TÜRKİYE
FIRAT UNIVERSITY
GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES

MONTE-CARLO FORECASTING AND OPTIMIZATION OF TIME SERIES DATA USING DATA MINING AND REGRESSION MODELS

Salim Jibrin DANBATTA

Doctoral Thesis

DEPARTMENT OF SOFTWARE ENGINEERING

Division of Software Engineering

APRIL 2022

TÜRKİYE
FIRAT UNIVERSITY
GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES

Department of Software Engineering

Doctoral Thesis

MONTE-CARLO FORECASTING AND OPTIMIZATION OF TIME SERIES DATA USING DATA MINING AND REGRESSION MODELS

Author
Salim Jibrin DANBATTA

Supervisor
Prof. Dr. Asaf Varol

APRIL 2022
ELAZIG

TÜRKİYE
FIRAT UNIVERSITY
GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES

Department of Software Engineering

Doctoral Thesis

Title:	Monte-Carlo Forecasting and Optimization of Time Series Data using Data Mining and Regression Models
Author:	Salim Jibrin DANBATTA
Submission Date:	14 March 2022
Defense Date:	29 April 2022

THESIS APPROVAL

This thesis, which was prepared according to the thesis writing rules of the Graduate School of Natural and Applied Sciences, Fırat University, was evaluated by the committee members who have signed the following signatures and was unanimously approved after the defense exam made open to the academic audience.

		Signature
Supervisor:	Prof. Dr. Asaf Varol Maltepe University, College of Engineering and Natural Sciences	Approved
Chair:	Prof. Dr. Mehmet KAYA Firat University, Faculty of Engineering	Approved
Member:	Assco. Prof. Dr. Fatih ÖZKAYNAK Firat University, Faculty of Technology	Approved
Member:	Assoc. Prof. Dr. Zafer CÖMERT Samsun University, Faculty of Engineering	Approved
Member:	Assoc. Prof. Dr. Ümit CAN Munzur University, Faculty of Engineering	Approved

This thesis was approved by the Administrative Board of the Graduate School on
.... / .... / 20 ....

Signature

Prof. Dr. Kürşat Esat ALYAMAÇ
Director of the Graduate School

DECLARATION

I hereby declare that I wrote this Doctoral Thesis titled “Monte-Carlo Forecasting and Optimization of Time Series Data using Data Mining and Regression Models ” in consistent with the thesis writing guide of the Graduate School of Natural and Applied Sciences, Firat University. I also declare that all information in it is correct, that I acted according to scientific ethics in producing and presenting the findings, cited all the references I used, express all institutions or organizations or persons who supported the thesis financially. I have never used the data and information I provide here in order to get a degree in any way.

29 April 2022

Salim Jibrin DANBATTA

PREFACE

The time-series data generated by multiple applications grow in both number and complexity. The time-series data is an important indicator for business success or failure, which makes modeling and forecasting it a valuable commodity and tool that assist in organizational decision-making

However, the time component makes time series modeling and forecasting problems more difficult to handle. This is mostly because it can disrupt the trend and seasonality pattern of the data, which also creates uncertainty in estimation. This study is limited only to time-series data. Hence, our study is motivated by the need to put forward a method that can capture and describe time series data while considering both trend and seasonal components of the data. Specifically, we focus on overcoming uncertainties in estimation during time series modeling and forecasting.

I would like to express my deepest appreciation and gratitude to my supervisor Prof. Dr. Asaf VAROL, without whose knowledge and assistance this study would not have been possible. I would also like to thank the Turkish Ministry of Culture and Tourism, Japan Tourism Research and Consulting Co, Tourism Malaysia Cooperate, Department of Statistics Singapore, and Candanedo for making the data used in this study available. Nevertheless, my profound gratitude goes to my loving, caring, and wonderful parents Alhaji Jibrin Surajo Dambatta and Hajiya Umma Salisu Muhammad for their love, care, understanding, and prayers. I thank my wife Bilkisu, and my children Ummukhulsum, Hauwa, and Ahmad. Lastly, I thank my siblings Habiba, Halima, Mukhtar, Khadija, Zainab, Rukayya, and Nusaiba for their constant encouragement without which this assignment would not be possible. A big thanks to my friends Ahmad and Abdullahi.

Salim Jibrin DANBATTA
ELAZIG, 2022

TABLE OF CONTENTS

Page

ABSTRACT

Monte-Carlo Forecasting and Optimization of Time Series Data using Data Mining and Regression Models

Salim Jibrin DANBATTA

Doctoral Thesis

FIRAT UNIVERSITY
Graduate School of Natural and Applied Sciences

Department of Software Engineering

April 2022,   Page: xiii + 108

Many different models for time series modeling and forecasting tasks are proposed in the literature. However, research on modeling and forecasting seasonal time series data with trends remains an interesting and challenging task despite these contributions. Lately, in the 21st century, with the outbreak of the COVID-19 pandemic, time-series data experience uncertainty in most possible event outcomes. Organizations are keen on models that would provide them with an insight into the inherent dynamics of their time-series data, especially in predicting the future values of the series, as it is a tool that assists in decision-making.

The best approach would be to have a model that can account for uncertainties in estimation. In this study, we proposed three hybrid time-series regression models: Polynomial-Fourier series, ANN-Polynomial-Fourier series, and ANN-Fourier series models to capture and describe different time-series data. The models combine the polynomial fitting, the Fourier fitting, and the artificial neural network. We ran 100 Monte Carlo simulations within 95% bounds of the models' regression curve to account for prediction uncertainties. The models were applied to many time-series data, which includes the tourism data of Turkey, Japan, Malaysia, and Singapore. The models have proven to be considered worthy candidates for time-series modeling and forecasting in each case. The Monte-Carlo simulation method was successful in predicting uncertain forecast outcomes. A Multi-step ahead forecast suggests a 10.22% increase in foreign visitors arrivals for Turkey in 2021, while Japan, Malaysia, and Singapore will expect a 92.42, 54.81%, and 70.55% decrease in the number of arrivals.

Keywords: Fourier fitting, Polynomial fitting, Monte Carlo simulation, Time series, Modeling, Forecasting

ÖZET

Monte-Carlo Forecasting and Optimization of Time Series Data using Data Mining and Regression Models

Salim Jibrin DANBATTA

Doktora Tezi

FIRAT ÜNİVERSİTESİ
Fen Bilimleri Enstitüsü

Yazılım Mühendisliği Anabilim Dalı

Nisan 2022, Sayfa: xiii + 108

Literatürde zaman serisi modelleme ve tahmin görevleri için birçok farklı model önerilmiştir. Bununla birlikte, bu katkılara rağmen, mevsimsel zaman serisi verilerinin trendlerle modellenmesi ve tahmin edilmesi üzerine araştırma, ilginç ve zorlu bir görev olmaya devam etmektedir. Son zamanlarda, 21. yüzyılda, COVID-19 pandemisinin patlak vermesiyle birlikte zaman serisi verileri, olası olay sonuçlarının çoğunda belirsizlik yaşıyor. Kuruluşlar, karar vermede yardımcı olan bir araç olduğu için, özellikle serilerin gelecekteki değerlerini tahmin etmede, zaman serisi verilerinin doğal dinamikleri hakkında bir fikir verecek modellere meraklıdır.

En iyi yaklaşım, tahmindeki belirsizlikleri hesaba katabilecek bir modele sahip olmak olacaktır. Bu çalışmada, farklı zaman serisi verilerini yakalamak ve tanımlamak için üç hibrit zaman serisi regresyon modeli önerdik: Polinom-Fourier serisi, ANN-Polinom-Fourier serisi ve ANN-Fourier serisi modelleri. Modeller polinom uydurma, Fourier uydurma ve yapay sinir ağını birleştirir. Tahmin belirsizliklerini hesaba katmak için modellerin regresyon eğrisinin %95 sınırları içinde 100 Monte Carlo simülasyonu çalıştırdık. Modeller, Türkiye, Japonya, Malezya ve Singapur turizm verilerini içeren birçok zaman serisi verisine uygulanmıştır. Modellerin, her durumda zaman serisi modelleme ve tahmin için değerli adaylar olarak kabul edildiği kanıtlanmıştır. Monte-Carlo simülasyon yöntemi, belirsiz tahmin sonuçlarını tahmin etmede başarılı oldu. Çok adımlı bir tahmin, 2021'de Türkiye'ye gelen yabancı ziyaretçi sayısında %10.22'lik bir artış olduğunu gösterirken, Japonya, Malezya ve Singapur, gelen sayısında %92.42, %54.81 ve %70.55'lik bir düşüş bekliyor.

Anahtar Kelimeler: Fourier uydurma, Polinom uydurma, Monte Carlo simülasyonu, Zaman serileri, Modelleme, Tahmin

LIST OF FIGURES

Page

LIST OF TABLES

Page

LIST OF APPENDICES

Page

SYMBOLS AND ABBREVIATIONS

Symbols

B : Backshift Operator
α : Intercept Term Estimation
β : Lag Coefficient of Model Estimation
ϕ : Coefficient of Moving Average and Model Parameter

Abbreviations

ACF : Autocorrelation
ADH : Augmented Dicky-Fuller Test
AIC : Akaike Information Criterion
ANN : Artificial Neural Network
AR : Autoregressive
ARIMA : Autoregressive Integrated Moving Average
ARMA : Autoregressive Moving Average
d : Number of Differencing
MA : Moving Average
MAPE : Mean Absolute Percentage Error
MER : Mean Estimation Error
MSE : Mean Square Error
p : Order of the Autoregressive Term
PACF : Partial Autocorrelation
q : Order of the Moving Average Term
R2 : R-Squared
RMSE : Root Mean Square Error

1. INTRODUCTION

Time series data has numerous definitions in Literature. For example, time-series data has been defined as any data collected over time at different points in time [1]. It is also referred to as a time-stamped sequence of data and indexed in time order [2] and [3]. Contrary to the data collected at a single point in time, also known as cross-sectional data [4]. In describing time series data, the standard practice in the literature is to split the series into the trend, cyclic, and seasonal components [5]. This practice is also known as time series decomposition. For simplicity, when time series decomposition occurs, we combine the trend and cyclic into one component, also known as a trend [6], [7], and [8]. One of the most efficient methods used in decomposing time series data is the seasonal and trend decomposition using loess (STL) [9]. The STL method developed by Cleveland et al. [10] has an advantage over the classical method by decomposing the time series data into a trend, seasonal and residual components.

Ideally, the series behavior is represented by the series trend and seasonal patterns for time series modeling and forecasting. This is because the trend and seasonal components are the stable and predictable components of the series. Thus, isolating these patterns in time series data is the primary goal of time series analysis [11]. Many different types of models for time series modeling and forecasting tasks are proposed in the literature. However, despite these contributions, research on modeling and forecasting seasonal time series data with trends remains one of the exciting and challenging tasks [12]. The critical challenge of the recent empirical and well-thought-out models, especially the machine learning methods for modeling and forecasting time series data, is the problem of overfitting [13] and [14].

This chapter presents the general overview of the thesis study. The chapter also illustrates the motivation of the thesis study and the problem definition. Moreover, the objectives and contributions of the work are discussed in this chapter. More details about the concept of time series data modeling and forecasting are presented in section 2.1.

1.1. Motivation

As the world gets technologically advanced, various industrial systems relentlessly produce streams of time series data that application is almost everywhere. The time-series data generated by multiple applications grow in both number and complexity [15] and [16]. Perhaps, this is what prompted researchers the development of different data mining techniques to assist in describing, analyzing, and extracting knowledge from the time series data. Because time is an important factor in determining the success or failure of our business activities and daily life, this makes time series data one of the most valuable commodities in the current information age we are living in.

For example, the outbreak of the COVID-19 pandemic in March 2020, as declared by the United Nations world tourism organization [17], shows how time-series data collection can affect our daily lives. This is evident from worldwide high demand and interest in time series data considering billions of people worldwide seeking information on the daily trend of the COVID-19 statistics in their countries [18]. Moreover, modern businesses and government organizations are racing to extract useful information by analyzing time-series data. Perhaps this is why time series databases are recently categorized as the fastest-growing database [19]. The rapid growth of time series data has drawn the attention of many organizations across the globe. They are often curious to know their activities' success or failure rate to invest in time series data analysis and forecasting [20] heavily.

This work is motivated by the need to model and forecast time series data while considering both the trend and seasonal components of the series and uncertainty in the data. A periodic time series data can also have a trend component or a trended series to have an element of seasonality. By its nature, time-series data mostly do not show linearity but rather show periodic spirit with the linear and or nonlinear trend. Capturing uncertainty possibility using a unique technique has become imperative while dealing with time-series data. This is because time series data are primarily collected in real-time, making future values vulnerable to many factors. For example, the uncertainty in the future number of international tourist arrival has plunged due to the outbreak of the COVID-19. Hence, a model alone might not capture enough uncertainties in the future values of the series using the past series values.

1.2. Problem definition and the main objective

Many studies have proposed modeling and forecasting time series data using approaches that best fit for trended or seasonal series. For example, the analyses of Masson et al. [20], Lewis et al. [21], Lange and Zeger [22], and Smart [23] show that the Fourier series method is among the best approach to modeling periodic data. However, suppose the seasonal time-series data have a trend component. In that case, a special technique is required to capture the trend component. This is probably why Ampaw et al. [25] obtained an undecent result after applying the ARIMA model to seasonal rainfall data without considering the trend of the series. This indicates that time series with a seasonal component can not be adequately represented and captured using the ARIMA model.

Similarly, most of the classical methods proposed in the literature do not adequately capture the nonlinear trend in the series because capturing the nonlinear trend also requires a particular technique. Additionally, the methods might not capture enough uncertainty in the future values of the series. The primary objective of this work is to design models that can adequately capture the trend and seasonal components of time series data. Nevertheless, the work aims at designing a Monte Carlo simulation scheme to generate distribution output that can reflect the uncertainties in the model result. The Monte Carlo simulation scheme is expected to capture all of the uncertainty in the future values of the series.

1.3. Research contribution

Time series data modeling and forecasting in the literature are mostly done with the presumption of linearity [24], which uplift the high use and implementation of linear models such as the ARIMA model. However, the stream of time series data arising in nature assumes a periodic or seasonal pattern with a linear and nonlinear trend [25], [26], and [27]. Lately, in the 21st century with the outbreak of the COVID pandemic, time-series data experience uncertainty in most possible event outcomes. Therefore, the best approach would be to have a model that can account for uncertainties in estimation.

Herein, we focus on overcoming these uncertainties in estimation during time series modeling and forecasting. Specifically, we present three novel hybrid models, "Polynomial-Fourier series," "ANN-Polynomial Fourier series," and "ANN-Fourier series," which are a combination of the polynomial fitting, Fourier series fitting, and the ANN to study different time-series data:

1.3.1. Polynomial fitting

In modeling and forecasting time series data, we use the polynomial fitting when presented with a polynomial expression of the relationship between the dependent and independent variables. Given a set of observations y_n and lag count x_n, we can express the relationship between x and y as $y^{r(x)} (r < n)$;

$$y^{r(x_i)} = a_0 + a_i x_i + a_2 x_i^2 + \cdots + a_{r-1} x_i^{r-1} + a_r x_i^r \tag{1.1}$$

Where $a_0, a_1, \dots, a_r$ are the regression coefficients that can be obtained using the least square method [28] and [29]. While n is the number of the data points. The value $e_{pi} = | y_i - y^r_{(x_i)} |$ denotes the residual between the fitted value $y^r_{(x_i)}$ and the original observation y_i. Hence, the error summation with regard to the predicted values is obtained as $e_p = \sum_{i-1}^{n} e_{pi}$.

1.3.2. Fourier fitting

In the case of capturing the seasonal component of the series, we represent the seasonal component using the Fourier series. Given a set of observations y_i that is observed at the time x_i, i = 1, …, n, the representation of the Fourier series fitting will give [30]:

$$y_{x_i}^{r} = a_0 + \sum_{j=1}^{r} a_i \cos(jwx) + \sum_{j=1}^{r} b_j \sin(jwx) \quad (1.2)$$

The Fourier fit error is represented as $e_{fi} = |\, y_i - y_{(x_i)}^{r} \,|$, and the sum of Fourier errors $e_f = \sum_{i=1}^{n} e_{fi}$. In equation 1.2, r is the number of Fourier series terms, respectively a_j and b_j are Fourier series coefficients, while w is the data frequency.

1.3.3. The ANN models

The adopted ANN model used in this study is the single-layer perceptron artificial neural network model consisting of input, hidden, and output layers. The mathematical representation of the feedforward ANN used in this study is given as :

$$y_t = g(w_0 + \sum_{j=1}^{q} w_j \,{}^{\circ}f(\sum_{i=1}^{p} w_{ij}\, y_{t-i} + w_{oj})) \quad (1.3)$$

1.3.4. The Monte Carlo scheme

To account for uncertainty in the estimation, we applied the Monte Carlo simulation scheme to the residue of the models' forecast. The aim is to simulate multiple possible values that yield different outcomes for a given forecasting period. The scheme works by first getting the bounds for the estimated coefficients. For each coefficient, collecting a random sample from the bounds using a suitable sampling distribution, applying these randomly sampled values to the model equation, and estimating a new curve Monte Carlo simulation path. The study highlights the significance of modeling residuals from classical or empirical models to improve forecast performance.

1.4. Conclusion

The chapter presents the general overview of time series modeling and forecasting, the overview of this thesis study, our motivation, the problem definition, our objectives, and our thesis study contribution. The remaining chapters of this study are structured as follows. Chapter 2 presents the theoretical concepts of time series modeling and forecasting, ARIMA model, Fourier series fitting, Monte Carlo simulation scheme, and the artificial neural network design. In chapter

3, some of the relevant related works of the theoretical concepts are found in the literature. While chapters 4 and 5 contain the central backbone of this study. They present the proposed models and methodology of the research and the study's results and discussions. Finally, chapter 6 provides the summary and conclusion of the thesis study. At the same time, the recommendation for organizations and future work can be found under the recommendation section.

2. THEORETICAL CONCEPTS

This chapter discusses and presents the background knowledge of the theoretical concept related to this study. In this chapter, section 2.1 explains the concept of time series modeling and forecasting. In contrast, sections 2.2 and 2.3 discussed the concept of the ARIMA family and Fourier series regression modeling techniques. The concept of the Monte Carlo simulation scheme is explained in section 2.4. The machine learning concept of artificial neural network algorithms is discussed in section 2.5.

2.1. Time Series Modeling and Forecasting

The time factor is counted as a significant game-changer that ensures business success. Although keeping up with the pace of time is seemly to be difficult, technological advancement has provided us with some methods that can be used to see things ahead of time. Time series forecasting involves working on time-based data (year, month, day, hour, and minutes) to extract meaningful information, statistics, and other data characteristics for decision making, as explained by Agrawal et al. in [31]. The extracted information, which describes the time series's inherent structure, is often referred to as a model. We use this model to forecast and generate future values of the series.

The two major components that make time series forecasting a challenge are the time dependence of a time series and seasonality in a time series. Because a time series is time-dependent, the basic assumption of linear regression models is that independent observations do not hold in time series. While due to increasing and decreasing trends, most time series have some form of variation specific to a particular period.

2.1.1. Pattern in a Time Series

Time-series data can be decomposed into the base level, trend, seasonality, and error components. We can locate a trend in data when we see an increasing or decreasing slope in the series. On the other hand, seasonality is present in time-series data when we observe a particular repeated pattern among regular intervals such as a month of the year, the day of the month, weekdays, or even times of the day. Even so, not all-time series data have trend and seasonality. A time series may not have a trend but could have seasonality or vice versa. Another factor related to the time series is cyclic behavior. Most of the time, cyclic behavior is confused with seasonality. However, we should note that cyclic occurs when the series's rise and fall pattern does not happen in fixed calendar-based intervals.

Observations in a time series data can be modeled as an additive or multiplicative depending on the nature of the trend and seasonality of the series. Classical decomposition of a time series can be done by considering the series to additive or multiplicative of the base level, trend, seasonal index, and residue.

Additive time series = Base Level + Trend + Seasonality + Error

Multiplicative time series = Base Level x Trend x Seasonality x Error

2.1.2. Stationarity in Time Series

In time series analysis, determining if a time series is stationary is essential. Time series stationarity is a crucial concept that analytical tools, statistical tests, and models rely upon. A stationary series is a series that is not time-dependent or a function of time. This means that a stationary series has statistical properties such as mean, variance, autocorrelation, etc, that do not change over time. We can accept a series as a stationary series if the series does not exhibit trend or seasonality effects. In the Python programming language, we can extract trend, seasonality, and residuals or error using the seasonal_decompose() function to splits the time series into seasonality, trend, and error components. For example, Figure 2.1 shows the decompose component that includes the real data, trend, seasonality, and residuals of monthly visitors to Turkey from the January 2014 to January 2020 dataset.

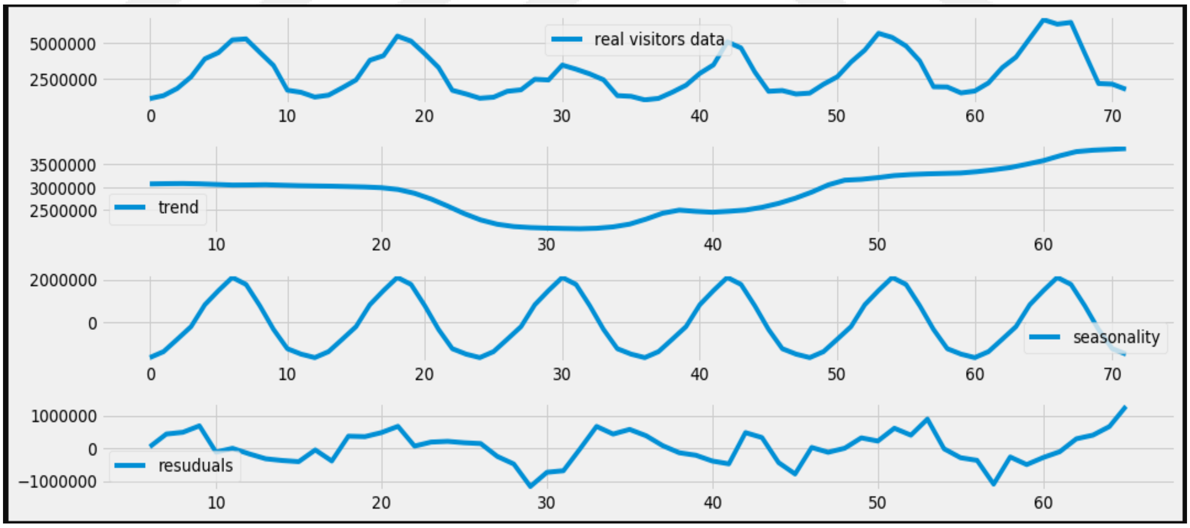


Figure 2.1. Component of a time series

Most statistical forecasting techniques and models, such as the regression and ARIMA models, are designed to work on a stationary time series. Making a time series stationary is simple. The first step of the forecasting process is transforming a non-stationary series into a stationary one. A non-stationary series can be transformed into a stationary series by either differencing the series, taking the series log, taking the nth root of the series, or combining any of the three options.

However, differencing is a well-known and convenient method to stationaries a time series dataset, as highlighted by Gruijter et al. [32]. For example, to differentiate a series, if Y_t is the value at a particular time t, then our first difference for $Y = Y_t - Y_t - 1$. To put it in a nonprofessional term, differencing is a process of subtracting the next value by the current value. While differencing a series, if, for example, the first differencing does not make the series stationary, we can go for one more differencing term and so on until the series is finally stationary. For example if Y is a series Y = [1, 5, 2, 12, 20] then the first differencing of Y will be Y = [5-1, 2-5, 12-2, 20-12] = [4, -3, 10, 8] and the second differencing will be Y = [-3-4, -10-3, 8-10] = [-7, -13, -2].

Time series forecasting is more accessible when the series is stationary, and the forecast is expected to be more reliable. One of the primary reasons as to why we should make a non-stationary series into a stationary series is that autoregressive forecasting models are fundamentally linear regression models and we know from theory linear regression models use lags of the series itself as the predictors. As such, stationary series comes with the benefit of removing any persistent autocorrelation, rendering the predictor lags of the forecasting models nearly independent.

Chaudhari and Choudhari [33] explain how the stationary of a series can be established by looking at the plot of the series. Nevertheless, unit root tests such as augmented Dickey-Fuller test (ADH test), Kwiatkowski-Phillips-Schmidt-Shin – KPSS test (trend stationery), and Philips-Perron test (PP Test) are statistical tests that can be applied on a time series dataset to quantitatively determine if a given series is stationary or not. The ADH test, which null hypothesis states that there is a unit root in the time series and the series is not stationary is the adopted unit root test for this study. We can only reject the null hypothesis in the augmented Dickey-Fuller hypothesis (ADH) if the obtained p-value of the ADH test is less than the significance level value (usually 0.05).

2.1.3. Autocorrelation and Partial autocorrelation in Time Series

Hyndman [5] define autocorrelation as a measure of the linear relationship between lagged values of a time series, and partial autocorrelation as a pure correlation of the time series with its lag and without the correlation contribution from intermediate lags. When we look at the autocorrelation of monthly visitors to the Turkey dataset in Figure 2.2, we will notice several autocorrelation coefficients that correspond to each panel in the lag plot.

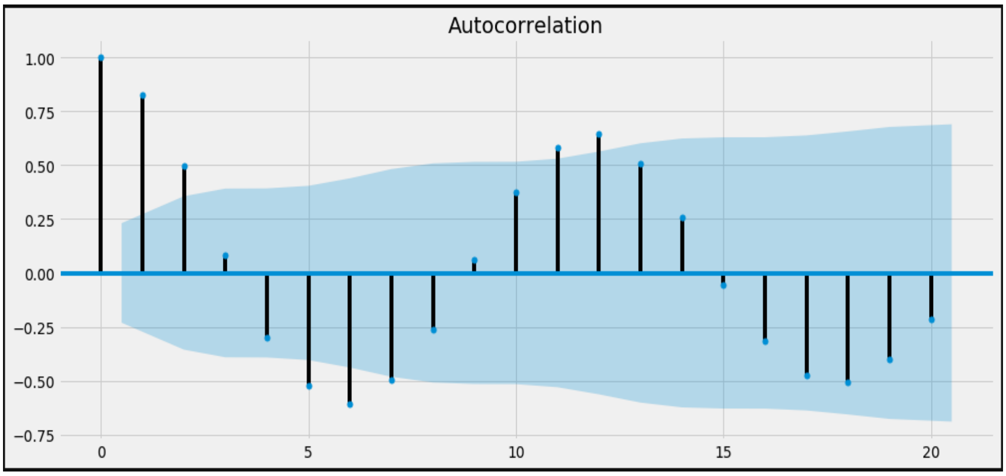


Figure 2.2. Autocorrelation plot of monthly visitors to Turkey dataset

The series trend and seasonality can be observed from the autocorrelation plots. If a trend exists in the data, the autocorrelation of small lags will be significant and positive. This is because nearby observations in time are also nearby observations in size. In short, a trended data series autocorrelation tends to get positive values that decrease slowly as the lags increase. Meanwhile, suppose the time series is seasonal. In that case, we can see the autocorrelation being more significant for the seasonal lags (occurring at a particular frequency) than the other lags. However, a series with both trend and seasonality will show the combination of the trend and seasonality effects. Figure 2.3 shows the partial autocorrelation plot of the monthly visit to the Turkey dataset. Partial autocorrelation is used for identifying the order of an autoregressive (AR) model. For example, suppose the autocorrelation plot indicates that the AR model could be appropriate for the series. In that case, a partial autocorrelation plot is used to identify the fittest order.

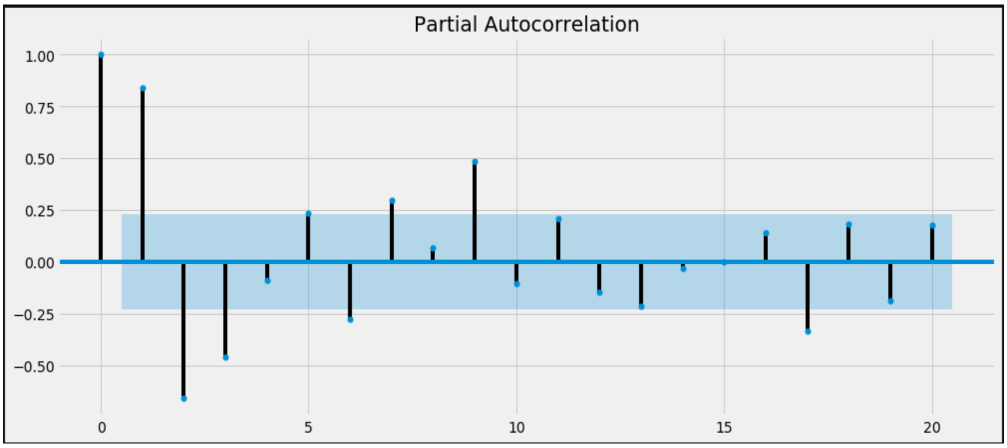


Figure 2.3. Partial autocorrelation plot of monthly visitors to Turkey dataset

2.2. ARIMA Model

The autoregressive integrated moving average (ARIMA) model is a statistical model that uses a time series of past values to forecast the series of future trends [31]. The main idea behind the ARIMA forecasting algorithm is that using only the time series previous or past values information, we can forecast the future values of the series. ARIMA model is characterized by three terms, namely p, d, and q, where p is the order of the autoregressive (AR) term, d is the required number of differencing needed for the series to be stationary, and q is the order of the moving average (MA) term. It is important to note that if a time series shows seasonal patterns, then a seasonal term must be added to the ARIMA so that it will become a seasonal autoregressive integrated moving average (SARIMA). In theory, p refers to the number of series lags to be used as predictors. Q refers to the number of lagged forecast errors used in the ARIMA model. At the same time, the value of d is the minimum number of differences needed to achieve a stationary series, thus, d = 0 for stationary series.

When we forecast a series using the AR model, it means that we are predicting our variable of interest using a linear combination of its past values. A model that Y_t only depends on its previous lags of Y'_t is known as the pure AR model, and the AR model of order p is shown in equation 2.1.

$$AR(1),\ y_t = \alpha + \beta_1\, y_{t-1} + \epsilon_t$$
$$AR(2),\ y_t = \alpha + \beta_1\, y_{t-1} + \beta_2\, y_{t-2} + \epsilon_t$$
$$AR(p),\ Y_t = \alpha + \beta_1\, Y_{t-1} + \beta_2\, Y_{t-2} + \cdots + \beta_p\, Y_{t-p} + \epsilon_1$$
$$AR(p),\ y_t = \alpha + \sum_{i=1}^{p} \beta_i\, y_{t-i} + \epsilon_t \qquad (2.1)$$

Where y_{t-1}, y_{t-2}, and y_{t-p} are the series lags, while β_1, β_2, and β_p are the lag regression coefficient of the model estimation, and α is the intercept term estimation of the model. Similarly, in a pure MA model, Y_t only depends on the lags of the forecast errors. An MA model of order q is shown in equation 2.2.

$$MA(1),\ y_t = \mu + \phi_1\, \epsilon_{t-1} + \epsilon_t$$
$$MA(2),\ y_t = \mu + \phi_1\, \epsilon_{t-1} + \phi_2\, \epsilon_{t-2} + \epsilon_t$$
$$MA(q),\ y_t = \mu + \phi_1\, \epsilon_{t-1} + \phi_2\, \epsilon_{t-2} + \cdots + \phi_q\, \epsilon_{t-q} + \epsilon_t$$
$$MA(q), y_t = \mu + \epsilon_t + \sum_{i=1}^{q} \phi_i\, \epsilon_{t-i} \qquad (2.2)$$

Where $\epsilon_{t-1}, \epsilon_{t-2}$, and ϵ_{t-q} are the forecast error terms, while μ is the mean of the series, is ϵ_t white noise and ϕ_1, ϕ_2, and ϕ_q are the lag regression coefficient of the model estimation.

An ARMA model is the model that forecasts a series using the combination of the AR and MA models which is used for describing weak stationary stochastic time series. The ARMA model is also known as ARMA(p,q) where: p is the order of AR and q is the order of MA respectively. The ARMA equation as described by Lee and Wu [34] is given in equation 2.3

$$ARMA(p,q),\ y_t = \alpha + \epsilon_t + \sum_{i=1}^{p} \beta_i\, y_{t-i} + \epsilon_t + \sum_{i=1}^{q} \phi_i\, \epsilon_{t-i} \tag{2.3}$$

An ARIMA model differs from the ARMA model by having the “integrated” part which refers to the number of differencing needed to make the series stationary. ARIMA which is a combination of the AR and MA terms is the one where the time series is differenced at least once to make the series stationary, and it forecast time series data by first back fitting to historical data and then forecasting the future using the backshift operator (B) as described by Newbold [35]. In time series analysis, the backshift operator (B) operates on an element of a time series to produce the previous element.

For example, given some time series Y = (Y1, Y2, ….)

then backshift operator B: $BY_t = Y_{t-1}$ for all t > 1

$B(BY_t) = B(Y_{t-1}) = B(Y_{t-2})$

Suppose Y_t is the first difference then $Y_t = y_t - y_{t-1}$, but $y_{t-1} = B(y_t)$

$= yt - B(yt) = yt\,(1 - B)$

Second difference $Z_t = Y_t - B(y_t)$

$= (1 - B)\, Y_t$

$= (1 - B)(1 - B)\, y_t$

$= (1 - B)\ 2\ Y_t$

nth difference = n = $(1 - B)\ n\ Y_t$

In other words, the ARIMA equation of Y_t = constant + linear combination of lags of Y up to p lags + linear combination of lagged forecast errors up to q lags. ARIMA (1,1,1) is described in the work of Weiqiang and Zhendong [36] as shown in equation 2.4 where p,d,q are all equal to one can be obtained as follows:

AR(1): $y_t = \alpha + \beta_1\, y_{t-1} + \epsilon_t$

MA(1): $y_t = \boldsymbol{\mu} + \epsilon_t + \phi_1\, \epsilon_{t-1}$

$y_t = y_t + y_t$

$y_t = (\alpha + \boldsymbol{\mu}) + \beta_1\, y_{t-1} + \epsilon_t - \phi_1\, \epsilon_{t-1}$

$y_t = (\alpha + \boldsymbol{\mu}) + \epsilon_t + \beta_1\, B(y_t) - \phi_1\, B(\epsilon_t)$

$y_t = (\alpha + \boldsymbol{\mu}) + (1 - B\phi_1)\,\epsilon_t + \beta_1 B(y_t)$

Assume you differenced the series once $y_t = (1 - B)Y_t$

$$(1 - B)\,Y_t = (\alpha + \boldsymbol{\mu}) + (1 - B\phi_1)\,\epsilon_t + \beta_1 B(y_t)$$

$$(1 - B)\,Y_t - \beta_1 B(y_t) = (\alpha + \boldsymbol{\mu}) + (1 - B\phi_1)\,\epsilon_t$$

$$(1 - B)\,Y_t - \beta_1 B(1 - B)Y_t = (\alpha + \boldsymbol{\mu}) + (1 - B\phi_1)\,\epsilon_t$$

$$(1 - B)\,(1 - \beta_1 B)\,Y_t = (\alpha + \boldsymbol{\mu}) + (1 - B\phi_1)\,\epsilon_t \qquad (2.4)$$

The objective in ARIMA is to find the values of p, d, and q. Let's start the discussion with how to find differencing order d in the ARIMA model. We should also recall that the purpose of finding d is to make the series stationary, yet, we must avoid over-differencing a series as an over differenced series may still be stationary which can affect the model parameters. Given this, getting the right order of d is getting a near stationary series that roams around the mean, which its ACF plot reaches to zero quickly. We can determine that a series needs further differencing if its ACFs are positive for many lags, for example, 10 or more. However, a series is over differenced if lag 1 of the ACF is too negative. The order that gives the least standard deviation is the one we should choose when we cannot decide between two orders of differencing.

Differencing is only needed for a non-stationary series; otherwise, the value of d is zero. The stationarity of a series can be determined using the ADH test. Therefore, if the p-value of the ADH test is less than the significant value (0.05), the ADH null hypothesis is rejected, and the series is considered to be stationary; thus, proceed to find d. On the other hand, if the p-value is greater than the significant value (or test statistics), the series is differenced to observe how the ACF plot will look.

The next step is to find p also known as the required AR term by inspecting the partial autocorrelation (PACF) plot. Since the PACF plot shows the pure correlation between the series and a lag, it will enable us to determine if a lag is needed for the AR term. To find the required number of AR terms we take the order of the AR term to be equal to as many lags that crosses the significance limit in the PACF plot. When a series is somehow slightly under difference, it can be reconciled by adding one or more AR terms. Similarly, the ACF plot is used to find the MA terms' order q, which is the lagged forecast error. The ACF will allow you to determine the required number of the MA terms that are needed to remove autocorrelation in a given stationary series. Likewise, a slightly over differenced series can be reconciled by adding MA terms. Having defined and determined the values of p, d, and q we have got everything to fit the ARIMA model.

2.3. Fourier Time-Series Fitting

The Fourier series was first introduced by Joseph Fourier in the year 1804, as reported by Soria et al. in [37]. The idea behind the Fourier series is that any periodic function can be represented using a harmonically related series of sines and cosines. These periodic functions in the Fourier series are the representation of the possibly finite sum of sine and cosine functions. Another alternative definition of the Fourier series provided by Yang [38] is that the Fourier series is an infinite series where the constant of terms is multiplied by the sine or cosine functions of the integer multiples of the variable and used in the analysis of periodic functions.

2.3.1. Fourier Series Function

The provided Fourier series definitions above show that the curve can be represented using the Fourier series for any given curve that repeats itself. For example, given in Figure 2.4 is a Fourier function f(x), representing the regression model that should be used to capture the data behavior for any value of x.

