On Student's-*t* ARMA Modelling of Missing Values

ABSTRACT

In this paper the study intends to mitigate the missing problem in the context of univariate ARMA time series models. Our main objective of this paper was to derive imputation estimators for ARMA models under the student's-*t* distribution assumptions and evaluate their imputation performance. The study also utilized the method of optimal interpolation criterion of missing values to build the novel imputation estimators for ARMA models. A data set of 1000 samples were generated using statistical R software. One hundred (100) points of missing values ware created within the generated sample data at a random mechanism. The study carried out an imputation of missing values using the developed estimators for ARMA (1,1) and ARMA (1,2) processes. Through this study, it was evident that the estimators ARMA (1,2) did imputed the missing values better than those of ARMA (1,1) process. Besides the development of the imputation estimators, the study did a comparison of the derived imputation estimators of time series with the convectional imputation techniques of missing values. They included K-Nearest Neighbors (KNN), Artificial Neural Networks (ANN) and Kalman filters imputations. The results obtained from the comparison of the simulation study was that the ANN, KNN and the Kalman filters were the best in imputing missing values in time series data. The proposed estimated models of ARMA also did compete well with the convectional techniques used in comparison in this study. The models that the study came up with can be of importance to data scientists, researchers in refilling missing data in time series contexts.

Keywords: Autoregressive Moving Average model, Artificial Neural Networks, Imputation, Kalman filters, Missing Values, Student t distribution, Optimal Linear interpolation

1. INTRODUCTION

An autoregressive moving average or (ARMA) model, is a process of a linear combination of an autoregressive or AR, a moving average or MA and an error or an innovation noise. The (ARMA) process is an important stationary time series model that plays a crucial role in the modeling of time series data. In most cases, the errors' noises for the general (ARMA) models are always assumed to take normal distribution its modelling [1].

Generally, most time series models assume normality in their estimation and predictions. Normality is always taken by these models because of its simplicity and availability of modeling techniques that can handle normality more comfortably [2, 3]. However, there are cases where some practical scenarios may not capture normality. Utilizing normality assumptions to model such cases where normality is not obeyed might not work and hard to handle [4].

Asymmetry and non-normality is an alternative way of handling data that do not obey normality. Asymmetric distribution offers more reliable and robust results when applied to real phenomena with asymmetric, non-normality, skewness and flat tails characteristics [5]. The most utilized distributions to capture asymmetry, flat and long tails, are; Student-*t* distribution, Normal inverse Gaussian and generalized error distribution just to mention a few. The characteristics of these distributions can be adjusted to adapt data that displays, heavy tails, skewness, non-normality and asymmetric features.

During modelling of time series data, researchers normally encounter with missing data. Missing values occurs when a value or data values in a given variable are not given for many reasons. Missing values masks and veils trends and patterns in time series analysis. Missing values has far reaching effects on analysis and forecasting associated to trends and patterns for time series data [6, 7]. Time series analysis does experiences this challenge quite often. Therefore, many researchers, have given rise to numerous ways of recouping missing data in various disciplines, including time series.

One of the ways to deals with missing values is through the deletion of missing parts. The deletion is a way of eliminating missing sets or the variables containing missing values by simply deleting the missing data. However, deletion of missing values has a major setback of eliminating vital information, which adversely impact the prediction, forecasts and eventually interfere with the inference on vital information for decision making [8, 9].

Imputation is the only way out to mitigate missing value problems. Imputation is an important step that should not be skipped in any data analysis process. Imputation enhances cleanliness in data for further handling and processing. A sufficient data imputation algorithm must improve on the data productivity, data analysis, data visualization and data investigations [2].

During imputation of missing data, there are various pertinent issues that need to be taken into consideration ensure adequate restoration of missing data. One of the important characteristics that any data should portray during analysis is normality. In practice, normality might not be realized for various reason. To cope up with asymmetry and non-normality, data can be transformed to make it capture normality. However, this step has been dismissed owing to the fact that vital statistical properties of the transformed data may be destroyed. Due to this reason, asymmetric distributions have been proven to capture and work well under non-normality, asymmetry and skewness cases [10]

Time series models like autoregressive moving average (ARMA) models were design in such a way that they only work with complete data. If missing data occurs during modeling using the time series models, then modeling might be hampered. This way, researches has been carried out to develop statistical imputation algorithms and estimators through robust statistical estimation techniques like Likelihood Estimation [11], Least Squares Method [12], Expectation Maximization [13], and Interpolation.

1.1 Autoregressive Moving Average Model

The Autoregressive moving average (ARMA) model is a time series model that is widely used for analyzing time series data. The general form of ARMA (p, q) is given by;

$$k_t = \sum_{i=1}^{p} \theta_i k_{t-i} + \sum_{j=1}^{q} \varphi_j e_{t-j} + e_t \tag{1}$$

Where e_t in the literature is assumed to be Gaussian white noise with zero mean and a constant variance δ_α^2. In this context, we will have e_t assuming the Student's *t* distribution. It is taken that this process should be causal-stationary and invertible so that the roots of the process lie outside the unit circle.

Time series models that has considered the imputation of missing values under Gaussian or normal assumptions are numerous in the literature. For instance, author [14] did proposed an algorithm for filling missing values for ARMA time series model when their innovations consider a stable Gaussian assumption. Authors [15, 16] considered the missing value problem under the AR processes of time series models. Their studies focused on letting their AR error terms assume Gaussian approach. Author [17] considered imputation of missing values with pure bilinear time series models. The innovations for the series in their study, assumed normal distribution.

Time series models that have considered imputation of missing values under asymmetric assumptions include [18, 19], their study considered a vector autoregressive (VAR) time series, to model data with missing data under Multivariate heavy tailed, Student *t* distribution approach.

The study made an observation, that there is a gap in imputing missing values under time series analysis, especially when their innovations follow asymmetric distributions. This is the gap that the study will bridge and have a contribution to missing values imputation problems. The objective of this paper is to develop some imputation estimators for ARMA time series modeling under Student *t*-distribution error assumptions via optimal linear interpolation.

1.2 The Student *t* distribution

The Student *t* distribution is a continuous distribution that belongs to the family of continuous probability distributions. It has wide range of applications ranging from statistics and other genres of sciences. This distribution has been used in capturing heavy tails, asymmetry and non-normality. Such attributes of heavy tails and asymmetry are mostly displayed by economic data, business data and finance datasets [20, 21].

The probability distribution function for the standard student *t* is given by;

$$f(x) = \frac{1}{\sqrt{v}\beta\left(\frac{v}{2},\frac{1}{2}\right)}\left(1+\frac{x^2}{v}\right)^{-\left(\frac{1+v}{2}\right)}$$

$$-\infty < x < \infty, \text{where } v > 0 \qquad (2)$$

Some known characteristics of the Student *t* distribution on mean, variance, skewness and kurtosis are given by,

i. Mean (α_1) = E(x) = 0
ii. Variance
Var (x) = $\beta_2 = \frac{v}{v-2}, v > 2$
iii. Skewness

$$\gamma_1(x) = \frac{\beta_3}{\beta_2^{\frac{3}{2}}} = 0$$

iv. Kurtosis

$$\gamma_2(x) = \frac{\beta_3}{\beta_2^2} = \frac{3(v-2)}{v-4}, \qquad v > 4$$

A challenge arises when utilizing this distribution to capture asymmetry and heavy tail-ness in presence of missing data. This is because the student *t*-distribution is crafted to be modeled and utilized under a case when the all data is present.

The student *t* distribution is one of the most commonly used heavy tailed distribution in modellings. The authors [22, 21, 22] have considered their time series errors following student's *t* assumptions for missing value estimations.

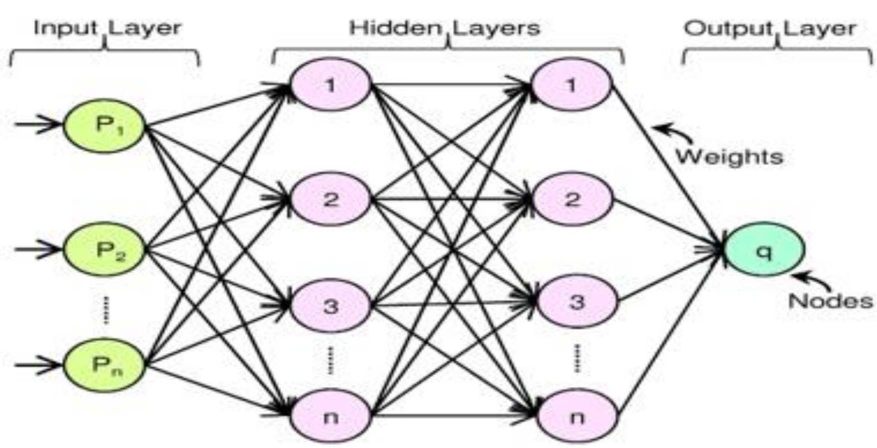
2. METHODOLOGY

This section provides an overview on what has been done on the state of the art techniques (machine learning techniques) of missing values imputation including the proposed optimal interpolation criterion for estimation of missing values in time series analysis. The imputation techniques will be utilized in the next sections in conducting comparisons of imputation of the missing values.

