

Article

Applying Hybrid ARIMA-SGARCH in Algorithmic Investment Strategies on S&P500 Index

Nguyen Vo ¹ and Robert Ślepaczuk ^{2,*} 

¹ Quantitative Finance Research Group, Faculty of Economic Sciences, University of Warsaw, Ul. Długa 44/50, 00-241 Warsaw, Poland; ainguyen.vo92@gmail.com

² Quantitative Finance Research Group, Department of Quantitative Finance, Faculty of Economic Sciences, University of Warsaw, Ul. Długa 44/50, 00-241 Warsaw, Poland

* Correspondence: rslepaczuk@wne.uw.edu.pl

Abstract: This research aims to compare the performance of ARIMA as a linear model with that of the combination of ARIMA and GARCH family models to forecast S&P500 log returns in order to construct algorithmic investment strategies on this index. We used the data collected from Yahoo Finance with daily frequency for the period from 1 January 2000 to 31 December 2019. By using a rolling window approach, we compared ARIMA with the hybrid models to examine whether hybrid ARIMA-SGARCH and ARIMA-EGARCH can really reflect the specific time-series characteristics and have better predictive power than the simple ARIMA model. In order to assess the precision and quality of these models in forecasting, we compared their equity lines, their forecasting error metrics (MAE, MAPE, RMSE, MAPE), and their performance metrics (annualized return compounded, annualized standard deviation, maximum drawdown, information ratio, and adjusted information ratio). The main contribution of this research is to show that the hybrid models outperform ARIMA and the benchmark (Buy&Hold strategy on S&P500 index) over the long term. These results are not sensitive to varying window sizes, the type of distribution, and the type of the GARCH model.



Citation: Vo, N.; Ślepaczuk, R. Applying Hybrid ARIMA-SGARCH in Algorithmic Investment Strategies on S&P500 Index. *Entropy* **2022**, *24*, 158. <https://doi.org/10.3390/e24020158>

Academic Editor: Vasily E. Tarasov

Received: 20 December 2021

Accepted: 17 January 2022

Published: 20 January 2022

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Copyright: © 2022 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

Over the past few decades, time series forecasting in finance has been an interesting and important research area. It has attracted the attention of not only the researcher community but also investors, speculators, and governments who are interested in verification of various models and approaches in predicting of future prices of various types of assets (Sakowski and Turovtseva, 2020 [1], Torre-Torres et al. (2021) [2]) with the use of more standard tools (Goubuzaitė and Teresiene, 2021 [3], Ivanyuk, 2021 [4]) or the newest ML techniques (Chlebus et al., 2021) [5]. The main aim of time series modeling is to carefully measure and analyze the historical observations of the time series in order to develop the most appropriate models. The most important function of these models is to forecast future values of the time series, i.e., to predict the movements, behaviors, and changes, usually by reflecting the characteristics of the historical observations. It is obvious that to obtain adequately low forecasting errors, a proper process for model fitting needs to be taken into consideration. Along with the development of more advanced forecasting techniques, a lot of effort has been put into improving forecasting accuracy by choosing, testing, and fitting more efficient models. As a result, various important theories and assumptions about modeling forecast have evolved. In particular, the analyses of the time series of an essential US stock index, such as the S&P500 have never failed to get attention and efforts from those interested in quantitative finance.

One of the most popular and frequently used time-series models is the autoregressive integrated moving average (ARIMA). In this model, there is a linear relationship between

past observation values (autoregressive) and random errors (moving average), in which random errors are assumed to be independent and identically distributed (i.i.d.) with a mean of zero and a constant variance σ_e^2 over time. It is not so surprising that financial time series often do not follow this assumption, and the S&P500 index is no exception. Its returns can be extremely volatile during booms and busts. This means that the existence of volatility clustering in time series can affect the forecasting performance of the mean models like ARIMA. Therefore, most researchers have started to use symmetric generalized autoregressive conditional heteroscedasticity (SGARCH) and its family models when modeling volatility in order to obtain accurate forecasts. This is the reason why, in our study, we evaluate and compare the forecasting performance among the ARIMA model, the hybrid ARIMA-SGARCH and ARIMA-EGARCH (EGARCH-Exponential Generalized Autoregressive Conditional Heteroscedasticity is a modified version of GARCH).

In that context, the paper addresses one main hypothesis (RH): whether the ARIMA(p,1,q)-SGARCH(1,1) (hybrid model) with window size $s = 1000$ can generate a trading strategy that outperforms ARIMA(p,1,q). In other words, the key objective of this study is to verify the efficiency of forecasts from hybrid and simple ARIMA models in algorithmic trading strategies.

Based on this hypothesis, a few research questions are constructed:

- RQ1. Is the result obtained from the main test robust to varying family of GARCH models?
- RQ2. Does the hybrid model with EGARCH outperform the one with SGARCH?
- RQ3. Is the result obtained from the main test robust to varying window sizes?
- RQ4. Is the hybrid model sensitive to different window sizes?
- RQ5. Is the result obtained from the main test robust to varying distributions?
- RQ6. Is the hybrid model sensitive to different distributions?

In order to verify the main hypothesis and answer the research questions mentioned above, empirical research was conducted based on the dataset of the S&P500 index. This study includes several novelties in comparison to other papers focused on similar topics. The data were collected on a daily basis over the period from 1 January 2000 to 31 December 2019. Firstly, we conducted a rolling forecast based on the ARIMA model with window size $s = 1000$, while the most common approach and not necessarily the best one was based on the simple division of the whole data sample into one in-sample and one out-of-sample set (usually 70/80% to 30/20%). The optimized combination of p and q which had the lowest Akaike information criterion (AIC) was used to predict the return for the next day. For the purpose of the out-of-sample results, the vector of forecasted values had a length of 3530 elements (starting on 20 December 2005). Secondly, we describe and review our implementation of ARIMA(p,1,q)-SGARCH(1,1) models with generalized error distribution (GED) and window size equal to 1000, in which optimized ARIMA(p,1,q) is taken from the first step. Thirdly, we evaluate the performance of SGARCH with different window sizes as well as various distributions to check the sensitivity of the results obtained in the main test. EGARCH, known as another family of GARCH models, was also applied in the sensitivity analysis in order to check the robustness of our initial assumptions. Therefore, we propose a more thorough approach, checking various types of GARCH models and various assumptions concerning the error distributions. In order to examine the precision and the quality of these models in predicting and their efficiency in algorithmic investment strategies, we compare their equity lines, their error metrics (mean absolute error (MAE), mean absolute percentage error (MAPE), root mean squared error (RMSE)) and their performance metrics: annualized return compounded (ARC), annualized standard deviation (ASD), maximum drawdown (MD), information ratio (IR), and adjusted information ratio (AIR). Therefore, the precision of forecasts is verified in the two-step procedure which combines the evaluation of econometric model forecasts with standard error metrics and the evaluation of investment signals constructed on these forecasts with the help of performance metrics calculated based on final equity lines. We expect that hybrid models can help build more efficient algorithmic investment strategies which outperform ARIMA.

The paper is structured as follows: Section 2 provides a literature review and an overview of ARIMA, SGARCH, EGARCH, and the hybrid models. Section 3 presents all the details concerning the dataset and explains the methodology of ARIMA and the hybrid models. In addition, it gives a full description of trading strategy construction and the technique of generating buy/sell signals for the investment strategy. Section 4 discusses the empirical results and conducts some sensitivity analysis to determine whether the result is robust to initial assumptions. The Section 5 draws conclusions and makes suggestions for future works.

2. Literature Review

According to Strickland [6], there are various ways to classify methods for time series analysis. Some examples are frequency-domain and time-domain and parametric and non-parametric. The category used in this study is linear and non-linear regression. Regression analysis is a process of estimating the relationship between a dependent variable and one (or more than one) independent one(s). If there is only a single independent variable, this is known as simple linear regression; otherwise, it is called multiple regression. Both of these models assume that the dependent variable is continuous.

2.1. Linear Forecasting Models—Autoregressive Integrated Moving Average Models

Box and Jenkins [7] first introduced Autoregressive Integrated Moving Average Models (ARIMA) in 1976. This model describes the linear relationship between past observation values and random errors (also known as shocks or disturbances). In order to estimate the ARIMA model correctly, we must identify and remove non-stationarity through differencing; hence, the differences between a value and its lagged values ($y_t - Ly_t$) need to be calculated. The ARIMA model can be regarded as an extension of the ARMA model.

There are many related studies in modelling and forecasting stock prices using ARIMA models. For instance, Ariyo et al. [8] revealed an extensive process describing how to obtain the most appropriate ARIMA model to anticipate stock prices (based on the smallest value of Schwarz Information Criterion (SIC)). Later, Kamruzzaman et al. [9] calculated returns by using the relative difference method and chose the ARIMA(2,1,2) model (based on the smallest value of AIC) as the most superior model for forecasting the stock market returns of the Dhaka Stock Exchange in Bangladesh. In addition, Abbasi et al. [10] applied this linear process into the flying cement industry and suggested that ARIMA(1,2,1) was a parsimonious model for forecasting cement stock prices in their case study.

2.2. Non-Linear Forecasting Models

2.2.1. Autoregressive Conditional Heteroskedasticity—ARCH(q)

This model was first proposed by Engle [11] to predict the conditional variance of return series. Despite the key strength as a simple model which produces volatility estimates with positive excess kurtosis (It means fat tails relative to the normal distribution which is in line with empirical observations about returns.), its weaknesses should also be taken into consideration. Firstly, due to the possible large value of the lag q, it could lead to a large number of parameters to be estimated. Hence it may result in difficulties to determine parameters [12].

Secondly, as it is well known in practice, the stock prices or financial assets in general react differently to positive and negative shocks. However, ARCH models assume these kinds of shock have the same effects on the volatility as they depend on the square of the previous shocks. As a result, this weakness should be taken into consideration in forecasting when applying ARCH models [13]. Additionally, since ARCH models respond slowly to large shocks, they are likely to overpredict the volatility [14]. Furthermore, the ARCH models do not provide any new insights to understand the source of volatility of financial time series. They merely provide a mechanical way to describe the behavior of the conditional variance and give no indication of what causes such behavior to occur [15].

2.2.2. Generalized Autoregressive Conditional Heteroscedasticity—GARCH(p,q)

The GARCH model is considered an extension of the ARCH model and was proposed by Bollerslev [16] by developing the symmetric GARCH (SGARCH). As demonstrated by many researchers and studies, the SGARCH (1,1) process is able to represent the majority of the time series [17]. The dataset which requires a model of higher orders like SGARCH(1,2) or SGARCH(2,1) is very rare [11]. However, financial time series inherits many characteristics that SGARCH is not able to incorporate well. Therefore, extensive generalizations with further features have been put forward in the literature.

One of the most essential properties of volatility that should be taken into consideration is the leverage effect, which describes the fact that there is a difference in reaction of volatility between notable price rises and notable price falls. This has led to the introduction of asymmetric GARCH models by purely adjusting the error term in the variance equation with a parameter to be responsible for this effect. These models were initially proposed by Engle et al. [18]. Nowadays, there are a variety of such models, including that by Higgins and Bera [19], in which a nonlinear asymmetric GARCH (N-GARCH) accounts for the leverage effect. Later, Glosten et al. [20] introduced the Glosten-Jagannathan-Runkle GARCH (GJR-GARCH) to precisely build up the volatility response from negative market shocks with an indicator function, while the quadratic ARCH (Q-ARCH) launched by Sentana [21] established asymmetric effects of both positive and negative shocks. The most recent model that allows for an asymmetric response due to the leverage effects is the exponential GARCH (EGARCH), which is discussed in the next Section.

2.2.3. The Conditional Variance Equation: Exponential GARCH

The EGARCH, presented by Nelson [22], is supposed to avoid imposing constraints on the coefficients by specifying the logarithm of the conditional volatility. In reality, “bad news” typically has a larger impact on volatility than “good news”. In other words, by applying EGARCH, we can mitigate the disadvantage of using GARCH in which negative innovations tend to increase the volatility more than positive innovations with the same magnitude. EGARCH is applied in this paper in the section concerning the comparison of its performance with SGARCH’s.

