TS_Final_Project_Sarah_Val

Valeri Antonova

6/3/2019

Arima model for terrorist acts after 1994

The first model attempted is for the total terrorist attacks accross the world. The time series is missing data for 1993, and the initial idea was to use only part of the dataset. As there is a huge trend and drift seen in the total dataset, it seemed that cutting out the data should not affect the prediction very much (please refer to python notebook for exploratory analysis)

```
data<-read.csv("~/Desktop/UofChicagoInfo/Time_Series/Final_Project/worldattack_94on.csv",
               row.names = 'Date')
head(data)
```

```
##            Australasia...Oceania Central.America...Caribbean Central.Asia
## 1994-01-31                     1                          18            2
## 1994-02-28                     0                          19            4
## 1994-03-31                     5                          17            8
## 1994-04-30                     0                           8            5
## 1994-05-31                     0                          13            9
## 1994-06-30                     1                          19           11
##            East.Asia Eastern.Europe Middle.East...North.Africa
## 1994-01-31         0              4                        131
## 1994-02-28         3              2                        106
## 1994-03-31         4             10                        178
## 1994-04-30         3             10                        104
## 1994-05-31         1             12                         87
## 1994-06-30         2             13                         59
##            North.America South.America South.Asia Southeast.Asia
## 1994-01-31            20            20         17              3
## 1994-02-28             2            44          9             11
## 1994-03-31            10            39         11             14
## 1994-04-30             8            12         18             15
## 1994-05-31             6            66         27             13
## 1994-06-30             5            39         11             18
##            Sub.Saharan.Africa Western.Europe Sum
## 1994-01-31                 36             76 328
## 1994-02-28                 56             62 318
## 1994-03-31                 76             49 421
## 1994-04-30                 93             89 365
## 1994-05-31                 31             49 314
## 1994-06-30                 24             41 243
```

```
tsdata<-ts(data[,13], start = c(1994,1), end=c(2017,12), frequency=12)
ncol(data)
```

```
## [1] 13
```

```
colnames(data)
```

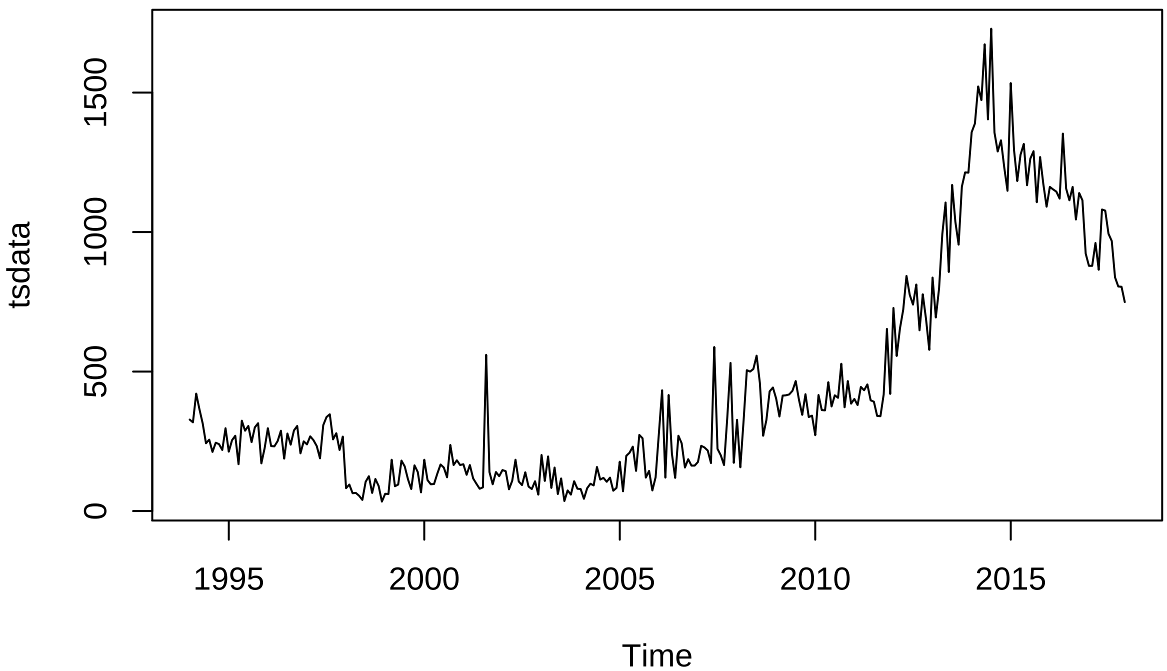
```
##  [1] "Australasia...Oceania"       "Central.America...Caribbean"
##  [3] "Central.Asia"                "East.Asia"
```

```
##  [5] "Eastern.Europe"              "Middle.East...North.Africa"
##  [7] "North.America"               "South.America"
##  [9] "South.Asia"                  "Southeast.Asia"
## [11] "Sub.Saharan.Africa"          "Western.Europe"
## [13] "Sum"
```

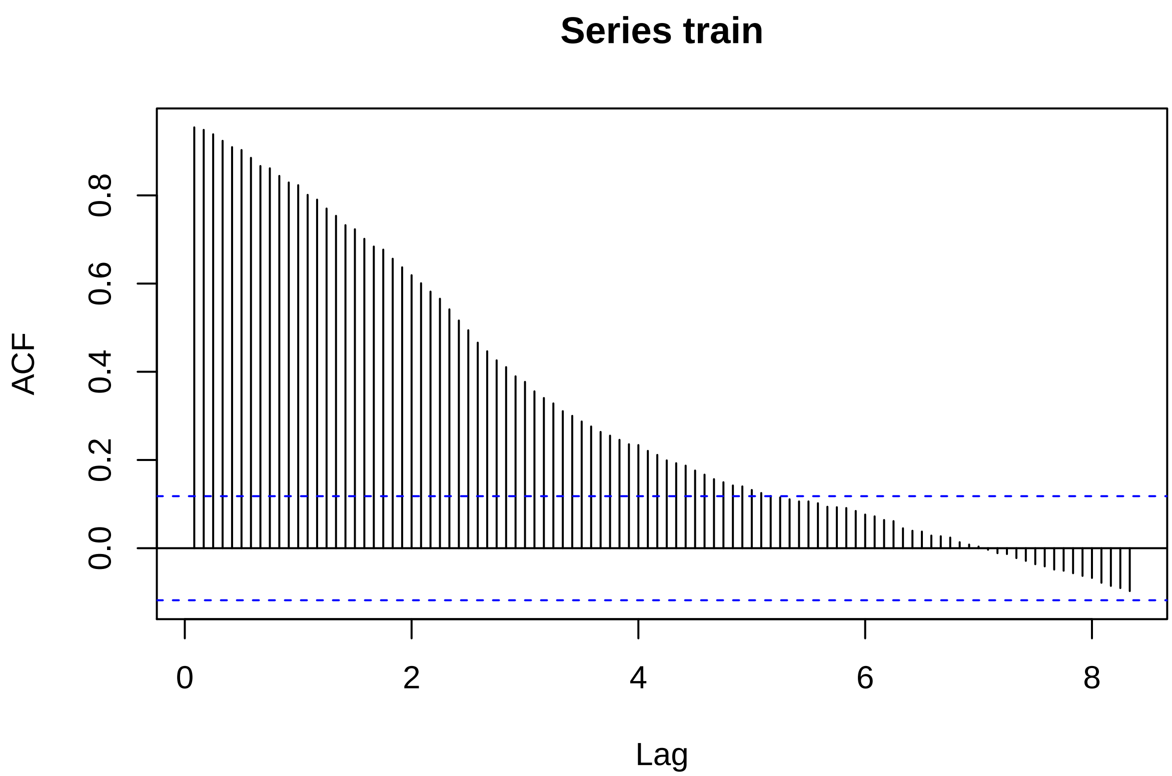
```
head(tsdata)
```

```
##      Jan Feb Mar Apr May Jun
## 1994 328 318 421 365 314 243
```

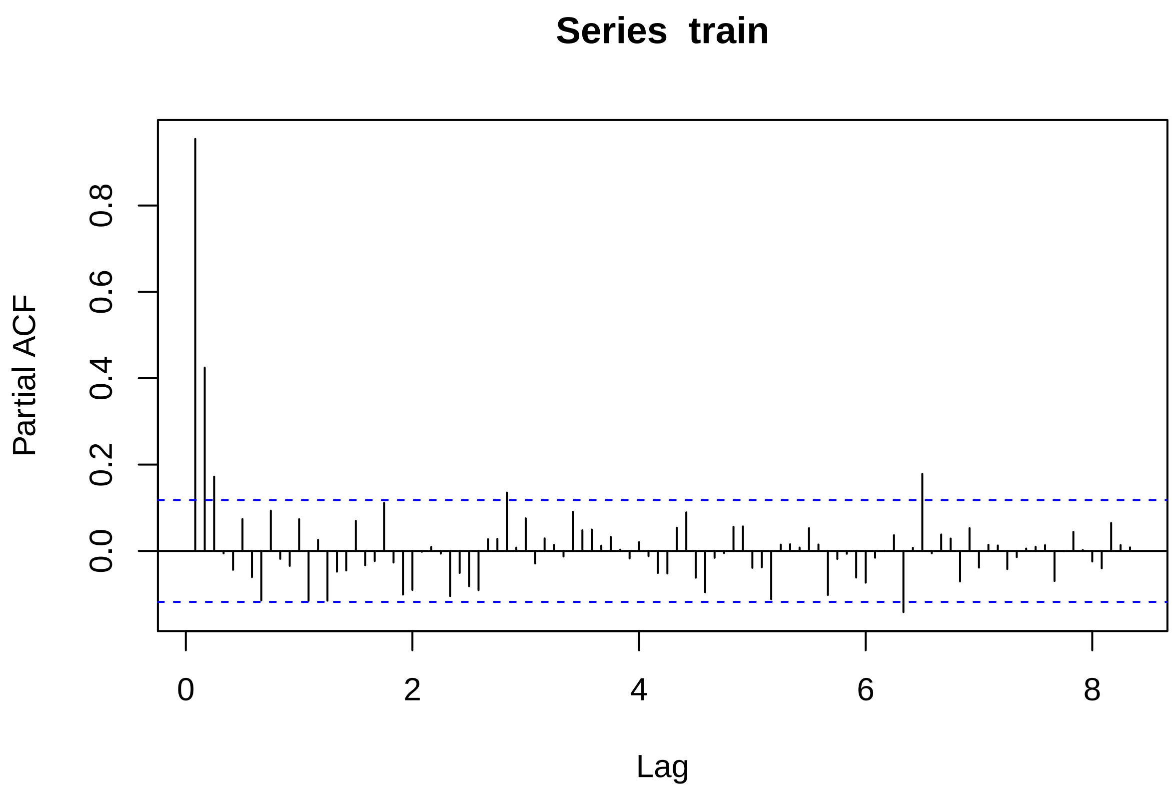
```
plot(tsdata)
```




```
train<-window(tsdata, start=c(1994, 1), end=c(2016,12), frequency=12)
test<-window(tsdata, start=c(2017,1)) #for testing model accuracy

acf(train,lag=100)
```




```r
pacf(train, lag=100) #According to the acf and pacf plots,
```




```r
#the model calls for an autoregressive model with p=3
```

```r
adf.test(train)
```

```
##
##  Augmented Dickey-Fuller Test
##
## data:  train
## Dickey-Fuller = -1.8083, Lag order = 6, p-value = 0.6568
## alternative hypothesis: stationary
```

```r
adf.test(train, alternative='explosive')
```

```
##
##  Augmented Dickey-Fuller Test
##
## data:  train
## Dickey-Fuller = -1.8083, Lag order = 6, p-value = 0.3432
## alternative hypothesis: explosive
```

```r
kpss.test(train)
```