2.1 Artificial Neural Networks (ANNs) Imputation

The artificial neural networks (ANNs) are machine learning computation models that imitate the biological brain's neurons. ANNs has been utilized in several applications including the speech recognition, image recognition, pattern recognition machine learning including missing values and data predictions and forecasting [23, 24]. The neural networks consist of multiple layers of hidden interconnections of codes. The outputs of each layer is fed into the adjacent layers of codes. The layers in the networks are arranged in a manner that they are fed forward and not backwards as shown in the following figure 1. The hidden layers of ANNs are sandwiched between the input and the output layers that carry out the calculations of the inputted data. ANNs has been used in estimation of missing data in time series like the authors in [25], utilized ANNs to impute missing values for univariate time series case. The authors in [26] did conducted a study on imputation of missing values in multivariate time series case. Their study involved the use of modified ANNs models in comparison to other statistical imputation models.

FIG 1: **Artificial Neural Networks (ANNs) Imputation**



2.2 The Kalman Filter Imputation

The kalman filter is a set of statistical equations, that can be used in estimation of future, present and past states of some given processes. They are typically expressed in the form of a linear difference equation as,

$$\mu_t = \alpha_{t-1}\beta_{t-1} + \omega_{t-1}U_{t-1} + \theta_{t-1} \tag{3}$$

Where: μ_t is the state of the system at time t.

α_{t-1} is the state transition matrix, which describes how the state evolves from time *t- 1* to time t

α_{t-1} is the input matrix, which describes how the system responds to inputs U_{t-1}

U_{t-1} is the input to the system at time t-1

θ_{t-1} is the process noise, which is assumed to be zero-mean and Gaussian With covariance matrix ω_{t-1}

Kalman filters provides a powerful recursive formula for predicting, racking and forecasting dynamics systems using current estimates and observations. Kalman filters has been heavily relied on some applications like signal processing, tracking navigations and imputations. The kalman filters can be adjusted accordingly to fit the model being required to be modeled. The kalman filter has been applied in imputing missing data or values in time series analysis. Author [27, 28, 29] conducted a comparison on the imputation of missing values in time series using Kalman filters technique verses other methods of estimating missing values.

2.3 K – Nearness Neighbors (KNN) Imputation

KNN is a popular imputation machine learning algorithm that utilizes the neighboring values to the missing data sets to impute the missing values. Using this algorithm, the most likely value of missing data can be calculated using the Euclidean distance of nearest neighbors [30]. The KNN also does the leveraging in the similarities between the missing observations.

2.4 Optimal Linear Interpolation of Missing Values

The proposed method was suggested by author [12]. Also, the above mentioned method has been utilized by author [22] for the imputation of missing values under bilinear time series models. For the purposes of computation of missing values in autoregressive moving average (ARMA) time series models. They explained how the imputation computation is arrived at using the following statements, that;
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 $\mathrm{k_m}$ based on observed values $k_t, k_{t-1}, \cdots k_{m-1}$ that are given by $Q_m^* = Q_p\{k_t: t \leq n; t \neq m\}$. The projection of k_t on to Q_m^* denoted as $P_{S_m}^{k_m}$ such that the disp $\{K_m - P_{S_m}^{K_m}\}$ 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 $Q_1 = Q_P\{k_{m-1}, k_{m-2}, \cdots\}$ and Q_m^* are not independent of each other and thus we consider the evaluation of the projection on to two disjoint subspace of $\mathrm{Q_m^*}$. To achieve this, we express $\mathrm{Q_m^*}$ as a direct sum of subspaces $\mathrm{Q_1}$ and another subspace, say $\mathrm{Q_t}$ such that $Q_m^* = Q_1 \oplus Q_*$. A possible sub-space is $Q_* = Q_p\{k_i - k_i^t; i > m+1\}$. Where k_i^t is based on the values $\{k_{m-1}, k_{m-2}, \cdots\}$. The existence of subspaces $\mathrm{Q_1}$and $\mathrm{Q_*}$ 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;

$$\mathrm{K}_* = \mathrm{Z}^{\mathrm{n}} + \sum \mathrm{a_i K_i} = (\mathrm{K} + \sum \mathrm{a_i K_i^t}) + \sum \mathrm{a_i K_i^t} \text{ where } \mathrm{K} \in \mathrm{S}_1 \quad (4)$$

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 sub-spaces S_1 and S_* Such that the dispersion given by disp $(K_m - P_{B_m}^K)$ is minimized so that;

$$K_m^* = P_{S_m^*}^{K_m} = P_{S_1}^{K_m} + P_{S_*}^{K_m} = K_m + P_{S_m^*}^{K_m} \quad (5)$$

When *n* is assumed to be finite large data, so that the coefficients $\{a_v : v \geq m+1\}$ 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. $K_m^* = P_{S_m^*}^{K_m} = P_{S_1}^{K_m} + P_{S_*}^{K_m} = K_m + P_{S_m^*}^{K_m}$ (6)

But $P_{S_m^*}^{K_m} = \left\{\sum_{v=m+1}^{n} \xi_v \left(K_v - K_v\right); disp\left(K_m - P_{S_m^*}^{K_m}\right)\right\}$ (7)

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

$$\mathrm{disp}\ \mathrm{X_m} = \mathrm{E}(\mathrm{K_m} - \mathrm{K_m^*}) = \left\{\left(K_m - \widehat{K}_m\right) - \sum_{v=m+1}^{n} \xi_v \left(K_v - \widehat{K}_m\right)\right\}^2 \quad (8)$$

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;

$$\widehat{K}_m^* = \hat{k}_m + \sum_{v=m+1}^{n} \xi_v \left(k_v - \hat{k}_v\right) \quad (9)$$

3. RESULTS AND DISCUSIONS

3.1 Derivation of Imputation Estimators for ARMA Process with student's t errors

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_*

Theorem 1

The imputation optimal interpolation estimator for ARMA (1,1) process is given by*;*

$$\hat{k}_m = \hat{\theta}_1 k_{t-1} + \hat{\varphi}_1 e_{t-1} + \sum_{v=m+1}^{n} (\theta_1 + \varphi_1) \ - \frac{(\theta_1 + \varphi_1)^{2(v-m)}}{2} \left(\frac{n}{n-2}\right)_{v-1} \left(k_v - \hat{k}_v\right) \quad (10)$$

Proof

The stationary ARMA (1,1) process is given by,

$$k_t = \theta_1 k_{t-1} + \varphi_1 \mathcal{E}_{t-1} + \mathcal{E}_t \text{ where } \mathcal{E}_t \sim t(0,1) \tag{11}$$

Getting the recursive form of equation (10) above, the it can be obtained as,

$$k_m = \sum_{i=1}^{\infty} \left\{ \prod_{j=1}^{i} (\theta_1 + \varphi_1) \mathcal{E}_{t-1} + \mathcal{E}_t \right\} \tag{12}$$

Obtaining the r-step future predicted form of the above equation (11), we write it as,

$$k_{m+r} = \sum_{i=1}^{\infty} \left\{ \prod_{j=1}^{i} (\theta_1 + \varphi_1) \mathcal{E}_{t+r-1} + \mathcal{E}_{t+r} \right\} \tag{13}$$

Or if we set $v = r + h$, then the above equation (12) can be rewritten as,

$$k_v = \sum_{i=1}^{\infty} \left\{ \prod_{j=1}^{i} (\theta_1 + \varphi_1) \mathcal{E}_{v-1} + \mathcal{E}_v \right\} \tag{14}$$

Obtaining r-step future predicted error of the above equation (13), which can be expressed as,

$$k_v - \hat{k}_v = \sum_{i=1}^{r-1} \left\{ \prod_{j=1}^{i} (\theta_1 + \varphi_1) \mathcal{E}_{v-1} + \mathcal{E}_v \right\} \tag{15}$$

From the above statement the dispersion is given as,

$$\text{disp } k_m = E\left(k_m - \hat{k}_m\right)^2 = E\left\{k_m - \hat{k}_m - \sum_{v=m+1}^{n} \xi_v \left(k_v - \hat{k}_v\right)\right\}^2 \tag{16}$$

If the second part of equation (15) above is simplified, then it can be obtained that the dispersion is given as,

$$\text{disp } k_m = E\left(k_m - \hat{k}_m\right)^2 - 2E\left\{\sum_{v=m+1}^{n} \xi_v \left(k_m - \hat{k}_m\right)\left(k_v - \hat{k}_v\right)\right\} + \left\{\sum_{v=m+1}^{n} \xi_v \left(k_v - \hat{k}_v\right)\right\}^2 \tag{17}$$