2.2.4. Underlying Return Distributions

GARCH models assume that the distribution of returns is normally distributed. However, this assumption has been proved in empirical financial markets to be inaccurate. This is because the distribution of financial returns tends to be leptokurtic [23]. It means that the tails are heavier in comparison with normal distribution. As a consequence, several fat tail distributions are applied in order to overcome this shortcoming. For instance, the Student-t was introduced by Bollerslev [24], the generalized error distribution (GED) by Nelson [22], and their skewed versions, which are all for leptokurtic distribution analysis.

2.3. The Hybrid ARIMA-GARCH

The class of ARIMA models with ARCH errors was proposed initially by Weiss [25]. The techniques were applied to U.S. macroeconomic time series. This approach was later adopted and extended by many researchers for modelling time series in various fields (e.g., Jabłecki et al. [26], Catellano and Ślepaczuk (2021) [27], Hauser and Kunst [28], Kijewski and Ślepaczuk [29]).

Alongside the theory development of the hybrid models in forecasting economic time series, Yaziz et al. [30] analyzed the performance of ARIMA-GARCH in forecasting gold price. The empirical results of 40-day gold price data series indicate that the hybrid ARIMA(1,1,1)-GARCH(0,2) model provides superior results and effectively improves evaluating and predicting precision in comparison to the linear models. Later, Sun [31] proposed the hybrid models to model and predict the equity returns for three US benchmark indices: Dow Transportation, S&P500, and VIX. The observed results suggested hybrid models are appropriate for anticipating the equity returns but have not been explored in the previous

works. The latest work discussed in this Section is by Ismail and Mustapa [32]. They presented the assessment in building the best fitted ARIMA-GARCH model to generate predicting values of the S&P500 stock prices. The ARIMA(2,1,2)-GARCH(1,1) model was determined to be the most appropriate model for forecasting stock prices.

Collectively, the presented papers reveal that heteroskedasticity can affect the validity or power of statistical tests when using ARIMA models, and the ARCH effect should be considered. Furthermore, the mentioned studies also indicate that the combination of ARIMA and the family of GARCH should be expected to perform well in modelling financial time series. In our study, we fit an optimal ARIMA-SGARCH as well as ARIMA-EGARCH in different window sizes and various return distributions in algorithmic investment strategies on the S&P500 index. It is based on the AIC, each day, using the rolling window approach.

3. Methodology and Data

In this Section, we provide a complete data analysis and a model-fitting procedure for the logarithmic returns of the S&P500 index. Additionally, this Section covers the description of methodology.

3.1. Data Analysis

3.1.1. Data Fetching and Preprocessing of Historical Data

We fetched the historical data from Yahoo Finance. Given the hybrid ARIMA-GARCH model proposed in Section 1, the data for the S&P500 index were collected with the period of 19 years for sufficiently reliable model fitting and forecasting purposes.

The first step in the process was cleaning the data. Then, we transformed the adjusted price into a daily logarithmic return, which was calculated according to the following formula:

$$r_t = \ln\left(\frac{P_t}{P_{t-1}}\right) \quad (1)$$

There are many reasons for choosing log returns instead of normal prices as a variable to forecast in this study. First, they can be added across time periods in order to create cumulative returns. Second, it is easy to convert between log return and simple return. Last but not least, log return follows normal distribution. Why is it an important advantage? This kind of distribution is solely dependent on the mean and the standard deviation of the sample. Based on these characteristics, any variable that exhibits normal distribution is feasible to be forecasted with higher accuracy, and our variable, in this case, is log return. Moreover, stock prices cannot be modelled by normal distribution because they have a negative side, and stock prices cannot fall below zero. In other words, prices are log-normally distributed, then the logarithm of each price will have a normal distribution. These relationships can be expressed by the equation below:

$$\ln\left(\frac{P_t}{P_0}\right) = \ln(P_t) - \ln(P_0) = \ln(1 + r_t) \quad (2)$$

3.1.2. Descriptive Statistics

Table 1 presents the descriptive statistics of the adjusted closing prices and log returns of the S&P500 index for the whole dataset.

As can be seen from Figure 1, there are a few periods, such as 2008, 2011, 2015, and 2018, that show high volatility of returns. Therefore, we can expect to build more accurate forecasting models if we are able to mitigate and “smooth” such periods. This is further explained in the Methodology section.

Table 1. Descriptive Statistics for S&P500 (Jan 2000–Dec 2019).

| Descript Statistics | S&P500 Original Prices | Log Returns |
|--------------------------|------------------------|-------------|
| Min | 676.5300 | -0.0947 |
| 1st Quantile | 1151.5349 | -0.0047 |
| Median | 1360.9550 | 0.0005 |
| Arithmetic Mean | 1574.6801 | 0.0002 |
| 3rd Quantile | 1986.2225 | 0.0057 |
| Max | 3240.0200 | 0.1096 |
| Skew | 0.9886 | -0.2295 |
| Kurtosis | -0.0715 | 8.6448 |
| Standard Error Mean (se) | 8.2559 | 0.0002 |
| Standard Deviation (sd) | 585.5315 | 0.0119 |

Note: The table covers the period between 1 January 2000 and 31 December 2019.

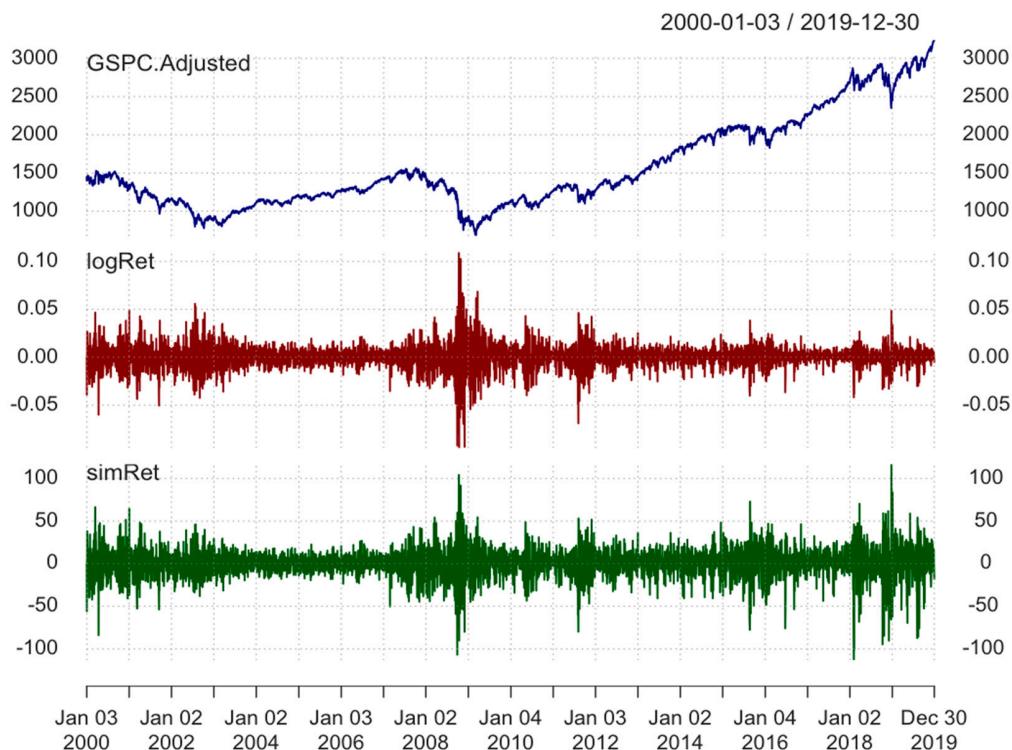


Figure 1. S&P500 index prices with its first differences and log returns. The fluctuations of S&P500 index prices, its first differences and log returns of S&P500 index prices in the period between 1 January 2000 and 31 December 2019.

Next, when we consider the central tendency, Table 1 shows two types of estimation as mean and median. The central tendency of a distribution is an estimation of the “center” of distribution, in this case of stock prices and log returns. If mean (or average) is computed by added up all the values and divided by the number of values, median, on the other hand, is the middle value or midpoint in data (also known as 50th percentile). In a normal distribution, these two metrics fall at the same midline point. In other words, mean and median are equal. In this study, with mean of 1574.6801 and median of 1360.9550, our initial assessment about stock prices of S&P500 is that they are not normally distributed. This it makes sense since normal distribution has 2 sides, while stock prices cannot be negative (below zero).

Figure 2 shows that log return series, on the other hand, has different characteristics of distribution in comparison to stock prices dataset. Intuitively, log returns are normally distributed because mean and median values are close to each other (0.0002 and 0.0005, respectively). Furthermore, values of first and third quantile (-0.0047 and 0.0057, re-

spectively) as well as min and max values (-0.0947 and 0.1096 , respectively) are quite symmetric. It is the main reason why we use log returns to build models.

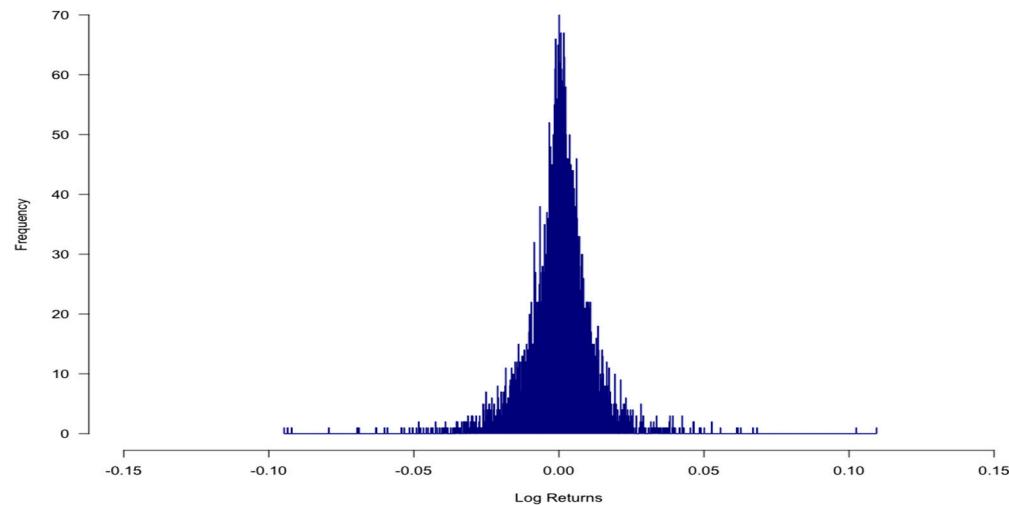


Figure 2. Histogram for log returns–S&P500. The histogram of S&P500 index log returns in the period between 1 January 2000 and 31 December 2019.

However, the kurtosis value of 8.6448 is larger than 3 (hence it is named leptokurtic) and skew value of -0.2295 (near but below zero), so we could say the log returns series is similar to double exponential distribution (<https://www.itl.nist.gov/div898/handbook/eda/section3/eda35b.htm> accessed on 15 July 2021). This kind of distribution is symmetric, but compared to the normal one, it has a stronger (higher and sharper) peak, more rapid decay, and heavier tails. Furthermore, looking at the histograms in Figure 3, it is perfectly clear that they show certain similarities to the normal distribution.

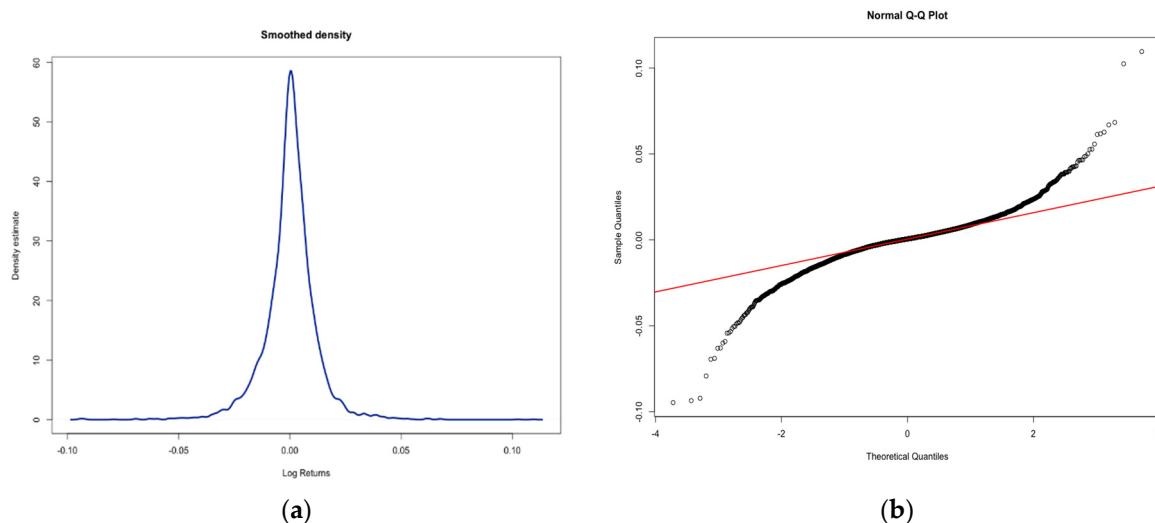


Figure 3. Smoothed density and Q-Q plot for log returns–S&P500. Smoothed density (a) and Q-Q plot (b) for S&P500 index log returns in the period between 1 January 2000 and 31 December 2019.