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

```
##
##  KPSS Test for Level Stationarity
##
## data:  train
## KPSS Level = 3.0434, Truncation lag parameter = 5, p-value = 0.01
```

```r
kpss.test(train, null='Trend')
```

```
## Warning in kpss.test(train, null = "Trend"): p-value smaller than printed
## p-value
```

```
##
##  KPSS Test for Trend Stationarity
##
## data:  train
## KPSS Trend = 0.92782, Truncation lag parameter = 5, p-value = 0.01
```

```r
#All stationarity tests show the time-series to be non-stationary,
#implying that drift is present. If only trend or seasonality was present,
#the 'trend' correction on the kpss test would allow the test to show the
#time series as stationary

summary(auto.arima(train))
```

```
## Series: train
## ARIMA(0,1,2)(0,0,1)[12]
##
## Coefficients:
##           ma1     ma2    sma1
##       -0.6408  0.1225  0.1562
## s.e.   0.0609  0.0622  0.0576
##
## sigma^2 estimated as 9368:  log likelihood=-1646.51
## AIC=3301.02   AICc=3301.17   BIC=3315.49
##
## Training set error measures:
```

```
##                     ME     RMSE      MAE       MPE     MAPE      MASE
## Training set 4.41687 96.08645 67.08135 -8.841302 26.60079 0.5393475
##                       ACF1
## Training set -0.0009331167
```

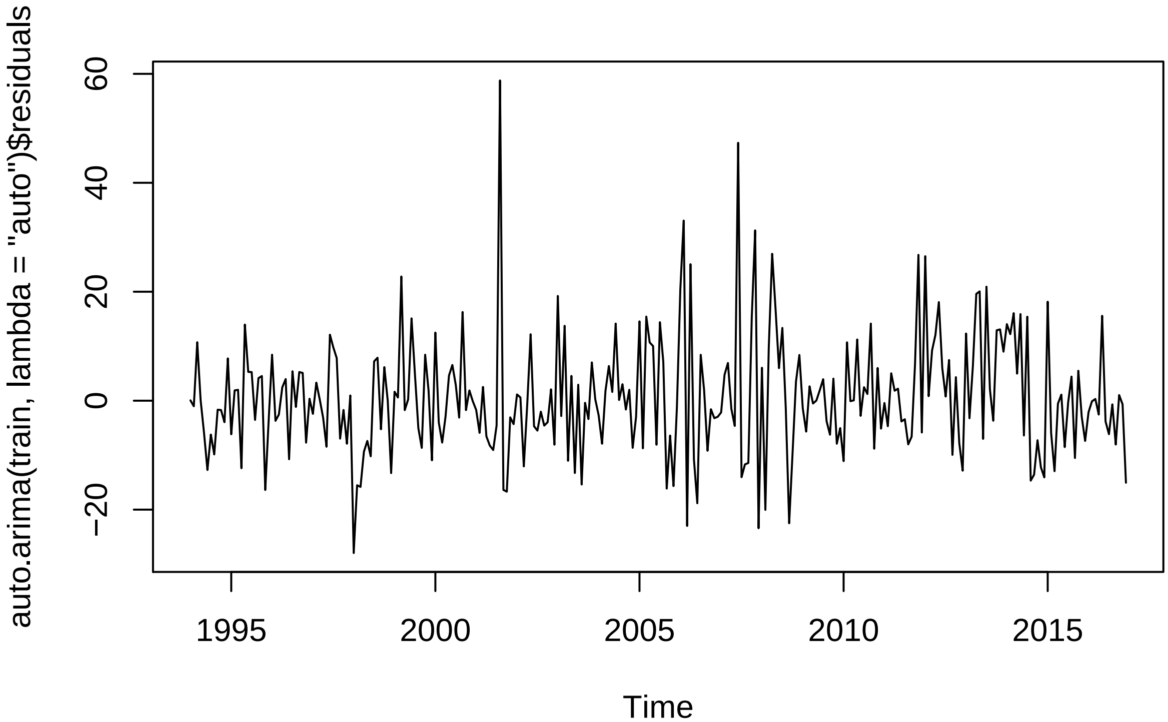
```
# interestingly enough the auto arima function does not
#return an AR, but and MA model with seasonality
summary(auto.arima(train, lambda = 'auto'))
```

```
## Series: train
## ARIMA(0,1,1)(0,0,1)[12]
## Box Cox transformation: lambda= 0.6349565
##
## Coefficients:
##           ma1    sma1
##       -0.6268  0.1089
## s.e.   0.0447  0.0615
##
## sigma^2 estimated as 120.7:  log likelihood=-1048.59
## AIC=2103.18   AICc=2103.26   BIC=2114.03
##
## Training set error measures:
##                     ME     RMSE      MAE       MPE     MAPE      MASE
## Training set 8.904553 96.97057 67.54718 -7.706319 26.00985 0.5430929
##                     ACF1
## Training set 0.001989198
```

```
#adding a box-cox transformation improves the AIC and BIC scores

Box.test(auto.arima(train, lambda = 'auto')$residuals, type=c('Ljung-Box'))
```

```
##
##  Box-Ljung test
##
## data:  auto.arima(train, lambda = "auto")$residuals
## X-squared = 0.54932, df = 1, p-value = 0.4586
```

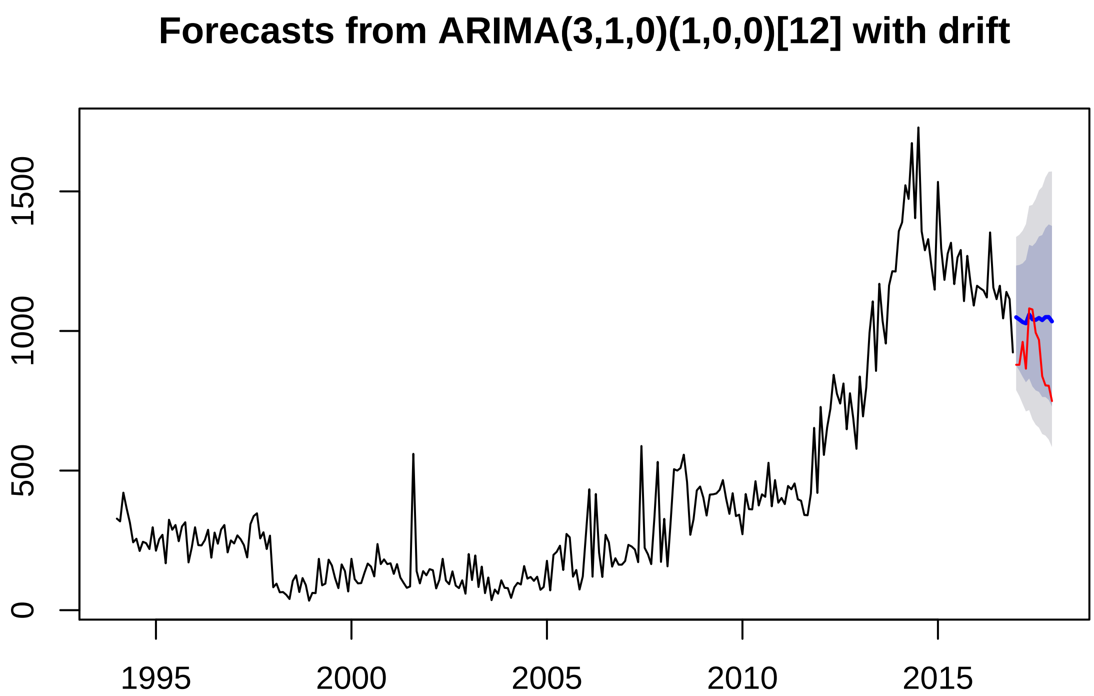
```
#residuals are independent, meaning the model is
#accounting for all explainable variance

plot(auto.arima(train, lambda = 'auto')$residuals)
```




```r
#plot of residuals after model fitting


summary(Arima(train, order=c(3,1,0), seasonal=c(1,0,0),include.drift = TRUE,lambda='auto'))
```

```
## Series: train
## ARIMA(3,1,0)(1,0,0)[12] with drift
## Box Cox transformation: lambda= 0.6349565
##
## Coefficients:
##           ar1      ar2      ar3    sar1   drift
##       -0.6549  -0.3509  -0.1298  0.0962  0.2242
## s.e.   0.0599   0.0688   0.0602  0.0604  0.3418
##
## sigma^2 estimated as 122.3:  log likelihood=-1048.82
## AIC=2109.63   AICc=2109.95   BIC=2131.34
##
## Training set error measures:
##                    ME     RMSE      MAE       MPE     MAPE      MASE
## Training set 3.575156 95.66097 66.82737 -9.098758 26.22891 0.5373055
##                    ACF1
## Training set 0.01714087
```

```r
#differencing to account for drift, setting p to that seen on
#PACF plot and accounting for seasonality significantly
#reduces the AIC and BIC scores, but does not change the MAPE

model<-Arima(train, order=c(3,1,0), seasonal=c(1,0,0),include.drift = TRUE,lambda='auto')
fcst<-forecast(model, h=12 )
plot(fcst)
lines(test, col='red') #the plot shows that the prediction is quite off
```



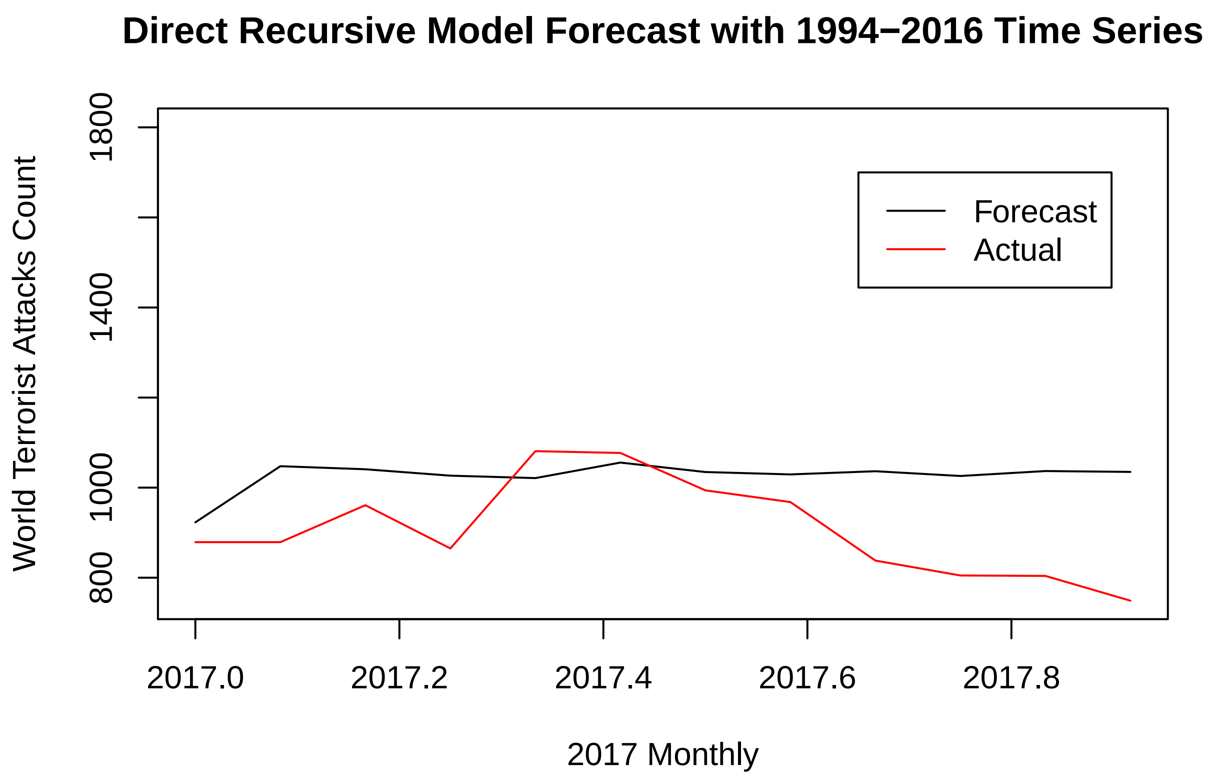

```
accuracy(fcst$mean, test) #MAPE score 17.09
```

```
##                    ME     RMSE      MAE       MPE     MAPE      ACF1
## Test set -134.1522 168.0606 143.6746 -16.20488 17.08779 0.5894664
##          Theil's U
## Test set  1.952262
```

```
train.xts<-as.xts(train) #direct reccursive model
for (month in 1:12) {
  dirrec<- Arima(train.xts, order=c(3,1,0), seasonal=c(1,0,0),include.drift = TRUE,lambda='auto')
  dirrec.fc<- forecast(dirrec, h=1)
  train.xts<-rbind(train.xts, dirrec.fc$mean)
  train.xts<-ts(train.xts, frequency = 12, start=c(1994, 1))
}
```

```
## Warning in rbind(deparse.level, ...): mismatched types: converting objects
## to numeric
```

```
train.xts<-window(train.xts, start=c(2017,1))

#plotting the direct reccursive forecast shows a closer fit
plot(train.xts,ylim=c(750,1800), main='Direct Recursive Model Forecast with 1994-2016 Time Series', ylal
lines(test, col='red')
legend(2017.65, 1700, legend=c("Forecast", 'Actual'), col=c('black','red'), lty=1, cex=1)
```




```r
accuracy(train.xts, test) # MAPE score 15.6
```

```
##                 ME     RMSE      MAE       MPE     MAPE      ACF1 Theil's U
## Test set -117.764 157.1804 131.2787 -14.36362 15.61504 0.4585521  1.903028
```

