PREDICTING TRAVEL MOVEMENTS OF SHORT-TERM VISITORS

Final Project – Time Series Analysis (MATH1318)

Contents

Tables & Figures

1. Introduction

Visiting Australia means encountering opportunities; with its young, multicultural, and ever-growing economy. Australia also attracts humankind with its impeccable coastlines, charming cities and breath-taking landscapes, the immense country is a must on anyone's travel bucket list. We are glad be living here and thus making the most of this chance, the analysis and prediction of travel movements of short-term visitors arriving in Australia has been attempted through this report.

Administrative information on people arriving in Australia is collected via various processing systems, passport documents, visa information, and incoming passenger cards. In order to understand and predict the number of visitors Australia welcomes, the administrative data (Statistics, 2019) collected by the Australian Government's Department of Home Affairs has been accessed.

2. Methodology

A high-level view of the methodology used is discussed in this section.

1. The dataset is examined and visualized to summarize its characteristics like trend, variance, seasonality, moving average and autoregressive behavior
2. The stationarity of the series is ensured (by transformations, d order differencing, ADF test, etc.)
3. Depending upon the characteristics of time series, especially trend, variance, stationarity and seasonality, the candidate models are chosen from linear, quadratic, cosine, and cyclical trend models along with ARIMA and SARIMA
4. The seasonality aspect will be handled by finding the parameter estimates for seasonal (P,D,Q) ARIMA model
5. The parameter values for models pertaining to ARIMA are identified using the following processes:
    a. ACF & PACF Plots
    b. EACF Matrix
    c. BIC Table
6. The models are fitted on the time series data using ML and CSS methods; and their significance and extent-of-fit are checked by comparing AIC and BIC scores
7. The overparameterized models, or overfitted models are then checked for their suitability
8. Coefficient tests are performed to test the significance of parameters estimates of shortlisted models, after checking for overfitting
9. The residuals of the best-fit models are examined for normality and resemblance to white noise
10. The model that is found to be optimal in the tests mentioned above will be chosen to give forecasts for the next 10 units of time, i.e. next 10 months

3. About Dataset

3.1 Data Description

The dataset consists of over 300 observations of monthly numbers of short-term arrivals in the country from January 1991 to November 2019. The total monthly visitor numbers range from 161 thousand to 1 million 57 thousand approximately, with a mean of 455 thousand arrivals.

Element	Value
Minimum	161,400
1st Quartile	343,450
Median	431,000
Mean	455,116
3rd Quartile	537,450
Maximum	1,057,900

Table 1: Summary of Time Series Data

3.2 Visual Analysis of Time Series

The data was converted into a time series and visualized against time, for understanding its features.

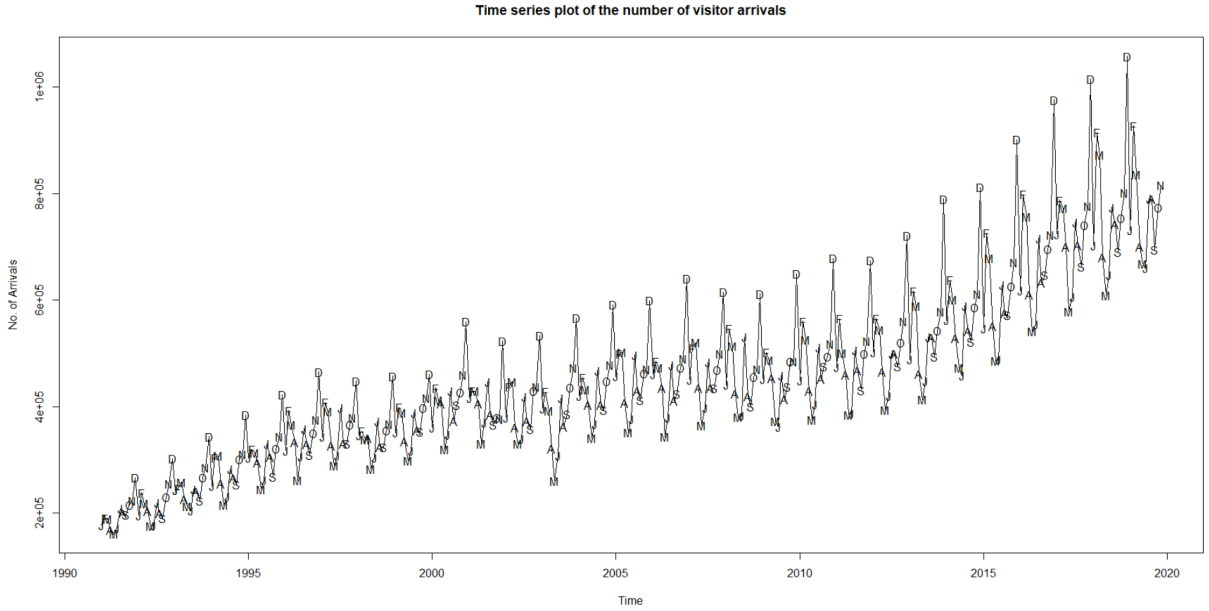

Figure 1: Time Series Plot of the number of visitor arrivals

From the time series plot, it can be observed that in a particular year, December experienced the highest number of arrivals, whereas May had the lowest. The following inferences were drawn from the time series plot:

- **Trend**: An upward trend can be seen in the plot which exhibits a moving average behavior
- **Variance**: Clear presence of change in variance
- **Seasonality**: Presence of seasonality

In order to understand the presence of correlation between consecutive observations, a **scatter plot** was obtained:

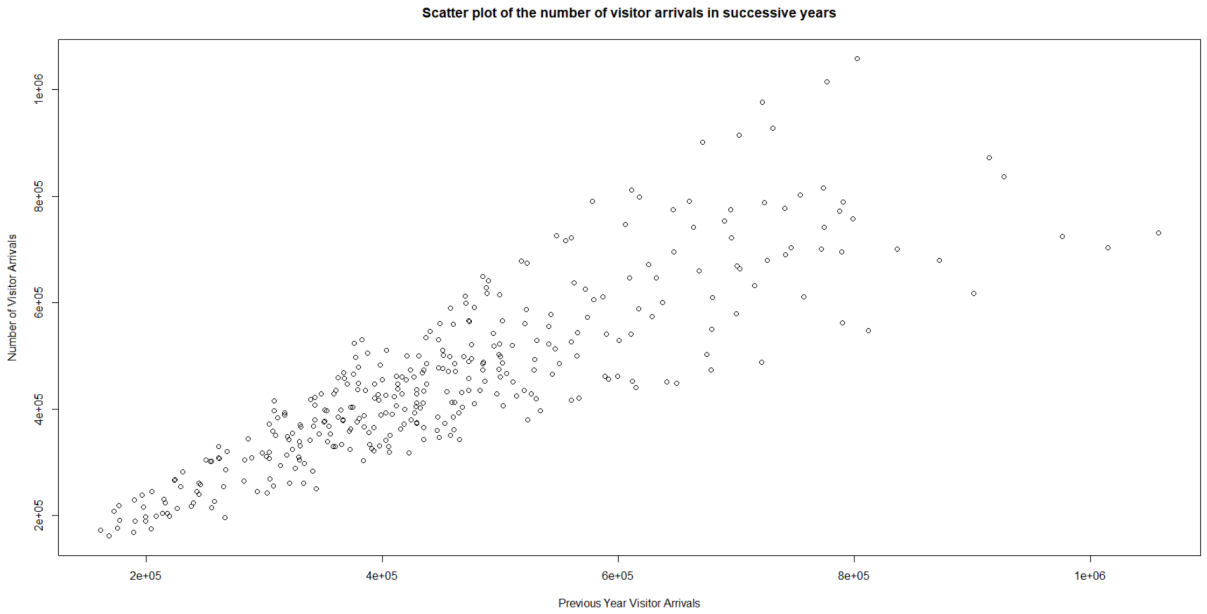

Figure 2: Scatter plot of the number of visitor arrivals in successive months

It is evident from the scatter plot above, that there is a strong presence of **upward trend**. The upward trend translates into a positive correlation between successive observations. Hence, the scatter plot confirms presence of **autoregressive behavior**. A strong correlation of **0.86** was observed between consecutive observations.

3.3 Checking for Seasonality and Stationarity

Before proceeding to model identification, the time series was checked for stationarity using an augmented Dickey-Fuller test, and the presence of seasonality was validated by examining its ACF & PACF plot.

3.3.1 ACF & PACF Plots

To verify the presence of seasonality, the ACF and PACF plot were obtained:

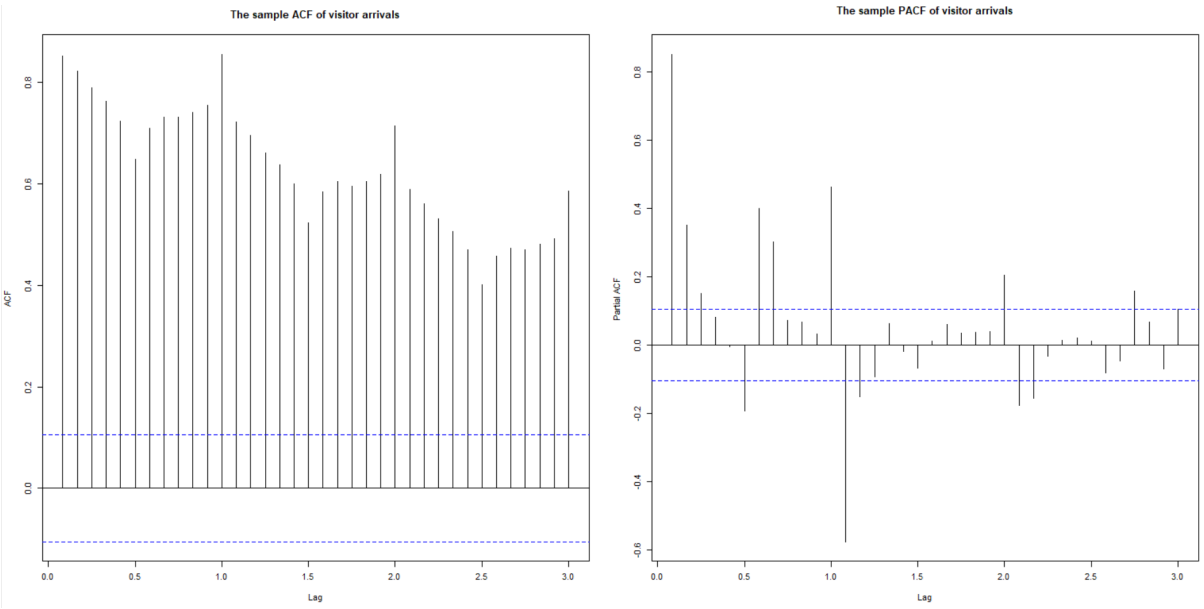

Figure 3: ACF & PACF plots of the time series

The wavy pattern of ACF-PACF plots confirm the presence of seasonality. It might be impacting extent of moving average by impacting the behavior of time series.

3.3.2 ADF Test

An augmented Dickey-Fuller test was performed to check if data was stationary:

```
Title:
 Augmented Dickey-Fuller Test

Test Results:
  PARAMETER:
    Lag Order: 14
  STATISTIC:
    Dickey-Fuller: 3.5393
  P VALUE:
    0.99
```

Figure 4: Stationarity Test for the time series

The **p-value (0.99)** was found to be greater than 5% significance level, hence failing to reject the null hypothesis stating non-stationarity. Hence, non-stationarity was assumed for the given time series.

4. Deterministic Trend Models

A strong positive correlation between consecutive values in the time series paired with the evident seasonality, might be the reason behind the perceived trend. For investigating deterministic trend modelling, 5 types of trend models were fitted to the concerned time series. The code for this task was executed in RStudio using TSA package.

4.1 Model Identification & Fitting

First, the linear, quadratic, cubic, seasonal, and harmonic trend models were simulated to understand how well they fit the given time series data. Seasonal and harmonic trend models were included in the investigation for factoring-in seasonality.

4.1.1 Linear Trend Model

The linear modelling was carried out using lm() function of TSA package.

The coefficient and intercept were both found to be significant. The intercept was found to be **-3.41×10^7**, while the coefficient was found to be **1.723×10^4**.

The p-value for F-statistic was found to be less than 5% significance level; hence the model can be assumed significant. The R squared value was found to be **0.76**.