Finally, when taking dispersion into consideration, if standard error (SE) of S&P500 prices is far away from zero (8.2559), log return series' one, on the other hand, is quite close to zero (0.0002). SE can be approximated by the following formula:

$$SE = \frac{\sigma}{\sqrt{n}} \quad (3)$$

where: σ is standard deviation and n is the number of observations (sample size).

Equation (3) informs us that the larger the sample size (more data points) involved in the calculation, the smaller the SE tends to be. That is, if SE is small, the data are said to be more representative of the true mean. So, with the value of 8.2559, SE of S&P500 prices shows that data may have some notable irregularities as the sample is less accurate (due to high value of SE). Obviously, with SE of 0.0002 (≈ 0), log return dataset can be expected to build more accurate models.

3.2. Methodology

3.2.1. Fundamental Concepts and Definitions

Autoregressive Moving Average Models—ARMA(p,q)

The ARMA process is the combination of the autoregressive model and moving average [2] designed for a stationary time series. Autoregression (AR) describes a stochastic process, and AR(p) can be denoted as shown below:

$$\text{AR}(1) : y_t = \phi y_{t-1} + \varepsilon_t$$

$$\text{AR}(p) : y_t = \phi_1 y_{t-1} + \phi_2 y_{t-2} + \dots + \phi_p y_{t-p} + \varepsilon_t \quad (4)$$

where: ϕ_p denotes the weights given to past observations at each lag, p is a positive integer providing the number of lags to be included, and ε_t is white noise.

We now introduce the lag, i.e., in the interest of notational convenience, which simply produces the previous element of the series, as shown below:

$$Ly_t = y_{t-1} \quad (5)$$

So, an AR (p), using lag notation, is now:

$$\left(1 - \sum_{i=1}^p \phi_i L^i\right) y_t = c + \varepsilon_t, \quad p = 1, 2, \dots \quad (6)$$

The moving average process of order q is denoted as MA(q) and the created time series contains a mean of q lagged white noise variables shifting along the series.

$$\text{MA}(1) : y_t = \mu + \varepsilon_t + \theta_1 \varepsilon_{t-1} \quad \text{MA}(q) : y_t = \mu + \varepsilon_t + \theta_1 \varepsilon_{t-1} + \dots + \theta_q \varepsilon_{t-q} \quad (7)$$

where: μ is the mean of the series, and θ_q are the weights given to each white noise value. Using lag notation, MA(q) can be written:

$$y_t = \mu + \left(1 + \sum_{i=1}^q \theta_i L^i\right) \varepsilon_t, \quad q = 1, 2, \dots \quad (8)$$

ARMA(p,q) is now expressed as below:

$$y_t = \phi_1 y_{t-1} + \phi_2 y_{t-2} + \dots + \phi_p y_{t-p} + \varepsilon_t - \theta_1 \varepsilon_{t-1} - \dots - \theta_q \varepsilon_{t-q} \quad (9)$$

where ε_t is independent of y_{t-1}, y_{t-2}, \dots

Autoregressive Integrated Moving Average Models—ARIMA(p,d,q)

The ARIMA model can be regarded as an extension of the ARMA model [7]. This process can be written as:

$$\left(1 - \sum_{i=1}^p \phi_i L^i\right) (1 - L)^d y_t = c + \left(1 + \sum_{i=1}^q \theta_i L^i\right) \varepsilon_t \quad (10)$$

where:

d is the number of differencing done to the series to achieve stationarity (Shumway and Stoffer [33]) with $I(d)$:

$$(1 - L)^d y_t = \mu + \varepsilon_t \quad (11)$$

p is the number of autoregressive terms (AR)

q is the number of moving average terms (MA)

The Autoregressive Conditional Heteroskedasticity—ARCH(q)

ARCH model [6] can be expressed as:

$$y_t = C + \varepsilon_t, \varepsilon_t = z_t \sigma_t \quad (12)$$

where:

y_t is an observed data series

C is a constant value

ε_t is residual

z_t is the standardized residual, independently and identically distributed with mean equal to 0 and variance tends toward 1 as sample size tends toward infinity, σ_t is the square root of the conditional variance, and it is a non-negative process.

ARCH(q) can be expressed in the following equation:

$$\sigma^2_t = \alpha_0 + \sum_{i=1}^q \alpha_i r^2_{t-i} \quad (13)$$

with $\alpha_0, \alpha_i \geq 0 (i = 1, \dots, q)$ so σ^2_t is non-negative.

Generalized Autoregressive Conditional Heteroscedasticity—GARCH(p,q)

The GARCH model is considered to be an extension of an ARCH model [16]. Unlike ARCH which involves only the most recent returns, generalized ARCH (GARCH) enhances the accuracy of forecasting by adding all the past squared returns with higher weights on more recent data and lower ones for faraway lags. Furthermore, GARCH is more restrained in comparison to ARCH, hence it can avoid overfitting and permit an infinite number of past squared errors to impact the current conditional variance [34]. So now, the conditional variance σ^2 is expressed by GARCH(p,q) as:

$$\sigma^2_t = \alpha_0 + \sum_{i=1}^q \alpha_i r^2_{t-i} + \sum_{j=1}^p \beta_j \sigma^2_{t-j} \quad (14)$$

GARCH(1,1) can be expressed by the equation below:

$$\sigma^2_t = \alpha_0 + \alpha_1 r^2_{t-1} + \beta_1 \sigma^2_{t-1} \quad \text{with } 0 < \alpha_1 + \beta_1 < 1 \quad (15)$$

and the rate of decay governed by $\alpha_1 + \beta_1$ where the closer $\alpha_1 + \beta_1$ is to 1, the slower the decay of the autocorrelation is. As proven by Bollerslev et al. [35], the valuations of GARCH(1,1) for stock returns usually yield $\alpha_1 + \beta_1$ very close to 1.

The Conditional Variance Equation: Exponential GARCH

The EGARCH model [17] is defined as $a_t = e_t \sigma_t$, in which:

$$\ln(\sigma^2_t) = \alpha_0 + g(e_{t-1}) + \beta_1 \ln(\sigma^2_{t-1}) \quad (16)$$

The function $g(e_{t-1})$ determines the asymmetry and is defined as the weighted innovation:

$$g(e_{t-1}) = \alpha_1 e_{t-1} + \gamma_1 [|e_{t-1}| - E(|e_{t-1}|)] \quad (17)$$

where: α_1 and γ_1 are real constants. This means the model can be written:

$$\ln(\sigma_t^2) = \alpha_0 + \alpha_1 e_{t-1} + \gamma_1 [|e_{t-1}| - E(|e_{t-1}|)] + \beta_1 \ln(\sigma_{t-1}^2) \quad (18)$$

Equation (18) informs us that a positive shock has the effect $(\alpha_1 + \gamma_1)e_{t-1}$, while the effect of the negative one has $(\alpha_1 - \gamma_1)e_{t-1}$. In reality, “good news” typically exerts a smaller impact on the volatility than “bad news”. For this reason, the use of $g(e_{t-1})$ allows the model to respond asymmetrically to “new information” in the market.

Equation (19) expresses the general EGARCH(s,r) models.

$$\ln(\sigma_t^2) = \alpha_0 + \sum_{i=1}^s g_i e_{t-i} + \sum_{j=1}^r \beta_j \ln(\sigma_{t-j}^2) \quad (19)$$

The Hybrid ARIMA-GARCH

As discussed above, ARIMA models are proposed for stationary time series with the assumption of constant variance, defined as “homoskedasticity”, while financial time series data often do not follow these assumptions. In practice, stock prices can be tremendously volatile during economic growth as well as recessions. In such scenarios, when homoskedasticity presumption is violated, it is said that the errors are heteroskedastic (a phenomenon known as heteroskedasticity). In other words, since heteroskedasticity is present, ARIMA or linear regression in general gives equal weights to all observations when observations with larger disturbance variance contain less information than the ones with smaller disturbance variance [36]. Given that heteroskedasticity can affect the validity or power of statistical tests when using ARIMA models, the ARCH effect should be considered.

Furthermore, according to Mandelbrot [37], large changes tend to be followed by large changes and vice versa. If volatility of a series exhibits such characteristics, it suggests that past variances might be predictive of the current variance. Hence, ARCH and GARCH models are the appropriate options in not only capturing the variance of each error term and correcting the deficiencies of heteroskedasticity for least squares but also dealing with the issue of volatility clustering. If one mechanism can simultaneously predict both the conditional mean and the conditional heteroscedasticity of the process, it is suggested as hybrid ARIMA-GARCH. It combines an ARIMA specification for modelling the mean behavior with the family of GARCH functions for simulating, estimating, and forecasting the variance behavior of the residuals from the ARIMA model. The hybrid ARIMA(p,d,q)-GARCH(r,s) can be specified as:

$$\begin{aligned} y_t &= \phi_1 y_{t-1} + \phi_2 y_{t-2} + \dots + \phi_p y_{t-p} + \varepsilon_t - \theta_1 \varepsilon_{t-1} - \dots - \theta_q \varepsilon_{t-q} \\ \sigma_t^2 &= \alpha_0 + \sum_{i=1}^q \alpha_i r_{t-i}^2 + \sum_{j=1}^p \beta_j \sigma_{t-j}^2 \\ \varepsilon_t &= z_t \sigma_t \quad (\varepsilon_t: N(0, \sigma_t^2)) \end{aligned} \quad (20)$$

3.2.2. Overview of the Methodology and Input Parameters

In order to achieve the goals that were mentioned in the Introduction section, the methodology of this research is structured in the following way:

Firstly, we conducted a rolling forecast based on an ARIMA model with window size(s) equal to 1000. The optimized combination of p and q which has the lowest AIC is used to predict return for the next point. At the end, the vector of forecasted values has the length of 3530 elements, with a starting point at 20 December 2005.

Next, we describe and review our implementation of dynamic ARIMA(p,1,q)-SGARCH(1,1) models with GED distribution and window size(s) equal to 1000 and where optimized ARIMA(p,1,q) is taken from the first step. Then, we evaluate the results based on error metrics, performance metrics, and equity curves.

After that, in the sensitivity analysis section, we build hybrid models with different input parameters: window size(s) equal to 500 and then 1500, with the following distributions: SNORM, SSTD, SGED.

Finally, we replaced SGARCH by EGARCH. We also conduct forecasting ARIMA on different window sizes in order to have a final conclusion regarding whether the hybrid model outperforms ARIMA in different input variables. We used the same criteria as in the main test to compare and evaluate the performance of each model.

To sum up, the forecasting models are centered around five sets displayed in Table 2 below.

Table 2. Sets of input parameters (ARIMA/hybrid ARIMA-xGARCH).

| Parameters | Values |
|--------------|---|
| Sample sizes | $s \in \{500, 1000, 1500\}$ (days) |
| Distribution | Generalized Error Distribution (GED) Skewed Normal Distribution (SNORM) Skewed Generalized Error Distribution (SGED) Skewed Student t Distribution (SSTD) |
| xGARCH MODEL | $x \in \{\text{SGARCH}, \text{eGARCH}\}$ (x represents the type of tested GARCH model. In other words, x is either symmetric (s)GARCH or exponential (e)GARCH.) |

Note: The letters in bold represent the parameters in the main test.

