ISyE 6402 Project

Siddharth Sen

11/04/2022

Part II - ARIMA Modelling

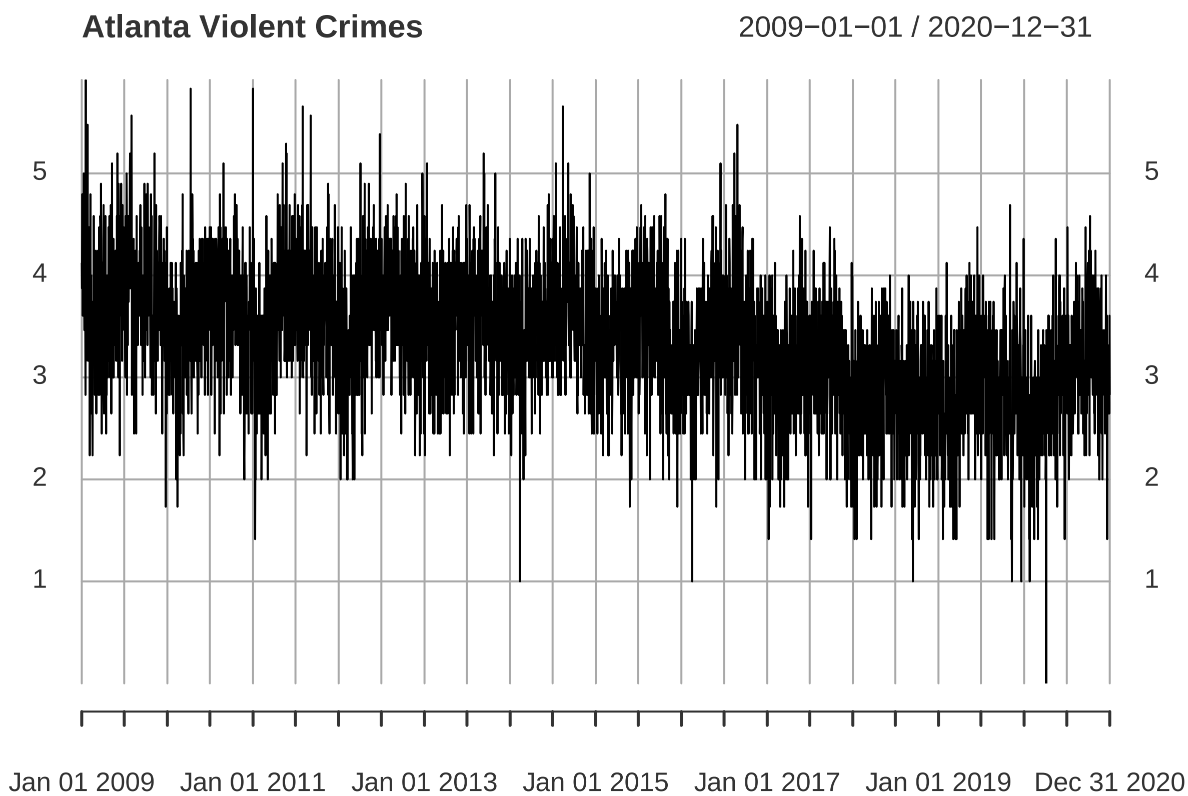
```
library(zoo)
library(xts)
library(lubridate)
library(mgcv)
library(lmtest)
```

Load data

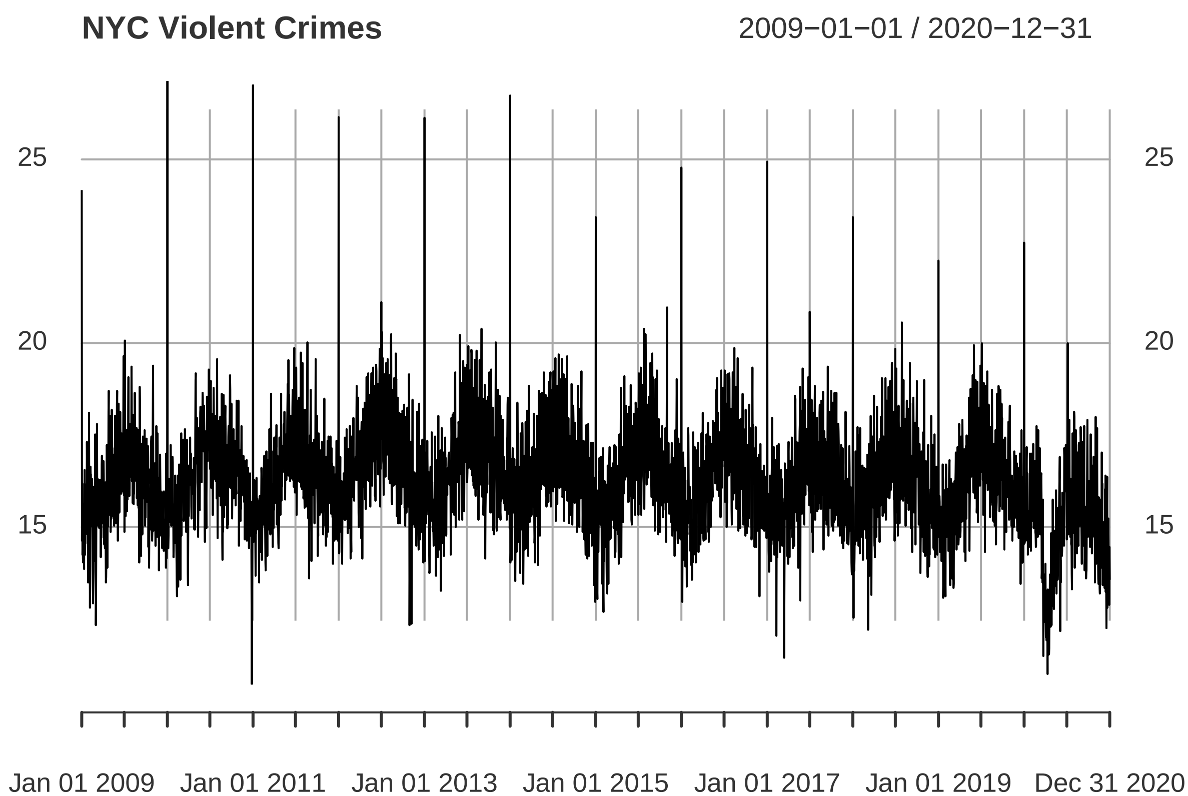
```
## Atlanta
atl.v.df <- read.csv("atl_violent_final.csv", head = TRUE)
atl.v.df$occurance_count <- sqrt(atl.v.df$occurance_count)
atl.p.df <- read.csv("atl_prop_final.csv", head = TRUE)
atl.p.df$occurance_count <- sqrt(atl.p.df$occurance_count)
colnames(atl.v.df) <- c("Date", "violentCrime")
colnames(atl.p.df) <- c("Date", "propertyCrime")

atl.df <- merge(atl.v.df, atl.p.df)
atl.df$Date <- as.Date(atl.df$Date, "%Y-%m-%d")
atl.df <- atl.df[atl.df$Date <= "2020-12-31", ]
atl.df <- na.locf(atl.df)

## New York City
nyc.v.df <- read.csv("nyc_violent_final.csv", head = TRUE)
nyc.v.df$occurance_count <- sqrt(nyc.v.df$occurance_count)
nyc.p.df <- read.csv("nyc_prop_final.csv", head = TRUE)
nyc.p.df$occurance_count <- sqrt(nyc.p.df$occurance_count)
colnames(nyc.v.df) <- c("Date", "violentCrime")
colnames(nyc.p.df) <- c("Date", "propertyCrime")

nyc.df <- merge(nyc.v.df, nyc.p.df)
nyc.df$Date <- as.Date(nyc.df$Date, "%Y-%m-%d")
nyc.df <- nyc.df[nyc.df$Date >= "2009-01-01", ]
nyc.df <- na.locf(nyc.df)
```