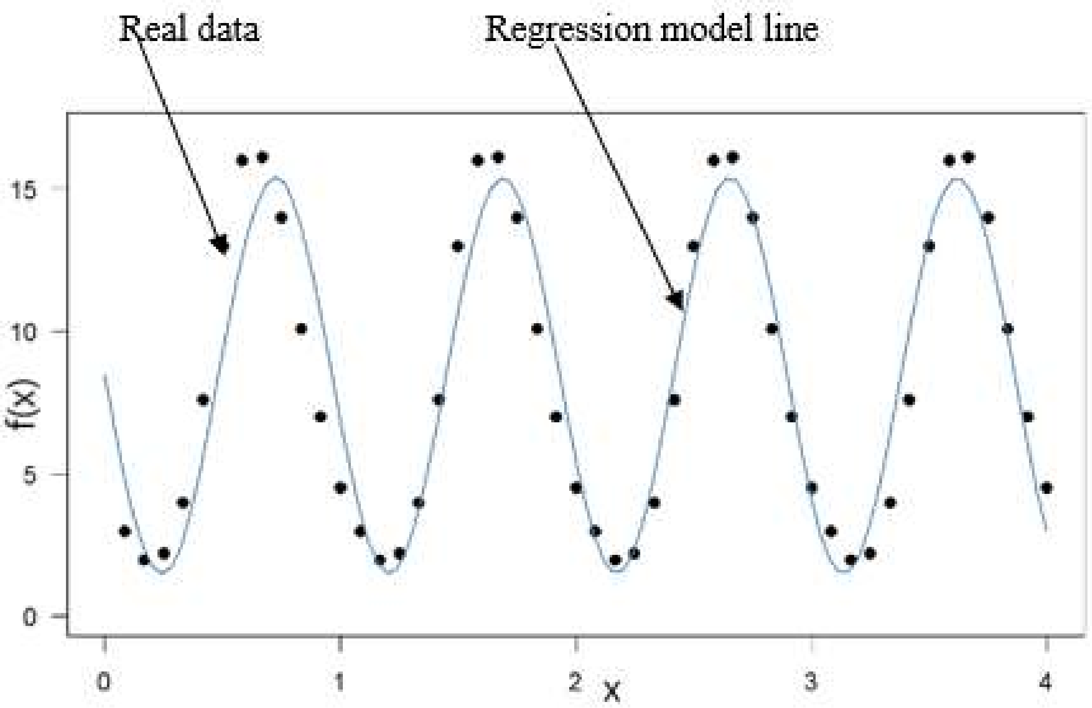


Figure 2.4. Fourier series function

In the case of Fourier, the function f(x) from Figure 2.4 is represented using the summation of sine and cosine together with their coefficient. Furthermore, x in Figure 2.4 is the representation of the time counts or rather known as lag count. The Fourier function f(x) as explained by Pollock [39] is obtained as;

$$f(x) = a_0 \cos(0\omega x) + b_0 \sin(0\omega x) + a_1 \cos(\omega x) + b_1 \sin(\omega x) + a_2 \cos(2\omega x) + b_2 \sin(2\omega x) + \cdots + a_n \cos(n\omega x) + b_n \sin(n\omega x)$$

$$f(x) = a_0 + a_1 \cos(\omega x) + b_1 \sin(\omega x) + a_2 \cos(2\omega x) + b_2 \sin(2\omega x) + \cdots + a_n \cos(n\omega x) + b_n \sin(n\omega x)$$

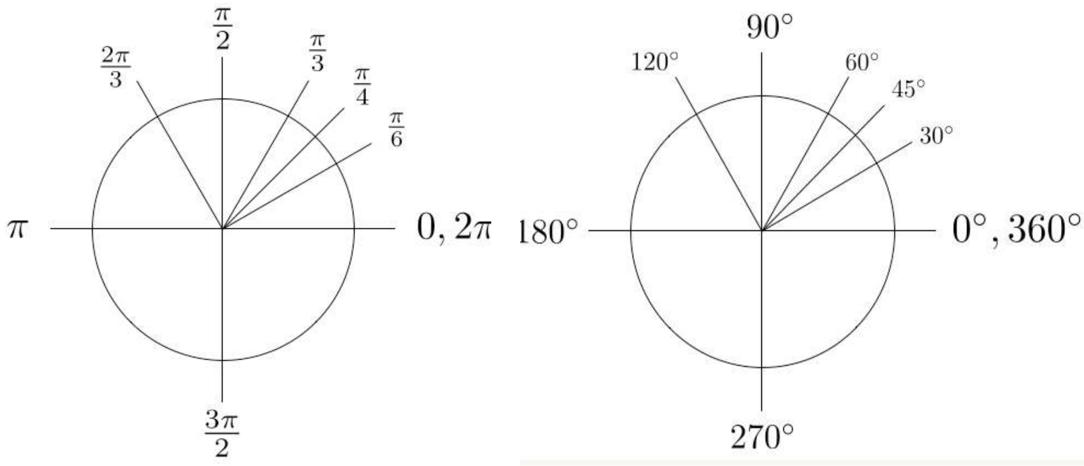
$$f(x) = a_0 + \sum_{n=1}^{\infty} a_n \cos(n\omega x) + \sum_{n=1}^{\infty} b_n \sin(n\omega x) \tag{2.5}$$

Where n is the number of terms in the series, and ω is the angular velocity, which we will see how it can become angular frequency to be used for time series in the next section. While a_0, a_n and b_n are the parameter coefficients of the series.

2.3.2. Angular Frequency in Fourier

The Fourier series applies to periodic signals and the time domain used in the series is periodic and continuous. Dudek et al. [40] explained how the periodic signals have a frequency spectrum that has harmonics. For example, a time-domain that repeats at 1000 hertz will have a frequency spectrum of 1000 hertz, 2000 hertz, 3000 hertz, etc. at first, second, and third harmonics with the first harmonic also known as the fundamental frequency. The study further states that frequencies between harmonics can be considered to have zero value or simply do not exist. The important point is that it does not contribute to the creation of the time domain signal.

In this study, for simplicity, we will explain how to obtain angular frequency ω using cycle angular distance ɵ, frequency f, and period T. Some special angels in radians and degrees as can be seen from Figure 2.5 are shown to explain better and understand how to obtained ω, ɵ, f, and T.



A B

Figure 2.5. A some special angles in radians. B some special angels in degrees

The angular velocity ω is expressed as angular distance ɵ per time t as shown in equation 2.6, while the frequency is expressed as the number of cycles (n) in a given time interval as can be seen in equation 2.7

.

$$\omega = \frac{\theta}{t} \tag{2.6}$$

$$f = \frac{n}{t} \tag{2.7}$$

Now, if we consider one full cycle, the time taken to complete one full cycle is known as period (T) and the angular distance in radians covered for one full cycle as can be seen in Figure 2.5 is given as ɵ = 2π. Therefore, equation 2.7 becomes;

$$\omega = \frac{2\pi}{T} \tag{2.8}$$

Where T is the period, and if we approach equation 2.7, in the same way, having n = 1, and t = T then we obtain equation 2.9 as;

$$f = \frac{1}{T} \tag{2.9}$$

Putting together equations 2.8 and 2.9, we get the angular frequency equation for ω in equation 2.10, which is the one we use in the Fourier function as seen in equation 2.7.

$$\omega = 2\pi f \tag{2.10}$$

2.3.3. Fourier Coefficients

The purpose of the number of terms (n) in equation 2.5 is to make the function f(x) capture more cycles in the data. To fit the Fourier function from Figure 2.4, x is considered as the time counts (i.e. x = t). This is because the Python programming language will always need to count the time as explained by Gashler and Ashmore in [41]. For example, in a monthly-based time-series dataset, the months will be the time counts t in Python.

Now, we should go through how to estimate the Fourier coefficients a_n and b_n from equation 2.5. This parameter or coefficient estimation is where the regression comes into play. There are various techniques available for parameter/coefficient estimation schemes such as those discussed in [42] and [43], but the most popular one is the least square fitting. Given in Figure 2.6, is a scatter line chart for the set of values of Y and time counts (t) that we will use to explain the least square fitting.

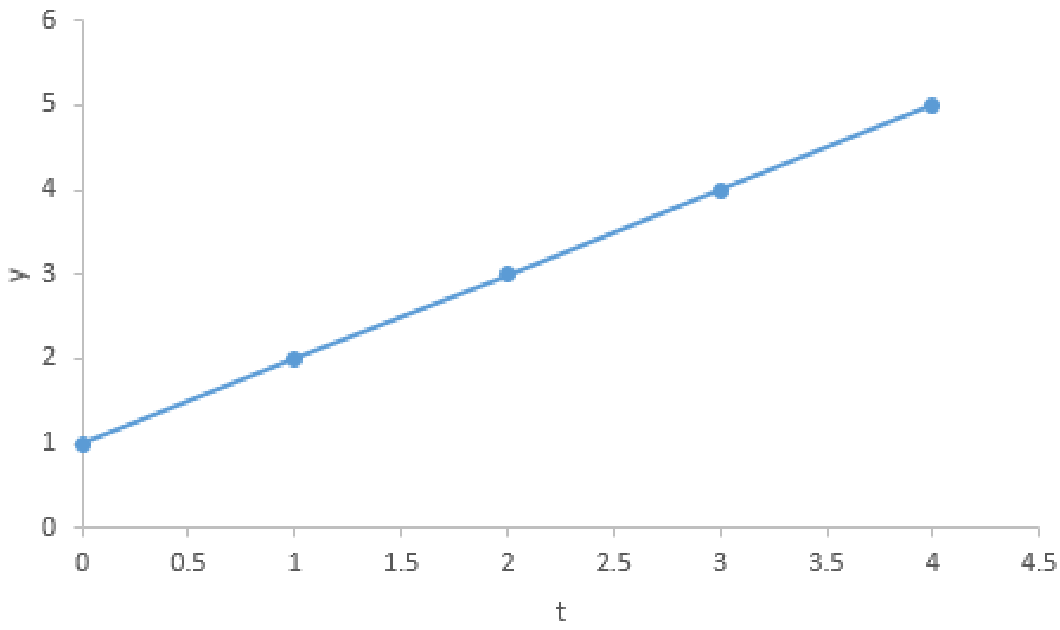


Figure 2.6. Scatter line chart for the set of values of Y and time counts (t)

To do the regression in the least square fitting, we begin by guessing what function we should use. When we look at Figure 2.6, we can elect to choose the function in equation 2.11.

$$f(t) = a_0 + a_1 t \tag{2.11}$$

We choose to go with the function in equation 2.11 because the nature of the data, which is a straight line, looks linear. To find the Fourier coefficient a_0 and $a_{1,}$ we need to do trial and error according to the following steps:

1) Guess any value for a_0 and $a_{1,}$

   Let a_0 = 2 and $a_{1,}$ = 1

   $$f(t) = 2 + t \quad (2.12)$$

2) Substitute any t value in $f(t)$ to get result number for equation 2.13. Example, if t = 2, then from equation 2.13 $f(2) = 2 + 2 = 4$.
3) Error = $| f(2) - Actual|^2$

We require the value with the least error from among the testing set of values in the least square fitting. This set of values for a_0 and a_1 with the least errors in fitting the curve are represented as a 1d array of vectors. Each of the coefficients $a_0, a_1, a_2, \ldots\ a_n$ have its estimated values with lower and upper estimation bounds. These estimated coefficients are the coefficients that we use in the main regression formula in equation 2.11. For example, if we get a_0 and a_1 as 1 and 2 respectively then from equation 2.11, we have;

$$f(t) = 1 + 2t$$

Therefore, for prediction, we only need to create an array of the t values starting from 1 to n, where n is the number of terms we are interested in.

2.4. Monte Carlo Scheme

The Monte Carlo scheme is a random way of simulating time series data paths. These simulations are used for forecasting and critical decision-making on future values and events of a series. Singhee and Rutenbar [44] described the Monte Carlo scheme as a method of solving deterministic problems and the computation of the expectation value of a random variable. According to the explanation of Malik in [45], the random numbers generated in the Monte Carlo simulation have similar properties to the risk factors which the simulation is attempting to simulate.

The fundamental idea behind using the Monte Carlo method is to run simulations repeatedly to get the number of possible outcomes of a given scenario over a large number of time counts in the series, which as a result, produces a large number of possible outcomes and probabilities of the variables. Also, the distribution of the probability of an unknown stochastic entity can be obtained using the Monte Carlo simulation. The design of the Monte Carlo scheme requires a powerful random number generation algorithm such as the Brownian motion algorithm and Markov chain

Monte Carlo algorithm. The design also requires the distribution from which the sample is drawn. This design can be applied to unstructured (un-modeled) and structured (modeled) data.

2.4.1. Monte Carlo Scheme for Un-Modelled Data

As previously discussed, the Monte Carlo scheme can be applied to un-modeled data by finding the distribution to which the data belongs and drawing a random sample m_i from the distribution. Given in Figure 2.7 is a sample data plot for un-modeled data, which we will use to demonstrate the Monte Carlo scheme for un-modeled data.

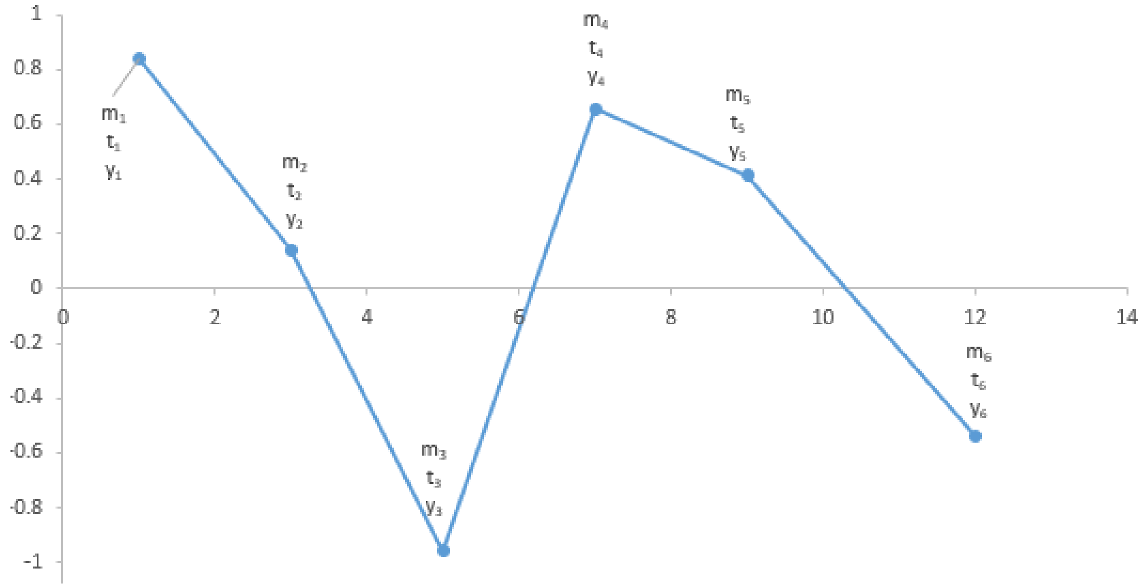


Figure 2.7. A random sample of an un-modeled data plot

From Figure 2.7 m_1, m_2… m_6 is the random sample of the data, while t_1, t_2…t_6 is the time counts, and y_1, y_2…y_6 are the lags of the data. If μ is the mean of the data and σ is the standard deviation of the data then the Monte Carlo of the data without random shock and with random shock is given in Equations 2.13 and 2.14 respectively.

$$lag_{ti} = \mu - y_{ti} + m_i \quad (2.13)$$

Random shock can be applied as $shock = \sigma \exp(\mu \Delta t) + M_i$

$$lag_{ti} = y_{ti} + shock \quad (2.14)$$

2.4.2. Monte Carlo Scheme for Modelled Data

As for the case of model data such as the one in this study, we can choose boundaries especially within ±2 of our model curve so that if $lag_{ti} > y_t + 2\sigma$ then $lag_{ti} = y_t + 2\sigma$ else if $lag_{ti} < y_t - 2\sigma$ then $lag_{ti} = y_t - 2\sigma$.

If $y_t = a_0 + a_1 y_{t-1} + a_2 y_{t-2}$ is a regression model then {a_0, a_1, a_2} are the regression coefficients estimated by software. The ranges of the regression coefficients are given as:

$$a_{0lb} \leq a_0 \leq a_{0ub}$$
$$a_{1lb} \leq a_1 \leq a_{1ub}$$
$$a_{2lb} \leq a_2 \leq a_{2ub}$$

The Monte Carlo scheme can be implemented using these bounds of the estimated coefficients; each random sampling from the bounds will give a new curve as shown in Figure 2.8.

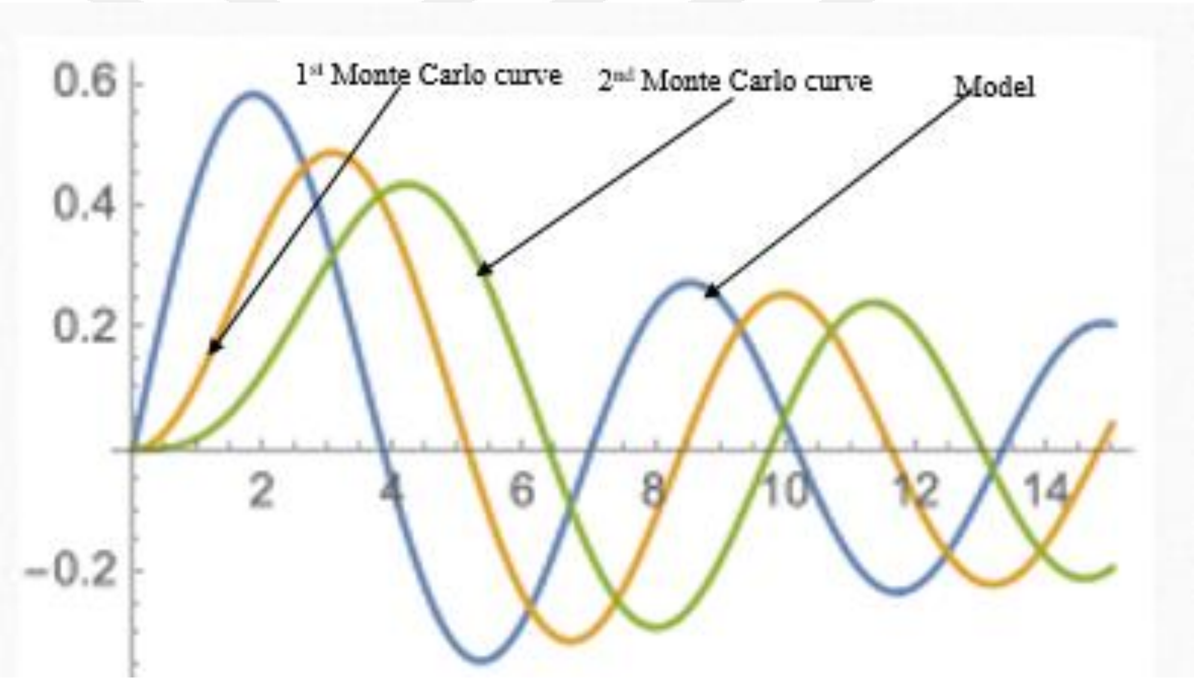


Figure 2.8. Monte Carlo simulation path for modeled data

The scheme work by first getting the bounds for the estimated coefficients, then for each coefficient, collecting a random sample from the bounds using a suitable sampling distribution, applying these randomly sampled values to the model equation, and estimating a new curve; that is Monte Carlo simulation path.

2.5. Artificial Neural Network

An artificial neural network (ANN) is a machine-learning model used for data processing that is inspired by well-known biological neural networks or a nervous system like the brain.

Panigrahi et al. [46] describe ANN as a mathematical model that mimics the human brain in processing the relationship between inputs and outputs. Although ANN is a machine learning algorithm used for building a forecasting model it is named ANN due to its resemblance with the typical neural networking as highlighted in [47]. The ANN algorithms are employed across a variety of tasks in different domains such as image processing, pattern recognition, prediction, etc. Palmer et al. [48] demonstrate how ANN performed best when there are enough datasets because of its ability to create the data input statistical model. Hence, the algorithm is attracting more interest from researchers.

2.5.1. ANN Method Design

The ANN is designed in such a way that it gets input and target data in a sequence then tries to predict the next possible value of the sequence according to a specified window slide. The data can be divided into any number of window slides. For example, if $Y_i = 1, 2, 3, 4, 5, 6, 7, 8$ is a data sequence, we can use a window of 4 as shown in Table 2.1.

Table 2.1. ANN input target example

Input	Target
[1, 2, 3, 4]	5
[2, 3, 4, 5]	6
[3, 4, 5, 6]	7
[4, 5, 6, 7]	8

It can be noted from Table 1 that the length of the data shrinks by the window size and this is the case for any given scenario. An artificial neural network can be implemented using a single or multiple layer perceptron. The major difference between the two is that the single-layer perceptron does not have a hidden layer while the multiple-layer perceptron has a hidden layer as shown in Figure 2.9 and Figure 2.10 respectively.

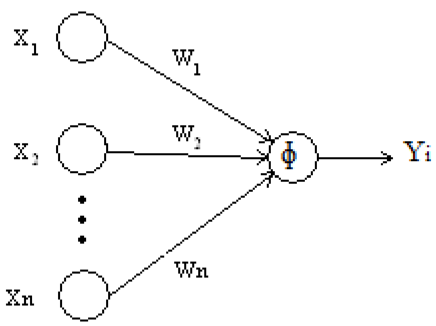


Figure 2.9. ANN single layer perceptron

From Figure 2.9 X_1, X_2, and X_n are the inputs, W_1, W_2, and W_n are the weights, while Y_i is the target output. Therefore, the target output can be obtained using equation 2.15.

$$Y_i = W_1X_1 + W_2X_2 + \cdots + W_nX_n + bias$$

$$Y_i = \sum_{i=1}^{n} W_iX_i + bias \tag{2.15}$$

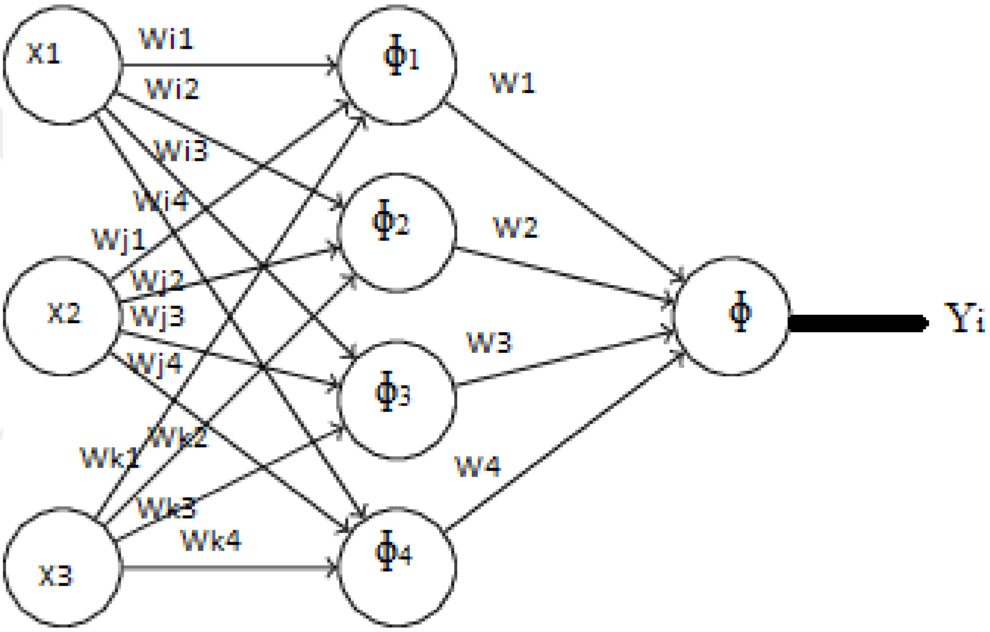


Figure 2.10. ANN multilayer perceptron with one hidden layer

Similarly, in Figure 2.10 X_1, X_2, and X_3 are the inputs, W_i, W_j, and W_k are the parameters of the input layer, W_1, W_2, W_3, and W_4 are the parameters of the hidden layer while Y_i is the target output and ϕ_i is the activation function and can be obtained using equation 2.16.

$$Y_i = \sum_{i=1}^{p} \phi_i W_{j,i} + Q \tag{2.16}$$

Where p is the number of hidden layers, j is the number of data, and Q is the bias for the rest of the layers.

$$\phi_i = \tan(\sum_{i=1}^{n} W_iX_i + q_i)$$

$$Q = \sum q_i$$

Equation 2.16 can be written as equation 2.17, a multi-linear perceptron with one hidden layer.

$$Y_i = \sum_{i=1}^{p} W_{j,i} \tan\left(\sum_{i=1}^{n} W_i X_i + q_i\right) + \sum q_i \tag{2.17}$$

3. RELATED WORKS

This chapter presents a literature review of relevant materials related to this study. The first section presents a systematic literature review on ARIMA models. Similarly, sections 3.2, 3.3, and 3.4 respectively discuss methods and results of previously published related studies on Fourier series, Monte Carlo scheme, and machine learning. Materials collection and selection criteria for inclusion or exclusion are discussed in the methodology chapter.

3.1. ARIMA Models

The study of Kriechbaumer et al. [49] with the title "an improved wavelet-ARIMA approach for forecasting metal prices" described wavelet analysis as a novel method that can capture the cyclic behavior of a time series. The study further explained how the wavelet analysis is an improvement on the traditional Fourier analysis in the sense that it decomposes a time series into frequency and time domain. Thus, different frequency variations in the series are identified and localized. The authors study the usefulness of the wavelet-ARIMA method for forecasting monthly prices of aluminum, copper, lead, and zinc traded on the London metal exchange market. The forecast results indicate how the forecast accuracy of the wavelet-ARIMA approach is highly correlated with the wavelet transform type, wavelet function, and decomposition levels. The study's major conclusion is that ARIMA models are not suitable for forecasting monthly metal prices, as the model performance does not show a significant difference from that of a naïve model.

In the work of Rusli et al. [50], the authors use an algorithm that is a hybrid of the ARIMA model and particle swarm optimization (PSO) to predict coal consumption from a combination of three-time series datasets namely capacity factor dataset, specific fuel consumption dataset, Persero coal online application dataset. The data belongs to Persero an Indonesian state-owned power generating company and the datasets are from 2013 to 2017. The study explained why the ARIMA model might be insufficient for the prediction problem. Hence, the author proposed the hybrid of the ARIMA and PSO methods. The ARIMA and hybrid ARIMA PSO evaluation performance were done using the MAPE calculations. The study results show how ARIMA and PSO can improve accuracy graphically compared to lone ARIMA output. In conclusion, this study indicates how the hybrid ARIMA PSO can increase the accuracy of the ARIMA model. It further shows how combining datasets is positively correlated with a smaller error rate. Mehrmolaei and Keyvanpour [51] review previous studies on time series forecasting in different application areas and proposed a novel approach to improving the ARIMA model using MER for time series forecasting. The study classifies time series forecasting techniques into short-lived estimators and long-range estimators based on forecast duration. New York City births for 1946 to 1946 that contains 168 data points is the time series data used for the study. The work proposed an approach of applying MER for time series on the basic ARIMA model and the results show that the improved ARIMA model produced a better result when compared to the basic ARIMA model.

Babu and Reddy in [52] present analysis and prediction of average global temperature from 1880 to 2010 using three variants of the ARIMA model. The three ARIMA models used in the study are the basic ARIMA, trended ARIMA, and wavelet ARIMA. The average temperature for the period of 2001 to 2010 is predicted using these ARIMA models and the individual performance of the models is compared to each other using the MAPE. The study results conclude that the wavelet ARIMA has the best performance among the three models. The trended ARIMA in the second position is slightly above the basic ARIMA. However, these models are all linear maybe using other nonlinear methods such as the artificial neural network can devise a better prediction method and results.

Similarly Amra and Maghari [53] in their study titled " Forecasting Groundwater Production and Rain Amounts Using ARIMA-Hybrid ARIMA: Case Study of Deir El-Balah City in Gaza" presents another analysis and prediction of groundwater production and rainfall amount using different variants of the ARIMA model. The ARIMA models were categorized into the ARIMA model and the Hybrid-ARIMA model. The latter is divided into ARIMA + NN, ARIMA + ets, and ARIMA + tbats. The datasets used in the research are the historical data for groundwater wells for the period of 2008 to 2017 collected from the Coastal Municipalities Water Utility in Dear El-Balah and rainfall amount data for the period of 1985 to 2017 collected from the Palestinian ministry of agriculture. A series forecast was done using the four different ARIMA models. The results show that the hybrid ARIMA + NN has the highest accuracy with MAPE 21.0% for rain amount forecast, while hybrid ARIMA + tbats has the best performance for groundwater forecast with MAPE 8.9%. The authors conclude that in the next five years rainfall amounts are expected to decrease by 8.4% likewise the well's production of groundwater is expected to decrease by 0.03% in comparison to the 2013 to 2017 period.

In [54] high volatility and multi-step prediction in financial time series data are identified as a major research problem in time series data mining by the authors. The ability of financial time series data mining to provide investors, banks, and insurance companies with useful information that assists them to channel their funds in the right direction for better returns is identified as the major motivation toward financial time series data forecasting in the study. However, the study identified maintaining strong prediction accuracy and preserving data trends across the forecast horizon as the two major issues of forecasting financial time series data. According to the paper, traditional models such as ARIMA and generalized autoregressive conditional heteroscedastic (GARCH) can preserve trends but at the cost of prediction accuracy while non-linear models like artificial neural networks provide prediction accuracy at the cost of data trend. It is because of these trends and prediction accuracy issues the authors proposed a linear hybrid model that can maintain both the prediction accuracy and data trend. The proposed model incorporated the MA filter-based preprocessing, partitioning, and interpolation techniques. Furthermore, the accuracy of the proposed model is justified using quantitative reasoning analysis. The results demonstrate that the proposed model produces better performance results in multi-step prediction than the traditional ARIMA and GARCH models in terms of both prediction accuracy and preserving data trends. The research model results were evaluated using MAPE, MaxAPE, and RMSE.

Koo et al. [55] in their paper "short-term electric load forecasting using data mining technique" present time series data forecasting using five models of seasonal ARIMA. It is highlighted in the study that there is an important difference between electric load during the weekdays and weekends. Even so, the weekday's electric loads may differ and behave differently from day to day. Furthermore, the paper explained how electric power consumption load can be treated as a time-series signal. The authors proposed a way to improve the traditional seasonal ARIMA by modeling five models of the seasonal ARIMA according to day type. The dataset used in the study was collected from the utility of Malaysia and in particular, is the electric load dataset of Peninsular Malaysia. The forecast results were 48 points ahead for each day type. From the simulation, the predicted results obtained were compared with the actual data obtained. In conclusion, the half-hourly load data for six weeks had been plotted according to day type to forecast the load demand for a day ahead.

In another work by Zeng et al. [56], the authors develop a real-time monitoring and warning algorithm for anomalies based on a modified ARIMA (M-ARIMA). The paper describes M-ARIMA as a model based on time series ARIMA and has the capability to achieve real-time monitoring and warning. Dataset used in the work was collected from fiber mechanical and thermal multi-parameter instruments. Three technical challenges of the study were finding corresponding prediction models in different time-period, selecting dynamic thresholds varying with the surrounding environment, and identifying the prediction phase associated with a time period as explained by the authors. The proposed M-ARIMA model was able to achieve data analysis of the collected data and the effective monitoring of states. Besides, the model can automatically trigger an alarm in a situation where an abnormality state is detected. The results show that the M-ARIMA produces high detection rates and can satisfy high real-time requirements. In the paper of Garcia et al. [57], the authors developed a theory of transgenic time series and used it to forecast several rare earth elements using the ARIMA model. The theory allows the study to eliminate challenges of metal price cycles and anomalous phenomena, which in turn improve the forecast accuracy. Since time series can be cut and pasted without further consideration, the authors create a genetically modified time series that represented the series in such a way that allows the genome to be sequenced with the help of restriction enzymes. The study results indicate that a transgenic time series for short-term forecast where consistent time series genome can be represented show high forecast accuracy with improved AIC, RMSE, MAE, and MAPE. In conclusion, the proposed algorithm chooses for each data set the predictive model that best fits the time series and, later, it compares the performance of the model proposed for the different data sets, presenting the optimal combination as a result.

The work of Khandelwal et al. [58] demonstrates how discrete wavelet transform (DWT) can be used to improve the accuracy of time series forecasting. The paper presents a unique way of forecasting time series data by using DWT to segregate the series into linear and nonlinear components. The linear and nonlinear components that are decomposed using the DWT are named detailed and approximate respectively. ARIMA model is used to recognize and predict the detailed component while ANN is used to recognize and predict the approximate component. Since in real-world time series, it is difficult to establish whether the series is linear or nonlinear, and it is known from literature ARIMA model assumes linear data generation function while ANN is for nonlinear, the study proposed a hybrid forecasting method that applies ARIMA and ANN to model the linear and nonlinear components of the series. Modeling in the proposed method is achieved by decomposing the series into low and high-frequency signals using the DWT. The proposed hybrid method in the study was tested on four real-world time series, whereas, the foresting results are compared with those of ARIMA, ANN, and another model from the literature. Comparison results show that the hybrid method performed better than the other models.

A comparative study of Matyjaszek et al. [59] evaluates the forecasting performance of time series models namely robust model, ARIMA model, generalized regression neural networks (GRNN), and multilayer feedforward networks. The evaluation of the models was done based on two categories: a full-time series and transgenic time series. The latter is based on the theory in [57]. The researchers consider the question of which forecast performance will provide the best projection of errors and choose MAPE since it is the only performance evaluation criteria that consider precentral deviation among RMSE, MAE, and MAPE. In the case where full-time series has been used the results of the study show that GRNN produces a more accurate forecast. However, in the case where the transgenic time series is used for the forecast, the ARIMA model gives the best forecast. In conclusion, the analyses of the results of the study show that the ARIMA model is the model that achieves the best performance in terms of MAPE and MAE when the transgenic time series is used. Because of this, the study concludes that the ARIMA model is the best among the models that were studied in the paper.

ARIMA model and deep belief network (DBN) are used by Qin et al. [60] to analyze related factors for red tide disasters using a proposed hybrid method that takes advantage of the strong forecasting ability of ARIMA and the powerful expression ability of DBN on nonlinear relationships. The dataset that was used in the study is that of the Zhoushan coastal area and

Wenhou coastal area all in China and from the period of 2008 to 2014. The hybrid method is designed to define temporal correlation and spatial heterogeneity by the corresponding ARIMA model for each environmental factor in different coastal areas. DBN serves to capture the complex nonlinear relationship between environmental factors and red tide biomass and then pre-realizes the red tide warning. The proposed ARIMA-DBN model is applied to the red tide forecast. Experimental results show that the proposed method makes a good estimate for red tide. In conclusion, the experimental results show that the proposed method works well in red tide prediction and has good applicability and robustness.

3.2. Fourier series

Brentan et al. [61] present a real-time water demand forecast in Franca, Brazil using the Fourier series and Chebyshev polynomials by real-time updating some adjustable coefficients. The study briefly explained the temporal behavior of water demand using an elliptical approach that shows the periodic nature of water demand, which is influenced, by social behavior. The authors review the polar coordinate representation of the weekly water demand for a week, helping to understand the periodic behavior of the water demand and propose new approaches to treat the water-demand prediction problem. The statistical analysis of the parameters used for the model evaluation shows that both models can predict the water demand in real-time with high accuracy. Even so, with thirty-six terms, the Fourier series produce slightly better results in the study. The study concludes that because there are facilities and agility to determine Fourier series coefficients, these give the models a chance to forecast in real-time and without any high computational efforts.

The study of Iwok in [62] with the title “seasonal modeling of Fourier series with a linear trend” was motivated by the need to model a periodic time series function with a linear trend. Because of this, the study proposed a Fourier series representation with a detrended linear function. In this representation, the time series is shown as the combination of the linear trend component and a linear combination of the seasonal number of the orthogonal trigonometric function. The study was carried out using the average monthly rainfall dataset of Calabar, Nigeria from 2005 to 2015, while the data analysis was done using Minitab and Gretl software. The proposed method was applied to this data and it produces a good fit. Moreover, diagnostic checks show that the proposed method performed better compared to the pure Fourier approach. To conclude, in this paper, it has been noticed that if the series is periodic then Fourier can be applied to the series, as for trended series, a special technique is needed for the modeling process.

In another comparative study, Iwok and Udoh [25] compared the performance of the ARIMA-Fourier model and that of a wavelet model by applying both models to a dataset of Nigeria's consumer price index series. The hybrid ARIMA-Fourier model was obtained through the combination of the linear and sinusoidal components of the time series. Both models were found to adequately fit the data. Evaluation of the models using MSE, MAE, and MAPE was done and the comparative analysis of the results shows that the wavelet model performs better than the hybrid ARIMA-Fourier model. The result of this study show how the wavelet method is flexible in handling non-stationary series and its ability to combine information from both the time domain and frequency domain.

A study by Fidino and Magle [63] illustrates another technique that can use the Fourier series to estimate periodic signals in dynamic occupancy models and parameterize the models using a large-scale long-term camera capture study dataset of medium to large mammals in Chicago, Illinois, USA. The periodic model proposed by the authors was able to capture 75% of the temporal variability concerning the coloring type ratios. Similarly, the same was done achieved from the dynamic occupancy models with temporally varying parameters. To sum it up, the proposed method in this study can generally divide variability between periodic and non-periodic sources and estimate the rate of temporal variability attributable to a periodic source in a model-based framework.

The problem of missing data in remote sensing analysis is addressed using a Fourier series approach in the work of Brooks et al. [64]. The authors explained how remotely sensed time-series data in the multi-temporal analysis is facing issues such as cloud cover and image-related problems that result in missing data for some aperiodic times of the year. The study applied a Fourier regression technique on a dataset of Landsat pixels with 30-m resolution. The result of the Fourier regression technique is then compared with that of STAR-FM, which shows that the Fourier has an R^2 of 90% over ¾ of all pixels. Moreover, compared to the STAR-FM the $R^2_{predicted}$ values of the Fourier regression technique are higher on about two-thirds of the pixels. The study's major conclusion is that the Fourier regression algorithm can be used to interpolate missing data of multi-temporal analysis at the Landsat scale.

Taiwo and Olatyo [65] in their study on an improved procedure for Fourier regression analysis described Fourier regression analysis as a better procedure that is extensively used for modeling periodic time series and explained how it can represent time series through elementary sine and cosine functions. The research addresses the issues of several components, estimation method, and stability of error term testing in a Fourier model. Using spectral analysis to determine the number of components, the Fourier transform-based estimation method, and upper and lower bounds to determine error stability are the three proposed solutions to the issues identified in the study respectively. The proposed Fourier regression analysis solutions are tested on a Nigerian road accidental death time-series dataset, which was able to show the period of significant frequencies and was used to fit the periodic trend. This shows that the proposed method performed adequately in studying periods and their frequencies and fitting the periodic trend for the dataset. In conclusion, the proposed Fourier regression analysis method is suitable for forecasting the time series dataset that was used in the study.

In the work of Prahutama et al. [66] inflation in the Indonesian food, sector is modeled using Fourier series regression for time series data. After the work explained parametric and non-parametric regressions techniques, the authors elect to go with the latter citing it does not need model assumption as the reason for its selection for the study. The study also highlights how the Fourier series approach is suitable for non-parametric regression because of its ability to overcome trigonometric distribution data. The study also shows that in the nonparametric regression model, the Fourier series requires just the K value to estimate the parameters. The result of the work found a multiple regression model with an R^2 value of more than 90% for the inflation modeling of the transport and communication sector while the Fourier model was able to give K value close to 99%. It can be observed from the result of the study that the Fourier model has 122 parameters while the multiple regression linear model requires only three parameters.

According to the study of Tsai et al. [67], the electricity consumption of buildings can be categorized into two or three categories, that is, according to their periodic and irregular activities, or according to long-term trends, periodic activity, and irregular activity. However, these categories vary in viewpoints concerning their time frame factor, which ranges from hourly to seasonal, thus, making it difficult for construction models to estimate electricity consumption accurately. Considering these viewpoints, to accurately analyze the periodic activities and long-term consumption trends that will be used by the building administrators and engineers the study proposed a polynomial Fourier series (PFS) model to address the identified issues. To determine predictive accuracy and analyzed the model result, the building types are categorized into three according to their purpose with electricity consumption ranging from 150mwh to o.6mwh. The results of the study show that the PFS model performed well with high accuracy in providing estimates of electricity consumption, which is very useful for evaluating energy policy and setting forward electricity budgets based on limited data and mathematical iteration.

An evolutionary technique for time series data modeling using a Fourier genetic series (FGS) is presented by De Lima et al. [12]. The study began by explaining the current trend in financial marketing and online trading highlighting why there is a need for better models to accurately forecast market behavior as best as possible. The FGS was used to map periodic functions in the range of any curve to find an optimal solution to large-scale combinatorial optimization problems identified in the study. Air passengers' time-series data obtained from Kaggle an open-source database was used to test the performance of the FGS model. The model results were benchmarked with that of other models, which proved to be an alternative technique of time series forecasting by producing low forecast error compared to the models that it was benchmarked with. In conclusion, a finite sum of highly inspiring mathematical expressions in the Fourier series shapes the time series with different features, summarizing the strengths of existing techniques, seasonality, trend deviation, and correlation characteristics, abstracting the characteristics of seasonality, trend deviation, and correlation.

3.3. Monte Carlo

Alamsyah and Nurriz [68] in their study about Monte Carlo simulation and clustering for customer segmentation in business organizations discussed how conducting business-related analysis has become a liability for organizations to retain their competitiveness. The study also demonstrates how to use the Monte Carlo simulation solution to address a situation whereby an organization could need a data attribute that is not available. As a case study of the work, clustering using the k-means algorithm and Monte Carlo simulation is applied to an Indonesian telecommunication company named Telkom Indonesia Makassar to generate the value of customer income. A segmentation model is developed by comparing the generated customer income value with customer spending. The work results show how to create customer revenue data and create a dataset to optimize customer potential using the data. In the paper of Ran et al. [69], the authors applied ID3 Algorithm to classify data according to their impact on equipment failure severity and then find the factors that most affect the equipment. Once they find the most important factors, they applied Monte Carlo simulation to predict the likelihood of equipment failure occurring in their operational life and accordingly pay more attention to easily trace the factors causing the failure.

