VS-LTGARCHX: A Flexible Subset Selection Approach for Estimation of log-TGARCHX Models and Its Application to BTC Markets

Victor Elvira, Samir Orujov, Audrey Poterie, Farid Rajabov, Francois Septier

▶ **To cite this version:**

Victor Elvira, Samir Orujov, Audrey Poterie, Farid Rajabov, Francois Septier. VS-LTGARCHX: A Flexible Subset Selection Approach for Estimation of log-TGARCHX Models and Its Application to BTC Markets. 2023. hal-04283159v2

HAL Id: hal-04283159

https://hal.science/hal-04283159v2

Preprint submitted on 15 Nov 2023

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Journal of the Royal Statistical Society, Series A (Statistics in Society), 2022, 1–19
doi: 10.1111/rssa.12993
Advance Access Publication Date: Day Month Year
Paper

VS-LTGARCHX: A Flexible Subset Selection Approach for Estimation of log-TGARCHX Models and Its Application to BTC Markets

Victor Elvira[1], **Samir Orujov**[2,*], **Audrey Poterie**[2], **Farid Rajabov**[3] **and Francois Septier**[2]

[1]School of Mathematics , University of Edinburgh, Peter Guthrie Tait Road, EH9 3FD, Edinburgh, United Kingdom, [2]Université Bretagne Sud, UMR CNRS 6205, LMBA, F-56000 Vannes, France and [3]Institute of Finance and Technology, University College of London, Gower Street, WC1E 6BT, London, United Kingdom

*Corresponding author. samir.orujov@univ-ubs.fr

Abstract

The log-TGARCHX model is less restrictive in terms of inclusion of exogenous variables and asymmetry lags compared to the GARCHX model. However, adding less (more) covariates than necessary may lead to underfitting (overfitting), respectively. In this context, we propose a new algorithm, called VS-LTGARCHX, which incorporates a variable selection procedure into the log-TGARCHX estimation process. Furthermore, the VS-LTGARCHX algorithm is applied to extremely volatile BTC markets using 42 conditioning variables. Interestingly, our results show that the VS-LTGARCHX models outperform the specified benchmark models in one-step-ahead forecasting.

Key words: GARCHX, log-GARCH, Variable Selection, Bitcoin Volatility

1. Introduction

The generalized autoregressive conditional heteroskedasticity model (GARCH) due to Bollerslev (1986) is a popular class of econometric models to capture the dynamics of conditional variance (volatility) of time series. A natural extension of the GARCH model, called GARCHX, uses exogenous variables to model the conditional variance of the time series. In fact, additional conditioning variables can significantly improve volatility modeling, as correctly stated by Sucarrat (2020). Nevertheless, in GARCHX models, a positivity restriction on the intercept parameter and non-negativity constraints on all other parameters and all exogenous variables should be imposed. The latter restrictions are to ensure the positivity of conditional volatility, but limit the use of GARCHX models (Sucarrat, 2020). Furthermore, if we are interested in invertibility and stationarity, then additional conditions should be imposed on the autoregressive and moving average parameters of the conditional variance equation (Hamilton, 1994). This restrictive nature of most members of the GARCH family models is relaxed in the *exponential GARCH model with exogenous variables* (EGARCHX) (Nelson, 1991) and *logarithmic GARCH model with exogenous variables* (log-GARCHX). The log-GARCHX model is indeed the restricted form of *logarithmic GARCH model with exogenous covariates and threshold variables* (log-TGARCHX), as we emphasize herein further. Therefore, the terms log-GARCHX and log-TGARCHX are sometimes used interchangeably throughout the paper. In the last two members of the GARCH family (EGARCHX and log-TGARCHX), fitted volatility is guaranteed to be positive, and non-negativity constraints on parameters and covariates are lifted. However, the advantage of log-GARCHX over EGARCHX is that unconditional variance in the EGARCH formulation may not exist if the errors are too fat-tailed (Sucarrat, 2019). Moreover, probabilistic properties of EGARCH and log-GARCH models have been studied, and it was shown that the consistency and asymptotic normality of the quasi-maximum likelihood estimation (QMLE) of log-GARCH models hold under conditions milder than those of EGARCH. Furthermore, log-GARCH models may capture richer asymmetry dynamics and its shock transmission mechanism may fit the reality better compared to EGARCH and GARCH (Francq et al., 2013).

Arguably, one of the most attractive features of the log-TGARCHX model is that it accepts the ARMA representation and therefore the parameter estimation can be carried out in any standard statistical software using least squares and quasi-maximal likelihood (Francq and Thieu, 2019; Sucarrat, 2019). Moreover, the latter makes the standard asymptotic theory about parameter estimates valid. Under these circumstances, hypothesis testing about the statistical significance of exogenous covariates and ARCH/GARCH parameters is standard. Conversely, because of the respective positivity and non-negativity constraints imposed on parameters in generic GARCHX

© The Author 2022. Published by Oxford University Press. All rights reserved. For permissions, please e-mail: journals.permissions@oup.com

models, the conventional asymptotic theory fails, and non-standard methods should be applied. These questions have been discussed in detail in Francq and Thieu (2019).

The contributions of this work are twofold. First, we propose an algorithm called VS-LTGARCHX to incorporate the variable selection procedure into the log-TGARCHX estimation process. The algorithm is very flexible in the sense that any variable selection procedure can be employed within its framework. However, in this paper, we use three widely used variable selection methods that will be elaborated on in Section 3, separately within the VS-LTGARCHX algorithm to select the best subset of regressors while estimating log-TGARCHX models. We show that enhancing log-TGARCHX models with suitable variable selection procedures generates useful results for conditional volatility modeling in terms of prediction and interpretation. Second, we apply the proposed VS-LTGARCHX algorithm to time series of Bitcoin (BTC), one of the most popular decentralized cryptocurrencies operating on the blockchain (distributed ledger) platform. In fact, BTC has drawn public attention since its inception because of the novel underlying technology that enables the fast transfer of funds internationally without third-party involvement. However, the log-TGARCHX model has not yet been used to study the volatility of BTC, despite the former's higher flexibility compared to many other GARCH family models. We contribute to the existing body of literature by filling in the latter gap.

The paper is structured as follows. In Section 2, we describe and contrast the GARCHX and log-TGARCHX models and emphasize the advantages of the latter over the former in terms of the inclusion of exogenous variables and model estimation. In Section 3, we present a novel procedure to select the subset of variables from a much larger subset of variables to be used within the log-TGARCHX model. In Section 4, we apply our procedure to the BTC market and show that, indeed, the one-step-ahead volatility predictions can be improved using the suggested procedure, and in Section 5 we conclude.

2. On generalized autoregressive conditional heteroskedasticity based models

This section briefly introduces the GARCHX and the log-TGARCHX, and we discuss why the latter may be superior to the former, particularly in the presence of multiple exogenous variables. The last subsection provides the ARMA representation of log-TGARCHX models, which helps to simplify the parameter estimation and inference procedures.

2.1. The GARCHX model

Assume a covariance stationary process given by

$$y_t = \mu_t + \epsilon_t, \quad \epsilon_t = \sigma_t \eta_t, \quad \sigma_t > 0, \quad \eta_t \sim iid\,(0,1)\,, \tag{1}$$

where $\{y_t\}_{t\geq 0}$ is the time series process (e.g. financial return), μ_t is the conditional mean function known *a priori*, and the error term ϵ_t is a random variable with mean zero and conditional variance σ_t^2. The term η_t is an independently and identically distributed innovation process with zero mean and unit variance, as indicated by $iid\,(0,1)$.

In Eq. (1), the conditional variance of y_t can evolve over time, while the unconditional variance is assumed to be fixed (which is implied by the stationarity of y_t). When the conditional variance of the errors (and thus that of the y_t) is fixed, they are referred to as homoscedastic errors. Conversely, if the conditional variance of ϵ_t is varying, we call them heteroskedastic errors. The presence of heteroskedasticity may be undesirable in the standard linear regression framework, where the sole purpose of the researcher is to model the conditional mean of a time series process. Indeed, it may distort the standard error estimates of the model parameters, and thus lead to misleading statistical inference about those parameters. Heteroskedasticity, its potential challenges for statistical inference about linear regression parameters, and remedies are addressed extensively in most of the introductory and intermediate econometrics textbooks. More information about this point can be found in Brooks (2008); Wooldridge (2015); Greene (2017).

In some cases, modeling the conditional variance directly may be more crucial for some entities than modeling the mean. For example, risk-averse investors may be more interested in the variance of returns on their investment and value exposed to risk than in the mean of them because the variance is a measure of the risk. In fact, changes in variance have crucial implications for financial markets, since investors usually require higher compensation for exposing themselves to higher risks (Hamilton, 1994). The latter creates a need for an explicit modeling of the conditional variance of returns in financial markets.

The GARCH model of Bollerslev (1986) is a parametric model for the direct modeling of conditional variance. This model is the generalized version of the *autoregressive conditional heteroskedasticity model* (ARCH) due to Engle (1982). Both models assume linear dynamics for the conditional variance of the ϵ_t process in Eq. (1). Formally, a GARCH(p_1,p_2) model is defined by

$$\begin{aligned} &\sigma_t^2 = \alpha_0 + \sum_{i=1}^{p_1} \alpha_i \epsilon_{t-i}^2 + \sum_{j=1}^{p_2} \alpha_{p_1+j} \sigma_{t-j}^2, \\ &\alpha_0 > 0, \quad \alpha_i \geq 0, \quad \forall i \in \{1,\ldots,p_1+p_2\}, \end{aligned} \tag{2}$$

where α_0 is the volatility intercept and $p_1 \in \mathbb{N}$ and $p_2 \in \mathbb{N}$ are ARCH (squared lags of ϵ_t) and GARCH (squared lags of σ_t) orders respectively. Note that if $p_2 = 0$, the GARCH(p_1,p_2) model is the ARCH(p_1) model.

