Open Access Library Journal
2025, Volume 12, e12978
ISSN Online: 2333-9721
ISSN Print: 2333-9705

Derivation of Imputation Estimators for ARMA Models with GARCH Innovations

Merary Kipkogei*, Wilfred Omwansa Arori, Joyce Akinyi Otieno

Department of Statistics and Actuarial Science, Maseno University, Kisumu, Kenya
Email: *kipkogeimerary01@gmail.com

How to cite this paper: Kipkogei, M., Arori, W.O. and Otieno, J.A. (2025) Derivation of Imputation Estimators for ARMA Models with GARCH Innovations. *Open Access Library Journal*, **12**: e12978.
https://doi.org/10.4236/oalib.1112978

Received: January 18, 2025
Accepted: February 24, 2025
Published: February 27, 2025

Abstract

The paper investigated the problem of restoring missing values in time series data analysis. The aim of this study was to advance the imputation of missing values for some autoregressive moving average models (ARMA) with generalized autoregressive conditioned heteroscedastic (GARCH) models. In this work, the novel imputation estimators for ARMA (1, 1) + GARCH (1, 1) and ARMA (2, 2) + GARCH (2, 2) were derived. The study utilized the method of optimal interpolation, whereby the innovation terms of the involved processes were minimized in the sense of dispersion. The study tested the consistency of the estimators using simulated data. A sample of a thousand (1000) observations was generated using R software following the proposed models. A hundred (100) positions of missing values were created at random within the data generated. Besides, the study carried out a comparison between the derived estimators and the celebrated machine learning Artificial Neural Networks (ANN), K-Nearness Neighbors (KNN) and the Kalman filters techniques. The imputation performance was carried out using the following metrics; Mean Error (ME), Root Mean Squared Error (RMSE), Mean Absolute Error (MAE), Mean Percentage Error (MPE), Mean Absolute Percentage Error (MAPE). The study found that the derived novel imputation estimator for ARMA (2, 2) +GARCH (2, 2) process was superior to the imputation estimator of ARMA (1, 1) + GARCH (1, 1) process. The derived estimators competed well compared to modern missing values imputation techniques. This paper gave a clear comparison that the ANN technique was the best followed by optimal interpolation technique while the Kalman technique was the last according to imputation performance. The study recommends that the derived estimators be utilized to input missing values for time series with GARCH innovations. The rational for this study is to contribute to the field of missing values imputation for nonlinear time series modeling.

Subject Areas

Applied Statistical Mathematics, Statistics and Econometrics

Keywords

Autoregressive Moving Average, Artificial Neural Networks, Generalized Autoregressive Conditional Heteroscedastic, Imputation, Kalman Filters, Missing Values

1. Introduction

A fashioned fact about time series data is the presence of missing values. In most cases missing values in time series arise due to many reasons that could include lost records, poor record keeping and recording, failure to record some data, failure of recording machines and gadgets among other reasons.

Time series data containing missing values were formerly approached in two eminent ways: ignoring missing values and/or deleting missing values. As a result of deletion and ignoring of missing values, a great deal of vital information is left out. This way, an analysis and handling of such data with missing and ignored data could give the wrong guides during data interpretation. Missing values always pose serious threats to any data analysis by hampering the estimation and forecasting of data being modeled.

In this endeavor, researchers have made use of substitutes to replace the missing values in time series using numerical values arrived from statistical methods. Some statistical methods are meant to replace missing values using some predicted values from the available data. The process of replacing missing values using some substitute values is called missing values imputation. Imputation of missing values is one of the critical steps in data cleaning processes that should not be avoided. Missing values imputation techniques have fair aspirations for handling missing values. One of the most vital aspects of missing values imputation is that imputation maintains accuracy and integrity of any data. Besides, imputation aids and supports data analysis by guaranteeing reliable data [1]. When missing values occur within any data, it is important to impute by estimating the missing values using an accurate technique.

Missing values imputations have been developed for most time series models, including linear and nonlinear models. For non-linear models, imputation has been done for autoregressive moving average (ARMA) models with stable Gaussian errors see reference [2]. Also, bilinear time series models with GARCH innovations have been carried out see reference [3]. However, there is little literature on imputation of missing values for ARMA models with GARCH innovations. This is the gap we wish to bridge through this study.

In this study, we developed imputation estimators for ARMA series models when the innovations of the assumed model have GARCH assumptions.

1.1. Autoregressive Moving Average Models

Let X_t be partly autoregressive and partly a moving average process, such that the process is generalized as,

$$X_t = \varphi_1 x_{t-1} + \varphi_2 x_{t-2} + \cdots + \varphi_p x_{t-p} + \epsilon_t - \vartheta_1 \epsilon_{t-1} - \vartheta_2 \epsilon_{t-2} + \cdots + \vartheta_q \epsilon_{t-q} \quad (1)$$

Then X_t is a linear combination of an autoregressive model of order (p) and a moving average model of order (q). Such process given by X_t is known as an autoregressive moving average (ARMA) model. The above model is always given as ARMA (p, q) model. The autoregressive moving average (ARMA) model is applicable to modeling the time series nature of processes.

1.2. Generalized Autoregressive Heteroscedastic Conditioned (GARCH) Model

Authors in reference [4], gave rise to the celebrated autoregressive heteroscedastic ARCH(q) model as a stationary process which has been well specified by X_t as, $X_t = \eta_t \mathcal{E}_t$, $\mathcal{E}_t \sim iid\ N(0,1)$ where η_t is a positive function defined by

$$\eta^2_{t|t-1} = \omega + \alpha_1 e^2_{t-1} + \cdots + \alpha_q e^2_{t-q} \quad (2)$$

with $\omega > 0$, $\alpha_j \geq 0$, $j = 1, \cdots, q$.

Where q is the order of the process. When fitting data, at times it is found better to relax Gaussian assumptions by the GARCH model's innovations and suppose that the model's innovations given by $\mathcal{E}_t$ can assume heavy tailed zero mean distribution such as the generalized error distribution. This can be possible when handling data that does not obey normality and has heteroscedastic variances.

The ARCH(q) model was modified by authors in reference to [5] so that it could contain and accommodate the conditioned variances or volatility. The above model was therefore renamed as Generalized autoregressive heteroscedastic conditioned (GARCH) model. The GARCH (q, p) model was given as a stationary process X_t, satisfying, $X_t = \eta_t \mathcal{E}_t$, $\mathcal{E}_t \sim iid\ N(0,1)$ where η_t is a positive function defined by

$$\eta^2_{t|t-1} = \omega + \alpha_1 e^2_{t-1} + \cdots + \alpha_q e^2_{t-u} + \beta_1 \sigma^2_{t-1} + \cdots + \beta_v \sigma^2_{t-p} \quad (3)$$

During modeling of this process, it can be assumed that either $X_t = \eta_t \mathcal{E}_t$, $\mathcal{E}_t \sim iid\ N(0,1)$ or that $\sqrt{\frac{v}{v-2}}\mathcal{E}_t \sim t_v$ where $v > 2$.

Where t_v denotes the students t distribution with v degrees of freedom. For t_v other distributions can be adopted.

1.3. Autoregressive Moving Average with Generalized Autoregressive Conditioned (GARCH) Models

In more general cases, generalized autoregressive conditioned (GARCH) model structure may be modeled by ARMA (p, q) models in various ways. One of the ways is that the GARCH model can be taken as the error terms of ARMA models. Such scenarios can be given as.

Let X_t be an ARMA (p, q) time series model with GARCH (u, v) innovations, this could be given by,

$$X_t = \varphi_1 x_{t-1} + \varphi_2 x_{t-2} + \cdots + \varphi_p x_{t-p} + \epsilon_t - \vartheta_1 \epsilon_{t-1} - \vartheta_2 \epsilon_{t-2} - \cdots - \vartheta_q \epsilon_{t-q} \quad (4)$$

Such that $\epsilon_t = \eta_{t|t-1} e_t$, where $e_t \sim iid(0,1)$ and $\eta^2_{t|t-1} = \omega + \alpha_1 e^2_{t-1} + \cdots + \alpha_u e^2_{t-u} + \beta_1 \sigma^2_{t-1} + \cdots + \beta_v \sigma^2_{t-v}$

Besides, $e_t \sim iid(0,1)$, can be taken to have a mean zero and unit variance and any distribution be it a symmetric like Normal distribution or asymmetric case like Gamma distribution and the Generalized Error distribution (GED).

The objectives of this study were:

1) To develop a novel ARMA model with GARCH innovations assumptions for imputing missing values.

2) To compare the imputation efficiency of the developed imputation estimators verses the modern imputation techniques.

2. Literature Review

2.1. Imputation of Missing Values for GARCH Models

The GARCH modeling with the presence of missing values has lately been done in time series analysis. An overview of what has been done includes.

