ts_2

Soumarya Basak

08/06/2022

```
library(ggplot2)
library(TSA)
```

```
##
## Attaching package: 'TSA'

## The following objects are masked from 'package:stats':
##
##     acf, arima

## The following object is masked from 'package:utils':
##
##     tar
```

```
library(tseries)
```

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

```
library(forecast)
```

```
## Registered S3 methods overwritten by 'forecast':
##   method       from
##   fitted.Arima TSA
##   plot.Arima   TSA
```

```
library(TTR)
```

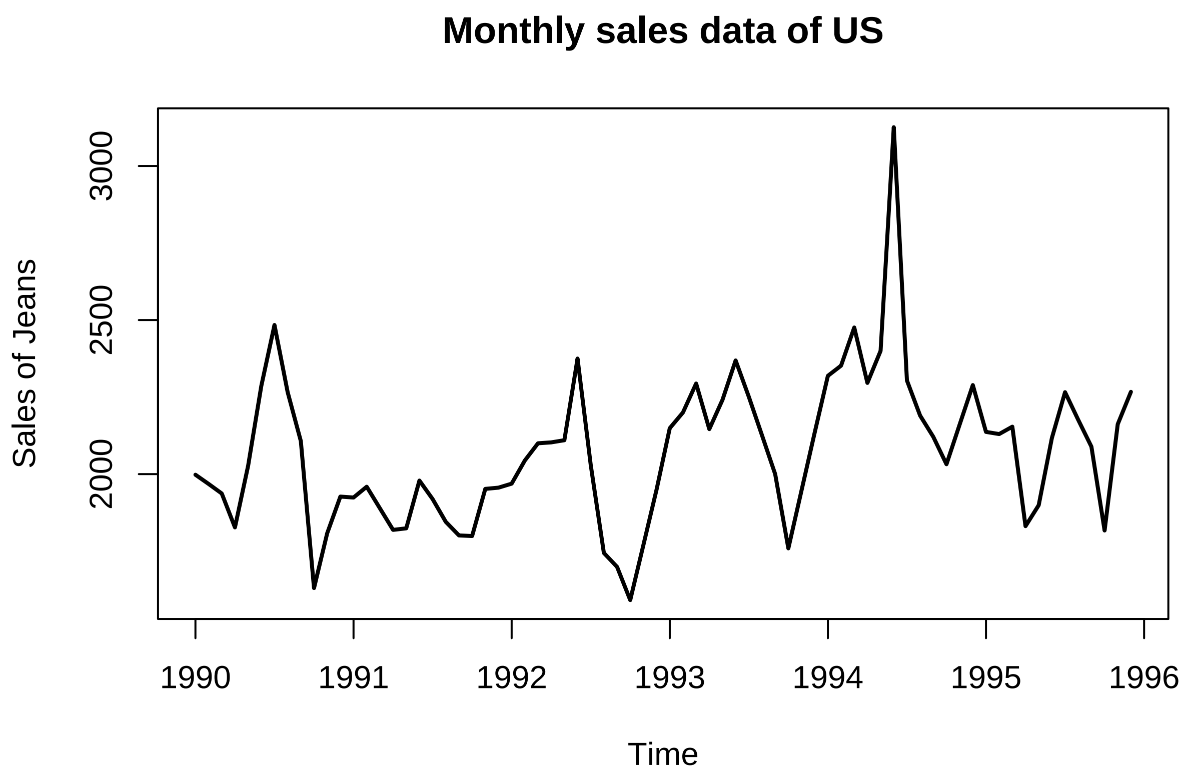
Importing the Data

```
df<- read.csv("C:\\Users\\souma\\Dropbox\\Mstat_CU\\Sem 2\\Regression_analysis_1\\Data Sets\\ts_2.csv",

# Convert this into time series data
ts2<- ts(df, start = c(1990,1), frequency = 12)
ts2
```

```
##       Jan  Feb  Mar  Apr  May  Jun  Jul  Aug  Sep  Oct  Nov  Dec
## 1990 1998 1968 1937 1827 2027 2286 2484 2266 2107 1630 1808 1927
## 1991 1924 1959 1889 1819 1824 1979 1919 1845 1801 1799 1952 1956
## 1992 1969 2044 2100 2103 2110 2375 2030 1744 1699 1591 1770 1950
## 1993 2149 2200 2294 2146 2241 2369 2251 2126 2000 1759 1947 2135
## 1994 2319 2352 2476 2296 2400 3126 2304 2190 2121 2032 2161 2289
## 1995 2137 2130 2154 1831 1899 2117 2266 2176 2089 1817 2162 2267
```

```
# Lets see the time series
plot(ts2, lwd=2, ylab="Sales of Jeans", main="Monthly sales data of US")
```



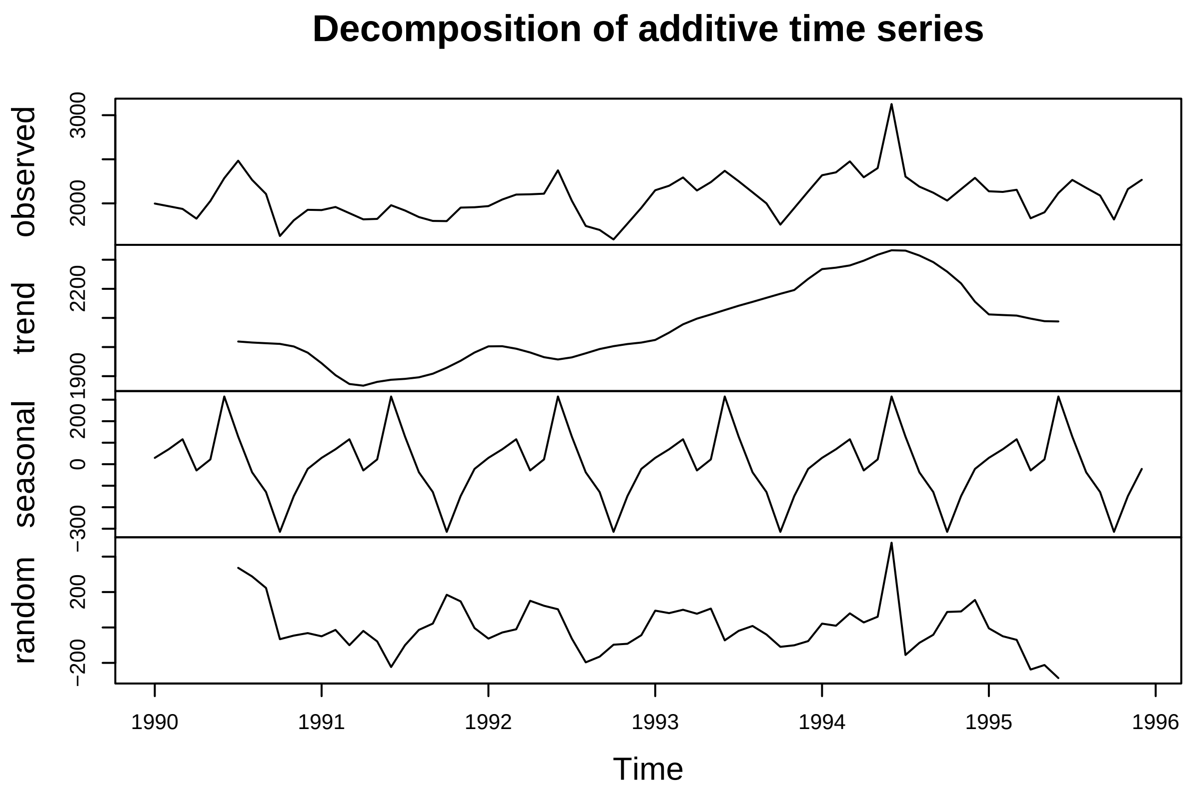
Docomposition

```
decompose(ts2)
```

```
## $x
##       Jan  Feb  Mar  Apr  May  Jun  Jul  Aug  Sep  Oct  Nov  Dec
## 1990 1998 1968 1937 1827 2027 2286 2484 2266 2107 1630 1808 1927
## 1991 1924 1959 1889 1819 1824 1979 1919 1845 1801 1799 1952 1956
## 1992 1969 2044 2100 2103 2110 2375 2030 1744 1699 1591 1770 1950
## 1993 2149 2200 2294 2146 2241 2369 2251 2126 2000 1759 1947 2135
## 1994 2319 2352 2476 2296 2400 3126 2304 2190 2121 2032 2161 2289
## 1995 2137 2130 2154 1831 1899 2117 2266 2176 2089 1817 2162 2267
##
## $seasonal
##            Jan        Feb        Mar        Apr        May        Jun
## 1990   29.54167   69.50833  116.00833  -29.00000   22.29167  314.90833
## 1991   29.54167   69.50833  116.00833  -29.00000   22.29167  314.90833
## 1992   29.54167   69.50833  116.00833  -29.00000   22.29167  314.90833
## 1993   29.54167   69.50833  116.00833  -29.00000   22.29167  314.90833
## 1994   29.54167   69.50833  116.00833  -29.00000   22.29167  314.90833
## 1995   29.54167   69.50833  116.00833  -29.00000   22.29167  314.90833
##            Jul        Aug        Sep        Oct        Nov        Dec
## 1990  128.31667  -37.59167 -129.35000 -314.59167 -148.15833  -21.88333
## 1991  128.31667  -37.59167 -129.35000 -314.59167 -148.15833  -21.88333
```

```
## 1992  128.31667  -37.59167 -129.35000 -314.59167 -148.15833  -21.88333
## 1993  128.31667  -37.59167 -129.35000 -314.59167 -148.15833  -21.88333
## 1994  128.31667  -37.59167 -129.35000 -314.59167 -148.15833  -21.88333
## 1995  128.31667  -37.59167 -129.35000 -314.59167 -148.15833  -21.88333
##
## $trend
##           Jan      Feb      Mar      Apr      May      Jun      Jul      Aug
## 1990       NA       NA       NA       NA       NA       NA 2019.000 2015.542
## 1991 1944.458 1903.375 1873.083 1867.375 1880.417 1887.625 1890.708 1896.125
## 1992 2002.375 2002.792 1994.333 1981.417 1965.167 1957.333 1964.583 1978.583
## 1993 2024.458 2049.583 2078.042 2097.583 2111.958 2127.042 2141.833 2155.250
## 1994 2267.792 2272.667 2280.375 2296.792 2317.083 2332.417 2331.250 2314.417
## 1995 2112.167 2110.000 2108.083 2097.792 2088.875 2088.000       NA       NA
##           Sep      Oct      Nov      Dec
## 1990 2013.167 2010.833 2002.042 1980.792
## 1991 1908.458 1929.083 1952.833 1981.250
## 1992 1993.167 2003.042 2010.292 2015.500
## 1993 2169.167 2183.000 2195.875 2234.042
## 1994 2291.750 2258.958 2218.708 2155.792
## 1995       NA       NA       NA       NA
##
## $random
##             Jan         Feb         Mar         Apr         May         Jun
## 1990         NA          NA          NA          NA          NA          NA
## 1991  -50.000000  -13.883333 -100.091667  -19.375000  -78.708333 -223.533333
## 1992  -62.916667  -28.300000  -10.341667  150.583333  122.541667  102.758333
## 1993   95.000000   80.908333   99.950000   77.416667  106.750000  -72.950000
## 1994   21.666667    9.825000   79.616667   28.208333   60.625000  478.675000
## 1995   -4.708333  -49.508333  -70.091667 -237.791667 -212.166667 -285.908333
##             Jul         Aug         Sep         Oct         Nov         Dec
## 1990  336.683333  288.050000  223.183333  -66.241667  -45.883333  -31.908333
## 1991 -100.025000  -13.533333   21.891667  184.508333  147.325000   -3.366667
## 1992  -62.900000 -196.991667 -164.816667  -97.450000  -92.133333  -43.616667
## 1993  -19.150000    8.341667  -39.816667 -109.408333 -100.716667  -77.158333
## 1994 -155.566667  -86.825000  -41.400000   87.633333   90.450000  155.091667
## 1995          NA          NA          NA          NA          NA          NA
##
## $figure
##  [1]   29.54167   69.50833  116.00833  -29.00000   22.29167  314.90833
##  [7]  128.31667  -37.59167 -129.35000 -314.59167 -148.15833  -21.88333
##
## $type
## [1] "additive"
##
## attr(,"class")
## [1] "decomposed.ts"
```

```
plot(decompose(ts2))
```