Furthermore, the natural extension of the GARCH model in which exogenous variables can be added is called the GARCHX model (Sucarrat, 2020). Then, a GARCHX(p_1,p_2,p_3) is defined by

$$\begin{aligned}\sigma_t^2 =&\alpha_0 + \sum_{i=1}^{p_1} \alpha_i \epsilon_{t-i}^2 + \sum_{j=1}^{p_2} \alpha_{p_1+j} \sigma_{t-j}^2 + \sum_{k=1}^{p_3} \beta_k x_{k,t}, \\ &\alpha_0 > 0, \quad \alpha_i \geq 0 \quad \forall i \in \{1, \ldots, p_1 + p_2\}, \\ &\beta_k \geq 0 \text{ and } x_{k,t} \geq 0 \quad \forall k = \{1, \ldots, p_3\},\end{aligned} \tag{3}$$

where $\{x_{k,t}\}_{k=1}^{p_3}$ are p_3 stationary exogenous covariates and $\{\beta_k\}_{k=1}^{p_3}$ are the unknown associated parameters to be estimated. Other parameters in Eq. (3) are GARCHX(p_1,p_2,p_3) parameters defined in Eq. (2). The inclusion of exogenous covariates into the GARCH equation comes with the additional cost of non-negativity restrictions on both the exogenous variables $\{x_{k,t}\}_{k=1}^{p_3}$ and their associated parameters $\{\beta_k\}_{k=1}^{p_3}$ in order to guarantee the positivity of the conditional variance. Furthermore, inference procedures under nullity are nonstandard (Sucarrat, 2020) and there is a need for alternative asymptotics, as in Pedersen and Rahbek (2019). This restrictive nature of GARCHX models is relaxed to a great extent by log-TGARCHX models, which we demonstrate in the next subsection.

2.2. The log-TGARCHX model

The log-GARCHX model (Sucarrat, 2019) assumes logarithmic specification of the conditional variance process and includes both asymmetry error terms and stationary covariates. Formally, the log-GARCHX(p_1,p_2,p_3,p_4) is defined as follows:

$$\begin{aligned}&\ln \sigma_t^2 = \alpha_0 + \sum_{i=1}^{p_1} \alpha_i \ln \epsilon_{t-i}^2 + \sum_{j=1}^{p_2} \alpha_{p_1+j} \ln \sigma_{t-j}^2 + \sum_{k=1}^{p_3} \beta_k x_{k,t} + \sum_{l=1}^{p_4} \gamma_l \mathbb{I}_{\{\epsilon_{t-l}<0\}} \ln \epsilon_{t-l}^2, \\ &\alpha_i, \beta_k, \gamma_l \in \mathbb{R}, \ \forall i \in \{0, \ldots, p_1 + p_2\}, \ \forall k \in \{1, \ldots, p_3\}, \ \forall l \in \{1, \ldots, p_4\},\end{aligned} \tag{4}$$

where α_0 is log-volatility intercept which controls the level of volatility in a multiplicative way, $(\alpha_i)_{i=\{1,\ldots,p_1\}}$ and $(\alpha_{p_1+j})_{j=\{1,\ldots,p_2\}}$ are real scalar sequences called ARCH-parameters and GARCH-parameters, respectively (Sucarrat, 2019). Note that, when $p_3 = p_4 = 0$, the log-GARCHX(p_1,p_2,p_3,p_4) model is none other than the log-GARCH(p_1, p_2) model introduced by Pantula (1986), Geweke (1986) and Milhoj (1987) independently.

The log-GARCHX model is referred to as the asymmetric log-GARCHX model in the literature (Sucarrat, 2019). However, we suggest calling the latter log-TGARCHX because it is equivalent to the TGARCHX model of Glosten et al. (1993) except for the logarithmic transformation.

2.3. ARMA representation of log-TGARCHX model

The log-GARCH family of volatility models has several attractive features. One of them is the representability via non-linear ARMA process, which makes those models amenable to estimation using ARMA inference methods and almost with any statistical software. Due to the latter fact, the standard inference procedures remain applicable. Furthermore, the addition of exogenous, deterministic, and / or predetermined conditioning variables does not change the relationship between the ARMA coefficients and the log-GARCH coefficients under suitable conditions (Sucarrat et al., 2016; Sucarrat, 2019). More precisely, the ARMAX representation for the log-TGARCHX model in Eq. (4) is as follows (Sucarrat, 2019):

$$\ln {\epsilon_t}^2 = \phi_0 + \sum_{i=1}^{p_1} \phi_i \ln \epsilon_{t-i}^2 + \sum_{j=1}^{p_2} \phi_{p_1+j} u_{t-j} + \sum_{k=1}^{p_3} \beta_k x_{k,t-1} + \sum_{l=1}^{p_4} \gamma_l \mathbb{I}_{\{\epsilon_{t-l}<0\}} \ln \epsilon_{t-l}^2 + u_t, \tag{5}$$

with

$$\begin{aligned}u_t &= \ln \eta_t^2 - \mathbb{E}\left(\ln \eta_t^2\right) \\ \phi_i &= \begin{cases} \alpha_0 + \left(1 - \sum_{j=1}^{p_2} \alpha_{p_1+j}\right) \times \mathbb{E}\left(\ln \eta_t^2\right), & i = 0 \\ \alpha_i + \alpha_{p_1+i}, & 1 \leq i \leq p_1 \\ -\alpha_i, & p_1 + 1 \leq i \leq p_2. \end{cases}\end{aligned} \tag{6}$$

The parameter values $(\beta_l)_{l=\{1,\ldots,p_3\}}$ for exogenous variables are the same in both the log-GARCHX specification and the corresponding ARMAX representation. The possibility of the ARMAX representation paves the way for a wide range of extensions of the log-GARCH-family models. In this paper, we propose enhancing log-GARCHX models by including a variable selection procedure during model estimation. Although there are many variable selection methods, in this work we focus on three of them. However, other selection methods can also be easily adapted to the proposed framework. The following section describes our original methodology and the three variable selection methods.

3. Enhancing log-TGARCHX model by adding variable selection algorithms

3.1. Variable selection procedure for log-TGARCHX model (VS-LTGARCHX)

As described by Sucarrat (2019), the log-TGARCHX model is estimated in two steps. The first step is to form the ARMAX (p_1, p_2, p_3, p_4) representation of the model as given in Eq. (5) and estimate the coefficients using the ordinary least squares (OLS) method. The second step consists of mapping the estimated coefficients to the coefficients of the model of interest, namely, log-TGARCHX specified in Eq. (4) using the maps in Eq. (6).

As an important contribution of this paper, we show that the log-TGARCHX model can be improved by selecting a relevant subset of exogenous variables and asymmetry lags, denoted by $K_\star \subseteq \{1, \ldots, p_3\}$ and $L_\star \subseteq \{1, \ldots, p_4\}$, respectively. To that end, we propose a heuristic strategy that incorporates variable selection within the log-TGARCHX estimation procedure. More precisely, this approach describes i) how to select both the subset of exogenous variables, through the $L_\star$ index set, and that of the asymmetry terms through the $K_\star$ index set to be used in the log-TGARCHX model, and then ii) how to estimate parameters of the model. This new strategy that is made up of three steps is described below.

1. *Step 1. Fit an ARMA representation and get the residuals*
   First, we fit the following ARMA(p_1,p_2) model:

$$\ln {\epsilon_t}^2 = \phi_0 + \sum_{i=1}^{p_1} \phi_i \ln \epsilon_{t-i}^2 + \sum_{j=1}^{p_2} \phi_{p_1+j} u_{t-j} + u_t. \tag{7}$$

   The model is fitted by using maximum likelihood estimation together with the innovations algorithm (Brockwell and Davis, 1991). Then, we compute the model residuals which are defined as:

$$\hat{z}_t = \ln {\epsilon_t}^2 - \widehat{\ln {\epsilon_t}^2}, \tag{8}$$

   where $\widehat{\ln {\epsilon_t}^2}$ denotes the adjusted value of $\ln {\epsilon_t}^2$ for model Eq. (7).
2. *Step 2. Perform the variable selection procedure for both exogenous variables and asymmetry orders*
   Next, variable selection is applied on the following model:

$$\hat{z}_t = a_0 + \sum_{k=1}^{p_3} b_k \mathbb{I}_{\{\epsilon_{t-k}<0\}} \ln \epsilon_{t-k}^2 + \sum_{l=1}^{p_4} c_l x_{l,t-1} + e_t, \tag{9}$$

   where $a_0, \{b_k\}_{k=1}^{p_3}, \{c_l\}_{l=1}^{p_4}$ are the model parameters to be estimated, $\mathbb{I}_{\{\epsilon_{t-k}<0\}} \ln \epsilon_{t-k}^2$ is the k-th lagged asymmetry term evaluated at time t, $x_{l,t-1}$ is the l-th stationary exogenous covariate evaluated at time t and e_t is the model error at time t. In this paper, we suggest applying one of the three variable selection methods introduced in Section 3.2.1. As a result of the variable selection process, we obtain the index sets $L_\star$ and $K_\star$ of size $p_{3\star}$ and $p_{4\star}$, respectively.
3. *Step 3. Fit the final log-TGARCHX model*
   Finally, we can estimate the log-TGARCHX(p_1, p_2, $p_{3\star}$, $p_{4\star}$) model:

$$\ln \sigma_t^2 = \alpha_0 + \sum_{i=1}^{p_1} \alpha_i \ln \epsilon_{t-i}^2 + \sum_{j=1}^{p_2} \alpha_{p_1+j} \ln \sigma_{t-j}^2 + \sum_{k \in K^*} \beta_k x_{k,t} + \sum_{l \in L^*} \gamma_l \mathbb{I}_{\{\epsilon_{t-l}<0\}} \ln \epsilon_{t-l}^2. \tag{10}$$

   Parameters $\{\alpha_i\}_{i=1}^{p_1+p_2}, \{\beta_k\}_{k\in K_\star}, \{\gamma_l\}_{l\in L_\star}$ of the final model are estimated using the two-step procedure described in Section 2.3.

In the next subsection, we shed light on the second step of the proposed methodology.