Plot Time Series

```
## ATL TS
atl.v.ts <- ts(atl.df$violentCrime, start = 2009, freq = 365.25)
plot(xts(atl.df$violentCrime, atl.df$Date), main="Atlanta Violent Crimes", lwd=1)
```




```
## NYC TS
nyc.v.ts <- ts(nyc.df$violentCrime, start = 2009, freq = 365.25)
plot(xts(nyc.df$violentCrime, nyc.df$Date), main="NYC Violent Crimes", lwd=1)
```



ARIMA Fitting and Forecasting

Test set = 2022 data (from Jan 1 2022 - end) Training Set = All previous data

Test Train Split

```
## X-axis points converted to 0-1 scale, common in nonparametric regression
scaler <- function(ts) {
  ts.pts = c(1:length(ts))
  ts.pts = c(ts.pts - min(ts.pts))/max(ts.pts)
  return(ts.pts)
}

train.ind = c(1:which(atl.df$Date == "2020-12-24"))

atl.train <- atl.df[train.ind, ]
atl.test <- atl.df[-train.ind, ]
nyc.train <- nyc.df[train.ind, ]
nyc.test <- nyc.df[-train.ind, ]
```

```r
atl.v.train <- ts(atl.train$violentCrime, start = 2009, freq = 365.25)

# Function to train ARIMA (p, d, q) Model
test_modelA <- function(ts, p, d, q) {
  mod = arima(ts, order = c(p, d, q), method = "ML")
  current.aic = AIC(mod)
  df = data.frame(p, d, q, current.aic)
  names(df) <- c("p","d","q","AIC")
  # print(paste(p,d,q,current.aic,sep=" "))
  return(df)
}

# Daily TS ARIMA (p, d, q) Fitting
atl.v.orders = data.frame(Inf, Inf, Inf, Inf)
names(atl.v.orders) <- c("p", "d", "q", "AIC")

for (p in 0:6) {
  for (d in 1:2) {
    for (q in 0:6) {
      possibleError <- tryCatch(
        atl.v.orders <- rbind(atl.v.orders, test_modelA(atl.v.train,p,d,q)),
        error = function(e) {e}
      )
      if (inherits(possibleError, "error"))
        next
    }
  }
}

atl.v.orders <- atl.v.orders[order(-atl.v.orders$AIC), ]
atl.v.ord <- atl.v.orders[nrow(atl.v.orders), ]
atl.v.orders[(nrow(atl.v.orders)-3):nrow(atl.v.orders), ]
```

a) Atlanta Violent Crime

```
##    p d q      AIC
## 64 4 1 6 7447.722
## 63 4 1 5 7446.246
## 77 5 1 5 7445.795
## 92 6 1 6 7444.383
```

```r
# (4, 1, 5)

# ARIMA Fitted Model
atl.v.arima = arima(atl.v.train, order = c(atl.v.ord$p, atl.v.ord$d, atl.v.ord$q), method='ML')
atl.v.arima
```

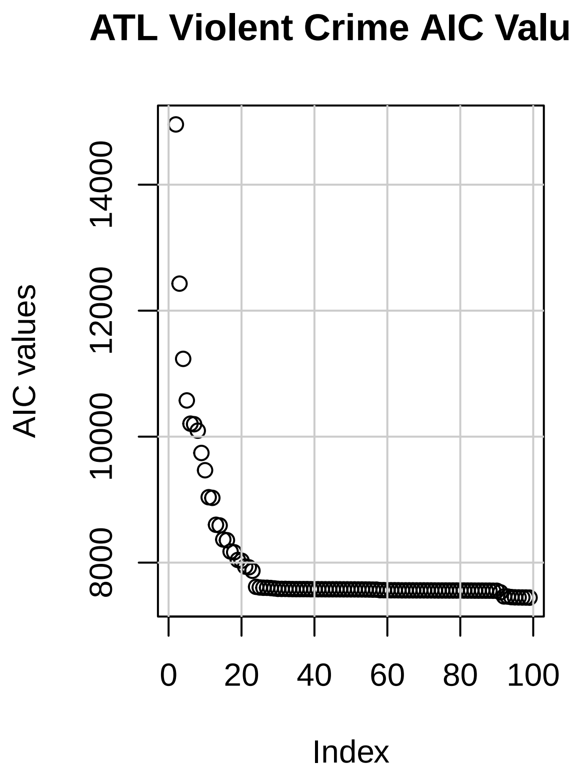
```
##
## Call:
## arima(x = atl.v.train, order = c(atl.v.ord$p, atl.v.ord$d, atl.v.ord$q), method = "ML")
##
```

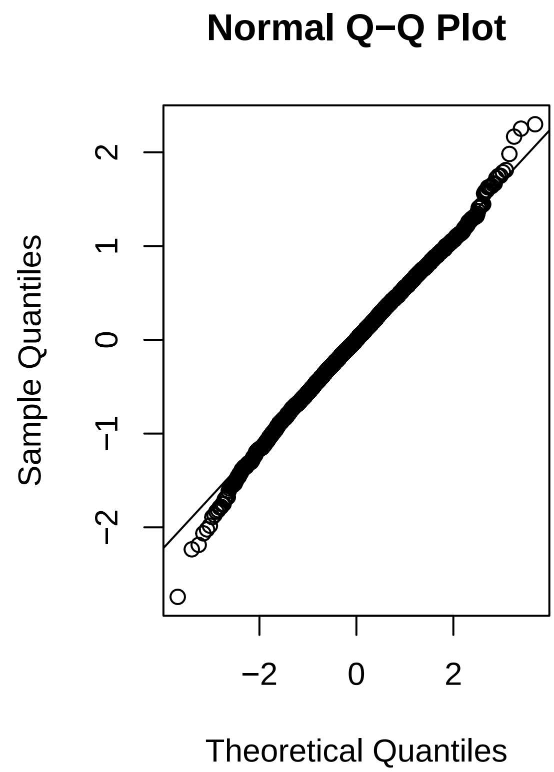
```
## Coefficients:
##           ar1      ar2      ar3      ar4      ar5     ar6     ma1      ma2
##       -1.1225  -0.3773  -0.3921  -1.1573  -0.9477  0.0141  0.1946  -0.6814
## s.e.   0.0030   0.0189   0.0041   0.0063   0.0121  0.0162     NaN      NaN
##          ma3     ma4      ma5      ma6
##       0.0177  0.7870  -0.1302  -0.9152
## s.e.  0.0027  0.0023      NaN      NaN
##
## sigma^2 estimated as 0.3186:  log likelihood = -3709.19,  aic = 7444.38
```

```
coeftest(atl.v.arima)
```