Arima model for terrorist attack from 1969 through 2017

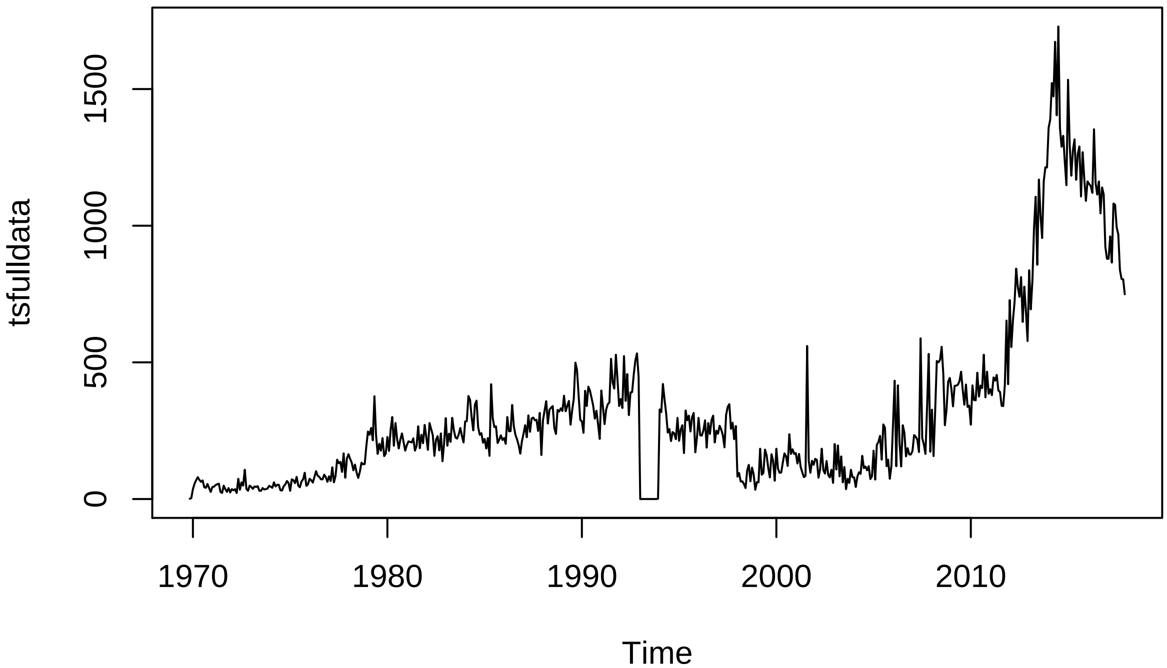
It was decided to try to fit an Arima model for the full time series. The data is the sum of all terorist attacks accross the world. Outlier analysis was used to estimate fill values for 1993.

```r
fulldata<-read.csv("~/Desktop/UofChicagoInfo/Time_Series/Final_Project/worldattacks.csv",
                   row.names = 'Date')
head(fulldata)
```

```
##            Australasia...Oceania Central.America...Caribbean Central.Asia
## 1969-11-30                     0                           0            0
## 1969-12-31                     0                           0            0
## 1970-01-31                     0                           1            0
## 1970-02-28                     0                           0            0
## 1970-03-31                     0                           3            0
## 1970-04-30                     0                           0            0
##            East.Asia Eastern.Europe Middle.East...North.Africa
## 1969-11-30         0              0                          0
## 1969-12-31         1              0                          0
## 1970-01-31         0              2                          0
## 1970-02-28         0              0                          1
## 1970-03-31         1              0                          2
## 1970-04-30         0              1                          2
##            North.America South.America South.Asia Southeast.Asia
## 1969-11-30             1             0          0              0
```

```
## 1969-12-31              0             0          0              1
## 1970-01-31             26             3          0              2
## 1970-02-28             50             0          0              1
## 1970-03-31             54             3          0              1
## 1970-04-30             71             2          0              2
##            Sub.Saharan.Africa Western.Europe Sum
## 1969-11-30                  0              0   1
## 1969-12-31                  0              1   3
## 1970-01-31                  1              1  36
## 1970-02-28                  0              4  56
## 1970-03-31                  2              3  69
## 1970-04-30                  0              2  80
```

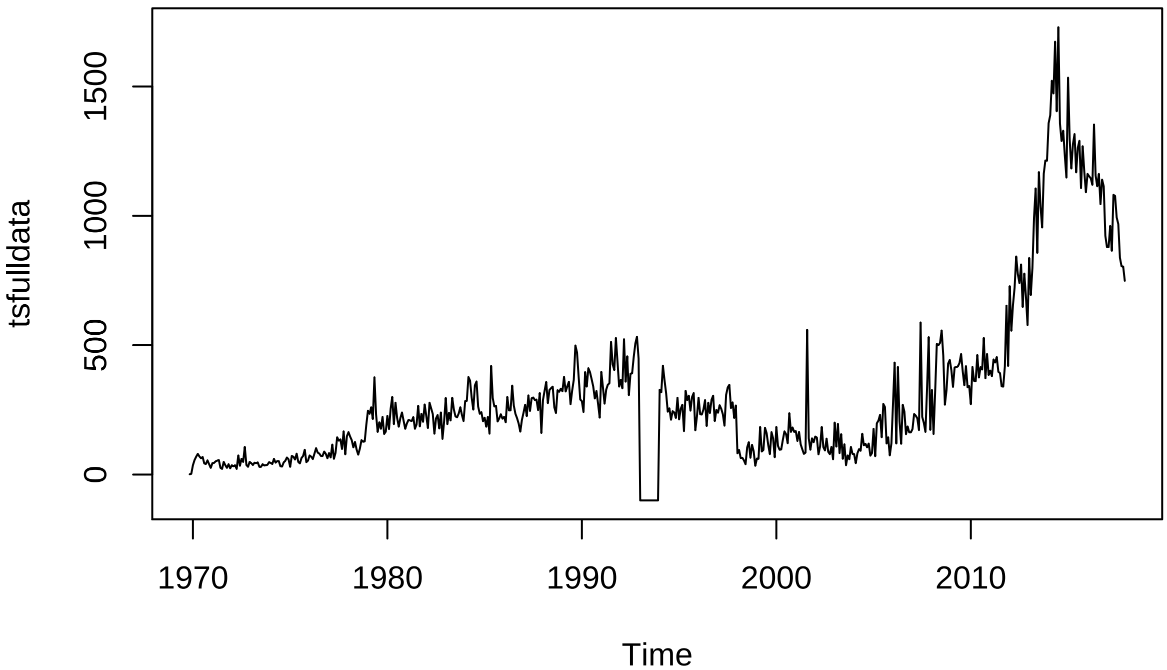
```r
tsfulldata<-ts(fulldata[,13], start = c(1969,11), end=c(2017,12), frequency=12)
plot(tsfulldata)
```




```r
tsfulldata[279:290]<--100
# an extreme value is used to set 1993 apart from the rest of the timeseries
tsfulldata[279:290]
```

```
##  [1] -100 -100 -100 -100 -100 -100 -100 -100 -100 -100 -100 -100
```

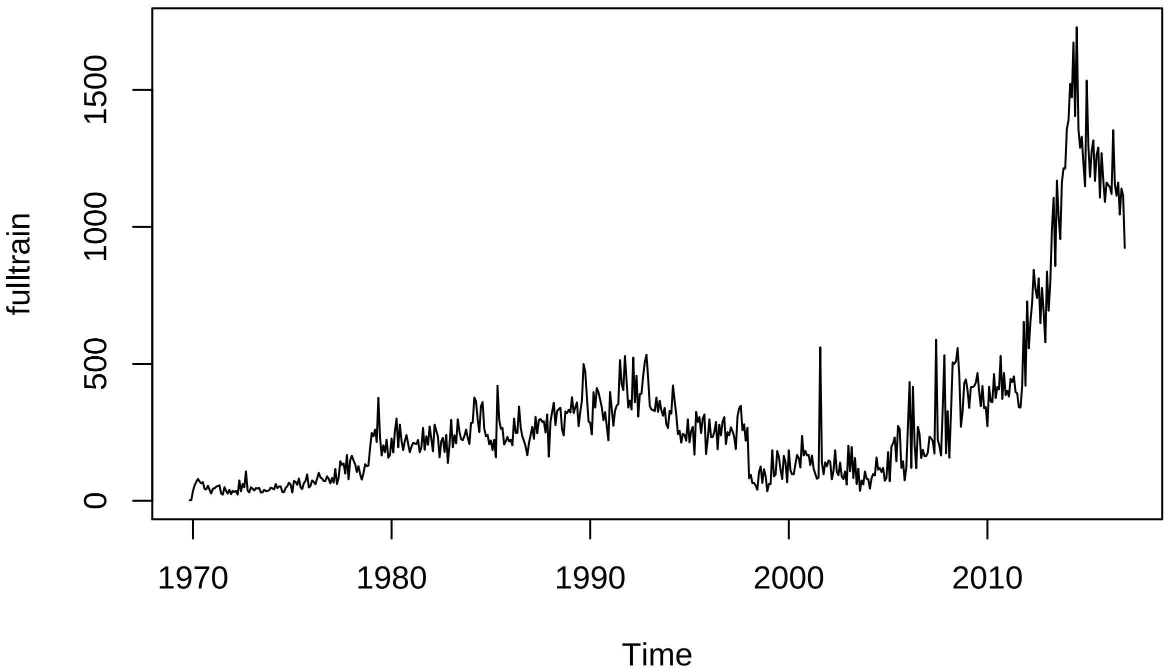
```r
plot(tsfulldata)
```



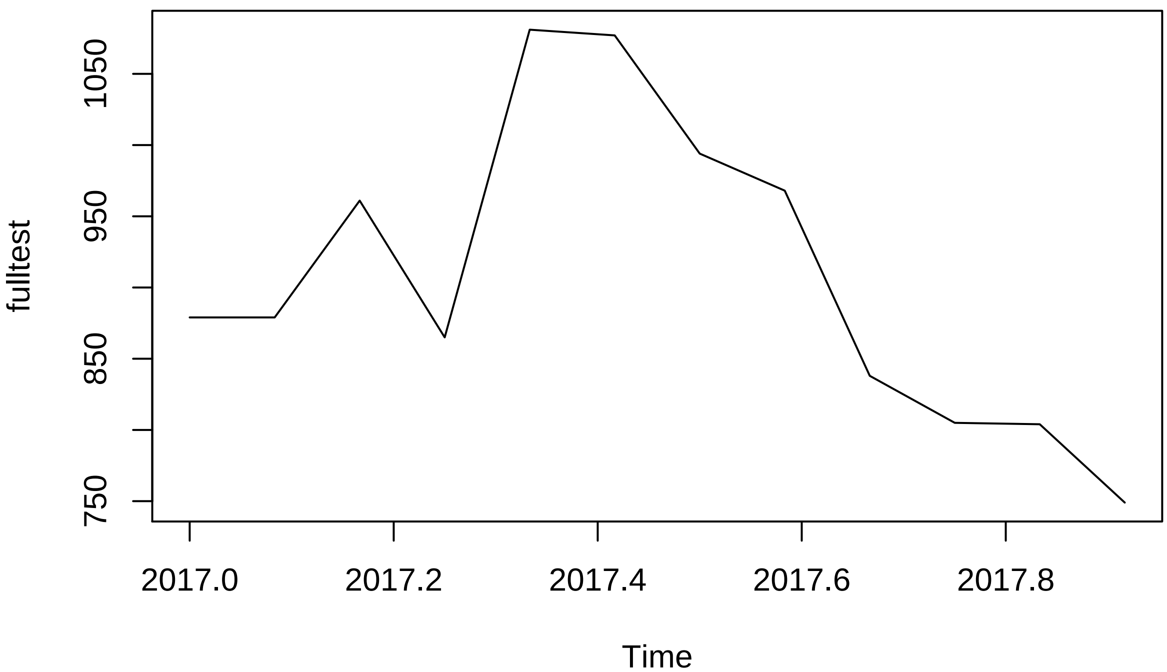

```
fulltrain<-window(tsfulldata, start=c(1969,11), end=c(2016,12), frequency=12)
fulltest<-window(tsfulldata, start=c(2017,1))

outliers<-tsoutliers(fulltrain, lambda='auto') #outliers are identified in the timeseries
outliers #outlier analysis identifies more than just 1993
```

```
## $index
##  [1] 269 275 276 277 278 279 280 281 282 283 284 285 286 287 288 289 290
## [18] 291 292 293 294 382
##
## $replacements
##  [1] 347.6385 361.0030 393.6366 340.9876 315.7355 345.9444 333.1190
##  [8] 331.3230 326.9273 376.9158 324.5525 364.7267 330.4509 310.3371
## [15] 339.4816 279.6183 265.6653 288.6494 275.3321 272.4516 268.0394
## [22] 119.4656
```

