Business Forecasting

Final Exam: Crime Data

Ronak Parikh

Submitted By: Deepti Khatri

Contents

Business Forecasting Final Exam

Introduction

Crime in US has been steadily decreasing over the years. In the US, FBI tracks crime data. Data is tracked by type of crime. For this exercise, we will focus on robberies, theft, and larceny. See https://www.fbi.gov/services/cjis/ucr for more detail.

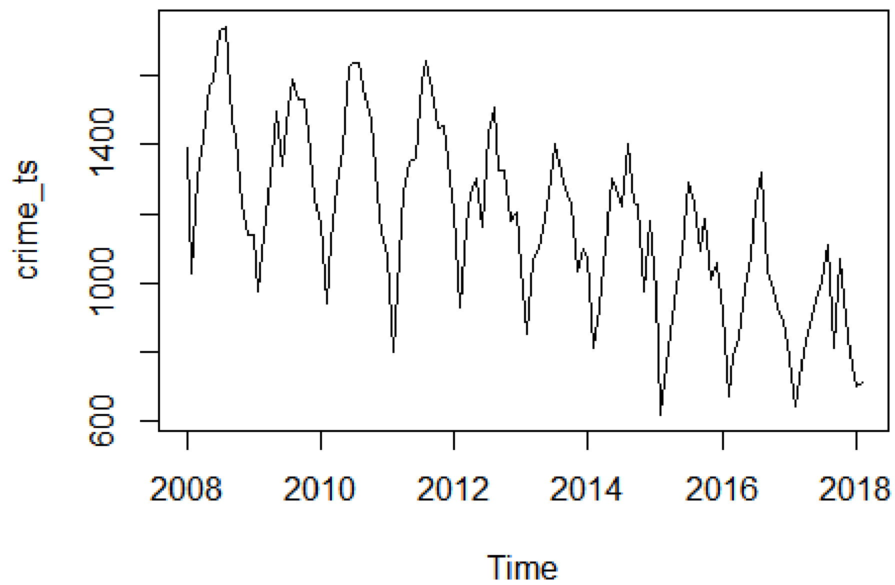
Libraries:-

```
library(fpp)

## Loading required package: forecast

## Loading required package: fma

## Loading required package: expsmooth

## Loading required package: lmtest

## Loading required package: zoo
```

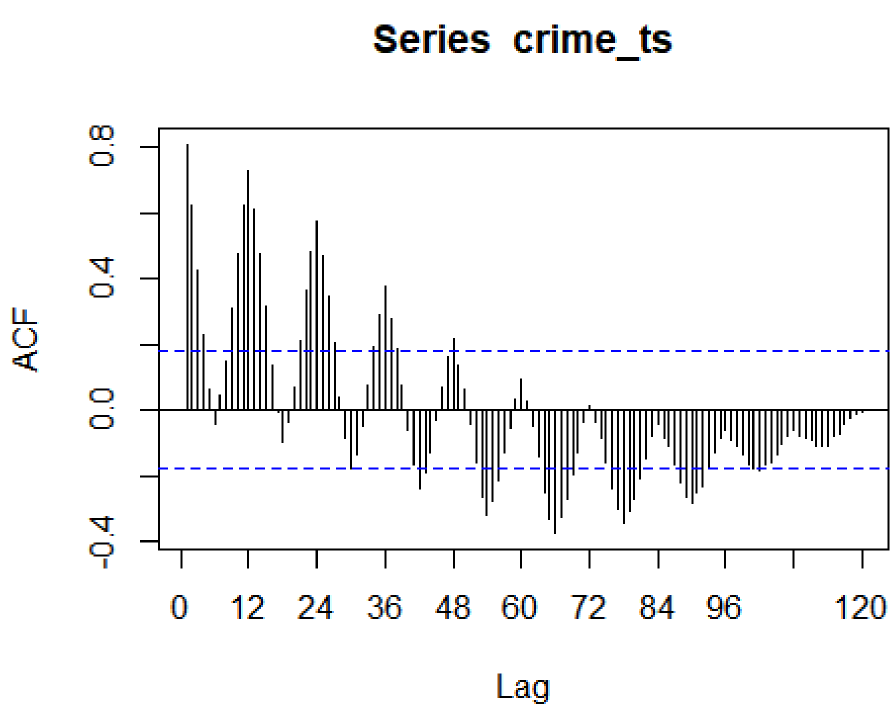
```
##
## Attaching package: 'zoo'

## The following objects are masked from 'package:base':
##
##     as.Date, as.Date.numeric

## Loading required package: tseries

library(fpp2)

## Loading required package: ggplot2

##
## Attaching package: 'fpp2'

## The following objects are masked from 'package:fpp':
##
##     ausair, ausbeer, austa, austourists, debitcards, departures,
##     elecequip, euretail, guinearice, oil, sunspotarea, usmelec

library(TTR)
library(forecast)
```

Import Data

```
crime <- read.csv("C:/Users/deept/Downloads/Data_Fall_2018_Crimes.csv")
crime_ts <- ts(crime$Data, start=c(2008,1),frequency = 12)
crime_ts1=window(crime_ts,start=c(2012,1),end=c(2018,2))
crime_ts1

##       Jan  Feb  Mar  Apr  May  Jun  Jul  Aug  Sep  Oct  Nov  Dec
## 2012 1206  928 1162 1254 1301 1162 1435 1510 1328 1324 1180 1201
## 2013 1047  852 1066 1098 1156 1254 1401 1318 1272 1229 1034 1100
## 2014 1077  814  926 1106 1306 1277 1222 1400 1235 1226  976 1179
## 2015  976  619  763  901 1031 1095 1292 1235 1090 1187 1011 1056
## 2016  914  670  786  841  987 1075 1227 1320 1035  983  931  891
## 2017  785  639  724  837  900  968 1005 1112  811 1069  900  795
## 2018  699  710

plot(crime_ts)
```

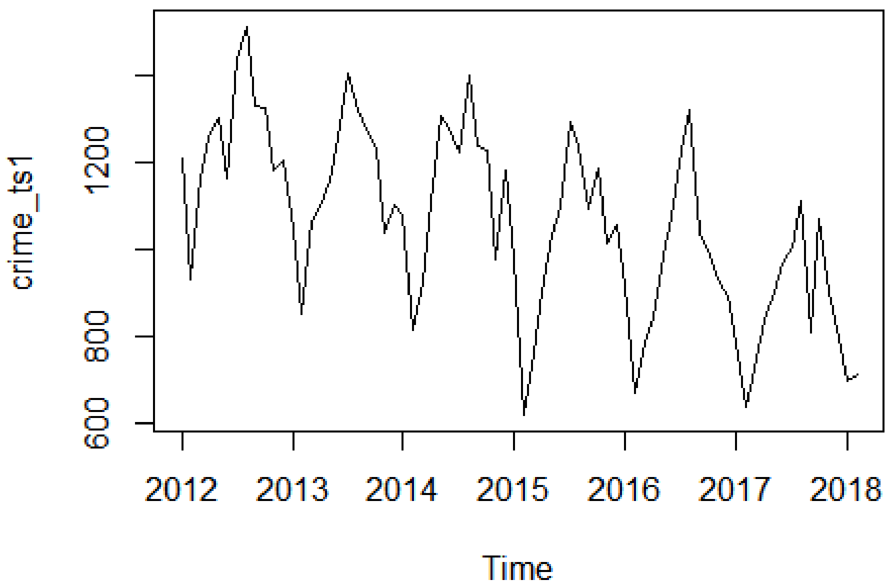



```
Acf(crime_ts,lag=120)
```

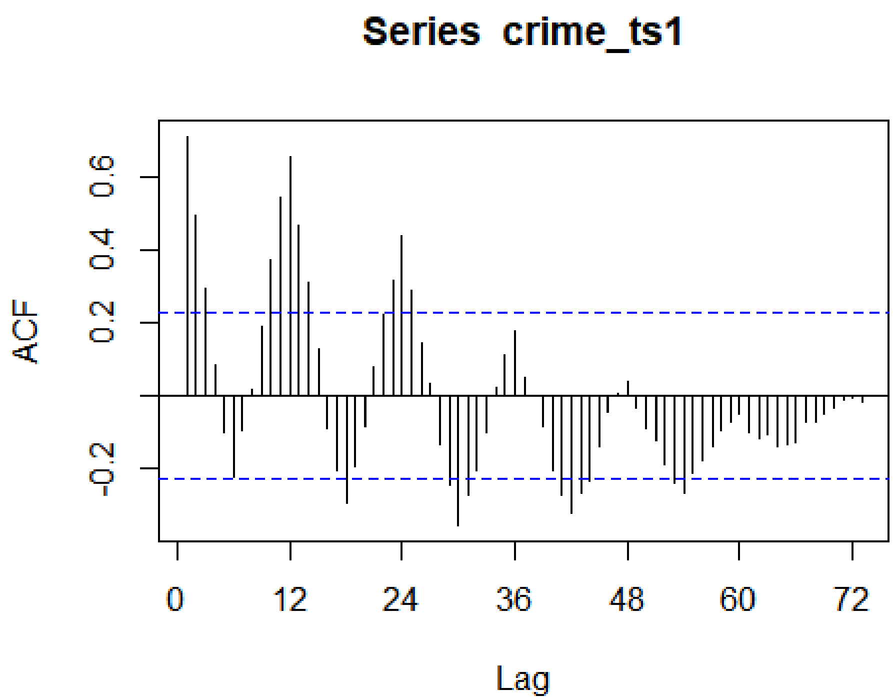


Plot and Inference

```r
#   Show a time series plot.
plot(crime_ts1)
```



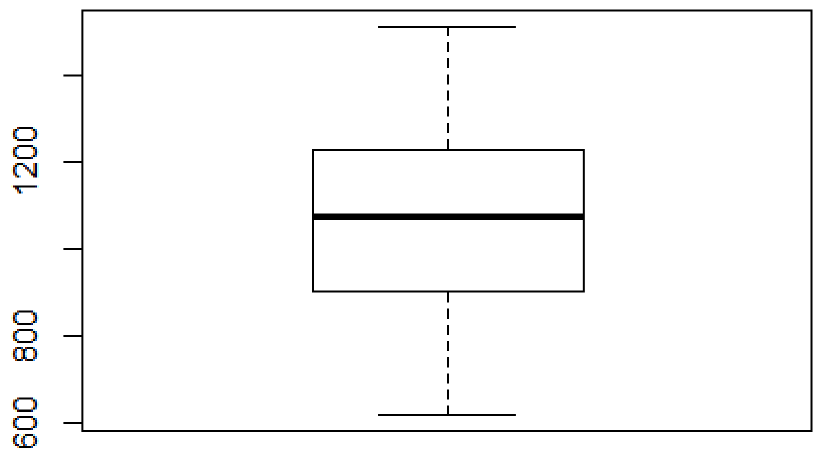

```r
Acf(crime_ts1,lag=74)
```



#There is seasonality in the data
#there were Seasonal peaks in Jan from year 2008 to 2014, afterward it is not visible
#Decreasing trend, crimes are decreased over the years

Central Tendency

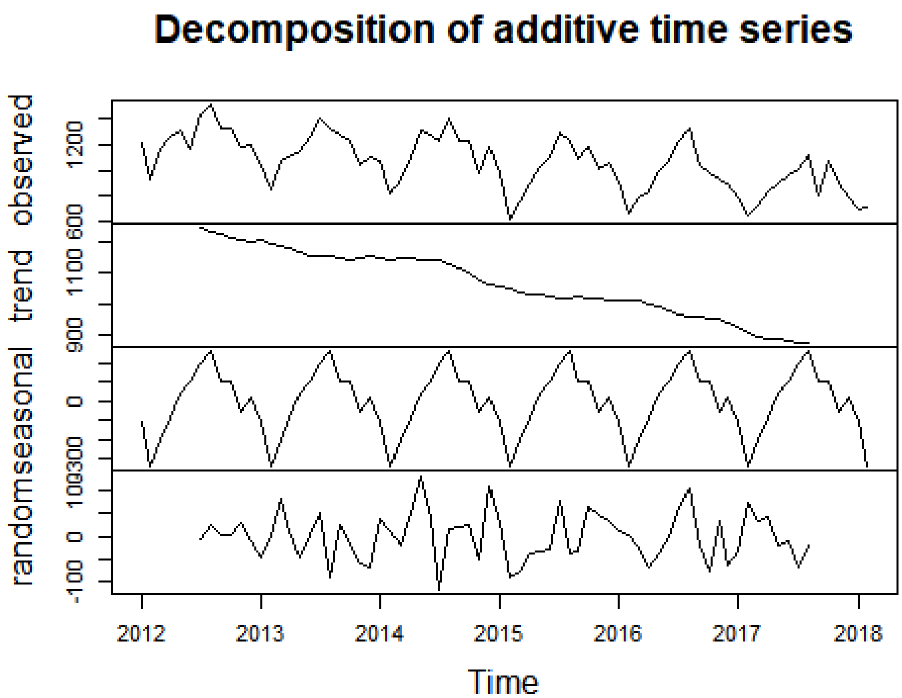
```
#   What are the min, max, mean, median, 1st and 3rd Quartile values of the
times series?

summary(crime_ts1)

##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max. 
##   619.0   904.2  1072.0  1059.9  1226.8  1510.0

#   Show the box plot.

boxplot(crime_ts1)
```




```
#Can you summarize your observation about the time series from the summary stats and box plot?
#On an average there happens 1072 crimes every year
#Either 619 crimes or 1510 crimes happens less probably
```

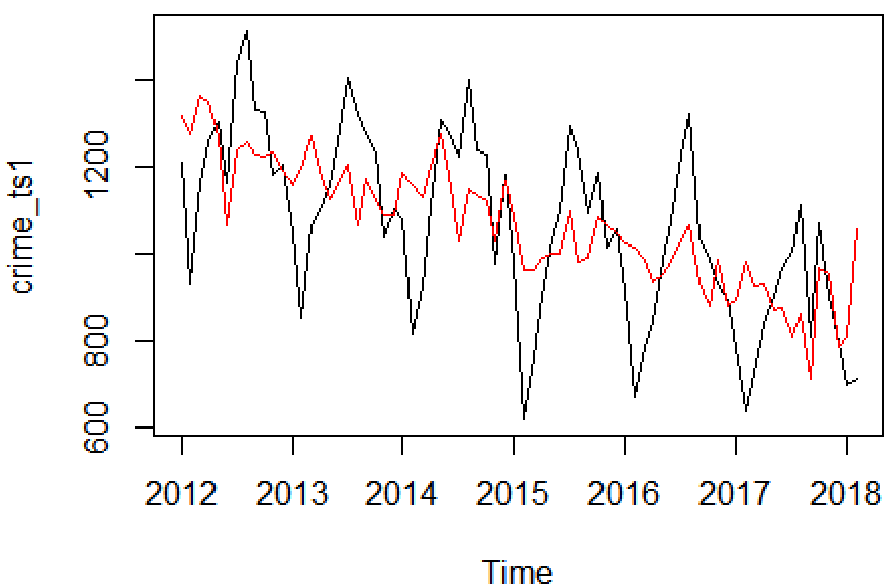
Decomposition

```
decompose_crimets1=decompose(crime_ts1)
plot(decompose_crimets1)
```




```
#Is the times series seasonal?
#Yes it is seasonal

#   Is the decomposition additive or multiplicative?
#Additive seasonal
decompose_crimets1$type

## [1] "additive"

#   If seasonal, what are the values of the seasonal monthly indices?
#Yes it is seasonal and following are Seasonal Indices
decompose_crimets1$figure

##  [1] -108.77627 -342.87627 -201.05127  -91.01794   32.84039   96.35706
##  [7]  197.23067  254.43206   99.32373  104.24873  -52.33461   11.62373

#For which month is the value of time series high and for which month is it l
ow?
#August Highest and Feb lowest
```

```
#Can you think of the reason behind the value being high in those months and low in those months?

Reason is the weather, in August weather is amazing, many people rome around, many tourists comes, so more the people higher is the crime rate
In febraury it is the coldest month most of the people remain inside, they co me out only for urgent work and get in quickly so less chances of crime
Therefore crime rate is highest in August and Lowest in Febraury.


#   Show the plot for time series adjusted for seasonality. Overlay this with the line for actual time series?
Does seasonality have big fluctuations to the value of time series?
seasonal_adj_crime=seasadj(decompose_crimets1)
plot(crime_ts1)
lines(seasonal_adj_crime,col='red')
```

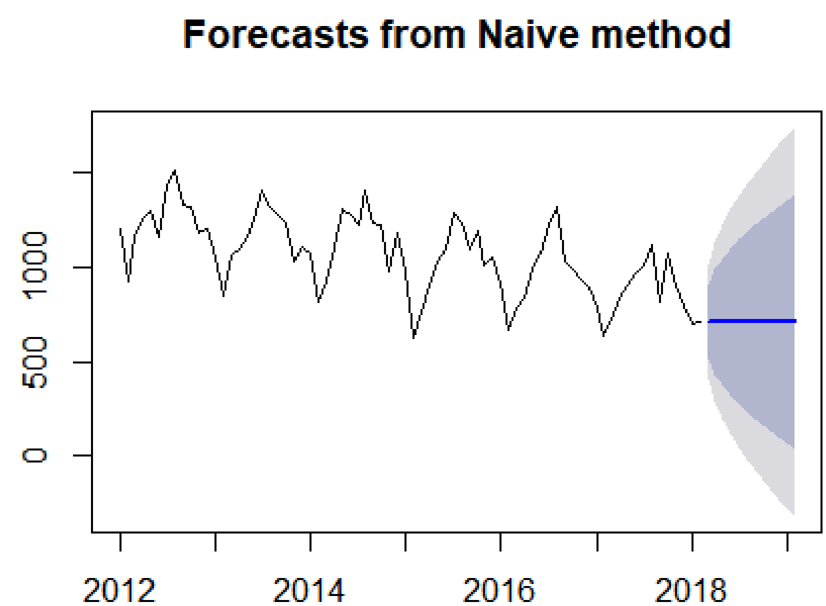



```
#Yes for seasonality has the big fluctuationis to the value of time series
```

Naive Forecast

#Output

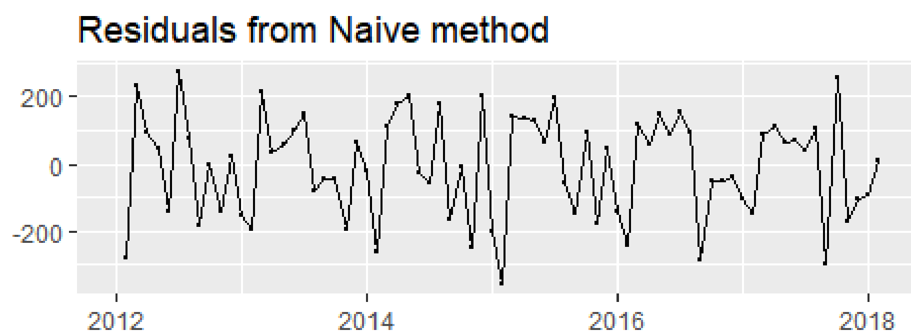
naive_forecast<-naive(crime_ts1,12)
plot(naive_forecast)

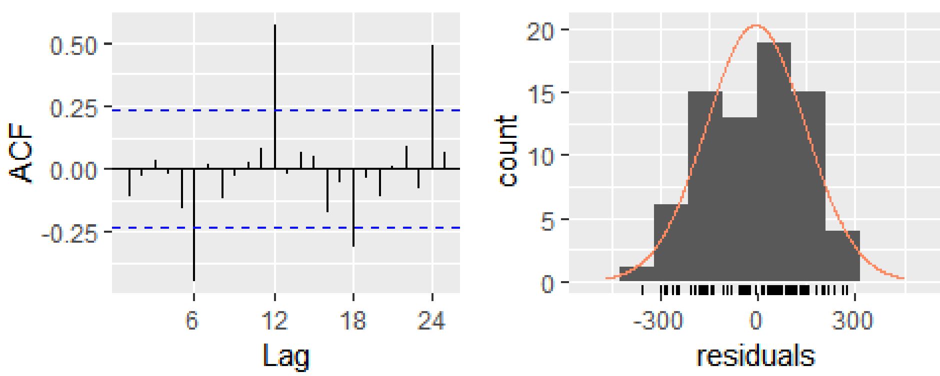



```
naive_forecast

##          Point Forecast     Lo 80     Hi 80      Lo 95    Hi 95
## Mar 2018            710 515.91615  904.0839  413.17436 1006.826
## Apr 2018            710 435.52398  984.4760  290.22515 1129.775
## May 2018            710 373.83690 1046.1631  195.88291 1224.117
## Jun 2018            710 321.83229 1098.1677  116.34872 1303.651
## Jul 2018            710 276.01531 1143.9847   46.27769 1373.722
## Aug 2018            710 234.59359 1185.4064  -17.07136 1437.071
```

```
## Sep 2018            710 196.50239 1223.4976  -75.32683 1495.327
## Oct 2018            710 161.04796 1258.9520 -129.54969 1549.550
## Nov 2018            710 127.74844 1292.2516 -180.47692 1600.477
## Dec 2018            710  96.25296 1323.7470 -228.64509 1648.645
## Jan 2019            710  66.29668 1353.7033 -274.45928 1694.459
## Feb 2019            710  37.67381 1382.3262 -318.23418 1738.234
```

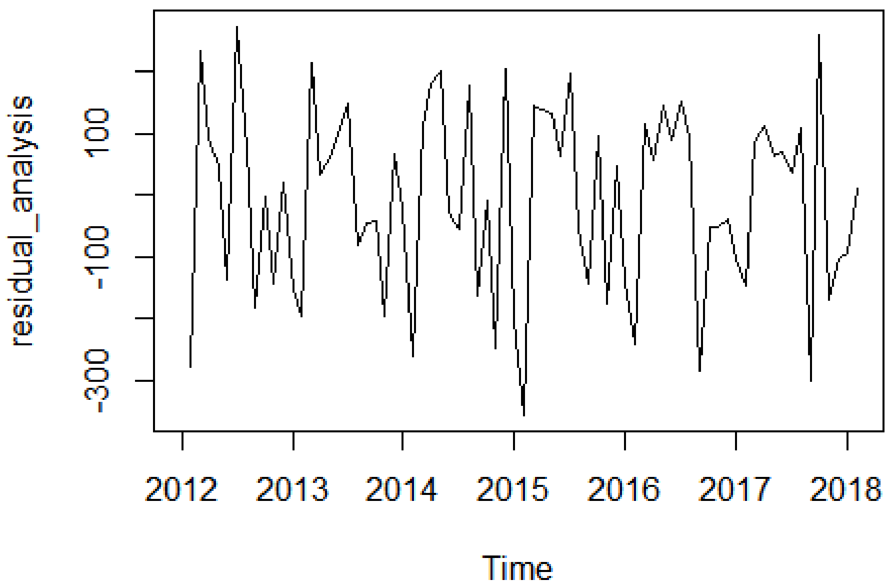
```
#   Perform Residual Analysis for this technique.
#   Do a plot of residuals. What does the plot indicate?
checkresiduals(naive_forecast)
```





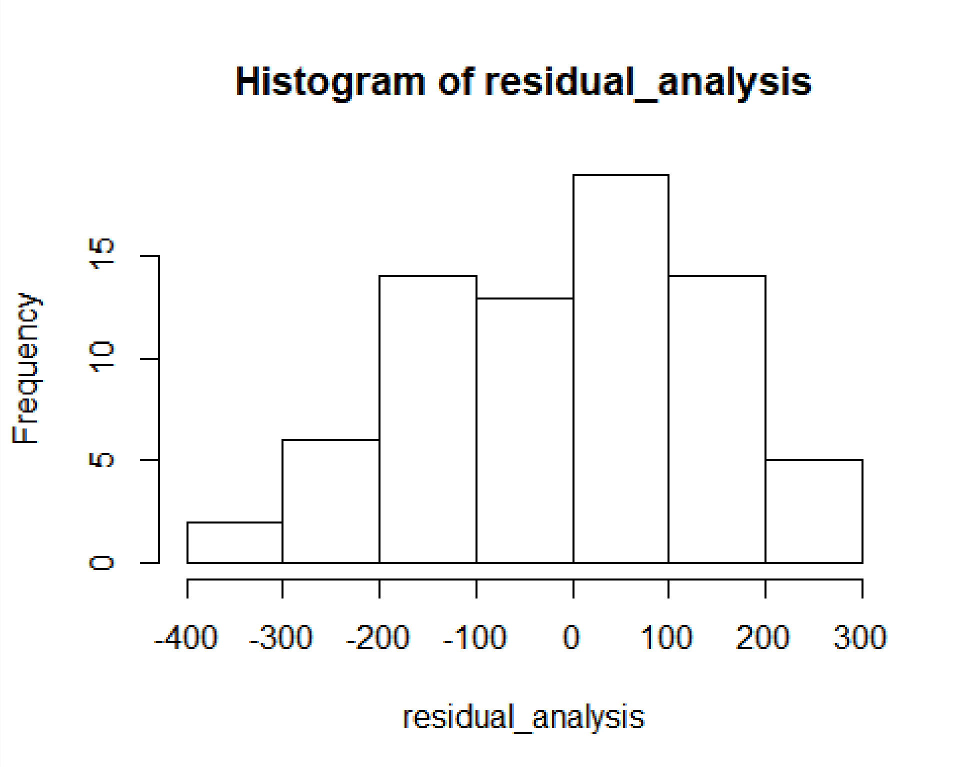

```
##
##  Ljung-Box test
##
## data:  Residuals from Naive method
## Q* = 50.58, df = 14.8, p-value = 8.463e-06
##
## Model df: 0.   Total lags used: 14.8
```

```
residual_analysis<-residuals(naive_forecast)
plot(residual_analysis)
```