Substituting equation (14) into equation (16) above, the dispersion is given as;

$$\text{disp } k_m = E\left(k_m - \hat{k}_m\right)^2 - 2E\left\{(\mathcal{E}_m)\left(\sum_{i=1}^{r-1}\left\{\prod_{j=1}^{i}(\theta_1 + \varphi_1)\mathcal{E}_{v-1} + \mathcal{E}_v\right\}\right)\right\}$$
$$+E\left\{\sum_{v=m+1}^{n} \xi_v \left(\sum_{i=1}^{r-1}\left\{\prod_{j=1}^{i}(\theta_1 + \varphi_1)\mathcal{E}_{v-1} + \mathcal{E}_v\right\}\right)\right\}^2 \tag{18}$$

Evaluating the above equation (17) then it can be found that;
The first term to be,

$$k_m = E\left(k_m - \hat{k}_m\right)^2 = E(\varepsilon_m)^2$$

The second term given by,

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

Which can be evaluated further to obtain,

$$-2E\begin{bmatrix}\xi_{m+1}(\theta_1+\varphi_1)\varepsilon_m\cdot\varepsilon_m+\xi_{m+1}\varepsilon_{m+1}\\+\xi_{m+2}(\theta_1+\varphi_1)\cdot(\theta_1+\varphi_1)\varepsilon_{m+1}\cdot\varepsilon_m+\xi_{m+2}\varepsilon_{m+2}\\+\xi_{m+3}(\theta_1+\varphi_1)\cdot(\theta_1+\varphi_1)\cdot(\theta_1+\varphi_1)\varepsilon_{m+2}\cdot\varepsilon_m\\+\xi_{m+3}\varepsilon_{m+3}+\cdots\end{bmatrix}$$

Or

$$-2E\begin{bmatrix}\xi_{m+1}(\theta_1+\varphi_1)\varepsilon_m^2+\xi_{m+1}\varepsilon_{m+1}\\+\xi_{m+2}(\theta_1+\varphi_1)^2\varepsilon_{m+1}\cdot\varepsilon_m+\xi_{m+2}\varepsilon_{m+2}\\+\xi_{m+3}(\theta_1+\varphi_1)^3\cdot\varepsilon_{m+2}\cdot\varepsilon_m+\xi_{m+3}\varepsilon_{m+3}+\cdots\end{bmatrix}$$

Which can be simplified further to be,

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

The third term is given by,

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

Can be evaluated to be,

$$+E\begin{bmatrix}\xi_{m+1}(\theta_1+\varphi_1)\varepsilon_m\cdot\varepsilon_m+\xi_{m+1}\varepsilon_{m+1}\\+\xi_{m+2}(\theta_1+\varphi_1)\cdot(\theta_1+\varphi_1)\varepsilon_{m+1}\cdot\varepsilon_m+\xi_{m+2}\varepsilon_{m+2}\\+\xi_{m+3}(\theta_1+\varphi_1)\cdot(\theta_1+\varphi_1)\cdot(\theta_1+\varphi_1)\varepsilon_{m+2}\cdot\varepsilon_m\\+\xi_{m+3}\varepsilon_{m+3}+\cdots\end{bmatrix}^2$$

Or

$$+E\begin{Bmatrix}\xi_{m+1}^2(\theta_1+\varphi_1)^2\varepsilon_m^2\varepsilon_m^2+\xi_{m+1}^2\varepsilon_{m+1}^2\\+\xi_{m+2}^2(\theta_1+\varphi_1)^4\varepsilon_{m+1}^2\varepsilon_m^2+\xi_{m+2}^2\varepsilon_{m+2}^2\\+\xi_{m+3}^2(\theta_1+\varphi_1)^6\varepsilon_{m+2}^2\varepsilon_m^2+\xi_{m+3}^2\varepsilon_{m+3}^2+\cdots\end{Bmatrix}$$

The above equation of the third term, can be simplified to obtain,

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

Putting all terms of the above equation (17) it can be obtained that the equation under the dispersion is given as;

$$k_m = \begin{bmatrix} (\varepsilon_m)^2 - 2E\{\xi_{m+1}(\theta_1+\varphi_1)\varepsilon_m^2\} \\ +\left\{\varepsilon_m^2 \sum_{v=m+1}^{n} \xi_v(\theta_1+\varphi_1)^{2(v-m)}\,\varepsilon_{v-1}^2 + \sum_{v=m+1}^{n} \xi_v^2\,\varepsilon_v^2\right\} \end{bmatrix} \quad (19)$$

Substituting for the errors in the above equation (18) using the assumption of the distribution taken, we can have the dispersion as;

$$\text{disp}\,(\hat{x}_m) = \left(\begin{bmatrix} \frac{n}{n-2} - 2\left\{\xi_{m+1}(\theta_1+\varphi_1)\frac{n}{n-2}\right\} + \frac{n}{n-2} \\ \sum_{v=m+1}^{n} \xi_v(\theta_1+\varphi_1)^{2(v-m)}\left(\frac{n}{n-2}\right)_{v-1} + \sum_{v=m+1}^{n} \xi_v^2\left(\frac{n}{n-2}\right)_v \end{bmatrix}\right) \quad (20)$$

Differentiating the above equation (19), then we obtain;

$$\frac{\partial}{\partial \xi_v}\{\text{disp}x_m\} = \left(\begin{matrix} \left(\frac{\partial}{\partial \xi_v}\left\{\frac{n}{n-2}\right\} - 2\frac{\partial}{\partial \xi_v}\left\{\xi_{m+1}(\theta_1+\varphi_1)\frac{n}{n-2}\right\}\right) \\ \left(+\frac{\partial}{\partial \xi_v}\left\{\frac{n}{n-2}\sum_{v=m+1}^{n} \xi_v(\theta_1+\varphi_1)^{2(v-m)}\left(\frac{n}{n-2}\right)_{v-1}\right\} + \frac{\partial}{\partial \xi_v}\left\{\sum_{v=m+1}^{n} \xi_v^2\left(\frac{n}{n-2}\right)_v\right\}\right) \end{matrix}\right) \quad (21)$$

Evaluating the above equation (20) under the derivative and equating it to zero we obtain;

$$-2\left\{(\theta_1+\varphi_1)\frac{n}{n-2}\right\} + \frac{n}{n-2}(\theta_1+\varphi_1)^{2(v-m)}\left(\frac{n}{n-2}\right)_{v-1} + 2\xi\left(\frac{n}{n-2}\right) = 0 \quad (22)$$

Making ξ_v to be the subject of the above equation (21), it can be found that;

$$\xi_v = (\theta_1+\varphi_1) - \frac{(\theta_1+\varphi_1)^{2(v-m)}}{2}\left(\frac{n}{n-2}\right)_{v-1} \quad (23)$$

Substituting the value of ξ_v from the above equation (22) into the expression of interpolation given by the equation (8), it can be found that the optimal linear interpolator for ARMA (1,1) is;

$$\hat{k}_m = \hat{\theta}_1 k_{t-1} + \hat{\varphi}_1 e_{t-1} + \sum_{v=m+1}^{n} (\theta_1+\varphi_1) - \frac{(\theta_1+\varphi_1)^{2(v-m)}}{2}\left(\frac{n}{n-2}\right)_{v-1}(k_v - \hat{k}_v) \quad (24)$$

Theorem 2

The imputation optimal interpolation estimator for ARMA (2, 1) process is given by

$$\hat{k}_t = \theta_2\hat{k}_{t-2} + \theta_1\hat{k}_{t-1} + \varphi_1\varepsilon_{t-1} + \sum_{v=m+1}^{n} \frac{\theta_2}{(\theta_2)^{2(v-m)} + (\theta_1+\varphi_1)^{2(v-m)} + 1}\left(k_v - \hat{k}_v\right) \tag{25}$$

Proof

A stationery ARMA (2,1) process is given as

$$k_t = \theta_2 k_{t-2} + \theta_1 k_{t-1} + \varphi_1\varepsilon_{t-1} + \varepsilon_t \text{ Where } \varepsilon_t \sim t(0,1) \tag{26}$$

The recursive form of the above equation (25) can be found to be;

$$k_t = \sum_{i=1}^{\infty}\left[\prod_{j=1}^{i}(\theta_2)\,\varepsilon_{t-2j}\right] + \sum_{i=1}^{\infty}\left[\prod_{j=1}^{i}(\theta_1+\varphi_1)\,\varepsilon_{t-j}\right] + \varepsilon_t \tag{27}$$

The r-step future forecast of the above process in the equation (26) can be given by;

$$k_{t+r} = \sum_{i=1}^{\infty}\left[\prod_{j=1}^{i}(\theta_2)\,\varepsilon_{t+r-2j}\right] + \sum_{i=1}^{\infty}\left[\prod_{j=1}^{i}(\theta_1+\varphi_1)\,\varepsilon_{t+r-j}\right] + \varepsilon_{t+r} \tag{28}$$

Setting $v = t + r$, then the above equation (27) can be rewrite as,

$$k_v = \sum_{i=1}^{\infty}\left[\prod_{j=1}^{i}(\theta_2)\,\varepsilon_{v-2j}\right] + \sum_{i=1}^{\infty}\left[\prod_{j=1}^{i}(\theta_1+\varphi_1)\,\varepsilon_{v-j}\right] + \varepsilon_v \tag{29}$$