Exponential smothing

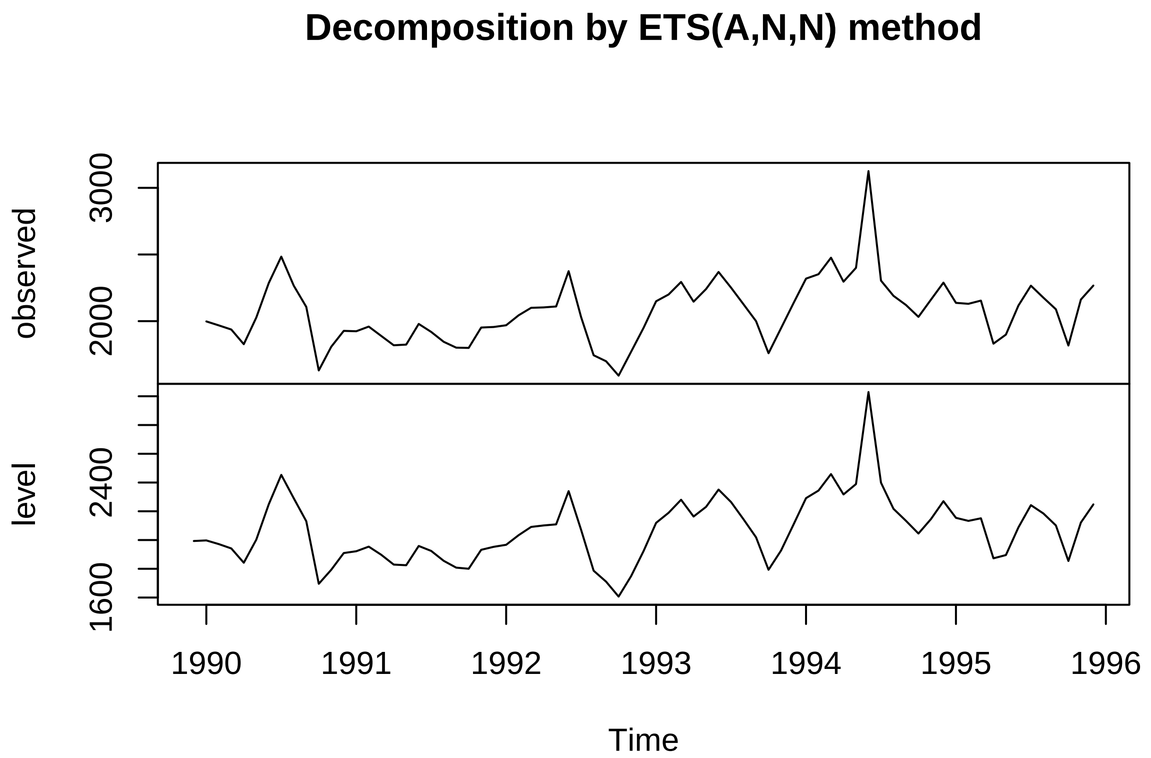
Now we will model this data using exponential smoothing method and observe the fitting and forecast for the next year.

```
m_es<- ets(ts2, model = "ANN")  ## ANN model indicates the simple exponential smoothing
summary(m_es)
```

```
## ETS(A,N,N)
##
## Call:
##  ets(y = ts2, model = "ANN")
##
##   Smoothing parameters:
##     alpha = 0.8692
##
##   Initial states:
##     l = 1993.2777
##
##   sigma:  205.4114
##
##      AIC     AICc      BIC
## 1078.694 1079.047 1085.524
##
## Training set error measures:
##                    ME     RMSE      MAE        MPE     MAPE      MASE
## Training set 4.070141 202.5383 149.1347 -0.2926178 7.121569 0.7509301
```

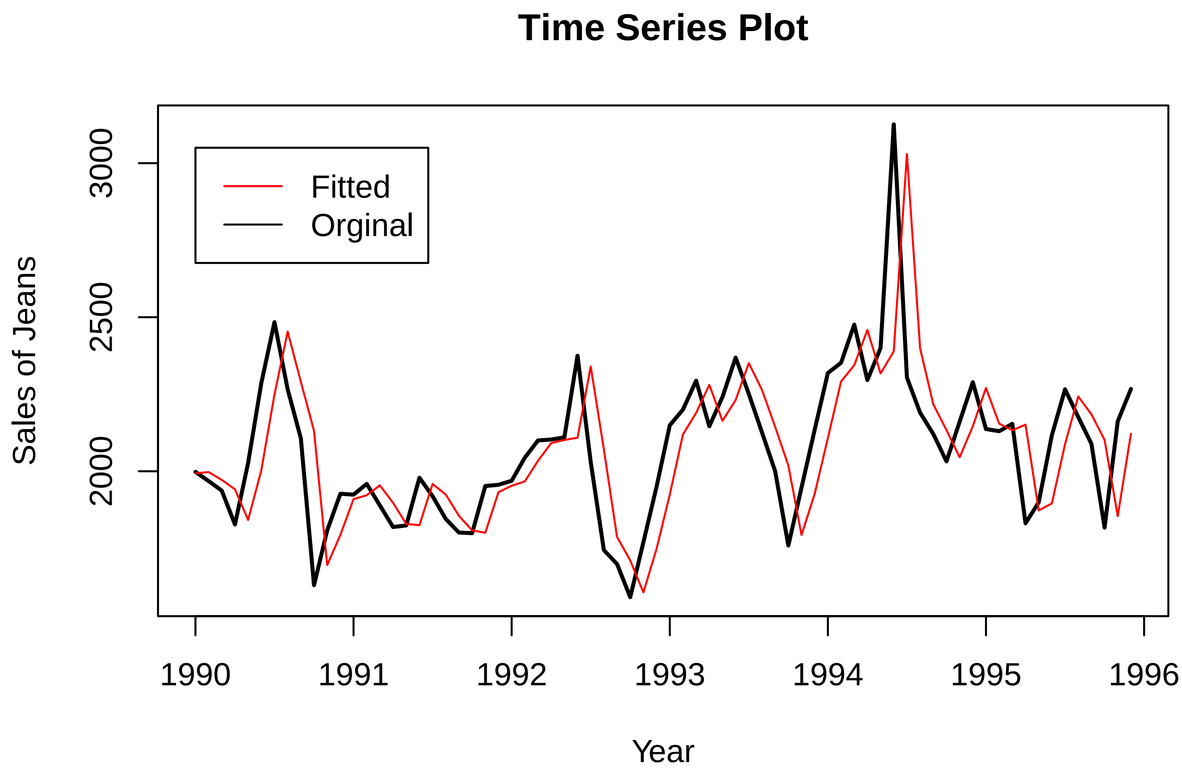
```
##                        ACF1
## Training set 0.02560041
```

```
plot(m_es)
```



SO we get the estiate of $\alpha = 0.869$

```
plot(ts2, lwd=2, main="Time Series Plot",ylab="Sales of Jeans", xlab="Year")
lines(fitted(m_es), col="red")
legend(1990,3050,legend=c("Fitted","Orginal"), col=c("red", "black"), lwd=c(1,1))
```