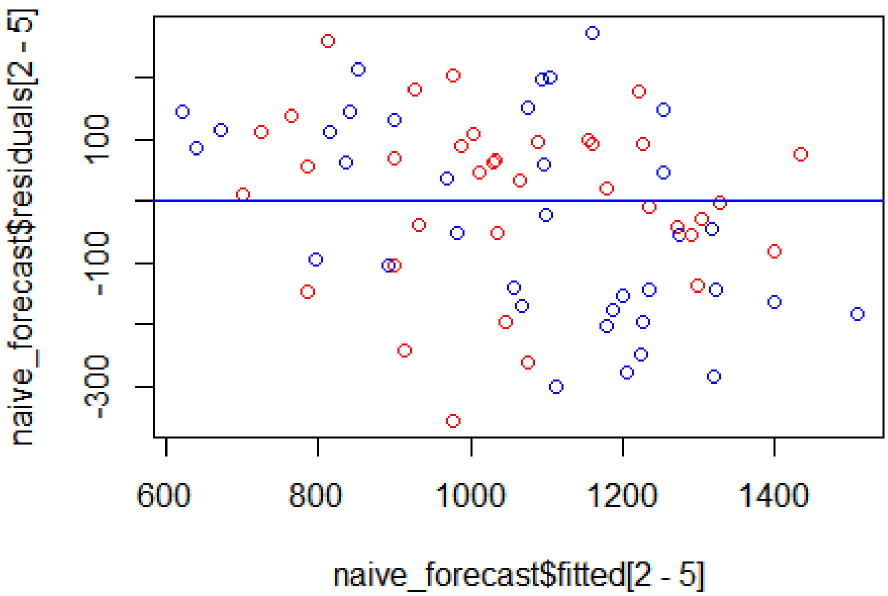

```
#There are highly significant values, as there are fluctuations it is not clo
se to 0
```

```r
#  Do a Histogram plot of residuals. What does the plot indicate?

hist(residual_analysis)
```



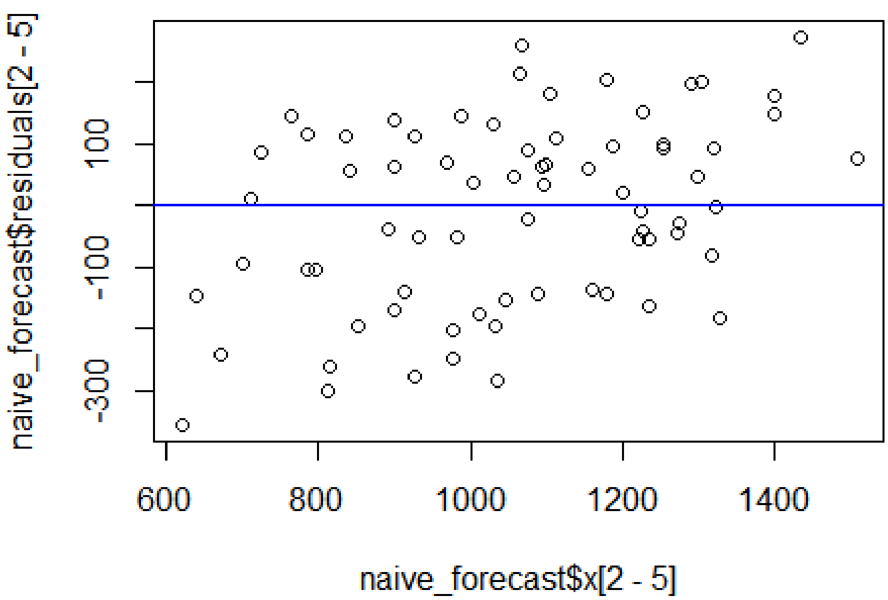

```r
#It is not normal, but skewed
```

```r
# Do a plot of fitted values vs. residuals. What does the plot indicate?

plot(naive_forecast$fitted[2-5],naive_forecast$residuals[2-5],col=c("red","bl
ue"))
abline(0,0,col='blue')
```



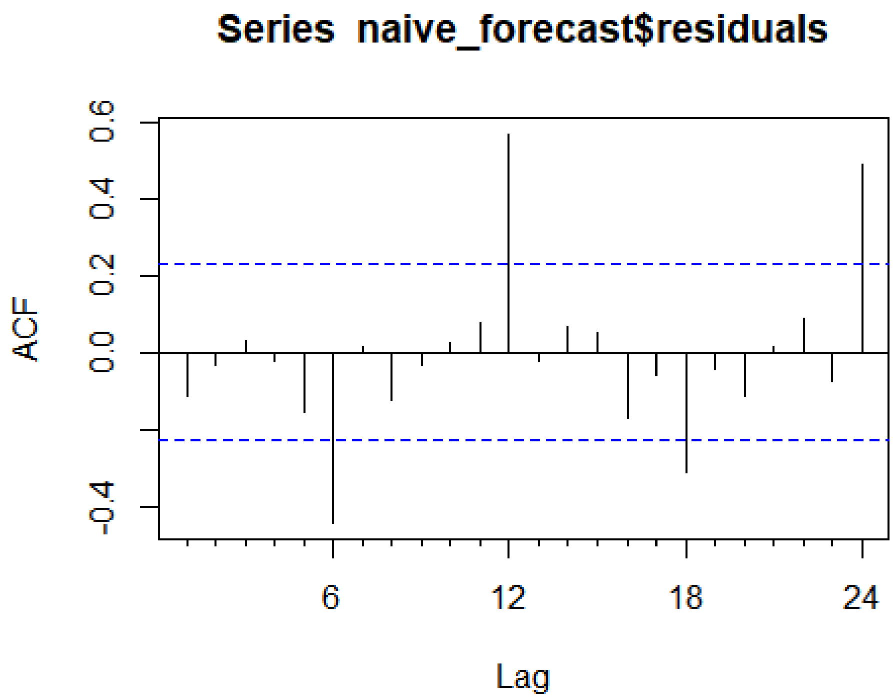

```r
#There is some pattern, error has some information
```

```r
  # Do a plot of actual values vs. residuals. What does the plot indicate?
attributes(naive_forecast)

## $names
##  [1] "method"    "model"     "level"     "mean"      "lower"
##  [6] "upper"     "x"         "series"    "fitted"    "residuals"
##
## $class
## [1] "forecast"

plot(naive_forecast$x[2-5],naive_forecast$residuals[2-5])
abline(0,0,col='blue')
```




```r
#there are many points above and below the mean line, some information is

left in the residual, residual is significant
```

```r
#   Do an ACF plot of the residuals? What does this plot indicate?

Acf(naive_forecast$residuals)
```

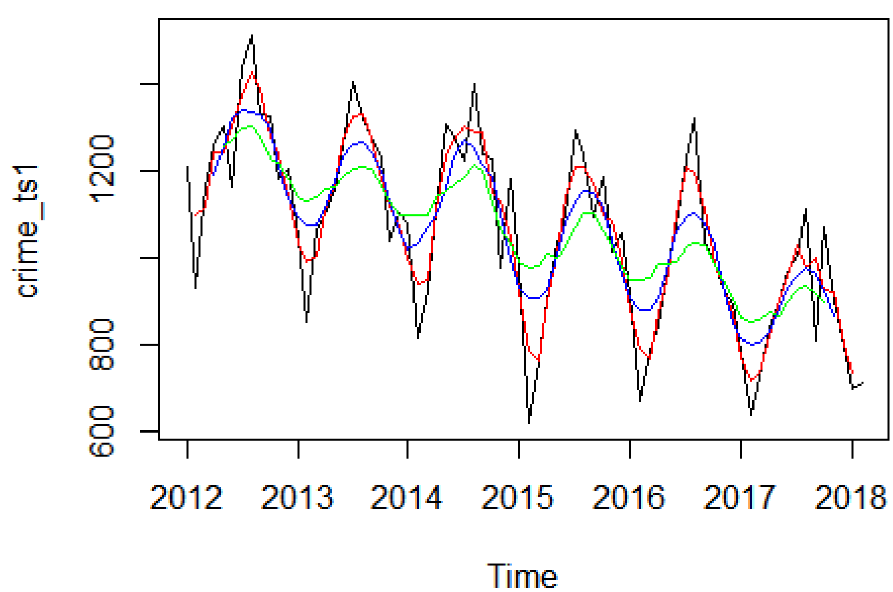



```r
# there are lag every 6 months, residual has some information left, this fore
casting method
#did not perform well


  # Print the 5 measures of accuracy for this forecasting technique
accuracy(naive_forecast)

##                     ME     RMSE      MAE       MPE     MAPE     MASE
## Training set -6.794521 151.4444 128.3562 -1.941344 13.03845 1.277176
##                    ACF1
## Training set -0.1100386

#   Forecast
#   Time series value for next year. Show table and plot
naive_forecast

##          Point Forecast     Lo 80     Hi 80     Lo 95    Hi 95
## Mar 2018            710 515.91615  904.0839  413.17436 1006.826
## Apr 2018            710 435.52398  984.4760  290.22515 1129.775
## May 2018            710 373.83690 1046.1631  195.88291 1224.117
## Jun 2018            710 321.83229 1098.1677  116.34872 1303.651
```

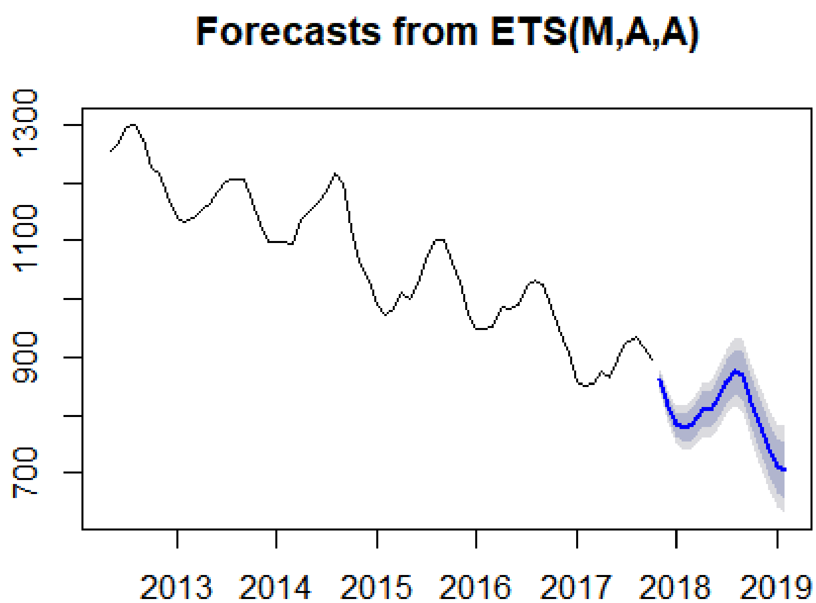
```
## Jul 2018           710 276.01531 1143.9847   46.27769 1373.722
## Aug 2018           710 234.59359 1185.4064  -17.07136 1437.071
## Sep 2018           710 196.50239 1223.4976  -75.32683 1495.327
## Oct 2018           710 161.04796 1258.9520 -129.54969 1549.550
## Nov 2018           710 127.74844 1292.2516 -180.47692 1600.477
## Dec 2018           710  96.25296 1323.7470 -228.64509 1648.645
## Jan 2019           710  66.29668 1353.7033 -274.45928 1694.459
## Feb 2019           710  37.67381 1382.3262 -318.23418 1738.234

#   Summarize this forecasting technique
#   How good is the accuracy?
 RMSE is 151.4444, which is high there could be better model

  # What does it predict the value of time series will be in one year?
#710 crimes in the coming year
naive_forecast

##          Point Forecast     Lo 80     Hi 80      Lo 95    Hi 95
## Mar 2018           710 515.91615  904.0839  413.17436 1006.826
## Apr 2018           710 435.52398  984.4760  290.22515 1129.775
## May 2018           710 373.83690 1046.1631  195.88291 1224.117
## Jun 2018           710 321.83229 1098.1677  116.34872 1303.651
## Jul 2018           710 276.01531 1143.9847   46.27769 1373.722
## Aug 2018           710 234.59359 1185.4064  -17.07136 1437.071
## Sep 2018           710 196.50239 1223.4976  -75.32683 1495.327
## Oct 2018           710 161.04796 1258.9520 -129.54969 1549.550
## Nov 2018           710 127.74844 1292.2516 -180.47692 1600.477
## Dec 2018           710  96.25296 1323.7470 -228.64509 1648.645
## Jan 2019           710  66.29668 1353.7033 -274.45928 1694.459
## Feb 2019           710  37.67381 1382.3262 -318.23418 1738.234

  # Other observation
#here it is showing the 710 crimes will happen next year, but this it not goo
d result as it is same as February, and above as mentioned February has lowes
t crime it is not going to be same in August or other months
```

Simple Moving Averages

```
#   Plot the graph for time series.
plot(crime_ts1)
#   Show the Simple Moving average of order 3 on the plot above in Red
ma3=ma(crime_ts1,order=3)
lines(ma3,col='RED')
#   Show the Simple Moving average of order 6 on the plot above in Blue
ma6=ma(crime_ts1,order=6)
lines(ma6,col='BLUE')

#   Show the Simple Moving average of order 9 on the plot above in Green
ma9=ma(crime_ts1,order=9)
lines(ma9,col='GREEN')
```



```
#(Bonus) show the forecast of next 12 months using one of the simple average
order that you feel works best for time series
ma_forecast=forecast(ma9,16)

## Warning in ets(object, lambda = lambda, biasadj = biasadj,
## allow.multiplicative.trend = allow.multiplicative.trend, : Missing values
## encountered. Using longest contiguous portion of time series

plot(ma_forecast)
```



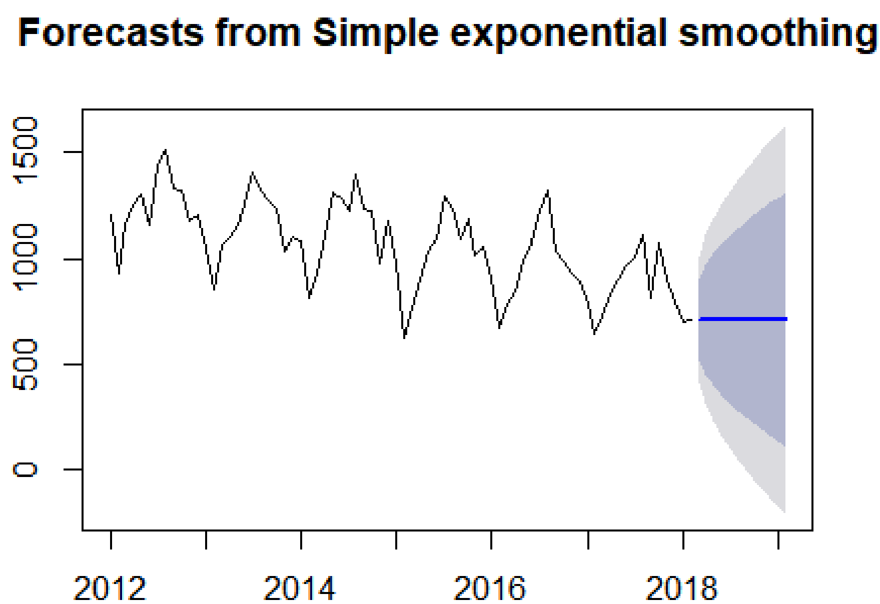

```
ma_forecast

##          Point Forecast    Lo 80    Hi 80    Lo 95    Hi 95
## Nov 2017       860.1857 847.3389 873.0325 840.5383 879.8332
## Dec 2017       815.2015 797.5091 832.8939 788.1434 842.2596
## Jan 2018       785.3394 764.1144 806.5644 752.8786 817.8003
## Feb 2018       778.2964 754.0959 802.4970 741.2849 815.3080
## Mar 2018       784.4370 757.5467 811.3273 743.3119 825.5622
## Apr 2018       810.6457 781.1508 840.1405 765.5372 855.7541
## May 2018       809.9477 778.0628 841.8327 761.1839 858.7115
## Jun 2018       830.1862 795.9680 864.4043 777.8541 882.5183
## Jul 2018       857.7419 821.1961 894.2878 801.8499 913.6339
## Aug 2018       876.1643 837.3371 914.9914 816.7832 935.5453
## Sep 2018       869.1390 828.1891 910.0889 806.5115 931.7665
## Oct 2018       824.2040 781.4323 866.9757 758.7904 889.6176
## Nov 2018       787.8372 743.4596 832.2148 719.9675 855.7068
```

```
## Dec 2018       742.8530 697.0956 788.6103 672.8732 812.8327
## Jan 2019       712.9909 665.9957 759.9861 641.1179 784.8638
## Feb 2019       705.9479 657.7683 754.1275 632.2636 779.6322
```

```
#   What are your observations of the plot as the moving average order goes u
p?
  #As the order goes up, line becomes smooth, better chances of good forecast
```

Smoothing

```
#   Perform a smoothing forecast for next 12 months for the time series.

ses_crime=ses(crime_ts1,12)
plot(ses_crime)
```



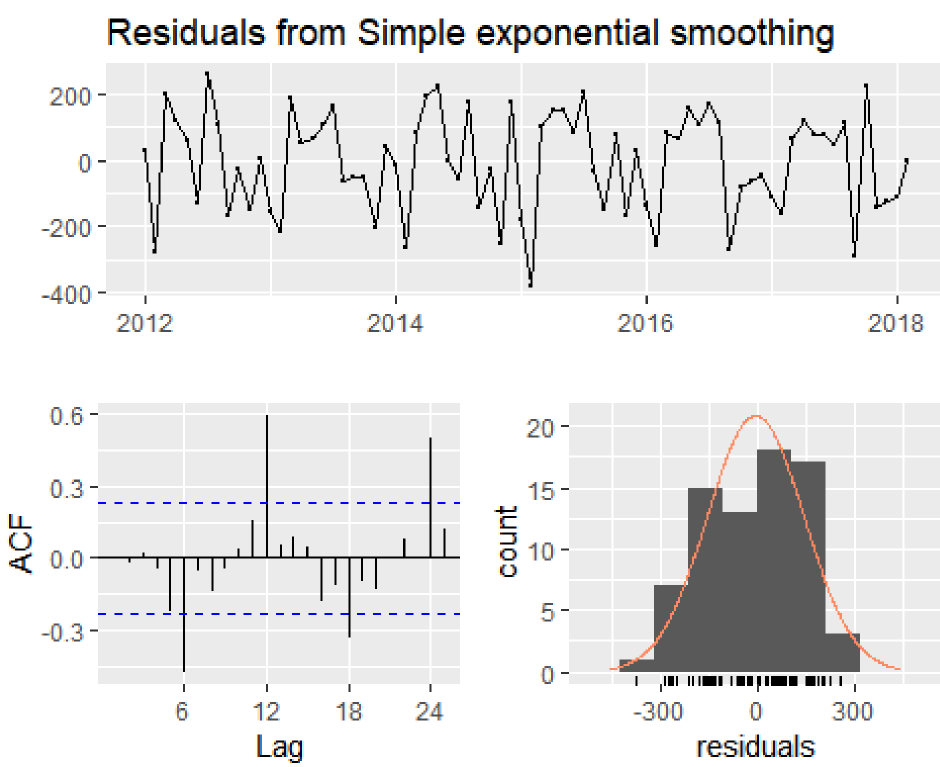

```
ses_crime

##           Point Forecast    Lo 80     Hi 80      Lo 95    Hi 95
## Mar 2018       710.2906 516.1634  904.4177 413.398695 1007.182
## Apr 2018       710.2906 451.8256  968.7555 315.002518 1105.579
## May 2018       710.2906 400.5763 1020.0048 236.623489 1183.958
## Jun 2018       710.2906 356.6782 1063.9029 169.487128 1251.094
## Jul 2018       710.2906 317.6578 1102.9233 109.810577 1310.771
## Aug 2018       710.2906 282.1793 1138.4019  55.550865 1365.030
```

```
## Sep 2018       710.2906 249.4239 1171.1572    5.455879 1415.125
## Oct 2018       710.2906 218.8469 1201.7342  -41.307595 1461.889
## Nov 2018       710.2906 190.0641 1230.5171  -85.327215 1505.908
## Dec 2018       710.2906 162.7923 1257.7889 -127.035842 1547.617
## Jan 2019       710.2906 136.8159 1283.7652 -166.763237 1587.344
## Feb 2019       710.2906 111.9663 1308.6149 -204.767483 1625.349
```

```
summary(ses_crime)
```

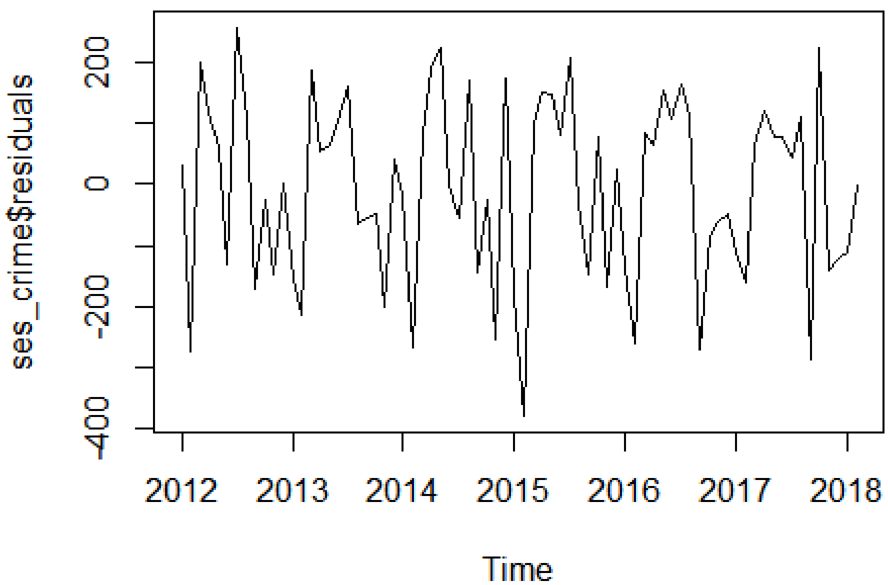
```
## 
## Forecast method: Simple exponential smoothing
## 
## Model Information:
## Simple exponential smoothing 
## 
## Call:
##  ses(y = crime_ts1, h = 12) 
## 
##   Smoothing parameters:
##     alpha = 0.879 
## 
##   Initial states:
##     l = 1176.0159 
## 
##   sigma:  151.4782
## 
##      AIC     AICc      BIC 
## 1065.499 1065.842 1072.411 
## 
## Error measures:
##                     ME     RMSE      MAE       MPE     MAPE     MASE
## Training set -7.15975 149.4172 126.7911 -2.105581 12.88012 1.261603
##                      ACF1
## Training set 0.0002632171
## 
## Forecasts:
##          Point Forecast    Lo 80     Hi 80       Lo 95    Hi 95
## Mar 2018       710.2906 516.1634  904.4177  413.398695 1007.182
## Apr 2018       710.2906 451.8256  968.7555  315.002518 1105.579
## May 2018       710.2906 400.5763 1020.0048  236.623489 1183.958
## Jun 2018       710.2906 356.6782 1063.9029  169.487128 1251.094
## Jul 2018       710.2906 317.6578 1102.9233  109.810577 1310.771
## Aug 2018       710.2906 282.1793 1138.4019   55.550865 1365.030
## Sep 2018       710.2906 249.4239 1171.1572    5.455879 1415.125
## Oct 2018       710.2906 218.8469 1201.7342  -41.307595 1461.889
## Nov 2018       710.2906 190.0641 1230.5171  -85.327215 1505.908
## Dec 2018       710.2906 162.7923 1257.7889 -127.035842 1547.617
## Jan 2019       710.2906 136.8159 1283.7652 -166.763237 1587.344
## Feb 2019       710.2906 111.9663 1308.6149 -204.767483 1625.349
```

```
  # What is the value of alpha?  What does that value signify?
 #alpha = 0.879 it signifies the optimal smoothing parameter for the model to
get minimum error

  # What is the value of initial state?
      #Initial states:
      # l = 1176.0159
  # What is the value of sigma?  What does the sigma signify?
      #sigma:  151.4782
      #signies the variation around the residual mean

  # Perform Residual Analysis for this technique.
checkresiduals(ses_crime)
```



