TUNKU ABDUL RAHMAN UNIVERSITY COLLEGE

FACULTY OF COMPUTING AND INFORMATION TECHNOLOGY

BMMS2074 Statistics for Data Science Assignment

Academic Year 2022/2023

Student Names	Student ID	Contribution (%)	Signature
Sit Yie Sian	21WMR03693	25	
Leong Sheng Mou	21WMR07568	25	
Tan Jacqueline	21WMR03259	25	
Loke Tze Min	21WMR07394	25	
	Total:	**100%**	

Programme : RDS2S1

Tutorial Group : G2

Date of Submission : 1st October 2022

Lecturer : Dr. Chin Wan Yoke

Tutor : Dr. Tan Pei Ling

Comments :

PART-II: Depth of Knowledge Assessment Rubrics

Program Learning Outcomes	Evaluation Criteria	Competency Levels				Score
		0-3 Unsatisfactory	4-7 Fair	8-11 Good	12-15 Outstanding	
Critical Thinking and Problem Solving (75%)	Written Communication **(15%)**	Attempts to use a consistent system for basic organization; minimal attempts to use sources to support ideas in the writing and these sources may not be correctly documented using an appropriate referencing style and/or may not be fully relevant to the task at hand.	Follows expectations appropriate to a specific discipline and/or writing task for basic organization, and content; use credible and/or relevant sources to support ideas and to document these sources properly using APA or Harvard referencing style.	Demonstrates consistent use of important conventions particular to a specific discipline and/or writing task; consistently use credible, relevant sources appropriate to the discipline and genre to support ideas and documents sources with few errors or exceptions using APA or Harvard referencing style.	Demonstrates detailed attention to and successful execution of a wide range of conventions particular to a specific discipline and/or writing task (including organization, content, formatting, and stylistic choices); synthesize a range of high-quality, credible, relevant sources that are appropriate for the discipline and genre to develop ideas and fully documents these sources using APA or Harvard referencing style.	
	Problem Solving Strategy and Approaches **(15%)**	Unable to identify an approach to possible solution.	Identifies a possible but very general approach to a solution without a clear sense of the steps to solve the problem.	Identifies a reasonable and problem specific possible approach to a solution with some sense of steps to be undertaken to reach a solution.	Identifies at least one reasonable and problem specific possible approach to a solution. Outlines several steps in detail and/or identifies another reasonable and problem specific possible approach.	
	Analysis **(15%)**	Demonstrates emerging understanding of the data analysis without showing evident e.g. diagrams, models, timelines, illustrations, explanations, or a series of steps in the report. The report fails to tie into basic concepts and build on prior knowledge.	Demonstrates moderate understanding of the data analysis that are somewhat evident e.g. diagrams, models, timelines, illustrations, explanations, or a series of steps within the report. The report may fail to tie into basic concepts and build on prior knowledge.	Demonstrates considerable understanding of the data analysis and are evident e.g. diagrams, models, timelines, illustrations, explanations, or a series of steps within the report. The report ties into basic concepts and builds on prior knowledge.	Demonstrates in-depth/ thorough understanding of the data analysis and are clearly evident e.g. diagrams, models, timelines, illustrations, explanations, or a series of steps throughout the report. The report ties into basic concepts and builds on prior knowledge.	
	Critical Thinking and Perspective-Taking **(15%)**	Specific position is stated but is simplistic and obvious.	Information is presented with some interpretation or evaluation, but not enough to develop a coherent analysis or synthesis.	Specific position takes into account the complexities of an issue and acknowledges other viewpoints.	Questions are examined from a range of viewpoints, taking into account the complexities of an issue.	
	Conclusions and Related Outcomes (Implications and Consequences) **(15%)**	Conclusion is inconsistently tied to some of the information discussed; related outcomes (consequences and implications) are oversimplified.	Conclusion is logically tied to information (because information is chosen to fit the desired conclusion); some related outcomes (consequences and implications) are identified clearly.	Conclusion is logically tied to a range of information, including opposing viewpoints; related outcomes (consequences and implications) are identified clearly.	Conclusions and related outcomes (consequences and implications) are logical and reflect student's informed evaluation and ability to place evidence and perspective discussed in priority order.	
					Total:	

Table Content

1.0 Introduction

Throughout this assignment, a time series dataset Yearly Drug Sales had been used for data analysis purposes.The dataset that we use is considered as a time series data which is also known as time- stamped data. Time series data consists of sequence and indexed according to time order such as daily, monthly and yearly. Time series analysis can be useful to see how a given asset, security, or economic variables change over time and also can be used to examine how the change associated with the chosen data point compares to shifts in other variables over the same time period(Adam Hayes, 2022). Time series can be used to measure the change in population over the time which means time series can be used in non-financial contexts.

Time series consist of stochastic processes , discrete time series and continuous time series. Stochastic process is different from a time series because it is a collection of random variables and a time series is a collection of numbers of a stochastic process. The time series is very useful because it can have a forecast feature which studies the information of the historical data and the related pattern to predict the future outcome.(Adam Hayes, 2022) Trend and seasonality analysis had played an important role during this process.

Also, there are a few types of time series models that will be used to forecast the event based on historical data which are verified. The common types of method used for forecasting which are Autoregressive Moving Average(ARMA), Autoregressive Integrated Moving Average(ARIMA), Seasonal Autoregressive Integrated Moving-Average(SARIMA), Autoregression(AR) and Moving Average(MA). The box-Jenkins Model is introduced which provide the technique to forecast data range based on inputs from a specified time series .In this model the forecast data will apply to the principles of autoregression , differencing and the moving average also known as the p,d,q.So the p will normally will associated with AR model which stand for auto-regression and is used to predict the future based on the historical data.While q

which associated with MA model which stand for moving average and using historical forecast error instead of historical data value to do the forecast. The combination of the principles that exist in the box-Jenkins Model will be known as ARIMA(p,d,q). Without the differencing , it will be ARMA (p,q) and with the seasonality will become SARIMA (p,d,q) (P,D,Q).

2.0 Objective

Throughout this assignment, we are using R studio to compute time series analysis. The time-series data has different uses in different multiple industries and scenarios. The main objective of this study is to predict and forecast drug sales for the next two years by finding the best model and using the selected model to do so. Moreover, the other objective is to determine the time series plot and also the trend of time series after decomposition. Also, another objective of this assignment is to determine the way to check the stationarity of the data. We are also interested in the trend and seasonality of the drug sales from 1991 until 2003 and to determine the optimal parameter for comparison by using Autocorrelation Function (ACF) and Partial Autocorrelation Function(PACF). This project's secondary goal is to identify the utilization of the ARIMA Family. Finally, by completing this project, we expect to increase our knowledge of the topic, our proficiency with R Studio, and our ability to forecast and make predictions now that we have a solid understanding of time series data.

3.0 Methodology

1.Time Plot

Time Plot, time series graph is a sequential series of numerical data points. The main purpose of this method is to visualize the observation or dataset over time. Most people will use a time plot as a starting point of the analysis to get the overview of the dataset. Besides, it is a suitable approach to display the changes of data over time. Unlike other commonly used types of x-y graphs, the x-y graphs can represent any type of x variable such as age, height, weight or other variables. For time plot,time it is only allows displaying specific time measure on the x-axis. Other than that, time plots also do not consist of categories like bar charts or pie charts. A line is drawn connecting all the data points in the chart. In this assignment project, we had to use a time plot to visualize and understand the dataset. The Year is plotted on the x-axis and the Drug Sales is plotted on the y-axis. The use of a time plot allows us to investigate if there are any trends in the dataset. Moreover, the time series graphs are considered as an effective way to predict models that consist of uncertainty factors for the future. Also, time series graphs are that trend can be identified easily and the data can be plotted.

2.Decomposition

Decomposition is a statistical task in which the time series data is decomposed into a few components or extracting seasonality, trend from a series data. Time series data are the combination of multiple components which include level, trend, seasonality and noise. The component of level is the average value in the series while component trend is the increasing or decreasing value which occurs in the series. The component seasonality is the repeating short-term cycle in the series and the component noise is the random variation in the series. Decomposition has provided a useful abstract model for thinking about the time series generally and for better understanding problems during time series analysis and forecasting. The stated components are either additively or multiplicatively combined in the time series data. When the variance of data does not change over different values of the time series is known as the additive model. The systematic component is the arithmetic sum of the individual effects of the predictors. It is linear and

the trend of the line is always a straight line, while the seasonality which is the height and width of the cycle will have the same frequency and amplitude. Multiplicative models are the one where as the data increases, so does the seasonal pattern or the increasing variance. Then, the trend and seasonal components are added to the error component. Multiplicative model is different from additive model because it is non-linear, such as quadratic or exponential and the trend is a curved line and seasonality has increasing or decreasing frequency and amplitude over time.The formula of the additive and multiplicative model is as below :

Additive Model	**Multiplicative Model**
$X_t = T_t + S_t + C_t + I_t$	$X_t = T_t S_t C_t I_t$
• **T = Trend Component** • **S = Seasonal Component** • **C = Cyclical Component** • **I = Irregular Component**	

3.Log transformation

Logarithmic transformation in R is one of the transformations that is typically used in time series forecasting. If your forecasting results have negative values, then log transformation of the target value will prevent from going below zero. In other words, logarithmic transformation stabilizes the variance of the time series and ensures that predictions stay positive.

If the data show variation that increases or decreases with the level of the series, then a transformation can be useful. For example, a logarithmic transformation is often useful. If we denote the original observations as y1,…,yT and the transformed observations as w1,…,wT, then wt=log(yt). Logarithms are useful because they are interpretable: changes in a log value are relative (or percentage) changes on the original scale. So if log base 10 is used, then an increase of 1 on the log scale corresponds to a multiplication of 10 on the

original scale. Another useful feature of log transformations is that they constrain the forecasts to stay positive on the original scale.

4.Seasonality

Seasonality, which is also known as seasonal variation, is a characteristic of a time series in which the data experience a same pattern yearly and the pattern is repeating yearly. Seasonality consists of various types of periods which are daily, weekly, quarterly or yearly depending on the data . Seasonal also refers to any predictable variation or pattern that recurs or repeats over the course of a year. It can be used to help analyze stock and economic trends which is not only important for investors but also act as a big role for the company to prepare solutions to prevent a big amount of loss. There are some differences between seasonal effects and cyclical effects. For seasonal effects, it will observe the data within one calendar year while the cyclical effects depend on the duration of certain events such as boosted sales due to clear stocks for supermarkets. The time span of the cyclical events can be long or short, however the seasonal effect normally is fixed within one calendar year. The type of seasonality has to be solved and identify whether the time series that we choose is either multiplicative or additive seasonality before it is solved. By observing the amplitude height, we are able to differentiate the model. The amplitude of the additive model will move constantly in every season while the amplitude of the multiplicative model will increase by the time. In this case, our seasonality of the time series is a multiplicative model.