A trend line corresponding to the linear model was plotted:

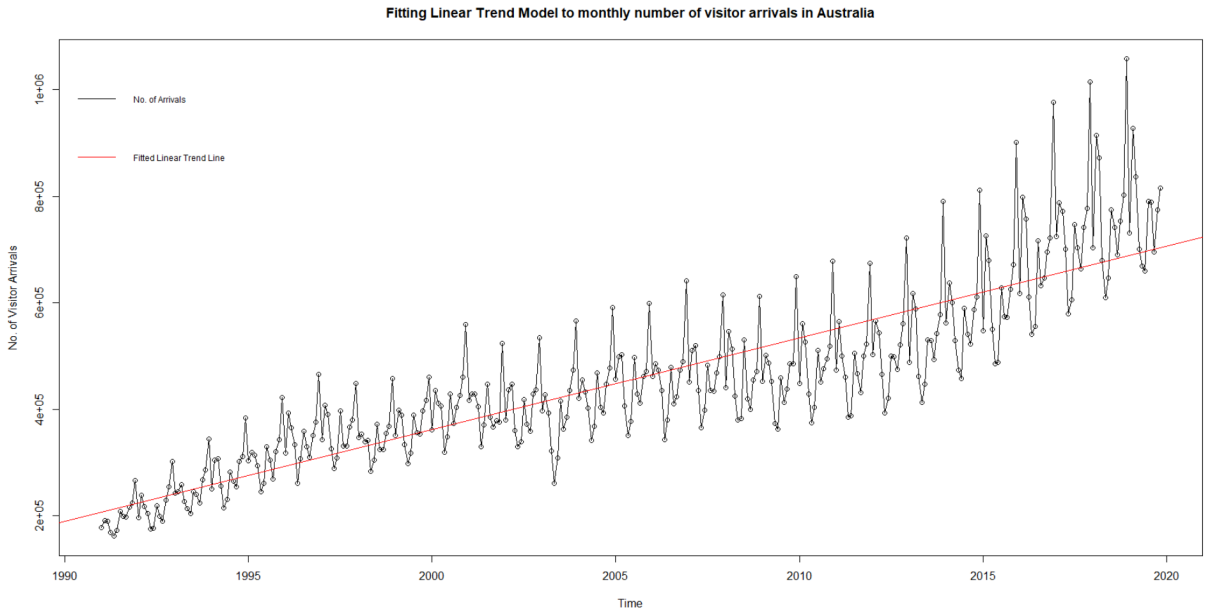

Figure 5: Fitted linear trend model

From the plot above, it was observed that the trendline of linear model failed to capture a substantial number of observations.

4.1.2 Quadratic Trend Model

The quadratic modelling was carried out using lm() function of TSA package.

The coefficients and intercept were all found to be significant. The intercept was found to be **1.242×10^9**, while the coefficients of t and t^2 were found to be **-1.256×10^6** and **317.4**.

The p-value was found to be less than 5% significance level; hence the model can be assumed significant. The R squared value was found to be **0.77,** lower than that of linear model.

A trend line corresponding to the quadratic model was plotted:

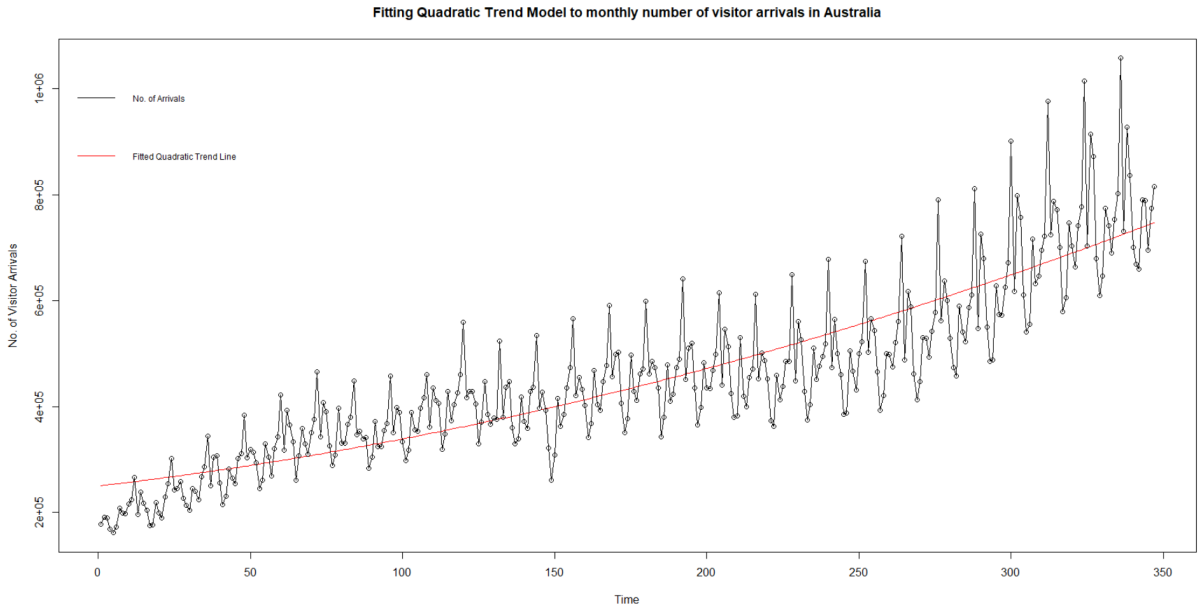

Figure 6: Fitted quadratic trend model

The trend line of quadratic model seems to follow the actual data more closely compared to linear model. However, it showed a weaker fit with a lower value of R-squared.

4.1.3 Cubic Trend Model

The cubic modelling was carried out using lm() function of TSA package.

The coefficients, t and t^2,and intercept were found to be significant. The intercept was found to be **1.242×10^9**, while the coefficients of t and t^2 were found to be **-1.256×10^6**, and **317.4**.

The p-value was found to be less than 5% significance level; hence the model can be assumed significant. The R squared value was found to be **0.77,** lower than that of linear model.

A trend line corresponding to the cubic model was plotted:

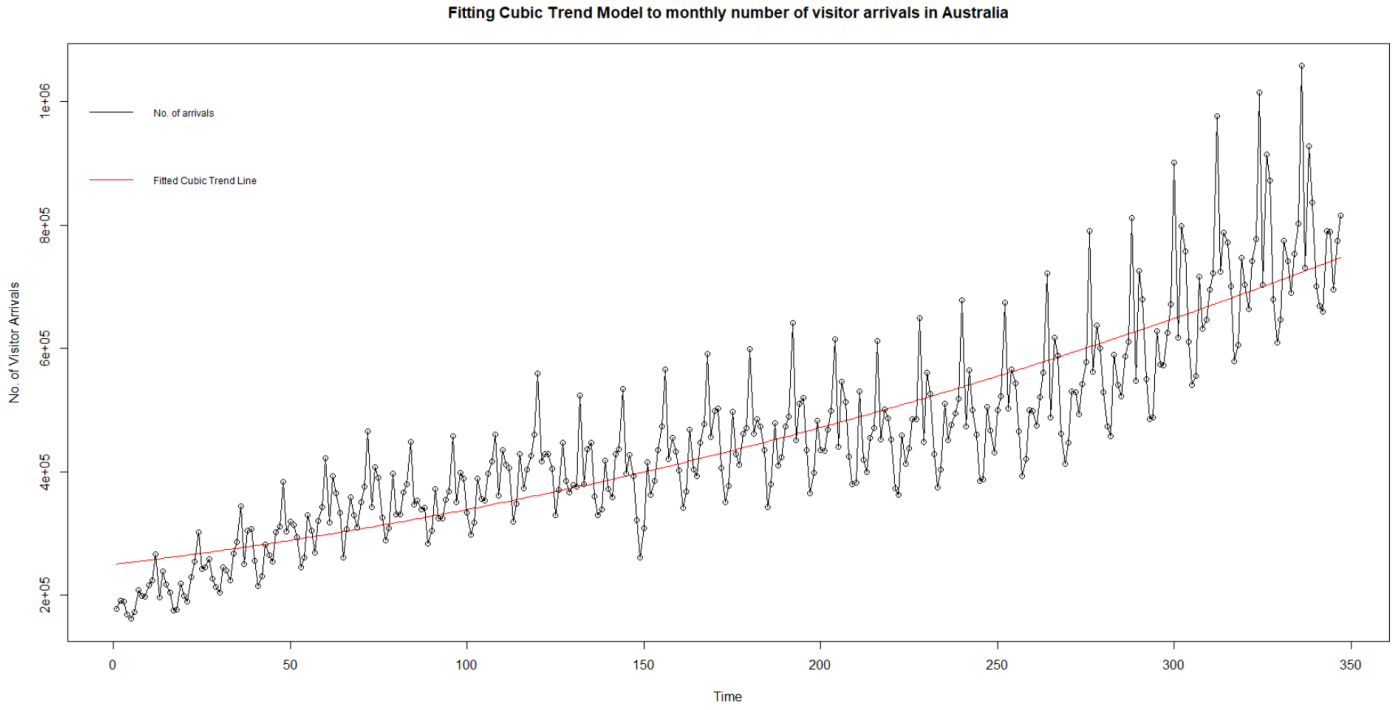

Figure 7: Fitted cubic trend model

The trend line of cubic model looks similar to that of quadratic model.

4.1.4 Seasonal Trend Model

The seasonal modelling was carried out using lm() function of TSA package.

All the coefficients were found to be significant. The p-value was found to be less than 5% significance level; hence the model can be assumed significant. The R squared value was found to be **0.89,** greater than that of linear model.

In this model, only the difference between November and December was found to be statistically significant at 5% significance level.

4.1.5 Harmonic Trend Model

The harmonic modelling was carried out using lm() function of TSA package.

The coefficients and intercept were all found to be significant. The intercept was found to be **455292**, while the coefficients of $\beta_1 cos(2\pi ft)$ and $\beta_2 sin(2\pi ft)$ were found to be **53356** and **-29519**.

The p-value was found to be smaller than 5% significance level; hence the model can be assumed significant.

A trend line corresponding to the harmonic model was plotted:

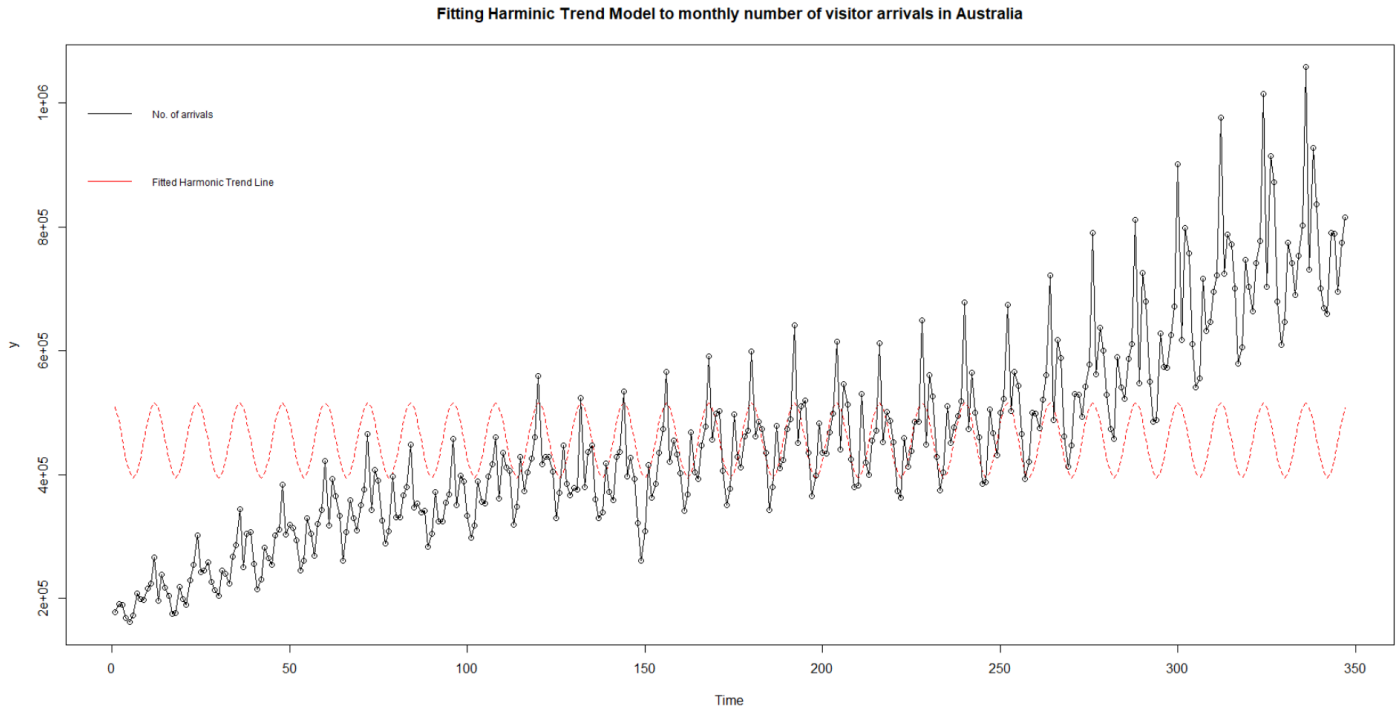

Figure 8: Fitted harmonic trend model

The harmonic trend model showed a weaker fit with the lowest value of R-squared(0.06).