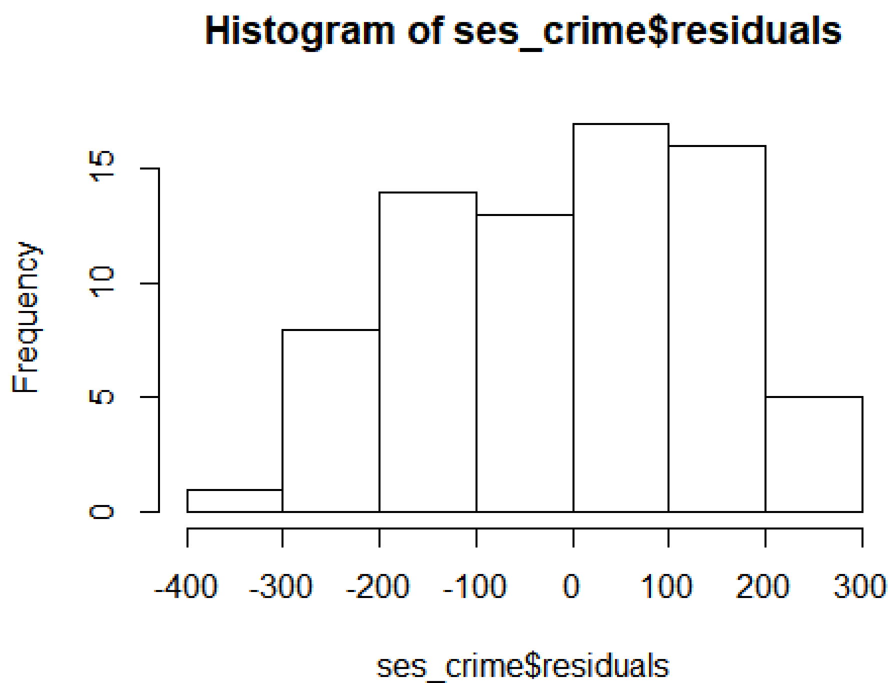

```
##
##  Ljung-Box test
##
## data:  Residuals from Simple exponential smoothing
## Q* = 59.161, df = 12.8, p-value = 6.293e-08
##
## Model df: 2.   Total lags used: 14.8
```

```r
# Do a plot of residuals. What does the plot indicate?
  plot(ses_crime$residuals)
```



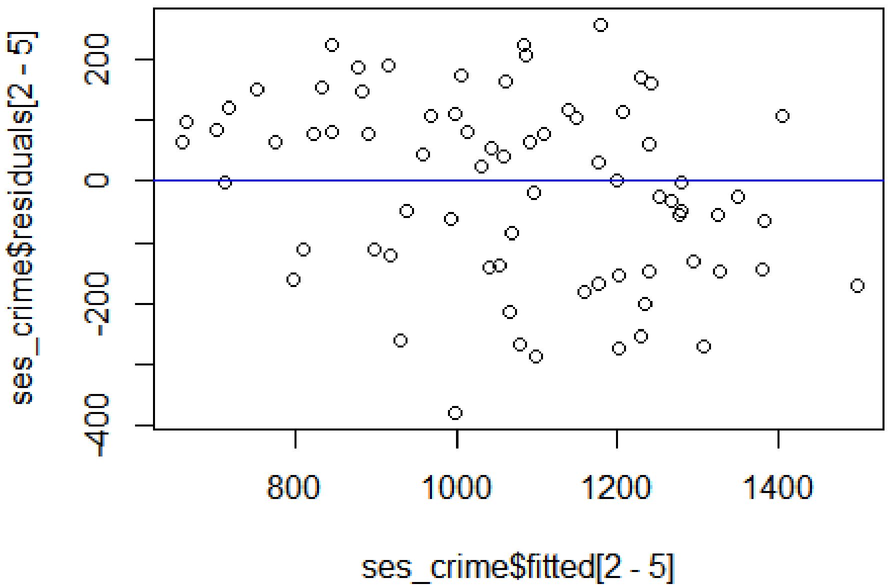

```r
  # The values highly fluctuating from 2012 onwards. Residuals should be clos
e to zero which indicates highly significant values.
```

```r
# Do a Histogram plot of residuals. What does the plot indicate?
hist(ses_crime$residuals)
```



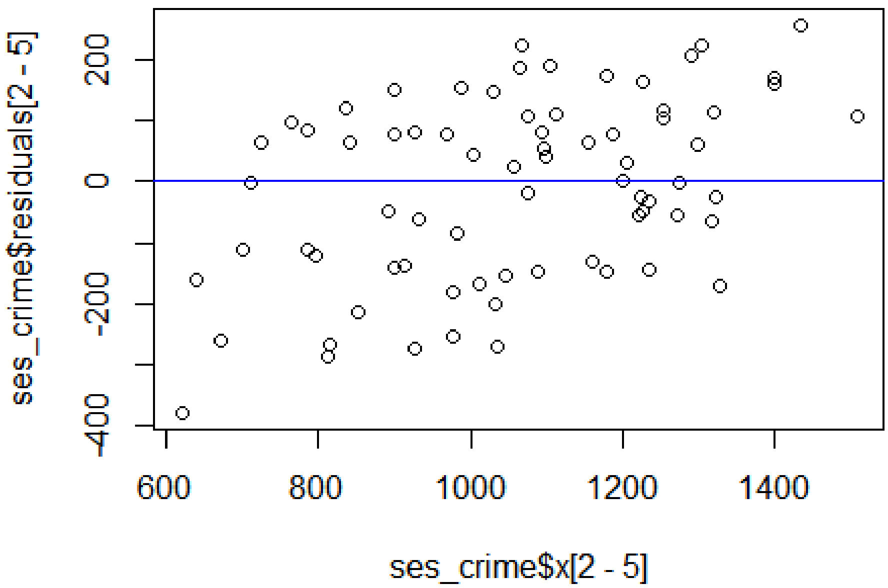

```r
#Histogram is not normal but skewed, indicates not a good forecast
```

```r
# Do a plot of fitted values vs. residuals. What does the plot indicate?
plot(ses_crime$fitted[2-5],ses_crime$residuals[2-5])
abline(0,0,col='blue')
```



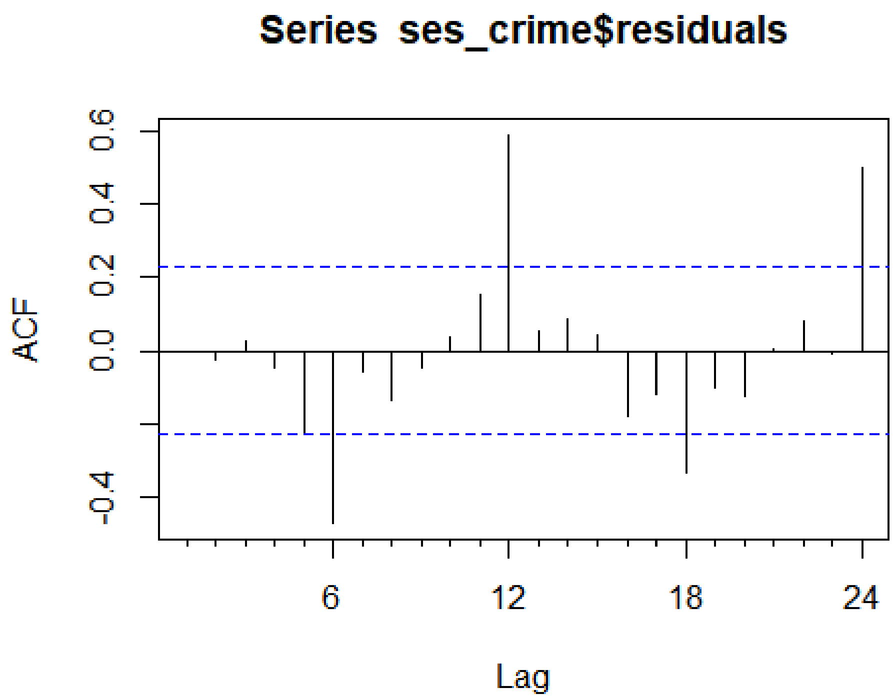

```r
#there are many points above mean line, thus there is a pattern which shows
that error component influences forecast model , there are information still
left in residual
```

```r
# Do a plot of actual values vs. residuals. What does the plot indicate?
plot(ses_crime$x[2-5],ses_crime$residuals[2-5])
abline(0,0,col='blue')
```




```r
#It shows a pattern, also there is leverage (many points at one place)
#information is still in the residual, which can be extracted with better m
odel
#Therefore we can say error component influences the forecast component
```

```r
# Do an ACF plot of the residuals? What does this plot indicate?
Acf(ses_crime$residuals)
```



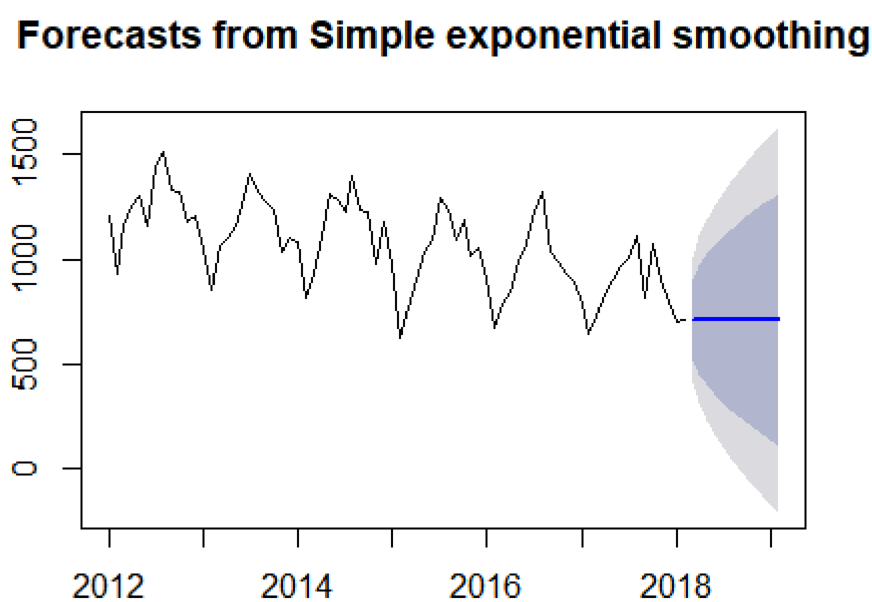

```r
showing the pattern still exists, there is lag every 6 months

# Print the 5 measures of accuracy for this forecasting technique
accuracy(ses_crime)
```

```
##                     ME     RMSE      MAE       MPE     MAPE     MASE
## Training set -7.15975 149.4172 126.7911 -2.105581 12.88012 1.261603
##                      ACF1
## Training set 0.0002632171
```

```r
# Forecast
 # Time series value for next year. Show table and plot
 ses_crime
```

```
##          Point Forecast    Lo 80     Hi 80       Lo 95    Hi 95
## Mar 2018       710.2906 516.1634  904.4177  413.398695 1007.182
## Apr 2018       710.2906 451.8256  968.7555  315.002518 1105.579
## May 2018       710.2906 400.5763 1020.0048  236.623489 1183.958
## Jun 2018       710.2906 356.6782 1063.9029  169.487128 1251.094
## Jul 2018       710.2906 317.6578 1102.9233  109.810577 1310.771
## Aug 2018       710.2906 282.1793 1138.4019   55.550865 1365.030
## Sep 2018       710.2906 249.4239 1171.1572    5.455879 1415.125
## Oct 2018       710.2906 218.8469 1201.7342  -41.307595 1461.889
## Nov 2018       710.2906 190.0641 1230.5171  -85.327215 1505.908
## Dec 2018       710.2906 162.7923 1257.7889 -127.035842 1547.617
## Jan 2019       710.2906 136.8159 1283.7652 -166.763237 1587.344
## Feb 2019       710.2906 111.9663 1308.6149 -204.767483 1625.349
```

```r
 plot(ses_crime)
```

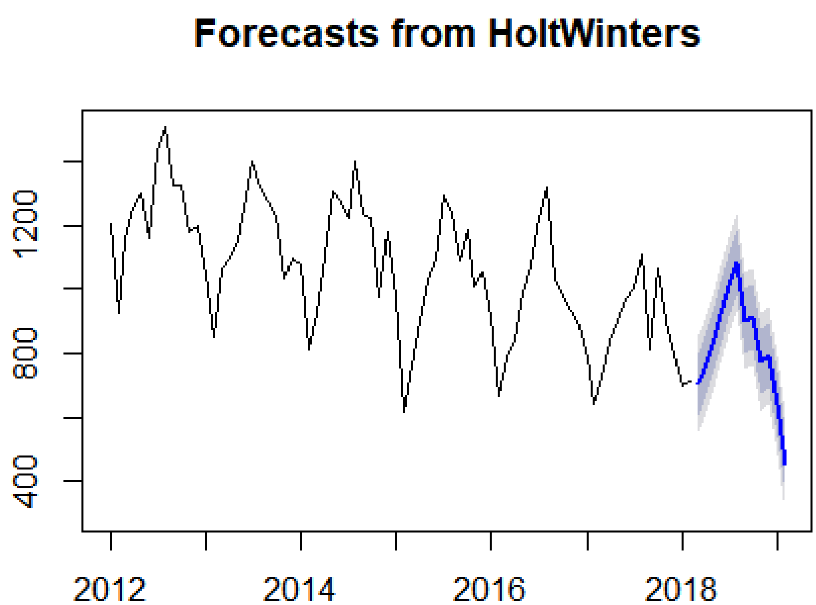



```
# Summarize this forecasting technique
#This is not efficient as proved by residuals analysis done above, there coul
d be better forecasting model.

# How good is the accuracy?
RMSE 149.4172, which is lower than naïve forecast, but it is still high and r
esidual analysis also shows it is not a good model
```

```r
# What does it predict the value of time series will be in one year?
  # for the next number of crimes will be 710.2906

# Other observation
  #it is better than naive bayes, as accuracy is higher
```

Holt-Winters

```r
#.    Perform Holt-Winters forecast for next 12 months for the time series.


hw_crime=HoltWinters(crime_ts1)

hw_crime_forecast=forecast(hw_crime,h=12)
plot(hw_crime_forecast)
```



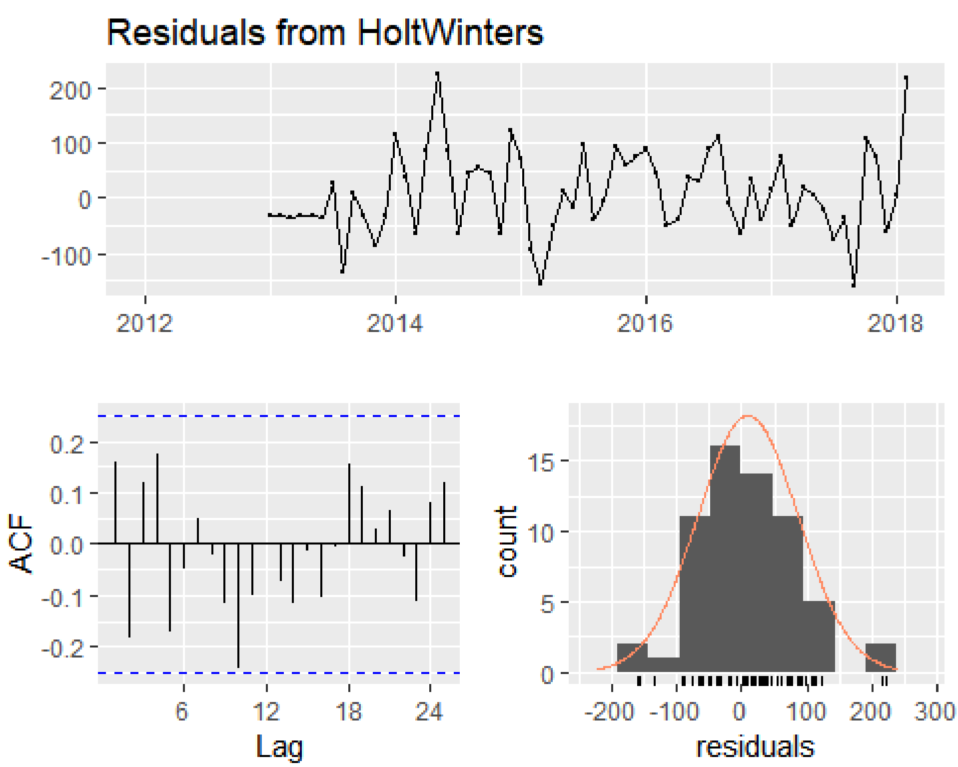

```r
  hw_crime

## Holt-Winters exponential smoothing with trend and additive seasonal compon
ent.
##
## Call:
```

```
## HoltWinters(x = crime_ts1)
##
## Smoothing parameters:
##  alpha: 0.04067075
##  beta : 0.08400935
##  gamma: 0.05016539
##
## Coefficients:
##            [,1]
## a    846.434014
## b     -5.287936
## s1  -138.703176
## s2   -83.911597
## s3    -3.206952
## s4    95.236618
## s5   191.609532
## s6   270.244682
## s7    92.640615
## s8   111.639168
## s9   -26.166524
## s10   -0.521382
## s11 -147.620238
## s12 -333.760689
```

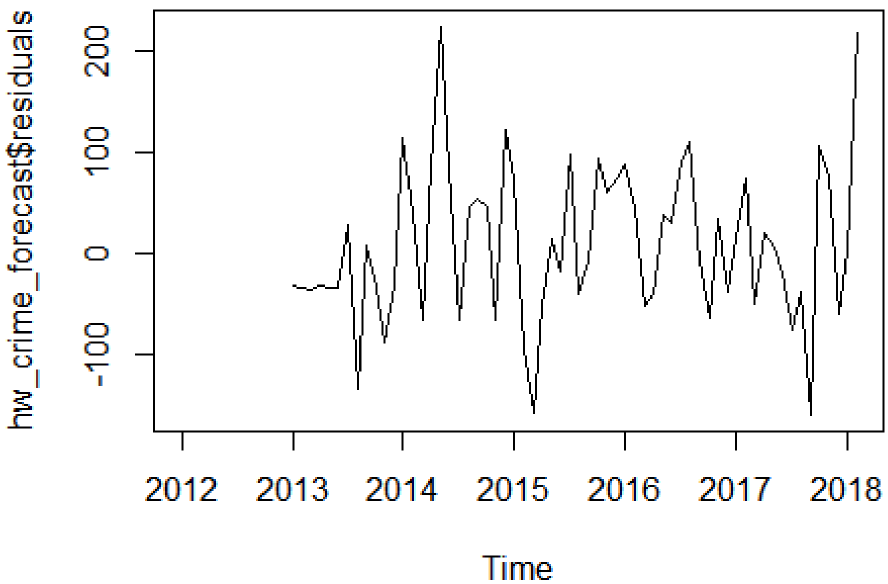
```
#What is the value of alpha?  What does that value signify?
#Value of alpha: 0.0406707, signifies level reacts to backdated observations
(in case if it close to 1, we say more weights are given to recent observatio
ns but it's not the case here)

#What is the value of beta? What does that value signify?
#Value of beta: 0.08400935, signifies trend depends on previous value

#What is the value of gamma? What does that value signify?
#Gamma is 0.05016539, signifies seasonality repeats according to cycle at reg
ular time period

#What is the value of initial states for the level, trend and seasonality? Wh
at do these values signify?
#a is level, b is trend, si to s12 is seasonality for 12 months respectively
  hw_crime$coefficients
```

```
##           a           b          s1          s2          s3          s4
##  846.434014   -5.287936 -138.703176  -83.911597   -3.206952   95.236618
##          s5          s6          s7          s8          s9         s10
##  191.609532  270.244682   92.640615  111.639168  -26.166524   -0.521382
##         s11         s12
## -147.620238 -333.760689
```

```
#What is the value of sigma?  What does the sigma signify?

  sd(complete.cases(hw_crime_forecast$residuals))

## [1] 0.3711156

  # Value of sigma =0.3711156, signifies value of standard deviation


#.  Perform Residual Analysis for this technique.

  checkresiduals(hw_crime_forecast)
```



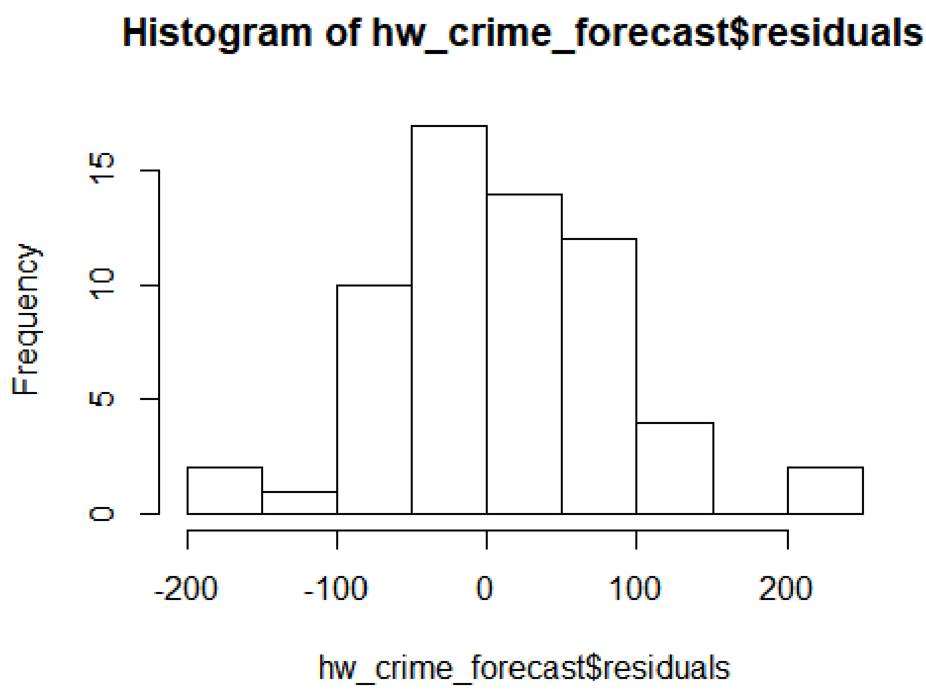
```r
#Do a plot of residuals. What does the plot indicate?
plot(hw_crime_forecast$residuals)
```




```r
#for year 2012 it is ok, but for rest it still has fluctuating values but it looks random

summary(hw_crime_forecast$residuals)
```

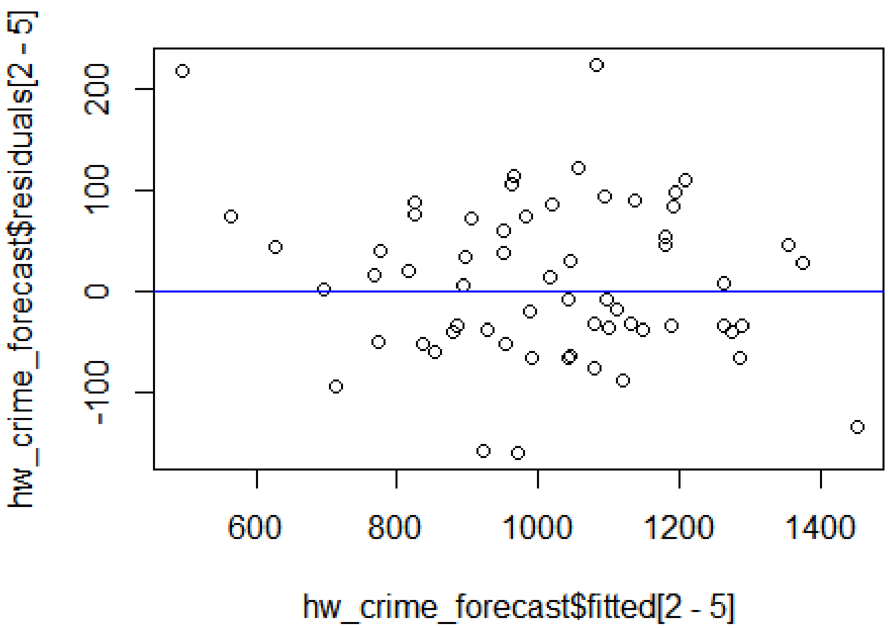
```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's 
## -158.73  -38.40    4.73    9.21   68.52  223.48      12
```

```r
#Do a Histogram plot of residuals. What does the plot indicate?
hist(hw_crime_forecast$residuals)
```



