**Mini Project – time series forecasting of GAS data preloaded in forecast package**

**24th - October – 2019**

**Submitted**

By: Shilpa pandey

**Table of Content**

**1. Project Objectives**

**2. Steps and approach**

**3. time series object for GAS dataset in R**

**4. Plot the time series data**

**5. General observation of dataset**

**6. Components of the time series are present in this dataset**

**7. periodicity of dataset**

**8. Is the time series Stationary**

**9. visual and ADF test for checking if time series is stationary**

**10. the null and alternate hypothesis for the stationarity test**

**11. DE seasonalize the dataset**

**12. Manual ARIMA Model with all steps**

**13. Auto ARIMA Model**

**14. ARIMA Model to forecast for next 12 periods**

**15. accuracy of the model**

**16. Code for reference**

**Project Objectives :**

Download the **Forecast** package in R. The package contains methods and tools for displaying and analyzing univariate time series forecasts including exponential smoothing via state space models and automatic ARIMA modelling. Explore the **gas** (Australian monthly gas production) dataset in Forecast package.

**Steps and approach**

Loaded the data and found there is only one column in the dataset

str(gas)

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

> head(gas)

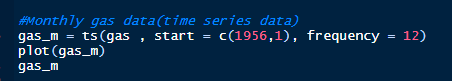
Jan Feb Mar Apr May Jun

1956 1709 1646 1794 1878 2173 2321

**Time series object for GAS dataset in R**

A time series is a sequence of measurements on the same variable collected over time.

The measurements are made at regular time intervals. This is monthly time series



**Plot the time series data** Machine generated alternative text:
Plot Zoom 
1960 
1970 
1980 
Time 
1990 

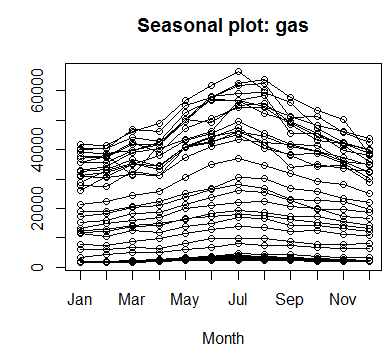
**General observation of dataset :**

How month behave every year, more gas data in July and least in Jan and December. With each year it is increasing, after being constant for sometimes.

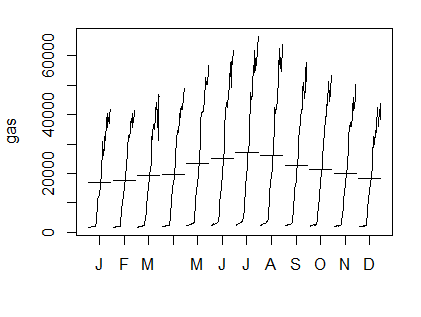
Also found that, initially till 1968 the data has seasonality but post that the dataset has both trend and seasonality.

seasonplot(gas)

monthplot(gas)



Every month of every year, gas volume is increasing



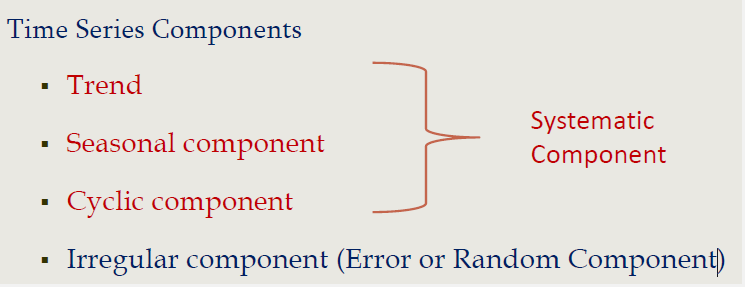
**Components of the time series are present in this dataset**

This dataset has trend and seasonality

Is there a Trend? Initially not but after a while it has Increasing trend till end

Is there a Seasonality? Yes

A TS can be split into several components, each representing one of the underlying categories of patterns,

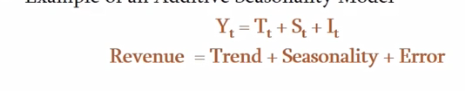


Trend: Long term movement of a series: either increasing or decreasing

Seasonality: Seasonality is the relative increase or decrease of data every period (quarter or month or week) compared to the yearly average.