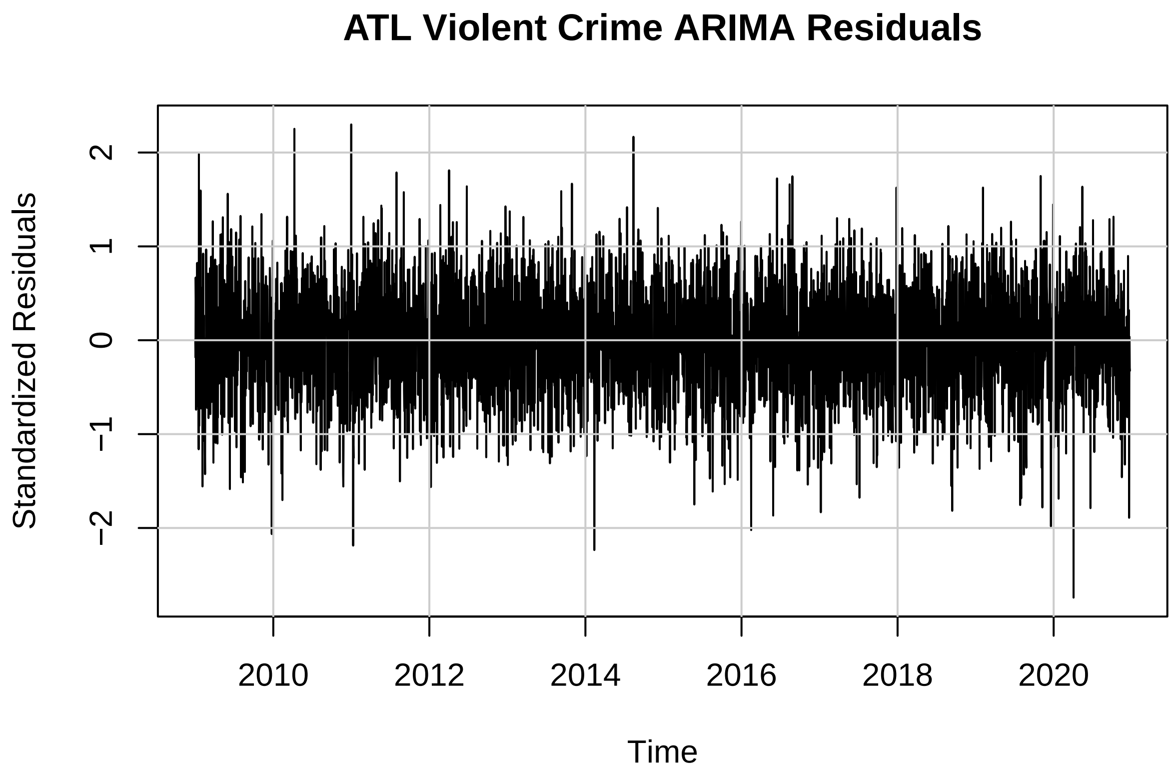
```
##
## z test of coefficients:
##
##       Estimate Std. Error   z value  Pr(>|z|)
## ar1 -1.1225315  0.0030037 -373.7120 < 2.2e-16 ***
## ar2 -0.3772850  0.0189106  -19.9510 < 2.2e-16 ***
## ar3 -0.3921351  0.0040757  -96.2130 < 2.2e-16 ***
## ar4 -1.1573128  0.0063224 -183.0501 < 2.2e-16 ***
## ar5 -0.9477160  0.0120690  -78.5246 < 2.2e-16 ***
## ar6  0.0140929  0.0161628    0.8719    0.3832
## ma1  0.1945653        NaN       NaN       NaN
## ma2 -0.6813645        NaN       NaN       NaN
## ma3  0.0176602  0.0027154    6.5036 7.841e-11 ***
## ma4  0.7869956  0.0022825  344.8023 < 2.2e-16 ***
## ma5 -0.1302488        NaN       NaN       NaN
## ma6 -0.9152075        NaN       NaN       NaN
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
par(mfrow=c(1,2))
plot(atl.v.orders$AIC, ylab="AIC values", main="ATL Violent Crime AIC Values")
grid(lty=1, col=gray(0.8))
qqnorm(resid(atl.v.arima))
qqline(resid(atl.v.arima))
```

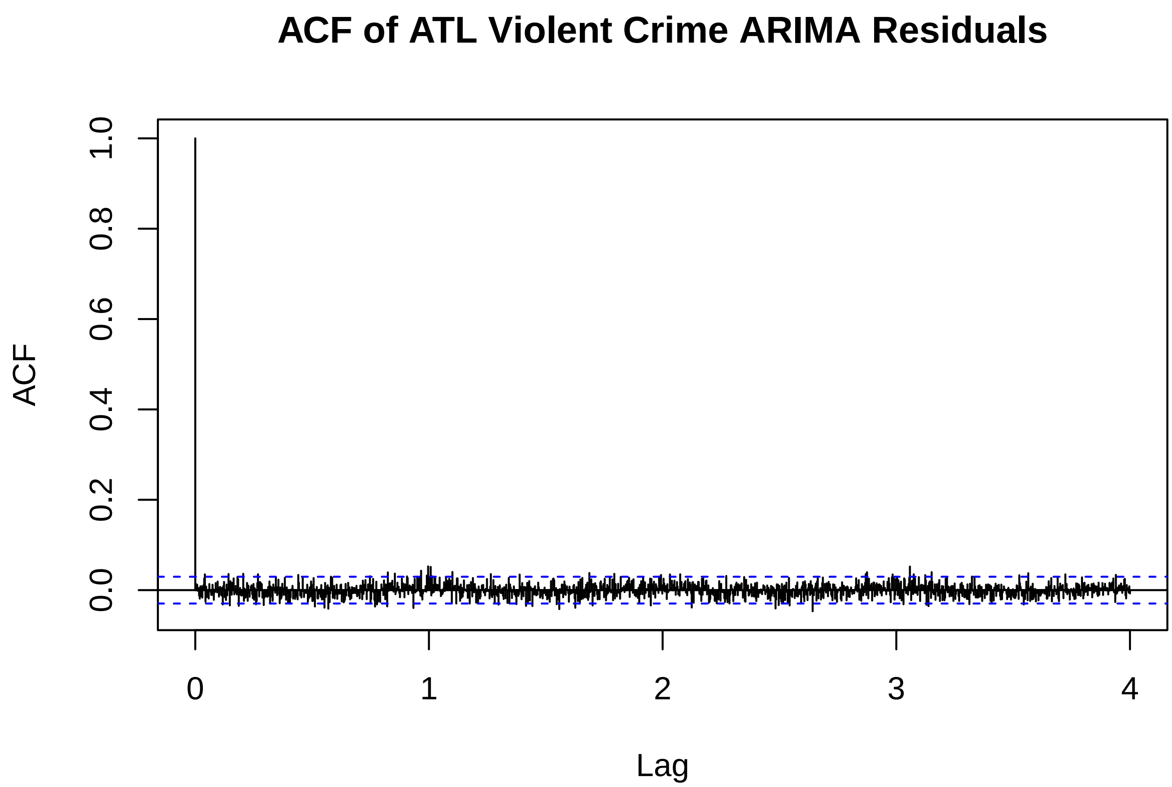





```
par(mfrow=c(1,1))
plot(residuals(atl.v.arima), ylab='Standardized Residuals', main="ATL Violent Crime ARIMA Residuals")
grid(lty=1, col=gray(0.8))
```




```
acf(residuals(atl.v.arima), lag.max = 365.25*4, main="ACF of ATL Violent Crime ARIMA Residuals")
```




```
nyc.v.train <- ts(nyc.train$violentCrime, start = 2009, freq = 365.25)

# Daily TS ARIMA (p, d, q) Fitting
nyc.v.orders = data.frame(Inf, Inf, Inf, Inf)
names(nyc.v.orders) <- c("p", "d", "q", "AIC")

for (p in 0:6) {
  for (d in 1:2) {
    for (q in 0:6) {
      possibleError <- tryCatch(
        nyc.v.orders <- rbind(nyc.v.orders, test_modelA(nyc.v.train,p,d,q)),
        error = function(e) {e}
      )
      if (inherits(possibleError, "error"))
        next
    }
  }
}

nyc.v.orders <- nyc.v.orders[order(-nyc.v.orders$AIC), ]
nyc.v.ord <- nyc.v.orders[nrow(nyc.v.orders), ]
nyc.v.orders[(nrow(nyc.v.orders)-3):nrow(nyc.v.orders), ]
```

b) NYC Violent Crime

```
##    p d q      AIC
## 50 3 1 6 12927.23
## 36 2 1 6 12925.04
## 75 5 1 3 12917.33
## 78 5 1 6 12777.38
```