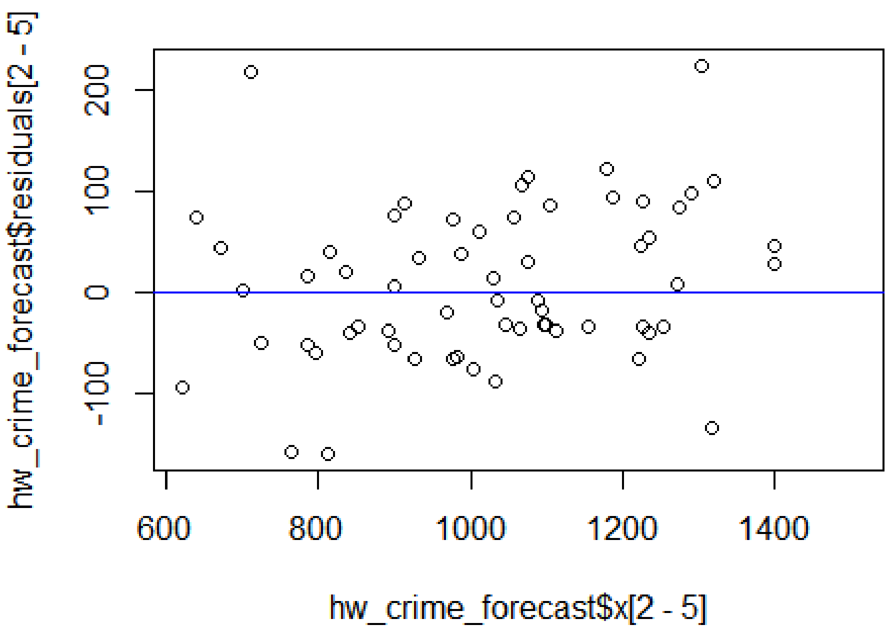

```
It is skewed, but as compared to other methods this is better
It has outliers
```

```r
#Do a plot of fitted values vs. residuals. What does the plot indicate?
 plot(hw_crime_forecast$fitted[2-5],hw_crime_forecast$residuals[2-5])
 abline(0,0,col='blue')
```



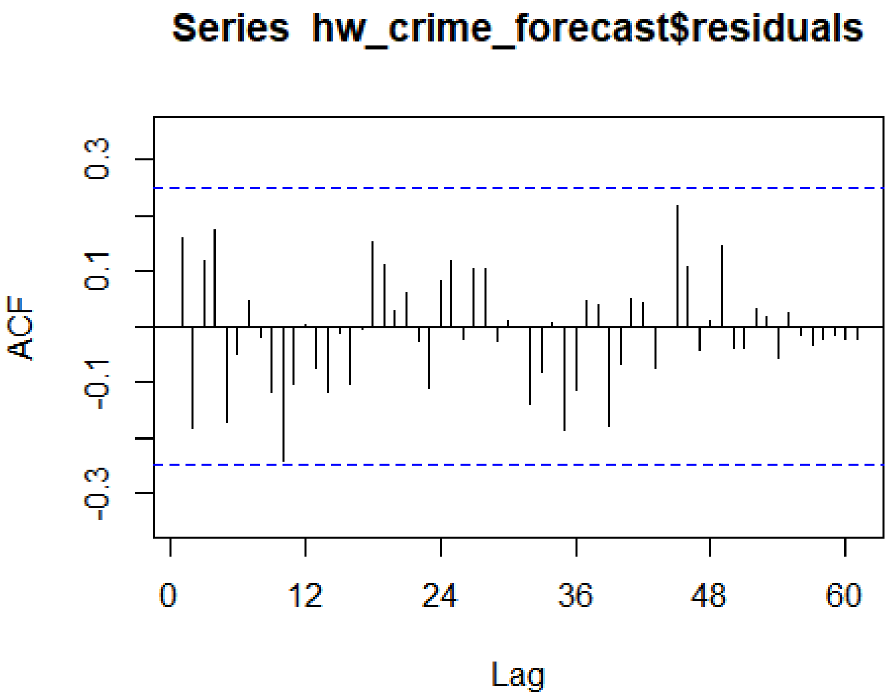

```r
#Variance is still there, it shows 5 outliers, some leverage,
#residual still has some significnce, possible there exists some method which can perform better
```

```r
#Do a plot of actual values vs. residuals. What does the plot indicate?
plot(hw_crime_forecast$x[2-5],hw_crime_forecast$residuals[2-5])
abline(0,0,col='blue')
```

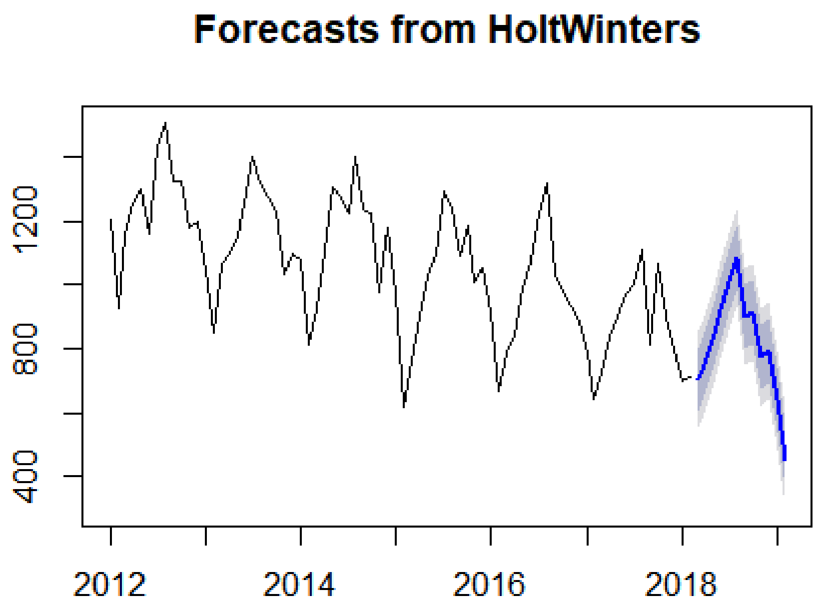



```r
#Variance is still there, it shows 5 outliers, some leverage,
#residual still has some significnce, possible there exists some method whi
ch can perform better
```

```
#Do an ACF plot of the residuals? What does this plot indicate?
 Acf(hw_crime_forecast$residuals,lag=74)
```




```
 #shows there is no autocorrelation (but one lag at 11, which is within limit
s), which shows it is good method of forecast

  #Print the 5 measures of accuracy for this forecasting technique
  accuracy(hw_crime_forecast)

##                    ME     RMSE      MAE       MPE     MAPE      MASE
## Training set 9.209882 77.38588 62.27083 0.6260094 6.368634 0.6196103
##                   ACF1
## Training set 0.1589406

  # Forecast

  # Time series value for next year. Show table and plot
  plot(hw_crime_forecast)
```




```
hw_crime_forecast

##          Point Forecast    Lo 80     Hi 80    Lo 95     Hi 95
## Mar 2018       702.4429 603.1699  801.7159 550.6180  854.2678
## Apr 2018       751.9465 652.5771  851.3160 599.9741  903.9190
## May 2018       827.3633 727.8820  926.8445 675.2198  979.5067
## Jun 2018       920.5189 820.9093 1020.1285 768.1791 1072.8586
## Jul 2018      1011.6039 911.8483 1111.3594 859.0409 1164.1668
## Aug 2018      1084.9511 985.0309 1184.8713 932.1363 1237.7658
## Sep 2018       902.0591 801.9545 1002.1636 748.9624 1055.1558
## Oct 2018       915.7697 815.4600 1016.0794 762.3593 1069.1801
## Nov 2018       772.6761 672.1395  873.2127 618.9186  926.4335
## Dec 2018       793.0333 692.2470  893.8196 638.8939  947.1726
## Jan 2019       640.6465 539.5867  741.7063 486.0889  795.2041
## Feb 2019       449.2181 347.8602  550.5760 294.2045  604.2317
```

#Following is the forecast for next 1 year
#Point Forecast
#Mar 2018 702.4429
#Apr 2018 751.9465
#May 2018 827.3633
#Jun 2018 920.5189
#Jul 2018 1011.6039
#Aug 2018 1084.9511
#Sep 2018 902.0591
#Oct 2018 915.7697
#Nov 2018 772.6761
#Dec 2018 793.0333
#Jan 2019 640.6465
#Feb 2019 449.2181

```
  # Summarize this forecasting technique
  #Holts Winter Forecast: It is built on simple smoothing forecast concept, here it has been adjusted for trend and seasonality both, which is done 2 scientists Holts and Winter. It comprises of forecast equation and 3 smoothing equations.
Smoothing equations involves calculation of level, trend and seasonality based on respective smoothing constant, which is calculated such that SSE should be minimum.
Once we have level, trend and seasonality, forecast model is built using forecast equation.
  Following are calculations
  - Forecast equation: Y^t+p = (Lt + p*Tt)*St-s+p
  - Level equation: Lt = @Yt/St-s + (1-@)(Lt-1 + T t-1)
  - Trend Equation: Tt = &(Lt-Lt-1) + (1-&)Tt-1
  - Seasonal Equation: !(Yt/Lt)+(1-!)St-s Where Lt = new smoothed Value @ = smoothing constant for level Yt = Actual forecast at time t & = Smoothing constant for trend Tt = trend estimate p = period for which to calculate forecast on Y^t+p = Forecast for p period into the future s = length of seasonality ! = Seasonality constant St = seasonality estimate.
It has advantage of simple smoothing and we have considered seasonality in the model, so it gives more accurate more forecast model.

  # How good is the accuracy?
 RMSE 77.38588, which is much lower than Naïve and simple smoothing, much higher accuracy


  # What does it predict the value of time series will be in one year?
  #Following is the forecast for next 1 year
  #Point Forecast
  #Mar 2018       702.4429
  #Apr 2018       751.9465
  #May 2018       827.3633
  #Jun 2018       920.5189
  #Jul 2018      1011.6039
  #Aug 2018      1084.9511
```

```
  #Sep 2018       902.0591
  #Oct 2018       915.7697
  #Nov 2018       772.6761
  #Dec 2018       793.0333
  #Jan 2019       640.6465
  #Feb 2019       449.2181
```

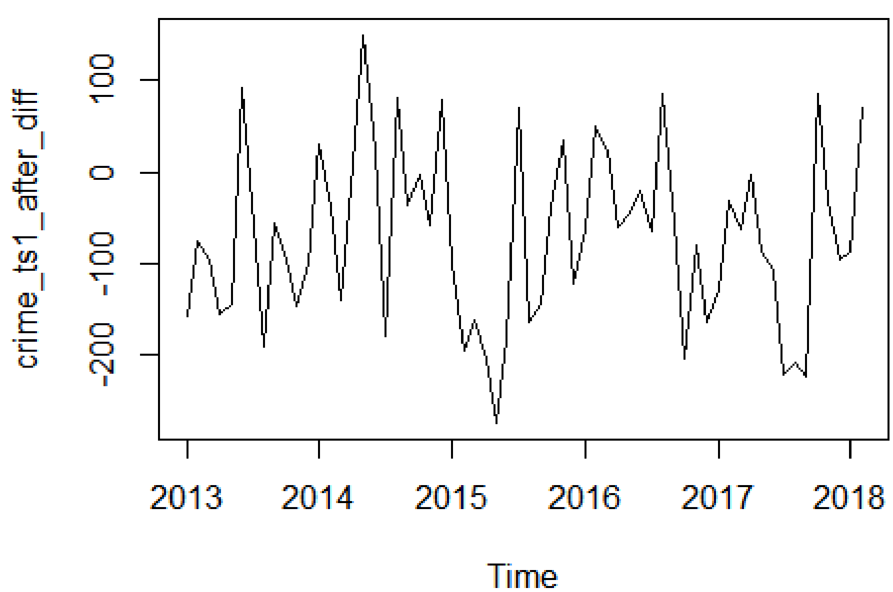
```
  # Other observation
 This is the better model than Naive, Simple smoothing
  Because of 2 reasons
  1. Acf plot of residuals show residuals is insignificant
  2. When we look at values of forecast for next 12 months, it shows high
in August and low in February which is matching our data
```

ARIMA or Box-Jenkins

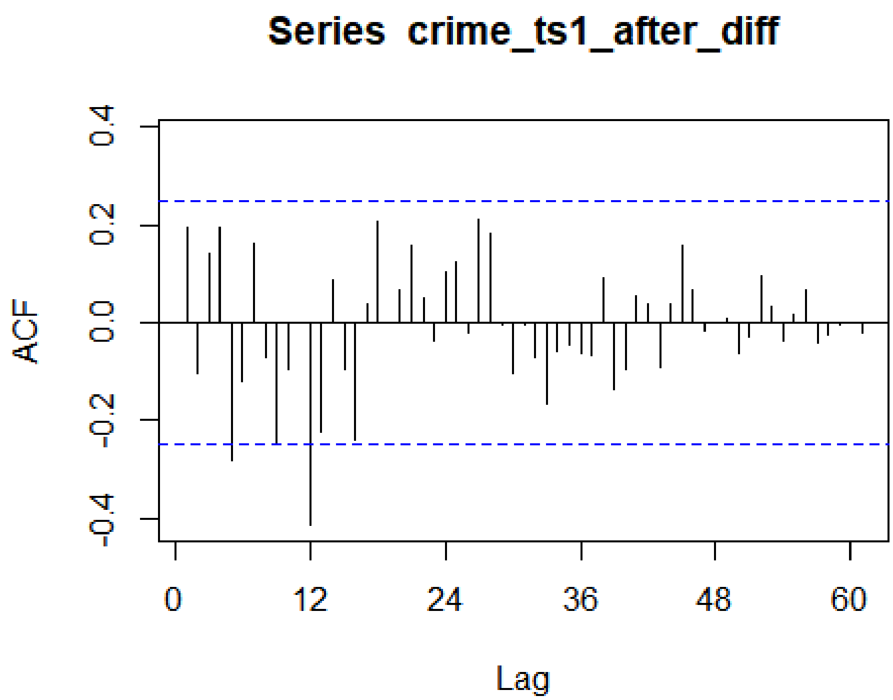
```
  # Is Time Series data stationary? How did you verify? Please post the outpu
t from one of the test.
  adf.test(crime_ts1,k=0)

## Warning in adf.test(crime_ts1, k = 0): p-value smaller than printed p-valu
e

##
##  Augmented Dickey-Fuller Test
##
## data:  crime_ts1
## Dickey-Fuller = -4.1352, Lag order = 0, p-value = 0.01
## alternative hypothesis: stationary

  #p value is .01<.05
  # ADF test says differences is required if p-value is > 0.05
  #It says it is stationary, trend stationary , no difference for trend is re
quired but other method shows difference is required becuase of seasonality


  kpss.test(crime_ts1)

## Warning in kpss.test(crime_ts1): p-value smaller than printed p-value

##
##  KPSS Test for Level Stationarity
##
## data:  crime_ts1
## KPSS Level = 0.93943, Truncation lag parameter = 3, p-value = 0.01
```

```
  # p value is .01 <.05
  # Kipps test says differences is required if p-value is < 0.05
  #Therefore  we can says its non-stationary and requires difference

  # How many differences are needed to make it stationary?
  nsdiffs(crime_ts1)

## [1] 1

  ndiffs(crime_ts1)

## [1] 1

  #1 difference for seasonality and one diff for trend, but actualy after 1 s
easonal diff ts became stationary

  crime_ts1_after_diff=diff(crime_ts1,12)


     adf.test(crime_ts1_after_diff,k=0)

## Warning in adf.test(crime_ts1_after_diff, k = 0): p-value smaller than
## printed p-value

##
##  Augmented Dickey-Fuller Test
##
## data:  crime_ts1_after_diff
## Dickey-Fuller = -6.1223, Lag order = 0, p-value = 0.01
## alternative hypothesis: stationary

  #p value is .01<.05
  # ADF test says differences is required if p-value is > 0.05
     #stationary

   kpss.test(crime_ts1_after_diff)

## Warning in kpss.test(crime_ts1_after_diff): p-value greater than printed p
-
## value

##
##  KPSS Test for Level Stationarity
##
## data:  crime_ts1_after_diff
## KPSS Level = 0.056938, Truncation lag parameter = 3, p-value = 0.1

  #p value is .1>.05
  # Kipps test says differences is required if p-value is < 0.05
  #There we can says its stationary now
  nsdiffs(crime_ts1_after_diff)
```

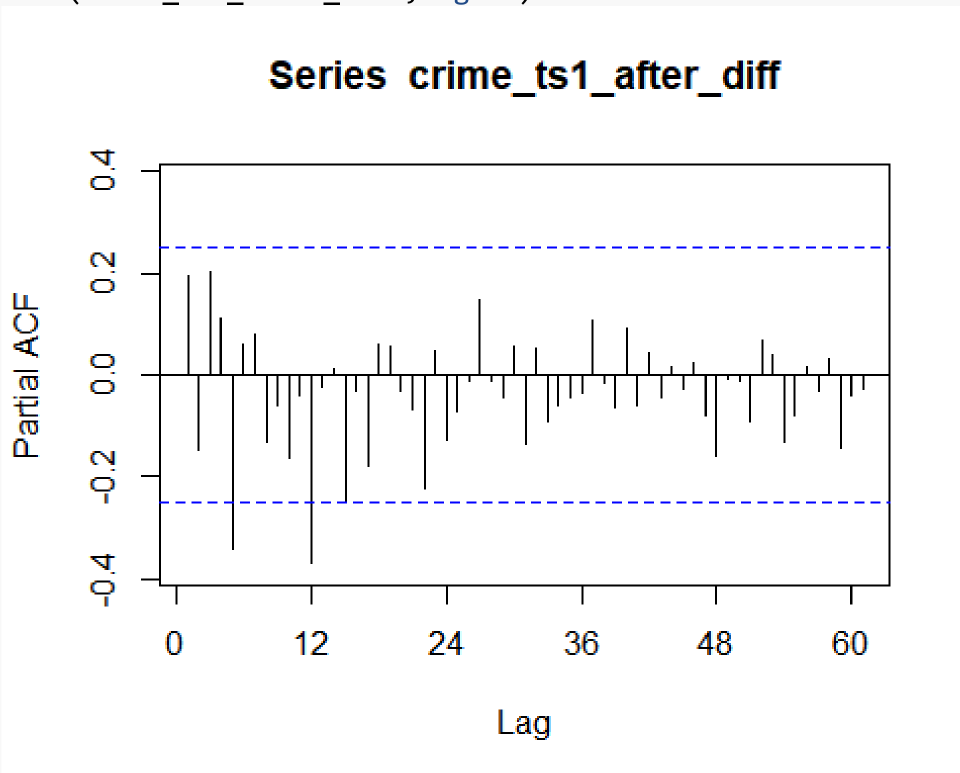
```
## [1] 0

  ndiffs(crime_ts1_after_diff)

## [1] 0

  #we don't need second difference
  #Now after 1 seasonal difference we have stationary time series

    #   Is Seasonality component needed?
      #Yes




#   Plot the Time Series chart of the differenced series.

plot(crime_ts1_after_diff)
```



```
# Plot the ACF and PACF plot of the differenced series.
Acf(crime_ts1_after_diff,lag=74)
```