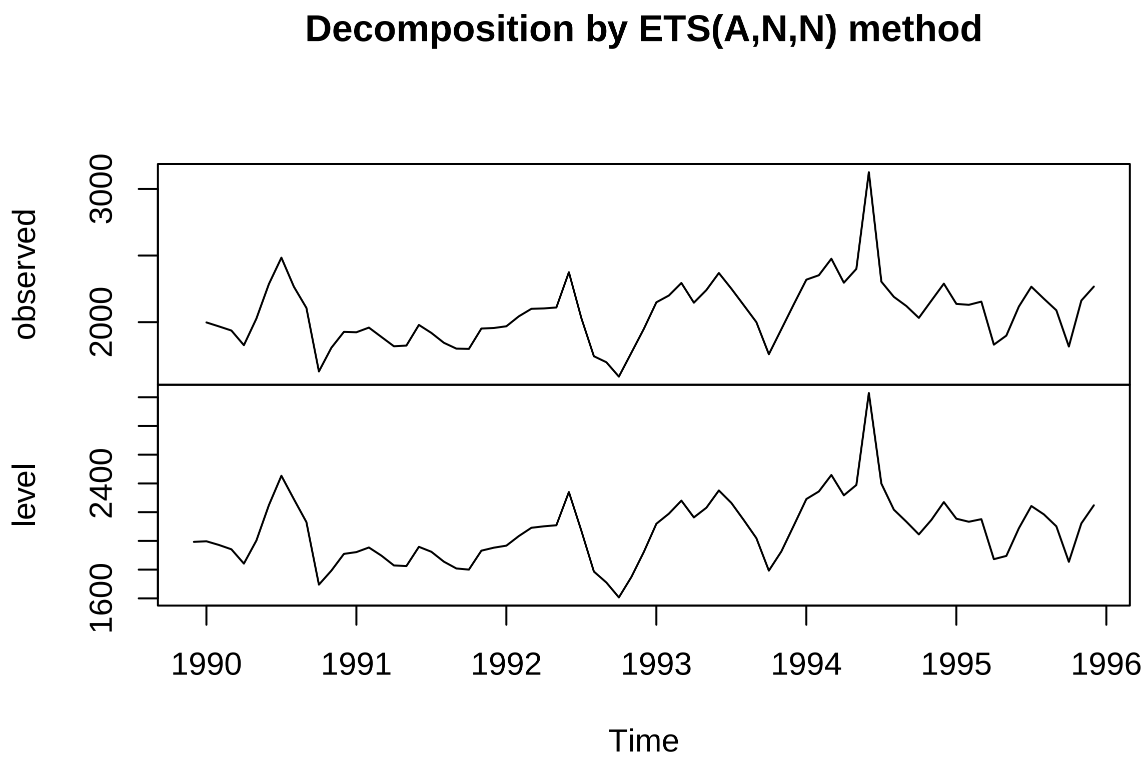


Prediction for subsequent year

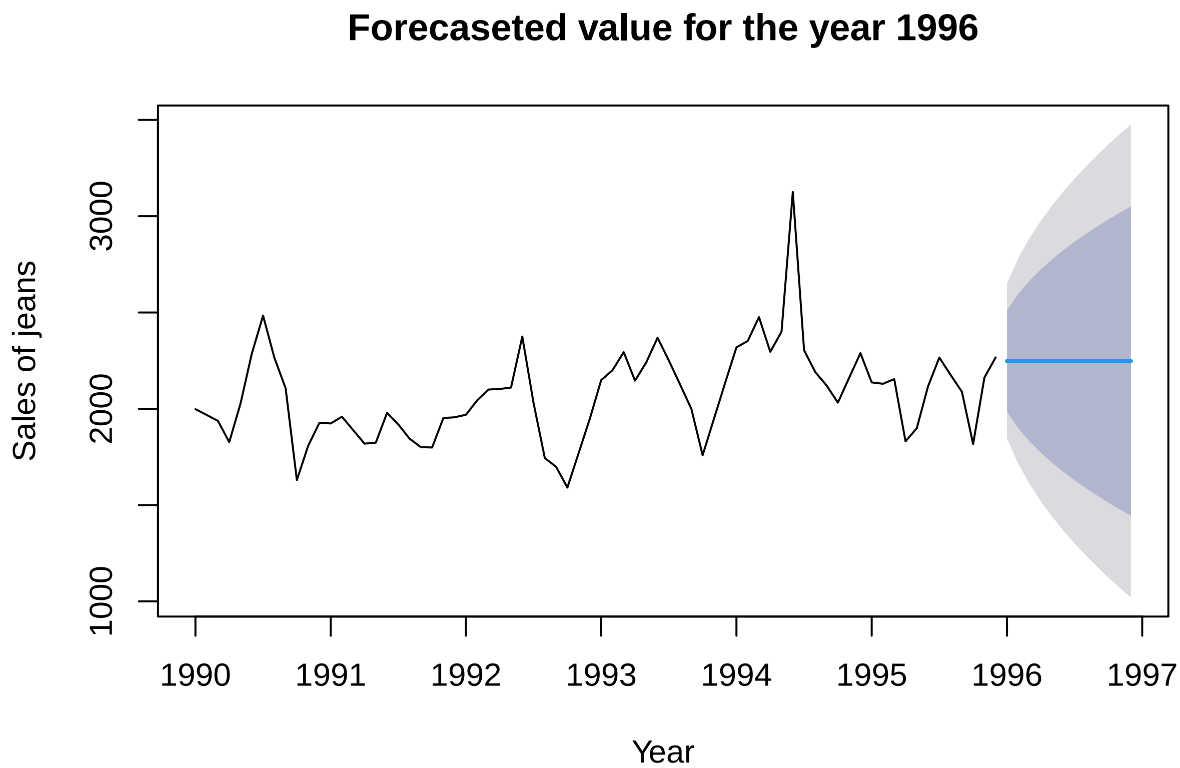
```
forecast(m_es, h=12)
```

```
##          Point Forecast    Lo 80    Hi 80    Lo 95    Hi 95
## Jan 1996       2248.006 1984.760 2511.251 1845.407 2650.604
## Feb 1996       2248.006 1899.212 2596.799 1714.572 2781.439
## Mar 1996       2248.006 1830.854 2665.157 1610.027 2885.984
## Apr 1996       2248.006 1772.217 2723.794 1520.350 2975.661
## May 1996       2248.006 1720.054 2775.958 1440.573 3055.438
## Jun 1996       2248.006 1672.599 2823.412 1367.998 3128.013
## Jul 1996       2248.006 1628.771 2867.240 1300.969 3195.042
## Aug 1996       2248.006 1587.847 2908.164 1238.380 3257.631
## Sep 1996       2248.006 1549.315 2946.696 1179.450 3316.561
## Oct 1996       2248.006 1512.800 2983.211 1123.605 3372.406
## Nov 1996       2248.006 1478.014 3017.997 1070.406 3425.605
## Dec 1996       2248.006 1444.734 3051.277 1019.508 3476.503
```

```
plot(m_es)
```




```
plot(predict(m_es, h=12),main= "Forecaseted value for the year 1996",
     xlab="Year", ylab = "Sales of jeans")
```



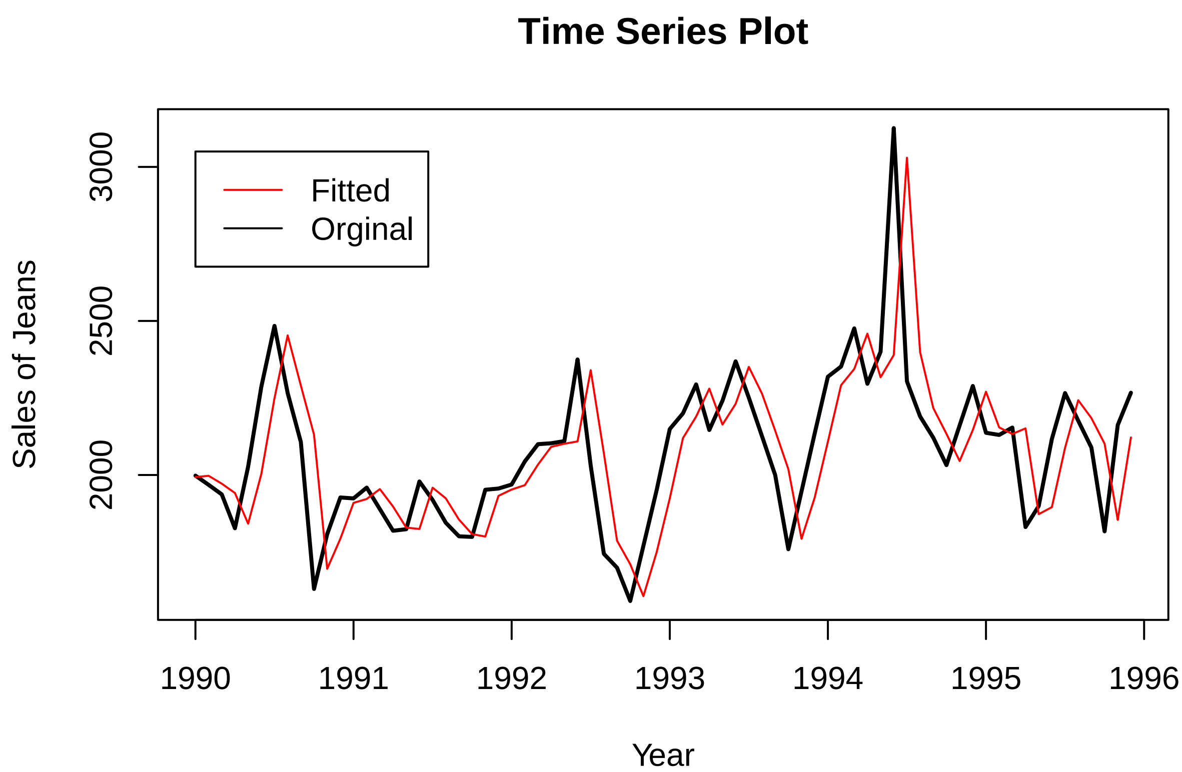
Another Approach For exponential smoonthing

```
exp_smooth<- ses(ts2)
summary(exp_smooth)
```

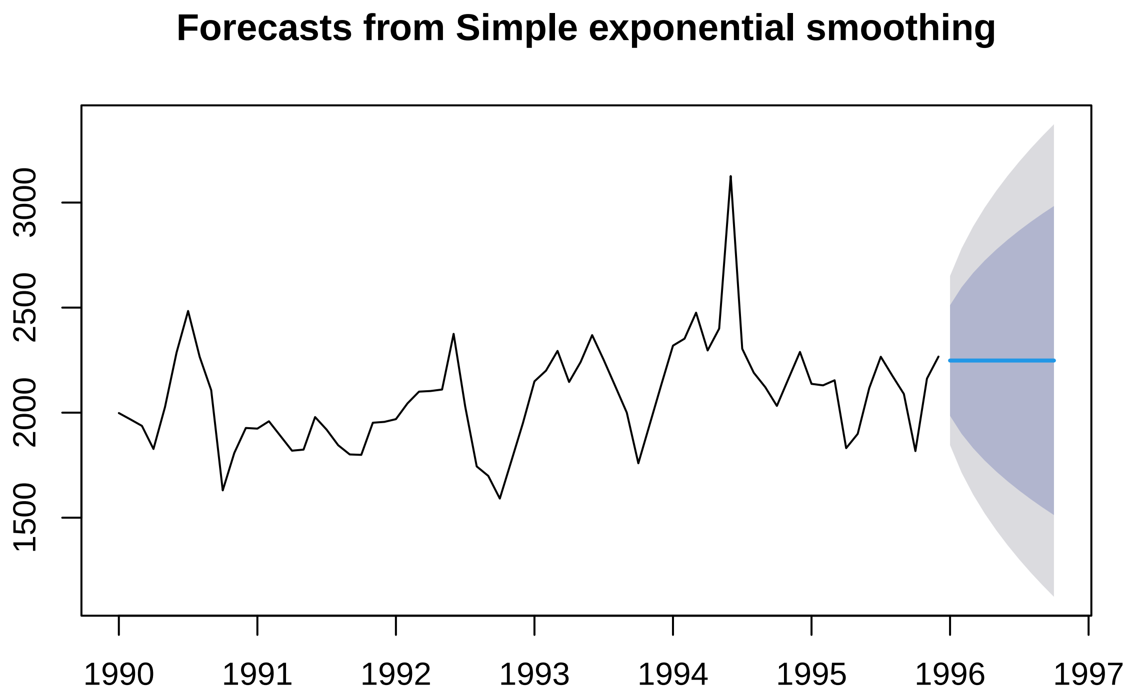
```
##
## Forecast method: Simple exponential smoothing
##
## Model Information:
## Simple exponential smoothing
##
## Call:
##  ses(y = ts2)
##
##   Smoothing parameters:
##     alpha = 0.8695
##
##   Initial states:
##     l = 1993.2154
##
##   sigma:  205.4114
##
##      AIC     AICc      BIC
## 1078.694 1079.047 1085.524
##
## Error measures:
```

```
##                      ME     RMSE      MAE        MPE     MAPE      MASE
## Training set 4.070739 202.5383 149.1293 -0.2924796 7.121298 0.7509029
##                    ACF1
## Training set 0.02542364
##
## Forecasts:
##          Point Forecast    Lo 80    Hi 80    Lo 95    Hi 95
## Jan 1996       2248.045 1984.800 2511.291 1845.447 2650.644
## Feb 1996       2248.045 1899.214 2596.877 1714.553 2781.537
## Mar 1996       2248.045 1830.830 2665.261 1609.969 2886.122
## Apr 1996       2248.045 1772.173 2723.918 1520.261 2975.829
## May 1996       2248.045 1719.992 2776.098 1440.458 3055.633
## Jun 1996       2248.045 1672.524 2823.567 1367.861 3128.230
## Jul 1996       2248.045 1628.682 2867.409 1300.811 3195.280
## Aug 1996       2248.045 1587.745 2908.346 1238.203 3257.888
## Sep 1996       2248.045 1549.202 2946.889 1179.257 3316.834
## Oct 1996       2248.045 1512.677 2983.414 1123.396 3372.695
```

```r
plot(ts2, lwd=2, main="Time Series Plot",ylab="Sales of Jeans", xlab="Year")
lines(fitted(exp_smooth), col="red")
legend(1990,3050,legend=c("Fitted","Orginal"), col=c("red", "black"), lwd=c(1,1))
```




```r
plot(forecast(exp_smooth))
```




```
#---------Residual Sum of squares of Exp_ smoothing-------------

res_e<- resid(exp_smooth)
RSS1<- sum(res_e^2)
RSS1
```

```
## [1] 2953568
```