5.Differencing (Ordinary and non-seasonal differencing)

Differencing is a widely used method for data transformation tools to make the time series become stationary. Differencing helps to eliminate changes in the level of time series to stabilize the mean of time series, thus reducing or eliminating trend and seasonality. In the section of differencing, there are two types of differencing which are seasonal differencing and ordinary differencing. Ordinary differencing is usually used on non-stationary time series data. It can be applied to first-order differencing, second-order differencing and more. The number of differencing can be decided by the previous result whether is stationary or not stationary. If the result is still not stationary after applying the

first differencing, the second or following differencing should be applied until the result becomes stationary. The difference between an observation and the previous observation from the previous data is also known as seasonal differencing. For seasonal differencing, it is applied when a time series looks like stationary noise. The following shows the formula for ordinary differencing and seasonal differencing:

For Ordinary Differencing:

First-order differencing:

$$Y_t' = Y_t - Y_{t-1} \quad \text{for } t = 2, 3, \ldots, n.$$

Second-order differencing:

$$\begin{aligned} Y_t'' &= Y_t' - Y_{t-1}' \\ &= (Y_t - Y_{t-1}) - (Y_{t-1} - Y_{t-2}) \\ &= Y_t - 2Y_{t-1} + Y_{t-2} \quad \text{for } t = 3,4, \ldots, n. \end{aligned}$$

For Seasonal Differencing:

$$y_t' = y_t - y_{t-m}$$

Remark: m = number of seasons

5. Auto-correlation function (ACF)

Auto-correlation function, ACF is recognized as the key statistic in the time series analysis. It is used to analyze the level of correlation between the variables. Besides, it also defines the correlation of time series with itself, lagged by 1, 2 or more periods. Moreover, ACF can be used to investigate the non-randomness in the data. Other than that, the correlation matrix which is also known as correlogram also applied in this section. By observing the correlogram, we can examine the relationship between each pair of variable and numeric values in the dataset. As the data is correlated with itself at lag 0, it shows a correlation of +1, indicating a strong positive correlation exists. The model types can be selected by identifying the pattern of correlation in the ACF. In a simple way to say, we can observe the number of spikes in the correlogram and define the

model types. A positively auto correlated pattern at lag 1 is used to indicate the AR model, while a negatively auto correlated pattern at lag 1 should be used to indicate the MA model. For referencing, the formula for auto-correlation function equation is shown as below:

$$r_k = \frac{\sum_{t=k+1}^{n} (y_t - \bar{y})(y_{t-k} - \bar{y})}{\sum_{t=1}^{n} (y_t - \bar{y})^2}$$

6.Partial Autocorrelation Function (PACF)

Partial autocorrelation function (PACF) is a measure of the correlation between time series and its own lagged values, returning the values of the time series at all shorter lags. The graph that is used in PACF also known as partial autocorrelation graphs. The PACF is used to conduct additional research and observation into the type of model that should be used. It is to determine the term used in the autoregressive model (AR model). Similar to the ways of observing the correlogram, the lags which are represented as a line in the partial autocorrelation graph that cross the confidence interval will be counted and used to determine the number of AR terms. The AR model will not be considered if none of the lags line is not exceeding the confidence interval. Moreover, if the number of AR and MA terms are equal to zero, it will be concluded that it is a pure random model which is also known as white noise model. The following is the equation for PACF:

The partial autocorrelation at lag k is

$$r_{kk} = \begin{cases} r_1 & \text{if } k = 1 \\ \dfrac{r_k - \sum_{j=1}^{k-1} r_{k-1,j} r_{k-j}}{1 - \sum_{j=1}^{k-1} r_{k-1,j} r_j} & \text{if } k = 2, 3, \ldots \end{cases}$$

where $r_{kj} = r_{k-1,j} - r_{kk} r_{k-1,k-j}$ for $j = 1, 2, 3, \ldots, k-1$

7.AR

AR, Auto-Regressive Model which is also known as conditional models, Markov models or transition model is the time period at t is impacted by the observation at various slots such as t-1, t-2, t-3,...,t-k and predicts future behavior based on past behavior. The impact of previous time spots was decided by the coefficient factor at that particular period of time. The AR model takes in one argument which is p, which helps to determine how many previous time steps will be inputted.The value for "p" is called the order.The AR process is an example of a stochastic process, which have degrees of uncertainty or randomness built in. For example, a milk distribution company that produces milk every month in the country. The amount of milk to be produced this month considering the milk generated in the last year needs to be calculated.First, the PACF values of all the 12 lags with respect to the current month. If the value of the PACF of any particular month is more than a significant value only those values will be considered for the analysis. In an AR model, the value of the outcome variable y, at the same point *t* in time is similar to regular linear regression. It is directly related to the predictor variable x. Where simple regression and AR models are different is that y is independent of x and previous values for y.

$$Yt = \beta_1 * y_{-1} + \beta_2 * y_{t-2} + \beta_3 * y_{t-3} + \ldots\ldots\ldots + \beta_k * y_{t-k}$$

8.MA

In time series analysis, the moving-average model (MA) is a common approach for modeling univariate time series. The moving-average model specifies that the model that the output variable is cross-correlated with a non-identical to itself random variable. Along with the AR model, the moving average model is a special case and key component of the more general ARMA and ARIMA models of time series, which have more complicated stochastic structure. The MA(q) refers to the moving average model of order q :

$$X_t = \mu + \varepsilon_t + \theta_1 \varepsilon_{t-1} + \cdots + \theta_q \varepsilon_{t-q} = \mu + \sum_{i=1}^{q} \theta_i \varepsilon_{t-i} + \varepsilon_t,$$

where μ is the mean of the series, the θ1,...,θq are the parameters of the ar$\varepsilon_t, \varepsilon_{t-1}, \ldots, \varepsilon_{t-q}$ are known as white noise error terms. The value of q is called the order of the MA model. This can be equivalently written in terms of the backshift operator B as:

$$X_t = \mu + (1 + \theta_1 B + \cdots + \theta_q B^q)\varepsilon_t.$$

The key difference between AR model and MA model is that the MA model relies on the autocorrelation function while AR model relies on the partial autocorrelation function because MA models are not based on the past period returns. Thus, determining which lagged values have a significant direct effect on the present day one is not relevant.

9.ARMA

ARMA, Autoregressive Moving Average is a model of forecasting in which the methods of autoregression(AR) analysis and moving average(MA) are both applied to time-series data that is well behaved. In the ARMA model, it is assumed that the time series is stationary and when it fluctuates, it does so uniformly around a particular type. Often this model is referred to as the ARMA(p,q) model; where:

- p is the order of the autoregressive polynomial,
- q is the order of the moving average polynomial.

The equation is given by:

$$X_t = c + \varepsilon_t + \sum_{i=1}^{p} \varphi_i X_{t-i} + \sum_{i=1}^{q} \theta_i \varepsilon_{t-i}.$$

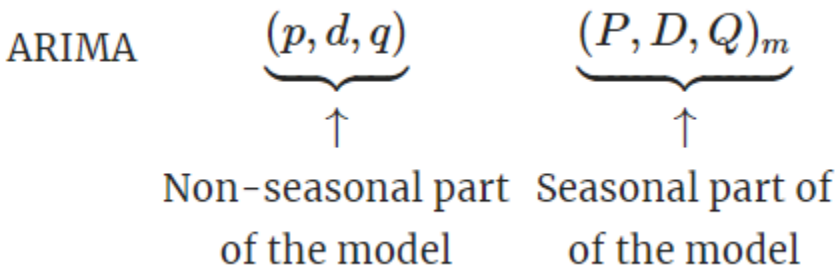
- φ = the autoregressive model's parameters,
- θ = the moving average model's parameters.

- c = a constant,
- Σ = summation notation,
- ε = error terms (white noise).

10.ARIMA

Auto-Regressive Integrated Moving Average is known as ARIMA. The forecasting equation's "autoregressive" elements are lags of the stationarized series. A time series that needs to be differentiated in order to become stationary is referred to as an "integrated" form of a stationary series. The lags of the forecast errors are referred to as "moving average" terms. Specialized ARIMA models include random-walk and random-trend models, autoregressive models, and exponential smoothing models.

So far, we have restricted our attention to non-seasonal data and non-seasonal ARIMA models. However, ARIMA models are also capable of modeling a wide range of seasonal data. A seasonal ARIMA model is formed by including additional seasonal terms in the ARIMA models we have seen so far. It is written as follows:



where m = number of observations per year. We use uppercase notation for the seasonal parts of the model, and lowercase notation for the non-seasonal parts of the model. The model's seasonal portion is made up of words that are comparable to its non-seasonal elements but involve backshifts of the seasonal period.

- p is the number of autoregressive terms,
- d is the number of nonseasonal differences needed for stationarity, and
- q is the number of lagged forecast errors in the prediction equation.

11.SARIMA

Seasonal ARIMA model (SARIMA) requires selecting hyperparameters for both the trend and seasonal elements of the series. There are three trend elements that require configuration.

They are the same as the ARIMA model; specifically:

p: Trend autoregression order.

d: Trend difference order.

q: Trend moving average order.

Seasonal Elements

There are four seasonal elements that are not part of ARIMA that must be configured; they are:

P: Seasonal autoregressive order.

D: Seasonal difference order.

Q: Seasonal moving average order.

m: The number of time steps for a single seasonal period.

Importantly, the m parameter influences the P, D, and Q parameters. For example, an m of 12 for monthly data suggests a yearly seasonal cycle. A P=1 would make use of the first seasonally offset observation in the model, e.g. t-(m * 1) or t-12. A P=2, would use the last two seasonally offset observations t-(m * 1), t-(m * 2). Similarly, a D of 1 would calculate a first order seasonal difference and a Q=1 would use a first order error in the model (e.g. moving average).

12.Box Pierce Test

The Ljung Box test which is also known as Box–Pierce test uses the test statistic, in the notation outlined above, given by

$$Q_{\mathrm{BP}} = n \sum_{k=1}^{h} \hat{\rho}_k^2,$$

and it uses the same critical region as defined above.Simulation studies have shown that the distribution for the Ljung–Box statistic is closer to a $x^2_{(h)}$ distribution then is the distribution for the Box–Pierce statistic for all sample sizes including small ones.

13.Augmented Dickey Fuller

Augmented Dickey Fuller test (ADF Test) is a common statistical test used to test whether a given Time series is stationary or not. It is one of the most commonly used statistical tests when it comes to analyzing the stationary of a series.

A Dickey-Fuller test is a unit root test that tests the null hypothesis that α=1 in the following model equation. Alpha is the coefficient of the first lag on Y.

Null Hypothesis (H0): alpha=1

$$y_t = c + \beta t + \alpha y_{t-1} + \phi \Delta Y_{t-1} + e_t$$

where,

- y(t-1) = lag 1 of time series
- delta Y(t-1) = first difference of the series at time (t-1)

It shares a fundamental null hypothesis with the unit root test. In other words, Y(t-1coefficient)'s is 1, indicating the existence of a unit root. The series is assumed to be non-stationary if it is not rejected.The Augmented Dickey-Fuller test, one of the most popular variations of the Unit Root test, developed from the equation above.