**periodicity of dataset :** Inconsistent seasonality and it is not periodic

Decompose the dataset using stl command and it will break into trend, seasonality and error



* The Additive model(periodic) is best used when the seasonal trend is of the same magnitude throughout the data set
* Multiplicative Model(non periodic) is preferred when the magnitude of seasonality changes as time increases.
* Visualization of Seasonality Helps to see the nature of seasonality : periodic or not

Scale of trend is small as compared to seasonality hence seasonality changes

Periodic seasonality : here the scale of seasonality is very large as compared to trend

stl(x = gas\_m, s.window = "periodic")

Components

seasonal trend remainder

Jan 1956 -4083.7994 3000.295 2792.50473

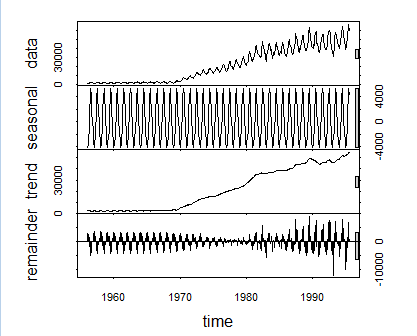
Feb 1956 -3715.9678 2792.712 2569.25583

Mar 1956 -2165.1863 2585.129 1374.05703

Apr 1956 -1809.3076 2461.267 1226.04094

May 1956 1713.8461 2337.404 -1878.25008

Jun 1956 3591.8664 2261.258 -3532.12392



Non-periodic seasonality: here the scale of seasonality is comparatively small and scale for trend has also increased and hence we will select non-periodic seasonality

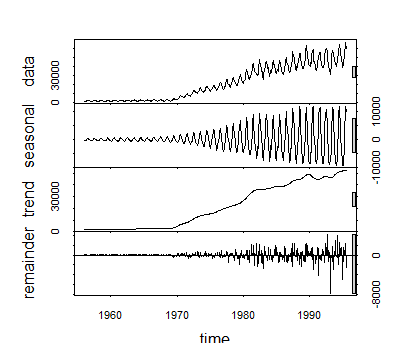
stl(x = gas\_m, s.window = 7)

Components

seasonal trend remainder

Jan 1956 -362.862450 2018.888 5.297466e+01

Feb 1956 -421.728504 2023.650 4.407834e+01



**Is the time series Stationary? :** No it is not stationary as it has trend and seasonality

Stationary time series: one whose properties do not depend on the time at which the series is observed.

* Time series with trends, or with seasonality are not stationary
* Trend and seasonality will affect the value of the time series at different times.

**Visual and ADF test for checking if time series is stationary**

Augmented Dickey-Fuller Test : Tests whether a time series is NON-STATIONARY

P values is more than 0.05 and hence the alternative hypothesis rejected and hence data is not stationary

* Augmented Dickey-Fuller Test
* data: gas\_m
* Dickey-Fuller = -2.7131, Lag order = 7, p-value = 0.2764
* alternative hypothesis: stationary

**The null and alternate hypothesis for the stationarity test**

* Null hypothesis H0: Time series non-stationary
* Alternative hypothesis Ha: Time series is stationary
* Rejection of null hypothesis implies that the series is stationary
* Augmented Dickey-Fuller Test
* data: gas\_m
* Dickey-Fuller = -2.7131, Lag order = 7, p-value = 0.2764
* alternative hypothesis: stationary

