Mini Project - Australian Gas Production

Name : Numer P

Table of Contents

1. Project Objective

The main objective of the report is to explore the Australian Gas Production Dataset ("from package::forecast") in R and generate insights about the data set. This exploration report will consist of the following,

- Importing dataset in R
- Understanding the structure of Dataset
- Graphical exploration
- Time Series Analysis

2. Assumptions

Australian gas production dataset is imported through the package called forecast to work on time series analysis. The dataset is univariate and gives the production of gas in Australia for the period Jan 1956 to Aug 1996. The production of gas in particular year is increased and again it is constant for the certain period. Time series analysis will show the forecast of gas production will be increased or decreased for the upcoming periods.

The ARIMA function will tells about the forecasted time series and provides the information for the growth of gas production in the particular time period. With forecasted values, the prediction will take for the similar production of gas or the dissimilarity exists in the Australian gas production for the period January 1970 to August 1995.

3. Exploratory Data Analysis – Step by Step Approach

1. Environment Set up and Data Import
2. Variable Identification
3. Univariate Analysis
4. Variable Transformation
5. Feature Exploration

3.1 Environment Setup and Data Import

3.1.1 Install necessary packages and Import Libraries

This section is used to install packages and invoke the associated libraries. Having all packages at the same places increase code readability.

3.1.2 Setup Working Directory

Setting a working directory on starting of the R session makes importing and exporting data files and code files easier. Basically, working directory is the location/ folder on the PC where you have the data, codes etc. related to the project.

Please refer Appendix A for source code.

3.1.3 Import and read the dataset

The Australian Gas Production dataset is imported from package forecast.

Please refer Appendix A for source code.

3.2 Variable Identification

- setwd() used for setup working directory to export data and files from the folder or location in PC.
- getwd() used to identify the location was correctly entered or not.
- Library function is used to load the installed packages like forecast, fpp2, quantmod, t series.
- class function is used to check the format of dataset.
- Accuracy function is used to identifies the error measures in the predicted variables.
- Forecast function is used to forecast the time series for the set of period.
- Window function is used to partition of the datasets.
- ts function is used to convert the dataset into time series data.
- acf function is used to analyse the auto correlation of the variable.
- pacf function is used to analyse the partial correlation of the variable.
- ts.plot is used to plot the time series data.
- Decompose function is used to find the components of time series data.
- stl function is used to decompose the components of time series data into seasonal, trend and irregular components using loess.
- Monthplot function is used to plot the time series data in month wise.
- Periodicity function is used to find the period of the dataset.
- Log function is used to find the log values of the time series data.

3.2.1 Variable Identification – Inferences

#getwd()

It shows the working directory dataset

#library(forecast)

(forecast) Methods and tools for analysing the univariate time series data.

#library(tseries)

(tseries) is used for the time series data analysis and computational data.

#library(fpp2)

(fpp2) is used for the forecast principles and practices

#library(quantmod)

(quantmod) is used for the quantitative financial modelling and trading.

#class

class function describes the full file in data.frame format. As the files includes category in season variables, it shows the values as character format.

#summary

It produces the results as summarised format for each variable.

#hist

This function is used to plot the histogram for the variables.

#accuracy

The accuracy function returns the values of the forecasted measures.

#box.test

Used to identify the Portmanteau test on the residuals.

#log

The function is used for get the log values for the variables.

#acf

Auto correlation of the datasets is predicted using this function.

#pacf

Partial auto correlation of the datasets is predicted using this function.

#diff

This function is used to convert the non-standardized datasets into standardized datasets.

#window

It is used in the datasets for the partition of the datasets into train and test.

3.3 Univariate Analysis

Univariate analysis is the analysis of data of one variable at time and it involves whether the datasets are descriptive or inferential statistics.

1. Read the data as a time series object in R. Plot the data.

```
> library(forecast)
```

The library forecast is installed for the analysis of univariate time series data and predicts the forecasted value of the time series.

```
Registered S3 method overwritten by 'quantmod':
  method           from
  as.zoo.data.frame zoo
```

```
> library(tseries)
```

The library tseries is installed for ARIMA analysis in the stationary data.

```
'tseries' version: 0.10-47

   'tseries' is a package for time series analysis and
computational finance.

   See 'library(help="tseries")' for details.
```

```
> library(fpp2)
```

The time series plot is generated by the package fpp2 and it is advanced package of ggplot which helps in showing the time series data.

```
Loading required package: ggplot2
Loading required package: fma
Loading required package: expsmooth
```

```
> library(quantmod)
```

The quantmod package is installed for the identification of periodicity of the time series data.

```
> mydata=forecast::gas
```

The gas dataset is imported form the package forecast and it is read in the object called mydata for the analysis.

```
> help(gas)
```

Description
Australian monthly gas production: 1956–1995.
Usage
gas
Format
Time series data

The dataset is already in time series format and period of the data is 1956-1995. The source of the data is collected from Australian Bureau of Statistics.

```
> class(mydata)
```

```
[1] "ts"
```

The class of the object is returned with time series data and it takes for further analysis of the time series analysis and forecasting.

```
> start(mydata)
```

```
[1] 1956    1
```

Gas dataset is starting with the January 1956.

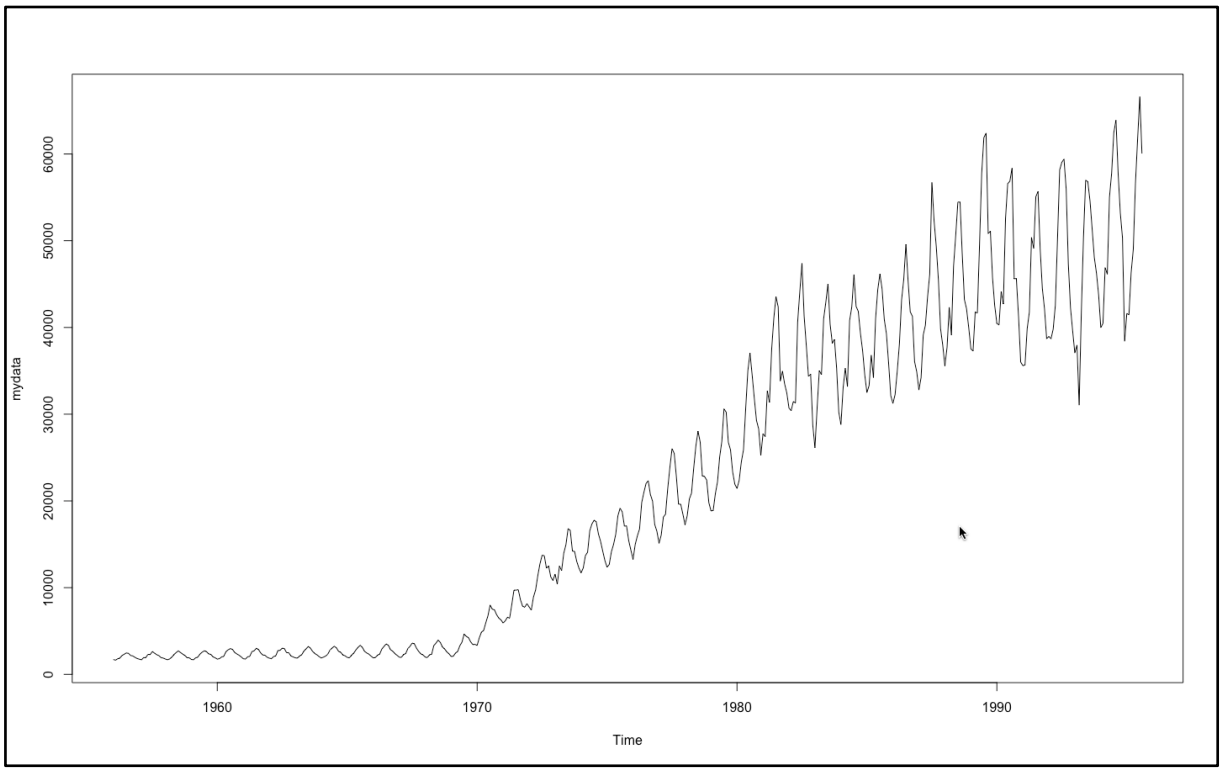
```
> end(mydata)
```

```
[1] 1995    8
```

The time series data is ended with August 1995.

Australian Gas Production shows the datasets are recorded from January 1956 to August 1995. The Datasets are analysed for the components present in the time series data.

```
> plot(mydata)
```

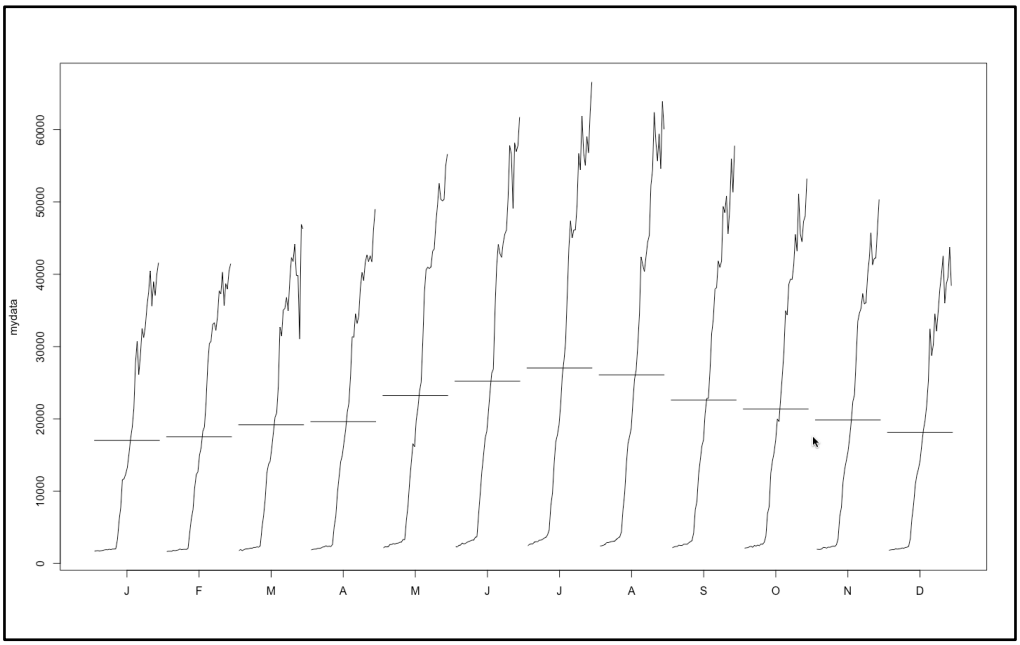


The graph showing the time in x axis and the data on y axis. The time series plot shows that the dataset is univariate and the data are plotted from 1956 to 1995.

Initially the graph showing that the gas production is constant over the period for 1956 to 1969. The mean for the time series is collected as same and from the year 1970, the plot showing there is some changes happened in the production of gas. The spikes are showing that the gas production for the particular time it is increasing and the spikes down are showing that the production of gas is decreasing for the certain period. The gas production attains its highest growth at July 1995 and slight drop in the end of August 1995. This shows that data has constant mean and variance are different of the variables. The plot shows there is increasing trend as well as increasing seasonality of the gas production.

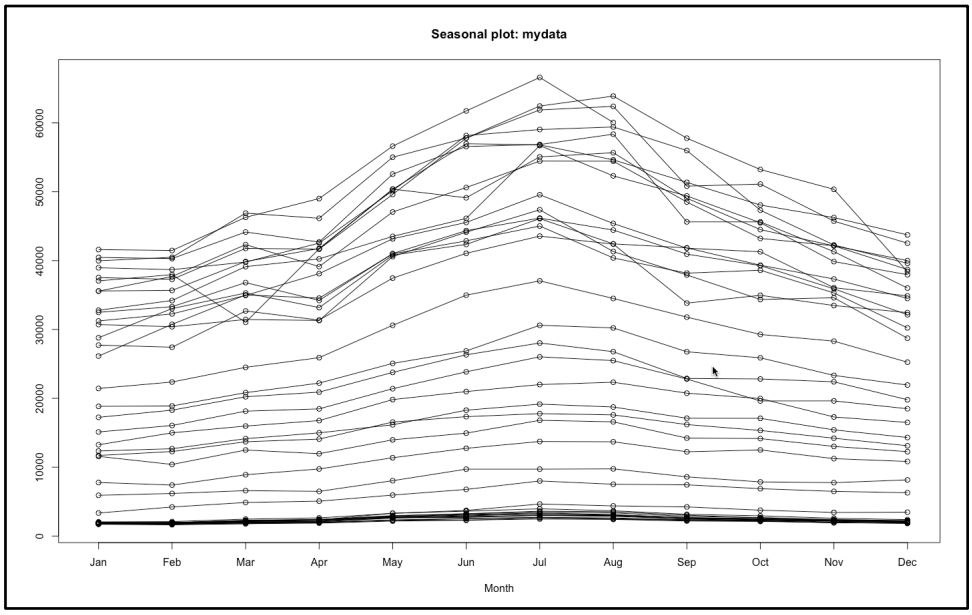
> monthplot(mydata)

Month plot shows how trends are continuously increasing over the period and it will determine the seasonality of the data present in the time series.



The Month Plot shows that the gas production is increasing for the period July Month and when compared for the other months the seasonality is some time constant, increasing, decreasing according to the months. The gas production in December, January, February, March are constant over the period and the production seasonality are fluctuations in the increasing year trend. The average mean are calculated in the medial of plots and the highest trend and seasonality showing in the time is for July month and it shows that slight seasonality is present in the dataset.

> forecast::seasonplot(mydata)



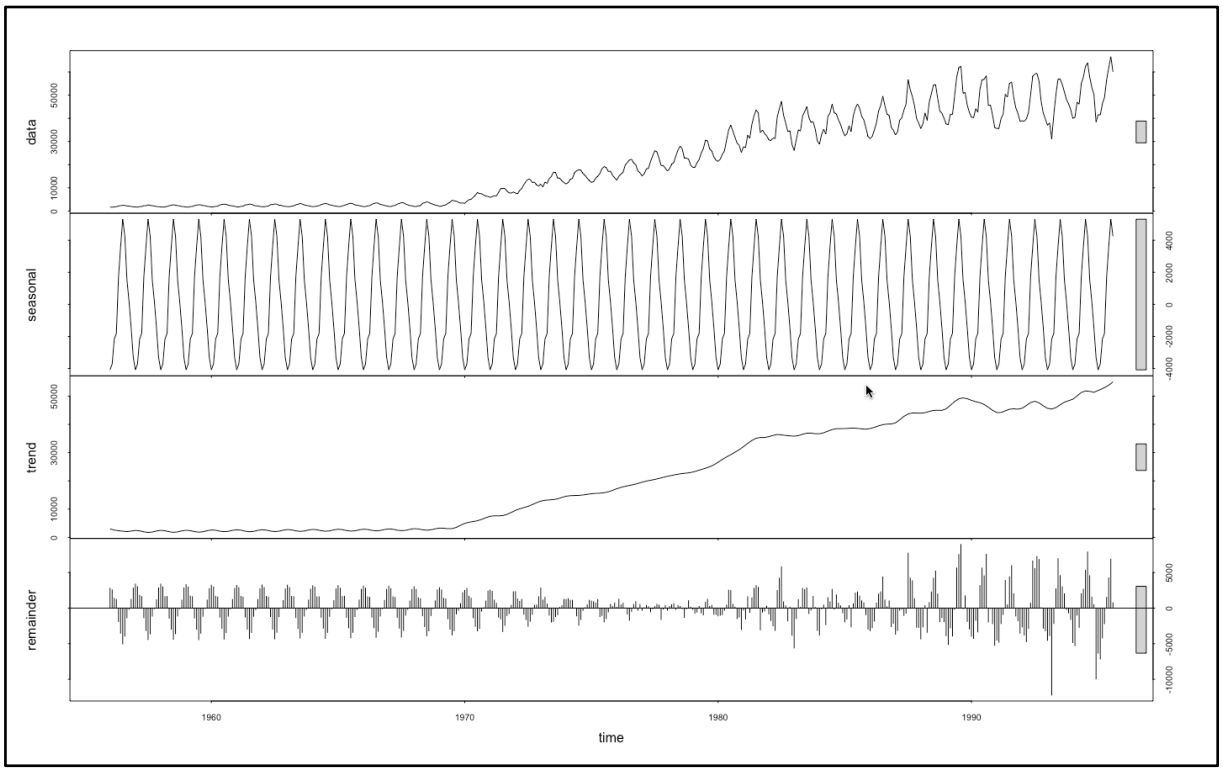
The seasonal plot shows that the various values are showing in the datasets are configured by the seasonality present in the data. The seasonal variations shows that the month July is reporting with the higher season and the values are constant over some initial years and the seasonality is increasing in the upcoming years and the last year 1995 shows the maximum seasonality and this plot that the slight seasonality is present.

2. What do you observe? Which components of the time series are present in this dataset?

> stl_for_mydata=stl(mydata,s.window = "periodic")#constant seasonality changes

The stl function is used to find the components present in the dataset. The periodic seasonality changes in the dataset shows trend, seasonality and residuals. The additive seasonality is taken for decompose of the gas production.