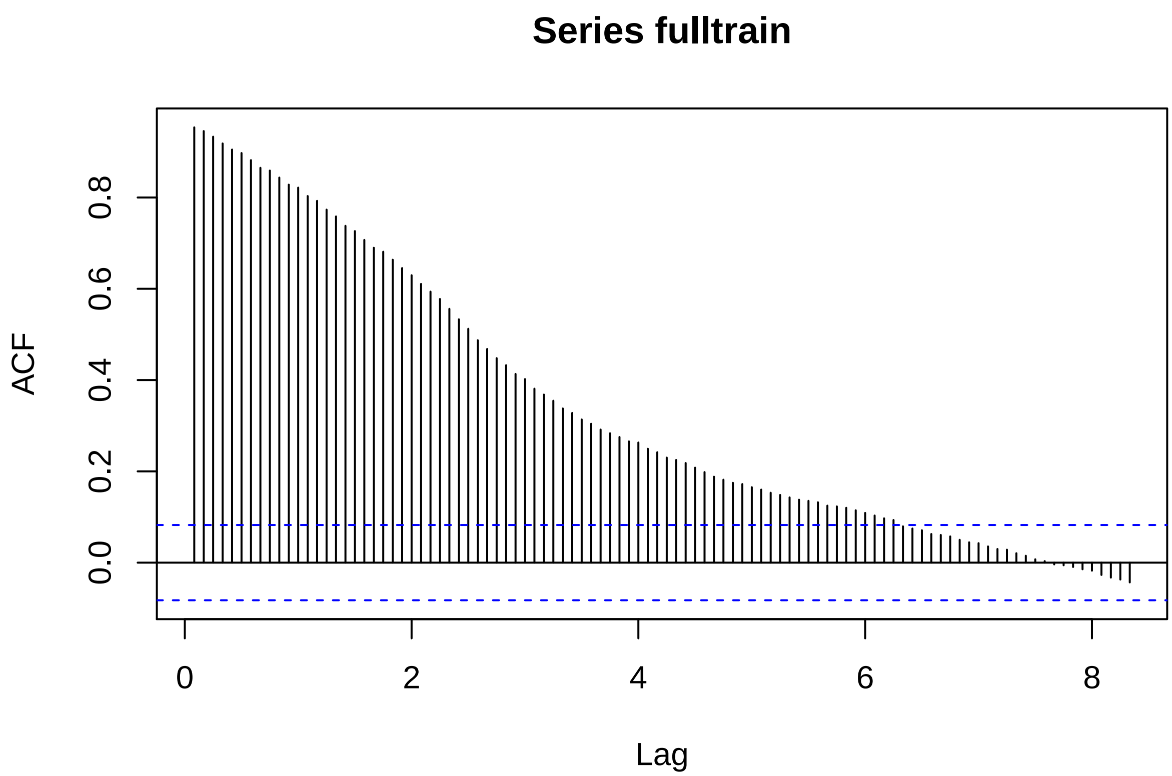
```
#values as outliers and proposes replacement values
fulltrain[279:290]<-outliers$replacements[6:17] #only 1994 values are replaced



plot(fulltrain)
```



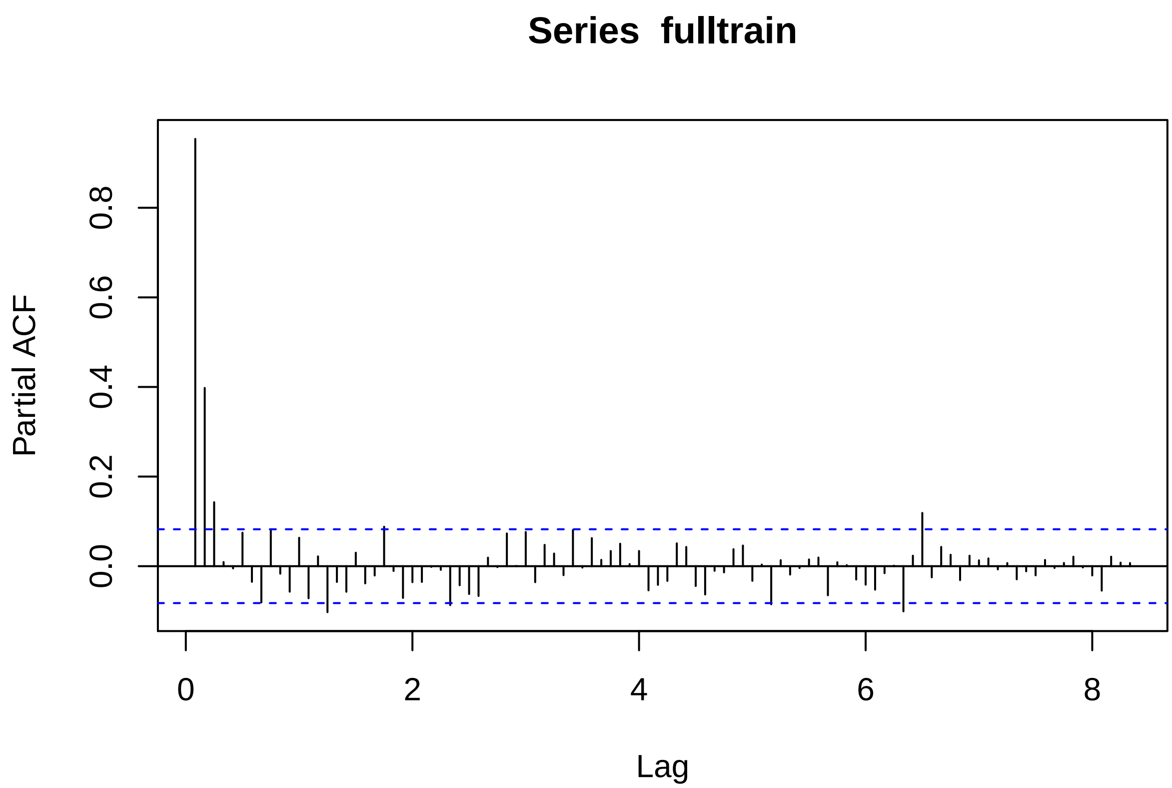

```r
plot(fulltest)
```




```r
acf(fulltrain,lag=100) # still suggesting an AR model
```




```r
pacf(fulltrain, lag=100)
```




```r
adf.test(fulltrain) #no stationarity, implying drift
```

```
##
```

```
##  Augmented Dickey-Fuller Test
##
## data:  fulltrain
## Dickey-Fuller = -1.1297, Lag order = 8, p-value = 0.9178
## alternative hypothesis: stationary
```

```
adf.test(fulltrain, alternative='explosive')
```

```
##
##  Augmented Dickey-Fuller Test
##
## data:  fulltrain
## Dickey-Fuller = -1.1297, Lag order = 8, p-value = 0.08221
## alternative hypothesis: explosive
```

```
kpss.test(fulltrain)
```

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

```
##
##  KPSS Test for Level Stationarity
##
## data:  fulltrain
## KPSS Level = 3.2554, Truncation lag parameter = 6, p-value = 0.01
```

```
kpss.test(fulltrain, null='Trend')
```

```
## Warning in kpss.test(fulltrain, null = "Trend"): p-value smaller than
## printed p-value
```

```
##
##  KPSS Test for Trend Stationarity
##
## data:  fulltrain
## KPSS Trend = 0.85961, Truncation lag parameter = 6, p-value = 0.01
```

```
summary(auto.arima(fulltrain)) #very high AIC and BIC scores with MAPE=23.7
```

```
## Series: fulltrain
## ARIMA(0,1,1)(0,0,2)[12]
##
## Coefficients:
##           ma1    sma1    sma2
##       -0.5844  0.1173  0.0727
## s.e.   0.0330  0.0424  0.0376
##
## sigma^2 estimated as 5827:  log likelihood=-3249.87
## AIC=6507.74   AICc=6507.82   BIC=6525.09
##
## Training set error measures:
##                    ME     RMSE     MAE       MPE     MAPE      MASE
## Training set 3.596095 76.06238 51.1038 -5.981913 23.77257 0.5874303
##                     ACF1
## Training set -0.03724038
```

```
summary(auto.arima(fulltrain, lambda = 'auto'))
```

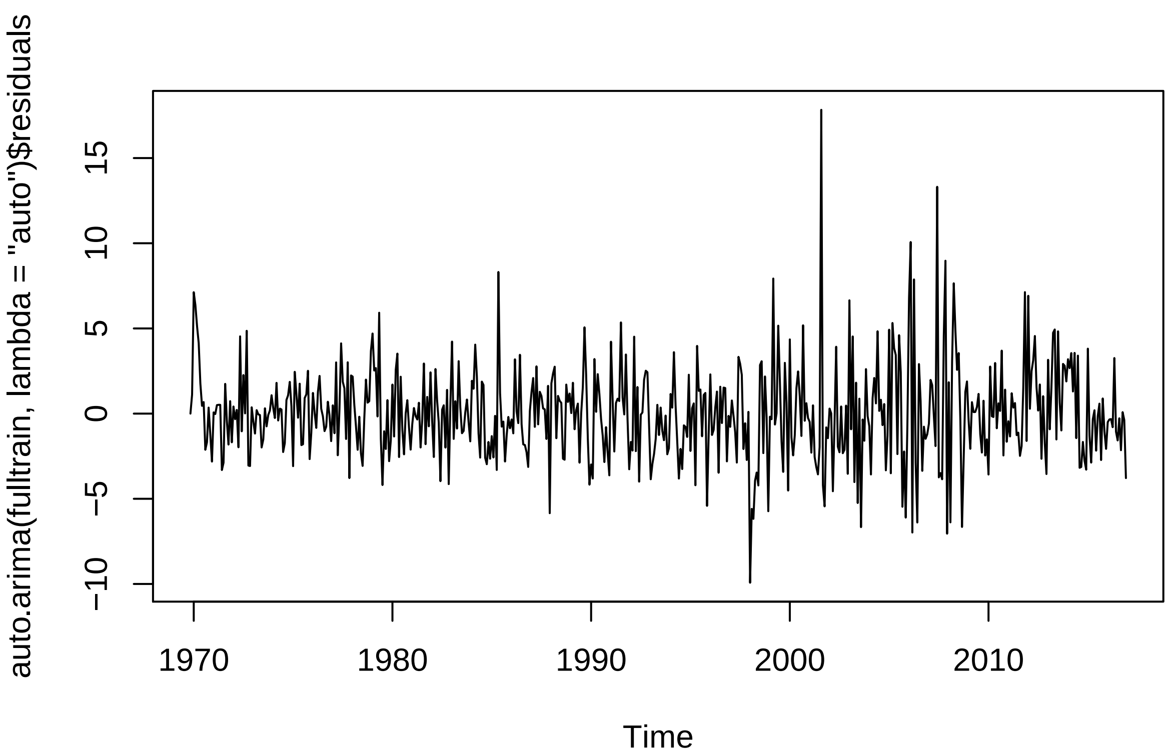
```
## Series: fulltrain
## ARIMA(0,1,1)(1,0,0)[12] with drift
```

```
## Box Cox transformation: lambda= 0.4245822
##
## Coefficients:
##           ma1    sar1   drift
##       -0.6493  0.0549  0.0680
## s.e.   0.0356  0.0426  0.0438
##
## sigma^2 estimated as 7.873:  log likelihood=-1383.41
## AIC=2774.82   AICc=2774.89   BIC=2792.16
##
## Training set error measures:
##                    ME     RMSE      MAE       MPE     MAPE      MASE
## Training set 3.71552 76.49694 51.21294 -5.929543 23.39871 0.5886849
##                    ACF1
## Training set 0.04632927
```

```r
# box-cox transformaiton allows for a lower AIC and BIC scores,
#but MAPE does not change much

Box.test(auto.arima(fulltrain, lambda = 'auto')$residuals, type=c('Ljung-Box'))
```

```
##
##  Box-Ljung test
##
## data:  auto.arima(fulltrain, lambda = "auto")$residuals
## X-squared = 0.067087, df = 1, p-value = 0.7956
```

```r
#residuals are independant

plot(auto.arima(fulltrain, lambda = 'auto')$residuals)
```



```r
m1<-Arima(fulltrain, order=c(3,1,0), seasonal=c(1,0,0),include.drift = TRUE,lambda='auto')
#creating an AR model with differencing
m3<-Arima(fulltrain, order=c(0,1,1), seasonal=c(0,1,1),lambda='auto')
#creating an auto arima recommended model
summary(m1) #MAPE= 23.4
```

```
## Series: fulltrain
## ARIMA(3,1,0)(1,0,0)[12] with drift
## Box Cox transformation: lambda= 0.4245822
##
## Coefficients:
##           ar1      ar2      ar3    sar1   drift
##       -0.6127  -0.3212  -0.1350  0.0475  0.0708
## s.e.   0.0419   0.0478   0.0423  0.0427  0.0603
##
## sigma^2 estimated as 8.055:  log likelihood=-1388.8
## AIC=2789.6   AICc=2789.75   BIC=2815.62
##
## Training set error measures:
##                    ME     RMSE      MAE      MPE     MAPE      MASE
## Training set 2.736755 76.13459 51.14963 -5.82275 23.43086 0.5879571
##                     ACF1
## Training set 0.004068574
```