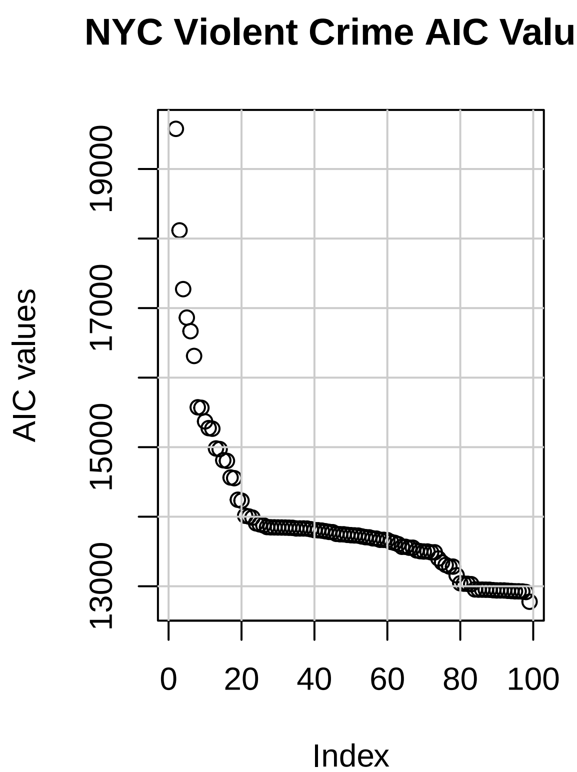
```r
# (5, 1, 6)

# ARIMA Fitted Model
nyc.v.arima = arima(nyc.v.train, order = c(nyc.v.ord$p, nyc.v.ord$d, nyc.v.ord$q), method='ML')
nyc.v.arima
```

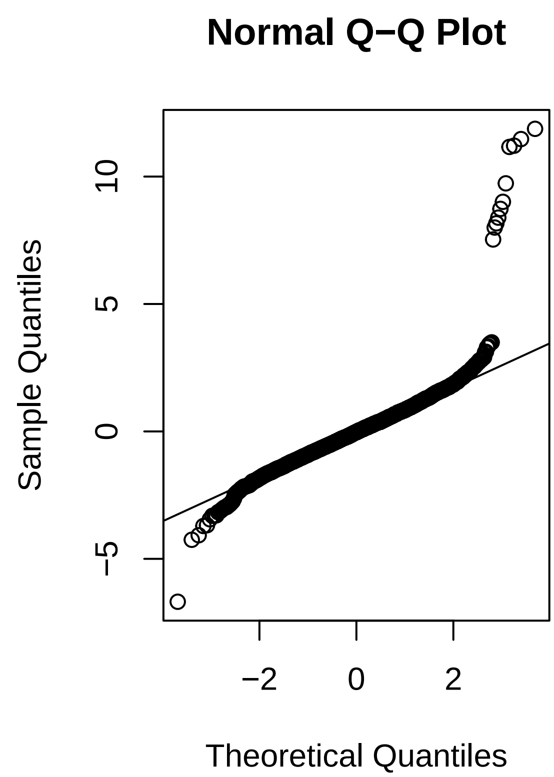
```
##
## Call:
## arima(x = nyc.v.train, order = c(nyc.v.ord$p, nyc.v.ord$d, nyc.v.ord$q), method = "ML")
##
## Coefficients:
##           ar1      ar2      ar3      ar4      ar5      ma1     ma2     ma3
##       -0.1967  -0.6381  -0.6443  -0.1982  -0.9945  -0.6922  0.4495  0.0965
## s.e.   0.0014   0.0028   0.0008   0.0014   0.0030   0.0099  0.0122  0.0119
##           ma4     ma5      ma6
##       -0.3790  0.8030  -0.8615
## s.e.   0.0078  0.0136   0.0128
##
## sigma^2 estimated as 1.078:  log likelihood = -6376.69,  aic = 12777.38
```

```r
coeftest(nyc.v.arima)
```

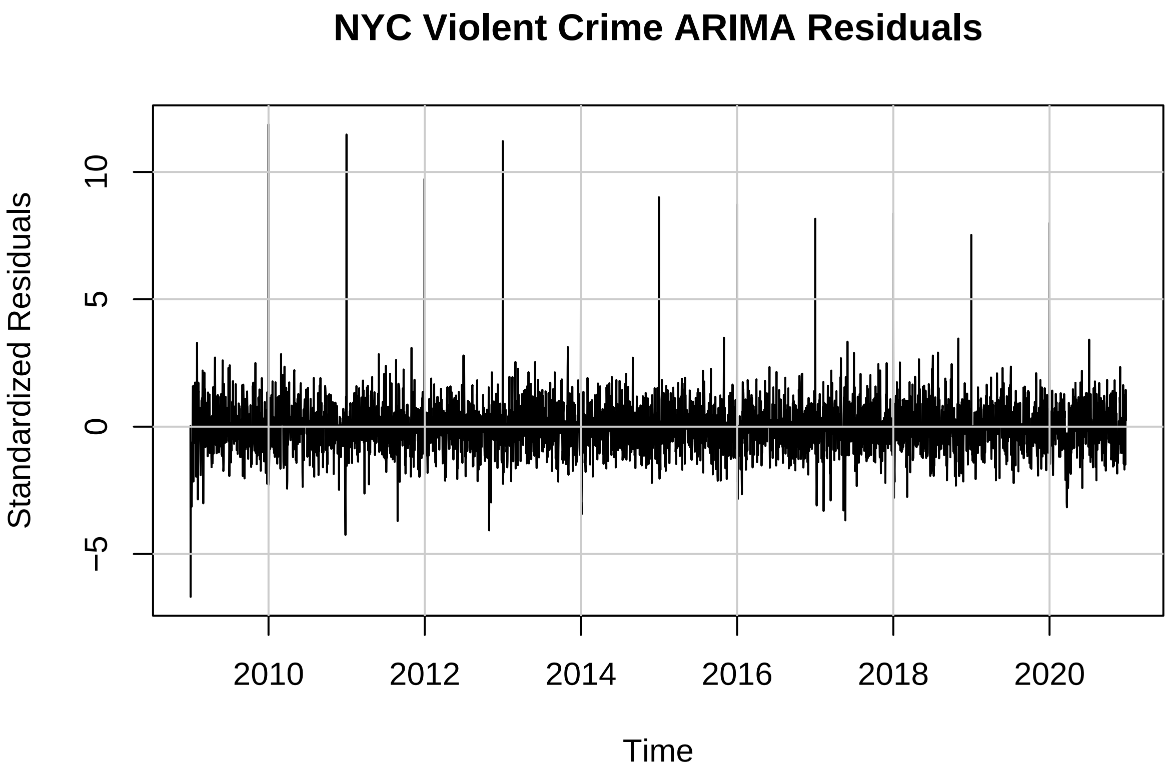
```
##
## z test of coefficients:
##
##        Estimate  Std. Error  z value  Pr(>|z|)
## ar1 -0.19667899  0.00137986 -142.535 < 2.2e-16 ***
## ar2 -0.63811043  0.00278659 -228.993 < 2.2e-16 ***
## ar3 -0.64431184  0.00075258 -856.137 < 2.2e-16 ***
## ar4 -0.19822549  0.00142030 -139.566 < 2.2e-16 ***
## ar5 -0.99451186  0.00300954 -330.453 < 2.2e-16 ***
## ma1 -0.69217334  0.00989389  -69.960 < 2.2e-16 ***
## ma2  0.44948699  0.01224785   36.699 < 2.2e-16 ***
## ma3  0.09654115  0.01189662    8.115 4.858e-16 ***
## ma4 -0.37903828  0.00784307  -48.328 < 2.2e-16 ***
## ma5  0.80303644  0.01361195   58.995 < 2.2e-16 ***
## ma6 -0.86151688  0.01281810  -67.211 < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
par(mfrow=c(1,2))
plot(nyc.v.orders$AIC, ylab="AIC values", main="NYC Violent Crime AIC Values")
grid(lty=1, col=gray(0.8))
qqnorm(resid(nyc.v.arima))
qqline(resid(nyc.v.arima))
```