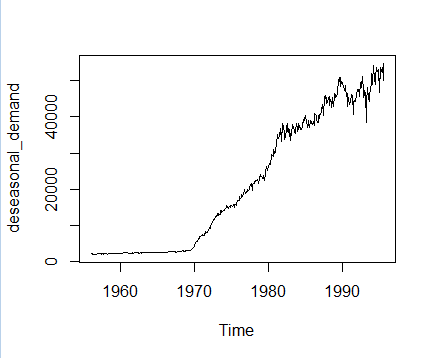
**De-seasonalize the series :**

deseasonal\_demand =seasadj(gas\_incon)

plot(deseasonal\_demand)

deseasonal\_demand

It will help to find how much trend is affected without seasonality



> deseasonal\_demand

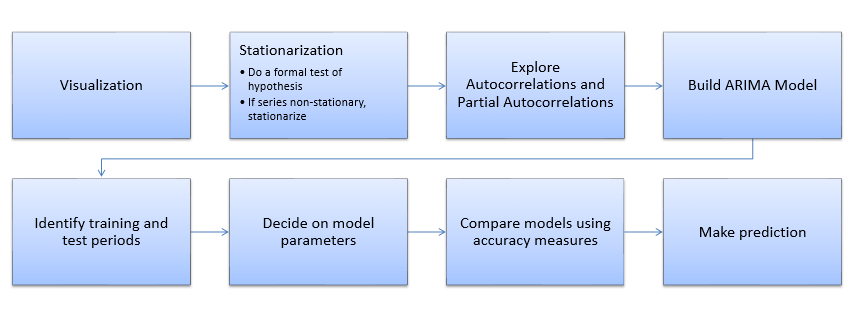
Jan Feb Mar Apr May Jun Jul Aug Sep Oct Nov Dec

1956 2071.862 2067.729 2047.950 2040.865 1990.354 2008.200 1944.032 2022.043 2018.048 2064.933 2126.943 2090.131

1957 2121.639 2114.350 2177.396 2106.258 2124.205 1956.682 2107.718 2046.762 2111.975 2108.576 2107.454 2147.752

**Manual ARIMA Model with all steps**

Steps are shown below:



Steps for Analysis

1.Visualization: already did and found that the data has trend and seasonality

2.Stationarization

Do a formal test of hypothesis and found the time series is not stationary and if series non-stationary then we need to stationarize. Stationary Series It is possible to make a non-stationary series stationary by taking differences between consecutive observations

Use diff command (order of 1) --- hence the value od d=1 as it stationarize the time series

The ADF test has p value less than 0.05 and hence now the data set is stationary

count\_d1 = diff(deseasonal\_demand, differences = 1)# used only on deseasonalized data, d=1

> plot(count\_d1)

> #test again if series is stationary

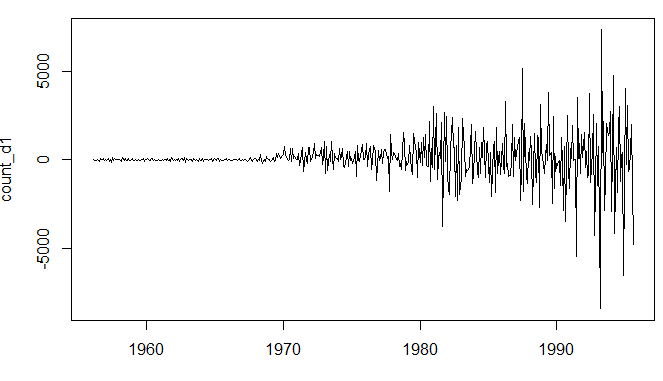
> adf.test(count\_d1, alternative = "stationary")#Differenced demand is stationary

