



Forecasting volatility by using wavelet transform, ARIMA and GARCH models

Lihki Rubio¹  · Adriana Palacio Pinedo¹  · Adriana Mejía Castaño¹  · Filipe Ramos² 

Received: 29 March 2023 / Revised: 19 September 2023 / Accepted: 14 October 2023 /

Published online: 10 December 2023

© The Author(s) 2023

Abstract

Forecasting volatility of certain stocks plays an important role for investors as it allows to quantify associated trading risk and thus make right decisions. This work explores econometric alternatives for time series forecasting, such as the ARIMA and GARCH models, which have been widely used in the financial industry. These techniques have the advantage that training the models does not require high computational cost. To improve predictions obtained from ARIMA, the discrete Fourier transform is used as ARIMA pre-processing, resulting in the wavelet ARIMA strategy. Due to the linear nature of ARIMA, non-linear patterns in the volatility time series cannot be captured. To solve this problem, two hybridisation techniques are proposed, combining wavelet ARIMA and GARCH. The advantage of applying this methodology is associated with the ability of each to capture linear and non-linear patterns present in a time series. These two hybridisation techniques are evaluated to verify which provides better prediction. The volatility time series is associated with Tesla stock, which has a highly volatile nature and it is of major interest to many investors today.

Keywords Volatility · ARIMA · GARCH · Wavelet transform · Hybrid model

✉ Filipe Ramos
frramos@fc.ul.pt

Lihki Rubio
lihkir@uninorte.edu.co

Adriana Palacio Pinedo
mapalacio@uninorte.edu.co

Adriana Mejía Castaño
mejiala@uninorte.edu.co

¹ Department of Mathematics and Statistics, Universidad del Norte, Km. 5 vía Puerto Colombia, 081007 Barranquilla, Atlántico, Colombia

² CEAUL-Centro de Estatística e Aplicações, Faculdade de Ciências, Universidade de Lisboa, 1749-016 Lisbon, Portugal

Abbreviations

ARIMA	Autoregressive integrated moving average
GARCH	Generalized autoregressive conditional heteroskedasticity
DWT	Discrete wavelet transform
IDWT	Inverse discrete wavelet transform
\mathcal{A}_{it}	i th approximation component from DWT at the time t
\mathcal{D}_{it}	i th detailed component from DWT at the time t
y_t, \tilde{y}_t	Time series observation and prediction at time t
$\tilde{h}_t^{\text{w-ARIMA}}$	W-ARIMA prediction obtained by applying ARIMA model to approximate (\mathcal{A}_{it}) and detailed (\mathcal{D}_{it}) components from (DWT) decomposition
$\tilde{h}_t^{\text{w-GARCH}}$	W-GARCH prediction obtained by applying GARCH model to approximate (\mathcal{A}_{it}) and detailed (\mathcal{D}_{it}) components from (DWT) decomposition
$\tilde{h}_t^{\text{w-ARIMA-GARCH}}$	W-ARIMA-GARCH prediction obtained by applying ARIMA to approximate components (\mathcal{A}_{it}) and GARCH to detailed components (\mathcal{D}_{it}) from (DWT) decomposition
$\tilde{h}_t^{\text{w-GARCH-ARIMA}}$	W-GARCH-ARIMA prediction obtained by applying GARCH to approximate components (\mathcal{A}_{it}) and ARIMA to detailed components (\mathcal{D}_{it}) from (DWT) decomposition
$\tilde{h}_t^{\text{ARIMA-GARCH}}$	ARIMA-GARCH prediction obtained by adding ARIMA forecast from the original time series y_t and GARCH prediction for its residuals
$\tilde{h}_t^{\text{GARCH-ARIMA}}$	GARCH-ARIMA prediction obtained by adding GARCH forecast from the original time series y_t and ARIMA prediction for its residuals
$\tilde{h}_t^{\text{w-ARIMA-SV}}$	Prediction obtained by applying ARIMA to approximate components (\mathcal{A}_{it}) and SV to detailed components (\mathcal{D}_{it}) from (DWT) decomposition

1 Introduction

With the creation of multiple financial indices and the increasing number of companies listing their shares on stock exchanges, the demand for forecasting models to quantify inherent risk in each of such instruments has grown considerably. Financial firms and traders are frequently concerned about the risks linked to the increasing volatility of stocks, such as Tesla, Inc. (TSLA) listed in Nasdaq, whose volatility will be predicted in this paper for illustrative purposes due to high volatility present in this stock. Tesla shares rose strongly at the time of the COVID-19 pandemic. However, the month-on-month decrease in the stock is deeper than the Represent the Bitcoin symbol (BTC) price drop. The problem is speculated to be associated with the CEO's sale of a portion of his stake to help fund new acquisitions. Investors are now questioning whether the sale has been finalized, which is causing high volatility in the stock.

On the other hand, accurate volatility prediction is a crucial part of risk management, most notably in asset allocation to diverse investment portfolios to adequately hedge the underlying risk. High volatility indicates that either the market is at risk or that securities values are unreliable and that capital markets are not performing well enough. Gaining a more accurate understanding of volatility, predicting it with accuracy and controlling portfolio exposure and its impact are essential to effective trading.

Different authors have studied volatility forecasting using statistical and machine learning models. For example, some of them have opted to use wavelet transform support vector machine (WSVM) to outperformed solutions obtained from (SVM) (Tang et al., 2009). Comparative studies on different versions of the Generalized Autoregressive Conditional Heteroskedasticity (GARCH) model, such as Cholesky Generalized Autoregressive Conditional Heteroskedasticity (CGARCH), Integrated Generalized Autoregressive Conditional Heteroskedasticity (IGARCH) and Fractionally Integrated Generalized Autoregressive Conditional Heteroskedasticity (FIGARCH), to detect which one delivers the best performance have been developed (Kang et al., 2009). Vector autoregressive models such as the VARIMA and VGARCH model along with aggregated causality factors have also been used in volatility forecasting (Hafner, 2009). The use of particle filtering (PF) techniques has also been applied in volatility forecasting for high frequency data, which contains patterns difficult to capture by classical statistical models (Gaoyu et al., 2009). In addition, techniques based on artificial neural networks together with classical time series models have been used in volatility forecasting and electricity demand to improve prediction accuracy (Hyup Roh, 2007; Luzia et al., 2023).

The main problem with volatility prediction lies in the fact that strong nonlinear patterns may be present in the associated time series. Therefore, linear models such as ARIMA or Exponential Smoothing (ES), which are widely used in statistics, for example to predict earnings before interest, taxes, depreciation and amortization (EBITA) index, mid-long-term electric energy consumption and exchange-rates (de Oliveira & Cyrino Oliveira, 2018; Maria & Eva, n.d.; Rubio et al., 2021) others are unable to predict volatility with high efficiency. These types of models are defined in terms of linear filters considering a finite number of historical data and residuals, to fit linear patterns in a time series. An alternative way to improve goodness-of-fit is to use moving average or discrete Fourier transform as pre-processing (Al Wadi et al., 2010; Alshammari et al., 2020).

Another option is to use machine learning (ML) or deep learning (DL) techniques which are able to capture strong fluctuations in a time series or have important properties associated with long term time dependencies, such as support vector regression (SVR) and long short-term memory (LSTM) models, respectively (Bathla, 2020; Chniti et al., 2017; Guo et al., 2019). A principal concern associated with the use of these approaches is the high computational cost required to train these models. GARCH stochastic processes have shown great effectiveness in predicting time series with implicit heteroscedasticity, such as volatility, and their evaluation does not require large computational resources. Therefore, the use of this technique on its own or in combination with another method will provide predictions with adequate scores for volatility forecasting.

In this paper, a hybridization technique based on the wavelet transform, using the ARIMA and GARCH models, is proposed. This technique uses selected models for each of the time series obtained from the decomposition into high and low frequency signals. Time series with high frequency are predicted using GARCH and those with low frequency are predicted using ARIMA or wavelet ARIMA. In this way, advantages offered by each technique when predicting time series volatility are maximized. A comparative study between this type of hybridization and other related techniques is performed to identify which approach provides the best combination of these predictive models.

The primary objective of this work is to demonstrate that the model, which incorporates wavelet transform, ARIMA, and GARCH, with the application of these models to both approximate and detailed components, yields superior predictions compared to other hybrid models utilized for volatility forecasting. To achieve this goal, various iterations of hybrid models, integrating wavelet transform and the incorporation of predicted residuals, are assessed. This assessment aims to establish that the suggested model provides the most favourable outcomes in terms of error rates and goodness of fit.

Having described the background of this research, the organization of the other sections shall be as follows: Sect. 2 includes a review of the literature associated with the problem addressed in this work. Section 3 describes formally predictive models used for volatility forecasting and associated important definitions. Section 4 introduce data set used for time series volatility estimation, its associated descriptive analysis. Section 5 and 6 present the methodology proposed for the optimal combination of ARIMA and GARCH models, experimental results obtained and discussions related. Section 7 present conclusions, future and on-going work.

2 Literature review

As an alternative to ARIMA, wavelet ARIMA model has been used by different authors for volatility forecasting using diverse approaches. For example, it has been shown that wavelet ARIMA provides better predictions than ARIMA for forecasting Amman stocks. In this case, ARIMA was applied to each of the Daubechies wavelets obtained from the decomposition (Al Wadi et al., 2010). Other authors use for example, maximal overlap discrete wavelet transform (MODWT) and hybrid ARIMA models with feedforward neural network FNN for volatility forecasting. In this case, it was found that the hybrid ARIMA-FNN model provides better predictions in terms of goodness-of-fit. ARIMA-FNN uses first ARIMA to fit linear patterns in the time series, then residuals are predicted using the FNN model. In this fashion, FNN adjusts prediction errors obtained from ARIMA (Xiao et al., 2014).