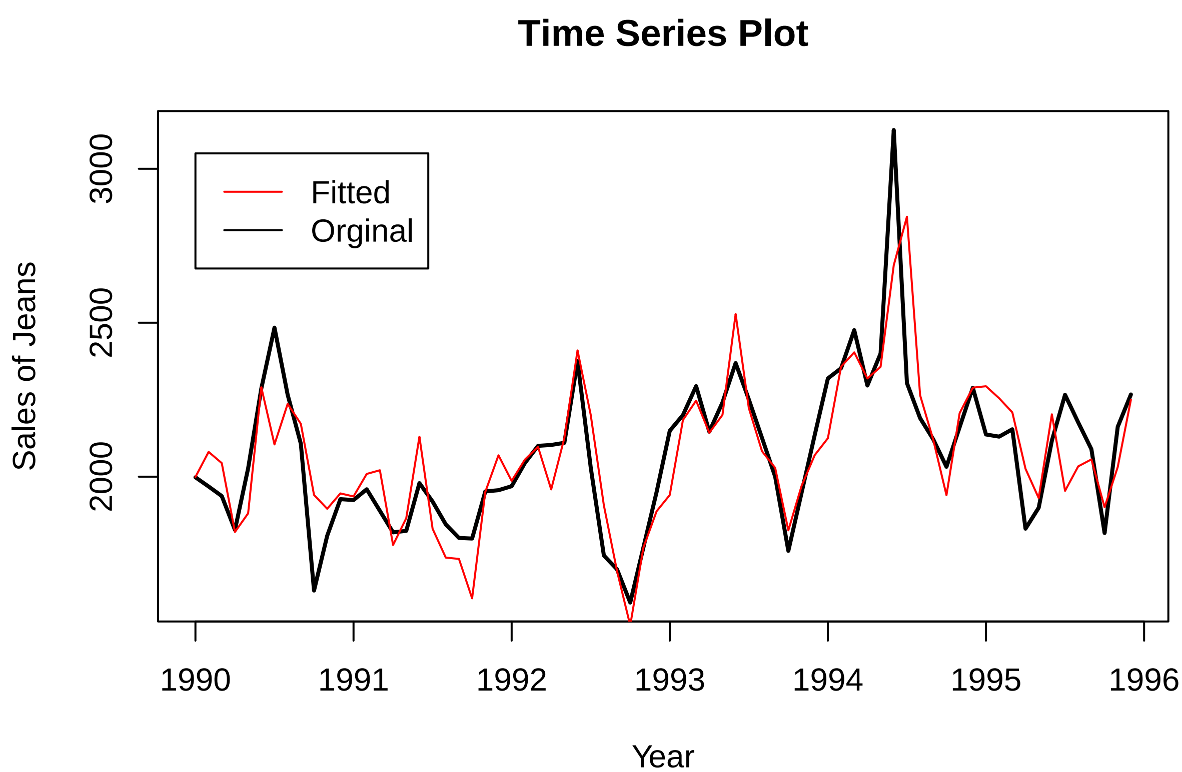
Holt Winter Method

```
m_hwt<- ets(ts2, model = "AAA")
summary(m_hwt)
```

```
## ETS(A,A,A)
##
## Call:
##  ets(y = ts2, model = "AAA")
##
##   Smoothing parameters:
##     alpha = 0.7737
##     beta  = 1e-04
##     gamma = 1e-04
##
##   Initial states:
##     l = 2003.6919
##     b = 4.072
```

```
##     s = -9.2434 -122.9214 -314.2085 -129.2761 -37.5282 128.379
##            314.7641 22.3801 -28.9488 115.9315 69.408 -8.7364
##
##   sigma:  156.5387
##
##      AIC     AICc      BIC
## 1051.501 1062.834 1090.204
##
## Training set error measures:
##                      ME     RMSE      MAE        MPE     MAPE     MASE
## Training set -0.4253545 138.0542 91.63662 -0.2571036 4.368596 0.461413
##                    ACF1
## Training set 0.0126827
```

```
plot(ts2, lwd=2, main="Time Series Plot",ylab="Sales of Jeans", xlab="Year")
lines(m_hwt$fitted, col="red")
legend(1990,3050,legend=c("Fitted","Orginal"), col=c("red", "black"), lwd=c(1,1))
```



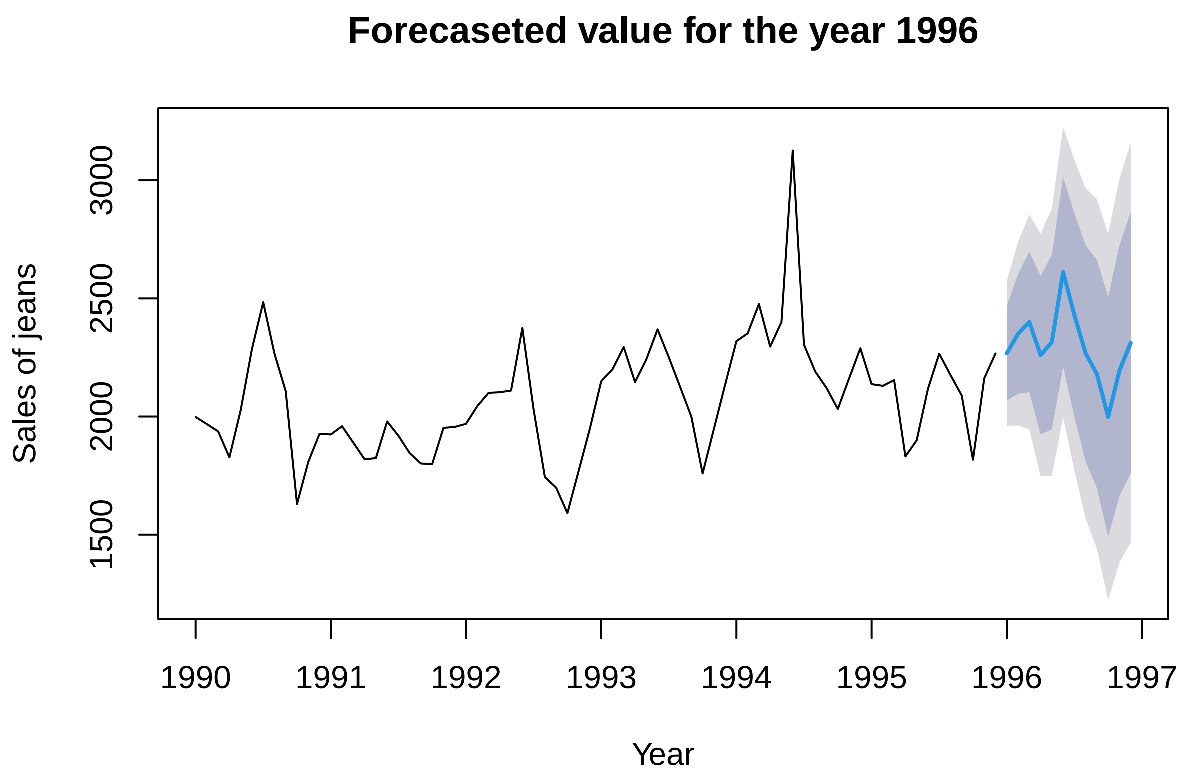
Forecast

```
forecast(m_hwt ,h=12)
```

```
##          Point Forecast    Lo 80    Hi 80    Lo 95    Hi 95
## Jan 1996       2267.789 2067.177 2468.402 1960.979 2574.600
## Feb 1996       2349.952 2096.296 2603.609 1962.019 2737.886
## Mar 1996       2400.556 2103.161 2697.951 1945.730 2855.383
## Apr 1996       2259.760 1924.272 2595.248 1746.675 2772.845
## May 1996       2315.174 1945.489 2684.859 1749.790 2880.559
```

```
## Jun 1996       2611.613 2210.629 3012.597 1998.361 3224.866
## Jul 1996       2429.306 1999.288 2859.324 1771.651 3086.962
## Aug 1996       2267.467 1810.248 2724.686 1568.211 2966.723
## Sep 1996       2179.781 1696.885 2662.677 1441.255 2918.307
## Oct 1996       1998.906 1491.624 2506.188 1223.086 2774.727
## Nov 1996       2194.270 1663.716 2724.823 1382.858 3005.681
## Dec 1996       2312.020 1759.168 2864.872 1466.506 3157.535
```

```
plot(forecast(m_hwt,h=12),,main= "Forecaseted value for the year 1996",
     xlab="Year", ylab = "Sales of jeans")
```



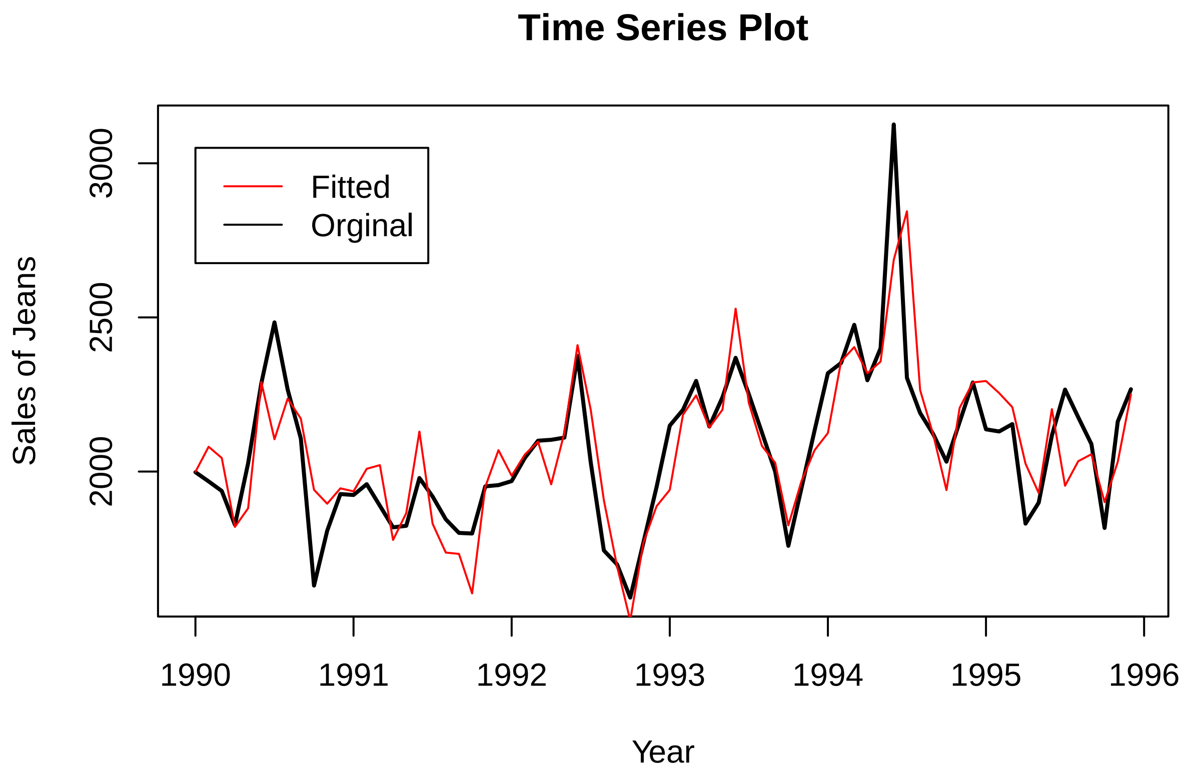
Another approach for Holt winter

```
holt<- hw(ts2)
summary(holt)
```