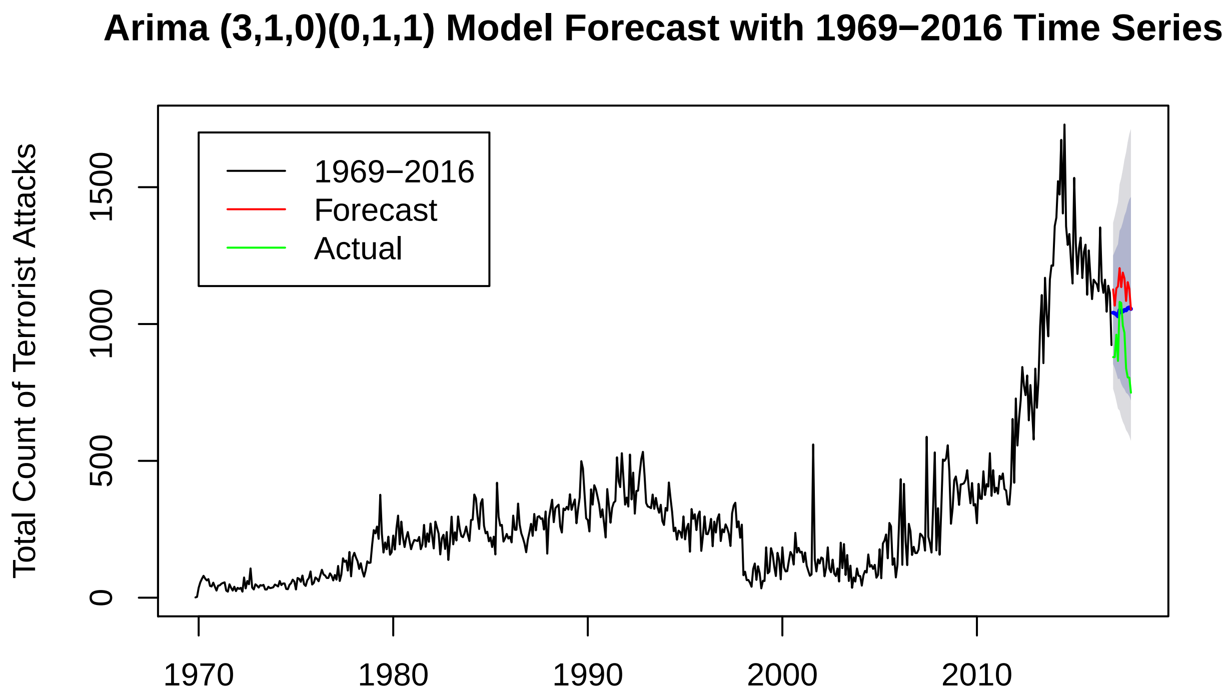
```r
summary(m3) # AIC and BIC scores are slightly lowel for this model
```

```
## Series: fulltrain
## ARIMA(0,1,1)(0,1,1)[12]
## Box Cox transformation: lambda= 0.4245822
##
## Coefficients:
##           ma1     sma1
##       -0.6428  -0.9539
## s.e.   0.0351   0.0254
##
## sigma^2 estimated as 7.622:  log likelihood=-1359.9
## AIC=2725.81   AICc=2725.85   BIC=2738.76
##
## Training set error measures:
##                    ME    RMSE      MAE       MPE     MAPE     MASE
## Training set 2.898047 72.8353 48.90852 -6.706486 23.01077 0.562196
##                    ACF1
## Training set 0.07310496
```

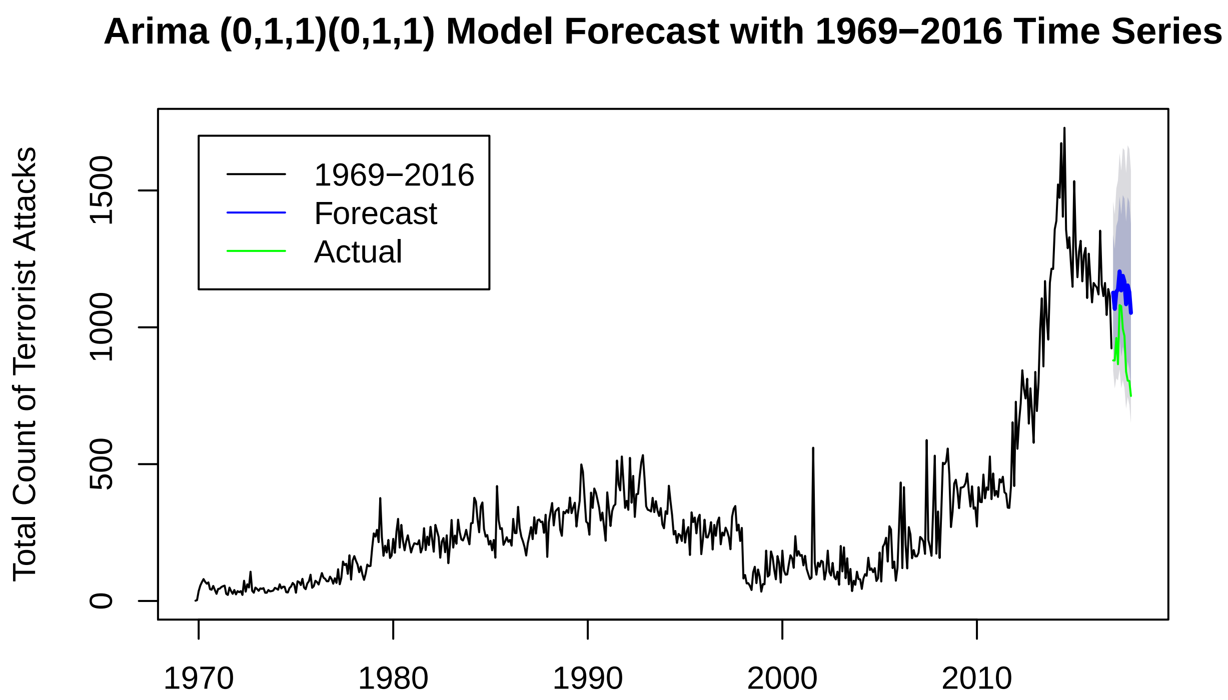
```r
fullfcst1<-forecast(m1, h=12 )
fullfcst2<-forecast(m3, h=12)

Box.test(fullfcst2$residuals, type=c('Ljung-Box')) # residuals are independant
```

```
##
##  Box-Ljung test
##
## data:  fullfcst2$residuals
## X-squared = 0.11083, df = 1, p-value = 0.7392
```

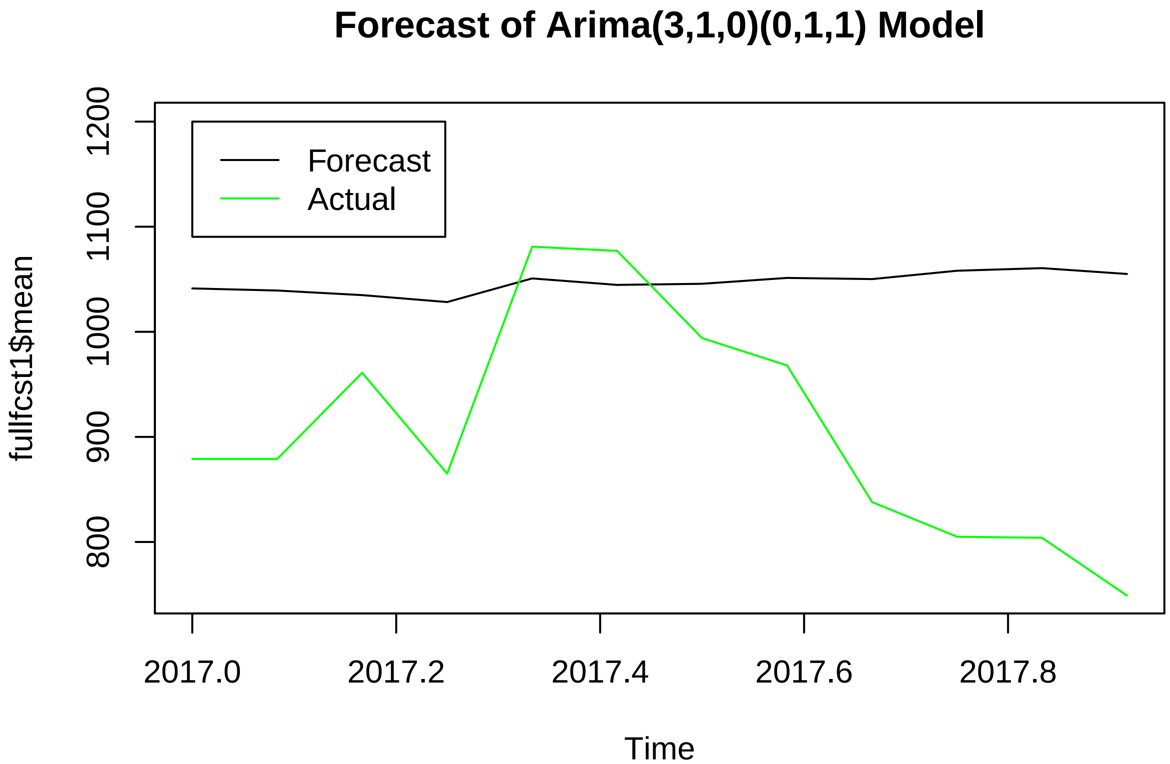
```r
plot(fullfcst1, main='Arima (3,1,0)(0,1,1) Model Forecast with 1969-2016 Time Series', ylab='Total Coun
lines(fullfcst2$mean, col='red')
lines(fulltest, col='green')
legend(1970, 1700, legend=c("1969-2016", 'Forecast', 'Actual'), col=c('black','red','green'), lty=1, ce
```




```r
plot(fullfcst2, main='Arima (0,1,1)(0,1,1) Model Forecast with 1969-2016 Time Series', ylab='Total Coun
lines(fulltest, col='green')
legend(1970, 1700, legend=c("1969-2016", 'Forecast', 'Actual'), col=c('black','blue','green'), lty=1, c
```



```r
plot(fullfcst1$mean,ylim=c(750,1200), main='Forecast of Arima(3,1,0)(0,1,1) Model') #AR model forecast
lines(fulltest, col='green') #actual values
legend(2017.0, 1200, legend=c('Forecast', 'Actual'), col=c('black','green'), lty=1, cex=1)
```