Although wavelet transform is crucial to improve volatility predictions obtained from ARIMA, there are some strong fluctuations that cannot be captured due to its linear structure. Some authors have opted for using machine learning and deep learning techniques to solve this problem. For example, using ML models such as neural networks and random forests for volatility prediction of the Dow Jones Industrial Average index stocks yields considerable predictive gains, since these models can

effectively capture multiple highly nonlinear patterns in a time series, which cannot be fitted by models such as heterogeneous autoregressive (HAR) (Christensen et al., 2021). Support vector machine models dominated the area of machine learning in the 1990s until 2010, displacing techniques based on artificial neural networks. After that, the deep learning-based techniques began to dominate, thanks to the current availability of computing power, advances in computer architectures, as well as the accumulation of big data. However, the SVM model has a great advantage over alternative models, as it is able to capture long-memory nonlinear patterns in time series such as the volatility of the S&P 500 index, it is additionally effective for high dimensionality data processing. SVM has been compared with classical models used in econometrics, for example, multiple versions of the GARCH model, obtaining comparable or better predictions using SVM than those obtained using the GARCH models (Gavrishchaka & Banerjee, 2006).

On the other hand, some authors prefer to use deep learning techniques such as long short-term memory (LSTM) for volatility prediction. It has been found that these techniques provide similar or better predictions in some cases to those provided by the support vector regression (SVR) model, which is highly suggested in the literature for this type of forecasting, as it is effective when it comes to adjusting marked patterns of heteroskedasticity. The use of LSTM is highlighted by the fact that it can be trained using graphics cards to reduce computational cost (Liu, 2019). Recent research has also considered to make use of stacked machine learning models based on artificial neural networks to forecast volatility of the S&P 500 index. In this case, predictions are obtained by evaluating under an activation function, forecasts obtained through Gradient Descent Boosting (GDC), Random Forest (RF) and Support Vector Machine (SVM) models, to obtain a single stacked forecast. This model has proven to be competitive with the classical models used individually (Ramos-Pérez et al., 2019).

Another alternative used to forecast realized volatility is to apply the hybrid ARIMA-GARCH model to each signal obtained from MODWT decomposition. The final forecast is obtained by adding forecasts from each MODWT wave component. Authors highlight advantages of using MODWT as a pre-processor for hybrid models such as ARIMA-GARCH to predict volatility of the Saudi Arabia stock market (Alshammari et al., 2020).

Several successful applications of wavelet transform-based methods has been used in finance applications. One such example is enhancing stock index prediction accuracy using a closed recurrent unit (GRU) neural network in conjunction with adaptive noise decomposition (CEEMDAN-wavelet). The results demonstrate that the GRU-CEEMDAN-wavelet approach yields lower errors compared to ARIMA and individual models (Qi et al., 2023). On the other hand, Li and Tang (2020) proposed the WT-FCD-MLGRU model to enhance accuracy of forecasting financial time series linked to stock indices. This model integrates wavelet transform, filter cycle decomposition, and multilag neural networks. Empirical analysis reveals that the WT-FCD-MLGRU model outperform alternative approaches in forecast precision, showing the lowest error when predicting stock indices, as compared to conventional models as ARIMA and the enhanced SVR machine learning model. Wang and Guo (2020) introduces the hybrid model DWT-ARIMA-GSXGB. The authors

use discrete wavelet transform to partition the dataset into approximation and error components. Subsequently, ARIMA is applied to the approximate partial data, while the enhanced XGBoost model (GSXGB) is used for the error data. Through experimental comparisons with stock market data, it was found that the DWT-ARIMA-GSXGB model exhibits lower errors when contrasted with four prediction models: ARIMA, XGBoost, GSXGB, and DWT-ARIMA-XGBoost. A hybrid model which integrates the Empirical Wavelet Transform (EWT) with the Improved Bee Colony Algorithm (ABC), the Extreme Learning Machine (ELM) neural network, and ARIMA has been proposed (Yu et al., 2020). EWT is employed to decompose, clean the data and removing noise, making it suitable for forecasting. Subsequently, the optimized ELM using GPS-EO-ABC and ARIMA are separately applied to generate diverse prediction outcomes, which are then combined with weighting. The optimized ELM demonstrated superior accuracy and stability when compared to the original ELM, ABC-ELM, LSTM, and ANN. This hybrid approach proves to be effective in not only prediction but also efficient in noise reduction and outlier correction within the data.

Each of the studies mentioned previously have proposed novel techniques for volatility forecasting, with a common purpose, to find a technique to get accurate and trustworthy predictions for this type of data, which has a large amount of noise and strong behaviour difficult to adjust, due to different external factors and current market conditions. The literature review did not reveal the existence of a model that combines wavelet transform, ARIMA, and GARCH for volatility prediction. In this proposed approach, ARIMA and GARCH are specifically employed for the approximate and detailed components derived from discrete wavelet transform.

3 Forecasting models

3.1 ARIMA model

ARIMA models, popularized by Box and Jenkins, are a flexible and powerful statistical tool for predictive modelling with time series data (Asteriou & Hall, 2016). Mainly, ARIMA models approximate time series future values as a linear function of past observations and white noise terms. The model consists of three components: non-stationary differences for stationarity, autoregressive model (AR) and moving average (MA) model (Montgomery et al., n.d.).

To define non-stationarity, the backshift operator, B is introduced. A time series, y_t , will be called homogeneous non-stationary if it is non-stationary but its first difference, i.e. $w_t = y_t - y_{t-1} = (1 - B)y_t$ or d th difference, $w_t = (1 - B)^d y_t$, yields a stationary time series. In addition, y_t will be called an autoregressive integrated moving average (ARIMA) process of orders p , d , and q , denoted ARIMA (p, d, q) if its d th difference yields a stationary process ARMA (p, q) . Therefore, an ARIMA (p, d, q) can be written as:

$$\Phi(B)(1 - B)^d y_t = \delta + \Theta(B)\varepsilon_t, \quad (1)$$

where:

$$\Phi(B) = 1 - \sum_{i=1}^p \phi_i B^i, \Theta(B) = 1 - \sum_{i=1}^q \theta_i B^i \quad (2)$$

are the backshift operator terms in the AR(p) and MA(q) defined as: $\Phi(B)y_t = \delta + \varepsilon_t$ and $y_t = \mu + \Phi(B)\varepsilon_t$, with $\delta = \mu - \phi\mu$, where μ is the mean and ε_t the white noise with $E(\varepsilon_t) = 0$.

Model orders p, q are determined by the nature of the autocorrelation and partial autocorrelation functions. The model coefficients are calculated using the maximum likelihood method (Box et al., 2008). The best model is identified by diagnostic checks such as the Akaike information criterion (AIC), the Bayes information criterion (BIC) and the Jarque–Bera normality test on the residual error series.

3.2 Wavelet-based ARIMA

The wavelet-based ARIMA is a composite technique, in which the raw data is decomposed first using wavelet transformation (WT) and then forecasting models are applied. The WT decomposes a signal into different time scales. It is defined as a set of basis functions $\psi_{a,b}(t)$ that can be generated by translating and scaling the so-called mother wavelet (Daubechies, 1992)

$$\psi_{a,b} = \frac{1}{\sqrt{a}} \psi\left(\frac{t-a}{a}\right), a > 0, -\infty < b < \infty, \quad (3)$$

where a is the scale parameter and b the location of the wavelet. There are several types of WT for which different mothers can be used, in this work we will concentrate on the mostly used Discrete Wavelet Transform (DWT).

A DWT is a discrete set of wavelet scales and translations. The DWT is adapted mainly for samples. Although, this transform decomposes the signal into a multi-orthogonal set of wavelets. DWT uses a dyadic grid, where the mother wavelet is scaled by the power two ($a = 2^j$) and shifted by an integer ($b = k2^j$), where k is a location index ranges from 1 to $2^{-j}N$ (N number of observations) and j from 0 to J (J total number of scales). DWT is given by the following equation:

$$\Psi_{j,k}(t) = 2^{-j/2} \Psi(2^{-j}t - k), \quad (4)$$

where the coefficients are calculated according to the following expression:

$$W_{j,k} = W(2^j, k2^j) = 2^{-j/2} \int_{-\infty}^{\infty} f(t) \overline{\Psi(2^{-j}t - k)} dt \quad (5)$$

The inverse discrete wavelet transform (IDWT) mean is then calculated to re-construct the original signal from the wavelet coefficients $W_{j,k}$ as follows:

$$f(t) = \sum_{j=-\infty}^{\infty} \sum_{k=-\infty}^{\infty} W_{j,k} \Psi_{j,k}(t) \tag{6}$$

There are different mother wavelets in the DWT, for example the Haar wavelet, the Daubechies wavelet, the orthogonal wavelet, the Symlet wavelet, the Meyer wavelet and the Coiflets wavelet (Mallat & Peyré, n.d.). The Haar wavelet is considered the simplest mother wavelet, while the Daubechies wavelet is a set of orthogonal wavelets. In this paper, Daubechies mother wavelets will be used for our first pre-processing. Symlet and Coiflet are defined as a modified version of Daubechies wavelets. On the other hand, Symlet is a Daubechies wavelet with a higher symmetry.

Multi-Resolution Analysis (MRA) also called pyramid algorithm, is defined by a hierarchical representation of DWT (Mallat, n.d.). This is based on the decomposition of the raw data into m levels by translation and convolution of the mother wavelet using low-pass (LP) and high-pass (HP) filters. Detail (D) and approximation (A) components are kept using these filters. The signal can be reconstructed through the sum of the last approximation component and every detail component.

