Model Report

<u>Mini Project : Australian Monthly Gas Production</u>

CONTENTS

1. Project Objective

The objective of the project is to explore the gas dataset from the Forecast package in R.

- Read the data as a time series object in R. Plot the data **(5 marks)**

- What do you observe? Which components of the time series are present in this dataset? **(5 marks)**

- What is the periodicity of dataset? **(5 marks)**

- 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? **(20 marks)**

- Develop an ARIMA Model to forecast for next 12 periods. Use both manual and auto.arima (Show & explain all the steps) **(20 marks)**

- Report the accuracy of the model **(5 marks)**

2. Exploratory Data Analysis

- It is a Time Series Data
- Number of rows in the dataset is 476
- There is only 1 column
- Frequency of the series is monthly
- There is an upward trend
- Seasonality is multiplicative
- Data starts from January 1956 and ends at August 1995

- **Structure of the dataset**

  It is a time series.

```
str(gas)
```

```
##  Time-Series [1:476] from 1956 to 1996: 1709 1646 1794 1878 2173 ...
```

- **Summary of the dataset**
  The summary of the dataset shows the min. value, 1st quartile, median, mean, 3rd quartile and the max. value of all the variables.

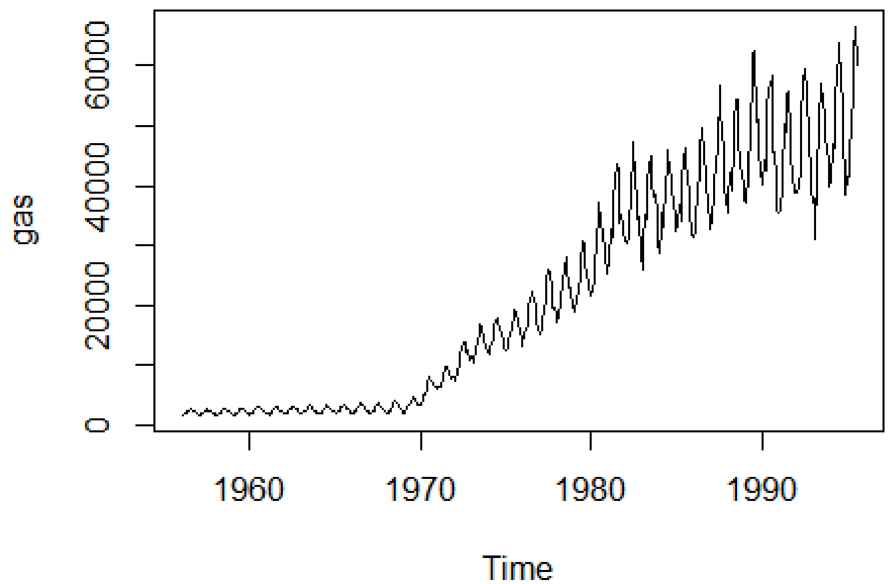
```
summary(gas)

##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##    1646    2675   16788   21415   38629   66600
```

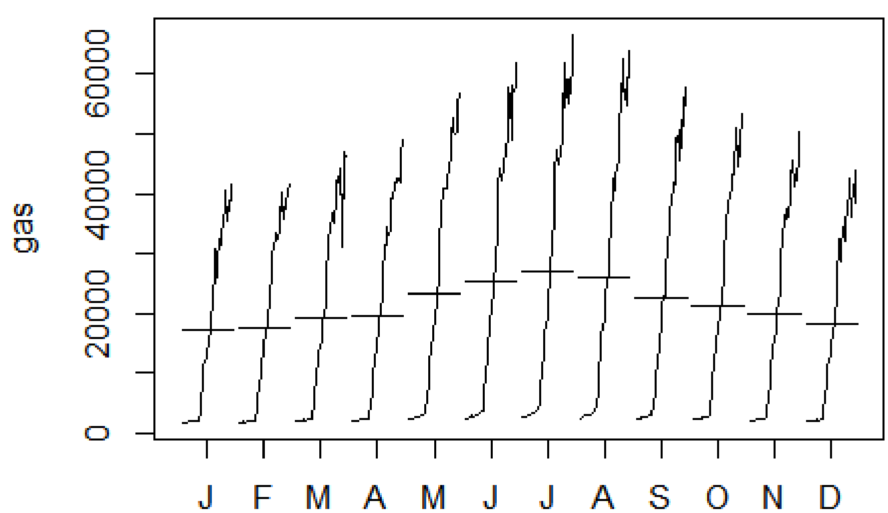
- **Missing Values Detection and omission**
  There are no **missing values** in the dataset. The test was done using sum(is.na) function.

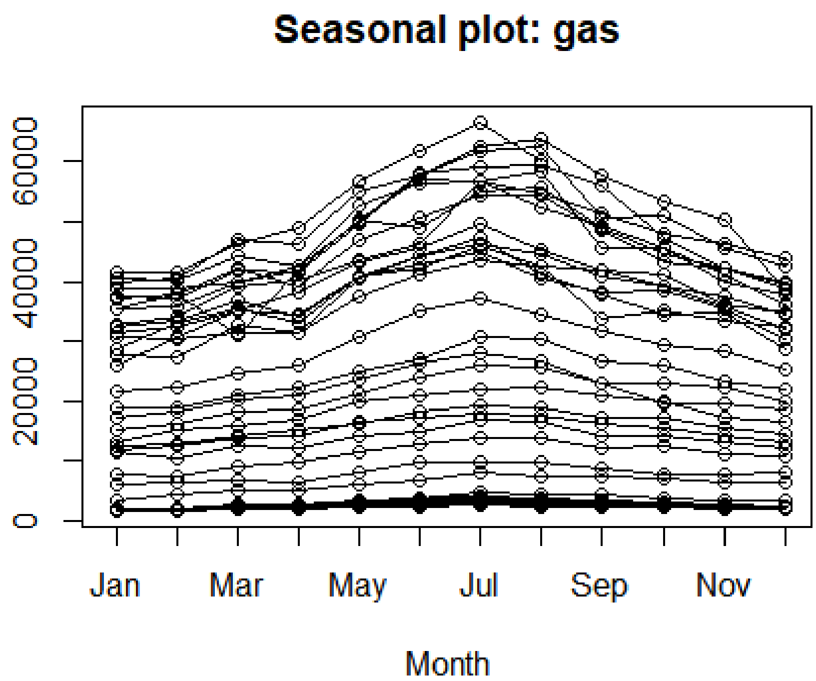
```
# Checking for any NA

sum(is.na(gas))

## [1] 0
```

3. Data Visualization

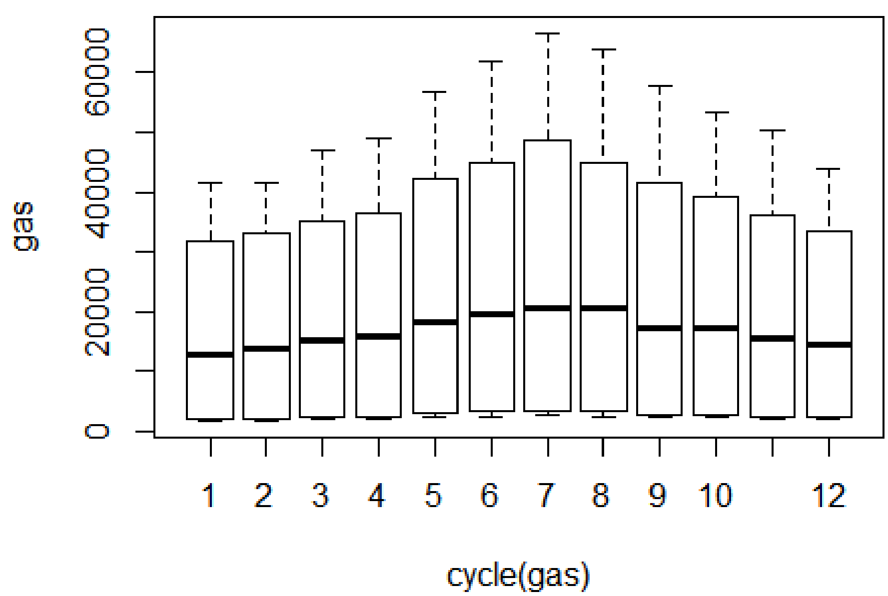
Below are the various plots


gas
0
20000
40000
60000
1960
1970
1980
1990
Time


gas
0
20000
40000
60000
J F M A M J J A S O N D



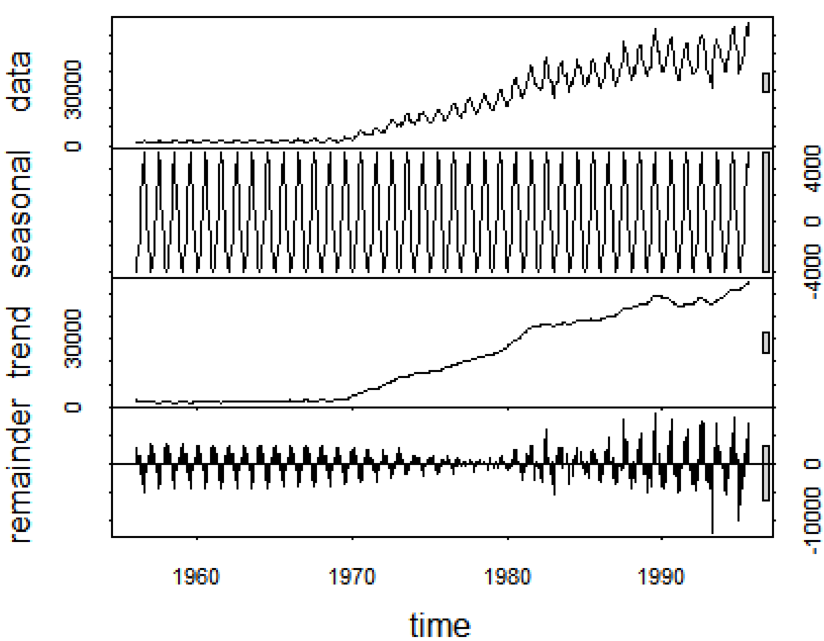

```r
boxplot(gas~cycle(gas))
```



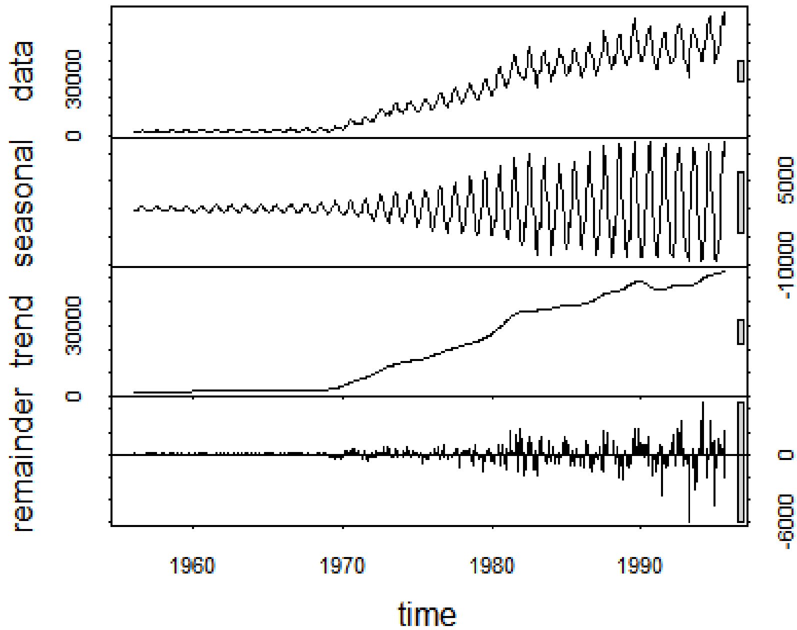
4. Insights

- From the visual observation, we can say that there is an "***upward trend***" in the series
- ***"Seasonality"*** is also there which is multiplicative
- Variance is not constant
- There is no missing value
- It is a non stationary series, we will have to stationarize it in order to perform further analysis
- Log transformation is required followed by differencing the series

```
# Decompose the data

gas_data_const <- stl(gas, s.window = "p")
plot(gas_data_const)
```




```
gas_data <- stl(gas, s.window = 5)
plot(gas_data)
```



5. **Periodicity**

Periodicity in a time series data means the pattern repeats at regular time intervals. The time series is called cyclic, if the time intervals at which the pattern repeats itself can not be defined and is not constant.

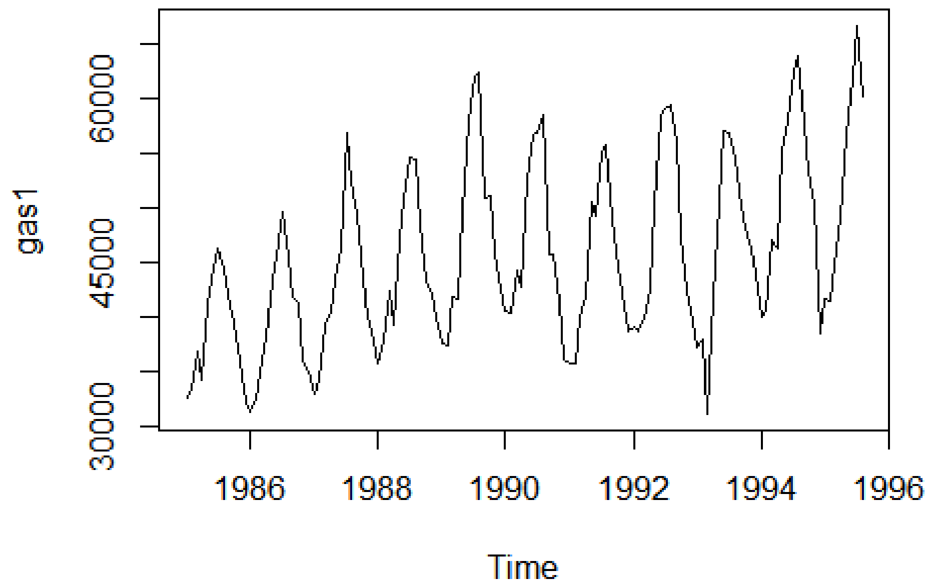
Here, in this case the periodicity is monthly.

