Times Series Forecasting

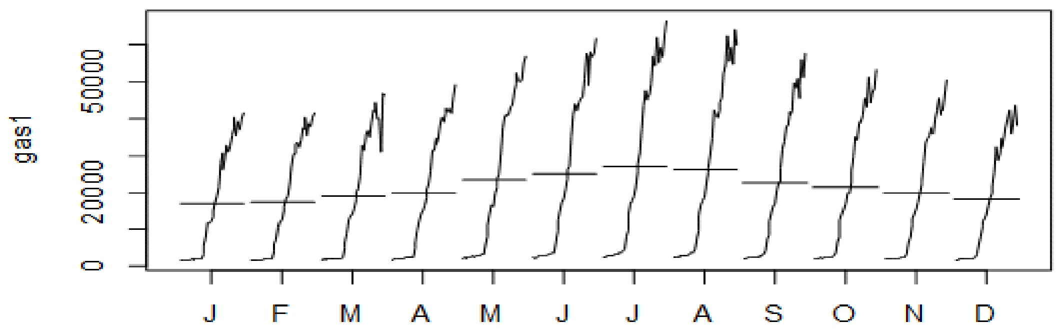
We are supposed to use the Gas dataset inside the forecast package. We read the Gas data as a time series object. Now, it's time to do some exploratory analysis with the data.

We find that the data is a monthly data (frequency = 12).

We plot the month plot, season plot & the overall plot with the time series data.

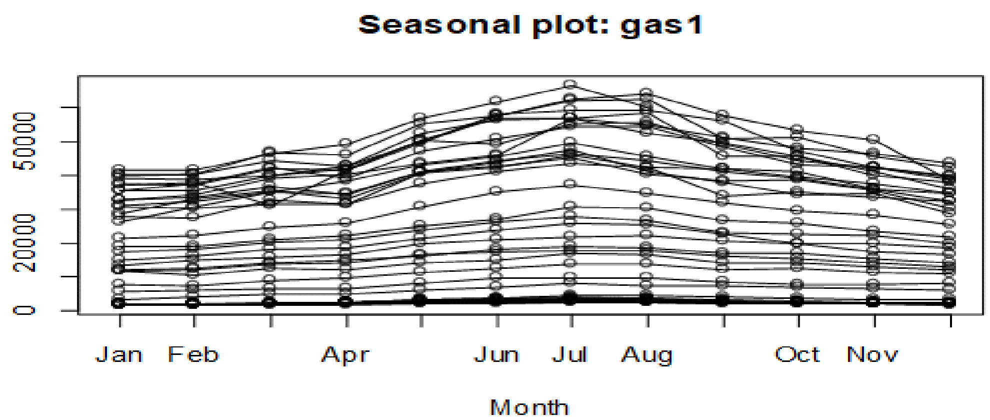
Month Plot

The month plot shows that the gas production has increased for every month of the year. We can also see a trend.