Augmented Dickey-Fuller Test

data: count\_d1

Dickey-Fuller = -8.1524, Lag order = 7, p-value = 0.01

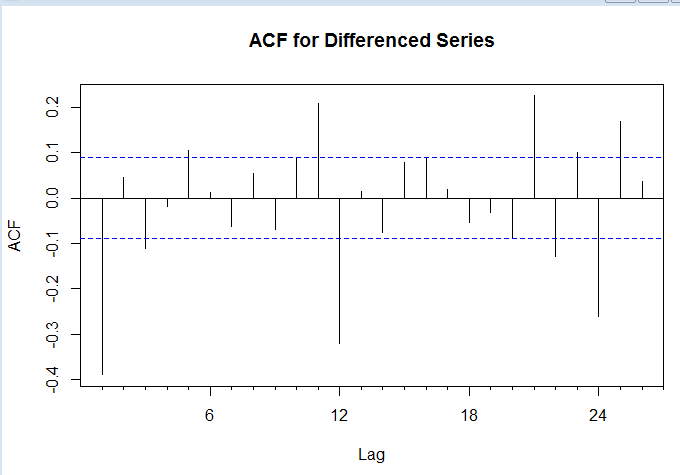
alternative hypothesis: stationary

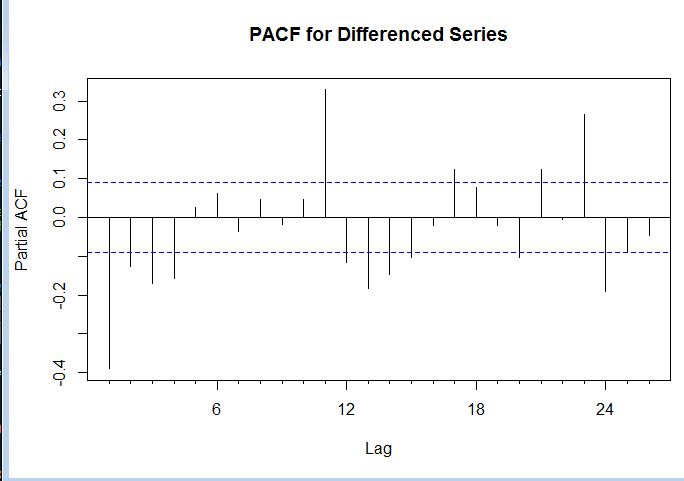


3.Explore Autocorrelations and Partial Autocorrelations

q=9 which is taken up to lag 24 (significant lags for acf)

p=14 which is taken up to lag 24(significant lags for pacf)





4. build ARIMA Model

5. Identify training and test periods: data up to 1992 into training and remaining in to test data set

gastrain = window(deseasonal\_demand, start=1956, end=c(1992,12))

gastest= window(deseasonal\_demand, start=1993, end=c(1993,12))

6. Decide on model parameters:

ARIMA(p, d, q) identifies a non-seasonal model which needs to be differenced d times to make it stationary and contains p AR terms and q MA terms

D=1, q=9, p=14

7. Compare models using accuracy measures

AIC : acrylic information criteria : which one is best we will use that (should have the lowest value)

demandARIMA2 and demandARIMA4 has lowest value

Call:

arima(x = gastrain, order = c(0, 1, 0))

sigma^2 estimated as 999302: log likelihood = -3688.57, aic = 7379.14

> demandARIMA2 = arima(gastrain, order=c(14,1,0))

> demandARIMA2# 2nd lowest aic vale

Call:

arima(x = gastrain, order = c(14, 1, 0))

Coefficients:

ar1 ar2 ar3 ar4 ar5 ar6 ar7 ar8 ar9 ar10 ar11 ar12 ar13 ar14

-0.4399 -0.1140 -0.0316 -0.0884 0.0278 0.2159 0.2230 0.0782 0.1587 0.2125 0.299 -0.2079 -0.2224 -0.0679

s.e. 0.0474 0.0513 0.0524 0.0499 0.0486 0.0481 0.0495 0.0496 0.0486 0.0491 0.050 0.0524 0.0535 0.0501

sigma^2 estimated as 660869: log likelihood = -3598.55, aic = 7227.11

> demandARIMA3 = arima(gastrain, order=c(0,1,9))

> demandARIMA3

Call:

arima(x = gastrain, order = c(0, 1, 9))

Coefficients:

ma1 ma2 ma3 ma4 ma5 ma6 ma7 ma8 ma9

-0.4823 0.0208 -0.0569 -0.0864 0.3133 0.3854 0.0883 -0.2174 0.0229

s.e. 0.0484 0.0593 0.0686 0.0658 0.0839 0.0930 0.0829 0.0748 0.0630

sigma^2 estimated as 726468: log likelihood = -3620.91, aic = 7261.82

> demandARIMA4 = arima(gastrain, order=c(14,1,9))

> demandARIMA4# lowest aic value

Call:

arima(x = gastrain, order = c(14, 1, 9))