To describe seasonal radon variation on the surface as well as its depth profile in the North Anatolian Fault Zone, Turkey, the work of Muhammad et al. [70] proposed a hybrid regression model that can capture and forecast feature values of radon seasonal anomaly and its depth profile. This proposed model, which includes parameter estimation of other radon-related parameters, was designed to cover a range of 300 possible variations by applying Monte Carlo simulation to the model. The result of the model shows a good performance concerning solving the radon transport equation using historical data to estimate parameters. The work of Moudrik et al. in [71] presents the use of the Monte Carlo search tree method to address the problem of algorithm program discovery instead of the well-known genetic programming technique that is used for addressing automated algorithm discovery. Representation of stack-based and typed approach programs was analyzed in the paper. The authors introduced the typed approach, which allows finely program size control. The experimental result of the approach generates a program of given size uniformly which from a Monte Carlo simulation viewpoint is a good property that shows a better control of exploration.

The study of Gonzale et al. [72] explains how Monte Carlo simulation is an important tool used for estimation of statistical properties of estimators and how regression techniques are employed in analyzing the large amount of data generated by the Monte Carlo simulation which put the researchers in an awkward position about what to report and what to test. Considering these limitations, the authors proposed an approach of classification and regression tree (CART) for analyzing Monte Carlo simulation data that shows promising performance in overcoming the limitation of representing many simulation factors. While the finding of the study indicates that the proposed approach shows similar conclusions with the existing descriptive and inferential approach, it gives more information about complex interactions among the simulation factors involved.

Similarly Qi and Hu [73] in their study with the title “Monte Carlo tree search-based intersection signal optimization model with channelized section spillover” highlights how signal optimization using the Monte Carlo method is getting significant attention among the research community and proposed a model based on Monte Carlo tree search to address the problem of intersection optimization. The proposed model also includes explicit modeling of channelized section spillover, which is dynamic. The results yielded by the proposed model always show better performance when compared and tested against Synchro. The Monte Carlo tree search-based algorithm proposed in this study could also apply to a network with multiple intersections. The work of Al-Dalkey et al. [74] considers how biologists felt the need to know the association between genes and diseases then proposed a classification system they named Monte Carlo for the genetic network. The proposed system can generate a genetic network and establish functionally related genes, which it use to forecast the relationship between the disease and the gene. The study evaluated Monte Carlo experimentally comparing it with nine approaches for a genetic network and the results were found to improve significantly.

3.4. Machine Learning

Zhongda et al. [75] proposed a forecasting model based on the combination of the empirical model and extreme machine learning to optimize the time-series prediction accuracy. The study uses Lorenz and network traffic time-series to test the prediction power and accuracy of the proposed model. The simulation result of the study produces a good performance with a minimized forecast error. In the work of Wang et al. [76], a combination of SVM and exponential smoothing algorithms was used to propose a machine learning approach to performance monitory and failure prediction in optical networks. The result of their work shows high-performance accuracy with an average of 95% in predicting equipment failure rate.

The work of Ağbulut et al. [77] applied different machine learning algorithms to the Turkish meteorological service data from 2008 to 2018 to forecast the monthly mean of daily solar radiation for the year 2018. Model diagnostics of the study are presented in terms of R^2, MPE, MPE, t-stat, and RMSE. For all the machine learning models used in the study, the R^2 estimate was found to be varying between 95% and 99% with respective MAPE and RMSE of 10% and 2MJ/m^2-day.

According to the model's diagnostics results, ANN presents the best performance in terms of R^2, MPE, MPE, t-stat, and RMSE. In the study of Khosravi et al. [78], time series prediction for wind speed at Osorio wind farm in Brazil was carried out using machine learning algorithms. Wind speed data at respective intervals of 5, 10, 15, and 30 minutes were used as input to seven different machine learning algorithms to predict the series. The result shows that the group method of data handling model has the best performance in terms of prediction accuracy for all the time intervals among the chosen machine learning algorithms.

The Qian [81] paper compares traditional regression methods used for time series forecasting and the modern machine learning approach. The paper used Dow 30, S&P 500, and NASDAQ real data sets to demonstrate how the precision of modern machine learning approaches surpasses the traditional regression methods in the time series forecasting domain. Pavlyshenko's [79] study to the role of machine learning in forecasting sales time series data while considering the generalization effect of machine learning. The study highlights how the generalization effect can predict sales series with small historic data. and concluded that forecasting sales data is a regression problem rather than a time-series problem.

The work of Parmezan et al. [80] presents statistical and machine learning methods evaluation for time-series data forecasting. The different state-of-the-art models and the individual models' conditions are identified. Respectively from 95-time series datasets 11 predictors, 7 parametric, and 4 nonparametric are identified by the means of multistep ahead projection strategies and 4 performance evaluation measures. The obtained results show that only SARIMA was found to outperform SVM and KNN machine learning algorithms from among the selected statistical models.

4. METHODOLOGY

This chapter presents the methodology and the algorithms of the proposed Polynomial Fourier series and ANN Polynomial Fourier series models of the study. The chapter discusses and presents the inclusion and exclusion selection criteria for articles and other scholarly written materials used in this study. The data transformation and pre-processing algorithm for the selected models are presented. Besides, the study's proposed Monte Carlo simulation scheme is presented in this chapter. Also, the chapter discusses the model selection criteria of our study and describes the time series data used for testing the performance of the models.

4.1. Inclusion and Exclusion Criteria

When designing a study, it is very important that researchers define appropriate inclusion and exclusion criteria and evaluate how these decisions will affect the external validity of the study results Meline [81] and Booth [82].

To collect the necessary study materials used in this study we searched online databases such as IEEE explore digital library, Wiley library, Elsevier, etc., and other scholarly written documents. We used a combination of search categories and terms to retrieve relevant materials for inclusion and exclusion screening. These search terms are combined using the Boolean "and" or "or" operators. The retrieved materials related to our study domain and published in the English language are screened for inclusion. Any material published in the English language or an ongoing study is excluded even if it falls within our study domain. Figure 4.1 shows all the steps involved in screening materials for inclusion or exclusion.

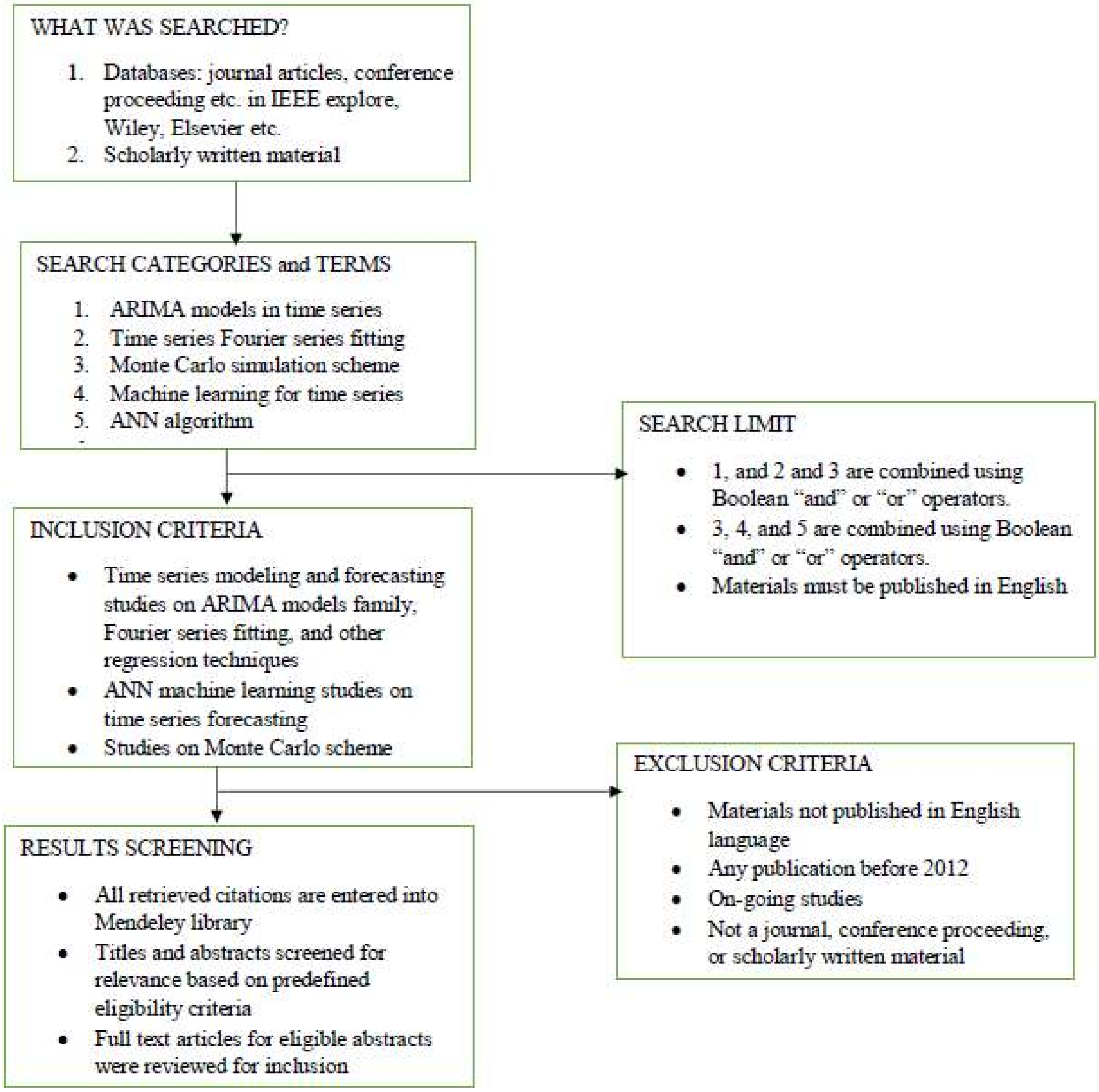


Figure 4.1. Inclusion and exclusion criteria

4.2. Data Pre-processing Algorithm

In this study, before feeding data to our model, we prepared it by preprocessing it according to the steps involved in Algorithm 1 provided in Table 4.1. It can be noted from the algorithm how the ARIMA model requires stationary data, Fourier series and Polynomial-Fourier series models require estimated data frequency while the ANN-Polynomial-Fourier series model requires both data frequency and window slides.

Table 4.1. Data pre-processing algorithm

Algorithm 1: Data pre-processing

```
Input:  data,
        model
Output: pre-processed data ready for modeling
train = [70%]
test = [30%]
if model = ARIMA
   perform stationarity test
     if series = stationary then
        split series into train and test
        submit for ARIMA modeling
     end if
     else if series != stationary then
        make the series stationary
        split series into train and test
        submit for ARIMA modeling
        end else if
end if
else if model = Fourier then
   estimate data frequency
   split series into train and test
   submit for Fourier series fitting
end else if
else if model = Polynomial Fourier series then
   estimate data frequency
   split series into train and test
   submit for Fourier series fitting
end else if
else if  model = ANN Polynomial Fourier series then
   estimate data frequency
   get slide window size
   split series into train and test
   submit for Fourier series fitting
end else if
else if model = ANN Fourier series then
    estimate data frequency
    get slide window size
   split series into train and test
   submit for Fourier series fitting
```

36: **end else if**

37: **return** processed data

4.3. Proposed Models

ARIMA family of models are regression models with the assumption that dependent variables are stationary and the independent variables are error lags or lags of the dependent variables. Hence, these families of models require the time series data to be transformed into a stationary series with a constant mean. With these characteristics, it can be established that the ARIMA family of models can handle series with trends but have a limitation when it comes to seasonal series [83] and [84]. Fourier series fitting on the other hand have limitations in handling trends but can very well handle a periodic or seasonal series[85].

Considering these strengths of the ARIMA and Fourier series, it can be said that they are respectively reasonable models to be applied to a trended and seasonal series. Our study considers the need to model time series data to account for both seasonality and trends in a series. In light of this, we proposed Polynomial-Fourier series, ANN-Polynomial-Fourier series, and ANN-Fourier series models to model series that can account for both trends (linear and nonlinear) and seasonality of the series as discussed in sections 4.3.1 and 4.3.2.

4.3.1. Polynomial-Fourier series model

In section 2.1.1 we explained how time-series data may have a trend or seasonal components or both trend and seasonal components in some cases. This trend that time series exhibit can be linear or nonlinear, in some cases, a series may have both linear and nonlinear trends. Our proposed Polynomial-Fourier series model used polynomial fitting in equation 4.1 to predict the linear and nonlinear trends of a series. While the seasonal component is predicted using the Fourier series in equation 2.5.

$$f(x) = \sum_{n=0}^{\infty} A_n X^n \tag{4.1}$$

From equation 4.1 X is the series time counts/lags, n is the number of the polynomial terms, and A is a coefficient. In this proposed model, the first polynomial term $a_1 X$ is used for capturing the series linear trend, and the second polynomial term $a_2 X^2$ is a gauge function that regulates the trend and seasonality, which we used to capture the series's nonlinear trend. Combining our respective polynomial and Fourier series equations in equations 4.1 and 2.5 will obtain equation 4.2. The Polynomial-Fourier series model equation is derived from equation 4.2 by considering the first and second terms of the polynomial and first term of the Fourier series as shown in equation 4.3.

$$f(x) = \sum_{n=0}^{\infty} A_n X^n + a_0 + \sum_{n=1}^{\infty} a_n \cos(n\omega X) + \sum_{n=1}^{\infty} b_n \sin(n\omega X) \quad (4.2)$$

Since we are using the first and second terms (a_1X anda_2X^2) of the polynomial and the first term of the Fourier series $a_1 cos(\omega X) + b_1 sin(\omega X)$ then we have:

$$f(x) = \sum_{n=0}^{2} A_n X^n + a_0 + \sum_{n=0}^{1} a_1 cos(n\omega X) + b_1 sin(n\omega X)$$

$$f(x) = a_0 + A_1 X + A_2 X^2 + a_1 cos(\omega X) + b_1 sin(\omega X) \quad (4.3)$$

Equation 4.3 is our proposed Polynomial-Fourier series model equation and the coefficients a_0, A_1, a_1, and b_1 of the model equation are estimated using the least square method. The Polynomial-Fourier series model algorithm given in Table 4.2 and algorithm 2 show how the estimated coefficients and covariance used model equation in equation 4.3 to generate the equation that will fit the given time series data.

Table 4.2. Polynomial-Fourier series model algorithm

Algorithm 2: Polynomial-Fourier series model

Input: Algorithm 1

Output: model residual
model training prediction
model test prediction

1: model_data_frame = [ ]
2: fit data using equation 4.3
3: estimate coefficients $\boldsymbol{a_0}$, $\boldsymbol{A_1}$, $\boldsymbol{a_1}$, and $\boldsymbol{b_1}$
4: store data model equation according to the estimated coefficients in model_data_frame
5: **return** model residue,
model training prediction,
model test prediction

The residual of the Polynomial-Fourier series model forecast result is collected and used for the Monte Carlo simulation scheme given in algorithm 4 of Table 4.4. The Monte Carlo simulation is done on the model prediction result to further optimize and forecast the model result. The chart in Figure 4.2 shows the polynomial and Fourier fittings forecasting steps involved in the proposed Polynomial-Fourier series model as presented by Danbatta and Varol [86].

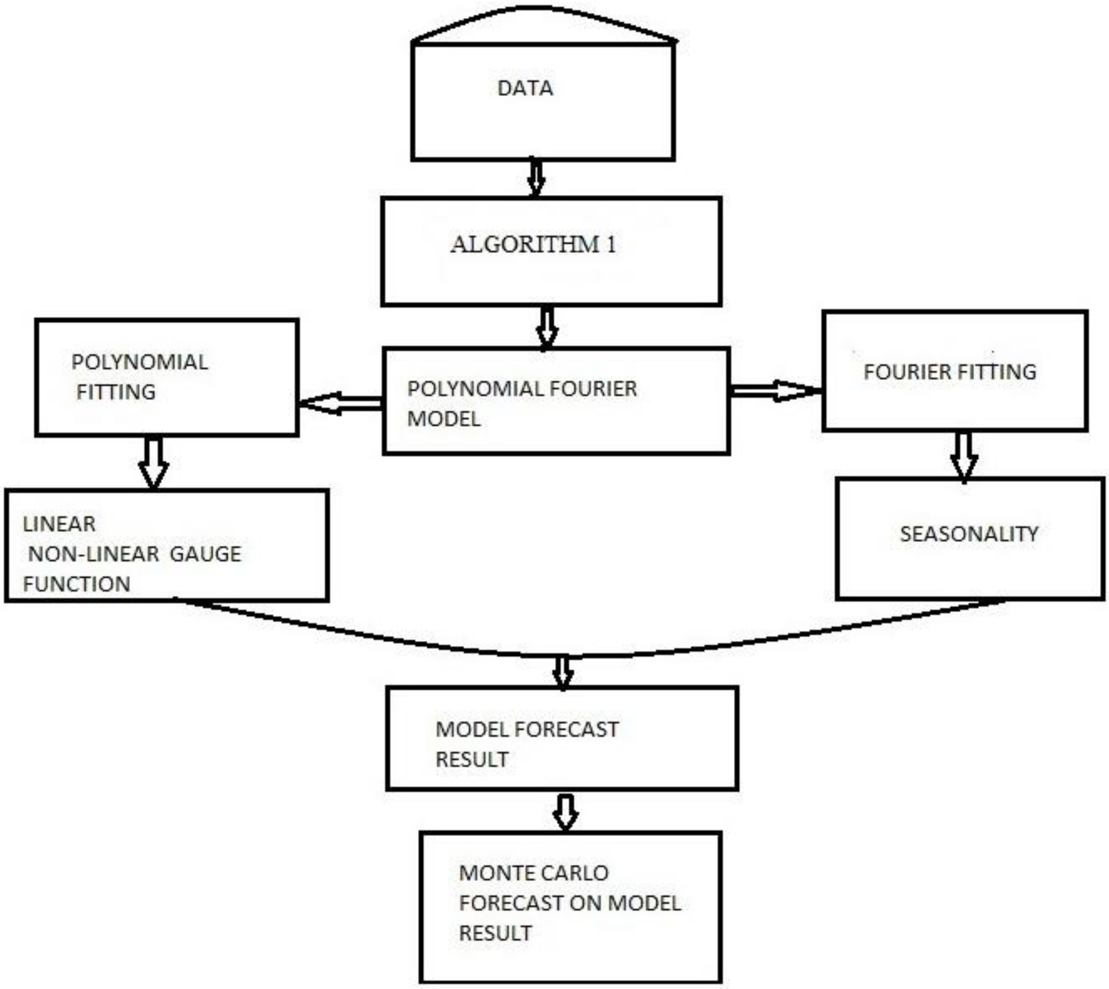


Figure 4.2. Polynomial-Fourier series model forecasting steps [86]

It can be seen that the Polynomial-Fourier series model forecasting steps given in Figure 4.2 involve model forecasting and Monte Carlo forecast of the model results steps. The model forecast result involves polynomial and Fourier fittings, which their summation yields the model equation in equation 4.3. The second step is running the Monte Carlo simulation scheme on the model result residue as discussed in section 4.4.

4.3.2. ANN-Polynomial-Fourier series model

We have seen how the Polynomial-Fourier series model is designed in section 4.3.1, in this section we will discuss and present the second proposed ANN-Polynomial-Fourier series model. The proposed ANN-Polynomial-Fourier series model begins by collecting the residue of the

Polynomial-Fourier series model. The collected residue is then modeled using an artificial neural network machine learning model. This ANN residual model is added to the Polynomial-Fourier series model given in equation 4.3, giving us the ANN-Polynomial-Fourier series equation as shown in equation 4.4.

$$f(x) = \sum_{n=0}^{\infty} A_n X^n + a_0 + \sum_{n=1}^{\infty} a_n \cos(n\omega X) + \sum_{n=1}^{\infty} b_n \sin(n\omega X) + ANN\ residual\ model + e$$

Where $\sum_{n=0}^{\infty} A_n X^n$ is the polynomial term, $\sum_{n=1}^{\infty} a_n \cos(n\omega X) + \sum_{n=1}^{\infty} b_n \sin(n\omega X)$ is the Fourier series term, and e is an error (0, σ) of the models which are assumed to be normally distributed. Respectively as discussed in section 4.3.1 we consider the first and second terms of the polynomial and first term of the Fourier series, hence, we obtain the ANN-Polynomial-Fourier series equation as:

$$f(x) = a_0 + A_1 X + A_2 X^2 + a_1 \cos(\omega X) + b_1 \sin(\omega X) + ANN\ residual\ model + e \qquad (4.4)$$

Algorithm 3 given in Table 4.3 shows how the ANN-Polynomial-Fourier series model collects residue from the Polynomial-Fourier series model and models the residue using ANN, which is added back to generate the ANN-Polynomial-Fourier series model in equation 4.4.

Table 4.3. ANN-Polynomial-Fourier series model algorithm

Algorithm 3: ANN-Polynomial-Fourier series model

Input: Polynomial-Fourier series model residual

Output: model residual
model training prediction
model test prediction

1: model the residue using ANN
2: add the modeled residue to equation 4.3
3: fit the data using equation 4.4
4: model data frame = [ ]
5: store data model equation according to the estimated coefficients in the model data frame
6: **return** model residue,
model training prediction,
model test prediction

After modeling the time series using the ANN-Polynomial-Fourier series model as shown in equation 4.4 and Table 4.3, the residual produced by the model is collected for Monte Carlo simulation. Similarly, the Monte Carlo simulation is done to forecast and optimize the model result so that it can account for more future value anomalies. The forecasting steps involved in the ANN-Polynomial-Fourier series model are given in Figure 4.3.

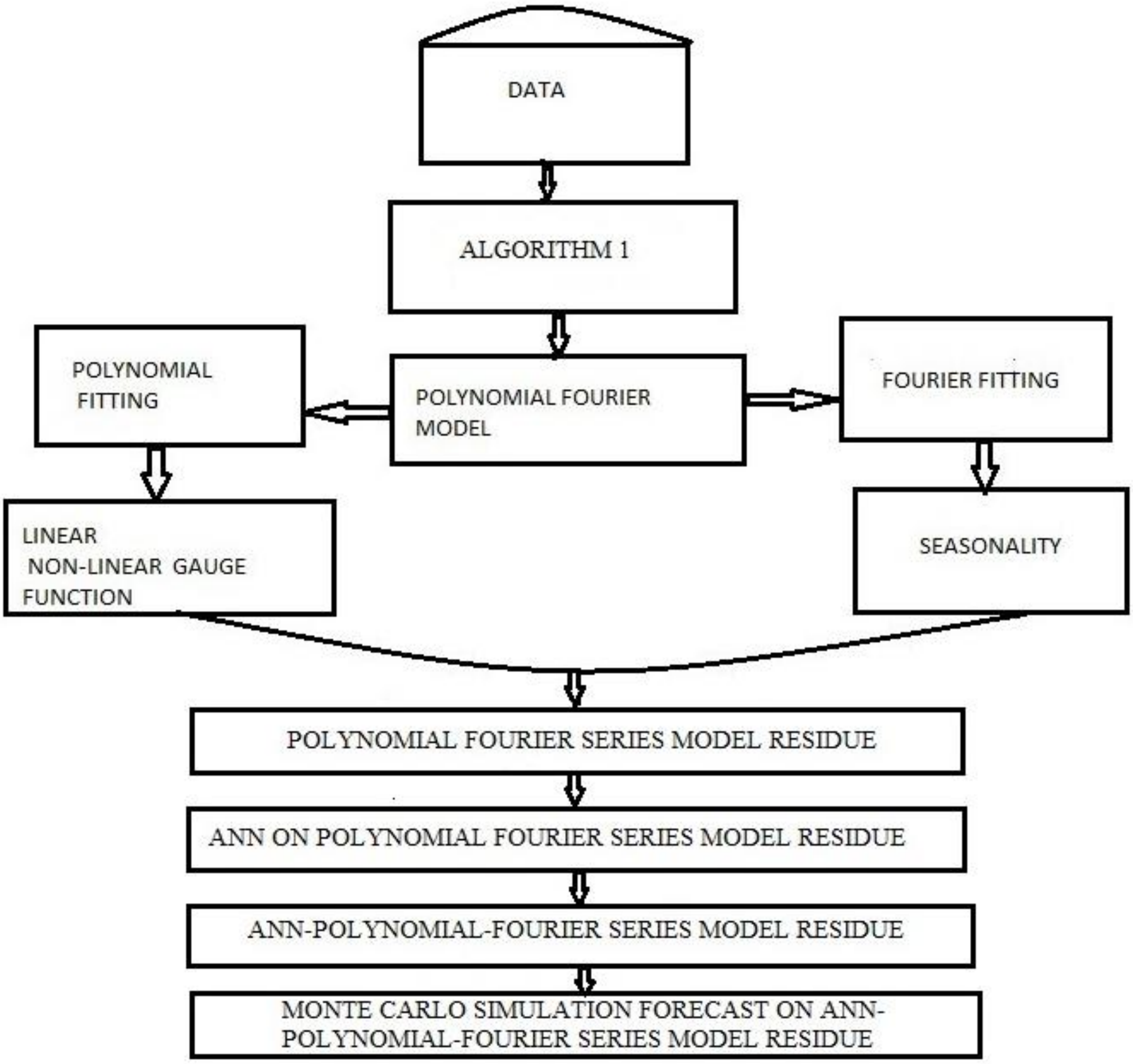


Figure 4.3. ANN-Polynomial-Fourier series model forecasting steps [87]

4.3.3. ANN-Fourier series model

In a situation where the said data observed high seasonality, the Fourier series is described as an approach that many researchers invoked to model seasonal time series data. Even so, the data might still have a trend component that is not included in the Fourier model. In this regard, and following the finding of Makridakis et al. [88] we proposed an ANN-Fourier series model. The proposed model is put forward as a combination of the Fourier series model and the artificial neural network (ANN) model. As opposed to most literature, our proposed ANN-Fourier is designed to capture and model the data using the Fourier series model fitting provided in equation 2.5. The ANN model is then used to capture and model the residue of the Fourier fitting in equation 2.5. Hence, we obtained the proposed ANN-Fourier series model in equation 4.5 as given by Danbatta and Varol [89].

$$f(x) = a_0 + \sum_{n=1}^{\infty} a_n \cos(n\omega x) + \sum_{n=1}^{\infty} b_n \sin(n\omega x) + ANN\ modeled\ residue + e \quad (4.5)$$

Where e is the proposed ANN-Fourier series model error while Table 4.4 shows the proposed ANN-Fourier series model algorithm.

Table 4.4. ANN-Fourier series model algorithm

Algorithm 4: ANN-Fourier series model
Input: Fourier series model residual
Output: model residual model training prediction model test prediction
1: model the residue using ANN
2: add the modeled residue to equation 2.5
3: fit the data using equation 4.5
4: model data frame = [ ]
5: store data model equation according to the estimated coefficients in the model data frame
6: **return** model residue, model training prediction, model test prediction

After fitting the time-series data using the algorithm presented in Table 4.4 and the equation in equation 4.5, we proceed to the Monte Carlo simulation scheme phase. The residue produce by the proposed ANN-Fourier series model is used as input to generate the 100 Monte Carlo simulation scheme paths. The mean of the simulation paths is used for forecasting and optimizing the model's result which enable to capturing more anomalies and uncertainties. The steps involve in the proposed ANN-Fourier series models given in the paper of Danbatta and Varol [89] are shown in Figure 4.4.

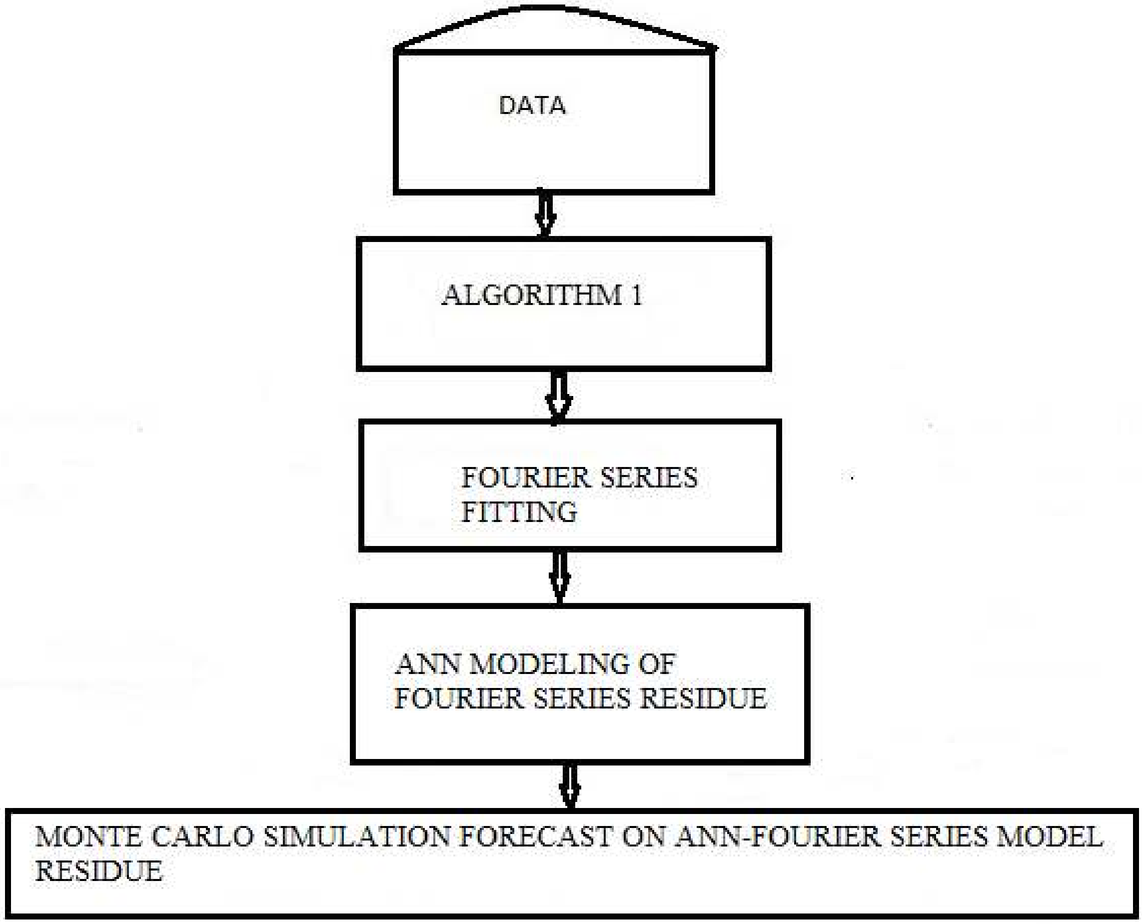


Figure 4.4. ANN-Fourier series model forecasting steps [89]

4.4. Monte Carlo Forecast Algorithm

We have mentioned in section 4.3 that the residual produced by the models is collected for running the Monte Carlo simulation scheme. The Monte Carlo simulation paths are used to forecast and optimize the series in case of future anomalies. In this study, instead of applying the Monte Carlo method on the estimated regression parameters of the model equations, we simulate the time series using the model residual and the Monte Carlo simulation is run within 95% ($\pm 2\sigma$) bounds from the model curves.

The Monte Carlo simulation begins by assembling the model prediction and residual into two separate data frames. Then the mean and standard deviation of the residue is calculated. The data, model prediction, and index are then submitted to a function that computes and returns the residual upper and lower bounds. Before simulating the model residue, we use a distribution fitter to find the distribution that fits the residue in case the residue is not normally distributed. When the data distribution is established, we run the Monte Carlo simulation, which returns all the Monte

Carlo simulation paths and the mean of the simulation paths, as shown in Table 4.5 and algorithm 4.

Table 4.5. Monte Carlo simulation scheme algorithm

Algorithm 4: Monte Carlo Simulation

Input: data
model residue
number of simulation paths = n

Output: Monte Carlo simulation paths (MCSP)
mean Monte Carlo simulation path (MMSP)

m = data frame (model prediction)
r = data frame (model residue)
μ = mean of r
σ = standard deviation of residual
MCSP = [ ] # *Monte Carlo simulation paths*
MMCSP = [ ] # *mean Monte Carlo simulation path*
upper-bound = m + 2σ
lower-bound = m - 2σ
random data frame = create random data frame (0,1) → (len(r) x n)
    for i in random data frame
        MCSP [i] = Norm.ppf[i] + model(m)
        x = MCSP [MCSP > upper-bound]
        y = MCSP [MCSP < lower-bound]
        MCSP [MCSP > upper-bound] = some random data (upper-bound, lower-bound, x)
        MCSP [MCSP < lower] = some random data (upper-bound, lower-bound, y)
        MMCSP = MMCSP (MMCSP-rows)
    end for
return MCSP, MMCSP

The steps involved in running the Monte Carlo simulation scheme showing how the model's residues are collected for Monte Carlo simulation are given in Figure 4.5

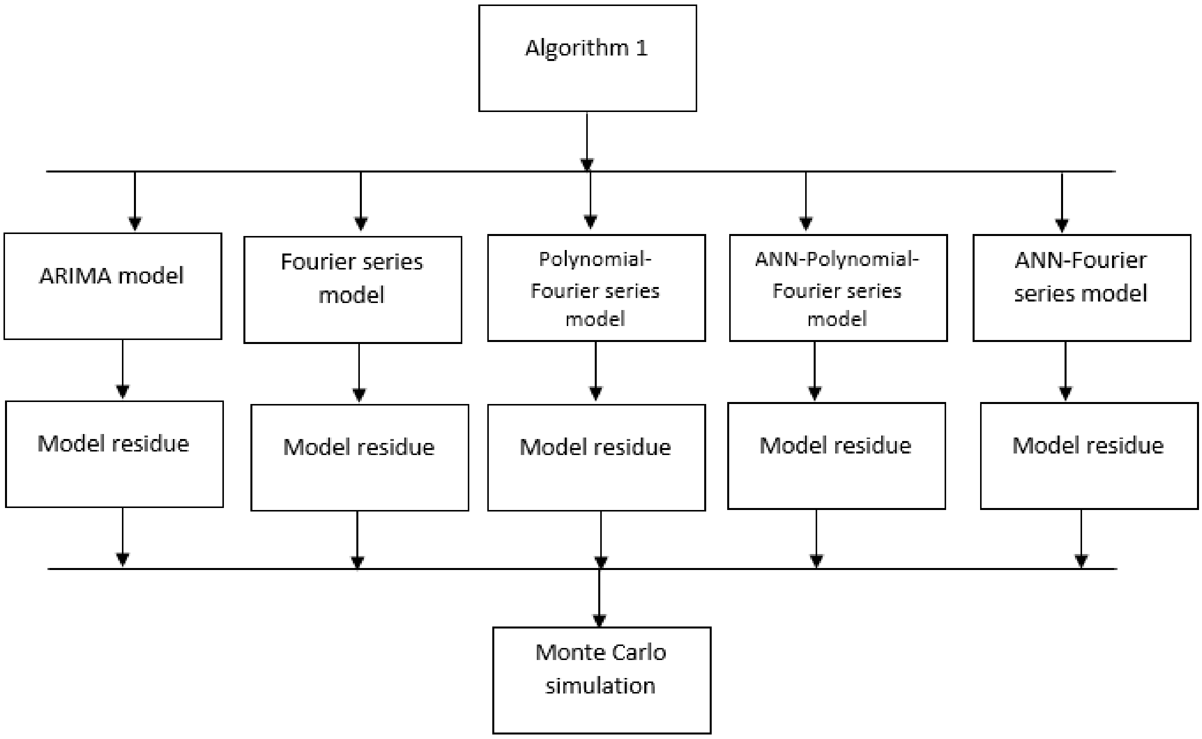


Figure 4.5. Proposed Monte Carlo simulation scheme

4.5. Model Selection Criteria

In this study, four time series forecasting models namely ARIMA, Fourier series, Polynomial-Fourier series, and ANN-Polynomial-Fourier series are employed from which one among them is selected for implementation on the given time series data. After obtaining the models' diagnostics results, then the model selection is done based on priority as shown in Figure 4.6 with ARIMA, Fourier series, and Polynomial-Fourier series having the same and highest priority while ANN-Polynomial-Fourier series and ANN-Fourier series models have the least selection priority.

This selection criterion is based on the findings of Makridakis et al. [88] that evaluate and compare the performance of regression and machine learning models across multiple forecasting horizons using a large subset of 1045 monthly time series forecasting problems. The study highlights the need for the results of classical methods to be used as a baseline when evaluating any time series machine learning forecasting model. This will demonstrate if the added complexity of the machine learning models is adding skill to the forecast or not.

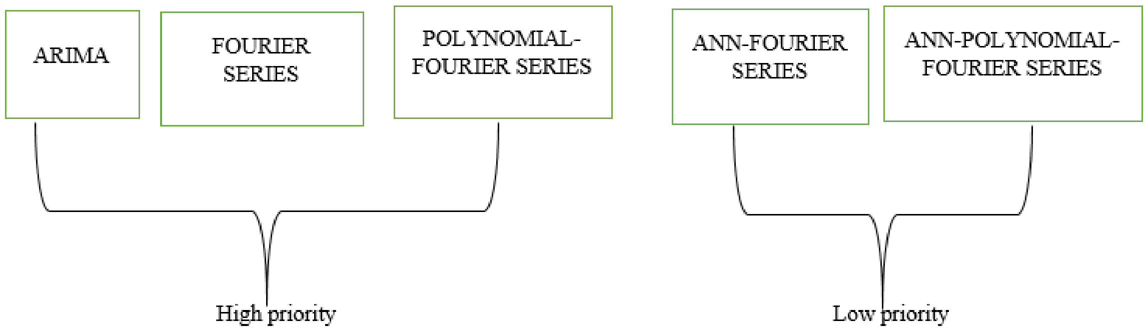


Figure 4.6. Model selection priority

4.6. Multi-step ahead forecasting

Nourani et al. [90] described the multi-step ahead forecasting or prediction as a forecasting technique that allows for estimating the future parameters of unknown time steps of a time series data. The multi-step ahead prediction is mostly needed and applied for time series data that required long time prediction according to Masum et al. [91]. In literature, direct and recursive algorithms are the most commonly used multi-step ahead forecasting [80]. Considering there is no dependency between the next and previous steps of the datasets used in this study, we elect to use the direct multi-step ahead forecasting technique. The major idea behind the direct multi-step ahead involves designing a separate model for every foresting time step in the series.

4.7. Testing Data Descriptions

The models proposed in this study are tested using two different time-series datasets. The first dataset is appliance energy consumption data collected from the UCI machine-learning repository, a collection of databases archive created by David Aha in 1987 at UC Irvine [92]. The dataset which is credited to the work of Candanedo et al. [93] contains information about the temperature and humidity conditions of a house that was monitored using a ZigBee wireless sensor network. The second dataset is tourism data that contains information about the number of monthly foreign visitors to Turkey from January 2004 to December 2020. The data is collected by the Turkish immigration service and available at the tourism portal of the Turkey Ministry of Culture and Tourism website (https://www.ktb.gov.tr/EN-99216/tourism.html). In addition to these datasets, we also used the monthly foreign visitors arrivals data of Japan, Malaysia, and Singapore from January 2004 to December 2020 for 12 months multi-step ahead forecasting of 2021. The Japan dataset was collected from the statistics portal of Japan Tourism Research and Consulting Co that is available at https://www.tourism.jp/en/tourism-database/stats/inbound/#monthly. The Malaysia dataset was collected from the Tourism Data section of the Tourism Malaysia Cooperate website available at the statistics portal http://mytourismdata.tourism.gov.my/. The Singapore dataset was collected from the Department of Statistics Singapore, which is available at the department's website at https://www.singstat.gov.sg/find-data/search-by-theme/industry/tourism/latest-data. Figure 4.7 depicts the monthly foreign visitor's arrivals datasets from January 2004 to December 2020.