3.2.3. The Implementation of Forecasting Models

ARIMA(p,1,q)

This section gives an in-depth outline of the actual implementation of ARIMA(p,1,q). As can be seen from the flowchart in Figure 4, this process spans fitting and forecasting from selecting sample size until the one-day-ahead return is obtained. Additionally, the model assessment framework will be provided, including return generating properties and an overview of the model's computational complexity.

One optimal ARIMA(p,1,q) forecasting model is fitted using a rolling window approach with different combinations of p and q for the values of the input variables. This optimized model which has the lowest value of AIC is used to generate one-day-ahead return. Since the rolling window approach is applied, the next data point is estimated based on the sample size equal to the length of window. The mechanism of this method is illustrated more specifically in Figure 5, with three iterations and sample size(s) equal to 1000. With the starting point at 20 December 2005, we have 3530 forecasted values for the ARIMA model, based on which we have constructed equity lines for each strategy.

Testing one combination of p and q is referred to as one iteration. An important condition we set up in this loop is that p and q cannot be equal to 0 at the same time. This means that ARIMA(0,1,0) is excluded. The best fitting model is selected based on the lowest AIC. In each iteration, the inner loop compares $6 * 6 - 1 = 35$ models together to pick up the best one with the lowest value of AIC. To put it in another way, excluding the situation $p = q = 0$, with 6 values from 0 to 5, p and q generate 35 combinations. Hence, with the starting point t at 20 December 2005, each point in the vector of 3530 elements is forecasted based on the most optimized ARIMA within these 35 models. For the whole process, the loop generates a predicting returns vector by checking $3530 * 35 = 123,550$ models in total.

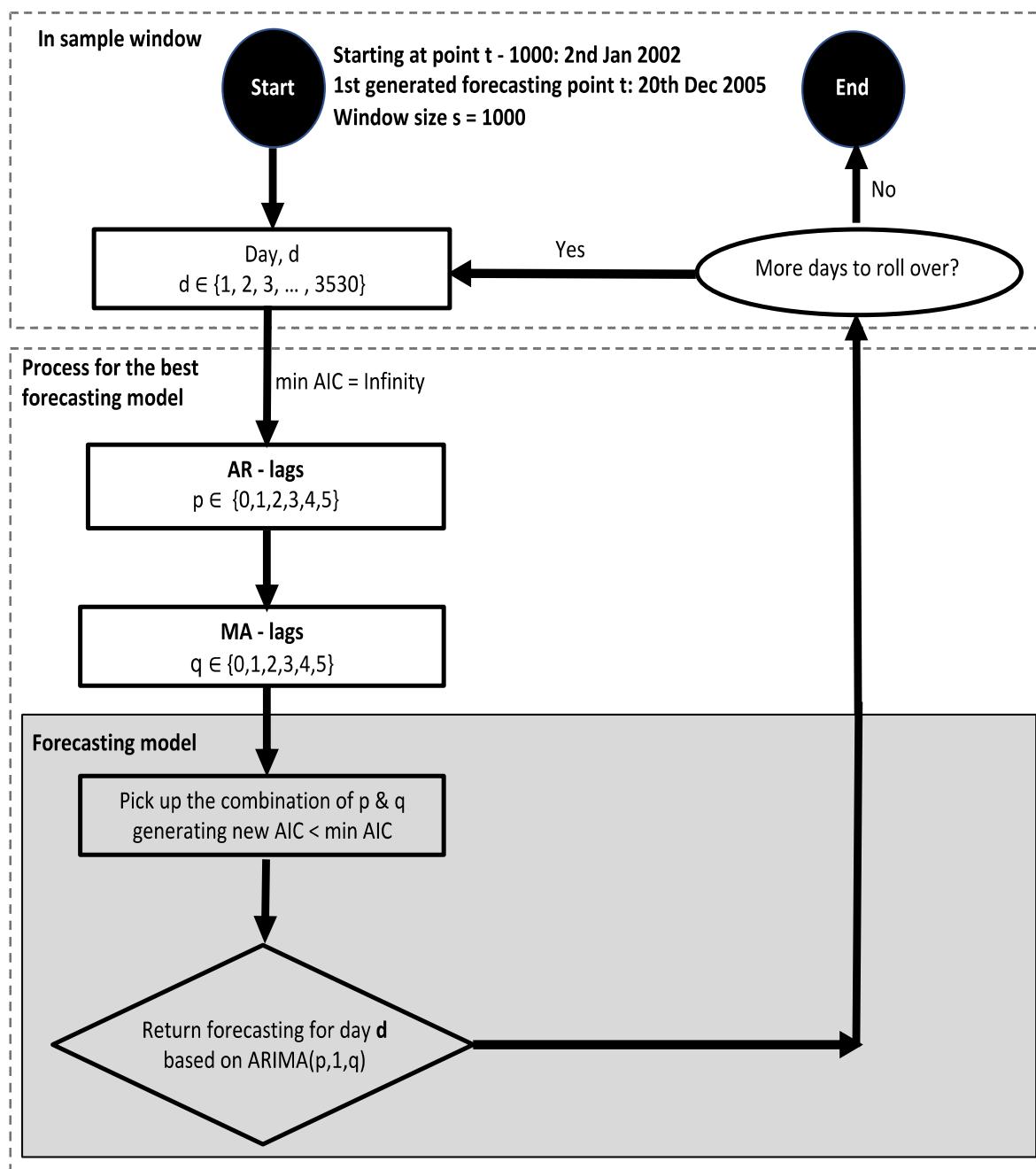


Figure 4. Flowchart of the forecasting model ARIMA(p,1,q). This flowchart is for models with window size $s = 1000$. For various values of window size, such as 500 or 1500, the first generated forecasting point t is the same, hence all models have the same length of forecasted log return vector of 3530. The difference here is the starting point where models select the first point to start a training model. For $s = 500$, the starting point should be at $t - 500$, and similarly, for $s = 1500$, the starting point should be at $t - 1500$.

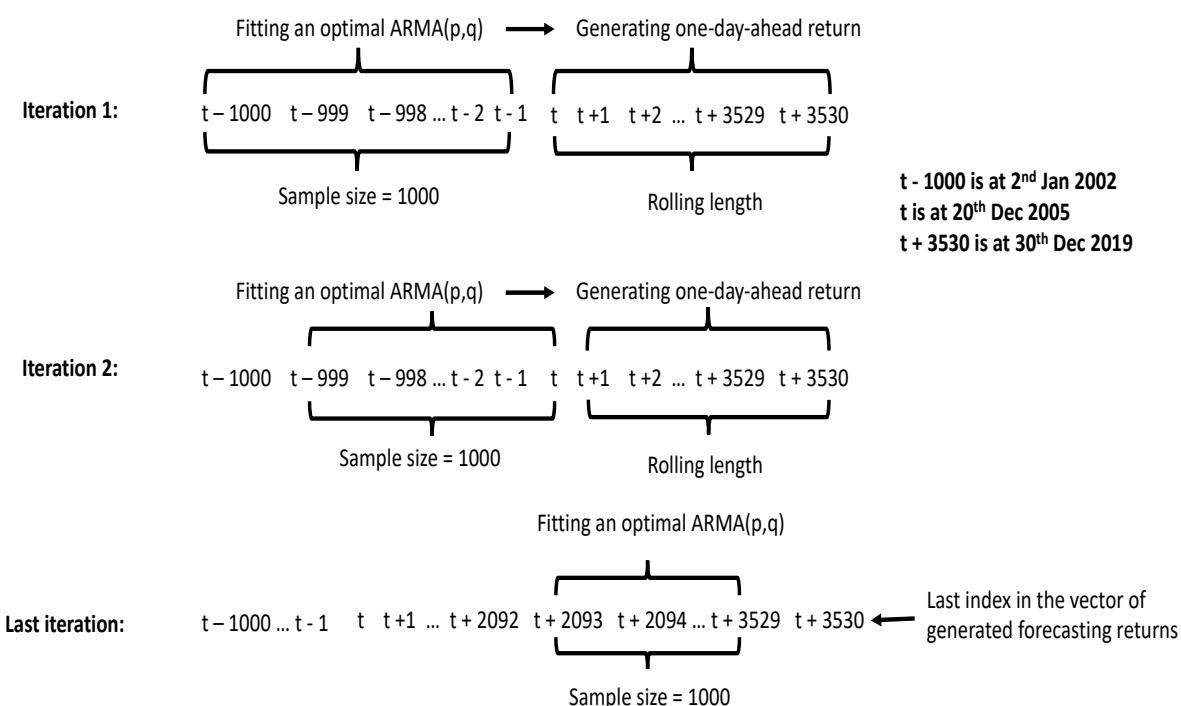


Figure 5. Rolling window illustration for sample size $s = 1000$. This illustration is for models with window size $s = 1000$ and applied the same for both the ARIMA and hybrid models in the next section. For various values of window size, such as 500 or 1500, the process is the same except the sample size understanding as window length. For $s = 500$, the starting point should be at $t - 500$, and similarly, for $s = 1500$, the starting point should be at $t - 1500$.

3.2.4. Dynamic ARIMA(p,1,q)-SGARCH(1,1)

In this section, we describe and review the implementation of dynamic ARIMA(p,1,q)-SGARCH(1,1). The steps applied to select parameters are similar to those used to fit ARIMA models, and they are described in Figure 6. It has similar steps as mentioned in Figure 4 regarding optimizing input parameters to fit model as well as the rolling process in Figure 5. The return distributions GED and SGARCH(1,1) with window size $s = 1000$ are used in building hybrid models. We have ARIMA(p,1,q)-SGARCH(1,1) as an optimal outcome per iteration to forecast the next value of log return, where SGARCH is applied to model the nonlinear patterns of the residuals. In other words, the error term ε_t of the ARIMA model in this process follows SGARCH(1,1) instead of being assumed constant like the ARIMA process in Figure 4.

3.2.5. Trading Strategy Criteria

In general, the rule for going long (buy) or short (sell) is as follows: if the forecasted log return is positive at time $t + 1$, we go long (buy stocks) at time t (direction would be $+1$); if the forecasted log return is negative at time $t + 1$, we go short (sell stocks) at time t (direction would be -1); and if the forecasted direction at time $t+1$ is the same as at time t , then there are no changes.

The initial investment is assumed to be \$1259.92 at the beginning. It is also the adjusted closing price of the S&P500 on 19 December 2005 (at $t - 1$), which is used as the starting point of the equity curves. For the benchmark, we used the Buy&Hold strategy on the S&P500 index in the period 20 December 2005–31 December 2019 and compared our strategy's performance with this benchmark.