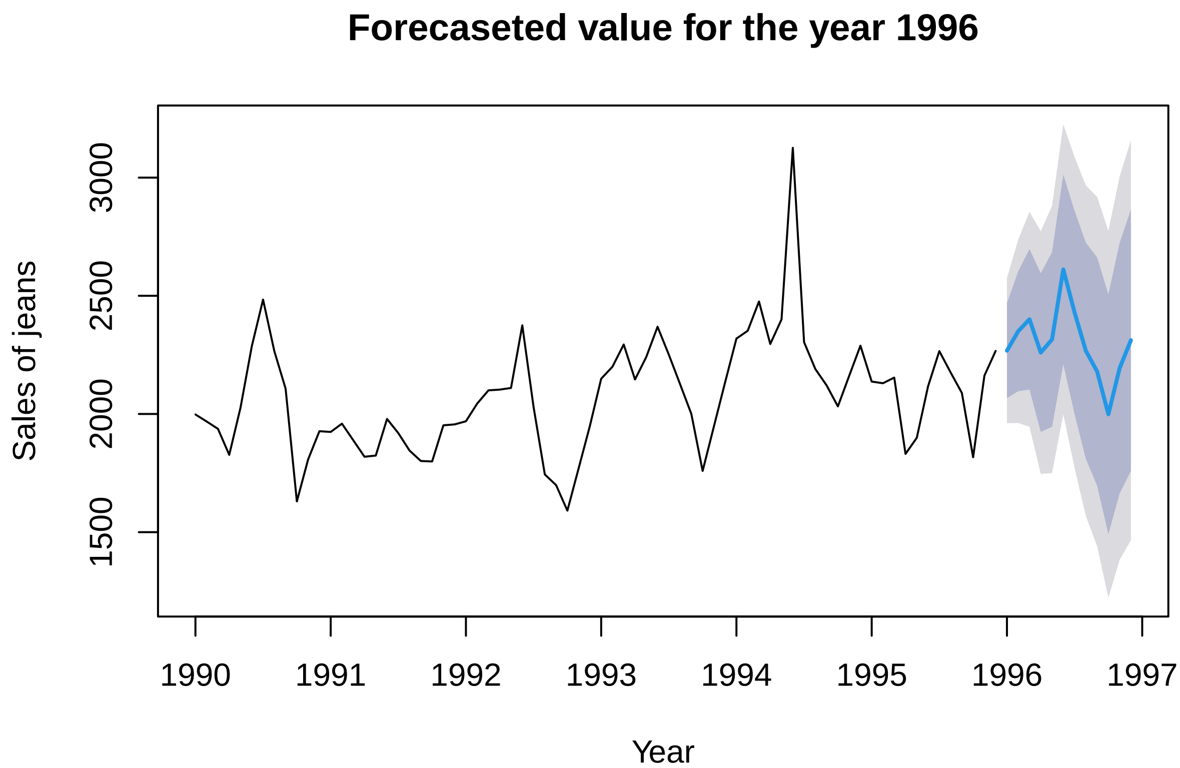
```
##
## Forecast method: Holt-Winters' additive method
##
## Model Information:
## Holt-Winters' additive method
##
## Call:
##  hw(y = ts2)
##
##   Smoothing parameters:
##     alpha = 0.7738
##     beta  = 1e-04
```

```
##     gamma = 1e-04
##
##   Initial states:
##     l = 2003.6918
##     b = 4.0719
##     s = -9.2432 -122.9205 -314.2081 -129.2762 -37.5283 128.379
##            314.7637 22.3801 -28.9485 115.931 69.4074 -8.7365
##
##   sigma:  156.5387
##
##      AIC     AICc      BIC
## 1051.501 1062.834 1090.204
##
## Error measures:
##                      ME     RMSE      MAE        MPE     MAPE      MASE
## Training set -0.4250807 138.0542 91.63232 -0.2570523 4.368377 0.4613913
##                    ACF1
## Training set 0.01252638
##
## Forecasts:
##          Point Forecast    Lo 80    Hi 80     Lo 95    Hi 95
## Jan 1996       2267.796 2067.184 2468.408 1960.9858 2574.606
## Feb 1996       2349.958 2096.282 2603.635 1961.9938 2737.923
## Mar 1996       2400.562 2103.133 2697.992 1945.6832 2855.442
## Apr 1996       2259.767 1924.233 2595.301 1746.6116 2772.922
## May 1996       2315.181 1945.440 2684.921 1749.7110 2880.650
## Jun 1996       2611.619 2210.571 3012.667 1998.2693 3224.969
## Jul 1996       2429.313 1999.223 2859.402 1771.5475 3087.078
## Aug 1996       2267.473 1810.176 2724.771 1568.0974 2966.849
## Sep 1996       2179.787 1696.806 2662.768 1441.1312 2918.443
## Oct 1996       1998.913 1491.540 2506.285 1222.9530 2774.872
## Nov 1996       2194.277 1663.627 2724.927 1382.7175 3005.836
## Dec 1996       2312.027 1759.072 2864.981 1466.3561 3157.697
## Jan 1997       2316.622 1742.219 2891.026 1438.1480 3195.097
## Feb 1997       2398.785 1803.704 2993.865 1488.6879 3308.882
## Mar 1997       2449.389 1834.321 3064.457 1508.7242 3390.054
## Apr 1997       2308.593 1674.163 2943.024 1338.3158 3278.871
## May 1997       2364.007 1710.783 3017.231 1364.9871 3363.027
## Jun 1997       2660.446 1988.949 3331.943 1633.4802 3687.411
## Jul 1997       2478.139 1788.850 3167.429 1423.9617 3532.317
## Aug 1997       2316.300 1609.661 3022.939 1235.5885 3397.011
## Sep 1997       2228.614 1505.037 2952.191 1121.9981 3335.229
## Oct 1997       2047.739 1307.607 2787.871  915.8054 3179.673
## Nov 1997       2243.103 1486.775 2999.431 1086.3992 3399.807
## Dec 1997       2360.853 1588.664 3133.042 1179.8917 3541.814
```

```
plot(ts2, lwd=2, main="Time Series Plot",ylab="Sales of Jeans", xlab="Year")
lines(holt$fitted, col="red")
legend(1990,3050,legend=c("Fitted","Orginal"), col=c("red", "black"), lwd=c(1,1))
```



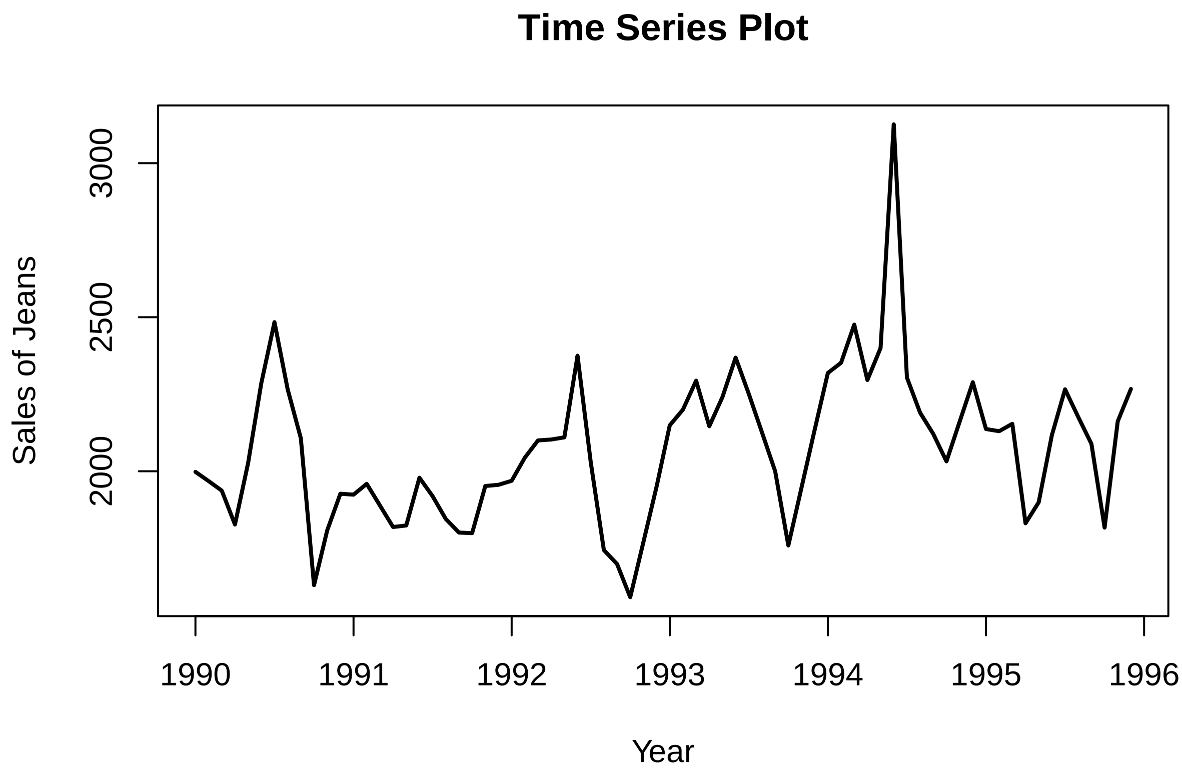

```
plot(forecast(holt,h=12),,main= "Forecaseted value for the year 1996",
     xlab="Year", ylab = "Sales of jeans")
```