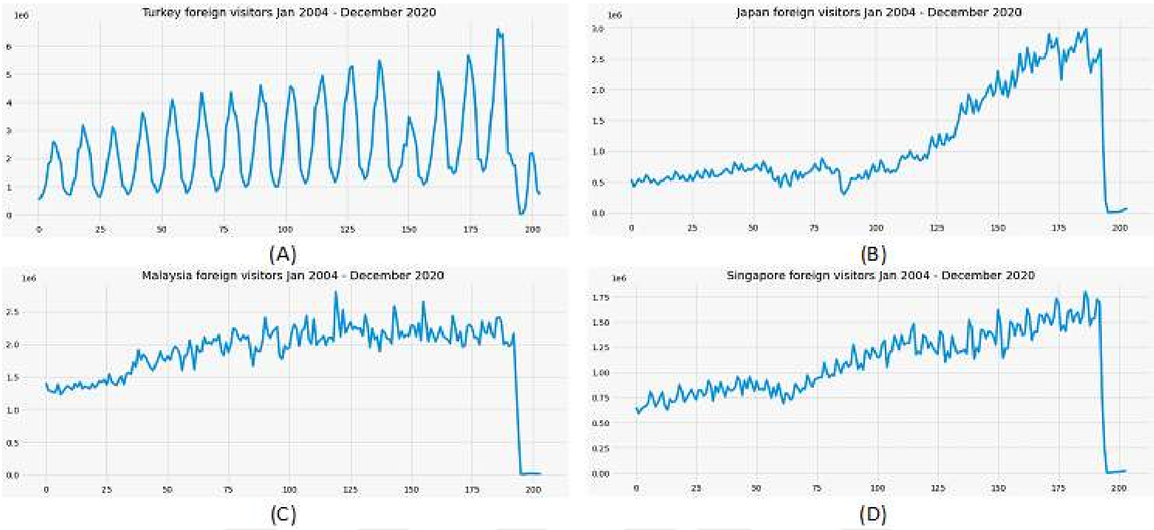


Figure 4.7. Monthly foreign visitors' arrival for January 2004 to December 2020 data plots for (A) Turkey (B) Japan (C) Malaysia (D) Singapore

Table 4.6 shows the characteristics of the datasets used in this study, which includes the number of attributes, instances, attribute type, and dataset type.

Table 4.6. Test dataset characteristics

Dataset	No. of Attributes	No. of Instances	Attribute Characteristics	Dataset Characteristic
Appliances energy consumption	29	19735	Real	Time series
Foreign visitors to Turkey	3	204	Integer	Time series
Foreign visitors to Japan	3	204	Integer	Time series
Foreign visitors to Malaysia	3	204	Integer	Time series
Foreign visitors to Singapore	3	204	Integer	Time series

5. RESULTS AND DISCUSSION

This chapter presents the models' training predictions and forecasts results on the datasets. The Monte Carlo simulation scheme is only applied to the models' forecasts region, not to the training prediction regions. This is done to achieve the objective of this study concerning using Monte Carlo simulation to forecast and optimize the proposed model's result. Because of the novel COVID-19 pandemic, the monthly foreign visitors to Turkey dataset is divided into two regions so that we can evaluate the models' performance before and during the pandemic separately.

For simplicity, we will refer to the monthly foreign visitor to Turkey dataset as tourism data in this chapter. Likewise, we will refer to the appliance consumption dataset as temperature data. This study will use the R^2 estimate value to evaluate how much variation in the data the models can account for. The R^2 estimate can be a biased estimator as explained by Akossou [94]. Yet, in the case of time series data analysis, the R2 estimate is a useful and good estimator according to the explanation of Monteiro and Costa [95]. Pierce [96] also discusses how good an estimator is R^2 when working with time-series data. As for the models, the evaluation metrics we will use in this study are the root means square error (RMSE) and mean absolute percentage error (MAPE). Zhang et al. [97] and Bi et al. [98] respectively provide the formulas for RMSE and MAPE as:

$$RMSE = \sqrt{\frac{1}{n}\sum_{t=1}^{n} e_t^2}$$

$$MAPE = \frac{1}{n}\sum_{t=1}^{n} \left|\frac{e_t}{y_t}\right| \times 100$$

Where n is the number of the data points, $e_t = y_t - f_t$ is the forecast error, and y_t is the actual value, f_t is the forecasted value.

The tourism data plot given in Figure 5.1 shows that the data deviated from its normal pattern with a sudden plunge in the number of foreign visitors around the month 195 (March 2020). Perhaps, this is because of the COVID-19 pandemic that causes the closure of airports, countries' borders, grounding of airplanes, canceled hotel reservations, etc., which leads to a low number of foreign visitors. This sudden change in the behavior of the data has caused uncertainty in forecasting the future number of foreign visitors, which is why the study of Günay et al. [99] call for urgent implementation of a forecasting model that can account for uncertainty in the future number of foreign visitors to Turkey.

To archive forecasting amid the effect of the COVID-19 pandemic on the tourism data and to test the robustness of our proposed models we will apply our models to the regions before and during the pandemic. The two forecasting regions will allow us to see and evaluate the model's performance before the pandemic when there are fewer uncertainties and during the pandemic when there are more uncertainties. However, the Monte Carlo simulation will be useful here as the scheme will be able to account for the uncertainties and optimize the forecast results of the proposed models.

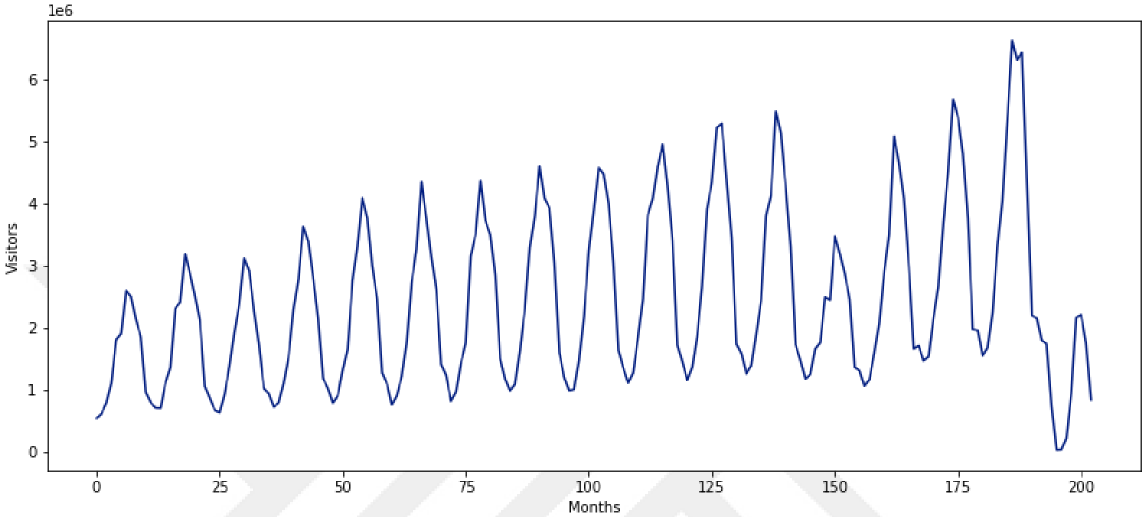


Figure 5.1. Tourism data plot from January 2004 to November 2020

5.1. Tourism data forecast results

As we mentioned earlier the tourism data forecast results would be presented in two forecasting horizons i.e. the region before and during the COVID-19 pandemic. Moreover, the Monte Carlo simulation scheme with 100 simulation paths will be applied to the model's forecast on the data to enable Monte Carlo forecasting and optimization of the model's forecast results. Since each of the 100 simulation paths will have its different direction, the mean of the simulation paths is calculated and used for presenting the Monte Carlo forecast result. The tourism data is divided into 70% for training and 30% for testing. The training period before the pandemic horizon is from January 2004 to March 2015, while the forecast period range is April 2015 to December 2019. During the pandemic period the whole available dataset is utilized and the period from January 2004 to July 2015 is used for training, the period from August 2015 to November 2020 is used for testing.

5.1.1. ARIMA forecast results before the COVID-19 pandemic

The ARIMA model was able to account for 81% variation in the tourism data during the training period before the COVID-19 pandemic as given in Table 5.1. The model also shows an error of RMSE of 5 people and forecast accuracy of MAPE of 16.75 in the training stage according to the values in Table 5.1.

Table 5.1. Simulation and ARIMA model results before COVID-19

	Number of simulations	R^2	RMSE	MAPE
ARIMA training prediction	1	0.81	5.41	16.75
ARIMA forecast	1	0.78	6.66	19.98
Monte Carlo mean forecast	100	0.80	5.82	17.54

The graphical forecast result of the ARIMA model before the pandemic period is shown in Figure 5.2. The model shows a better performance during the training stage when compared to the R^2 estimate of 78%, RMSE of 7 people, and MAPE of 19.98% obtained during the forecasting stage in the period before the pandemic.

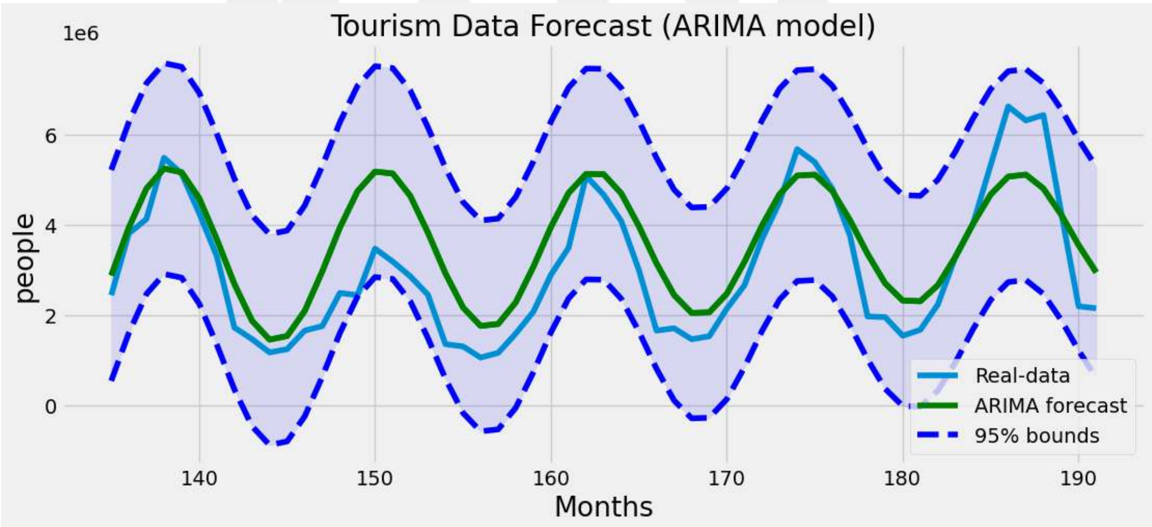
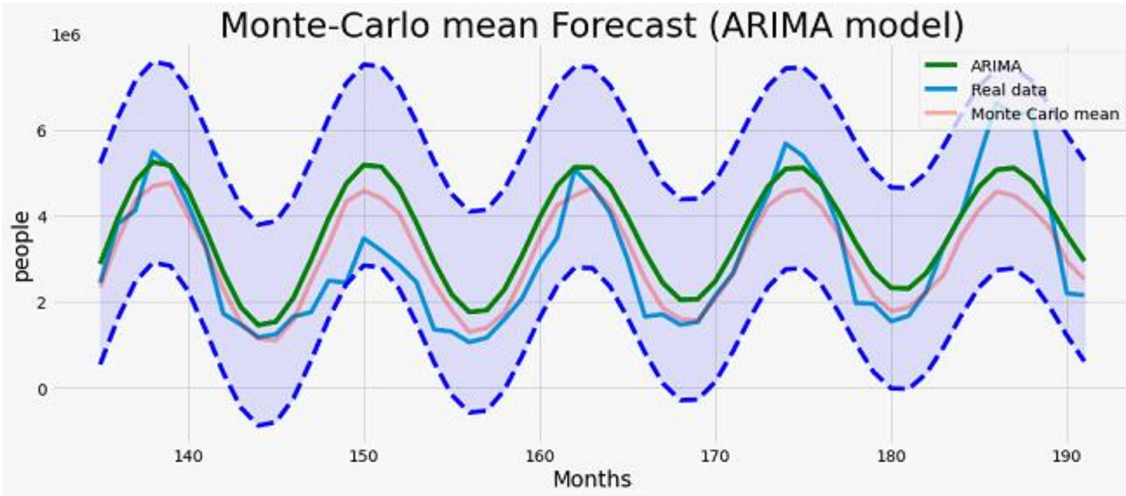


Figure 5.2. ARIMA model forecast before COVID-19 pandemic

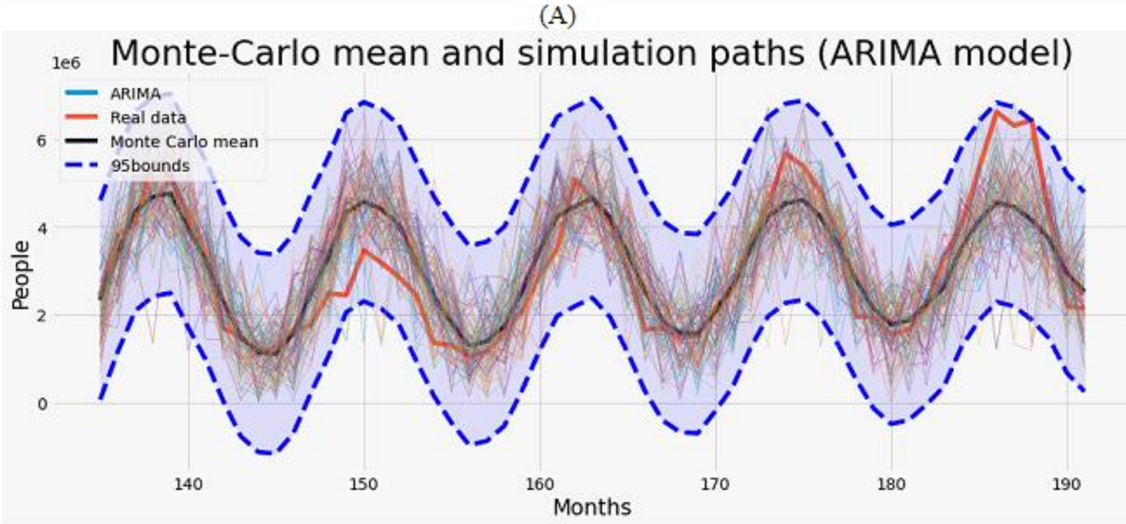
5.1.2. Monte Carlo forecast result before COVID-19 for ARIMA model

The Monte Carlo simulation scheme is only applied to the forecasting region of the ARIMA model. We run 100 Monte Carlo simulations on the tourism data and compute the mean of the generated simulation paths. In Figure 5.3 (A), the mean of the Monte Carlo simulation can be seen to resemble the ARIMA model line with a slight increase in inaccuracy. The thin lines in Figure 5.3 (B) represent each of the 100 Monte Carlo simulation paths. The increase in accuracy can be observed from the numerical values given in Table 5.1. The Monte Carlo forecast was able to account for 80% variation in the tourism data, which shows an increase of 2% from the 78% R^2 estimate value that was achieved by the ARIMA model.

The Monte Carlo simulation also reduces the model error from the previous error of 7 people produced by the ARIMA model to the error of 6 people according to the RMSE value in Table 5.1. Similarly, the forecast accuracy measured by MAPE shows that the Monte Carlo forecast has increased the ARIMA model forecast accuracy from having a MAPE of 19.98% obtained from the model forecast to 17.54% during the Monte Carlo forecast. This result shows that the Monte Carlo simulation was able to increase the performance of the ARIMA model in the forecasting region before the COVID-19 pandemic for the tourism data.



(A)



(B)

Figure 5.3. (A) Monte Carlo mean forecast for ARIMA model before COVID-19 (B) 100 Monte Carlo simulation paths for ARIMA model before COVID-19

5.1.3. ARIMA forecast results during the COVID-19 pandemic

In the training period between January 2004 and July 2015 during the COVID-19 pandemic, the ARIMA model accounts for 82% variation in the tourism data according to the obtained R^2 estimate value given in Table 5.2. The ARIMA model's respective RMSE and MAPE performance in this training period are 6 people and 17.53%.

Table 5.2. Simulation and ARIMA model results during COVID-19

	Number of simulations	R^2	RMSE	MAPE
ARIMA training prediction	1	0.82	6.24	17.53
ARIMA forecast	1	0.75	7.90	21.07
Monte Carlo mean forecast	100	0.80	6.41	18.77

The ARIMA model forecast for the tourism data shown in Figure 5.4 shows that the model performance during the pandemic is not as good as the performance before the pandemic. While in the forecast region before the pandemic the model accounts for 78% variation in the data, in the forecast region during the pandemic it accounts for 75% which indicates a reduction in the ability of the ARIMA model to forecast the tourism data during the pandemic. The numerical values reported in Table 5.2 also show that the ARIMA model forecast produces RMSE of 8 people and MAPE of 21.07% during the pandemic.

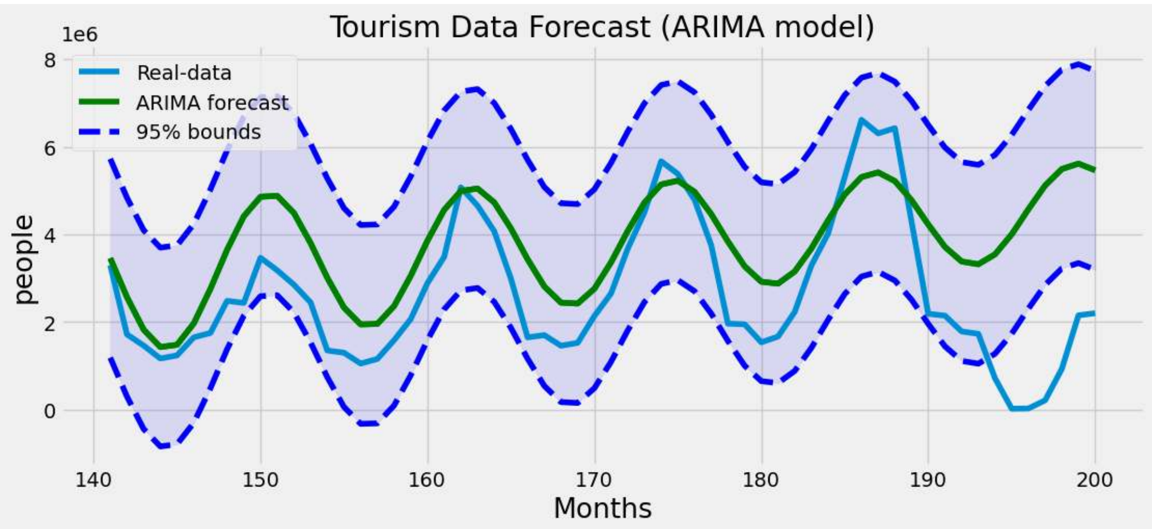
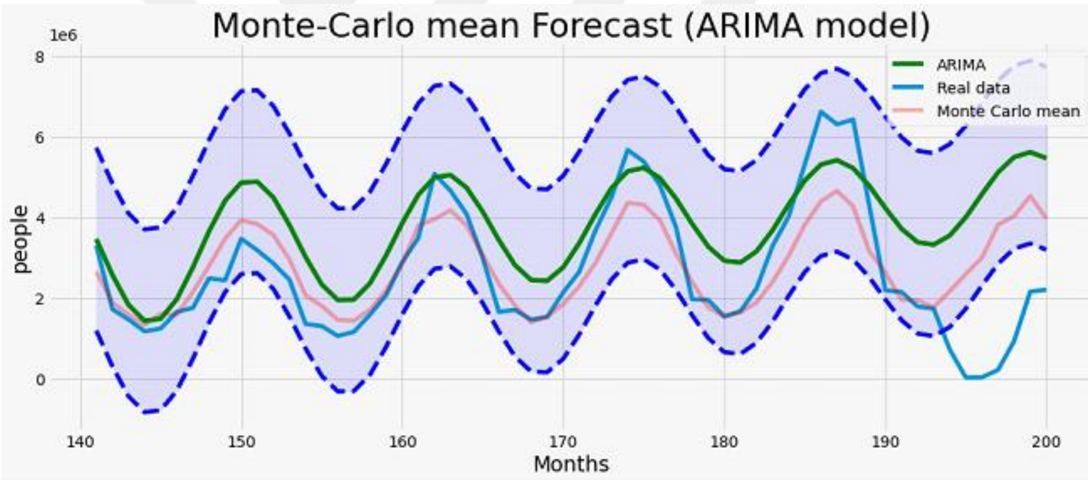


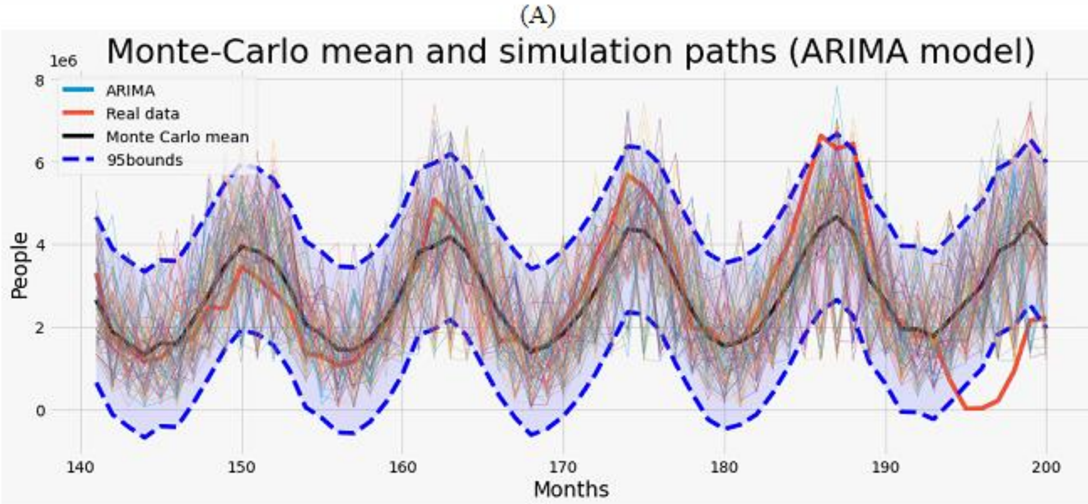
Figure 5.4. ARIMA model forecast during the COVID-19 pandemic

5.1.4. Monte Carlo forecast result during the COVID-19 pandemic for ARIMA model

We have seen from section 5.1.3 how the uncertainty in the tourism data caused by the COVID-19 pandemic has affected the ARIMA model's ability to forecast the data. However, the Monte Carlo scheme captured more of the uncertainties and increased the performance accuracy of the model in the forecasting region. This can be seen in Figure 5.5 (A) how the Monte Carlo mean assume values that are closer to the real data and the model regression curve. Figure 5.5. (B) also shows how the 100 simulation paths can capture all the uncertainties in the tourism data during the pandemic forecasting horizon. Moreover, it is evident from the values in Table 5.2 the Monte Carlo forecast has achieved an 80% R^2 estimate value, which is an increase of 5% from the ARIMA model's 75% R^2 value. Accordingly, the Monte Carlo forecast reduces the ARIMA model error from RMSE of 8 people to 6 people, and it also increases the forecast accuracy from MAPE of 21.07% to 18.77%.



(A)

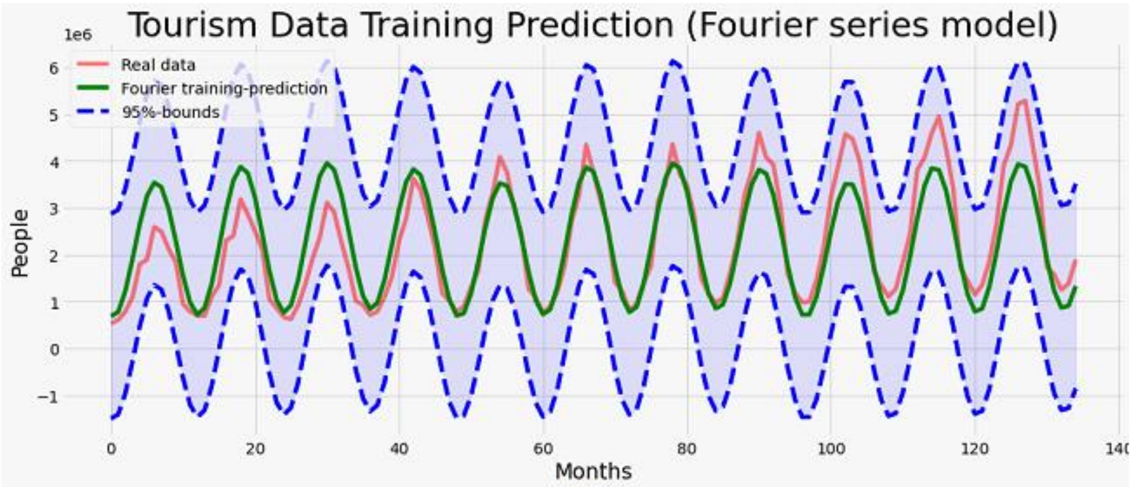


(B)

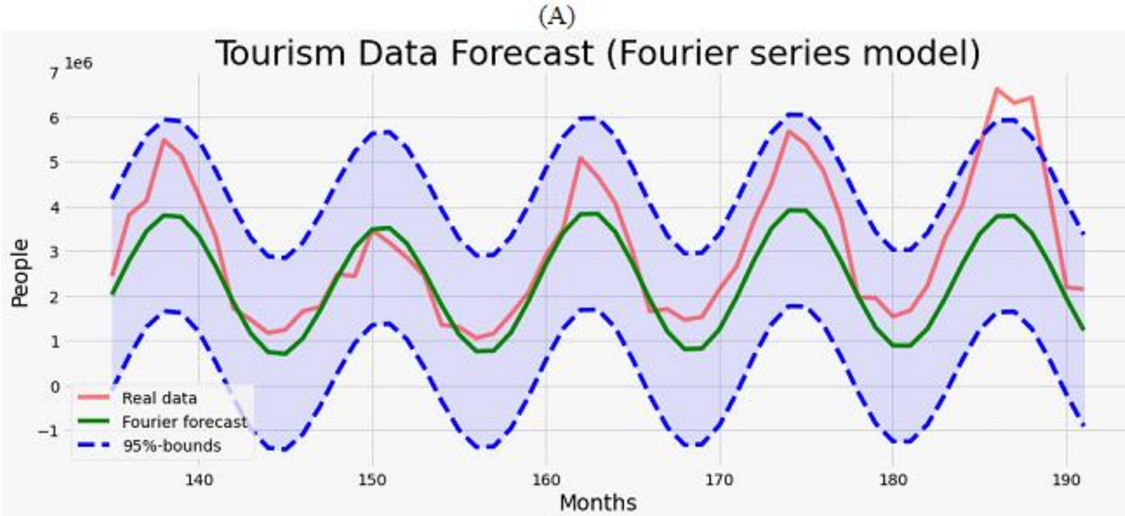
Figure 5.5. (A) Monte Carlo mean forecast for ARIMA model during the COVID-19 (B) 100 Monte Carlo simulation paths for ARIMA model during the COVID-19

5.1.5. Fourier series model forecast results before the COVID-19 pandemic

The Fourier series model training prediction and forecast results in the training and forecasting periods before the COVID-19 pandemic are respectively shown in Figure 5.6 (A) and (B). The results are shown within ±2σ from the regression curve and the model training and forecast numerical values are given in Table 5.3.



(A)



(B)

Figure 5.6. (A) Fourier series model training prediction before COVID-19 (B) Fourier series model forecast before COVID-19

In the training period before the pandemic, the Fourier series model can account for 89% variation in the data with RMSE of 4 people and MAPE of 12.88% according to the values given in Table 5.3. The Fourier series model training performance in the period before the pandemic is better than its performance in the testing period.

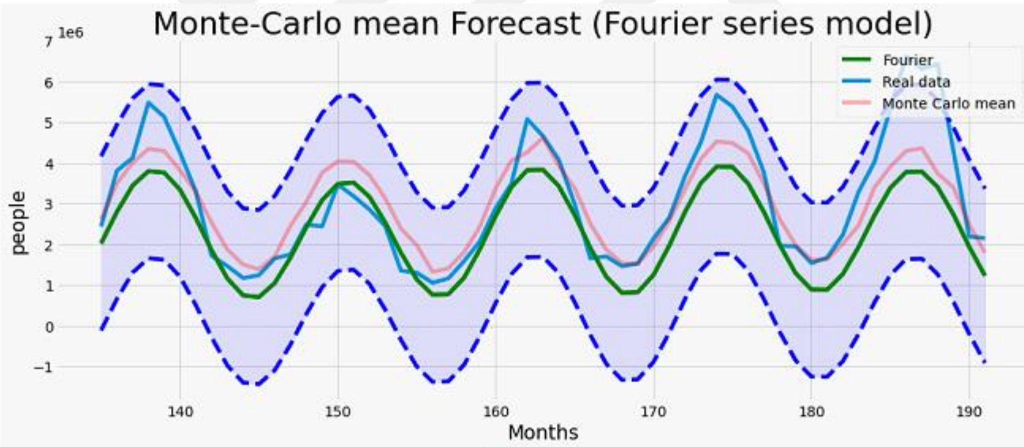
Table 5.3. Simulation and Fourier series model results before COVID-19

	Number of simulations	R^2	RMSE	MAPE
Fourier series training prediction	1	0.89	4.09	12.88
Fourier series forecast	1	0.71	7.32	23.45
Monte Carlo mean forecast	100	0.79	6.11	18.98

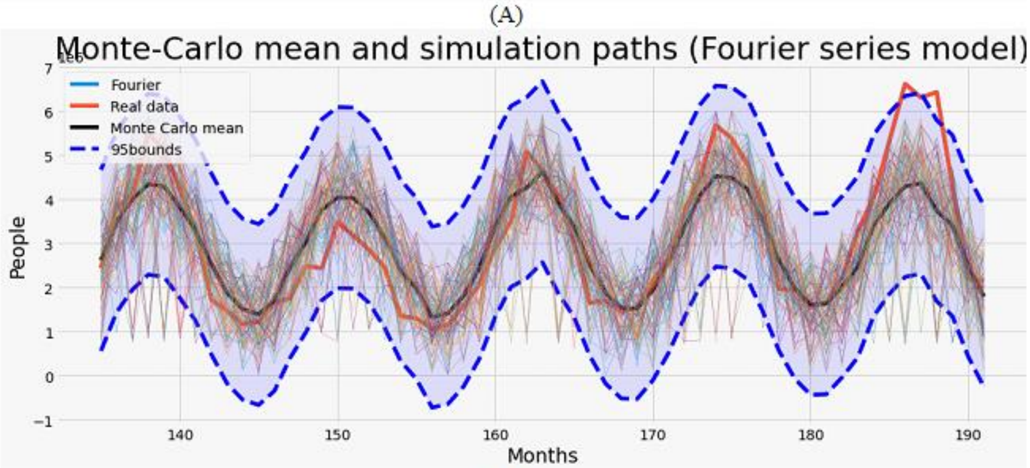
The forecast result shows that the Fourier series model can account for 71% variation in the tourism data according to the R^2 estimate value given in Table 5.3. It also shows that the Fourier series model has a forecast error of RMSE of 7 people and forecast accuracy of MAPE of 23.45%.

5.1.6. Monte Carlo forecast result before the COVID-19 for Fourier series model

The mean of the Monte Carlo simulation and the 100 Monte Carlo simulation paths for the Fourier series model are given in Figure 5.7 (A) and (B).



(A)



(B)

Figure 5.7. (A) Monte Carlo mean forecast for the Fourier series model before the COVID-19 (B) 100 Monte Carlo simulation paths for the Fourier series model before the COVID-19

The Monte Carlo simulation scheme is applied to the Fourier series model forecast region in the forecasting period before the pandemic. From Table 5.3 we can see that the Monte Carlo forecast was able to increase the Fourier series model performance with an R^2 estimate of 79% from the previous model's R^2 estimate value of 71%. Similarly, the RMSE value was reduced by the Monte Carlo scheme from the model's RMSE of 7 people to 6 people, and the MAPE is also increased from 23.45% to 18.98%.

5.1.7. Fourier series model forecast results during COVID-19 pandemic

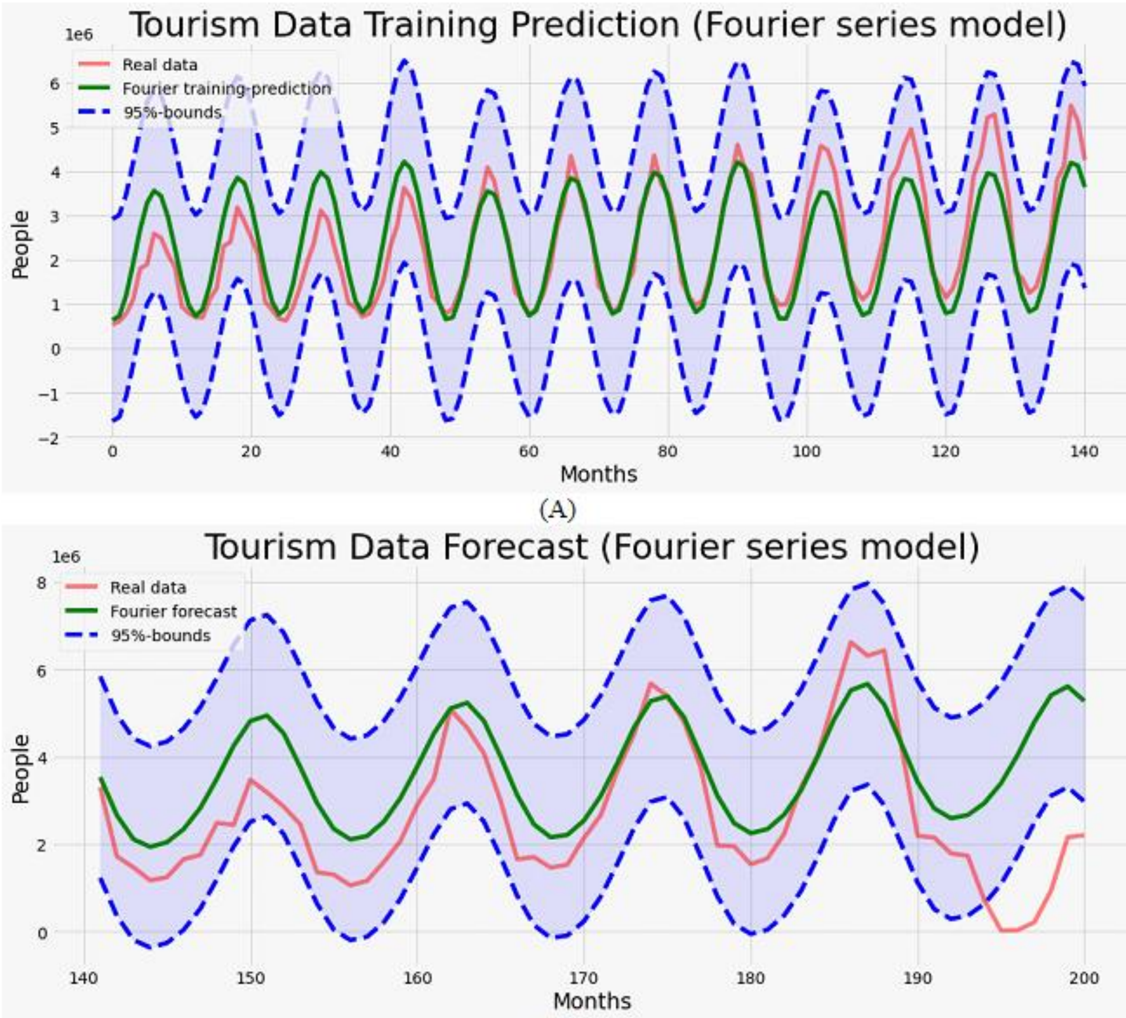
During the training period in the pandemic region, the Fourier series model was able to get a 91% R^2 estimate value, with respective forecast error and accuracy of RMSE of 4 people and MAPE of 13.13% as per the numerical results given in Table 5.4. The performance of the model in the training period during the pandemic is slightly better than its performance before the pandemic. This might be because of having more training data since more training data can give a better training prediction [100].

Table 5.4. Simulation and Fourier series model results during COVID-19

	Number of simulations	**R^2**	**RMSE**	**MAPE**
Fourier series training prediction	1	0.91	3.97	13.13
Fourier series forecast	1	0.69	8.34	24.91
Monte Carlo mean forecast	100	0.75	6.55	19.78

However, the Fourier model forecast in the forecast region during the pandemic is not as good as before. Perhaps, this has to do with the pandemic's effect on the data behavior that causes the data to change its direction. Figure 5.8 (A) and (B) depict the graphical results of the Fourier series model training prediction and model forecast during the COVID-19 pandemic period.

The sudden drop in the number of foreign visitors can be observed around months 195 and 196 (March and April 2020) from Figure 5.8 (B). This anomaly that was caused by the pandemic has brought about uncertainty and change of behavior in the data. Hence, why the Fourier series model was model forecast performance is not outstanding as the model only accounted for 69% variation in the data. The RMSE value in Table 5.4 also shows that the model error went up to 8 people with a MAPE of 24.91%.



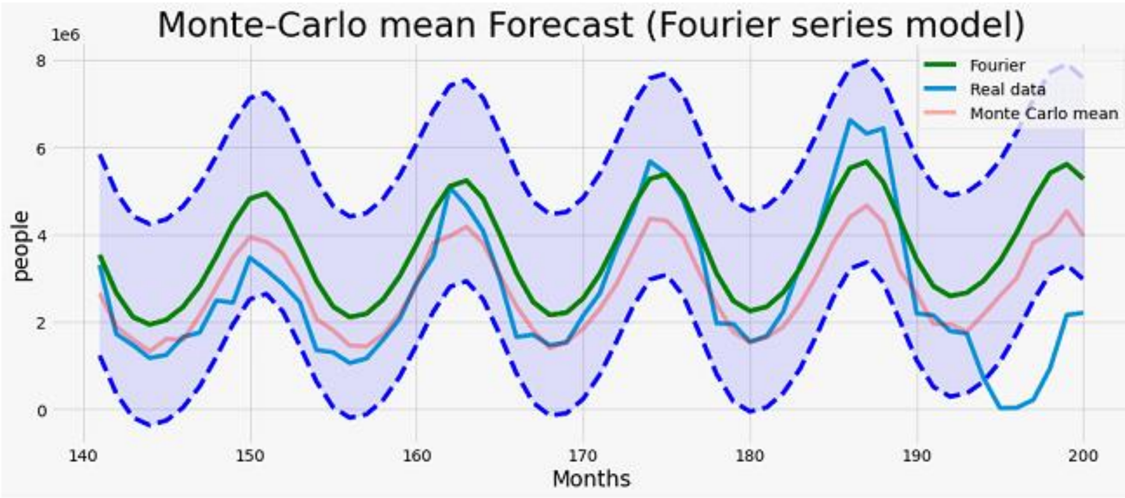
(A)

(B)

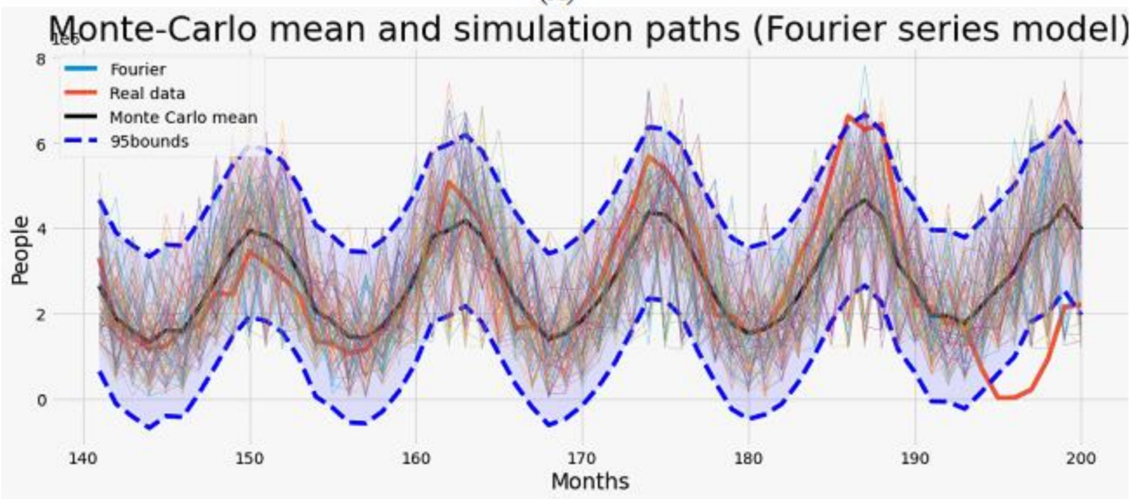
Figure 5.8. (A) Fourier series model training prediction during COVID-19 (B) Fourier series model forecast during COVID-19

5.1.8. Monte Carlo forecast result during the COVID-19 for Fourier series model

The Monte Carlo simulation forecast on the residue of the Fourier series model was able to increase the model performance. This can be seen in Table 5.8 where the simulation was able to achieve a 75% R^2 value, which is an increase of 6% from the 69% R^2 estimate value produced by the Fourier model. Similarly, the RMSE and MAPE of the model were improved by the Monte Carlo simulation scheme by achieving an RMSE of 7 people from the model's 8 people and a MAPE of 19.78% from the model's 24.91%. Figure 5.9 (A) and (B) show the Monte Carlo simulation means and the 100 Monte Carlo simulation paths respectively.