> plot(stl_for_mydata)

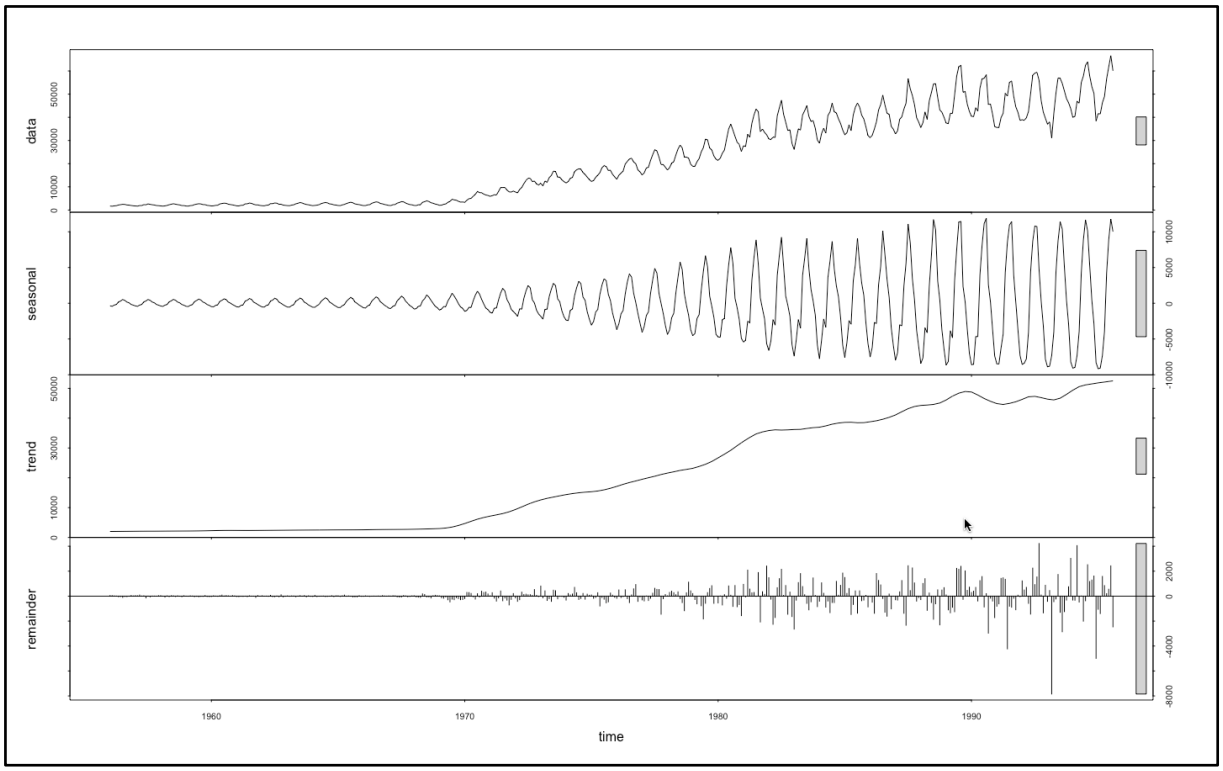


The scale of the dataset is small and the values are measured from 0 to 50000. Seasonality showing the constant changes around the dataset and the values are measure with -4000 to +4000 with large scale to predict the values. The trend is continuously increasing from the years and the values are treated with less scaling measure of 0 to 50000. The higher values are predicted with the residuals are increasing negatively in the constant period as it indicates the seasonality changes in the gas production.

The Main components present in the Australian Gas Production dataset is Trend, Seasonality, Residuals. The values are predicted by the periodic changes in the seasonality.

> stl_for_mydata7=stl(mydata,s.window = 7)#seasonality changes

The decompose stl function is now predicted with the value 7 as it indicates the seasonality changes in the dataset. The residuals are picked from the original dataset values and the values are measured with different scaling methods.



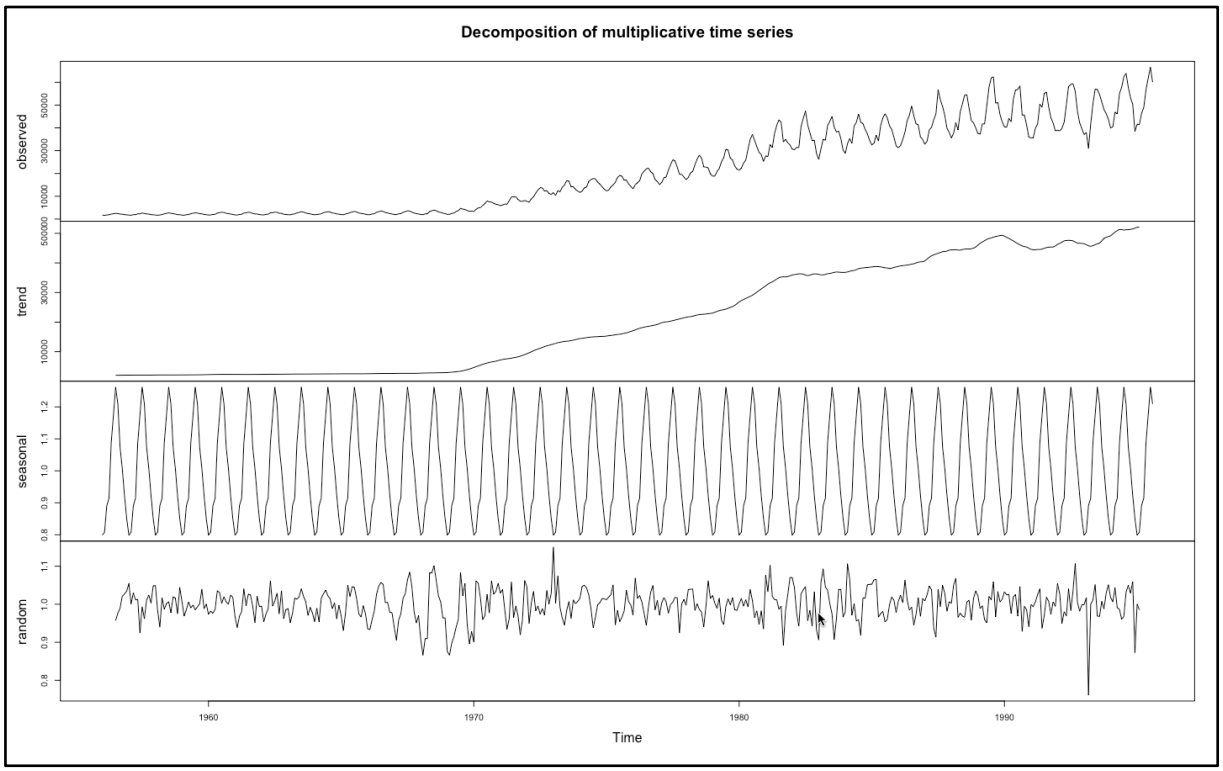
The seasonality shows that the increasing factors shows increased in values and scale measured the -10000 to +10000 of the variables. Increasing factor of seasonality indicates that multiplicative seasonality is calculated. Trend shows the increasing factor with small scale measures the higher value 0 to 50000. Residuals are making the significance results with seasonality and trend as it shows the initial stage of residuals are reported null and the factor increasing in seasonality is showing the increased factors in the residuals as well in trend.

This plot significant with increasing trend, multiplicative seasonality and the increased residuals reported in increased seasonality of factors.

```
> components_of_my_data=decompose(mydata,type = "multiplicative")#since seasonality is in multiplicative
```

Decompose function is the basic function which shows the various components present in the gas production dataset. The decomposed factors are analysed by the univariate dataset and makes the various functions on the datasets.

> plot(components_of_my_data)



The decomposition chart sources the univariate series is having the increase time series and the seasonality of the data is constant over the time period which shows the residuals are diversified by the increased factor value in 1975 and the negative factor residuals are measured in 1995 of the time series data. The decomposition of the time series is displaying the increased time series factor and seasonality and the verified residuals with impact from the trend and seasonal of the datasets.

3. What is the periodicity of dataset?

> periodicity(mydata)

Monthly periodicity from Jan 1956 to Aug 1995

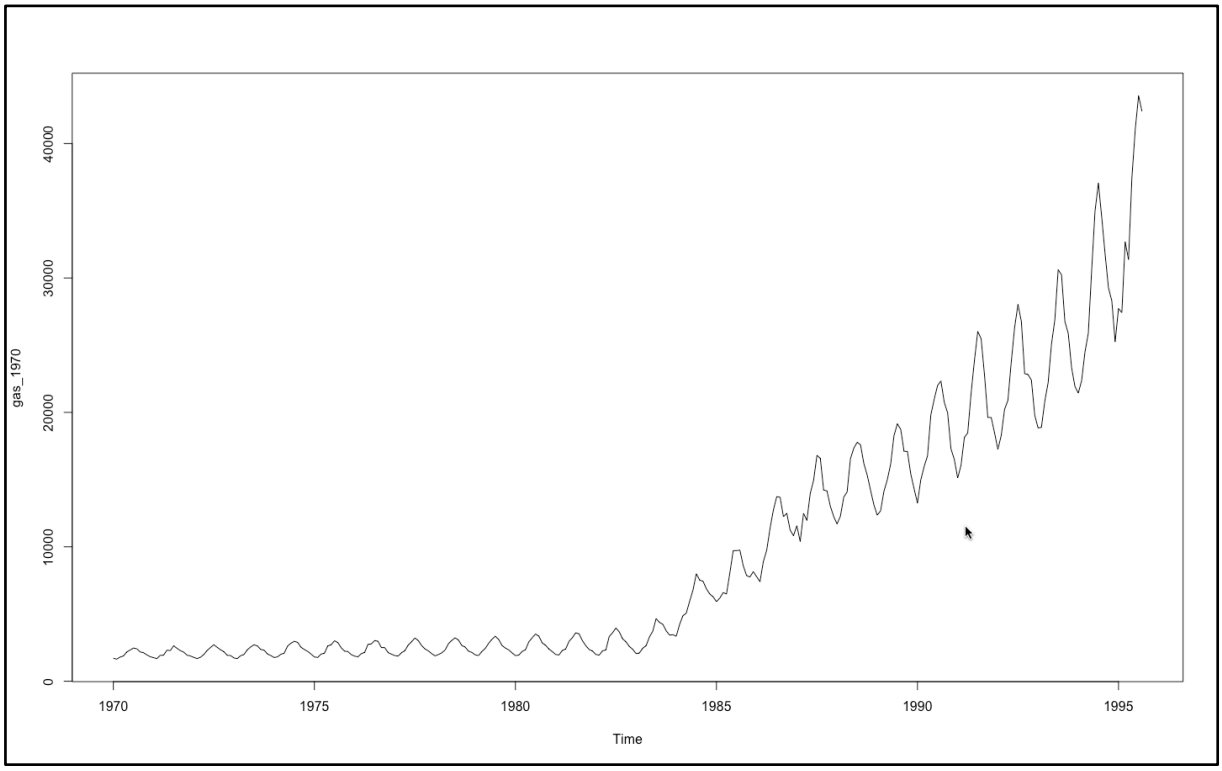
The periodicity of the original Australian gas production is from January 1956 to August 1995.

1.1. Read the data as a time series object in R. Plot the data. (Time Series Data from 1970s)

```
> gas_1970=ts(mydata,start = c(1970,1),end = c(1995,8),frequency = 12)
```

The Australian Gas Production is taken from January 1970 and ended with August 1995 for Time Series Analysis.

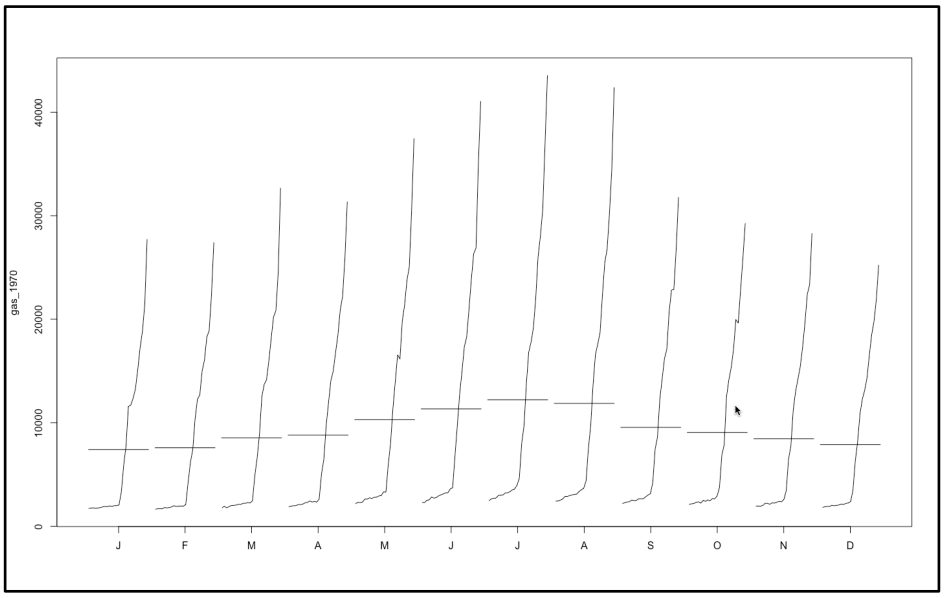
```
> ts.plot(gas_1970)
```



The plot shows the time series data of the Australian gas production from January 1970 to August 1995. The plot shows the constant waves till 1984 and the time series fluctuations starts with the 1985 as it shows the continuous increasing trend in the time series. The time series is separated by the constant growth of the gas production and the gas production from 1985 is increasing as well the spikes increasing in the some of the months and the spikes are decreasing in the months which shows that there is slight seasonality present in the time series.

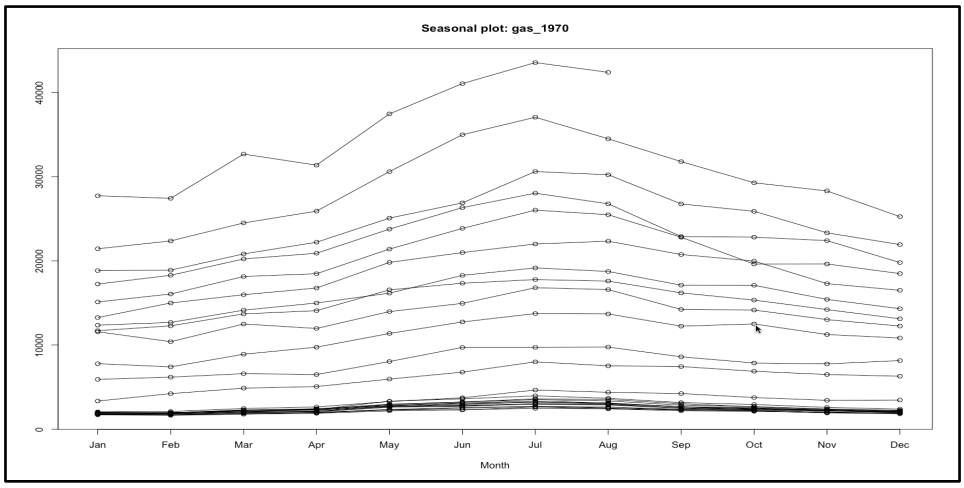
The spikes increased and it finds the highest growth at August 1995. This shows the chart is well defined in sourcing the seasonality and increasing trend after a period of the time.

> monthplot(gas_1970)



The month plot of the time series shows the increased trend in July month and the seasonality is reflected with the slight changes in the seasonality. The variables are predicted for the various trends measured in the values and the series is decreased by December month. The variables are measured in the constant mean for before and after July month of the gas production in the time series. The month plot is measured in the various values for the predicted higher trend from the year 1985 and the seasonality is increased by the constant of two months and then decreased by constant of two months.