The wavelet transform is employed to eliminate noise from the time series, thereby enhancing the stability of the data structure. As an example, consider a time series y_t . By performing Daubechies wavelet filtering on y_t , two sets of filtered and decomposed series are generated, the (LP) and (HP) versions, named the i -th approximate and detailed components at the time t , denoted by \mathcal{A}_{it} and \mathcal{D}_{it} , respectively (see Fig. 1).

Each component can be further filtered to obtain a second level for each one. The decomposed time series is predicted by applying the according time series model to each component. For example, the ARIMA model described in Sect. 3.1 can be used to predict the \mathcal{D}_{it} and \mathcal{A}_{it} components, the final predictive model for the original time series, denoted by w -ARIMA can be obtained by the following addition.

$$\tilde{y}_t^{w\text{-ARIMA}} = \tilde{\mathcal{A}}_{2t}^{\text{ARIMA}} + \tilde{\mathcal{D}}_{2t}^{\text{ARIMA}} + \tilde{\mathcal{D}}_{1t}^{\text{ARIMA}} \tag{7}$$

where each term of the sum is obtained from the inverse reconstruction of signals (IDWT) (see Fig. 2). This model uses Daubechies filter as pre-processing to reduce noise in the time series data and then apply ARIMA to each component, to obtain higher prediction accuracy than the classical ARIMA.

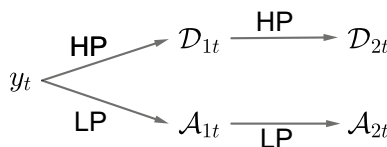


Fig. 1 Multi-resolution analysis applied to the original time series y_t , where \mathcal{A}_{it} represent the approximation components, and \mathcal{D}_{it} , the detailed. (HP) is the high pass filter and (LP) is the low pass filter

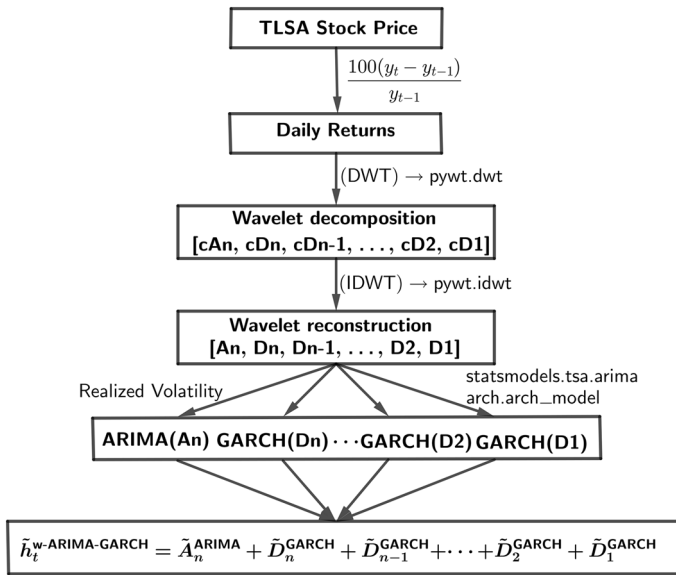


Fig. 2 Flow chart for Python implementation of partitioned models to forecast volatility using PyWavelets, statsmodels.tsa.arima and arch.arch model libraries

3.3 GARCH models

Heteroscedastic conditional autoregressive (ARCH) models first studied by Engle (1982) have been widely used for high volatility forecasting. These models effectively describe the variance changing over time i.e. heteroscedasticity, by employing a deterministic mapping based on historical errors. Formally, the ARCH model of order $p \geq 0$ is defined as follows:

$$y_t = \sqrt{h_t} \varepsilon_t, h_t = \alpha_0 + \sum_{i=1}^p \alpha_i y_{t-i}^2 \tag{8}$$

The generalized autoregressive conditional heteroskedasticity (GARCH) model can be considered as a generalization of the ARCH model defined by Eq. (9). GARCH aggregates linear combination of conditional variance squares and for orders $p \geq 0$ and $q \geq 0$ is given by

$$y_t = \sqrt{h_t} \varepsilon_t, h_t = \alpha_0 + \sum_{i=1}^p \alpha_i y_{t-i}^2 + \sum_{j=1}^q \beta_j h_{t-j}^2, \tag{9}$$

where $\varepsilon_t \sim \text{iid}(0, 1)$, $\alpha_0 > 1$, $\alpha_i \geq 0, i = 1, 2, \dots, p$, $\beta_j \geq 0, j = 1, 2, \dots, q$,

$$\sum_{i=1}^p \alpha_i + \sum_{j=1}^q \beta_j < 1, \quad (10)$$

and ε_t, y_{t-j} are independent for $j \geq 1$. The stochastic process defined by Eq. (9) is known as the GARCH process of order p, q and is denoted as GARCH (p, q). Let Σ_t be a σ -field associated with the sequence $\{\varepsilon_t, \varepsilon_{t-1}, \dots\}$, hence, $\mathbb{E}(\varepsilon_t^2 | \Sigma_t) = h_t$, i.e. conditional variance of h_t is time varying rather than constant as volatility time series studied in this work.

Given that linear GARCH models are not efficient in capturing possible asymmetries in a time series, one alternative is to use non-linear GARCH techniques, one of these is the Exponential GARCH approach, denoted as (EGARCH), which is defined as follows:

$$\log(h_t) = \alpha_0 + \sum_{i=1}^p \left\{ \frac{|\varepsilon_{t-i}|}{\sqrt{h_{t-i}}} - \sqrt{\frac{2}{\pi}} \right\} + \omega \frac{\varepsilon_{t-i}}{\sqrt{h_{t-i}}} + \sum_{j=1}^q \beta_j \log(h_{t-j}), \quad (11)$$

where $\alpha_i, \beta_j, i = 1, 2, \dots, p, j = 1, 2, \dots, q$ and ω are positive constants. The Akaike information criteria (AIC) and Bayesian information criteria (BIC) were used to find the best model in terms of goodness of fit, as well as to add penalty criteria to the number of selected parameters under minimization of these constants. For estimation of the associated parameters, the maximum likelihood approach (MLE) was used. Forecasts were carried out using one-step-ahead predictions for a fixed forecast horizon.

3.4 Hybrid and wavelet models

3.4.1 Hybrid model

To improve predictions obtained by each predictive model individually, hybrid models are proposed, which exploit the advantages of each model and have been demonstrated to be effective in different applications (Khashei & Bijari, 2011, 2012; Rubio & Alba, 2022; Wang et al., 2013; Zhang, 2003). The first type of hybrid model used in this work consider for example the use of ARIMA to extract linear components in the time series and then forecast the associated residuals, being non-linear data with heteroscedasticity, using for example the GARCH model. With this strategy, prediction of linear and nonlinear patterns will be outperformed, thus obtaining a final forecast with higher accuracy. Hybrid model \mathcal{Y}_t is then expressed as:

$$\mathcal{Y}_t = \tilde{\mathcal{L}}_t + \tilde{\mathcal{N}}_t, \quad (12)$$

where $\tilde{\mathcal{L}}_t$ and $\tilde{\mathcal{N}}_t$ correspond to the linear and nonlinear components part of the decomposition of the time series data. Let $\tilde{\mathcal{L}}_t$ be the forecast obtained from ARIMA in the volatility time series at t , hence, the corresponding residual ε_t are given by

$$\hat{\varepsilon}_t = \mathcal{Y}_t - \hat{\mathcal{L}}_t, \quad (13)$$

and are predicted applying the GARCH model and may be represented as

$$\hat{\varepsilon}_t = \hat{f}_{\text{GARCH}}(\hat{\varepsilon}_{t-1}, \hat{\varepsilon}_{t-2}, \dots, \hat{\varepsilon}_{t-n}) + \Delta_t, \tag{14}$$

where \hat{f}_{GARCH} is the nonlinear mapping corresponding to the GARCH model with random error Δ_t and its forecast can be represented as \tilde{N}_t . Accordingly, the hybrid model (ARIMA-GARCH) is represented by

$$\hat{Y}_t = \tilde{L}_t + \tilde{N}_t, \tag{15}$$

where \tilde{L}_t and \tilde{N}_t represent predicted results of both linearity and nonlinearity based on ARIMA and GARCH models, correspondingly. In the present work a second alternative (GARCH-ARIMA) is considered. Similarly, to the definition of the hybrid model presented above, but instead of applying ARIMA as a pre-processing method before to apply GARCH model. Therefore, the most appropriate strategy for predicting volatility using hybrid models is also analysed. In other words, the models associated with the terms \tilde{L}_t and \tilde{N}_t are swapped with the GARCH and ARIMA models, respectively.

3.4.2 Wavelet model

Another technique studied in this work, which also performs a mixed combination of predictive models, is considered with the purpose of performing a comparative study between these partitioned techniques. The technique propose below is related to the approach presented in Sect. 3.2. In this approach, each signal $\tilde{A}_{it}, \tilde{D}_{it}$ resulting from the discrete wavelet decomposition (DWT) is predicted by applying ARIMA or GARCH model to each component. These predictions are denoted by $\tilde{h}_t^{\text{w-ARIMA}}$ or $\tilde{h}_t^{\text{w-GARCH}}$, depending on the model used.

This work proposes to use ARIMA and GARCH models in a mixed strategy to predict low and high frequency signals form (DWT). Considering each possible combination in the selection of the models employed for each component, with the objective of identifying which of these methodologies should be considered, for volatility forecasting.