The future prediction error of equation (28) can be given by,

$$k_v - \hat{k}_v = \sum_{i=1}^{v-1}\left[\prod_{j=1}^{i}(\theta_2)\,\varepsilon_{v-2j}\right] + \sum_{i=1}^{v-1}\left[\prod_{j=1}^{i}(\theta_1+\varphi_1)\,\varepsilon_{v-j}\right] + \varepsilon_v \tag{30}$$

From the above lemma, the dispersion is given by,

$$\text{disp}\,k_m = E\left(k_m - \hat{k}_m\right)^2 = E\left\{k_m - \hat{k}_m - \sum_{v=m+1}^{n}\xi_v\left(k_v - \hat{k}_v\right)\right\}^2 \tag{31}$$

If the second part of the above equation (30) is simplified, then it can be obtained that the dispersion is given by,

$$\text{disp}\,k_m = E\left(k_m - \hat{k}_m\right)^2 - 2\left\{\sum_{v=m+1}^{n}\xi_v\left(k_m - \hat{k}_m\right)\left(k_v - \hat{k}_v\right)\right\} + E\left\{\sum_{v=m+1}^{n}\xi_v\left(k_v - \hat{k}_v\right)\right\}^2 \tag{32}$$

Substituting the above equation (29) in to the equation (31), it can be obtained that the dispersion given by $disp\ k_m$ is;

$$\mathrm{disp}\ k_m = E\left(k_m - \hat{k}_m\right)^2 - 2E(e_m)\left[\sum_{v=m+1}^{n} \xi_v (e_m)\begin{pmatrix} \sum_{i=1}^{\infty}\left[\prod_{j=1}^{i}(\theta_2)\varepsilon_{v-2j}\right] \\ +\sum_{i=1}^{\infty}\left[\prod_{j=1}^{i}(\theta_1+\varphi_1)\varepsilon_{v-j}\right]+\varepsilon_v \end{pmatrix}\right]$$
$$+E\left[\sum_{v=m+1}^{n} \xi_v \begin{pmatrix} \sum_{i=1}^{\infty}\left[\prod_{j=1}^{i}(\theta_2)\varepsilon_{v-2j}\right] \\ +\sum_{i=1}^{\infty}\left[\prod_{j=1}^{i}(\theta_1+\varphi_1)\varepsilon_{v-j}\right]+\varepsilon_v \end{pmatrix}\right]^2 \qquad (33)$$

Evaluating further the above equation (32) to its equivalent terms, then it can be expressed that,

The first term is

$$E\left(k_m - \hat{k}_m\right)^2 = E(\mathcal{E}_m)^2$$

The second term is,

$$-2E[e_m]\begin{bmatrix} \xi_{m+1}\theta_2\mathcal{E}_{m-1} + \xi_{m+1}(\theta_1+\varphi_1)\mathcal{E}_{m-1} + \xi_{m+1}\mathcal{E}_{m-1} \\ \xi_{m+2}\theta_2\mathcal{E}_{m} + \xi_{m+2}(\theta_1+\varphi_1)^2\mathcal{E}_{m+1} + \xi_{m+2}\mathcal{E}_{m+2} \\ \xi_{m+3}\theta_3^3\theta_2\mathcal{E}_{m+1} + \xi_{m+3}(\theta_1+\varphi_1)^2\mathcal{E}_{m+2} + \xi_{m+3}\mathcal{E}_{m+3} + \cdots \end{bmatrix}$$

Which can further be simplified to be,

$$-2E\left[\sum_{v=m+2}^{n}(\mathcal{E}_m^2)\xi_v\theta_2\right]$$

The third term can be expressed as,

$$+\sum_{v=m+1}^{n}\xi_v^2\cdot\sum_{i=1}^{\infty}\left[\prod_{j}^{i}(\theta_2)^2\,\mathcal{E}_{v-2j}^2\right]+\sum_{v=m+1}^{n}\xi_v^2\cdot\sum_{j}^{n}\left[\prod_{i}^{j}(\theta_1+\varphi_1)^2\mathcal{E}_{v-i}^2\right]+\sum_{v=m+1}^{n}\xi_v^2\,\mathcal{E}_v^2$$

Can be evaluated as,

$$+E\begin{bmatrix} \xi_{m+1}^2\cdot(\theta_2)^2\mathcal{E}_{m-1}^2 + \xi_{m+1}^2\cdot(\theta_1+\varphi_1)^2\mathcal{E}_m^2 + \xi_{m+1}^2\mathcal{E}_{m+1}^2 \\ \xi_{m+2}^2\cdot(\theta_2)^4\mathcal{E}_m^2 + \xi_{m+1}^2\cdot(\theta_1+\varphi_1)^4\mathcal{E}_{m+1}^2 + \xi_{m+2}^2\mathcal{E}_{m+2}^2 \\ \xi_{m+3}^2\cdot(\theta_2)^8\mathcal{E}_{m+1}^2 + \xi_{m+3}^2\cdot(\theta_1+\varphi_1)^6\mathcal{E}_{m+1}^2 + \xi_{m+3}^2\mathcal{E}_{m+3}^2 + \cdots \end{bmatrix}$$

Which can be written as,

$$+E\left(\sum_{v=m+1}^{n}\xi_v^2\cdot(\theta_2)^{2(v-m)}\mathcal{E}_v^2 + \sum_{v=m+1}^{n}\xi_v^2\cdot(\theta_1+\varphi_1)^{2(v-m)}\mathcal{E}_v^2 + \sum_{v=m+1}^{n}\xi_v^2\mathcal{E}_v^2 + \cdots\right)$$

Which can generalize it as;

$$+E\left\{\sum_{v=m+1}^{v-m}\xi_v^2\mathcal{E}_v^2\left[(\theta_2)^{2(v-m)} + (\theta_1+\varphi_1)^{2(v-m)} + 1\right]\right\}$$

Putting all the worked terms of equation (32) together, then it can be found that the dispersion becomes,

$$\text{disp } k_t = \mathrm{E}(\mathcal{E}_\mathrm{m})^2 - 2\mathrm{E}\left[\sum_{\mathrm{v=m+2}}^{n} (\mathcal{E}_\mathrm{m}^2)\xi_\mathrm{v}\theta_2\right] + \mathrm{E}\left\{\sum_{\mathrm{v=m+1}}^{n} \xi_\mathrm{v}^2\mathcal{E}_\mathrm{v}^2\left[(\theta_2)^{2(\mathrm{v-m})} + (\theta_1+\varphi_1)^{2(\mathrm{v-m})} + 1\right]\right\} \tag{34}$$

Substituting for the above equation (10) for the errors using the characteristics of the assumed distribution, then we will have the dispersion as,

$$\text{disp } k_t = \frac{n}{n-2} - 2\frac{n}{n-2}\left[\sum_{\mathrm{v=m+2}}^{\mathrm{n}} \xi_\mathrm{v}\theta_2\right] + \frac{n}{n-2}\left\{\sum_{\mathrm{v=m+1}}^{\mathrm{n}} \xi_\mathrm{v}^2\left[(\theta_2)^{2(\mathrm{v-m})} + (\theta_1+\varphi_1)^{2(\mathrm{v-m})} + 1\right]\right\} \tag{35}$$

Differentiating the above equation with respect to ξ_v

$$\frac{\partial}{\partial\xi_\mathrm{v}}(\text{disp } k_t) = \frac{\partial}{\partial\xi_\mathrm{v}}\left(\frac{n}{n-2}\right) - 2\frac{\partial}{\partial\xi_\mathrm{v}}\left[\frac{n}{n-2}\left[\sum_{\mathrm{v=m+2}}^{\mathrm{n}} \xi_\mathrm{v}\theta_2\right]\right] + \frac{\partial}{\partial\xi_\mathrm{v}}\left[\frac{n}{n-2}\left\{\sum_{\mathrm{v=m+1}}^{\mathrm{n}} \xi_\mathrm{v}^2\left[(\theta_2)^{2(\mathrm{v-m})} + (\theta_1+\varphi_1)^{2(\mathrm{v-m})} + 1\right]\right\}\right] \tag{36}$$

Setting it to zero, it can be obtained that ξ_v can be obtained as,

$$\xi_\mathrm{v} = \sum_{v=m+1}^{n} \frac{\theta_2}{(\theta_2)^{2(\mathrm{v-m})} + (\theta_1+\varphi_1)^{2(\mathrm{v-m})} + 1} \tag{36}$$

Substituting the above equation (36) into the expression of the interpolation given by equation (8), it can be found that the optimal linear interpolator for ARMA (1,2) can be given by;

$$\hat{k}_\mathrm{t} = \theta_2\hat{k}_{\mathrm{t-2}} + \theta_1\hat{k}_{\mathrm{t-1}} + \varphi_1\mathcal{E}_{\mathrm{t-1}} + \sum_{v=m+1}^{n} \frac{\theta_2}{(\theta_2)^{2(\mathrm{v-m})} + (\theta_1+\varphi_1)^{2(\mathrm{v-m})} + 1}\left(\mathrm{k_v} - \hat{\mathrm{k}}_\mathrm{v}\right)$$

3.2 Synthetic data Generation and Simulation

The study generated 1000 samples using the scripts in R statistical package version 4.42. The R scripts were used to generate the required synthetic data sets. The generated samples took into consideration the student's *t* assumptions. The study considered the generation of 100 missing value positions within the created synthetics data set. The packages for imputations were also installed accordingly to perform the imputation. The missing mechanism for the missing data in the simulated data was missing at random (MAR) mechanism. The codes for simulations are provided here in.

