniques. *Applied Soft Computing*, 75, 596–606.
- Markowitz, H. (1952). Portfolio selection. *The Journal of Finance*, 7(1), 77–91.
- Mirikitani, D. T., & Nikolaev, N. (2010). Recursive Bayesian recurrent neural networks for time-series modeling. *IEEE Transactions on Neural Networks*, 21(2), 262–274.
- Pagnotta, E., & Buraschi, A. (2018). An equilibrium valuation of Bitcoin and decentralized network assets. *SSRN Electronic Journal*. <https://doi.org/10.2139/ssrn.3142022>.
- Pesaran, M. H., & Timmermann, A. (2004). How costly is it to ignore breaks when forecasting the direction of a time series? *International Journal of Forecasting*, 20(3), 411–425.
- Pichl, L., Kaizoji, T., Pichl, L., & Kaizoji, T. (2017). Volatility analysis of Bitcoin price time series. *Quantitative Finance and Economics*, 1(4), 474–485.
- Pilbeam, K., & Langeland, K. N. (2015). Forecasting exchange rate volatility: GARCH models versus implied volatility forecasts. *International Economics and Economic Policy*, 12(1), 127–142.

- Pineda, F. (1987). Generalization of back propagation to recurrent and higher order neural networks. *Undefined*.
- Pratas, T. (2022). *Forecasting Bitcoin's volatility: Exploring the potential of deep-learning*. Instituto Universitário de Lisboa - ISCTE Business School, Lisboa, Portugal. Retrieved from <https://repositorio.iscte-iul.pt/handle/10071/26641>.
- Qin, Y., Song, D., Chen, H., Cheng, W., Jiang, G., & Cottrell, G. (2017). A dual-stage attention-based recurrent neural network for time series prediction. ArXiv preprint arXiv:1704.02971. Retrieved from <https://arxiv.org/abs/1704.02971v4>.
- Ramos, F. R. (2021). *Data science in economic-financial time series modeling and forecasting: from classical methodologies to deep learning*. PhD Thesis, Instituto Universitário de Lisboa - ISCTE Business School, Lisboa, Portugal.
- Ramos, F. R., Costa, A., Mendes, D., & Mendes, V. (2018). Forecasting financial time series: A comparative study. In *JOCLAD 2018, XXIV Jornadas de Classificação e Análise de Dados*. Escola Naval – Alfeite.
- Ramos, F. R., Lopes, D. R., Costa, A., & Mendes, D. (2021). Exploiting the memory power of LSTM neural networks in modeling and forecasting the PSI 20. In *SPE 2021—XXV Congresso da Sociedade Portuguesa de Estatística*. Évora - Portugal.
- Ramos, F. R., Lopes, D. R., & Pratas, T. E. (2022). Deep neural networks: A hybrid approach using box&jenkins methodology. *Innovations in mechatronics engineering II. icieng 2022. Lecture notes in mechanical engineering* (pp. 51–62). Springer.
- Rubio, L., & Alba, K. (2022). Forecasting selected colombian shares using a hybrid ARIMA-SVR model. *Mathematics*, 10(13), 2181.
- Rumelhart, D. E., Hinton, G. E., & Williams, R. J. (1986). Learning representations by back-propagating errors. *Nature*, 323(6088), 533–536.
- Senarathne, C. W. (2019). The leverage effect and information flow interpretation for speculative bitcoin prices: Bitcoin volume vs ARCH effect. *European Journal of Economic Studies*, 8(1), 77–84.
- Sezer, O. B., Gudelek, M. U., & Ozbayoglu, A. M. (2020). Financial time series forecasting with deep learning: A systematic literature review: 2005–2019. *Applied Soft Computing*, 90, 106–181.
- Smyl, S. (2020). A hybrid method of exponential smoothing and recurrent neural networks for time series forecasting. *International Journal of Forecasting*, 36(1), 75–85.
- Taleb, N. N. (2021). Bitcoin, currencies, and fragility. *Quantitative Finance*, 21(8), 1249–1255.
- Tang, L. B., Tang, L. X., & Sheng, H. Y. (2009). Forecasting volatility based on wavelet support vector machine. *Expert Systems with Applications*, 36(2), 2901–2909.
- Tealab, A. (2020). Time series forecasting using artificial neural networks methodologies: A systematic review. *Future Computing and Informatics Journal*, 3(2), 334–340.
- Tkáč, M., & Verner, R. (2016). Artificial neural networks in business: Two decades of research. *Applied Soft Computing*, 38, 788–804.
- Wang, P., Xu, K., Ding, Z., Du, Y., Liu, W., Sun, B., et al. (2022). An online electricity market price forecasting method via random forest. *IEEE Transactions on Industry Applications*, 58(6), 7013–7021.
- Willmott, C., & Matsuura, K. (2005). Advantages of the mean absolute error (MAE) over the root mean square error (RMSE) in assessing average model performance. *Climate Research*, 30(1), 79–82.
- Wilson, J. H., & Spralls, S. A., III. (2018). What do business professionals say about forecasting in the marketing curriculum? *International Journal of Business, Marketing, & Decision Science*, 11(1), 1–20.
- Yermack, D. (2015). Is Bitcoin a real currency? An economic appraisal. In D. Lee Kuo Chuen (Ed.), *Handbook of digital currency* (pp. 31–43). Academic Press.