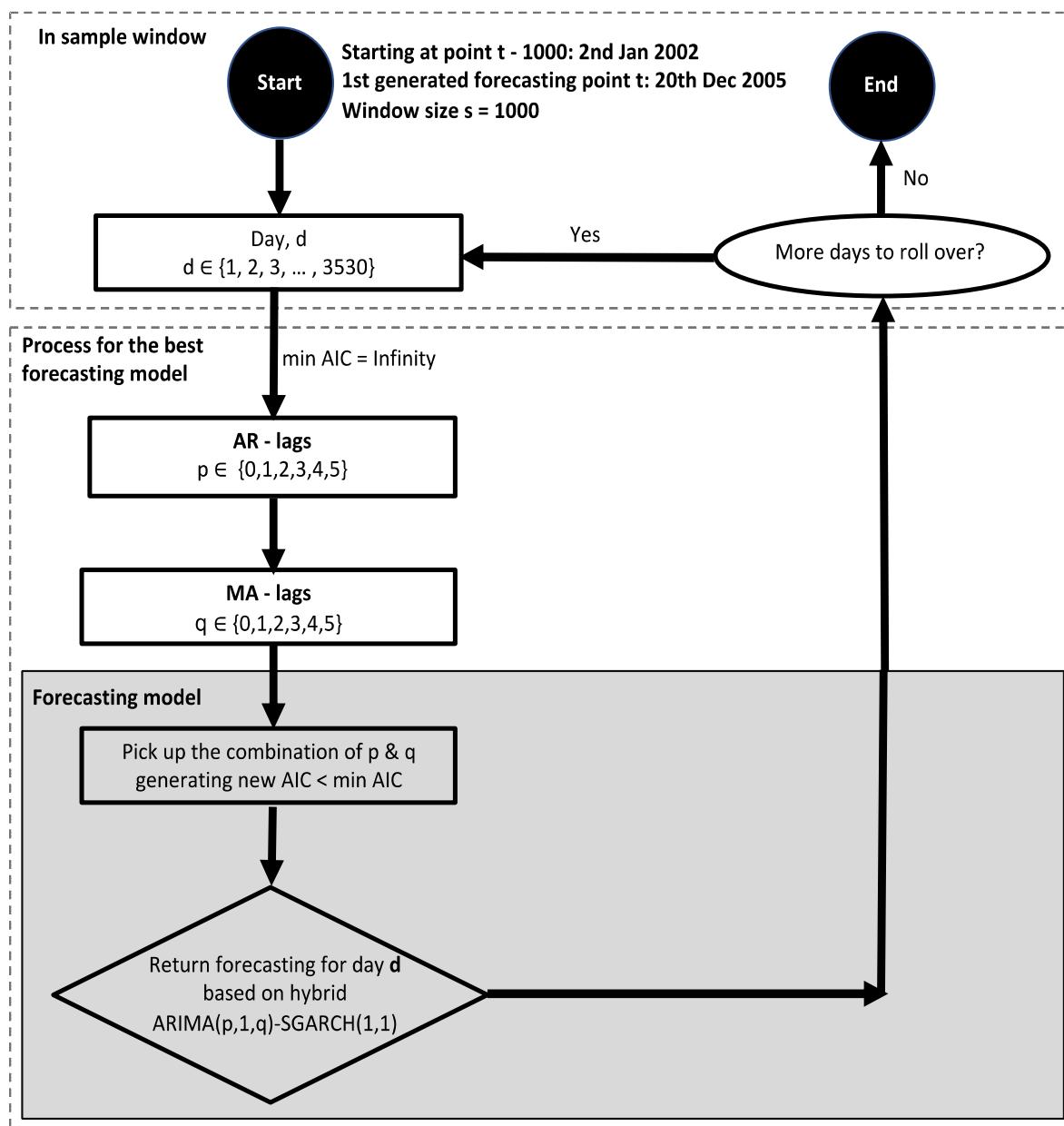


Figure 6. Flowchart of the forecasting model ARIMA(p,1,q)-SGARCH(1,1). This flowchart is for models with window size $s = 1000$. For various values of window size, such as 500 or 1500, the first generated forecasting point t is the same, hence all models have the same length of forecasted log return vector of 3530. The difference here is the starting point where models select the first point to start the training model. For $s = 500$, the starting point should be at $t - 500$ and similarly, for $s = 1500$, the starting point should be at $t - 1500$.

3.2.6. Criteria and Evaluation of Statistic Fit and Forecasting Akaike Information Criterion (AIC)

When a statistical model is selected to represent the process that generates the data, it will not be completely accurate. In other words, some “information” will be lost by applying this model in forecasting, and it might lead astray if the missing information is of great importance and has a huge effect on adopted data. However, there is a trade-off between the goodness of fit (how well the model fits a set of observations) and the number of parameters (more parameters \rightarrow more information) in the model. In order to avoid the risks of overfitting and underfitting, we apply the Akaike information criterion–AIC

(<https://www.statisticshowto.com/akaike-information-criterion/>, accessed on 15 July 2021). In general terms, AIC is an estimator of the relative quality of statistical models for a given dataset and also provides means for the model selection, which is expressed by the following formula:

$$AIC = 2k - 2\ln(\hat{L}) \quad (21)$$

where: k is the number of estimated parameters in the model and \hat{L} is the maximum value of the likelihood function for the model.

Error Metrics

In order to evaluate forecast from estimated models, we calculated the following error metrics:

mean absolute error (MAE)

$$MAE = \frac{1}{n} \sum_{t=1}^n |A_i - F_i| \quad (22)$$

where: n is the number of errors; A_i is the actual value and F_i is the forecasted value computed by the given model.

mean square error (MSE)

$$MSE = \frac{1}{n} \sum_{t=1}^n (A_i - F_i)^2 \quad (23)$$

root mean square error (RMSE)

$$RMSE = \sqrt{\frac{1}{n} \sum_{i=1}^n (A_i - F_i)^2} \quad (24)$$

mean absolute percentage error (MAPE)

$$MAPE = \frac{1}{n} \sum_{t=1}^n \left| \frac{A_i - F_i}{A_i} \right| \quad (25)$$

Performance Statistics

Moreover, in order to evaluate the efficiency of algorithmic investment strategies built based on the signals from econometric models, we calculated the performance metrics based on the created equity lines and formulas from Kość et al. [38] and Zenkova and Ślepaczuk [39].

Annualized return compounded (ARC)

ARC is expressed as percentage (%) and computed as:

$$ARC = \prod_{i=1}^N (1 + R_i)^{252/N} - 1 \quad (26)$$

where: R_i is the percentage rate of return and N is the sample size

Annualized standard deviation (ASD)

ASD is expressed as percentage (%) and computed as:

$$ASD = \sqrt{252} * \sqrt{\frac{1}{N-1} \sum_{i=1}^N (R_i - \bar{R})^2} \quad (27)$$

where: R_i is the percentage rate of return, \bar{R} is the average rate of return, and N is the sample size

Maximum drawdown (MD)

MD is the difference between the global maximum and the consecutive global minimum of the equity curve. The importance here is the time order, which means the global maximum must occur before the global minimum. It is expressed as below:

$$MD(S)_{t_1}^{t_2} = \max_{(x,y) \in \{[t_1,t_2]^2 : x \leq y\}} \frac{S_x - S_y}{S_x} \quad (28)$$

where: S is the price process; $[t_1, t_2]$ is the period between time t_1 and t_2

Information ratio (IR)

IR is the ratio between ARC and ASD informing us about risk adjusted returns for tested strategy

$$IR = \frac{ARC}{ASD} \quad (29)$$

Adjusted information ratio (IR*)

IR* is similar to IR, but it also takes into account MD as one of the risk factors. Then we have:

$$IR^* = \frac{ARC^2 * \text{sign}\{\text{ARC}\}}{ASD * MD} \quad (30)$$

where: sign {ARC} is the sign of ARC and can take values of 0, -1 or +1.

4. Results and Robustness Tests

4.1. Results

The performance of ARIMA (ARIMA 1000) and hybrid model ARIMA(p,1,q)-SGARCH(1,1) with GED distribution (SGARCH.GED 1000) as well as benchmark (Buy&Hold-S&P500) for window size(s) equal to 1000 are presented in Table 3. As the result shows, the hybrid model outperforms ARIMA and benchmark strategy evaluated based on error metrics and performance statistics. These results are much better even if compared with the ensemble models built from ML techniques (LSTM model) for S&P500 index (Michałkow et al.) [40] or rather complex approach using pair trading strategies for the constituents of the Nasdaq 100 index (Bui and Ślepaczuk) [41]. In particular, SGARCH.GED 1000 is more accurate than ARIMA 1000 in predicting returns and has the lowest values of MAE, MSE, RMSE, and MAPE (11.831 against 12.122; 303.044 against 310.372; 17.408 against 17.617; 0.00754 against 0.00775, respectively).

Table 3. Forecasting performance of ARIMA(p,1,q) and ARIMA(p,1,q)-SGARCH(1,1).

| METHOD | Error Metrics | | | | | Performance Statistics | | | | |
|-----------------|---------------|----------------|---------------|----------------|----------------|------------------------|----------------|--------------|--------------|--|
| | MAE | MSE | RMSE | MAPE | ARC | ASD | MD | IR | IR* | |
| BUY&HOLD S&P500 | | | | | 6.931% | 18.826% | 56.775% | 0.368 | 0.045 | |
| ARIMA 1000 | 12.122 | 310.372 | 17.617 | 0.00775 | 8.084% | 18.878% | 50.007% | 0.428 | 0.069 | |
| SGARCH.GED 1000 | 11.831 | 303.044 | 17.408 | 0.00754 | 14.026% | 18.893% | 25.885% | 0.742 | 0.402 | |

Note: In order to simplify the structure of the table, SGARCH.GED 1000 is understood as ARIMA(p,1,q)-SGARCH(1,1) with GED distribution and window size equal to 1000 days. MAE: mean absolute error; MSE: mean squared error; RMSE: root mean squared error; MAPE: mean absolute percentage error; ARC: annualized return compounded; ASD: annualized standard deviation; MD: maximum drawdown; IR = ARC / ASD: information ratio; IR* = (ARC^2 * sign(ARC)) / (ASD * MD): adjusted information ratio. Figures in bold indicate the best results.

Concerning performance statistics, the hybrid model generates the highest IR among the 3 methods with 0.742, the second one is ARIMA 1000 with IR of 0.428, and the last one is the benchmark with IR of 0.368. Although the hybrid model gives the highest ARC equal to 14.026%, its ASD is also the highest with 18.893%. However, it is not significantly different from the lowest value of 18.826% belonging to the benchmark. In terms of adjusted IR*, we can see that SGARCH.GED 1000 also outperforms ARIMA 1000. The difference between IR and IR* is that we additionally take into account MD as a measure of risk beside ASD. We can see that MD of ARIMA (50.007%) is almost 2 times higher than the MD of the hybrid

model (25.884%), while the ARC in the numerator of IR* (8.084%) is nearly 1.75 times lower than that of the hybrid model. As a result, the IR* of ARIMA 1000 is approximately 6 times smaller than that of SGARCH.GED 1000 (0.069 against 0.402). With this value of IR*, the hybrid method again beats the market with IR* of 0.045 (approximately 10 times higher).

To visualize the performance of ARIMA(p,1,q) and the hybrid model with GED distribution as well as the benchmark, the cumulative returns of these strategies are shown in Figure 7. The equity curves of ARIMA and the hybrid model remain below the Buy&Hold strategy for almost 2 years, but during the financial market crisis of 2008–2009, they behave tremendously well. In particular, from the latter half of 2010, ARIMA proved to be a good candidate, which even outperformed the hybrid model impressively in almost 2 years (2010–2011). However, at the end of 2011, ARIMA performance showed a dramatic decline and then remained below the hybrid model until the end of our data period. After the financial market crisis of 2008–2009, the hybrid model underwent an upward trend with small breaks until the end of 2019. In general, in spite of being under the ARIMA in short periods of time from 2008–2011, it is depicted as the most superior model in the whole discussed data frame. At the onset, it is clear it captures well all the movements of time series and is much better when compared with the benchmark.

The consequences of such results are very important from the point of view of the possibility of creating market-beating investment strategies and the allocation of assets by financial institutions actively managing assets on financial markets. Our results contradict the accepted Efficient Market Hypothesis in the informational sense in the weak form and should be the reason for further intensive research on this topic.

In general, error metrics, performance statistics, and equity curves imply that the hybrid model outperforms ARIMA and the benchmark. Referring to the main hypothesis of this paper, we can conclude that the combination between ARIMA(p,1,q) and SGARCH(1,1) is efficient.



Figure 7. Equity curves of ARIMA(p,1,q) & ARIMA(p,1,q)-SGARCH(1,1). In order to simplify the structure of the legend, SGARCH.GED 1000 is understood as ARIMA(p,1,q)-SGARCH(1,1) with GED distribution and window size equal to 1000 days; ARIMA 1000 is ARIMA(p,1,q) with window size $s = 1000$; BUY&HOLD-S&P500 is the benchmark strategy.

4.2. Robustness Test

In this Section, we verify whether the results we obtained above are robust to varying family of GARCH, various distributions, as well as different window lengths. In the previously obtained results, we conducted rolling forecasting on a hybrid model ARIMA(p,1,q)-

SGARCH with GED distribution and window size equal to 1000. In order to check the sensitivity of this result, we changed input parameters to conduct three extra tests. In particular, the first robustness test is to substitute SGARCH to EGARCH (keeping the same GED distribution and window size of 1000 days used in the main test). The second one is changing GED to a variety of distributions, such as SNROM, SSTD, and SGED (the other conditions of the main test remain unchanged). The last one is replacing the window size of 1000 to 500 and 1500 (the remaining conditions of the main test are kept the same).