3.2. Variable selection algorithms

In this section, three variable selection strategies are introduced to select $L_\star$ and $K_\star$ (see *Step 2* in Section 3.1). As described in the previous section, the variable selection procedure is performed on the linear model Eq. (9). The subset selection procedure is important, especially when the set of potential regressors is large or when the regressors are highly correlated. For instance, when the number of regressors is larger than the number of observations, it is well known that linear models may suffer from high variance, which is referred to as overfitting in the literature. Furthermore, parsimonious models are more easily interpretable than complex models with many regressors. Therefore, the selection of variables to identify the most influential variables can improve both predictive accuracy and interpretability (Miller, 2002). In fact, there are many variable selection methods in the related literature. Here, we have chosen three approaches: *the least absolute shrinkage and selection operator* (LASSO), *adaptive best subset selection* (ABESS), and *Boruta*.

3.2.1. Variable selection using LASSO

LASSO is a regression method introduced by Tibshirani (1996). The method is still widely used to perform both variable selection and regularization. This is achieved by adding a penalty function of the model parameters. Consider the linear regression setting in Eq. (9). Then, the LASSO regression problem consists of estimating the model parameter vector $\Theta = \left\{a_0, \{b_k\}_{k=1}^{p_3}, \{c_l\}_{l=1}^{p_4}\right\}$ that solves the

following constrained OLS problem:

$$\widehat{\Theta}^{\text{lasso}} = \underset{\Theta\in\mathbb{R}^{(p_3+p_4+1)}}{\arg\min} \Big(\sum_{t=1}^{T} \big(z_t - a_0 - \sum_{k=1}^{p_3} b_k \mathbb{I}_{\{\epsilon_{t-k}<0\}} \ln \epsilon_{t-k}^2 - \sum_{l=1}^{p_4} c_l x_{l,t-1}\big)^2 + \lambda(\sum_{k=1}^{p_3} |b_k| + \sum_{l=1}^{p_4} |c_l|)\Big), \tag{11}$$

where the objective function is nothing else than the sum of mean squared errors and $\lambda > 0$ is a regularization parameter. LASSO regression shrinks the magnitude of all model coefficients and also sets some of them to zero. Then, in *Step 2* we use LASSO to perform the selection of variables in the model Eq. (9). The subsets of indices $L_\star$ and $K_\star$ obtained using LASSO are denoted by

$$\begin{aligned} K_\star^{\text{lasso}} &= \{k|\hat{b}_k^{\text{lasso}} \neq 0, \quad \forall k = 1,\dots,p_3\} \\ L_\star^{\text{lasso}} &= \{l|\hat{c}_l^{\text{lasso}} \neq 0, \quad \forall l = 1,\dots,p_4\} \end{aligned} \tag{12}$$

corresponds to the sets of indices of non-zero parameters. Note that the value of the regularization parameter λ is usually unknown and therefore needs to be tuned. Several approaches have been suggested to adjust λ. Here, we adopt the algorithm outlined in Section 2.5 of Friedman et al. (2010). The algorithm finds the minimum Lagrange multiplier value in the Lagrangian representation of the constrained optimization problem in the Eq. (11) which shrinks all slope coefficients to zero simultaneously, and that value is denoted by λ_{max}. Then λ_{min} is set to $0.001\lambda_{max}$ and the grid is formed as a decreasing sequence of 100 values from λ_{max} to λ_{min} on the log scale.

3.2.2. Adaptive best subset selection (ABESS) algorithm

ABESS is a recently developed polynomial algorithm due to Zhu et al. (2020) for the vaiable selection. In simple linear regression setting e.g. Eq. (9), this algorithm chooses the subset of covariates by solving the following minimization problem:

$$\widehat{\Theta}^{\text{abess}} = \underset{\Theta\in\mathbb{R}^{(p_3+p_4+1)}}{\arg\min} \sum_{t=1}^{T} \big(z_t - a_0 - \sum_{k=1}^{p_3} b_k \mathbb{I}_{\{\epsilon_{t-k}<0\}} \ln \epsilon_{t-k}^2 - \sum_{l=1}^{p_4} c_l x_{l,t-1}\big)^2 \quad \text{subject to} \quad (\sum_{k=1}^{p_3} ||b_k||_0 + \sum_{l=1}^{p_4} ||c_l||_0) \leq s, \tag{13}$$

where $s \in \{1,\dots,p_3 + p4\}$ is a regularization parameter and $||\cdot||_0$ is the ℓ_0 norm.

Here, the Lagrangian of the problem is not continuous, and its solution may be computationally infeasible with large p. However, a recently developed splicing algorithm and special information criterion (SIC) have been proven to solve the minimization problem above in polynomial times and for an unknown sparsity parameter (Zhu et al., 2020).

Although the ABESS algorithm is extremely fast compared to its competitors as shown in the latter reference, it seems to be more aggressive compared to LASSO in penalization as the form of the constraint function suggests. Therefore, when the set of the covariates upon which variable selection needs to be performed is relatively small, LASSO should be preferred. However, a comparison of the two is outside the scope of our study and needs further research.

Finally, one of the main advantages of ABESS is that the potential values of the tuning parameter s are finite, as the l_0 norm penalty suggests. The selected variables by the ABESS algorithm are the ones which receive non-zero coefficients after the minimization of the latter equation. More precisely,

$$\begin{aligned} K_\star^{\text{abess}} &= \{k|\hat{b}_k^{\text{abess}} \neq 0, \quad k = 1,\dots,p_3\} \\ L_\star^{\text{abess}} &= \{l|\hat{c}_l^{\text{abess}} \neq 0, \quad l = 1,\dots,p_4\}. \end{aligned} \tag{14}$$

3.2.3. Boruta algorithm

Boruta is an iterative feature selection algorithm based on the random forest algorithm (Kursa and Rudnicki, 2010). At each iteration, the algorithm creates randomly shuffled copies of the original features. Then, a random forest is trained on the extended data that includes both the original features and their respective shuffled copies. To measure the importance of each feature, the algorithm computes a Z score on each feature and compares it to a threshold defined as the maximum Z score of all shuffled copies. Then, only variables for which importance is significantly higher than the threshold are kept in the model while the others are removed. This process is repeated until either all features are labeled as important/nonimportant or until the maximum number of iterations is reached. The detail of the algorithm is given in the original article by Kursa and Rudnicki (2010). Ultimately, the two sets $K_\star^{\text{Boruta}}$ and $L_\star^{\text{Boruta}}$ correspond to the sets of indices of the retained exogenous variables and the asymmetric lags.

4. Application to BTC market

In this section, we apply the VS-LTGARCHX algorithm to the BTC market and study its performance compared to the log-GARCH (1,1) model and the full-blown log-TGARCHX model. The log-GARCH(1,1) model is a log-GARCH model with no exogenous variable, while the full-blown log-TGARCHX model is a log-GARCH model with the complete set of exogenous variables and asymmetry terms. These *two extremes* form the benchmark models with which to compete. The next subsection describes the BTC data used.

4.1. Data: data sources and data preprocessing

The sample period runs from December 18, 2017 to June 17, 2022 and includes 1643 observations in total collected at daily frequency. It is worth mentioning that the BTC price data were collected both in 5-minute and daily frequencies, which were taken from Binance

exchange. The BTC price data at the daily frequency were used to obtain the BTC daily returns. Moreover, the BTC prices at 5-minute frequency were used to obtain the daily realized variance as described in (Bergsli et al., 2022). Formally, let R_t, the return on the t-th day, be defined as

$$R_t = \ln P_t - \ln P_{t-1}, \tag{15}$$

where P_t is the close price of BTC on day t. Then, we define m equally spaced intraday returns between each consecutive two days t and $t+1$. More precisely, the j-th intraday return for the the t-th day is denoted by $R_{t+\frac{j}{m}}$ and defined as:

$$R_{t+\frac{j}{m}} = \ln\left(P_{t+\frac{j}{m}}\right) - \ln\left(P_{t+\frac{j-1}{m}}\right), \quad \text{for } j = 0, \ldots, m-1, \tag{16}$$

where $\ln\left(P_{t+\frac{j}{m}}\right)$ (resp. $\ln\left(P_{t+\frac{j-1}{m}}\right)$ is the logarithm of the intraday price of BTC at time $t+\frac{j}{m}$ (resp. $t+\frac{j-1}{m}$). Thus, the daily realized variance, also called volatility, for the t-th day can be defined as the sum of squared intraday returns following the previous reference:

$$V_t^{(d)} = \sum_{j=0}^{m-1} R^2_{t+\frac{j}{m}}. \tag{17}$$

where the subscript d is used to emphasize the daily frequency of the realized variance.

For the purposes of this paper, we used BTC prices in 5-minute frequency to estimate the daily realized variance (volatility), which implies that m in Eq. (16) and Eq. (17) is set to 288 throughout our analysis.

Furthermore, we classify the exogenous variables into groups using a classification similar to, but not exactly the same as Chen et al. (2021). More precisely, the groups we defined are Blockchain technology, Public Opinion, Risks and Uncertainties, Financials, and Macroeconomic development. Also, the number of variables we use is more than in the latter reference. To be more precise, we consider 39 exogenous variables, 3 variables which are transformations of high-frequency returns (daily, weekly and monthly realized volatility) and a threshold variable, which constitute 42 variables in total. The threshold variable is nothing but a binary variable which receives the value of unity if the previous day return is negative and the value of zero if otherwise. The complete list of exogenous variables is given in the Appendix A. All variables were collected at daily frequency. However, unlike BTC prices, measurements on most exogenous variables are missing on weekends and public holidays. Then, to overcome inconsistency in frequency, the last observation carried forward (LOCF) method was used to fill the missing values.

In addition, we used only stationary exogenous variables in our study as shown in (Sucarrat, 2019). Therefore, Augmented Dickey-Fuller (ADF) test due to Dickey and Fuller (1979) was applied to each exogenous variable to identify non-stationary ones. The non-stationary variables that take only positive values were exposed to the logarithmic difference transformation ($d\ln(X_t) = \ln X_t - \ln X_{t-1}$) and the prefix $dl\_$ was added to their names. In contrast, non-stationary variables with non-positive and positive values were subject to first differencing ($X_t \mapsto X_t - X_{t-1}$), and the prefix $d\_$ was added to their names. Furthermore, we applied the test of stationarity (Kwiatkowski et al., 1992) to the data to confirm the results of the ADF test which is a standard practice (Brooks, 2008, p. 331). If the stationarity hypothesis is rejected by KPSS those series were subject to the first differencing and the prefix $d\_$ was added to their names. In addition, all covariates were standardized to have zero mean and unit variance to prevent variables from having a different measurement scale that could affect the results of subset selection.