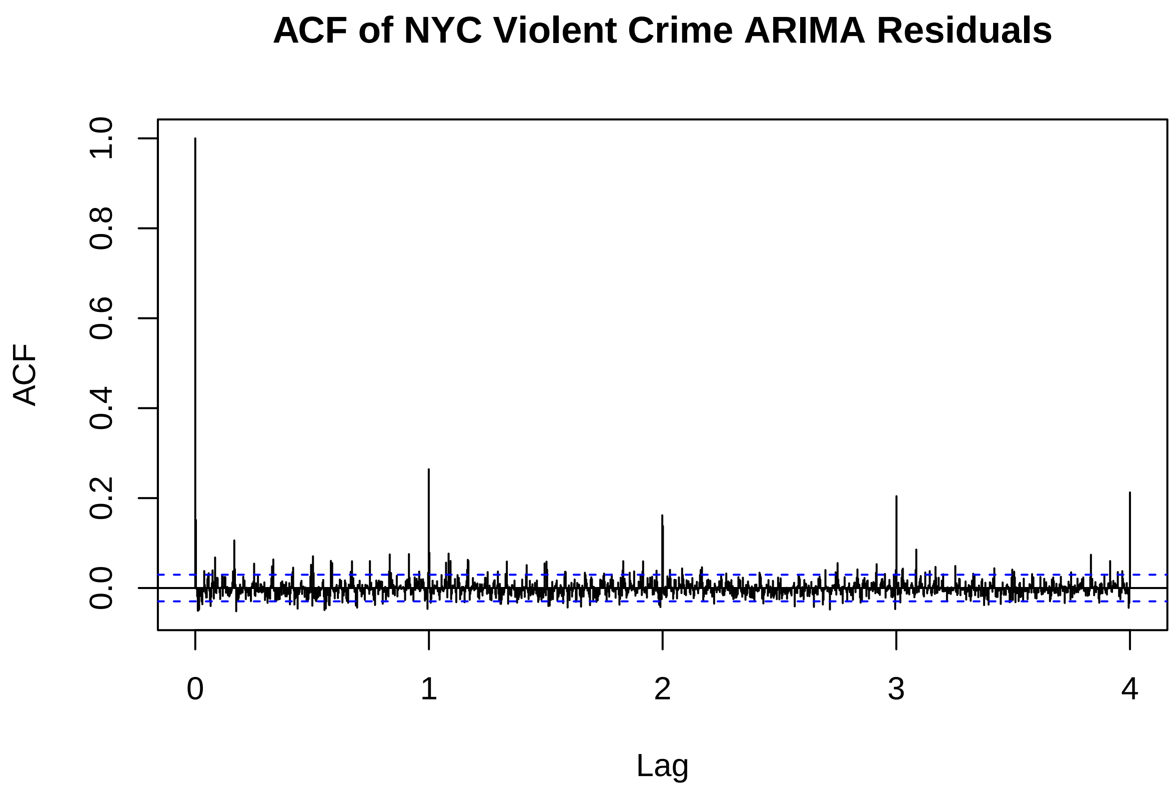





```
par(mfrow=c(1,1))
plot(residuals(nyc.v.arima), ylab='Standardized Residuals', main="NYC Violent Crime ARIMA Residuals")
grid(lty=1, col=gray(0.8))
```




```
acf(residuals(nyc.v.arima), lag.max = 365.25*4, main="ACF of NYC Violent Crime ARIMA Residuals")
```



Residual Analysis

```
## Test for Uncorrelated Residuals for the final model

# ATL
Box.test(atl.v.arima$resid, lag = (atl.v.ord$p+atl.v.ord$q+1),
         type = "Ljung-Box", fitdf = (atl.v.ord$p+atl.v.ord$q))
```

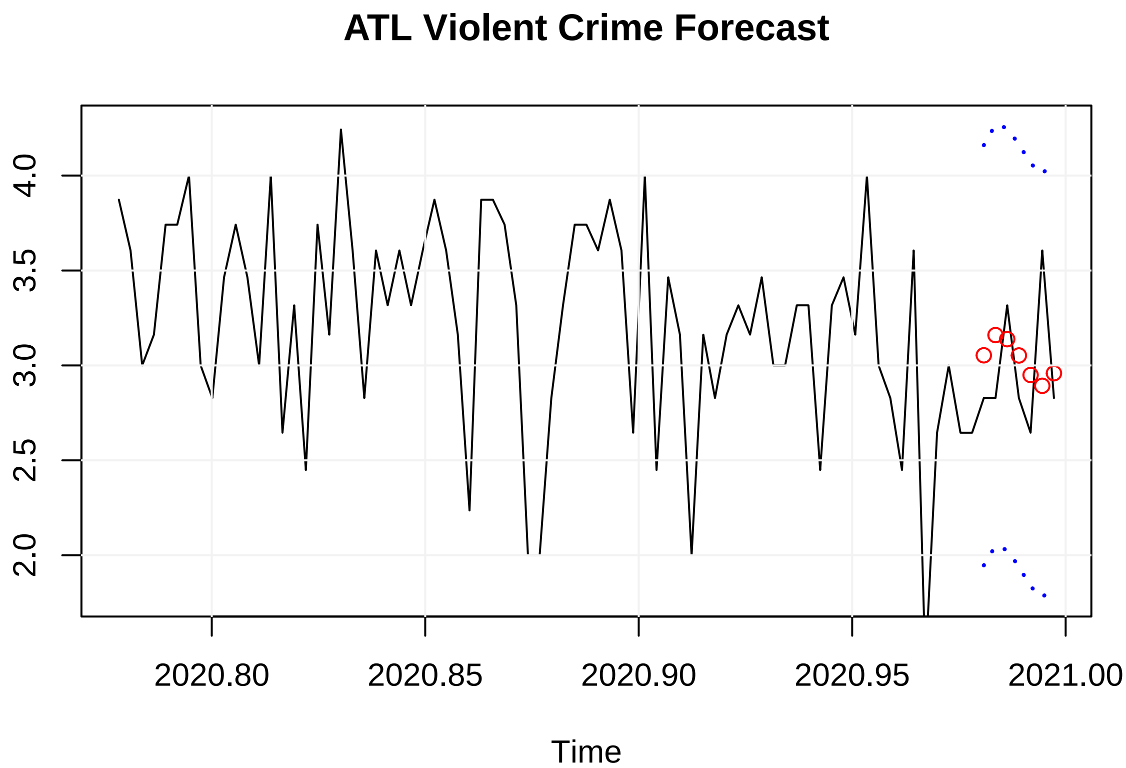
```
##
##  Box-Ljung test
##
## data:  atl.v.arima$resid
## X-squared = 6.9424, df = 1, p-value = 0.008418
```

```
# NYC
Box.test(nyc.v.arima$resid, lag = (nyc.v.ord$p+nyc.v.ord$q+1),
         type = "Ljung-Box", fitdf = (nyc.v.ord$p+nyc.v.ord$q))
```