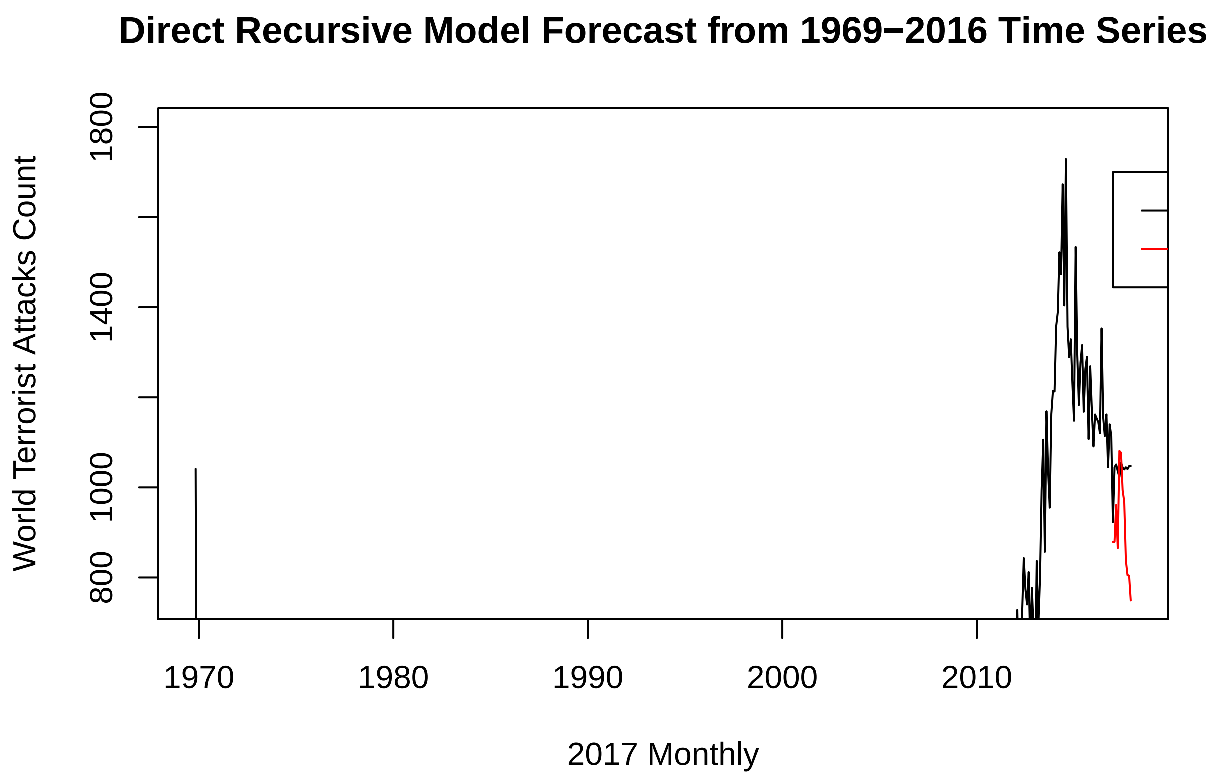

```r
accuracy(fullfcst1$mean, fulltest) #AR MAPE=17.7
```

```
##                    ME     RMSE      MAE       MPE     MAPE     ACF1 Theil's U
## Test set -138.3426 174.2376 148.7679 -16.73401 17.70027 0.589988  2.041191
```

```r
accuracy(fullfcst2$mean, fulltest) #MA MAPE = 25.75
```

```
##                    ME     RMSE      MAE       MPE     MAPE     ACF1 Theil's U
## Test set -222.8302 237.1199 222.8302 -25.75483 25.75483 0.489722  2.675834
```

```r
train.xts1<-as.xts(fulltrain) #direct reccursive model
for (month in 1:12) {
  dirrec<- Arima(train.xts1, order=c(3,1,0), seasonal=c(1,0,0),include.drift = TRUE,lambda='auto')
  dirrec.fc<- forecast(dirrec, h=1)
  train.xts1<-rbind(train.xts1, dirrec.fc$mean)
  train.xts1<-ts(train.xts1, frequency = 12, start=c(1969, 11))
}


plot(train.xts1,ylim=c(750,1800), main='Direct Recursive Model Forecast from 1969-2016 Time Series',ylal
lines(fulltest, col='red')
legend(2017.0, 1700, legend=c("Forecast", 'Actual'), col=c('black','red'), lty=1, cex=1)
```




```r
#Plot is refusing to output correctly in pdf foremat

accuracy(train.xts1, fulltest) # MAPE score 16.42
```

```
##                 ME     RMSE    MAE       MPE     MAPE      ACF1 Theil's U
## Test set -124.7453 164.5845 138.17 -15.17714 16.42029 0.4729929  1.991901
```

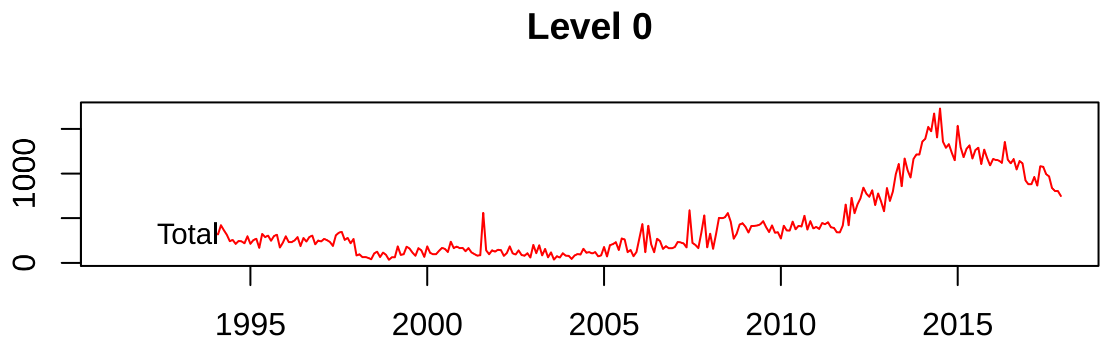
Hierarchical model for terrorist acts after 1994

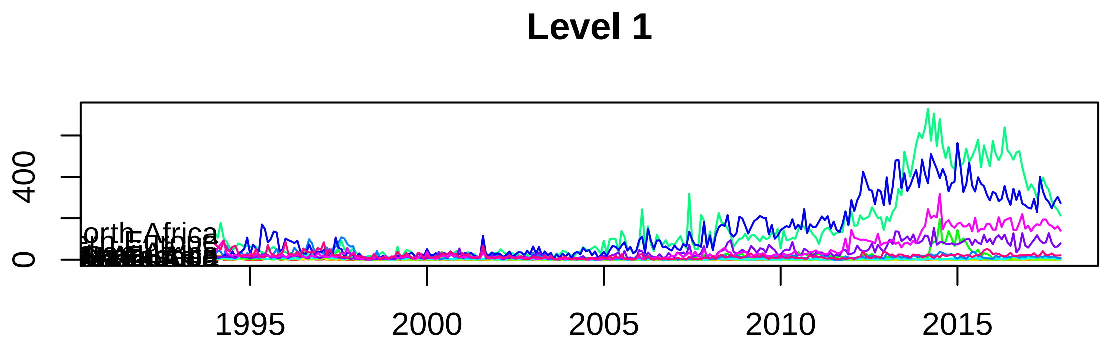
The hierarchical model was built on 2 levels: Total terrorist attacks per world region and a summation of all terrorist attacks in the world per month since 1994. Again this data set is being reduced, due to missing values for 1994.

```r
#https://github.com/earowang/hts sample hts models


ts_hdata<-ts(data[,0:12], start = c(1994,1), end=c(2017,12), frequency=12)
hdata<-hts(ts_hdata)
```

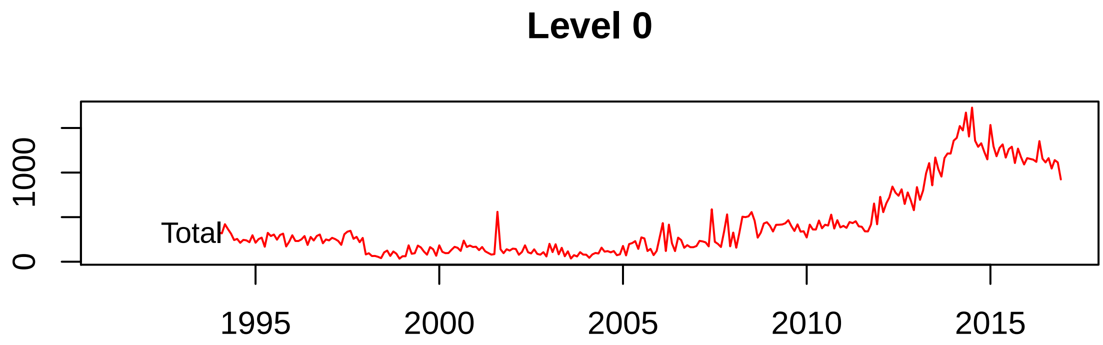
```
## Since argument characters are not specified, the default labelling system is used.
```

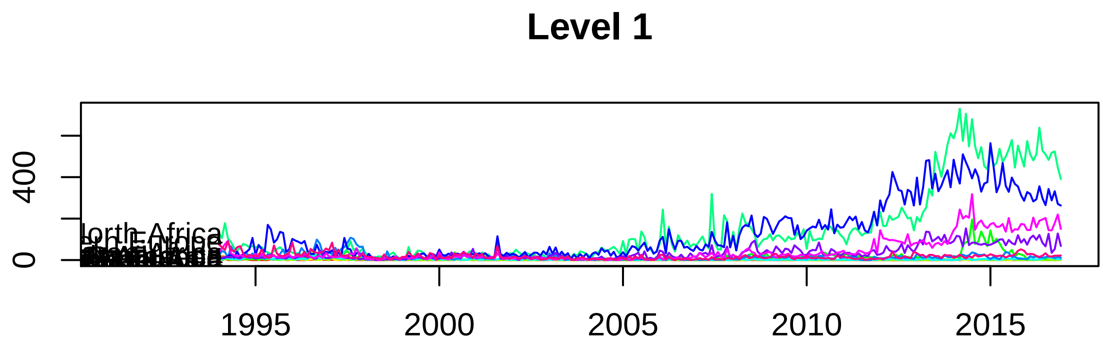
```r
plot(hdata)
```






```
hdata.train<-window(hdata, start=c(1994,1), end=c(2016,12), frequency=12)
hdata.test<-window(hdata, start=c(2017,1))
plot(hdata.train)
```

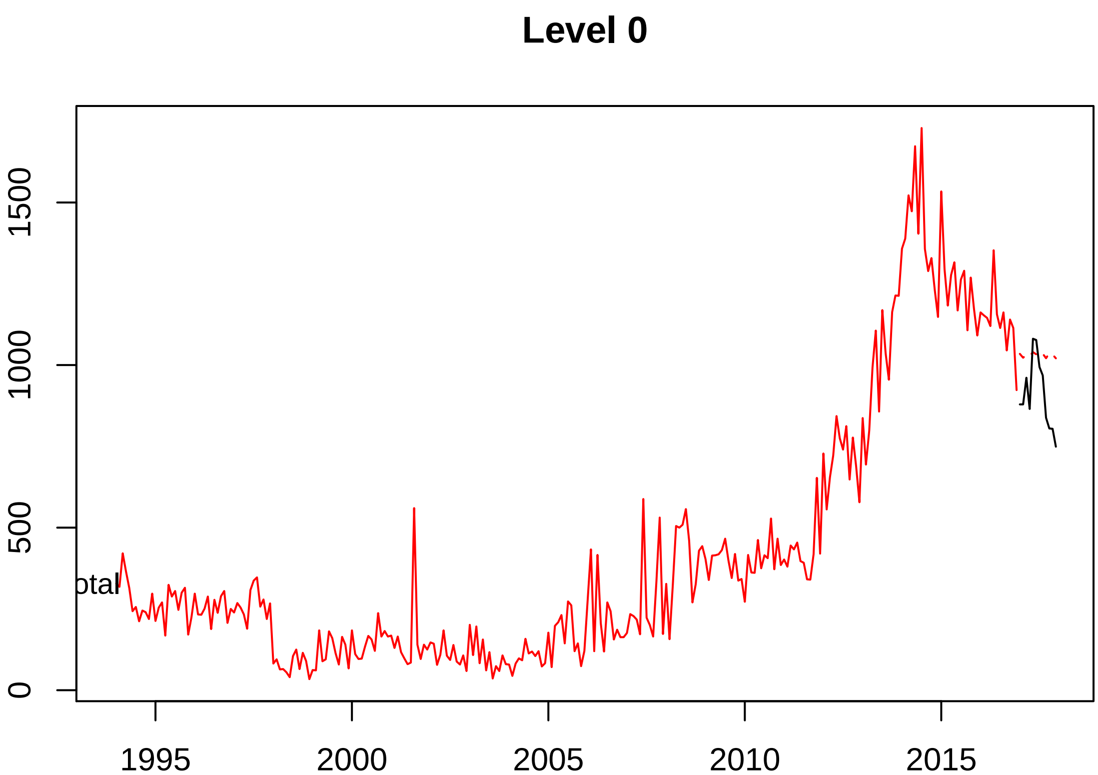




```r
hdatafcst<-forecast(hdata.train, h=12, method="bu", fmethod="arima")
#bottom up approach is used in this case, as using historical data,
#not too many divisions and no mid-level to use the middle out method.

hdatafcst
```

```
## Hierarchical Time Series
## 2 Levels
## Number of nodes at each level: 1 12
## Total number of series: 13
## Number of observations in each historical series: 276
## Number of forecasts per series: 12
## Top level series of forecasts:
##           Jan      Feb      Mar      Apr      May      Jun      Jul
## 2017 1034.474 1022.597 1031.509 1027.382 1039.847 1033.104 1026.720
##           Aug      Sep      Oct      Nov      Dec
## 2017 1034.927 1021.016 1036.690 1032.286 1020.725
```

```r
plot(hdatafcst, levels=0)
lines(aggts(hdata.test, level=0))
```




```r
accuracy(hdatafcst,hdata.test, levels=0) #MAPE is 16.13
```

```
##            Total
## ME   -121.773104
## RMSE  157.698543
## MAE   135.947965
## MAPE   16.126260
## MPE   -14.812474
## MASE    1.093049
```