4.2 Result

Summarizing the results of fitting deterministic trend models on the time series:

Trend Model	Significance of Intercept	Coefficients		Model p-value	Adj. R-squared	Model Significant?
		Significant	**Total**			
Linear	Yes	1	1	<0.05	0.7573	Yes
Quadratic	Yes	2	2	<0.05	0.771	Yes
Cubic	Yes	3	3	<0.05	0.771	Yes
Seasonal	No	12	12	<0.05	0.8966	Yes
Harmonic	Yes	2	2	<0.05	0.06249	Yes

Table 2: Summary of fitted deterministic trend models

All the models were found to be significant with some faults, but none of the deterministic trend models were found to capture the trend and magnitude of the given time series. This was confirmed from the visualization of fitted model on the concerned time series.

5. Model Building Strategy

To determine an appropriate model for the time series data the following multistage model-building strategy would be followed:

- Model Specification, (or identification).
- Model Fitting.
- Model Diagnostics.

5.1 Model Specification

Fitting the seasonal-deterministic-trend models, it is observed that the model fit is not suitable. So, we will now fit stochastic seasonal models. We follow the residual approach and specify the SARIMA models.

From the time-series plot of the original data in Figure 1, it is apparent that the series has changing variance. First, a log transformation of the series is taken to stabilise the changing variance. All the further approaches as carried on this transformed data series.

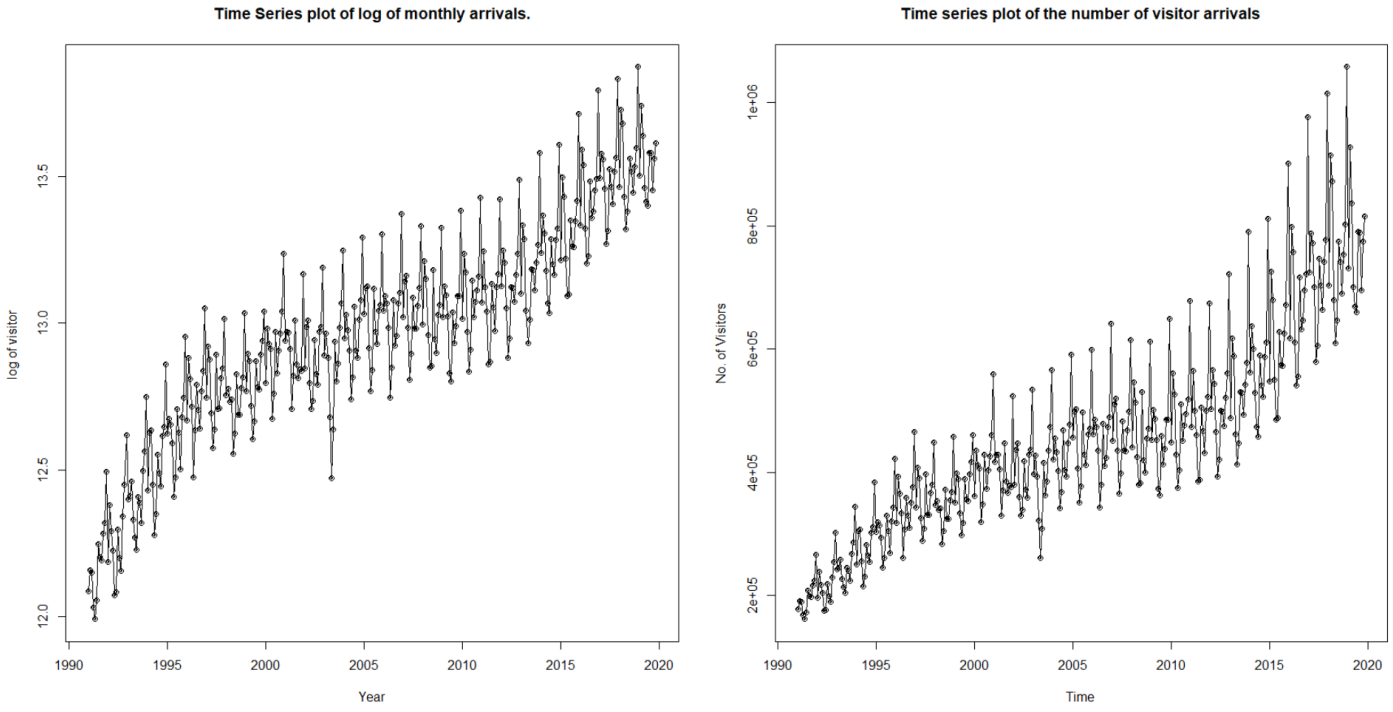

Figure 9: Original vs. Log Transformed Time Series

Observing Figure 3, the ACF and PACF plots of the original series we noticed persistent correlations at lags 12, 24 and so on indicating periodicity with period s = 12.

5.1.1 Specification of the seasonal part

Considering the SARIMA(p,d,q) x (P,D,Q) model we begin with specification of seasonal orders (P,D,Q). where D indicates the seasonal difference and P and Q refer to seasonal autoregressive and moving average parts respectively.

Fitting a plain model with first seasonal difference with order D = 1, we try to get rid of the seasonal trend effect. On inspecting the autocorrelation structure of the residuals, we find no significant lags. The PACF shows 2 significant lags, indicating SAR (1) and SAR (2) for the seasonal part i.e. P = 1 or 2.

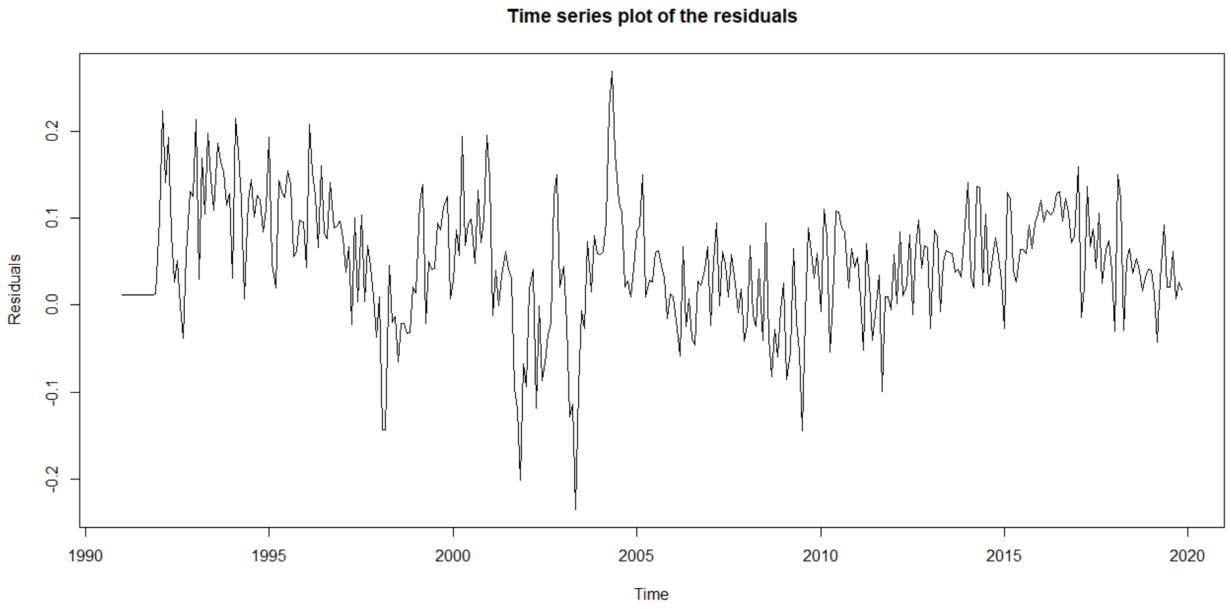

Figure 10: Time series plot for residuals

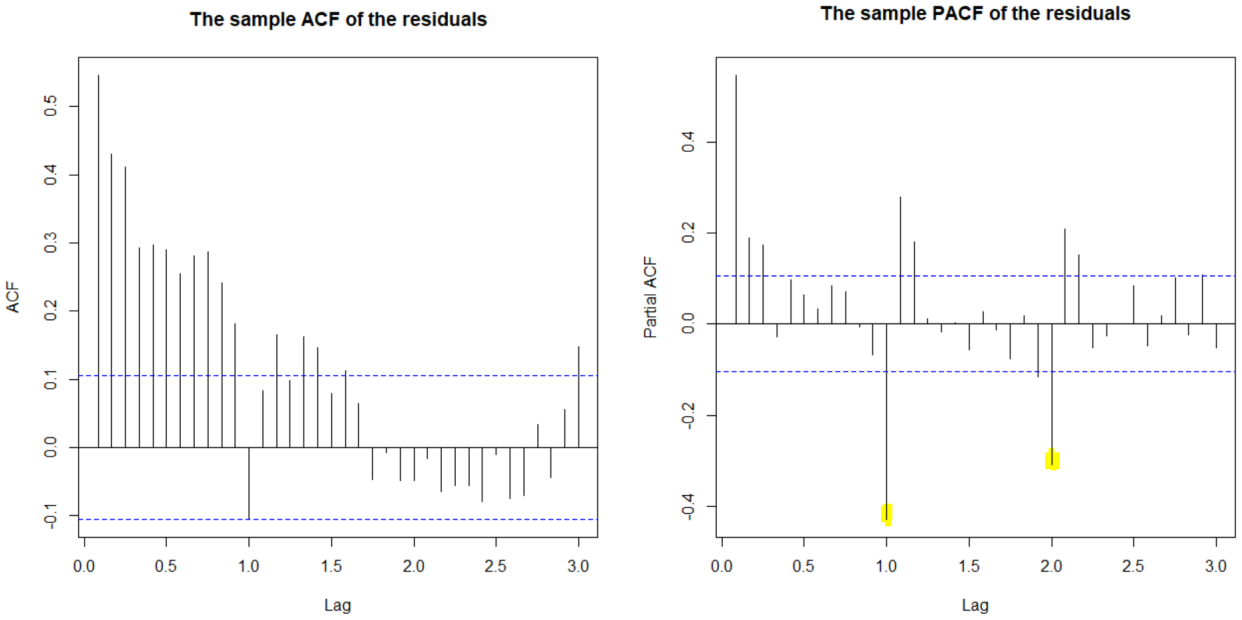

Figure 11: TS and ACF/PACF plot for SARIMA (0,0,0)X(0,1,0)

So, we add the SAR(1) and SAR(2) components and check if the effect of seasonal component on residuals still exist.

Though, significant correlations are still observed at seasonal lags 1 and 2 in PACF, we move forward with ordinary differencing remove the ordinary trend. Taking first ordinary difference, and inspecting the ACF and PACF plots of residuals, we observe that the there is no trend nor seasonality remaining. This is observed with SAR(2). The time series plot of the residuals also follows white noise.

The series is now stationary. This is also supported by the ADF test carried the residual obtained at this stage. With p-value less than 0.05 the test is statistically significant, and series is stationary.

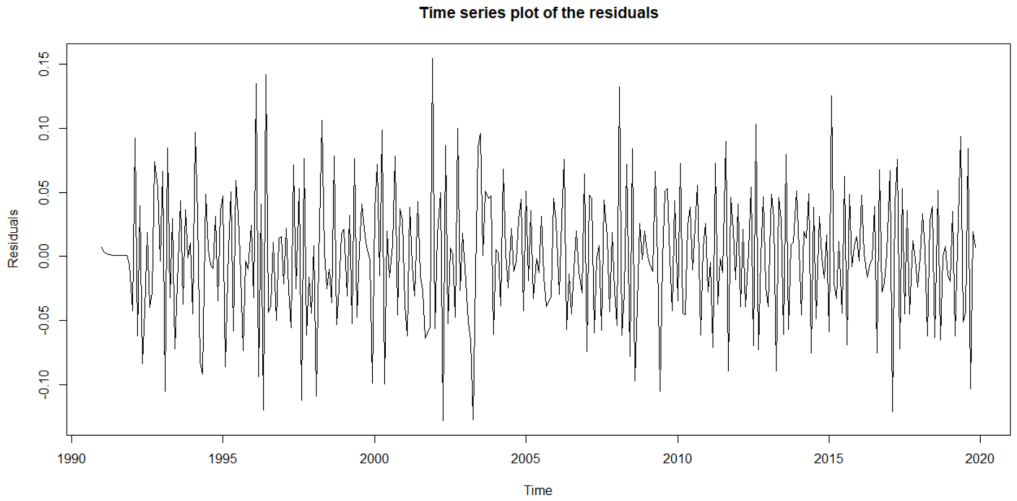

Figure 12: TS plot of plot for SARIMA (0,1,0)x(2,1,0)

The fixed components of our model at this stage are d= 1, P=2, D=1, Q=0

Model
SARIMA (p,1,q) x(2,1,0)

Table 3: Fixed components after examining specification of seasonal part

5.1.2 Specification of ordinary part

With no more seasonality or trend left in the series, we propose values for the ARIMA order p and q thereby, specifying a set of candidate models.