Formally, it is proposed to use models of the type:

$$\tilde{h}_t^{\text{w-ARIMA-GARCH}} = \tilde{A}_{2t}^{\text{ARIMA}} + \tilde{D}_{2t}^{\text{GARCH}} + \tilde{D}_{1t}^{\text{GARCH}},$$

where for each signal obtained from the wavelet decomposition, either high or low frequency, the ARIMA and GARCH models are applied in a mixed form. In our literature review, it was found that the usual way to apply these wavelet models is by using an individual model, ARIMA or GARCH for low-frequency and high-frequency components respectively, however, in this work a comparative study of all possible options is performed. Selection of individual models for each wave is described in Sects. 3.1 and 3.3.

4 Stochastic volatility model

To make accuracy and performance comparisons between the proposed model and one of the currently used for volatility forecasting, the stochastic volatility model (SV) is introduced. Asset prices exhibit fluctuations in volatility over time, characterized by periods of both high and low return variability. Stochastic volatility models capture this phenomenon by incorporating a latent volatility variable, which is modelled as a stochastic process (Hoffman & Gelman, n.d.; Kim et al., 1998).

$$\begin{aligned}\sigma &\sim \epsilon_t \exp\left(\frac{h_t}{2}\right), \\ h_{t+1} &= \mu + \phi(h_t - \mu) + \delta_t \sigma, \\ h_1 &\sim N\left(\mu, \frac{\sigma}{\sqrt{1-\phi^2}}\right), \\ \epsilon_t, \delta_t &\sim N(0, 1),\end{aligned}$$

where, ϵ_t represent the asset return white noise, at time t , δ_t is the shock on volatility and h_t is the latent parameter for log volatility. The primary idea of this model is identifying hidden factors that impact alterations in asset values. These undisclosed elements, encompassing shifts in market sentiment, news, or other relevant variables, play a role in determining the extent of market volatility. By representing volatility as a dynamic and stochastic process, the SV model provides a more accurate representation of financial markets, distinguishing it from simpler models that assume a constant level of volatility.

5 Methodology

Prediction of volatility time series has been performed in different works using machine learning and deep learning techniques as well as classical time series models. On the other hand, results obtained using a single forecasting strategy can be improved by using mixed models, combining techniques that have different predictive properties. Hybrid models have been used previously for time series forecasting, usually by decomposing the original data into two components, associated with the linear and nonlinear part of the series. In this manner, the use of two predictive methods is combined, where usually the non-linear part is adjusted using models able to capture strong fluctuations as ANNs for example (Zhang & Zhang, 2018). Another hybrid technique found in the literature consider outputs from a GARCH model as input of an ANN and conversely for forecasting volatility (Lu et al., 2016). Stacking models have also been used to forecast volatility, where predictions are dynamically selected from a set of trained machine learning models based on feature extraction and selection (Aras, 2021).

Some authors proposed to use wavelet transform as a pre-processing to decompose a time series into multiple high and low frequency waves, and depending on the nature of the data different strategies are proposed to forecast each component. For example, D. Liu et al. proposed to use SVM to forecast wind speed based just

on the approximate signal. On the other hand, Glosten–Jagannathan–Runkle (GJR)-GARCH model has been used together with wavelet transform, to predict financial returns, in this case each wavelet component is adjusted with the same model (Berger, 2016). To forecast electricity prices, a wavelet transform based on ARIMA have also been used, in this case the prediction is obtained applying ARIMA-GARCH to the approximate signal and GARCH to the detailed signals (Tan et al., 2010).

In this work, a comparative study between hybridization techniques based on residuals decomposition and partitioned strategies according to wavelet transform for volatility forecasting is presented. The strategy what indeed provide better predictions in terms of the accuracy metrics, MAPE, RMSE, and R^2 is proposed. Initially it is proposed to use a technique based on wavelet transform and the ARIMA and GARCH models is address, where ARIMA, being linear, is used to forecast the low frequency wave, while GARCH is used for high frequency waves. For realized volatility forecasting, the use of this strategy was not found in the literature found to date, nor a detail description of the implementation process of this type of models in the selected programming language Python. This choice of models has been found to be the most natural and efficient for volatility forecasting, and consequently, it is the one proposed in this work.

Most of the applications of the wavelet transform found in Python are mostly targeted at digital image processing. Therefore, this paper describes how this implementation was done to serve also as a starting point for those interested in using these types of partitioned models. In addition, given that in the related literature, limited detail is provided regarding the functioning of the associated libraries for the wavelet decomposition and recovery of the original signals. Figure 2 describes the process of implementing the hybridization technique suggested.

One of the main challenges presented during the implementation of the wavelet transform for time series forecasting is that there is limited development of the wavelet transform when using the Python language. Most of the applications of the wavelet transform found in Python are mostly targeted at digital image processing. Therefore, this paper describes how this implementation was done to serve also as a starting point for those interested in using these types of partitioned models. In addition, in the related literature, limited detail is provided regarding the functioning of the associated libraries for the wavelet decomposition and recovery of the original signals. Figure 2 describes the process of implementing the hybridization technique proposed in this work. As can be seen in Fig. 2, to obtain the final prediction, first, the respective decomposition of the daily returns is obtained by using the discrete wavelet transform. Then, by using the discrete wavelet transform inverse, the original signal is reconstructed through approximate and detailed components. Finally, each signal is predicted by using, for example, ARIMA for approximate components and GARCH for detailed components to get the prediction $\tilde{h}_t^{\text{w-ARIMA-GARCH}}$.

Table 1 Descriptive parameters for TSLA closed prices

Closed price	
N_{records}	3178
μ	59.95
σ	95.69
y_{min}	1.05
Q_1	9.30
\tilde{x}	16.36
Q_3	27.01
y_{max}	409.97
<i>Kurtosis</i>	1.82
<i>Skewness</i>	1.79

6 Data

To carry out the empirical part of this investigation, the daily closing prices of TSLA stocks were considered. Data was loaded directly from Yahoo Finance using the free Python API, for a total of 3178 observations (between 2010-07-07 to 2023-02-17). Descriptive parameters are calculated to obtain relevant information related to measures of central tendency, dispersion, kurtosis and skewness of closed prices (see Table 1). The values presented point to an asymmetrical (positive) data distribution, with values of Q_1 , Q_2 and Q_3 relatively close (in the first ten years considered in the study, historical data show low values). Around 75% of the observations show a closed price below USD 27, far from the maximum value observed (USD 409.97) at the end of 2021.

Figure 3 shows for the TSLA closed price and trading volume strong fluctuations associated with high volatility. Strong uptrend and a large trading volume ending

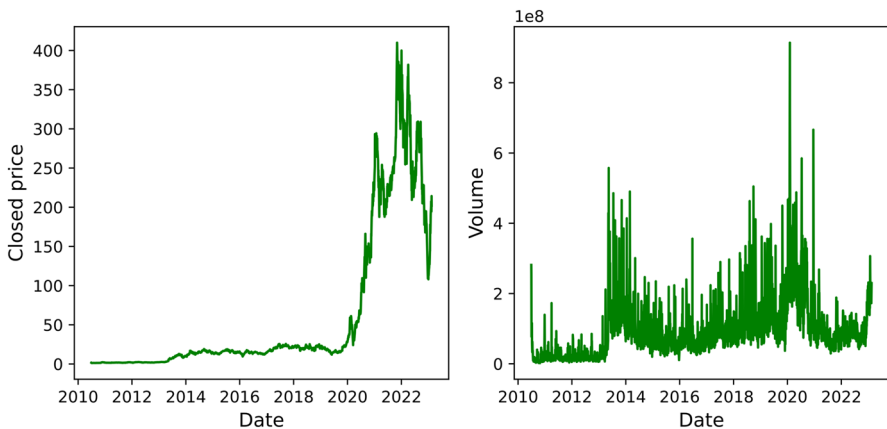


Fig. 3 Time series for TSLA closed price (left) and trading volume (right)

Table 2 Normality tests (Jarque–Bera test and skewness and kurtosis tests), stationarity/existence of unit root (ADF test and KPSS test) and independence test (BDS test) for TSLA closed prices

Test	Normality			Unit root/stationary		Independence
	Kurtosis	skewness	Jarque–Bera	ADF	KPSS	BDS (Dim.2–Dim.6)
Statistic	11.62	28.80	2139.65	−0.98	6.15	11.67–23.88
<i>p</i> -value	0.00*	0.00*	0.00*	0.75	–	0.00*

*Rejected for the significance levels 1%, 5%, 10%

around 2021 can be identified. In terms of trends, we can identify three clearly distinct moments. Until 2020, historical data shows a relatively constant trend (with low closed price). Globally, there is a marked growth between 2020 and the end of 2021, although with some fluctuations, namely in early 2021 (due to the instability of the financial markets, partly caused by the COVID-19 pandemic). From the end of 2021, the data is marked by an abrupt drop in the closed price.

To analyse some features of the closed price time series, Table 2 contains the statistic test and the following hypothesis tests: Normality tests (Jarque–Bera test and Skewness and Kurtosis tests), Stationarity/Existence of unit root (ADF test and KPSS test) and independence test (BDS test). As expected, for any significance level, the normality, stationarity, and independence tests are rejected. In fact, there is statistical evidence to: (i) do not reject the nonnormality of the distribution of the data (with rejection of the null hypothesis, which indicates normal behaviour, in all tests performed); (ii) assume the non-stationarity (due to the null hypothesis not being rejected when doing ADF test, and due to the statistical value corresponding to KPSS being superior to the critical reference values); (iii) infer about the non-iid, since the null hypothesis of the data being iid has been rejected through BDS test.