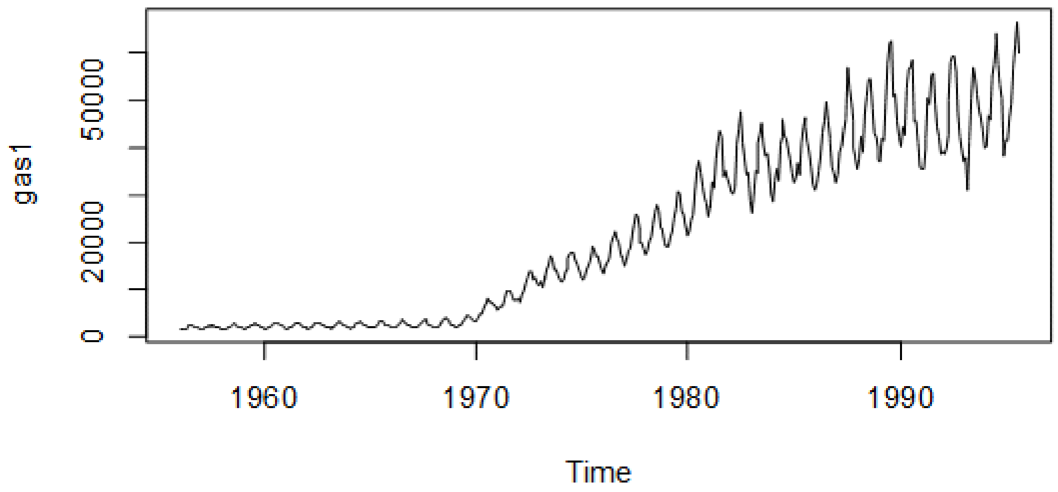
Season Plot

The season plot shows a strong presence of seasonality.



Time Series Plot

We have made a plot of the time series data. We can observe that till 1970, the graph is level and after that we can see a trend starting from 1970 to 1996. We can see a trend & seasonality.



We found that the periodicity is monthly.

Test & Train Data

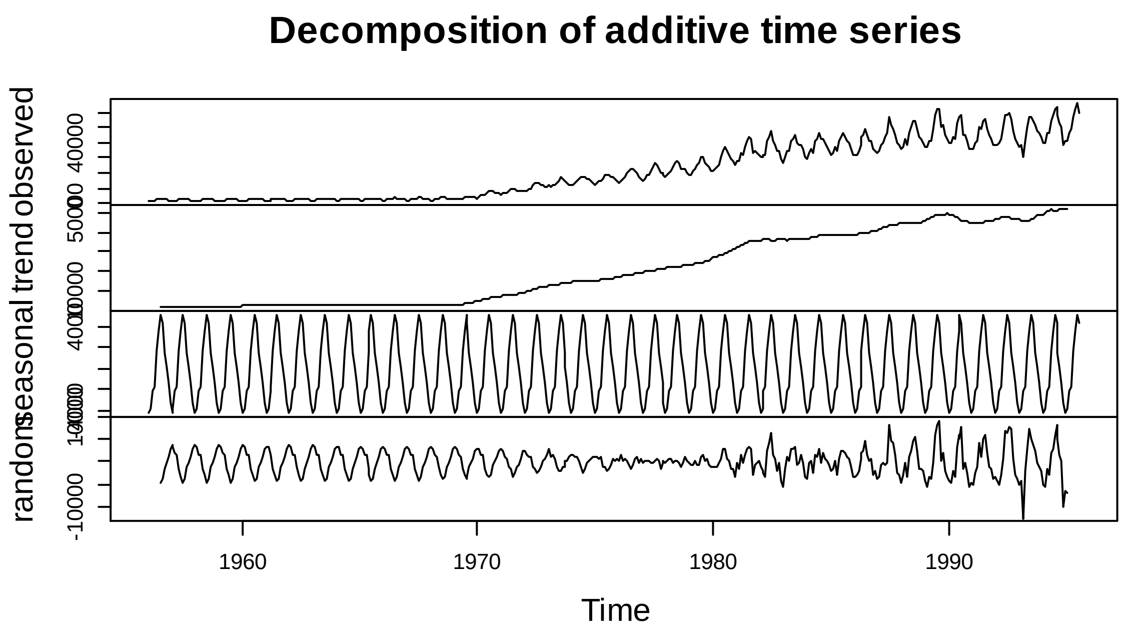
We divide the time series data into test & train data. Below is the code snippet for the same:

```
gas.train <- window(gas1, end = c(1994,12))

str(gas.train)

gas.test<- window(gas1, start = c(1995,1))

str(gas.test)
```

Time Series Decomposition

We have done additive decomposition of the Time series in order to understand the components better:

```
gasComp = decompose(gas1) #additive

plot(gasComp)
```



We can see, that our gas time series data has:

- Level
- Trend
- Seasonality
- Random Component

<u>Holt-Winters' Model – Forecasting</u>

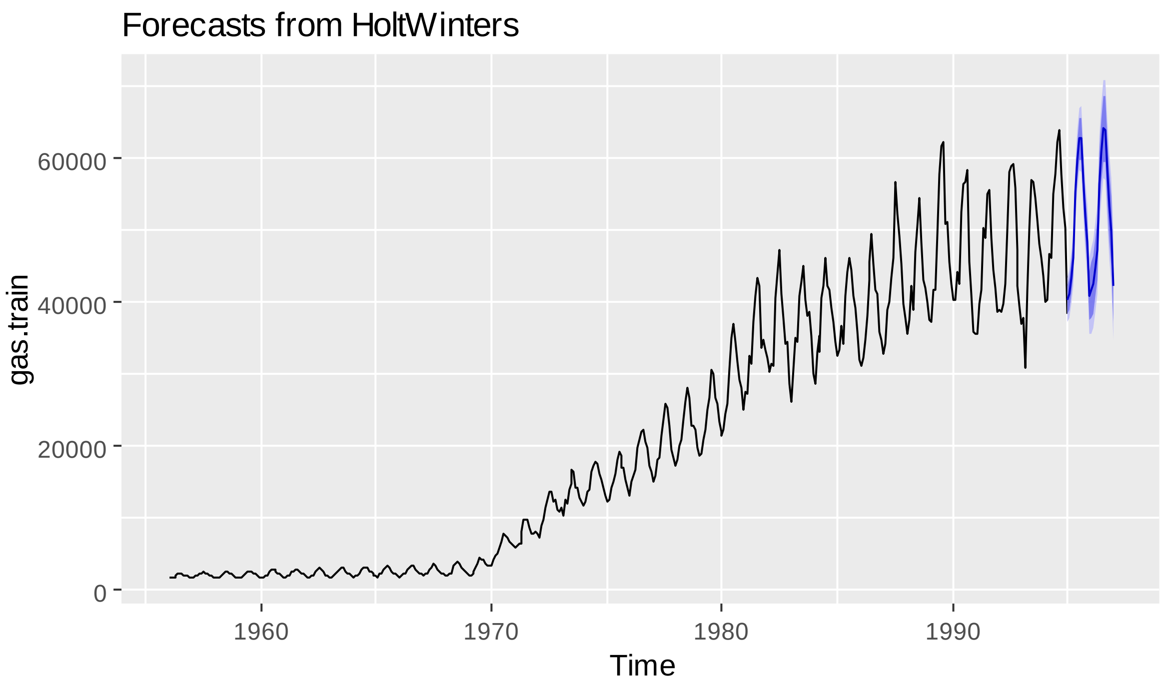
We can see from decomposition that our time series data contains level, trend & seasonality. Hence we are applying Holt Winter's model for forecasting. It has three parameters: alpha, beta & gamma, where alpha signifies Trend, beta signifies Trend and gamma signifies seasonality.

```
HoltWinters.gas = HoltWinters(gas.train, alpha = NULL, beta = NULL, gamma = NULL,
                              seasonal = c("additive", "multiplicative"))
HW.forecast=forecast(HoltWinters.gas)
HW.forecast
HW.forecast$model
HW.forecast$mean
HW.forecast$fitted
accuracy(HW.forecast, gas.test)
autoplot(HW.forecast)
```

Below are the values of alpha, beta & gamma:

```
Smoothing parameters:
 alpha: 0.3766451
 beta : 0.009235467
 gamma: 0.7605161
```