Authors in reference [6], conducted a study on an approximation of intractable likelihood of GARCH (1, 1) models with missing values through sequential Monte Carlo (SMC) and maximum likelihood estimation of imputation estimators. The authors, in reference [7], carried out a study that refuted the use of Gaussian modeling for a time varying mean for modeling missing values in GARCH models and they instead gave a novel estimation criterion based on an indirect inference to estimate missing values with GARCH models. Again, reference in reference [8] conducted a study on an estimation of missing values in GARCH models using quasi likelihood estimator. Authors in reference [9] proposed an algorithm for Log-GARCH model that treats the missing values as zero values. Their algorithm handled the missing values by estimating them through ARMA representation. Authors in reference [6] estimated missing values for GARCH model using expectation Maximization (EM) estimation. Their study never gave an empirical case of model predictability and accuracy. Studies authors in reference [3] conducted a study on estimation and imputation of missing values for bilinear time series models with GARCH innovations. Their study used the optimal linear interpolation technique for estimation. Besides, their study involved measurement of the efficiency of the derived estimates. Their estimates were better for imputing missing values for bilinear time series data. The authors in reference [10] did a presentation on a non-linear state formulation of GARCH model with missing values. Their study proposed that Kalman filters could handle the missing value problem more comfortably. Authors in reference [11] fitted a simple ARCH model in presence of missing data, their study utilized least squares estimation to tackle the missing value problems. Through Bondon's study, there was a suggestion for advancements in the imputation of missing values for time series with

GARCH models.

To this end, there is no study that has been carried out on derivation and formulation of imputation estimators for imputing missing values for ARMA models especially when the models under consideration have taken the generalized autoregressive conditioned heteroscedastic (GARCH) innovations using the optimal linear interpolation estimation.

2.2. Optimal Linear Interpolation Technique

The proposed technique was first presented by authors [2] for the estimation of missing values in ARMA with stable symmetric Gaussian error assumptions. The method has been utilized by authors [12] for the development of imputation estimators for pure bilinear time series models. They explained how the imputation computation is arrived at using the following statements.

Suppose an observation k_m is a missing value out of a set of *n*-possible observations generated by an ARMA (*p*, *q*) process. Let the subspace Q_m^* be the allowable space of a linear estimator of k_m based on observed values $k_t, k_{t-1}, \cdots, k_{m-1}$ that are given by $Q_m^* = Q_p\left\{k_t : t \le n; t \ne m\right\}$. The projection of k_t on to Q_m^* denoted as $P_{S_m}^{k_m}$ such that the disp $\left\{K_m - P_{S_m}^{K_m}\right\}$ is minimized, that is basically the minimum dispersion of the linear estimator. Direct computation of the projection of the K_m on to Q_m^* would be complicated since the subspace $Q_1 = Q_P\left\{k_{m-1}, k_{m-2}, \cdots\right\}$ and Q_m^* are not independent of each other and thus we consider the evaluation of the projection on to two disjoint subspaces of Q_m^*. To achieve this, we express Q_m^* as a direct sum of subspaces Q_1 and another subspace, say Q_t such that $Q_m^* = Q_1 \oplus Q_*$. A possible sub-space is $Q_* = Q_p\left\{k_i - k_i^t; i > m+1\right\}$. Where k_i^t is based on the values $\left\{k_{m-1}, k_{m-2}, \cdots\right\}$. The existence of subspaces Q_1 and Q_* are shown in the following lemma;

Lemma

Suppose k_t is non-determined stationary process defined on the probability space (Ω, β, ρ). Then the subspace Q_m^* is the direct sum of subspaces Q_1 and Q_* as defined in the above norm.

Proof

Suppose $K_* \in S_m^*$ then K_* can be represented as;

$$K_* = Z^n + \sum a_i K_i = \left(K + \sum a_i K_i^t\right) + \sum a_i K_i^t \quad \text{where} \quad K \in S_1 \tag{5}$$

So clearly, the two components in the above Equation (2) are independent. The best linear estimators for K_m can be evaluated as a projection over the two subspaces S_1 and S_* Such that the dispersion given by $\text{disp}\left(K_m - P_{B_m}^K\right)$ is minimized so that;

$$K_m^* = P_{S_m^*}^{K_m} = P_{S_1}^{K_m} + P_{S_*}^{K_m} = K_m + P_{S_m^*}^{K_m} \tag{6}$$

When *n* is assumed to be finite large data, so that the coefficients $\left\{a_v : v \ge m+1\right\}$ are estimated such that the dispersion error of the estimate is minimized. This is achieved as follows:

We use Equations (2) and (3) above to estimate the dispersion, such that the

disp $\left\{K_m - P_{S_m^*}^{K_m}\right\}$ is minimized *i.e.*

$$K_m^* = P_{S_m^*}^{K_m} = P_{S_1}^{K_m} + P_{S_*}^{K_m} = K_m + P_{S_m^*}^{K_m} \tag{7}$$

But

$$P_{S_m^*}^{K_m} = \left\{\sum_{v=m+1}^{n} \xi_v \left(K_v - K_v\right); \text{disp}\left(K_m - P_{S_m^*}^{K_m}\right)\right\} \tag{8}$$

Squaring both sides and taking the expectations, we obtain the dispersion error as;

$$\text{disp } X_m = E\left(K_m - K_m^*\right) = \left\{\left(K_m - \hat{K}_m\right) - \sum_{v=m+1}^{n} \xi_v \left(K_v - \hat{K}_m\right)\right\}^2 \tag{9}$$

By minimizing the dispersion with respect to the coefficients (differentiating with respect to ξ_v and solving for ξ_v), we should obtain the coefficients ξ_v, for $v \geq m+1$, which are used in estimating the missing values. The missing value at point k_v is estimated as;

$$\hat{K}_m^* = \hat{k}_m + \sum_{v=m+1}^{n} \xi_v \left(k_v - \hat{k}_v\right) \tag{10}$$

3. Methodology

The study derived some imputation estimators for some ARMA time series models that had lower orders of GARCH innovations. The study considered developing the estimators for ARMA (1, 1) + GARCH (1, 1) and ARMA (2, 2) + (2, 2) processes. The derived imputation estimators were used in imputation of missing data. The study utilized some simulated data with missing values for some imputation techniques. Besides, the study intended to carry out some comparisons between the developed imputation estimators against the convectional imputation techniques. The imputation performance on comparison was measured for all the imputation techniques used in the study.

3.1. Method of Optimal Linear Interpolation

The suggested method for estimation of imputation estimators in this paper was the optimal linear interpolation. The method was used to derive optimal estimators for some ARMA models with assumptions about GARCH innovations. The optimal estimation technique minimized the dispersion error of the ARMA model process.

3.2. Data

The study utilized the generated data from R statistical software version 4.4.2. A thousand (1000) samples were randomly generated for each process of ARMA (1, 1) + GARCH (1, 1) and ARMA (2, 2) + GARCH (2, 2) for the study.

3.3. Data Amputation and Missing Values Generation

The missing values were created into the synthetic data generated by R software. The process of creating missing values in the available data is also known as data

amputation. Appropriate packages like forecast, gglot2, imputeTS and zoo, were included in the R scripts to carry out all the required amputation and imputation for the simulated time series data.

3.4. Measurements of Model Performance

In this study, imputation efficiencies of different imputation techniques were evaluated to check how well the models predicted the imputation estimates using the five different criteria. The imputation techniques included Mean Error (ME), Root Mean Squared Error (RMSE), Mean Absolute Error (MAE), Mean Percentage Error (MPE), Mean Absolute Percentage Error (MAPE).

4. Results and Discussions

4.1. Derivation of Imputation Estimators with GARCH Innovations

4.1.1. An ARMA (1, 1) with Model with GARCH (1, 1) Innovations

The stationary ARMA (1, 1) model with GARCH (1, 1) Innovations is given by, $x_t = \varphi_1 x_{t-1} + \theta_1 \varepsilon_{t-1} + \varepsilon_t$ where $\varepsilon_t = \eta_t \delta_t$, such that

$$\delta_t \sim iid\ t(0,1) \tag{11}$$

$$\text{and} \quad \Upsilon_\Psi = \eta^2_{t|t-1} = \omega + \alpha_1 e^2_{t-1} + \beta_1 \sigma^2_{t-1}$$

Theorem 1

The optimal imputation interpolation estimator for ARMA (1, 1) process with GARCH (1, 1) innovations is given by,

$$x_t = \varphi_1 x_{t-1} + \theta_1 e_{t-1} + \sum_{v=m+1}^{n} \frac{\theta_1 + \varphi_1}{\left(\theta_1 + \varphi_1\right)^{2(v-m)} + 1}\left(k_v - \hat{k}_v\right)$$

Proof

The stationery ARMA (1, 1) model is given by, $x_t = \varphi_1 x_{t-1} + \theta_1 \varepsilon_{t-1} + \varepsilon_t$ where $\varepsilon_t = \eta_t \delta_t$, such that $\delta_t \sim iid(0,1)$ and

$$\Upsilon_\Psi = \eta^2_{t|} = \omega + \alpha_1 e^2_{t-1} + \beta_1 \sigma^2_{t-1} \tag{12}$$

The recursive form of the process in the Equation (12) is given by,

$$x_t = \sum_{j=1}^{\infty}\left[\prod_{i=1}^{j}\left(\theta_1 + \varphi_1\right)\varepsilon_{t-1}\right] + \varepsilon_t \tag{13}$$

The r^{th}-step future ahead predictor for Equation (13) is given as

$$x_{t+r} = \sum_{j=1}^{\infty}\left[\prod_{i=1}^{j}\left(\theta_1 + \varphi_1\right)\varepsilon_{t+r-1}\right] + \varepsilon_{t+r} \tag{14}$$