> forecast::seasonplot(gas_1970)



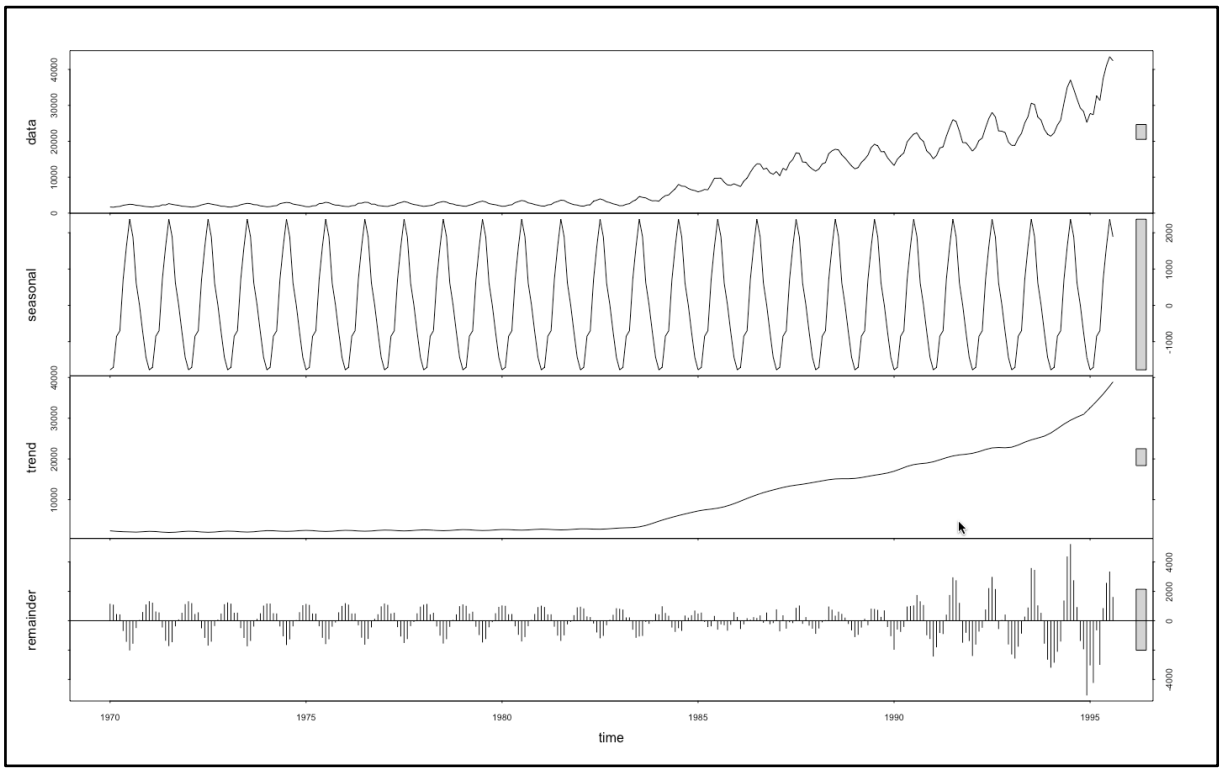
The season plot shows the increased seasonality in the year 1995 and then decreased and it shows the seasonality is present in the dataset.

2.1 What do you observe? Which components of the time series are present in this dataset? (Time Series Data from 1970s)

```
> stl_for_gas_1970=stl(gas_1970,s.window = "periodic")#constant seasonality changes
```

The decomposition of the data is taken for the analysis of the datasets and to predict the components which are present in the time series.

```
> plot(stl_for_gas_1970)
```



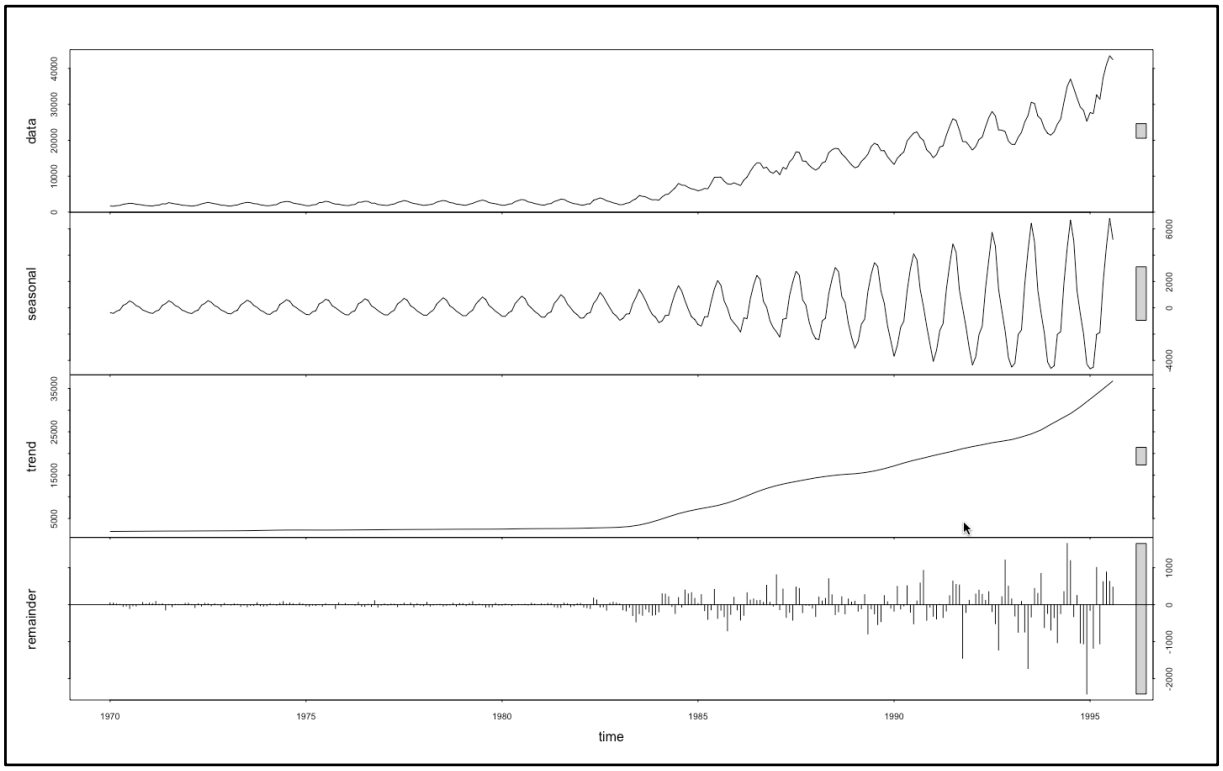
The plot shows the periodic constant seasonality of the gas production. The trend is well exhibited and shows the seasonality is showing constant. The residuals are treated by very less values between 1985 to 1990. This shows the values are certainly working in the less errors for creating the seasonality increases and trend increasing features. The constant seasonality will work on additive models and the slight increase in seasonality is measured from the gas production.

The main components present in the gas production from 1970 to 1995 is exhibited with increasing trend, slight seasonality and minimal residuals. The factors can be treated with the changes in seasonality for further decompositions.

> stl_for_gas_1970_7=stl(gas_1970,s.window = 7)#seasonality changes

The seasonality is decomposed by the constant value 7 and the decompositions shows the various components treated and given accurate values of gas production from the year 1970 to 1995.

> plot(stl_for_gas_1970_7)

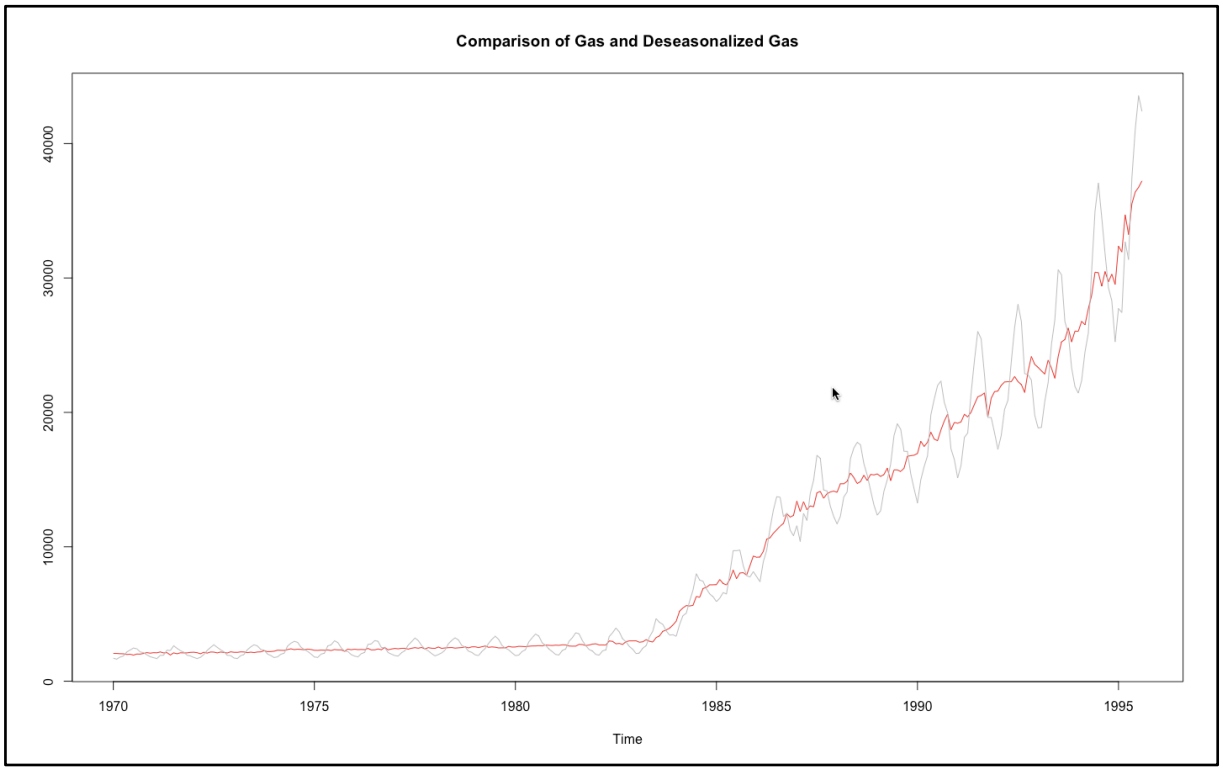


The plot implies that the seasonality is increasing in accordance to trend and measures the residuals increasing with trend. The univariate analysis of decomposition of the gas production shows the values are increased residuals in January 1995. The scaling measures of the seasonality is predicted with minimum changes in seasonality and the multiplicative seasonality is present in the univariate series. The variables are increasing from 1983 and exhibit the good increasing trend of the gas production.

> gas_production=(stl_for_gas_1970_7$time.series[,2]+stl_for_gas_1970_7$time.series[,3])

The object is created for the time series calculation and to find the predicted values are correlated values. Since, seasonality is present in the values the residuals and trends are calculated for the relations between actual and decomposed values.

> ts.plot(gas_production,gas_1970,col=c("red","grey"),main="Comparison of Gas and Deseasonalized Gas")

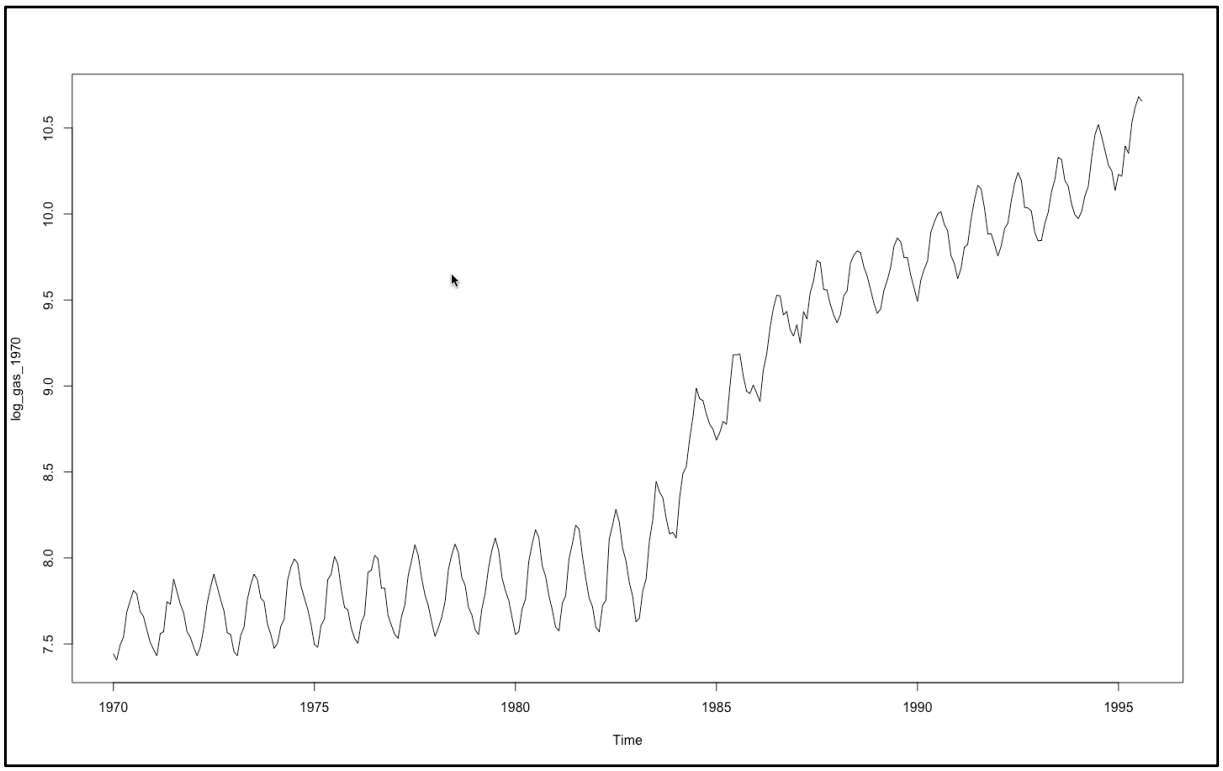


The comparison plot shows the trend exhibits in good increasing factor and the variables are decomposed in the seasonality factor. This plot shows the trend and seasonality present in the variables for the increasing seasonality and increasing trend which shows the higher residuals when calculated for the decompositions. The univariate analysis is based on the decomposition seasonality factor in year 1995 the trend is showing the stop values but the seasonality is increasing for the certain period.

> log_gas_1970=log(gas_1970)

The log values are taken to convert the multiplicative seasonality factor into additive seasonality factor. The log values are treated with exponential factor for constant seasonality and without changes in increasing trend. The decomposition factor are responsible for the increased trend and constant seasonality for the whole period in gas production from 1970 to 1995. The log values are then treated for the predicted charts with actual and predicted values.

> plot(log_gas_1970)

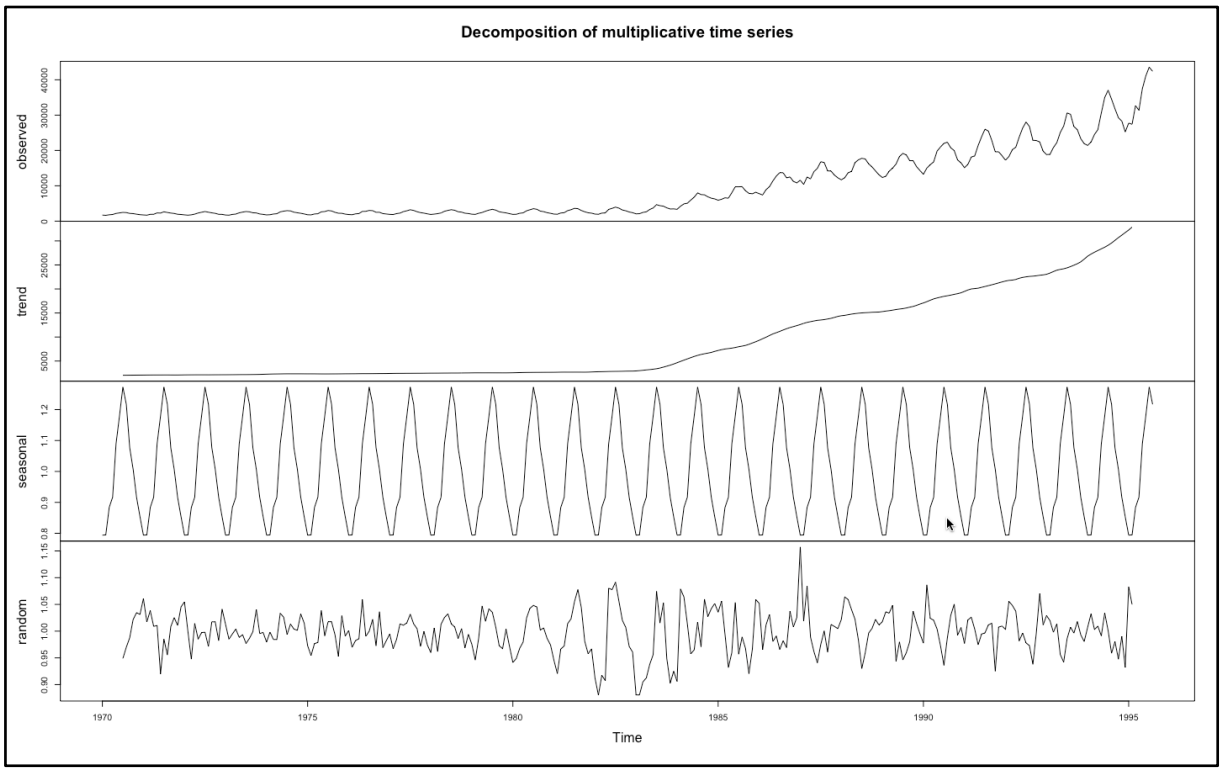


The log values plots shows the increased trend from the year 1985 and it shows the two different seasonality. In 1970, the seasonality is constant over the period 1984 and the values are increased with trend in 1985 and again it is maintained with the constant seasonality. The mean and variance are differed in each series and the variables are treated with most increasing factors and the variables are increased in each seasons. The increased trend in most factors for log values and the decomposition is perfectly exhibited in the univariate series.