Finally, in the next subsection, we will describe in detail how the VS-LTGARCHX algorithm is applied to BTC market data and how the performance of different variants of the algorithm is evaluated.

4.2. Model performance and comparison to benchmark

Once the data preprocessing is performed, we divide the data set into train, test, and validation sets. The hyperparameters of the variable selection algorithms are then tuned. Ultimately, the final models are built and their respective performances are measured. The following subsections describe the methodology we have used to perform each of the latter steps.

4.2.1. Data split

First, the time series data set was divided into train, validation, and test sets. More precisely, the data partitioning consisted of dividing the sample period (from December 18, 2017 to June 17, 2022) into three sub-periods corresponding, respectively, to the train, validation and test sets.

The train set is used to build a prediction model (*a learner*) with a fixed set of hyperparameters. The validation set is used to select the best combination of hyperparameters for the respective variable selection method based on some performance metrics. Ultimately, the test set is used to evaluate the performance of the final model, the one fitted with the best combination of hyperparameters (Hastie et al., 2001).

Moreover, the benchmark models considered in this paper do not have any hyperparameters to tune. Therefore, the validation set is not needed, so it was merged with the train set for the benchmark models.

Finally, to respect the dynamic structure of the data, we use the recursive and rolling window scheme while partitioning the data set for all models considered (Elliott and Timmermann, 2008).

4.2.2. Hyperparameter tuning

As has already been mentioned, the three variable selection methods, namely LASSO, ABESS, and Boruta, have hyperparameters that need to be tuned. To do that, we used two strategies: one for LASSO and ABESS and another for Boruta. In fact, a more efficient

strategy can be used for Boruta, since it is a wrapper around the random forest algorithm (Breiman, 1996) that is heavily based on bootstrap aggregation (bagging).

To tune the hyperparameters of LASSO and ABESS, first, we fit the prediction models (*learners*) to the training set by fixing the hyperparameter value based on a predefined grid. Then, for each hyperparameter value, we obtain the prediction errors of the *learners* in the validation set. The value of the tuning parameter associated with the smallest validation *root mean squared error* (RMSE) is considered optimal.

The hyperparameter tuning procedure for the random forest behind the Boruta wrapper can be done more efficiently due to the bootstrap resampling applied to the initial training set (Breiman, 2001). However, the use of the original bootstrap strategy would destroy the underlying temporal dependence in time series. Therefore, we utilize the moving block bootstrap mechanism to respect the dynamic nature of the data while conducting the bagging process. The idea is to divide the initial training set into N small blocks so that the temporal structure is preserved within each block, and then to perform a bootstrapping using the N blocks (Kunsch, 1989; Y, 1992). Unlike LASSO and ABESS which have only one hyperparameter to be tuned each, in Boruta there are multiple of them, which makes it costly to train the latter. Finally, we tune four hyperparameters; three of them are intrinsic to Boruta, namely, the number of trees, the minimum node size, the number of splitting variables, and one is due to the moving block bootstrap algorithm, namely, the block size (see Algorithm 2 in the Appendix C). Again, the set of hyperparameters associated with the smallest RMSE is considered optimal.

4.2.3. Evaluation of the predictive performance and comparison to the benchmark models

Once the optimal values of the hyperparameters are found, the variable selection is carried out by combining the training and the validation sets, and this is the end of *Step 2* of the VS-LTGARCHX algorithm. The final step *Step 3* consists of fitting the log-TGARCHX model as explained in detail in Escribano and Sucarrat (2018) and Sucarrat (2019) with the variables selected by the respective variable selection algorithms.

Moreover, we use the same portion of time series to build the final model for both benchmarks and VS-LTGARCHX variants, and use the same strategy for all models considered to generate one-step-ahead predictions. More precisely, to measure performance of each model, we generate one-step-ahead predictions from the test set (the same portion of time series for all models for fair comparison) using the rolling window scheme and finally calculate six different conventional metrics for each model based on their respective forecasts. The procedure is outlined in Algorithm 3.

Remark 1 *The rolling window scheme is used instead of the recursive window scheme because it is known that the recursive window scheme may pose problems when evaluating nested models (Violante and Laurent, 2012, p. 466) and that the stationarity assumption of relative losses underlying the Model Confidence Set (MCS) test may not hold when using the expanding window scheme (Hansen et al., 2011).*

Remark 2 *We use the square root of daily realized volatility shown in Eq.* (17) *as a proxy for the true conditional standard deviation* (σ_t) *when calculating the performance metrics in accordance with the related literature (Violante and Laurent, 2012).*

Indeed, the judgment about predictive superiority of models relying solely on performance metrics may not be convincing due to underlying sampling issues. Therefore, we use the MCS procedure due to (Hansen et al., 2011), which involves hypotheses testing and model selection, to make inferences about the ranking of the models. When several models have been built, the MCS procedure helps in determining whether the performance of a given model in the set of tested models is statistically significant in comparison to the performance of other tested models, see Hansen et al. (2011) for more details. Furthermore, in the MCS procedure, different aspects of the models can be evaluated by using user-specified loss functions (Bernardi and Catania, 2015). Here, we use two loss functions, namely, RMSE and QLIKE, within the MCS procedure due to the robustness of the latter (Patton, 2011). However, other metrics such as mean error (ME), mean absolute error (MAE), mean absolute percentage error (MAPE), and mean percentage error (MPE) are also reported independently in the Appendix E.

The strategy used to assess the performance of the VS-LTGARCHX algorithm compared to the benchmark models is summarized in the Algorithm 3 displayed in the Appendix D.

4.3. Numerical results

4.3.1. Data preprocessing, data partition and parameter tuning

We collect daily data on the BTC market and 44 variables measured in daily frequency covering the period from December 18, 2017 to June 17, 2022. The missing values, if any, were filled using the last observation carried forward principle. The complete list of the variables is given in Table 4, 5 and 6 of the Appendix A. After the data are preprocessed and all variables are stationarized and standardized to have zero mean and unit variance, we partition the data set into train, validation and test sets in 60%-20%-20% chunks, initially (see Algorithm 1).

In the next subsections, we will shed light into the set of variables selected by LASSO, ABESS, and Boruta, respectively, and then compare the VS-LTGARCHX variants' predictive accuracy to those of the benchmark models.

4.3.2. Selected variables by VS-LTGARCHX algorithms

Before performing variable selection, the variable selection algorithms, namely LASSO, ABESS, and Boruta, must be calibrated. We follow the strategies introduced in Subsection 4.2.2. The hyperparameter tuning process for LASSO and ABESS is outlined in Algorithm 1, and that for random forest underlying the Boruta wrapper is summarized in Algorithm 2. The optimal values of the tuning parameters help us to choose a subset of variables from a larger set for each of the respective methods. The variables selected by various selection procedures are given in Table 1.

Table 1. Variables Selected by ABESS, LASSO and Boruta, respectively

#	ABESS	LASSO	Boruta
1	d_VIX	dl_MKTCP	dl_CPTRA
2	d_gtrends	d_VIX	dl_ETRVU
3	rv_w	d_gtrends	dl_HRATE
4		rv_w	dl_MWNTD
5			dl_MWTRV
6			dl_NADDU
7			dl_NTRAN
8			dl_NTRBL
9			dl_NTREP
10			dl_SPY
11			dl_DJI
12			d_CPTRV
13			d_ETRAV
14			d_TRFEE
15			d_TRVOU
16			d_VIX
17			d_gtrends
18			rv_d
19			rv_w

ABESS selected the most parsimonious model as expected. Interestingly, the composition of the variables selected by LASSO and ABESS is almost the same except LASSO selects one extra variable, namely, the first lag of the logarithmic difference of BTC market capitalization (dl_MKTCP). Moreover, the first lag of the difference in VIX index is selected by the three selection procedures. In fact, it has recently been found that the logarithmic difference of the VIX index has contemporaneous effects on the volatility of BTC (López-Cabarcos et al., 2021). In our study, we focus solely on lagged effects because contemporaneous feedback is not usually helpful for investor decision-making.

Another variable selected by the three procedures is the lagged difference of Google trends (d_gtrends). Interestingly, Google trends lags were found to have predictive power over BTC returns (Arratia and López-Barrantes, 2021). Moreover, several other recent studies have found the link of the Google trends variable with conditional volatility (Bystrom and Krygier, 2018; Aslanidis et al., 2022; Bakas et al., 2022).

Furthermore, the lag of the weekly realized volatility selected by the three selection procedures has already been shown to be effective in predicting the volatility of BTC (Aharon and Qadan, 2019; Bergsli et al., 2022). However, it is still an open question why the lag of daily and monthly realized volatilities were not selected by LASSO and ABESS as opposed to the latter references.

Interestingly, among the three variable selection methods, only Boruta selects a large set of technological variables. Indeed, we could not find a study to support this interesting finding, and the effect of BTC technology variables on its volatility should be further researched.

4.3.3. Predictive accuracy of VS-LTGARCHX algorithms

Before delving into the analysis of results and comparative statistics, one fact is worth noting to assert the importance of explanatory variables for conditional volatility forecasting. Figure 1 illustrates the forecasts generated from the two benchmark models. According to the latter, the full-blown log-TGARCHX model seems to perform better than log-GARCH(1,1) particularly during large volatility periods. Moreover, after the variable selection procedure is finished we generate one-step-ahead predictions from three variants of VS-LTGARHCX algorithm using the test set as shown in Algorithm 3. Figure 2 shows one-step-ahead forecasts of BTC conditional volatility from three different variants of the VS-LTGARCHX algorithm. As the figure reveals, the forecasts generated using the variants of the VS-LTGARCHX algorithm are more precise compared to the benchmark during relatively high volatility periods.

Furthermore, according to Table 2, the three variants of VS-LTGARCHX lead to better predictive accuracy than the log-GARCH (1,1) and log-TGRACHX benchmarks in terms of RMSE and QLIKE. Moreover, the ABESS and LASSO variants always outperform the Boruta alternative with regard to the latter criteria.