Coefficients:

ar1 ar2 ar3 ar4 ar5 ar6 ar7 ar8 ar9 ar10 ar11 ar12 ar13 ar14

-0.1422 -0.1865 -0.2408 -0.4644 -0.4192 0.2982 -0.2215 -0.2553 -0.0240 0.1839 0.2717 -0.2729 -0.0153 0.1102

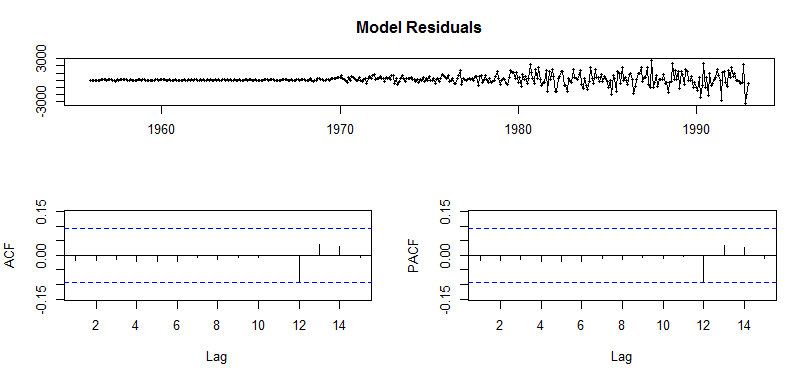
s.e. 0.2424 0.2713 0.0859 0.0704 0.0987 0.1064 0.1414 0.1638 0.1065 0.0794 0.0567 0.0685 0.1138 0.1182

ma1 ma2 ma3 ma4 ma5 ma6 ma7 ma8 ma9

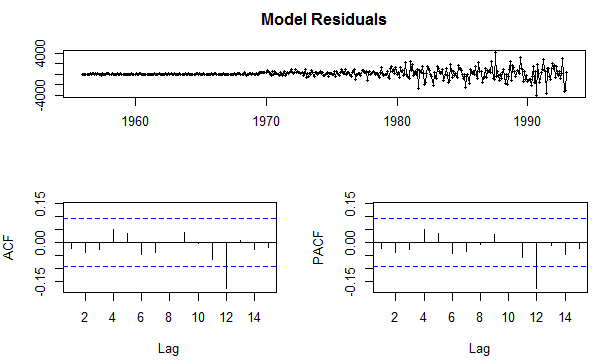
-0.3152 0.2336 0.1885 0.4773 0.4418 -0.5417 0.7542 0.2132 0.2646

s.e. 0.2444 0.3303 0.2071 0.0883 0.0945 0.0903 0.2041 0.3361 0.2494

sigma^2 estimated as 460453: log likelihood = -3530.65, aic = 7109.29



This one touching blue line : so not a good model



Since the demandARIMA2 has error crossing blue line and hence error is more significant and so ignoring that and accepting demandARIMA4

ARIMA (14,1,9)

AIC: 7109.2997109.109.29

**Auto ARIMA Model**

Series: gastrain

ARIMA(0,1,1) with drift

Coefficients:

ma1 drift

-0.4381 101.9776

s.e. 0.0443 24.4818

sigma^2 estimated as 841880: log likelihood=-3649.71

AIC=7305.41 AICc=7305.47 BIC=7317.69

AIC value = 7305.47

ARIMA(0,1,1)

**ARIMA Model to forecast for next 12 periods**

Residuals should be insignificant and independent for forecasting

Ljung box test

H0: Residuals are independent

Ha: Residuals are not independent

> Box.test(fit$residuals)