4.2.1. Varying Family of GARCH Models

Table 4 informs us that ARIMA(p,1,q) has the worst performance, with the highest values of MAE, MSE, RMSE, and MAPE in comparison to EGARCH.GED 1000 (12.122 against 11.828, 310.372 against 301.745, 17.617 against 17.371, 0.00775 against 0.00753, respectively). Moreover, with the highest values of all key performance indicators (KPIs), where ARC = 11.010%, IR = 0.582 and IR* = 0.220, EGARCH.GED 1000 beats not only ARIMA with ARC = 8.084%, IR = 0.428 and IR* = 0.069, but also the benchmark with ARC = 6.931%, IR = 0.368 and IR* = 0.045.

The equity curves of all models and the benchmark are plotted in Figure 8. In general, despite being under ARIMA in some periods of time from 2008–2011, EGARCH.GED 1000 is depicted as a superior model in the whole analyzed data period. It leads to the conclusion that the transformation from SGARCH to EGARCH seems to be insensitive. That is to say, we can conclude that the combination of ARIMA(p,1,q) and EGARCH(1,1) outperforms ARIMA in a similar way as the combination of ARIMA(p,1,q) and SGARCH(1,1), and we can treat it as the answer to the first research question of this paper.

Table 4. Forecasting performance of ARIMA(p,1,q) & ARIMA(p,1,q)-EGARCH(1,1).

| METHOD | Error Metrics | | | | | Performance Statistics | | | |
|-------------------|---------------|----------------|---------------|----------------|----------------|------------------------|----------------|--------------|--------------|
| | MAE | MSE | RMSE | MAPE | ARC | ASD | MD | IR | IR* |
| BUY & HOLD S&P500 | | | | | 6.931% | 18.826% | 56.775% | 0.368 | 0.045 |
| ARIMA 1000 | 12.122 | 310.372 | 17.617 | 0.00775 | 8.084% | 18.878% | 50.007% | 0.428 | 0.069 |
| SGARCH.GED 1000 | 11.831 | 303.044 | 17.408 | 0.00754 | 14.026% | 18.893% | 25.885% | 0.742 | 0.402 |
| EGARCH.GED 1000 | 11.828 | 301.745 | 17.371 | 0.00753 | 11.010% | 18.901% | 29.150% | 0.582 | 0.220 |

Note: In order to simplify the structure of the table, EGARCH.GED 1000 is understood as ARIMA(p,1,q)-EGARCH(1,1) with GED distribution and window size equal to 1000 days. MAE: mean absolute error; MSE: mean squared error; RMSE: root mean squared error; MAPE: mean absolute percentage error; ARC: annualized return compounded; ASD: annualized standard deviation; MD: maximum drawdown; IR = ARC / ASD: information ratio; IR* = (ARC^2 * sign(ARC)) / (ASD * MD): adjusted information ratio. The figures in bold indicate the best results.

However, as can be seen in Table 4, although error metrics of EGARCH.GED.1000 have the lowest values, it cannot beat the SGARCH.GED 1000 in terms of performance statistics. With the ARC of 14.026%, IR* of 0.402, the hybrid model with SGARCH is the most superior strategy in comparison with the other 3 methods in Table 4. Furthermore, Figure 8 shows that from the beginning of 2016, we observe a big difference in the cumulative returns of these two hybrid models. EGARCH is introduced as more advanced than SGARCH since it takes the magnitude of volatility into consideration. In other words, EGARCH mitigates the disadvantage of GARCH by putting more weight on negative innovation since it tends to increase the volatility. However, the result based on IR*, which is selected as the most important performance statistic to evaluate the model, does not support this theory. This leads to the observation that the best model is not necessarily the same when the selection is based on the best error metrics or the best performance statistics. In a nutshell, as a response to the second research question in this paper, based on IR* as the main indicator for selecting the best model, SGARCH.GED 1000 outperforms EGARCH.GED 1000.



Figure 8. Equity curves of ARIMA($p,1,q$) & ARIMA($p,1,q$)-EGARCH(1,1). In order to simplify the structure of the legend, SARCH.GED 1000 is understood as ARIMA($p,1,q$)-SARCH(1,1) with GED distribution and window size equal to 1000 days; EGARCH.GED 1000 is understood as ARIMA($p,1,q$)-EGARCH(1,1) with GED distribution and window size equal to 1000; ARIMA 1000 is ARIMA($p,1,q$) with window size $s = 1000$; BUY&HOLD-S&P500 is the benchmark strategy.

4.2.2. Varying Window Sizes

Table 5 demonstrates the performance of ARIMA and SARCH.GED with window sizes of 500 and then 1500. The result of the main test with window size = 1000 is also included. Error metrics for the window size of 500 show that SARCH.GED 500, with the lower values of MAE, MSE, RMSE, and MAPE (11.91 against 12.216, 307.812 against 318.342, 17.545 against 17.842, 0.00758 against 0.00777, respectively), outperforms ARIMA 500. Moreover, with the higher values of KPIs in performance statistics (ARC = 5.912%, IR = 0.313 and IR* = 0.052), SARCH.GED 500 also beats ARIMA 500 (ARC = -0.574%, IR = -0.03 and almost zero for IR*).

Table 5. Performance of ARIMA($p,1,q$) and hybrid models in different window sizes.

| METHOD | Error Metrics | | | | | Performance Statistics | | | | |
|-------------------|---------------|----------------|---------------|----------------|----------------|------------------------|----------------|--------------|--------------|--|
| | MAE | MSE | RMSE | MAPE | ARC | ASD | MD | IR | IR* | |
| BUY & HOLD S&P500 | | | | | 6.931% | 18.826% | 56.775% | 0.368 | 0.045 | |
| ARIMA 500 | 12.216 | 318.342 | 17.842 | 0.00777 | -0.573% | 18.830% | 46.471% | -0.030 | 0.000 | |
| SARCH.GED 500 | 11.91 | 307.812 | 17.545 | 0.00758 | 5.912% | 18.871% | 35.666% | 0.313 | 0.052 | |
| ARIMA 1000 | 12.122 | 310.372 | 17.617 | 0.00775 | 8.084% | 18.878% | 50.007% | 0.428 | 0.069 | |
| SARCH.GED 1000 | 11.831 | 303.044 | 17.408 | 0.00753 | 14.026% | 18.893% | 25.885% | 0.742 | 0.402 | |
| ARIMA 1500 | 12.069 | 308.983 | 17.578 | 0.00771 | 5.005% | 18.852% | 50.733% | 0.265 | 0.026 | |
| SARCH.GED 1500 | 11.825 | 303.298 | 17.415 | 0.00753 | 12.186% | 18.896% | 25.885% | 0.645 | 0.304 | |

Note: In order to simplify the structure of the table, SARCH.GED 500 is understood as ARIMA($p,1,q$)-SARCH(1,1) with GED distribution and window size $s = 500$ days, similar for $s = 1000$ and $= 1500$ days. MAE: mean absolute error; MSE: mean squared error; RMSE: root mean squared error; MAPE: mean absolute percentage error; ARC: annualized return compounded; ASD: annualized standard deviation; MD: maximum drawdown; IR = ARC / ASD: information ratio; IR* = (ARC^2 * sign(ARC)) / (ASD * MD): adjusted information ratio. The figures in bold indicate the best results.

When window size is switched from 500 to 1500, we have the same results in which the hybrid models are demonstrated to be superior to ARIMA. Particularly, SARCH.GED 1500 has an ARC of 12.186%, which is almost 2.5 times higher than that of ARIMA (5.005%),

and ARIMA's $IR^* = 0.026$ is nearly 10 times lower than that of the hybrid models (0.304). The difference in the result of IR^* is because of the lower value of ARC as the numerator and the higher value of MD as the denominator leading to the lower final value of ARIMA's IR^* in comparison with the hybrid models.

Figures 9 and 10 illustrate the equity curves of tested strategies and the benchmark with the window sizes of 500 and 1500, respectively. Noticeably, in Figure 9, both ARIMA 500 and SGARCH.GED 500 underperform benchmark at the end. We could see that the Buy&Hold strategy is not necessarily the worst for various values. In general, our hybrid models seem to be sensitive to the values of window size. However, we can still conclude that hybrid models outperform ARIMA regardless of the values of window size as an input parameter. To recap, in response to the third research question in this paper, the results obtained from the main test are robust to varying window sizes.

Table 5 shows that with the highest value of ARC and IR^* (14.026% and 0.402, respectively), the hybrid models with the window size of 1000 are the best strategy among these three different window sizes. According to Figure 11, SGARCH.GED 1000 beats all mentioned methods. Based on IR^* as the main performance indicator in choosing the best model, SGARCH.GED 1000 outperforms the other hybrid models with different values of $s = 500$ and $= 1500$, and their differences are rather significant. In conclusion, the reply to the fourth research question in this paper is the hybrid models are sensitive to different window sizes.

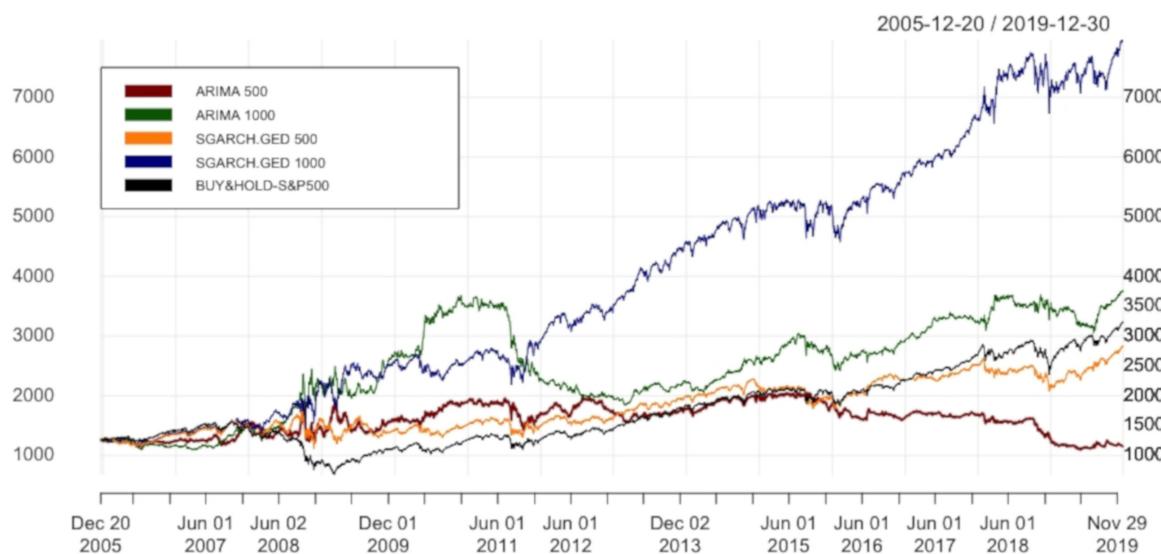


Figure 9. Equity curves of ARIMA($p,1,q$) and hybrid models with window sizes = 500 and 1000. In order to simplify the structure of the legend, SGARCH.GED 500/SGARCH.GED 1000 is understood as ARIMA($p,1,q$)-SGARCH(1,1) with GED distribution and window size(s) equal to 500/1000 days.