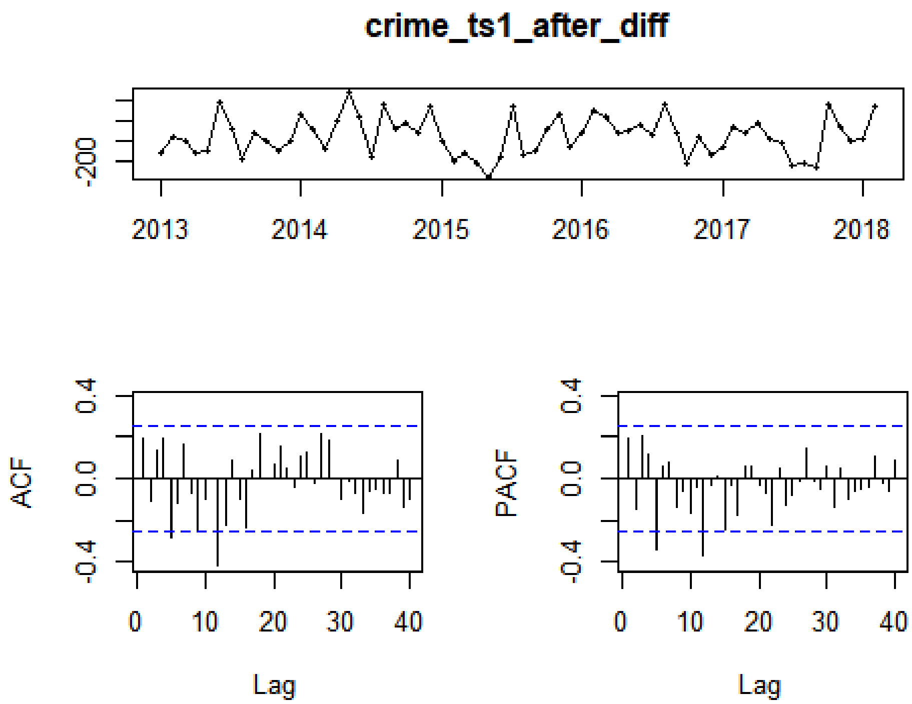


```
#q = 0,1,2,3,4,5 and Q=0,1,2 and d=0
Pacf(crime_ts1_after_diff,lag=74)
```




```
#p = 0,1,2,3,4,5 and P=0,1,2 and D=1
```

```
tsdisplay(crime_ts1_after_diff,lag.max=40)
```




```
# Based on the ACF and PACF, which are the possible ARIMA model possible?
Following are possible ARIMA models
ARIMA(0,0,0)(0,1,0)[12]
ARIMA(0,0,0)(0,1,1)[12]
 ARIMA(0,0,0)(0,1,2)[12]
 ARIMA(0,0,0)(1,1,0)[12]
 ARIMA(0,0,0)(1,1,1)[12]
 ARIMA(0,0,0)(1,1,2)[12]
 ARIMA(0,0,0)(2,1,0)[12]
 ARIMA(0,0,0)(2,1,1)[12]
 ARIMA(0,0,0)(2,1,2)[12]
 ARIMA(0,0,1)(0,1,0)[12]
 ARIMA(0,0,1)(0,1,1)[12]
 ARIMA(0,0,1)(0,1,2)[12]
 ARIMA(0,0,1)(1,1,0)[12]
 ARIMA(0,0,1)(1,1,1)[12]
 ARIMA(0,0,1)(1,1,2)[12]
 ARIMA(0,0,1)(2,1,0)[12]
 ARIMA(0,0,1)(2,1,1)[12]
 ARIMA(0,0,1)(2,1,2)[12]
 ARIMA(0,0,2)(0,1,0)[12]
 ARIMA(0,0,2)(0,1,1)[12]
 ARIMA(0,0,2)(0,1,2)[12]
 ARIMA(0,0,2)(1,1,0)[12]
 ARIMA(0,0,2)(1,1,1)[12]
 ARIMA(0,0,2)(1,1,2)[12]
 ARIMA(0,0,2)(2,1,0)[12]
```

```
ARIMA(0,0,2)(2,1,1)[12]
ARIMA(0,0,3)(0,1,0)[12]
ARIMA(0,0,3)(0,1,1)[12]
ARIMA(0,0,3)(0,1,2)[12]
ARIMA(0,0,3)(1,1,0)[12]
ARIMA(0,0,3)(1,1,1)[12]
ARIMA(0,0,3)(2,1,0)[12]
ARIMA(0,0,4)(0,1,0)[12]
ARIMA(0,0,4)(0,1,1)[12]
ARIMA(0,0,4)(1,1,0)[12]
ARIMA(0,0,5)(0,1,0)[12]
ARIMA(1,0,0)(0,1,0)[12]
ARIMA(1,0,0)(0,1,1)[12]
ARIMA(1,0,0)(0,1,2)[12]
ARIMA(1,0,0)(1,1,0)[12]
ARIMA(1,0,0)(1,1,1)[12]
ARIMA(1,0,0)(1,1,2)[12]
ARIMA(1,0,0)(2,1,0)[12]
ARIMA(1,0,0)(2,1,1)[12]
ARIMA(1,0,0)(2,1,2)[12]
ARIMA(1,0,1)(0,1,0)[12]
ARIMA(1,0,1)(0,1,1)[12]
ARIMA(1,0,1)(0,1,2)[12]
ARIMA(1,0,1)(1,1,0)[12]
ARIMA(1,0,1)(1,1,1)[12]
ARIMA(1,0,1)(1,1,2)[12]
ARIMA(1,0,1)(2,1,0)[12]
ARIMA(1,0,1)(2,1,1)[12]
ARIMA(1,0,2)(0,1,0)[12]
ARIMA(1,0,2)(0,1,1)[12]
ARIMA(1,0,2)(0,1,2)[12]
ARIMA(1,0,2)(1,1,0)[12]
ARIMA(1,0,2)(1,1,1)[12]
ARIMA(1,0,2)(2,1,0)[12]
ARIMA(1,0,3)(0,1,0)[12]
ARIMA(1,0,3)(0,1,1)[12]
ARIMA(1,0,3)(1,1,0)[12]
ARIMA(1,0,4)(0,1,0)[12]
ARIMA(2,0,0)(0,1,0)[12]
ARIMA(2,0,0)(0,1,1)[12]
ARIMA(2,0,0)(0,1,2)[12]
ARIMA(2,0,0)(1,1,0)[12]
ARIMA(2,0,0)(1,1,1)[12]
ARIMA(2,0,0)(1,1,2)[12]
ARIMA(2,0,0)(2,1,0)[12]
ARIMA(2,0,0)(2,1,1)[12]
ARIMA(2,0,1)(0,1,0)[12]
ARIMA(2,0,1)(0,1,1)[12]
ARIMA(2,0,1)(0,1,2)[12]
ARIMA(2,0,1)(1,1,0)[12]
ARIMA(2,0,1)(1,1,1)[12]
ARIMA(2,0,1)(2,1,0)[12]
ARIMA(2,0,2)(0,1,0)[12]
ARIMA(2,0,2)(0,1,1)[12]
ARIMA(2,0,2)(1,1,0)[12]
```

```
ARIMA(2,0,3)(0,1,0)[12]
ARIMA(3,0,0)(0,1,0)[12]
ARIMA(3,0,0)(0,1,1)[12]
ARIMA(3,0,0)(0,1,2)[12]
ARIMA(3,0,0)(1,1,0)[12]
ARIMA(3,0,0)(1,1,1)[12]
ARIMA(3,0,0)(2,1,0)[12]
ARIMA(3,0,1)(0,1,0)[12]
ARIMA(3,0,1)(0,1,1)[12]
ARIMA(3,0,1)(1,1,0)[12]
ARIMA(3,0,2)(0,1,0)[12]
ARIMA(4,0,0)(0,1,0)[12]
ARIMA(4,0,0)(0,1,1)[12]
ARIMA(5,0,0)(0,1,0)[12]


fit1=Arima(crime_ts1, order=c(0,0,0), seasonal=c(0,1,1))
fit2=Arima(crime_ts1, order=c(0,0,1), seasonal=c(0,1,1))
fit3=Arima(crime_ts1, order=c(1,0,2), seasonal=c(0,1,1))
fit4=Arima(crime_ts1, order=c(1,0,2), seasonal=c(1,1,1))
#ARIMA(0,0,0)(0,1,1)[12]
#AIC=772.91   AICc=773.12   BIC=777.17
#ARIMA(0,0,1)(0,1,1)[12]
#AIC=758.98   AICc=759.39   BIC=765.36
#ARIMA(1,0,2)(0,1,1)[12]
#AIC=737.53   AICc=738.6   BIC=748.17
#Arima(crime_ts1, order=c(1,0,2), seasonal=c(1,1,1))
#IC=739.38   AICc=740.91   BIC=752.14

#Show the AIC, BIC and Sigma^2 for the possible models?
#all possible model and there AIC are as follows


for (p in c(0,1,2,3)){
  for (q in c(0,1,2,3,4)){
    for(pp in c(0,1,2)){
      for(qq in c(0,1,2)){
  print(paste("ARIMA (",p,0,q,")","(", pp,1,qq,")"))
        fit=Arima(crime_ts1, order=c(p,0,q), seasonal=c(pp,1,qq))
        print(paste("AIC ",fit$aic))
        print(paste("BIC ",fit$bic))
        print(paste("SIGMA2 ",fit$sigma2))
}}}
}
[1] "ARIMA ( 0 0 0 ) ( 0 1 0 )"
[1] "AIC  770.914089197612"
[1] "BIC  773.041223582657"
[1] "SIGMA2  14242.4834176378"
[1] "ARIMA ( 0 0 0 ) ( 0 1 1 )"
[1] "AIC  772.913607608731"
[1] "BIC  777.167876378821"
```

```
[1] "SIGMA2  14475.8390268803"
[1] "ARIMA ( 0 0 0 ) ( 0 1 2 )"
[1] "AIC  765.635561935435"
[1] "BIC  772.01696509057"
[1] "SIGMA2  11344.4617708254"
[1] "ARIMA ( 0 0 0 ) ( 1 1 0 )"
[1] "AIC  772.913199078687"
[1] "BIC  777.167467848777"
[1] "SIGMA2  14475.7070703569"
[1] "ARIMA ( 0 0 0 ) ( 1 1 1 )"
[1] "AIC  771.992376451078"
[1] "BIC  778.373779606213"
[1] "SIGMA2  12659.6392905252"
[1] "ARIMA ( 0 0 0 ) ( 1 1 2 )"
[1] "AIC  764.71296013102"
[1] "BIC  773.2214976712"
[1] "SIGMA2  10224.3410360091"
[1] "ARIMA ( 0 0 0 ) ( 2 1 0 )"
[1] "AIC  764.5053275341"
[1] "BIC  770.886730689235"
[1] "SIGMA2  11374.7904976455"
[1] "ARIMA ( 0 0 0 ) ( 2 1 1 )"
[1] "AIC  764.346358297111"
[1] "BIC  772.854895837292"
[1] "SIGMA2  10229.798821955"
[1] "ARIMA ( 0 0 0 ) ( 2 1 2 )"
[1] "AIC  766.300387399091"
[1] "BIC  776.936059324316"
[1] "SIGMA2  9992.54370289367"
[1] "ARIMA ( 0 0 1 ) ( 0 1 0 )"
[1] "AIC  758.248199346754"
[1] "BIC  762.502468116844"
[1] "SIGMA2  11384.835671154"
[1] "ARIMA ( 0 0 1 ) ( 0 1 1 )"
[1] "AIC  758.979536637857"
[1] "BIC  765.360939792992"
[1] "SIGMA2  11294.226902486"
[1] "ARIMA ( 0 0 1 ) ( 0 1 2 )"
[1] "AIC  756.387514417996"
[1] "BIC  764.896051958176"
[1] "SIGMA2  10100.0556138473"
[1] "ARIMA ( 0 0 1 ) ( 1 1 0 )"
[1] "AIC  758.331229838068"
[1] "BIC  764.712632993204"
[1] "SIGMA2  11126.0746425456"
[1] "ARIMA ( 0 0 1 ) ( 1 1 1 )"
[1] "AIC  759.340901379896"
[1] "BIC  767.849438920076"
[1] "SIGMA2  11055.5969703462"
[1] "ARIMA ( 0 0 1 ) ( 1 1 2 )"
```

```
[1] "AIC  755.87977136558"
[1] "BIC  766.515443290805"
[1] "SIGMA2  9132.83227546468"
[1] "ARIMA ( 0 0 1 ) ( 2 1 0 )"
[1] "AIC  757.230845489578"
[1] "BIC  765.739383029758"
[1] "SIGMA2  10431.8507997334"
[1] "ARIMA ( 0 0 1 ) ( 2 1 1 )"
[1] "AIC  756.051259091664"
[1] "BIC  766.686931016889"
[1] "SIGMA2  8503.11901706588"
[1] "ARIMA ( 0 0 1 ) ( 2 1 2 )"
[1] "AIC  757.879433620511"
[1] "BIC  770.642239930782"
[1] "SIGMA2  9309.96401354871"
[1] "ARIMA ( 0 0 2 ) ( 0 1 0 )"
[1] "AIC  759.057839339125"
[1] "BIC  765.43924249426"
[1] "SIGMA2  11346.4318987998"
[1] "ARIMA ( 0 0 2 ) ( 0 1 1 )"
[1] "AIC  758.2801084787"
[1] "BIC  766.78864601888"
[1] "SIGMA2  10901.5155173642"
[1] "ARIMA ( 0 0 2 ) ( 0 1 2 )"
[1] "AIC  756.673821505719"
[1] "BIC  767.309493430944"
[1] "SIGMA2  10064.9152562114"
[1] "ARIMA ( 0 0 2 ) ( 1 1 0 )"
[1] "AIC  756.97502811001"
[1] "BIC  765.483565650191"
[1] "SIGMA2  10568.7524147554"
[1] "ARIMA ( 0 0 2 ) ( 1 1 1 )"
[1] "AIC  758.440124671178"
[1] "BIC  769.075796596403"
[1] "SIGMA2  10640.7965635709"
[1] "ARIMA ( 0 0 2 ) ( 1 1 2 )"
[1] "AIC  757.368835064633"
[1] "BIC  770.131641374903"
[1] "SIGMA2  9481.70920040389"
[1] "ARIMA ( 0 0 2 ) ( 2 1 0 )"
[1] "AIC  757.671633725535"
[1] "BIC  768.307305650761"
[1] "SIGMA2  10423.6200119808"
[1] "ARIMA ( 0 0 2 ) ( 2 1 1 )"
[1] "AIC  757.442822695836"
[1] "BIC  770.205629006106"
[1] "SIGMA2  8866.97690022976"
[1] "ARIMA ( 0 0 2 ) ( 2 1 2 )"
[1] "AIC  759.346976884355"
[1] "BIC  774.236917579671"
```

```
[1] "SIGMA2  9669.15474525496"
[1] "ARIMA ( 0 0 3 ) ( 0 1 0 )"
[1] "AIC  760.142255387628"
[1] "BIC  768.650792927809"
[1] "SIGMA2  11356.87052933"
[1] "ARIMA ( 0 0 3 ) ( 0 1 1 )"
[1] "AIC  760.050375307719"
[1] "BIC  770.686047232945"
[1] "SIGMA2  11081.0125950179"
[1] "ARIMA ( 0 0 3 ) ( 0 1 2 )"
[1] "AIC  758.345877498532"
[1] "BIC  771.108683808803"
[1] "SIGMA2  10052.877190619"
[1] "ARIMA ( 0 0 3 ) ( 1 1 0 )"
[1] "AIC  758.941662809447"
[1] "BIC  769.577334734673"
[1] "SIGMA2  10732.1501566723"
[1] "ARIMA ( 0 0 3 ) ( 1 1 1 )"
[1] "AIC  760.400190254287"
[1] "BIC  773.162996564557"
[1] "SIGMA2  10805.1058928795"
[1] "ARIMA ( 0 0 3 ) ( 1 1 2 )"
[1] "AIC  759.111514629088"
[1] "BIC  774.001455324404"
[1] "SIGMA2  9481.26314652835"
[1] "ARIMA ( 0 0 3 ) ( 2 1 0 )"
[1] "AIC  759.55773308402"
[1] "BIC  772.320539394291"
[1] "SIGMA2  10544.5279887819"
[1] "ARIMA ( 0 0 3 ) ( 2 1 1 )"
[1] "AIC  759.206954481632"
[1] "BIC  774.096895176947"
[1] "SIGMA2  8833.1769983651"
[1] "ARIMA ( 0 0 3 ) ( 2 1 2 )"
[1] "AIC  761.105105931959"
[1] "BIC  778.122181012319"
[1] "SIGMA2  9698.49220119922"
[1] "ARIMA ( 0 0 4 ) ( 0 1 0 )"
[1] "AIC  749.015338324764"
[1] "BIC  759.651010249989"
[1] "SIGMA2  9179.95883909723"
[1] "ARIMA ( 0 0 4 ) ( 0 1 1 )"
[1] "AIC  746.923731682095"
[1] "BIC  759.686537992365"
[1] "SIGMA2  8558.81193703222"
[1] "ARIMA ( 0 0 4 ) ( 0 1 2 )"
[1] "AIC  748.143676611304"
[1] "BIC  763.03361730662"
[1] "SIGMA2  8570.29920218246"
[1] "ARIMA ( 0 0 4 ) ( 1 1 0 )"
```

```
[1] "AIC  746.471839565429"
[1] "BIC  759.234645875699"
[1] "SIGMA2  8492.11162619569"
[1] "ARIMA ( 0 0 4 ) ( 1 1 1 )"
[1] "AIC  748.471341169076"
[1] "BIC  763.361281864392"
[1] "SIGMA2  8642.81592810684"
[1] "ARIMA ( 0 0 4 ) ( 1 1 2 )"
[1] "AIC  749.819810650297"
[1] "BIC  766.836885730658"
[1] "SIGMA2  8473.59570193809"
[1] "ARIMA ( 0 0 4 ) ( 2 1 0 )"
[1] "AIC  748.470914725649"
[1] "BIC  763.360855420965"
[1] "SIGMA2  8641.88419777145"
[1] "ARIMA ( 0 0 4 ) ( 2 1 1 )"
[1] "AIC  750.068717353409"
[1] "BIC  767.08579243377"
[1] "SIGMA2  8402.130883188"
[1] "ARIMA ( 0 0 4 ) ( 2 1 2 )"
[1] "AIC  751.831673112996"
[1] "BIC  770.975882578402"
[1] "SIGMA2  8698.64334864873"
[1] "ARIMA ( 1 0 0 ) ( 0 1 0 )"
[1] "AIC  755.759827166745"
[1] "BIC  760.014095936835"
[1] "SIGMA2  10927.3453276609"
[1] "ARIMA ( 1 0 0 ) ( 0 1 1 )"
[1] "AIC  752.297864547824"
[1] "BIC  758.67926770296"
[1] "SIGMA2  9842.03259325381"
[1] "ARIMA ( 1 0 0 ) ( 0 1 2 )"
[1] "AIC  752.940459780983"
[1] "BIC  761.448997321164"
[1] "SIGMA2  9763.61258316081"
[1] "ARIMA ( 1 0 0 ) ( 1 1 0 )"
[1] "AIC  751.250444940426"
[1] "BIC  757.631848095562"
[1] "SIGMA2  9674.1039335427"
[1] "ARIMA ( 1 0 0 ) ( 1 1 1 )"
[1] "AIC  753.243899736568"
[1] "BIC  761.752437276749"
[1] "SIGMA2  9843.35843331595"
[1] "ARIMA ( 1 0 0 ) ( 1 1 2 )"
[1] "AIC  754.824630488472"
[1] "BIC  765.460302413697"
[1] "SIGMA2  9832.68817119838"
[1] "ARIMA ( 1 0 0 ) ( 2 1 0 )"
[1] "AIC  753.240261866794"
[1] "BIC  761.748799406974"
```

```
[1] "SIGMA2  9846.23600510966"
[1] "ARIMA ( 1 0 0 ) ( 2 1 1 )"
[1] "AIC  755.250428213078"
[1] "BIC  765.886100138304"
[1] "SIGMA2  10007.345113753"
[1] "ARIMA ( 1 0 0 ) ( 2 1 2 )"
[1] "AIC  756.709987486894"
[1] "BIC  769.472793797164"
[1] "SIGMA2  9944.68101423601"
[1] "ARIMA ( 1 0 1 ) ( 0 1 0 )"
[1] "AIC  752.896132336709"
[1] "BIC  759.277535491844"
[1] "SIGMA2  10208.1702034292"
[1] "ARIMA ( 1 0 1 ) ( 0 1 1 )"
[1] "AIC  738.356534506197"
[1] "BIC  746.865072046377"
[1] "SIGMA2  6741.03223116038"
[1] "ARIMA ( 1 0 1 ) ( 0 1 2 )"
[1] "AIC  740.30992423821"
[1] "BIC  750.945596163435"
[1] "SIGMA2  6898.79542810249"
[1] "ARIMA ( 1 0 1 ) ( 1 1 0 )"
[1] "AIC  740.05098556875"
[1] "BIC  748.55952310893"
[1] "SIGMA2  7379.0873902348"
[1] "ARIMA ( 1 0 1 ) ( 1 1 1 )"
[1] "AIC  740.300450050377"
[1] "BIC  750.936121975602"
[1] "SIGMA2  6915.97561012062"
[1] "ARIMA ( 1 0 1 ) ( 1 1 2 )"
[1] "AIC  741.78435892614"
[1] "BIC  754.54716523641"
[1] "SIGMA2  6917.39505772165"
[1] "ARIMA ( 3 0 0 ) ( 0 1 0 )"
[1] "AIC  750.675203677829"
[1] "BIC  759.18374121801"
[1] "SIGMA2  9688.7475178767"
[1] "ARIMA ( 3 0 0 ) ( 0 1 1 )"
[1] "AIC  743.691437084456"
[1] "BIC  754.327109009681"
[1] "SIGMA2  7957.56298737239"
[1] "ARIMA ( 3 0 0 ) ( 0 1 2 )"
[1] "AIC  745.649246658254"
[1] "BIC  758.412052968525"
[1] "SIGMA2  8107.83618686946"
[1] "ARIMA ( 3 0 0 ) ( 1 1 0 )"
[1] "AIC  745.837617296726"
[1] "BIC  756.473289221951"
[1] "SIGMA2  8560.28698876352"
[1] "ARIMA ( 3 0 0 ) ( 1 1 1 )"
```

```
[1] "AIC  745.669704303152"
[1] "BIC  758.432510613423"
[1] "SIGMA2  8104.19998336689"
[1] "ARIMA ( 3 0 0 ) ( 1 1 2 )"
[1] "AIC  746.725340012429"
[1] "BIC  761.615280707744"
[1] "SIGMA2  7437.84466655635"
[1] "ARIMA ( 3 0 0 ) ( 2 1 0 )"
[1] "AIC  743.189529944925"
[1] "BIC  755.952336255195"
[1] "SIGMA2  7322.84948014396"
[1] "ARIMA ( 3 0 0 ) ( 2 1 1 )"
[1] "AIC  744.610315134036"
[1] "BIC  759.500255829351"
[1] "SIGMA2  7151.36909205626"
[1] "ARIMA ( 3 0 0 ) ( 2 1 2 )"
[1] "AIC  746.60353693898"
[1] "BIC  763.620612019341"
[1] "SIGMA2  7251.85916406055"
[1] "ARIMA ( 3 0 1 ) ( 0 1 0 )"
[1] "AIC  751.209725016205"
[1] "BIC  761.84539694143"
[1] "SIGMA2  9611.04470388488"
[1] "ARIMA ( 3 0 1 ) ( 0 1 1 )"
[1] "AIC  738.312712958565"
[1] "BIC  751.075519268835"
[1] "SIGMA2  6563.46502434085"
[1] "ARIMA ( 3 0 1 ) ( 0 1 2 )"
[1] "AIC  739.9980857423"
[1] "BIC  754.888026437616"
[1] "SIGMA2  6729.78261556644"
[1] "ARIMA ( 3 0 1 ) ( 1 1 0 )"
[1] "AIC  739.602900509689"
[1] "BIC  752.36570681996"
[1] "SIGMA2  7070.51903206744"
[1] "ARIMA ( 3 0 1 ) ( 1 1 1 )"
[1] "AIC  739.93851388194"
[1] "BIC  754.828454577255"
[1] "SIGMA2  6727.97449661076"
[1] "ARIMA ( 3 0 1 ) ( 1 1 2 )"
[1] "AIC  741.873713472511"
[1] "BIC  758.890788552871"
[1] "SIGMA2  6866.58824563834"
[1] "ARIMA ( 3 0 1 ) ( 2 1 0 )"
[1] "AIC  742.717204666551"
[1] "BIC  757.607145361866"
[1] "SIGMA2  7290.16210140853"
[1] "ARIMA ( 3 0 1 ) ( 2 1 1 )"
[1] "AIC  744.657410509052"
[1] "BIC  761.674485589413"
```

```
[1] "SIGMA2  7321.37817937898"
[1] "ARIMA ( 3 0 1 ) ( 2 1 2 )"
[1] "AIC  746.254600131411"
[1] "BIC  765.398809596817"
[1] "SIGMA2  6469.69164840464"
[1] "ARIMA ( 3 0 2 ) ( 0 1 0 )"
[1] "AIC  748.287807375105"
[1] "BIC  761.050613685376"
[1] "SIGMA2  8918.50087454264"
[1] "ARIMA ( 3 0 2 ) ( 0 1 1 )"
[1] "AIC  739.7570258364"
[1] "BIC  754.646966531716"
[1] "SIGMA2  6629.46540224212"
[1] "ARIMA ( 3 0 2 ) ( 0 1 2 )"
[1] "AIC  741.436968218502"
[1] "BIC  758.454043298862"
[1] "SIGMA2  6799.01681764895"
[1] "ARIMA ( 3 0 2 ) ( 1 1 0 )"
[1] "AIC  746.178153106196"
[1] "BIC  761.068093801512"
[1] "SIGMA2  8289.77224458533"
[1] "ARIMA ( 3 0 2 ) ( 1 1 1 )"
[1] "AIC  747.296757447427"
[1] "BIC  764.313832527787"
[1] "SIGMA2  8159.52726339099"
[1] "ARIMA ( 3 0 2 ) ( 1 1 2 )"
[1] "AIC  743.432632657207"
[1] "BIC  762.576842122613"
[1] "SIGMA2  6896.19819121235"
[1] "ARIMA ( 3 0 2 ) ( 2 1 0 )"
[1] "AIC  744.717393995744"
[1] "BIC  761.734469076104"
[1] "SIGMA2  7420.77514561706"
[1] "ARIMA ( 3 0 2 ) ( 2 1 1 )"
[1] "AIC  745.402931892661"
[1] "BIC  764.547141358067"
[1] "SIGMA2  5896.85551555371"
[1] "ARIMA ( 3 0 2 ) ( 2 1 2 )"
[1] "AIC  747.401429653503"
[1] "BIC  768.672773503954"
[1] "SIGMA2  6029.41904059747"
[1] "ARIMA ( 3 0 3 ) ( 0 1 0 )"
[1] "AIC  743.229315338591"
[1] "BIC  758.119256033906"
[1] "SIGMA2  7547.46576064515"
[1] "ARIMA ( 3 0 3 ) ( 0 1 1 )"
[1] "AIC  735.520312724861"
[1] "BIC  752.537387805222"
[1] "SIGMA2  5891.50961096133"
[1] "ARIMA ( 3 0 3 ) ( 0 1 2 )"
```

```
[1] "AIC  737.466738816128"
[1] "BIC  756.610948281534"
[1] "SIGMA2  5912.48709662384"
[1] "ARIMA ( 3 0 3 ) ( 1 1 0 )"
[1] "AIC  737.69734854232"
[1] "BIC  754.714423622681"
[1] "SIGMA2  6363.14690697586"
[1] "ARIMA ( 3 0 3 ) ( 1 1 1 )"
[1] "AIC  737.467398491252"
[1] "BIC  756.611607956658"
[1] "SIGMA2  5913.99781994175"
[1] "ARIMA ( 3 0 3 ) ( 1 1 2 )"
[1] "AIC  739.426268057971"
[1] "BIC  760.697611908422"
[1] "SIGMA2  5904.53071320146"
[1] "ARIMA ( 3 0 3 ) ( 2 1 0 )"
[1] "AIC  738.16424127795"
[1] "BIC  757.308450743356"
[1] "SIGMA2  6146.87853632464"
[1] "ARIMA ( 3 0 3 ) ( 2 1 1 )"
[1] "AIC  739.410403109765"
[1] "BIC  760.681746960216"
[1] "SIGMA2  5931.73865440184"
[1] "ARIMA ( 3 0 3 ) ( 2 1 2 )"
[1] "AIC  741.416411611588"
[1] "BIC  764.814889847084"
[1] "SIGMA2  6084.56045845931"
[1] "ARIMA ( 3 0 4 ) ( 0 1 0 )"
[1] "AIC  751.376645470861"
[1] "BIC  768.393720551222"
[1] "SIGMA2  9030.78782503261"
[1] "ARIMA ( 3 0 4 ) ( 0 1 1 )"
[1] "AIC  736.421970101866"
[1] "BIC  755.566179567272"
[1] "SIGMA2  5687.91385393613"
[1] "ARIMA ( 3 0 4 ) ( 0 1 2 )"
[1] "AIC  738.236925340554"
[1] "BIC  759.508269191004"
[1] "SIGMA2  5820.90700807459"
[1] "ARIMA ( 3 0 4 ) ( 1 1 0 )"
[1] "AIC  741.064418934959"
[1] "BIC  760.208628400365"
[1] "SIGMA2  6465.44429914474"
[1] "ARIMA ( 3 0 4 ) ( 1 1 1 )"
[1] "AIC  738.208200180365"
[1] "BIC  759.479544030816"
[1] "SIGMA2  5870.39659947868"
[1] "ARIMA ( 3 0 4 ) ( 1 1 2 )"
[1] "AIC  739.938008494357"
[1] "BIC  763.336486729853"
```

```
[1] "SIGMA2  5881.45934432296"
[1] "ARIMA ( 3 0 4 ) ( 2 1 0 )"
[1] "AIC  739.289924006253"
[1] "BIC  760.561267856703"
[1] "SIGMA2  6211.19001843974"
[1] "ARIMA ( 3 0 4 ) ( 2 1 1 )"
[1] "AIC  739.84689727147"
[1] "BIC  763.245375506966"
[1] "SIGMA2  5711.94024241078"
[1] "ARIMA ( 3 0 4 ) ( 2 1 2 )"
[1] "AIC  741.759621619951"
[1] "BIC  767.285234240492"
[1] "SIGMA2  5764.06152921603"
```

```
auto.arima(crime_ts1)
```

```
## Series: crime_ts1
## ARIMA(0,0,0)(0,1,1)[12] with drift
##
## Coefficients:
##          sma1    drift
##       -0.6938  -5.8455
## s.e.   0.2026   0.4432
##
## sigma^2 estimated as 6315:  log likelihood=-362.1
## AIC=730.2   AICc=730.61   BIC=736.58
```

```
auto.arima(crime_ts1,trace=TRUE, stepwise = FALSE)
```

```
##
##  ARIMA(0,0,0)(0,1,0)[12]                    : 770.9808
##  ARIMA(0,0,0)(0,1,0)[12] with drift         : 745.1077
##  ARIMA(0,0,0)(0,1,1)[12]                    : 773.117
##  ARIMA(0,0,0)(0,1,1)[12] with drift         : 730.6116
##  ARIMA(0,0,0)(0,1,2)[12]                    : 766.0494
##  ARIMA(0,0,0)(0,1,2)[12] with drift         : 732.8794
##  ARIMA(0,0,0)(1,1,0)[12]                    : 773.1166
##  ARIMA(0,0,0)(1,1,0)[12] with drift         : 733.0104
##  ARIMA(0,0,0)(1,1,1)[12]                    : Inf
##  ARIMA(0,0,0)(1,1,1)[12] with drift         : 732.8712
##  ARIMA(0,0,0)(1,1,2)[12]                    : 765.4147
##  ARIMA(0,0,0)(1,1,2)[12] with drift         : Inf
##  ARIMA(0,0,0)(2,1,0)[12]                    : 764.9191
##  ARIMA(0,0,0)(2,1,0)[12] with drift         : 734.3973
##  ARIMA(0,0,0)(2,1,1)[12]                    : Inf
##  ARIMA(0,0,0)(2,1,1)[12] with drift         : Inf
##  ARIMA(0,0,0)(2,1,2)[12]                    : Inf
##  ARIMA(0,0,0)(2,1,2)[12] with drift         : Inf
##  ARIMA(0,0,1)(0,1,0)[12]                    : 758.4516
##  ARIMA(0,0,1)(0,1,0)[12] with drift         : 743.6372
```

```
##  ARIMA(0,0,1)(0,1,1)[12]                    : 759.3933
##  ARIMA(0,0,1)(0,1,1)[12] with drift         : 730.6412
##  ARIMA(0,0,1)(0,1,2)[12]                    : 757.0893
##  ARIMA(0,0,1)(0,1,2)[12] with drift         : 732.9028
##  ARIMA(0,0,1)(1,1,0)[12]                    : 758.745
##  ARIMA(0,0,1)(1,1,0)[12] with drift         : 732.5183
##  ARIMA(0,0,1)(1,1,1)[12]                    : 760.0427
##  ARIMA(0,0,1)(1,1,1)[12] with drift         : 732.8807
##  ARIMA(0,0,1)(1,1,2)[12]                    : 756.9512
##  ARIMA(0,0,1)(1,1,2)[12] with drift         : 735.1751
##  ARIMA(0,0,1)(2,1,0)[12]                    : 757.9326
##  ARIMA(0,0,1)(2,1,0)[12] with drift         : 733.654
##  ARIMA(0,0,1)(2,1,1)[12]                    : Inf
##  ARIMA(0,0,1)(2,1,1)[12] with drift         : 735.1694
##  ARIMA(0,0,1)(2,1,2)[12]                    : 759.4067
##  ARIMA(0,0,1)(2,1,2)[12] with drift         : 737.6401
##  ARIMA(0,0,2)(0,1,0)[12]                    : 759.4716
##  ARIMA(0,0,2)(0,1,0)[12] with drift         : 745.3147
##  ARIMA(0,0,2)(0,1,1)[12]                    : 758.9819
##  ARIMA(0,0,2)(0,1,1)[12] with drift         : 731.6442
##  ARIMA(0,0,2)(0,1,2)[12]                    : 757.7453
##  ARIMA(0,0,2)(0,1,2)[12] with drift         : 733.9135
##  ARIMA(0,0,2)(1,1,0)[12]                    : 757.6768
##  ARIMA(0,0,2)(1,1,0)[12] with drift         : 733.8564
##  ARIMA(0,0,2)(1,1,1)[12]                    : 759.5116
##  ARIMA(0,0,2)(1,1,1)[12] with drift         : 733.8918
##  ARIMA(0,0,2)(1,1,2)[12]                    : 758.8961
##  ARIMA(0,0,2)(1,1,2)[12] with drift         : 736.3545
##  ARIMA(0,0,2)(2,1,0)[12]                    : 758.7431
##  ARIMA(0,0,2)(2,1,0)[12] with drift         : 734.6265
##  ARIMA(0,0,2)(2,1,1)[12]                    : Inf
##  ARIMA(0,0,2)(2,1,1)[12] with drift         : 736.3722
##  ARIMA(0,0,3)(0,1,0)[12]                    : 760.844
##  ARIMA(0,0,3)(0,1,0)[12] with drift         : 747.5011
##  ARIMA(0,0,3)(0,1,1)[12]                    : 761.1218
##  ARIMA(0,0,3)(0,1,1)[12] with drift         : 732.7913
##  ARIMA(0,0,3)(0,1,2)[12]                    : 759.8732
##  ARIMA(0,0,3)(0,1,2)[12] with drift         : 734.6269
##  ARIMA(0,0,3)(1,1,0)[12]                    : 760.0131
##  ARIMA(0,0,3)(1,1,0)[12] with drift         : 733.3479
##  ARIMA(0,0,3)(1,1,1)[12]                    : 761.9275
##  ARIMA(0,0,3)(1,1,1)[12] with drift         : 734.4542
##  ARIMA(0,0,3)(2,1,0)[12]                    : 761.085
##  ARIMA(0,0,3)(2,1,0)[12] with drift         : 734.8265
##  ARIMA(0,0,4)(0,1,0)[12]                    : 750.0868
##  ARIMA(0,0,4)(0,1,0)[12] with drift         : 743.4597
##  ARIMA(0,0,4)(0,1,1)[12]                    : 748.451
##  ARIMA(0,0,4)(0,1,1)[12] with drift         : 730.3228
##  ARIMA(0,0,4)(1,1,0)[12]                    : 747.9991
##  ARIMA(0,0,4)(1,1,0)[12] with drift         : 733.0894
```