6. **Data Subset**

From the original series of data, we can observe and interpret that the entire data from 1956 is not relevant for our analysis. Hence, we will perform our analysis from 1985 onwards.

```
# Taking the subset of the data as the entire data is irrelevant
```

```r
gas1<- window(gas, start=c(1985,1), end=c(1995,8))
plot(gas1)
```

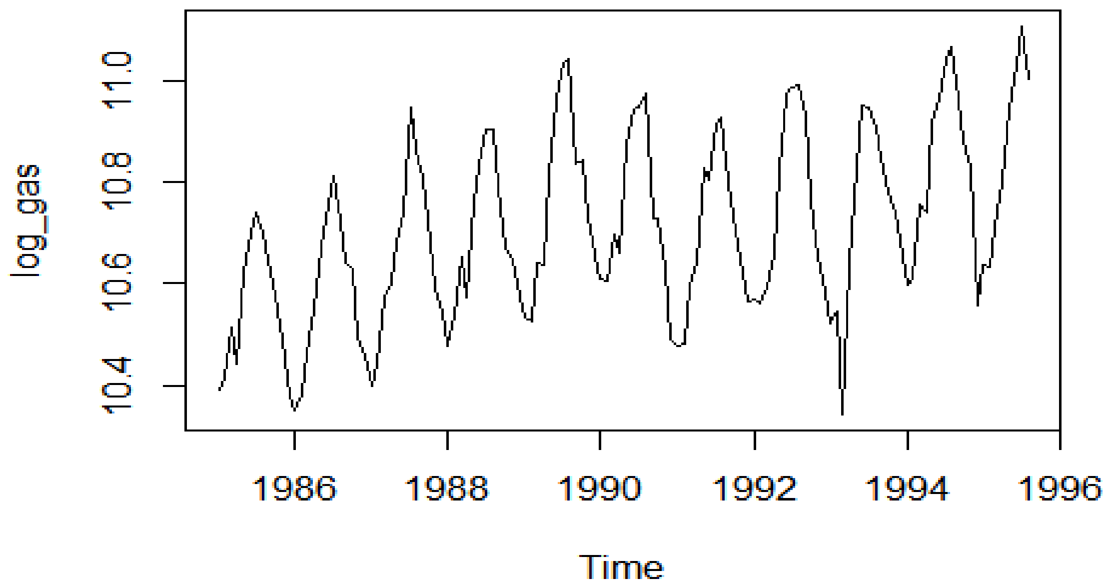


7. Is The Time Series Stationary?

The time series in the context is ***not stationary*** as there is an upward trend along with multiplicative seasonality.

We will have to make it stationarize (Taking Log and then differencing)

```r
# Taking log of the series to make the variance constant

log_gas <- log(gas1)
plot(log_gas)
```



8. Augmented Dickey-Fuller Test (ADF Test)

ADF Test is a common statistical test used to check the stationarity of the series. It tests the null hypothesis. It is the most common test used for this purpose.

Here we have taken the log transform of the series and then differenced the log transform (difference=1). After this, on applying ADF Test we found that the p-value is 0.01. Therefore, it means the series is now stationary.

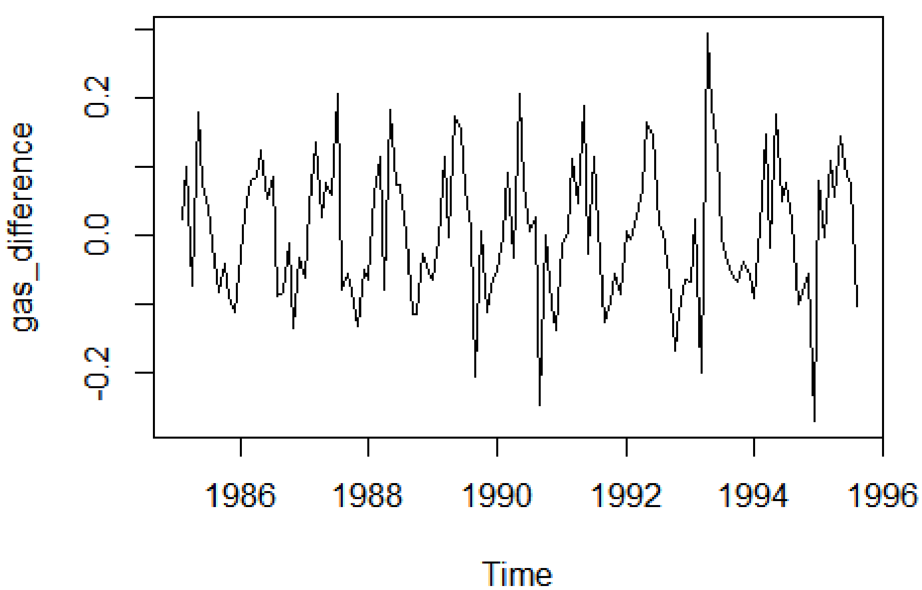
Null and Alternative Hypothesis For The Stationarity Test

Null Hypothesis, H0 : Non-Stationary

Alternative Hypothesis, Ha: Stationary

```
library(tseries)

gas_augdf <- adf.test(log_gas, alternative = "stationary")
```

```
## Warning in adf.test(log_gas, alternative = "stationary"): p-value smaller
## than printed p-value

gas_augdf

##
##  Augmented Dickey-Fuller Test
##
## data:  log_gas
## Dickey-Fuller = -7.3234, Lag order = 5, p-value = 0.01
## alternative hypothesis: stationary

# Differencing to address the trend in the series

gas_difference <- diff(log_gas, differences = 1)
plot(gas_difference)
```




```
gas_adf <- adf.test(gas_difference, alternative = "stationary")

## Warning in adf.test(gas_difference, alternative = "stationary"): p-value
## smaller than printed p-value

gas_adf

##
##  Augmented Dickey-Fuller Test
##
## data:  gas_difference
```

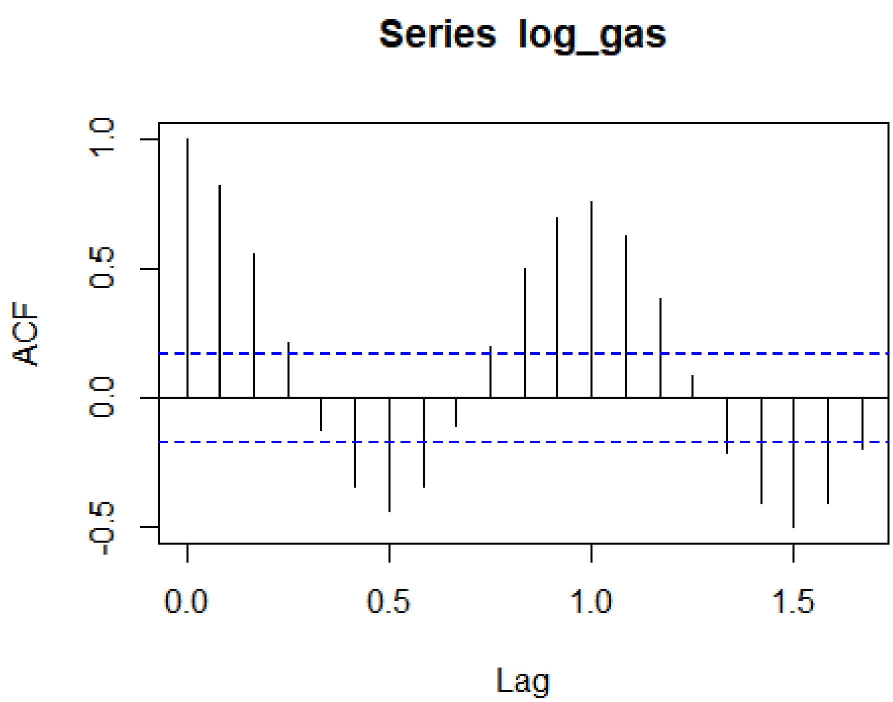
```
## Dickey-Fuller = -10.319, Lag order = 5, p-value = 0.01
## alternative hypothesis: stationary
```

```
# Now the series is stationary with p-value = 0.01
```

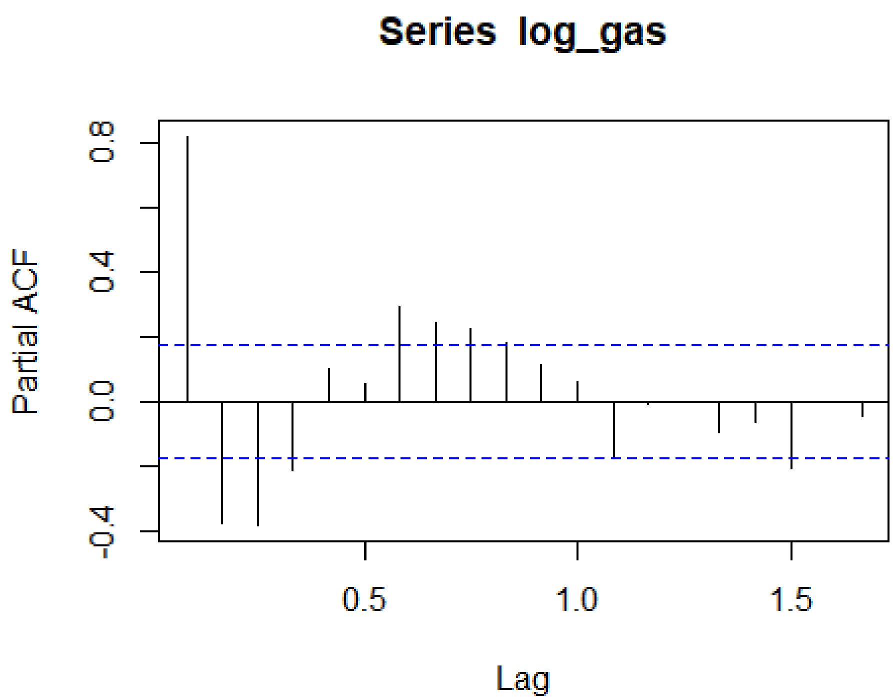
9. ACF and PACF Plots

ACF is used to get the p value (AR) and PACF is used to get the q value (MA).

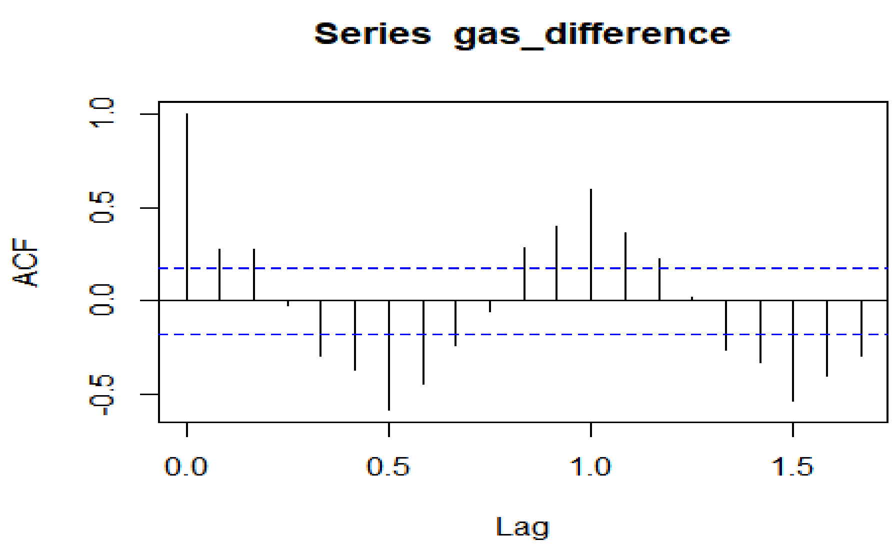
```
# ACF and PACF plots

acf(log_gas, lag.max = 20)
```