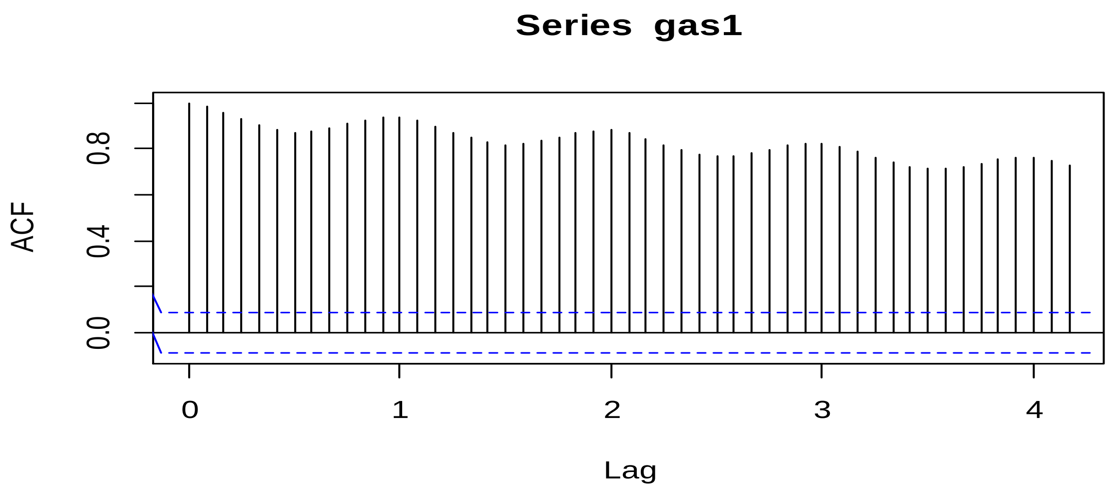
Below is the graph showing forecasts from holt winter:



Stationary Series Test for ARIMA Forecasting

Now, we will do ARIMA Forecasting. For applying ARIMA forecasting, we need a Stationary series. We have already established that the series has seasonality, but still we can check a stationary series by ADF Test.

We have done the ACF plot. From the ACF plot, we can determine that the series is seasonal.



```
library(tseries)
autoplot(gas1)
acf(gas1, lag.max = 50)
adf.test(gas1)
```

We have to make Null Hypothesis and Alternate Hypothesis for ADF test.

Ho: Series not Stationary

H1: Series Stationary

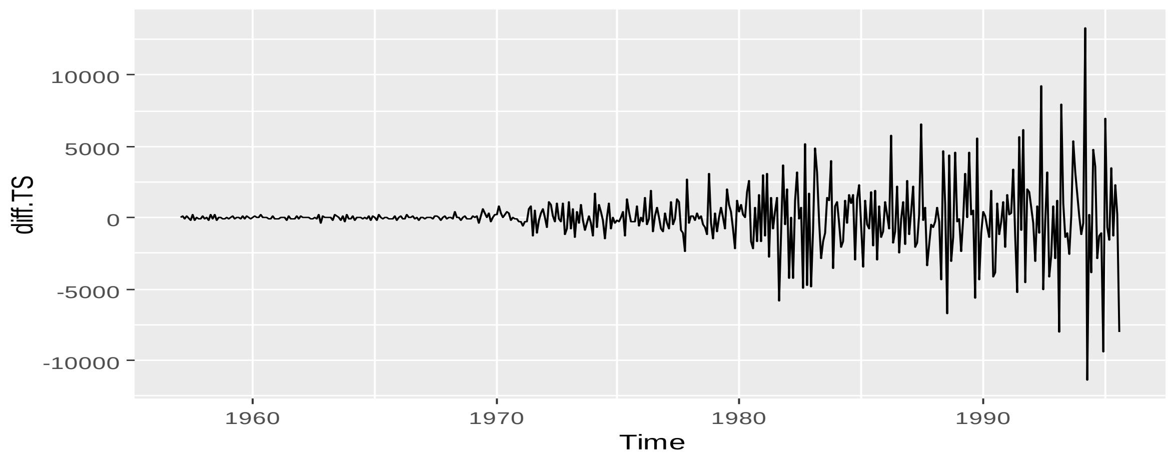
```
> adf.test(gas1)

        Augmented Dickey-Fuller Test

data:  gas1
Dickey-Fuller = -2.7131, Lag order = 7, p-value = 0.2764
alternative hypothesis:  stationary
```

Since the p value is 0.2764 we have to accept the null hypothesis, that is the series is non stationary.