The error of the predictor form of Equation (14) can be given as

$$x_{t+r} - \hat{x}_{t+r} = \sum_{j=1}^{r-1}\left[\prod_{i=1}^{j}\left(\theta_1 + \varphi_1\right)\varepsilon_{t+r-1}\right] + \varepsilon_{t+r} \tag{15}$$

Or if we set $v = t + r$ we can have the predictor as

$$x_v - \hat{x}_v = \sum_{j=1}^{r-1}\left[\prod_{i=1}^{j}\left(\theta_1 + \varphi_1\right)\varepsilon_{v-1}\right] + \varepsilon_v \tag{16}$$

The dispersion is given by the following expression

$$\begin{aligned}\operatorname{disp}\hat{x}_m &= E\left[x_m-\hat{x}_m\right]^2-2E\sum_{v=m+1}^{n}\xi_v\left[\left(x_m-\hat{x}_m\right)\left(x_v-\hat{x}_v\right)\right]\\&\quad+E\sum_{v=m+1}^{n}\left[\xi_v\left(x_v-\hat{x}_v\right)\right]^2\end{aligned}\tag{17}$$

Substituting the above Equation (16) into the Equation (17) then, it can be given that the dispersion for the above process is,

$$\begin{aligned}\operatorname{disp}\hat{x}_m &= E\left[x_m-\hat{x}_m\right]^2-2E\sum_{v=m+1}^{n}\xi_v\left[e_m\left(\sum_{j=1}^{r-1}\left[\prod_{i=1}^{j}\left(\theta_1+\varphi_1\right)\varepsilon_{v-1}\right]+\varepsilon_v\right)\right]\\&\quad+E\sum_{v=m+1}^{n}\left[\xi_v\left(\sum_{j=1}^{r-1}\left[\prod_{i=1}^{j}\left(\theta_1+\varphi_1\right)\varepsilon_{v-1}\right]+\varepsilon_v\right)\right]^2\end{aligned}\tag{18}$$

The above Equation (18) can be simplified to its equivalent terms so that we can have,

The first term can be evaluated to be,

$$E\left[x_m-\hat{x}_m\right]^2=\hat{\varepsilon}_m^2=\Upsilon_\Psi$$

The second term is given as

$$-2E\sum_{v=m+1}^{n}\xi_v\left[\varepsilon_m\left(\sum_{j=1}^{r-1}\left[\prod_{i=1}^{j}\left(\theta_1+\varphi_1\right)\varepsilon_{v-1}\right]+\varepsilon_v\right)\right]$$

Which can be written as,

$$\begin{aligned}&-2E\left[\xi_{m+1}\left(\theta_1+\varphi_1\right)\varepsilon_m^2+\xi_{m+1}\varepsilon_{m+1}+\xi_{m+2}\left(\theta_1+\varphi_1\right)^2\varepsilon_m\varepsilon_{m+1}+\xi_{m+2}\varepsilon_{m+2}\right.\\&\left.+\xi_{m+3}\left(\theta_1+\varphi_1\right)^3\varepsilon_m\varepsilon_{m+2}+\xi_{m+3}\varepsilon_{m+3}+\cdots\right]\end{aligned}$$

Which can be simplified to be

$$-2\left[\xi_{m+1}\left(\theta_1+\varphi_1\right)\Upsilon_\Psi\right]$$

The third term can be given by

$$+E\sum_{v=m+1}^{n}\left[\xi_v\left(\sum_{j=1}^{r-1}\left[\prod_{i=1}^{j}\left(\theta_1+\varphi_1\right)\varepsilon_{v-1}\right]+\varepsilon_v\right)\right]^2$$

The third term above can be evaluated and written as,

$$\begin{aligned}&+E\left[\xi_{m+1}^2\left(\theta_1+\varphi_1\right)^2\varepsilon_m^2+\xi_{m+1}^2\varepsilon_{m+1}^2+\xi_{m+2}^2\left(\theta_1+\varphi_1\right)^4\varepsilon_{m+1}^2+\xi_{m+2}^2\varepsilon_{m+2}^2\right.\\&\left.+\xi_{m+3}^2\left(\theta_1+\varphi_1\right)^6\varepsilon_{m+2}^2+\xi_{m+3}^2\varepsilon_{m+3}^2+\cdots\right]\end{aligned}$$

Which can be simplified to be,

$$+\sum_{v=m+1}^{n}\xi_v^2\Upsilon_\Psi\left[\left(\theta_1+\varphi_1\right)^{2(v-m)}+1\right]$$

Putting together all the terms of Equation (18) so that we have the generalized dispersion equation given by,

$$\text{disp}\,\hat{x}_m = \Upsilon_\Psi - 2\left[\xi_{m+1}\left(\theta_1+\varphi_1\right)h_\Psi\right] + \sum_{v=m+1}^{n}\xi_v^2\Upsilon_\Psi\left[\left(\theta_1+\varphi_1\right)^{2(v-m)}+1\right] \tag{19}$$

Differentiating Equation (19) with respect to ξ_v and making ξ_v to be the subject of the formula, then it can be obtained that,

$$\xi_v = \frac{\theta_1+\varphi_1}{\left(\theta_1+\varphi_1\right)^{2(v-m)}+1} \tag{20}$$

Substituting the estimated Equation (20) into the interpolation Equation (10) given by the above lemma. It can be found that the imputation estimator of ARMA (1, 1) with GARCH (1, 1) innovations can be given by,

$$\hat{x}_t = \varphi_1 x_{t-1} + \theta_1 e_{t-1} + \sum_{v=m+1}^{n}\frac{\theta_1+\varphi_1}{\left(\theta_1+\varphi_1\right)^{2(v-m)}+1}\left(k_v - \hat{k}_v\right)$$

4.1.2. An ARMA (2, 2) with Model with GARCH (2, 2) Innovations

The stationary ARMA (2, 2) model with GARCH (2, 2) Innovations is given by,

$$x_t = \varphi_2 x_{t-2} + \varphi_1 x_{t-1} + \theta_2\varepsilon_{t-2} + \theta_1\varepsilon_{t-1} + \varepsilon_t \quad \text{where} \quad \varepsilon_t = \eta_t\delta_t, \tag{21}$$

such that $\delta_t \sim iid\ t(0,1)$ and $\Upsilon_\Psi = \eta_{t|}^2 = \omega + \alpha_1 e_{t-1}^2 + \alpha_2 e_{t-2}^2 + \beta_1\sigma_{t-1}^2 + \beta_2\sigma_{t-2}^2$

Theorem 2

The optimal imputation interpolation estimator for ARMA (2, 2) process with GARCH (2, 2) innovation is given by,

$$\hat{x}_t = \varphi_2 x_{t-2} + \varphi_1 x_{t-1} + \theta_2\varepsilon_{t-2} + \theta_1\varepsilon_{t-1} + \sum_{v=m+1}^{n}\frac{\left(\varphi_2+\theta_2\right)+\left(\varphi_1+\theta_1\right)^2}{\left(\varphi_2+\theta_2\right)^{2(v-m)}+\left(\varphi_1+\theta_1\right)^2+1}\left(k_v-\hat{k}_v\right) \tag{22}$$

Proof

For the purposes of effective computation and evaluation, the ARMA (2, 2) process is always written as,

$$x_t = \left(\varphi_2+\theta_2\right)\varepsilon_{t-2} + \left(\varphi_1+\theta_1\right)\varepsilon_{t-1} + \varepsilon_t$$

where $\varepsilon_t = \eta_t\delta_t$ so that $\delta_t \sim iid\left(0,1\right)$ and let

$$h_\Psi = \eta_{t|t-1}^2 = \omega + \alpha_1 e_{t-1}^2 + \alpha_2 e_{t-2}^2 + \beta_1\sigma_{t-1}^2 + \beta_2\sigma_{t-2}^2 \tag{22}$$

The recursive form of the above Equation (22) is given by,

$$x_t = \sum_{i=1}^{\infty}\left[\prod_{j=1}^{i}\left(\varphi_2+\theta_2\right)\varepsilon_{t-2j}\right] + \sum_{k=1}^{\infty}\left[\prod_{i=1}^{k}\left(\varphi_1+\theta_1\right)\varepsilon_{t-i}\right] + \varepsilon_t \tag{24}$$

The r^{th}-step ahead predictor of Equation (24) can be given by the following expression,

$$x_{t+r} = \sum_{i=1}^{\infty}\left[\prod_{j=1}^{i}\left(\varphi_2+\theta_2\right)\varepsilon_{t+r-2j}\right] + \sum_{k=1}^{\infty}\left[\prod_{i=1}^{k}\left(\varphi_1+\theta_1\right)\varepsilon_{t+r-i}\right] + \varepsilon_{t+r} \tag{25}$$

If we let $v = t+r$, we can re-write the above Equation (25) as,

$$x_v = \sum_{i=1}^{\infty}\left[\prod_{j=1}^{i}\left(\varphi_2+\theta_2\right)\varepsilon_{v-2j}\right] + \sum_{k=1}^{\infty}\left[\prod_{i=1}^{k}\left(\varphi_1+\theta_1\right)\varepsilon_{v-i}\right] + \varepsilon_v \tag{26}$$

It can be written that the predicted error for Equation (26) is given by,

$$x_v - \hat{x}_v = \sum_{i=1}^{r-1}\left[\prod_{j=1}^{i}(\varphi_2+\theta_2)\varepsilon_{v-2j}\right] + \sum_{k=1}^{r-1}\left[\prod_{i=1}^{k}(\varphi_1+\theta_1)\varepsilon_{v-i}\right] + \varepsilon_v \tag{27}$$