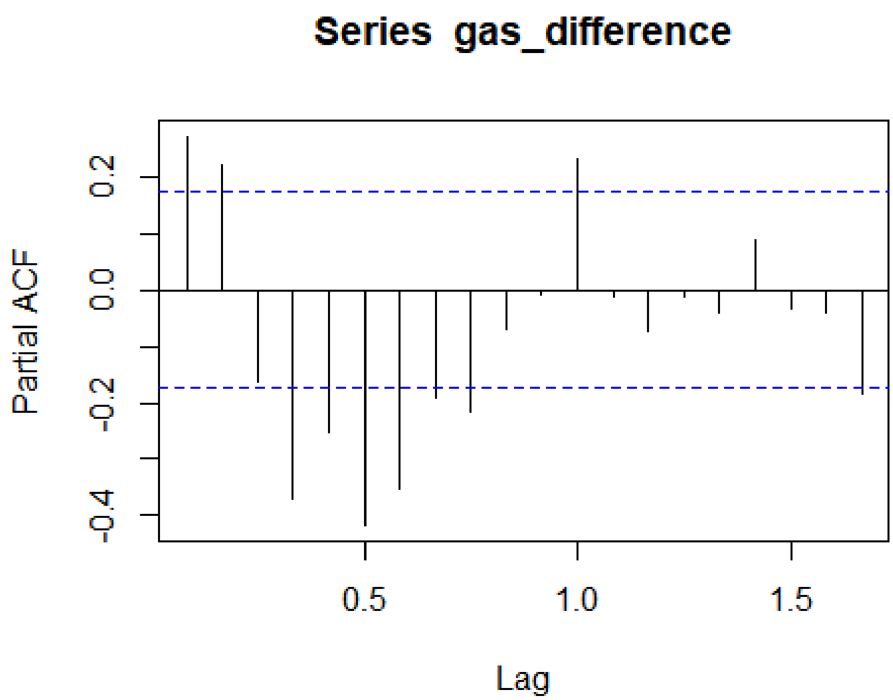

```
pacf(log_gas, lag.max = 20)
```




```r
# ACF and PACF plots on differeced series

acf(gas_difference, lag.max = 20)
```



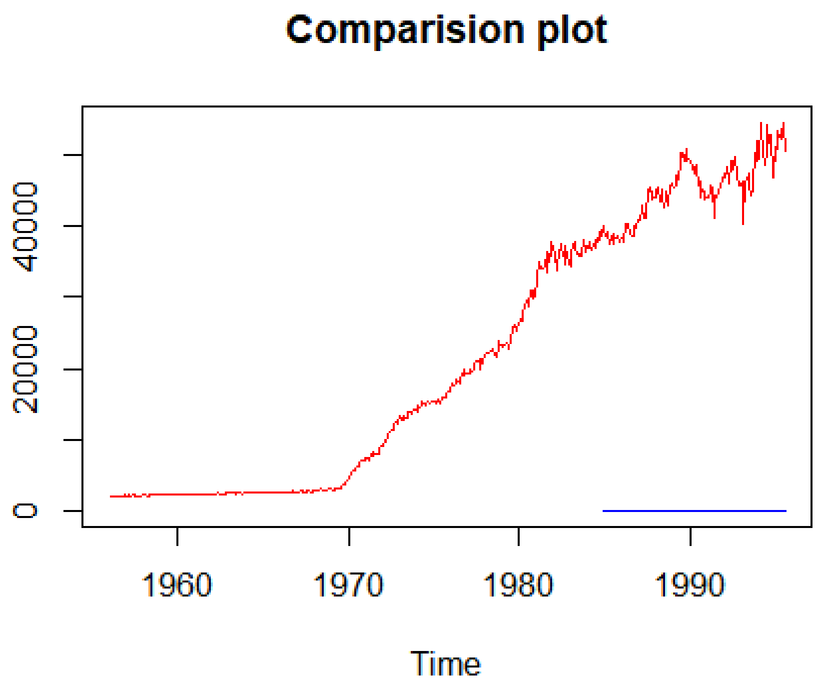

```r
pacf(gas_difference, lag.max = 20)
```



10. Deseasonalise

Since seasonality is present in this data, we will perform the decompose operation to remove the seasonality effect from this series.

```
# Deseasonalise the data

deseason_gas <- (gas_data$time.series[,2] + gas_data$time.series[,3])
plot <- ts.plot(deseason_gas, log_gas, col = c("red", "blue"), main = "Compar
ision plot")
```



11. Training and Test Data

Dividing the data into training and test data for our analysis purpose.
We have taken the subset data here for this division.

```
# Dividing the series into train and test

gas_train = window(log_gas, start=c(1985,1), end=c(1992,12))
gas_test= window(log_gas, start=c(1993,1), end=c(1995,8))
```

12. ARIMA Model

ARIMA is called as 'Auto Regressive Integrated Moving Average'. It is a class of models that explains a given time series based on its own past values i.e its own lags and the lagged forecast errors, so that equation can be used to forecast future values.

Here we are required to build an ARIMA model (Both manual and auto ARIMA) to forecast for next 12 periods.

<u>We have tested the manual ARIMA with various SAR and SMA values to get the best model.</u>

```
# ARIMA Model on train data (checking with various SAR and SMA components)

arima_train <- arima(gas_train, order = c(2,1,2), seasonal = c(0,1,0))
arima_train

##
## Call:
## arima(x = gas_train, order = c(2, 1, 2), seasonal = c(0, 1, 0))
##
## Coefficients:
##           ar1      ar2     ma1      ma2
##       -0.9841  -0.1157  0.5660  -0.4340
## s.e.   0.2071   0.2021  0.1945   0.1918
##
## sigma^2 estimated as 0.002916:  log likelihood = 123.36,  aic = -236.72

Box.test(arima_train$residuals, lag = 20, type = "Ljung-Box")

##
##  Box-Ljung test
##
## data:  arima_train$residuals
## X-squared = 26.959, df = 20, p-value = 0.1364

arima_train <- arima(gas_train, order = c(2,1,2), seasonal = c(1,1,0))

## Warning in log(s2): NaNs produced

## Warning in log(s2): NaNs produced

arima_train

##
## Call:
## arima(x = gas_train, order = c(2, 1, 2), seasonal = c(1, 1, 0))
##
## Coefficients:
##           ar1      ar2     ma1      ma2     sar1
##       -0.9903  -0.1232  0.4984  -0.5016  -0.3461
## s.e.   0.1795   0.1767  0.1644   0.1608   0.1125
##
## sigma^2 estimated as 0.002567:  log likelihood = 127.6,  aic = -243.2

Box.test(arima_train$residuals, lag = 20, type = "Ljung-Box")
```

```
##
##  Box-Ljung test
##
## data:  arima_train$residuals
## X-squared = 25.856, df = 20, p-value = 0.1706
```

```
arima_train <- arima(gas_train, order = c(2,1,2), seasonal = c(0,1,1))
arima_train
```

```
##
## Call:
## arima(x = gas_train, order = c(2, 1, 2), seasonal = c(0, 1, 1))
##
## Coefficients:
##           ar1      ar2     ma1      ma2     sma1
##       -0.9620  -0.1098  0.4791  -0.5209  -0.7750
## s.e.   0.1786   0.1770  0.2306   0.1795   0.1746
##
## sigma^2 estimated as 0.001966:  log likelihood = 133.38,  aic = -254.76
```

```
Box.test(arima_train$residuals, lag = 20, type = "Ljung-Box")
```

```
##
##  Box-Ljung test
##
## data:  arima_train$residuals
## X-squared = 18.608, df = 20, p-value = 0.5474
```

```
arima_train <- arima(gas_train, order = c(2,1,2), seasonal = c(1,1,1))
arima_train
```

```
##
## Call:
## arima(x = gas_train, order = c(2, 1, 2), seasonal = c(1, 1, 1))
##
## Coefficients:
##           ar1      ar2     ma1      ma2    sar1     sma1
##       -0.9487  -0.0967  0.4944  -0.5056  0.1829  -0.9992
## s.e.   0.1884   0.1866  0.2037   0.1783  0.1372   0.4041
##
## sigma^2 estimated as 0.001711:  log likelihood = 134.06,  aic = -254.13
```

```
Box.test(arima_train$residuals, lag = 20, type = "Ljung-Box")
```

```
##
##  Box-Ljung test
##
## data:  arima_train$residuals
## X-squared = 16.717, df = 20, p-value = 0.6713
```

```
arima_train <- arima(gas_train, order = c(2,1,2), seasonal = c(2,1,0))
arima_train
```

```
##
## Call:
## arima(x = gas_train, order = c(2, 1, 2), seasonal = c(2, 1, 0))
##
## Coefficients:
##           ar1      ar2     ma1      ma2     sar1     sar2
##       -0.9813  -0.1467  0.5185  -0.4458  -0.4711  -0.3808
## s.e.   0.1841   0.1878  0.1780   0.1833   0.1193   0.1299
##
## sigma^2 estimated as 0.002239:  log likelihood = 131.79,  aic = -249.57
```

```
Box.test(arima_train$residuals, lag = 20, type = "Ljung-Box")
```

```
##
##  Box-Ljung test
##
## data:  arima_train$residuals
## X-squared = 22.04, df = 20, p-value = 0.3383
```

```
arima_train <- arima(gas_train, order = c(2,1,2), seasonal = c(2,1,1))
arima_train
```

```
##
## Call:
## arima(x = gas_train, order = c(2, 1, 2), seasonal = c(2, 1, 1))
##
## Coefficients:
##           ar1      ar2     ma1      ma2    sar1     sar2     sma1
##       -0.9537  -0.1109  0.5019  -0.4691  0.0972  -0.1582  -0.7743
## s.e.   0.1974   0.2131  0.2349   0.2272  0.2745   0.2159   0.3828
##
## sigma^2 estimated as 0.001921:  log likelihood = 134.45,  aic = -252.9
```

```
Box.test(arima_train$residuals, lag = 20, type = "Ljung-Box")
```

```
##
##  Box-Ljung test
##
## data:  arima_train$residuals
## X-squared = 15.199, df = 20, p-value = 0.7649
```

```
arima_train <- arima(gas_train, order = c(2,1,2), seasonal = c(2,1,2))
arima_train
```

```
##
## Call:
## arima(x = gas_train, order = c(2, 1, 2), seasonal = c(2, 1, 2))
##
## Coefficients:
```

```
## Warning in sqrt(diag(x$var.coef)): NaNs produced
```

```
##           ar1      ar2     ma1      ma2    sar1     sar2     sma1    sma2
##       -0.9588  -0.1183  0.4924  -0.4688  0.8626  -0.4122  -1.6392  0.9231
## s.e.      NaN      NaN     NaN      NaN  0.1295      NaN      NaN     NaN
##
## sigma^2 estimated as 0.001694:  log likelihood = 134.7,  aic = -251.4
```

```
Box.test(arima_train$residuals, lag = 20, type = "Ljung-Box")
```

```
##
##  Box-Ljung test
##
## data:  arima_train$residuals
## X-squared = 14.253, df = 20, p-value = 0.8174
```

```
arima_train <- arima(gas_train, order = c(2,1,2), seasonal = c(0,1,2))
arima_train
```

```
##
## Call:
## arima(x = gas_train, order = c(2, 1, 2), seasonal = c(0, 1, 2))
##
## Coefficients:
##           ar1      ar2     ma1      ma2     sma1     sma2
##       -0.9471  -0.0963  0.5047  -0.4953  -0.7902  -0.2082
## s.e.   0.1922   0.1901  0.2142   0.1850   0.4233   0.1592
##
## sigma^2 estimated as 0.001699:  log likelihood = 134.25,  aic = -254.5
```

```
Box.test(arima_train$residuals, lag = 20, type = "Ljung-Box")
```

```
##
##  Box-Ljung test
##
## data:  arima_train$residuals
## X-squared = 16.135, df = 20, p-value = 0.7082
```

```
arima_train <- arima(gas_train, order = c(2,1,2), seasonal = c(1,1,2))
arima_train
```

```
##
## Call:
## arima(x = gas_train, order = c(2, 1, 2), seasonal = c(1, 1, 2))
##
## Coefficients:
##           ar1      ar2     ma1      ma2     sar1     sma1     sma2
##       -0.9495  -0.1034  0.5069  -0.4927  -0.5150  -0.2052  -0.5985
## s.e.   0.1900   0.1876  0.9752   0.5058   1.1448   1.3133   0.7082
##
## sigma^2 estimated as 0.001824:  log likelihood = 134.42,  aic = -252.83
```

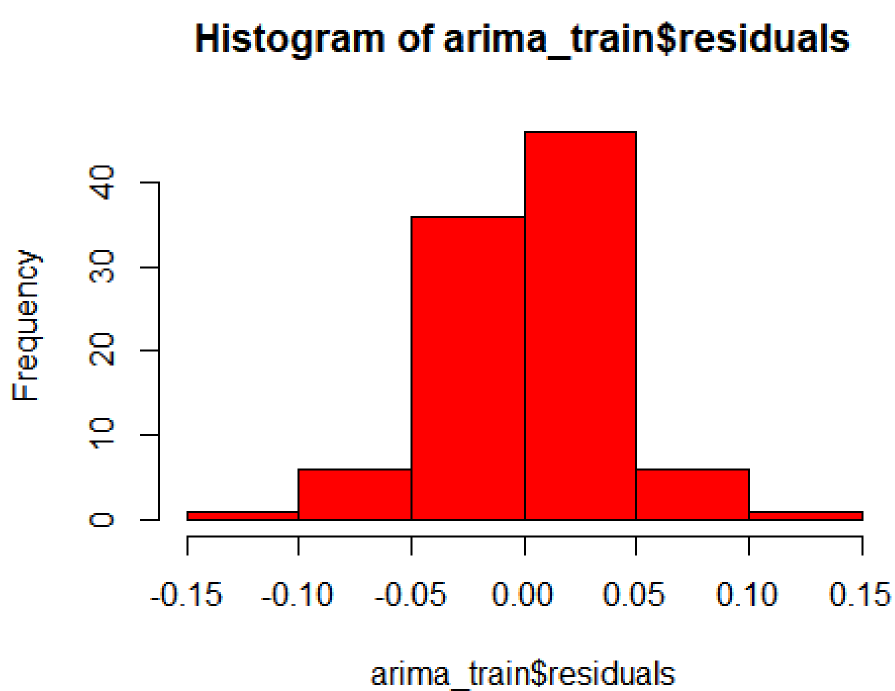
```
Box.test(arima_train$residuals, lag = 20, type = "Ljung-Box")
```

```
##
##  Box-Ljung test
##
## data:  arima_train$residuals
## X-squared = 15.79, df = 20, p-value = 0.7296
```