> components_of_gas_1970=decompose(gas_1970,type = "multiplicative")#seasonalized gas – components

The components are highly efficient in the decompose functions and shows the factors variables are measured with the increased trend and seasonality. The components showing the variables are exhibited with the increased trend, constant seasonality and minimal residuals with higher variance for the factors treated with trend and seasonality of the values.

> plot(components_of_gas_1970)



The decomposition variables of the plotted series is exhibited well increased trend and constant seasonality and accurate residuals for the various factors. The original values are decomposed with the seasonality factor and the scaling variables are treated for the minimal residuals and the comparison are treated with 0.5 variance in the each univariate analysis. The variance in the seasonality is treated for the 0.5 variance with mean 0. The values in increased trend are measured with the 0 to 30000 it shows the univariate series of the gas production in the variables.

3.1 What is the periodicity of dataset? (Time Series Data from 1970s)

> periodicity(gas_1970)

Monthly periodicity from Jan 1970 to Aug 1995

The periodicity of the gas production taken for the analysis part is January 1970 to August 1995.

Partition your dataset in such a way that you have the data 1994 onwards in the test data.

```
> train=window(gas_1970,start=c(1970,1),end=c(1993,12),frequency=12)
```

The train dataset is created for the analysis of the model creation in ARIMA and auto ARIMA forecasting the observed values in train dataset is 288 and it starts from January 1970 to December 1993.

```
> test=window(gas_1970,start=c(1994,1),frequency=12)
```

The test datasets is created for the validation of the model in ARIMA and auto ARIMA forecasted values. The test dataset is created with the year starting January 1994 and it ends with the August 1995. The observed values in the validation dataset is 20 time series values.

4. Is the time series Stationary? Inspect visually as well as conduct an ADF test? Write down the null and alternate hypothesis for the stationarity test? De-seasonalise the series if seasonality is present?

Augmented Dickey Fuller test shows that there is stationary is present in the time series data.

Null Hypothesis H0 = Time Series Non-Stationary

Alternative Hypothesis Ha = Time Series Stationary

If the p-value is greater than 5%, then we have to reject the Alternative Hypothesis, hence the time will be non-stationary.

```
> tseries::adf.test(train)
```

```
Augmented Dickey-Fuller Test

data:  train
Dickey-Fuller = -0.80309, Lag order = 6, p-value = 0.9606
alternative hypothesis: stationary
```

Test proves that the p-value is greater than 5% and the alternative hypothesis Ha is rejected and H0 is accepted. The time series is Non-Stationary.

> diff_train=diff(train)

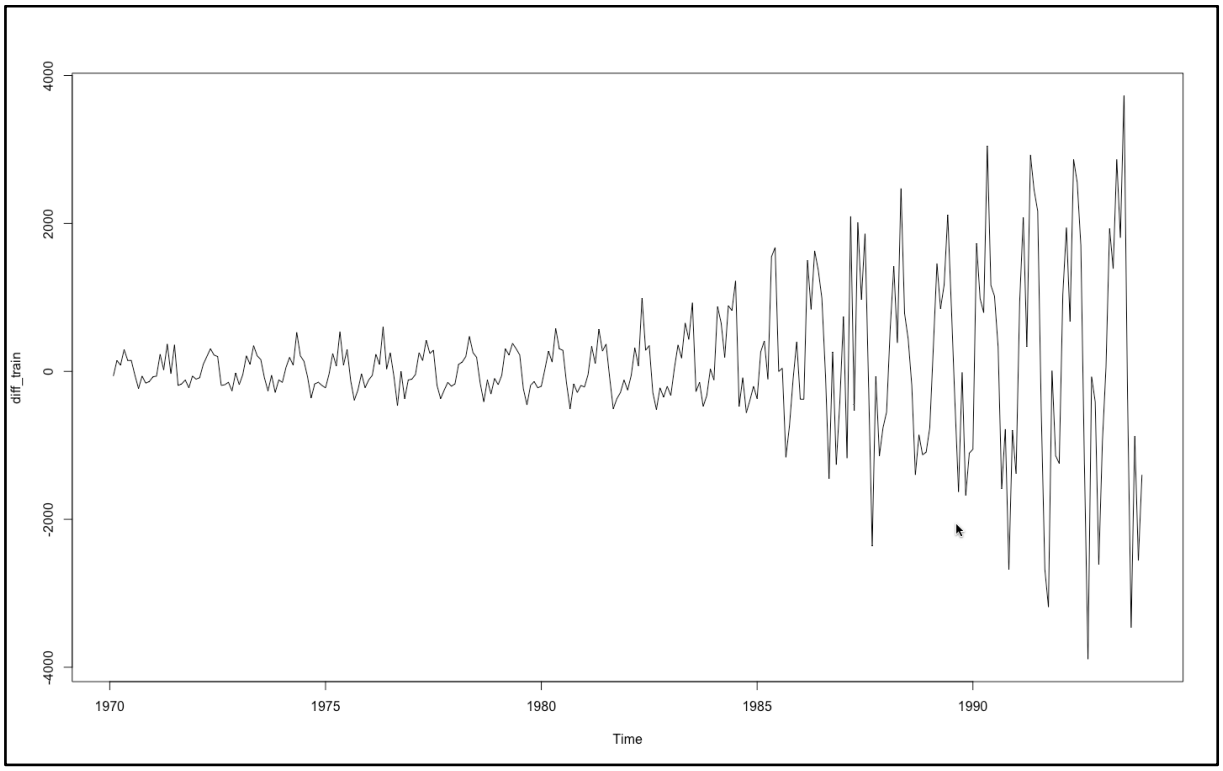
The differentiation function is used to stationaries the time series data and the stationary data will be used in fitting the ARIMA model.

> tseries::adf.test(diff_train)

```
Augmented Dickey-Fuller Test

data:  diff_train
Dickey-Fuller = -15.365, Lag order = 6, p-value = 0.01
alternative hypothesis: stationary
```

Augmented Dickey Fuller test proves that the differentiated data is stationary as the p-value is smaller than 5% and the Null Hypothesis H0 is rejected. Hence, the time series is stationary.

> plot(diff_train)

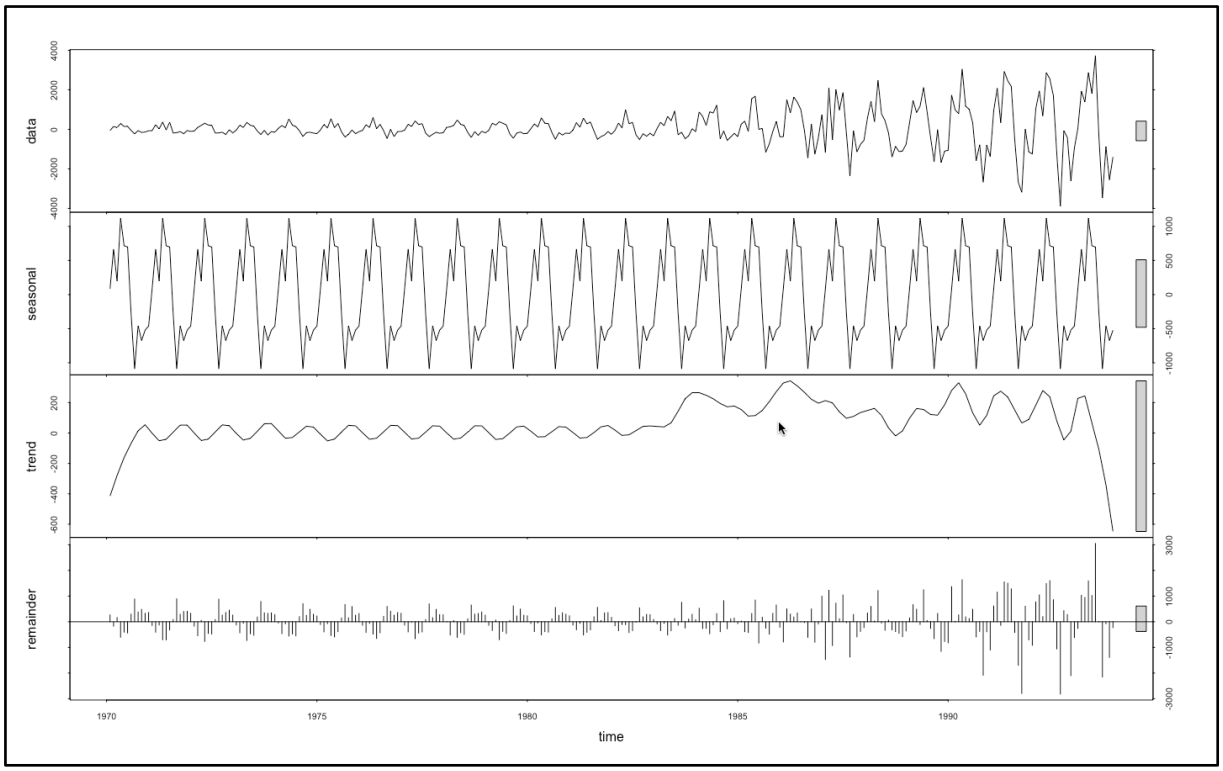


The plot shows the values are stationary. The segments are showing the constant mean over one period and the variance are equal in other segments. This differentiation will help in get the accuracy of the model in validating with the test dataset.

```
> stl_for_train=stl(diff_train,s.window = "periodic")
```

The decomposition of stationary time series is taken for the analysis of components present in the datasets. The decomposition is taken with the periodic constant over the observed train dataset.

```
> plot(stl_for_train)
```



The plot shows the data is having seasonality and the slight seasonality is reflected with the minimal residuals in the train datasets. The trend is predicted with small values and the doesn't shows any increase or decreased value. The trend is projected with -600 to +200 values of the stationary series. The trend is not measured properly and the seasonality is slightly taken place in the stationary series. The residuals are similar to the trend and seasonality growth of the time series data.

The stationary train data is decomposed and the seasonality is predicted in the plot and the slight variance is accordance to the plots of the datasets.

5. Develop an initial forecast for next 20 periods. Check the same using the various metrics, after finalising the model, develop a final forecast for the 12 time periods. Use both manual and auto.arima (Show & explain all the steps)

Auto ARIMA for Initial Forcast

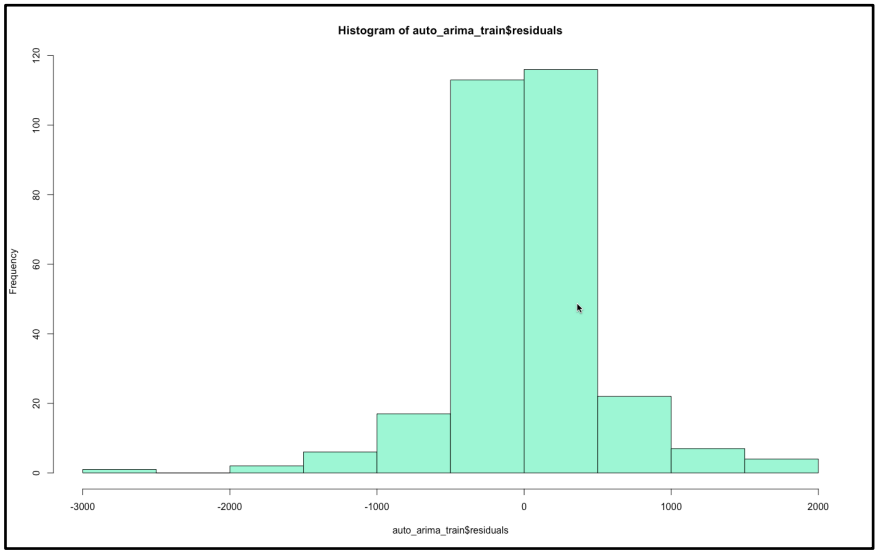
```
> auto_arima_train=forecast::auto.arima(train,seasonal = TRUE,trace = TRUE)
```

The Auto ARIMA is the auto regression techniques which does not needs the original dataset is to be stationary or seasonality should be removed. The Auto ARIMA is helpful in predicting the accurate model for the best fit of the time series and will help the model to be a good fit. The trace is given to predict the future values from the previous data.

```
Fitting models using approximations to speed things up...

ARIMA(2,1,2)(1,1,1)[12]                : 4067.871
ARIMA(0,1,0)(0,1,0)[12]                : 4121.319
ARIMA(1,1,0)(1,1,0)[12]                : 4079.142
ARIMA(0,1,1)(0,1,1)[12]                : 4055.107
ARIMA(0,1,1)(0,1,0)[12]                : 4093.322
ARIMA(0,1,1)(1,1,1)[12]                : 4068.263
ARIMA(0,1,1)(0,1,2)[12]                : 4055.442
ARIMA(0,1,1)(1,1,0)[12]                : 4070.888
ARIMA(0,1,1)(1,1,2)[12]                : Inf
ARIMA(0,1,0)(0,1,1)[12]                : 4065.624
ARIMA(1,1,1)(0,1,1)[12]                : 4048.631
ARIMA(1,1,1)(0,1,0)[12]                : 4088.215
ARIMA(1,1,1)(1,1,1)[12]                : 4062.553
ARIMA(1,1,1)(0,1,2)[12]                : 4050.224
ARIMA(1,1,1)(1,1,0)[12]                : 4066.789
ARIMA(1,1,1)(1,1,2)[12]                : Inf
ARIMA(1,1,0)(0,1,1)[12]                : 4059.176
ARIMA(2,1,1)(0,1,1)[12]                : 4051.502
ARIMA(1,1,2)(0,1,1)[12]                : 4050.444
ARIMA(0,1,2)(0,1,1)[12]                : 4052.755
ARIMA(2,1,0)(0,1,1)[12]                : 4058.224
ARIMA(2,1,2)(0,1,1)[12]                : 4053.783

Now re-fitting the best model(s) without approximations...

ARIMA(1,1,1)(0,1,1)[12]                : 4215.942

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

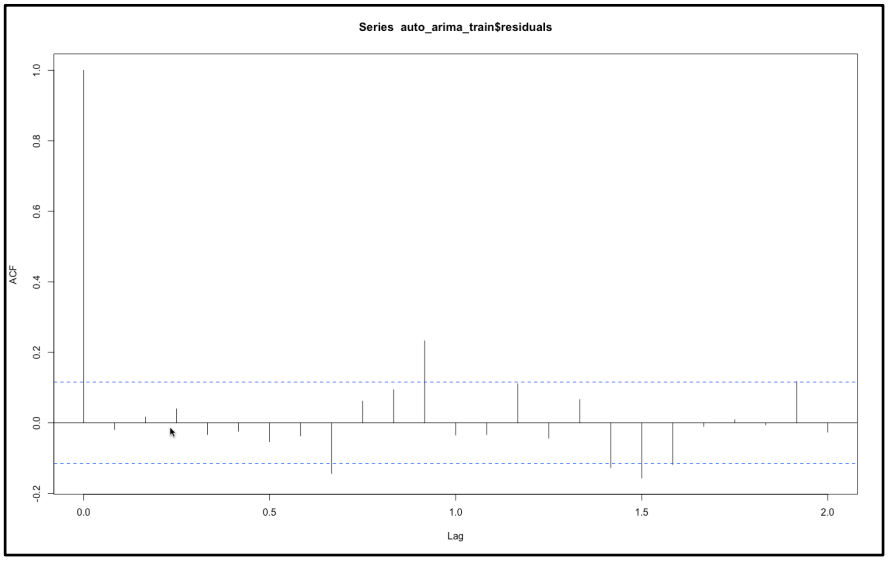
The best model is predicted with Auto ARIMA forecast is developed by the (p, d, q) is (1, 1, 1) with the seasonality values is selected as (p, d, q) is (0, 1, 1) with monthly frequency. The Best Auto ARIMA value is measured with the approximation of 4215.942 and the values are auto regretted with the previous values and the measures are correlated with the original train data.

```
> hist(auto_arima_train$residuals,col = "aquamarine")
```



The histogram shows the residual values are peak in each series. The univariate graph shows the maximum residuals are counted from -1 to +1 and the residuals are highly correlated for the prediction. This residuals values are measured for the minimal residual series in time series.

```
> acf(auto_arima_train$residuals)
```



The Auto correlation of the residuals in the time series shows that the beyond 1, there is no significance in the plots and the values are not much significant in proving the residual are impacting in the forecasting of the models.