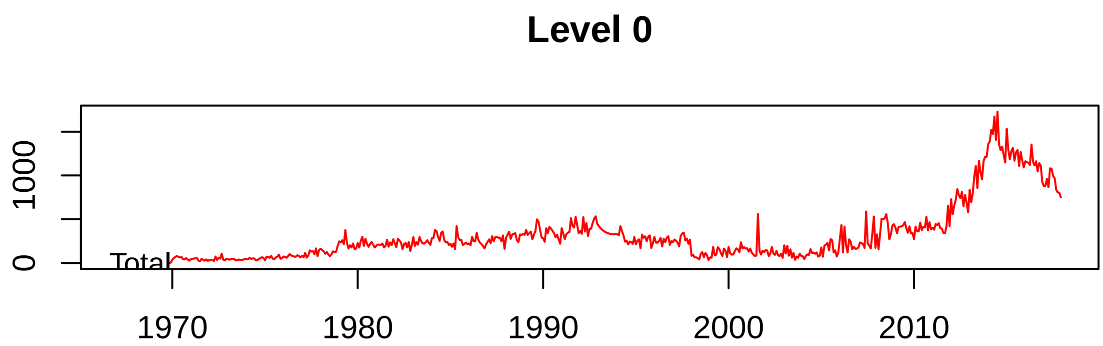
Hierarchical model for terrorist acts between 1969 and 2017

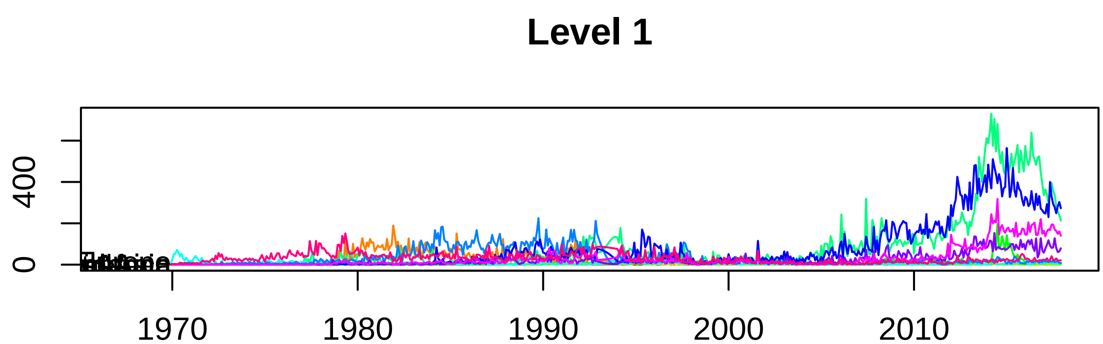
The hierarchical model was built on 2 levels: Total terrorist attacks per world region and a summation of all terrorist attacks in the world per month between 1969 through 2017. Missing value imputation was used to estimate and fill the missing values for 1993.

```
ts_fullhdata<-ts(fulldata[,0:12], start = c(1969,11), end=c(2017,12), frequency=12)

for (i in 1:12) {
  ts_fullhdata[,i][279:290]<- NA
  ts_fullhdata[,i]<-na.interpolation(ts_fullhdata[,i], option = "stine")

} #filling missing values for 1993 with estimation

fullhdata<-hts(ts_fullhdata)
```

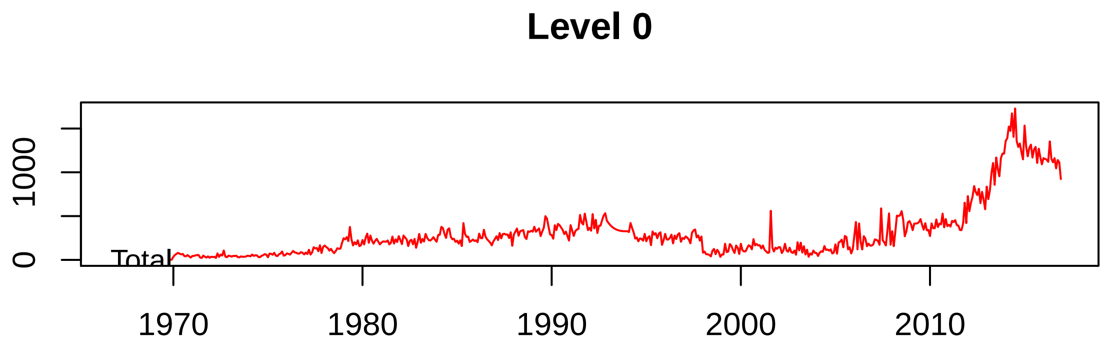
```
## Since argument characters are not specified, the default labelling system is used.
```

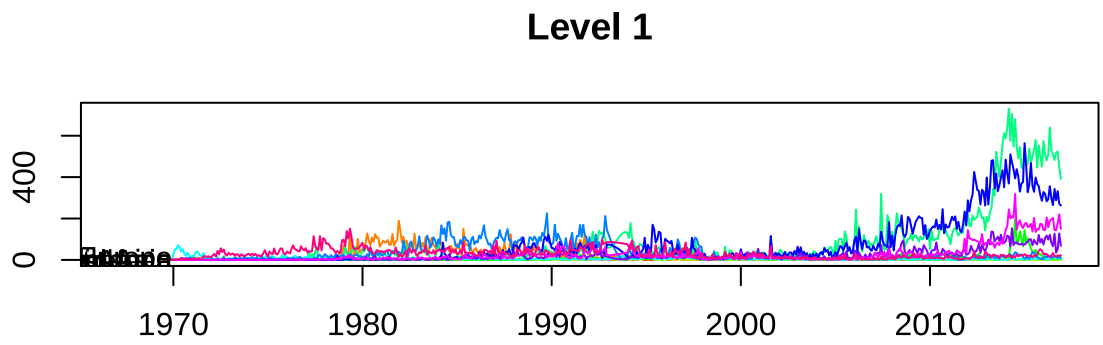
```
plot(fullhdata)
```






```
fullhdata.train<-window(fullhdata, start=c(1969,11), end=c(2016,12), frequency=12)
fullhdata.test<-window(fullhdata, start=c(2017,1))
plot(fullhdata.train)
```





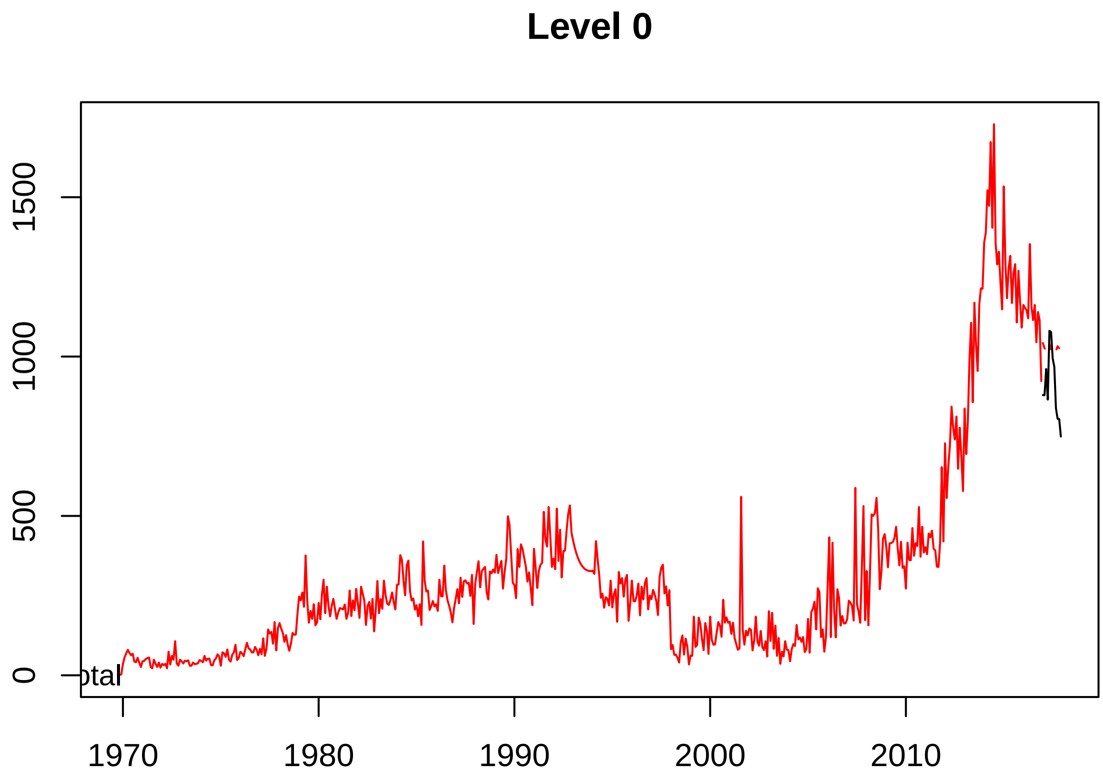

```r
fullhdatafcst<-forecast(fullhdata.train, h=12, method="bu", fmethod="arima")
#again using an arima model with a bottom up approach for hierarchical modeling
fullhdatafcst
```

```
## Hierarchical Time Series
## 2 Levels
## Number of nodes at each level: 1 12
## Total number of series: 13
## Number of observations in each historical series: 566
## Number of forecasts per series: 12
## Top level series of forecasts:
##           Jan      Feb      Mar      Apr      May      Jun      Jul
## 2017 1042.540 1026.098 1020.290 1018.521 1039.217 1022.064 1023.330
##           Aug      Sep      Oct      Nov      Dec
## 2017 1029.292 1017.683 1032.588 1026.158 1016.309
```