4.0 Data Sources

The dataset we obtained for this study is “Monthly Drug Sales” which we got from the “Kaggle” website. This dataset consists of 2 columns which are “Month” and “Monthly Drug Sales (per million)”. This dataset recorded the drug sales data from July 1991 to June 2008, totally 17 years. In “Monthly Drug Sales” column recorded the amount of drug sales per million during that month of year.

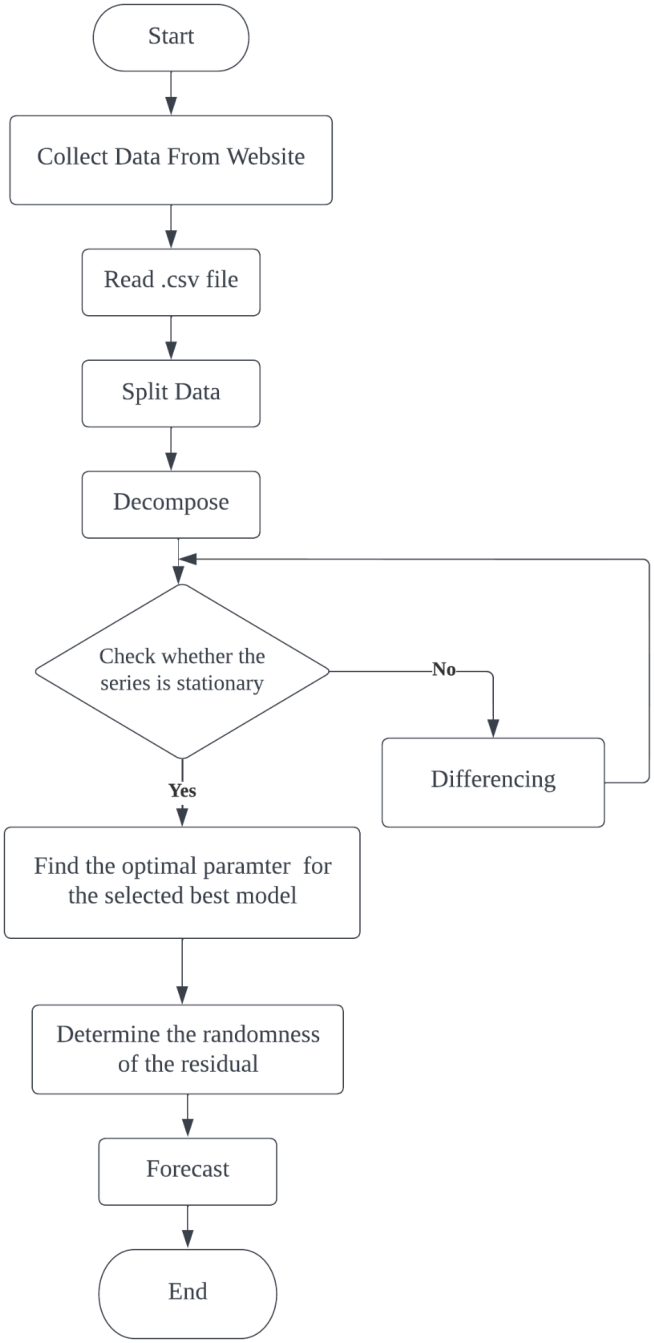
Data used: Drug Sales.csv (204 rows)

```
> data <- read.csv("Drug Sales.csv")
> data
        Month Monthly.drug.sales
1    1/7/1991           3.526591
2    1/8/1991           3.180891
3    1/9/1991           3.252221
4   1/10/1991           3.611003
5   1/11/1991           3.565869
6   1/12/1991           4.306371
7    1/1/1992           5.088335
8    1/2/1992           2.814520
9    1/3/1992           2.985811
10   1/4/1992           3.204780
11   1/5/1992           3.127578
12   1/6/1992           3.270523
13   1/7/1992           3.737851
14   1/8/1992           3.558776
15   1/9/1992           3.777202
```

First 15 rows of drug sales data

5.0 Data Analysis

Flowchart



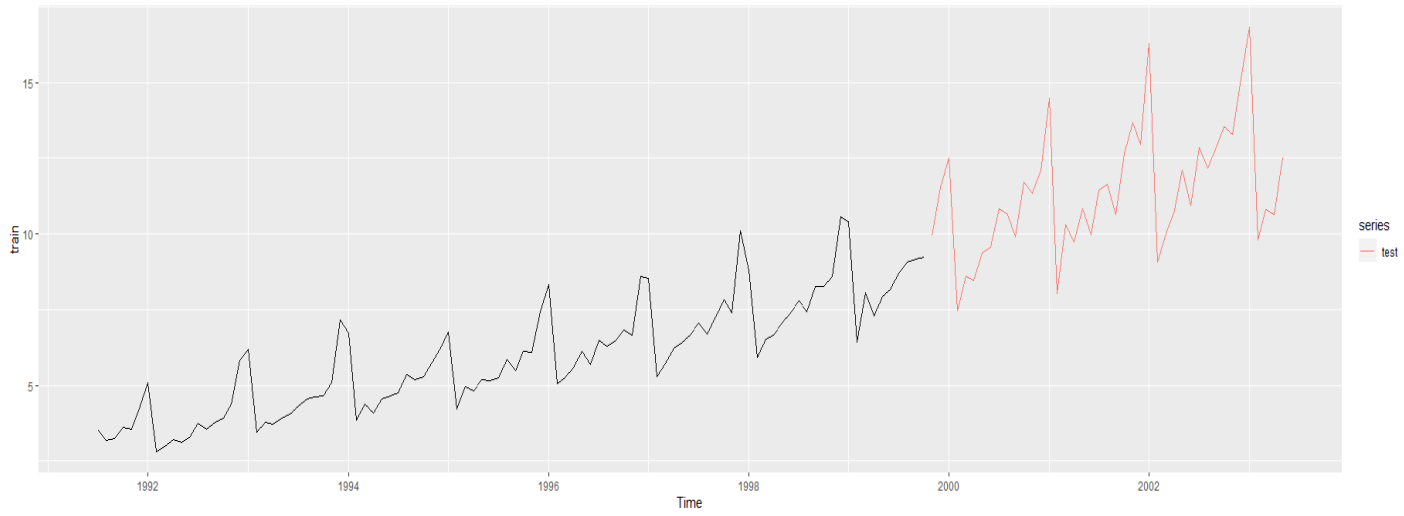
Step 0: Import and Split Data

```
#read file
print(getwd())
setwd("C:/Users/yiesi/Downloads")
data <- read.csv("Drug Sales.csv")
data
```

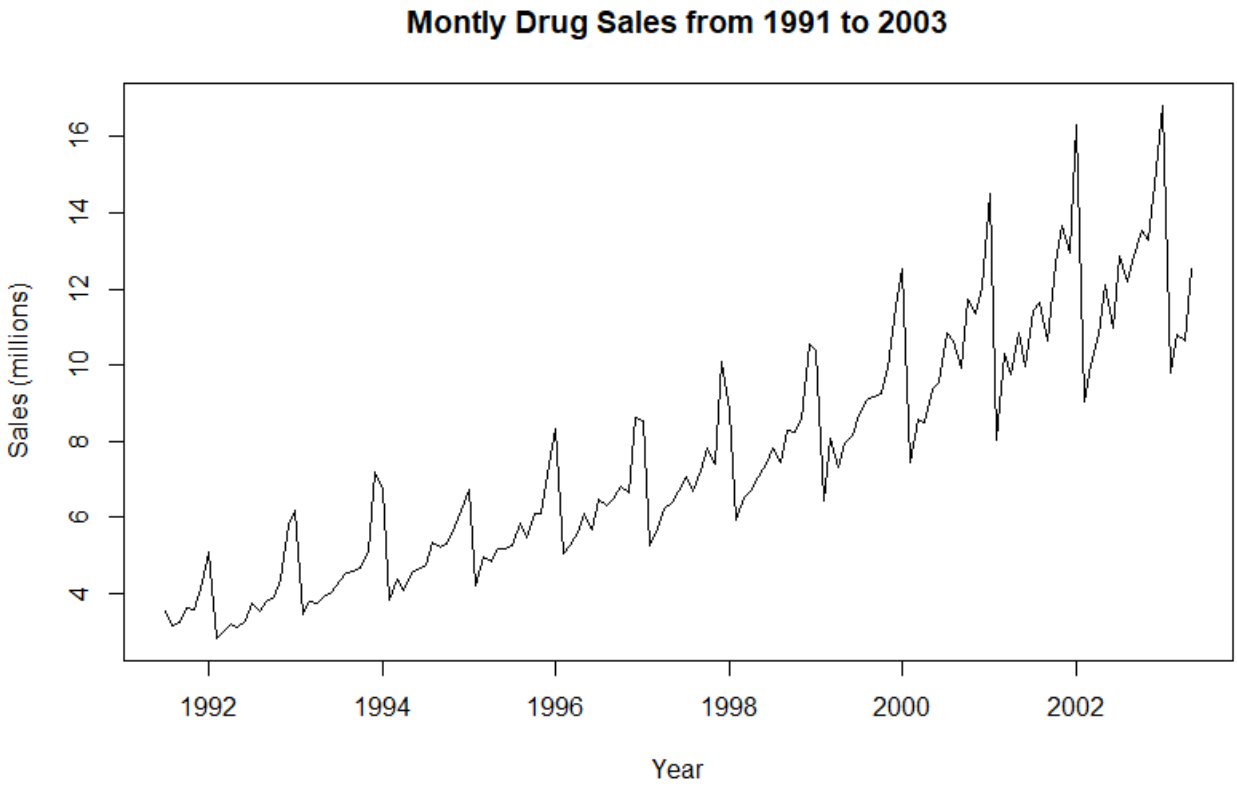
getwd() returns an absolute file path representing the current working directory of the R process while setwd(dir) is used to set the working directory to dir. Reads a file in table format and creates a data frame from it, with cases corresponding to lines and variables to fields in the file.

```
#Split data
train <- head(y, round(length(y) * 0.70))
h <- length(y) - length(train)
test <- tail(y, h)
train
test
autoplot(train) + autolayer(test)
```