Box-Jenkins Model

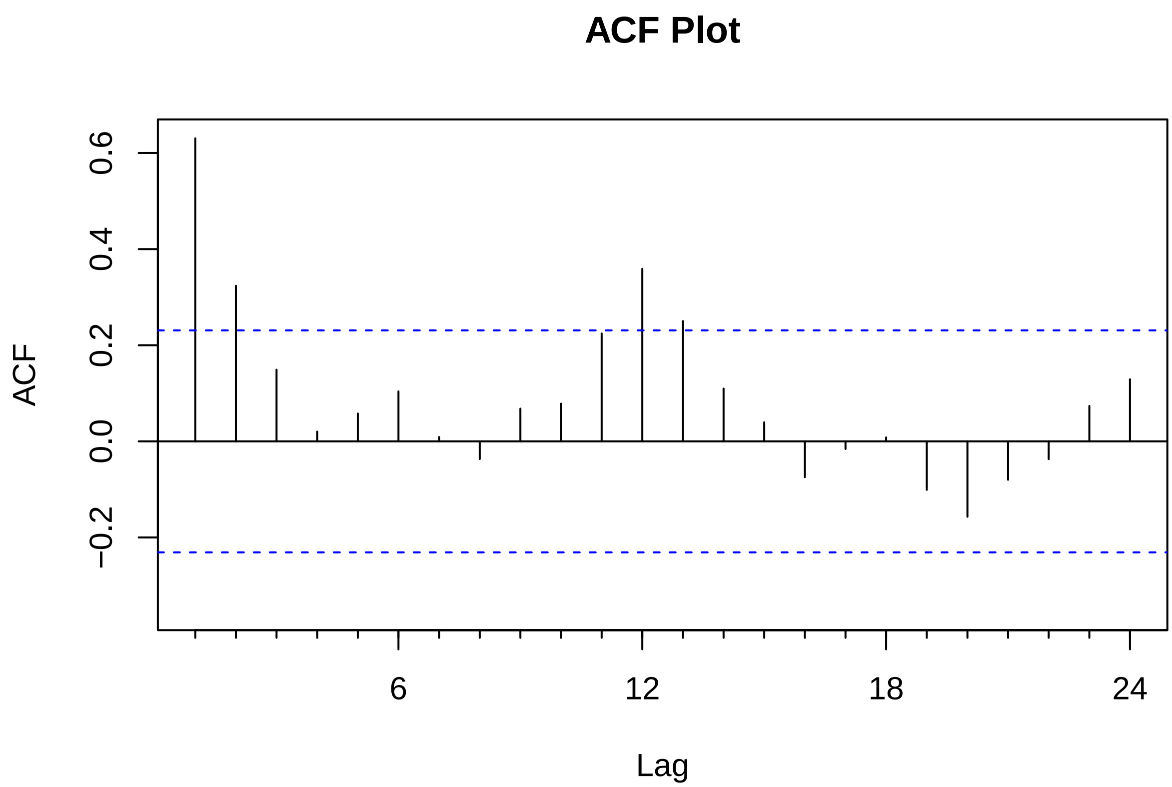
Lets starts wit the plot

```
plot(ts2, lwd=2, main="Time Series Plot",ylab="Sales of Jeans", xlab="Year")
```

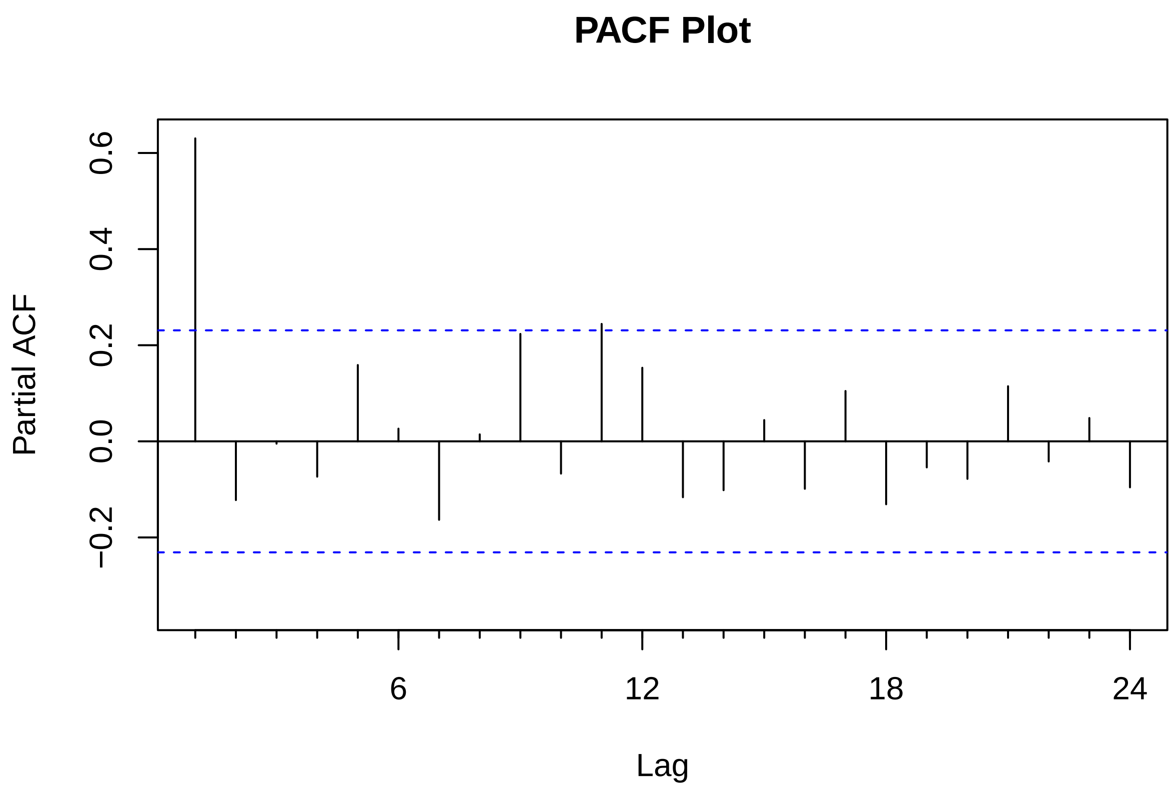


Acf and Pacf plot

```
Acf(ts2, main="ACF Plot")
```




```
Pacf(ts2,main="PACF Plot")
```



Acf plot shows that their is a presence of seasonality.

ADF test for finding d

```
adf.test(ts2)
```

```
##
##  Augmented Dickey-Fuller Test
##
## data:  ts2
## Dickey-Fuller = -3.0414, Lag order = 4, p-value = 0.1513
## alternative hypothesis: stationary
```

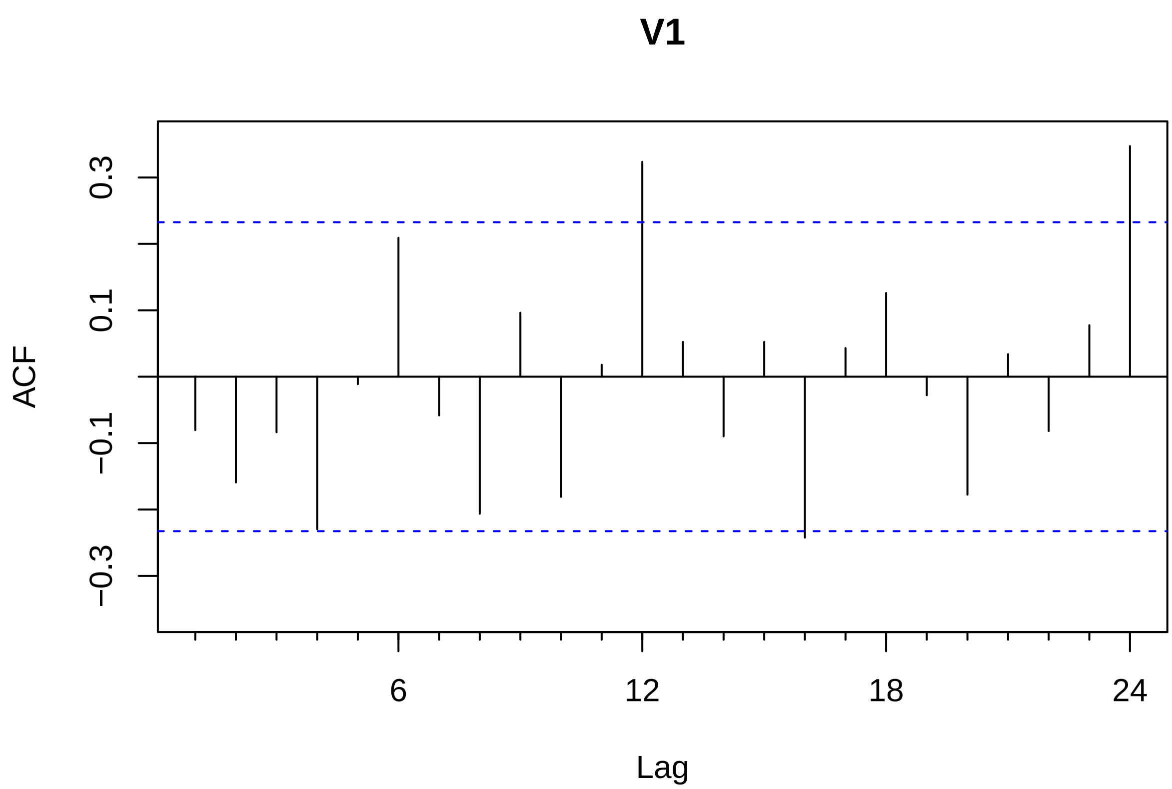
```
# Adf test after 1st difference
dif_1<- diff(ts2, 1)
adf.test(dif_1)
```

```
## Warning in adf.test(dif_1): p-value smaller than printed p-value
```

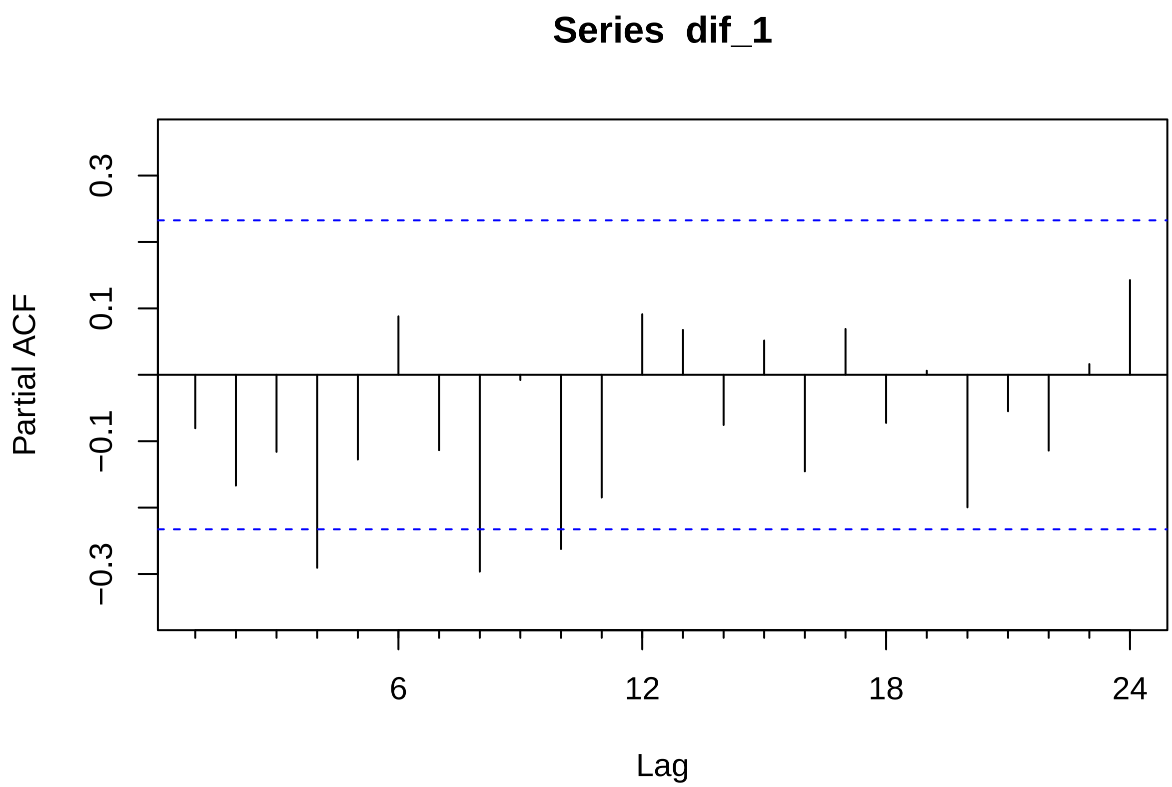
```
##
##  Augmented Dickey-Fuller Test
##
## data:  dif_1
## Dickey-Fuller = -5.5452, Lag order = 4, p-value = 0.01
## alternative hypothesis: stationary
```

The p-value is lesser than 0.05 , so we reject the null hypothesis. which indicates that the data is stationary at 1st differentiating. so, `d=1`

```r
#Acf plot for first difference
Acf(dif_1)
```




```r
Pacf(dif_1)
```



After 1st difference the data becomes nonstationary in trend.