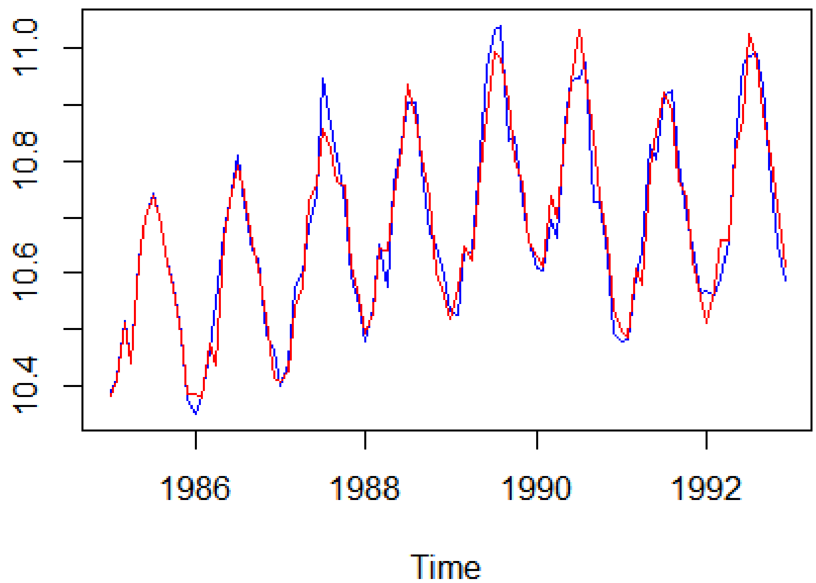
```r
# The best model obtained with SAR = 2 and SMA = 2

arima_train <- arima(gas_train, order = c(2,1,2), seasonal = c(2,1,2))
arima_train
```

```
##
## Call:
## arima(x = gas_train, order = c(2, 1, 2), seasonal = c(2, 1, 2))
##
## Coefficients:

## Warning in sqrt(diag(x$var.coef)): NaNs produced

##           ar1      ar2     ma1      ma2    sar1     sar2     sma1    sma2
##       -0.9588  -0.1183  0.4924  -0.4688  0.8626  -0.4122  -1.6392  0.9231
## s.e.      NaN      NaN     NaN      NaN  0.1295      NaN      NaN     NaN
##
## sigma^2 estimated as 0.001694:  log likelihood = 134.7,  aic = -251.4
```

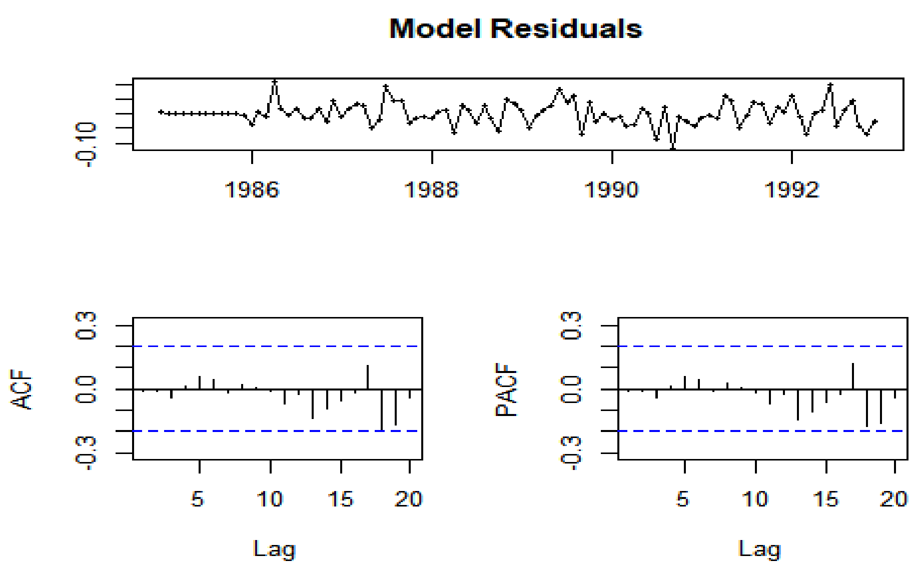
```r
hist(arima_train$residuals, col = "red")
```



```r
arima_train_fit <- fitted(arima_train)
ts.plot(gas_train,arima_train_fit, col = c("blue", "red"))
```




```r
tsdisplay(residuals(arima_train), lag.max = 20, main = 'Model Residuals')
```



13. Box-Test on Manual ARIMA

```
# Box Test

Box.test(arima_train$residuals, lag = 20, type = "Ljung-Box")

##
##  Box-Ljung test
##
## data:  arima_train$residuals
## X-squared = 14.253, df = 20, p-value = 0.8174
```

We have got the best model with order(2,1,2) seasonal(2,1,2)
Its AIC value is -251.4 and p-value from Box-Test is 0.8174 which is the best among all other seasonal effects. It means the residulas are independent because we do not reject the null hypothesis, H0

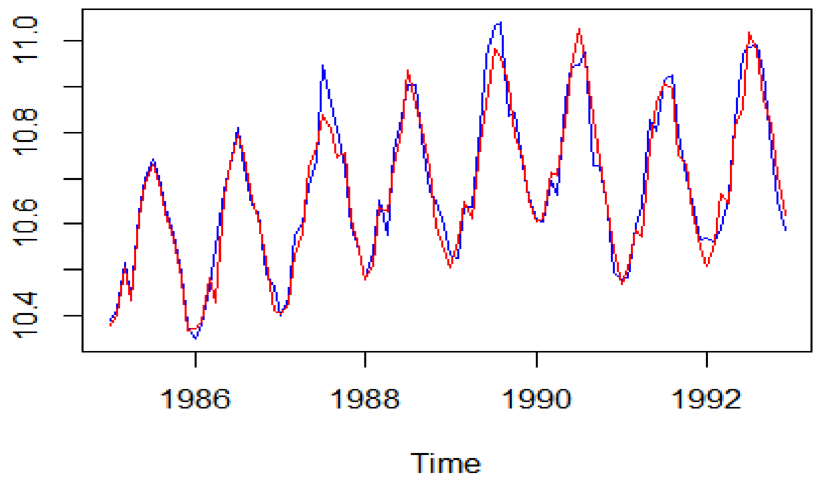
As, Null Hypothesis, Ho : Residuals are independent
Alternative Hypothesis, Ha : Residuals are dependent

14. Auto ARIMA

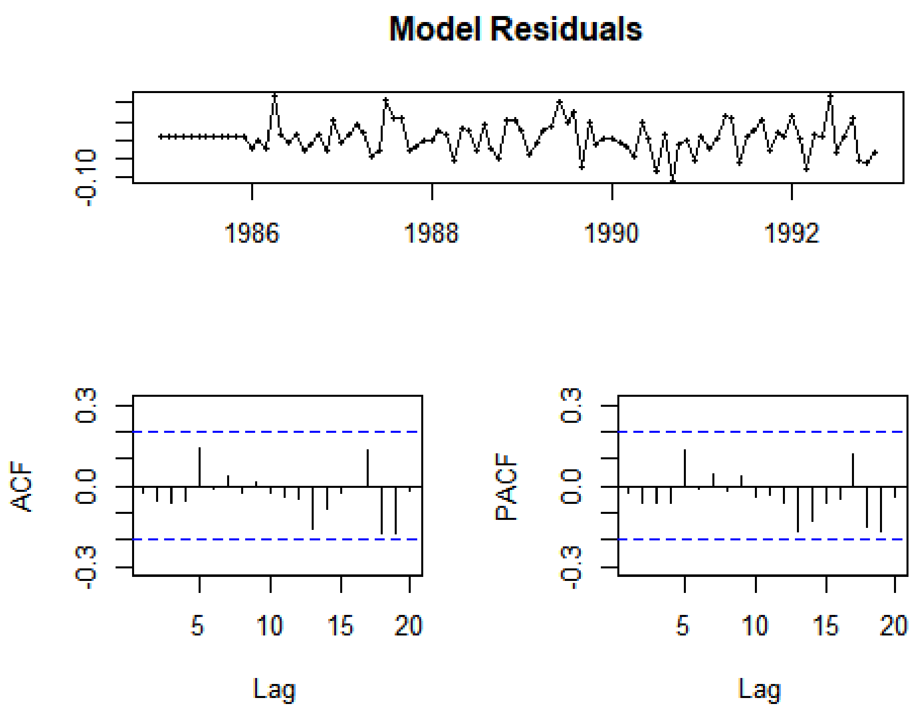
On performing auto ARIMA, the model with order(3,0,0) seasonal(2,1,1) was the best one.

```
# Auto ARIMA on train data

autoarima_train <- auto.arima(gas_train, seasonal = TRUE)
autoarima_train

## Series: gas_train
## ARIMA(3,0,0)(2,1,1)[12]
##
## Coefficients:
##          ar1     ar2     ar3    sar1     sar2     sma1
##       0.4810  0.1932  0.2537  0.0796  -0.2110  -0.7060
## s.e.  0.1092  0.1165  0.1064  0.2252   0.1707   0.2894
##
## sigma^2 estimated as 0.002198:  log likelihood=135.75
## AIC=-257.51   AICc=-256.03   BIC=-240.49
```

```
autoarima_train_fit <- fitted(autoarima_train)
ts.plot(gas_train,autoarima_train_fit, col = c("blue", "red"))
```




```
tsdisplay(residuals(autoarima_train), lag.max = 20, main = 'Model Residuals')
```



15. Box-Test on Auto ARIMA

```
# Box Test on auto arima

Box.test(autoarima_train$residuals, lag = 20, type = "Ljung-Box")

##
##  Box-Ljung test
##
## data:  autoarima_train$residuals
## X-squared = 17.196, df = 20, p-value = 0.6402
```

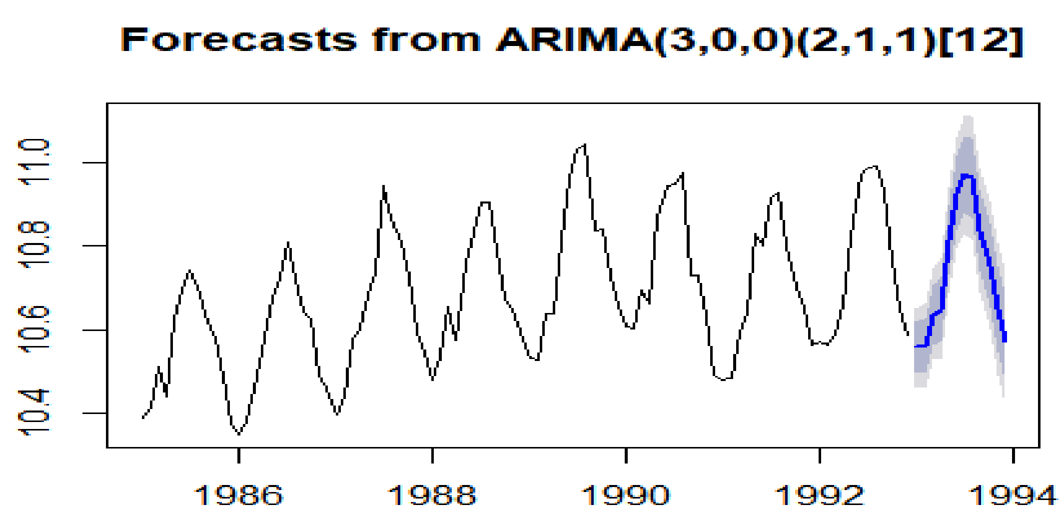
The Box Test on Auto ARIMA gives a p-value of 0.6402 which means the residuals are independent and we do not reject null hypothesis,H0

16. Forecast

The forecast for the next 12 periods from Auto ARIMA results

The plot shows the forecast with 80 and 95 percent confidende intervals in dark grey and light grey.

```
# Forecast on train data

gas_train_autofc <- forecast(autoarima_train, h = 12)
plot(gas_train_autofc)
```



*17.*Accuracy

The accuracy of the model is a measure to judge how good a model is for the forecasting purpose. In time series data, we broadly use the MAPE value to check the accuracy of the model. MAPE stands for Mean Absolute Percentage Error.