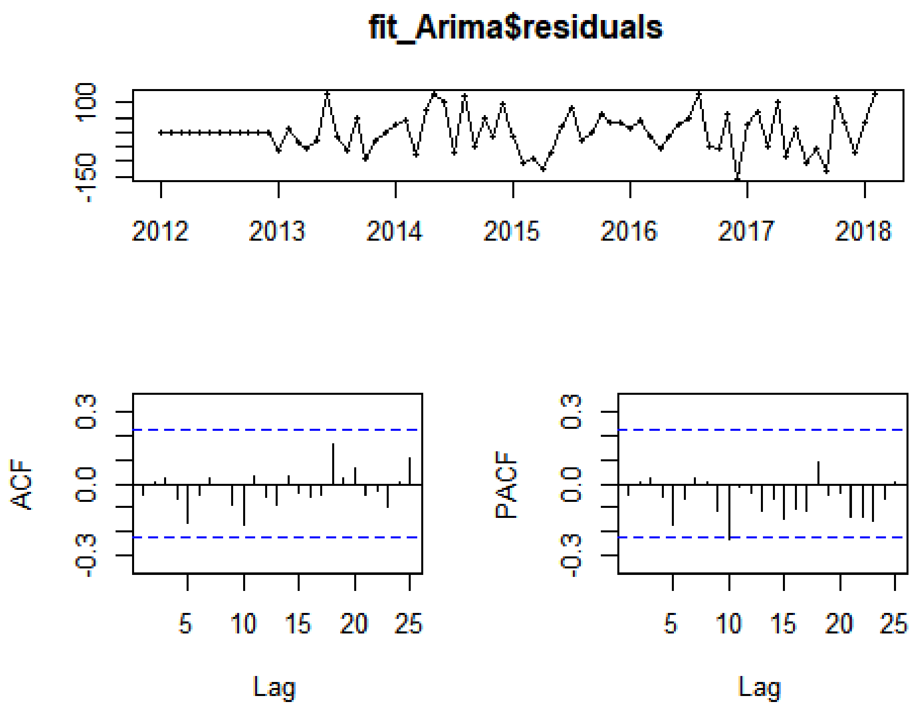
```
##  ARIMA(0,0,5)(0,1,0)[12]                    : 751.2237
##  ARIMA(0,0,5)(0,1,0)[12] with drift         : 741.145
##  ARIMA(1,0,0)(0,1,0)[12]                    : 755.9632
##  ARIMA(1,0,0)(0,1,0)[12] with drift         : 744.8156
##  ARIMA(1,0,0)(0,1,1)[12]                    : 752.7117
##  ARIMA(1,0,0)(0,1,1)[12] with drift         : 731.6546
##  ARIMA(1,0,0)(0,1,2)[12]                    : 753.6422
##  ARIMA(1,0,0)(0,1,2)[12] with drift         : 733.9809
##  ARIMA(1,0,0)(1,1,0)[12]                    : 751.6642
##  ARIMA(1,0,0)(1,1,0)[12] with drift         : 733.6177
##  ARIMA(1,0,0)(1,1,1)[12]                    : 753.9457
##  ARIMA(1,0,0)(1,1,1)[12] with drift         : 733.9659
##  ARIMA(1,0,0)(1,1,2)[12]                    : 755.8961
##  ARIMA(1,0,0)(1,1,2)[12] with drift         : 736.0767
##  ARIMA(1,0,0)(2,1,0)[12]                    : 753.942
##  ARIMA(1,0,0)(2,1,0)[12] with drift         : 735.0817
##  ARIMA(1,0,0)(2,1,1)[12]                    : 756.3219
##  ARIMA(1,0,0)(2,1,1)[12] with drift         : Inf
##  ARIMA(1,0,0)(2,1,2)[12]                    : 758.2373
##  ARIMA(1,0,0)(2,1,2)[12] with drift         : 738.3558
##  ARIMA(1,0,1)(0,1,0)[12]                    : 753.3099
##  ARIMA(1,0,1)(0,1,0)[12] with drift         : 745.4934
##  ARIMA(1,0,1)(0,1,1)[12]                    : Inf
##  ARIMA(1,0,1)(0,1,1)[12] with drift         : 732.3101
##  ARIMA(1,0,1)(0,1,2)[12]                    : Inf
##  ARIMA(1,0,1)(0,1,2)[12] with drift         : 734.6362
##  ARIMA(1,0,1)(1,1,0)[12]                    : Inf
##  ARIMA(1,0,1)(1,1,0)[12] with drift         : 734.3964
##  ARIMA(1,0,1)(1,1,1)[12]                    : Inf
##  ARIMA(1,0,1)(1,1,1)[12] with drift         : 734.6184
##  ARIMA(1,0,1)(1,1,2)[12]                    : Inf
##  ARIMA(1,0,1)(1,1,2)[12] with drift         : 737.0631
##  ARIMA(1,0,1)(2,1,0)[12]                    : Inf
##  ARIMA(1,0,1)(2,1,0)[12] with drift         : 735.3627
##  ARIMA(1,0,1)(2,1,1)[12]                    : Inf
##  ARIMA(1,0,1)(2,1,1)[12] with drift         : 737.0774
##  ARIMA(1,0,2)(0,1,0)[12]                    : Inf
##  ARIMA(1,0,2)(0,1,0)[12] with drift         : 747.6816
##  ARIMA(1,0,2)(0,1,1)[12]                    : Inf
##  ARIMA(1,0,2)(0,1,1)[12] with drift         : Inf
##  ARIMA(1,0,2)(0,1,2)[12]                    : Inf
##  ARIMA(1,0,2)(0,1,2)[12] with drift         : Inf
##  ARIMA(1,0,2)(1,1,0)[12]                    : Inf
##  ARIMA(1,0,2)(1,1,0)[12] with drift         : Inf
##  ARIMA(1,0,2)(1,1,1)[12]                    : Inf
##  ARIMA(1,0,2)(1,1,1)[12] with drift         : Inf
##  ARIMA(1,0,2)(2,1,0)[12]                    : Inf
##  ARIMA(1,0,2)(2,1,0)[12] with drift         : Inf
##  ARIMA(1,0,3)(0,1,0)[12]                    : Inf
##  ARIMA(1,0,3)(0,1,0)[12] with drift         : 749.7802
```

```
##  ARIMA(1,0,3)(0,1,1)[12]                    : Inf
##  ARIMA(1,0,3)(0,1,1)[12] with drift         : 731.1183
##  ARIMA(1,0,3)(1,1,0)[12]                    : Inf
##  ARIMA(1,0,3)(1,1,0)[12] with drift         : 732.5298
##  ARIMA(1,0,4)(0,1,0)[12]                    : 751.2311
##  ARIMA(1,0,4)(0,1,0)[12] with drift         : 743.1642
##  ARIMA(2,0,0)(0,1,0)[12]                    : 757.9247
##  ARIMA(2,0,0)(0,1,0)[12] with drift         : 745.6547
##  ARIMA(2,0,0)(0,1,1)[12]                    : 753.8279
##  ARIMA(2,0,0)(0,1,1)[12] with drift         : 731.2003
##  ARIMA(2,0,0)(0,1,2)[12]                    : 755.5037
##  ARIMA(2,0,0)(0,1,2)[12] with drift         : 733.3849
##  ARIMA(2,0,0)(1,1,0)[12]                    : 753.1208
##  ARIMA(2,0,0)(1,1,0)[12] with drift         : 733.164
##  ARIMA(2,0,0)(1,1,1)[12]                    : 755.4855
##  ARIMA(2,0,0)(1,1,1)[12] with drift         : 733.3378
##  ARIMA(2,0,0)(1,1,2)[12]                    : 757.941
##  ARIMA(2,0,0)(1,1,2)[12] with drift         : 735.7694
##  ARIMA(2,0,0)(2,1,0)[12]                    : 755.4854
##  ARIMA(2,0,0)(2,1,0)[12] with drift         : 734.0734
##  ARIMA(2,0,0)(2,1,1)[12]                    : 757.9415
##  ARIMA(2,0,0)(2,1,1)[12] with drift         : 735.7686
##  ARIMA(2,0,1)(0,1,0)[12]                    : 760.3661
##  ARIMA(2,0,1)(0,1,0)[12] with drift         : 747.2848
##  ARIMA(2,0,1)(0,1,1)[12]                    : Inf
##  ARIMA(2,0,1)(0,1,1)[12] with drift         : 733.1668
##  ARIMA(2,0,1)(0,1,2)[12]                    : 758.4644
##  ARIMA(2,0,1)(0,1,2)[12] with drift         : 735.435
##  ARIMA(2,0,1)(1,1,0)[12]                    : Inf
##  ARIMA(2,0,1)(1,1,0)[12] with drift         : 735.4085
##  ARIMA(2,0,1)(1,1,1)[12]                    : Inf
##  ARIMA(2,0,1)(1,1,1)[12] with drift         : 735.4192
##  ARIMA(2,0,1)(2,1,0)[12]                    : Inf
##  ARIMA(2,0,1)(2,1,0)[12] with drift         : Inf
##  ARIMA(2,0,2)(0,1,0)[12]                    : Inf
##  ARIMA(2,0,2)(0,1,0)[12] with drift         : Inf
##  ARIMA(2,0,2)(0,1,1)[12]                    : Inf
##  ARIMA(2,0,2)(0,1,1)[12] with drift         : Inf
##  ARIMA(2,0,2)(1,1,0)[12]                    : Inf
##  ARIMA(2,0,2)(1,1,0)[12] with drift         : Inf
##  ARIMA(2,0,3)(0,1,0)[12]                    : Inf
##  ARIMA(2,0,3)(0,1,0)[12] with drift         : Inf
##  ARIMA(3,0,0)(0,1,0)[12]                    : 751.377
##  ARIMA(3,0,0)(0,1,0)[12] with drift         : 745.2747
##  ARIMA(3,0,0)(0,1,1)[12]                    : 744.7629
##  ARIMA(3,0,0)(0,1,1)[12] with drift         : 732.3464
##  ARIMA(3,0,0)(0,1,2)[12]                    : 747.1765
##  ARIMA(3,0,0)(0,1,2)[12] with drift         : 734.7142
##  ARIMA(3,0,0)(1,1,0)[12]                    : 746.909
##  ARIMA(3,0,0)(1,1,0)[12] with drift         : 735.1265
```

```
##  ARIMA(3,0,0)(1,1,1)[12]                    : 747.197
##  ARIMA(3,0,0)(1,1,1)[12] with drift         : 734.73
##  ARIMA(3,0,0)(2,1,0)[12]                    : 744.7168
##  ARIMA(3,0,0)(2,1,0)[12] with drift         : 734.7938
##  ARIMA(3,0,1)(0,1,0)[12]                    : 752.2812
##  ARIMA(3,0,1)(0,1,0)[12] with drift         : 747.5407
##  ARIMA(3,0,1)(0,1,1)[12]                    : Inf
##  ARIMA(3,0,1)(0,1,1)[12] with drift         : 734.7801
##  ARIMA(3,0,1)(1,1,0)[12]                    : Inf
##  ARIMA(3,0,1)(1,1,0)[12] with drift         : 737.6333
##  ARIMA(3,0,2)(0,1,0)[12]                    : 749.8151
##  ARIMA(3,0,2)(0,1,0)[12] with drift         : Inf
##  ARIMA(4,0,0)(0,1,0)[12]                    : 750.9048
##  ARIMA(4,0,0)(0,1,0)[12] with drift         : 746.9431
##  ARIMA(4,0,0)(0,1,1)[12]                    : 742.2707
##  ARIMA(4,0,0)(0,1,1)[12] with drift         : 734.2759
##  ARIMA(4,0,0)(1,1,0)[12]                    : 745.4075
##  ARIMA(4,0,0)(1,1,0)[12] with drift         : 737.3346
##  ARIMA(4,0,1)(0,1,0)[12]                    : 750.8774
##  ARIMA(4,0,1)(0,1,0)[12] with drift         : 746.7845
##  ARIMA(5,0,0)(0,1,0)[12]                    : 750.2877
##  ARIMA(5,0,0)(0,1,0)[12] with drift         : 741.8248
##
##
##
##  Best model: ARIMA(0,0,4)(0,1,1)[12] with drift

## Series: crime_ts1
## ARIMA(0,0,4)(0,1,1)[12] with drift
##
## Coefficients:
##          ma1      ma2      ma3     ma4     sma1    drift
##       0.3623  -0.1544  -0.1207  0.3215  -0.6800  -5.8300
## s.e.  0.1246   0.1481   0.1529  0.1232   0.1978   0.5674
##
## sigma^2 estimated as 5726:  log likelihood=-357.12
## AIC=728.25   AICc=730.32   BIC=743.14

#We have to choose between ARIMA(0,0,4)(0,1,1)[12] and ARIMA(0,0,0)(0,1,1)[12
]


fit_Arima <- Arima(crime_ts1, order=c(0,0,4), seasonal=c(0,1,1), include.drif
t = TRUE)
fit_Arima

## Series: crime_ts1
## ARIMA(0,0,4)(0,1,1)[12] with drift
##
## Coefficients:
##          ma1      ma2      ma3     ma4     sma1    drift
```

```
##       0.3623  -0.1544  -0.1207  0.3215  -0.6800  -5.8300
## s.e.  0.1246   0.1481   0.1529  0.1232   0.1978   0.5674
##
## sigma^2 estimated as 5726:  log likelihood=-357.12
## AIC=728.25   AICc=730.32   BIC=743.14
```

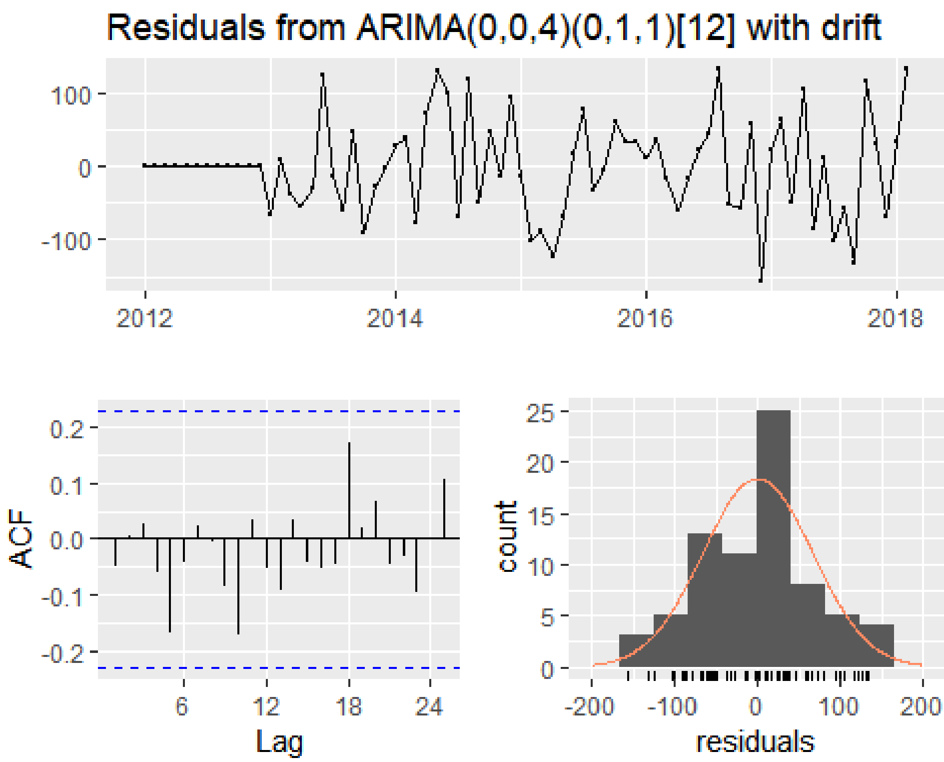
```
tsdisplay(fit_Arima$residuals)
```




```
fit1_Arima <- Arima(crime_ts1, order=c(0,0,0), seasonal=c(0,1,1), include.dri
ft = TRUE)
fit1_Arima
```

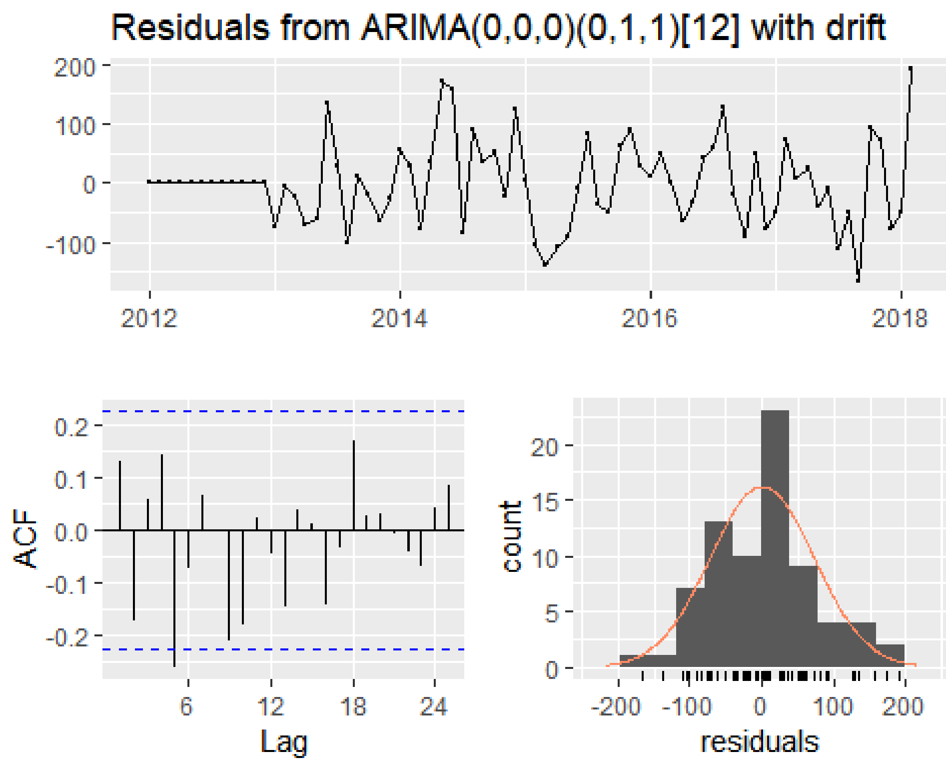
```
## Series: crime_ts1
## ARIMA(0,0,0)(0,1,1)[12] with drift
##
## Coefficients:
##          sma1    drift
##       -0.6938  -5.8455
## s.e.   0.2026   0.4432
##
## sigma^2 estimated as 6315:  log likelihood=-362.1
## AIC=730.2   AICc=730.61   BIC=736.58
```

```
fit_res <- residuals(fit_Arima)
fit1_res <- residuals(fit1_Arima)
checkresiduals(fit_Arima)
```




```
##
##  Ljung-Box test
##
## data:  Residuals from ARIMA(0,0,4)(0,1,1)[12] with drift
## Q* = 7.4665, df = 8.8, p-value = 0.569
##
## Model df: 6.   Total lags used: 14.8
```

```
checkresiduals(fit1_Arima)
```




```
##
##  Ljung-Box test
##
## data:  Residuals from ARIMA(0,0,0)(0,1,1)[12] with drift
## Q* = 20.959, df = 12.8, p-value = 0.0687
##
## Model df: 2.   Total lags used: 14.8
```

```
Box.test(fit_res, lag=16, fitdf=4, type="Ljung")
```