(A)



(B)

Figure 5.9. (A) Monte Carlo mean forecast for the Fourier series model during the COVID-19 (B) 100 Monte Carlo simulation paths for the Fourier series model during the COVID-19

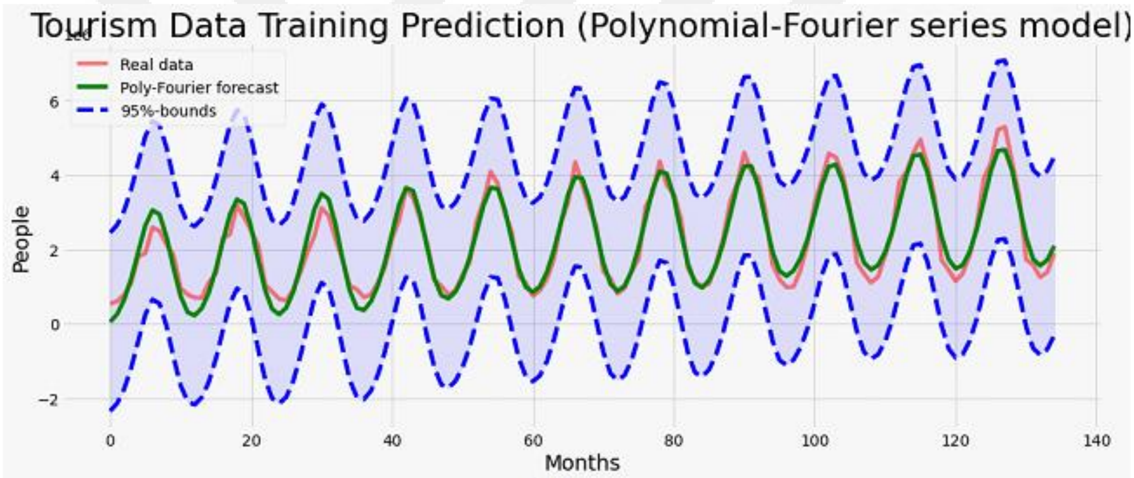
5.1.9. Polynomial-Fourier series model forecast results before COVID-19 pandemic

The Polynomial-Fourier series model is one of the proposed models of this study that was designed to capture the trend and seasonal component of any time series data using the combination of polynomial and Fourier series fittings. The training prediction and forecast results of the proposed model are shown in Table 5.5.

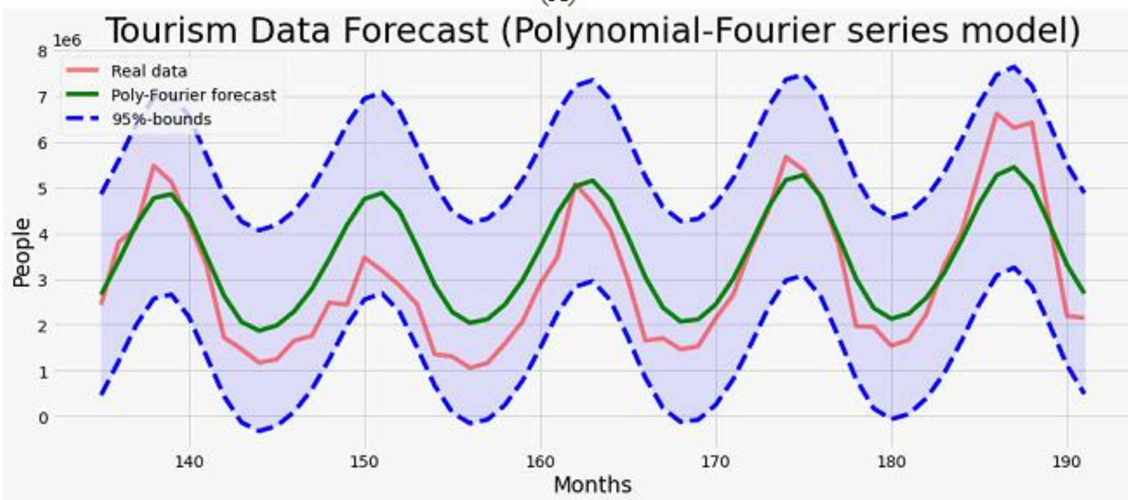
Table 5.5. Simulation and Polynomial-Fourier series model results before COVID-19

	Number of simulations	**R^2**	**RMSE**	**MAPE**
Polynomial-Fourier series training prediction	1	0.95	2.41	08.69
Polynomial-Fourier series forecast	1	0.90	3.04	10.65
Monte Carlo mean forecast	100	0.92	2.55	09.32

In the training period before the pandemic, the proposed Polynomial-Fourier series model accounts for 95% variation in the tourism data as per the numerical values presented in Table 5.5. The proposed model also shows an RMSE of 2 people and a MAPE of 08.69%. Meanwhile, in the forecasting region for the period before the pandemic, the model was able to achieve a 90% R^2 estimate value, RMSE of 3 people, and MAPE of 10.65%. The results show that the model has outperformed the performance of the ARIMA and Fourier series models in this forecasting region before the pandemic. This indicates that the Polynomial-Fourier series has demonstrated more accuracy in capturing and forecasting the data using the series historical data. The training prediction and forecasting of the proposed Polynomial-Fourier series model are respectively shown in Figure 5.10 (A) and (B).



(A)



(B)

Figure 5.10. (A) Polynomial-Fourier series model training prediction before COVID-19 (B) Polynomial-Fourier series model forecast before COVID-19

5.1.10. Monte Carlo forecast result before COVID-19 for Polynomial-Fourier series model

The Monte Carlo simulation scheme is applied to the proposed Polynomial-Fourier series model forecast result. The simulation was able to increase the model performance from having an R^2 estimate value of 90% to an R^2 estimate value of 92% according to the values provided in Table 5.5. Similarly, the Monte Carlo simulation respectively improves the forecast error and accuracy. In the forecast region before the pandemic, the Monte Carlo was able to decrease the proposed model error from RMSE of 3 people to RMSE of 2 people. The forecast accuracy was also increased from the model's MAPE of 10.65% to MAPE of 9.32%. Figure 5.11 (A) and (B) respectively show the mean of the Monte Carlo simulation and the 100 simulation paths. The thin lines in Figure 5.11 (B) are the 100 simulation paths that cover the possible future scenarios in the data.

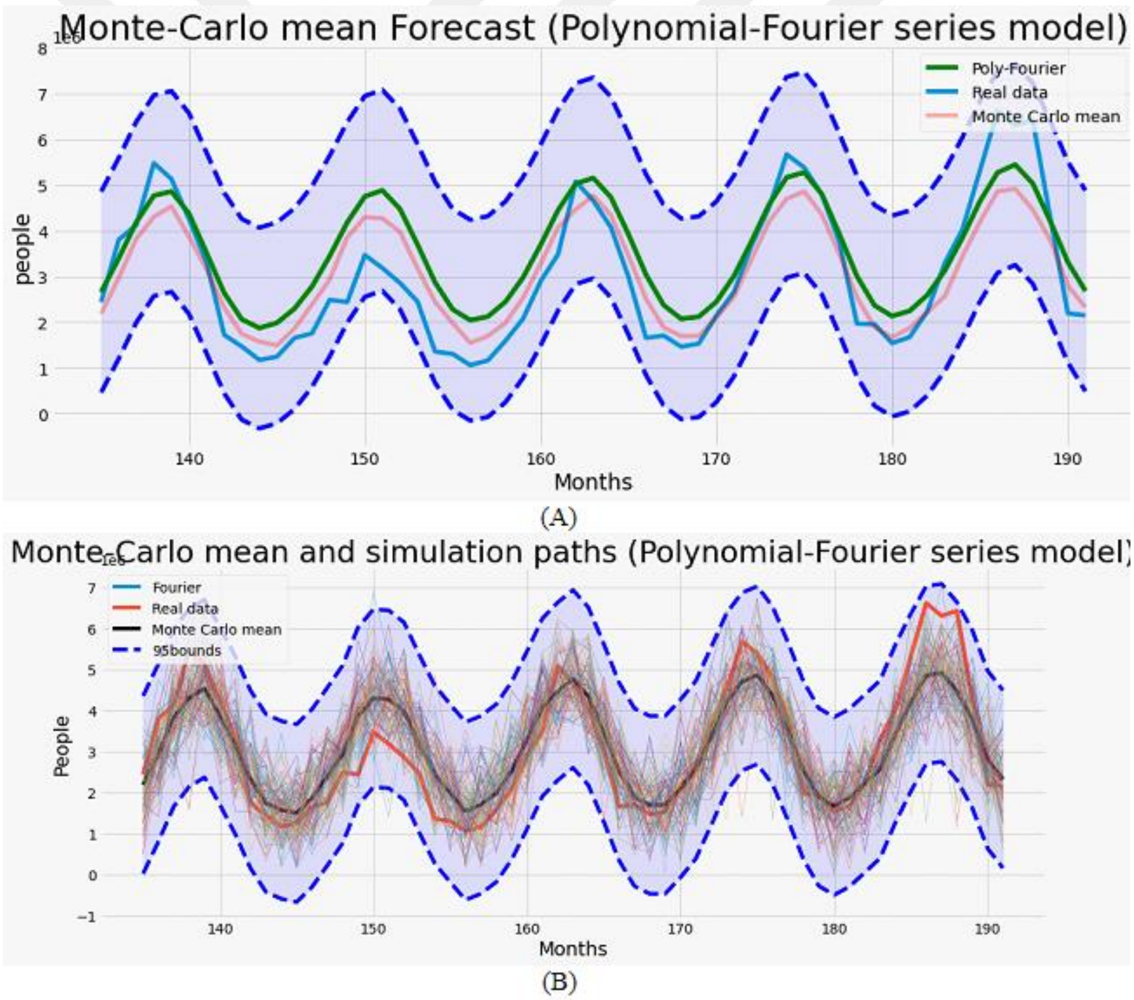


Figure 5.11. (A) Monte Carlo mean forecast for the Polynomial-Fourier series model before the COVID-19 (B) 100 Monte Carlo simulation paths for the Polynomial-Fourier series model before the COVID-19

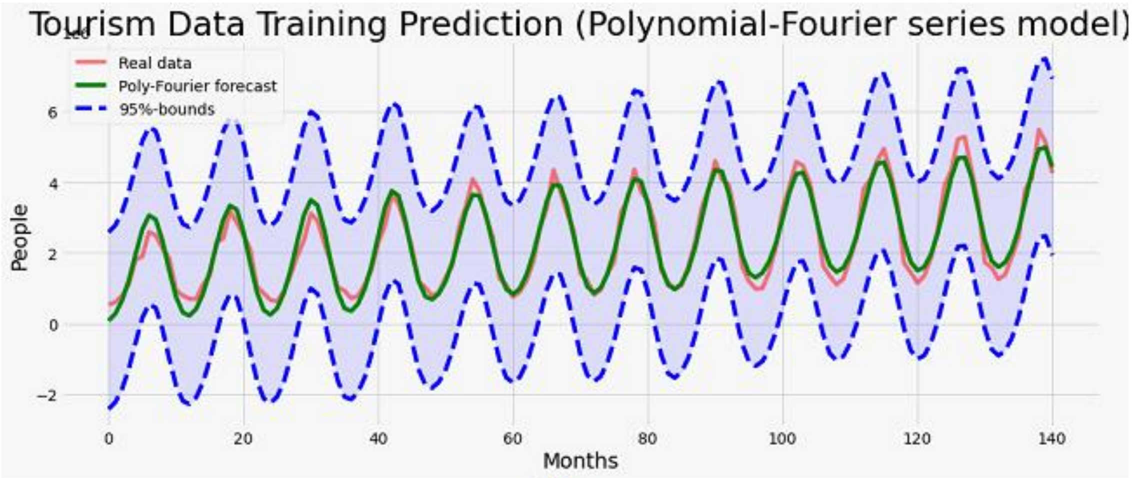
5.1.11. Polynomial-Fourier series model forecast results during COVID-19 pandemic

The proposed Polynomial-Fourier series model performance for both the training and forecast region in the period during the pandemic is reported in Table 5.6. The proposed model shows a decent performance in the training region of the forecasting horizon during the pandemic. The model was able to account for 96% variation in the tourism data while having an RMSE of 2 people and MAPE of 8.44%.

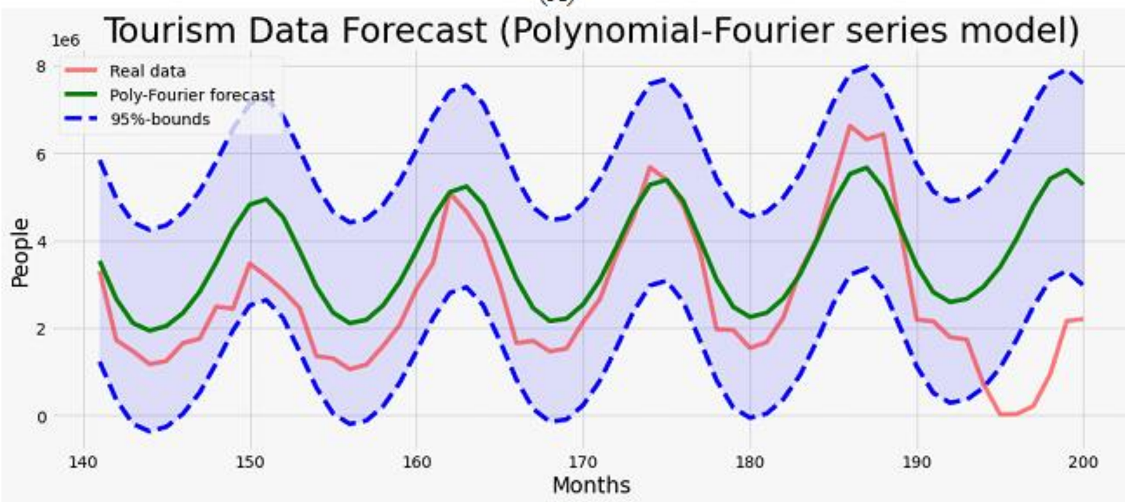
Table 5.6. Simulation and Polynomial-Fourier series model results during COVID-19

	Number of simulations	**R^2**	**RMSE**	**MAPE**
Polynomial-Fourier series training prediction	1	0.96	2.02	08.44
Polynomial-Fourier series forecast	1	0.88	4.09	11.75
Monte Carlo mean forecast	100	0.90	3.35	10.98

In the forecast region during the pandemic, the proposed model was able to account for 88% variation in the data with RMSE of 4 people and MAPE of 11.75%. The training prediction and forecast result of the Polynomial-Fourier series model are respectively shown in Figure 5.12 (A) and (B). Similarly, the forecast result of the proposed Polynomial-Fourier series model in the pandemic period is better than that of the ARIMA and Fourier series models. This shows that the model was able to capture some level of uncertainty in the data that was not captured by the ARIMA and Fourier series models.



(A)

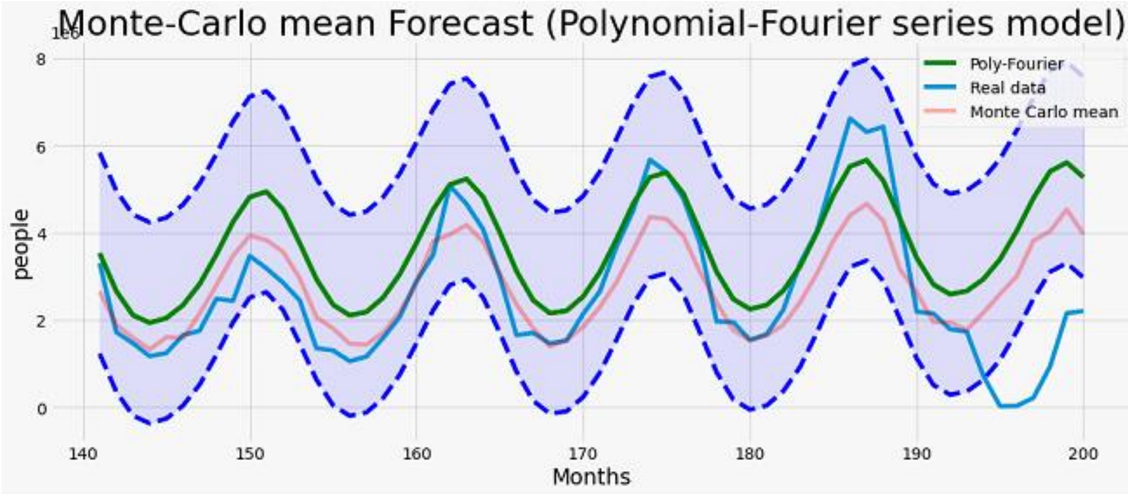


(B)

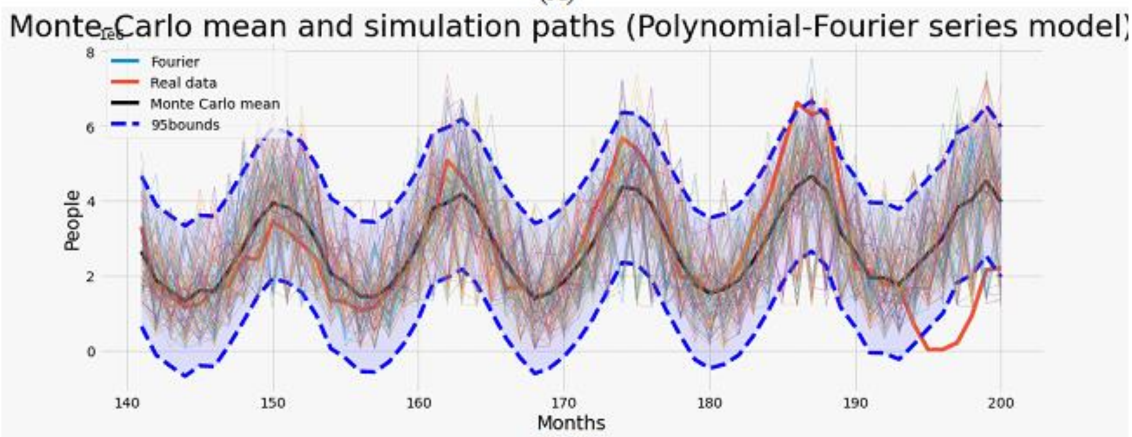
Figure 5.12. (A) Polynomial-Fourier series model training prediction during COVID-19 (B) Polynomial-Fourier series model forecast during COVID-19

5.1.12. Monte Carlo forecast result during COVID-19 for Polynomial-Fourier series model

Although the Polynomial-Fourier series model has captured some level of uncertainty in the tourism data compared with the ARIMA and Fourier series models, the Monte Carlo simulation captured all of the uncertainties during the pandemic period, which in turn improved the model's forecast result. The mean of the Monte Carlo simulation and the 100 Monte Carl simulation paths are respectively graphically shown in Figure 5.13 (A) and (B). The mean of the Monte Carlo simulation was able to account for 90% variation in the data, which is a 2% increase in the R^2 estimate value obtained by the proposed model according to the results presented in Table 5.6. Accordingly, the values in Table 5.6 show that the Monte Carlo simulation has an RMSE of 3 people and a prediction accuracy of 10.98%.



(A)



(B)

Figure 5.13. (A) Monte Carlo mean forecast for the Polynomial-Fourier series model during the COVID-19 (B) 100 Monte Carlo simulation paths for the Polynomial-Fourier series model during the COVID-19

5.1.13. ANN forecast on Polynomial-Fourier series model residue

The residue of the Polynomial-Fourier series model forecast is collected and modeled using the artificial neural network method. The ANN modeled residue results in the period before and during the COVID-19 pandemic periods are given in Table 5.7.

Table 5.7. ANN forecast on the Polynomial-Fourier series model residue

	R^2	RMSE	MAPE
ANN forecast on Polynomial-Fourier series model residue before COVID-19	0.98	1.09	04.91
ANN forecast on Polynomial-Fourier series model residue during COVID-19	0.97	1.48	05.67

In the period before the pandemic, the ANN forecast on the residue produces a 98% R^2 estimate value with just an RMSE of one person and MAPE of 4.91% according to the result in Table 5.7. The ANN forecast on the residue during the pandemic also yields a decent result with the forecast accounting for 97% variation in the data, forecast error of RMSE of one person, and forecast accuracy of MAPE of 5.67%. Figure 5.14 (A) and (B) respectively show the ANN forecast on the Polynomial-Fourier series model in the forecasting region before and during the COVID-19 pandemic.

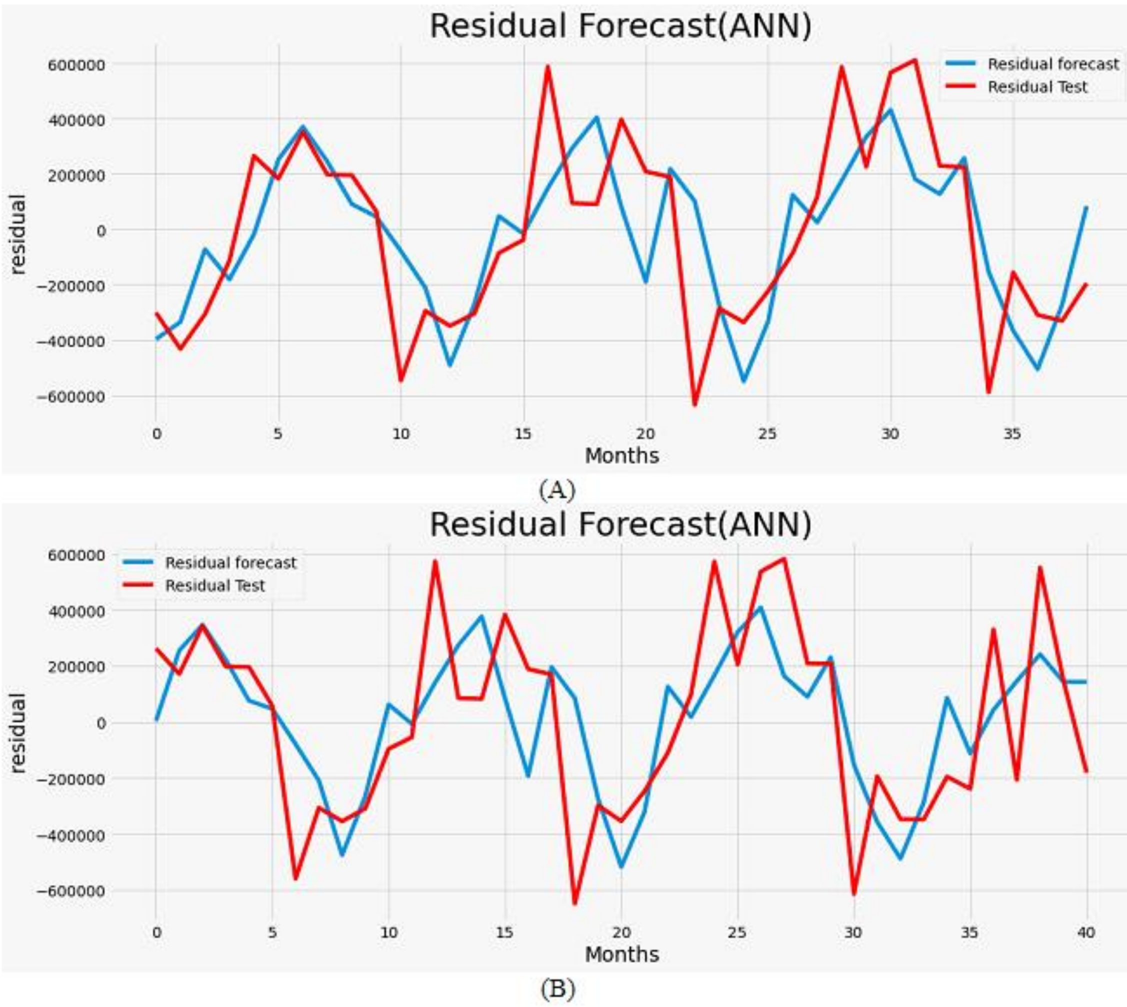
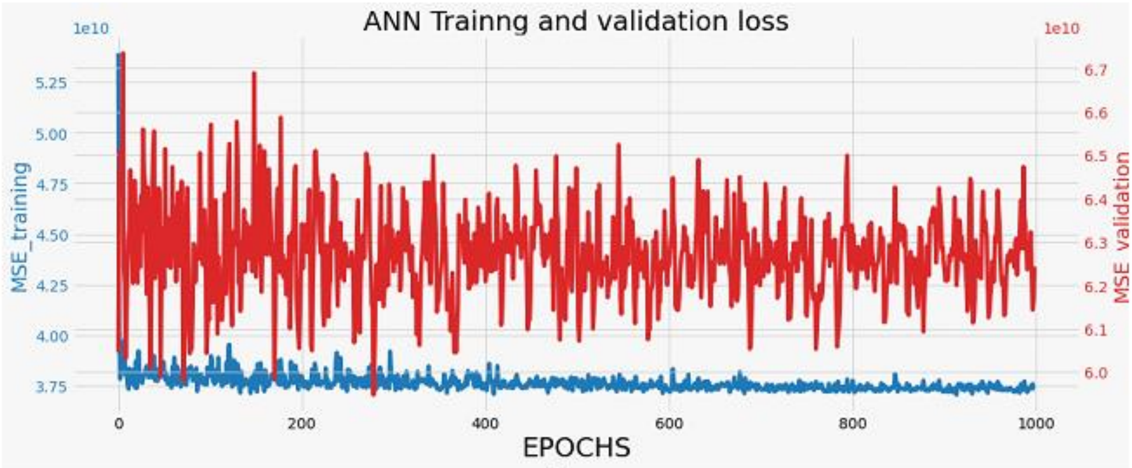
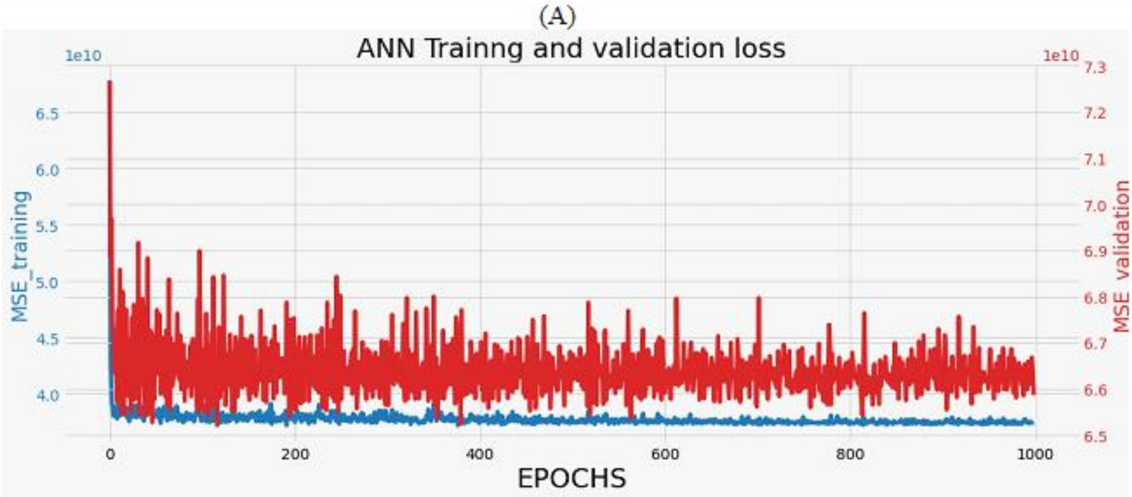


Figure 5.14. ANN forecast on Polynomial-Fourier series model (A) before COVID-19 (B) during COVID-19

The ANN training and validation loss results before and during the pandemic are respectively shown in Figure 5.15 (A) and (B). The training errors demonstrated by the blue curves in Figure 5.15 (A) and (B) indicate that the model has shown a good learning rate. Accordingly, the model has also demonstrated decent learning from the training set as can be observed from the red curves in Figure 5.15 (A) and (B), which are the validation curves.



(A)



(B)

Figure 5.15. ANN on Polynomial-Fourier series model training and validation loss (A) before COVID-19 (B) during COVID-19

5.1.14. ANN-Polynomial-Fourier series model forecast results before COVID-19 pandemic

The proposed ANN-Polynomial-Fourier series model was able to account for 97% variation in the tourism data in the training period of the region before the pandemic as presented in the numerical results given in Table 5.8. Similarly, in the same training prediction region before the pandemic, the proposed model achieved a prediction of an RMSE of 2 people and a prediction accuracy of a MAPE of 6.96% as shown in Table 5.8.

Table 5.8. Simulation and ANN-Polynomial-Fourier series model before COVID-19

	Number of simulations	R^2	RMSE	MAPE
ANN-Polynomial-Fourier series training prediction	1	0.97	1.58	06.96
ANN-Polynomial-Fourier series forecast	1	0.94	2.43	08.70
Monte Carlo mean forecast	100	0.95	1.97	07.31

In the forecast region before the pandemic, the proposed ANN-Polynomial-Fourier series model shows better performance when compared to the proposed Polynomial-Fourier series model and the benchmarked ARIMA and Fourier series models. The training prediction and the forecast results of the model are shown in Figure 5.16 (A) and (B) respectively. According to the results given in Table 5.8, the proposed ANN-Polynomial-Fourier series model was able to achieve an R^2 estimate value of 94% that accounts for the variation in the data. The RMSE obtained by the proposed model in the forecasting region before the pandemic is given as 2 people, and the MAPE produced by the proposed model is 8.70% as given in Table 5.8.

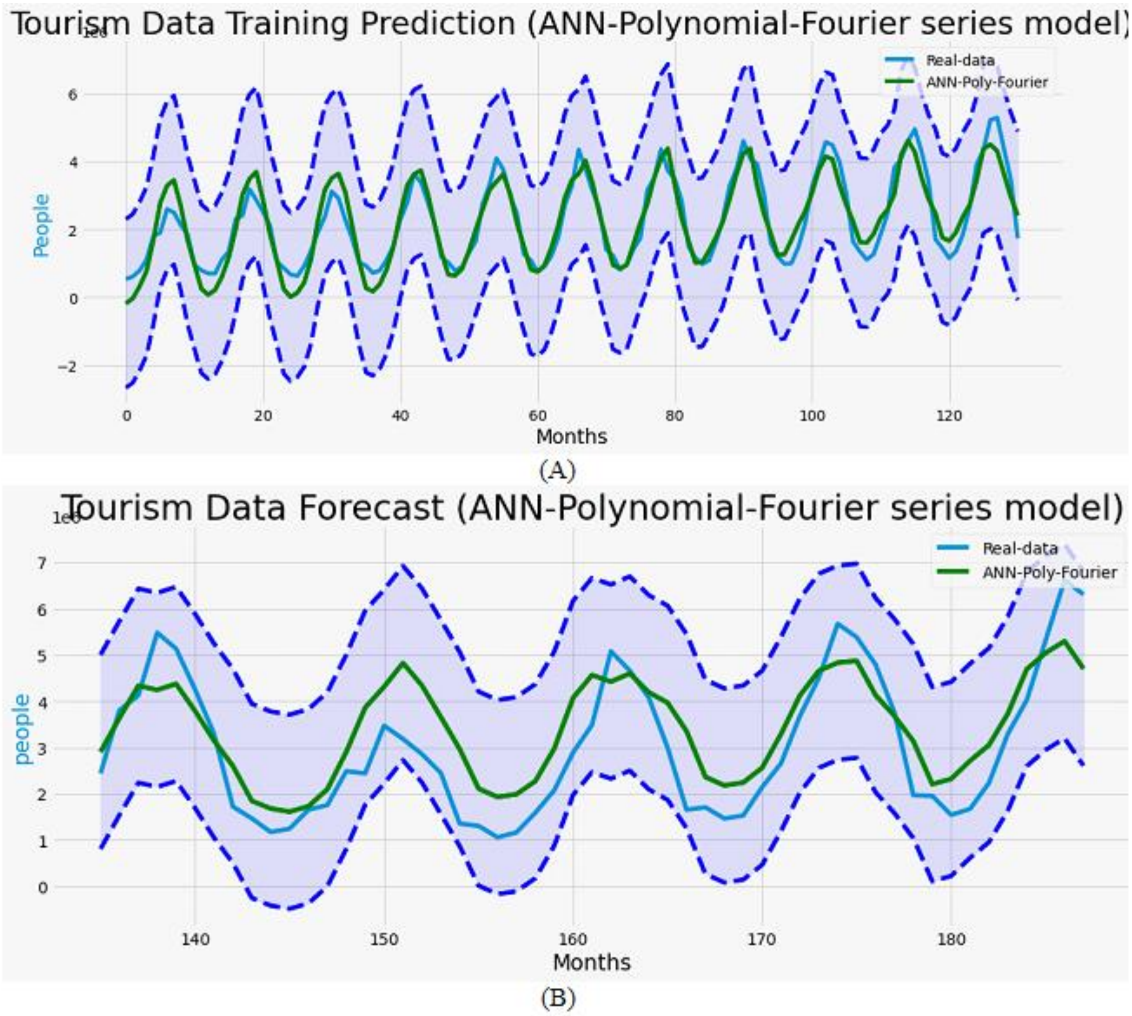


Figure 5.16. (A) Polynomial-Fourier series model training prediction before COVID-19 (B) Polynomial-Fourier series model forecast before COVID-19

5.1.15. Monte Carlo forecast result before COVID-19 for ANN-Polynomial-Fourier series model

The Monte Carlo simulation scheme is also applied to the result of the forecasting region before the pandemic obtained by the proposed ANN-Polynomial-Fourier series model. The numerical results obtained during the Monte Carlo simulation on the result of the proposed model are shown in Table 5.8. The findings show that the simulation scheme was able to increase the performance of the proposed model in the forecasting region before the pandemic. It was able to achieve a 95% R^2 estimate value from the previous 94% R^2 estimate value obtained by the proposed model. The RMSE of 2 people and MAPE of 7.31% obtained during the Monte Carlo simulation scheme are also slightly better than the model. Figure 5.17 (A) and (B) respectively depicts the mean of the Monte Carlo simulation and the 100 Monte Carlo simulation paths obtained during the Monte Carlo simulation scheme on the forecast region before the pandemic of the proposed ANN-Polynomial-Fourier series model.

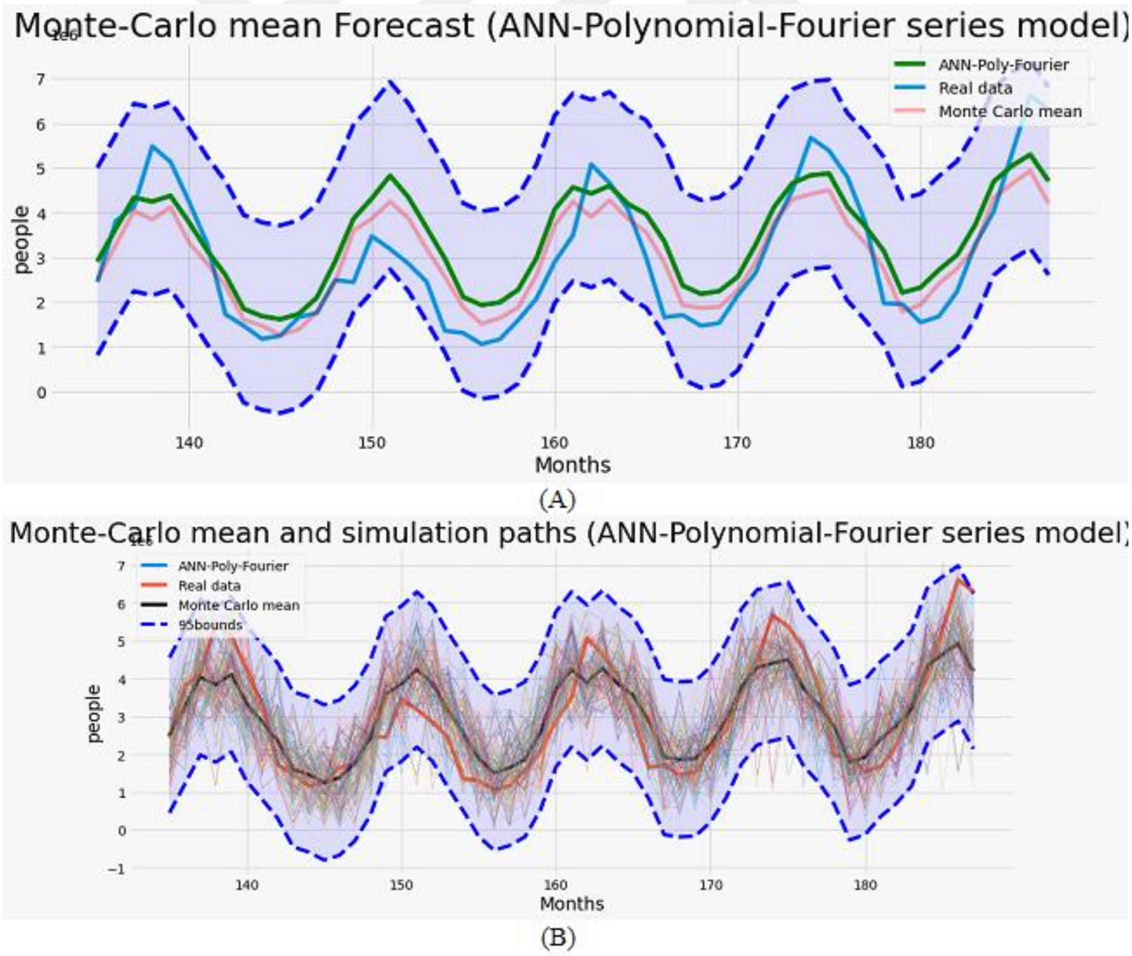


Figure 5.17. (A) Monte Carlo mean forecast for the ANN-Polynomial-Fourier series model before the COVID-19 (B) 100 Monte Carlo simulation paths for the ANN-Polynomial-Fourier series model before the COVID-19

5.1.16. ANN-Polynomial-Fourier series model forecast results during COVID-19 pandemic

In the training period of the region during the pandemic, the proposed ANN-Polynomial-Fourier model accounts for 97% variation in the tourism data as shown by the R^2 estimate value given in Table 5.9. The respective RMSE and MAPE performance of the proposed ANN-Polynomial-Fourier series model in this training period of the region during the pandemic is 2 people for RMSE and 17.53% for MAPE as given in Table 5.9.