The MAPE for the traing data is around .30 whereas for test data it is around 0.56

As the error is very small, we can say that the model is good to be used for forecasting the future production of gas.

```
# Accuracy

accuracy <- accuracy(gas_train_autofc, gas_test)
accuracy

##                        ME       RMSE        MAE         MPE      MAPE
## Training set  0.007063177 0.04226108 0.03211989  0.06486765 0.2992514
## Test set     -0.013297032 0.09760529 0.05938956 -0.13259204 0.5614692
##                   MASE       ACF1 Theil's U
## Training set 0.4702325 -0.0250175        NA
## Test set     0.8694582  0.1636908 0.7575444
```

APPENDIX

```r
options(repos=c(CRAN="http://cran.rstudio.com"))
install.packages("forecast")

## Installing package into 'C:/Users/user/Documents/R/win-library/3.6'
## (as 'lib' is unspecified)

## package 'forecast' successfully unpacked and MD5 sums checked

## Warning: cannot remove prior installation of package 'forecast'

## Warning in file.copy(savedcopy, lib, recursive =
## TRUE): problem copying C:\Users\user\Documents\R\win-
## library\3.6\00LOCK\forecast\libs\x64\forecast.dll to C:
## \Users\user\Documents\R\win-library\3.6\forecast\libs\x64\forecast.dll:
## Permission denied

## Warning: restored 'forecast'

##
## The downloaded binary packages are in
##  C:\Users\user\AppData\Local\Temp\RtmpcflgRS\downloaded_packages

library(forecast)

## Registered S3 method overwritten by 'xts':
##   method     from
##   as.zoo.xts zoo

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

## Registered S3 methods overwritten by 'forecast':
##   method             from
##   fitted.fracdiff    fracdiff
##   residuals.fracdiff fracdiff

# Reading the data from the forecast package

data("gas")
View(gas)

# Checking for any NA

sum(is.na(gas))

## [1] 0
```

```r
# Exploratory data analysis

class(gas)

## [1] "ts"

start(gas)

## [1] 1956    1

end(gas)

## [1] 1995    8

frequency(gas)

## [1] 12

cycle(gas)

##      Jan Feb Mar Apr May Jun Jul Aug Sep Oct Nov Dec
## 1956   1   2   3   4   5   6   7   8   9  10  11  12
## 1957   1   2   3   4   5   6   7   8   9  10  11  12
## 1958   1   2   3   4   5   6   7   8   9  10  11  12
## 1959   1   2   3   4   5   6   7   8   9  10  11  12
## 1960   1   2   3   4   5   6   7   8   9  10  11  12
## 1961   1   2   3   4   5   6   7   8   9  10  11  12
## 1962   1   2   3   4   5   6   7   8   9  10  11  12
## 1963   1   2   3   4   5   6   7   8   9  10  11  12
## 1964   1   2   3   4   5   6   7   8   9  10  11  12
## 1965   1   2   3   4   5   6   7   8   9  10  11  12
## 1966   1   2   3   4   5   6   7   8   9  10  11  12
## 1967   1   2   3   4   5   6   7   8   9  10  11  12
## 1968   1   2   3   4   5   6   7   8   9  10  11  12
## 1969   1   2   3   4   5   6   7   8   9  10  11  12
## 1970   1   2   3   4   5   6   7   8   9  10  11  12
## 1971   1   2   3   4   5   6   7   8   9  10  11  12
## 1972   1   2   3   4   5   6   7   8   9  10  11  12
## 1973   1   2   3   4   5   6   7   8   9  10  11  12
## 1974   1   2   3   4   5   6   7   8   9  10  11  12
## 1975   1   2   3   4   5   6   7   8   9  10  11  12
## 1976   1   2   3   4   5   6   7   8   9  10  11  12
## 1977   1   2   3   4   5   6   7   8   9  10  11  12
## 1978   1   2   3   4   5   6   7   8   9  10  11  12
## 1979   1   2   3   4   5   6   7   8   9  10  11  12
## 1980   1   2   3   4   5   6   7   8   9  10  11  12
## 1981   1   2   3   4   5   6   7   8   9  10  11  12
## 1982   1   2   3   4   5   6   7   8   9  10  11  12
## 1983   1   2   3   4   5   6   7   8   9  10  11  12
## 1984   1   2   3   4   5   6   7   8   9  10  11  12
## 1985   1   2   3   4   5   6   7   8   9  10  11  12
## 1986   1   2   3   4   5   6   7   8   9  10  11  12
```

```
## 1987   1   2   3   4   5   6   7   8   9  10  11  12
## 1988   1   2   3   4   5   6   7   8   9  10  11  12
## 1989   1   2   3   4   5   6   7   8   9  10  11  12
## 1990   1   2   3   4   5   6   7   8   9  10  11  12
## 1991   1   2   3   4   5   6   7   8   9  10  11  12
## 1992   1   2   3   4   5   6   7   8   9  10  11  12
## 1993   1   2   3   4   5   6   7   8   9  10  11  12
## 1994   1   2   3   4   5   6   7   8   9  10  11  12
## 1995   1   2   3   4   5   6   7   8
```

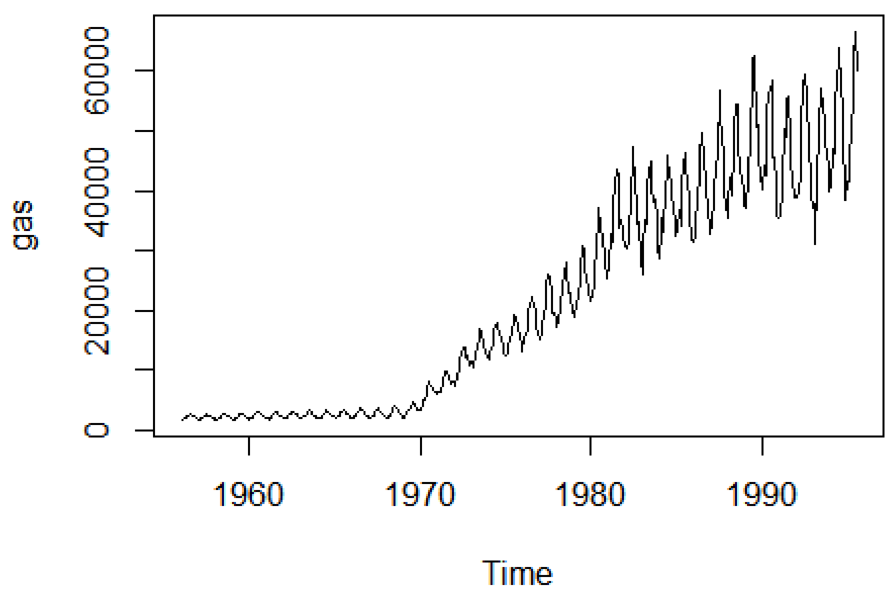
```
summary(gas)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max. 
##    1646    2675   16788   21415   38629   66600
```

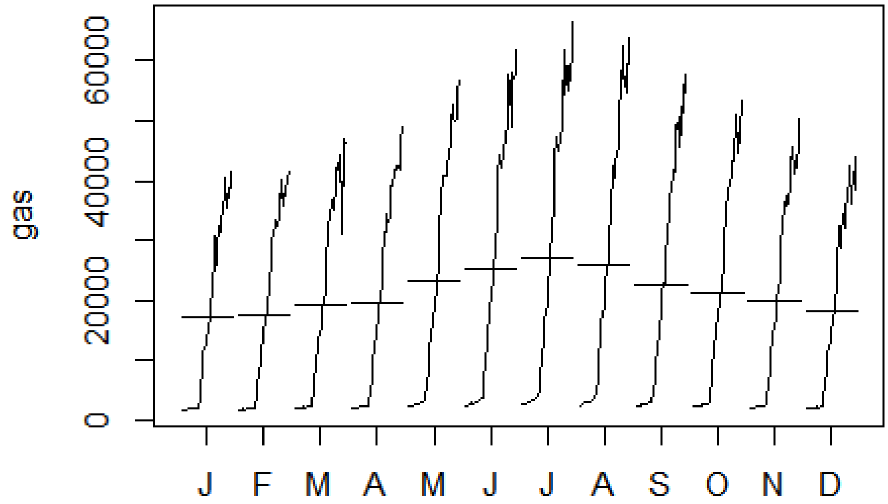
```
str(gas)
```

```
##  Time-Series [1:476] from 1956 to 1996: 1709 1646 1794 1878 2173 ...
```

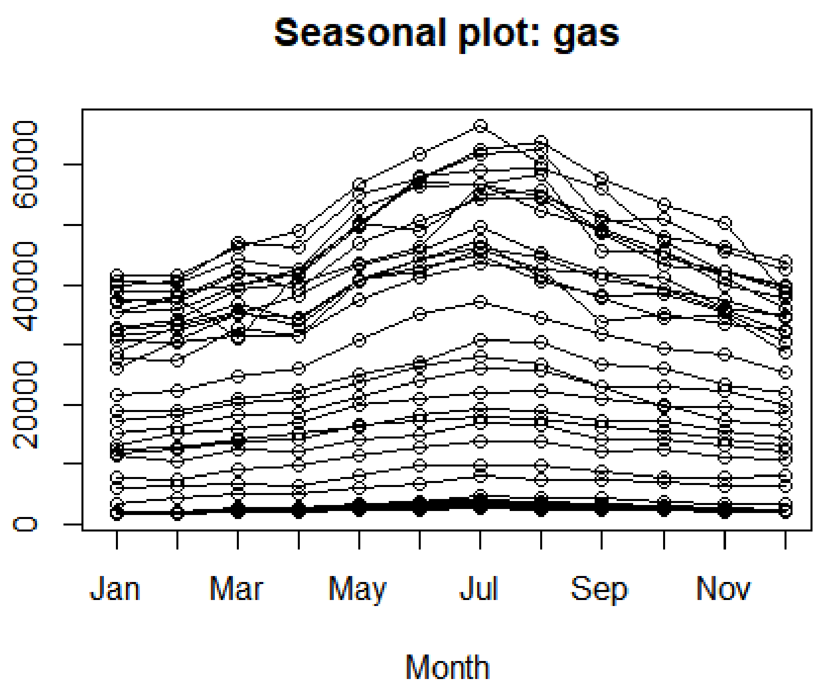
```
# Plot of the time series data

plot(gas)
```