From the lemma above, the dispersion error is given by,

$$\begin{aligned}\operatorname{disp}\hat{x}_m = {} & E\left[x_m - \hat{x}_m\right]^2 - 2E\sum_{v=m+1}^{n}\xi_v\left[\left(x_m-\hat{x}_m\right)\left(x_v-\hat{x}_v\right)\right] \\ & + E\sum_{v=m+1}^{n}\left[\xi_v\left(x_v-\hat{x}_v\right)\right]^2\end{aligned} \tag{28}$$

Substituting Equation (27) into the dispersion Equation (28), it can be obtained that,

$$\begin{aligned}\operatorname{disp}\hat{x}_m = {} & E\left[x_m-\hat{x}_m\right]^2 \\ & -2E\sum_{v=m+1}^{n}\xi_v\left[\varepsilon_m\sum_{i=1}^{r-1}\left[\prod_{j=1}^{i}(\varphi_2+\theta_2)\varepsilon_{v-2j}\right]+\sum_{k=1}^{r-1}\left[\prod_{i=1}^{k}(\varphi_1+\theta_1)\varepsilon_{v-i}\right]+\varepsilon_v\right] \\ & +E\left[\sum_{V=m+1}^{n}\xi_v\left\{\sum_{i=1}^{r-1}\left[\prod_{j=1}^{i}(\varphi_2+\theta_2)\varepsilon_{v-2j}\right]+\sum_{k=1}^{r-1}\left[\prod_{i=1}^{k}(\varphi_1+\theta_1)\varepsilon_{v-i}\right]+\varepsilon_v\right\}\right]^2\end{aligned} \tag{29}$$

Evaluating further the above Equation (29) under the dispersion, then it can be obtained that,

The first term is $E\left[x_m-\hat{x}_m\right]^2 = E\left(\varepsilon_m\right)^2 = h_\Psi$

The second term can be evaluated to be,

$$-2E\sum_{v=m+1}^{n}\xi_v\left[\varepsilon_m\sum_{i=1}^{r-1}\left[\prod_{j=1}^{i}(\varphi_2+\theta_2)\varepsilon_{v-2j}\right]+\sum_{k=1}^{r-1}\left[\prod_{i=1}^{k}(\varphi_1+\theta_1)\varepsilon_{v-i}\right]+\varepsilon_v\right]$$

Again, the second term can be simplified further to be,

$$\begin{aligned}& -2E\varepsilon_m\left[\xi_{m+1}\left\{(\varphi_2+\theta_2)\varepsilon_{m-1}+(\varphi_1+\theta_1)\varepsilon_m+\varepsilon_{m+1}\right\}\right. \\ & +\xi_{m+2}\left\{(\varphi_2+\theta_2)^2\varepsilon_m+(\varphi_1+\theta_1)^2\varepsilon_{m+1}+\varepsilon_{m+2}\right\} \\ & \left.+\xi_{m+3}\left\{(\varphi_1+\theta_1)^3\varepsilon_{m-1}+(\varphi_1+\theta_1)^3\varepsilon_m+\varepsilon_{m+1}\right\}+\cdots\right]\end{aligned}$$

Which can then be simplified further to achieve,

$$-2E\left[\xi_{m+1}\left\{(\varphi_1+\theta_1)\varepsilon_m^2\right\}+\xi_{m+2}\left\{(\varphi_1+\theta_1)^2\varepsilon_m^2\right\}\right]$$

It can also be expressed as

$$-2h_\Psi\left[\xi_{m+1}\left\{(\varphi_1+\theta_1)\right\}+\xi_{m+2}\left\{(\varphi_1+\theta_1)^2\right\}\right]$$

The third term can be given by,

$$E\left[\sum_{V=m+1}^{n}\xi_v\left\{\sum_{i=1}^{r-1}\left[\prod_{j=1}^{i}(\varphi_2+\theta_2)\varepsilon_{v-2j}\right]+\sum_{k=1}^{r-1}\left[\prod_{i=1}^{k}(\varphi_1+\theta_1)\varepsilon_{v-i}\right]+\varepsilon_v\right\}\right]^2$$

It can also be evaluated to be

$$\begin{aligned}& \xi_{m+1}^2\left[(\varphi_2+\theta_2)\varepsilon_{m-1}+(\varphi_1+\theta_1)\varepsilon_m+\varepsilon_{m+1}\right]^2 \\ & +\xi_{m+2}^2\left[(\varphi_2+\theta_2)^2\varepsilon_m+(\varphi_1+\theta_1)^2\varepsilon_{m+1}+\varepsilon_{m+2}\right]^2\end{aligned}$$

$$+\xi_{m+3}^2\left[\left(\varphi_1+\theta_1\right)^3\varepsilon_{m-1}+\left(\varphi_1+\theta_1\right)^3\varepsilon_m+\varepsilon_{m+1}\right]^2+\cdots$$

And it can be simplified to be

$$E\left(\sum_{v=m+1}^{n}\xi_{m+1}^2\left(\varphi_2+\theta_2\right)^{2(v-m)}\varepsilon_{v-2}^2+\left(\varphi_1+\theta_1\right)^{v-m}\varepsilon_{v-1}^2+\sum_{v=m+1}^{n}\xi_v^2\varepsilon_v^2\right)$$

Substituting the distributional assumptions into the error terms, then the above term can be written as,

$$h_\Psi\left(\sum_{v=m+1}^{n}\xi_v^2\left(\varphi_2+\theta_2\right)^{2(v-m)}+\xi_v^2\left(\varphi_1+\theta_1\right)^{v-m}+\sum_{v=m+1}^{n}\xi_v^2\right)$$

Putting together all the terms of the above simplifications, the dispersion can be given as,

$$\begin{aligned}\operatorname{disp} x_m &= h_\Psi-2h_\Psi\left[\xi_{m+1}\left\{\left(\varphi_2+\theta_2\right)\right\}+\xi_{m+2}\left\{\left(\varphi_1+\theta_1\right)^2\right\}\right]\\&\quad+h_\Psi\left(\sum_{v=m+1}^{n}\xi_v^2\left(\varphi_2+\theta_2\right)^{2(v-m)}+\xi_v^2\left(\varphi_1+\theta_1\right)^{v-m}+\sum_{v=m+1}^{n}\xi_v^2\right)\end{aligned}\tag{30}$$

Differentiating Equation (30) with respect ξ_v, and setting ξ_v to be subject of the formula then it can be obtained that,

$$\xi_v=\frac{\left(\varphi_2+\theta_2\right)+\left(\varphi_1+\theta_1\right)^2}{\left(\varphi_2+\theta_2\right)^{2(v-m)}+\left(\varphi_1+\theta_1\right)^2+1}\tag{31}$$

Substituting the above estimated Equation (31) into the interpolation Equation (10) given by the above lemma. It can be obtained that the interpolation estimator for the ARMA (2, 2) with GARCH (2, 2) innovations can be obtained as,

$$\begin{aligned}\hat{x}_t &= \varphi_2x_{t-2}+\varphi_1x_{t-1}+\theta_2\varepsilon_{t-2}+\theta_1\varepsilon_{t-1}\\&\quad+\sum_{v=m+1}^{n}\frac{\left(\varphi_2+\theta_2\right)+\left(\varphi_1+\theta_1\right)^2}{\left(\varphi_2+\theta_2\right)^{2(v-m)}+\left(\varphi_1+\theta_1\right)^2+1}\left(k_v-\hat{k}_v\right)\end{aligned}$$

4.2. Simulation Illustrations and Results

In this section, we underlined how simulation was carried out to achieve the required results. We wrote the R-codes using R studio (Version 4.4.2). The running and performance of the codes were executed through the following: forecast, gglot2, imputeTS and zoo packages. The R-codes performed the following:

1) Generated some data sets of samples (1000) from ARMA processes with GARCH specifications. The code created the ACF, and the residual and normality check of the simulated data.

2) To amputed the generated data in the above step (i) to have (100) missing values and visualize the missing data using plots.

3) Each imputation technique (Linear interpolation, ANN, KNN and Kalman imputations) was coded and trained to predict, replace the missing values using the generated data samples and to visualize the imputed data for every technique.

4) Imputation performance and efficiency was calculated for every imputation technique in R studio using "R-package metrics".