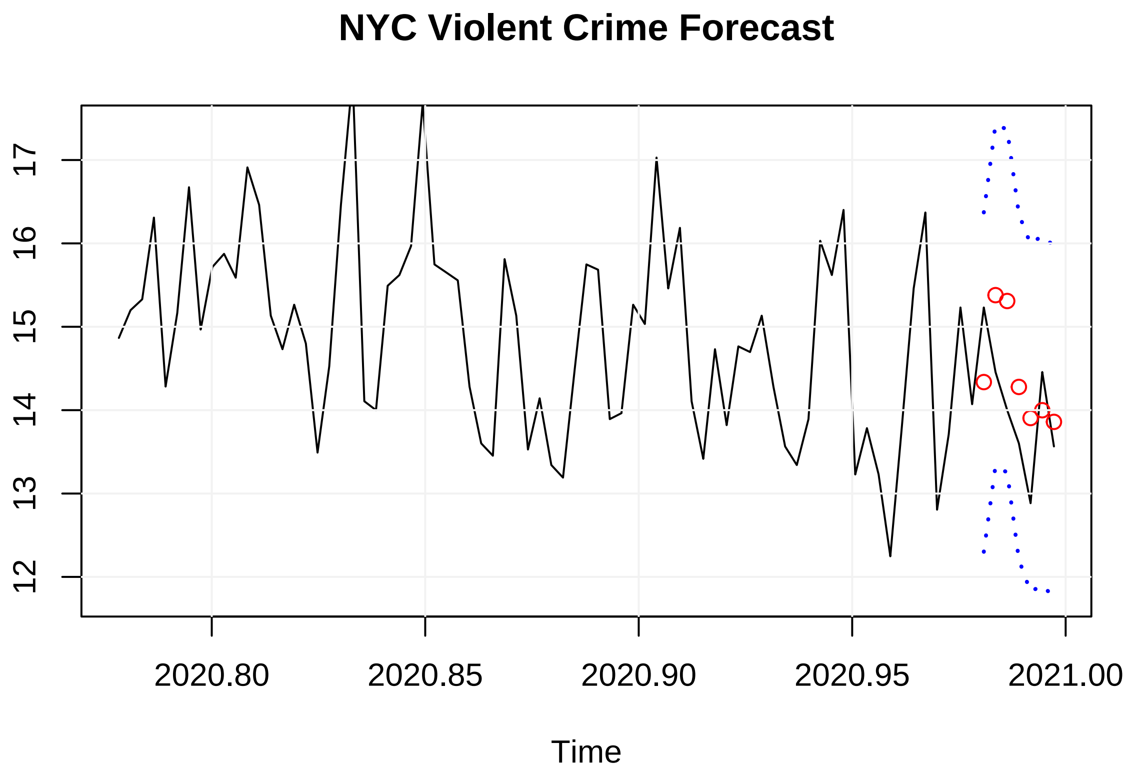
```
##
##  Box-Ljung test
##
## data:  nyc.v.arima$resid
## X-squared = 145.74, df = 1, p-value < 2.2e-16
```

Forecast

```
plot_forecast <- function(ts, out_pred, days_ahead, plot_title, conf) {
  n = length(ts)
  nfit = n-days_ahead

  timevol=time(ts)
  ubound = out_pred$pred+conf*out_pred$se
  lbound = out_pred$pred-conf*out_pred$se
  ymin = min(lbound, min(out_pred$pred))
  ymax = max(ubound, max(out_pred$pred))

  par(mfrow=c(1,1))
  plot(timevol[(n-80):n],ts[(n-80):n],type="l", ylim=c(ymin, ymax), xlab="Time",
       ylab="", main=plot_title)
  points(timevol[(nfit+1):n],out_pred$pred,col="red")
  lines(timevol[(nfit+1):n],ubound,lty=3,lwd= 2, col="blue")
  lines(timevol[(nfit+1):n],lbound,lty=3,lwd= 2, col="blue")
}

n.ahead <- nrow(atl.test)

## Forecast ATL Violent Crime
atl.v.pred <- as.vector(predict(atl.v.arima, n.ahead=n.ahead))
plot_forecast(atl.v.ts, atl.v.pred, n.ahead, conf=1.96, plot_title = "ATL Violent Crime Forecast")
grid(lty=1, col=gray(0.95))
```




```r
## Forecast NYC Violent Crime
nyc.v.pred <- as.vector(predict(nyc.v.arima, n.ahead=n.ahead))
plot_forecast(nyc.v.ts, nyc.v.pred, n.ahead, conf=1.96, plot_title = "NYC Violent Crime Forecast")
grid(lty=1, col=gray(0.95))
```



Prediction Evaluation

```
mape <- function(y, y_pred) {
  mape <- mean(abs((y-y_pred)/y))
  return(mape)
}

pm <- function(obs, pred) {
  pm <- sum((pred-obs)^2)/sum((obs-mean(obs))^2)
  return(pm)
}

atl.v.mape <- mape(atl.test$violentCrime, atl.v.pred$pred)
atl.v.pm <- pm(atl.test$violentCrime, atl.v.pred$pred)

nyc.v.mape <- mape(nyc.test$violentCrime, nyc.v.pred$pred)
nyc.v.pm <- pm(nyc.test$violentCrime, nyc.v.pred$pred)

cat("ATL Violent:\nMAPE =", atl.v.mape, "\nPM =", atl.v.pm,
    "\n\nNYC Violent:\nMAPE =", nyc.v.mape, "\nPM =", nyc.v.pm)
```

```
## ATL Violent:
## MAPE = 0.0983468
## PM = 1.214297
```

```
##
## NYC Violent:
## MAPE = 0.05693062
## PM = 1.465883
```

SARIMA Fitting and Forecasting

```
atl.v.sarima = arima(atl.v.train, order = c(5,1,6), seasonal = list(order =c(2,0,2), period=7), method=
atl.v.sarima
```