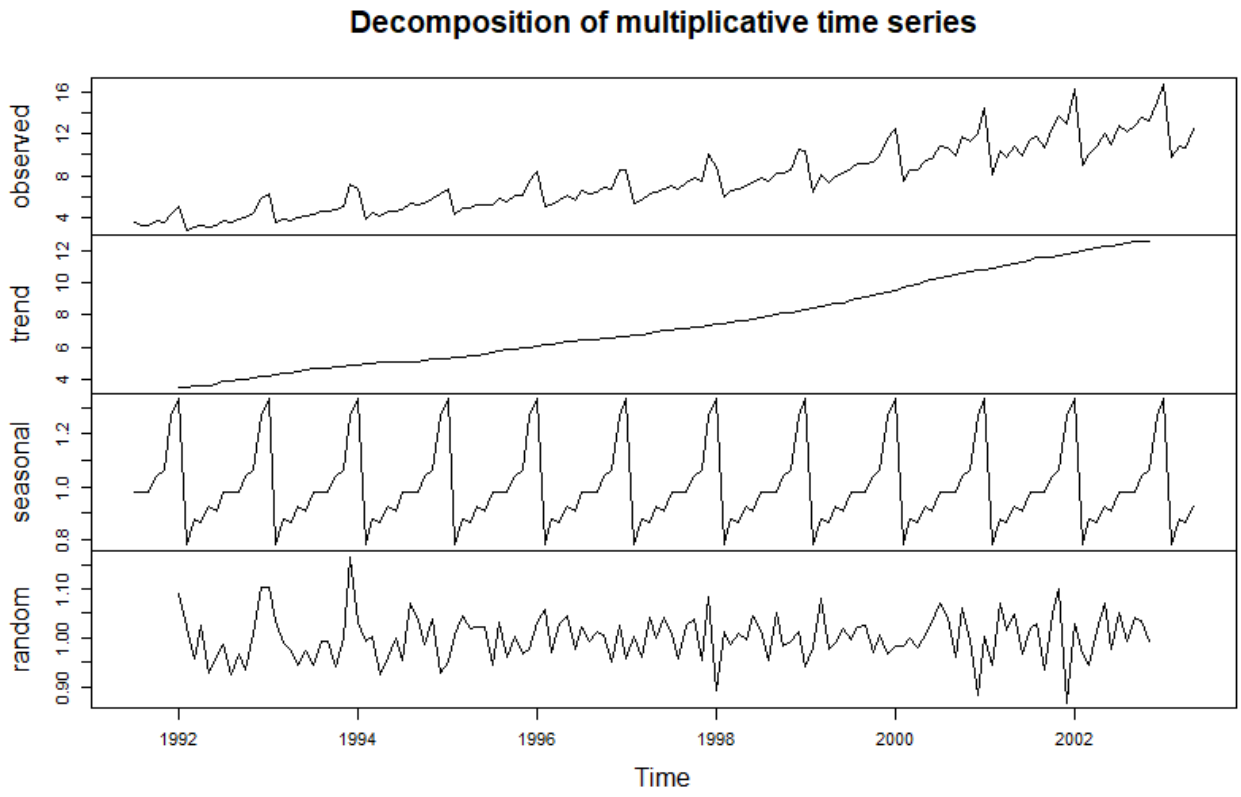
Split the train data into 70% and another 30% split into test data. 70% of the head data is in the train data while the last 30% of the tail data is in test data. After that, autoplot and auto layer to view the train and test data in a time series graph. The figure below shows the split of train and test data.



Step 1 : Visualizing our Time Series Dataset



Based on the figure above, it shows the times series plot of the sales for the drug against the dyear. The time plot starts from July 1991 to May 2003. The sales of drugs increase over time and it indicates a linear increase from year to year. The number of drugs increases year by year, so does the seasonal pattern. Both trend and seasonal components are multiplied and then added to the error component. From the plot, we can see that during (December to January), the number of drug sales is the highest. It will drop significantly in February and steadily increase afterward.



Based on the decomposition, there will be 4 components: observed, trend, seasonal and random. The random actually is the residue when the trend and seasonal effects were no longer inside the data. So the purpose of doing decomposition actually is to visualize the time series. So from the visualization , we can briefly know the pattern of the dataset chosen, so that we can select the suitable model for further analysis. From the trend graph we are able to see an increasing trend. The seasonal graph shows constant variance for each year. While in the random graph, when the data without the trend and seasonal effect showed a different pattern from the observed.

Observed

The observed graph is actually the time series plot. And it shows a positive climb in the number of drug sales every year, and this shows a multiplicative model as the seasonal variation becomes bigger every year and the trend is gradually rising.

Trend

Trend is a long term relatively smooth pattern of the drug sales against the month. The figure above shows a positive trend as the amount of alcohol sales increased from July 1991 to May 2003.

Seasonal

Seasonal is used to identify the seasonal factor of time, such as which month can be considered peak season in a year, and it occurs when there is a repeating month in the data that exhibits seasonality at regular intervals. From the seasonal chart we can see that there is a similar pattern (up and down) that repeats itself every year. This time series shows a very clear and distinct seasonal behavior. Therefore, we know approximately that we need to differentiate in a later step.

Random

The random graph shows periods of variability in the early and later years of the series. Especially in the years 1992, 1993, 1994, 1998, 2001 and 2002. Statistically, these residuals for the multiplicative model are calculated by dividing the observed number of drug sales and the estimated number of drug sales (trend) and the estimated seasonal component (seasonal). From the graph above we can see that the random graph has a mean of 1 and a constant variance. This is very important to ensure that a model adequately captures the information in the data.

Next, we will check the stationary of the data before moving to the model part. This is because non-stationary are more complex and require more calculations especially for the forecast purpose. So to stationarize the series , there are some common techniques such as detrending , differencing and seasonality. But for the detrend and the seasonality actually can briefly be seen in the decomposition visualization. So will check from many aspects to determine how many rounds of differencing needed for the time series.

Step 2a : Stationarize (Differencing)

In step 2, we have differencing our time series to stabilize the mean of our time series by removing changes in the level of a time series, and therefore eliminating trend and seasonality.

Original ACF

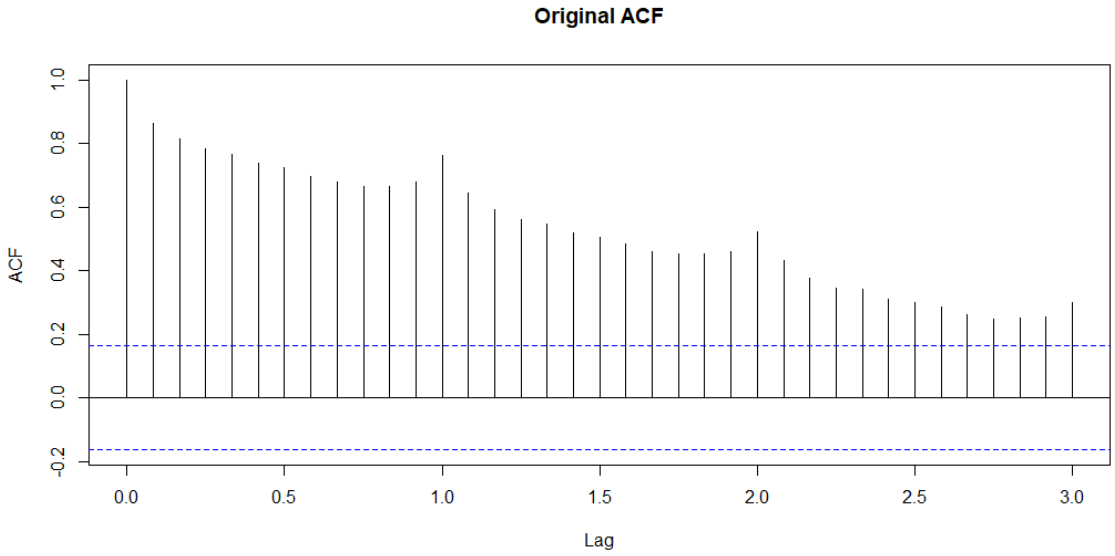


Figure 2.1 Original ACF

In order to determine the stability of the data, ACF had applied to the drug sales. The slow linear decay pattern found in the autocorrelation function(ACF) of the drug monthly sales per year indicates that the dataset is non- stationary. Most of the autocorrelations had shown that it is positive, and based on the original ACF figure we can observe that some seasonal patterns have the large values of 12th and 14th. Based on the observations, a seasonal difference was suggested.

Original PACF

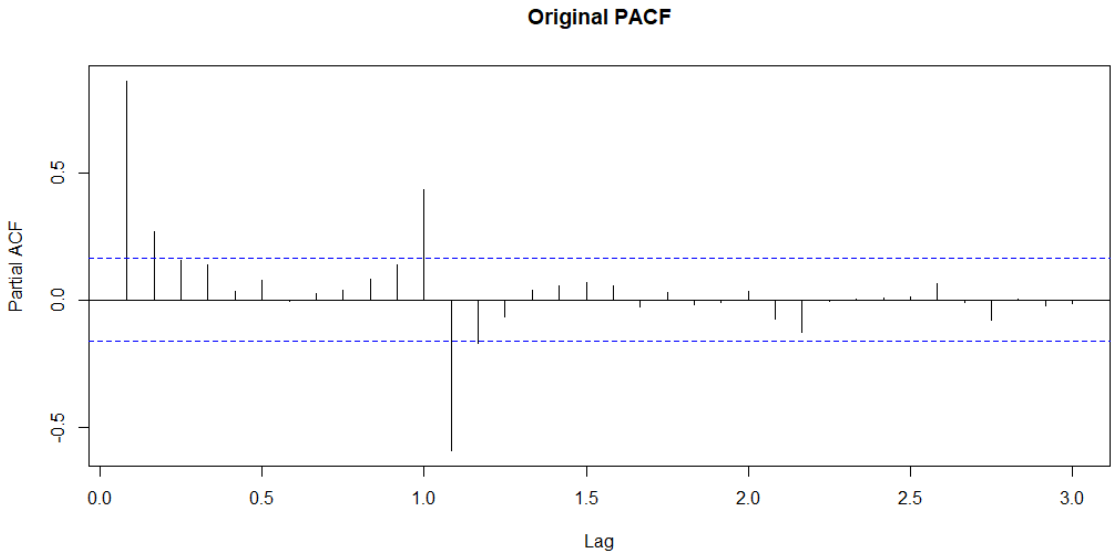


Figure 2.2 Original PACF

The partial autocorrelation plot had shown the higher order moving average in the data. We can observe that there are more than 1 spike after lag 0. Since lag 1.0 and lag 1.1 have the higher spikes, it can be concluded that both lag have correlated among each other that the result at lag 1.0 can affect the result in lag 1.1.

First Differencing

Hypothesis testing: Augmented-Dickey Fuller Test ($\alpha = 0.05$)

H0 : The time series is not stationary

H1 : The time series is stationary.