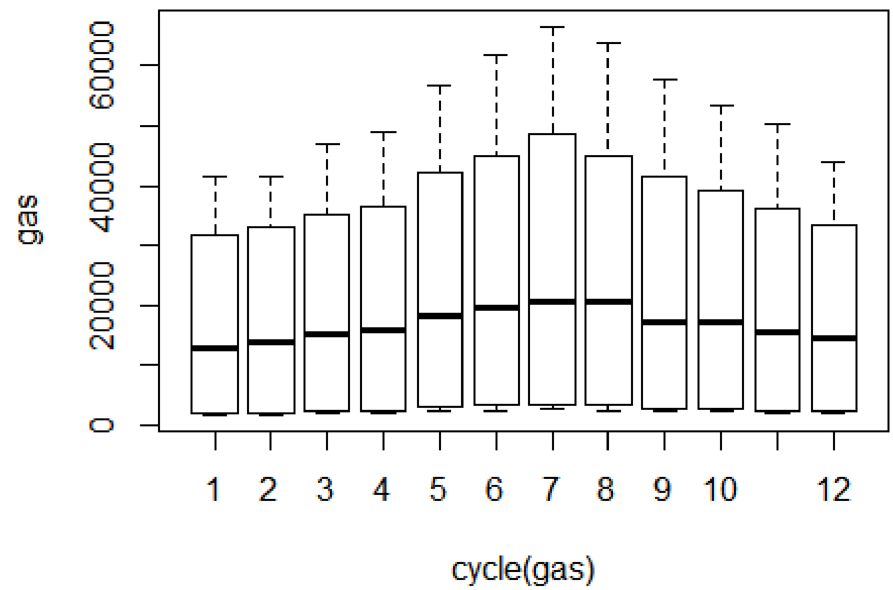

```
monthplot(gas)
```



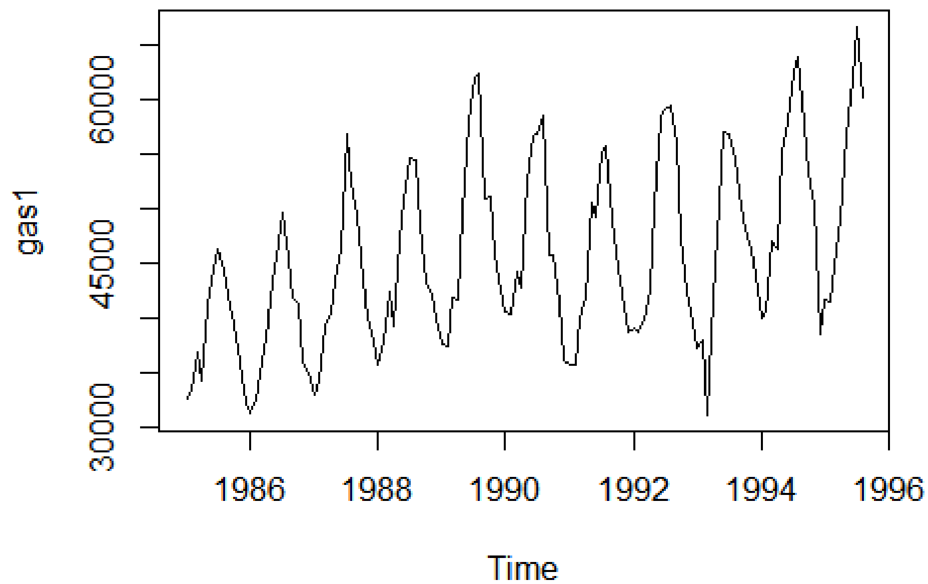

```r
seasonplot(gas)
```



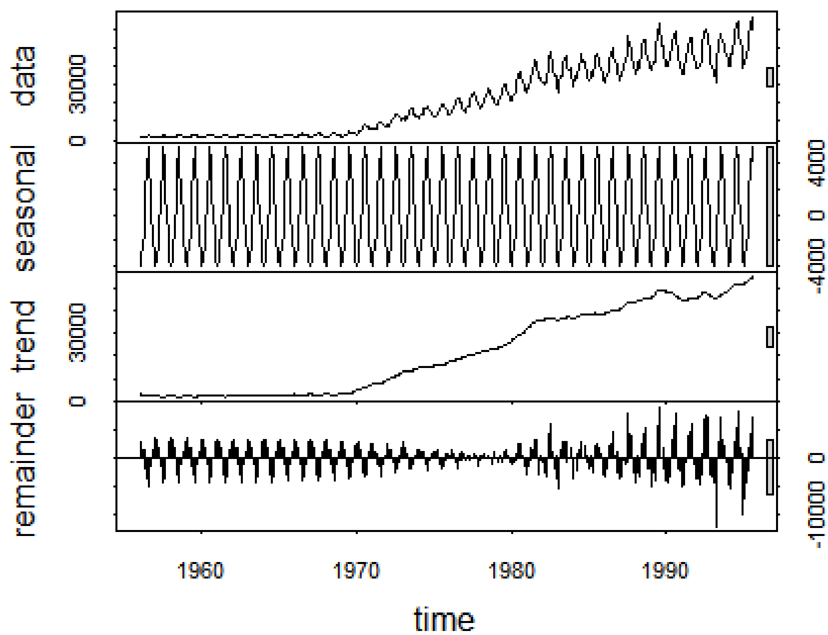

```r
boxplot(gas~cycle(gas))
```



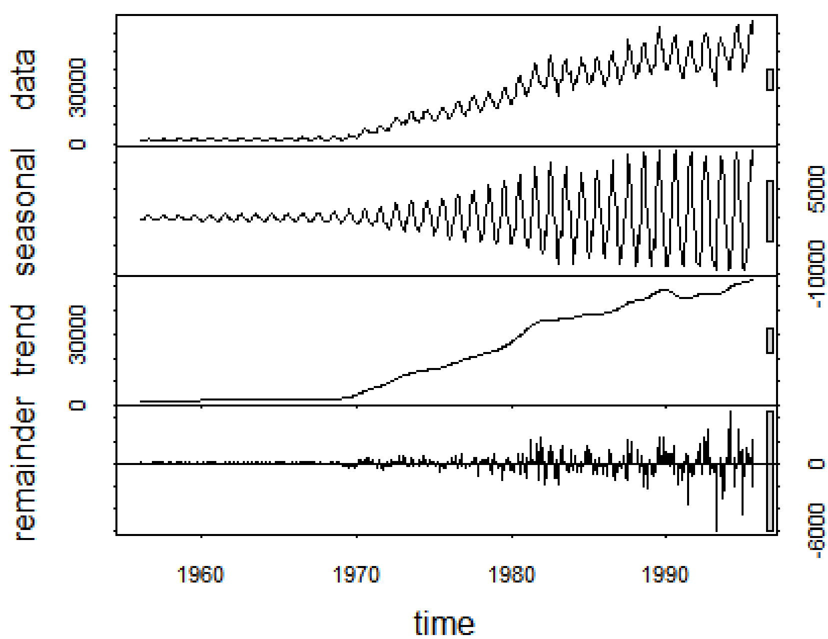

```
# Taking the subset of the data as the entire data is irrelevant

gas1<- window(gas, start=c(1985,1), end=c(1995,8))
plot(gas1)
```



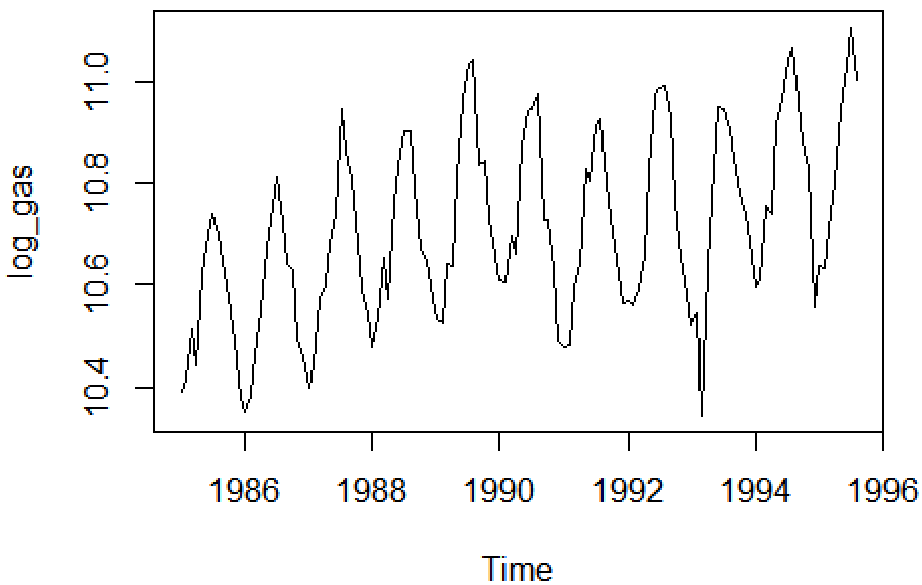

```r
# Decompose the data

gas_data_const <- stl(gas, s.window = "p")
plot(gas_data_const)
```




```r
gas_data <- stl(gas, s.window = 5)
plot(gas_data)
```

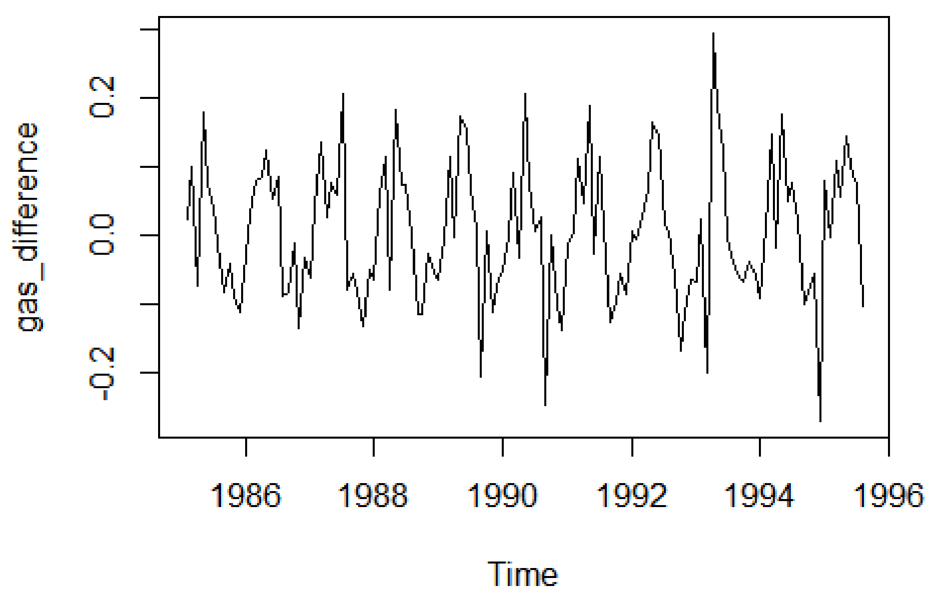


```r
# Taking log of the series to make the variance constant

log_gas <- log(gas1)
plot(log_gas)
```




```r
# Augmented Dickey Fuller Test

install.packages("tseries")

## Installing package into 'C:/Users/user/Documents/R/win-library/3.6'
## (as 'lib' is unspecified)

## package 'tseries' successfully unpacked and MD5 sums checked

## Warning: cannot remove prior installation of package 'tseries'

## Warning in file.copy(savedcopy, lib, recursive = TRUE): problem copying C:
## \Users\user\Documents\R\win-library\3.6\00LOCK\tseries\libs\x64\tseries.dl
l
## to C:\Users\user\Documents\R\win-library\3.6\tseries\libs\x64\tseries.dll:
## Permission denied

## Warning: restored 'tseries'

##
## The downloaded binary packages are in
##  C:\Users\user\AppData\Local\Temp\RtmpcflgRS\downloaded_packages
```

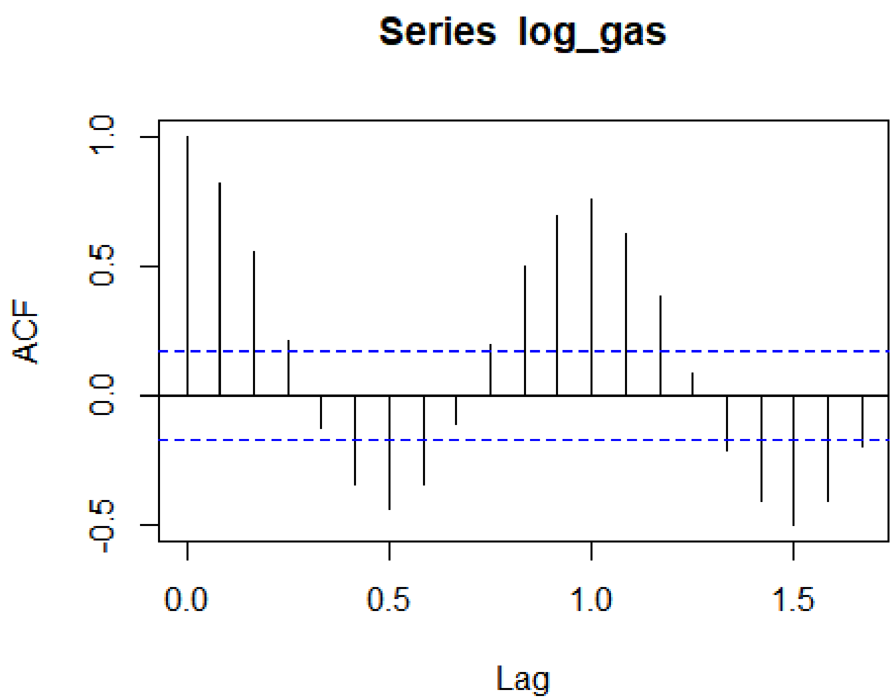
```r
library(tseries)

gas_augdf <- adf.test(log_gas, alternative = "stationary")

## Warning in adf.test(log_gas, alternative = "stationary"): p-value smaller
## than printed p-value

gas_augdf

##
##  Augmented Dickey-Fuller Test
##
## data:  log_gas
## Dickey-Fuller = -7.3234, Lag order = 5, p-value = 0.01
## alternative hypothesis: stationary

# Differencing to address the trend in the series

gas_difference <- diff(log_gas, differences = 1)
plot(gas_difference)
```



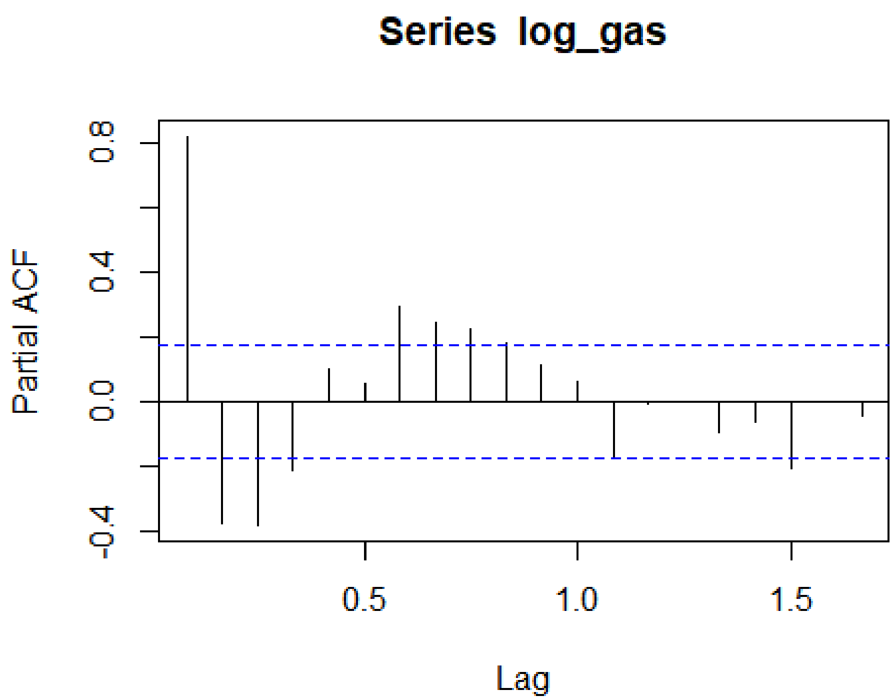

```r
gas_adf <- adf.test(gas_difference, alternative = "stationary")

## Warning in adf.test(gas_difference, alternative = "stationary"): p-value
## smaller than printed p-value

gas_adf
```

```
##
##  Augmented Dickey-Fuller Test
##
## data:  gas_difference
## Dickey-Fuller = -10.319, Lag order = 5, p-value = 0.01
## alternative hypothesis: stationary
```

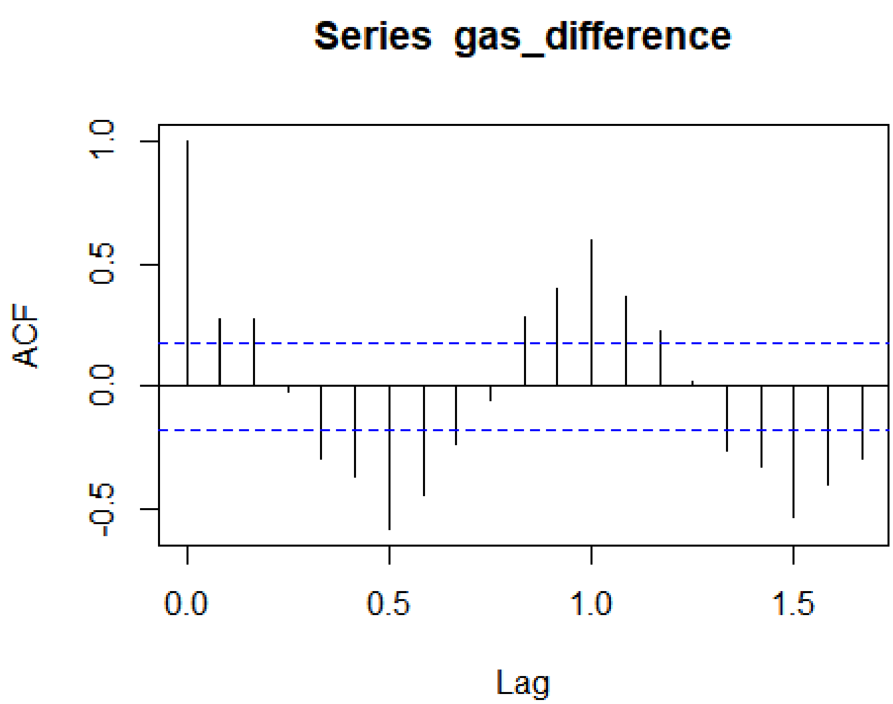
```r
# Now the series is stationary with p-value = 0.01

# ACF and PACF plots

acf(log_gas, lag.max = 20)
```