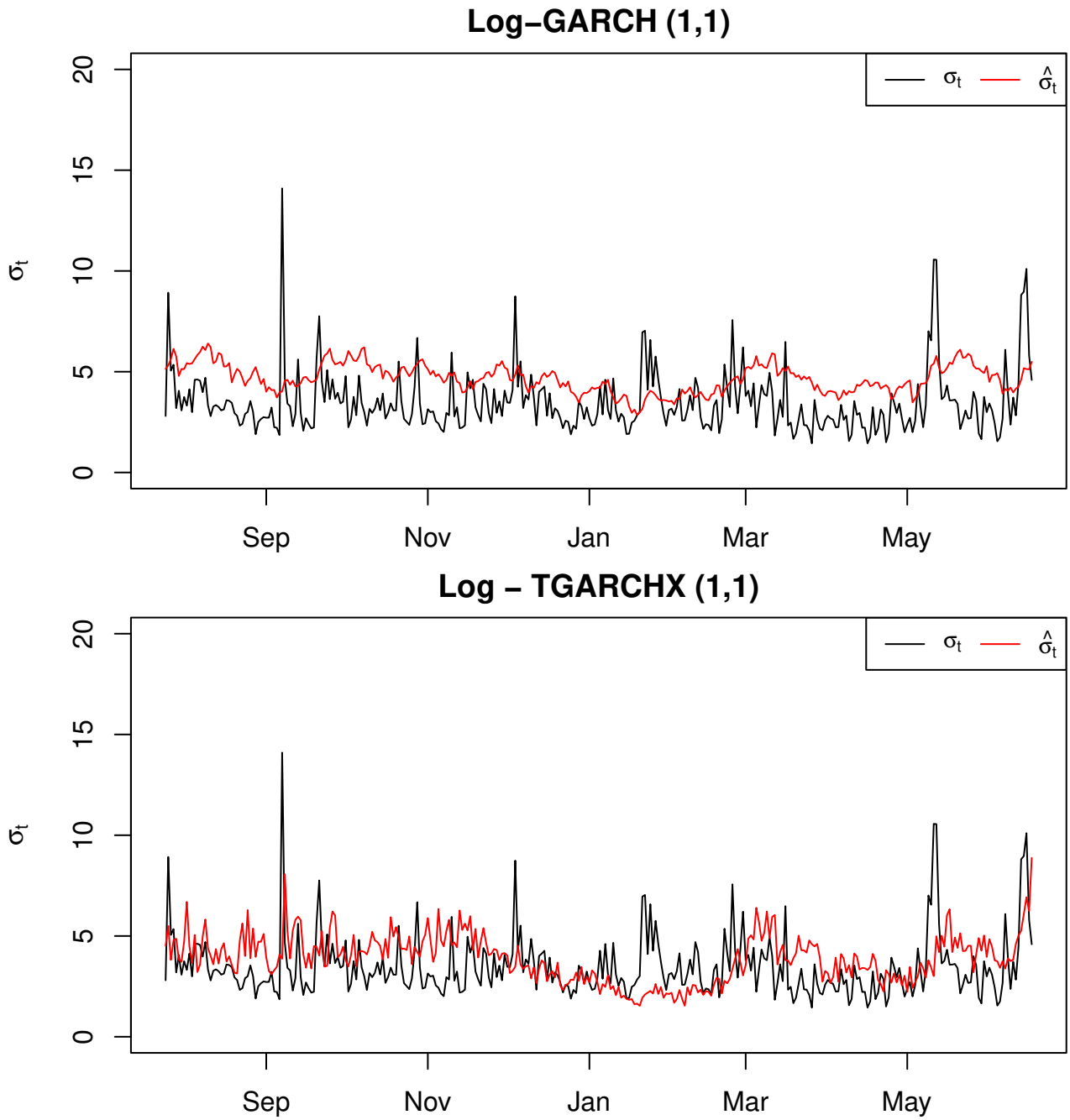

Fig. 1: Plot of one-step-ahead forecasts from log-GARCH, log-TGARCHX models and realized standard deviation

On the other hand, the results of the MCS procedure are summarized in Table 3. The intrinsic hypothesis test for MCS was carried out at the significance level 5%. The null hypothesis states that the model has predictive performance equivalent to all other models considered (Bernardi and Catania, 2015). Then, the p-value for each model represents the probability of observing a sample performance of that particular model at least as adverse to the null hypothesis as the current one. Therefore, it is natural for models with a p-value lower than the significance level (5%) to be removed from the set considered. The MCS procedure (hypothesis test+elimination) continues until no model can be removed from the set and the final set of surviving models is called the superior set models (SSM) (Hansen et al., 2011; Bernardi and Catania, 2015).

As shown in Table 3, the MCS procedure has eliminated VS-LTGARCHX (Boruta) from SSM while using RMSE as a loss criterion. This means that the latter is inferior to VS-LTGARHCX (LASSO) and VS-LTGARHCX (ABESS). However, it does not mean that VS-LTGARCHX (Boruta) has the same predictive ability as log-GARCH (1,1) and log-TGARCHX models. In another experiment, we

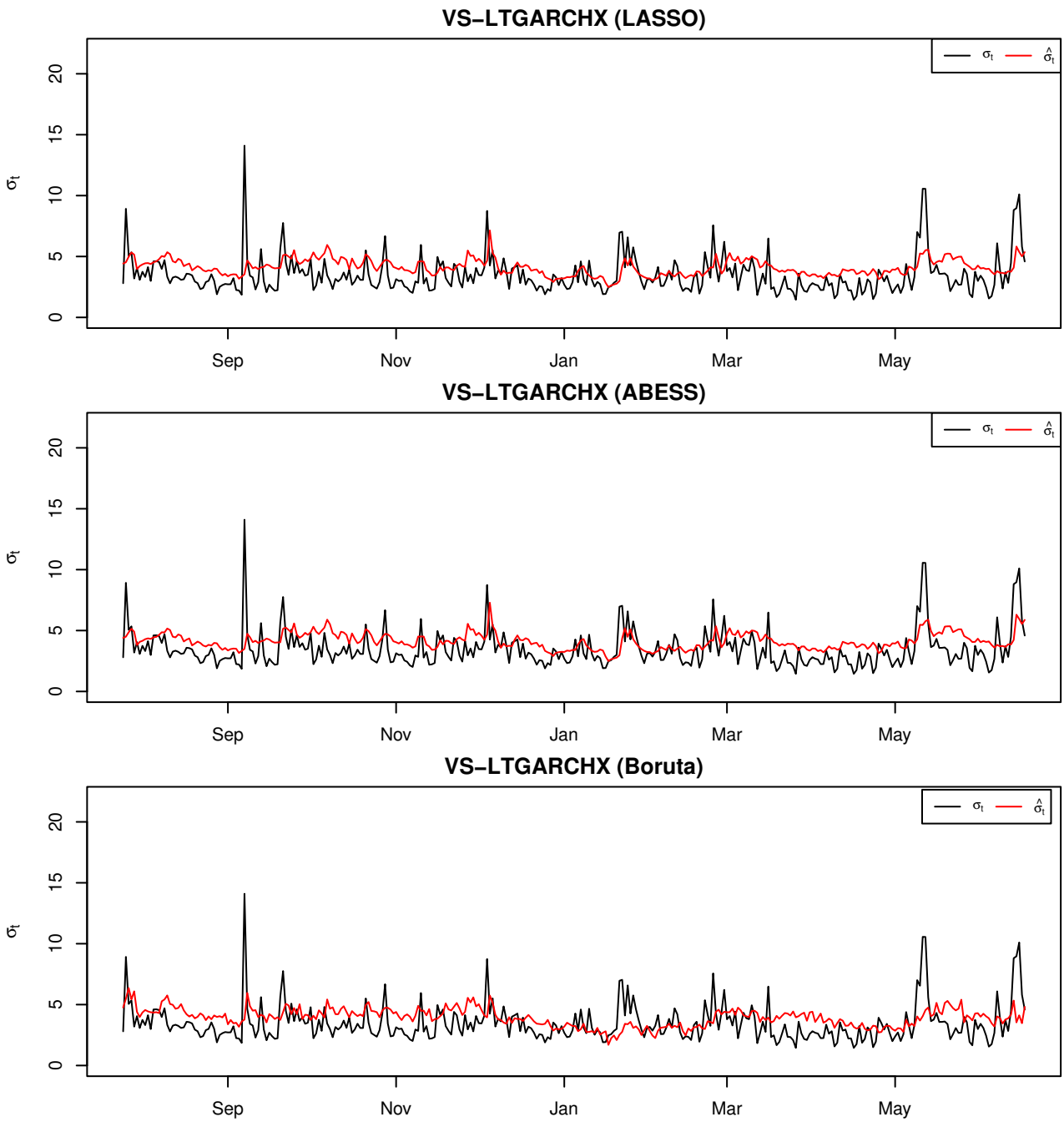

Fig. 2: Plot of forecasts from VS-LTGARCHX using LASSO, ABESS and Boruta variable selection procedures, respectively

restricted the initial set of the models considered ($\mathcal{M}^0$) to include only the models VS-LTGARCHX (Boruta), log-GARCH (1,1) and log-TGARCHX. Not surprisingly, VS-LTGARCHX (Boruta) was found to be superior to the other two models, using both MSE and QLIKE as loss criteria. This means that the VS-LTGARCHX algorithm outperformed the benchmarks in all of its variants. However, the Boruta variant is inferior to the LASSO and ABESS variants of VS-LTGARCHX.

As a robustness check, we report the extended list of performance metrics for the rolling window forecasting scheme in Table 7 of the Appendix E. The extended list of performance metrics includes not only QLIKE and RMSE, but also mean error (ME), mean absolute error (MAE), mean absolute percentage error (MAPE) and mean percentage error (MPE) (see Algorithm 3 in the Appendix D and Hyndman and Athanasopoulos (2018) for details).