From figure 2.1, we can observe that our data has seasonal effects but not that obvious as the value in ACF spikes at every lag 12,24. Apart from that, the Augmented Dicket-Fuller(ADF) Test to check the stationarity with default lag order =5. The p-value is observed from figure 2.3 which is 0.257 which is larger than $\alpha = 0.05$, so we failed to reject H0 and conclude that the time series is not stationary after the first differencing. The time series has a clearer visualization of the seasonal effect and we decided to have second differencing which is non-seasonal differencing.

```
        Augmented Dickey-Fuller Test

data:  diff1
Dickey-Fuller = -2.7684, Lag order = 5, p-value = 0.257
alternative hypothesis: stationary
```

Figure 2.3 ADF Test of first differencing

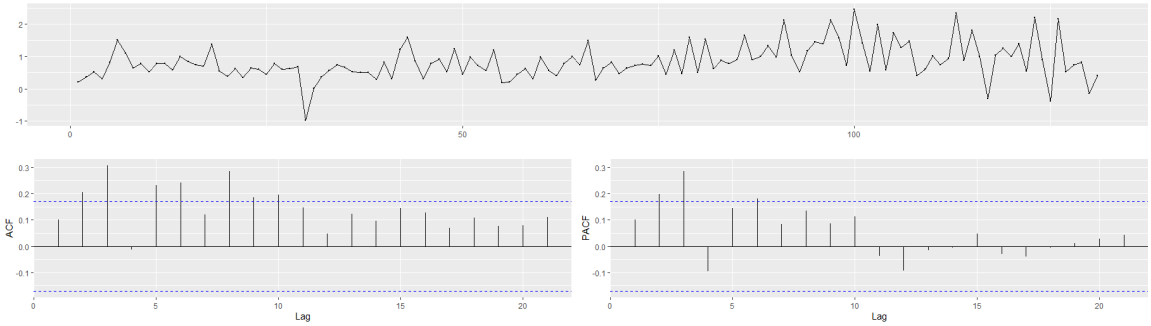


Figure 2.4 First Differencing

Second Differencing

Hypothesis testing: Augmented-Dickey Fuller Test ($\alpha = 0.05$)

H0 : The time series is not stationary

H1 : The time series is stationary.

Figure 2.5 shows our data after second differencing and without lag. ADF test is used to calculate the p-value with the default lag order which is lag order = 5. The p-value from the ADF test which is 0.01 and smaller than $\alpha = 0.05$. In this case, we can reject H0 and conclude that the time series is stationary after second differencing.

```
        Augmented Dickey-Fuller Test

data:  diff2
Dickey-Fuller = -7.8536, Lag order = 5, p-value = 0.01
alternative hypothesis: stationary
```

Figure 2.5 ADF Test of second differencing

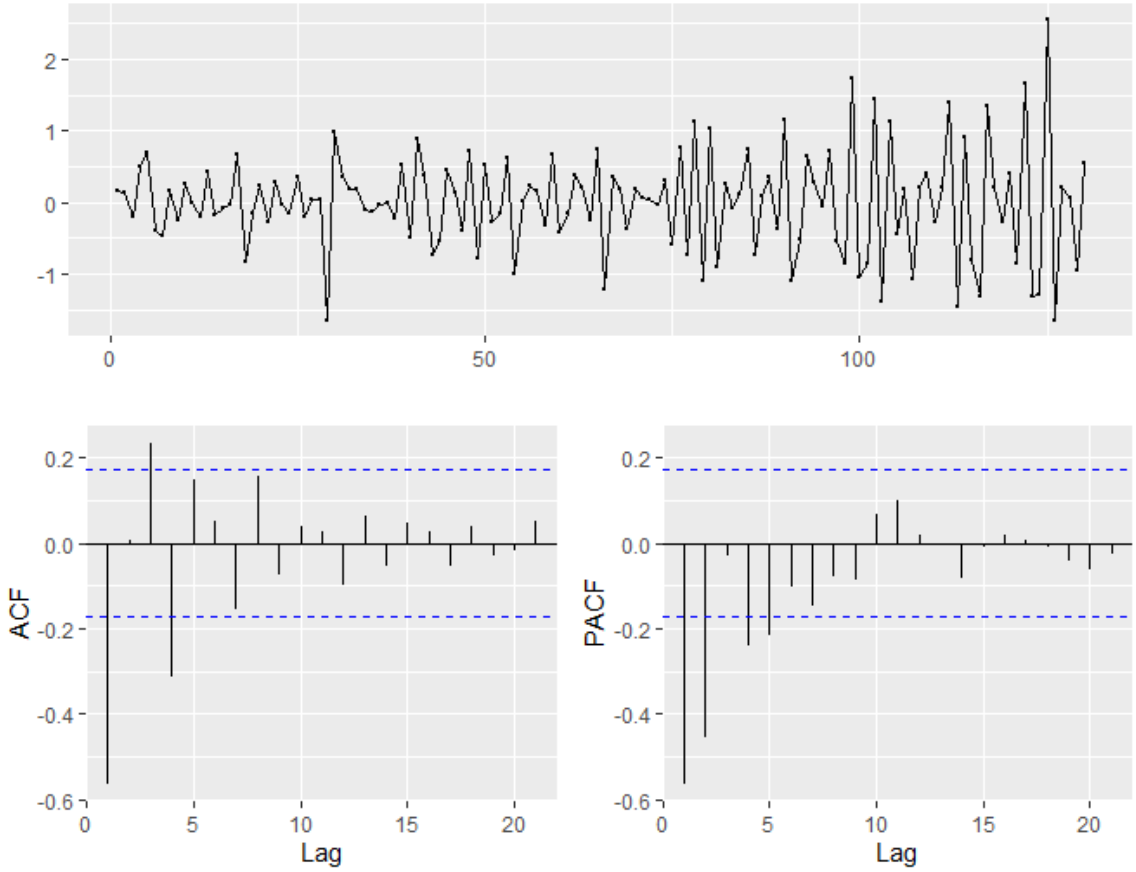


Figure 2.6 Second Differencing

Thus, we are confident that the trend and seasonal component had successfully been removed from our time series. The ACF plot and PACF plot from figure 2.6, will be used to make a manual guess on the orders of our ARIMA model (p,d,q)(P,D,Q)[12].

Step 3 : Find Optimal Parameters (Multiplicative Model)

Step 3a: Compare and Determine the best model

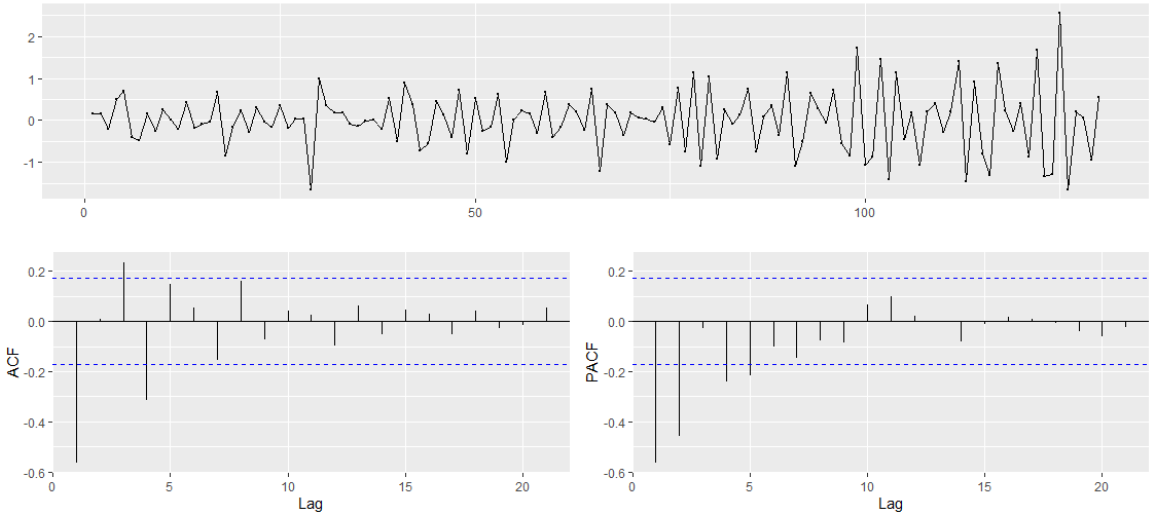


Figure 3.1 first seasonal and second non-seasonal differencing ACF and PACF

```
> manual_model<- arima(y, order =c(2,1,1), seasonal = list(order=c(0,1,0),period=12))
> summary(manual_model)

Call:
arima(x = y, order = c(2, 1, 1), seasonal = list(order = c(0, 1, 0), period = 12))

Coefficients:
          ar1      ar2      ma1
      -0.1710  -0.0407  -0.8336
s.e.   0.1122   0.1066   0.0742

sigma^2 estimated as 0.253:  log likelihood = -95.89,  aic = 199.77

Training set error measures:
                     ME      RMSE       MAE         MPE     MAPE      MASE         ACF1
Training set 0.0144006 0.4795537 0.3234603 0.05473026 4.000691 0.3414008 0.001892127
> manual_model$aic
[1] 199.7733
```

Figure 3.2 Manual mode 1 from acf and pacf

From figure 3.2 show that we found our model 1 manually. Based on the figure 3.1,we found the two spikes at PACF, so the p value is 2 and one spike at ACF, so the q value is 1. The ARIMA is (2,1,1) and the seasonal of our model is (0,1,0). We found no spike at

seasonal 12 in ACF and PACF. Based on our observations, our initial guess of the model is ARIMA(2, 1, 1)(0, 1, 0)[12]. The AIC value of the model is 199.7733.

```
> #manual model 2
> manual_model<- arima(y, order =c(2,1,1), seasonal = list(order=c(0,1,1),period=12))
> summary(manual_model)

Call:
arima(x = y, order = c(2, 1, 1), seasonal = list(order = c(0, 1, 1), period = 12))

Coefficients:
          ar1      ar2      ma1     sma1
      -0.2058  -0.0573  -0.8082  -0.2093
s.e.   0.1134   0.1064   0.0768   0.1035

sigma^2 estimated as 0.2448:  log likelihood = -94.02,  aic = 198.05

Training set error measures:
                     ME      RMSE       MAE        MPE    MAPE      MASE        ACF1
Training set 0.01766999 0.4717961 0.3269248 0.03010548 4.05141 0.3450575 0.003384292
> manual_model$aic
[1] 198.045
>
```

Figure 3.3 Manual model from acf and pacf

Figure 3.3 shows that we found our model 2 manually. Based on the figure 3.1,we found the two spikes at PACF, so the p value is 2 and one spike at ACF, so the q value is 1. The arima is (2,1,1) and the seasonal of our model is (0,1,1). We found no spike at seasonal 12, 24 and 36 in PACF, but we found that there was one spike at 36 at ACF.. Based on our observations, our initial guess of the model is ARIMA(2, 1, 1)(0, 1, 1)[12]. The AIC value of the model is 198.045. It is better than model 1.

```
> #manual model 3 (best)
> manual_model<- arima(y, order =c(4,1,1), seasonal = list(order=c(0,1,0),period=12))
> summary(manual_model)

Call:
arima(x = y, order = c(4, 1, 1), seasonal = list(order = c(0, 1, 0), period = 12))

Coefficients:
          ar1      ar2     ar3      ar4      ma1
      -0.1848  -0.0757  0.0439  -0.2649  -0.7781
s.e.   0.1190   0.1218  0.1105   0.0932   0.0985

sigma^2 estimated as 0.2335:  log likelihood = -90.79,  aic = 193.58

Training set error measures:
                     ME      RMSE       MAE        MPE     MAPE      MASE         ACF1
Training set 0.01185589 0.4606881 0.3251983 0.02170972 4.043746 0.3432353 -0.01698755
> manual_model$aic
[1] 193.5832
```