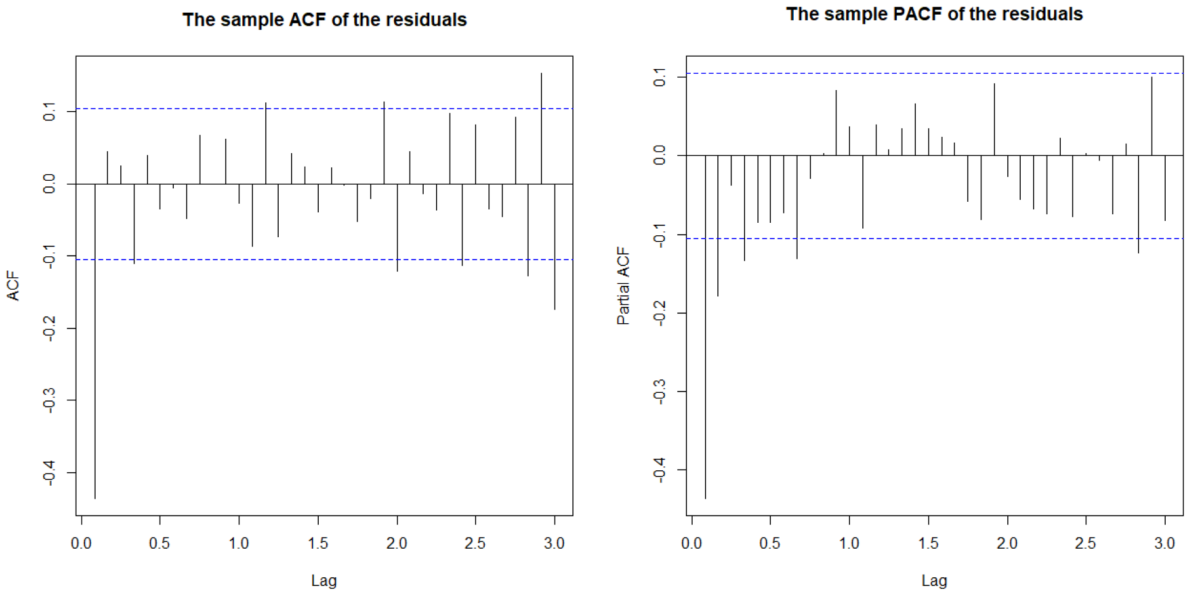

Figure 13: ACF-PACF plot of residuals

Models
SARIMA (p,1,q) x(2,1,0)

Table 4: Fixed components after examining specification of ordinary part

In this part, our aim would be to determine the values of p and q of ARIMA. From the previous section, by performing differencing on the time series with order 2, we have determined the value of d=2.As the time series data holds AR and MA component, to address this we would proceed further and fit Stochastic models. ARIMA class models would

be used to fit stochastic trend to the time series data. We would proceed with choosing appropriate values for the model parameters p, d and q by using the following model specification tools:

- *ACF and PACF*
- *Extended Autocorrelation Function*
- *Bayesian Information Criterion (BIC)*
- *Dickey Fuller unit root test*

Using the above procedures, we would fit the following models:

- *AR(p)*
- *MA(q)*
- *IMA (d, q)*
- *ARI (p, d)*
- *ARIMA (p, d, q)*

where:

p is the **number of Autoregressive terms.**
d is the **order of Differencing** needed for stationarity.
q is the **number of lagged forecast errors in the prediction equation.**

5.1.2.1 ACF and PACF

Firstly, using the ACF and PACF plots we determine the values of p and q for ARMA.

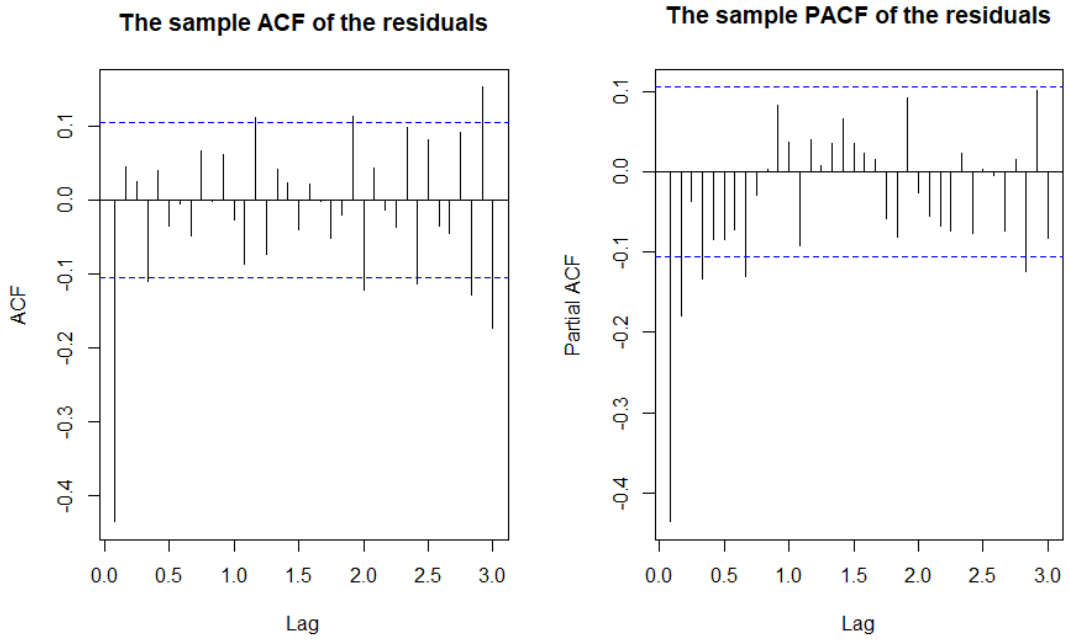

Figure 14: ACF & PACF Plots of residuals

As per Figure 14, we infer the following:
ACF*:* The plot seems to have sine damped decaying pattern with an apparent significant lag at 1 in the plot, lag 4 is barely significant thus we disregard it, as lag 2 and 3 are insignificant.

PACF*:* Lag 1,2 and 4 are significant.

To determine the final set of models we use one of the following approaches:

- ***Approach 1***- determines p and q based on the number of lags with significant autocorrelations.
- ***Approach 2***- determines p and q based on the "cut-off" of lag.

Using approach 1, we get p=1,2,3 and q=1. Thus, observing the ACF and PACF plots of the first differenced time series, in conclusion the set of candidate models are as follows:

Models
ARIMA (1,1,1)

ARIMA (2,1,1)
ARIMA (3,1,1)

Table 5: Candidate ARIMA Models from ACF-PACF Plots

5.1.2.2 Extended Autocorrelation Function

This is an effective method to identify the orders p and q of an ARMA (p, q) model.
The EACF table has two symbols: x and 0.
x: if sample correlation of AR residuals are significantly different from 0
0: if not.
For our interpretation, we observe 0s. Based on the inference of these values we would decide our models.
EACF matrix of an ARMA model for our time series is given below:

```
## AR/MA
##   0 1 2 3 4 5 6 7 8 9 10 11 12 13
## 0 x o o x o o o o o o  o  o  x
## 1 x x o x o o o o o o  o  o  o
## 2 x x x x o o o o o o  o  o  o
## 3 x x o o o o o o o o  o  o  o
## 4 x x o o o o o o o o  o  o  o
## 5 x x o x o x o o o o  o  o  x
## 6 x o o x o x o o o o  o  o  o
## 7 x x o x x x o x o o  o  o  o
```

The Extended Autocorrelation (EACF) method implies the existence of white noise behavior. We would consider the row corresponding to p = 0 and q=4 as the vertex. AR and MA orders corresponding to this vertex determines the order of ARMA (p, q), as our series is first differenced, we would include d=1.
Thus, the resulting set of possible models would be:

Models
ARIMA (0,1,4)
ARIMA (0,1,5)
ARIMA (1,1,4)
ARIMA (1,1,5)

Table 6: Candidate ARIMA models from EACF Matrix

However, adhering to the ***Principal of parsimony***, we eliminate model with more than 6 parameters, that is, we would choose the models which would require the smallest number of parameters to adequately represent the time series. Thus, the final set of candidate models would be as mentioned below:

Models
ARIMA (0,1,4)
ARIMA (0,1,5)
ARIMA (1,1,4)

Table 7: Final set of candidate ARIMA models from EACF Matrix

5.1.2.3 Bayesian Information Criterion (BIC)

Here, we would use Bayesian Information Criterion (BIC) method which is based on maximum likelihood estimation (MLE) to determine the orders of p and q. Orders of ARMA (p, q) model is determined by the orders specified by minimizing the BIC.

(BIC) or Schwartz's Bayesian Criterion is given as:

$$BIC = -2\log(\text{maximum likelihood}) + k\log(n)$$

For this purpose, we would examine a subset of ARMA models to derive at some tentative models. Best subset models can be summarized and determined by observing the pattern of time series lags and the error process.

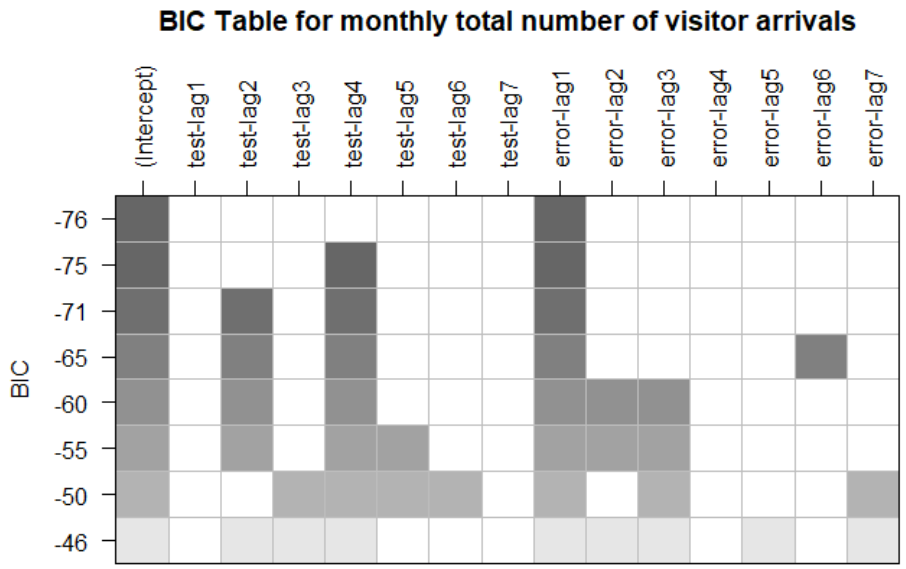

Figure 15: BIC Table for monthly number of visitor arrivals

In Figure 15, the models are sorted according to their BIC, with better models (lower BIC) placed in higher rows with darker shades.
Thus, our orders of p and q based on this table (row wise) is as follows:

- Model containing lag 1 of the error process; MA (1).
- Model containing lag at 4 of the observed time series and lag 1 of the error process; AR (4) and MA (1).
- Model containing lag at 2 and 4 of the observed time series and lag 1 of the error process; AR (2), AR (4) and MA (1).

Thus, using the shaded columns corresponding to AR (2), AR (4) and MA (1) coefficients, we include the following models mentioned in Table 15 (Note: All models are displayed in ARIMA format) in the set of candidate models.

Models
ARIMA (2,1,1)
ARIMA (4,1,1)

Table 8: Candidate ARIMA models from BIC Table

5.1.2.4 Final set of Candidate Models

Consolidated set of candidate models is mentioned below in Table 9 (Note: All models are displayed in ARIMA format)

Models
ARIMA (1,1,1)
ARIMA (2,1,1)
ARIMA (3,1,1)
ARIMA (0,1,4)
ARIMA (0,1,5)
ARIMA (1,1,4)
ARIMA (4,1,1)

Table 9: Final Set of Candidate Models for ARIMA

5.2 Model Fitting

As per Table 9, using our final set of candidate models, in this section, firstly, we would fit the models, analyze their respective residuals and based on this we would short list our set of candidate models for next process which involves sorting them based on AIC and BIC, here, we would choose a set of top performing models, overfit them respectively and perform parameter estimation on each model to acquire appropriate results. We would apply both maximum likelihood estimation and Conditional sum of squares (CSS) on each model, to derive at a best model , i.e. a model with approximately all significant coefficients, appropriate residuals and lowest AIC and BIC values.