Now, we will remove the seasonality of the time series by differencing of the series. We will do a differencing by 12 since seasonal series.

```
diff.TS=diff(diff(gas1,lag =12),1)
autoplot(diff.TS)
adf.test(diff.TS)
acf(diff.TS, lag.max=50)
acf(diff.TS, lag.max=50, plot = FALSE)
pacf(diff.TS, lag.max=50)
pacf(diff.TS, lag.max=50, plot = FALSE)
```

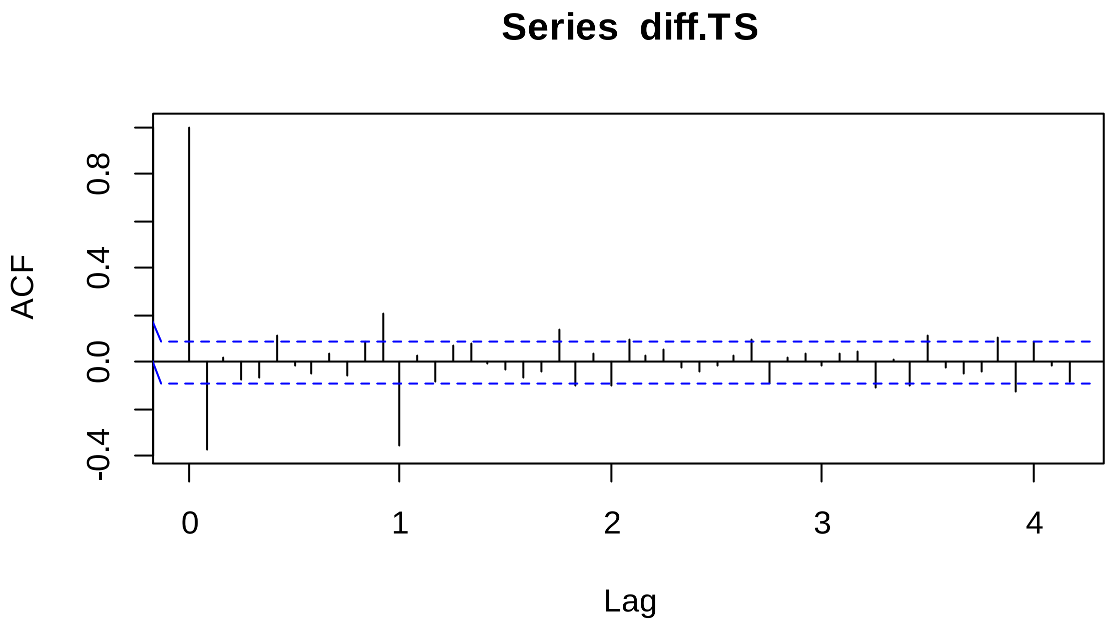
We get the following series by differencing.