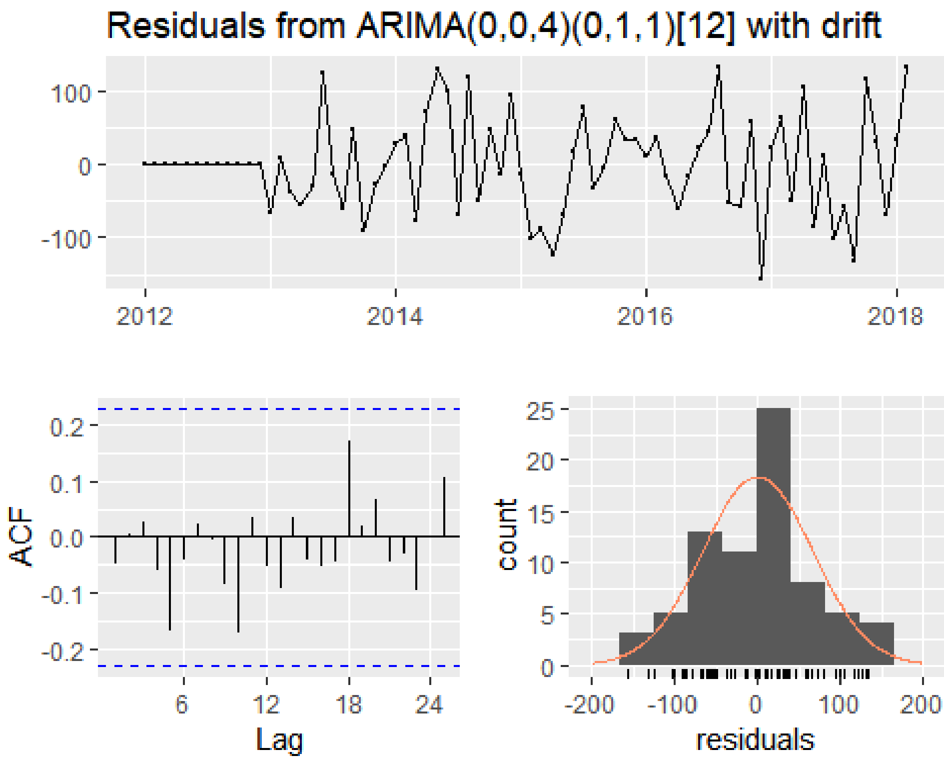
```
##
##  Box-Ljung test
##
## data:  fit_res
## X-squared = 7.9008, df = 12, p-value = 0.7928
```

```
Box.test(fit1_res, lag=16, fitdf=4, type="Ljung")
```

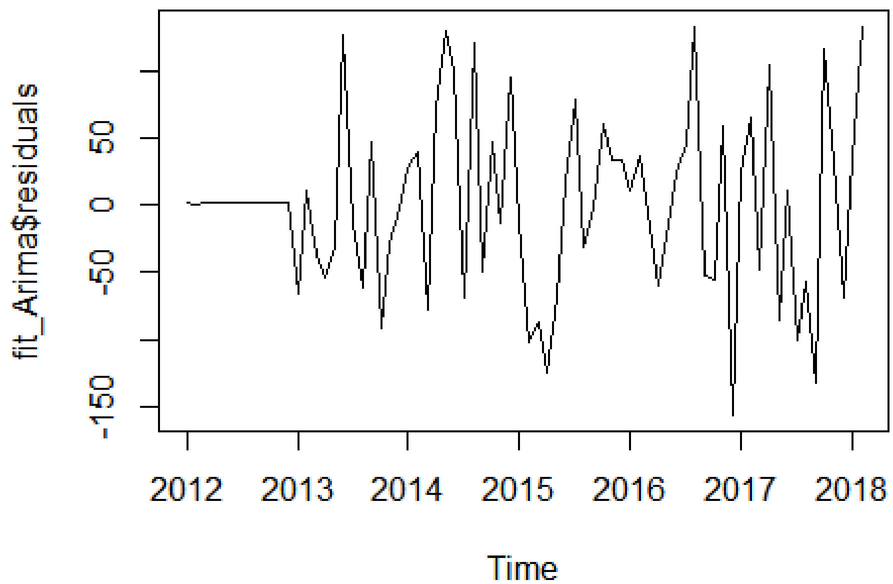
```
##
##  Box-Ljung test
##
## data:  fit1_res
## X-squared = 22.908, df = 12, p-value = 0.02852
```

```
    #   Based on the above AIC, BIC and Sigma^2 values, which model will you
select?
#According to principle of Parsimony, I decided to choose simple model i.e. A
RIMA(0,0,0)(0,1,1)[12] because
```

```
# AIC are close, but when i did residual analysis and Box test it is not a go
od model
#So I will choose ARIMA(0,0,4) (0,1,1)

#What is the final formula for ARIMA with the coefficients?
Arima(crime_ts1, order=c(0,0,4), seasonal=c(0,1,1), include.drift = TRUE)
# Perform Residual Analysis for this technique.
checkresiduals(fit_Arima)
```

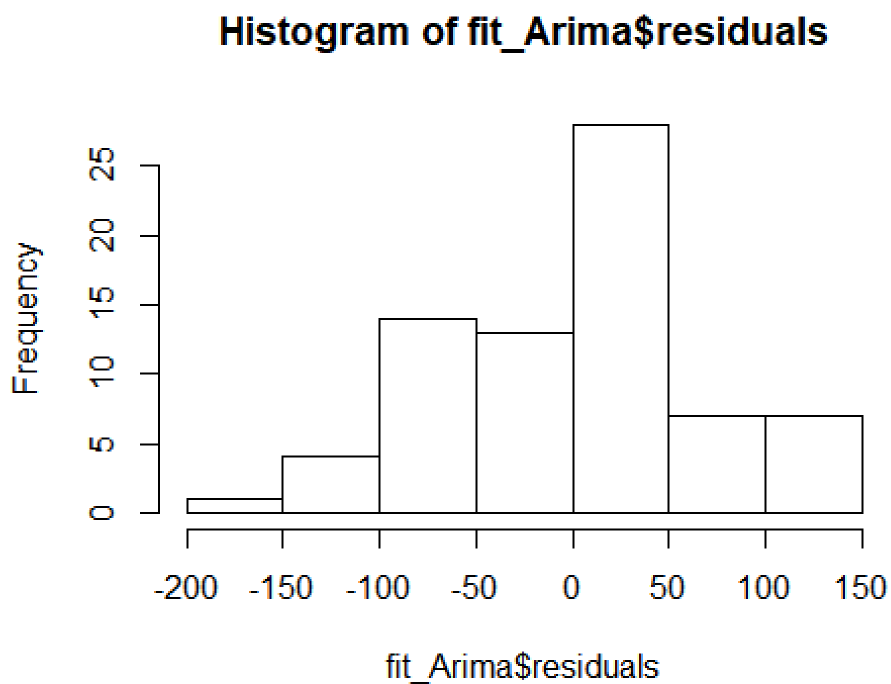



```
##
##  Ljung-Box test
##
## data:  Residuals from ARIMA(0,0,4)(0,1,1)[12] with drift
## Q* = 7.4665, df = 8.8, p-value = 0.569
##
## Model df: 6.   Total lags used: 14.8

  # Do a plot of residuals. What does the plot indicate?
plot(fit_Arima$residuals)
```



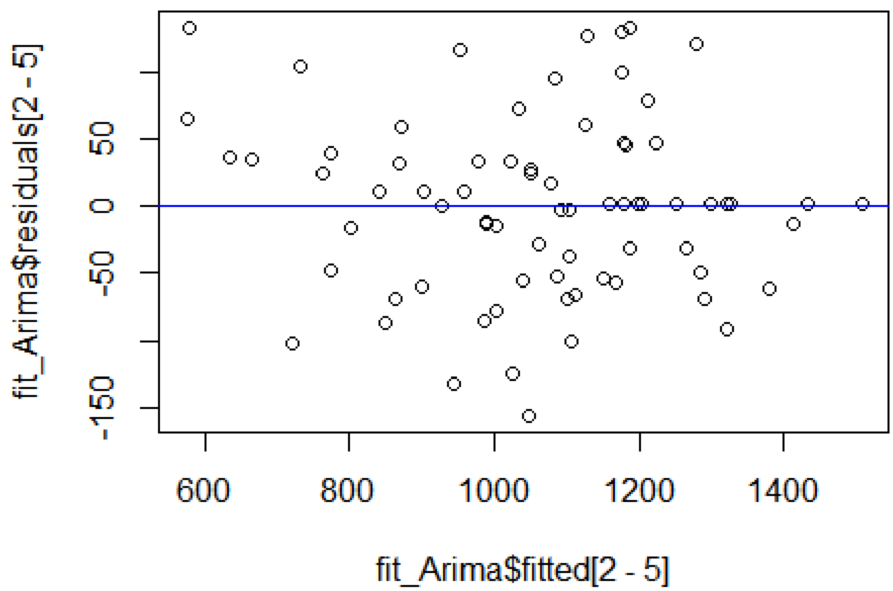
The values highly fluctuating from 2013 onwards. Residuals should be close to zero, but it looks random, Acf plot will clear whether it has significant information or not.

```r
    #   Do a Histogram plot of residuals. What does the plot indicate?
hist(fit_Arima$residuals)
```



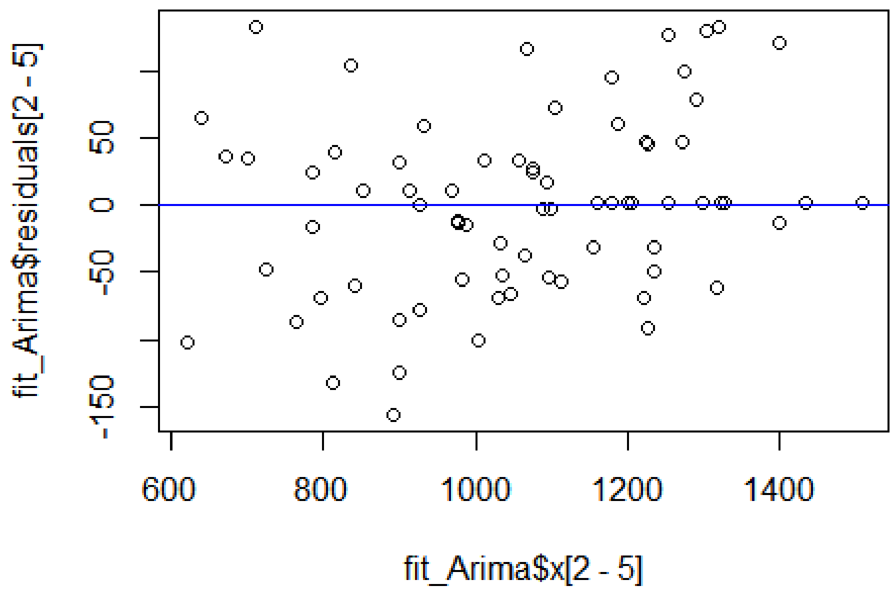
Histogram is skewed

```r
#   Do a plot of fitted values vs. residuals. What does the plot indicate ?
plot(fit_Arima$fitted[2-5],fit_Arima$residuals[2-5])
abline(0,0,col='blue')
```



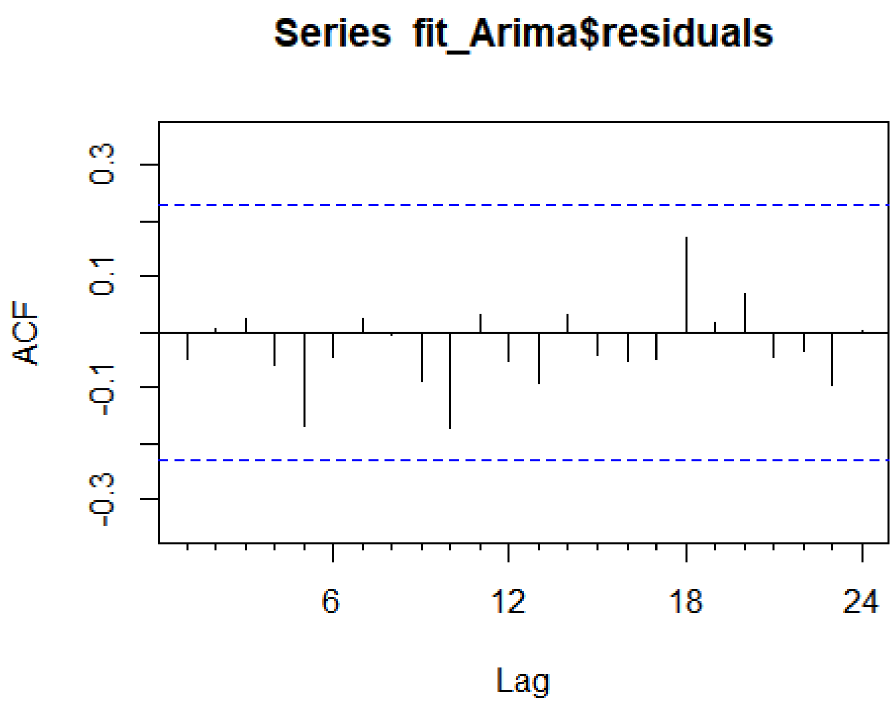
Above plot show that's there still a pattern which shows that the error component influences forecasting element

```r
#   Do a plot of actual values vs. residuals. What does the plot indicate ?
plot(fit_Arima$x[2-5],fit_Arima$residuals[2-5])
abline(0,0,col='blue')
```



Above plot show that's there still a pattern which shows that the error component influences forecasting element

```r
#   Do an ACF plot of the residuals? What does this plot indicate?
Acf(fit_Arima$residuals)
```



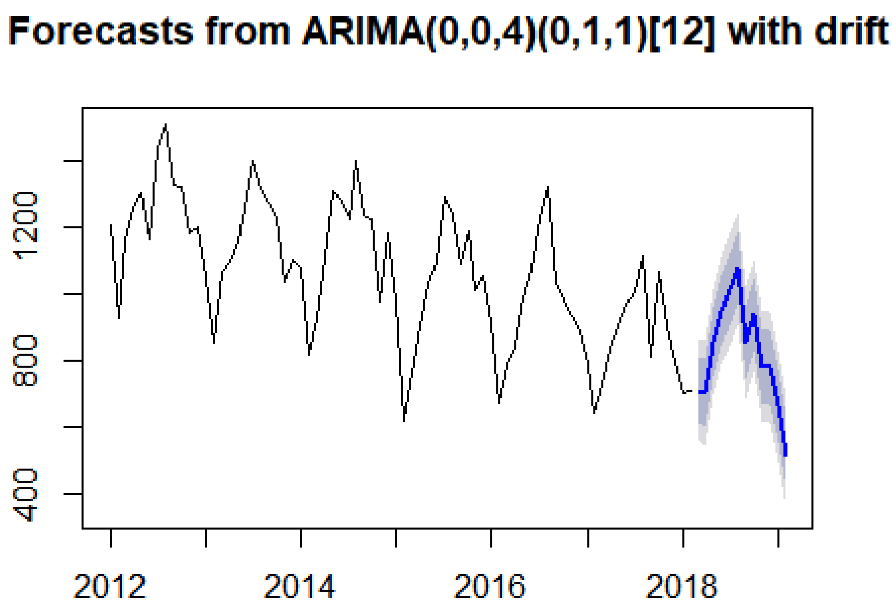
There are no lags, residuals are not correlated means there is no information left in error

```r
#   Print the 5 measures of accuracy for this forecasting technique.
arima_forecast=forecast(fit_Arima,12)
accuracy(arima_forecast)
```

```
##                     ME     RMSE      MAE        MPE     MAPE      MASE
## Training set 0.2662675 65.82532 50.74318 -0.2440167 5.164153 0.5049072
##                     ACF1
## Training set -0.04830753
```

```r
  # Forecast
arima_forecast=forecast(fit_Arima,12)
  # Next one year. Show table and plot
arima_forecast
```

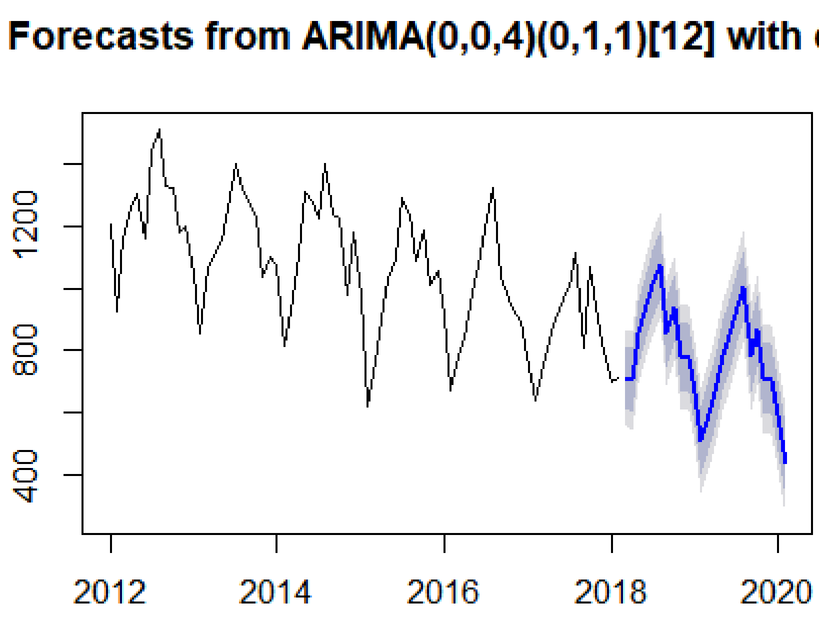
```
##            Point Forecast    Lo 80     Hi 80    Lo 95     Hi 95
## Mar 2018       709.3136 612.0531  806.5741 560.5665  858.0608
## Apr 2018       703.4688 600.0323  806.9054 545.2763  861.6614
## May 2018       852.4855 747.9808  956.9901 692.6594 1012.3116
## Jun 2018       946.9715 841.8129 1052.1301 786.1453 1107.7977
## Jul 2018      1010.6543 900.9524 1120.3561 842.8798 1178.4288
## Aug 2018      1076.8079 967.1060 1186.5097 909.0334 1244.5824
## Sep 2018       853.2275 743.5383  962.9166 685.4724 1020.9825
## Oct 2018       935.8941 826.2068 1045.5815 768.1418 1103.6465
## Nov 2018       783.0792 673.3947  892.7637 615.3313  950.8272
## Dec 2018       777.6034 667.9350  887.2717 609.8801  945.3267
## Jan 2019       663.6195 554.0742  773.1649 496.0844  831.1546
## Feb 2019       511.3674 401.8221  620.9127 343.8323  678.9025

plot(arima_forecast)
```




```
  # Next two years. Show table and plot

arima_forecast_2yr=forecast(fit_Arima,24)
arima_forecast_2yr

##            Point Forecast    Lo 80     Hi 80    Lo 95     Hi 95
## Mar 2018       709.3136 612.0531  806.5741 560.5665  858.0608
## Apr 2018       703.4688 600.0323  806.9054 545.2763  861.6614
## May 2018       852.4855 747.9808  956.9901 692.6594 1012.3116
## Jun 2018       946.9715 841.8129 1052.1301 786.1453 1107.7977
## Jul 2018      1010.6543 900.9524 1120.3561 842.8798 1178.4288
```

```
## Aug 2018      1076.8079 967.1060 1186.5097 909.0334 1244.5824
## Sep 2018       853.2275 743.5383  962.9166 685.4724 1020.9825
## Oct 2018       935.8941 826.2068 1045.5815 768.1418 1103.6465
## Nov 2018       783.0792 673.3947  892.7637 615.3313  950.8272
## Dec 2018       777.6034 667.9350  887.2717 609.8801  945.3267
## Jan 2019       663.6195 554.0742  773.1649 496.0844  831.1546
## Feb 2019       511.3674 401.8221  620.9127 343.8323  678.9025
## Mar 2019       597.7660 483.7899  711.7422 423.4545  772.0775
## Apr 2019       665.2962 550.7522  779.8402 490.1162  840.4762
## May 2019       785.7014 671.0559  900.3468 610.3662  961.0365
## Jun 2019       848.0576 733.3495  962.7656 672.6267 1023.4884
## Jul 2019       940.6941 825.5414 1055.8468 764.5832 1116.8049
## Aug 2019      1006.8477 891.6950 1122.0004 830.7368 1182.9585
## Sep 2019       783.2672 668.1266  898.4078 607.1749  959.3596
## Oct 2019       865.9339 750.7950  981.0728 689.8442 1042.0237
## Nov 2019       713.1190 597.9829  828.2552 537.0335  889.2046
## Dec 2019       707.6431 592.5223  822.7640 531.5811  883.7052
## Jan 2020       593.6593 478.6557  708.6629 417.7765  769.5421
## Feb 2020       441.4072 326.4036  556.4108 265.5244  617.2900
```

```
plot(arima_forecast_2yr)
```




```
  # Summarize this forecasting technique
```

ARIMA: - It stands for Autoregressive(AR) Integrated(I) Moving Average(MA)

In a multiple regression model, we forecast the variable of interest using a linear combination of predictors. In an autoregression model, we forecast the variable of interest using a linear combination of past values of the variable. The term autoregression indicates that it is a regression of the variable against itself. We use Pacf plot (partial autocorrelation graph to find the lags) and apply AR(p) model.

Rather than using past values of the forecast variable in a regression, a moving average model uses past forecast errors in a regression-like model. We refer to this as an MA(q) model, a moving average model of order q. We use Acf plot to find q.

If we combine differencing with autoregression and a moving average model, we obtain a non-seasonal ARIMA model i.e. ARMA(p,d,q)
When we have seasonal component in time series we apply ARIMA model.
ARIMA(p,d,q)(P,D,Q)[F]
Where
(p,d,q) is for trend
(P,D,Q) is for seasonal
F frequency in the time series (4,12)(quarterly, monthly)respectively
p = order of autoregressive part for non-seasonal
d = degree of first differencing involved for non-seasonal
q=order of moving average part for non-seasonal
P= order of autoregressive part for seasonal
D= degree of first differencing involved for seasonal
Q = order of moving average part for seasonal

Box – Jenkins methodology refers to set of procedures for identifying, fitting, and checking ARIMA models with time series data. Forecast follows directly from the form of the fitted model.

ARIMA is known as Box-Jenkins method as they really bought it mainstream.