```
> Box.test(auto_arima_train$residuals,lag = 30,type = "Ljung-Box")
```

Portmanteau Test is used to check whether the residuals are independent till lag 30.

Null Hypothesis H0 = Residuals are independent

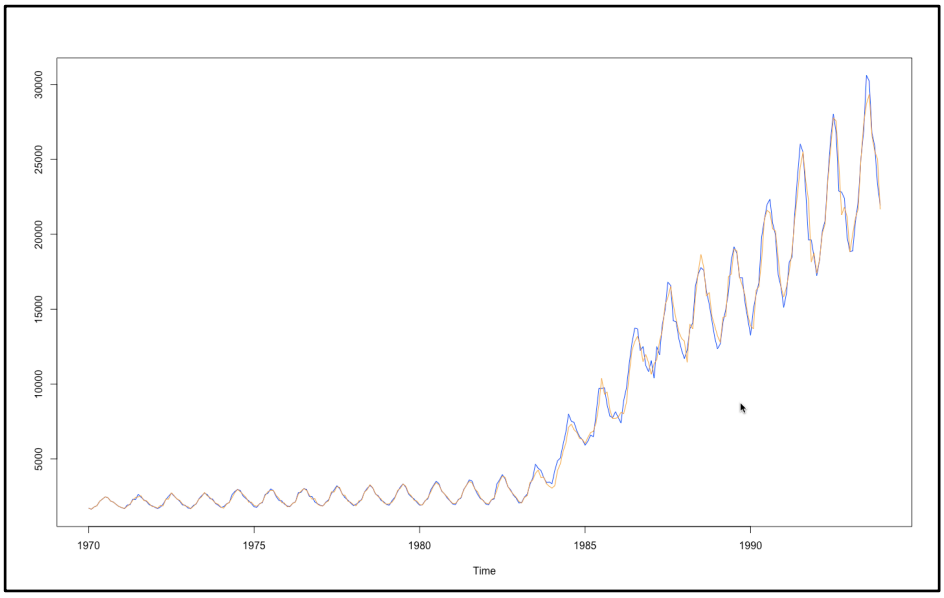
Alternative Hypothesis Ha = Residuals are not-independent

If the p-value is greater than 5%, then the model will be fit with accepting the Null Hypothesis.

```
Box-Ljung test

data:  auto_arima_train$residuals
X-squared = 90.356, df = 30, p-value = 5.799e-08
```

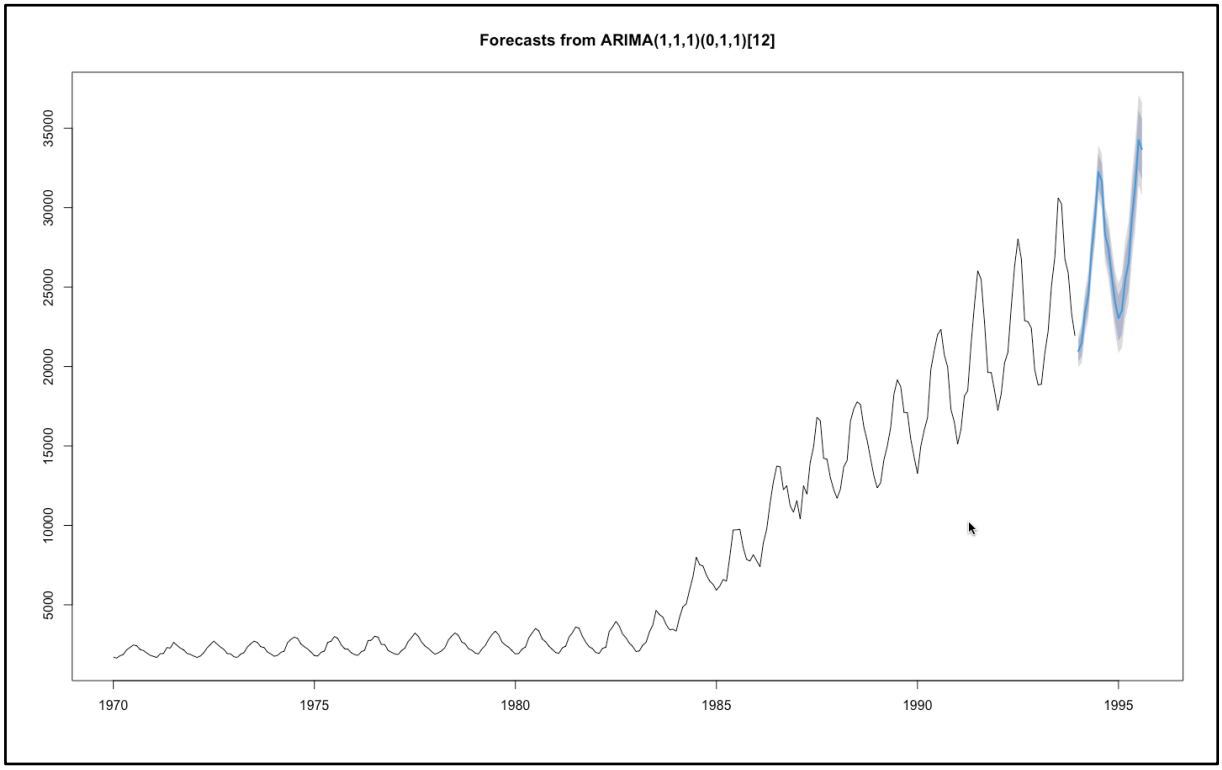
The p-value is predicted by very low value and nearer to 0. Hence the Null Hypothesis is rejected in the case of the lag transformation is till 30. The rejection of Null Hypothesis states that the residuals are not independent till lag 30. Hence, the model is not best accurate fit in terms of residuals of the time series.

```
> ts.plot(train,fitted(auto_arima_train),col=c("blue","orange"))
```



The fitted values are plotted along with the original train dataset for the correlation between actual and predicted values. The plot shows the similar values and pertains the correlation is less significant in the values and in year 1989, the fitted values are higher than the actual values and results in less correlation of the variables.

```
> forecast_auto_arima_train=plot(forecast::forecast(auto_arima_train,h=20))
```



The forecasted values are displayed for the next 20 periods i.e. 20 months of the train dataset. Forecasted period shows the same frequency which developed in the next years. The forecasted values (between 80% and 90%) are more similar in the original forecast of the train datasets.

```
> forecast_auto_arima_train
```

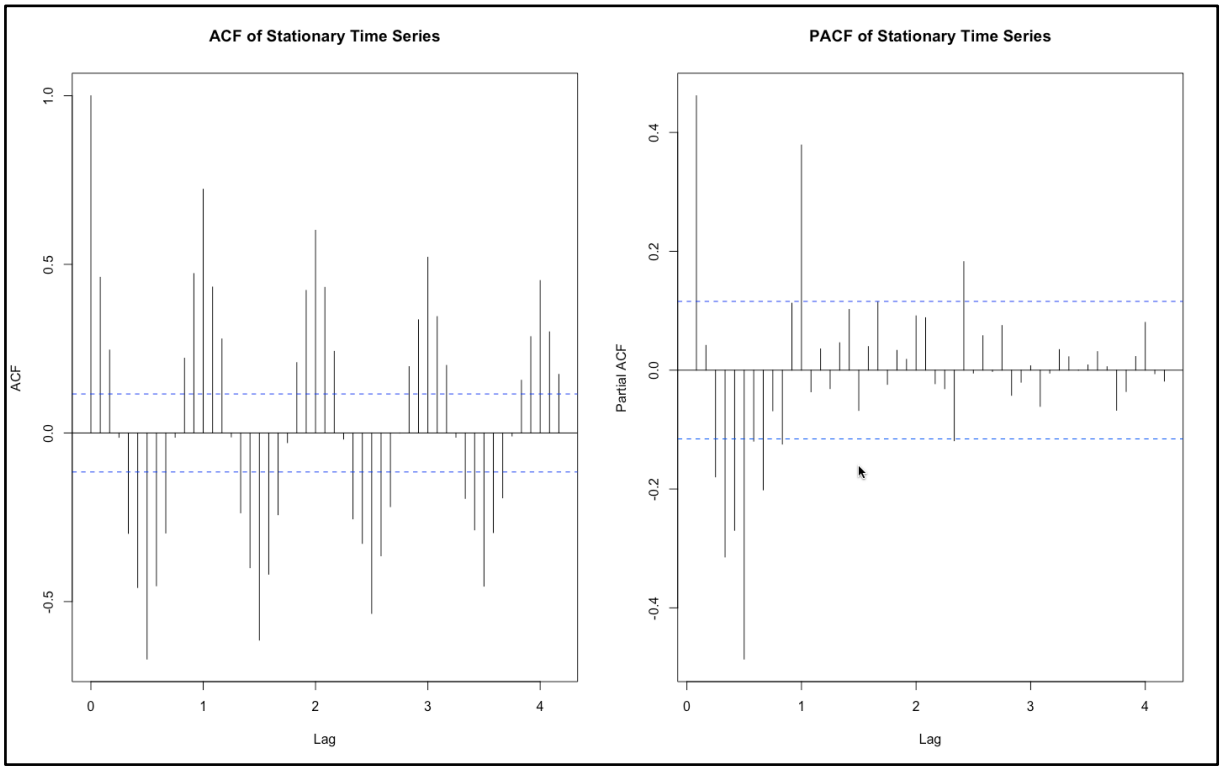
$mean	Jan	Feb	Mar	Apr	May	Jun	Jul	Aug	Sep	Oct	Nov	Dec
1994	20964.82	21472.27	23397.32	24475.53	27325.34	29352.03	32242.76	31663.37	28306.14	27416.17	25664.8	24035.73
1995	23031.38	23517.32	25430.58	26502.3	29348.56	31373.29	34262.96	33682.98				

The forecasted values are predicted by the mean values of the train data. The frequency are produced for the next 20 periods which starts from January 1994 to August 1995. The values are significant in produce the predicted values and makes its significant connection with the original datasets of the gas production.

ARIMA for Initial Forecast

```
> par(mfrow=c(1,2))

> acf(diff_train,lag=50,main="ACF of Stationary Time Series")

> pacf(diff_train,lag=50,main="PACF of Stationary Time Series")
```

The Auto Correlation and Partial Correlation Factors are measured for the analysis of manual ARIMA and the stationary of datasets is previously done with the maximum differentiation value is 1. The AR and MA movement is predicted with the help of this plots to improvise the ARIMA model.



The ACF plot shows the MA movement in the time series and the analysis of the predicted values. The ACF plot predicts with the maximum q value is between lag 2. The lag beyond 3 is unpredictable and doesn't shows the better significance in the auto correlation. The AR model is proven by the PACF plot and the p value is 1. The lags beyond 1 is not significant in the better fitting model.

> arima_train=arima(train,order = c(1,1,2))#with Differentiation, with MA and with AR

The ARIMA model is built without seasonality of train dataset. The ARIMA model shows the better fir than the acquired values of the predicted forecast values. The differentiation is 1.

> arima_train

```
Call: arima(x = train, order = c(1, 1, 2))

Coefficients:
         ar1      ma1     ma2
      0.3847  0.0451  0.1693
s.e.  0.1389  0.1426  0.0661

sigma^2 estimated as 713715:  log likelihood = -2341.51,  aic = 4691.03
```

The AR and MA values are predicted for the ARIMA model without seasonality in the data is present. The AR value is given with lag 1 is about significance value of 0.1389 and MA value is created with lag 1 and lag 2 is about 0.1426 and 0.0661. the values are measured for the prediction of the variables in the featured model.

> arima_train_seasonal=arima(train,order = c(1,1,2),seasonal = list(order=c(0,1,1),period=12))

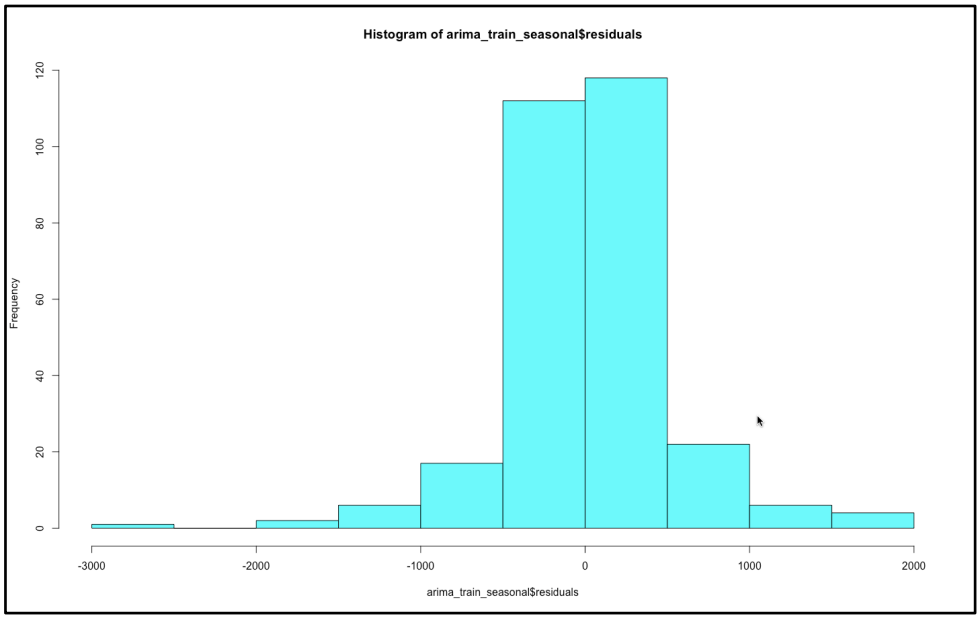
The ARIMA model is built with the seasonality of the datasets and the frequency is 12. The model is predicted without AR and with differentiation of the dataset as the ARIMA can be built only with stationary data.

> arima_train_seasonal

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

Coefficients:
         ar1      ma1     ma2      sma1
      0.6168  -0.8908  0.0521  -0.4116
s.e.  0.1553   0.1670  0.0998   0.0572

sigma^2 estimated as 256005:  log likelihood = -2103.76,  aic = 4217.53
```

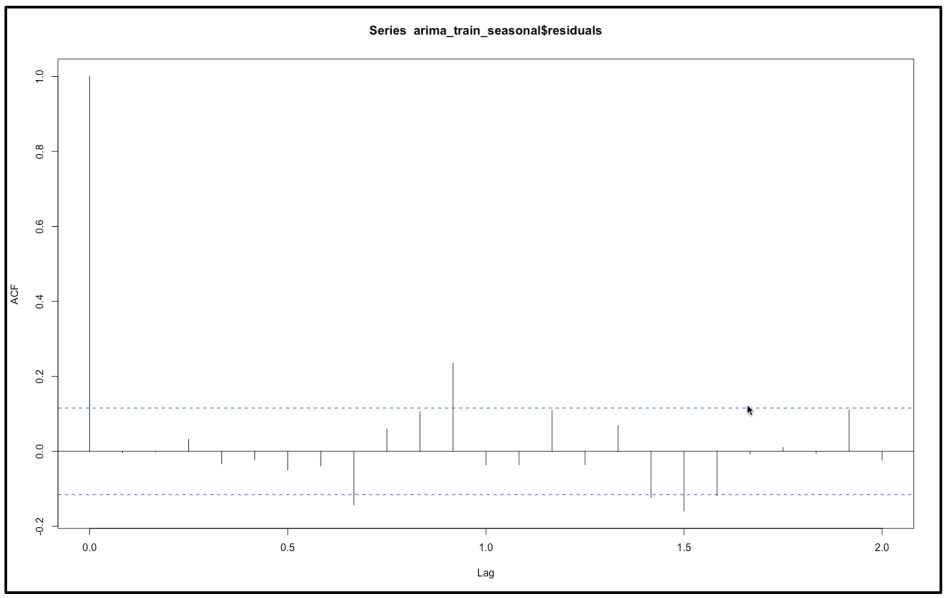
The seasonal factor has influenced in the train data and predicted the AR and MA model is significant about 0.1553 in AR model and MA models are predicted with 0.1670 and 0.0998 respectively of the ARIMA models. The seasonal MA models is predicted with the 0.0572.

```
> hist(arima_train_seasonal$residuals,col = "cyan")
```



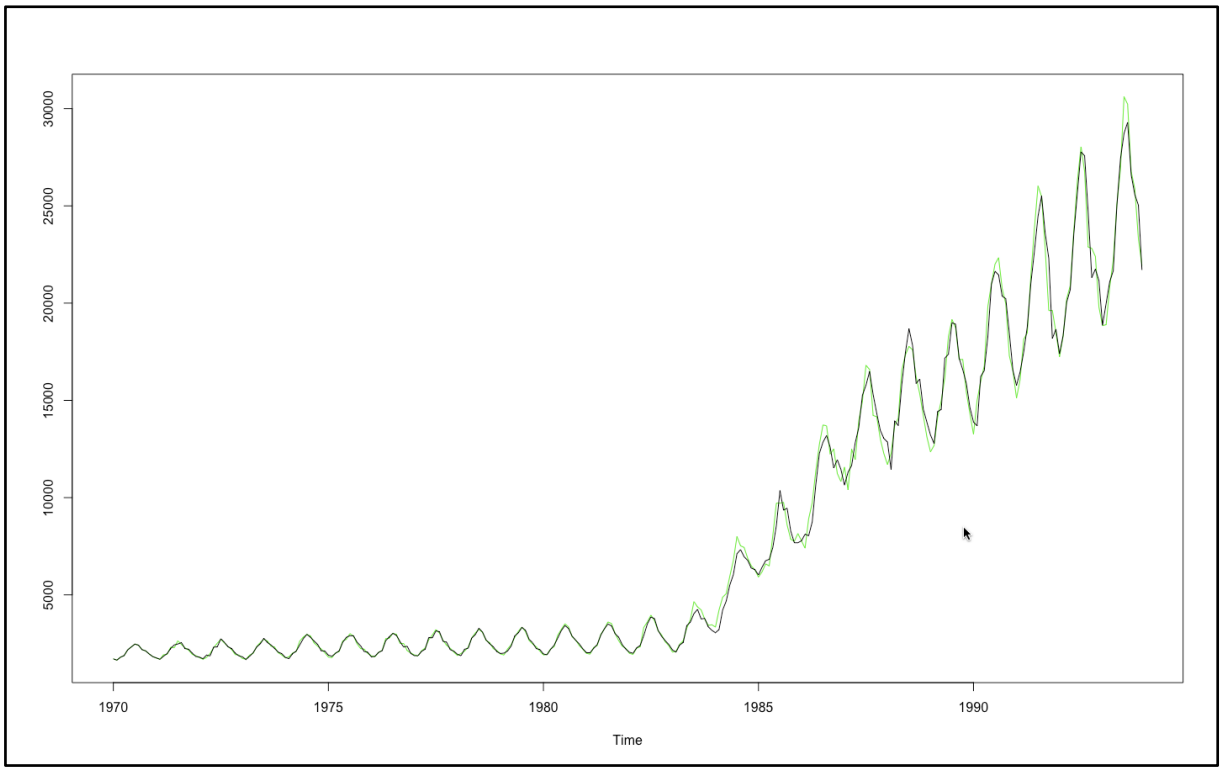
The histogram is measured with univariate analysis of the correlation of residuals are maximum within -500 to +500 in the values. The maximum residuals are measured with the 5% and the residuals are highly treatable with model.

```
> acf(arima_train_seasonal$residuals)
```



The ACF plot of residuals shows that lag 1 is highly significant and the beyond lag 1, there is no significance of the plots.

```
> ts.plot(train,fitted(arima_train_seasonal),col=c("green","black"))
```



The plot shows the accuracy and the significance of the forecasted values with the original series for the prediction of analysis. The fitted values are measured with lesser significance of the original dataset. The model describes to less significant in seasonality of the dataset.