4.2.1. Table and Figures Visualizations

Visualization of the plots and tables of the simulated data are given in this sub section. The plots provided were created from the data for the two models of

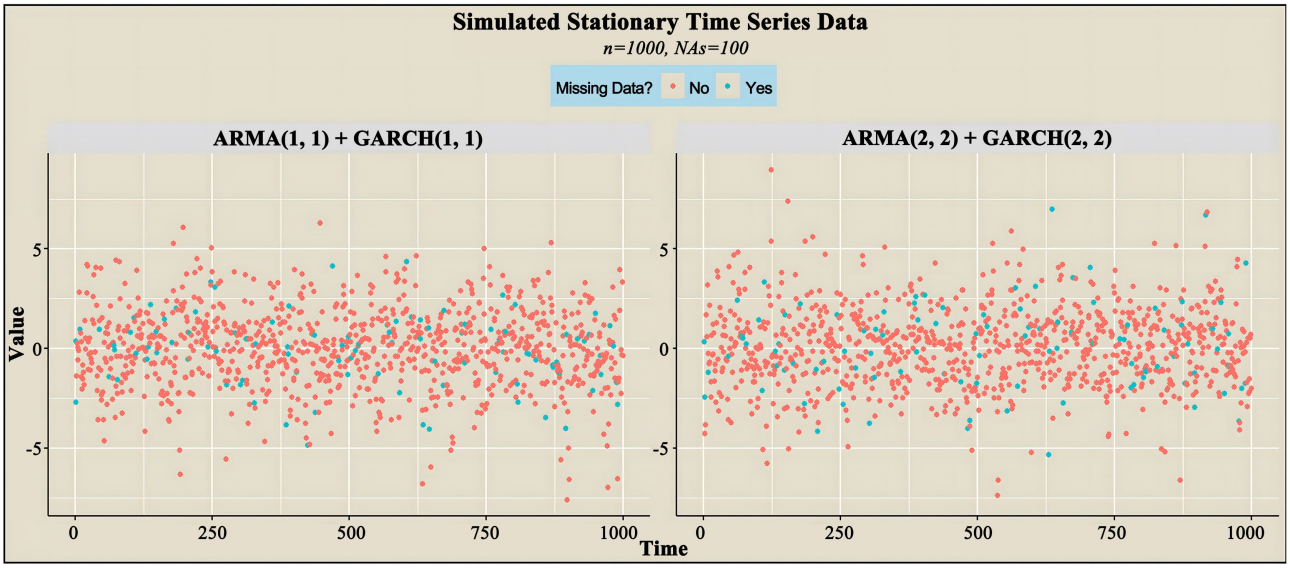


Figure 1. Plot of simulated stationary data with missing values.

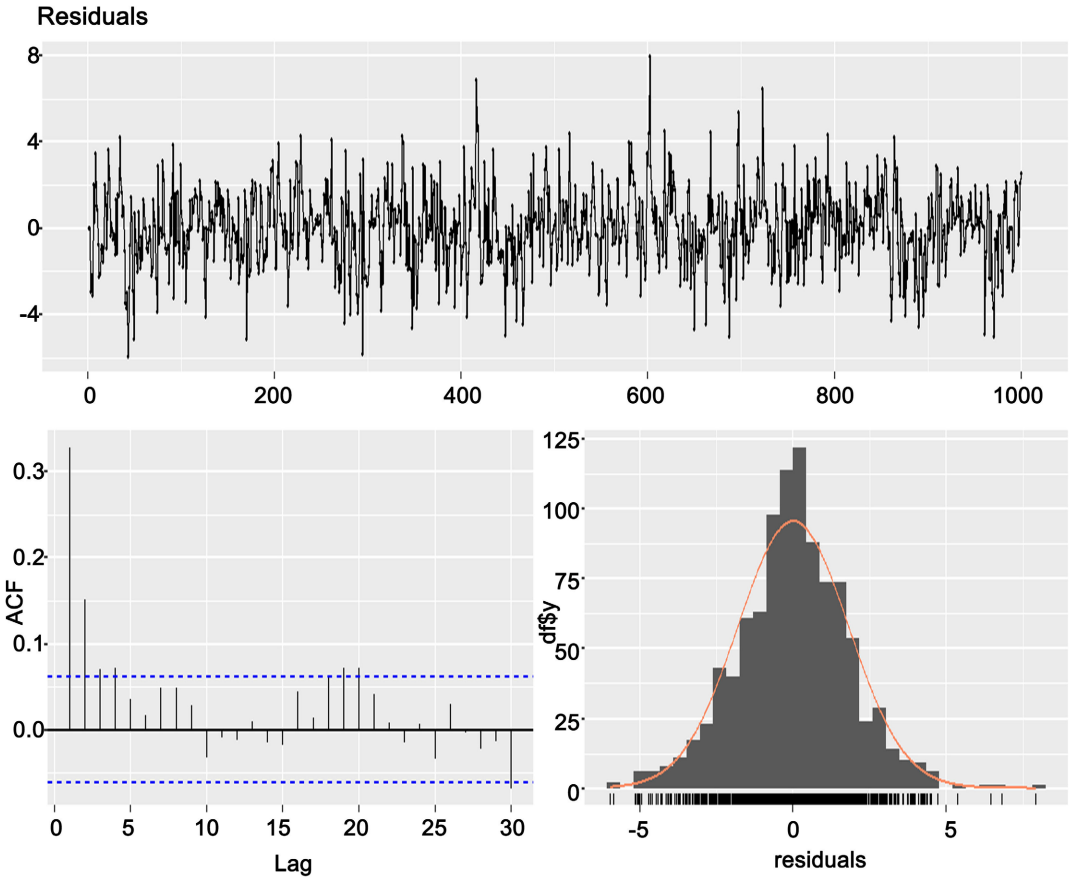


Figure 2. The residuals and the ACF plots of the simulated data for ARMA (2, 2) + GARCH (2, 2) model.

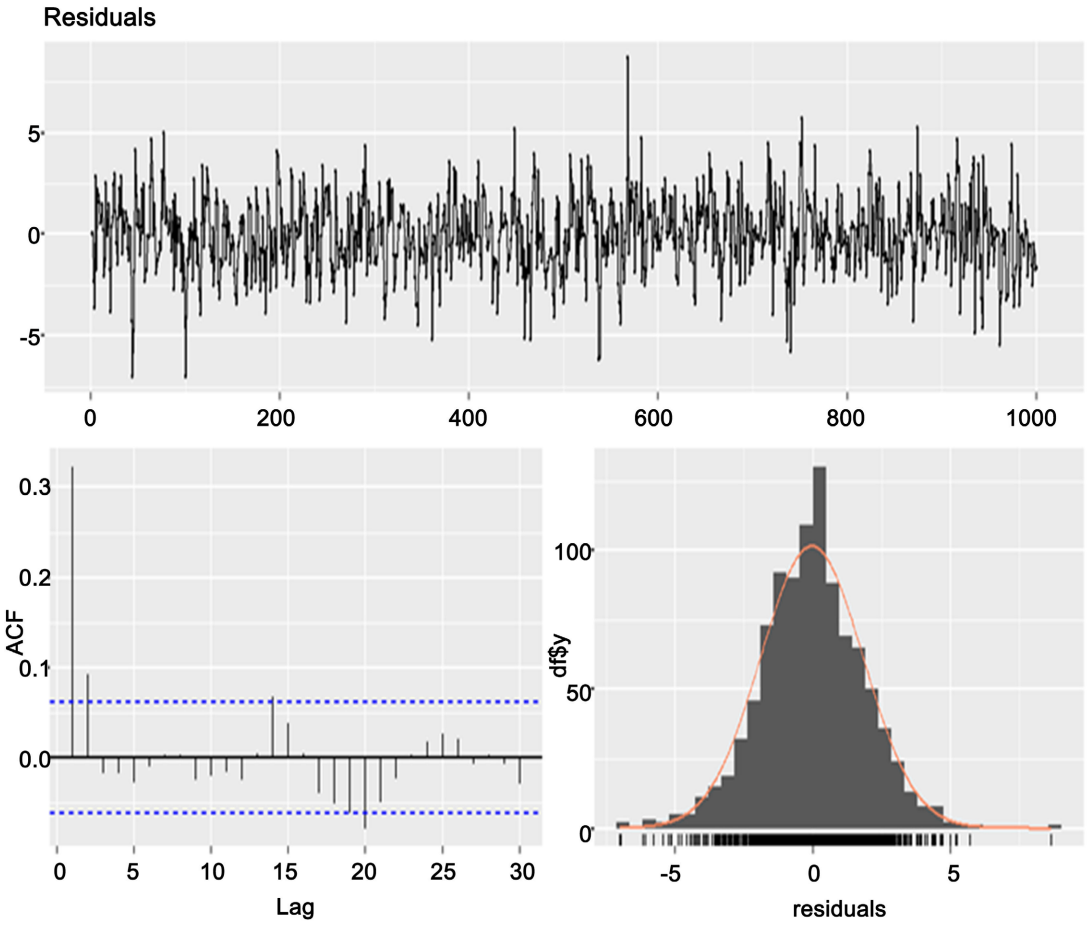


Figure 3. The residuals and the ACF plots of the simulated data for ARMA (1, 1) + GARCH (1, 1) model.