Figure 3.4 Best model found from acf and pacf

Figure 3.4 shows that we found our model 2 manually. Based on the figure 3.1, we found that four spikes at PACF, so the p value is 4 and one spike at ACF, so the q value is 1. The ARIMA is (4,1,1) and the seasonal of our model is (0,1,1). We found no spike at seasonal 12, 24 and 36 in ACF and PACF. Based on our observations, our initial guess of the model is ARIMA(4, 1, 1)(0, 1, 1)[12]. The AIC value of the model is 193.5832. It is better than model 2 and model 1.

```
> fit<-auto.arima(y, ic="aic", trace=TRUE)

 ARIMA(2,1,2)(1,1,1)[12]                    : 198.9571
 ARIMA(0,1,0)(0,1,0)[12]                    : 284.3108
 ARIMA(1,1,0)(1,1,0)[12]                    : 236.1755
 ARIMA(0,1,1)(0,1,1)[12]                    : 197.46
 ARIMA(0,1,1)(0,1,0)[12]                    : 198.2019
 ARIMA(0,1,1)(1,1,1)[12]                    : 197.9865
 ARIMA(0,1,1)(0,1,2)[12]                    : 197.7948
 ARIMA(0,1,1)(1,1,0)[12]                    : 198.132
 ARIMA(0,1,1)(1,1,2)[12]                    : 199.6432
 ARIMA(0,1,0)(0,1,1)[12]                    : 284.1451
 ARIMA(1,1,1)(0,1,1)[12]                    : 196.3334
 ARIMA(1,1,1)(0,1,0)[12]                    : 197.9182
 ARIMA(1,1,1)(1,1,1)[12]                    : 197.4393
 ARIMA(1,1,1)(0,1,2)[12]                    : 197.1347
 ARIMA(1,1,1)(1,1,0)[12]                    : 197.1666
 ARIMA(1,1,1)(1,1,2)[12]                    : 199.1309
 ARIMA(1,1,0)(0,1,1)[12]                    : 234.9612
 ARIMA(2,1,1)(0,1,1)[12]                    : 198.045
 ARIMA(1,1,2)(0,1,1)[12]                    : 198.534
 ARIMA(0,1,2)(0,1,1)[12]                    : 196.2679
 ARIMA(0,1,2)(0,1,0)[12]                    : 197.9019
 ARIMA(0,1,2)(1,1,1)[12]                    : 197.4475
 ARIMA(0,1,2)(0,1,2)[12]                    : 197.1799
 ARIMA(0,1,2)(1,1,0)[12]                    : 197.0732
 ARIMA(0,1,2)(1,1,2)[12]                    : 199.1756
 ARIMA(0,1,3)(0,1,1)[12]                    : 198.1404
 ARIMA(1,1,3)(0,1,1)[12]                    : 199.0806

 Best model: ARIMA(0,1,2)(0,1,1)[12]

> fit
Series: y
ARIMA(0,1,2)(0,1,1)[12]

Coefficients:
          ma1     ma2     sma1
      -1.0075  0.1559  -0.2081
s.e.   0.0859  0.0839   0.1044

sigma^2 = 0.2511:  log likelihood = -94.13
AIC=196.27   AICc=196.59   BIC=207.74
> summary(y)
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
  2.815   5.091   6.829   7.561   9.921  16.828
>
```

Figure 3.5 Find auto model using fit auto model

```
> #auto arima
> auto_model<- arima(y,order = c(0,1,2), seasonal = list(order= c(0,1,1), Period=12))
> auto_model

Call:
arima(x = y, order = c(0, 1, 2), seasonal = list(order = c(0, 1, 1), Period = 12))

Coefficients:
          ma1     ma2     sma1
      -1.0075  0.1559  -0.2081
s.e.   0.0859  0.0839   0.1044

sigma^2 estimated as 0.2453:  log likelihood = -94.13,  aic = 196.27
> auto_model$aic
[1] 196.2679
```

Figure 3.6 Using auto model to find aic

The Figure 3.5 and Figure 3.6 above show how we are using auto.arima to find the best model automatically. After that we found out that the suggested model by the auto.arima function is Arima(0, 1, 2)(0, 1, 1)[12]. The AIC value of the suggested auto model by the auto.arima function is 196.2679. It is worse than our model from Figure 3.4.

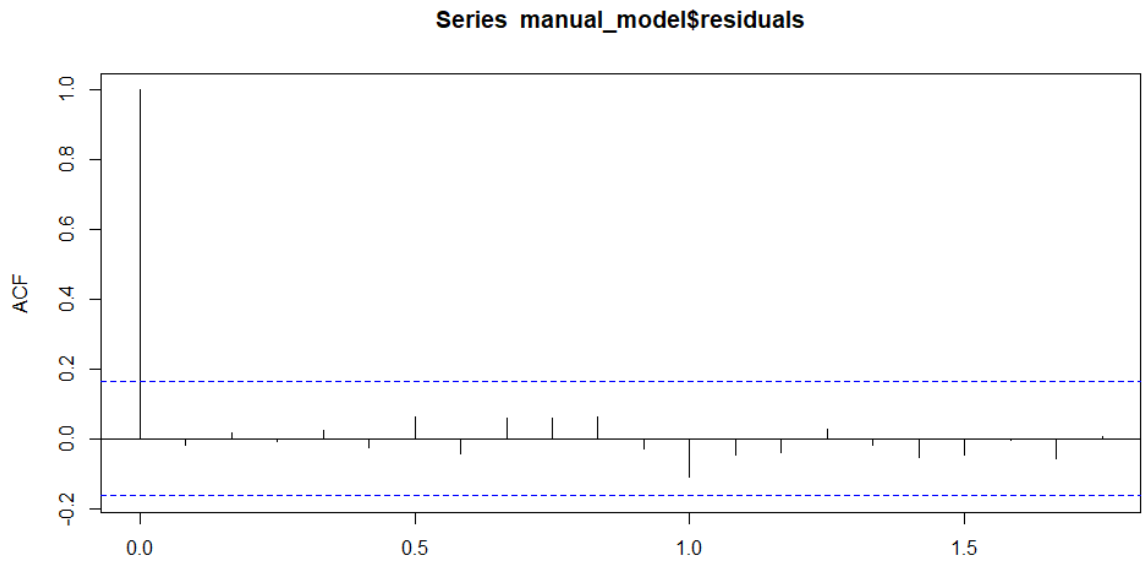


Figure 3.7 Best model from manual model with acf

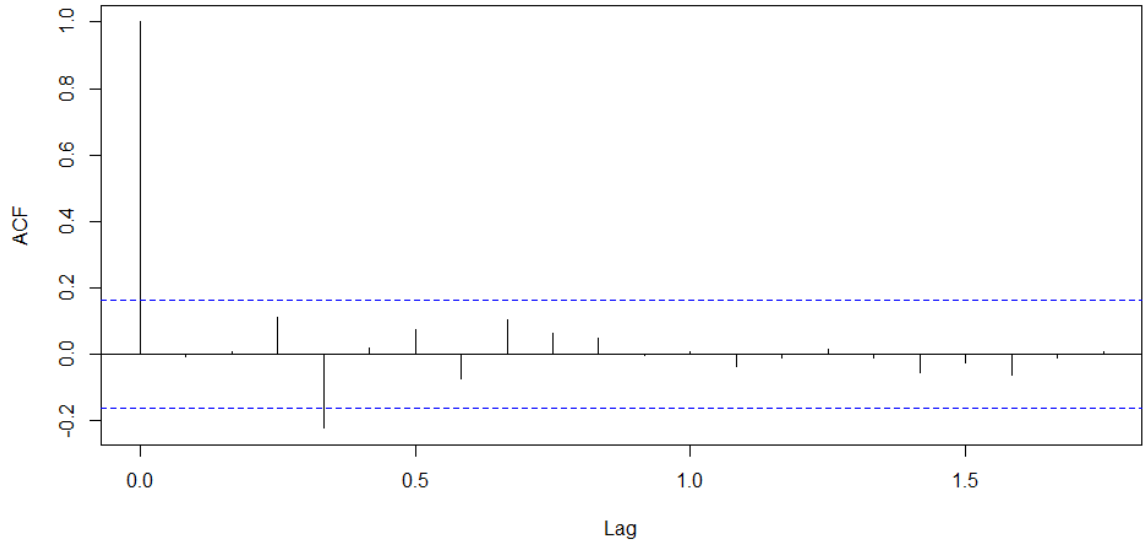


Figure 3.8 Best model from auto model with acf

From Figure 3.7 and Figure 3.8, we compare the model we found with manual model and auto.arima model by using acf. As we can see, Figure 3.7 is better than Figure 3.8. So we used both models to compare in our assignment.

```
> ets(y=y)
ETS(M,A,M)

Call:
 ets(y = y)

  Smoothing parameters:
    alpha = 0.1449
    beta  = 0.0071
    gamma = 1e-04

  Initial states:
    l = 3.1918
    b = 0.0511
    s = 0.9113 0.9276 0.8641 0.8722 0.777 1.3338
           1.2676 1.0573 1.0363 0.9801 0.9802 0.9925

  sigma:  0.055

     AIC     AICc      BIC
449.9963 454.8923 500.3646
> Box.test(y)

        Box-Pierce test

data:  y
X-squared = 106.56, df = 1, p-value < 2.2e-16
```

Figure 3.9

Figure 3.9, that is the step for finding error, trend, and seasonality of our model. It is an exponential smoothing function for forecasting. The ETS with "MAM" type is a multiplicative Holt-Winters' model with multiplicative errors and seasonality. The AIC value for this model is 449.9963.