Non-stationary illustrations of the simulated data are given by the following time series plot

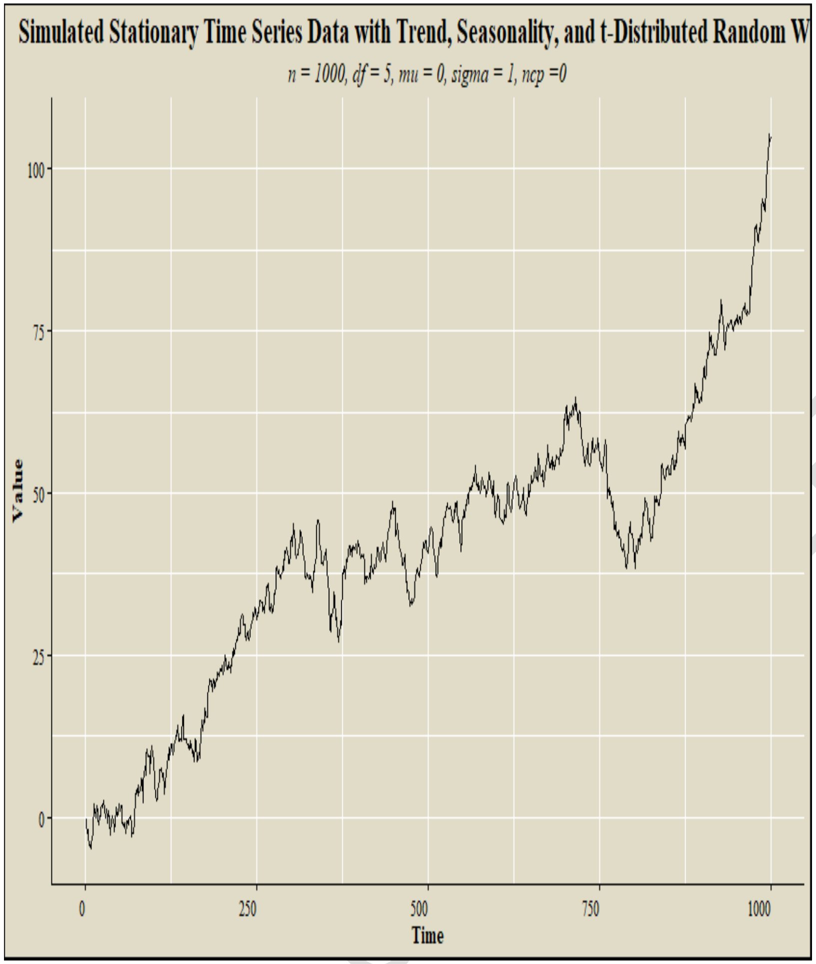


Figure 2

The illustration of the missing data at random in the simulated data sets. The figure below has 100 values of missing data

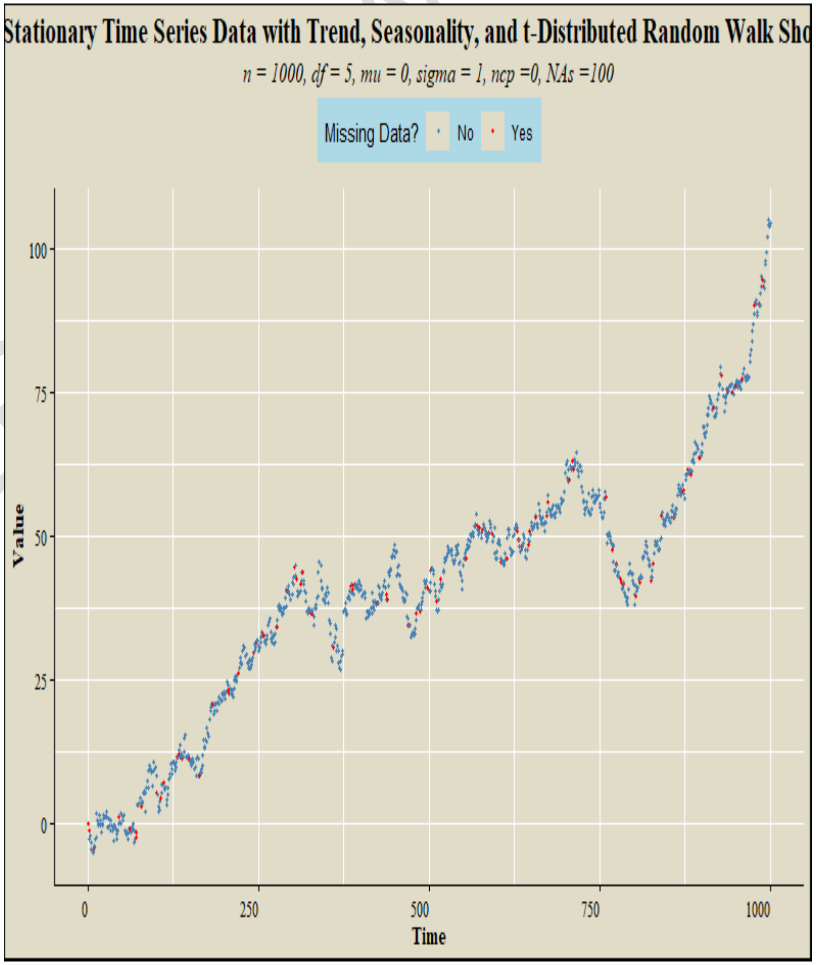


Figure 3

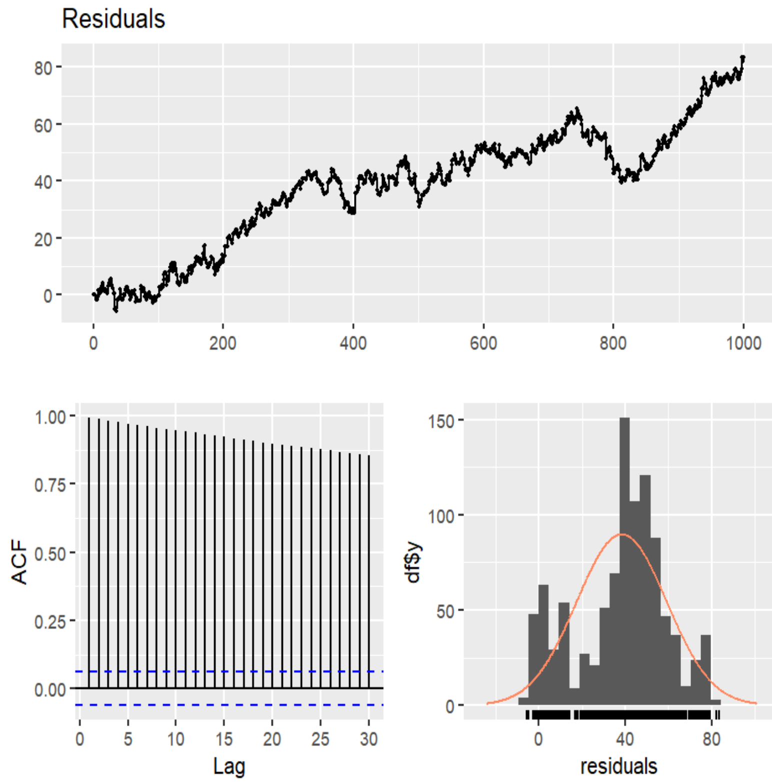


Figure 4 ACF and the residuals of the used data sets

The above figure 4, gives the ACF and the residuals of the used data sets.
Imputing the created missing values using the derived model for ARMA (1,1) process, the illustration is given below in figure 4.