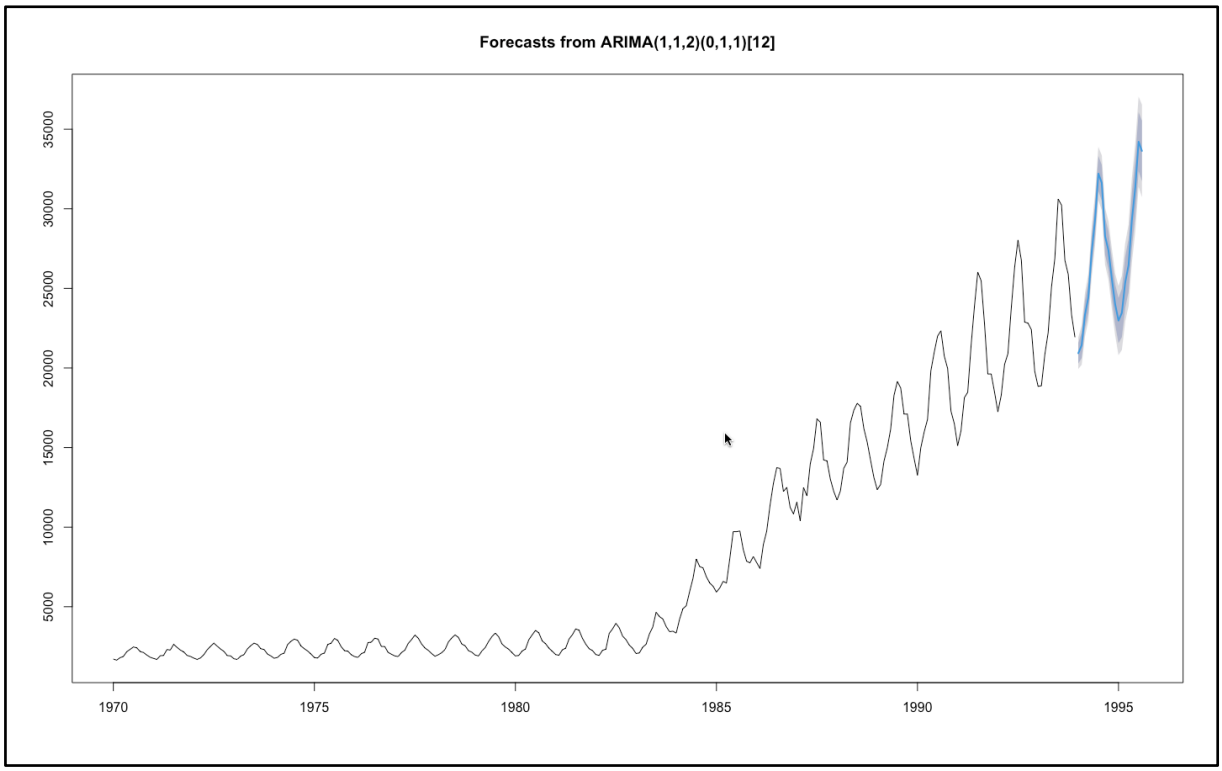
```
> Box.test(arima_train_seasonal$residuals,lag=30,type = "Ljung-Box")
```

```
        Box-Ljung test

data:  arima_train_seasonal$residuals
X-squared = 89.976, df = 30, p-value = 6.623e-08
```

Portmanteau Test is predicted for the lag 30. The p-value is measured with smaller than 5%, then it is failed to prove the Null Hypothesis and it shows the residuals are not independent (Ha). Hence the model is required to develop the most accuracy in the forecasted values.

> forecast_arima_train=plot(forecast::forecast(arima_train_seasonal,h=20))

The forecast of the model is developed with train data with the seasonal factors. The frequency measured for the next 20 periods in train model.



The plot shows the forecasted values are between 85% and 90% accuracy for the next 20 periods the forecasted values are exactly in the ratio of the model developed in the actual data. The forecasted values are measured for the higher rate of increased seasonality on the predicted values.

> forecast_arima_train

$mean	Jan	Feb	Mar	Apr	May	Jun	Jul	Aug	Sep	Oct	Nov	Dec
1994	20926.69	21427.55	23353.23	24434.69	27287.23	29315.81	32211.26	31633.04	28275.05	27386.17	25633.09	24004.06
1995	22977.61	23459.36	25373.26	26447.44	29295.5	31321.31	34215.06	33635.79				

The mean values are average predicted values for next 20 periods of the test dataset. The values are measured with similar to ARIMA model with the actual predicted values in the datasets.

5.1 Develop a final forecast for the 12 time periods. Use both manual and auto.arima

Auto ARIMA for final forecast

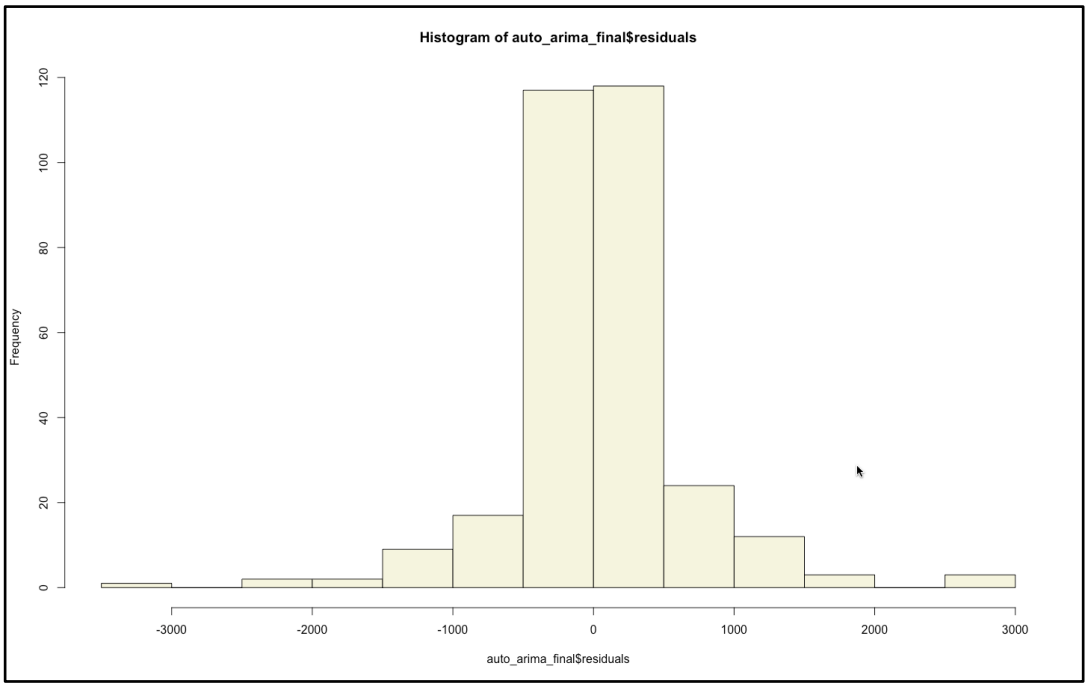
```
> auto_arima_final=forecast::auto.arima(gas_1970,seasonal = TRUE,trace = TRUE)
```

The Auto ARIMA is used for the forecasted of the model with the seasonality factor and makes trace of the original previous data in the models.

```
Fitting models using approximations to speed things up...

ARIMA(2,1,2)(1,1,1)[12]                : 4491.266
ARIMA(0,1,0)(0,1,0)[12]                : 4550.831
ARIMA(1,1,0)(1,1,0)[12]                : 4493.929
ARIMA(0,1,1)(0,1,1)[12]                : 4480.656
ARIMA(0,1,1)(0,1,0)[12]                : 4510.515
ARIMA(0,1,1)(1,1,1)[12]                : 4493.144
ARIMA(0,1,1)(0,1,2)[12]                : 4478.691
ARIMA(0,1,1)(1,1,2)[12]                : Inf
ARIMA(0,1,0)(0,1,2)[12]                : 4505.447
ARIMA(1,1,1)(0,1,2)[12]                : 4479.599
ARIMA(0,1,2)(0,1,2)[12]                : 4480.474
ARIMA(1,1,0)(0,1,2)[12]                : 4478.123
ARIMA(1,1,0)(0,1,1)[12]                : 4479.47
ARIMA(1,1,0)(1,1,2)[12]                : Inf
ARIMA(1,1,0)(1,1,1)[12]                : 4492.097
ARIMA(2,1,0)(0,1,2)[12]                : 4481.094
ARIMA(2,1,1)(0,1,2)[12]                : 4480.472

Now re-fitting the best model(s) without approximations...

ARIMA(1,1,0)(0,1,2)[12]                : 4650.831

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

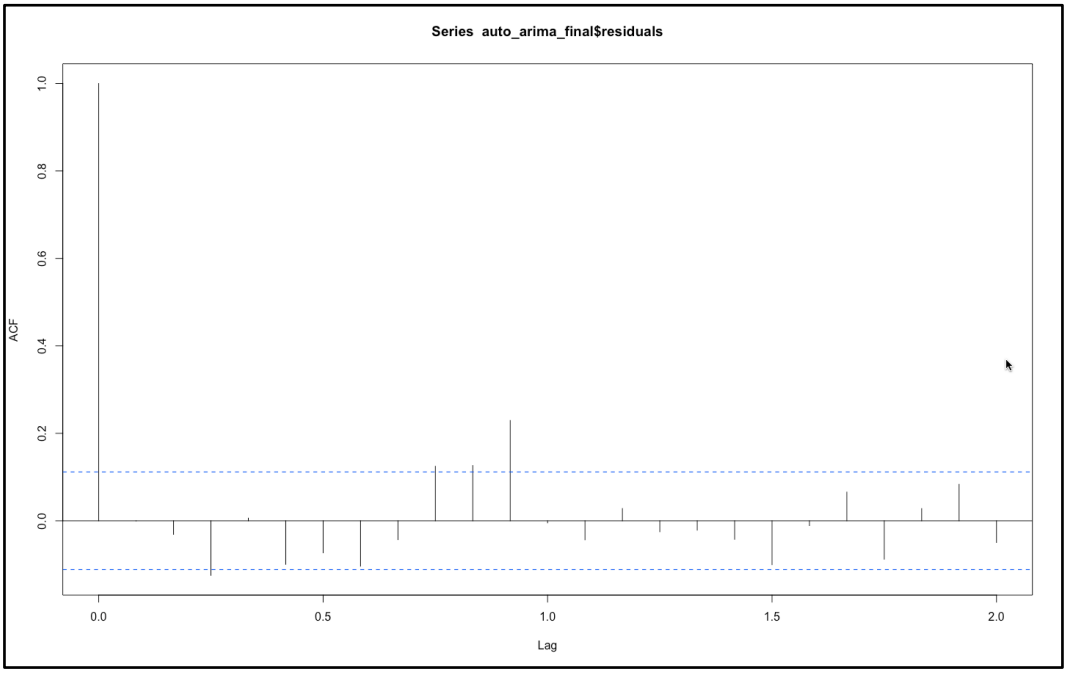
The forecasted value of the best model is predicted with AR and MA values in the frequency of (p, d, q) is (1, 1, 0) and the seasonal frequency are predicted with (p, d, q) is about (0, 1, 2) in the frequency 12. The best model is based on the values are predicted as 4650.831 in the measured built model.

> hist(auto_arima_final$residuals,col = "beige")



The residuals are treated for the range between -500 to +500 values. The histogram value is treated with the values are measured with minimum residual of the original dataset.

> acf(auto_arima_final$residuals)



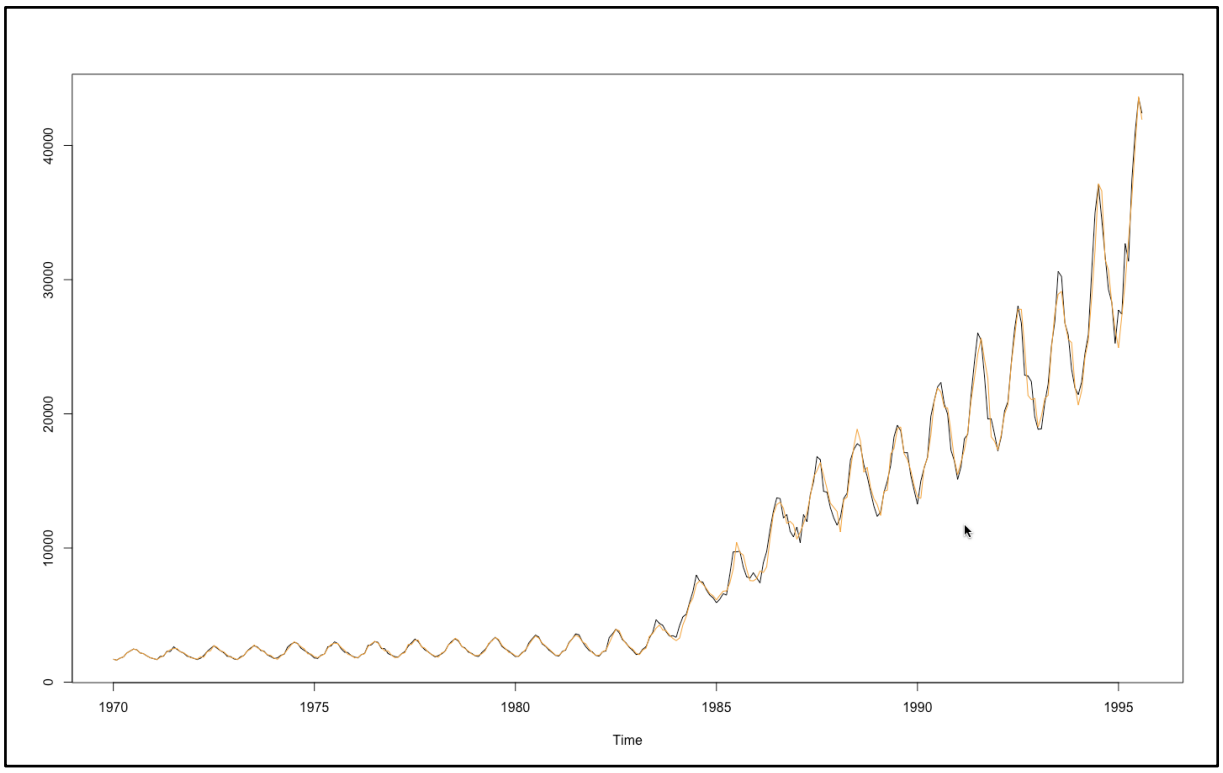
The ACF plot is calculated for the lag 1and beyond lag 1 there is no measured significant values.