5.2.1 Analyzing ACF and PACF of Residuals

5.2.1.1 SARIMA (1,1,1) x (2,1,0) - ML

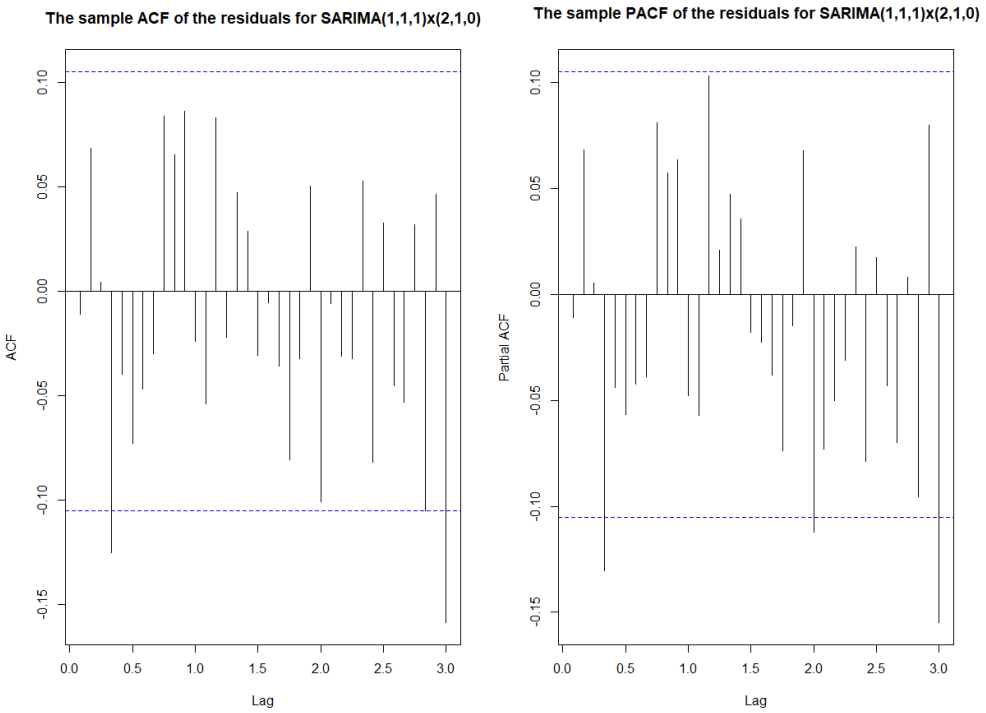

Figure 16: ACF-PACF Plots for SARIMA (1,1,1)x(2,1,0) - ML

We observe significant correlations in both the ACF and PACF plots, thus, as the residuals are not white noise, we reject this model.

5.2.1.2 SARIMA (1,1,1) x (2,1,0) - CSS

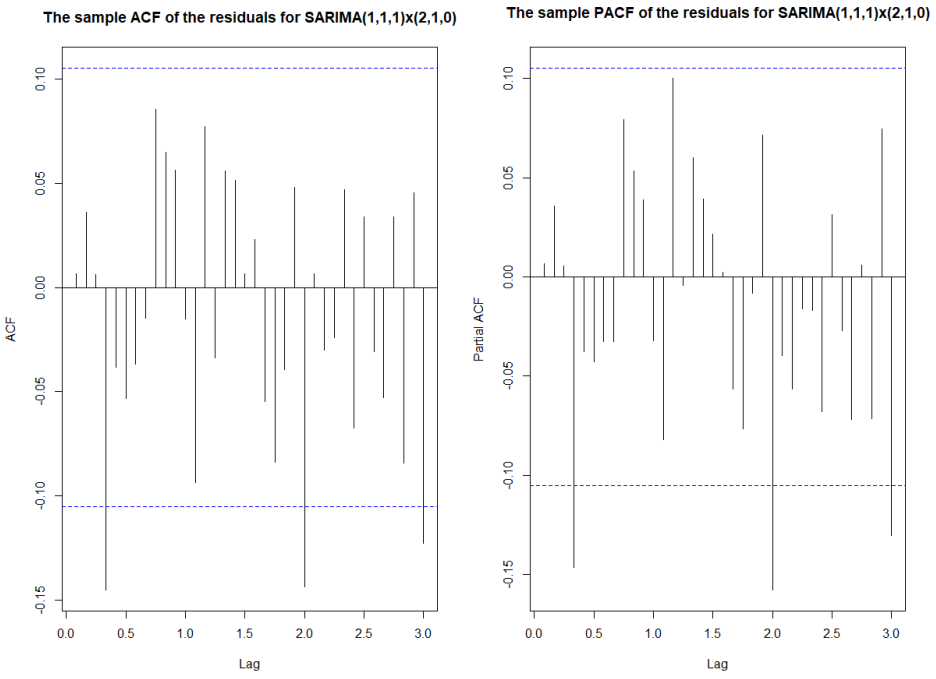

Figure 17: ACF-PACF Plots for SARIMA (1,1,1)x(2,1,0) - CSS

We observe significant correlations in both the ACF and PACF plots, thus, as the residuals are not white noise, we reject this model.

5.2.1.3 SARIMA (2,1,1) x (2,1,0) - CSS

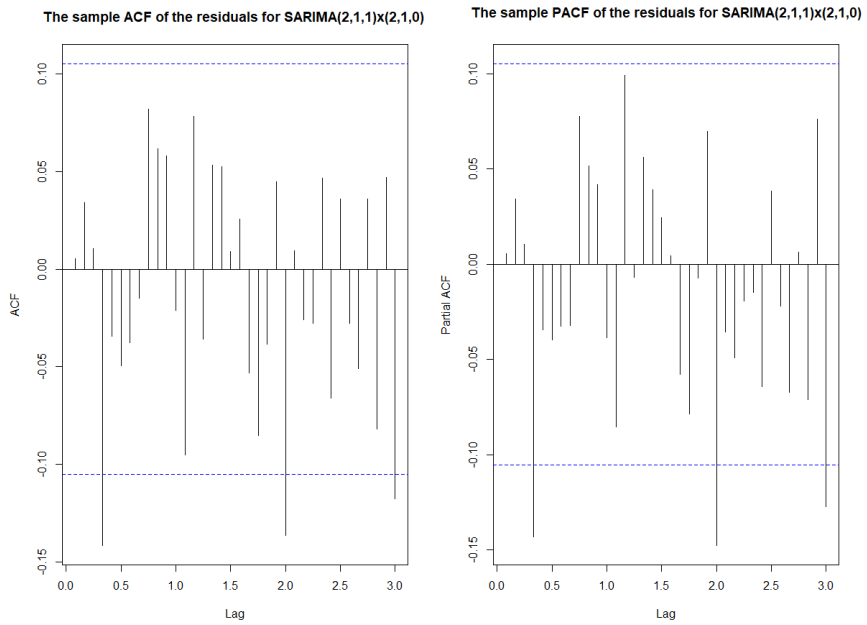

Figure 18: ACF-PACF Plots for SARIMA (2,1,1)x(2,1,0) - CSS

We observe significant correlations in both the ACF and PACF plots, thus, as the residuals are not white noise, we reject this model.

5.2.1.4 SARIMA (2,1,1) x (2,1,0) - ML

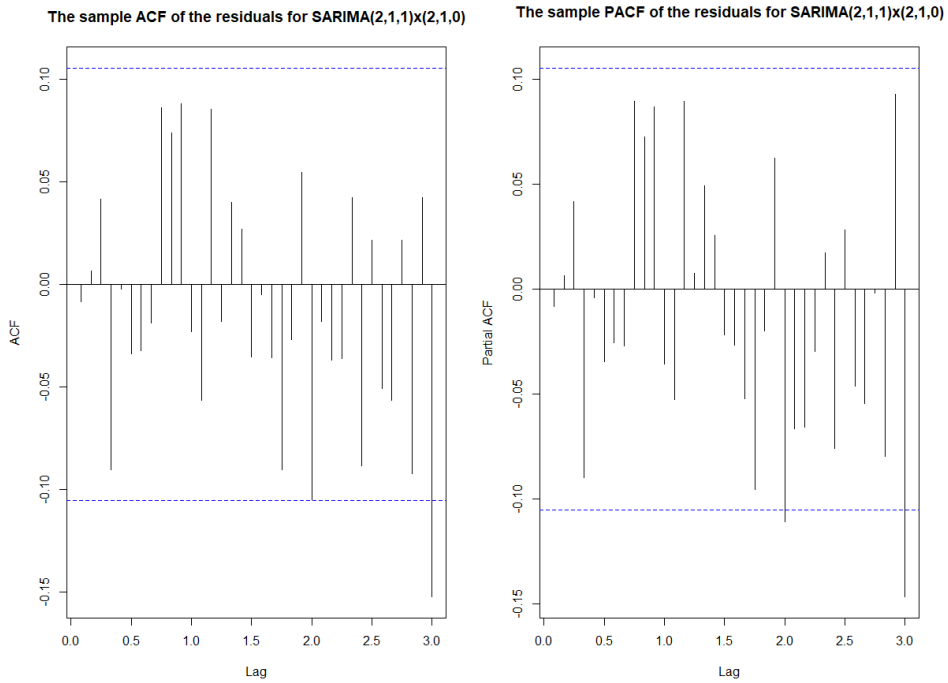

Figure 19: ACF-PACF Plots for SARIMA (2,1,1)x(2,1,0) - ML

From the ACF and PACF plots, it is evident that the residuals follow a white noise process. Thus, we consider this model in our short list it for our candidate set to perform next procedure. Furthermore, similar results (the residuals were white noise/uncorrelated) was observed for all the remaining models in both maximum likelihood estimation and Conditional sum of squares (CSS) methods. Thus, our shortlisted set of candidate models are as follows:

CSS Models	ML Models
SARIMA (3,1,1) x (2,1,0)	SARIMA (2,1,1) x (2,1,0)
SARIMA (0,1,4) x (2,1,0)	SARIMA (3,1,1) x (2,1,0)
SARIMA (0,1,5) x (2,1,0)	SARIMA (0,1,4) x (2,1,0)

SARIMA (1,1,4) x (2,1,0)	SARIMA (0,1,5) x (2,1,0)
SARIMA (2,1,1) x (2,1,0)	SARIMA (1,1,4) x (2,1,0)
SARIMA (4,1,1) x (2,1,0)	SARIMA (2,1,1) x (2,1,0)
	SARIMA (4,1,1) x (2,1,0)

Table 10: Final Set of Candidate Models from CSS and ML Method

5.2.2 Sort by AIC and BIC

In this section, we would separately sort the ML and CSS models (Table 11) based on AIC and BIC.

Model	df	BIC
m211_ml.tourism	6	-1075.648
m014_ml.tourism	7	-1072.701
m311_ml.tourism	7	-1070.056
m411_ml.tourism	8	-1068.178
m114_ml.tourism	8	-1067.801
m015_ml.tourism	8	-1067.612

- ML Models

Model	df	AIC
m014_ml.tourism	7	-1099.379
m411_ml.tourism	8	-1098.668
m211_ml.tourism	6	-1098.515
m114_ml.tourism	8	-1098.290
m015_ml.tourism	8	-1098.101
m311_ml.tourism	7	-1096.734

Table 11: AIC-BIC Score Comparison of ML Models

For ML models,from Table 11, we infer that SARIMA (0,1,4) x (2,1,0), SARIMA (4,1,1) x (2,1,0) and SARIMA (2,1,1) x (2,1,0) perform best in terms of AIC with respect to their counter parts. In terms of BIC, SARIMA (2,1,1) x (2,1,0) performs best followed by SARIMA (0,1,4) x (2,1,0), furthermore, SARIMA (4,1,1) x (2,1,0) with a close score was not too far behind in magnitude. Thus, our shortlisted set of ML models would be :

ML Models
SARIMA (0,1,4) x (2,1,0)
SARIMA (4,1,1) x (2,1,0)
SARIMA (2,1,1) x (2,1,0)

Table 12: Shortlisted Set of ML Models from AIC-BIC Scores

Model	df	BIC
m311_css.tourism	7	-1095.286
m411_css.tourism	8	-1093.582
m014_css.tourism	7	-1089.082
m114_css.tourism	8	-1087.898
m015_css.tourism	8	-1084.497

- CSS Models

Model	df	AIC
m411_css.tourism	8	-1120.528
m311_css.tourism	7	-1118.382
m114_css.tourism	8	-1114.843
m014_css.tourism	7	-1112.178
m015_css.tourism	8	-1111.442

Table 13: AIC-BIC Score Comparison of CSS Models

For CSS models, from Table 13, we infer that SARIMA (4,1,1) x (2,1,0) and SARIMA (3,1,1) x (2,1,0) perform best in terms of both AIC and BIC, with respect to their counter parts. However, adhering to the ***Principal of parsimony***, we eliminate model SARIMA (4,1,1) x (2,1,0), and choose the model SARIMA (3,1,1) x (2,1,0) as it requires smaller number of parameters to adequately represent the time series, also, it must be noted that, SARIMA (4,1,1) x (2,1,0) is in fact an over-fitting model of SARIMA (3,1,1) x (2,1,0).Thus, the final candidate yielded from CSS method would be:

CSS Model
SARIMA (3,1,1) x (2,1,0)

Table 14: Shortlisted CSS Model from AIC-BIC Scores

We would analyse the coefficients of all these shortlisted models in the next step to determine if they are significant.

5.2.3 Overfitting

We have shortlisted models according to the residuals and their AIC, BIC score. In this section, we will add ordinary AR and MA coefficients w.r.t the specified models to check their validity. We consider the models specified in Table 15 enlist the overfit models. The table below shows the specified model obtained from previous steps with their overfitting counterparts models and we further decide which overfitting models are suitable for consideration.