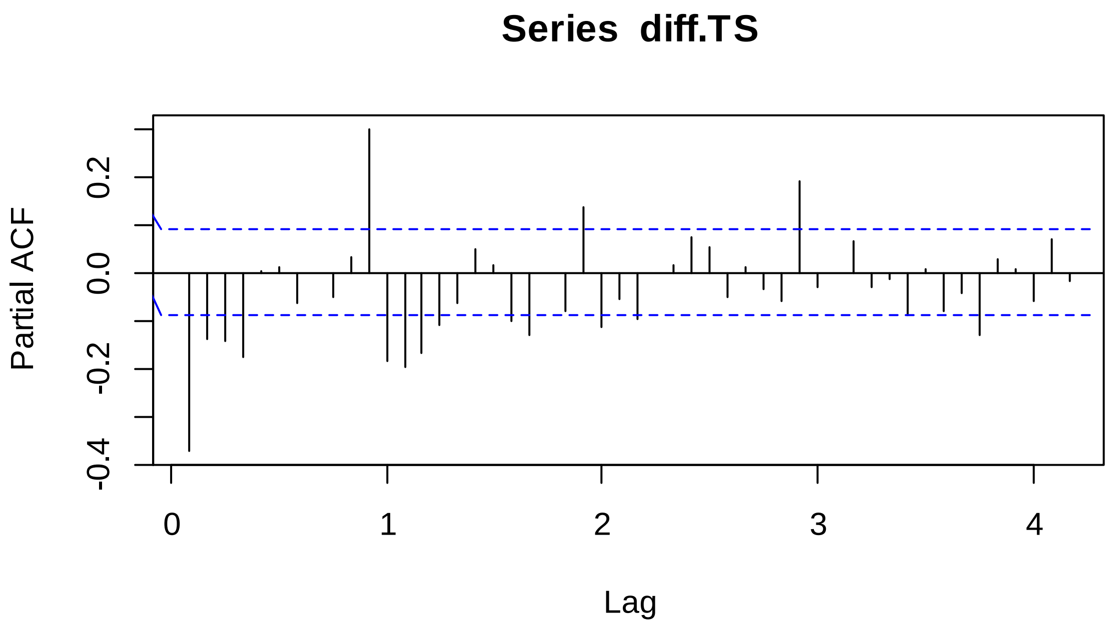


Now, we do the ADF test again to check for stationarity.

```
> adf.test(diff.TS)

	Augmented Dickey-Fuller Test

data:  diff.TS
Dickey-Fuller = -9.0548, Lag order = 7, p-value = 0.01
alternative hypothesis: stationary
```

The low value of p value suggests that the series is stationary. The ACF test is done for the difference series. Now we can apply ARIMA Forecasting.





By the ACF and the PACF plot, we determine the value of p,d,q of ARIMA(p,d,q)

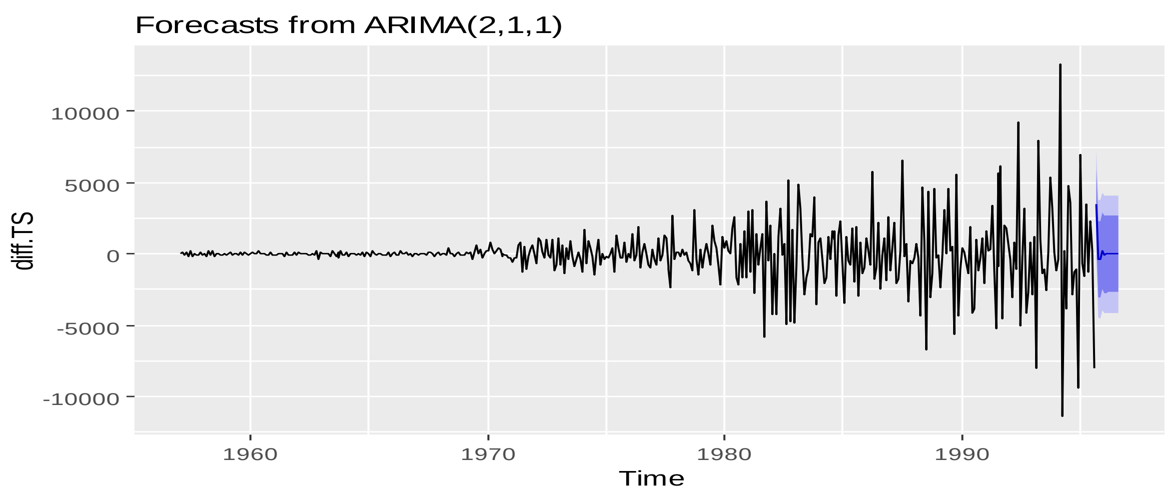
We will consider p = 2; d = 1; q = 1 for the ARIMA Modeling.