SARIMA Model

Finding d

```
adf.test(ts2,k=1)
```

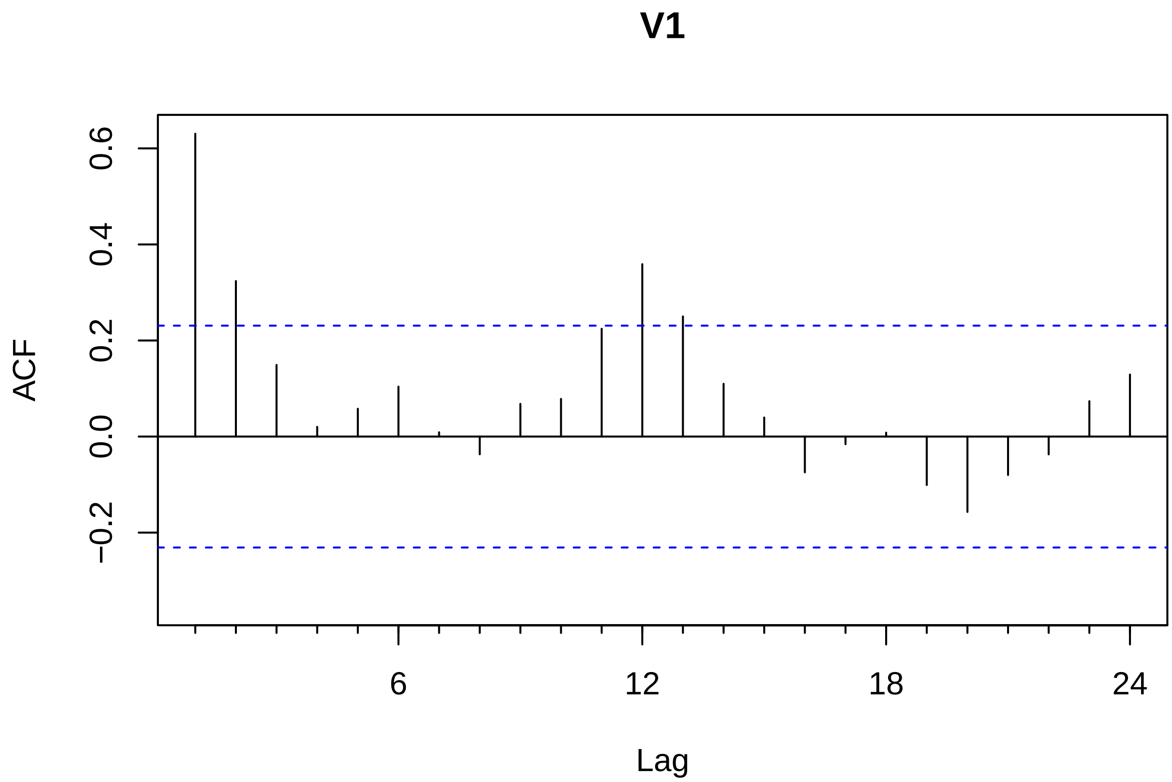
```
## Warning in adf.test(ts2, k = 1): p-value smaller than printed p-value
```

```
##
##  Augmented Dickey-Fuller Test
##
## data:  ts2
## Dickey-Fuller = -4.3153, Lag order = 1, p-value = 0.01
## alternative hypothesis: stationary
```

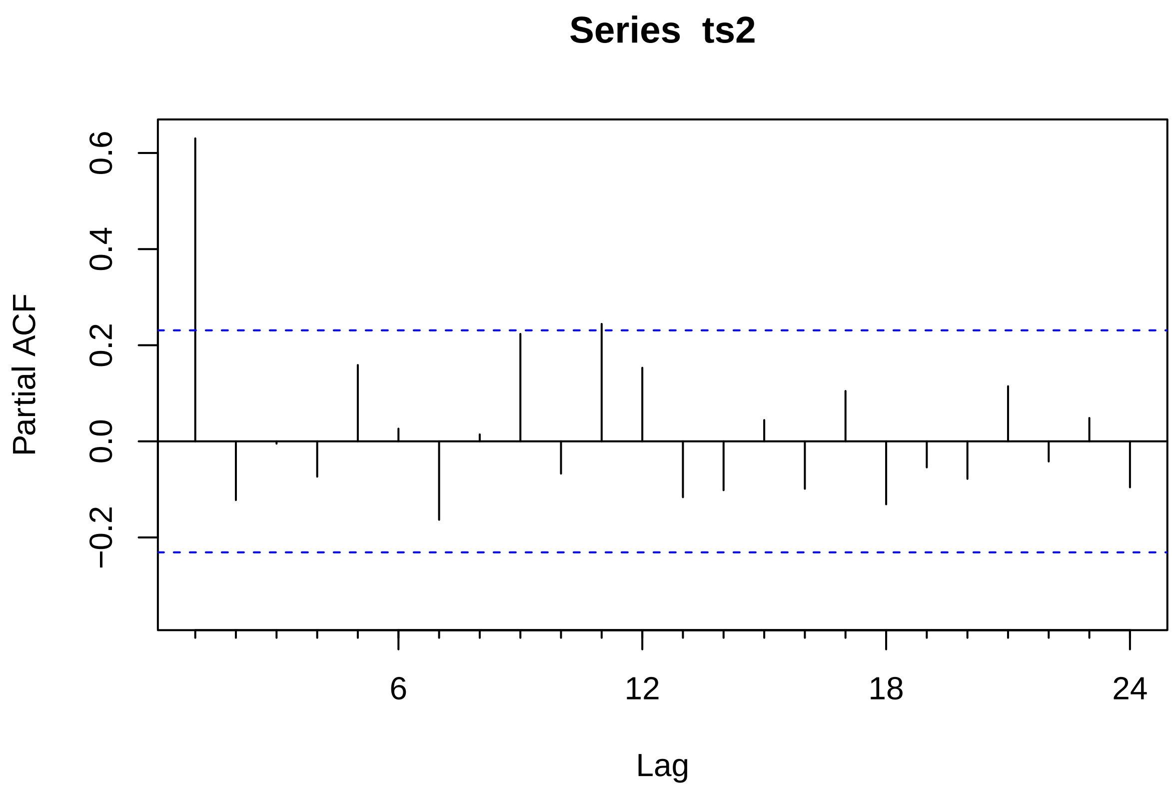
So the data becomes non stationary in trend after first difference.

Finding P and Q

```
dif_12<- diff(ts2,12) #12 th different
Acf(ts2)
```




```
Pacf(ts2)
```



We can find that there is a significant spike at lag 12 and 13 on ACF and at 11 point in PAcf So, the the seasonal MA order is `Q=1`. and we may consider `P=1 or 0`

Finding `D`

```
dif_12<- diff(ts2,12) # 12th difference of the ts
adf.test(dif_12)
```

```
##
##  Augmented Dickey-Fuller Test
##
## data:  dif_12
## Dickey-Fuller = -2.4061, Lag order = 3, p-value = 0.4106
## alternative hypothesis: stationary
```

The p value is larger than 0.05, so we need to accept the fact that the process is non stationary.

After s^{th} difference the the model is still non-stationary,

```
dif_24<- diff(ts2, 24) # 24th order difference
adf.test(dif_24)
```

```
##
##  Augmented Dickey-Fuller Test
##
## data:  dif_24
## Dickey-Fuller = -2.4978, Lag order = 3, p-value = 0.3753
## alternative hypothesis: stationary
```

After $2s^{th}$ difference the the model is still non-stationary,

Sarima

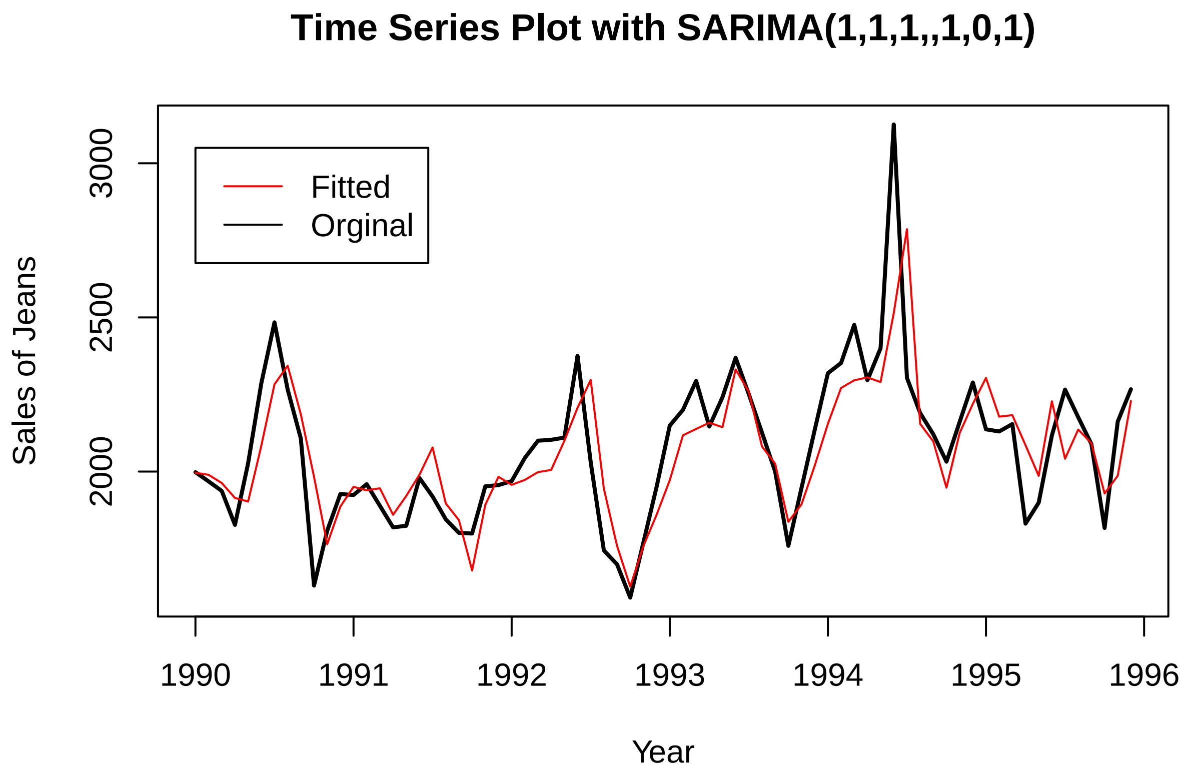
```r
# First model SARIMA(1,1,1)(1,0,1)
m_sarima<- Arima(ts2, order = c(1,1,1), seasonal = list(order=c(1,0,1), period=12))
summary(m_sarima)
```

```
## Series: ts2
## ARIMA(1,1,1)(1,0,1)[12]
##
## Coefficients:
##          ar1      ma1    sar1     sma1
##       0.6832  -0.9732  0.9988  -0.9574
## s.e.  0.1017   0.0288  0.0076   0.1311
##
## sigma^2 = 22380:  log likelihood = -463.4
## AIC=936.81   AICc=937.73   BIC=948.12
##
## Training set error measures:
##                    ME     RMSE      MAE       MPE     MAPE      MASE
## Training set 12.14234 144.3115 99.42659 0.1671923 4.694486 0.5006374
##                     ACF1
## Training set -0.01656748
```