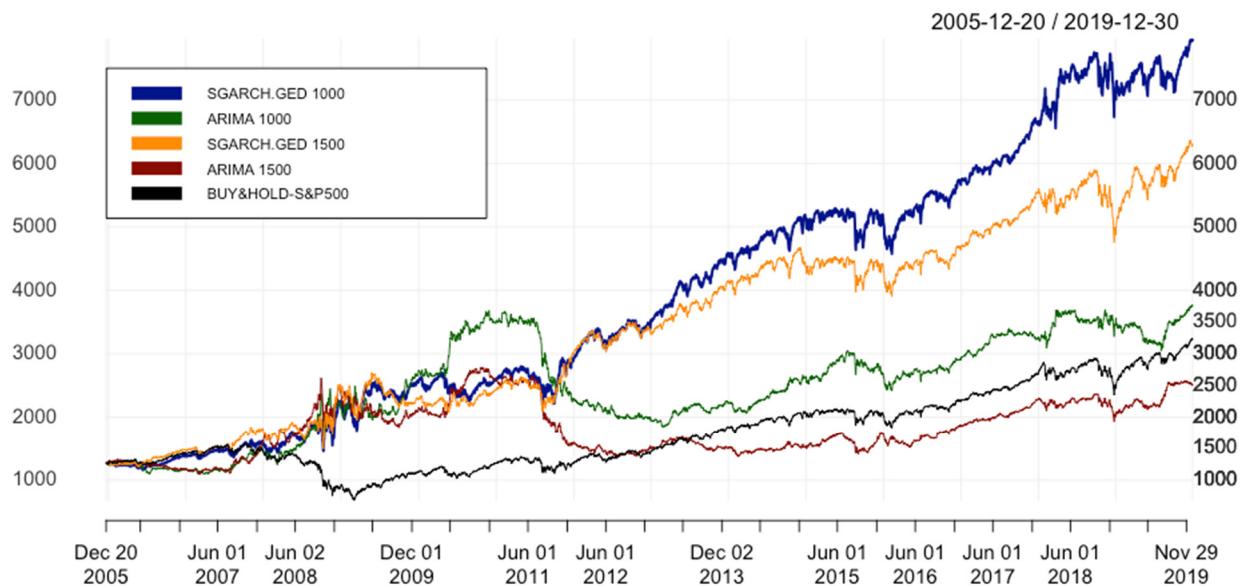


Figure 10. Equity curves of ARIMA(p,1,q) and hybrid models with window sizes = 1000 and 1500. In order to simplify the structure of the legend, SARCH.GED 1000/SARCH.GED 1500 is understood as ARIMA(p,1,q)-SGARCH(1,1) with GED distribution and window size(s) equal to 1000/1500 days.

4.2.3. Varying Distributions

Table 6 presents the results of hybrid models with varying distributions, including GED, SNORM, SSTD, and SGED. In terms of error metrics, it is quite evident that ARIMA 1000 has the worst performance with the highest values of MAE, MSE, RMSE, and MAPE in comparison with all hybrid models. Although ARIMA 1000's KPIs of performance statistics are higher than those of the benchmark, these figures are still lower than all hybrid models. Particularly, in terms of IR*, SGARCH.SNORM 1000 with 0.129, SGARCH.SSTD 1000 with 0.147 and SGARCH.SGED 1000 with 0.119 beat ARIMA 1000 with 0.069.

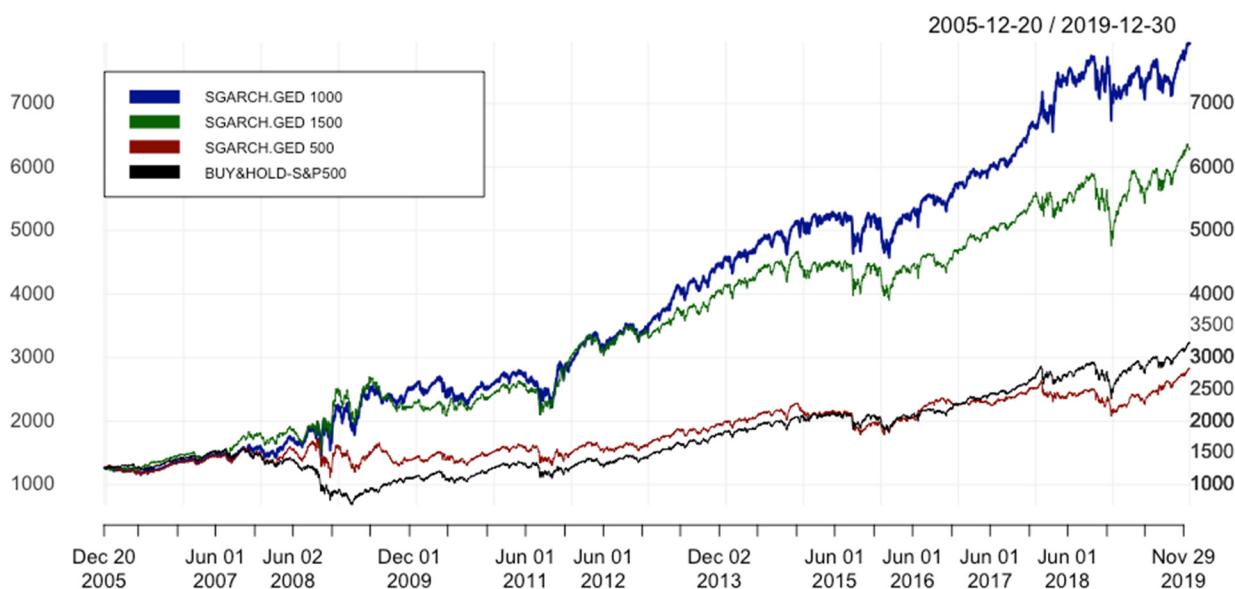


Figure 11. Equity curves of ARIMA(p,1,q)-SGARCH(1,1) with different window sizes. In order to simplify the structure of the legend, SARCH.GED 500/SARCH.GED 1000/SARCH.GED 1500 is understood as ARIMA(p,1,q)-SGARCH(1,1) with GED distribution and window size(s) equal to 500/1000/1500 days.

Table 6. Performance of ARIMA(p,1,q) and hybrid models in different distributions.

| METHOD | Error Metrics | | | | | Performance Statistics | | | |
|-------------------|---------------|----------------|---------------|----------------|----------------|------------------------|----------------|--------------|--------------|
| | MAE | MSE | RMSE | MAPE | ARC | ASD | MD | IR | IR* |
| BUY & HOLD S&P500 | | | | | 6.931% | 18.826% | 56.775% | 0.368 | 0.045 |
| ARIMA 1000 | 12.122 | 310.372 | 17.617 | 0.00775 | 8.084% | 18.879% | 50.007% | 0.428 | 0.069 |
| SGARCH.GED 1000 | 11.831 | 303.044 | 17.408 | 0.00754 | 14.026% | 18.893% | 25.885% | 0.742 | 0.402 |
| SGARCH.SNORM 1000 | 11.880 | 303.151 | 17.411 | 0.00758 | 8.987% | 18.890% | 33.079% | 0.476 | 0.129 |
| SGARCH.SSTD 1000 | 11.928 | 305.642 | 17.483 | 0.00762 | 8.860% | 18.881% | 28.373% | 0.469 | 0.147 |
| SGARCH.SGED 1000 | 11.848 | 302.362 | 17.389 | 0.00755 | 9.201% | 18.859% | 37.566% | 0.488 | 0.119 |

Note: In order to simplify the structure of the table, SGARCH.GED 1000 is understood as ARIMA(p,1,q)-SGARCH(1,1) with GED distribution and window size(s) equal to 1000 days, similar for SNORM, SSTD, and SGED. MAE: mean absolute error; MSE: mean squared error; RMSE: root mean squared error; MAPE: mean absolute percentage error; ARC: annualized return compounded; ASD: annualized standard deviation; MD: maximum drawdown; IR = ARC / ASD: information ratio; IR* = (ARC^2 * sign(ARC)) / (ASD * MD): adjusted information ratio. Figures in bold indicate the best results.

Figure 12 plots the equity curves of all hybrid models with various distributions as well as ARIMA, while window size remains 1000. It can be seen that the cumulative returns of hybrid models with SNORM, SSTD, and SGED distributions show no significant differences at the end, but all of them surpass ARIMA's. The results in Figure 12 and Table 6 draw the conclusion that regardless of the distributions, our hybrid models are more profitable than ARIMA. In short, the fifth research question can be answered as follows: the results obtained from the main test are robust to varying distributions. As can be seen in Figure 12, it is obvious that the best model is SGARCH.GED 1000, whose ending point is significantly far away from the rest and performance metrics are much better. As for the last research question of this paper, hybrid models are sensitive to different distribution and this with GED distribution outperforms the ones with other distributions, such as SNORM, SSTD, and SGED.

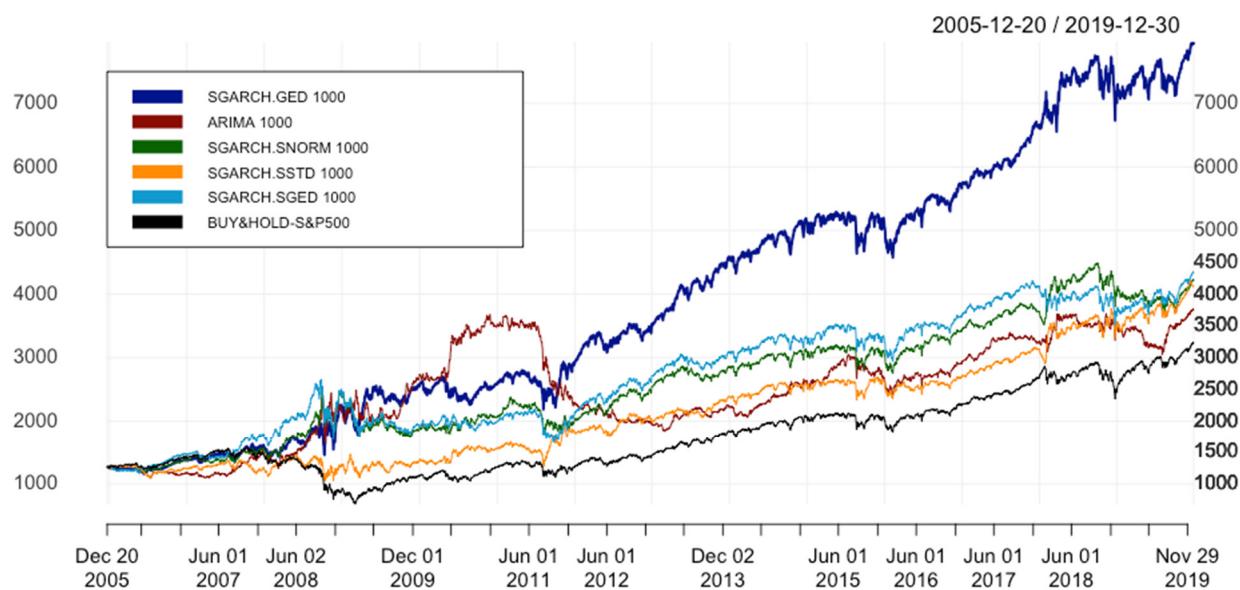


Figure 12. Equity curves of all hybrid models with different distributions. In order to simplify the structure of the legend, SGARCH.GED 1000 is understood as ARIMA(p,1,q)-SGARCH(1,1) with GED distribution and window size(s) equal to 1000 days, similar for SNORM, SSTD, and SGED.

5. Conclusions

The main hypothesis of this paper is whether the ARIMA(p,1,q)-SGARCH(1,1) (hybrid model) with window size equal to 1000 can generate an algorithmic trading strategy that outperforms ARIMA(p,1,q). Based on this hypothesis, the research questions are con-

structed as follows: whether the results in the main test are robust to (RQ1) varying family of GARCH models, (RQ3) varying window sizes, and (RQ5) varying distributions. We also evaluated and examined more research questions regarding whether the performance of hybrid models in the main test change with (RQ2) varying family of GARCH model; (RQ4) varying window sizes and (RQ6) varying distributions.

The dataset used for this research consists of the quotations of the S&P500 index. The data were collected on a daily basis over the period from 1 January 2000 to 31 December 2019. Next, the forecasted value was generated based on the best ARIMA model as a result of the best combination of p from 0 to 5 and q from 0 to 5, which has the lowest value of AIC. A rolling window of 1000 with one day ahead moving was selected for the main test. The vector of forecasted log returns with 3530 elements was generated. Based on these values, we set up the trading signals in which we entered the long position if forecasted log return was positive and the short one if forecasted log return was negative. By assuming the initial investment of \$1259.92 (the level of S&P500 index at the starting date), we calculated returns from the starting point of our out-of-sample window on 19 December 2005. Similar steps and process were conducted for ARIMA(p,1,q)-SGARCH(1,1) with GED distribution and window size $s = 1000$. The difference was that by combining ARIMA with SGARCH to create a hybrid ARIMA-GARCH model, conditional mean and variance could be simultaneously modeled (unlike ARIMA(p,1,q) where only conditional mean is modeled). We then calculated error metrics and performance statistics. Our benchmark was simply the Buy&Hold strategy on the S&P500 index. For the robustness test analysis, we conducted the same procedure step by step, as we did for the main test with changing input parameters, such as replacing GARCH with EGARCH, varying window sizes (500 & 1500), as well as distributions (SNORM, SSTD, SGED). In order to evaluate the performance of these models, we compute error metrics (MAE, MSE, MAPE, RMSE) and performance statistics (ARC, ASD, MD, IR, IR*) and present the equity curve for each model and also the benchmark.