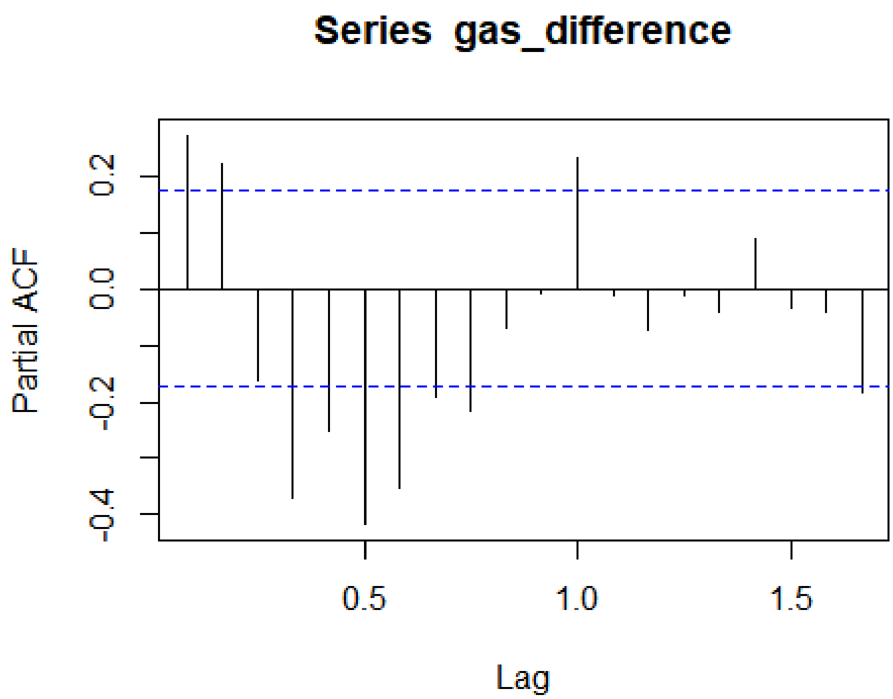

```r
pacf(log_gas, lag.max = 20)
```



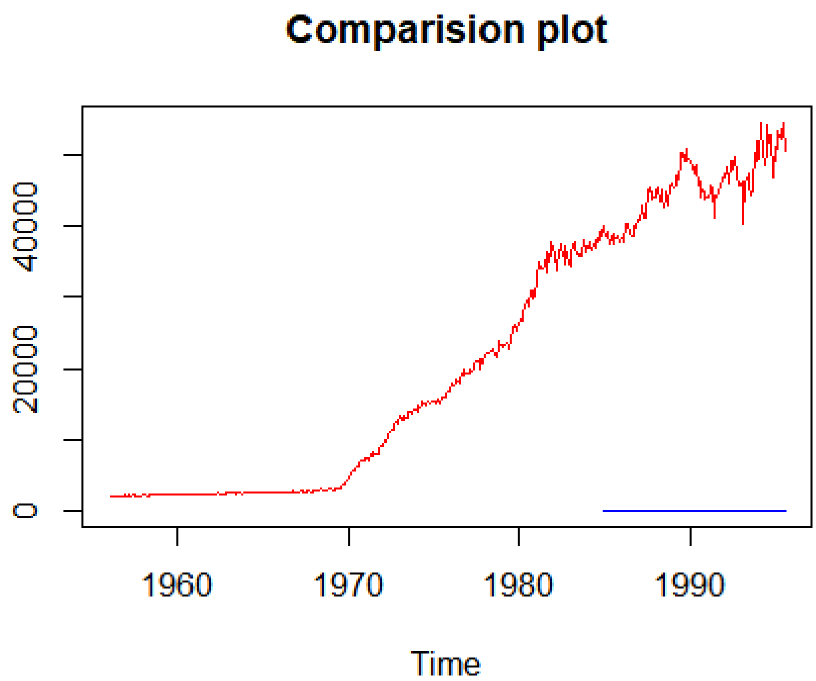

```
# ACF and PACF plots on differeced series

acf(gas_difference, lag.max = 20)
```



```r
pacf(gas_difference, lag.max = 20)
```




```r
# Deseasonalise the data

deseason_gas <- (gas_data$time.series[,2] + gas_data$time.series[,3])
plot <- ts.plot(deseason_gas, log_gas, col = c("red", "blue"), main = "Compar
ision plot")
```




```
# Dividing the series into train and test

gas_train = window(log_gas, start=c(1985,1), end=c(1992,12))
gas_test= window(log_gas, start=c(1993,1), end=c(1995,8))

# ARIMA Model on train data (checking with various SAR and SMA components)

arima_train <- arima(gas_train, order = c(2,1,2), seasonal = c(0,1,0))
arima_train

##
## Call:
## arima(x = gas_train, order = c(2, 1, 2), seasonal = c(0, 1, 0))
##
## Coefficients:
##           ar1      ar2     ma1      ma2
##       -0.9841  -0.1157  0.5660  -0.4340
## s.e.   0.2071   0.2021  0.1945   0.1918
##
## sigma^2 estimated as 0.002916:  log likelihood = 123.36,  aic = -236.72

Box.test(arima_train$residuals, lag = 20, type = "Ljung-Box")

##
##  Box-Ljung test
##
```

```
## data:  arima_train$residuals
## X-squared = 26.959, df = 20, p-value = 0.1364

arima_train <- arima(gas_train, order = c(2,1,2), seasonal = c(1,1,0))

## Warning in log(s2): NaNs produced

## Warning in log(s2): NaNs produced

arima_train

##
## Call:
## arima(x = gas_train, order = c(2, 1, 2), seasonal = c(1, 1, 0))
##
## Coefficients:
##           ar1      ar2     ma1      ma2     sar1
##       -0.9903  -0.1232  0.4984  -0.5016  -0.3461
## s.e.   0.1795   0.1767  0.1644   0.1608   0.1125
##
## sigma^2 estimated as 0.002567:  log likelihood = 127.6,  aic = -243.2

Box.test(arima_train$residuals, lag = 20, type = "Ljung-Box")

##
##  Box-Ljung test
##
## data:  arima_train$residuals
## X-squared = 25.856, df = 20, p-value = 0.1706

arima_train <- arima(gas_train, order = c(2,1,2), seasonal = c(0,1,1))
arima_train

##
## Call:
## arima(x = gas_train, order = c(2, 1, 2), seasonal = c(0, 1, 1))
##
## Coefficients:
##           ar1      ar2     ma1      ma2     sma1
##       -0.9620  -0.1098  0.4791  -0.5209  -0.7750
## s.e.   0.1786   0.1770  0.2306   0.1795   0.1746
##
## sigma^2 estimated as 0.001966:  log likelihood = 133.38,  aic = -254.76

Box.test(arima_train$residuals, lag = 20, type = "Ljung-Box")

##
##  Box-Ljung test
##
## data:  arima_train$residuals
## X-squared = 18.608, df = 20, p-value = 0.5474
```

```r
arima_train <- arima(gas_train, order = c(2,1,2), seasonal = c(1,1,1))
arima_train

##
## Call:
## arima(x = gas_train, order = c(2, 1, 2), seasonal = c(1, 1, 1))
##
## Coefficients:
##           ar1      ar2     ma1      ma2    sar1     sma1
##       -0.9487  -0.0967  0.4944  -0.5056  0.1829  -0.9992
## s.e.   0.1884   0.1866  0.2037   0.1783  0.1372   0.4041
##
## sigma^2 estimated as 0.001711:  log likelihood = 134.06,  aic = -254.13

Box.test(arima_train$residuals, lag = 20, type = "Ljung-Box")

##
##  Box-Ljung test
##
## data:  arima_train$residuals
## X-squared = 16.717, df = 20, p-value = 0.6713

arima_train <- arima(gas_train, order = c(2,1,2), seasonal = c(2,1,0))
arima_train

##
## Call:
## arima(x = gas_train, order = c(2, 1, 2), seasonal = c(2, 1, 0))
##
## Coefficients:
##           ar1      ar2     ma1      ma2     sar1     sar2
##       -0.9813  -0.1467  0.5185  -0.4458  -0.4711  -0.3808
## s.e.   0.1841   0.1878  0.1780   0.1833   0.1193   0.1299
##
## sigma^2 estimated as 0.002239:  log likelihood = 131.79,  aic = -249.57

Box.test(arima_train$residuals, lag = 20, type = "Ljung-Box")

##
##  Box-Ljung test
##
## data:  arima_train$residuals
## X-squared = 22.04, df = 20, p-value = 0.3383

arima_train <- arima(gas_train, order = c(2,1,2), seasonal = c(2,1,1))
arima_train

##
## Call:
## arima(x = gas_train, order = c(2, 1, 2), seasonal = c(2, 1, 1))
##
## Coefficients:
```

```
##           ar1      ar2     ma1      ma2    sar1     sar2     sma1
##       -0.9537  -0.1109  0.5019  -0.4691  0.0972  -0.1582  -0.7743
## s.e.   0.1974   0.2131  0.2349   0.2272  0.2745   0.2159   0.3828
##
## sigma^2 estimated as 0.001921:  log likelihood = 134.45,  aic = -252.9
```

```
Box.test(arima_train$residuals, lag = 20, type = "Ljung-Box")
```

```
##
##  Box-Ljung test
##
## data:  arima_train$residuals
## X-squared = 15.199, df = 20, p-value = 0.7649
```

```
arima_train <- arima(gas_train, order = c(2,1,2), seasonal = c(2,1,2))
arima_train
```

```
##
## Call:
## arima(x = gas_train, order = c(2, 1, 2), seasonal = c(2, 1, 2))
##
## Coefficients:
```

```
## Warning in sqrt(diag(x$var.coef)): NaNs produced
```

```
##           ar1      ar2     ma1      ma2    sar1     sar2     sma1    sma2
##       -0.9588  -0.1183  0.4924  -0.4688  0.8626  -0.4122  -1.6392  0.9231
## s.e.      NaN      NaN     NaN      NaN  0.1295      NaN      NaN     NaN
##
## sigma^2 estimated as 0.001694:  log likelihood = 134.7,  aic = -251.4
```

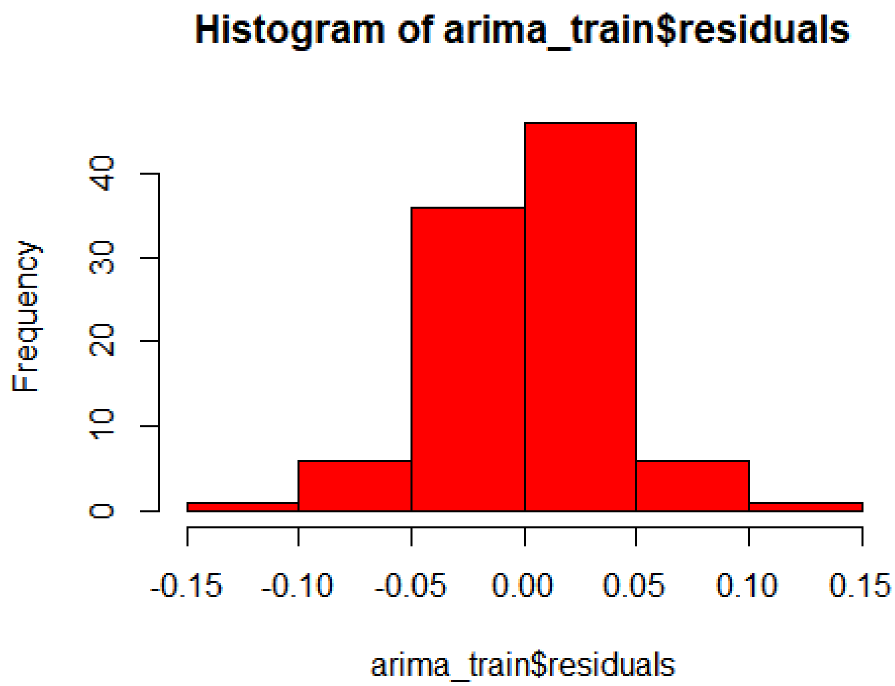
```
Box.test(arima_train$residuals, lag = 20, type = "Ljung-Box")
```

```
##
##  Box-Ljung test
##
## data:  arima_train$residuals
## X-squared = 14.253, df = 20, p-value = 0.8174
```