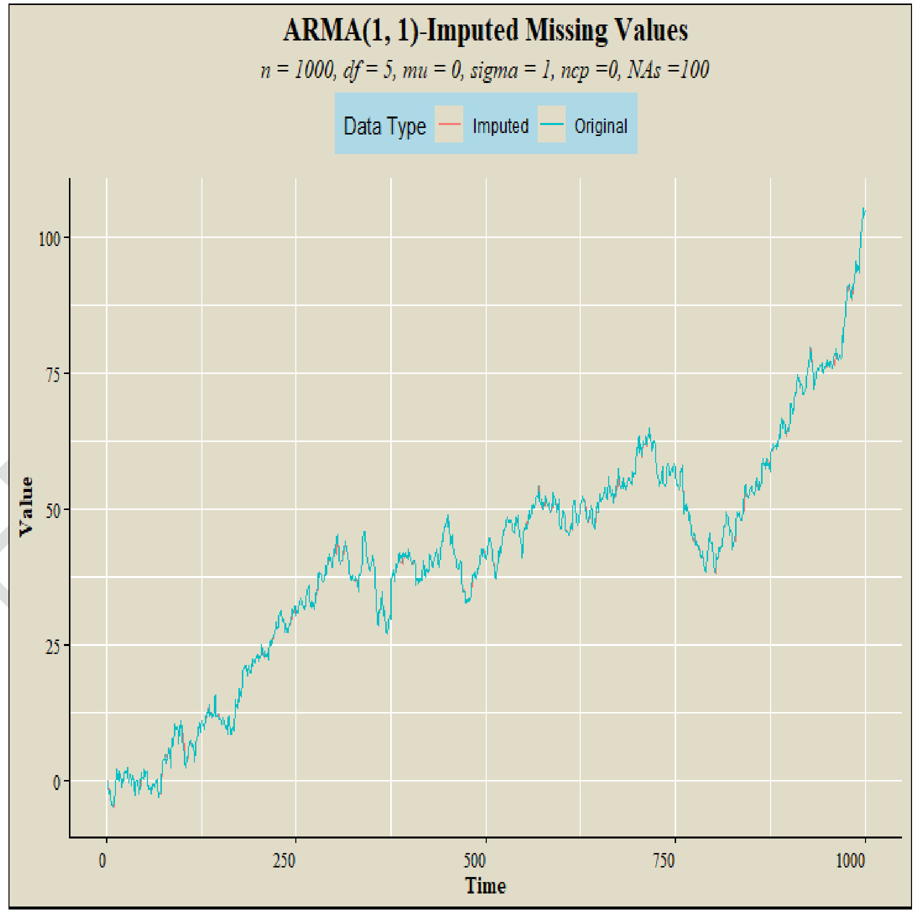


Figure 5: ARMA(1,1)-Imputed Missing values

The above figure 5, shows how the imputation with the estimator for ARMA (1,1) was plotted. It's clear that the imputation is almost actuate.

Imputing missing values for ARMA (2,1) process, then this is how the ACF and the Residuals behaved from the used data.

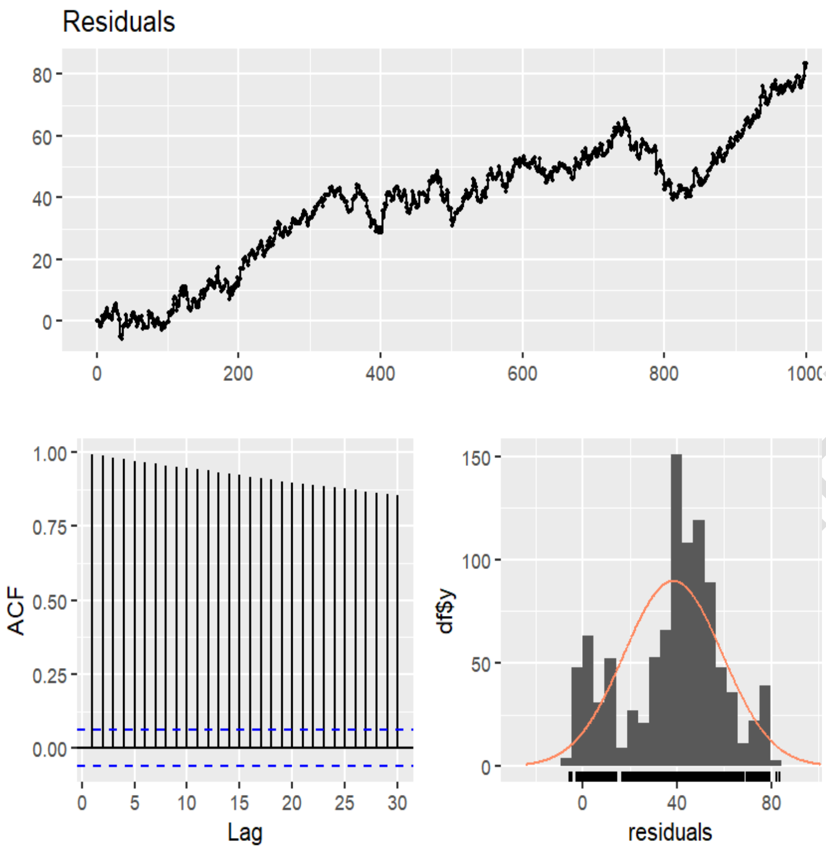


Figure 6 ACF and the Residuals behaved from the used data

The illustration of the imputed missing data using the time series plot is given by the following figure.

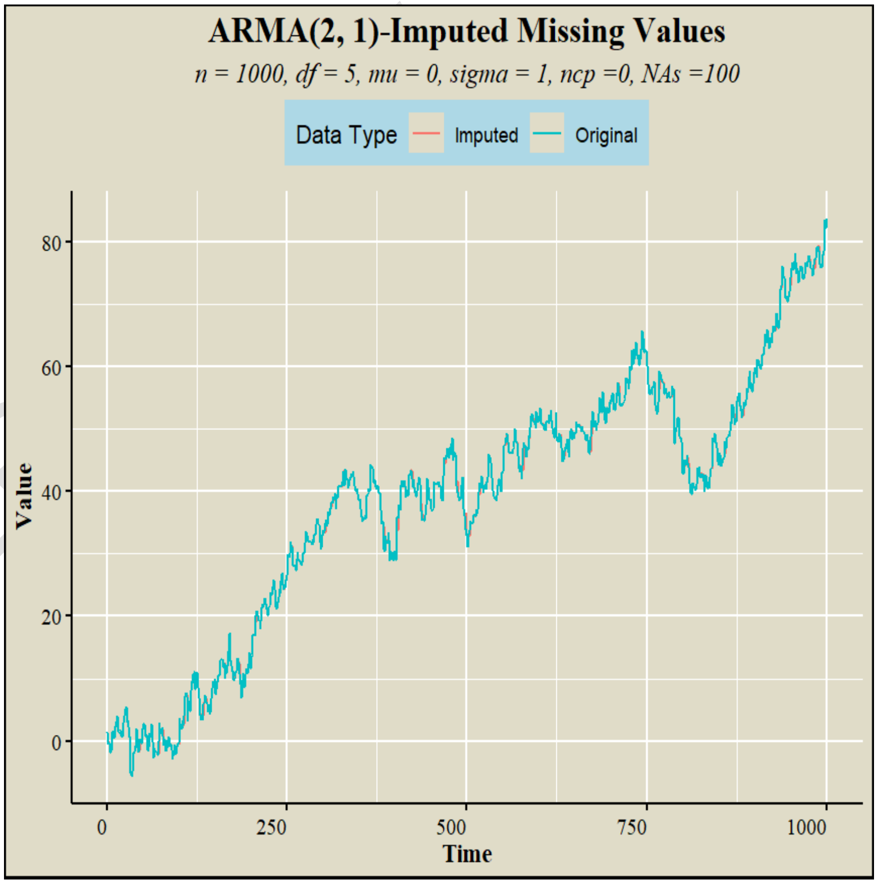


Figure 7: ARMA(2,1)-Imputed Missing values

The imputation from the state of the art imputation methods ware done using the ANN imputation. Its observation is given by the following plots of the AFC and the residuals.

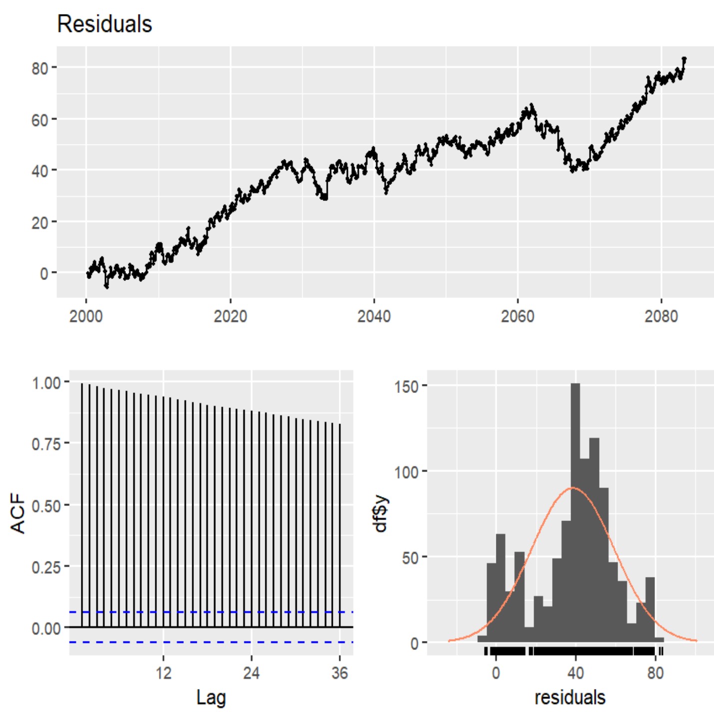


Figure 8 State of the art **imputation methods ware done using the ANN imputation**