Furthermore, it is not surprising that the full-blown log-TGARCHX model outperforms all other models in terms of ME and MPE as shown in Table 7. Tibshirani correctly points out that : "...the OLS estimates often have low bias but large variance; prediction accuracy

Table 2. Predictive accuracy of models in the test set using rolling window

Model	RMSE	QLIKE
Log-GARCH	2.00	3.74
Log-TGARCHX	1.85	3.82
LASSO	1.60	**3.64**
ABESS	**1.59**	**3.64**
Boruta	1.66	3.69

Table 3. MCS Superior Set Models (SSM) using MSE and QLIKE as loss criteria

MCS (QLIKE)		MCS (MSE)	
Superior Set Models (SSM)	p-value	Superior Set Models (SSM)	p-value
VS-LTGARCHX (ABESS)	1	VS-LTGARCHX (ABESS)	1
VS-LTGARCHX (LASSO)	0.33	VS-LTGARCHX (LASSO)	1
		VS-LTGARCHX (Boruta)	0.15

can sometimes be improved by shrinking or setting to 0 some coefficients. By doing so, we sacrifice a little bias to reduce the variance of the predicted values and hence may improve the overall prediction accuracy"(Tibshirani, 1996, p. 267). The unrestricted log-TGARCHX model (estimated using OLS) should have less bias compared to its restricted alternatives, as inferred from the latter quote. Therefore, ME and MPE which are nothing but the average and the weighted average of forecast errors, respectively, should have the smallest value for the unrestricted log-TGARCHX model.

Moreover, as an additional robustness check, we follow Hansen et al. (2011) and test our algorithm using the recursive forecasting scheme. An extended list of performance metrics for the recursive window forecasts ìn Table 8 of the Appendix E supports our conclusions in this subsection, thus showing the robustness of the proposed algorithms to varying environments.

5. Conclusion

Conditional volatility forecasting of a financial asset is important from the perspective of risk management, hedging, and portfolio management. Therefore, there are a plethora of models dealing with volatility forecasting, with the GARCH family of models being arguably the main approach. However, generic GARCH models are restrictive in nature with respect to the inclusion of exogenous variables that may contain important information about conditional volatility. Fortunately, log-GARCH models put very mild restrictions on inclusion of covariates in the conditional variance equation, and their estimation procedure is rather simple. However, this flexibility brings about a potential overfitting problem. To overcome this problem, we propose the VS-LTGARCHX algorithm to incorporate the variable selection procedure into the log-TGARCHX estimation process. The algorithm is very flexible in the sense that any variable selection procedure can be used within its framework. However, in this paper, we use LASSO, ABESS, and Boruta variable selection procedures separately within VS-LTGARCHX algorithm to select the subset of exogenous variables and asymmetry lags while estimating log-TGARCHX models. Furthermore, we apply the VS-LTGARCHX algorithm to extremely volatile BTC markets using 42 conditioning variables. Three different restricted versions of log-TGARHCX models obtained with the VS-LTGARCHX algorithm (using LASSO, ABESS, and Boruta selection procedures, respectively) outperform the benchmark log-GARCH (1,1) and log-TGARCHX (1,1) models in one-step ahead conditional volatility prediction based on different accuracy metrics. To assert the superiority of the VS-LTGARCHX variants, we use a formal procedure called MCS which indicates that the forecasts generated from the three variants of VS-LTGARCHX are superior to those from the benchmark models.

As a result, the main contributions of this paper are twofold. On the one hand, our study proposes a novel approach to model conditional volatility in the GARCH models domain which may be useful for academicians, investors, and policy-makers. On the other hand, we analyze volatility in the evolving and relatively young BTC market with 42 different exogenous variables. To the best of our knowledge, no study has been conducted to study BTC volatility with as many exogenous covariates as we do.

Ultimately, irrespective of the novelties of our paper, there are several limitations which paves the way for further research. First, the set of exogenous variables can be extended. Second, the selection procedures in this study cannot handle the simultaneous selection of the subset of ARCH, GARCH, asuymmetry terms and exogenous variables. Rather, our approach is based on the stepwise selection, which may be suboptimal due to possible correlation of variables. In addition, we have not considered ARCH/GARCH orders and asymmetry orders greater than 1, which presumably fettered further improvement of the predictive accuracy of VS-LTGARCHX.

A. Abbreviations

Table 4. Abbreviations for Blockchain Technology Variables

#	Abbreviation	Expansion
1	ATRCT	Bitcoin Median Transaction Confirmation Time in Minutes
2	AVBLS	Bitcoin Average Block Size
3	BLCHS	Bitcoin api.blockchain Size
4	CPTRA	Bitcoin Cost Per Transaction
5	CPTRV	Bitcoin Cost % of Transaction Volume
6	DIFF	Bitcoin Difficulty
7	ETRAV	Bitcoin Estimated Transaction Volume
8	ETRVU	Bitcoin Estimated Transaction Volume USD
9	HRATE	Bitcoin Hash Rate
10	MIREV	Bitcoin Miners Revenue
11	MKPRU	Bitcoin Market Price USD
12	MKTCP	Bitcoin Market Capitalization
13	MWNTD	Bitcoin My Wallet Number of Transaction Per Day
14	MWNUS	Bitcoin My Wallet Number of Users
15	MWTRV	Bitcoin My Wallet Transaction Volume
16	NADDU	Bitcoin Number of Unique Bitcoin Addresses Used
17	NTRAN	Bitcoin Number of Transactions
18	NTRAT	Bitcoin Total Number of Transactions
19	NTRBL	Bitcoin Number of Transaction per Block
20	NTREP	Bitcoin Number of Transactions Excluding Popular Addresses
21	TOTBC	Total Bitcoins
22	TOUTV	Bitcoin Total Output Volume
23	TRFEE	Bitcoin Total Transaction Fees
24	TRFUS	Bitcoin Total Transaction Fees USD
25	TRVOU	Bitcoin USD Exchange Trade Volume

Source: `www.nasdaq.com`

Table 5. Names and abbreviations of the exogenous variables within public opinion, risks and uncertainties, financials, and macroeconomic development classes.

Public Opinion[1]	**Risks and Uncertainties**	**Financials**[2]	**Macroeconomic Development**[3]
Number of tweets (tweet)	Geopolitical risks (GPRD)[4]	US Federal Funds Effective Rate (FFER)[5]	Crude Oil (CL=F)
Google trends (gtrends)	Daily China economic policy uncertainty Index (CNEPU)[6]	Ripple-USD Exchange Rate (XRP-USD)	Nasdaq (NDAQ)
		Bitcoin Futures price (BTC=F)	S&P500 (SPY)
		Chinese Yuan to dollar exchange rate (CNYUSD=X)	DOW30 (DJI)
			The Chicago Board Options Exchange Volatility Index (VIX)
			The Chicago Board Options Exchange Gold Volatility Index (GVZ)

[1] `www.bitinfocharts.com`

[2] `www.finance.yahoo.com`

[3] `www.finance.yahoo.com`

[4] `www.matteoiacoviello.com`

[5] `www.fred.stlouisfed.org`

[6] `www.economicpolicyuncertaintyinchina.weebly.com`

Table 6. Abbreviations for Realized Volatility Measures and Prefixes

#	Abbreviation	Expansion
1	rv_d	Daily realized variance[1]
2	rv_w	Weekly realized variance
3	rv_m	Monthly realized variance
4	d_	Prefix in variable names, denotes for the first difference (d_$X := X_t - X_{t-1}$)
5	dl_	Prefix in variable names, denotes the first logarithmic difference (dl_$X := \ln X_t - \ln X_{t-1}$)

[1] BTC price data were taken in 5-minute frequencies from Binance exchange to calculate the daily realized volatily.

B. Time series CV for LASSO or ABESS

The following algorithm outlines the methodology we follow for data partitioning to perform cross-validation for LASSO and ABESS.

Algorithm 1 Time Series CV For Variable Selection With LASSO or ABESS

Input: $\mathcal{D}_t = \{(z_t, \boldsymbol{x}_{t-1}, \boldsymbol{\sigma}_t | t = 1, ..., T\}$ where $\boldsymbol{\sigma_t}$ is the sequence of the realized standard deviation, $\boldsymbol{x}_{t-1}$ are regressors including the asymmetries, z_t are residuals within Eq. (9), Π is the parameter grid

function BESTSUBSETSELECTOR($\mathcal{D}_t$, Π)

Data Split: Training Sets (Ω_k) and Validation Sets (Υ_k), $\forall k = 1, 2, ..., \lfloor 0.2N \rfloor$, where cardinalities $\#(\Omega_1) = \lfloor 0.6N \rfloor = \tau_1$ and $\#(\Upsilon_1) = \lceil 0.2N \rceil = \tau_2$. Then, the collection of train and validation sets are:

$$
\begin{aligned}
\Omega_1 &:= \{\mathcal{D}_t | t = 1, 2, ..., \tau_1\} \\
\Upsilon_1 &:= \{\mathcal{D}_t | t = \tau_1 + 1\} \\
\Omega_2 &:= \{\mathcal{D}_t | t = 1, 2, ..., \tau_1 + 1\} \\
\Upsilon_2 &:= \{D_t | t = \tau_1 + 2\} \\
&\vdots \\
\Omega_{\tau_2} &:= \{\mathcal{D}_t | t = 1, 2, ..., \tau_1 + \tau_2 - 1\} \\
\Upsilon_{\tau_2} &:= \{\mathcal{D}_t | t = \tau_1 + \tau_2\}
\end{aligned}
$$

Define the set of parameters to be tuned with a search grid Π.
for each parameter set $\pi \in \Pi$ **do**
for each $(\Omega_k \subseteq \Omega, \Upsilon_k \subset \Upsilon), \quad \forall k = 1, ..., \tau_2$ **do**
Fit LASSO or ABESS
Generate Predictions for Υ_k using LASSO or ABESS, respectively
end for
Calculate average performance (RMSE) across Υ and $\forall \pi \in \Pi$.

$$
RMSE_\pi = \sqrt{\frac{1}{\tau_2} \sum_{t=\tau_1}^{\tau_1+\tau_2} (\boldsymbol{\sigma}_t - \hat{\boldsymbol{\sigma}}_t)^2}
$$

end for
Determine the optimal set of tuning parameters based on the lowest RMSE
Fit the ABESS or LASSO with respective optimal tuning parameters to $\Omega_{\tau_2} \cup \Upsilon_{\tau_2}$
Get The Best Subset of Covariates, $\mathbf{x}^*$
end function

Output: Selected subset of covariates ($\mathbf{x}^*$) using LASSO or ABESS.

C. Time series CV for random forest (Boruta)

The following algorithm describes the data partitioning technique to perform time series cross-validation for the random forest (Boruta).