According to (Karasan, n.d., 2005), to model volatility we need to calculate the return volatility, which is also known as realized volatility. Realized volatility is the square root of realized variance, which is the sum of squared return. Realized volatility is used to calculate the performance of the volatility prediction method, it is denoted by h_t in the model description of GARCH (Eq. 9).

7 Numerical results

To continue the study, based on historical data on TSLA closing prices, described in previous section, the time series of the daily returns and realized volatility were obtained. Figure 4 shows daily returns plot, whose values are around the zero mean. As can be seen, the variance is a function of time, which confirms the effect of heteroscedasticity in the time series, and the associated histogram shows that the daily returns have a non-normal distribution. Regarding realized volatility time series, note the peaks of values observed during the period 2020/2021 (instability of the financial markets, partly motivated by the COVID-19 pandemic).

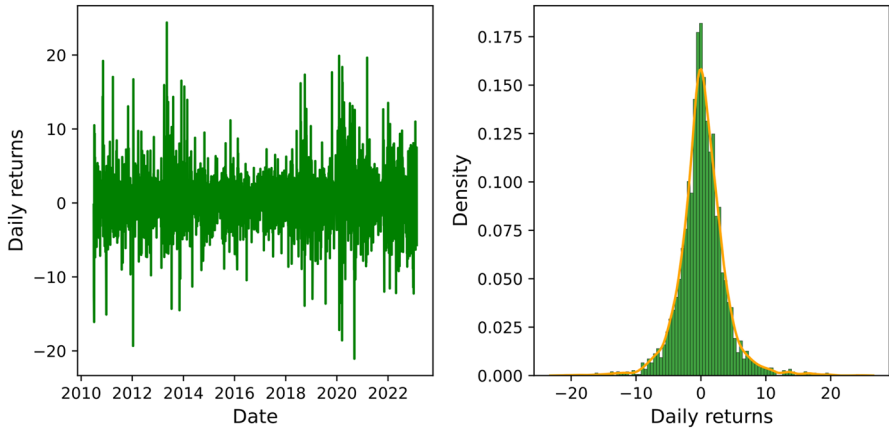


Fig. 4 Daily return series (left) and KDE histogram (right) plots for TSLA stock

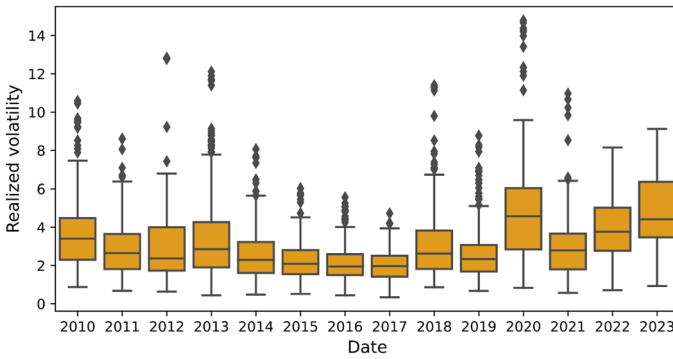


Fig. 5 Realized volatility time series for TSLA stock. Graphical representation of annual box plots

Table 3 Descriptive parameters for TSLA daily returns and realized volatility

	Daily returns	Realized volatility
$N_{records}$	3181	3177
μ	0.21	3.06
σ	3.61	1.90
y_{min}	-21.06	0.33
Q_1	-1.55	1.76
\tilde{x}	0.12	2.57
Q_3	1.93	3.83
y_{max}	24.39	14.78
<i>Kurtosis</i>	5.01	5.37
<i>Skewness</i>	0.32	1.87

Table 4 Normality tests (Jarque–Bera test and skewness and kurtosis tests), stationarity/existence of unit root (ADF test and KPSS test) and independence test (BDS test) for TSLA daily returns and realized volatility

Test	Normality			Unit root/stationary		Independence BDS (Dim.2–Dim.6)
	Kurtosis	Skewness	Jarque–Bera	ADF	KPSS	
D. return statistic	18.86	8.28	3385.62	−56.93	0.08	0.44–1.36
<i>p</i> -value	0.00*	0.00*	0.00*	0.00*	–	0.65–0.17
R. volatility Statistic	19.38	29.58	5681.73	−6.10	0.10	4.84–7.47
<i>p</i> -value	0.00*	0.00*	0.00*	0.00*	–	0.00*

*Means rejected for the significance levels 1%, 5%, 10%

This detail can be better perceived when evaluating the annual box plots (Fig. 5), where a considerable sample amplitude is observed in 2020 (with several outliers observations above the maximum barrier of the box-plot) and a notable interquartile amplitude, in comparison with the other years.

Complementing the graphical representation, some descriptive statistical measures are presented in Table 3. Based on these values, a positive kurtosis (leptokurtic distribution) and a slight asymmetry to the right are observed.

As was done in previous section, to analyse some features of the time series, Table 4 contains the statistic test and the following hypothesis tests: Normality tests (Jarque–Bera test and Skewness and Kurtosis tests), Stationarity/Existence of unit root (ADF test and KPSS test) and independence test (BDS test). As expected, for any significance level, the normality is rejected for the two series under study, concluding that it does not have a normal distribution.

In addition, for both time series, ADF test and KPSS tests confirm that the series is stationary (due to the null hypothesis being rejected when doing ADF test, and due to the statistical value corresponding to KPSS being less to the critical reference values). When testing for Independency, for daily returns it is concluded that the null hypothesis is not rejected for any significance level and therefore the data are i.i.d. For realized volatility time series, there is statistical evidence to infer about

Table 5 Normality tests on daily returns and realized volatility for TLSEA stocks

Test	Shapiro–Wilk	D’Agostino’s K^2	Anderson–Darling	Significance	Critical value
				level	
<i>p</i> -value	0.00000	0.00000	42.514	15	0.575
				10	0.655
				5	0.786
				2.5	0.917
				1	1.091

p-values for Shapiro–Wilk, D’Agostino’s K^2 and Anderson–Darling tests are registered

the non-iid, since the null hypothesis of the data being iid has been rejected through BDS test. Different normality test such as: Shapiro–Wilk, D’Agostino’s K^2 and Anderson–Darling tests, were implemented to verify normality and results are presented in Table 5. As can be seen, based on the null hypothesis:

H_0 : dataset has normal distribution

Shapiro–Wilk, and D’Agostino’s K^2 test yields p -values far below the significance threshold of 0.05. Therefore, the null hypothesis is rejected, i.e. the distribution of daily returns is not normal. On the other hand, it can be seen that daily returns distributions are peaked around the mean, which is confirmed with positive kurtosis values larger than 2, it means most of the daily return values are concentrated in the mean. Furthermore, this implies a high level of risk but the possibility of higher returns due to large price movements. Since the statistic (42.514), on Anderson–Darling test, is greater than all critical values for different significance levels, non-normality is confirmed.

7.1 Application of ARIMA model

Daily returns and realized volatility for TLSEA show stationary signals, with p -values less than 0.05 on Augmented Dickey–Fuller test. Therefore, the null hypothesis

H_0 : dataset has nonstationary distribution

is rejected, thus each share shows a stationary signal; consequently, an integration order greater than zero is not necessary. The ARIMA model was trained based on the Box and Jenkins strategy to get the best model in terms of goodness of fit. Optimal orders p, d, q were obtained based on the Akaike information criterion (AIC), which provides best orders for model quality, based on the minimization of the AIC.

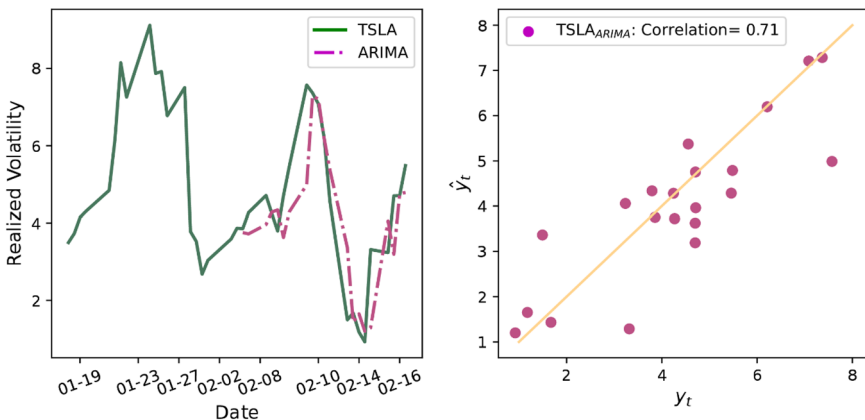
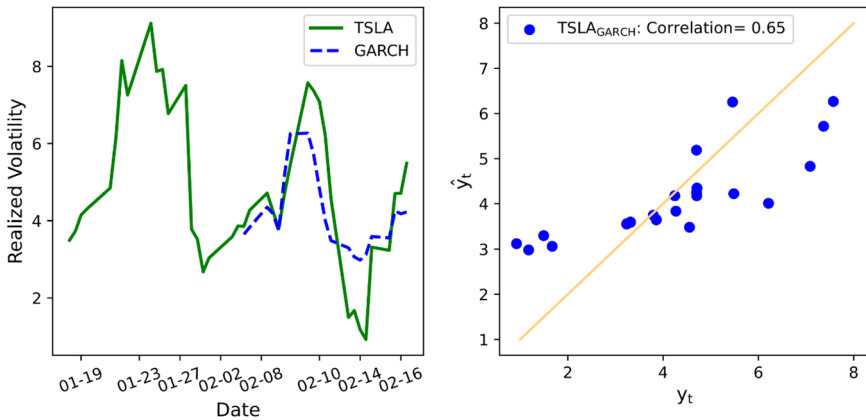


Fig. 6 Real vs. ARIMA adjustment for TSLA realized volatility (left). Correlation plots for the test set and prediction with its corresponding R^2 value (right)

Table 6 Accuracy metrics MAPE, MAE, RMSE, and R^2 for TSLA realized volatility forecast using ARIMA and GARCH

Model	MAPE	MAE	RMSE	R^2
ARIMA	0.25	0.75	1.03	0.71
GARCH	0.25	0.99	1.24	0.65

**Fig. 7** Real vs. GARCH adjustment for TSLA realized volatility (left). Real time series is represented in green and its prediction in blue colour. Correlation plots for the test set and prediction with its corresponding R^2 value (right)

The minimum AIC was obtained by doing a loop over a fixed range for p, q where the order d was set to zero since no differentiation was necessary given that Dickey-Fuller test confirmed stationarity.