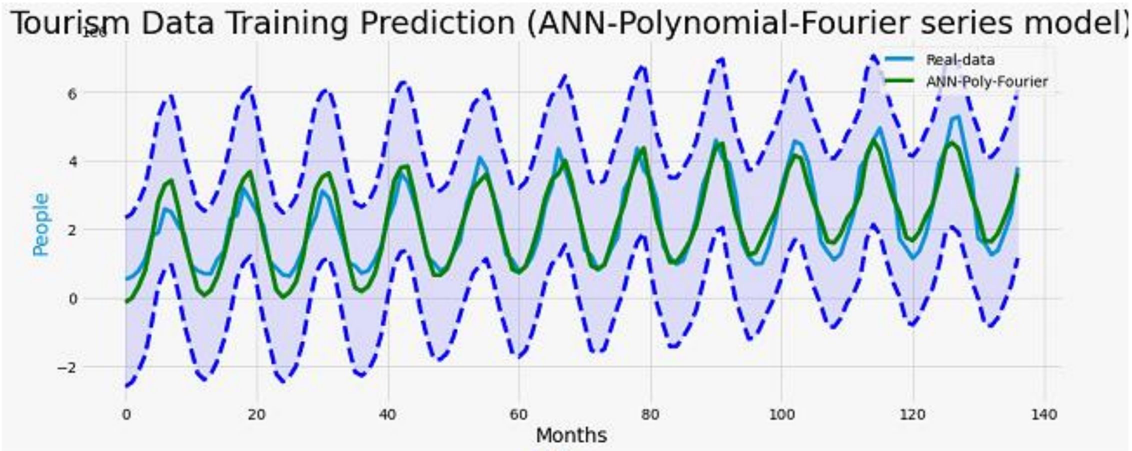
Table 5.9. Simulation and ANN-Polynomial-Fourier series model during COVID-19

	Number of simulations	**R^2**	**RMSE**	**MAPE**
ANN-Polynomial-Fourier series training prediction	1	0.97	2.06	07.13
ANN-Polynomial-Fourier series forecast	1	0.92	3.11	09.67
Monte Carlo mean forecast	100	0.93	2.71	09.08

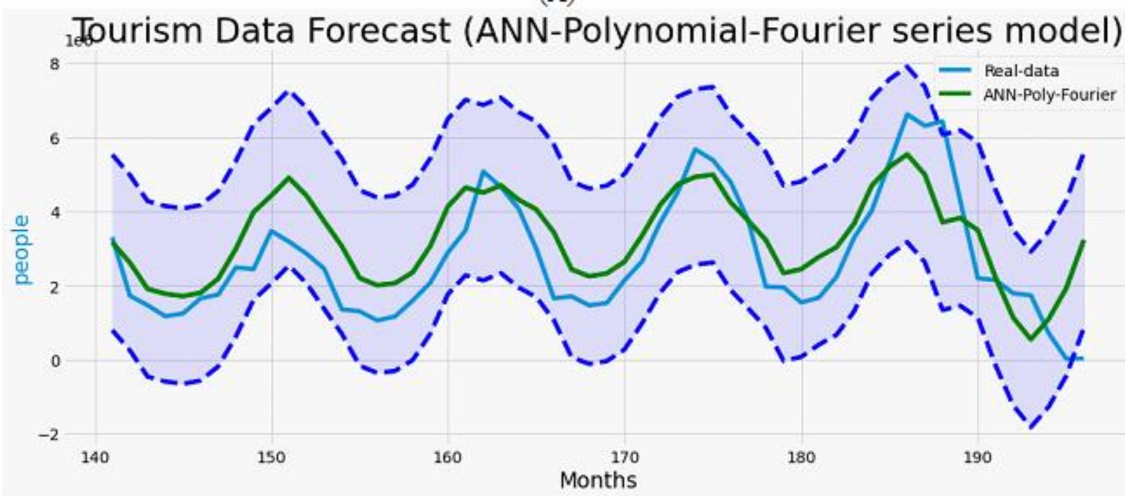
The proposed ANN-Polynomial-Fourier series was able to account for 92% variation in the data during the pandemic period, which is shown in the R^2 estimate value shown in the numerical values presented in Table 5.9. The model was also able to achieve prediction error and accuracy of RMSE of 3 people and MAPE of 9.67% respectively as per the results given in Table 5.9. The model's training prediction and forecast graphical results are shown in Figure 5.18 (A) and (B) respectively.

The proposed Polynomial-Fourier series has shown a statistically decent result in the forecasting region of the tourism data during the COVID-19 pandemic when compared to the benchmarked literature models of ARIMA and Fourier series models. However, the proposed ANN-Polynomial-Fourier series model outperforms all the three models in both the forecasting regions before and during the pandemic. It is the same for the training prediction; the proposed ANN-Polynomial-Fourier series model has outperformed the three models in the training period for both the period before and during the pandemic.

The good performance of the ANN-Polynomial-Fourier series model has shown that the empirical machine learning (the artificial neural network) component of the model has improved the result of the classical regression component of the polynomial and the Fourier series models. This shows that the suggestion of Makridakis et al. [88] can be applied in choosing this model for the tourism data as the ANN is improving the result of the classical method. However, the proposed Polynomial-Fourier series model can also be applied to this data since it has demonstrated a decent performance in terms of all the evaluation metrics used in this study.



(A)



(B)

Figure 5.18. (A) ANN-Polynomial-Fourier series model training prediction during COVID-19 (B) ANN-Polynomial-Fourier series model forecast during COVID-19

5.1.17. Monte Carlo forecast result during COVID-19 for ANN-Polynomial-Fourier series model

The Monte Carlo simulation scheme applied to the proposed ANN-Polynomial-Fourier series model in the forecasting region during the COVID-19 pandemic was able to achieve an R^2 estimate value of 93% as per the given numerical values in Table 5.9. The simulation also gives a prediction error of RMSE of 2 people and forecast accuracy of MAPE of 9.08% as given in Table 5.9. It can also be seen from Table 5.9 that the Monte Carlo simulation scheme was able to show slightly better performance when compared to the model forecast results. The mean of the Monte Carlo simulation scheme and the 100 Monte Carlo simulation paths are respectively shown in Figure 5.19 (A) and (B).

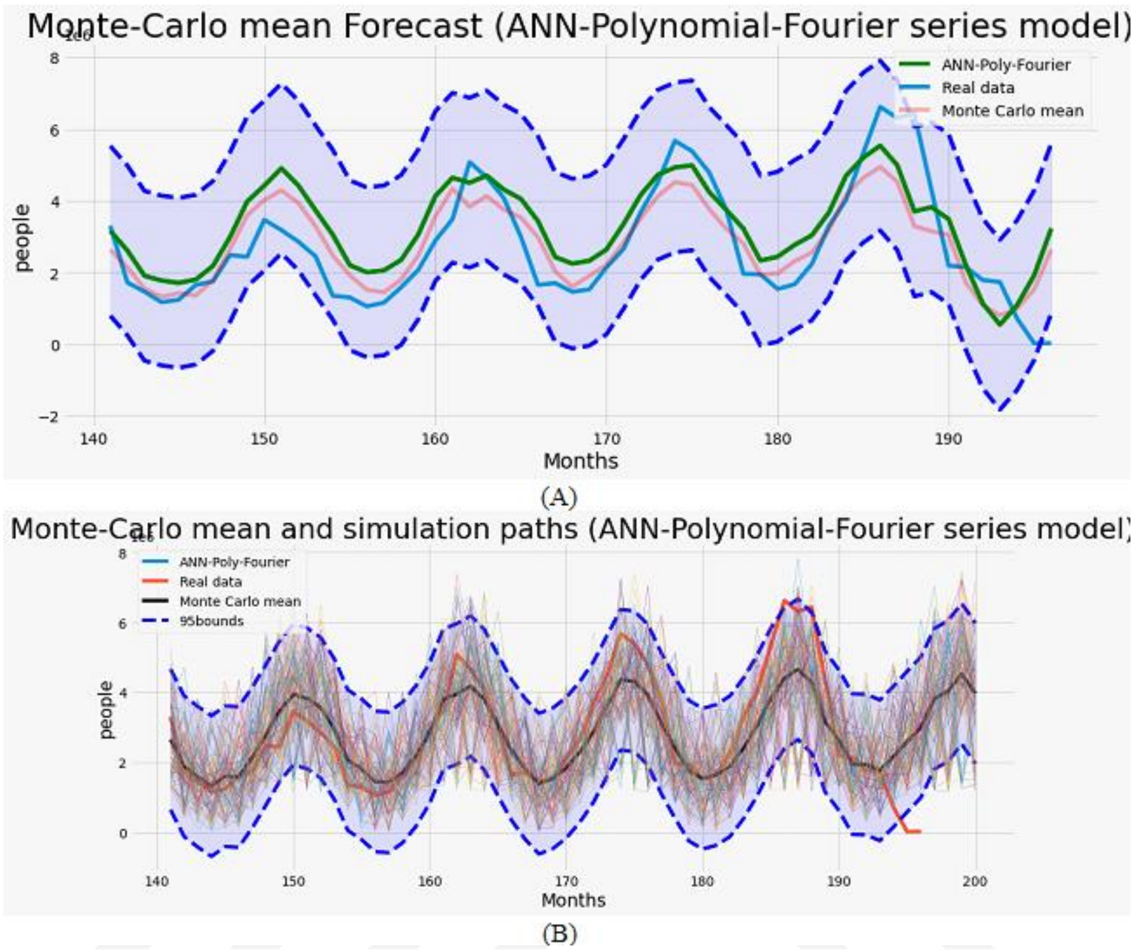


Figure 5.19. (A) Monte Carlo mean forecast for the ANN-Polynomial-Fourier series model during the COVID-19 (B) 100 Monte Carlo simulation paths for the ANN-Polynomial-Fourier series model during the COVID-19

5.1.18. ANN-Fourier series model forecast results before and during the COVID-19 pandemic

The values given in Table 5.10 show that in the forecasting region before the pandemic, the ANN-Fourier series was able to account for up to 93% variation in the data (see the R^2 value in Table 5.10). The values also show that the proposed model was able to achieve an RMSE of 4 people and MAPE of 9.41%.

Table 5.10. Simulation and ANN-Fourier series model result before COVID-19

	Number of simulations	**R^2**	**RMSE**	**MAPE**
ANN forecast on Fourier series residue	1	0.97	2.53	07.21
ANN-Fourier series forecast	1	0.93	4.02	09.41

In the forecasting region during the COVID-19 pandemic, the values presented in Table 5.11 show that the ANN-Fourier series was able to have a respective R^2, RMSE, and an MAPE values of 90%, 5 people, and 10.46%. The Monte Carlo simulation was able to add to the ANN-Fourier series performance by optimizing the result with an error of RMSE of 4 people and forecast accuracy of MAPE of 09.71% with the simulation accounting for 93% R^2 estimate value.

Table 5.11. Simulation and ANN-Fourier series model result during COVID-19

	Number of simulations	R^2	RMSE	MAPE
ANN forecast on Fourier series residue	1	0.94	3.53	08.79
ANN-Fourier series forecast	1	0.90	5.45	10.46
Monte Carlo mean forecast	100	0.93	4.39	09.71

The 100 Monte Carlo simulation scheme paths are shown in Figure 5.20 (A). The mean of the Monte Carlo simulation paths is used for presenting the simulation result in Figure 5.20 (A). The mean of the simulation paths can be seen to graphically resemble that of the proposed ANN-Fourier series model. Moreover, with the mean used for presenting the Monte Carlo simulation paths, it is easier to understand the simulation behavior within the forecast region as seen in Figure 5.20 (B).

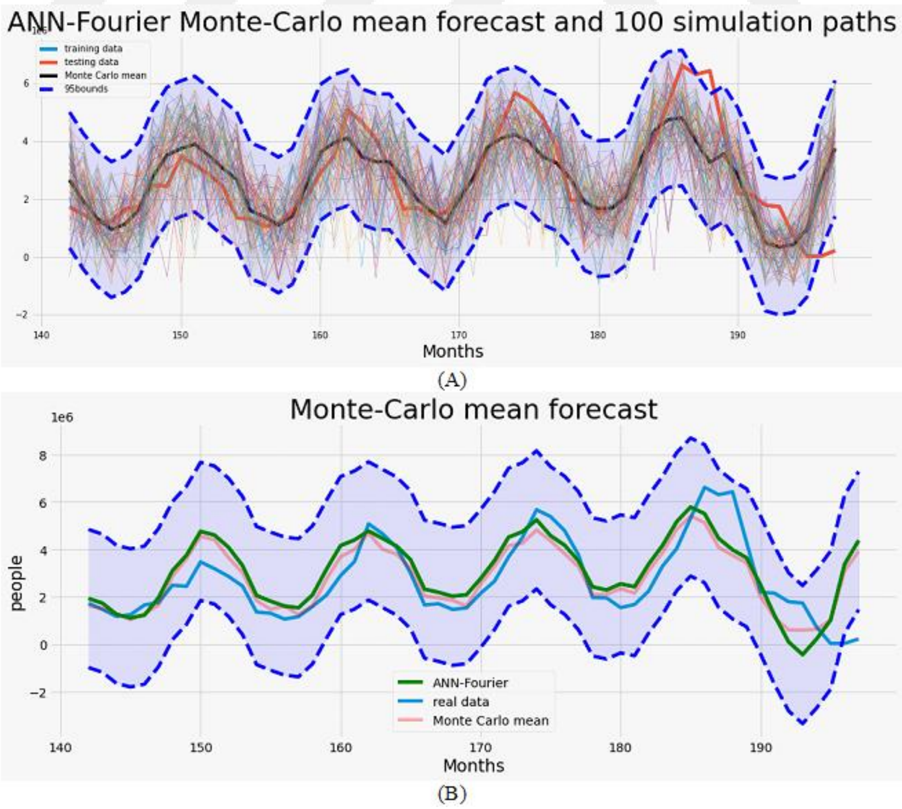


Figure 5.20. (A) 100 Monte Carlo simulation paths forecast during COVID-19 (B) Monte Carlo mean forecast

5.2. Residue analysis

The residue of the Fourier series model and the Polynomial-Fourier series model is evaluated to demonstrate the individual models' capabilities. The residue of the ARIMA model is not considered for two reasons, i.e. the ARIMA model work with stationary data, and the residue of the ARIMA model is not used for generating any of the other three models used in this study. Rather, it was only used for running the Monte Carlo simulation scheme.

5.2.1. Fourier series model residue

The residue plot diagram in Figure 5.21 (A) and (B) respectively depicts the Fourier series model’s residue before and during the COVID-19 pandemic periods. We can see from the figure that the model was able to capture the seasonal component of the tourism data, however, the Fourier series model fails to capture the trend component of the data. This shows that the Fourier series model is an appropriate model that can be applied to purely seasonal data as demonstrated in the work of Borowik et al. [101], Kamir et al. [102], and Chen et al. [103]. Similarly, the upward trend in Figures 5.21 (A) and (B) shows that the data have a trend that can only be captured using a special technique.

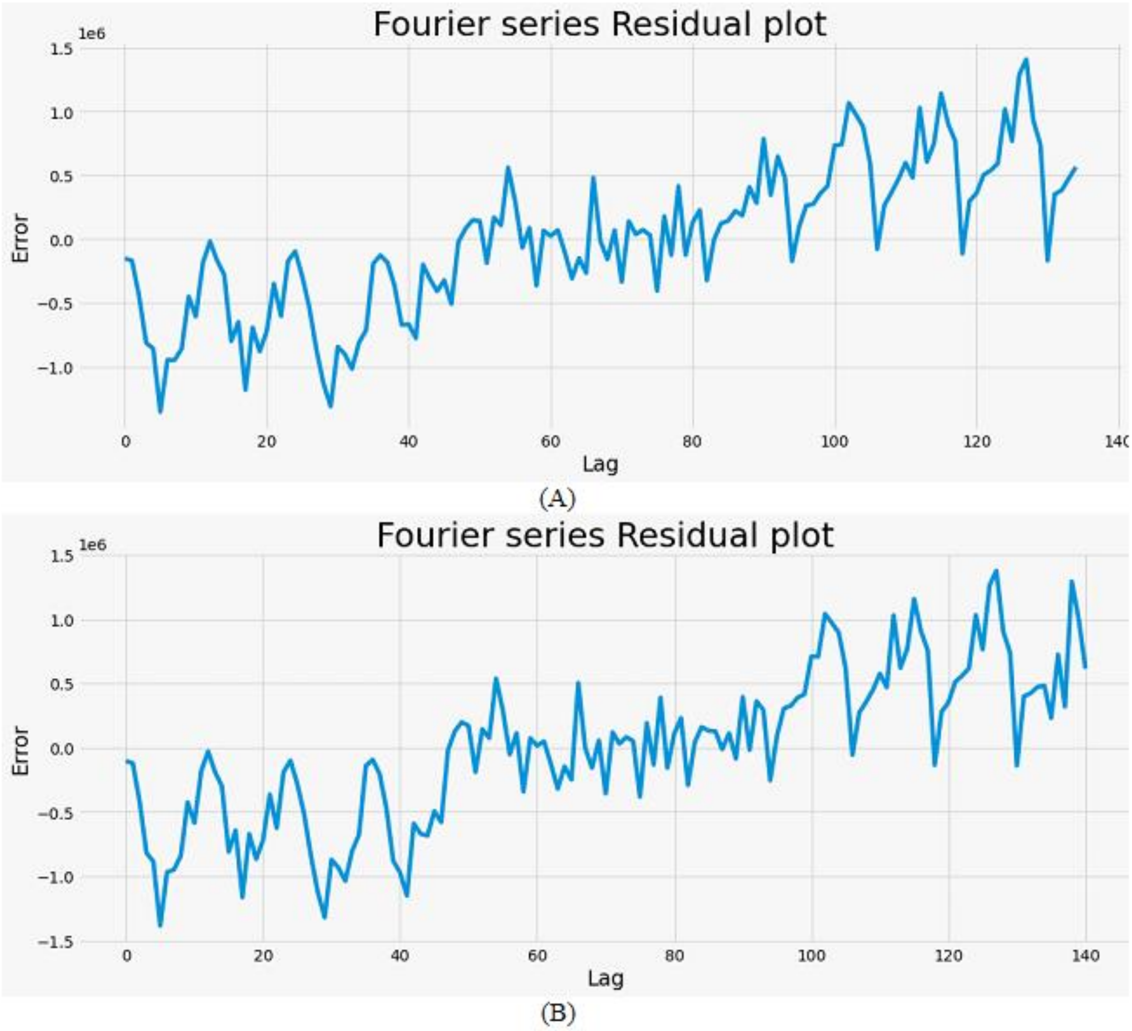
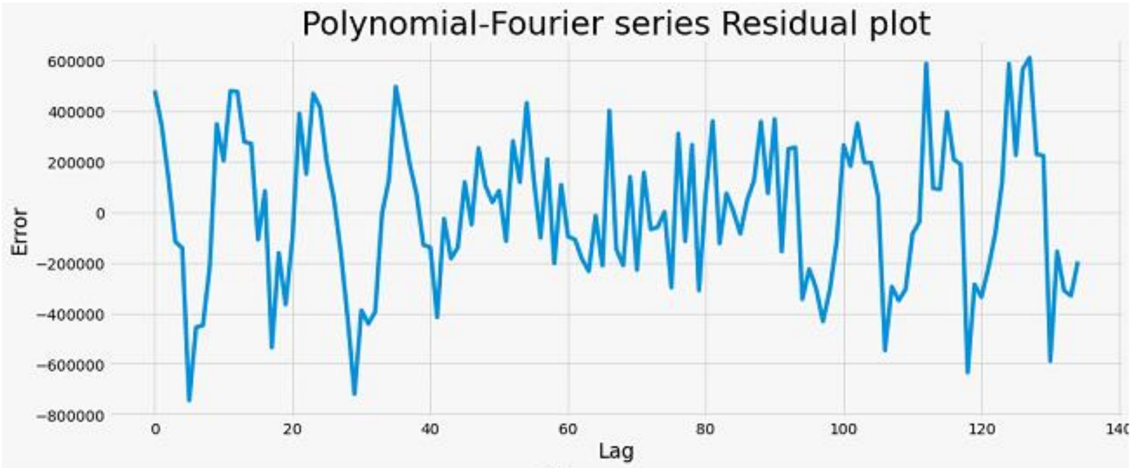


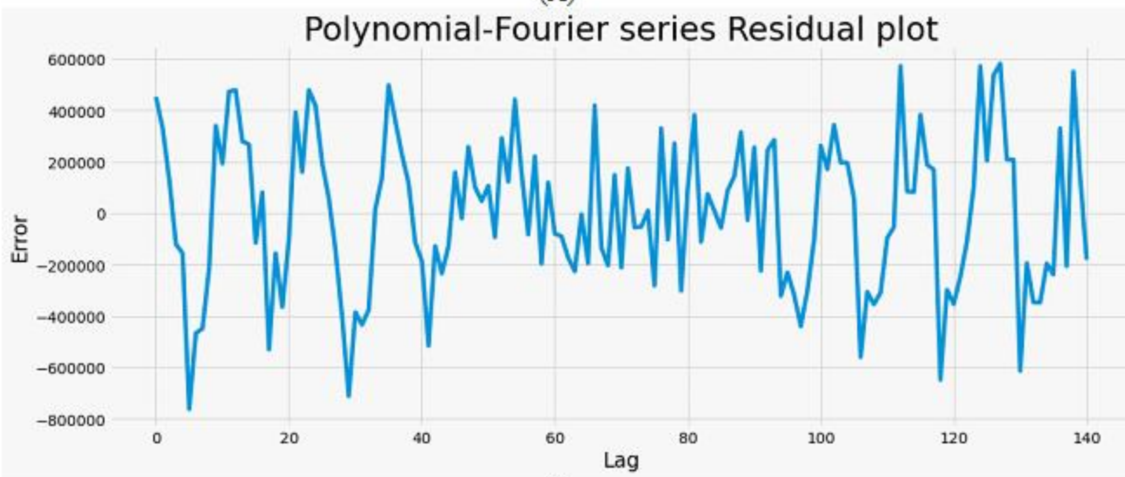
Figure 5.21. Fourier series model residue (A) before the COVID-19 pandemic (B) during the COVID-19 pandemic

5.2.2. Polynomial-Fourier series model residue

The Polynomial-Fourier series model residue plot shown in Figure 5.22 (A) and (B) respectively for the forecasting regions before and during the pandemic for the tourism data show that the proposed model was able to capture both the trend and seasonal components of the data. This shows that the special technique of using the polynomial terms to capture the trend component of the data, and the Fourier series terms to capture the seasonal component of the data used in this study is successful in addressing the trend and seasonal component of the data. The residue plot in Figure 5.22 shows that the residue is stable without having a trend or seasonal component unaccounted for in the data residue.



(A)



(B)

Figure 5.22. Fourier series model residue (A) before the COVID-19 pandemic (B) during the COVID-19 pandemic

5.3. Scatter and Q-Q plot

The proposed Polynomial-Fourier series has a person correlation of 91% for the tourism data training period, as shown in Figure 5.23 (A). In the forecasting period of the tourism data, the Polynomial-Fourier series model obtained a person correlation of 75% as given in Figure 5.23 (C), which demonstrates that there is a linear relationship between the tourism data and model. The Polynomial-Fourier series model residue for the training and forecast periods respectively shown in the Q-Q plot in Figure 2.23 (B) and (D) demonstrate the proposed model residue fits the theoretical normal distribution. This indicates the goodness of fit of the proposed Polynomial-Fourier series model considering there are only a few deviations at the tail.

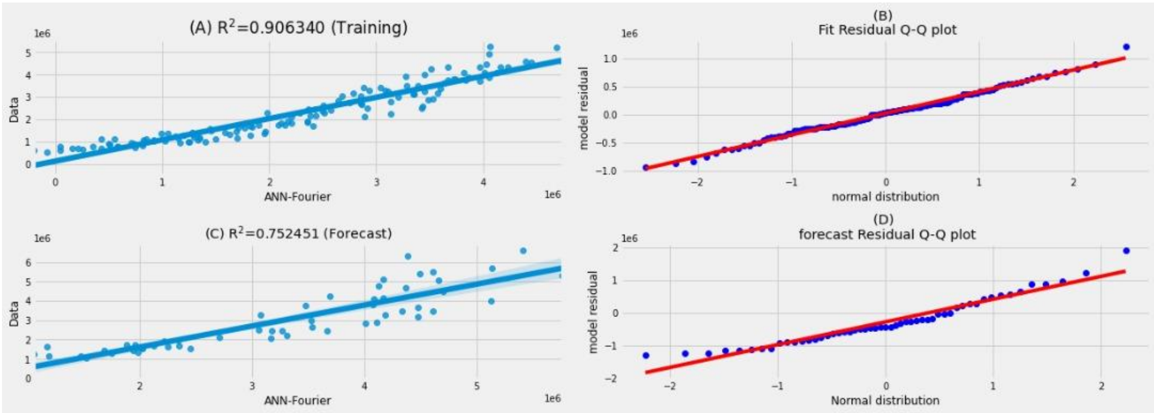


Figure 5.23. Scatter and Q-Q plots for the Polynomial-Fourier series model and tourism data

There is an existing linear relationship between the proposed ANN-Polynomial-Fourier series model and the tourism data. This is demonstrated in the training scatter plot in Figure 5.24 (A) and forecast scatter plot in Figure 5.24 (C), which show a person's correlation of 97% and 84% in the training and forecasting periods respectively. Moreover, with just a few deviations at the tails, the Q-Q plot shown in Figures 5.24 (B) and (D) also show that the proposed ANN-Polynomial-Fourier series fits the theoretical normal distribution, which implies goodness of fit of the proposed model.

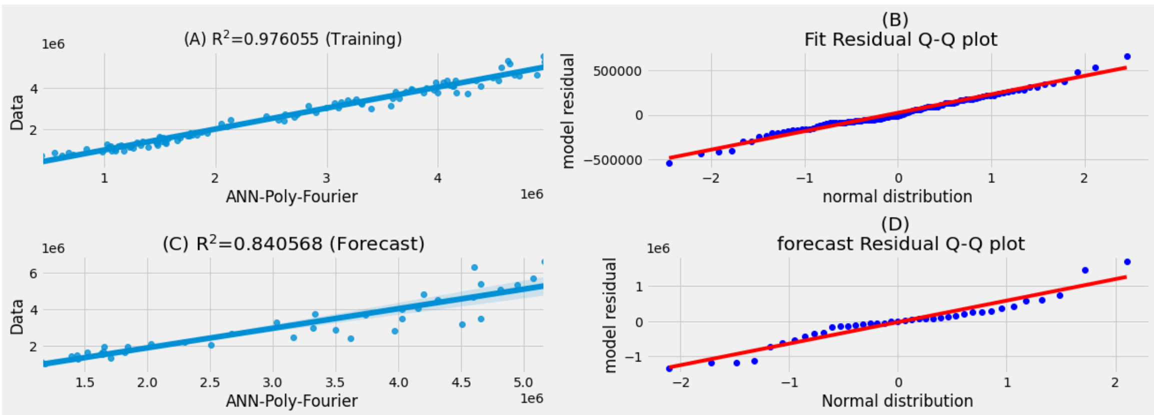


Figure 5.24. Scatter and Q-Q plots for the ANN-Polynomial-Fourier series model and tourism data

5.4. Temperature data forecast results

The benchmarked and proposed models of the study are also applied to the temperature data, and the numerical results of the models are presented in Table 5.12. In the training prediction period of the temperature data, the ARIMA model was able to obtain an R^2 value of 90% with respective prediction error and accuracy of RMSE of 2.55°c and MAPE of 8.08% (see Table 5.10). In the same training period, the Fourier series model training prediction results show that the model can account for 80% variation in the temperature data with RMSE of 5.33°c and MAPE of 11.39% according to the numerical values presented in Table 5.12.

Table 5.12. Simulation and models result for temperature data

	Number of simulations	**R^2**	**RMSE**	**MAPE**
ARIMA model training prediction	1	0.90	2.55	08.08
ARIMA model forecast	1	0.85	4.49	09.15
Monte Carlo mean forecast (ARIMA)	100	0.88	3.89	07.91
Fourier series model training prediction	1	0.80	5.33	11.39
Fourier series model forecast	1	0.73	7.12	13.73
Monte Carlo mean forecast (Fourier series)	100	0.75	6.06	12.11
Polynomial-Fourier series training prediction	1	0.92	2.04	07.40
Polynomial-Fourier series forecast	1	0.89	2.98	07.88
Monte Carlo mean forecast (Polynomial-Fourier series)	100	0.90	2.51	07.57
ANN-Polynomial-Fourier series training prediction	1	0.95	1.70	03.16
ANN-Polynomial-Fourier series forecast	1	0.90	2.72	05.46
Monte Carlo mean forecast (ANN-Polynomial-Fourier series)	100	0.92	3.01	06.65

The proposed Polynomial-Fourier series model was able to account for 92% variation in the temperature data during the training prediction period as given in the R^2 value in Table 5.10. The proposed model also produces an error of RMSE of 2.04°c and an accuracy of MAPE of 7.40%. During the same training prediction period, the proposed ANN-Polynomial-Fourier series model was able to achieve a 95% R^2 estimate value with RMSE of 1.70 people and MAPE of 3.16% as per the values given in Table 5.10.

In the forecasting region of the temperature data, the ARIMA model yields a decent result obtaining an R^2 estimate value of 85%, RMSE of 4.49°c, and MAPE of 9.15% according to the results given in Table 5.10. The Fourier series model produces less performance in the forecasting region of the temperature data when compared to the ARIMA model. It yields an R^2 estimate value of 73%, with forecast error and accuracy of RMSE of 7.12°c and MAPE of 13.73% as per the numerical results presented in Table 5.10.

The proposed Polynomial-Fourier series and ANN-Polynomial-Fourier series models were respectively able to account for 89% for the Polynomial-Fourier series model and 90% for ANN-Polynomial-Fourier series model variation in the temperature data as recorded in the R^2 estimate value in Table 5.10. The prediction error of the Polynomial-Fourier series model is RMSE of 2.98°c while the prediction error of the ANN-Polynomial-Fourier series model is also RMSE of 2.72°c (see Table 5.10). As for the prediction accuracy of the proposed models, the Polynomial-Fourier series model was able to show a prediction accuracy of MAPE of 7.88%. The ANN-Polynomial-Fourier series model gives prediction accuracy MAPE of 5.46% as given in Table 5.10.

The Monte Carlo simulation results for all models increased and optimized the model's forecast performance. This shows that the Monte Carlo simulation when applied to the results of the models used in this study can produce a more accurate performance. For example, the ARIMA model forecast R^2 estimate value result of 85% was increased to 88% by the Monte Carlo simulation scheme as shown in Table 5.10. Similarly, the simulation scheme reduced the ARIMA model forecast error from the model's RMSE of 4.49ºc to RMSE of 3.49ºc.

The prediction accuracy is also the same as the Monte Carlo simulation yields a MAPE of 7.91% from the model's MAPE of 9.15%. The Fourier series model result performance is also increased and optimized by the Monte Carlo simulation scheme in which the simulation was able to account for 75% variation in the data which is a 2% increase from the model's R^2 values of 73%. The Monte Carlo simulation also gives prediction error and accuracy of RMSE of 6.06ºc from the model's 7.12ºc and MAPE of 12.11% from the model's 13.73%.

The proposed model's result performance was also optimized by the Monte Carlo simulation scheme. The Polynomial-Fourier series model was able to account for 89% variation in the forecast region of the temperature data, while the Monte Carlo simulation scheme on the model result can account for 90% variation in the data. The simulation also produces an error of RMSE of 2 people from the model forecast error of RMSE of 2.51ºc. The forecast accuracy also increased, as the Monte Carlo scheme was able to yield a MAPE of 7.07% from the model MAPE of 7.88%.

The proposed ANN-Polynomial-Fourier series model achieves a 90% R2 estimate value improved by the Monte Carlo simulation to 92% R^2 estimate value. The Monte Carlo simulation scheme also improves the proposed model performance in terms of both prediction error and accuracy. The simulation show error of RMSE of 2.01ºc from the model's RMSE of 2.72ºc and an accuracy of MAPE of 4.65% from the model's MAPE of 5.46% as given in the numerical values presented in Table 5.10.

5.5. Results of the multi-step ahead forecasting

In the case of the tourism data, we demonstrate how the ANN-Polynomial Fourier series model results surpass the results of the other compared literature and proposed models. In this regard, we employed the multi-step ahead direct prediction algorithm to forecast the future number of foreign visitors from January to December 2021 using the result of the ANN-Polynomial Fourier series and Monte Carlo simulation. The 12 months multi-step ahead forecasting is not only applied to the Turkey data, it is also applied to the foreign visitor's data of Japan, Malaysia, and Singapore for the same period. The Monte Carlo forecast result for the 12 months multi-step ahead forecasting is presented using the mean of the simulations. Nevertheless, the year 2019 is considered as a threshold considering it is the tourism peak year at all the countries and the year that was not affected by the COVID-19 pandemic.

5.5.1. Multi-step ahead forecasts for Turkey

The result of the 12 months of 2020 shows that when compared to the threshold year (2019) there is a decrease of 71.74% in the number of foreign visitors to Turkey as given in Table 5.13. The results provided in Table 5.13 also show that the number of foreign visitors to Turkey in 2021 will increase by 10.22% when compared to 2020. Even with this forecast, the number of visitors will be decreased by 68.85% in 2021 if we compare the result to that of 2019 (see Table 5.13). Overall, the result shows that the year 2021 will be better than 2020 in terms of the number of foreign arrivals, which means that the country could see some positive changes in 2021 as opposed to the devastating outcome seen in 2020.

Table 5.13. Multi-step ahead modeling results for the number of monthly foreign visitors to Turkey

Month	2019	2020	2021 (Monte Carlo mean)	% Rate of change 2019/2020	2019/2021 (Monte Carlo mean)	2020/2021 (Monte Carlo mean)
Jan	1 539 496	1 787 435	509 787	16.11	-66.89	-71.48
Feb	1 670 238	1 733 112	937 343	3.76	-43.88	-45.92
Mar	2 232 358	718 097	812 326	-67.83	-63.61	13.12
Apr	3 293 176	24 238	258 471	-99.26	-92.15	966.39
May	4 022 254	29 829	311 578	-99.26	-92.25	944.55
Jun	5 318 984	214 768	858 255	-95.96	-83.86	299.62
Jul	6 617 380	932 927	1 540 879	-85.90	-76.71	65.17
Aug	6 307 508	1 814 701	2 392 081	-71.23	-62.08	31.82
Sep	5 426 818	2 203 482	2 552 569	-59.40	-52.96	15.84
Oct	4 291 574	1 742 303	1 907 339	-59.40	-55.56	9.47
Nov	2 190 622	833 991	1 002 081	-61.93	-54.26	20.15
Dec	2 147 878	699 330	952 569	-67.44	-55.65	36.21
Total	**45 058 286**	**12 734 213**	**14 035 278**	**-71.74**	**-68.85**	**10.22**

5.5.2. Multi-step ahead forecasts for Japan

The obtained values given in Table 5.14 for the 12 month multi-step ahead forecasting for the year 2021 show that Japan will expect a total of 311,781 arrivals. This number shows that compared to the 4,115,828 number of arrivals in 2020 (see Table 5.14), the country will continue to decrease the number of foreign visitors arrivals in the year 2021 by about 92.42%. The number of arrivals in 2021 will also be decreased by 99.02% when compared to the threshold of the 2019 year. Likewise, the foreign visitor's arrivals in 2020 have shown a decrease of 87.09% when compared to the 2019 arrivals as given in Table 5.14.

Table 5.14. Multi-step ahead modeling results for the number of monthly foreign visitors to Japan

				% Rate of change		
Month	2019	2020	2021 (Monte Carlo mean)	2019/2020	2019/2021 (Monte Carlo mean)	2020/2021 (Monte Carlo mean)
Jan	2 689 339	2 661 022	48 900	-01.05	-98.18	-98.16
Feb	2 604 322	1 085 147	10 489	-58.33	-99.59	-99.03
Mar	2 760 136	193 658	14 200	-99.49	-63.61	-92.67
Apr	2 926 685	2 917	10 356	-99.90	-99.64	255.02
May	2 773 091	1 663	4 630	-99.94	-99.83	178.41
Jun	2 880 041	2 565	6 542	-99.91	-99.77	155.05
Jul	2 991 189	3 782	7 622	-99.87	-99.75	101.53
Aug	2 520 134	8 658	13 706	-99.66	-99.46	58.30
Sep	2 272 883	13 684	22 719	-99.40	-99.04	66.03
Oct	2 496 568	27 386	41 368	-98.90	-98.34	51.05
Nov	2 441 274	56 673	65 028	-97.68	-97.33	14.74
Dec	2 526 387	58 673	66 221	-97.68	-97.37	12.86
Total	**31 882 049**	**4 115 828**	**311 781**	**-87.09**	**-99.02**	**-92.42**

5.5.3. Multi-step ahead forecasts for Malaysia

In the case of Malaysia, there were 26,100,784 visitors in 2019 and 4,332,722 visitors in 2020. The 12-month multi-ahead forecasts for 2021 shows that the country will expect 1,958,029 visitors in 2021 (see Table 5.15). The numbers show that in 2020 there was a decrease of 83.4% in the number of arrivals when compared to 2019. In 2021, the forecasts indicate that the country will continue to see a decrease in the number of foreign visitors of about 92.4% compared to 2019, and 54.81% compared to 2020 as per the given values in Table 5.15.

Table 5.15. Multi-step ahead results for the number of monthly foreign visitors to Malaysia

Month	2019	2020	2021 (Monte Carlo mean)	% Rate of change 2019/2020	2019/2021 (Monte Carlo mean)	2020/2021 (Monte Carlo mean)
Jan	2 195 684	2 164 459	481 483	-1.42	-78.07	-77.76
Feb	2 165 933	1 397 912	570 505	-35.46	-73.66	-59.19
Mar	2 334 613	671 084	748 186	-71.26	-67.95	11.49
Apr	2 159 517	7 546	9 130	-99.58	-99.58	20.99
May	2 098 267	5 411	9 275	-99.74	-99.56	71.41
Jun	2 400 561	6 585	8 352	-99.73	-99.65	26.83
Jul	2 415 097	18 660	19 701	-99.18	-99.18	5.58
Aug	2 342 438	11 631	19 819	-99.50	-99.15	70.40
Sep	1 997 093	16 131	30 772	-99.19	-98.46	90.76
Oct	2 031 198	11 315	21 409	-99.44	-98.95	89.21
Nov	1 969 315	11 420	20 635	-99.42	-98.95	80.69
Dec	1 991 068	10 568	18 762	-99.47	-99.06	77.54
Total	**26 100 784**	**4 332 722**	**1 958 029**	**-83.40**	**-92.49**	**-54.81**

5.5.4. Multi-step ahead forecasts for Singapore

The values in Table 5.16 show that with an expected 807.708 number of foreign visitors to Singapore, the country's arrivals will decrease by about 70.55% and 95.77% compared to 2020 and 2019 respectively. The numbers presented in Table 5.16 also show a decrease of 85.65% in 2020 compared to the number of arrivals in 2019.