```
> Box.test(auto_arima_final$residuals,lag=30,type = "Ljung-Box")
```

```
Box-Ljung test

data:  auto_arima_final$residuals
X-squared = 73.841, df = 30, p-value = 1.459e-05
```

Portmanteau test is predicted with p-value is smaller than 5%. The smaller p-value is leads to reject the Null Hypothesis and it takes the residuals are not independent in the measured values of the model.

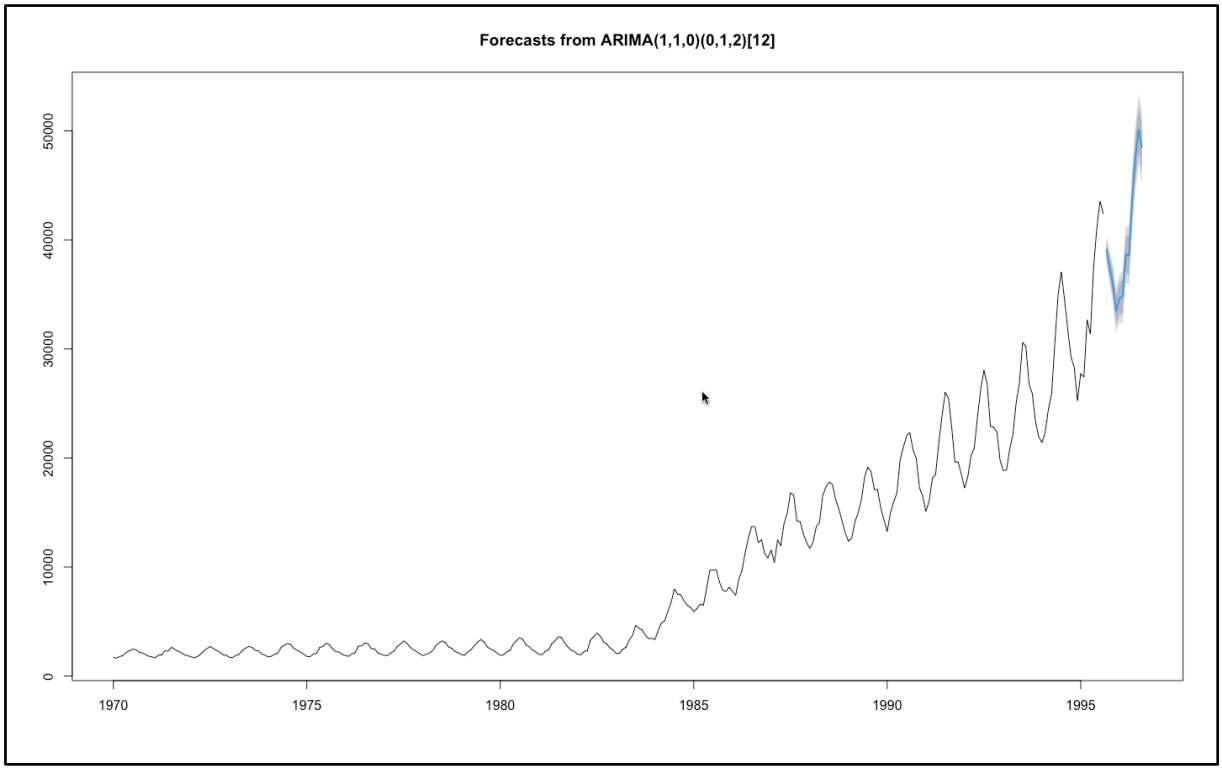
```
> ts.plot(gas_1970,fitted(auto_arima_final),col=c("black","orange"))
```



The plotted values are significant in the actual plotted values for the forecasted model of Auto ARIMA model. The seasonality is selected for the two segments with increasing trend patterns and the significance of the fitted values are predicted with higher measure of the models.

> auto_arima_final_forecast=plot(forecast::forecast(auto_arima_final,h=12))

The Auto ARIMA forecast is built model with the next 12 periods for the forecasted plot.



The forecasted plot at the year is showing the decreased growth next month followed by August 1995. The gap is filled the forecasted values for the increase and decrease trend growth of 85% and 90% in the seasonality trend patterns in the variables defined.

> auto_arima_final_forecast

$mean	Jan	Feb	Mar	Apr	May	Jun	Jul	Aug	Sep	Oct	Nov	Dec
1995									39109.45	37444.17	35953.3	33509.72
1996	34649.27	34803.11	38674.39	38568.96	43908.08	47649.68	50122.94	48512.79				

The final forecast of Auto ARIMA model is predicted for the next 12 months and the values are predicted till August 1996. The values are showing the same pattern in increase and decreased factor of the model. The increased pattern in July is consistent till the pattern is designed in the values for the higher prediction of the assumed actual values.

ARIMA for final forecast

```
> tseries::adf.test(gas_1970)#stationary check
```

The final forecast is taken with original dataset form January 1970 to August 1995. The final forecast for the 12 periods are measured for the original dataset.

```
Augmented Dickey-Fuller Test

data:  gas_1970
Dickey-Fuller = 0.73972, Lag order = 6, p-value = 0.99
alternative hypothesis: stationary
```

Augmented Dickey Fuller Test is predicted with p-value is 0.99 which is greater than 5% and the time series is not stationary. Hence the time series is not stationary.

```
> diff_gas=diff(gas_1970)#differentiation as the data is not stationary
```

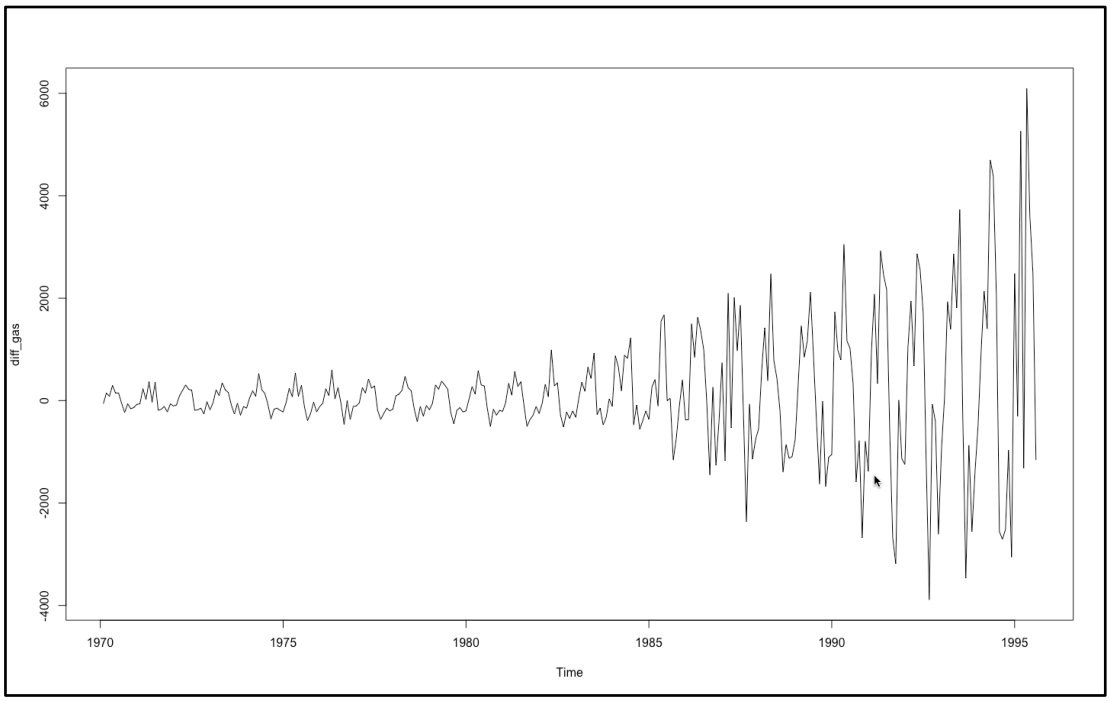
The differentiation is taken as the time series is not stationary and it need to be differentiated from the dataset.

```
> tseries::adf.test(diff_gas)#checking the stationarity for the differentiation
```

```
Augmented Dickey-Fuller Test

data:  diff_gas
Dickey-Fuller = -15.575, Lag order = 6, p-value = 0.01
alternative hypothesis: stationary
```

The ADF shows the p-value is measured with smaller value and the time series is stationary in the time series. This differentiation is taken for ARIMA forecasted analysis.

> plot(diff_gas)

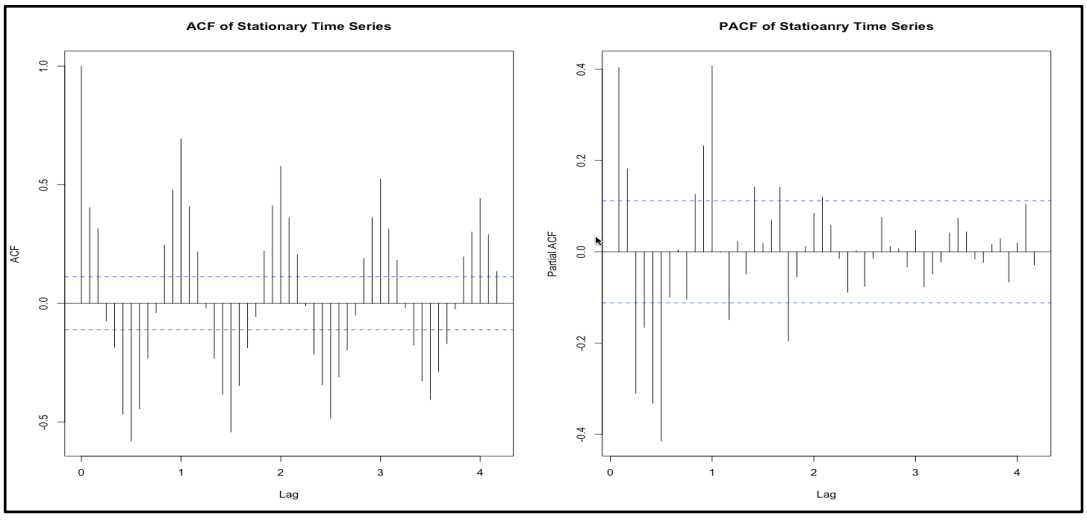


The plot is showing the differentiated values for the year 1970 to 1995. the differentiated values are stationary with two segments. The segments are measured form mean 0 and differed variance for the datasets.

> par(mfrow=c(1,2))

> acf(diff_gas,lag=50,main="ACF of Stationary Time Series")

> pacf(diff_gas,lag=50,main="PACF of Statioanry Time Series")



The ACF and PACF are performed for the stationary time series and the values are predicted for AR and MA movement is about for p is significant more than 5 and the q is significant of about 2. The differentiation value is measured as 1.

```
> arima_final=arima(gas_1970,order = c(5,1,2),seasonal = list(order=c(1,1,1),period=12))
```

The ARIMA final forecasting model is defined with the seasonality of the time series and the values are predicted for the frequency is 12. The seasonal (p, d, q) is (1, 1, 1) since the maximum lag is taken for the seasonality in auto correlation and partial auto correlation of the original data series.

```
> arima_final
```

Call: arima(x = gas_1970, order = c(5, 1, 2), seasonal = list(order = c(1, 1, 1), period = 12))

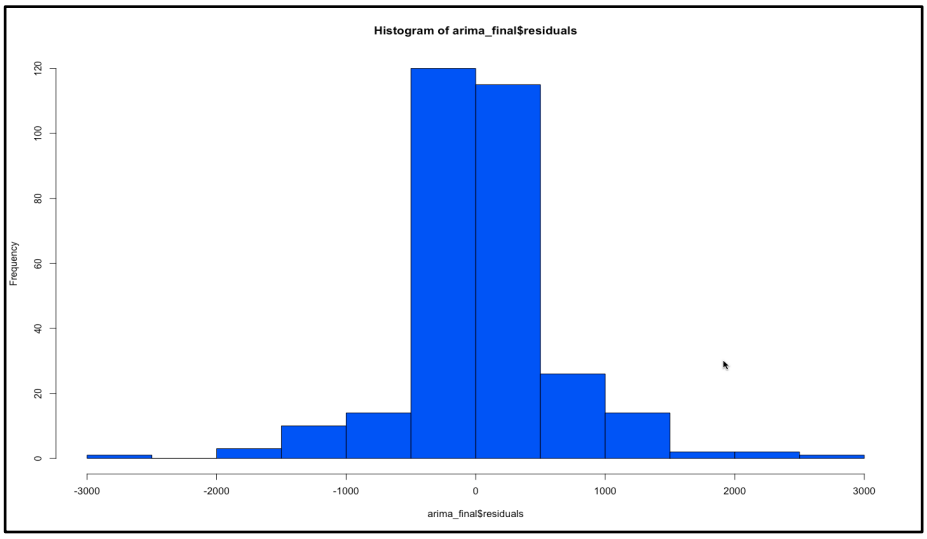
Coefficients:

	ar1	ar2	ar3	ar4	ar5	ma1	ma2	sar1	sma1
	0.9907	-0.5036	-0.367	0.1799	-0.2213	-1.334	0.9398	-0.1262	-0.29
s.e.	0.0655	0.0854	0.0837	0.0833	0.0598	0.0304	0.0412	0.1339	0.133

sigma^2 estimated as 364512: log likelihood = -2309.14, aic = 4638.29

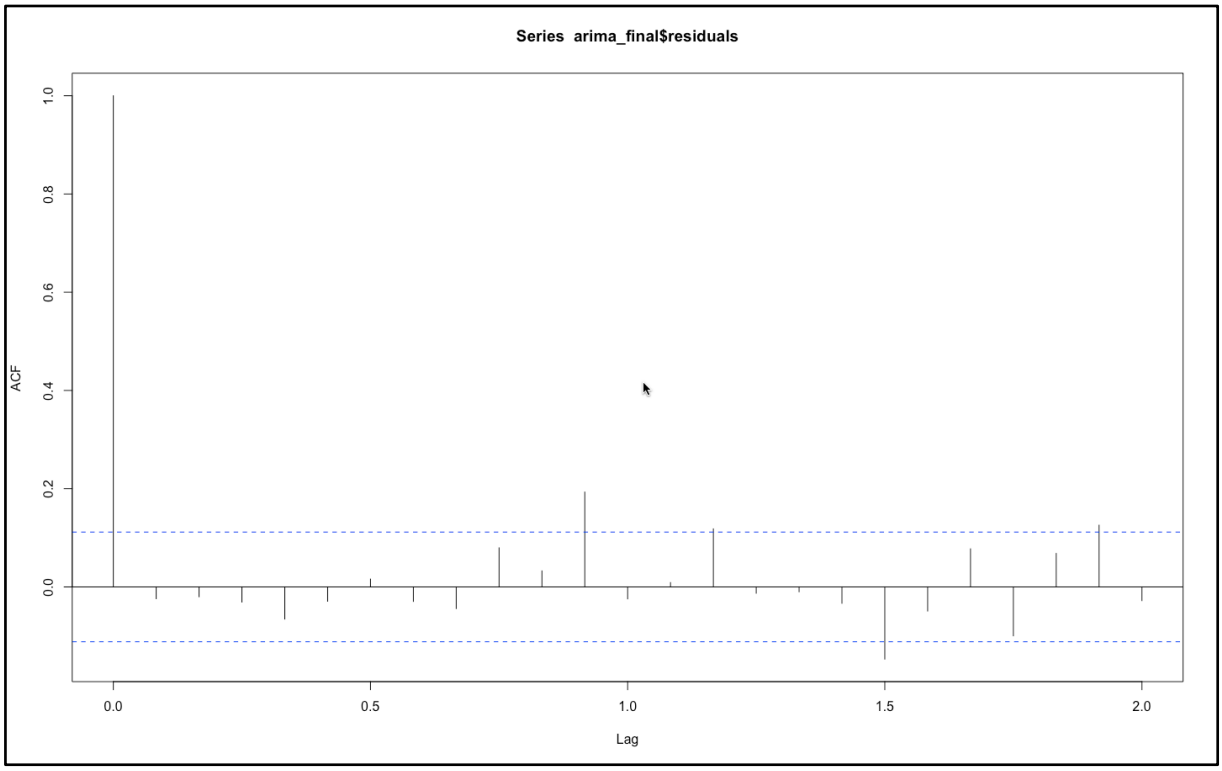
The ARIMA model is calculated for the AR and MA movement of the original dataset. The predicted values that are significant of seasonal AR and MA values in the each variables for the univariate entries. The ARIMA seasonal values are increased for the period of forecasted value is 12.

```
> hist(arima_final$residuals,col = "blue")
```



The histogram plot shows the values are plotted from the correlation to negative to positive values. The increased maximum residual are settled in centre of the plots and the other minimal residuals are created for the correlation with the original datasets.