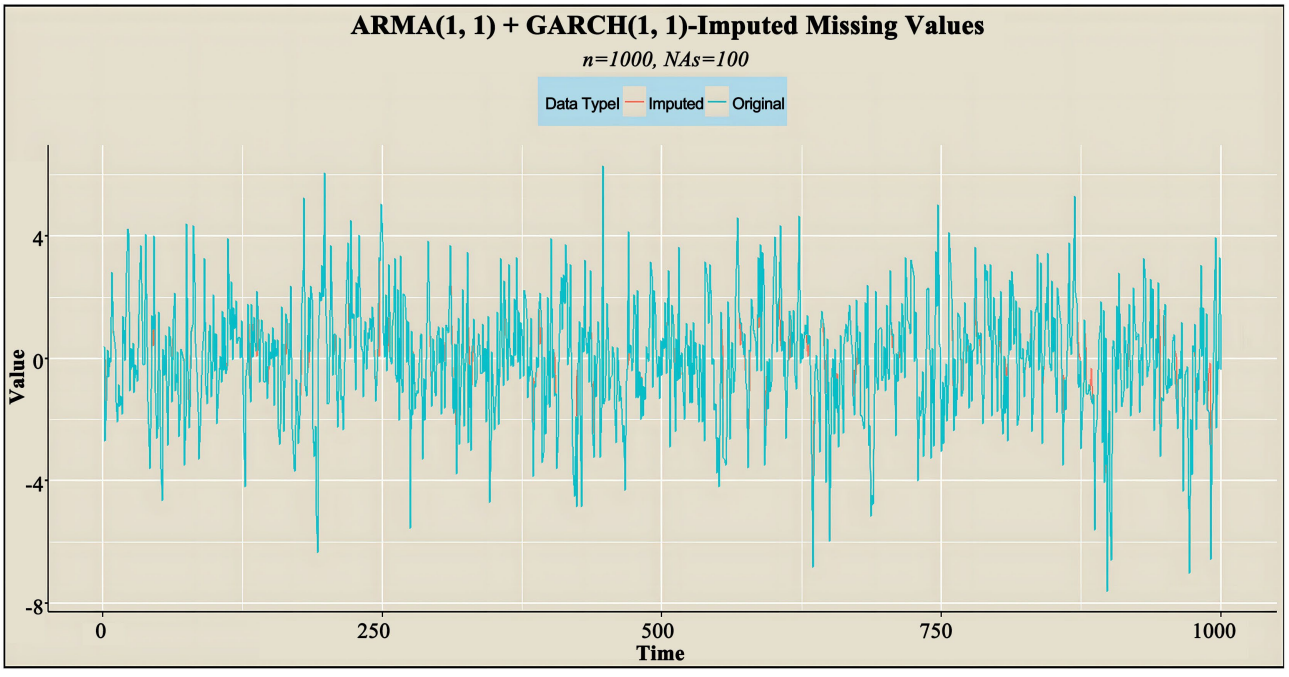


Figure 4. A plot of an imputed missing data using estimated model for ARMA (1, 1) + GARCH (1, 1).

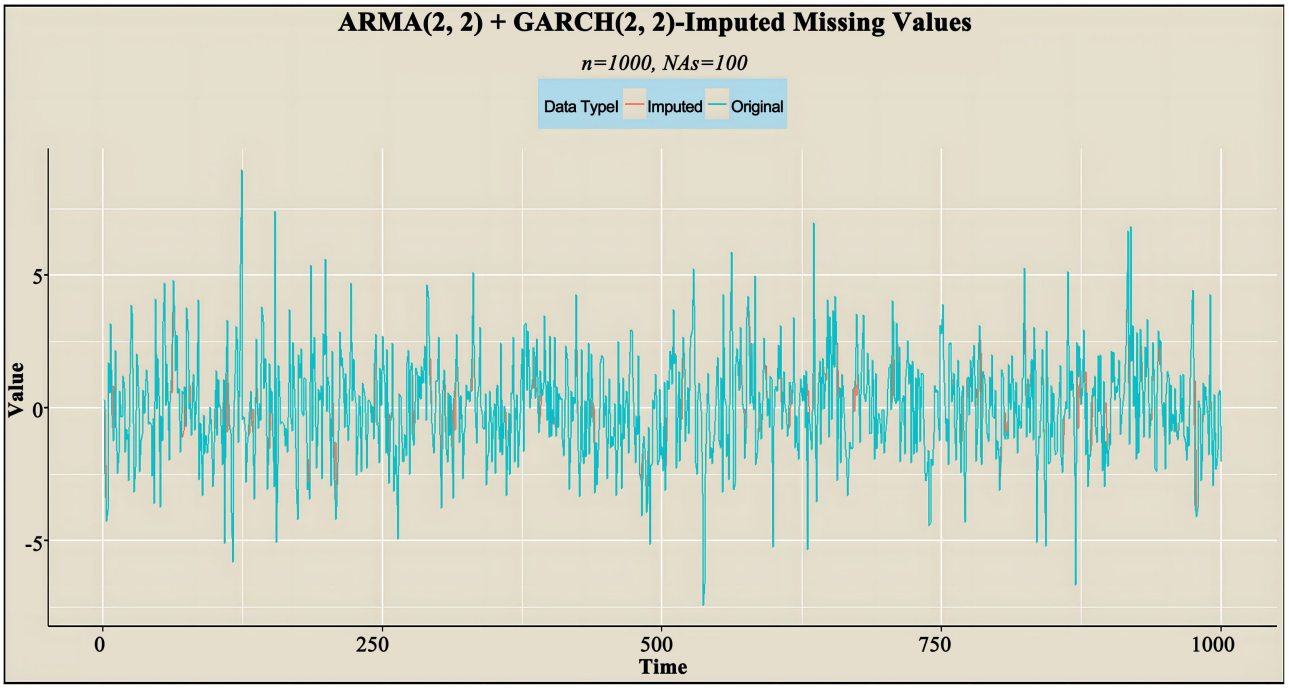


Figure 5. A plot of an imputed missing data using estimated model for ARMA (2, 2) + GARCH (2, 2).

ARMA (1, 1) + GARCH (1, 1) and that of ARMA (2, 2) + GARCH (2, 2). We also had some autocorrelation functions (ACF) and the residuals for the data sets used in the study (See **Figures 1-5**).

4.2.2. Validation of Imputation Methods

In the above **Table 1**, the results were obtained from the simulated data for ARMA (1, 1) + GARCH (1, 1) process. The results measured the imputation performance of the four imputation techniques namely, Optimal linear interpolation (OLI), Artificial Neural Networks (ANNs), K- Nearest Neighbors (KNN), and the Kalman filters technique. It is evident that the KNN, ANN and Optimal linear interpolation techniques have desired results across the performance metrics used. The metrics used were Mean Error (ME), root mean squared error (RMSE), Mean absolute error (MAE), Mean percentage error (MPE). Comparison made from the results of **Table 1**, is that the KNN was the best technique for imputing missing values, it had the lowest values of ME = −0.01, RMASE = 0.53, compared to ANN with ME = 0.18, RMASE = 0.52 while the Optimal linear interpolation technique came at the third position with ME = −0.02, RMSE = 0.51. Imputation performance metric values of ANN and optimal linear interpolation were so close to one another. Kalman was the poorest in estimating missing values with this type of data, it had the highest values by its metrics.

The imputation performance of **Table 2** above on ARMA (2, 2) + GARCH (2, 2) simulated data was as follows: the optimal linear interpolation was the second-best technique after the ANN technique. This is illustrated by lower values of performance techniques for optimal interpolation estimation ME = −0.0063, RMSE

= 0.66 and MAPE = 20.25%, while the ANN had ME = −0.05, RMSE = 0.62 and MAPE = 16%. The imputation performance of Optimal linear interpolation and ANN were very close. The Kalman filter imputation was the poorest technique for imputation. This was evident by higher numbers from performance metrics e.g., ME = −0.0009, MPE = 15.02% higher than the values of the other imputation techniques in the study.

Table 1. Imputation performance of the techniques on ARMA (1, 1) + (1, 1) data.

	ME	RMSE	MAE	MPE	MAPE
OLI	−0.02006	0.5160437	0.124112	8.254586	12.97376
ANNs	−0.01814	0.5043942	0.1206323	7.402212	13.85182
KNN	−0.010598	0.5271044	0.130292	10.33006	14.00209
KALMAN	0.01711178	0.26917	0.05734335	−4.33466	22.20235

Table 2. Imputation performance of the techniques on ARMA (2, 2) + (2, 2) data.

	ME	RMSE	MAE	MPE	MAPE
OLI	−0.0063109	0.6644256	0.1685092	11.61931	20.25187
ANN	−0.004829	0.626470	0.153301	0.153301	16.0088
KALMAN	0.000922339	0.662133	0.1612853	15.02621	15.91684

5. Conclusion

The study recommends that the above imputation estimators for ARMA with GARCH errors can be applied to real life data. More comparisons can be made with other imputation techniques.

Acknowledgements

I wish to thank my doctoral advisors for the great way they contributed to the development of this work. Their guidance, suggestions, and alignment of this work have always motivated the success of this paper. I am gratefully indebted to them.

Conflicts of Interest

The authors declare no conflicts of interest.

References

[1] Gao, Y.B., Semiromi, M.T. and Merz, C. (2023) Efficacy of Statistical Algorithms in Imputing Missing Data of Streamflow Discharge Imparted with Variegated Variances and Seasonalities. *Environmental Earth Sciences*, **82**, Article No. 476. https://doi.org/10.1007/s12665-023-11139-z

[2] Nassiuma, D.K. (1994) Symmetric Stable Sequence with Missing Observations. *Journal of Time Series Analysis*, **15**, 313-323. https://doi.org/10.1111/j.1467-9892.1994.tb00196.x

[3] Owili1, P.A., Nassiuma, D. and Orawo, L. (2015) Efficiency of Imputation Techniques

for Missing Values of Pure Bilinear Models with GARCH Innovations. *American Journal of Mathematics and Statistics*, **5**, 316-324.