Overall, the result shows that hybrid methods can generate a strategy that can outperform the ARIMA model even when we change our initial assumptions concerning the family of GARCH (RQ1), window sizes (RQ3), and distributions (RQ5). However, they are not always more efficient than our benchmark, as it was in the case of the hybrid model of SGARCH with GED distribution and window size of $s = 500$. Additionally, the hybrid model of SGARCH with GED distribution and window size equal to 1000 performs the best in comparison to other hybrid models in terms of changing different GARCH model (RQ2), window sizes (RQ4), and distributions (RQ6). Even being introduced as a model which can mitigate the disadvantages of SGARCH, the EGARCH model could not beat SGARCH in this research. To conclude, from the obtained results, the hybrid ARIMA-GARCH can generate a trading strategy that outperforms ARIMA(p,1,q) and should be taken into consideration in predicting returns and building trading strategies instead of applying only ARIMA to the series.

There are some limitations of this paper which can be improved in future works. The first and the biggest is that the proposed ugarchroll function in rugarch package (Author: Alexios Ghalanosonly) supports moving one period ahead only. It is time-consuming and reduces efficiency in practice. This issue may be addressed if we incorporate an extra wrapped function into the main one by making use of the underlying functions in the package. In addition, we applied fixed window sizes of 500, 1000, and 1500 without checking which exact value of the window size would deliver the best results. To solve this issue, we can try to build a loop function with the input parameters that can be a range of window sizes (for example the range of 500, 501, 503, . . . , 1499, 1500). In this way, the best value of window size which delivers the best trading strategy can be selected. Moreover, we applied a trading strategy based on ideal conditions without any transaction costs, the discussion of the influence of taxes on our results (Batrancea) [42], or a certain threshold. This means we should take into consideration the magnitude of forecast return value instead of building a strategy based on the sign of forecasted values. This issue may

be addressed by assigning cost for any transaction and setting a threshold with which we can compare forecasted return value before generating the direction for trading (+1 for entering long or -1 for entering short).

Author Contributions: Conceptualization, N.V. and R.Ś.; Data curation, N.V.; Formal analysis, N.V.; Investigation, N.V. and R.Ś.; Methodology, N.V. and R.Ś.; Project administration, R.Ś.; Software, N.V.; Supervision, R.Ś.; Validation, N.V. and R.Ś.; Visualization, N.V.; Writing—original draft, N.V.; Writing—review & editing, R.Ś. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: All relevant data are within the paper.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Sakowski, P.; Turovtseva, A. *Verification of Investment Opportunities on the Cryptocurrency Market within the Markowitz Framework*; Working Papers 2020-41; Faculty of Economic Sciences, University of Warsaw: Warsaw, Poland, 2020.
2. Torre-Torres, O.V.; Venegas-Martínez, F.; Martínez-Torre-Enciso, M. Enhancing portfolio performance and VIX futures trading timing with markov-switching GARCH models. *Mathematics* **2021**, *9*, 185. [[CrossRef](#)]
3. Guobužaité, R.; Teresienė, D. Can Economic Factors Improve Momentum Trading Strategies? The Case of Managed Futures during the COVID-19 Pandemic. *Economies* **2021**, *9*, 86. [[CrossRef](#)]
4. Ivanyuk, V. Formulating the concept of an investment strategy adaptable to changes in the market situation. *Economies* **2021**, *9*, 95. [[CrossRef](#)]
5. Chlebus, M.; Dyczko, M.; Woźniak, M. Nvidia’s Stock Returns Prediction Using Machine Learning Techniques for Time Series Forecasting Problem. *Cent. Eur. Econ. J.* **2021**, *8*, 44–62. [[CrossRef](#)]
6. Strickland, J. *Predictive Analytics Using R*; Published by Lulu, Inc.: Morrisville, NC, USA, 2014; ISBN 978-1-312-84101-7.
7. Box, G.E.P.; Jenkins, G.M. *Time Series Analysis Forecasting and Control*; Holden Day: San Francisco, CA, USA, 1976.
8. Ariyo, A.; Adewumi, A.; Ayo, C. Stock Price Prediction Using the ARIMA Model. In Proceedings of the UKSim-AMSS 16th International Conference on Computer Modelling and Simulation, Cambridge, UK, 26–28 March 2014; pp. 105–111. [[CrossRef](#)]
9. Kamruzzaman, M.; Khudri, M.M.; Rahman, M.M. Modeling and Predicting Stock Market Returns: A Case Study on Dhaka Stock Exchange of Bangladesh. *Dhaka Univ. J. Sci.* **2017**, *65*, 97–101. [[CrossRef](#)]
10. Abbasi, U.; Almarashi, A.; Khan, K.; Hanif, S. Forecasting cement stock prices using ARIMA model: A case study of flying cement. *Sci. Int.* **2017**, *29*, 1939–1043.
11. Engle, R.F. Autoregressive conditional heteroscedasticity with estimates of the variance of United Kingdom inflation. *Econometrica* **1982**, *50*, 987–1007. [[CrossRef](#)]
12. Ghani, I.M.; Rahim, H.A. Modeling and Forecasting of Volatility using ARIMA-GARCH: Case Study on Malaysia Natural Rubber Prices. *IOP Conf. Ser. Mater. Sci. Eng.* **2018**, *548*, 012023. [[CrossRef](#)]
13. Chua, C.L.; Tsipalias, S. Information flows and stock market volatility. *J. Econom.* **2019**, *34*, 129–148. [[CrossRef](#)]
14. Jansen, S. *Machine Learning for Algorithmic Trading: Predictive Models to Extract Signals from Market and Alternative Data for Systematic Trading Strategies with Python*, 2nd ed.; Packt Publishing Ltd.: Birmingham, UK, 2020.
15. Tsay, R.S. *An Introduction to Analysis of Financial Data with R*; John Wiley & Sons: Street Hoboken, NJ, USA, 2012.
16. Bollerslev, T. Generalized autoregressive conditional heteroscedasticity. *J. Econom.* **1986**, *31*, 307–327. [[CrossRef](#)]
17. Engle, R.F. GARCH 101: The Use of ARCH/GARCH Models in Applied Econometrics. *J. Econ. Perspect.* **2001**, *15*, 157–168. [[CrossRef](#)]
18. Engle, R.F.; Ng, V.K.; Rothschild, M. Asset pricing with a factor-ARCH covariance structure: Empirical estimates for treasury bills. *J. Econom.* **1990**, *45*, 213–237. [[CrossRef](#)]
19. Higgins, M.L.; Bera, A.K. A class of nonlinear ARCH models. *Int. Econ. Rev.* **1992**, *33*, 137–158. [[CrossRef](#)]
20. Glosten, L.R.; Jagannathan, R.; Runkle, D.E. On the relation between the expected value and the volatility of the nominal excess return on stocks. *J. Financ.* **1993**, *48*, 1779–1801. [[CrossRef](#)]
21. Sentana, E. Quadratic ARCH Models. *Rev. Econ. Stud.* **1995**, *62*, 639–661. [[CrossRef](#)]
22. Nelson, D.B. Conditional Heteroskedasticity in Asset Returns: A New Approach. *Econometrica* **1991**, *59*, 347–370. [[CrossRef](#)]
23. Wilhelmsson, A. Garch Forecasting Performance under Different Distribution Assumptions. *J. Forecast.* **2006**, *25*, 561–578. [[CrossRef](#)]
24. Bollerslev, T. A Conditionally Heteroskedastic Time Series Model for Speculative Prices and Rates of Return. *Rev. Econ. Stat.* **1987**, *69*, 542–547. [[CrossRef](#)]

25. Weiss, A.A. ARMA Models With ARCH Errors. *J. Time Ser. Anal.* **1984**, *5*, 129–143. [[CrossRef](#)]
26. Jabłecki, J.; Kokoszczyński, R.; Sakowski, P.; Ślepaczuk, R.; Wójcik, P. *Volatility as an Asset Class: Obvious Benefits and Hidden Risks*; Peter Lang: Frankfurt am Main, Germany, 2015; ISBN 978-3-653-97884-1. [[CrossRef](#)]
27. Castellano Gomez, S.; Ślepaczuk, R. Optimisation in algorithmic investment strategies. *Work. Pap. Fac. Econ. Sci. Univ. Wars.* 2021. Available online: https://www.wne.uw.edu.pl/application/files/8616/3847/9793/WNE_WP375.pdf (accessed on 19 December 2021).
28. Hauser, M.; Kunst, R. Fractionally Integrated Models with ARCH Errors: With an Application to the Swiss One-Month Euromarket Interest Rate. *Rev. Quant. Financ. Account.* **1998**, *10*, 95–113. [[CrossRef](#)]
29. Kijewski, M.; Ślepaczuk, R. Predicting prices of S&P500 index using classical methods and recurrent neural networks. *Work. Pap. Fac. Econ. Sci. Univ. Wars.* 2020. Available online: https://www.wne.uw.edu.pl/index.php/download_file/5769/4919/ (accessed on 15 July 2021).
30. Yaziz, S.R.; Azizan, N.A.; Zakaria, R.; Ahmad, M. The performance of hybrid ARIMA-GARCH modeling in forecasting gold price. In Proceedings of the 20th International Congress on Modelling And Simulation, Adelaide, Australia, 1–6 December 2013; pp. 1201–1207.
31. Sun, K. Equity Return Modeling and Prediction Using Hybrid ARIMA-GARCH Model. *Int. J. Financ. Res.* **2017**, *8*, 154–161. [[CrossRef](#)]
32. Ismail, M.T.; Mustapa, F. Modelling and Forecasting S&P500 Stock Prices Using Hybrid ARIMA-GARCH Model. *J. Phys. Conf. Ser.* **2019**, *1366*, 012130. [[CrossRef](#)]
33. Shumway, R.H.; Stoffer, D.S. *Time Series Analysis and Its Applications With R Examples*, 2nd ed.; Springer: Berlin/Heidelberg, Germany, 2000.
34. Brooks, C. *Introductory Econometrics for Finance*, 2nd ed.; Cambridge University Press: Cambridge, UK, 2008.
35. Bollerslev, T.; Chou, R.; Kroner, K.F. ARCH Modeling in Finance: A Review of The Theory and Empirical Evidence. *J. Econom.* **1992**, *52*, 5–59. [[CrossRef](#)]
36. Allison, P.D. *Multiple Regression: A Primer*; Pine Forge Press: Thousand Oaks, CA, USA, 1999.
37. Mandelbrot, B. The Variation of Certain Speculative Prices. *J. Bus.* **1963**, *36*, 394–419. [[CrossRef](#)]
38. Kość, K.; Sakowski, P.; Ślepaczuk, R. Momentum and Contrarian Effects on the Cryptocurrency Market. *Phys. A Stat. Mech. Its Appl.* **2019**, *523*, 691–701. [[CrossRef](#)]
39. Zenkova, M.; Ślepaczuk, R. Robustness of Support Vector Machines in Algorithmic Trading on Cryptocurrency Market. *Cent. Eur. Econ. J.* **2018**, *5*, 186–205. [[CrossRef](#)]
40. Michańkow, J.; Sakowski, P.; Ślepaczuk, R. The comparison of LSTM in algorithmic investment strategies on BTC and SP500 index on various frequencies. *Sensors* **2022**, in press.
41. Bui, Q.; Ślepaczuk, R. Applying Hurst Exponent in Pair Trading Strategies on Nasdaq 100 index. *Phys. A Stat. Mech. Its Appl.* **2021**, *592*, 126784. [[CrossRef](#)]
42. Batrincea, L. An Econometric Approach Regarding the Impact of Fiscal Pressure on Equilibrium: Evidence from Electricity, Gas and Oil Companies Listed on the New York Stock Exchange. *Mathematics* **2021**, *9*, 630. [[CrossRef](#)]