Specified Model	Over fitted Models	Comments
SARIMA (3,1,1) x (2,1,0) – CSS		
one extra ordinary MA coefficient	SARIMA (3,1,2) x (2,1,0) – CSS	Considered.
one extra ordinary AR coefficient	SARIMA (4,1,1) x (2,1,0) – CSS	Dropped. Already fitted and from parsimony
SARIMA (0,1,4) x (2,1,0) - ML		
one extra ordinary MA coefficient	SARIMA (0,1,5) x (2,1,0) - ML	Dropped. Already fitted and AIC, BIC lower than specified model
one extra ordinary AR coefficient	SARIMA (1,1,4) x (2,1,0) - ML	Dropped. Already fitted and AIC, BIC lower than specified model
SARIMA (2,1,1) x (2,1,0) - ML		
one extra ordinary MA coefficient	SARIMA (2,1,2) x (2,1,0) - ML	Considered.
one extra ordinary AR coefficient	SARIMA (3,1,1) x (2,1,0) - ML	Dropped. Already fitted and AIC, BIC lower than specified model
SARIMA (4,1,1) x (2,1,0) - ML		
one extra ordinary MA coefficient	SARIMA (4,1,2) x (2,1,0) - ML	Considered.
one extra ordinary AR coefficient	SARIMA (5,1,1) x (2,1,0) - ML	Considered.

Table 15: Analyzing overfitted counterparts of candidate models

We compare the adequacy of the models using overfitting approach. Taking the specified and overfitted models we test the significance of parameters.

Candidate Model	Coefficients		
	Significant	**Total**	
SARIMA (3,1,1) x (2,1,0) – CSS	3	4	AR(1), AR(2), MA(1)
SARIMA (3,1,2) x (2,1,0) – CSS (OF)	1	5	MA(1)
SARIMA (0,1,4) x (2,1,0) - ML	1	4	MA(1)
SARIMA (2,1,1) x (2,1,0) - ML	3	3	All, AR(1),AR(2), MA(1)
SARIMA (2,1,2) x (2,1,0) – ML (OF)	1	4	MA(1)
SARIMA (4,1,1) x (2,1,0) - ML	2	5	AR(4), MA(1)
SARIMA (4,1,2) x (2,1,0) – ML (OF)	4	6	AR(1),AR(3), MA(1),MA(2)
SARIMA (5,1,1) x (2,1,0) – ML (OF)	2	6	AR(4), MA(1)

Table 16: Comparing adequacy of models using overfitting

It is observed that overfitting models do not fit well and have insignificant coefficients. From, the table 16, we find that model with ARIMA(2,1,1) have the best parameter estimation with all significant coefficients. Thus, we select SARIMA (2,1,1) x (2,1,0) model using ML method as our best model.

5.3 Model Diagnostics

In this section, we perform an analysis of the residuals of our final model. Determining if the residuals of the selected model closely follows the behaviour of white noise, we can further support the statement that the selected model is the best candidate model.

```
        Shapiro-Wilk normality test

data:  res.model
W = 0.99442, p-value = 0.2373
```

Figure 20: Normality test for residuals of chosen model

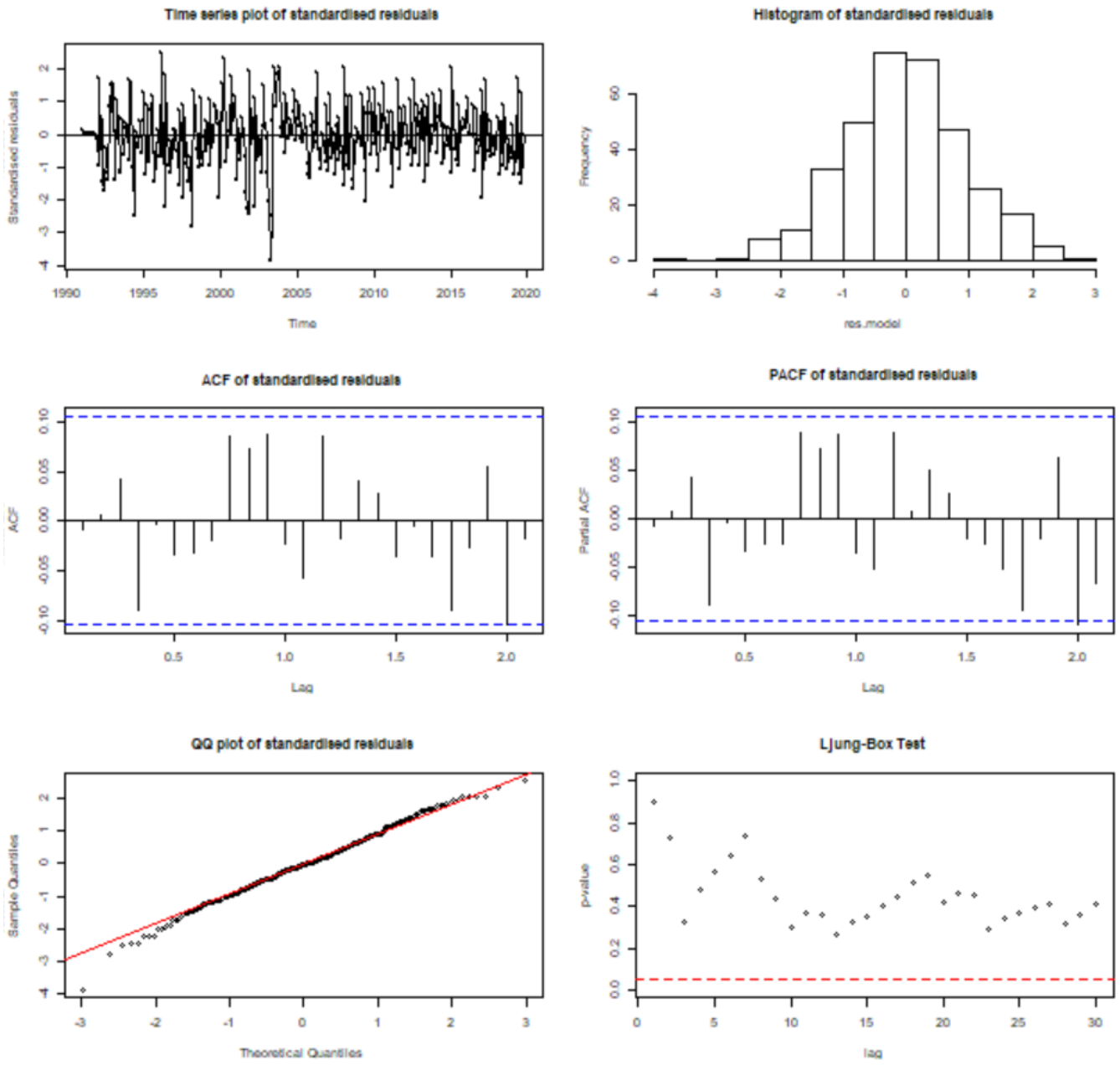

Figure 21: Residual analysis of chosen model

- **Time series plot of the residuals**: (Top left, fig.21) The plot shows no trend nor seasonality, close to the white noise plot.
- **Normality of the residuals**: From the histogram (Top right, fig.21) we can say residuals are normally distributed. The QQ-plot (Bottom-left, fig.21) shows minor deviations at tails but mostly follows the normal behaviour. The results of the Shapiro-Wilks test (fig.22) which give p-value >0.05 and fail to reject the normality hypothesis support that the residual follow a normal distribution.
- **Autocorrelation of the residuals**: With no significant lags in the ACF plot (Middle-Right, fig.21) and only one slightly significant lag at the near end in PACF plot(Middle-Right, fig.21), we say that there are no significant correlations As there are no significant lags in the ACF Plot (Middle-Right, fig.21)
- **The L-Jung Box Test**: With all points above the red-dashed line in the plot (Bottom -Right, fig.21), it depicts that there are no residual correlations

Thus, we can support the suitability of our selected model as its residual, closely follow the characteristics of white noise.
The best model is SARIMA (2,1,1) x (2,1,0)

6. Forecasting

Based on the analysis undertaken, we have selected SARIMA (2,1,1) x (2,1,0) as the best model and this is used to predict the future values of the number of short-term visitor to Australia for the next 10 months

	Point Forecast <dbl>	Lo 80 <dbl>	Hi 80 <dbl>	Lo 95 <dbl>	Hi 95 <dbl>
Dec 2019	1090425.6	1028519.7	1156057.6	997184.3	1192385.4
Jan 2020	780587.8	732166.1	832211.9	707761.2	860908.0
Feb 2020	931586.8	868720.7	999002.4	837177.4	1036643.0
Mar 2020	889338.5	826209.1	957291.6	794625.1	995341.1
Apr 2020	755089.5	699151.3	815503.2	671237.0	849417.1
May 2020	661695.0	610874.4	716743.5	585571.2	747714.8
Jun 2020	686743.1	632257.2	745924.5	605186.4	779290.7
Jul 2020	833020.5	764932.9	907168.6	731172.1	949055.8
Aug 2020	799089.7	731941.1	872398.6	698709.9	913890.5
Sep 2020	739404.2	675634.2	809193.1	644133.9	848765.4

Figure 22: 10-month forecast values & their CIs for number of visitor arrivals

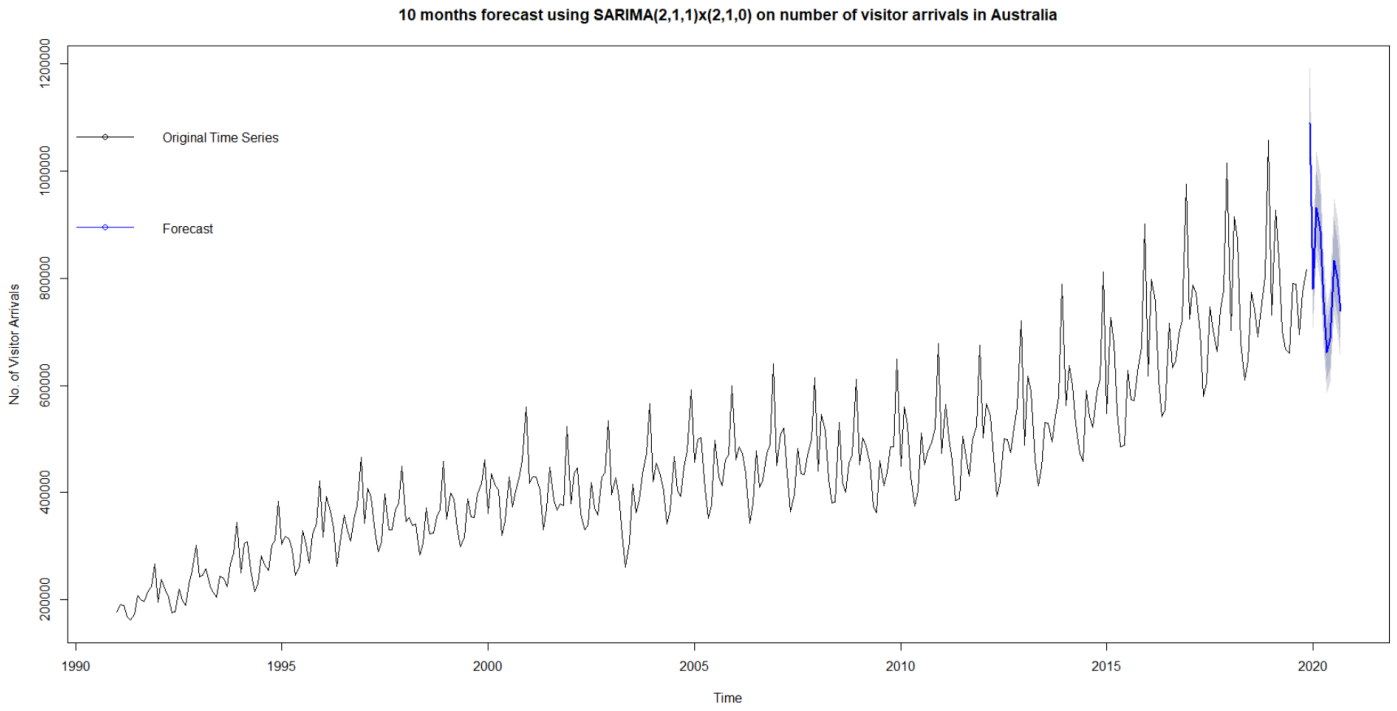

Figure 23: Plotting forecasts with original time series

The forecast shows that trend and seasonality will continue for the coming months. The number of visitors will increase during the peak travel times to Australia. This scenario is without taking the impact of COVID-19 global pandemic in consideration. The forecast shows that tourism in Australia would keep on flourishing with highest number of visitors predicted to be received in Dec 2019 within a range of (997,184 - 1,192,385).

7. Conclusion

The best model for predicting the monthly number of visitor arrivals in Australia was **SARIMA (2,1,1)x(2,1,0)**. This was confirmed by assessing:

- Lowest AIC/BIC score, hence the best amongst all candidates
- Coefficient was found to be significant
- Residuals were normally distributed
- No correlation values were found to be greater than the confidence bounds

References

Statistics, A. B. o., 2019. *3401.0 - Overseas Arrivals and Departures, Australia, Nov 2019.* [Online] Available at: https://www.abs.gov.au/AUSSTATS/abs@.nsf/DetailsPage/3401.0Nov%202019?OpenDocument [Accessed 23 May 2020].