Table 5.16. Multi-step ahead modeling results for the number of monthly foreign visitors to Singapore

Month	2019	2020	2021 (Monte Carlo mean)	% Rate of change 2019/2020	2019/2021 (Monte Carlo mean)	2020/2021 (Monte Carlo mean)
Jan	1 624 593	1 688 099	78 265	3.91	-95.18	-95.36
Feb	1 499 051	732 965	45 198	-51.10	-96.98	-93.83
Mar	1 564 664	240 001	405 329	-84.66	-74.09	68.89
Apr	1 596 724	750	25 920	-99.95	-98.38	3356.00
May	1 487 791	880	29 467	-99.94	-98.02	3248.52
Jun	1 551 847	2 171	30 175	-99.86	-98.06	1289.91
Jul	1 802 593	6 843	31 835	-99.62	-98.23	365.22
Aug	1 736 102	8 912	28 573	-99.49	-98.35	220.61
Sep	1 463 543	9 500	22 842	-99.35	-98.44	140.44
Oct	1 530 479	13 397	35 724	-99.12	-97.67	166.66
Nov	1 533 668	14 676	37 246	-99.04	-97.57	153.79
Dec	1 724 961	24 014	37 134	-98.61	-97.85	54.63
Total	**19 116 016**	**2 742 208**	**807 708**	**-85.65**	**-95.77**	**-70.55**

6. CONCLUSION

Time series modeling and forecasting can be affected by the series trend and seasonal components. Being an important practice, many organizations are keen on models that would provide them with an insight into the inherent dynamics of their series data. Especially in forecasting the future values of the series, it is a tool that will greatly assist them in decision-making. We present three hybrid models that can capture and describe a time-series data trend and seasonal components reasonably. These hybrid models are the Polynomial-Fourier series, ANN-Polynomial-Fourier series, and ANN-Fourier series, models.

In the proposed Polynomial-Fourier series and ANN-Polynomial-Fourier series models, the linear component of the series is captured using the first polynomial term ($a_1\, x$). Likewise, the second polynomial term ($a_2\, x^2$) serves as the nonlinear gauge function that regulates the series trend and seasonality. The Fourier series part used in all three models uses the series frequency and equation 2.5 to capture the seasonal component of the series. In the case of the Polynomial-Fourier series model, the model is the summation of the linear, nonlinear, and seasonal components that are captured using the polynomial and the Fourier series respectively.

The ANN-Polynomial-Fourier series model is obtained by collecting the residue of the Polynomial-Fourier series model and modeling it using the ANN model. Hence, the summation of the ANN modeled residue and the Polynomial-Fourier series model gives the proposed ANN-Polynomial-Fourier series model. However, in the case of the ANN-Fourier series model, only the residue of the Fourier model is collected and modeled using the ANN model. Consequently, the ANN modeled residue and the Fourier series model are put together to yield the proposed ANN-Fourier series model.

In addition to the modeling process, the residue produced by each model (including the ARIMA and Fourier series models from literature) is collected for the Monte Carlo simulation scheme. Therefore; to forecast, optimize, and capture the future values and anomalies of the series, a 100 optimal Monte Carlo simulation scheme is carried out within 95% bounds of the proposed and literature models bounds. The goal here is to account for uncertainty in estimations as well as to simulate all possible future values of the series in question. In the Monte Carlo simulation mechanism, we randomly get the sample from our proposed model's training residues and then add it to our model forecast paths. Doing this means that we will get a distinct outcome that is generated for each simulation. Since we use the 95% regression curve, we reject any Monte Carlo generated path that crosses the prediction bounds.

The mean of the 100 Monte Carlo simulation paths is used for presenting the Mote Carlo model forecast results. Multiple paths were generated, and the mean of the paths was able to appear similar to the anomalous change that occurred in all the time series data used in this study.

Considering the Monte Carlo forecast reasonably captured the data, we used it to apply a multi-step ahead forecast for the tourism time series data of Turkey, Japan, Malaysia, and Singapore to forecast the future expected number of visitors' arrivals for 2021. The practice will make it feasible to study and access multiple future outcomes of the time series data. Nevertheless, the statistically acceptable results yielded by the model make it a worthy candidate for time series modeling and forecasting. The Monte Carlo simulation method was very useful in predicting uncertain forecast outcomes.

6.1. Highlights of the study

This study highlights the significance of considering both the trend and seasonal components of time series data during modeling and forecasting. It also highlights the significance of modeling residuals from classical/empirical models to improve forecast performance, specifically;

- Seasonal time series data may contain linear and or nonlinear trends
- Fourier series fitting can be used to capture and forecast any periodic or seasonal time-series data
- The polynomial fitting can be used to capture and forecast the trend component of periodic or seasonal time-series data
- Monte Carlo simulation scheme can be used to capture any data anomalies including nonlinear trends
- Classical method results should be evaluated and used as a baseline when evaluating machine learning methods

6.2. Publications generated from this study

Four publications are generated from this study, which includes two publications in two international refereed journals and two publications in an international conference proceeding. The Polynomial-Fourier series model algorithm is described in the study of Danbatta and Varol [86] titled "Monte Carlo Forecasting of Time-series Data using Polynomial-Fourier series model" in the International Journal of Modeling, Simulation, and Scientific Computing. The ANN-Polynomial-Fourier series algorithm is described in our ANN-Polynomial-Fourier series Modeling and Monte Carlo simulation paper that was published in the study of Danbatta and Varol [87] in the Journal of Forecasting.

The performances of the two algorithms were compared to that the ARIMA and Fourier series from the literature as well as that of Shabri et al. [104], Silva et al. [105], Álvarez et al. [106], Kulendran et al. [107], and within the proposed models. The ANN-Fourier series model's algorithm was discussed in Danbatta and Varol [92], an IEEE conference proceeding paper presented in the

9th International Symposium on Digital Forensics and Security (ISDFS). The title of the paper is "Modeling and Forecasting of Tourism Time-series Data using ANN-Fourier Series Model and Monte Carlo Simulation". Another IEEE conference proceeding that was generated from the study and published in the 8th ISDFS conference is the work of Danbatta and Varol [108] titled "Predicting Student's Final Graduation CGPA Using Data Mining and Regression Methods: A Case Study of Kano Informatics Institute". The paper describes the general and applies the theoretical concepts of data mining and regression modeling used in this study. In summary, this thesis work generated four main papers, and the summary of the publications are provided in Table 6.1.

Table 6.1. Summary of publications generated from the thesis study

SN	Paper title	Paper type	Journal/Conference	To be cited as
1	ANN-Polynomial-Fourier series Modeling and Monte Carlo Forecasting of Tourism Data	Journal	Journal of Forecasting	Danbatta, S. J., & Varol, A. (2022). ANN–polynomial–Fourier series modeling and Monte Carlo forecasting of tourism data. Journal of Forecasting, (October), 1–13. https://doi.org/10.1002/for.2845
2	Monte Carlo Forecasting of Time Series Data Using Polynomial-Fourier Series Model	Journal	International Journal of Modelling, Simulation, and Scientific Computing	Danbatta, S. J., & Varol, A. (2021). Monte Carlo forecasting of time series data using Polynomial-Fourier series model. International Journal of Modeling, Simulation, and Scientific Computing, 12(03), 2141004. https://doi.org/10.1142/S179396232141004X
3	Modeling and Forecasting of Tourism Time Series Data using ANN-Fourier Series Modeland Monte Carlo Simulation	Conference	9th International Symposium on Digital Forensics and Security	Danbatta, S. J., & Varol, A. (2021a). Modeling and Forecasting of Tourism Time Series Data using ANN-Fourier Series Model and Monte Carlo Simulation. In 2021 9th International Symposium on Digital Forensics and Security (ISDFS) (pp. 1–6). IEEE. https://doi.org/10.1109/ISDFS52919.2021.9486325
4	Predicting Student's Final Graduation CGPA Using Data Mining and Regression Methods: A Case Study of Kano Informatics Institute	Conference	8th International Symposium on Digital Forensics and Security	Danbatta, S. J., & Varol, A. (2020). Predicting Student's Final Graduation CGPA Using Data Mining and Regression Methods: A Case Study of Kano Informatics Institute. In 2020 8th International Symposium on Digital Forensics and Security (ISDFS) (pp. 1–7). IEEE. https://doi.org/10.1109/ISDFS49300.2020.9116389

RECOMMENDATIONS

With the high global demand for time series data, modeling and forecasting time series data has become necessary. One of the major challenges in forecasting some time-series data such as the tourism data that we used in this study is the uncertainties in estimation. This is mostly due to the current ongoing COVID pandemic that is resulting in different policies by different governments and organizations around the globe. Our models' results show that the method can be regarded as a robust method for studying such time-series data. Therefore, policymakers can consider the models' results in guiding their decision.

Future research may use the method while considering other related attributes such as airline tickets reservations, hotels reservations, etc. Moreover, environmental factors such as special seasons, festivals, and COVID-19 (confirmed cases and asymptomatic cases) can be combined in predicting the future number of visitors for a specific country. In addition to this, scenario forecasting and judgmental forecasting can be incorporated into the forecast exercise.

REFERENCES

[1] N. K. Ahmed, A. F. Atiya, N. El Gayar, and H. El-Shishiny, "An empirical comparison of machine learning models for time series forecasting," *Econom. Rev.*, vol. 29, no. 5, pp. 594–621, 2010.

[2] S. T. Li, S. C. Kuo, Y. C. Cheng, and C. C. Chen, "Deterministic vector long-term forecasting for fuzzy time series," *Fuzzy Sets Syst.*, vol. 161, no. 13, pp. 1852–1870, 2010.

[3] G. Krempl *et al.*, "Open challenges for data stream mining research," *ACM SIGKDD Explor. Newsl.*, vol. 16, no. 1, pp. 1–10, 2014.

[4] D. Rapach and G. Zhou, "Time-Series and Cross-Sectional Stock Return Forecasting: New Machine Learning Methods," *SSRN Electron. J.*, pp. 1–33, 2019.

[5] Rob J Hyndman, "Forecasting: Forecasting: Principles & Practice," no. September, p. 138, 2014.

[6] R. J. Hyndman and G. Athanasopoulos, *Forecasting: Principles and Practice (2nd ed)*. 2018.

[7] C. Bergmeir, R. J. Hyndman, and B. Koo, "A note on the validity of cross-validation for evaluating autoregressive time series prediction," *Comput. Stat. Data Anal.*, vol. 120, pp. 70–83, 2018.

[8] G. Athanasopoulos, R. A. Ahmed, and R. J. Hyndman, "Hierarchical forecasts for Australian domestic tourism," *Int. J. Forecast.*, vol. 25, no. 1, pp. 146–166, 2009.

[9] X. Li and R. Law, "Forecasting Tourism Demand with Decomposed Search Cycles," *J. Travel Res.*, vol. 59, no. 1, pp. 52–68, 2020.

[10] R. B. Cleveland, W. S. Cleveland, J. E. McRae, and I. Terpenning, "stl-a-seasonal-trend-decomposition-procedure-based-on-loess.pdf," *J. Off. Stat.*, vol. 6, no. 1, pp. 3–73, 1990.

[11] E. Bee Dagum and S. Bianconcini, "Time Series Components," pp. 29–57, 2016.

[12] R. Di. T. De Lima, S. M. M. I. Fernandes, and I. P. L. Melo, "Fourier Genetic Series: An Evolutionary Time Series Modeling Technique," *2019 IEEE Lat. Am. Conf. Comput. Intell. LA-CCI 2019*, 2019.

[13] Y. Peng and M. H. Nagata, "An empirical overview of nonlinearity and overfitting in machine learning using COVID-19 data," *Chaos, Solitons and Fractals*, vol. 139, 2020.

[14] J. Subramanian and R. Simon, "Overfitting in prediction models - Is it a problem only in high dimensions?," *Contemp. Clin. Trials*, vol. 36, no. 2, pp. 636–641, 2013.

[15] C. Ordonez and J. Garcia-Garcia, "Managing Big Data Analytics Workflows with a Database System," *Proc. - 2016 16th IEEE/ACM Int. Symp. Clust. Cloud, Grid Comput. CCGrid 2016*, pp. 649–655, 2016.

[16] N. Zhang and W. F. Lu, "An efficient data preprocessing method for mining customer survey data," *IEEE Int. Conf. Ind. Informatics*, vol. 1, pp. 573–578, 2007.

[17] B. Gopalakrishnan, R. Peters, and D. Vanzetti, "Covid-19 and Tourism," in *Economic Consequences (United Nations Conference on Trade and Development)*, 2020, no. July, pp. 1–27.

[18] G. Cetin, "Impact of Covid-19 on Tourism Industry," *J. Tour.*, no. April, pp. 1–3, 2020.

[19] A. Kulkarni and R. Booz, "Time-Series Database Analysis & Data Modeling Guide," 2020. [Online]. Available: https://blog.timescale.com/blog/what-the-heck-is-time-series-data-and-why-do-i-need-a-time-series-database-dcf3b1b18563/. [Accessed: 08-Mar-2021].

[20] P. J. Masson, P. M. Silveira, C. Duque, and P. F. Ribeiro, "Fourier series: Visualizing Joseph Fourier's imaginative discovery via FEA and time-frequency decomposition," *ICHQP 2008 13th Int. Conf. Harmon. Qual. Power*, pp. 1–5, 2008.

[21] B. G. Lewis, R. D. Herbert, and R. D. Bell, "The Application of Fourier Analysis to Forecasting the

Inbound Call Time Series of a Call Centre," *MODSIM 2003 Int. Congr. Model. Simul.*, pp. 1281–1286, 2003.

[22] N. Lange and S. L. Zeger, "Non-linear Fourier time series analysis for human brain mapping by functional magnetic resonance imaging," *J. R. Stat. Soc. Ser. C Appl. Stat.*, vol. 46, no. 1, pp. 1–29, 1997.

[23] J. L. Smart, "Estimation of time of death with a fourier series unsteady-state heat transfer model," *J. Forensic Sci.*, vol. 55, no. 6, pp. 1481–1487, 2010.

[24] A. Tealab, H. Hefny, and A. Badr, "Forecasting of nonlinear time series using ANN," *Future Computing and Informatics Journal*, vol. 2, no. 1. pp. 39–47, 2017.

[25] I. A. Iwok and G. M. Udoh, "A Comparative Study between the ARIMA- Fourier Model and the Wavelet model," no. 2015, pp. 137–144, 2016.

[26] B. T. Wilson, J. F. Knight, and R. E. McRoberts, "Harmonic regression of Landsat time series for modeling attributes from national forest inventory data," *ISPRS J. Photogramm. Remote Sens.*, vol. 137, pp. 29–46, 2018.

[27] R. T. R. Qiu, J. Park, S. N. Li, and H. Song, "Social costs of tourism during the COVID-19 pandemic," *Ann. Tour. Res.*, vol. 84, no. June, p. 102994, 2020.

[28] J. Gao, W. Ji, L. Zhang, S. Shao, Y. Wang, and F. Shi, "Fast Piecewise Polynomial Fitting of Time-Series Data for Streaming Computing," *IEEE Access*, vol. 8, pp. 43764–43775, 2020.

[29] X. Liu and Y. . Wang, "Research of automatically piecewise polynomial curve-fitting method based on least-square principle," *Sci. Technol. Eng.*, vol. 14, no. 3, pp. 55–58, 2014.

[30] P. W. Talbot *et al.*, "Correlated synthetic time series generation for energy system simulations using Fourier and ARMA signal processing," *International Journal of Energy Research*, vol. 44, no. 10. pp. 8144–8155, 2020.

[31] R. K. R. A. K. Agrawal, R. Adhikari, and R. K. R. A. K. Agrawal, "An Introductory Study on Time Series Modeling and Forecasting Ratnadip Adhikari R. K. Agrawal," *arXiv Prepr. arXiv1302.6613*, vol. 1302.6613, pp. 1–68, 2013.

[32] J. J. de Gruijter, M. F. P. Bierkens, D. J. Brus, and M. Knotters, *Time-Series Models*. 2006.

[33] M. S. Chaudhari and N. K. Choudhari, "Data visualization as a preprocessing step in designing of data mining tools visualizing time series pattern of rainfall data," *Proc. 2018 Int. Conf. Syst. Model. Adv. Res. Trends, SMART 2018*, pp. 99–103, 2018.

[34] J. Y. Lee and T. N. Wu, "Discovery of temporal variation of arsenic in a historical Blackfoot disease territory by time series analysis," *6th Int. Conf. Fuzzy Syst. Knowl. Discov. FSKD 2009*, vol. 2, pp. 542–546, 2009.

[35] P. Newbold, "ARIMA model building and the time series analysis approach to forecasting," *J. Forecast.*, vol. 2, no. 1, pp. 23–35, 1983.

[36] W. Weiqiang and N. Zhendong, "Time series analysis of NASDAQ composite based on seasonal ARIMA model," *Proc. - Int. Conf. Manag. Serv. Sci. MASS 2009*, vol. 2001, pp. 1–4, 2009.

[37] J. Soria, J. Baptiste, and J. Fourier, "Fourier Series and Recent Developments in Analysis," no. June, pp. 1–35, 2003.

[38] X.-S. Yang, "Fourier Series and Transform," *Eng. Math. with Examples Appl.*, pp. 173–185, 2017.

[39] D. S. G. Pollock, "The fourier decomposition of a time series," no. 3, pp. 1–7.

[40] A. E. Dudek, H. Hurd, and W. Wójtowicz, "Periodic autoregressive moving average methods based on Fourier representation of periodic coefficients," *Wiley Interdiscip. Rev. Comput. Stat.*, vol. 8, no. 3, pp. 130–149, 2016.

[41] M. S. Gashler and S. C. Ashmore, "Modeling time series data with deep Fourier neural networks,"

Neurocomputing, vol. 188, pp. 3–11, 2016.

[42] F. Capezzuoli, “A Basic Guide to Fitting of Time-Series with Fourier Series,” no. April, 2018.

[43] W. Gurker and R. Viertl, “Statistical Parameter Estimation,” vol. II, .

[44] A. Singhee and R. A. Rutenbar, “Statistical Blockade: A novel method for very fast Monte Carlo simulation of rare circuit events, and its application,” *Des. Autom. Test Eur. Most Influ. Pap. 10 Years Date*, pp. 235–251, 2008.

[45] F. Malik, “Monte Carlo Simulation Engine In Python Let ' s Understand The Theory First How Does Monte-Carlo Simulation Work,” 2020.

[46] S. Panigrahi and H. S. Behera, “A hybrid ETS–ANN model for time series forecasting,” *Eng. Appl. Artif. Intell.*, vol. 66, no. July, pp. 49–59, 2017.

[47] B. Wang, Y. Fang, J. Sheng, W. Gui, and Y. Sun, “BTP prediction model based on ANN and regression analysis,” *Proc. - 2009 2nd Int. Work. Knowl. Discov. Data Mining, WKKD 2009*, pp. 108–111, 2009.

[48] A. Palmer, J. José Montaño, and A. Sesé, “Designing an artificial neural network for forecasting tourism time series,” *Tour. Manag.*, vol. 27, no. 5, pp. 781–790, 2006.

[49] T. Kriechbaumer, A. Angus, D. Parsons, and M. Rivas Casado, “An improved wavelet-ARIMA approach for forecasting metal prices,” *Resour. Policy*, vol. 39, no. 1, pp. 32–41, 2014.

[50] R. Rusli, A. N. Hidayanto, and Y. Ruldeviyani, “Consumption Prediction on Steam Power Plant Using Data Mining Hybrid Particle Swarm Optimization (PSO) and Auto Regressive Integrated Moving Average (ARIMA),” *2019 Int. Work. Big Data Inf. Secur. IWBIS 2019*, pp. 15–20, 2019.

[51] S. Mehrmolaei and M. R. Keyvanpour, “Time series forecasting using improved ARIMA,” *2016 Artif. Intell. Robot. IRANOPEN 2016*, pp. 92–97, 2016.

[52] C. N. Babu and B. E. Reddy, “Temperature Using Variants of ARIMA models,” *IEEE-International Conf. Adv. Eng. Sci. Manag. (ICAESM -2012)*, no. 1, pp. 256–260, 2012.

[53] I. Abu Amra and A. Maghari, “Forecasting Groundwater Production and Rain Amounts Using ARIMA-Hybrid ARIMA: Case Study of Deir El-Balah City in Gaza,” *Proc. - 2018 Int. Conf. Promis. Electron. Technol. ICPET 2018*, pp. 135–140, 2018.

[54] C. Narendra Babu and B. Eswara Reddy, “Prediction of selected Indian stock using a partitioning–interpolation based ARIMA–GARCH model,” *Appl. Comput. Informatics*, vol. 11, no. 2, pp. 130–143, 2015.

[55] B. G. Koo, M. S. Kim, K. H. Kim, H. T. Lee, J. H. Park, and C. H. Kim, “Short-term electric load forecasting using data mining technique,” *7th Int. Conf. Intell. Syst. Control. ISCO 2013*, no. PECon 08, pp. 153–157, 2013.

[56] J. Zeng, L. Zhang, G. Shi, T. Liu, and K. Lin, “An ARIMA based real-time monitoring and warning algorithm for the anomaly detection,” *Proc. Int. Conf. Parallel Distrib. Syst. - ICPADS*, vol. 2017-Decem, pp. 469–476, 2018.

[57] M. V. Riesgo García, A. Krzemień, M. Á. Manzanedo del Campo, C. Escanciano García-Miranda, and F. Sánchez Lasheras, “Rare earth elements price forecasting by means of transgenic time series developed with ARIMA models,” *Resour. Policy*, vol. 59, no. June, pp. 95–102, 2018.

[58] I. Khandelwal, R. Adhikari, and G. Verma, “Time series forecasting using hybrid arima and ann models based on DWT Decomposition,” *Procedia Comput. Sci.*, vol. 48, no. C, pp. 173–179, 2015.

[59] M. Matyjaszek, P. Riesgo Fernández, A. Krzemień, K. Wodarski, and G. Fidalgo Valverde, “Forecasting coking coal prices by means of ARIMA models and neural networks, considering the transgenic time series theory,” *Resour. Policy*, vol. 61, no. February, pp. 283–292, 2019.

[60] M. Qin, Z. Li, and Z. Du, “Red tide time series forecasting by combining ARIMA and deep belief network,” *Knowledge-Based Syst.*, vol. 125, pp. 39–52, 2017.

[61] B. Brentan, E. Luvizotto Jr., J. Izquierdo, and R. Pérez-García, “Fourier series and Chebyshev polynomial applied to real-time water demand forecasting,” *Acta Univ.*, vol. 26, pp. 74–81, 2017.

[62] I. A. Iwok, “Seasonal Modelling of Fourier Series with Linear Trend,” *Int. J. Stat. Probab.*, vol. 5, no. 6, p. 65, 2016.

[63] M. Fidino and S. B. Magle, “Using Fourier series to estimate periodic patterns in dynamic occupancy models,” *Ecosphere*, vol. 8, no. 9, 2017.

[64] E. B. Brooks, V. A. Thomas, R. H. Wynne, and J. W. Coulston, “Fitting the multitemporal curve: A fourier series approach to the missing data problem in remote sensing analysis,” *IEEE Trans. Geosci. Remote Sens.*, vol. 50, no. 9, pp. 3340–3353, 2012.

[65] A. Taiwo and T. Olatayo, “an Improved Procedure for Fourier Regression Analysis,” *Ann. Comput. Sci. Ser. Vol. XVI fasc. 2 – 2018*, vol. XVI, 2018.

[66] A. Prahutama, Suparti, and T. W. Utami, “Modelling fourier regression for time series data - A case study: Modelling inflation in foods sector in Indonesia,” *J. Phys. Conf. Ser.*, vol. 974, no. 1, 2018.

[67] C. L. Tsai, W. T. Chen, and C. S. Chang, “Polynomial-Fourier series model for analyzing and predicting electricity consumption in buildings,” *Energy Build.*, vol. 127, pp. 301–312, 2016.

[68] A. Alamsyah and B. Nurriz, “Monte Carlo simulation and clustering for customer segmentation in business organization,” *Proceeding - 2017 3rd Int. Conf. Sci. Technol. ICST 2017*, pp. 104–109, 2017.

[69] R. Li, L. Chen, and X. Gu, “Condition-based maintenance of electrical equipments based on ID3 algorithm and Monte Carlo stimulation,” *Proc. IEEE Power Eng. Soc. Transm. Distrib. Conf.*, vol. 2005, pp. 1–5, 2005.

[70] A. Muhammad, F. Külahcı, and P. Akram, “Modeling radon time series on the North Anatolian Fault Zone, Turkiye: Fourier transforms and Monte Carlo simulations,” *Nat. Hazards*, no. 0123456789, 2020.

[71] J. Moudrik, T. Kren, and R. Neruda, “Algorithm discovery with Monte-Carlo search: Controlling the size,” *Proc. - Int. Conf. Tools with Artif. Intell. ICTAI*, vol. 2017-Novem, pp. 390–395, 2018.

[72] O. Gonzalez, H. P. O’Rourke, I. C. Wurpts, and K. J. Grimm, “Analyzing Monte Carlo Simulation Studies With Classification and Regression Trees,” *Struct. Equ. Model.*, vol. 25, no. 3, pp. 403–413, 2018.

[73] H. Qi and X. Hu, “Monte Carlo Tree Search-based intersection signal optimization model with channelized section spillover,” *Transp. Res. Part C Emerg. Technol.*, vol. 106, no. December 2018, pp. 281–302, 2019.

[74] R. Al-Dalky, K. Taha, D. Al Homouz, and M. Qasaimeh, “Applying Monte Carlo Simulation to Biomedical Literature to Approximate Genetic Network,” *IEEE/ACM Trans. Comput. Biol. Bioinforma.*, vol. 13, no. 3, pp. 494–504, 2016.

[75] Z. Tian, Y. Ren, and G. Wang, “Short-term wind power prediction based on empirical mode decomposition and improved extreme learning machine,” *J. Electr. Eng. Technol.*, vol. 13, no. 5, pp. 1841–1851, 2018.

[76] Z. Wang *et al.*, “Failure prediction using machine learning and time series in optical network,” *Opt. Express*, vol. 25, no. 16, p. 18553, 2017.

[77] Ü. Ağbulut, A. E. Gürel, and Y. Biçen, “Prediction of daily global solar radiation using different machine learning algorithms: Evaluation and comparison,” *Renew. Sustain. Energy Rev.*, vol. 135, no. August 2020, 2021.

[78] A. Khosravi, L. Machado, and R. O. Nunes, “Time-series prediction of wind speed using machine learning algorithms: A case study Osorio wind farm, Brazil,” *Appl. Energy*, vol. 224, no. December 2017, pp. 550–566, 2018.

[79] B. M. Pavlyshenko, “Machine-learning models for sales time series forecasting,” *Data*, vol. 4, no. 1,

pp. 1–11, 2019.

[80] A. R. S. Parmezan, V. M. A. Souza, and G. E. A. P. A. Batista, "Evaluation of statistical and machine learning models for time series prediction: Identifying the state-of-the-art and the best conditions for the use of each model," *Inf. Sci. (Ny).*, vol. 484, pp. 302–337, 2019.

[81] T. Meline, "Selecting Studies for Systemic Review: Inclusion and Exclusion Criteria," *Contemp. Issues Commun. Sci. Disord.*, vol. 33, no. Spring, pp. 21–27, 2006.

[82] A. Booth, "Searching for qualitative research for inclusion in systematic reviews: A structured methodological review," *Syst. Rev.*, vol. 5, no. 1, pp. 1–23, 2016.

[83] K. Kalpakis, D. Gada, and V. Puttagunta, "Distance measures for effective clustering of ARIMA time-series," *Proc. - IEEE Int. Conf. Data Mining, ICDM*, pp. 273–280, 2001.

[84] C. Y. Lam, W. H. Ip, and C. W. Lau, "A business process activity model and performance measurement using a time series ARIMA intervention analysis," *Expert Syst. Appl.*, vol. 36, no. 3 PART 2, pp. 6986–6994, 2009.

[85] L. F. Chaparro and A. Akan, *Frequency Analysis: The Fourier Series*. 2019.

[86] S. J. Danbatta and A. Varol, "Monte Carlo forecasting of time series data using Polynomial-Fourier series model," *Int. J. Model. Simulation, Sci. Comput.*, vol. 12, no. 03, p. 2141004, Jun. 2021.

[87] S. J. Danbatta and A. Varol, "ANN–polynomial–Fourier series modeling and Monte Carlo forecasting of tourism data," *J. Forecast.*, no. October, pp. 1–13, Jan. 2022.

[88] S. Makridakis, E. Spiliotis, and V. Assimakopoulos, "Statistical and Machine Learning forecasting methods: Concerns and ways forward," *PLoS One*, vol. 13, no. 3, pp. 1–26, 2018.

[89] S. J. Danbatta and A. Varol, "Modeling and Forecasting of Tourism Time Series Data using ANN-Fourier Series Model and Monte Carlo Simulation," in *2021 9th International Symposium on Digital Forensics and Security (ISDFS)*, 2021, pp. 1–6.

[90] V. Nourani, G. Elkiran, and J. Abdullahi, "Multi-step ahead modeling of reference evapotranspiration using a multi-model approach," *J. Hydrol.*, vol. 581, p. 124434, 2020.

[91] S. Masum, Y. Liu, and J. Chiverton, *Multi-step time series forecasting of electric load using machine learning models*, vol. 10841 LNAI. Springer International Publishing, 2018.

[92] D. Dua and C. Graff, "{UCI} Machine Learning Repository." 2017.

[93] L. M. Candanedo, V. Feldheim, and D. Deramaix, "Data driven prediction models of energy use of appliances in a low-energy house," *Energy Build.*, vol. 140, pp. 81–97, 2017.

[94] A. Akossou, "Impact of data structure on the estimators R-square and adjusted R-square in linear regression.," *Int. J. Math. Comput.*, vol. 20, no. 03, pp. 84–93, 2013.

[95] M. Monteiro and M. Costa, "A time series model comparison for monitoring and forecasting water quality variables," *Hydrology*, vol. 5, no. 3, pp. 1–20, 2018.

[96] D. A. Pierce, "R ~ Measures for Time Series," *J. Am. Stat. Assoc.*, vol. 74, no. 368, pp. 901–910, 2007.

[97] Y. Zhang, G. Li, B. Muskat, R. Law, and Y. Yang, "Group pooling for deep tourism demand forecasting," *Ann. Tour. Res.*, vol. 82, no. March, p. 102899, 2020.

[98] J. W. Bi, Y. Liu, and H. Li, "Daily tourism volume forecasting for tourist attractions," *Ann. Tour. Res.*, vol. 83, no. September 2019, p. 102923, 2020.

[99] F. Günay, E. Bayraktaroğlu, and K. Özkul, "Ekonomi Assessing the Short-term Impacts of COVID-19 Pandemic on Foreign Visitor's Demand for Turkey: A Scenario Analysis," *J. Ekon.*, vol. 04, no. April, pp. 80–85, 2020.

[100] O. Claveria and S. Torra, "Forecasting tourism demand to Catalonia: Neural networks vs. time series models," *Econ. Model.*, vol. 36, pp. 220–228, 2014.

[101] G. Borowik, Z. M. Wawrzyniak, and P. Cichosz, "Time series analysis for crime forecasting," *26th Int. Conf. Syst. Eng. ICSEng 2018 - Proc.*, 2019.

[102] E. Kamir, F. Waldner, and Z. Hochman, "Estimating wheat yields in Australia using climate records, satellite image time series and machine learning methods," *ISPRS J. Photogramm. Remote Sens.*, vol. 160, no. February 2019, pp. 124–135, 2020.

[103] W. Chen, H. Ning, X. Mao, and B. Wang, "Data mining modeling for electromagnetic scattering computing," *Proc. - 5th Int. Conf. Fuzzy Syst. Knowl. Discov. FSKD 2008*, vol. 2, pp. 137–140, 2008.

[104] A. Shabri, R. Samsudin, and Y. Yusoff, "Combining Deep Neural Network and Fourier Series for Tourist Arrivals Forecasting," *IOP Conf. Ser. Mater. Sci. Eng.*, vol. 864, no. 1, 2020.

[105] E. S. Silva, H. Hassani, S. Heravi, and X. Huang, "Forecasting tourism demand with denoised neural networks," *Ann. Tour. Res.*, vol. 74, no. November 2018, pp. 134–154, 2019.

[106] M. Álvarez-Díaz, M. González-Gómez, and M. Otero-Giráldez, "Forecasting International Tourism Demand Using a Non-Linear Autoregressive Neural Network and Genetic Programming," *Forecasting*, vol. 1, no. 1, pp. 90–106, 2018.

[107] N. Kulendran and K. K. F. Wong, "Modeling seasonality in tourism forecasting," *J. Travel Res.*, vol. 44, no. 2, pp. 163–170, 2005.

[108] S. J. Danbatta and A. Varol, "Predicting Student's Final Graduation CGPA Using Data Mining and Regression Methods: A Case Study of Kano Informatics Institute," in *2020 8th International Symposium on Digital Forensics and Security (ISDFS)*, 2020, pp. 1–7.

APPENDICES

APPENDIX- 1: MONTE CARLO SIMULATION PATHS RESULTS FOR THE TOURISM DATA DURING THE COVID-19 PANDEMIC PERIOD

	0	1	2	3	4 \
143	1.647641e+06	1.058464e+06	2.254387e+06	5.319908e+05	2.172985e+06
144	1.982171e+06	2.067067e+06	1.150223e+06	2.573523e+05	7.930202e+05
145	5.694130e+05	1.335623e+06	2.084852e+06	1.849507e+06	1.695127e+06
146	6.495340e+05	3.680822e+06	5.716560e+05	2.583281e+06	1.006132e+06
147	5.464924e+05	1.538797e+06	2.614183e+06	2.780456e+06	1.649427e+06
148	2.573790e+06	2.898582e+06	3.861046e+06	1.584587e+06	1.146052e+06
149	4.331338e+06	3.894241e+06	2.151202e+06	3.792330e+06	4.399953e+06
150	5.282511e+06	5.227048e+06	2.580188e+06	5.676020e+06	4.420509e+06
151	3.305264e+06	4.781955e+06	5.125932e+06	6.244620e+06	4.667996e+06
152	4.411654e+06	4.514356e+06	5.558579e+06	2.376482e+06	3.462694e+06
153	3.929691e+06	2.353658e+06	3.650030e+06	4.166113e+06	3.599810e+06
154	3.388426e+06	1.603539e+06	3.499619e+06	1.541360e+06	3.417788e+06
155	1.529100e+06	1.912009e+06	2.701435e+06	2.537855e+06	1.839797e+06
156	7.441661e+05	2.115590e+06	1.918264e+06	8.014981e+05	1.650175e+06
157	5.899973e+05	1.966261e+06	4.316929e+05	3.929152e+06	7.860998e+05
158	3.285271e+06	1.797623e+06	3.548412e+05	6.299976e+05	2.253719e+06
159	2.947722e+06	2.675918e+06	2.409692e+06	6.054026e+05	2.268203e+06
160	1.598828e+06	3.397848e+06	4.011580e+06	2.870117e+06	2.693476e+06
161	7.930202e+05	3.197303e+06	4.335043e+06	4.095186e+06	9.235349e+05
162	1.054050e+06	7.495153e+05	9.960431e+05	3.477385e+06	3.972206e+06
163	2.829256e+06	3.792999e+06	4.608254e+06	5.664869e+06	3.879088e+06
164	3.086008e+06	3.733900e+06	4.984353e+06	8.365251e+05	6.195724e+06
165	4.161659e+06	4.600739e+06	4.594826e+05	5.190852e+06	6.067624e+06
166	2.528309e+06	2.255271e+06	2.881775e+06	2.657186e+06	2.662462e+06
167	9.977929e+05	2.293676e+06	7.640169e+05	1.761263e+06	3.665140e+06
168	2.268873e+06	7.096177e+05	2.685687e+06	1.192051e+06	1.374363e+06
169	3.547679e+05	9.380366e+05	9.405367e+05	1.501827e+06	2.836797e+06
170	1.975070e+06	2.452888e+06	9.670398e+05	1.283962e+06	2.198265e+06
171	1.783208e+06	2.627398e+06	2.011778e+06	3.319975e+06	1.786344e+06
172	2.927389e+06	3.210332e+06	3.699819e+06	4.247929e+06	2.987828e+06

173	3.624271e+06	3.505850e+06	3.558637e+06	2.943821e+06	2.095371e+06
174	3.769336e+06	4.880645e+06	4.723761e+06	4.559742e+06	1.068551e+06
175	6.351971e+06	4.791948e+06	4.847684e+06	1.025046e+06	4.995234e+06
176	2.917019e+06	2.566723e+06	3.446571e+06	4.647035e+06	3.643262e+06
177	2.098244e+06	4.242326e+06	3.315608e+06	3.449238e+06	3.648294e+06
178	3.966150e+06	2.906224e+06	3.708870e+06	3.455412e+06	2.041525e+06
179	2.145633e+06	2.793263e+06	2.641690e+06	3.401535e+06	3.036217e+05
180	8.945317e+05	2.477926e+06	3.289233e+06	4.143633e+05	2.315975e+06
181	1.162827e+06	1.492182e+06	3.599056e+06	2.011023e+06	2.816470e+05
182	2.249116e+06	3.737016e+06	4.178148e+06	1.357147e+06	3.165329e+06
183	1.559742e+06	2.685032e+06	1.558438e+06	2.359074e+06	2.915579e+06
184	3.170980e+06	4.100321e+06	4.371750e+06	4.306585e+06	4.697121e+06
185	4.457700e+06	5.533723e+06	3.620279e+06	5.190952e+06	6.190006e+05
186	4.319693e+06	3.713326e+06	4.227054e+06	4.022774e+06	5.499471e+06
187	1.054050e+06	5.350236e+06	4.937001e+06	4.930198e+06	4.363253e+06
188	1.515765e+06	3.156913e+06	2.330694e+06	5.033287e+06	2.151209e+06
189	3.570566e+06	2.948497e+06	3.484352e+06	3.531333e+06	1.829716e+06
190	4.634103e+06	1.523239e+06	9.670398e+05	8.794134e+05	4.364529e+06
191	4.594826e+05	2.021483e+06	5.901847e+05	1.815074e+06	1.228069e+06
192	4.594826e+05	1.029652e+06	6.335022e+05	1.097555e+06	2.776025e+06
193	7.350137e+05	6.480039e+05	9.927701e+05	1.126558e+06	8.365251e+05
194	1.336323e+06	1.057047e+06	1.010545e+06	1.155561e+06	2.929438e+06
195	2.438182e+06	3.232108e+06	1.828115e+05	2.339442e+06	7.205120e+05
196	2.502656e+06	2.183669e+06	2.012578e+06	5.490578e+06	2.445531e+06
197	2.907639e+06	5.235046e+06	4.348087e+06	5.382426e+06	6.270277e+06
198	4.606232e+06	7.495153e+05	4.672101e+06	4.964217e+06	6.037974e+06

	5	6	7	8	9 \
143	1.299919e+06	4.380245e+05	7.495153e+05	4.004278e+06	9.235349e+05
144	7.640169e+05	8.018114e+05	8.945317e+05	1.089653e+06	2.139920e+06
145	2.933769e+06	2.498148e+06	2.900507e+06	5.754957e+05	7.876464e+05
146	3.704915e+05	3.361478e+06	8.846790e+05	1.945512e+06	2.424001e+04
147	3.736769e+05	2.929044e+05	2.361230e+06	2.567383e+06	1.347664e+06
148	4.205901e+06	2.362077e+06	4.739842e+05	1.666174e+06	1.680298e+06
149	3.629720e+06	3.684070e+06	4.883660e+06	1.126558e+06	1.805346e+06
150	4.161029e+06	6.684910e+06	3.038449e+06	6.898377e+06	8.075218e+05