ARIMA Forecasting

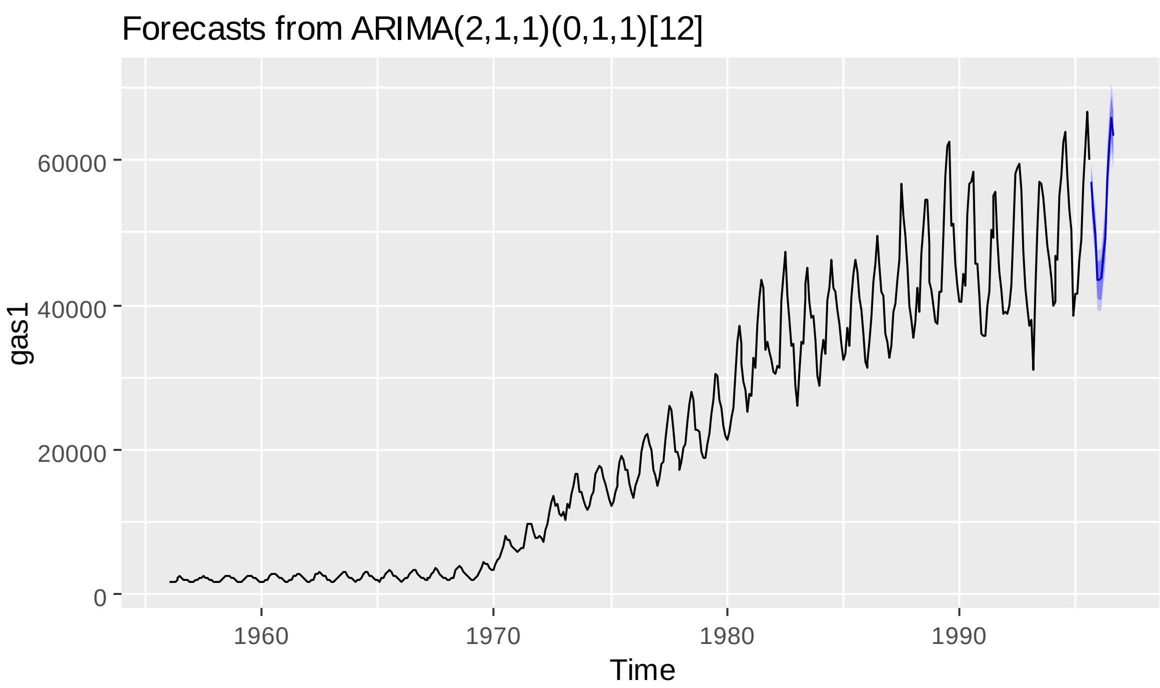
Now, since we have established that the diff.TS series is stationary.

```
library(fpp2)

gas.arima.fit <- Arima(diff.TS,order =c(2,1,1))

gas.arima.fit

checkresiduals(gas.arima.fit)

Box.test(gas.arima.fit$residuals, lag=30, type="Ljung-Box")

autoplot(forecast(gas.arima.fit, h=12))

gas.auto.arima.fit <- auto.arima(gas.train)

ARIMA.forecast=forecast(gas.auto.arima.fit)

accuracy(ARIMA.forecast,gas.test)

gas.auto.arima.fit <- auto.arima()

gas.auto.arima.fit <- auto.arima(gas1)

autoplot(forecast(gas.auto.arima.fit, h=12))
```

We will forecast for the next 12 periods using ARIMA Forecasting.



We have used the auto.arima() function to forecast the model. The below graph shows the forecasting.



ACCURACY of the model

We have tested the auto.arima model with the test & train data set. We have found the Mean Absolute Percentage Error(MAPE) of the test set = 4.5%

```
> accuracy(ARIMA.forecast,gas.test)
                      ME     RMSE       MAE       MPE     MAPE      MASE
ACF1 Theil's U
Training set    18.43311 1551.733  876.3621 0.1845762 3.887880 0.4723883 -0
.007149862        NA
Test set      1797.05962 2623.714 2420.3779 3.4661214 4.504051 1.3046641 -0
.417516251 0.5264559
```