Appendix

```
---
title: "filefor report"
output: word_document
---



```{r setup, include=FALSE}
knitr::opts_chunk$set(echo = TRUE)
# Importing all the necessary packages
library(knitr)
library(TSA)
library(tseries)
library(lmtest)
library(dLagM)
library(FitAR)
library(forecast)
library(readxl)
library(fUnitRoots)
```


```{r echo=FALSE }
# Function for AIC BIC Sort - ML Models
sort.score <- function(x, score = c("bic", "aic")){
  if (score == "aic"){
    x[with(x, order(AIC)),]
  } else if (score == "bic") {
    x[with(x, order(BIC)),]
  } else {
    warning('score = "x" only accepts valid arguments ("aic","bic")')
  }
}
```
```

```
# Function for AIC BIC Sort - CSS Models
calc_aic_bic <- function(model,data)
{
 len <- length(data)
 aic=list()
 bic=list()
 k=list()
 df=list()
 for(i in 1:length(model)){
  k[[i]] <- length(model[[i]]$coef)
  aic[[i]] <- -2*model[[i]]$loglik +2*k[[i]]
  bic[[i]] <- -2*model[[i]]$loglik +k[[i]]*log(len)
  df[[i]] <- k[[i]] +1+1-1
 }
 return(list(aic,bic,df))
}

# Function for Residual Analysis on shortlisted ML and CSS Models
residual.analysis <- function(model, std = TRUE,start = 2, class = c("ARIMA","GARCH","ARMA-GARCH")[1]){
 # If you have an output from arima() function use class = "ARIMA"
 # If you have an output from garch() function use class = "GARCH"
 # If you have an output from ugarchfit() function use class = "ARMA-GARCH"

 if (class == "ARIMA"){
  if (std == TRUE){
   res.model = rstandard(model)
  }else{
   res.model = residuals(model)
  }
 }else if (class == "GARCH"){
  res.model = model$residuals[start:model$n.used]
 }else if (class == "ARMA-GARCH"){
  res.model = model@fit$residuals
 }else {
```

```
  stop("The argument 'class' must be either 'ARIMA' or 'GARCH' ")
 }
 par(mfrow=c(3,2))
 plot(res.model,type='o',ylab='Standardised residuals', main="Time series plot of standardised residuals")
 abline(h=0)
 hist(res.model,main="Histogram of standardised residuals")
 acf(res.model,main="ACF of standardised residuals")
 pacf(res.model,main="PACF of standardised residuals")
 qqnorm(res.model,main="QQ plot of standardised residuals")
 qqline(res.model, col = 2)
 print(shapiro.test(res.model))
 k=0
 #Ljung Box independence test for every lag H0:independent, H1: series is correlated at lags
 LBQPlot(res.model, lag.max = 30, StartLag = k + 1, k = 0, SquaredQ = FALSE)
}
```

```{r }
# Setting the path for file access
setwd("C:\\Users\\DELL\\OneDrive\\Studies\\Semester 3\\Time Series\\Assignment 3")
# Reading the data into R
tourism <- read_excel("A3.xls", sheet = "Data1", range = "BR11:BR357", col_names = FALSE)
# Displaying the first 6 rows of the series
head(tourism)
```

```{r }
# Converting the dataframe into a time series object
tourism.ts = ts(tourism,start=c(1991,1),end=c(2019,11),frequency = 12)
```

```
class(tourism.ts)

head(tourism.ts)
```

```{r }
# Plotting the Time Series Plot

par(mfrow=c(1,1))

# Plotting the time series

plot(tourism.ts,type='l',ylab='No. of Arrivals', main="Time series plot of the number of visitor arrivals")

points(y=tourism.ts,x=time(tourism.ts), pch=as.vector(season(tourism.ts)))
```

```{r }
# Scatter plot of observations for checking correlation in consecutive values of Neighboring number of visitor arrivals

plot(y=tourism.ts,x=zlag(tourism.ts),ylab='Number of Visitor Arrivals', xlab='Previous Year Visitor Arrivals',main = "Scatter plot of the number of visitor arrivals in successive years")
```

```{r }
# Determining the correlation between succeeding data points

y = tourism.ts

x = zlag(tourism.ts)

ind = 2:length(x)

cor(y[ind],x[ind])
```

```{r }
#ACF and PACF

par(mfrow=c(1,2))

acf(tourism.ts,  lag.max = 36,main="The sample ACF of visitor arrivals")

pacf(tourism.ts,  lag.max = 36,main="The sample PACF of visitor arrivals")
```

```{r }

# Implementing ADF Test on the series

order = ar(diff(tourism.ts))$order

adfTest(tourism.ts, lags = order,  title = NULL,description = NULL)


```


```{r }

########### Fitting Deterministic Trend Models ##########


# Defining the Linear regression model

model1 = lm(tourism.ts~time(tourism.ts))

plot(tourism.ts,type='o',ylab='No. of Visitor Arrivals',main ="Fitting Linear Trend Model to monthly number of visitor arrivals in Australia")

legend ("topleft", lty = 1,bty = "n",text.width = 8,text.size ,col = c("black","red"),c("No. of Arrivals","Fitted Linear Trend Line"),cex = 0.75)

abline(model1, col="red") # add the fitted least squares line to the model

summary(model1)
```


```{r }

# Defining the Quadratic regression model

t = time(tourism.ts)

t2 = t^2

quadmodel2 = lm(tourism.ts~t+t2) # label the quadratic trend model as quadmodel2

summary(quadmodel2) # determining the summary

plot(ts(fitted(quadmodel2)),

    ylim=c(min(c(fitted(quadmodel2),as.vector(tourism.ts))),

        max(c(fitted(quadmodel2),as.vector(tourism.ts)))),

    col=c('red'), ylab='No. of Visitor Arrivals',main = "Fitting Quadratic Trend Model to monthly number of visitor arrivals in Australia")

lines(as.vector(tourism.ts),type="o",col="black")

legend ("topleft", lty = 1,bty = "n",text.width = 5, col = c("black","red"),c("No. of Arrivals","Fitted Quadratic Trend Line"),cex = 0.75)


```

```{r }

# Defining Cubic trend Model

t = time(tourism.ts)

t2 = t^2

t3 = t^3

model3 = lm(tourism.ts~t+t2+t3)

summary(model3) #Determining the summary

plot(ts(fitted(model3)),

    ylim=c(min(c(fitted(model3),as.vector(tourism.ts))),

        max(c(fitted(model3),as.vector(tourism.ts)))),

    col=c('red'), ylab='No. of Visitor Arrivals',main = "Fitting Cubic Trend Model to monthly number of visitor arrivals in Australia")

lines(as.vector(tourism.ts),type="o",col="black")

legend ("topleft", lty = 1,bty = "n",text.width = 10, col = c("black","red"),c("No. of arrivals","Fitted Cubic Trend Line"),cex = 0.75)

```

```{r }

# Defining Seasonal trend Model

month.=season(tourism.ts) # period added to improve table display and this line sets up indicators

model4=lm(tourism.ts~month.-1) # -1 removes the intercept term

summary(model4)

model4.1=lm(tourism.ts~month.) # remove -1 to include the intercept term in the model

summary(model4.1)