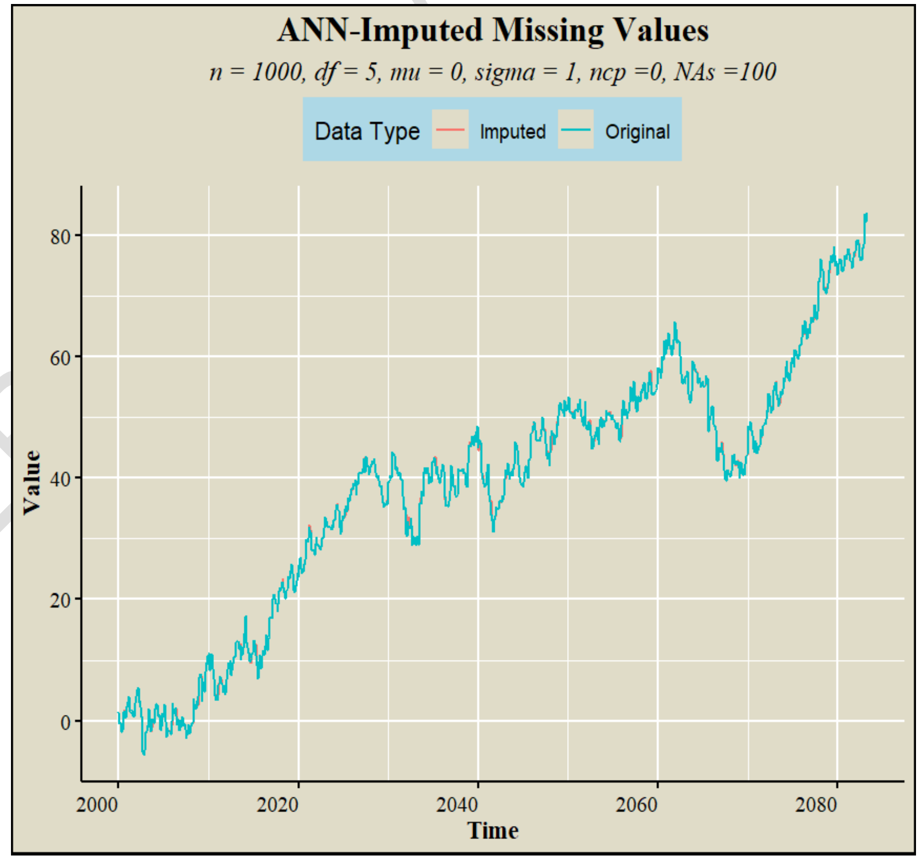


Figure 9: ANN-Imputed Missing values

Imputing missing values using the Kalman filters was also illustrated using the following plot on **Figure 10**

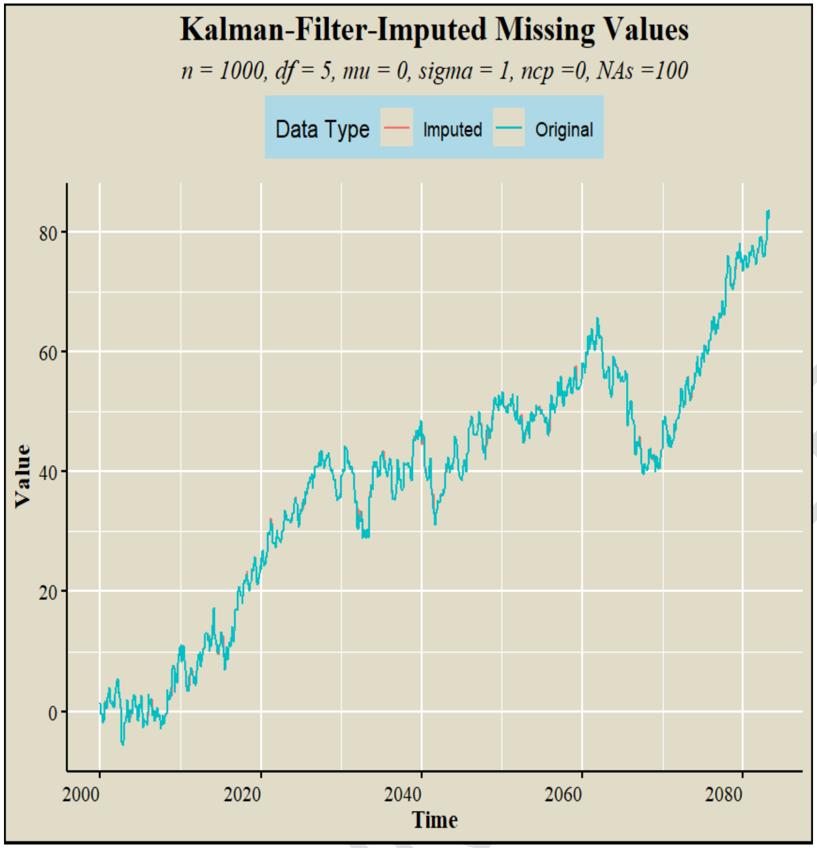


Figure 10: Kalman-Filter-Imputed Missing values

Imputing missing values using the KNN algorithm, was also illustrated using the following plot on **Figure 11**

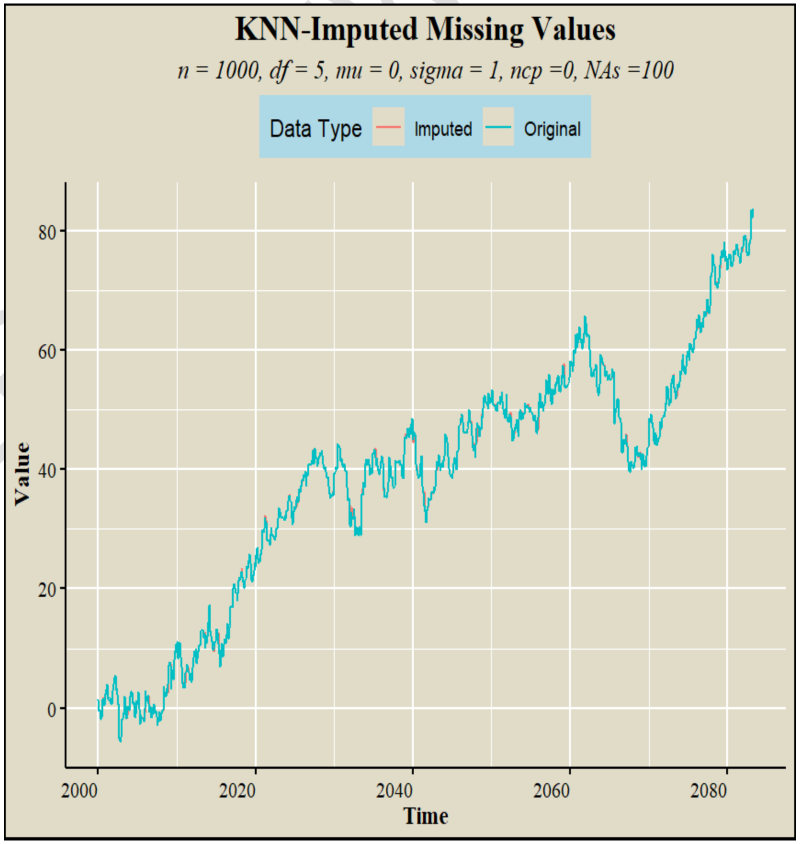


Figure 11:KMN-Imputed missing values

4. CONCLUSION

The derived and adjusted ARMA imputation estimators, were able to compete favorably with the state-of-the-art imputation techniques like ANN, a machine learning technique that has taken the center stage at the prediction and forecasting, Kalman filters and the KNN. There is need to conduct a clear comparison of the developed imputation estimators and other methods of imputation like Bayesian, nonparametric imputation estimators. Also, using real data sets in assessing the imputation efficiency can be conducted.

REFERENCES

[1] Introduction to Time Series and Forecast. Peter J. Brockwell and Richard A. Davis Third edition DOI 10.1007/978-3-319-29854-2

[2] Alfredo Vellido (2005). Missing data Imputation through GTM as a mixture of *t* distribution. 19 (2006) 1624- 1635. www.elsevier.com/locate/neunet

[3] McLachlan, G. J., & Peel, D. (2000a). On computational aspects of clustering via mixtures of normal and *t*-components. In Proceedings of the American statistical association (Bayesian Statistical Science Section). Alexandria, VA: American Statistical Association.

[4] Student's-*t* VAR Modeling with Missing Data Via Stochastic EM and Gibbs Sampling. Rui Zhou, Junyan Liu, Sandeep Kumar and Daniel Palomar. Vol. 68, 2020. Pp. 6198-6211. IEEE TRANSACTIONS ON SIGNAL PROCESSING

[5] Mohammad Ahsanullah, B. M. Golam Kibria, Mohammad Shakil Normal and Student's-*t* Distributions and Their Applications ISBN 978-94-6239-061-4 (eBook) DOI 10.2991/978-94-6239-061-4 Library of Congress Control Number: 2013957385

[6] SICE: An improved missing data imputation technique. Shahldul Islam Khan, Abu Sayed Md Latiful Hoque. (2020) *7:37 https://doi.org/10.1186/s40537-020-00313-w*

[7] Kwak Sang Kyu, Kim Jong Hae. Statistical data Preparation: Management of Missing Values and Otliers. Korean J Anesthesiol. 2017;70(4):407.