ARIMA was adjusted using the TSLA stock volatility training set composed of all the available historical data, except for the last 21 days considered for the test set or prediction horizon. One-step-ahead forecasting was considered in this work. The Best ARIMA model obtained was ARIMA(4, 0, 4) for realized volatility prediction. Figure 6 shows realized volatility prediction and correlation plots between the original and predicted time series. Goodness of fit and simulation performance can be confirmed with the R^2 and accuracy metrics, such as MAPE, MAE, and RMSE (Table 6). As can be seen, ARIMA shows good performance forecasting implicit linear patterns in TSLA realized volatility, maintaining trends for a long-term horizon; nevertheless, it is essential to use another method to detect non-linear patterns.

7.2 Application of GARCH models

Similar to ARIMA training and considering the same prediction horizon, the best orders p, q for the GARCH model were obtained by minimization of the AIC

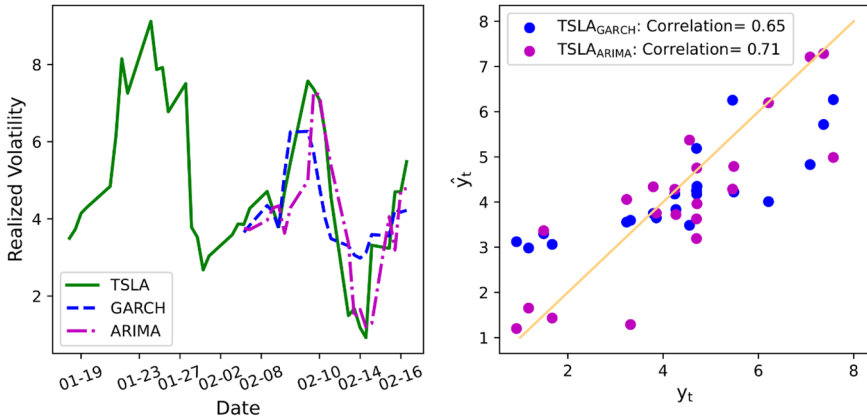


Fig. 8 Real vs. ARIMA, and GARCH adjustment for Tesla realized volatility (left). Correlation plots with its corresponding R^2 value (right)

coefficient. Parameters obtained for the ARCH model with a prediction horizon of 21 days were $p = 4, q = 2$, considering a zero-volatility process and the exponential version of the GARCH model as optimal parameters after a grid search was applied. Coefficients for the model were obtained based on the maximum likelihood estimation (MLE). Figure 7 shows model adjustment for the real time series in the test set and correlation plots between the original and predicted realized volatility. It is remarkable the difference between goodness of fit between GARCH and ARIMA models, although in terms of error metrics they are only slightly different (Table 6). ARIMA remains as the model with the best goodness-of-fit. Figure 8 shows ARIMA and GARCH model predictions and their correlations with the original time series.

7.3 Application of hybrid and partitioned models

A first approximation to combine ARIMA and GARCH models is to predict realized volatility first using for example ARIMA and then predict its residuals with ARCH to have the first hybrid model implemented in this work denoted by ARIMA-GARCH (see Sect. 3.4). The second hybrid model can be obtained, changing roles for the first and second model applied to the original time series and its forecasted residuals, this model will be denoted by GARCH-ARIMA. This second option has

Table 7 Accuracy metrics MAPE, MAE, RMSE, and R^2 for realized volatility forecast using individual and hybrid models

Model	MAPE	MAE	RMSE	R^2
ARIMA	0.25	0.75	1.03	0.71
GARCH	0.25	0.99	1.24	0.65
ARIMA-GARCH	0.24	1.07	1.26	0.70
GARCH-ARIMA	0.25	0.99	1.24	0.65

the disadvantage that residuals are extremely small, and therefore the contribution of ARIMA is not significant, resulting in the same accuracy results provided by GARCH (see Table 7).

The second option for combining the econometric models proposed in this work is to consider the wavelet transform as a pre-processing method for the time series of interest. Daily return time series is decomposed in an approximate signal named \mathcal{A}_t and a detailed signal denoted by \mathcal{D}_t using the wavelet transform. Usually, detailed signal has higher frequency and therefore in this work it is proposed to use GARCH model to predict this component and ARIMA as a linear model for the low frequency signal. This first approximation will be denoted by W-ARIMA-GARCH. It means, over the signals \mathcal{A}_t and \mathcal{D}_t , ARIMA and GARCH models are applied respectively. By W-GARCH-ARIMA shall be denoted the opposite, for selecting predictive models for low and high frequency decomposed signals (see Fig. 2). In addition, it was investigated the case where for each of the wavelets obtained by wavelet transform the same model, either ARIMA or GARCH, was applied to study the behaviour of these predictions. Prediction results are presented in Fig. 9, where the original TSLA time series is plotted over the prediction and variance explained by each model is at the right side.

For W-ARIMA model, the following parameters were obtained from the training datasets for TSLA daily returns after a manual grid search in pursuit of the best decomposition and models: Daubechey wave db2 cast the orders (4, 1, 4) and (4, 0, 4) for high and low frequency waves. Figure 10 shows time series decomposition using the db2-Daubechey filter. Next, the W-GARCH model was implemented, and the parameters obtained were: Daubechey wave db3, (4, 4) and (4, 4) for low and high frequency signals considering zero mean model and exponential GARCH. Figure 11 shows the model adjustment for the real time series, with its respective correlation plots. As can be seen, W-GARCH prediction are strongly penalizing the high values of the time series. A possible reason can be

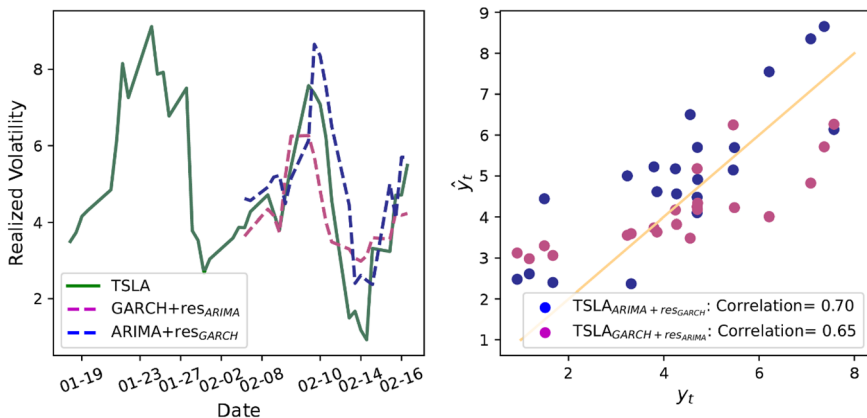


Fig. 9 Real vs. ARIMA-GARCH and GARCH-ARIMA predictions for TSLA realized volatility (left). Correlation plot with its corresponding R^2 value (right)

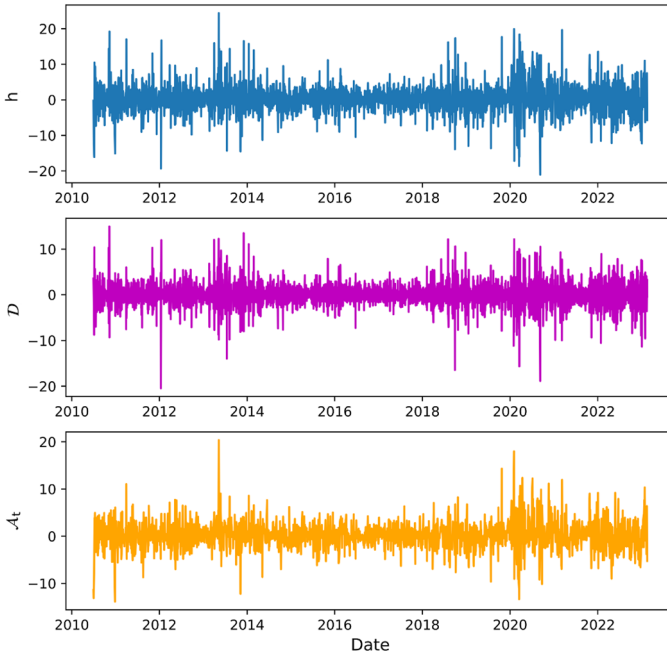


Fig. 10 Decomposition with db2-Daubechji filter and ARIMA-WAVELET adjustment for each decomposition