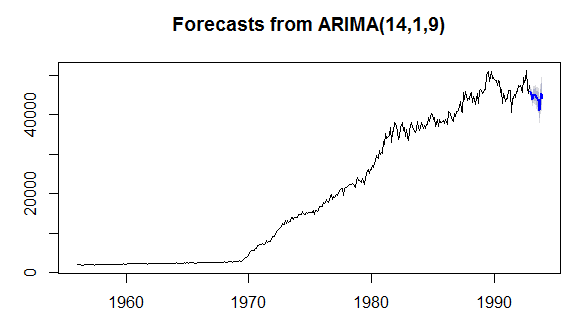
Box-Pierce test

data: fit$residuals

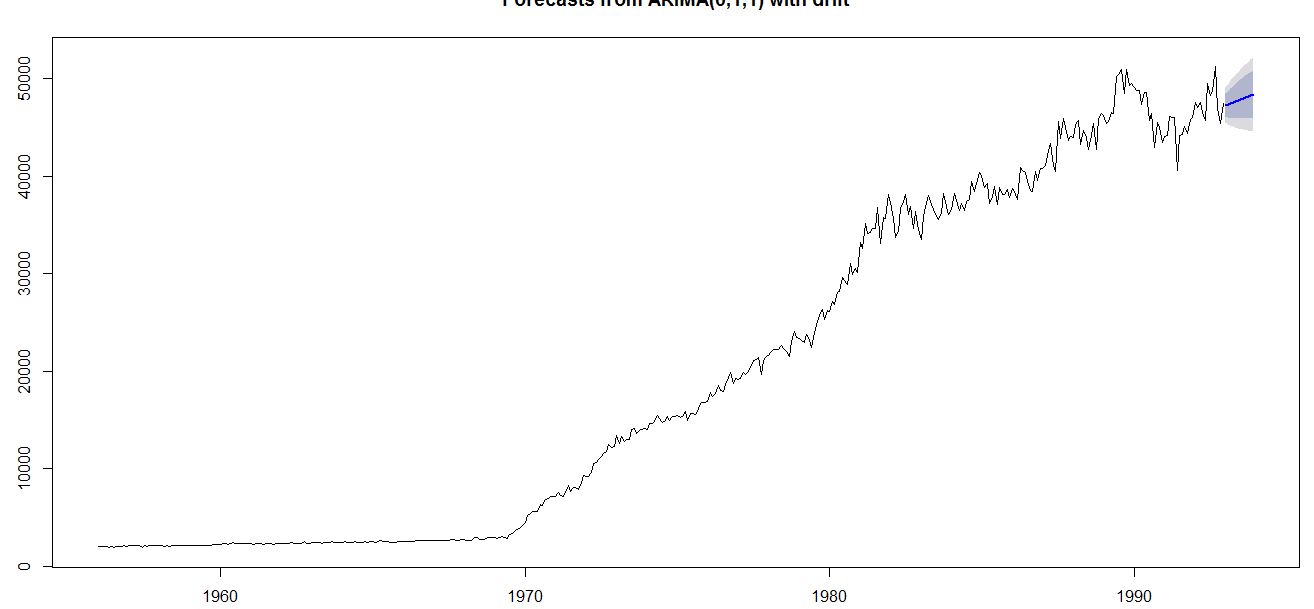
X-squared = 0.08229, df = 1, p-value = 0.7742

P value is more than 0.05 percent, hence rejecting this test and hence residuals are independent.

**Forecast of manual Arima model**



**Forecast of auto arima model:**



**Accuracy of the model**

MAPE (error values) value for training and test data for manual arima is 2.13 and 6.03

MAPE value for training and test data for manual arima is 4.39 and 5.32

|  |
| --- |
| > accuracy(f7, gastest)  ME RMSE MAE MPE MAPE MASE ACF1 Theil's U  Training set 82.58943 677.802 403.1309 0.5888861 2.134369 0.2357057 -0.01730936 NA  Test set 1864.67157 3645.732 2803.0349 3.5926965 6.038963 1.6389004 0.26322309 0.933764  > f8=forecast(fit)  > accuracy(f8, gastest)  ME RMSE MAE MPE MAPE MASE ACF1 Theil's U  Training set -0.2750405 914.435 559.5491 -2.193468 4.399203 0.3271615 0.01361389 NA  Test set -1506.9060913 3268.666 2328.2746 -3.720113 5.329286 1.3613139 0.11295123 0.84321 |
|  |
| |  | | --- | | > | |

For the training , we have less error in training data of manual arima

Since the error in auto arima is 5.32 which is less than manual arima and hence auto arima is better model.