ARIMA is data hungry method, it need at least 6 to 10 years of seasonal data to forecast

It is useful because it is most powerful tool for accurate short-range forecast. Moreover, Models are quite flexible and can represent a wide range of characteristics of time series occurring in practices.
Formal procedures for testing adequacy of models are available. Forecast and predictions intervals follow directly from the fitted model.

```r
 # How good is the accuracy?
accuracy(arima_forecast)

##                     ME     RMSE      MAE        MPE     MAPE      MASE
## Training set 0.2662675 65.82532 50.74318 -0.2440167 5.164153 0.5049072
##                     ACF1
## Training set -0.04830753
```

```
RMSE is 65.82 which is lowest as compared to other forecasting models,
Accuracy is highest

# What does it predict time series will be in one year and next two years?
For 1 year
         Point Forecast
Mar 2018       709.3136
Apr 2018       703.4688
May 2018       852.4855
Jun 2018       946.9715
Jul 2018      1010.6543
Aug 2018      1076.8079
Sep 2018       853.2275
Oct 2018       935.8941
Nov 2018       783.0792
Dec 2018       777.6034
Jan 2019       663.6195
Feb 2019       511.3674

For Next 2 years it is
          Point Forecast
 Mar 2018       709.3136
 Apr 2018       703.4688
 May 2018       852.4855
 Jun 2018       946.9715
 Jul 2018      1010.6543
 Aug 2018      1076.8079
 Sep 2018       853.2275
 Oct 2018       935.8941
 Nov 2018       783.0792
 Dec 2018       777.6034
 Jan 2019       663.6195
 Feb 2019       511.3674
 Mar 2019       597.7660
 Apr 2019       665.2962
 May 2019       785.7014
 Jun 2019       848.0576
 Jul 2019       940.6941
 Aug 2019      1006.8477
 Sep 2019       783.2672
 Oct 2019       865.9339
 Nov 2019       713.1190
 Dec 2019       707.6431
 Jan 2020       593.6593
 Feb 2020       441.4072
```

Accuracy Summary

Show a table of all the forecast method above with their accuracy measures.

```r
accuracy(naive_forecast)

##                     ME     RMSE      MAE       MPE     MAPE     MASE
## Training set -6.794521 151.4444 128.3562 -1.941344 13.03845 1.277176
##                    ACF1
## Training set -0.1100386

accuracy(ses_crime)

##                    ME     RMSE      MAE       MPE     MAPE     MASE
## Training set -7.15975 149.4172 126.7911 -2.105581 12.88012 1.261603
##                      ACF1
## Training set 0.0002632171

accuracy(hw_crime_forecast)

##                    ME     RMSE      MAE       MPE     MAPE      MASE
## Training set 9.209882 77.38588 62.27083 0.6260094 6.368634 0.6196103
##                   ACF1
## Training set 0.1589406

accuracy(arima_forecast)

##                     ME     RMSE      MAE        MPE     MAPE      MASE
## Training set 0.2662675 65.82532 50.74318 -0.2440167 5.164153 0.5049072
##                     ACF1
## Training set -0.04830753
```

Model	ME	RMSE	MAE	MPE	MAPE	MASE
Naive	-6.794521	151.4444	128.3562	-1.941344	13.03845	1.277176
Smoothing	-7.15975	149.4172	126.7911	-2.105581	12.88012	1.261603
HoltsWinter	9.209882	77.38588	62.27083	0.6260094	6.368634	0.6196103
ARIMA	0.2662675	65.82532	50.74318	-0.244016	5.164153	0.5049072

Separately define each forecast method and why it is useful. Show the best and worst forecast method for each of the accuracy measures

Naïve Forecast: This forecast is a simple model assumes that the recent data provides the best predictions of the future.

That is Y’t+1=Yt

It is useful in the scenario to validate the result obtained from the complex forecasting model.

Simple Smoothing Forecast: In this forecast model, the weights are assigned to the observations based on its relevance. If the more recent observations have more information than more weights are assigned to them and weights are decreased in the exponential order

Y^t+1(point forecast) = @Yt + @(1-@)Yt-1 +…..

@= smoothing factor

It is useful in short term forecasting where it assumes extreme fluctuations represent randomness in a series of historical observation. Also, where there is no proper upward or downward trend present.

Holts Winter Forecast: It is built on simple smoothing forecast concept, here it has been adjusted for trend and seasonality both, which is done 2 scientists Holts and Winter. It comprises of forecast equation and 3 smoothing equations.

Smoothing equations involves calculation of level, trend and seasonality based on respective smoothing constant, which is calculated such that SSE should be minimum.

Once we have level, trend and seasonality, forecast model is built using forecast equation.

Following are calculations

- Forecast equation: Y^t+p = (Lt + p*Tt)*St-s+p
- Level equation: Lt = @Yt/St-s + (1-@)(Lt-1 + T t-1)
- Trend Equation: Tt = &(Lt-Lt-1) + (1-&)Tt-1
- Seasonal Equation: !(Yt/Lt)+(1-!)St-s

  Where Lt = new smoothed Value
  @ = smoothing constant for level
  Yt = Actual forecast at time t
  & = Smoothing constant for trend
  Tt = trend estimate
  p = period for which to calculate forecast on
  Y^t+p = Forecast for p period into the future
  s = length of seasonality
  ! = Seasonality constant
  St = seasonality estimate.

It has advantage of simple smoothing and we have considered seasonality in the model, so it gives more accurate more forecast model.

ARIMA: - It stands for Autoregressive(AR) Integrated(I) Moving Average(MA)

In a multiple regression model, we forecast the variable of interest using a linear combination of predictors. In an autoregression model, we forecast the variable of interest using a linear combination of *past values of the variable*. The term *auto*regression indicates that it is a regression of the variable against itself. We use Pacf plot (partial autocorrelation graph to find the lags) and apply **AR(p)** model.

Rather than using past values of the forecast variable in a regression, a moving average model uses past forecast errors in a regression-like model. We refer to this as an **MA(**q**) model**, a moving average model of order q. We use Acf plot to find q.

If we combine differencing with autoregression and a moving average model, we obtain a non-seasonal ARIMA model i.e. **ARMA(p,d,q)**
When we have seasonal component in time series we apply ARIMA model.
ARIMA(p,d,q)(P,D,Q)[F]
Where
(p,d,q) is for trend
(P,D,Q) is for seasonal
F frequency in the time series (4,12)(quarterly, monthly)respectively
p = order of autoregressive part for non-seasonal
d = degree of first differencing involved for non-seasonal
q=order of moving average part for non-seasonal
P= order of autoregressive part for seasonal
D= degree of first differencing involved for seasonal
Q = order of moving average part for seasonal

Box – Jenkins methodology refers to set of procedures for identifying, fitting, and checking ARIMA models with time series data. Forecast follows directly from the form of the fitted model.

ARIMA is known as Box-Jenkins method as they really bought it mainstream.

ARIMA is data hungry method, it need at least 6 to 10 years of seasonal data to forecast

It is useful because it is most powerful tool for accurate short-range forecast. Moreover, Models are quite flexible and can represent a wide range of characteristics of time series occurring in practices.
Formal procedures for testing adequacy of models are available. Forecast and predictions intervals follow directly from the fitted model.

Best and worst forecasting Model: -

According to ME, ARIMA is best and Simple Smoothing is worst
According to RMSE, ARIMA is best and Naïve is worst
According to MAE, ARIMA is best and Naïve is worst
According to MPE, ARIMA is best and Simple Smoothing is worst
According to MAPE, ARIMA is best and Naïve is worst
According to MASE, ARIMA is best and Naïve is worst

Conclusion

- Summarize your analysis of time series value over the time-period.
  Time Series value will decrease and will follow seasonality, there will be decreased crime with their respective months i.e. there will decrease in crime next year but there will be maximum crime in August and least in February.

- Based on your analysis and forecast above, do you think the value of the time series will increase, decrease or stay flat over the next year? How about next 2 years?

  As per our ARIMA model for next 2 years there is a decreasing pattern along with seasonality : every month of the year will not have same number of crimes.

  August is having highest crimes and February has lowest crimes in next 2 years because of seasonality.

- Rank forecasting methods that best forecast for this time series based on historical values.

  Rank 1 : ARIMA
  Rank 2 : Holt's Winter
  Rank 3: Simple Smoothing
  Rank 4: Naive

Final Question

- If you were me, what final grade would you give yourself for this class?

  A

- Indicate the reasons why you gave yourself this grade?

  Because I learned great deal in this course, learned every forecasting techniques meticulously, honestly did all the assignments, attended every class with full attention and participated in every possible conversation.

R code

```
library(fpp)

library(fpp2)
```

```
library(TTR)

library(forecast)

crime <- read.csv("C:/Users/deept/Downloads/Data_Fall_2018_Crimes.csv")

crime_ts <- ts(crime$Data, start=c(2008,1),frequency = 12)

crime_ts1=window(crime_ts,start=c(2012,1),end=c(2018,2))

crime_ts1

plot(crime_ts)

Acf(crime_ts,lag=120)


#Plot and Inference

#       Show a time series plot.

plot(crime_ts1)

Acf(crime_ts1,lag=74)


#There is seasonality in the data

#there were Seasonal peaks in Jan from year 2008 to 2014, afterward it is not visible

#Decreasing trend, crimes are decreased over the years




#Central Tendency

#       What are the min, max, mean, median, 1st and 3rd Quartile values of the times
series?


summary(crime_ts1)


#       Show the box plot.


boxplot(crime_ts1)
```

#Can you summarize your observation about the time series from the summary stats and box plot?

#On an average there happens 1072 crimes every year

#Either 619 crimes or 1510 crimes happens less probably

#Plot the decomposition of the time series.

decompose_crimets1=decompose(crime_ts1)

plot(decompose_crimets1)

#Is the times series seasonal?

#Yes it is seasonal

Is the decomposition additive or multiplicative?

#Additive seasonal

decompose_crimets1$type

If seasonal, what are the values of the seasonal monthly indices?

#Yes it is seasonal and following are Seasonal Indices

decompose_crimets1$figure

#For which month is the value of time series high and for which month is it low?

#August Highest and Feb lowest

#Can you think of the reason behind the value being high in those months and low in those months?

#Reason is the weather, in August weather is amazing, many people rome around, many tourists comes,

#so more the people higher is the crime rate

#In febraury it is the coldest month most of the people remain inside, they come out only for urgent work and getin quickly

#so less chances of crime

#Therefore crime rate is highest in August and lowest in Febraury

Show the plot for time series adjusted for seasonality. Overlay this with the line for actual time series?

#Does seasonality have big fluctuations to the value of time series?

```
seasonal_adj_crime=seasadj(decompose_crimets1)

plot(crime_ts1)

lines(seasonal_adj_crime,col='red')
```

#Yes for seasonality has the big fluctuationis to the value of time series

#Naive Forecast

#Output

```
naive_forecast<-naive(crime_ts1,12)

plot(naive_forecast)

naive_forecast
```

Perform Residual Analysis for this technique.

Do a plot of residuals. What does the plot indicate?

```
checkresiduals(naive_forecast)

residual_analysis<-residuals(naive_forecast)

plot(residual_analysis)


#There are highly significant values, as there is fluctuations it is not close to 0


 #     Do a Histogram plot of residuals. What does the plot indicate?


hist(residual_analysis)

#It is not normal,but skewed

  #     Do a plot of fitted values vs. residuals. What does the plot indicate?


plot(naive_forecast$fitted[2-5],naive_forecast$residuals[2-5],col=c("red","blue"))

abline(0,0,col='blue')


#There is some pattern, error has some information


  #     Do a plot of actual values vs. residuals. What does the plot indicate?

attributes(naive_forecast)

plot(naive_forecast$x[2-5],naive_forecast$residuals[2-5])

abline(0,0,col='blue')

#there are many points above and below the mean line, some information is left in the
residual

#residual is significant


#     Do an ACF plot of the residuals? What does this plot indicate?


Acf(naive_forecast$residuals)
```

there are lag every 6 months, residual has some information left, this forecasting method
#did not perform well

# Print the 5 measures of accuracy for this forecasting technique
accuracy(naive_forecast)

Forecast
Time series value for next year. Show table and plot
naive_forecast

Summarize this forecasting technique
How good is the accuracy?

What does it predict the value of time series will be in one year?
#710 crimes in the coming year
naive_forecast

Other observation
#here it is showing the 719 crimes will happen next year, but this it not good result
#as it is same as Febraury, and above as mentioned Febraury has lowest crime
it is not going to be same in August or other months

#Simple Moving Averages

Plot the graph for time series.

```
plot(crime_ts1)
```

Show the Simple Moving average of order 3 on the plot above in Red

```
ma3=ma(crime_ts1,order=3)
lines(ma3,col='RED')
```

Show the Simple Moving average of order 6 on the plot above in Blue

```
ma6=ma(crime_ts1,order=6)
lines(ma6,col='BLUE')
```

Show the Simple Moving average of order 9 on the plot above in Green

```
ma9=ma(crime_ts1,order=9)
lines(ma9,col='GREEN')
```

(Bonus) show the forecast of next 12 months using one of the simple average order that you feel works best for time series

```
ma_forecast=forecast(ma9,16)
plot(ma_forecast)
ma_forecast
```

What are your observations of the plot as the moving average order goes up?

#As the order goes up, line becomes smooth, better chances of good forecast

#Smoothing

Perform a smoothing forecast for next 12 months for the time series.

```
ses_crime=ses(crime_ts1,12)
plot(ses_crime)
ses_crime
```

```
summary(ses_crime)
 #	What is the value of alpha?  What does that value signify?
   #alpha = 0.879
   #it signifies the optimal smoothing parameter for the model to get minimum erro

 #	What is the value of initial state?
   #Initial states:
   # l = 1176.0159
 #	What is the value of sigma?  What does the sigma signify?
   #sigma:  151.4782
   #signies the variation around the residual mean

 #	Perform Residual Analysis for this technique.
checkresiduals(ses_crime)
 #	Do a plot of residuals. What does the plot indicate?
 plot(ses_crime$residuals)
 #Flutations are there, residuals significant

 #	Do a Histogram plot of residuals. What does the plot indicate?
 hist(ses_crime$residuals)
  #Histogram is not normal but skewed, indicates not a good forecast

 #	Do a plot of fitted values vs. residuals. What does the plot indicate?
 plot(ses_crime$fitted[2-5],ses_crime$residuals[2-5])
 abline(0,0,col='blue')
 #there are many points above mean line, thus there is a pattern which shows
```

#that error component influences forecast model , there are information still left in residual

#	Do a plot of actual values vs. residuals. What does the plot indicate?

plot(ses_crime$x[2-5],ses_crime$residuals[2-5])

abline(0,0,col='blue')

#It shows a pattern, also there is leverage (many points at one place)

#information is still in the residual, which can be extracted with better model

#Therefore we can say error component influences the forecast component

#	Do an ACF plot of the residuals? What does this plot indicate?

Acf(ses_crime$residuals)

#showing the pattern still exists, there is lag every 6 months

#	Print the 5 measures of accuracy for this forecasting technique

accuracy(ses_crime)

#	Forecast

#	Time series value for next year. Show table and plot

ses_crime

plot(ses_crime)

#	Summarize this forecasting technique

#This is not efficient as mentioned above by residuals analysis, there could be better

#forecasting model than this

#	How good is the accuracy?

```
#	What does it predict the value of time series will be in one year?
  # for the next number of crimes will be  710.2906


#	Other observation
  #it is better than naive bayes, as accuracy is higher


#Holt-Winters
#.	Perform Holt-Winters forecast for next 12 months for the time series.
?HoltWinters
hw_crime=HoltWinters(crime_ts1)
hw_crime_forecast=forecast(hw_crime,h=12)
plot(hw_crime_forecast)
hw_crime
#What is the value of alpha?  What does that value signify?
#Value of  alpha: 0.0406707, signifies level reacts to backdated observations
#(in case if it close to 1, we say more weights are given to recent observations but
#it's not the case here)


#What is the value of beta? What does that value signify?
#Value of beta : 0.08400935, signifies trend depends on previous value


#What is the value of gamma? What does that value signify?
#Gamma is 0.05016539, signifies seasonality repeats according to cycle at regular time
period


#What is the value of initial states for the level, trend and seasonality? What do these
values signify?
#a is level, b is trend, si to s12 is seasonality for 12 months respectively
```

```
hw_crime$coefficients

#What is the value of sigma?  What does the sigma signify?

sd(complete.cases(hw_crime_forecast$residuals))
# Value of sigma =0.3711156, signifies value of standard deviation

 #.     Perform Residual Analysis for this technique.

checkresiduals(hw_crime_forecast)

#Do a plot of residuals. What does the plot indicate?
plot(hw_crime_forecast$residuals)
#for year 2012 it is ok, but for rest it still has values but it looks random
summary(hw_crime_forecast$residuals)
 #Do a Histogram plot of residuals. What does the plot indicate?
hist(hw_crime_forecast$residuals)
#it is skewed, but as compared to other methods this is better

#Do a plot of fitted values vs. residuals. What does the plot indicate?
plot(hw_crime_forecast$fitted[2-5],hw_crime_forecast$residuals[2-5])
abline(0,0,col='blue')
#Variance is still there, it shows 5 outliers, some leverage ,
#residual still has some significnce, possible there exists some method which can perform
better

#Do a plot of actual values vs. residuals. What does the plot indicate?
plot(hw_crime_forecast$x[2-5],hw_crime_forecast$residuals[2-5])
```

abline(0,0,col='blue')

#Variance is still there, it shows 5 outliers, some leverage ,

#residual still has some significnce, possible there exists some method which can perform better

#Do an ACF plot of the residuals? What does this plot indicate?

Acf(hw_crime_forecast$residuals,lag=74)

#shows there is no autocorelation, which shows it is good method of forecast

#Print the 5 measures of accuracy for this forecasting technique

accuracy(hw_crime_forecast)

#	Forecast

#	Time series value for next year. Show table and plot

plot(hw_crime_forecast)

hw_crime_forecast

#Following is the forecast for next 1 year

#Point Forecast

#Mar 2018   702.4429

#Apr 2018   751.9465

#May 2018   827.3633

#Jun 2018   920.5189

#Jul 2018   1011.6039

#Aug 2018   1084.9511

#Sep 2018   902.0591

#Oct 2018   915.7697

#Nov 2018   772.6761

#Dec 2018   793.0333

#Jan 2019   640.6465

#Feb 2019   449.2181

Summarize this forecasting technique

#Holts Winter Forecast: It is built on simple smoothing forecast concept, here it has been adjusted for trend and seasonality both, which is done 2 scientists Holts and Winter. It comprises of forecast equation and 3 smoothing equations.

Smoothing equations involves calculation of level, trend and seasonality based on respective smoothing constant, which is calculated such that SSE should be minimum.

Once we have level, trend and seasonality, forecast model is built using forecast equation.

Following are calculations

- Forecast equation: Y^t+p = (Lt + p*Tt)*St-s+p

- Level equation: Lt = @Yt/St-s + (1-@)(Lt-1 + T t-1)

#- Trend Equation: Tt = &(Lt-Lt-1) + (1-&)Tt-1

- Seasonal Equation: !(Yt/Lt)+(1-!)St-s Where Lt = new smoothed Value @ = smoothing constant for level Yt = Actual forecast at time t & = Smoothing constant for trend Tt = trend estimate p = period for which to calculate forecast on Y^t+p = Forecast for p period into the future s = length of seasonality ! = Seasonality constant St = seasonality estimate.

It has advantage of simple smoothing and we have considered seasonality in the model, so it gives more accurate more forecast model.

How good is the accuracy?

What does it predict the value of time series will be in one year?

#Following is the forecast for next 1 year

#Point Forecast

#Mar 2018   702.4429

#Apr 2018   751.9465

#May 2018   827.3633

#Jun 2018   920.5189

#Jul 2018   1011.6039

#Aug 2018   1084.9511

#Sep 2018   902.0591

#Oct 2018   915.7697

#Nov 2018   772.6761

#Dec 2018   793.0333

#Jan 2019   640.6465

#Feb 2019   449.2181

Other observation

#This is the better model than Naive, Simple smoothing

#Because of 2 reasons

#1. Acf plot of residuals show residuals is insignificant

#2. When we look at values of forecast for next 12 months, it shows high in August and low in feb

which is matching our data

Is Time Series data stationary? How did you verify? Please post the output from one of the test.

adf.test(crime_ts1,k=0)

#p value is .01<.05

ADF test says differences is required if p-value is > 0.05

#It says it is stationary, trend stationary , no difference for trend is required but other method shows difference is required becuase of seasonality

kpss.test(crime_ts1)

```
# p value is .01 <.05

# Kipps test says differences is required if p-value is < 0.05

#Therefore  we can says its non-stationary and requires difference


#       How many differences are needed to make it stationary?

nsdiffs(crime_ts1)

ndiffs(crime_ts1)

#1 difference for seasonality and one diff for trend, but actualy after 1 seasonal diff ts became stationary


crime_ts1_after_diff=diff(crime_ts1,12)




  adf.test(crime_ts1_after_diff,k=0)

#p value is .01<.05

# ADF test says differences is required if p-value is > 0.05

  #stationary


 kpss.test(crime_ts1_after_diff)

#p value is .1>.05

# Kipps test says differences is required if p-value is < 0.05

#There we can says its stationary now

nsdiffs(crime_ts1_after_diff)

ndiffs(crime_ts1_after_diff)

#we don't need second difference

#Now after 1 seasonal difference we have stationary time series


 #      Is Seasonality component needed?
```

#Yes

Plot the Time Series chart of the differenced series.

plot(crime_ts1_after_diff)

Plot the ACF and PACF plot of the differenced series.

Acf(crime_ts1_after_diff,lag=74)

#q = 0,1,2,3,4,5 and Q=0,1,2 and d=0

Pacf(crime_ts1_after_diff,lag=74)

#p = 0,1,2,3,4,5 and P=0,1,2 and D=1

tsdisplay(crime_ts1_after_diff,lag.max=40)

auto.arima(crime_ts1)

auto.arima(crime_ts1,trace=TRUE, stepwise = FALSE)

Based on the ACF and PACF, which are the possible ARIMA model possible?

tsdisplay(crime_ts1_after_diff)

#p = 0,1 q=0,1,2,3

#ARIMA(0,1,0) , ARIMA(0,1,1), ARIMA(0,1,2), ARIMA(0,1,3) ,ARIMA(1,1,0) , ARIMA(1,1,1),ARIMA(1,1,2), ARIMA(1,1,3)

fit1=Arima(crime_ts1, order=c(0,0,0), seasonal=c(0,1,1))

fit2=Arima(crime_ts1, order=c(0,0,1), seasonal=c(0,1,1))

fit3=Arima(crime_ts1, order=c(1,0,2), seasonal=c(0,1,1))

fit4=Arima(crime_ts1, order=c(1,0,2), seasonal=c(1,1,1))

#ARIMA(0,0,0)(0,1,1)[12]

```
#AIC=772.91   AICc=773.12   BIC=777.17

#ARIMA(0,0,1)(0,1,1)[12]

#AIC=758.98   AICc=759.39   BIC=765.36

#ARIMA(1,0,2)(0,1,1)[12]

#AIC=737.53   AICc=738.6   BIC=748.17

#Arima(crime_ts1, order=c(1,0,2), seasonal=c(1,1,1))

#IC=739.38   AICc=740.91   BIC=752.14


  #      Show the AIC, BIC and Sigma^2 for the possible models?

#all possible model and there AIC are as follows

auto.arima(crime_ts1)

auto.arima(crime_ts1,trace=TRUE, stepwise = FALSE)


#We have to choose between ARIMA(0,0,4)(0,1,1)[12] and ARIMA(0,0,0)(0,1,1)[12]


fit_Arima <- Arima(crime_ts1, order=c(0,0,4), seasonal=c(0,1,1), include.drift = TRUE)

fit_Arima

tsdisplay(fit_Arima$residuals)


fit1_Arima <- Arima(crime_ts1, order=c(0,0,0), seasonal=c(0,1,1), include.drift = TRUE)

fit1_Arima


fit_res <- residuals(fit_Arima)

fit1_res <- residuals(fit1_Arima)

checkresiduals(fit_Arima)

checkresiduals(fit1_Arima)

Box.test(fit_res, lag=16, fitdf=4, type="Ljung")

Box.test(fit1_res, lag=16, fitdf=4, type="Ljung")
```

Based on the above AIC, BIC and Sigma^2 values, which model will you select?

#According to principle of Parsimony, I decided to choose simple model i.e. ARIMA(0,0,0)(0,1,1)[12] because

AIC are close, but when i did residual analysis and Box test it is not a good model

#So i will choose

What is the final formula for ARIMA with the coefficients?

#Arima(crime_ts1, order=c(0,0,4), seasonal=c(0,1,1), include.drift = TRUE)

Perform Residual Analysis for this technique.

checkresiduals(fit_Arima)

Do a plot of residuals. What does the plot indicate?

plot(fit_Arima$residuals)

Do a Histogram plot of residuals. What does the plot indicate?

hist(fit_Arima$residuals)

Do a plot of fitted values vs. residuals. What does the plot indicate?

plot(fit_Arima$fitted[2-5],fit_Arima$residuals[2-5])

abline(0,0,col='blue')

Do a plot of actual values vs. residuals. What does the plot indicate?

plot(fit_Arima$x[2-5],fit_Arima$residuals[2-5])

abline(0,0,col='blue')

Do an ACF plot of the residuals? What does this plot indicate?

Acf(fit_Arima$residuals)

Print the 5 measures of accuracy for this forecasting technique.

arima_forecast=forecast(fit_Arima,12)

accuracy(arima_forecast)

Forecast

```
arima_forecast=forecast(fit_Arima,12)
 #        Next one year. Show table and plot
arima_forecast
plot(arima_forecast)
 #        Next two years. Show table and plot

arima_forecast_2yr=forecast(fit_Arima,24)
arima_forecast_2yr
plot(arima_forecast_2yr)
 #        Summarize this forecasting technique
 #        How good is the accuracy?
accuracy(arima_forecast)

   # What does it predict time series will be in one year and next two years?

accuracy(naive_forecast)
accuracy(ses_crime)
accuracy(hw_crime_forecast)
accuracy(arima_forecast)
```