Algorithm 2 Time Series CV For Random Forest Behind The Boruta

Input: $\mathcal{D}_t = \{(z_t, \boldsymbol{x}_{t-1}, \boldsymbol{\sigma}_t | t = 1, ..., T\}$ where $\boldsymbol{\sigma_t}$ is the sequence of the realized standard deviation, $\boldsymbol{x}_{t-1}$ are regressors that also include asymmetries, z_t are residuals within Eq. (9), number of trees (n_{tree}), minimum node size (n_{min}), block size (l_n), number of split variables (m), $\Pi := (n_{tree}, n_{min}, l_n, m)'$ is four-dimensional parameter grid

Data Split: Training Set $\mathcal{T} = \{\mathcal{D}_t | 1 < t < \lfloor 0.8T \rfloor\}$.
Define the set of parameters to be tuned with a search grid Π.
for $\pi \in \Pi$ **do**
  for each b = 1 to n_{tree} **do**
    Draw block bootstrap sample $Z_\star$ of size $\#(Z_\star) \leq \lfloor 0.8T \rfloor$ with parameter l_n.
    while node size $n > n_{min}$ **do**
      Chose a random set of m variables among the p variables and apply the CART criterion on this subset.
      Cut on the best split.
    end while
  end for
  Make out-of-bag (OOB) predictions $\hat{\boldsymbol{\sigma}}_t$
  Calculate average OOB performance (RMSE) $\forall \pi \in \Pi$.
end for
Determine the optimal set of tuning parameters based on the lowest RMSE
Apply Boruta with the optimal parameters
Get The Best Subset of Covariates, $\mathbf{x}^*$

Output: Selected subset of covariates ($\mathbf{x}^*$) by Boruta.

D. Evaluation of the predictive performance of VS-LTGARCHX variants and the benchmark models

The following algorithm summarizes the strategy we follow to evaluate the predictive performance of the different variants of the VS-LTGARCHX algorithm.

Algorithm 3 Evaluation of the Predictive Performance of VS-LTGARCHX variants and The Benchmark Models

Inputs: $\mathcal{D}_t = \{(r_t, \boldsymbol{x}^*_{t-1}, \boldsymbol{\sigma}_t) | t = 1, ..., T\}$, r_t are log-returns, x^*_{t-1} is deemed to be the selected variables from Algorithm 1 or 2 for VS-LTGARCHX variants, the full set of the variables at hand for the benchmark log-TGARCHX model, and an empty set for the benchmark log-GARCH(1,1).

1: **Data Split**: Define set of indices $\mathcal{S} = \{\tau : \lfloor 0.8T \rfloor \leq \tau \leq T\}$ for the test data, with cardinality $\#(\mathcal{S}) = \lceil 0.2T \rceil$
2: **for** $\tau \in \mathcal{S}$ **do**
3: Fit the intended model using $\mathcal{D}_\tau = \{\mathcal{D}_t | \tau - \lfloor 0.8T \rfloor + 1 \leq t \leq \tau\}$
4: Predict $\sigma_{\tau+1}$:

$$\mathbb{E}(\sigma_{\tau+1} | \mathcal{D}_\tau) = f(r_\tau, \boldsymbol{x}^*_{\tau-1})$$

5: Calculate the prediction errors, $\bar{d}_\tau = \sigma_\tau - \hat{\sigma_\tau}$
6: **end for**
7: Evaluate predictive performance using the following:

$$\begin{aligned} ME &= \frac{1}{\#(\mathcal{S})} \sum_{\tau \in \mathcal{S}} \bar{d}_\tau, \quad RMSE = \sqrt{\frac{1}{\#(\mathcal{S})} \sum_{\tau \in \mathcal{S}} \bar{d}^2_\tau} \\ MAE &= \frac{1}{\#(\mathcal{S})} \sum_{\tau \in \mathcal{S}} |\bar{d}_\tau|, \quad MAPE = \frac{1}{\#(\mathcal{S})} \sum_{\tau \in \mathcal{S}} \left| \frac{\bar{d}_\tau}{\sigma_\tau} \right| \\ MPE &= \frac{1}{\#(\mathcal{S})} \sum_{\tau \in \mathcal{S}} \frac{\bar{d}_\tau}{\sigma_\tau}, \quad QLIKE = \frac{1}{\#(\mathcal{S})} \sum_{\tau \in \mathcal{S}} (\ln \hat{\sigma_\tau}^2 + \frac{\sigma^2_\tau}{\hat{\sigma_\tau}^2}) \end{aligned} \tag{18}$$

Output: The table of predictive performance scores of the models considered.

E. Study of robustness

The table below (Table 7) is an extended version of Table 2. In Table 7, together with RMSE and QLIKE metrics, we report mean error (ME), Mean Absolute Error (MAE), Mean Absolute Percentage Error (MAPE) and Mean Percentage Error (MPE) for rolling window forecasts (see Eq. (18) and Hyndman and Athanasopoulos (2018) for details).

Table 7. Predictive accuracy of models in the test set.

	ME	RMSE	MAE	MPE	MAPE	QLIKE
Log-GARCH	1.20	2.00	1.70	24.78	36.04	3.74
Log-TGARCHX	**0.42**	1.85	1.39	**3.02**	36.66	3.82
LASSO	0.65	1.60	**1.23**	15.47	29.66	**3.64**
ABESS	0.67	**1.59**	1.24	16.11	**29.53**	**3.64**
Boruta	0.47	1.66	1.25	9.94	31.61	3.69

Although we are aware of the potential problems posed by the recursive forecasting scheme when evaluating nested models, we follow Hansen et al. (2011) and test our algorithm using the recursive scheme, also for the sake of robustness. Interestingly, our conclusions about the superiority of forecasts generated by the VS-LTGARCHX models over those generated by the benchmark models remain valid. Table 8 describes the performance metrics of the models in the test set (20% of the overall data). The training window is expanded by one observation at each forecast origin. The initial training set comprises 1314 (80% of the data) observations. The results of the MCS procedure are exactly the same as those in the rolling window forecasting scheme, and thus are omitted.

Table 8. Recursive window forecasts.

	ME	RMSE	MAE	MPE	MAPE	QLIKE
Log-GARCH	1.22	2.03	1.74	24.45	36.60	3.75
Log-TGARCHX	**0.07**	1.78	1.31	**7.91**	40.34	3.93
LASSO	0.59	1.62	1.24	14.07	30.29	**3.65**
ABESS	0.61	**1.61**	1.24	14.66	**30.02**	**3.65**
Boruta	0.45	1.64	**1.22**	9.65	31.15	3.68

6. Competing interests

No competing interest is declared.

7. Author contributions statement

F.R. wrote a Python program to download data and reviewed the literature, the rest of the work was divided and completed equally by F.S., A.P., V.E and S.O.

8. Data availability statement

The data and code underlying this article are available in `https://github.com/salahaddiniayyubi/Log-TGARCHX-Subset-Selection.git`.

9. Acknowledgements

Samir Orujov gratefully acknowledges financial support from Total Energies and ADA University, Azerbaijan.

References

D. Y. Aharon and M. Qadan. Bitcoin and the day-of-the-week effect. *Finance Research Letters*, 31, Dec. 2019. ISSN 1544-6123. doi: 10.1016/j.frl.2018.12.004. URL `https://www.sciencedirect.com/science/article/pii/S1544612317307894`.

A. Arratia and A. X. López-Barrantes. Do Google Trends forecast bitcoins? Stylized facts and statistical evidence. *Journal of Banking and Financial Technology*, Mar. 2021. ISSN 2524-7956, 2524-7964. doi: 10.1007/s42786-021-00027-4. URL `http://link.springer.com/10.1007/s42786-021-00027-4`.

N. Aslanidis, A. F. Bariviera, and Ó . G. López. The link between cryptocurrencies and Google Trends attention. *Finance Research Letters*, 47:102654, June 2022. ISSN 1544-6123. doi: 10.1016/j.frl.2021.102654. URL `https://www.sciencedirect.com/science/article/pii/S1544612321005833`.

D. Bakas, G. Magkonis, and E. Y. Oh. What drives volatility in Bitcoin market? *Finance Research Letters*, 50:103237, Dec. 2022. ISSN 1544-6123. doi: 10.1016/j.frl.2022.103237. URL https://www.sciencedirect.com/science/article/pii/S1544612322004378.

L. Ø. Bergsli, A. F. Lind, P. Molnár, and M. Polasik. Forecasting volatility of Bitcoin. *Research in International Business and Finance*, 59:101540, Jan. 2022. ISSN 0275-5319. doi: 10.1016/j.ribaf.2021.101540. URL https://www.sciencedirect.com/science/article/pii/S0275531921001616.

M. Bernardi and L. Catania. The Model Confidence Set package for R. *CEIS Research Paper*, Nov. 2015. URL https://ideas.repec.org//p/rtv/ceisrp/362.html. Number: 362 Publisher: Tor Vergata University, CEIS.

T. Bollerslev. Generalized autoregressive conditional heteroskedasticity. *Journal of Econometrics*, 31(3):307–327, Apr. 1986. ISSN 0304-4076. doi: 10.1016/0304-4076(86)90063-1. URL https://www.sciencedirect.com/science/article/pii/0304407686900631.

L. Breiman. Bagging predictors. *Machine Learning*, 24(2):123–140, Aug. 1996. ISSN 1573-0565. doi: 10.1007/BF00058655. URL https://doi.org/10.1007/BF00058655.

L. Breiman. Random Forests. *Machine Learning*, 45(1):5–32, Oct. 2001. ISSN 1573-0565. doi: 10.1023/A:1010933404324. URL https://doi.org/10.1023/A:1010933404324.

P. J. Brockwell and R. A. Davis. Estimation for ARMA Models. In P. J. Brockwell and R. A. Davis, editors, *Time Series: Theory and Methods*, Springer Series in Statistics, pages 238–272. Springer, New York, NY, 1991. ISBN 978-1-4419-0320-4. doi: 10.1007/978-1-4419-0320-4_8. URL https://doi.org/10.1007/978-1-4419-0320-4_8.