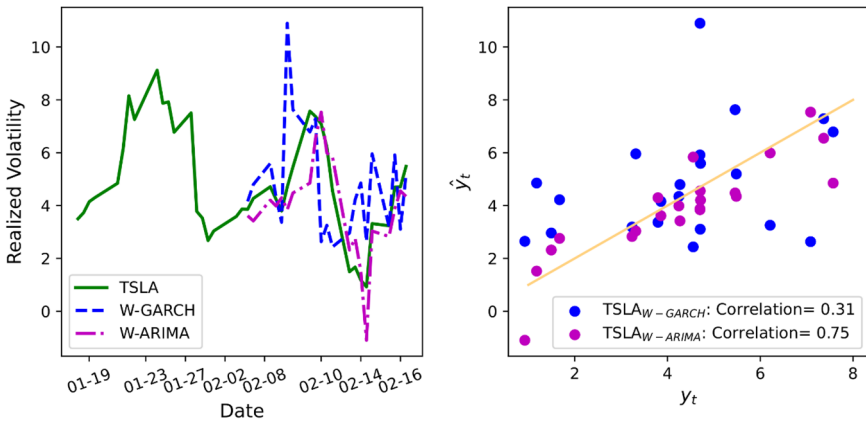


Fig. 11 Real vs. W-GARCH and W-ARIMA models predictions for TSLA realized volatility. Correlation plots for the test set and prediction with its corresponding R^2 value (right)

to use GARCH process to adjust low frequency signal from the decomposition. Comparing with W-ARIMA an improvement can be seen and confirmed with the R^2 as goodness of fit measure and accuracy metrics (see Table 8). Therefore,

Table 8 Accuracy metrics MAPE, MAE, RMSE, and R^2 for realized volatility forecast using the W-ARIMA, W-GARCH models

Model	MAPE	MAE	RMSE	R^2
W-ARIMA	0.26	0.79	1.00	0.68
W-GARCH	0.35	1.80	2.38	0.36

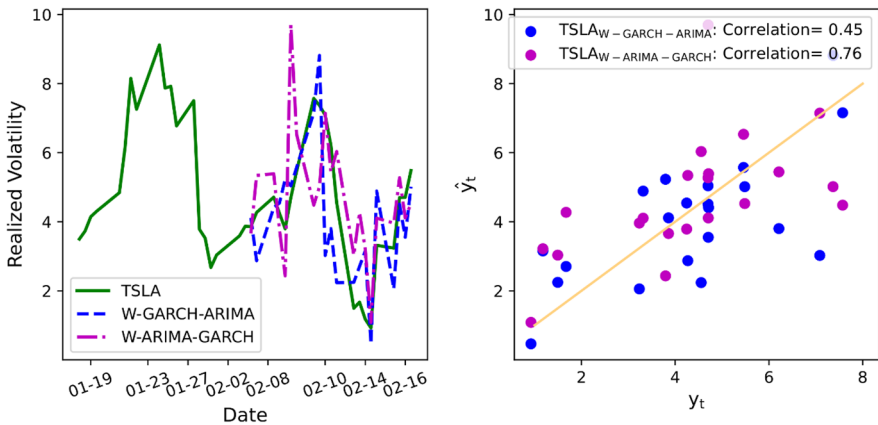


Fig. 12 Real vs. W-GARCH-ARIMA and W-ARIMA-GARCH models predictions for TSLA realized volatility. Correlation plots with its corresponding R^2 value (right)

Table 9 Accuracy metrics MAPE, MAE, RMSE, and R^2 for realized volatility forecast using all the models studied

Model	MAPE	MAE	RMSE	R^2
ARIMA	0.25	0.75	1.03	0.71
GARCH	0.25	0.99	1.24	0.65
SV	0.44	1.12	1.73	0.24
W-ARIMA	0.26	0.79	1.00	0.68
W-GARCH	0.35	1.80	2.38	0.36
ARIMA-GARCH	0.24	1.07	1.26	0.70
GARCH-ARIMA	0.25	0.99	1.24	0.65
W-ARIMA-GARCH	0.24	1.27	1.59	0.76
W-GARCH-ARIMA	0.37	1.07	1.34	0.62
W-ARIMA-SV	0.37	2.53	3.02	0.13

applying ARIMA to each component in terms of goodness of fit is better than using GARCH on each one.

For W-ARIMA-GARCH model the following were parameters obtained for TSLA daily returns: Daubechey wave db3, orders (4, 2), with optional exogenous

regressors and exponential GARCH) version for high frequency signal and (4, 0, 4) as ARIMA orders for low frequency. W-GARCH-ARIMA model yielded the following parameters for the best model: Daubechey wave db2, (4, 1, 4) as ARIMA orders for approximate signal and (4, 2), zero mean, and exponential GARCH for the low frequency component.

Figure 12 shows models adjustment for the real time series of realized volatility and the respective correlation plots. Goodness of fit can be confirmed with R^2 and accuracy metrics MAPE, MAE, and RMSE are resumed in Table 9 for all the models studied in this work. Thus, applying ARCH and ARIMA process in high and low frequency signal provides better predictions in terms of goodness-of-fit. As before, applying a GARCH process on the low frequency decomposition fails to give optimal prediction results.

Finally, the stochastic volatility model was implemented for the same prediction horizon, using rolling-forecast. However, the model proposed in this work showed better error metrics and goodness of fit. For the implementation of the SV model, defined in Sect. 4, $\phi \sim \text{uniform}(-1, 1)$, $\sigma \sim \text{cauchy}(0, 5)$ and $\mu \sim \text{cauchy}(0, 10)$ were considered (Fig. 13).

8 Conclusions

The present work provides a starting point for any research related to the use of hybrid and wavelet transform models for volatility forecasting. It also proposed ARIMA and GARCH models which do not require significant computational training costs as alternative strategies to predict each signal. This same strategy can be used considering even any other pair of models, e.g. machine learning or deep learning. A detailed description has been made with outlines, which show how the implementation of these models in Python language should be done due to the fact that there is very little in the literature on the subject.

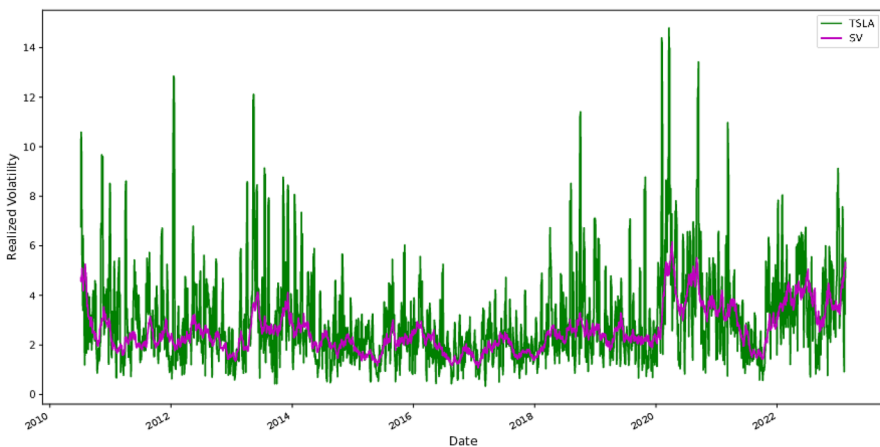


Fig. 13 Real vs. SV model with $\phi \sim \text{uniform}(-1, 1)$, $\sigma \sim \text{cauchy}(0, 5)$ and $\mu \sim \text{cauchy}(0, 10)$

ARIMA models, when used in combination with the wavelet transform to predict realized volatility, will provide better predictions than those obtained with the GARCH model in terms of goodness of fit and each accuracy metric. By using multiple hybridization techniques between the proposed models, prediction scores can be improved. Selection of models to be used in each component of the wavelet transform decomposition is crucial to obtain more accurate predictions. In conclusion, the best alternative is to forecast low frequency signals with linear models such as ARIMA and high frequency signals with models that can capture strong nonlinear patterns, such as the GARCH model.

Models based on wavelet transform are compared with hybrid models, based on forecasting the original time series using for example a linear model and then its residual using a model able to detect strong fluctuations, such as GARCH. The two possible combinations were studied, to reach the conclusion that in order to obtain improved predictions, the model that should always be applied first is the linear model, in this case ARIMA. The hybrid model was compared with each model individually and the partitioned wavelet transform approach. In terms of goodness of fit, the best model for forecasting volatility is the wavelet transform where the approximate signal is predicted with a linear model, in this case ARIMA and the detailed high frequency using GARCH.

The present W-ARIMA-GARCH model has been applied to Tesla stock volatility, to quantify risk, however, it can be applied to any other time series with strong fluctuations, such as cryptocurrencies, for example. One of the advantages of this model is that it takes advantage of the ARIMA and GARCH models to capture different types of patterns in the time series. In addition, another advantage of implementing this type of forecasts is that they can be optimized, allowing to obtain predictions in short CPU time. For example, for algorithmic trading firms, where fast predictions must be executed, to reduce risk and detect market and arbitrage opportunities, it is of high importance.

As future work, it is proposed to make use of this technique in different applications where it is not yet used, to prove its effectiveness. Investigate which alternative models can be used in this type of decomposition. Also, study techniques based on genetic algorithms and stacked neural networks to compare their performance in comparison with the models studied in this work. The use of sentiment analysis can be used to refine a financial decision, therefore as future work is also proposed to study natural language processing and sentiment analysis to obtain even more accurate predictions.

Acknowledgements This work is partially financed by national funds through FCT—Fundação para a Ciência e a Tecnologia under the project UIDB/00006/2020. L.R. is supported by Universidad del Norte. Research Agenda 2021-009. The paper has also benefited from discussions at the 42nd EBES Conference, in Lisbon.

Funding Open access funding provided by FCT/IFCCN (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.

Data availability Datasets used in this work were acquired through the use of the free Yahoo Finance Python API.

Declarations

Conflict of interest The authors declare no conflict of interest.