[4] Engle, R.F. (1982) Autoregressive Conditional Heteroskedasticity with Estimates of the Variance of United Kingdom Inflation. *Econometrica*, **50**, 987-1007. https://doi.org/10.2307/1912773

[5] Bollerslev, T. (1986) Generalized Autoregressive Conditional Heteroskedasticity. *Journal of Econometrics*, **31**, 307-327. https://doi.org/10.1016/0304-4076(86)90063-1

[6] Wee, D.C.H., Chen, F. and Dunsmuir, W.T.M. (2023) Estimating GARCH (1, 1) in the Presence of Missing Data. *The Annals of Applied Statistics*, **17**, 2596-2618. https://doi.org/10.1214/23-AOAS1734

[7] Blasques, F., Gorgi, P. and Koopman, S. (2021) Missing Observations in Observation-Driven Time Series Models. *Journal of Econometrics*, **221**, 542-568. https://doi.org/10.1016/j.jeconom.2020.07.043

[8] Cascone, M.H. and Hotta, L.K. (2018) Quasi-Maximum Likelihood Estimation of GARCH Models in the Presence of Missing Values. *Journal of Statistical Computation and Simulation*, **89**, 292-314. https://doi.org/10.1080/00949655.2018.1546860

[9] Sucarrat, G. and Escribano, A. (2017) Estimation of Log-GARCH Models in the Presence of Zero Returns. *The European Journal of Finance*, **24**, 809-827. https://doi.org/10.1080/1351847X.2017.1336452

[10] Ossandón, S. and Bahamonde, N. (2013) A New Nonlinear Formulation for GARCH Models. *Comptes Rendus Mathematique*, **351**, 235-239. https://doi.org/10.1016/j.crma.2013.02.014

[11] Bondon, P. and Bahamonde, N. (2012) Least Squares Estimation of ARCH Models with Missing Observations. *Journal of Time Series Analysis*, **33**, 880-891. https://doi.org/10.1111/j.1467-9892.2012.00803.x

[12] Owili, P.A. (2016) Imputation of Missing Values for Bilinear Time Series Models. Ph.D. Thesis, Kabarak University Repository.

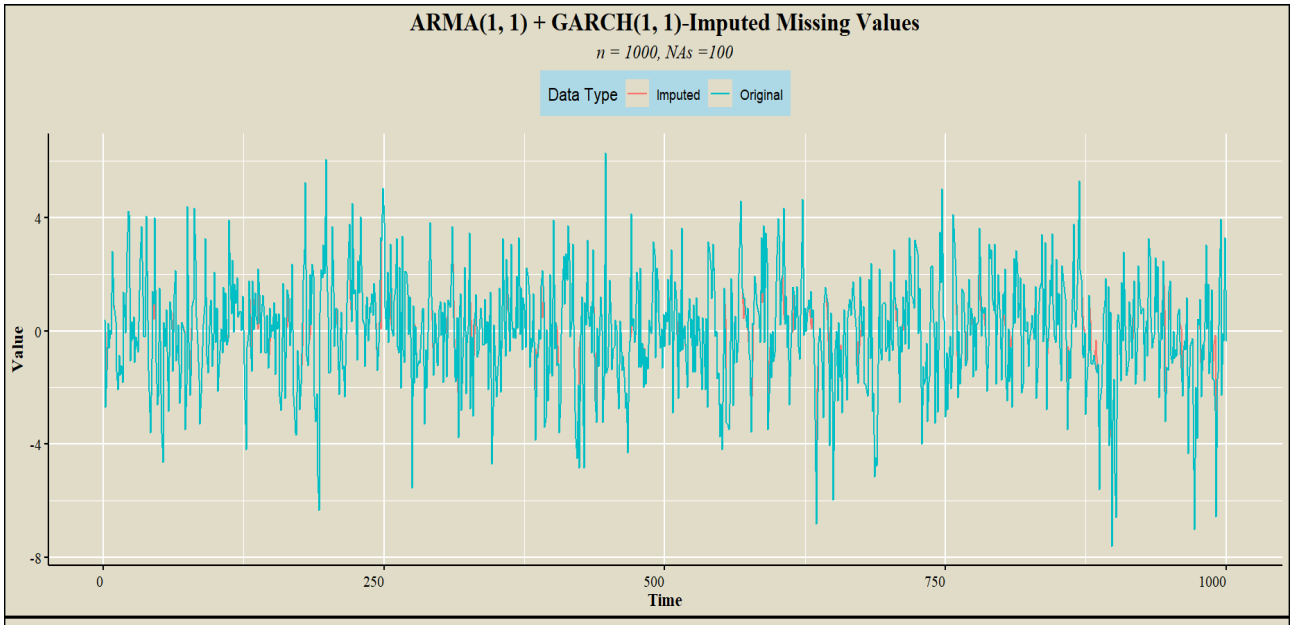
Appendices

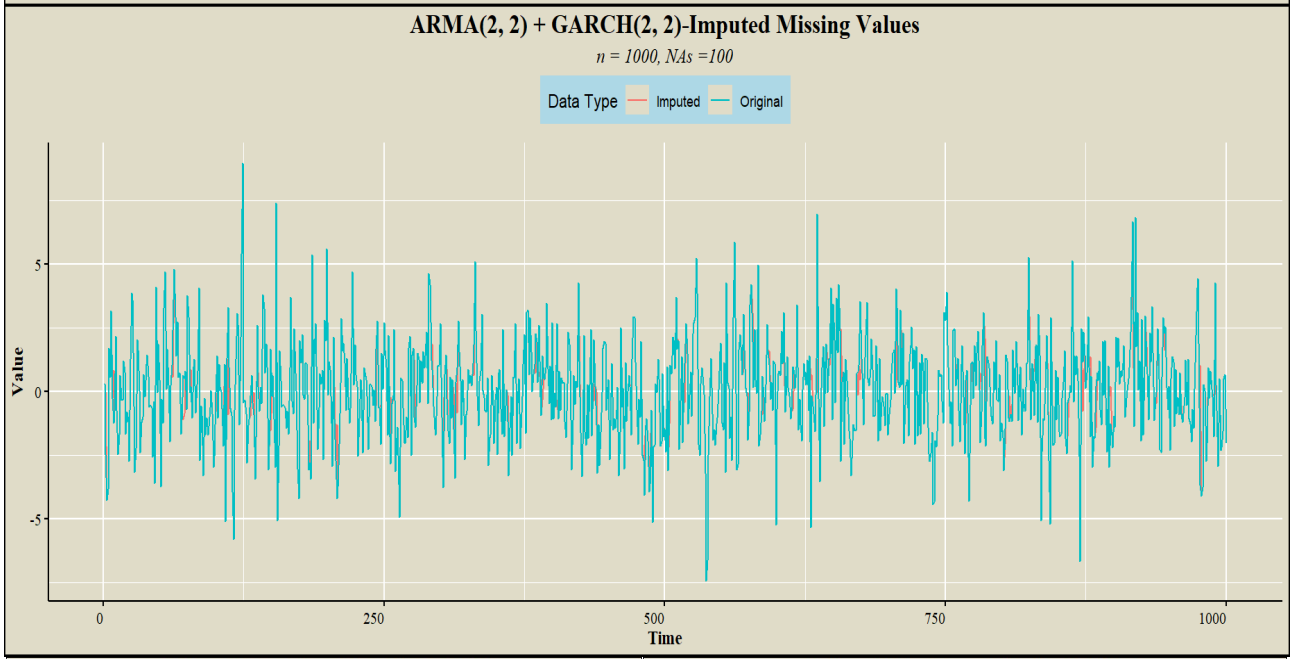
R Script for Data Generation

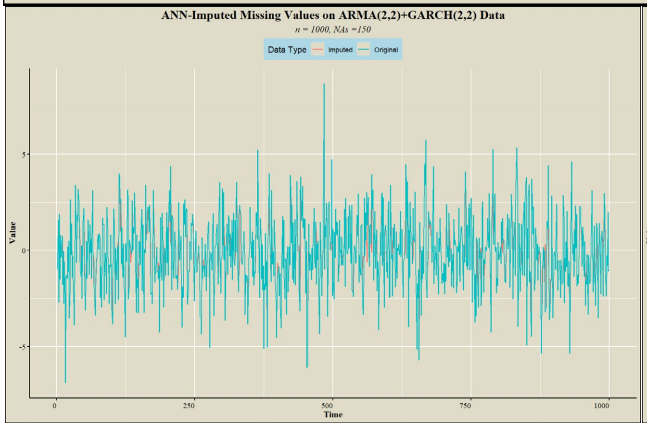
```
# load packages
library(fGarch)
library(forecast)
library(ggplot2)
# set seed for reproducibility
set.seed(44)
# set parameters
n <- 1000 # number of records
NAs <- 100 # number of missing records
# Simulate ARMA(1,1) and ARMA(2,2) process
arma_1_1 <- arima.sim(n = n, model = list(ar = 0.5, ma = 0.3))
arma_2_2 <- arima.sim(n = n, model = list(ar = c(0.5, -0.2),
ma = c(0.3, 0.1)))
# Fit GARCH(1,1) and GARCH(2,2) model to the ARMA residuals
garch_model_1_1 <- garchFit(~ garch(1, 1), data = arma_1_1, trace = FALSE)
garch_model_2_2 <- garchFit(~ garch(2, 2), data = arma_2_2, trace = FALSE)
# Extract the conditional volatility (GARCH model's estimate of volatility)
volatility_1_1 <- volatility(garch_model_1_1)
volatility_2_2 <- volatility(garch_model_2_2)
# Simulate the ARMA(1,1) + GARCH(1,1) process by adding the GARCH vola-
tility
variable_1 <- arma_1_1 + volatility_1_1 * rt(n,15)
variable_2 <- arma_2_2 + volatility_2_2 * rt(n,15)
# Combine into a data frame
simulated_data <- data.frame(index = 1:n,
variable_1 = variable_1,
variable_2 = variable_2)
# introduce missing data at random (data frame), ensuring reproducibility
set.seed(44)
missing_indices_1 <- sample(1:n, NAs)
simulated_data$variable_1_miss <- simulated_data$variable_1
simulated_data$variable_1_miss[missing_indices_1] <- NA
set.seed(44)
missing_indices_2 <- sample(1:n, NAs)
simulated_data$variable_2_miss <- simulated_data$variable_2simu-
lated_data$variable_2_miss[missing_indices_2] <- NA
# for visualization purposes, transform the data to long format
simulated_viz_data <- simulated_data |>
dplyr::mutate(missing_values = dplyr::if_else(!is.na(variable_1_miss), "No",
dplyr::if_else(is.na(variable_1_miss),
"Yes", ""))) |>
```