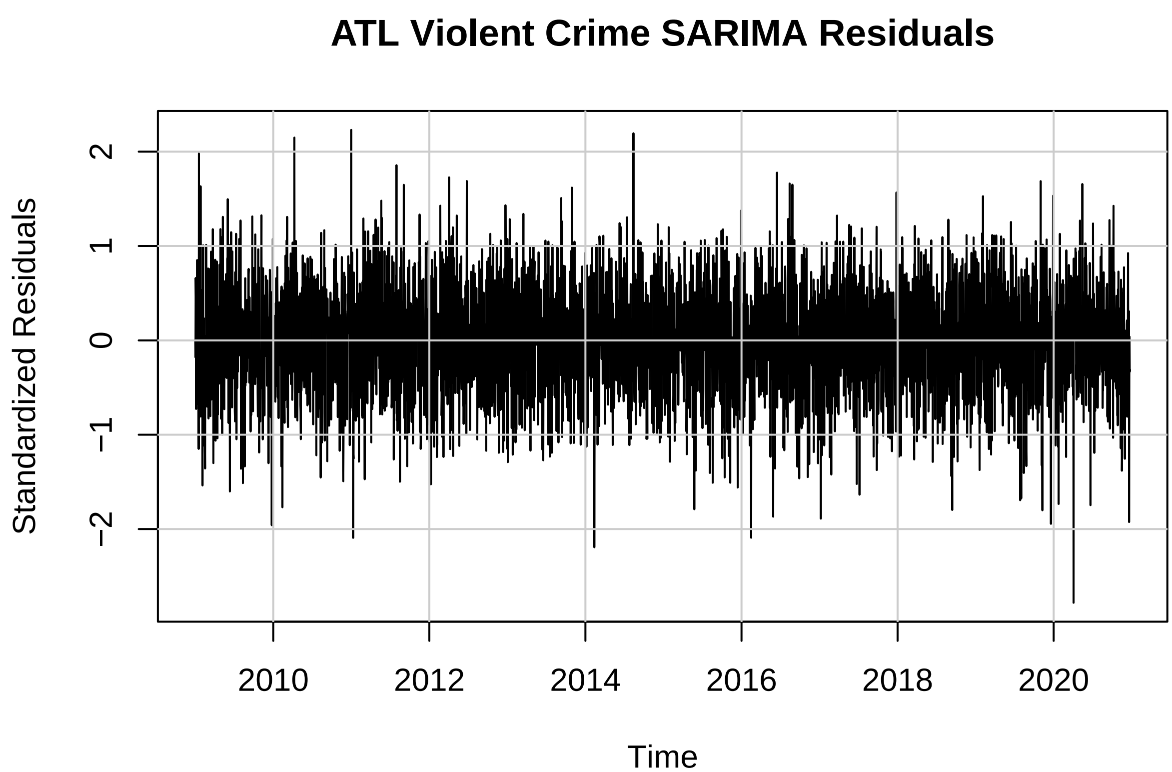
a) ATL Violent Crime

```
##
## Call:
## arima(x = atl.v.train, order = c(5, 1, 6), seasonal = list(order = c(2, 0, 2),
##     period = 7), method = "ML")
##
## Coefficients:
##           ar1      ar2     ar3     ar4     ar5      ma1     ma2      ma3
##       -0.3117  -0.2662  0.0083  0.0132  0.2237  -0.6178  -0.021  -0.2676
## s.e.   0.2749      NaN     NaN  0.4509  0.2588   0.2790     NaN   0.4221
##           ma4      ma5     ma6    sar1    sar2     sma1     sma2
##       -0.0173  -0.1916  0.1848  0.0046  0.9949  -0.0032  -0.9874
## s.e.   0.2353      NaN  0.2501  0.0049  0.0049   0.0062   0.0062
##
## sigma^2 estimated as 0.3164:  log likelihood = -3697.65,  aic = 7427.31
```

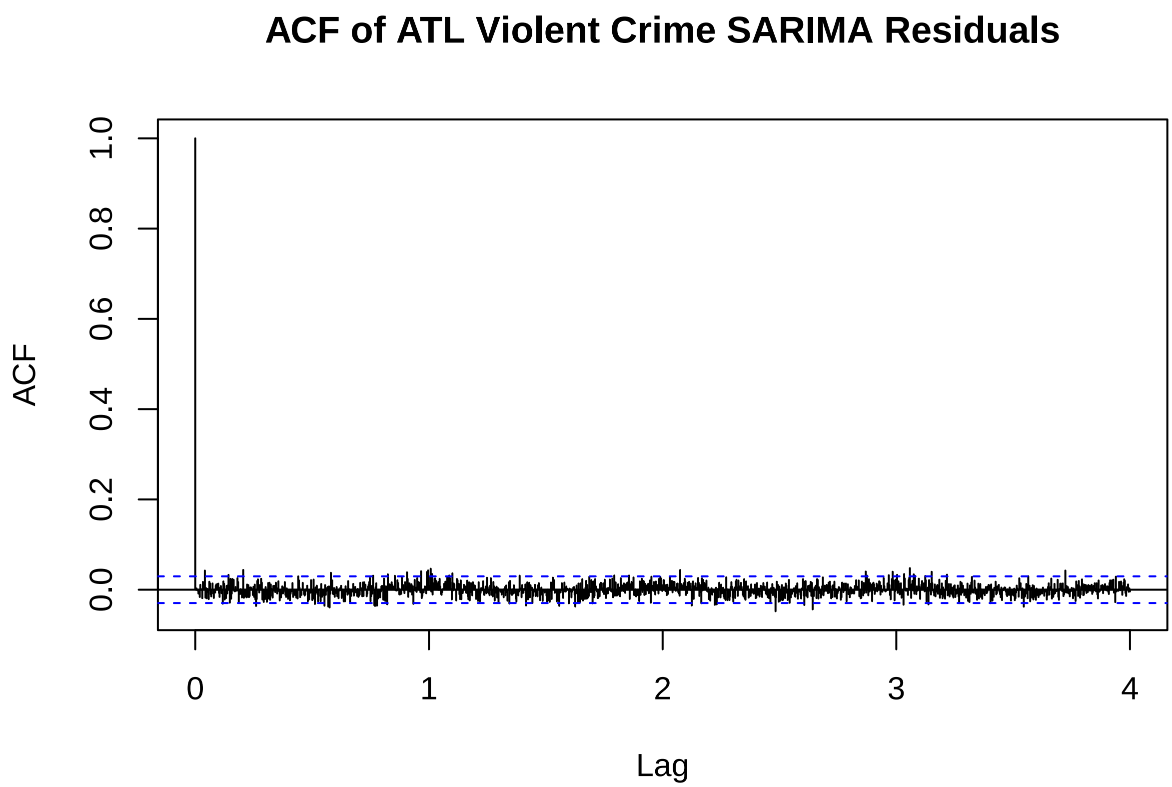
```
coeftest(atl.v.sarima)
```

```
##
## z test of coefficients:
##
##        Estimate Std. Error   z value Pr(>|z|)
## ar1  -0.3116551  0.2749122   -1.1337   0.2569
## ar2  -0.2661876        NaN       NaN      NaN
## ar3   0.0083429        NaN       NaN      NaN
## ar4   0.0131595  0.4509087    0.0292   0.9767
## ar5   0.2237308  0.2588139    0.8644   0.3873
## ma1  -0.6177669  0.2789681   -2.2145   0.0268 *
## ma2  -0.0210306        NaN       NaN      NaN
## ma3  -0.2675961  0.4221474   -0.6339   0.5262
## ma4  -0.0172830  0.2352841   -0.0735   0.9414
## ma5  -0.1915679        NaN       NaN      NaN
## ma6   0.1847715  0.2500624    0.7389   0.4600
## sar1  0.0045851  0.0048716    0.9412   0.3466
## sar2  0.9949347  0.0048703  204.2880   <2e-16 ***
## sma1 -0.0032469  0.0061744   -0.5259   0.5990
## sma2 -0.9874367  0.0061644 -160.1839   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
par(mfrow=c(1,1))
plot(residuals(atl.v.sarima), ylab='Standardized Residuals', main="ATL Violent Crime SARIMA Residuals")
grid(lty=1, col=gray(0.8))
```




```
acf(residuals(atl.v.sarima), lag.max = 365.25*4, main="ACF of ATL Violent Crime SARIMA Residuals")
```




```
nyc.v.sarima = arima(nyc.v.train, order = c(5,1,6),seasonal = list(order =c(2,0,2), period=7), method='
nyc.v.sarima
```