```
> acf(arima_final$residuals)
```



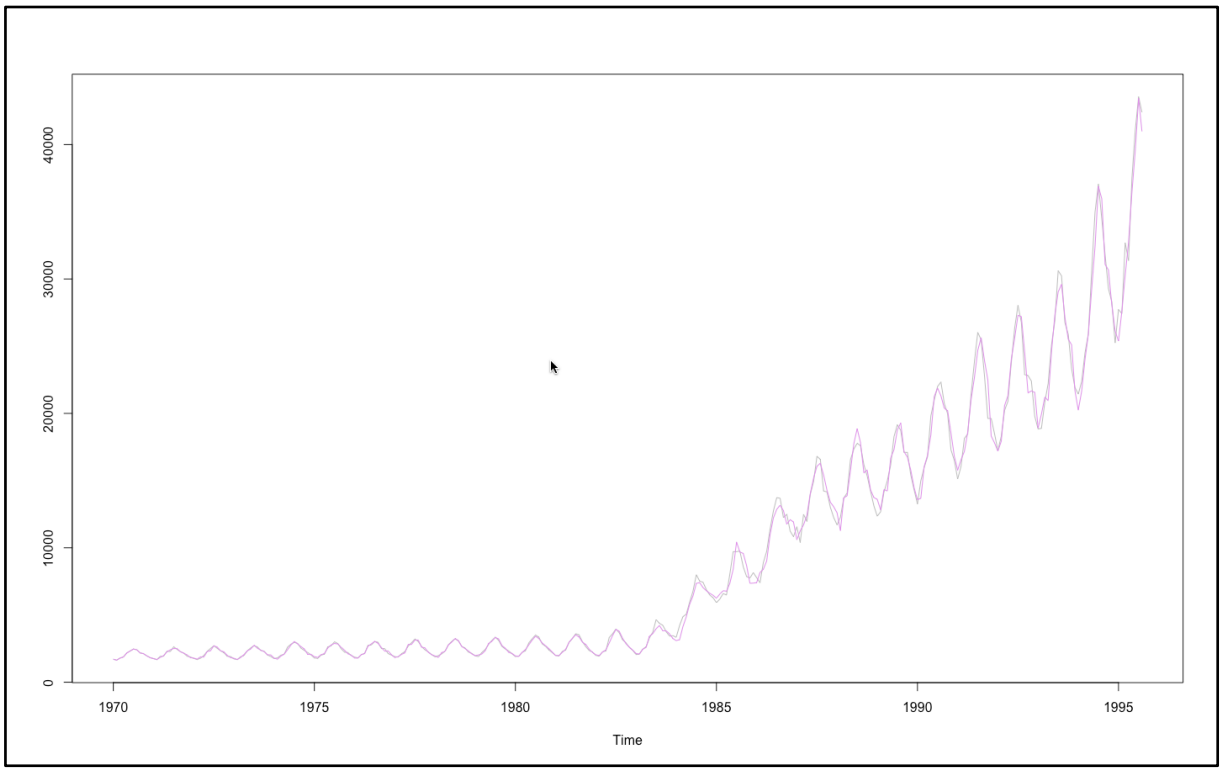
The residuals lag points are significant on lag 1 and there is no much significance in the values present in the values for the maximum significance. The ACF plots shows that beyond lag 1, the values are not correlated in the maximum side of the plots.

```
> Box.test(arima_final$residuals,lag = 30,type = "Ljung-Box")
```

```
Box-Ljung test

data:  arima_final$residuals
X-squared = 58.747, df = 30, p-value = 0.001302
```

Portmanteau test shows the original fit of the model residuals are not independent. Thus the p-value is not greater than 5% till the lag 30 and the residuals values are not independent for the ARIMA model.

```
> ts.plot(gas_1970,fitted(arima_final),col=c("grey","violet"))
```

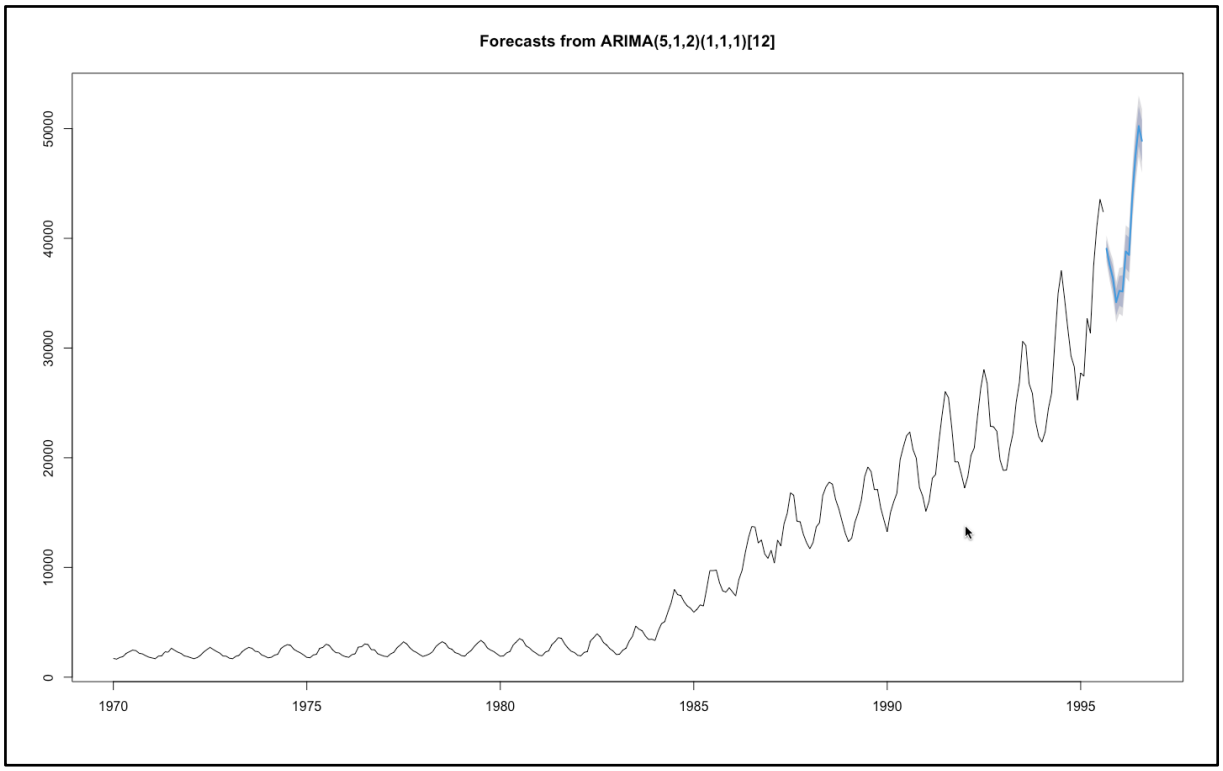


The time series plot is significant in the values generated in the model fit for the values and the values are created for next forecasted method on the time series data. The forecasted value is treated with the actual and predicted values for the increased seasonality and increased trend in the univariate analysis of the forecasted model.

```
> arima_final_forecast=plot(forecast::forecast(arima_final,h=12))
```

The final forecast of the ARIMA model is built with the forecasted value is 12. The next final forecast on 12 months is predicted form the previous values of the original dataset in the values present in the gas production.

The forecasted values are treated with increased and decreased trend and seasonality of the values is about 85% and 90% of the growth of gas production. The values are based on the lag produced in the seasonality presented ARIMA function and it will involves the various functions of the components. The values are based on the 12 months which gives the best prediction in the previous pattern of the seasonality.



The forecasted values are still in gap of producing the values for the continuous entries in the values for the increased trend and the values on the higher prediction of the previous values. The lag produced in the previous values are increased in the constant variance of the prediction in each segment.

> arima_final_forecast

$mean	Jan	Feb	Mar	Apr	May	Jun	Jul	Aug	Sep	Oct	Nov	Dec
1995									39060.96	37546.4	36366.1	34176.44
1996	35221.28	35149.21	38805.7	38478.89	43788.58	47593.05	50241.64	48882.93				

The final forecasted values are predicted for the next 12 months. The univariate series is forecasted with the September 1995 to August 1996. The forecasted values are measured in the highest mean value of July and the values are measured with decreasing and increasing factors for the variables and exhibits the seasonality and increased trend in the values for the original dataset in the forecasted trend. The forecasted values are treated for the highest 80% and 95% of the forecasted values.

6. Report the accuracy of the model

> forecast::accuracy(auto_arima_train)

	ME	RMSE	MAE	MPE	MAPE	MASE	ACF1
Training set	27.33581	494.6562	290.8485	0.3410585	3.494452	0.2957792	-0.01884895

The accuracy of the model created with MAPE (Mean Absolute Percent Error) shows the value is 3.49% which is good fit and exhibits the better accuracy in the each factors. The accuracy of the fit of Auto ARIMA model is measured with ACF Factor of -0.01.

> forecast::accuracy(arima_train_seasonal)

	ME	RMSE	MAE	MPE	MAPE	MASE	ACF1
Training set	27.39739	494.4178	289.7535	0.3477467	3.472744	0.4718336	-0.00418693

Accuracy of the ARIMA factor is measured with MAPE value of 3.47% and this model also exhibits the good fit. The auto correlation value is treated and measured with -0.00 which is very low value factor in the ARIMA fitted values.

> forecast::accuracy(auto_arima_final)

	ME	RMSE	MAE	MPE	MAPE	MASE	ACF1
Training set	41.98972	616.6393	343.2864	0.2485466	3.398262	0.2686541	-0.00030606

The MAPE value of the final forecast on Auto ARIMA model is measured with 3.39% and the value is good fit in the Auto correlation value of -0.00 and exhibits the minimum errors in the model.

> forecast::accuracy(arima_final)

	ME	RMSE	MAE	MPE	MAPE	MASE	ACF1
Training set	50.10252	590.8697	347.5626	0.2886987	3.483081	0.4702567	-0.02396623

The MAPE for the ARIMA final model is measured for 3.48% and the Auto correlation plot is about 2% in the forecasted model. This ensure the slight fit in the ARIMA model and the forecast value is good fit.

- The accuracy of the model is exhibiting the good forecast value in each models. The best fit of forecasted value is determined by MAPE (Mean Absolute Percent Error) and the outcome of the best fit is well determined in the Auto ARIMA model of final forecast.
- The forecasted values are examined in the fitted values of the train and original dataset of the gas production. The values with lower MAPE is making the model to be more fitted in the errors as well as the minimal errors in the model. The model is structured with the time forecasting for 12 months.
- Accuracy of the model measures are taken in the consideration of the models created in the variance of trend, seasonality and residuals of the decomposed factors.
- The stationary values for the accuracy is measured in the AR (Auto Regression) and MA (Moving Average) are maintained in the prediction of the previous values for increased trend and seasonality of the factors present in the datasets.
- Accuracy measures on the model are making the time series most fitted models in the previous values and measured in the period of 12 months. The values are maintained in the sources of the values are measured in the decomposition of the time series in the constant seasonality present in the data.
- Hence, the best model which showing the best accuracy model is Auto ARIMA model predicted for final 12 months in the Auto ARIMA forecasted value.

3.5 Outlier Identification

The outlier identification for the time series is found as the maximum values which is usual data in the univariate as well as bivariate analysis of the data.

Outliers are measured in the time series analysis and cannot be eliminated or treated. The maximum knowledge can be imputed form the decomposition of time series model.

3.6 Variable Transformation/ Feature creation

The variable transformation of the data is taken for the analysis of the time series data from 1970 for analysis.

```
gas_1970=ts(mydata,start = c(1970,1),end = c(1995,8),frequency = 12)
```

The time series data is selected from January 1970 and ended with August 1995 for the forecast analysis in ARIMA model.

4. Conclusion

Australian Gas Production from January 1956 to August 1995 is predicted for Time Series Forecasting. In which the data period from January 1970 to August 1995 is taken for analysis of the ARIMA model and Auto ARIMA model. The dataset is predicted with components like increasing trend on certain period and slight seasonality in certain period. In seasonality period July month is predicting with higher gas production. The increased July month production is showing the seasonality is present. Auto ARIMA models are predicting the forecasting values for the 12 months which predicted the Australian Gas Production till August 1996. In this period the seasonality exhibits with increased value in July month. The forecasted value is better fit in Auto ARIMA model for final forecast of 12 periods. Hence, the variance is correlated with Auto ARIMA forecasted values.

5. Appendix

```
#=========================================Set Working Directory==========================================
setwd("/Users/numerp/Documents/PGP-BABI/Module 7 Time Series Forecasting/Project 6")
getwd()
#=========================================Install Required Packages=======================================
library(forecast)
library(tseries)
library(fpp2)
library(quantmod)
#=========================================Calling Dataset from forecast Package=========================
mydata=forecast::gas
help(gas)
class(mydata)
start(mydata)
end(mydata)
plot(mydata)
monthplot(mydata)
forecast::seasonplot(mydata)
stl_for_mydata=stl(mydata,s.window = "periodic")#constant seasonality changes
plot(stl_for_mydata)
stl_for_mydata7=stl(mydata,s.window = 7)#seasonality changes
plot(stl_for_mydata7)
components_of_my_data=decompose(mydata,type = "multiplicative")#since seasonality is in multiplicative
plot(components_of_my_data)
periodicity(mydata)
#=========================================Calling Dataset from 1970 for Analysis=========================
gas_1970=ts(mydata,start = c(1970,1),end = c(1995,8),frequency = 12)
ts.plot(gas_1970)
```

```
monthplot(gas_1970)
forecast::seasonplot(gas_1970)
stl_for_gas_1970=stl(gas_1970,s.window = "periodic")#constant seasonality changes
plot(stl_for_gas_1970)
stl_for_gas_1970_7=stl(gas_1970,s.window = 7)#seasonality changes
plot(stl_for_gas_1970_7)
gas_production=(stl_for_gas_1970_7$time.series[,2]+stl_for_gas_1970_7$time.series[,3])
ts.plot(gas_production,gas_1970,col=c("red","grey"),main="Comparison of Gas and
Deseasonalized Gas")
log_gas_1970=log(gas_1970)
plot(log_gas_1970)
components_of_gas_1970=decompose(gas_1970,type = "multiplicative")#seasonalized gas -
components
plot(components_of_gas_1970)
periodicity(gas_1970)
#=======================================Partition of
Dataset==========================================
train=window(gas_1970,start=c(1970,1),end=c(1993,12),frequency=12)
test=window(gas_1970,start=c(1994,1),frequency=12)
#=======================================Stationary Check for train and
test=============================
tseries::adf.test(train)
diff_train=diff(train)
tseries::adf.test(diff_train)
plot(diff_train)
plot(test)
stl_for_train=stl(diff_train,s.window = "periodic")
plot(stl_for_train)
#=======================================Auto ARIMA Forecast for next 20
periods=========================
auto_arima_train=forecast::auto.arima(train,seasonal = TRUE,trace = TRUE)
hist(auto_arima_train$residuals,col = "aquamarine")
acf(auto_arima_train$residuals)
Box.test(auto_arima_train$residuals,lag = 30,type = "Ljung-Box")
```

```
ts.plot(train,fitted(auto_arima_train),col=c("blue","orange"))
forecast_auto_arima_train=plot(forecast::forecast(auto_arima_train,h=20))
forecast_auto_arima_train
#=======================================ARIMA forecast for next 20
periods==============================
par(mfrow=c(1,2))
acf(diff_train,lag=50,main="ACF of Stationary Time Series")
pacf(diff_train,lag=50,main="PACF of Stationary Time Series")
arima_train=arima(train,order = c(1,1,2))#with Differentiation, with MA and with AR
arima_train
arima_train_seasonal=arima(train,order = c(1,1,2),seasonal = list(order=c(0,1,1),period=12))
arima_train_seasonal
par(mfrow=c(1,1))
hist(arima_train_seasonal$residuals,col = "cyan")
acf(arima_train_seasonal$residuals)
ts.plot(train,fitted(arima_train_seasonal),col=c("green","black"))
Box.test(arima_train_seasonal$residuals,lag=30,type = "Ljung-Box")
forecast_arima_train=plot(forecast::forecast(arima_train_seasonal,h=20))
forecast_arima_train
#=======================================Auto ARIMA Final Forecast for 12
Periods=========================
auto_arima_final=forecast::auto.arima(gas_1970,seasonal = TRUE,trace = TRUE)
hist(auto_arima_final$residuals,col = "beige")
acf(auto_arima_final$residuals)
Box.test(auto_arima_final$residuals,lag=30,type = "Ljung-Box")
ts.plot(gas_1970,fitted(auto_arima_final),col=c("black","orange"))
auto_arima_final_forecast=plot(forecast::forecast(auto_arima_final,h=12))
auto_arima_final_forecast
#=======================================ARIMA Forecast for 12
Periods=====================================
tseries::adf.test(gas_1970)#stationary check
diff_gas=diff(gas_1970)#differentiation as the data is not stationary
tseries::adf.test(diff_gas)#checking the stationarity for the differentiation
plot(diff_gas)
```

```
par(mfrow=c(1,2))
acf(diff_gas,lag=50,main="ACF of Stationary Time Series")
pacf(diff_gas,lag=50,main="PACF of Statioanry Time Series")
arima_final=arima(gas_1970,order = c(5,1,2),seasonal = list(order=c(1,1,1),period=12))
arima_final
par(mfrow=c(1,1))
hist(arima_final$residuals,col = "blue")
acf(arima_final$residuals)
Box.test(arima_final$residuals,lag = 30,type = "Ljung-Box")
ts.plot(gas_1970,fitted(arima_final),col=c("grey","violet"))
arima_final_forecast=plot(forecast::forecast(arima_final,h=12))
arima_final_forecast
#=====================================Accuracy of the
Model=============================================
forecast::accuracy(auto_arima_train)
forecast::accuracy(arima_train_seasonal)
forecast::accuracy(auto_arima_final)
forecast::accuracy(arima_final)
```