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

- Alshammari, T. S., Ismail, M. T., Al-wadi, S., Saleh, M. H., & Jaber, J. J. (2020). Modeling and forecasting Saudi stock market volatility using wavelet methods. *The Journal of Asian Finance, Economics and Business*, 7(11), 83–93. <https://doi.org/10.13106/jafeb.2020.vol7.no11.083>
- Al Wadi, S., Ismail, M. T., Altaher, A. M., & Addul Karim, S. A. (2010). Forecasting volatility data based on Wavelet transforms and ARIMA model. In *2010 International Conference on Science and Social Research (CSSR 2010)* (pp. 86–90). <https://doi.org/10.1109/CSSR.2010.5773909>
- Aras, S. (2021). Stacking hybrid GARCH models for forecasting Bitcoin volatility. *Expert Systems with Applications*, 174, 114747. <https://doi.org/10.1016/J.ESWA.2021.114747>
- Asteriou, D., & Hall, S. G. (2016). ARIMA models and the Box–Jenkins methodology. In *Applied Econometrics* (pp. 275–296). Macmillan Education UK. https://doi.org/10.1057/978-1-137-41547-9_13
- Bathla, G. (2020). Stock price prediction using LSTM and SVR. In *2020 Sixth International Conference on Parallel, Distributed and Grid Computing (PDGC)* (pp. 211–214). <https://doi.org/10.1109/PDGC50313.2020.9315800>
- Berger, T. (2016). Forecasting based on decomposed financial return series: A wavelet analysis. *Journal of Forecasting*, 35(5), 419–433. <https://doi.org/10.1002/for.2384>
- Box, G. E. P., Jenkins, G. M., & Reinsel, G. C. (2008). *Time Series Analysis: Forecasting and Control*. Wiley.
- Chniti, G., Bakir, H., & Zaher, H. (2017). E-commerce time series forecasting using LSTM neural network and support vector regression. In *Proceedings of the international conference on big data and internet of thing* (pp. 80–84). <https://doi.org/10.1145/3175684.3175695>
- Christensen, K., Siggaard, M., & Veliyev, B. (2021). A machine learning approach to volatility forecasting. *Journal of Financial Econometrics*. <https://doi.org/10.1093/jffinec/nbac020>
- Daubechies, I. (1992). Ten lectures on wavelets. *Society for Industrial and Applied Mathematics*. <https://doi.org/10.1137/1.9781611970104>
- de Oliveira, E. M., & Cyrino Oliveira, F. L. (2018). Forecasting mid-long term electric energy consumption through bagging ARIMA and exponential smoothing methods. *Energy*, 144, 776–788. <https://doi.org/10.1016/J.ENERGY.2017.12.049>
- Engle, R. F. (1982). Autoregressive conditional heteroscedasticity with estimates of the variance of United Kingdom inflation. *Econometrica*, 50(4), 987. <https://doi.org/10.2307/1912773>
- Gaoyu, Z., Qiongfei, L., Qing, L., & Zhizhao, Z. (2009). High frequency financial time series forecasting via particle filtering. In *2009 International Conference on Information Management, Innovation Management and Industrial Engineering* (pp. 62–65). <https://doi.org/10.1109/ICIMI.2009.477>
- 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. <https://doi.org/10.1007/s10287-005-0005-5>
- Guo, J., Xie, Z., Qin, Y., Jia, L., & Wang, Y. (2019). Short-term abnormal passenger flow prediction based on the fusion of SVR and LSTM. *IEEE Access*, 7, 42946–42955. <https://doi.org/10.1109/ACCESS.2019.2907739>

- Hafner, C. M. (2009). Causality and forecasting in temporally aggregated multivariate GARCH processes. *Econometrics Journal*, 12(1), 127–146. <https://doi.org/10.1111/j.1368-423X.2008.00276.x>
- Hoffman, M. D., & Gelman, A. (n.d.). *The no-U-turn sampler: adaptively setting path lengths in Hamiltonian Monte Carlo*. <http://mcmc-jags.sourceforge.net>
- Hyup Roh, T. (2007). Forecasting the volatility of stock price index. *Expert Systems with Applications*, 33(4), 916–922. <https://doi.org/10.1016/J.ESWA.2006.08.001>
- Kang, S. H., Kang, S. M., & Yoon, S. M. (2009). Forecasting volatility of crude oil markets. *Energy Economics*, 31(1), 119–125. <https://doi.org/10.1016/J.ENERCO.2008.09.006>
- Karasan, A. (n.d.). Machine learning for financial risk management with Python: algorithms for modeling risk.
- Khashei, M., & Bijari, M. (2011). A novel hybridization of artificial neural networks and ARIMA models for time series forecasting. *Applied Soft Computing*, 11(2), 2664–2675. <https://doi.org/10.1016/J.ASOC.2010.10.015>
- Khashei, M., & Bijari, M. (2012). A new class of hybrid models for time series forecasting. *Expert Systems with Applications*, 39(4), 4344–4357. <https://doi.org/10.1016/J.ESWA.2011.09.157>
- Kim, S., Shepherd, N., & Chib, S. (1998). Stochastic volatility: Likelihood inference and comparison with ARCH models. *Review of Economic Studies*, 65(3), 361–393. <https://doi.org/10.1111/1467-937X.00050>
- Li, X., & Tang, P. (2020). Stock index prediction based on wavelet transform and FCD-MLGRU. *Journal of Forecasting*, 39(8), 1229–1237. <https://doi.org/10.1002/for.2682>
- Liu, Y. (2019). Novel volatility forecasting using deep learning—Long short term memory recurrent neural networks. *Expert Systems with Applications*, 132, 99–109. <https://doi.org/10.1016/J.ESWA.2019.04.038>
- Lu, X., Que, D., & Cao, G. (2016). Volatility forecast based on the hybrid artificial neural network and GARCH-type models. *Procedia Computer Science*, 91, 1044–1049. <https://doi.org/10.1016/J.PROCS.2016.07.145>
- Luzia, R., Rubio, L., & Velasquez, C. E. (2023). Sensitivity analysis for forecasting Brazilian electricity demand using artificial neural networks and hybrid models based on autoregressive integrated moving average. *Energy*, 274, 127365. <https://doi.org/10.1016/J.ENERGY.2023.127365>
- Mallat, S. G. (n.d.). Fundamental papers in wavelet theory.
- Mallat, S. G., & Peyré, G. (n.d.). A wavelet tour of signal processing: the sparse way.
- Maria, F. C., & Eva, D. (n.d.). Exchange-rates forecasting: exponential smoothing techniques and ARIMA models.
- Montgomery, D. C., Jennings, C. L., & Kulahci, M. (n.d.). Wiley series in probability and statistics.
- Qi, C., Ren, J., & Su, J. (2023). GRU neural network based on CEEMDAN-wavelet for stock price prediction. *Applied Sciences*, 13(12), 7104. <https://doi.org/10.3390/app13127104>
- Ramos-Pérez, E., Alonso-González, P. J., & Núñez-Velázquez, J. J. (2019). Forecasting volatility with a stacked model based on a hybridized artificial neural network. *Expert Systems with Applications*, 129, 1–9. <https://doi.org/10.1016/J.ESWA.2019.03.046>
- Rubio, L., & Alba, K. (2022). Forecasting selected colombian shares using a hybrid ARIMA-SVR model. *Mathematics*, 10(13), 2181. <https://doi.org/10.3390/math10132181>
- Rubio, L., Gutiérrez-Rodríguez, A. J., & Forero, M. G. (2021). EBITDA index prediction using exponential smoothing and ARIMA model. *Mathematics*, 9(20), 2538. <https://doi.org/10.3390/math9202538>
- Tan, Z., Zhang, J., Wang, J., & Xu, J. (2010). Day-ahead electricity price forecasting using wavelet transform combined with ARIMA and GARCH models. *Applied Energy*, 87(11), 3606–3610. <https://doi.org/10.1016/J.APENERGY.2010.05.012>
- 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. <https://doi.org/10.1016/j.eswa.2008.01.047>
- Wang, L., Zou, H., Su, J., Li, L., & Chaudhry, S. (2013). An ARIMA-ANN hybrid model for time series forecasting. *Systems Research and Behavioral Science*, 30(3), 244–259. <https://doi.org/10.1002/sres.2179>
- Wang, Y., & Guo, Y. (2020). Forecasting method of stock market volatility in time series data based on mixed model of ARIMA and XGBoost. *China Communications*, 17(3), 205–221. <https://doi.org/10.23919/JCC.2020.03.017>
- Xiao, Y., Xiao, J., Liu, J., & Wang, S. (2014). A multiscale modeling approach incorporating ARIMA and ANNs for financial market volatility forecasting. *Journal of Systems Science and Complexity*, 27(1), 225–236. <https://doi.org/10.1007/s11424-014-3305-4>

- Yu, H., Ming, L. J., Sumei, R., & Shuping, Z. (2020). A hybrid model for financial time series forecasting—Integration of EWT, ARIMA with the improved ABC optimized ELM. *IEEE Access*, 8, 84501–84518. <https://doi.org/10.1109/ACCESS.2020.2987547>
- Zhang, P. G. (2003). Time series forecasting using a hybrid ARIMA and neural network model. *Neuro-computing*, 50, 159–175. [https://doi.org/10.1016/S0925-2312\(01\)00702-0](https://doi.org/10.1016/S0925-2312(01)00702-0)
- Zhang, Y., & Zhang, J. (2018). Volatility forecasting of crude oil market: A new hybrid method. *Journal of Forecasting*, 37(8), 781–789. <https://doi.org/10.1002/for.2502>

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