C. Brooks. *Introductory Econometrics for Finance*. Cambridge University Press, Cambridge, 2 edition, 2008. doi: 10.1017/CBO9780511841644. URL https://www.cambridge.org/core/books/introductory-econometrics-for-finance/4F3AB9473A63F11982D6902D813BC521.

H. Bystrom and D. Krygier. What Drives Bitcoin Volatility?, July 2018. URL https://papers.ssrn.com/abstract=3223368.

W. Chen, H. Xu, L. Jia, and Y. Gao. Machine learning model for Bitcoin exchange rate prediction using economic and technology determinants. *International Journal of Forecasting*, 37(1):28–43, Jan. 2021. ISSN 0169-2070. doi: 10.1016/j.ijforecast.2020.02.008. URL https://www.sciencedirect.com/science/article/pii/S0169207020300431.

D. Dickey and W. Fuller. Distribution of the Estimators for Autoregressive Time Series With a Unit Root. *JASA. Journal of the American Statistical Association*, 74, June 1979. doi: 10.2307/2286348.

G. Elliott and A. Timmermann. Economic Forecasting. *Journal of Economic Literature*, 46(1):3–56, Feb. 2008. ISSN 0022-0515. doi: 10.1257/jel.46.1.3. URL https://pubs.aeaweb.org/doi/10.1257/jel.46.1.3.

R. F. Engle. Autoregressive Conditional Heteroscedasticity with Estimates of the Variance of United Kingdom Inflation. *Econometrica*, 50(4):987–1007, 1982. ISSN 0012-9682. doi: 10.2307/1912773. URL https://www.jstor.org/stable/1912773. Publisher: [Wiley, Econometric Society].

A. Escribano and G. Sucarrat. Equation-by-equation estimation of multivariate periodic electricity price volatility. *Energy Economics*, 74:287–298, Aug. 2018. ISSN 0140-9883. doi: 10.1016/j.eneco.2018.05.017. URL https://www.sciencedirect.com/science/article/pii/S0140988318301841.

C. Francq and L. Q. Thieu. QML INFERENCE FOR VOLATILITY MODELS WITH COVARIATES. *Econometric Theory*, 35(1):37–72, Feb. 2019. ISSN 0266-4666, 1469-4360. doi: 10.1017/S0266466617000512. URL https://www.cambridge.org/core/product/identifier/S0266466617000512/type/journal_article.

C. Francq, O. Wintenberger, and J.-M. Zakoïan. GARCH models without positivity constraints: Exponential or log GARCH? *Journal of Econometrics*, 177(1):34–46, Nov. 2013. ISSN 0304-4076. doi: 10.1016/j.jeconom.2013.05.004. URL https://www.sciencedirect.com/science/article/pii/S0304407613001267.

J. Friedman, R. Tibshirani, and T. Hastie. Regularization paths for generalized linear models via coordinate descent. *Journal of Statistical Software*, 33(1):1–22, 2010. doi: 10.18637/jss.v033.i01.

J. Geweke. Commet. *Econometric Reviews*, 5(1):57–61, Jan. 1986. ISSN 0747-4938. doi: 10.1080/07474938608800097. URL https://doi.org/10.1080/07474938608800097. Publisher: Taylor & Francis _eprint: https://doi.org/10.1080/07474938608800097.

L. R. Glosten, R. Jagannathan, and D. E. Runkle. On the Relation between the Expected Value and the Volatility of the Nominal Excess Return on Stocks. *The Journal of Finance*, 48(5):1779–1801, Dec. 1993. ISSN 00221082. doi: 10.1111/j.1540-6261.1993.tb05128.x. URL https://onlinelibrary.wiley.com/doi/10.1111/j.1540-6261.1993.tb05128.x.

W. Greene. *Econometric Analysis*. Pearson, New York, NY, 8th edition edition, Mar. 2017. ISBN 978-0-13-446136-6.

J. D. Hamilton. *Time series analysis*. Princeton university press, 1994.

P. R. Hansen, A. Lunde, and J. M. Nason. The Model Confidence Set. *Econometrica*, 79(2):453–497, 2011. ISSN 0012-9682. URL https://www.jstor.org/stable/41057463. Publisher: [Wiley, Econometric Society].

T. Hastie, R. Tibshirani, and J. H. Friedman. *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. Springer Science & Business Media, 2001. ISBN 978-0-387-95284-0. Google-Books-ID: VRzITwgNV2UC.

R. J. Hyndman and G. Athanasopoulos. *Forecasting: principles and practice*. OTexts, Lexington, Ky., 2nd edition edition, May 2018. ISBN 978-0-9875071-1-2.

H. R. Kunsch. The Jackknife and the Bootstrap for General Stationary Observations. *The Annals of Statistics*, 17(3):1217–1241, Sept. 1989. ISSN 0090-5364, 2168-8966. doi: 10.1214/aos/1176347265. URL https://projecteuclid.org/journals/annals-of-statistics/volume-17/issue-3/The-Jackknife-and-the-Bootstrap-for-General-Stationary-Observations/10.1214/aos/1176347265.full. Publisher: Institute of Mathematical Statistics.

M. B. Kursa and W. R. Rudnicki. Feature Selection with the Boruta Package. *Journal of Statistical Software*, 36:1–13, Sept. 2010. ISSN 1548-7660. doi: 10.18637/jss.v036.i11. URL https://doi.org/10.18637/jss.v036.i11.

D. Kwiatkowski, P. C. B. Phillips, P. Schmidt, and Y. Shin. Testing the null hypothesis of stationarity against the alternative of a unit root: How sure are we that economic time series have a unit root? *Journal of Econometrics*, 54(1):159–178, Oct. 1992. ISSN 0304-4076. doi: 10.1016/0304-4076(92)90104-Y. URL https://www.sciencedirect.com/science/article/pii/030440769290104Y.

M. Á. López-Cabarcos, A. M. Pérez-Pico, J. Piñeiro-Chousa, and A. Šević. Bitcoin volatility, stock market and investor sentiment. Are they connected? *Finance Research Letters*, 38:101399, Jan. 2021. ISSN 15446123. doi: 10.1016/j.frl.2019.101399. URL https://linkinghub.elsevier.com/retrieve/pii/S1544612319309274.

A. Milhoj. A Conditional Variance Model for Daily Deviations of an Exchange Rate. *Journal of Business & Economic Statistics*, 5(1): 99–103, 1987. URL https://ideas.repec.org//a/bes/jnlbes/v5y1987i1p99-103.html. Publisher: American Statistical Association.

A. Miller. *Subset Selection in Regression*. Chapman and Hall/CRC, New York, 2 edition, Apr. 2002. ISBN 978-0-429-11918-7. doi: 10.1201/9781420035933.

D. B. Nelson. Conditional Heteroskedasticity in Asset Returns: A New Approach. *Econometrica*, 59(2):347–370, 1991. ISSN 0012-9682. doi: 10.2307/2938260. URL http://www.jstor.org/stable/2938260. Publisher: [Wiley, Econometric Society].

S. G. Pantula. Comment. *Econometric Reviews*, 5(1):71–74, Jan. 1986. ISSN 0747-4938, 1532-4168. doi: 10.1080/07474938608800099. URL http://www.tandfonline.com/doi/abs/10.1080/07474938608800099.

A. J. Patton. Volatility forecast comparison using imperfect volatility proxies. *Journal of Econometrics*, 160(1):246–256, Jan. 2011. ISSN 0304-4076. doi: 10.1016/j.jeconom.2010.03.034. URL https://www.sciencedirect.com/science/article/pii/S030440761000076X.

R. S. Pedersen and A. Rahbek. TESTING GARCH-X TYPE MODELS. *Econometric Theory*, 35(05):1012–1047, Oct. 2019. ISSN 0266-4666, 1469-4360. doi: 10.1017/S026646661800035X. URL https://www.cambridge.org/core/product/identifier/S026646661800035X/type/journal_article.

G. Sucarrat. The log-GARCH model via ARMA representations. In *Financial Mathematics, Volatility and Covariance Modelling*. Routledge, 2019. ISBN 978-1-315-16273-7. Num Pages: 24.

G. Sucarrat. garchx: Flexible and Robust GARCH-X Modelling. *MPRA Paper*, May 2020. URL https://ideas.repec.org//p/pra/mprapa/100301.html. Number: 100301 Publisher: University Library of Munich, Germany.

G. Sucarrat, S. Grønneberg, and A. Escribano. Estimation and inference in univariate and multivariate log-GARCH-X models when the conditional density is unknown. *Computational Statistics & Data Analysis*, 100:582–594, Aug. 2016. ISSN 01679473. doi: 10.1016/j.csda.2015.12.005. URL https://linkinghub.elsevier.com/retrieve/pii/S0167947315003059.

R. Tibshirani. Regression Shrinkage and Selection via the Lasso. *Journal of the Royal Statistical Society. Series B (Methodological)*, 58 (1):267–288, 1996. ISSN 0035-9246. URL https://www.jstor.org/stable/2346178. Publisher: [Royal Statistical Society, Wiley].

F. Violante and S. Laurent. Volatility Forecasts Evaluation and Comparison. In *Handbook of Volatility Models and Their Applications*, pages 465–486. John Wiley & Sons, Ltd, 2012. ISBN 978-1-118-27203-9. doi: 10.1002/9781118272039.ch19. URL https://onlinelibrary.wiley.com/doi/abs/10.1002/9781118272039.ch19. Section: Nineteen _eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/9781118272039.ch19.

J. M. Wooldridge. *Introductory Econometrics: A Modern Approach*. Cengage Learning, Sept. 2015. ISBN 978-1-305-44638-0. Google-Books-ID: wUF4BwAAQBAJ.

L. R. Y. Moving blocks jackknife and bootstrap capture weak dependence. *Exploring the Limits of Bootstrap*, 1992. URL https://cir.nii.ac.jp/crid/1573668924689635584. Publisher: Wiley.

J. Zhu, C. Wen, J. Zhu, H. Zhang, and X. Wang. A polynomial algorithm for best-subset selection problem. *Proceedings of the National Academy of Sciences*, 117(52):33117–33123, Dec. 2020. doi: 10.1073/pnas.2014241117. URL https://www.pnas.org/doi/10.1073/pnas.2014241117. Publisher: Proceedings of the National Academy of Sciences.