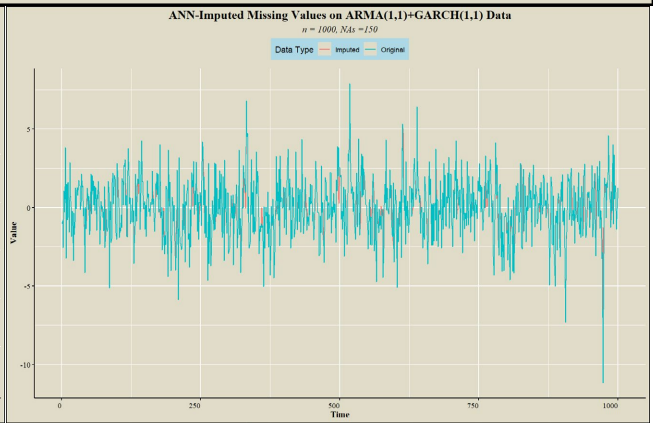
```
dplyr::select(-c(variable_1_miss, variable_2_miss),
"ARMA (1,1) + GARCH (1,1)" = variable_1,
"ARMA (2,2) + GARCH (2,2)" = variable_2) |>
tidyr::pivot_longer(cols = starts_with("ARMA"),
names_to = "variable_names",
values_to = "variable_values")
# custom graph theme
cust_theme <- function(){
theme(plot.title = element_text(face = "bold",
hjust = 0.5,
size = 16,
family = "serif",
color = "black"),
plot.subtitle=element_text(face = 'italic',
hjust = 0.5,
size = 12,
family = "serif",
color = 'black'),
axis.title = element_text(face = "bold",
size = 11.5,
family = "serif",
color = "black"),
axis.text = element_text(face = "plain",
size = 10,
family = "serif",
color = "black"),
strip.text.x = element_text(face = "bold",
size = 13.5,
family = "serif",
color = "black"),
axis.text.x = element_text(angle = 0,
hjust = 1,
vjust = 0.5),
plot.background = element_rect(fill = "#E0DCC8",
color = "black",
linewidth = 1),
panel.background = element_rect(fill = "#E0DCC8"),
axis.line = element_line(color = "black"),
axis.ticks = element_line(color = "black"),
legend.position = "top",
legend.direction = "horizontal",
legend.background = element_rect(fill = "lightblue")
)
```

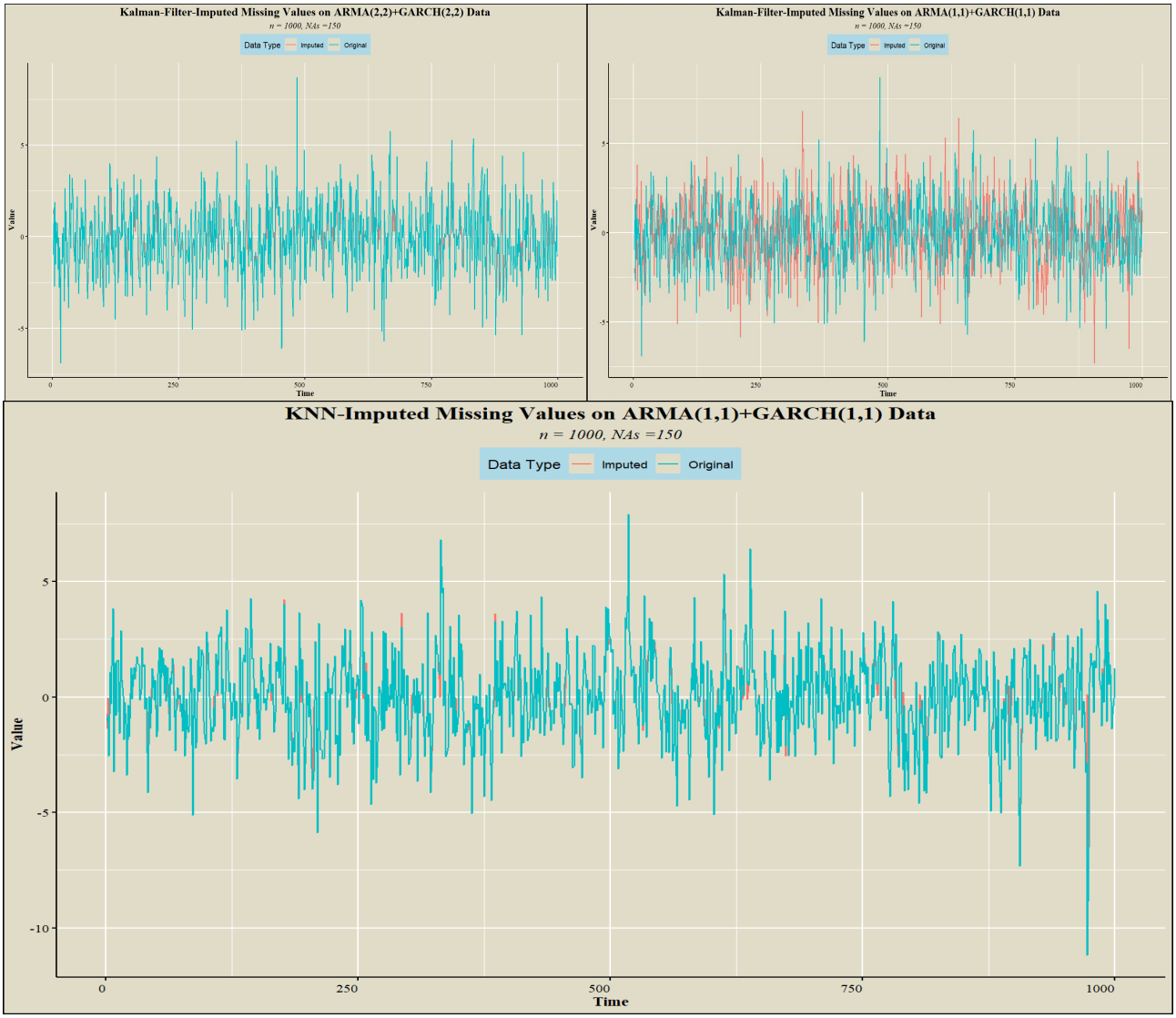
```
}
# visualize the data (both missing and non-missing values)
(line_graph_t_series_miss <- simulated_viz_data |>
ggplot(aes(x = index,
y = variable_values,
color = missing_values)) +
geom_jitter() +
labs(title = "Simulated Stationary Time Series Data",
x = "Time",
y = "Value",
color = "Missing Data?",
subtitle = paste0("n = ", n,
", NAs =", NAs)) +
facet_wrap(~variable_names, axes = "all") +
cust_theme())

# save data into disc
data_dir <- paste0(getwd(), "/SimulatedData")
if(!dir.exists(data_dir)){
dir.create(data_dir)
} else {print("Directory Exists!")}
write.csv(x = simulated_data,
file = paste0(data_dir, "/simulated_t_series_data2.csv"),
row.names = FALSE)
```


ARMA(1, 1) + GARCH(1, 1)-Imputed Missing Values
n = 1000, NAs =100
Data Type
Imputed
Original
Value
Time


ARMA(2, 2) + GARCH(2, 2)-Imputed Missing Values
n = 1000, NAs =100
Data Type
Imputed
Original
Value
Time


ANN-Imputed Missing Values on ARMA(2,2)+GARCH(2,2) Data
n = 1000, NAs =150
Data Type
Imputed
Original
Value
Time


ANN-Imputed Missing Values on ARMA(1,1)+GARCH(1,1) Data
n = 1000, NAs =150
Data Type
Imputed
Original
Value
Time


Kalman-Filter-Imputed Missing Values on ARMA(2,2)+GARCH(2,2) Data
n = 1000, NAs =150
Data Type Imputed Original
Value
Time
Kalman-Filter-Imputed Missing Values on ARMA(1,1)+GARCH(1,1) Data
n = 1000, NAs =150
Data Type Imputed Original
Value
Time
KNN-Imputed Missing Values on ARMA(1,1)+GARCH(1,1) Data
n = 1000, NAs =150
Data Type Imputed Original
Value
Time