Step 3b: Estimated model coefficients (best model)

```
> #model coefficients(manual model)
> arima(y,order = c(4,1,1),seasonal=list(order=c(0,1,0),period=12))

Call:
arima(x = y, order = c(4, 1, 1), seasonal = list(order = c(0, 1, 0), period = 12))

Coefficients:
          ar1      ar2     ar3      ar4      ma1
      -0.1848  -0.0757  0.0439  -0.2649  -0.7781
s.e.   0.1190   0.1218  0.1105   0.0932   0.0985

sigma^2 estimated as 0.2335:  log likelihood = -90.79,  aic = 193.58
```

Figure 3.11

The model's coefficient information is as below :

ar1, $\phi 1$ = -0.1848

ar2, $\phi 2$ = -0.0757

ar3, $\phi 3$ = 0.0439

ar4, $\phi 4$ = -0.2649

ma1, $\theta 1$ = -0.7781

Step 3c: Test the significance of coefficient

Manual model

```
> #test the significance of the coefficients (manual model)
> fit<-arima(y,order = c(4,1,1),seasonal=list(order=c(0,1,0),period=12))
> coeftest(fit)

z test of coefficients:

     Estimate Std. Error z value  Pr(>|z|)
ar1 -0.184834   0.118963 -1.5537  0.120253
ar2 -0.075684   0.121790 -0.6214  0.534317
ar3  0.043882   0.110507  0.3971  0.691300
ar4 -0.264948   0.093164 -2.8439  0.004456 **
ma1 -0.778065   0.098510 -7.8983 2.826e-15 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Figure 3.12

Hypothesis Testing: Coefficient Test **(α = 0.05)**

H0: Coefficient is not significant

H1: Coefficient is significant

From the figure 3.12 above, we can observe that the p-value of ϕ_1 , ϕ_2, ϕ_3, ϕ_4 and θ_1 are 0.120253, 0.534317, 0.691300, 0.004456 and 2.826e-15 (0.000000000000002826) respectively. Based on the p-values for all 5 models shown, only the p-value of ϕ_4, and θ_1 is lower than α = 0.05. Therefore, we will reject the H0 for ϕ_4, and θ_1 and conclude that this two coefficient are significant. For ϕ_1 , ϕ_2, and ϕ_3,, we will accept H0 and conclude that this three coefficient are not significant. By this, we also can conclude that the coefficient for ϕ_4 and θ_1 are important in the model while doing forecasting.

Auto ARIMA Model

```
> #test the significance of the coefficients (auto model)
> fit1<-auto_model
> coeftest(fit1)

z test of coefficients:

      Estimate Std. Error  z value Pr(>|z|)
ma1  -1.007470   0.085892 -11.7295  < 2e-16 ***
ma2   0.155927   0.083915   1.8582  0.06315 .
sma1 -0.208112   0.104444  -1.9926  0.04631 *
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Figure 3.13

Hypothesis Testing: Coefficient Test **(α = 0.05)**

H0: Coefficient is not significant

H1: Coefficient is significant

From the figure 3.13 above, it shows the test significance of the coefficients for the auto ARIMA model. We can observe that the p-value of θ_1 , θ_2 and Θ_1 are 2e-16 (0.0000000000000002), 0.06315, and 0.04631 respectively. Based on the p-values for all 3 models shown, only the p-value of θ_1 and Θ_1 is lower than α = 0.05. Therefore, we will

reject the H0 for θ_1 and Θ_1 and conclude that this two coefficient are significant. For θ_2, we will accept H0 and conclude that this coefficient are not significant.

Step 3d: Form Equation for the Best Model

```
z test of coefficients:

     Estimate Std. Error z value  Pr(>|z|)
ar1 -0.184834   0.118963 -1.5537  0.120253
ar2 -0.075684   0.121790 -0.6214  0.534317
ar3  0.043882   0.110507  0.3971  0.691300
ar4 -0.264948   0.093164 -2.8439  0.004456 **
ma1 -0.778065   0.098510 -7.8983 2.826e-15 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Figure 3.14 Coefficient Test

$$(1 - B - B^2 - B^3 - B^4)(1 - B^{12})Y = (1 + 0.184834B + 0.075684B^2 - 0.043882B^3 + 0.264948\,B^4)(1 + 0.778065B^{12})\varepsilon$$

$$(B^6 + B^5 - 2B^2 - B + 1)Y = (0.20614B^6 - 0.03414B^5 + 0.32383B^4 + 0.09993B^3 + 0.853749B^2 + 0.184834B + 1)\varepsilon$$

$$Y_t - Y_{t-1} - 2Y_{t-2} + Y_{t-5} + Y_{t-6} = \varepsilon + 0.184834\,\varepsilon_{-1} + 0.853749\,\varepsilon_{-2} + 0.09993\,\varepsilon_{-3} + 0.32383\,\varepsilon_{-4} - 0.03414\,\varepsilon_{-5} + 0.20614\,\varepsilon_{-6}$$

$$Y_t = Y_{t-1} + 2Y_{t-2} - Y_{t-5} - Y_{t-6} + \varepsilon + 0.184834\,\varepsilon_{-1} + 0.853749\,\varepsilon_{-2} + 0.09993\,\varepsilon_{-3} + 0.32383\,\varepsilon_{-4} - 0.03414\,\varepsilon_{-5} + 0.20614\,\varepsilon_{-6}$$

Step 4: Randomness

1. Manual Best Model

```
> #box test
> acf(manual_model$residuals)
> Box.test(manual_model$residuals, lag = 40)

        Box-Pierce test

data:  manual_model$residuals
X-squared = 23.958, df = 40, p-value = 0.9791
```

Figure 4.1 Box-Pierce Test (Manual Best Model)

Portmanteau Test: Box-Pierce Test for residual **(α = 0.01)**

$$H_0 : \varepsilon_t \sim NID(0, \sigma^2)$$
$$H_1 : \varepsilon_t \text{ do not follow } NID(0, \sigma^2)$$

In figure 4.1, we used the Box-Pierce Test to test whether the residual of our best model is a white noise(pure random) series or not. Using the test, we get a p-value of 0.9791 > 0.01. We fail to reject H0 and conclude that the residual of our model is white noise as it follows Normal Independent Distribution. This means that there are no remaining patterns in the residual and our model is a good fit.

2. Auto Best Model

```
> acf(auto_model$residuals)
> Box.test(auto_model$residuals, lag = 40)

        Box-Pierce test

data:  auto_model$residuals
X-squared = 37.653, df = 40, p-value = 0.5764
```

Figure 4.2 Box-Pierce Test (Auto Best Model)

Portmanteau Test: Box-Pierce Test for residual **(α = 0.01)**

$$H_0 : \varepsilon_t \sim NID(0, \sigma^2)$$
$$H_1 : \varepsilon_t \text{ do not follow } NID(0, \sigma^2)$$

In figure 4.2, we used the Box-Pierce Test to test whether the residual of our best model is a white noise(pure random) series or not. Using the test, we get a p-value of 0.5764 > 0.01. We fail to reject H0 and conclude that the residual of our model is white noise as it follows Normal Independent Distribution. This means that there are no remaining patterns in the residual and our model is a good fit.

Figure 4.1 and Figure 4.2 both achieve white noise but the value of the manual best model is higher than the auto best model which means manual best model is better than auto best model.

6.0 Results

Our final model will be ARIMA(4,1,1)(0,1,0)[12] :

$$Y_t = Y_{t-1} + 2Y_{t-2} - Y_{t-5} - Y_{t-6} + \varepsilon + 0.184834\ \varepsilon_{-1} + 0.853749\ \varepsilon_{-2} + 0.09993\ \varepsilon_{-3} + 0.32383\ \varepsilon_{-4} - 0.03414\ \varepsilon_{-5} + 0.20614\ \varepsilon_{-6}$$