[8] Ana Almeida, Susana Brás, Felipe Cabral Pinto. Focalize K-NN: An imputation algorithm for time series datasets. Pattern Analysis and Applications (2024) 27:39 https://doi.org/10.1007/s10044-024-01262-3

[9] Lovorka Gotal Dmitrović, Vesna Dušak, Jasminka Dobša: Missing data problems in non-Gaussian probability distributions Informatol. 49, 2016. 138-152

[10] Ané Neethling, Johan Ferreira, Andriëtte Bekker and Mehrdad Naderi. Skew Generalized Normal Innovations for the AR(p) Process Endorsing Asymmetry. 2020, 12, 1253; doi:10.3390/sym12081253

[11] Demet AYDIN, Birdal ŞENOĞLU. Estimating the Missing Value in one-way ANOVA under Long-tailed Symmetric Error Distributions. Sigma J Eng & Nat Sci 36 (2), 2018, 523-538

[12] Pascal Bondona, and Natalia Bahamonde. Least squares estimation of ARCH models with missing observations. J. Time Ser. Anal. 2012, 33 880–891. (wileyonlinelibrary.com) DOI: 10.1111/j.1467-9892.2012. 00803.x

[13] Dempster A.P., Laird N.M. and Rubin D., (1977) Maximum Likelihood Estimation from Incomplete Data via the EM Algorithm, *Journal of the Royal Statistical Society* 39, 1–38.

[14] Nassiuma, D. K. (1994). Symmetric stable sequence with missing observations. J.T.S.A. volume 15, page 317

[15] J. Ding, L. Han, and X. Chen, "Time series AR modeling with missing observations based on the polynomial transformation," *Math. Compute. Model.*, vol. 51, no. 5/6, pp. 527–536, 2010.

[16] V. A. Voloshko and Y. S. Kharin, "Robust estimation of AR coefficients under simultaneously influencing outliers and missing values," *J. Statistical Planning Inference*, vol. 141, no. 9, pp. 3276–3288, 2011.

[17] Poti Abaja Owili, Luke Orawo, Dankit Nassiuma, Imputation of Missing Values for Pure Bilinear Time Series Models

with Normal Distributed Innovations. 2015, Vol. 3 No. 5, pp. 199-202. DOI: 10.12691/ajams-3-5-4.

[18] J. Liu, S. Kumar, and D. P. Palomar, “Parameter estimation of heavy-tailed AR(p) model from incomplete data,” in *Proc. Eur. Signal Process. Conf.*, A Coruña, Spain, Sep. 2019, pp. 2–6.

[19] Rui Zhou, Junyam Liu, Sandeep Kumar and Daniel P. Paloma. Student t VAR Modelling with Missing Data via Stochastic E-M and Gibbs Sampling. 2020, Vol. 68, pp. 6199-6211

[20] Sandya Nilmini Kumari, Abby Tan. Characterization of Student’s t- Distribution with some Application to Finance Mathematical Theory and Modeling, ISSN 2224-5804 (Paper) ISSN 2225-0522 (Online) Vol.3, No.10, 2013

[21] Bruno Dinga, Jimbo Henry Claver, Kum Kwa Cletus and Shu Felix Che. Modeling and Predicting Exchange Rate Volatility: Application of Symmetric GARCH and Asymmetric EGARCH and GJR-GARCH Models. JOURNAL OF THE CAMEROON ACADEMY OF SCIENCES Vol. 19 No. 2 (JULY 2023). DOI: https://dx.doi.org/10.4314/jcas.v19i2.6

[22] Poti Abaja Owili, Luke Orawo, Dankit Nassiuma, Estimation of Missing Values for Pure Bilinear Time Series Models with Student-t Innovations, *International Journal of Statistics and Applications*, Vol. 5 No. 6, 2015, pp. 293-301. http://doi: 10.5923/j.statistics.20150506.05.

[23] Safa, B., Arkebauer, T.J., Zhu, Q.M., Suyker, A. and Irmak, S. (2021) Gap Filling of Net Ecosystem CO_2 Exchange (NEE) above Rain-Fed Maize Using Artificial Neural Networks (ANNs). Journal of Software Engineering and Applications, 14, 150-171. https://doi.org/10.4236/jsea.2021.145010

[24] Faru, S.H., Waititu, A. and Nderu, L. (2023) A Hybrid Neural Network Model Based on Transfer Learning for Forecasting Forex Market. Journal of Data Analysis and Information Processing, 11, 103-120. https://doi.org/10.4236/jdaip.2023.112007

[25] Ana Almeida, Susana Brás, Susana Sargento & Filipe Cabral Pinto. Focalize K-NN: an imputation algorithm for time series datasets. Pattern Analysis and Applications (2024) 27:39 https://doi.org/10.1007/s10044-024-01262-3

[26] Wei C., Dong W., Jian L., Hao Z, Y., Tan L., Lei Li. Bidirectianl Recurrent Imputation for Time Series. 32nd Conference on Neural Information Processing Systems (NeurIPS 2018), Montréal, Canada.

[27] Adejumo O.A., Onifade O.C., Albert S. (2021), Kalman Filter Algorithm versus Other Methods of Estimating Missing Values: Time Series Evidence. African Journal of Mathematics and Statistics Studies 4(2), 1-9. DOI: 10.52589/AJMSS-VFVNMQLX.

[28] Poti Abaja Owili, Estimation of Missing Values for BL (p, 0, p, p) Time Series Models with Student-t Innovations, *International Journal of Statistics and Applications*, Vol. 6 No. 3, 2016, pp. 113-122. doi: 10.5923/j.statistics.20160603.04.

[29] Poti Owili Abaja. Imputation of Missing Values for Bilinear Time Series Models (2016). PhD Thesis dissertation.

[30] Rizvi, S. T. H., Latif, M. Y., Amin, M. S., Telmoudi, A. J., & Shah, N. A. (2023). Analysis of Machine Learning Based Imputation of Missing Data. *Cybernetics and Systems*, 1–15. https://doi.org/10.1080/01969722.2023.2247257

APPENDIX

```
# load packages
library(ggplot2)
library(forecast)
# set seed for reproducibility
set.seed(44)
# set parameters
n <- 1000  # number of records
df <- 5    # degrees of freedom the t-distribution
mu <- 0    # mean of the time series
sigma <- 1 # standard deviation of the time series
ncp <- 0   # non-centrality parameter (optional)
NAs <- 100 # number of missing records
```

```
# create time indices
time_index <- 1:n
# generate trend and seasonal components
trend <- 0.1 * time_index
seasonal_component <- sin(2 * pi * time_index / 12)
# simulate the random walk component using t-distribution
random_walk <- cumsum(rt(n, df) * sigma + mu)
# combine the components and create a time series
ts_data <- ts(trend + seasonal_component + random_walk, start = c(2000, 1), frequency = 12)
# Convert time series to data frame
df_ts_data <- data.frame(Time = time_index, Value = as.numeric(ts_data))
# 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 time series using ggplot2
line_graph_t_series <- ggplot(df_ts_data, aes(x = Time, y = Value)) +
geom_line() +
labs(title = "Simulated Non-Stationary Time Series Data with Trend, Seasonality
, and t-Distributed Random Walk",
x = "Time",
y = "Value",
subtitle = paste0("n = ", n,
", df = ", df,
", mu = ", mu,
", sigma = ", sigma,
```

```
", ncp =", ncp)) +
cust_theme()
# print visual
line_graph_t_series
# introduce missing data at random (data frame), ensuring reproducibility
set.seed(44)
missing_indices <- sample(1:n, NAs)
df_ts_data$Value_miss <- df_ts_data$Value
df_ts_data$Value_miss[missing_indices] <- NA
# introduce missing data at random (non-data frame), ensuring reproducibility
ts_data_miss <- ts_data
ts_data_miss[missing_indices] <- NA
# visualize missing Values
df_ts_data <- df_ts_data |>
dplyr::mutate(Missing = dplyr::if_else(!is.na(Value_miss), "No",
dplyr::if_else(is.na(Value_miss), "Yes", "")))
(line_graph_t_series_miss <- df_ts_data |>
ggplot(aes(x = Time,
y = Value,
color = Missing)) +
geom_jitter(size = 1, pch = 20) +
scale_color_manual(values = c("Yes" = "red", "No" = "steelblue")) +
labs(title = "Simulated Non-Stationary Time Series Data with Trend, Seasonality,
and t-Distributed Random Walk Showing Missing Values",
x = "Time",
y = "Value",
color = "Missing Data?",
subtitle = paste0("n = ", n,
", df = ", df,
", mu = ", mu,
", sigma = ", sigma,
", ncp =", ncp,
", NAs =", NAs)) +
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 = df_ts_data,
file = paste0(data_dir, "/simulated_t_series_data.csv"),
row.names = FALS
```