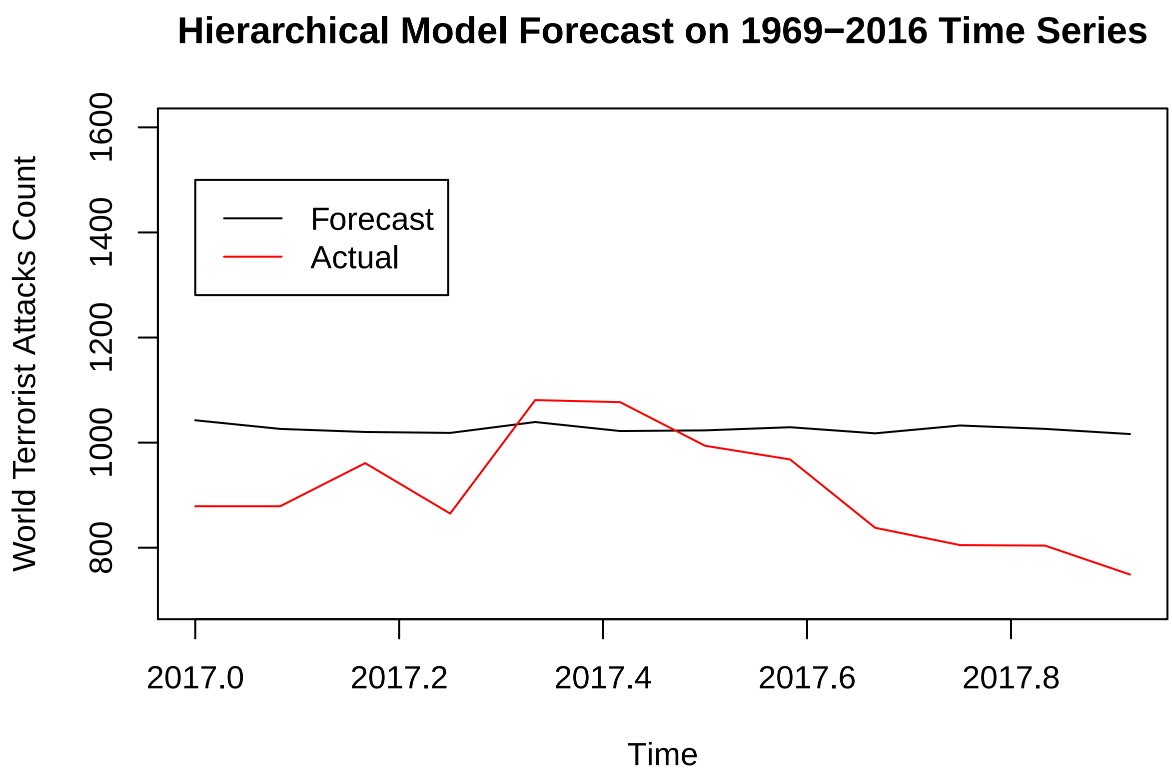
```r
accuracy(fullhdatafcst,fullhdata.test, levels=0) # MAPE 15.87
```

```
##             Total
## ME    -117.840899
## RMSE   155.386220
## MAE    133.960699
## MAPE    15.878361
## MPE    -14.384021
## MASE     1.537724
```

```r
plot(fullhdatafcst, levels=0)
lines(aggts(hdata.test, level=0))
```




```
modeltest<-aggts(fullhdatafcst)


plot(modeltest[,1], ylim=c(700,1600), main = "Hierarchical Model Forecast on 1969-2016 Time Series", yla
lines(test, col='red')
legend(2017.0, 1500, legend=c("Forecast", 'Actual'), col=c('black','red'), lty=1, cex=1)
```

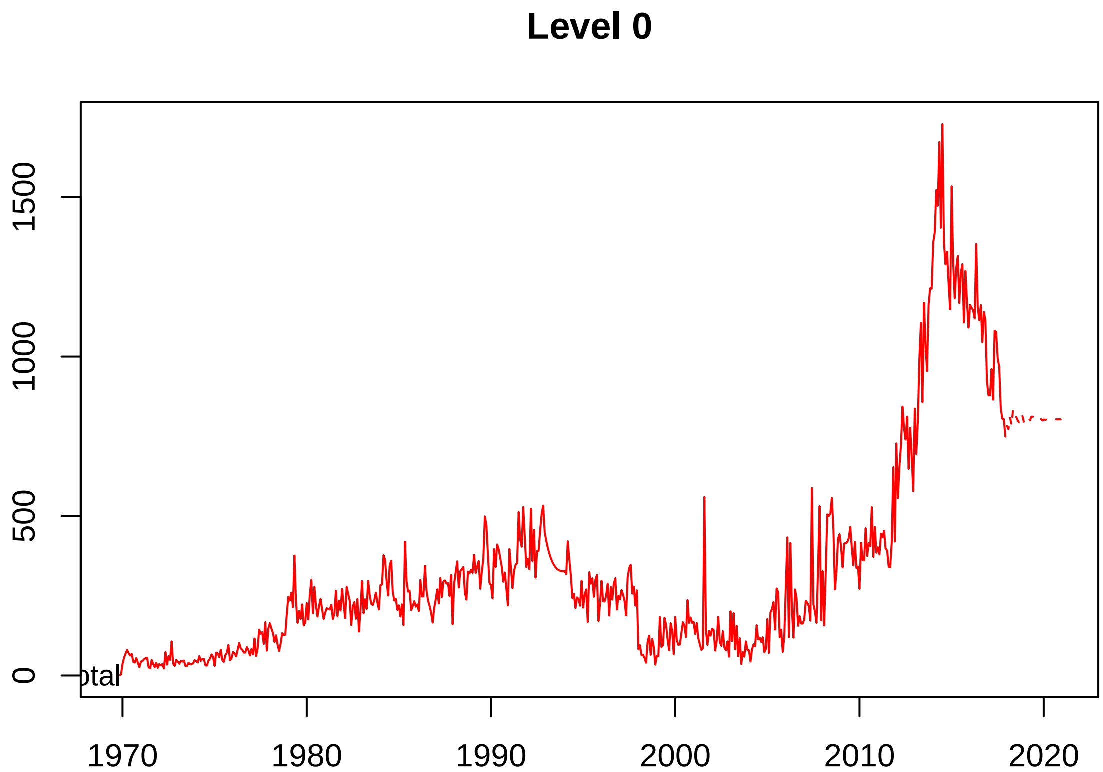



```
#plot prediction of total terrorist attacks on the world in 2017,
#predicted vs actual
```

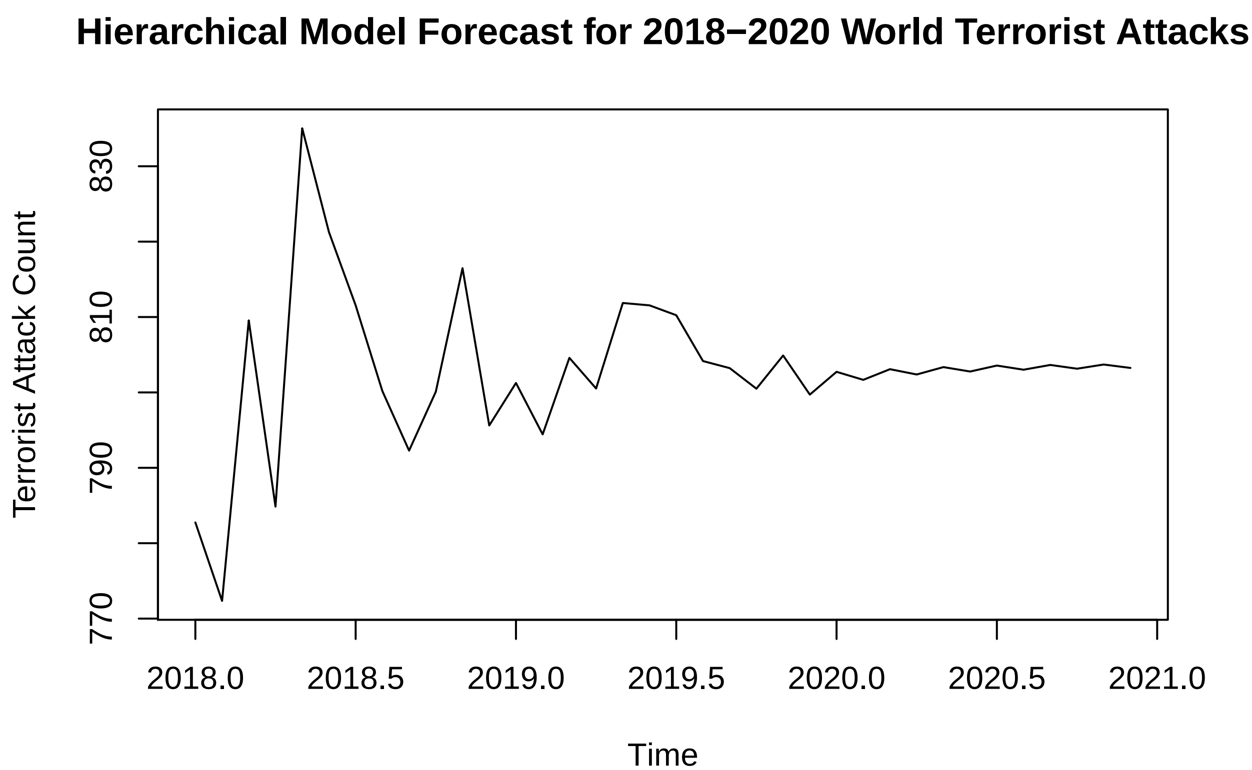
Final Model

Direct reccursive model for 1994 through 2017 data showed the same results as the hierarchical model with the full time series. Both use the Arima model and MAPE was at 15%. The hierarchial model was chosen as it allows to predict for multiple timeseries at the same time. Also the full time period may allow for better prediction for a longer time period.

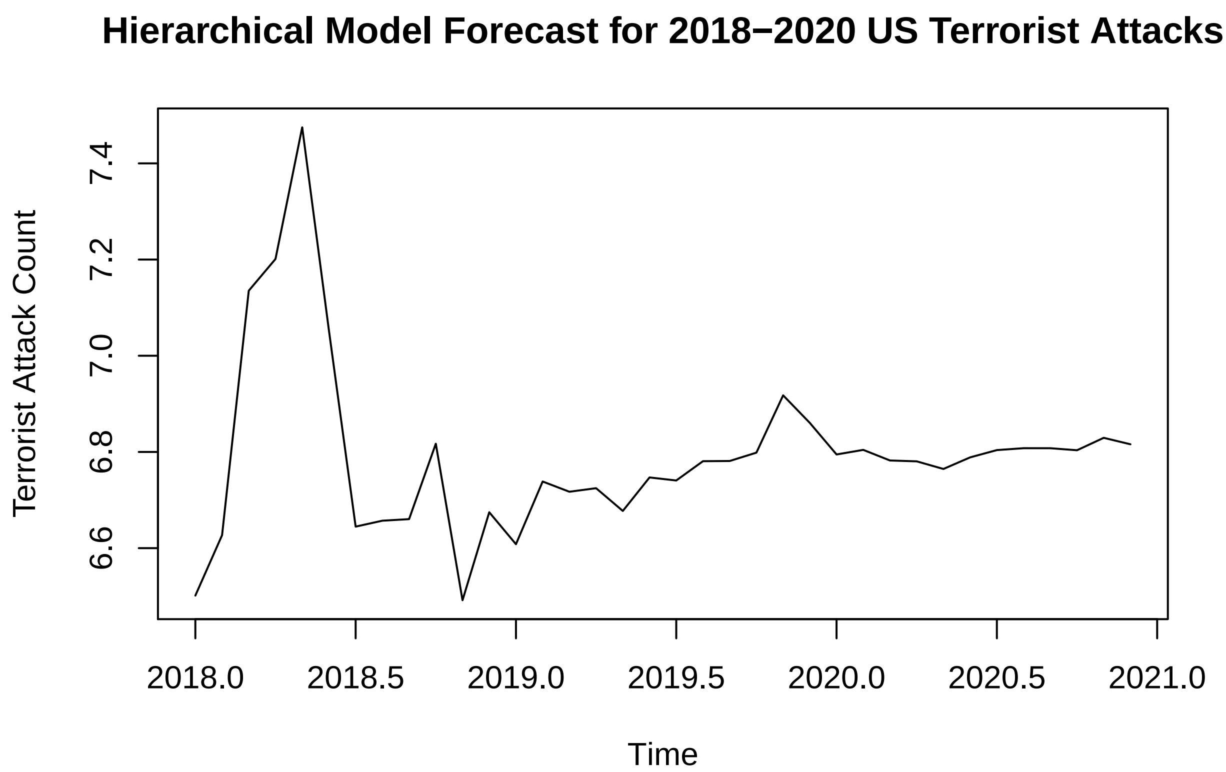
```
finalfcst<-forecast(fullhdata, h=36, method='bu', fmethod='arima')
# using the full data set from 1969-2017 to predict terroristic
#acts from 2018 through 2020
plot(finalfcst, levels=0) #full time series with forecast
```




```
final<-aggts(finalfcst)
plot(final[,1], main='Hierarchical Model Forecast for 2018-2020 World Terrorist Attacks', ylab='Terrori
```




```
plot(final[,8],  main='Hierarchical Model Forecast for 2018-2020 US Terrorist Attacks', ylab='Terrorist
```


Hierarchical Model Forecast for 2018−2020 US Terrorist Attacks
Terrorist Attack Count
Time
7.4
7.2
7.0
6.8
6.6
2018.0
2018.5
2019.0
2019.5
2020.0
2020.5
2021.0