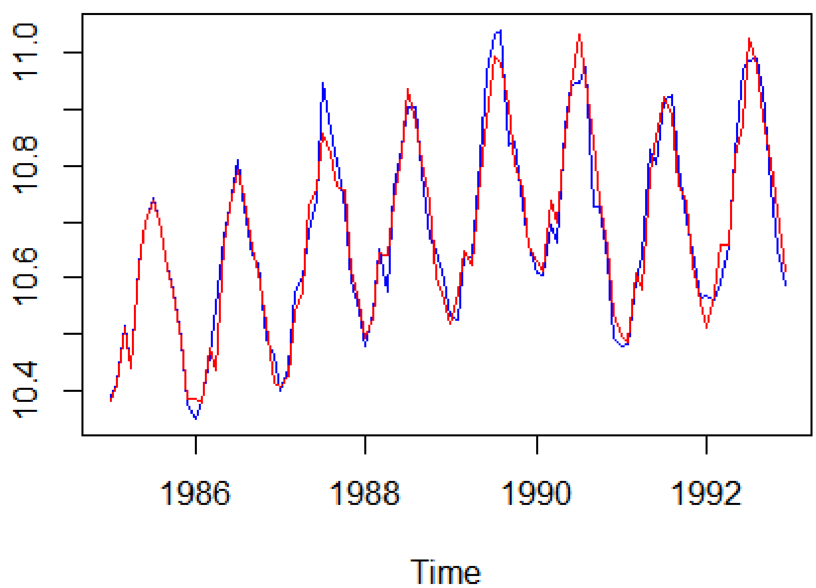
```
arima_train <- arima(gas_train, order = c(2,1,2), seasonal = c(0,1,2))
arima_train
```

```
##
## Call:
## arima(x = gas_train, order = c(2, 1, 2), seasonal = c(0, 1, 2))
##
## Coefficients:
##           ar1      ar2     ma1      ma2     sma1     sma2
##       -0.9471  -0.0963  0.5047  -0.4953  -0.7902  -0.2082
## s.e.   0.1922   0.1901  0.2142   0.1850   0.4233   0.1592
##
## sigma^2 estimated as 0.001699:  log likelihood = 134.25,  aic = -254.5
```

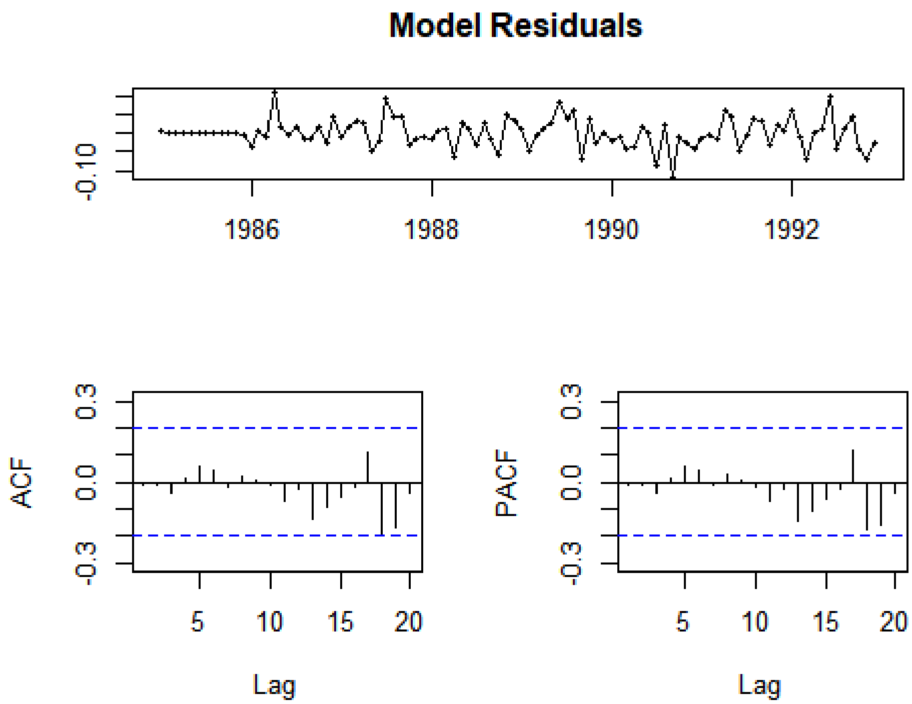
```r
Box.test(arima_train$residuals, lag = 20, type = "Ljung-Box")

##
##  Box-Ljung test
##
## data:  arima_train$residuals
## X-squared = 16.135, df = 20, p-value = 0.7082

arima_train <- arima(gas_train, order = c(2,1,2), seasonal = c(1,1,2))
arima_train

##
## Call:
## arima(x = gas_train, order = c(2, 1, 2), seasonal = c(1, 1, 2))
##
## Coefficients:
##           ar1      ar2     ma1      ma2     sar1     sma1     sma2
##       -0.9495  -0.1034  0.5069  -0.4927  -0.5150  -0.2052  -0.5985
## s.e.   0.1900   0.1876  0.9752   0.5058   1.1448   1.3133   0.7082
##
## sigma^2 estimated as 0.001824:  log likelihood = 134.42,  aic = -252.83

Box.test(arima_train$residuals, lag = 20, type = "Ljung-Box")

##
##  Box-Ljung test
##
## data:  arima_train$residuals
## X-squared = 15.79, df = 20, p-value = 0.7296

# The best model obtained with SAR = 2 and SMA = 2

arima_train <- arima(gas_train, order = c(2,1,2), seasonal = c(2,1,2))
arima_train

##
## Call:
## arima(x = gas_train, order = c(2, 1, 2), seasonal = c(2, 1, 2))
##
## Coefficients:

## Warning in sqrt(diag(x$var.coef)): NaNs produced

##           ar1      ar2     ma1      ma2    sar1     sar2     sma1    sma2
##       -0.9588  -0.1183  0.4924  -0.4688  0.8626  -0.4122  -1.6392  0.9231
## s.e.      NaN      NaN     NaN      NaN  0.1295      NaN      NaN     NaN
##
## sigma^2 estimated as 0.001694:  log likelihood = 134.7,  aic = -251.4

hist(arima_train$residuals, col = "red")
```




```
arima_train_fit <- fitted(arima_train)
ts.plot(gas_train,arima_train_fit, col = c("blue", "red"))
```

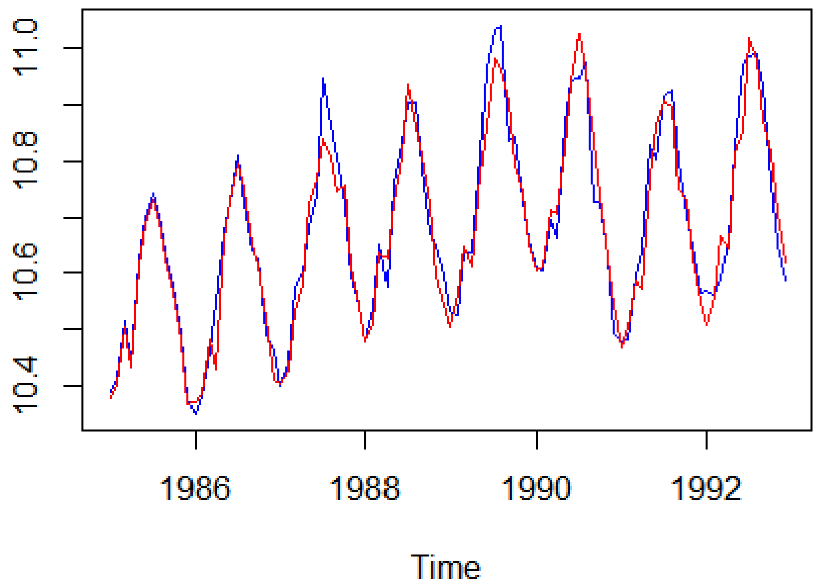


```r
tsdisplay(residuals(arima_train), lag.max = 20, main = 'Model Residuals')
```

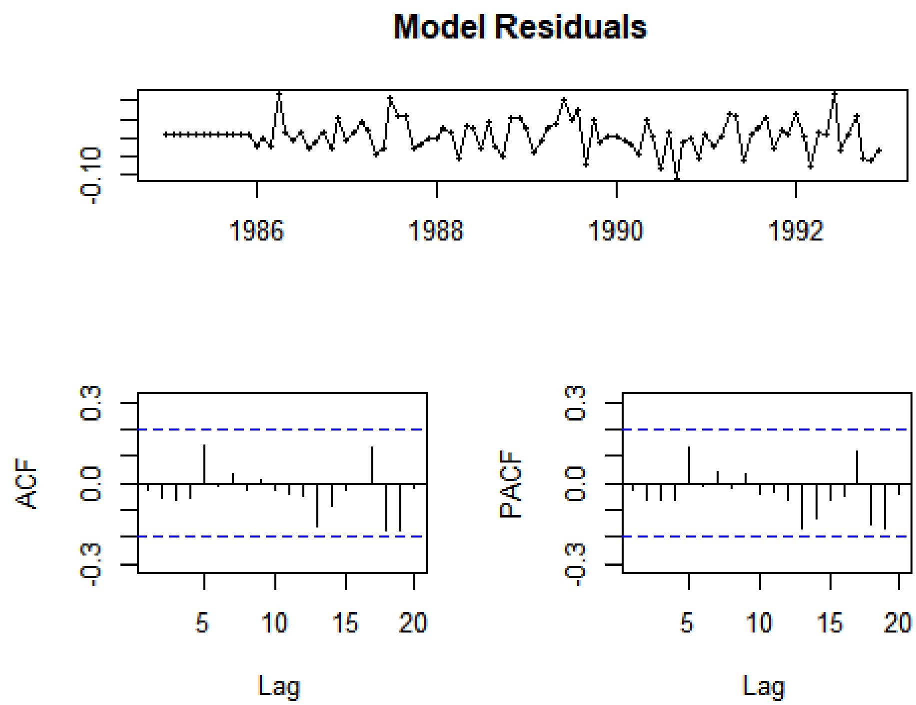



```r
# Box Test

Box.test(arima_train$residuals, lag = 20, type = "Ljung-Box")

##
##  Box-Ljung test
##
## data:  arima_train$residuals
## X-squared = 14.253, df = 20, p-value = 0.8174

# Auto ARIMA on train data

autoarima_train <- auto.arima(gas_train, seasonal = TRUE)
autoarima_train

## Series: gas_train
## ARIMA(3,0,0)(2,1,1)[12]
##
## Coefficients:
##          ar1     ar2     ar3    sar1     sar2     sma1
##       0.4810  0.1932  0.2537  0.0796  -0.2110  -0.7060
## s.e.  0.1092  0.1165  0.1064  0.2252   0.1707   0.2894
##
## sigma^2 estimated as 0.002198:  log likelihood=135.75
## AIC=-257.51   AICc=-256.03   BIC=-240.49
```

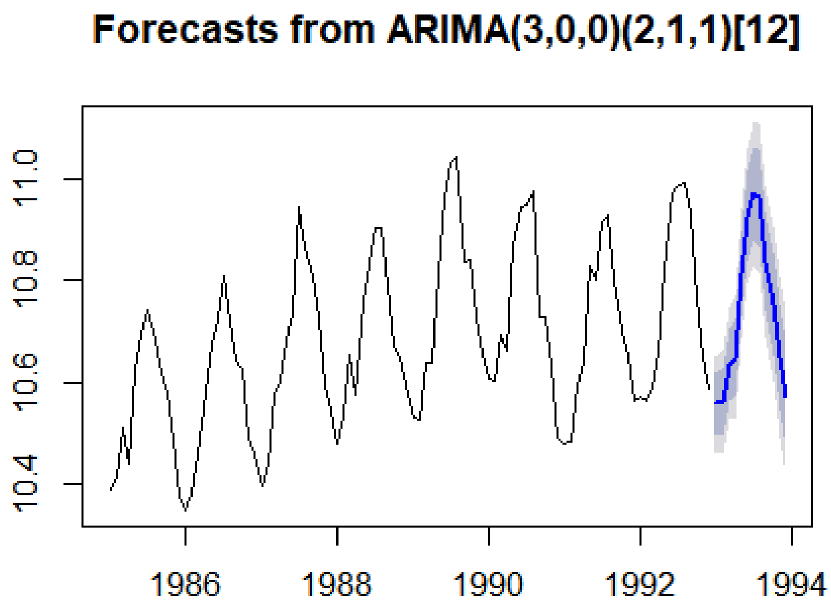
```r
autoarima_train_fit <- fitted(autoarima_train)
ts.plot(gas_train,autoarima_train_fit, col = c("blue", "red"))
```




```r
tsdisplay(residuals(autoarima_train), lag.max = 20, main = 'Model Residuals')
```




```r
# Box Test on auto arima

Box.test(autoarima_train$residuals, lag = 20, type = "Ljung-Box")

##
##  Box-Ljung test
##
## data:  autoarima_train$residuals
## X-squared = 17.196, df = 20, p-value = 0.6402

# Forecast on train data

gas_train_autofc <- forecast(autoarima_train, h = 12)
plot(gas_train_autofc)
```




```
# Accuracy

accuracy <- accuracy(gas_train_autofc, gas_test)
accuracy

##                        ME       RMSE        MAE         MPE      MAPE
## Training set  0.007063177 0.04226108 0.03211989  0.06486765 0.2992514
## Test set     -0.013297032 0.09760529 0.05938956 -0.13259204 0.5614692
##                   MASE       ACF1 Theil's U
## Training set 0.4702325 -0.0250175        NA
## Test set     0.8694582  0.1636908 0.7575444
```