**16. Code for reference**

library('ggplot2')

library('forecast')

library('tseries')

str(gas)

head(gas)

data(gas)

write.csv(gas, "gas\_data.csv",row.names = FALSE)

#plot gas

plot(gas)

seasonplot(gas)

monthplot(gas)

#Quarterly gas data

gas\_q = ts(gas, start = c(1956,1), frequency = 4)

plot(gas\_q)

gas\_q

#Monthly gas data(time series data)

gas\_m = ts(gas , start = c(1956,1), frequency = 12)

plot(gas\_m)

gas\_m

#Decompose the data

gas\_const = stl(gas\_m, s.window = "periodic")#constant seasonality

plot(gas\_const)

gas\_const

gas\_incon<-stl(gas\_m, s.window=7) #seasonality changes

plot(gas\_incon)

gas\_incon

# to deseasonalize

deseasonal\_demand =seasadj(gas\_incon)

plot(deseasonal\_demand)

deseasonal\_demand

ts.plot(deseasonal\_demand, gas, col=c("red", "blue"), main="Comparison of gas and Deseasonalized gas")

#Check for stationarity

#Dickey-Fuller Test

adf.test(gas\_m, alternative = "stationary")

#Differencing the time series data

count\_d1 = diff(deseasonal\_demand, differences = 1)# used only on deseasonalized data, d=1

plot(count\_d1)

#test again if series is stationary

adf.test(count\_d1, alternative = "stationary")#Differenced demand is stationary

#acf and pacf for dif time series

Acf(count\_d1, main='ACF for Differenced Series')#q=9 which is taken upto lag 24

Pacf(count\_d1, main='PACF for Differenced Series')#p=14 which is taken upto lag 24

acf(gas\_m, lag.max = 24)

pacf(gas\_m, lag.max = 24)

#From the ACF plot, there is a cut off after lag 0. This implies that q=0. PACF cuts off after lag 1. Hence p=1.

#Splitting into training and test sets

str(gas)

gastrain = window(deseasonal\_demand, start=1956, end=c(1992,12))

gastest= window(deseasonal\_demand, start=1993, end=c(1993,12))

# (ACF, diff, pacf) or (q,d,p)

demandARIMA1 = arima(gastrain, order=c(0,1,0))

demandARIMA1

demandARIMA2 = arima(gastrain, order=c(14,1,0))

demandARIMA2# 2nd lowest aic vale

demandARIMA3 = arima(gastrain, order=c(0,1,9))

demandARIMA3

demandARIMA4 = arima(gastrain, order=c(14,1,9))

demandARIMA4# lowest aic value

tsdisplay(residuals(demandARIMA4), lag.max=15, main='Model Residuals')# better model none of the errors are significat and touching the ble line

tsdisplay(residuals(demandARIMA2), lag.max=15, main='Model Residuals')

#There are no significant autocorrelations present. If the model is not correctly specified, that will usually be reflected in residuals in the form of trends, skeweness, or any other patterns not captured by the model. Ideally, residuals should look like white noise, meaning they are normally distributed. Residuals plots show a smaller error range, more or less centered around 0.

# this was the manual Arima method

#Fitting with Auto ARIMA

fit<-auto.arima(gastrain, seasonal=FALSE)

fit

tsdisplay(residuals(fit), lag.max=45, main='Auto ARIMA Model Residuals')

#Auto ARIMA also fits the same p and q parameters for the model, but has a slightly lower AIC.

#Ljung box test

#H0: Residuals are independent

#Ha: Residuals are not independent

library(stats)

Box.test(demandARIMA$residuals)

Box.test(fit$residuals)

#Forecasting with the ARIMA model

fcast <- forecast(demandARIMA4, h=12)

fcast1 <- forecast(fit, h=12)

plot(fcast)

plot(fcast1)

fit1<-auto.arima(demandTS, seasonal=FALSE)

fcast2=forecast(fit1, h=12)

plot(fcast2)

#Accuracy of the forecast

f7=forecast(demandARIMA4)

accuracy(f7, gastest)

f8=forecast(fit)

accuracy(f8, gastest)