```
> forecast(best_model,h=24)
         Point Forecast     Lo 80    Hi 80     Lo 95    Hi 95
Jun 2003       11.75833 11.139123 12.37754 10.811332 12.70533
Jul 2003       13.47463 12.854989 14.09426 12.526973 14.42228
Aug 2003       13.10374 12.478122 13.72936 12.146939 14.06054
Sep 2003       13.59832 12.955685 14.24096 12.615494 14.58115
Oct 2003       14.18159 13.536228 14.82696 13.194591 15.16860
Nov 2003       14.01715 13.357818 14.67648 13.008790 15.02550
Dec 2003       15.77505 15.108102 16.44201 14.755040 16.79507
Jan 2004       17.51700 16.846682 18.18732 16.491837 18.54216
Feb 2004       10.51816  9.833569 11.20275  9.471170 11.56515
Mar 2004       11.49811 10.809389 12.18682 10.444804 12.55141
Apr 2004       11.36573 10.671252 12.06021 10.303616 12.42785
May 2004       13.20943 12.507276 13.91159 12.135576 14.28329
Jun 2004       12.44643 11.456925 13.43593 10.933114 13.95974
Jul 2004       14.17657 13.179559 15.17358 12.651774 15.70136
Aug 2004       13.79512 12.781923 14.80832 12.245568 15.34467
Sep 2004       14.29402 13.253834 15.33421 12.703190 15.88486
Oct 2004       14.88030 13.839400 15.92119 13.288383 16.47221
Nov 2004       14.71083 13.645789 15.77588 13.081989 16.33968
Dec 2004       16.47243 15.390074 17.55478 14.817110 18.12775
Jan 2005       18.21306 17.118729 19.30739 16.539426 19.88669
Feb 2005       11.21317 10.094895 12.33144  9.502918 12.92341
Mar 2005       12.19490 11.063959 13.32584 10.465275 13.92452
Apr 2005       12.06124 10.915713 13.20677 10.309307 13.81317
May 2005       13.90535 12.743220 15.06747 12.128027 15.68266
```

Figure 6.1 forecast value for next two years

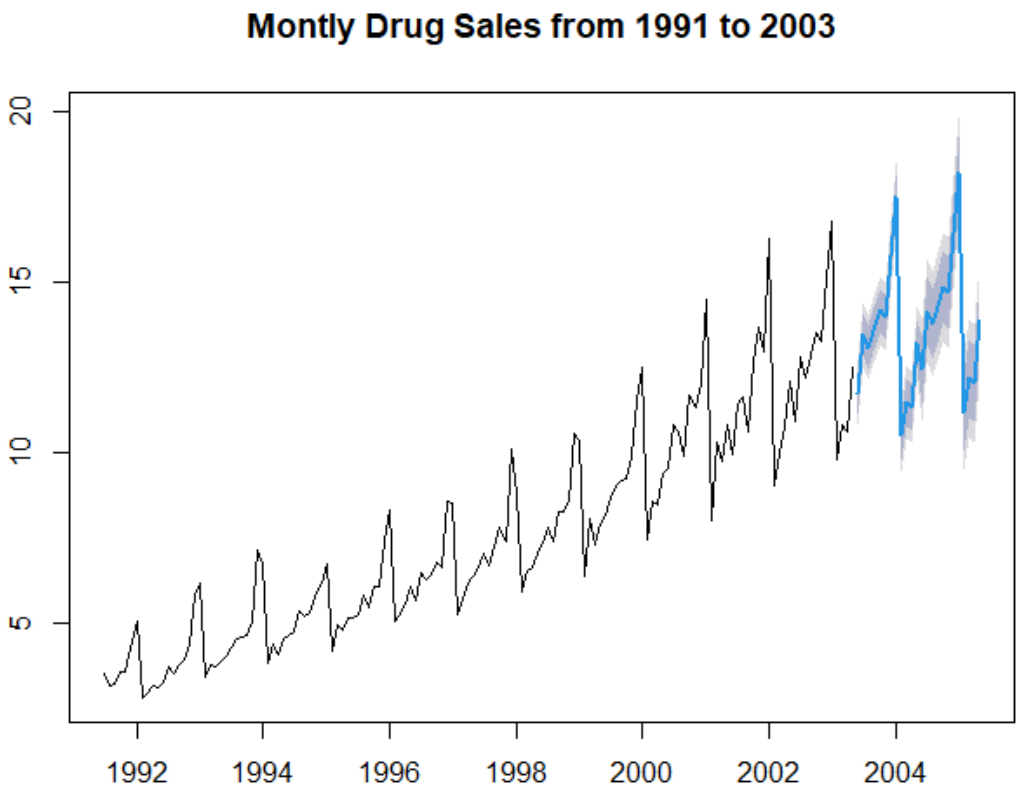


Figure 6.1 Forecast with ARIMA(4,1,1)(0,1,0)[12]

The figures above had shown the forecast value of Monthly Drug Sale. Based on the figures, we can observe that the drug sales have some seasonality and have a slightly increasing trend. We had forecasted that the trend of the drug sales will increase in the next two years which is 2004 and 2005.

7.0 Discussion and interpretations

As the result above, we can see that the forecasting result gradually increased until January 2004. It started to drop significantly in February 2004 and increased afterward until 2005. The next cycle is the same as before. The reason it reached its peak in December or January might be because of the seasonal effect. Seasonal influenza (flu) viruses are detected year-round in the United States, flu viruses typically circulate during the fall and winter during what's known as the flu season. The exact timing and duration of flu seasons varies, but flu activity often begins to increase in October. Most of the time flu activity peaks between December and February, although significant activity can last as late as May. Due to factors of flu season, the drug sales mostly brought in October until January. In February the drug sales have significantly dropped to the lowest point.

Limitation and recommendations for future work

The limitation of our project is that our dataset only provides the data until 2008. Other than that, we could not find the latest dataset with 2020 or 2021 data. This makes us unable to make an up-to-date forecasting for the dataset. Besides that, we should use the more recent data of 2020 and 2021 online to make more recent forecasts so our model can be more applicable in the real world.

8.0 Conclusion

By concluding the forecast graph, the best model for this dataset is ARIMA(4,1,4)$(0,1,0)_{12}$. By using this model to do forecasting, it has predicted an obvious increase for the upcoming two years based on the graph. We can conclude that for this project assignment, the most suitable model is ARIMA compared to other models.

On the other hand, the expansion estimation model equation for the which we define as the best model is shown as below:

$$Y_t = Y_{t-1} + 2Y_{t-2} - Y_{t-5} - Y_{t-6} + \varepsilon + 0.184834\,\varepsilon_{-1} + 0.853749\,\varepsilon_{-2} + 0.09993\,\varepsilon_{-3} + 0.32383\,\varepsilon_{-4} - 0.03414\,\varepsilon_{-5} + 0.20614\,\varepsilon_{-6}$$

In a nutshell, we had fulfilled and achieved the objective which were stated in the earlier part of this report. The most important thing is we are able to find the best model with all the previous procedures. In the last part, we also completed our forecast successfully.

Throughout this project assignment, we had a better understanding of time series forecasting and how it works overall. Even though this is the first time we touch on this topic and we are facing some trouble in the middle of the process, we are grateful that we have this opportunity to learn new things. We would like to specially say thank you to our lecturer, Dr Chin Wan Yoke and tutor Dr. Tan Pei Ling for the patience guidance along the study of statistics for data science.

9.0 Reference

Brownlee, J. (2018), *A Gentle Introduction to SARIMA for Time Series Forecasting in Python,* viewed 29 September 2022, <https://machinelearningmastery.com/sarima-for-time-series-forecasting-in-python/#:~:text=A%20seasonal%20ARIMA%20model%20uses,average%20terms%20at%20lag%20s>.

Brownlee, J. (2017), *How to Decompose Time Series Data into Trend and Seasonality* , viewed 29 September 2022, <https://machinelearningmastery.com/decompose-time-series-data-trend-seasonality/#:~:text=Time%20series%20decomposition%20involves%20thinking,time%20series%20analysis%20and%20forecasting>.

Chourasia, A (2020), *Decomposition in Time Series Data,* viewed 29 September 2022, <https://medium.com/analytics-vidhya/decomposition-in-time-series-data-b20764946d63>.

Esprabens, J. , & Arango, A., & Kim, J.(2022), *Time Series for Beginners*, viewed 29 September 2022,<https://bookdown.org/JakeEsprabens/431-Time-Series/modelling-time-series.html#ar-and-ma>.

Hayes, A (2022), *Time Series Definition*, viewed 29 September 2022, <https://www.investopedia.com/terms/t/timeseries.asp>.

Hyndman, R.J., & Athanasopoulos, G. (2018) *Forecasting: principles and practice*, 2nd edition, OTexts: Melbourne, Australia. OTexts.com/fpp2. Accessed on <29 September 2022>.

Influx Data(2022), *What is time series data?*, viewed 27 September 2022, <https://www.influxdata.com/what-is-time-series-data/#:~:text=Time%20series%20data%20can%20be%20classified%20into%20two%20types%3A,at%20irregular%20time%20intervals%20(events)>.

Mehandzhiyski, V. (2020), *What is a Moving Average Model?,* viewed 29 September 2022, <https://365datascience.com/tutorials/time-series-analysis-tutorials/moving-average-model/>.

Minitab® 21 Support (2022), Interpret the partial autocorrelation function (PACF), viewed 29 September 2022, <https://support.minitab.com/en-us/minitab/21/help-and-how-to/statistical-modeling/time-series/how-to/partial-autocorrelation/interpret-the-results/partial-autocorrelation-function-pacf/>.

EPM Information Development Team. (2017), *Autocorrelation Statistics,* viewed 29 September 2022, <https://docs.oracle.com/cd/E57185_01/CBREG/ch06s03s03s03.html>.

Quanstart (2021), '*Autoregressive Integrated Moving Average ARIMA(p, d, q) Models for Time Series Analysis*', viewed 29 September 2022 <https://www.quantstart.com/articles/Autoregressive-Integrated-Moving-Average-ARIMA-p-d-q-Models-for-Time-Series-Analysis/>.

Rochelle P. Walensky, MD, MPH (2022), *Influenza (Flu),* viewed 29 September 2022,, <https://www.cdc.gov/flu/about/season/flu-season.htm>.

Shetty, C. (2020), *Time Series Model AR, MA, ARMA, ARIMA,* viewed 28 September 2022 <https://towardsdatascience.com/time-series-models-d9266f8ac7b0>.

Stephanie, G.(2022), *ARMA model*, viewed 29 September 2022, <https://www.statisticshowto.com/arma-model/>.

Stephanie, G.(2022), *Autoregressive Model: Definition & The AR Process*, viewed 29 September 2022, < https://www.statisticshowto.com/autoregressive-model/>.

Stūris, J. (2021), *Logarithmic transformation in R, inverse logarithmic transformation in R*, viewed 29 September 2022, <https://datacornering.com/logarithmic-transformation-in-r-inverse-logarithmic-transformation-in-r/>.

Wikipedia (2011), *Ljung–Box test*, viewed 29 September 2022, <https://en.wikipedia.org/wiki/Ljung%E2%80%93Box_test>.

10.0 Appendix

```
library(forecast)
library(lmtest)
library(ggplot2)
library(tseries)
library(MLmetrics)
library(graphics)
library(lmtest)

#read file
data <- read.csv("C:\\Users\\tanja\\Downloads\\Statistic Assignment\\Drug Sales.csv")

#label y
y <- data$Monthly.drug.sales
y <- ts(y, start =c(1991,7), frequency=12)

#Split data
train <- head(y, round(length(y) * 0.70))
h <- length(y) - length(train)
test <- tail(y, h)
train
test
autoplot(train) + autolayer(test)

#plot graph
y <- data$Monthly.drug.sales
y <- ts(train, start =c(1991,7), frequency=12)
plot (y,ylab="Sales", main="Montly Drug Sales from 1991 to 2003")
ggtsdisplay(y)
adf.test(y, k=12)

#plot original acf and pacf
acf(y,main = "Original ACF",lag =36)
pacf(y,main="Original PACF", lag= 36)

#decomposition
components <- decompose(y, type = "multiplicative")
plot(components)
cbind(components$x,components$trend,components$seasonal,components$random)

#log plot
y1<-log(y);y1
y1<-ts(y, frequency = 12, start=c(1991,7))
plot(y, ylab="Log(Drug Sales)",main="Montly Drug Sales from 1991 to 2003")
```

```r
#seasonal differencing
diff1<-ts(diff(y, lag=12, lag.max=40))
ggtsdisplay(diff1)
adf.test(diff1)

#non-seasonal differencing
diff2<-ts(diff(diff1))
ggtsdisplay(diff2)
adf.test(diff2)

#plot acf and pacf
acf(diff2,main = "ACF Plot",lag =36)
pacf(diff2,main="PACF Plot", lag= 36)

#manual model 1
manual_model<- arima(y, order =c(2,1,1), seasonal = list(order=c(0,1,0),period=12))
summary(manual_model)
manual_model$aic

#manual model 2
manual_model<- arima(y, order =c(2,1,1), seasonal = list(order=c(0,1,1),period=12))
summary(manual_model)
manual_model$aic

#manual model 3 (best)
manual_model<- arima(y, order =c(4,1,1), seasonal = list(order=c(0,1,0),period=12))
summary(manual_model)
manual_model$aic

#box test
acf(manual_model$residuals)
Box.test(manual_model$residuals, lag = 40)

#find best fit
fit<-auto.arima(y, ic="aic", trace=TRUE)
fit
summary(y)

#auto arima
auto_model<- arima(y,order = c(0,1,2), seasonal = list(order= c(0,1,1), Period=12))
auto_model
auto_model$aic
acf(auto_model$residuals)
Box.test(auto_model$residuals, lag = 40)
```

```r
#best model -> auto model
best_model<-manual_model

#ets approach
ets(y)
Box.test(y)

#arima approach
fit<- arima(y,order = c(4,1,1),seasonal=list(order=c(0,1,0),period=12))
        Box.test(fit$residuals, lag=24)

#Alternative auto models
fit<-auto.arima(y)
summary(y)
auto.arima(y,ic="aic",trace=TRUE)

#accuracy manual model
accuracy(best_model)

#accuracy auto model
accuracy(auto_model)

#model coefficients(manual model)
arima(y,order = c(4,1,1),seasonal=list(order=c(0,1,0),period=12))

#test the significance of the coefficients (manual model)
fit<-arima(y,order = c(4,1,1),seasonal=list(order=c(0,1,0),period=12))
coeftest(fit)

#test the significance of the coefficients (auto model)
fit1<-auto_model
coeftest(fit1)

#residual manual model
r<-resid(best_model)
plot(r)
acf(r)

#residual auto model
d<-resid(auto_model)
plot(d)
acf(d)



#forecast 1
forecast(best_model,h=24)
plot(forecast(best_model,h=24), main="Montly Drug Sales from 1991 to 2003")

#forecast 2
fit <- ets(y)
plot(forecast(fit))
```