151	5.781879e+06	3.024306e+06	5.697585e+06	3.100948e+06	3.579050e+06
152	2.808926e+06	4.997997e+06	1.141059e+06	2.653151e+06	3.787913e+06
153	4.719846e+06	2.061277e+06	5.121857e+06	3.773297e+06	4.947781e+06
154	3.664887e+06	1.015255e+06	9.686199e+05	2.099271e+06	1.768831e+06
155	4.217918e+05	4.667876e+05	8.388249e+05	1.638474e+06	1.581739e+06
156	1.035437e+06	5.006918e+05	2.963016e+06	3.848025e+06	2.175094e+06
157	2.488802e+06	1.257073e+06	3.566186e+06	9.670398e+05	2.969785e+06
158	7.607671e+05	5.349393e+05	2.042883e+06	3.809174e+06	2.033291e+06
159	1.912511e+06	2.971268e+06	3.028595e+06	2.824384e+06	2.877935e+06
160	3.520573e+06	3.449778e+06	3.575636e+06	2.192984e+06	3.713787e+06
161	3.613425e+06	4.762331e+06	5.454288e+06	6.610804e+06	3.130455e+06
162	3.401233e+06	5.186828e+06	3.734197e+06	3.312486e+06	4.796212e+06
163	2.839257e+06	4.584602e+06	4.457445e+06	4.317708e+06	4.832488e+06
164	1.155561e+06	5.371206e+06	2.522160e+06	2.695216e+06	4.138680e+06
165	3.380392e+06	2.221904e+06	4.115934e+06	2.992250e+06	3.474313e+06
166	2.257992e+06	3.715263e+06	3.231435e+06	2.459998e+06	1.897686e+06
167	1.059253e+06	1.112056e+06	9.188922e+05	2.902505e+06	1.938130e+06
168	2.352565e+06	3.329446e+05	5.609941e+05	1.821842e+06	4.739842e+05
169	1.732692e+06	3.268600e+05	4.208219e+06	1.748780e+06	3.395201e+06
170	2.407509e+06	5.609941e+05	3.948915e+06	2.336836e+06	1.800009e+06
171	4.445246e+06	4.119660e+06	3.380987e+06	3.091116e+06	1.914051e+06
172	6.139272e+06	3.977567e+06	2.358885e+06	3.587355e+06	2.508690e+06
173	3.438657e+06	4.382441e+06	2.678530e+06	4.002107e+06	5.767472e+06
174	3.216601e+06	3.303704e+06	3.109508e+06	1.184564e+06	2.496976e+06
175	3.984230e+06	5.356077e+06	4.206937e+06	2.959007e+06	4.898170e+06
176	1.025046e+06	2.989652e+06	3.767588e+06	3.734552e+06	3.539170e+06
177	3.141298e+06	2.124442e+06	3.117317e+06	4.029829e+06	3.984756e+06
178	2.533242e+06	2.625846e+06	2.413106e+06	1.623185e+06	1.576472e+06
179	9.533498e+05	1.238929e+06	1.141059e+06	1.503476e+06	3.754343e+06
180	2.253291e+06	2.220377e+06	1.023971e+06	3.861594e+06	3.058031e+06
181	1.790925e+06	3.451084e+06	1.228069e+06	2.353378e+06	3.805691e+06
182	1.646496e+06	2.970996e+06	2.534730e+06	1.702158e+06	4.516050e+06
183	3.189638e+06	3.616718e+06	2.123839e+06	5.219406e+06	1.996077e+06
184	3.104507e+06	4.302014e+06	5.721267e+06	4.834324e+06	3.914186e+06
185	4.341041e+06	6.133179e+06	3.106649e+06	5.721156e+06	4.297871e+06
186	5.781688e+06	4.307834e+06	4.981166e+06	6.225488e+06	3.133061e+06

187	4.650636e+06	3.753690e+06	3.747520e+06	3.925055e+06	3.228205e+06
188	3.112114e+06	4.132132e+06	4.541633e+06	2.144042e+06	3.500185e+06
189	4.032073e+06	3.230273e+06	2.615436e+06	4.384428e+06	3.633984e+06
190	2.577971e+06	2.486867e+06	4.599239e+06	2.452835e+06	3.760114e+06
191	3.164137e+05	1.130122e+06	6.915088e+05	1.328877e+06	1.326613e+06
192	5.899973e+05	4.739842e+05	4.324457e+05	6.480039e+05	1.422667e+06
193	5.464924e+05	7.495153e+05	8.574144e+05	4.396566e+05	1.121812e+06
194	9.815415e+05	3.007747e+04	2.610712e+06	9.525382e+05	1.892724e+06
195	5.174892e+05	3.066566e+06	2.152942e+06	1.083053e+06	8.695062e+05
196	1.190293e+06	1.182645e+06	5.616931e+06	1.043099e+06	2.179022e+06
197	4.093558e+06	4.873589e+06	2.496044e+06	4.291919e+06	4.642497e+06
198	4.721335e+06	5.411676e+06	4.856657e+06	5.703997e+06	4.725584e+06

	...	90	91	92	93 \
143	...	1.712682e+06	6.346213e+05	1.126558e+06	4.425099e+05
144	...	1.025046e+06	1.408918e+06	2.335808e+06	8.694762e+05
145	...	2.430741e+06	1.133676e+06	2.934313e+05	1.068551e+06
146	...	1.184564e+06	1.507096e+06	1.661578e+06	1.213568e+06
147	...	3.810755e+04	1.229846e+06	2.260648e+05	2.693820e+06
148	...	3.633208e+06	2.253626e+06	1.065568e+06	1.470269e+06
149	...	1.511306e+06	2.927752e+06	3.224954e+06	4.551437e+06
150	...	3.684829e+06	2.784276e+06	2.476855e+06	3.200361e+06
151	...	5.289595e+06	3.104550e+06	3.432428e+06	3.776203e+06
152	...	3.672637e+06	4.434846e+06	4.557585e+06	4.099323e+06
153	...	3.513392e+06	1.588577e+06	2.792157e+06	2.718357e+06
154	...	3.584016e+06	2.294584e+06	3.022034e+06	6.133087e+05
155	...	2.687302e+06	3.874662e+06	1.213568e+06	2.261099e+06
156	...	2.069457e+06	7.053743e+05	4.739842e+05	2.242806e+06
157	...	1.602412e+06	1.827469e+06	2.015686e+06	2.792535e+06
158	...	2.248895e+06	2.227641e+06	1.694832e+06	1.039548e+06
159	...	3.327802e+06	9.972790e+05	1.054050e+06	2.394274e+06
160	...	2.464626e+06	2.852173e+06	3.835121e+06	3.724845e+06
161	...	2.073529e+06	3.284407e+06	4.639367e+06	3.558567e+06
162	...	4.516742e+06	2.826086e+06	3.764319e+06	4.290317e+06
163	...	6.412155e+06	2.251098e+06	4.566604e+06	2.424279e+06
164	...	5.174892e+05	4.709861e+06	3.025249e+06	3.684510e+06

165	...	2.164938e+06	9.235349e+05	3.296734e+06	2.341858e+06
166	...	1.188002e+06	3.422202e+06	3.461292e+06	1.975833e+06
167	...	3.059305e+06	3.385406e+06	1.352890e+06	1.640145e+06
168	...	1.958619e+06	7.812012e+05	6.423578e+05	1.996135e+06
169	...	2.840260e+05	1.659494e+06	1.044385e+06	3.441979e+06
170	...	8.819238e+05	2.222774e+06	4.352040e+06	1.083053e+06
171	...	4.737338e+06	2.475542e+06	2.763876e+06	2.346910e+06
172	...	4.378929e+06	1.644818e+06	4.456946e+06	2.037131e+06
173	...	2.858499e+06	3.404439e+06	2.339992e+06	3.515572e+06
174	...	4.204554e+06	5.447944e+06	3.664422e+06	2.932123e+06
175	...	4.170942e+06	5.510894e+06	4.442038e+06	3.666901e+06
176	...	4.200634e+06	4.098027e+06	3.212998e+06	3.697750e+06
177	...	3.995166e+06	3.536679e+06	2.163082e+06	4.491859e+06
178	...	2.600200e+06	2.618171e+06	3.599540e+06	3.335740e+06
179	...	1.285053e+06	2.100838e+06	1.960029e+06	2.022151e+06
180	...	1.860355e+06	1.631589e+06	3.423284e+06	8.378552e+05
181	...	3.728508e+06	2.741192e+06	2.152467e+06	2.056843e+06
182	...	4.029495e+06	3.039955e+06	7.778716e+05	4.611904e+06
183	...	3.925067e+06	2.685989e+06	9.090333e+05	1.722539e+06
184	...	2.347172e+06	3.734989e+06	4.125756e+06	2.595346e+06
185	...	5.701768e+06	3.694633e+06	4.273337e+06	3.825416e+06
186	...	1.228069e+06	6.688924e+06	5.710610e+06	4.037557e+06
187	...	3.653084e+06	5.169599e+06	3.934973e+06	3.722305e+06
188	...	4.075862e+06	1.010545e+06	2.568175e+06	3.899986e+06
189	...	2.353745e+06	2.507430e+06	3.822093e+06	2.618292e+06
190	...	2.512834e+06	3.795876e+06	1.713362e+06	3.590945e+06
191	...	1.937511e+06	4.909754e+05	1.682646e+06	1.037788e+06
192	...	1.242571e+06	3.489983e+05	7.786415e+05	2.176512e+06
193	...	6.246751e+05	6.874081e+05	1.010545e+06	1.213568e+06
194	...	1.557480e+06	5.418314e+05	1.083053e+06	7.495153e+05
195	...	1.136470e+06	1.291179e+06	1.622032e+06	9.544336e+04
196	...	3.402879e+06	2.335697e+06	5.321281e+06	3.109582e+06
197	...	4.635343e+06	1.170063e+06	3.661888e+06	4.379301e+06
198	...	5.703995e+06	4.937637e+06	5.687245e+06	4.988695e+06

	94	95	96	97	98 \
143	7.495153e+05	1.925939e+05	1.233517e+06	6.349810e+05	1.909405e+06
144	9.601858e+05	2.279748e+06	1.097617e+06	9.454852e+05	3.959850e+05
145	3.161481e+05	1.492610e+06	9.979659e+05	1.919276e+06	7.329161e+05
146	2.625246e+05	1.986997e+06	3.131144e+05	1.428815e+05	1.097555e+06
147	9.815415e+05	8.655284e+05	1.410916e+06	4.493578e+06	2.128659e+06
148	2.212188e+06	3.769430e+06	1.002192e+06	1.304144e+06	2.404440e+06
149	3.423200e+06	5.281278e+06	1.588505e+06	7.060104e+05	6.219367e+06
150	6.762567e+06	1.964075e+06	4.370987e+06	4.015480e+06	2.421598e+06
151	2.707393e+06	5.449575e+06	4.707367e+06	5.174892e+05	3.785468e+06
152	2.611910e+06	6.490168e+06	3.004929e+06	4.830632e+06	4.871986e+06
153	3.435591e+06	3.706303e+06	4.176176e+06	4.065583e+06	4.517400e+06
154	3.176325e+06	1.215416e+06	3.346189e+06	1.047116e+06	7.516098e+05
155	1.840956e+06	6.190006e+05	1.689719e+05	1.997219e+06	2.176493e+06
156	1.928769e+06	2.205534e+06	4.580979e+05	7.594949e+05	2.087114e+05
157	3.218920e+05	9.217497e+05	1.192774e+06	8.249072e+05	3.143101e+06
158	7.127719e+05	1.674155e+06	2.433235e+06	1.783849e+06	1.354951e+06
159	2.914369e+06	2.623812e+06	7.193682e+05	1.420234e+06	3.089770e+06
160	2.489918e+06	4.642661e+06	5.196587e+06	7.060104e+05	2.469668e+06
161	4.030320e+06	4.211787e+06	4.631710e+06	7.785186e+05	4.198457e+06
162	6.628694e+06	3.372018e+06	5.167088e+06	2.832647e+06	3.127260e+06
163	4.146687e+06	6.341449e+06	3.147441e+06	4.346983e+06	3.107660e+06
164	2.460353e+06	5.851399e+06	3.336966e+06	3.360747e+06	3.933117e+06
165	4.890694e+06	4.063158e+06	5.557354e+06	4.480478e+06	3.090266e+06
166	2.093455e+06	3.760104e+06	3.282098e+06	4.052633e+06	2.892471e+06
167	1.931149e+06	9.235349e+05	2.232721e+06	2.615995e+06	2.820435e+06
168	5.461706e+05	2.821154e+06	3.881177e+05	1.184564e+06	3.570213e+06
169	2.663924e+06	4.739842e+05	5.076330e+05	1.173815e+06	7.728269e+05
170	4.723828e+05	1.376740e+06	8.816283e+04	1.823140e+06	1.483765e+06
171	4.545082e+06	3.940176e+06	4.594826e+05	2.440720e+06	4.882692e+06
172	5.645421e+06	2.974216e+06	5.368718e+06	3.899535e+06	2.085582e+06
173	4.757007e+06	4.623738e+06	6.297932e+06	5.193458e+06	2.969348e+06
174	3.108393e+06	4.849526e+06	3.293517e+06	2.632413e+06	3.681302e+06
175	4.176196e+06	4.804278e+06	4.370140e+06	9.380366e+05	3.921754e+06
176	3.625462e+06	4.539625e+06	5.200848e+06	2.690309e+06	2.700905e+06
177	3.634726e+06	1.718791e+06	3.555833e+06	2.559876e+06	4.130680e+06

178	3.649451e+06	2.824570e+06	2.810334e+06	2.308221e+06	1.459726e+06
179	2.081324e+06	7.684011e+05	1.369278e+06	1.257073e+06	5.496662e+05
180	8.622138e+05	8.054929e+05	3.685524e+06	1.318321e+06	3.050892e+06
181	1.033247e+06	6.625055e+05	2.956883e+06	1.141059e+06	2.198910e+06
182	2.472411e+06	3.688166e+06	1.951897e+06	3.755416e+06	9.304673e+05
183	2.948647e+06	3.789122e+06	1.594426e+06	1.068551e+06	4.168944e+06
184	5.230750e+06	4.430270e+06	4.609048e+06	4.558115e+06	3.329909e+06
185	3.447402e+06	3.309837e+06	5.411825e+06	4.118411e+06	4.978847e+06
186	4.865826e+06	5.566724e+06	5.396961e+06	3.523548e+06	3.107667e+06
187	3.727655e+06	4.110084e+06	5.438809e+06	3.696070e+06	4.794678e+06
188	2.940348e+06	2.680250e+06	3.522239e+06	2.924746e+06	3.627204e+06
189	3.186767e+06	3.683007e+06	4.019317e+06	1.936815e+06	2.671941e+06
190	2.926795e+06	5.208413e+06	3.891776e+06	1.985306e+06	2.281318e+06
191	2.567709e+06	6.106379e+05	8.510267e+05	4.032340e+06	6.150134e+05
192	7.205120e+05	4.877215e+05	1.218480e+06	7.596827e+05	2.442219e+06
193	1.097555e+06	7.640169e+05	8.220235e+05	9.525382e+05	9.960431e+05
194	5.899973e+05	1.472519e+06	1.010545e+06	2.584290e+06	4.739842e+05
195	3.038429e+06	8.220235e+05	1.682249e+06	8.404752e+05	6.589565e+05
196	2.360295e+06	3.128296e+06	1.257654e+06	3.597659e+06	1.947939e+06
197	3.499018e+06	3.461048e+06	3.733740e+06	4.786916e+06	5.102342e+06
198	4.079778e+06	4.757324e+06	3.666731e+06	4.271485e+06	3.883541e+06

	99
143	1.566140e+06
144	6.320987e+05
145	2.036814e+06
146	1.310820e+06
147	3.661276e+06
148	3.433176e+06
149	4.432661e+06
150	4.181335e+06
151	4.772707e+06
152	3.713710e+06
153	2.669646e+06
154	2.582647e+06
155	3.321945e+06

156	1.476878e+06
157	3.194839e+06
158	3.494774e+06
159	1.209381e+06
160	3.842452e+06
161	5.587937e+06
162	5.529190e+06
163	4.565596e+06
164	3.313380e+06
165	1.025046e+06
166	3.515564e+06
167	2.047966e+06
168	8.790541e+05
169	2.678870e+06
170	1.503704e+06
171	4.233543e+06
172	3.441743e+06
173	4.496427e+06
174	4.874890e+06
175	4.771618e+06
176	4.267661e+06
177	4.081760e+06
178	2.297075e+06
179	1.791589e+06
180	3.964099e+06
181	3.379542e+06
182	3.135320e+06
183	6.625055e+05
184	2.824001e+06
185	4.518907e+06
186	6.973174e+06
187	3.722028e+06
188	2.375264e+06
189	4.307879e+06
190	9.815415e+05
191	1.157308e+06

192	6.044990e+05
193	7.703948e+05
194	1.722266e+06
195	3.929725e+05
196	3.426555e+06
197	2.639590e+06
198	5.532860e+06

APPENDIX- 2: MONTE CARLO SIMULATION PATHS RESULTS FOR THE TOURISM DATA BEFORE THE COVID-19 PANDEMIC PERIOD

	0	1	2	3	4 \
135	2.825783e+06	2.474598e+06	3.187439e+06	1.124699e+06	3.138919e+06
136	3.788735e+06	3.839338e+06	3.292845e+06	2.760642e+06	5.162897e+06
137	3.620017e+06	4.076723e+06	4.523308e+06	4.383028e+06	4.291008e+06
138	3.535807e+06	5.342632e+06	3.489387e+06	4.688433e+06	3.748360e+06
139	5.857443e+06	3.973016e+06	4.614009e+06	4.713117e+06	4.038958e+06
140	3.531608e+06	3.725203e+06	4.298889e+06	2.941985e+06	2.680593e+06
141	3.342398e+06	3.081863e+06	2.042909e+06	3.021118e+06	3.383297e+06
142	3.044682e+06	3.011622e+06	1.433938e+06	3.279236e+06	2.530878e+06
143	8.377462e+05	1.717941e+06	1.922971e+06	2.589775e+06	1.650015e+06
144	1.628653e+06	1.689870e+06	2.312288e+06	4.155710e+05	1.063017e+06
145	1.703401e+06	7.639922e+05	1.536706e+06	1.844322e+06	1.506772e+06
146	1.925596e+06	8.616982e+05	1.991873e+06	8.246356e+05	1.943097e+06
147	1.666588e+06	1.894824e+06	2.365369e+06	2.267865e+06	1.851782e+06
148	2.118920e+06	2.936369e+06	2.818752e+06	2.153094e+06	2.658955e+06
149	2.371202e+06	3.763124e+06	2.848431e+06	4.933122e+06	3.059678e+06
150	4.845032e+06	3.958307e+06	3.098324e+06	3.262334e+06	4.230167e+06
151	4.706391e+06	4.544380e+06	4.385693e+06	3.310231e+06	4.301358e+06
152	2.795550e+06	3.867872e+06	4.233692e+06	3.553314e+06	3.448025e+06
153	1.575804e+06	2.776719e+06	3.454879e+06	3.311910e+06	1.201705e+06
154	1.513454e+06	1.000106e+06	4.405366e+06	2.320775e+06	2.615717e+06
155	9.887691e+05	1.563216e+06	2.049156e+06	2.678961e+06	1.614530e+06
156	1.164583e+06	1.550765e+06	2.296108e+06	5.356994e+04	3.018156e+06
157	2.015101e+06	2.276818e+06	1.357579e+06	2.628560e+06	3.151168e+06
158	1.714157e+06	1.551410e+06	1.924843e+06	1.790975e+06	1.794120e+06
159	2.079675e+06	2.852097e+06	1.465272e+06	2.534748e+06	3.669570e+06
160	4.070222e+06	3.140814e+06	4.318668e+06	3.428373e+06	3.537041e+06
161	3.391027e+06	2.909019e+06	3.740180e+06	4.074742e+06	4.870462e+06
162	4.050339e+06	4.335146e+06	2.832466e+06	3.638398e+06	4.183376e+06
163	3.669864e+06	4.173051e+06	3.806105e+06	4.585868e+06	3.671734e+06
164	3.468932e+06	3.637583e+06	3.929346e+06	4.256052e+06	3.504957e+06
165	3.324447e+06	3.253861e+06	3.285325e+06	2.918858e+06	2.413132e+06
166	2.664623e+06	3.327029e+06	3.233517e+06	3.135752e+06	1.586334e+06

167	3.196549e+06	2.266684e+06	2.299905e+06	5.711067e+05	2.387854e+06
168	1.365036e+06	1.156239e+06	1.680680e+06	2.396227e+06	1.797920e+06
169	1.236555e+06	2.514554e+06	1.962176e+06	2.041827e+06	2.160476e+06
170	3.017739e+06	2.385960e+06	2.864385e+06	2.713309e+06	1.870549e+06
171	3.182879e+06	3.568905e+06	3.478559e+06	3.931470e+06	2.084932e+06
172	2.570731e+06	4.145917e+06	4.629504e+06	2.915913e+06	4.049385e+06
173	3.707525e+06	3.903840e+06	5.159660e+06	4.213099e+06	3.182290e+06
174	4.397857e+06	5.284733e+06	5.547673e+06	3.866191e+06	4.943974e+06
175	3.670275e+06	4.341013e+06	3.669497e+06	4.146723e+06	4.478433e+06
176	3.313727e+06	3.867669e+06	4.029457e+06	3.990615e+06	4.223397e+06
177	3.280591e+06	3.921964e+06	2.781439e+06	3.717653e+06	1.274446e+06
178	2.406739e+06	2.045309e+06	2.351521e+06	2.229758e+06	3.109956e+06
179	5.708205e+05	2.422491e+06	2.176180e+06	2.172124e+06	1.834191e+06
180	9.781297e+05	1.956350e+06	1.463875e+06	3.074779e+06	1.356892e+06
181	2.611659e+06	2.240869e+06	2.560270e+06	2.588274e+06	1.574010e+06
182	3.880151e+06	2.025894e+06	1.411495e+06	1.642136e+06	3.719469e+06
183	2.040682e+06	3.779991e+06	2.926853e+06	3.656959e+06	1.956195e+06
184	3.664246e+06	4.735763e+06	3.661847e+06	1.461496e+06	5.776704e+06
185	4.461630e+06	3.973090e+06	5.319459e+06	3.853842e+06	4.166371e+06
186	5.441376e+06	5.274911e+06	1.534238e+06	3.436425e+06	6.390966e+06
187	5.035339e+06	5.508565e+06	3.691005e+06	4.976484e+06	3.559346e+06

	5	6	7	8	9 \
135	2.618520e+06	2.104780e+06	1.259076e+06	4.230476e+06	1.544629e+06
136	2.325539e+06	3.085171e+06	2.493622e+06	3.256742e+06	3.882763e+06
137	5.029312e+06	4.769657e+06	5.009486e+06	2.674633e+06	3.750097e+06
138	3.369481e+06	5.152285e+06	3.675967e+06	4.308286e+06	3.163095e+06
139	3.278536e+06	3.230391e+06	4.463234e+06	4.586113e+06	3.859089e+06
140	4.504442e+06	3.405415e+06	2.116856e+06	2.990616e+06	2.999035e+06
141	2.924192e+06	2.956588e+06	3.671614e+06	1.494783e+06	1.836759e+06
142	2.376213e+06	3.880594e+06	1.707089e+06	4.007833e+06	4.148179e+06
143	2.313954e+06	6.702787e+05	2.263710e+06	7.159619e+05	1.000939e+06
144	6.733330e+05	1.978148e+06	1.108180e+06	5.804820e+05	1.256867e+06
145	2.174379e+06	5.897157e+05	2.414002e+06	1.610180e+06	2.310242e+06
146	2.090382e+06	5.110459e+05	4.832489e+05	1.157183e+06	9.602217e+05
147	1.006567e+06	1.033387e+06	1.255143e+06	1.731781e+06	1.697964e+06

148	2.292535e+06	1.973795e+06	3.441485e+06	3.969003e+06	2.971837e+06
149	4.074589e+06	1.534238e+06	4.716773e+06	2.212990e+06	4.361283e+06
150	3.340280e+06	3.205673e+06	4.104496e+06	5.157309e+06	4.098779e+06
151	4.089345e+06	4.720426e+06	4.754596e+06	4.632874e+06	4.664794e+06
152	3.941023e+06	3.898825e+06	3.973844e+06	3.149702e+06	4.056190e+06
153	3.024752e+06	3.709567e+06	4.122015e+06	4.811366e+06	2.736874e+06
154	2.275384e+06	3.339704e+06	2.473850e+06	2.222486e+06	3.106873e+06
155	9.947298e+05	2.035058e+06	1.959265e+06	1.875974e+06	2.182813e+06
156	2.941836e+05	2.526695e+06	8.284966e+05	9.316478e+05	1.792037e+06
157	1.549420e+06	8.588931e+05	1.987846e+06	1.318064e+06	1.605402e+06
158	1.553032e+06	2.421651e+06	2.133261e+06	1.673440e+06	1.338269e+06
159	2.116308e+06	1.081369e+06	2.032645e+06	3.214995e+06	2.640171e+06
160	4.120107e+06	2.916295e+06	2.457616e+06	3.803765e+06	2.281224e+06
161	4.212351e+06	3.374392e+06	5.687910e+06	4.221940e+06	5.203304e+06
162	4.308097e+06	2.530730e+06	5.226866e+06	4.265972e+06	3.945992e+06
163	5.256596e+06	5.062527e+06	4.622235e+06	4.449454e+06	3.747855e+06
164	5.383403e+06	4.094900e+06	3.130070e+06	3.862311e+06	3.219363e+06
165	3.213810e+06	3.776360e+06	2.760730e+06	3.549659e+06	4.601920e+06
166	2.335161e+06	2.387080e+06	2.271328e+06	1.409571e+06	1.906222e+06
167	1.785236e+06	2.602938e+06	1.917983e+06	1.174143e+06	2.329998e+06
168	6.132781e+05	1.408330e+06	1.872025e+06	1.852334e+06	1.735874e+06
169	1.858277e+06	1.252171e+06	1.843983e+06	2.387894e+06	2.361027e+06
170	2.163641e+06	2.218839e+06	2.092033e+06	1.621194e+06	1.593350e+06
171	2.472209e+06	2.642430e+06	1.641479e+06	2.800116e+06	4.141765e+06
172	4.012022e+06	3.992403e+06	3.279274e+06	4.970665e+06	4.491694e+06
173	4.081908e+06	5.071460e+06	3.001040e+06	4.417163e+06	5.282826e+06
174	4.038660e+06	4.828140e+06	4.568100e+06	4.071838e+06	5.749082e+06
175	4.641788e+06	4.896353e+06	4.006510e+06	5.851649e+06	3.930356e+06
176	3.274106e+06	3.987891e+06	4.833849e+06	4.305178e+06	3.756722e+06
177	3.211056e+06	4.279275e+06	2.475286e+06	4.033685e+06	3.185324e+06
178	3.278174e+06	2.399671e+06	2.801016e+06	3.542705e+06	1.699437e+06
179	2.005489e+06	1.470856e+06	1.467178e+06	1.573000e+06	1.157637e+06
180	1.929647e+06	2.537638e+06	2.781725e+06	1.352620e+06	2.160960e+06
181	2.886744e+06	2.408824e+06	2.042345e+06	3.096768e+06	2.649459e+06
182	2.654576e+06	2.600273e+06	3.859370e+06	2.579988e+06	3.359202e+06
183	2.763670e+06	3.248688e+06	2.400745e+06	3.367157e+06	3.365808e+06

184	3.877741e+06	4.100735e+06	4.379793e+06	4.054990e+06	4.970023e+06
185	3.944140e+06	3.924806e+06	5.238779e+06	4.989771e+06	5.396375e+06
186	3.890398e+06	4.662777e+06	6.200987e+06	4.472455e+06	5.773024e+06
187	2.906376e+06	5.409892e+06	4.865319e+06	3.475298e+06	4.100316e+06

	...	90	91	92	93 \
135	...	2.864551e+06	2.221963e+06	1.565728e+06	2.107454e+06
136	...	2.485057e+06	3.447043e+06	3.999524e+06	3.125504e+06
137	...	4.729478e+06	3.956351e+06	3.455516e+06	3.049939e+06
138	...	2.901231e+06	4.046964e+06	4.139045e+06	2.934299e+06
139	...	3.078517e+06	3.788863e+06	3.190550e+06	4.661477e+06
140	...	4.163084e+06	3.340772e+06	2.632619e+06	2.873845e+06
141	...	1.661493e+06	2.505778e+06	2.682928e+06	3.473590e+06
142	...	2.092369e+06	1.555587e+06	1.372346e+06	1.803598e+06
143	...	2.020524e+06	7.181088e+05	9.135433e+05	1.118454e+06
144	...	1.188156e+06	1.642477e+06	1.715637e+06	1.442486e+06
145	...	1.455262e+06	3.079592e+05	1.025363e+06	9.813744e+05
146	...	2.042178e+06	1.273602e+06	1.707204e+06	2.714625e+05
147	...	2.356944e+06	3.064681e+06	7.233886e+05	2.102902e+06
148	...	2.908871e+06	2.095798e+06	1.306013e+06	3.012198e+06
149	...	3.546248e+06	3.680395e+06	3.792584e+06	4.255631e+06
150	...	4.227291e+06	4.214623e+06	3.897037e+06	2.717535e+06
151	...	4.932941e+06	3.543813e+06	3.218005e+06	4.376503e+06
152	...	3.311617e+06	3.542617e+06	4.128512e+06	4.062781e+06
153	...	2.106883e+06	2.828638e+06	3.636274e+06	2.992053e+06
154	...	2.940293e+06	1.932563e+06	2.491805e+06	2.805330e+06
155	...	3.124387e+06	6.441527e+05	2.024331e+06	7.473791e+05
156	...	1.544629e+06	2.132494e+06	1.128367e+06	1.521325e+06
157	...	8.249379e+05	3.285678e+05	1.499554e+06	9.303922e+05
158	...	9.152549e+05	2.246970e+06	2.270269e+06	1.384849e+06
159	...	3.308457e+06	3.502832e+06	2.291334e+06	2.462555e+06
160	...	3.885292e+06	3.183482e+06	3.100723e+06	3.907654e+06
161	...	3.348861e+06	4.168720e+06	3.802079e+06	5.231186e+06
162	...	3.398759e+06	4.197985e+06	5.467152e+06	2.826478e+06
163	...	5.430699e+06	4.082536e+06	4.254400e+06	4.005864e+06
164	...	4.334136e+06	2.704445e+06	4.380638e+06	2.938286e+06

165	...	2.868001e+06	3.193414e+06	2.558941e+06	3.259656e+06
166	...	2.924039e+06	3.665172e+06	2.602089e+06	2.165595e+06
167	...	1.896527e+06	2.695218e+06	2.058117e+06	1.596089e+06
168	...	2.130146e+06	2.068986e+06	1.541457e+06	1.830397e+06
169	...	2.367232e+06	2.093947e+06	1.275203e+06	2.663291e+06
170	...	2.203552e+06	2.214264e+06	2.799217e+06	2.641977e+06
171	...	2.669923e+06	3.156179e+06	3.072248e+06	3.109277e+06
172	...	3.777809e+06	3.641451e+06	4.709406e+06	3.168339e+06
173	...	5.236821e+06	4.648322e+06	4.297408e+06	4.240411e+06
174	...	5.459067e+06	4.869243e+06	3.520909e+06	5.806217e+06
175	...	5.080147e+06	4.341584e+06	3.128802e+06	3.767311e+06
176	...	2.822690e+06	3.649910e+06	3.882830e+06	2.970616e+06
177	...	4.022129e+06	2.825758e+06	3.170700e+06	2.903713e+06
178	...	1.582142e+06	3.818940e+06	3.235807e+06	2.238570e+06
179	...	1.410889e+06	2.314821e+06	1.578912e+06	1.452149e+06
180	...	2.504098e+06	3.531708e+05	1.605428e+06	2.399265e+06
181	...	1.886362e+06	1.977967e+06	2.761584e+06	2.044048e+06
182	...	2.615750e+06	3.380518e+06	2.139218e+06	3.258368e+06
183	...	3.729938e+06	2.867719e+06	3.578025e+06	3.193651e+06
184	...	3.710433e+06	4.330053e+06	4.586146e+06	5.419359e+06
185	...	5.100052e+06	5.137445e+06	4.433816e+06	4.507270e+06
186	...	5.573198e+06	4.967812e+06	4.415039e+06	1.555021e+06
187	...	4.259442e+06	4.351658e+06	4.548865e+06	3.638929e+06

	94	95	96	97	98 \
135	1.772772e+06	1.958489e+06	2.578940e+06	2.222178e+06	2.981809e+06
136	3.179572e+06	3.966108e+06	3.261489e+06	3.170810e+06	2.843275e+06
137	3.469056e+06	4.170296e+06	3.875460e+06	4.424614e+06	3.717474e+06
138	3.305127e+06	4.333013e+06	3.335281e+06	3.233812e+06	2.930441e+06
139	2.778953e+06	6.068956e+06	3.896791e+06	5.734238e+06	4.324608e+06
140	3.316072e+06	4.244280e+06	2.594844e+06	2.774825e+06	3.430666e+06
141	2.801095e+06	3.908618e+06	1.707509e+06	5.122523e+06	4.467774e+06
142	3.926882e+06	1.066699e+06	2.501360e+06	2.289457e+06	1.339410e+06
143	4.813800e+05	2.115882e+06	1.673482e+06	1.606979e+06	1.123976e+06
144	5.558998e+05	2.867570e+06	7.901626e+05	1.878389e+06	1.903038e+06
145	1.408888e+06	1.570248e+06	1.850320e+06	1.784400e+06	2.053709e+06

146	1.799171e+06	6.303536e+05	1.900420e+06	5.300372e+05	3.538981e+05
147	1.852472e+06	5.447937e+05	8.558717e+05	1.945614e+06	2.052472e+06
148	2.825013e+06	2.989981e+06	1.948407e+06	2.128057e+06	1.799758e+06
149	2.782983e+06	3.140534e+06	3.302080e+06	3.082810e+06	4.464590e+06
150	3.311672e+06	3.884713e+06	4.337169e+06	3.950097e+06	3.694448e+06
151	4.686511e+06	4.513322e+06	3.378161e+06	3.795918e+06	4.791060e+06
152	3.326692e+06	4.609854e+06	4.940026e+06	2.494514e+06	3.314622e+06
153	3.273246e+06	3.381411e+06	3.631710e+06	2.066708e+06	3.373466e+06
154	4.199140e+06	2.257970e+06	3.327937e+06	1.936473e+06	2.112080e+06
155	1.774035e+06	3.082242e+06	1.178426e+06	1.893423e+06	1.154714e+06
156	7.916560e+05	2.812918e+06	1.314169e+06	1.328343e+06	1.669509e+06
157	2.449649e+06	1.956388e+06	2.847017e+06	2.205135e+06	1.376488e+06
158	1.454958e+06	2.448379e+06	2.163460e+06	2.622743e+06	1.931219e+06
159	2.636010e+06	4.664754e+06	2.815764e+06	3.044218e+06	3.166077e+06
160	3.043390e+06	4.399414e+06	2.949181e+06	2.713439e+06	4.845897e+06
161	4.767419e+06	2.966674e+06	3.482144e+06	3.879227e+06	3.640215e+06
162	3.154649e+06	3.693699e+06	2.925631e+06	3.959780e+06	3.757492e+06
163	5.316104e+06	4.955544e+06	2.973394e+06	4.061780e+06	5.517339e+06
164	5.089039e+06	3.496844e+06	4.924108e+06	4.048389e+06	2.967166e+06
165	3.999624e+06	3.920188e+06	4.918105e+06	4.259774e+06	2.934074e+06
166	2.270663e+06	3.308480e+06	2.381007e+06	1.986950e+06	2.612150e+06
167	1.899659e+06	2.274033e+06	2.015261e+06	4.117088e+05	1.747997e+06
168	1.787309e+06	2.332204e+06	2.726332e+06	1.229904e+06	1.236219e+06
169	2.152389e+06	1.010379e+06	2.105364e+06	1.511715e+06	2.448006e+06
170	2.828967e+06	2.337290e+06	2.328804e+06	2.029515e+06	1.523762e+06
171	3.144547e+06	2.361968e+06	2.720126e+06	1.797059e+06	2.231589e+06
172	3.182858e+06	3.149049e+06	4.865716e+06	3.454724e+06	4.487438e+06
173	3.630287e+06	3.069258e+06	4.776887e+06	6.466048e+06	4.325091e+06
174	4.530954e+06	5.255615e+06	4.220696e+06	5.295700e+06	3.611865e+06
175	4.498143e+06	4.999116e+06	3.690948e+06	3.026806e+06	5.225512e+06
176	4.541472e+06	4.064338e+06	4.170901e+06	4.140542e+06	3.408459e+06
177	2.678395e+06	2.596397e+06	3.849306e+06	3.078355e+06	3.591225e+06
178	2.732266e+06	3.150043e+06	3.048854e+06	1.932190e+06	1.684301e+06
179	1.455338e+06	1.683288e+06	2.475286e+06	1.436512e+06	2.091346e+06
180	1.827265e+06	1.672231e+06	2.174106e+06	1.817965e+06	2.236672e+06
181	2.382892e+06	2.678680e+06	2.879140e+06	1.637847e+06	2.076025e+06

182	2.862496e+06	4.222474e+06	3.437681e+06	2.301313e+06	2.477753e+06
183	4.105574e+06	2.939045e+06	2.152782e+06	4.978580e+06	2.941653e+06
184	3.400452e+06	4.412740e+06	4.848315e+06	4.574845e+06	5.577736e+06
185	3.118636e+06	4.037521e+06	1.066613e+06	4.028681e+06	4.466273e+06
186	4.534759e+06	5.522557e+06	4.139882e+06	6.185238e+06	4.340984e+06
187	5.393121e+06	3.474990e+06	4.584758e+06	4.083012e+06	3.974816e+06

	99
135	2.777203e+06
136	2.984013e+06
137	4.494674e+06
138	3.929972e+06
139	5.238138e+06
140	4.043853e+06
141	3.402792e+06
142	2.388316e+06
143	1.712428e+06
144	1.212637e+06
145	9.523396e+05
146	1.445304e+06
147	2.735229e+06
148	2.555660e+06
149	4.495428e+06
150	4.969909e+06
151	3.670238e+06
152	4.132882e+06
153	4.201677e+06
154	3.543772e+06
155	2.023730e+06
156	1.300110e+06
157	1.379594e+05
158	2.302619e+06
159	2.705640e+06
160	3.241808e+06
161	4.776328e+06
162	3.769377e+06

163	5.130408e+06
164	3.775518e+06
165	3.844303e+06
166	3.323599e+06
167	2.254566e+06
168	2.170098e+06
169	2.418847e+06
170	2.022872e+06
171	2.971848e+06
172	5.031764e+06
173	5.028817e+06
174	4.926086e+06
175	6.537158e+06
176	3.106908e+06
177	3.317074e+06
178	3.988370e+06
179	1.451984e+06
180	1.490442e+06
181	3.051140e+06
182	1.377519e+06
183	3.264892e+06
184	2.917325e+06
185	5.186910e+06
186	5.671421e+06
187	3.816274e+06

CURRICULUM VITAE

Salim Jibrin DANBATTA

[redacted]

[redacted]

[redacted]

[redacted]

[redacted]

[redacted]

- [redacted]
- [redacted] data using Polynomial-[redacted]

Proceedings:

1. Danbatta, S. J., & Varol, A. (2021a). Modeling and Forecasting of Tourism Time Series Data using ANN-Fourier Series Model and Monte Carlo Simulation. In 2021 9th International

Symposium on Digital Forensics and Security (ISDFS) (pp. 1–6). IEEE. https://doi.org/10.1109/ISDFS52919.2021.9486325

2. Danbatta, S. J., & Varol, A. (2020). Predicting Student's Final Graduation CGPA Using Data Mining and Regression Methods: A Case Study of Kano Informatics Institute. In 2020 8th International Symposium on Digital Forensics and Security (ISDFS) (pp. 1–7). IEEE. https://doi.org/10.1109/ISDFS49300.2020.9116389
3. Danbatta, S. J., & Varol, A. (2019). Comparison of Zigbee, Z-Wave, Wi-Fi, and Bluetooth Wireless Technologies Used in Home Automation. In 2019 7th International Symposium on Digital Forensics and Security (ISDFS) (pp. 1–5). IEEE. https://doi.org/10.1109/ISDFS.2019.8757472