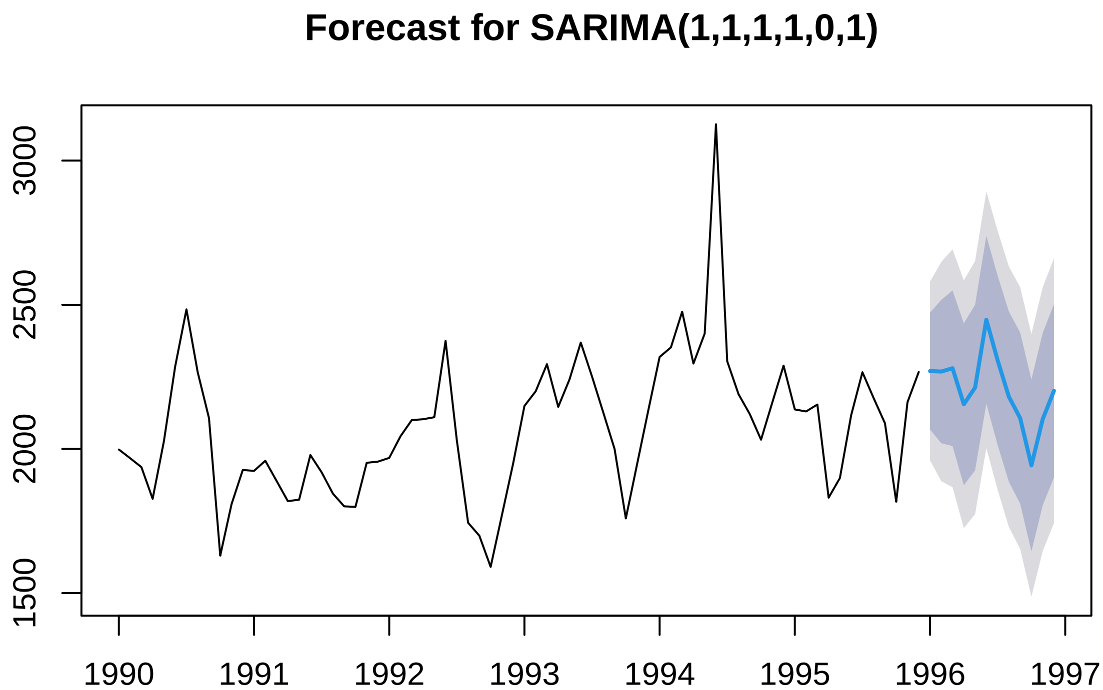
```r
forecast(m_sarima,h=12)
```

```
##          Point Forecast    Lo 80    Hi 80    Lo 95    Hi 95
## Jan 1996       2270.113 2067.454 2472.773 1960.172 2580.055
## Feb 1996       2268.359 2019.814 2516.904 1888.242 2648.476
## Mar 1996       2279.827 2010.052 2549.602 1867.241 2692.412
## Apr 1996       2154.720 1873.854 2435.586 1725.172 2584.268
## May 1996       2212.855 1925.699 2500.011 1773.688 2652.022
## Jun 1996       2448.260 2157.258 2739.262 2003.211 2893.309
## Jul 1996       2307.385 2013.839 2600.930 1858.446 2756.323
## Aug 1996       2181.095 1885.721 2476.469 1729.359 2632.830
## Sep 1996       2106.892 1810.083 2403.701 1652.961 2560.823
## Oct 1996       1942.953 1644.918 2240.988 1487.147 2398.758
## Nov 1996       2103.067 1803.914 2402.219 1645.553 2560.581
## Dec 1996       2201.672 1901.481 2501.862 1742.569 2660.774
```

```r
plot(ts2, lwd=2, main="Time Series Plot with SARIMA(1,1,1,,1,0,1)",ylab="Sales of Jeans", xlab="Year")
lines(m_sarima$fitted, col="red")
legend(1990,3050,legend=c("Fitted","Orginal"), col=c("red", "black"), lwd=c(1,1))
```




```
plot(forecast(m_sarima,h=12), main = " Forecast for SARIMA(1,1,1,1,0,1)")
```




```r
# m0del 2;
m_sarima2<- Arima(ts2, order = c(1,1,1), seasonal = list(order=c(1,1,1), period=12))
#m_sarima3<- Arima(ts2, order = c(1,1,1), seasonal = list(order=c(1,2,1), period=12)) #
m_sarima4<- Arima(ts2, order = c(1,1,1), seasonal = list(order=c(0,0,1), period=12))
m_sarima5<- Arima(ts2, order = c(1,1,1), seasonal = list(order=c(1,0,0), period=12))
m_sarima6<- Arima(ts2, order = c(1,1,1), seasonal = list(order=c(1,1,0), period=12))
# m_sarima7<- Arima(ts2, order = c(1,1,1), seasonal = list(order=c(1,2,0), period=12))
m_sarima8<- Arima(ts2, order = c(1,1,1), seasonal = list(order=c(0,2,1), period=12))
m_sarima9<- Arima(ts2, order = c(1,1,1), seasonal = list(order=c(1,2,3), period=12))

AIC<- c(m_sarima$aic, m_sarima2$aic, m_sarima4$aic, m_sarima5$aic, m_sarima6$aic, m_sarima8$aic, m_sarin

AIC
```

```
## [1] 936.8054 785.3913 947.3973 943.5624 785.7980 682.7522 678.7668
```

So, the model `SARIMA(1,1,1)(1,2,3)` is the best model as it has low AIC value.

SARIMA(1,1,1)(1,2,3)

```r
summary(m_sarima9)
```

```
## Series: ts2
## ARIMA(1,1,1)(1,2,3)[12]
##
## Coefficients:
```

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

```
##          ar1      ma1     sar1     sma1    sma2     sma3
##       0.7621  -0.9307  -0.6054  -0.6609  0.1982  -0.0236
## s.e.  0.2071   0.1875   0.1479      NaN  0.4979      NaN
##
## sigma^2 = 50758:  log likelihood = -332.38
## AIC=678.77   AICc=681.64   BIC=691.72
##
## Training set error measures:
##                     ME     RMSE      MAE       MPE     MAPE      MASE
## Training set -18.04647 170.0116 97.77025 -1.159542 4.587605 0.4922974
##                     ACF1
## Training set -0.02447572
```

The model will be good enough if the residuals for the model be random in nature.
To test for the randommess of the residuals we would perform Ljung Box test here.

```
Box.test(m_sarima9$residuals,lag = 71,type="Ljung-Box")
```

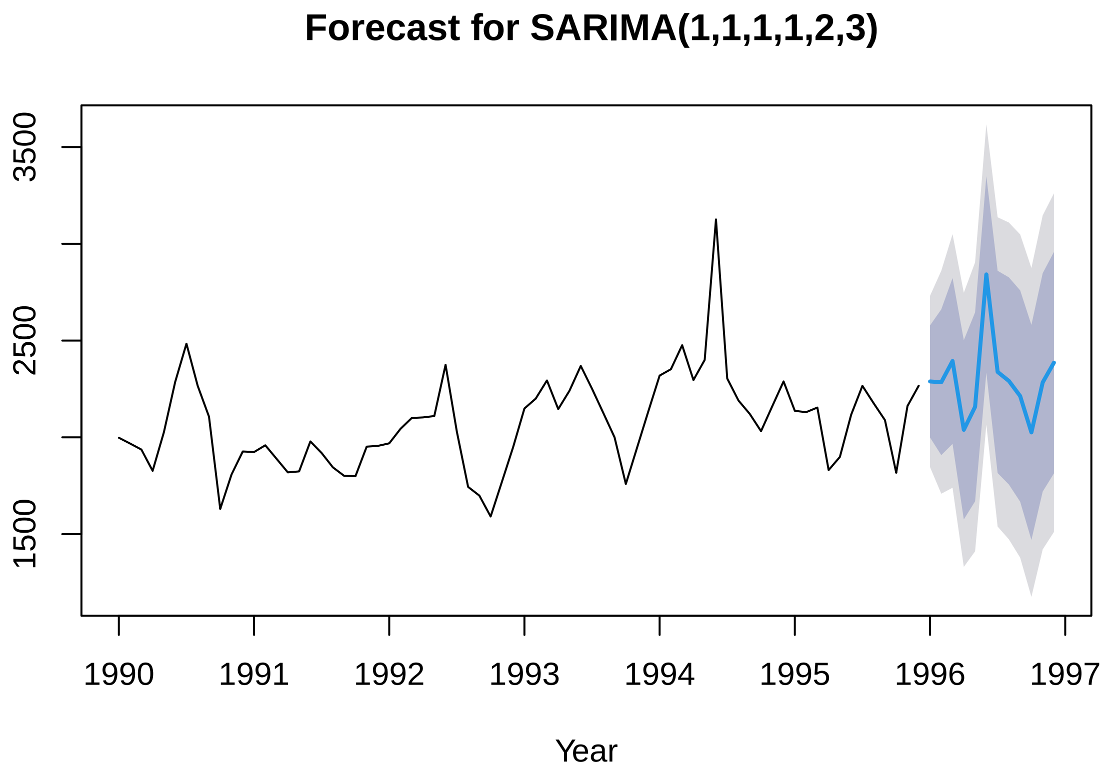
```
##
##  Box-Ljung test
##
## data:  m_sarima9$residuals
## X-squared = 40.353, df = 71, p-value = 0.9987
```

Comment: Here the high p value indicates that the residuals are random in nature so we can work with this model.

```
forecast(m_sarima9, h=12)
```

```
##          Point Forecast    Lo 80    Hi 80    Lo 95    Hi 95
## Jan 1996       2288.715 1999.395 2578.035 1846.238 2731.192
## Feb 1996       2284.679 1908.151 2661.207 1708.828 2860.530
## Mar 1996       2394.501 1966.195 2822.807 1739.463 3049.539
## Apr 1996       2038.889 1575.843 2501.935 1330.722 2747.056
## May 1996       2157.696 1669.583 2645.809 1411.191 2904.201
## Jun 1996       2841.821 2334.601 3349.042 2066.094 3617.549
## Jul 1996       2338.075 1815.615 2860.536 1539.041 3137.110
## Aug 1996       2291.437 1756.343 2826.531 1473.081 3109.793
## Sep 1996       2213.450 1667.530 2759.370 1378.538 3048.362
## Oct 1996       2025.364 1469.904 2580.824 1175.861 2874.867
## Nov 1996       2283.708 1719.639 2847.777 1421.039 3146.377
## Dec 1996       2385.916 1813.925 2957.906 1511.132 3260.699
```

```
plot(forecast(m_sarima9, h=12), main = " Forecast for SARIMA(1,1,1,1,2,3)", xlab = "Year")
```


Forecast for SARIMA(1,1,1,1,2,3)
3500
2500
1500
1990
1991
1992
1993
1994
1995
1996
1997
Year