```

```{r }

# Defining Harmonic Trend Model

har.=harmonic(tourism.ts,1)

model.tourism.har=lm(tourism.ts~har.)

summary(model.tourism.har)

plot(ts(fitted(model.tourism.har)), ylim = c(min(c(fitted(model.tourism.har),

                                as.vector(tourism.ts))), max(c(fitted(model.tourism.har),as.vector(tourism.ts)))),
```

```
    ylab='y' , main = "Fitting Harminic Trend Model to monthly number of visitor arrivals in Australia",
type="l",lty=2,col="red")

lines(as.vector(tourism.ts),type="o")

legend ("topleft", lty = 1,bty = "n",text.width = 10, col = c("black","red"),c("No. of arrivals","Fitted Harmonic Trend
Line"),cex = 0.75)
```

```{r }
# Data has strong changing variance,implementing log transformation

log.tourism.ts = log(tourism.ts)

par(mfrow=c(1,1))

plot(log.tourism.ts,ylab='log of visitor ',xlab='Year',type='o', main = "Time Series plot of log of monthly arrivals.")

points(log.tourism.ts,cex = .6, col = "black")

plot(tourism.ts,type='o',main="Time series plot of the number of visitor arrivals", ylab = 'No. of Visitors')

points(tourism.ts,cex = .6, col = "black")
```

```{r }

# Seasonal differencing and fitting a plain seasonal model & residual analysis

m1.tourism = arima(log.tourism.ts,order=c(0,0,0),seasonal=list(order=c(0,1,0), period=12))

res.m1 = residuals(m1.tourism);

par(mfrow=c(1,1))

plot(res.m1,xlab='Time',ylab='Residuals',main="Time series plot of the residuals")

# Plotting ACF and PACF plots

par(mfrow=c(1,2))

acf(res.m1, lag.max = 36, main = "The sample ACF of the residuals")

pacf(res.m1, lag.max = 36, main = "The sample PACF of the residuals")
```

```{r }
```

```
# Implement SARMA(1,0) component
m2.tourism = arima(log.tourism.ts,order=c(0,0,0),seasonal=list(order=c(1,1,0), period=12))
res.m2 = residuals(m2.tourism);
par(mfrow=c(1,1))
plot(res.m2,xlab='Time',ylab='Residuals',main="Time series plot of the residuals")

# Plotting ACF and PACF plots
par(mfrow=c(1,2))
acf(res.m2, lag.max = 36, main = "The sample ACF of the residuals")
pacf(res.m2, lag.max = 36, main = "The sample PACF of the residuals")
```

```{r }
# Implement SARMA(2,0) component
m3.tourism = arima(log.tourism.ts,order=c(0,0,0),seasonal=list(order=c(2,1,0), period=12))
res.m3 = residuals(m3.tourism);
par(mfrow=c(1,1))
plot(res.m3,xlab='Time',ylab='Residuals',main="Time series plot of the residuals")

# Plotting ACF and PACF plots
par(mfrow=c(1,2))
acf(res.m3, lag.max = 36, main = "The sample ACF of the residuals")
pacf(res.m3, lag.max = 36, main = "The sample PACF of the residuals")
```

```{r }
# Performing first ordinary difference with SARMA(1,0) component
m4.tourism = arima(log.tourism.ts,order=c(0,1,0),seasonal=list(order=c(1,1,0), period=12))
res.m4 = residuals(m4.tourism);
par(mfrow=c(1,1))
```

```
plot(res.m2,xlab='Time',ylab='Residuals',main="Time series plot of the residuals")

# Plotting ACF and PACF plots
par(mfrow=c(1,2))
acf(res.m4, lag.max = 36, main = "The sample ACF of the residuals")
pacf(res.m4, lag.max = 36, main = "The sample PACF of the residuals")
```

```{r }
# Performing first ordinary difference with SARMA(2,0) component
m5.tourism = arima(log.tourism.ts,order=c(0,1,0),seasonal=list(order=c(2,1,0), period=12))
res.m5 = residuals(m5.tourism);
par(mfrow=c(1,1))
plot(res.m5,xlab='Time',ylab='Residuals',main="Time series plot of the residuals")

# Plotting ACF and PACF plots
par(mfrow=c(1,2))
acf(res.m5, lag.max = 36, main = "The sample ACF of the residuals")
pacf(res.m5, lag.max = 36, main = "The sample PACF of the residuals")
```

```{r }
#ADF Test to check if the ordinary differenced series is stationary
order = ar(diff(res.m5))$order
adfTest(res.m5, lags = order,  title = NULL,description = NULL)
```

```{r }
################ Model Building Strategy #################

# Plotting ACF and PACF Plots for the first differenced Monthly Visitor Arrival series
```

```
par(mfrow=c(1,2))
acf(res.m5, lag.max = 36, main = "The sample ACF of the residuals")
pacf(res.m5, lag.max = 36, main = "The sample PACF of the residuals")
```

```{r }
# Plotting EACF Matrix
eacf(res.m5)
```

```{r }
# Plotting the BIC Table
par(mfrow=c(1,1))
res = armasubsets(y=res.m5,nar=7,nma=7,y.name='test',ar.method='ols')
plot(res)
title('BIC Table for monthly total number of visitor arrivals', line = 6)
```

```{r }
# Checking the candidate models: Residual analysis using ACF and PACF plots

#SARIMA(1,1,1)x(2,1,0) with Method ML
m111_ml.tourism = arima(log.tourism.ts,order=c(1,1,1),seasonal=list(order=c(2,1,0), period=12),  method="ML")
res.m111_ml = residuals(m111_ml.tourism)
par(mfrow=c(1,1))
plot(res.m111_ml,xlab='Time',ylab='Residuals',main="Time series plot of the residuals")
par(mfrow=c(1,2))
acf(res.m111_ml, lag.max = 36, main = "The sample ACF of the residuals for SARIMA(1,1,1)x(2,1,0) ")
pacf(res.m111_ml, lag.max = 36, main = "The sample PACF of the residuals for SARIMA(1,1,1)x(2,1,0) ")
# Model Rejected - has significant Correlations
```

```
#SARIMA(1,1,1)x(2,1,0) with Method CSS
m111_css.tourism = arima(log.tourism.ts,order=c(1,1,1),seasonal=list(order=c(2,1,0), period=12),  method="CSS")
res.m111_css = residuals(m111_css.tourism)
par(mfrow=c(1,1))
plot(res.m111_css,xlab='Time',ylab='Residuals',main="Time series plot of the residuals")
par(mfrow=c(1,2))
acf(res.m111_css, lag.max = 36, main = "The sample ACF of the residuals for SARIMA(1,1,1)x(2,1,0)")
pacf(res.m111_css, lag.max = 36, main = "The sample PACF of the residuals for SARIMA(1,1,1)x(2,1,0)")
# Model Rejected - has significant Correlations

#SARIMA(2,1,1)x(2,1,0) with Method ML
m211_ml.tourism = arima(log.tourism.ts,order=c(2,1,1),seasonal=list(order=c(2,1,0), period=12),method="ML")
res.m211_ml = residuals(m211_ml.tourism);
par(mfrow=c(1,1))
plot(res.m211_ml,xlab='Time',ylab='Residuals',main="Time series plot of the residuals")
par(mfrow=c(1,2))
acf(res.m211_ml, lag.max = 36, main = "The sample ACF of the residuals for SARIMA(2,1,1)x(2,1,0)")
pacf(res.m211_ml, lag.max = 36, main = "The sample PACF of the residuals for SARIMA(2,1,1)x(2,1,0)")
# Model Considered- White Noise

#SARIMA(2,1,1)x(2,1,0) with Method CSS
m211_css.tourism = arima(log.tourism.ts,order=c(2,1,1),seasonal=list(order=c(2,1,0), period=12),method="CSS")
res.m211_css = residuals(m211_css.tourism);
par(mfrow=c(1,1))
plot(res.m211_css,xlab='Time',ylab='Residuals',main="Time series plot of the residuals")
par(mfrow=c(1,2))
acf(res.m211_css, lag.max = 36, main = "The sample ACF of the residuals for SARIMA(2,1,1)x(2,1,0) ")
pacf(res.m211_css, lag.max = 36, main = "The sample PACF of the residuals for SARIMA(2,1,1)x(2,1,0)")
# Model Rejected - has significant Correlations

#SARIMA(3,1,1)x(2,1,0) with method ML
m311_ml.tourism = arima(log.tourism.ts,order=c(3,1,1),seasonal=list(order=c(2,1,0), period=12),method="ML")
res.m311_ml = residuals(m311_ml.tourism);
```

```
par(mfrow=c(1,1))
plot(res.m311_ml,xlab='Time',ylab='Residuals',main="Time series plot of the residuals")
par(mfrow=c(1,2))
acf(res.m311_ml, lag.max = 36, main = "The sample ACF of the residuals for SARIMA(3,1,1)x(2,1,0)")
pacf(res.m311_ml, lag.max = 36, main = "The sample PACF of the residuals for SARIMA(3,1,1)x(2,1,0)")
# Model Considered- White Noise

#SARIMA(3,1,1)x(2,1,0) with method CSS
m311_css.tourism = arima(log.tourism.ts,order=c(3,1,1),seasonal=list(order=c(2,1,0), period=12),method="CSS")
res.m311_css = residuals(m311_css.tourism);
par(mfrow=c(1,1))
plot(res.m311_css,xlab='Time',ylab='Residuals',main="Time series plot of the residuals")
par(mfrow=c(1,2))
acf(res.m311_css, lag.max = 36, main = "The sample ACF of the residuals for SARIMA(3,1,1)x(2,1,0)")
pacf(res.m311_css, lag.max = 36, main = "The sample PACF of the residuals for SARIMA(3,1,1)x(2,1,0)")
# Model Considered- White Noise

#SARIMA(0,1,4)x(2,1,0) with method ML
m014_ml.tourism = arima(log.tourism.ts,order=c(0,1,4),seasonal=list(order=c(2,1,0), period=12),method="ML")
res.m014_ml = residuals(m014_ml.tourism);
par(mfrow=c(1,1))
plot(res.m014_ml,xlab='Time',ylab='Residuals',main="Time series plot of the residuals")
par(mfrow=c(1,2))
acf(res.m014_ml, lag.max = 36, main = "The sample ACF of the residuals for SARIMA(0,1,4)x(2,1,0)")
pacf(res.m014_ml, lag.max = 36, main = "The sample PACF of the residuals for SARIMA(0,1,4)x(2,1,0)")
# Model Considered- White Noise

#SARIMA(0,1,4)x(2,1,0) with method CSS
m014_css.tourism = arima(log.tourism.ts,order=c(0,1,4),seasonal=list(order=c(2,1,0), period=12),method="CSS")
res.m014_css = residuals(m014_css.tourism);
par(mfrow=c(1,1))
plot(res.m014_css,xlab='Time',ylab='Residuals',main="Time series plot of the residuals")
par(mfrow=c(1,2))
acf(res.m014_css, lag.max = 36, main = "The sample ACF of the residuals for SARIMA(0,1,4)x(2,1,0)")
```

```
pacf(res.m014_css, lag.max = 36, main = "The sample PACF of the residuals for SARIMA(0,1,4)x(2,1,0)")
# Model Considered- White Noise

#SARIMA(1,1,4)x(2,1,0) with method ML
m114_ml.tourism = arima(log.tourism.ts,order=c(1,1,4),seasonal=list(order=c(2,1,0), period=12),method = "ML")
res.m114_ml = residuals(m114_ml.tourism);
par(mfrow=c(1,1))
plot(res.m114_ml,xlab='Time',ylab='Residuals',main="Time series plot of the residuals")
par(mfrow=c(1,2))
acf(res.m114_ml, lag.max = 36, main = "The sample ACF of the residuals for SARIMA(1,1,4)x(2,1,0)")
pacf(res.m114_ml, lag.max = 36, main = "The sample PACF of the residuals for SARIMA(1,1,4)x(2,1,0)")
# Model Considered- White Noise

#SARIMA(1,1,4)x(2,1,0) with method CSS
m114_css.tourism = arima(log.tourism.ts,order=c(1,1,4),seasonal=list(order=c(2,1,0), period=12),method = "CSS")
res.m114_css = residuals(m114_css.tourism);
par(mfrow=c(1,1))
plot(res.m114_css,xlab='Time',ylab='Residuals',main="Time series plot of the residuals")
par(mfrow=c(1,2))
acf(res.m114_css, lag.max = 36, main = "The sample ACF of the residuals for SARIMA(1,1,4)x(2,1,0)")
pacf(res.m114_css, lag.max = 36, main = "The sample PACF of the residuals for SARIMA(1,1,4)x(2,1,0)")
# Model Considered- White Noise

#SARIMA(0,1,5)x(2,1,0) with method ML
m015_ml.tourism = arima(log.tourism.ts,order=c(0,1,5),seasonal=list(order=c(2,1,0), period=12),method = "ML")
res.m015_ml = residuals(m015_ml.tourism);
par(mfrow=c(1,1))
plot(res.m015_ml,xlab='Time',ylab='Residuals',main="Time series plot of the residuals")
par(mfrow=c(1,2))
acf(res.m015_ml, lag.max = 36, main = "The sample ACF of the residuals for SARIMA(0,1,5)x(2,1,0)")
pacf(res.m015_ml, lag.max = 36, main = "The sample PACF of the residuals for SARIMA(0,1,5)x(2,1,0)")
# Model Considered- White Noise

#SARIMA(0,1,5)x(2,1,0) with method CSS
```

```
m015_css.tourism = arima(log.tourism.ts,order=c(0,1,5),seasonal=list(order=c(2,1,0), period=12),method = "CSS")
res.m015_css = residuals(m015_css.tourism);
par(mfrow=c(1,1))
plot(res.m015_css,xlab='Time',ylab='Residuals',main="Time series plot of the residuals")
par(mfrow=c(1,2))
acf(res.m015_css, lag.max = 36, main = "The sample ACF of the residuals for SARIMA(0,1,5)x(2,1,0)")
pacf(res.m015_css, lag.max = 36, main = "The sample PACF of the residuals for SARIMA(0,1,5)x(2,1,0)")
# Model Considered- White Noise

#SARIMA(4,1,1)x(2,1,0) with method ML
m411_ml.tourism = arima(log.tourism.ts,order=c(4,1,1),seasonal=list(order=c(2,1,0), period=12),method="ML")
res.m411_ml = residuals(m411_ml.tourism);
par(mfrow=c(1,1))
plot(res.m411_ml,xlab='Time',ylab='Residuals',main="Time series plot of the residuals")
par(mfrow=c(1,2))
acf(res.m411_ml, lag.max = 36, main = "The sample ACF of the residuals for SARIMA(4,1,1)x(2,1,0)")
pacf(res.m411_ml, lag.max = 36, main = "The sample PACF of the residuals for SARIMA(4,1,1)x(2,1,0)")
# Model Considered- White Noise

#SARIMA(4,1,1)x(2,1,0) with method CSS
m411_css.tourism = arima(log.tourism.ts,order=c(4,1,1),seasonal=list(order=c(2,1,0), period=12),method="CSS")
res.m411_css = residuals(m411_css.tourism);
par(mfrow=c(1,1))
plot(res.m411_css,xlab='Time',ylab='Residuals',main="Time series plot of the residuals")
par(mfrow=c(1,2))
acf(res.m411_css, lag.max = 36, main = "The sample ACF of the residuals for SARIMA(4,1,1)x(2,1,0)")
pacf(res.m411_css, lag.max = 36, main = "The sample PACF of the residuals for SARIMA(4,1,1)x(2,1,0)")
# Model Considered- White Noise

```



```{r }
#AIC Sort for ML Models
```

```
sc.AIC =
AIC(m211_ml.tourism,m311_ml.tourism,m014_ml.tourism,m114_ml.tourism,m015_ml.tourism,m411_ml.tourism)

sort.score(sc.AIC, score = "aic")


#BIC Sort for ML Models

sc.BIC =
BIC(m211_ml.tourism,m311_ml.tourism,m014_ml.tourism,m114_ml.tourism,m015_ml.tourism,m411_ml.tourism)

sort.score(sc.BIC, score = "bic")


```



```{r }

# Performing AIC BIC Sort for CSS Models

Model<-list("m311_css.tourism","m014_css.tourism","m114_css.tourism","m015_css.tourism","m411_css.tourism")

aic_bic
<-calc_aic_bic(list(m311_css.tourism,m014_css.tourism,m114_css.tourism,m015_css.tourism,m411_css.tourism),log.
tourism.ts)

AIC <-aic_bic[[1]]

BIC <- aic_bic[[2]]

df <-  aic_bic[[3]]

aic_table <- as.data.frame(cbind(Model,AIC,df))

bic_table <- as.data.frame(cbind(Model,BIC,df))


BICtable <- as.data.frame(lapply(bic_table, unlist))

BICtable <- BICtable[order(BICtable$BIC),]

BICtable


AICtable <- as.data.frame(lapply(aic_table, unlist))

AICtable <- AICtable[order(AICtable$AIC),]

AICtable


```


```{r }

########## Overfitting CSS Model ##############
```

```
# Performing overfitting for CSS Model
m312_css.tourism = arima(log.tourism.ts,order=c(3,1,2),seasonal=list(order=c(2,1,0), period=12),method="CSS")

# Performing coef test on CSS models
coeftest(m311_css.tourism)
coeftest(m312_css.tourism)
```

```{r }
########## Overfitting ML Models ##############

# Performing overfitting for shortlisted ML Models
m511_ml.tourism = arima(log.tourism.ts,order=c(5,1,1),seasonal=list(order=c(2,1,0), period=12),method="ML")
m412_ml.tourism = arima(log.tourism.ts,order=c(4,1,2),seasonal=list(order=c(2,1,0), period=12),method="ML")
m212_ml.tourism = arima(log.tourism.ts,order=c(2,1,2),seasonal=list(order=c(2,1,0), period=12),method="ML")

# Performing coef test on ML models
coeftest(m014_ml.tourism)
coeftest(m411_ml.tourism)
coeftest(m511_ml.tourism)
coeftest(m412_ml.tourism)
coeftest(m212_ml.tourism)
coeftest(m211_ml.tourism)
```

```{r }
# Residual Analysis of SARIMA(2,1,1)x(2,1,0) to determine Model Diagnostics
win.graph(width=10, height=10,pointsize=8)
residual.analysis(model = m211_ml.tourism)
```

```

```{r }

# Forecasting using SARIMA(2,1,1)x(2,1,0)

par(mfrow=c(1,1))

m211_ml.tourism = Arima(tourism.ts,order=c(2,1,1),seasonal=list(order=c(2,1,0), period=12),method = "ML",lambda = 0)

preds = forecast(m211_ml.tourism, h = 10)

plot(preds,main = "10 months forecast using SARIMA(2,1,1)x(2,1,0) on number of visitor arrivals in Australia",ylab = "No. of Visitor Arrivals",xlab="Time")

```

```{r }

# Determining CI Range

predsdf<-as.data.frame(preds)

predsdf

```
```