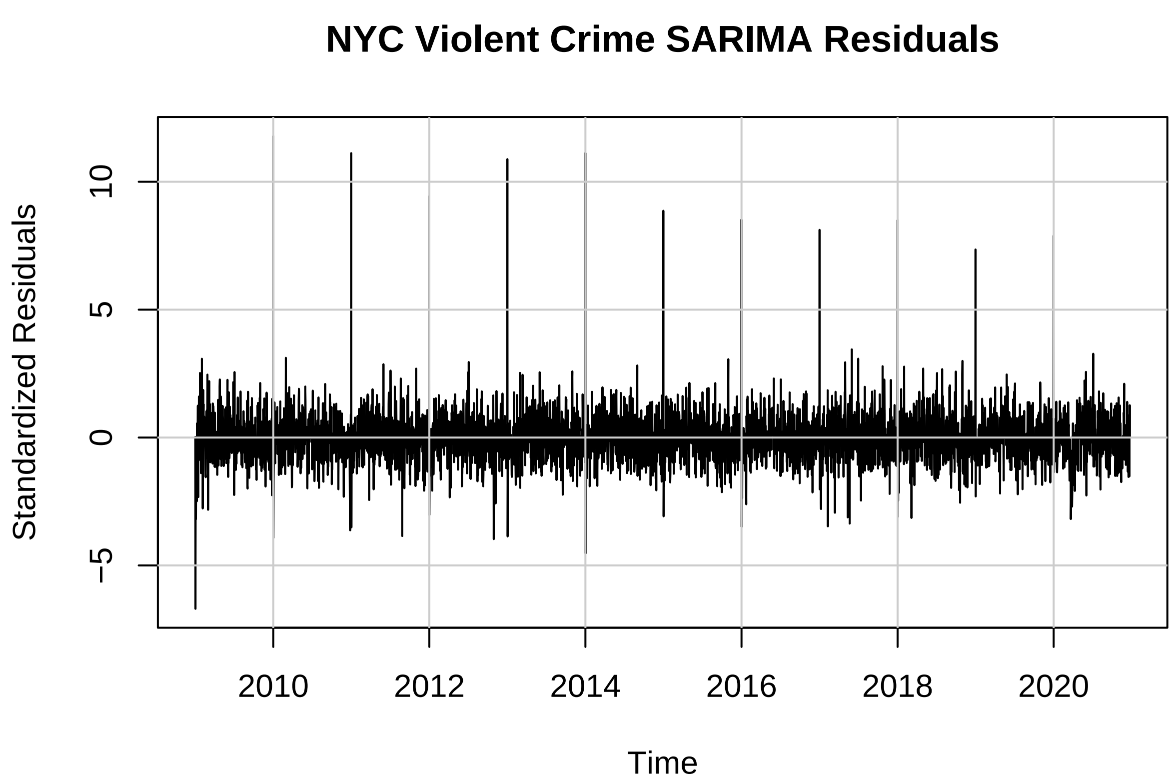
b) NYC Violent Crime

```
##
## Call:
## arima(x = nyc.v.train, order = c(5, 1, 6), seasonal = list(order = c(2, 0, 2),
##     period = 7), method = "ML")
##
## Coefficients:
##           ar1      ar2      ar3      ar4      ar5      ma1      ma2     ma3
##       -0.6730  -0.4059  -0.6252  -0.8598  -0.1270  -0.0660  -0.2456  0.2253
## s.e.   0.1556   0.1121   0.1065   0.1063   0.0914   0.1543   0.1713  0.1620
##          ma4      ma5      ma6     sar1     sar2     sma1     sma2
##       0.3166  -0.6417  -0.2483  0.0110  0.9888  -0.0016  -0.9849
## s.e.  0.1202   0.0911   0.0799  0.0074  0.0077   0.0013   0.0101
##
## sigma^2 estimated as 1.03:  log likelihood = -6282.99,  aic = 12597.98
```

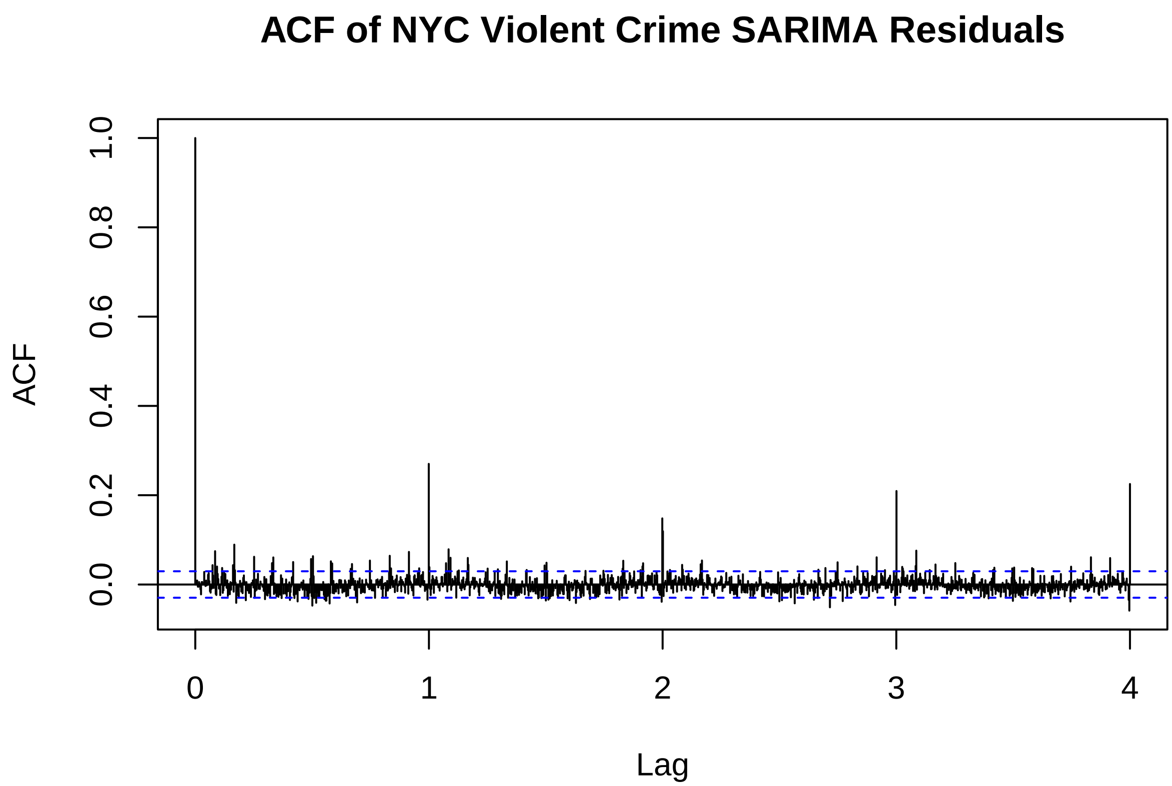
```
coeftest(nyc.v.sarima)
```

```
##
## z test of coefficients:
##
##        Estimate Std. Error  z value  Pr(>|z|)
## ar1  -0.6729602  0.1555966  -4.3250 1.525e-05 ***
## ar2  -0.4059085  0.1120734  -3.6218 0.0002925 ***
## ar3  -0.6251983  0.1064586  -5.8727 4.288e-09 ***
## ar4  -0.8598435  0.1063188  -8.0874 6.095e-16 ***
## ar5  -0.1270087  0.0913581  -1.3902 0.1644592
## ma1  -0.0659577  0.1543031  -0.4275 0.6690478
## ma2  -0.2456195  0.1712505  -1.4343 0.1514951
## ma3   0.2252845  0.1620041   1.3906 0.1643438
## ma4   0.3166466  0.1201666   2.6351 0.0084122 **
## ma5  -0.6417228  0.0910708  -7.0464 1.836e-12 ***
## ma6  -0.2483220  0.0798663  -3.1092 0.0018758 **
## sar1  0.0109827  0.0074432   1.4755 0.1400684
## sar2  0.9888487  0.0077322 127.8868 < 2.2e-16 ***
## sma1 -0.0015507  0.0013223  -1.1728 0.2408931
## sma2 -0.9849050  0.0100829 -97.6810 < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
par(mfrow=c(1,1))
plot(residuals(nyc.v.sarima), ylab='Standardized Residuals', main="NYC Violent Crime SARIMA Residuals")
grid(lty=1, col=gray(0.8))
```

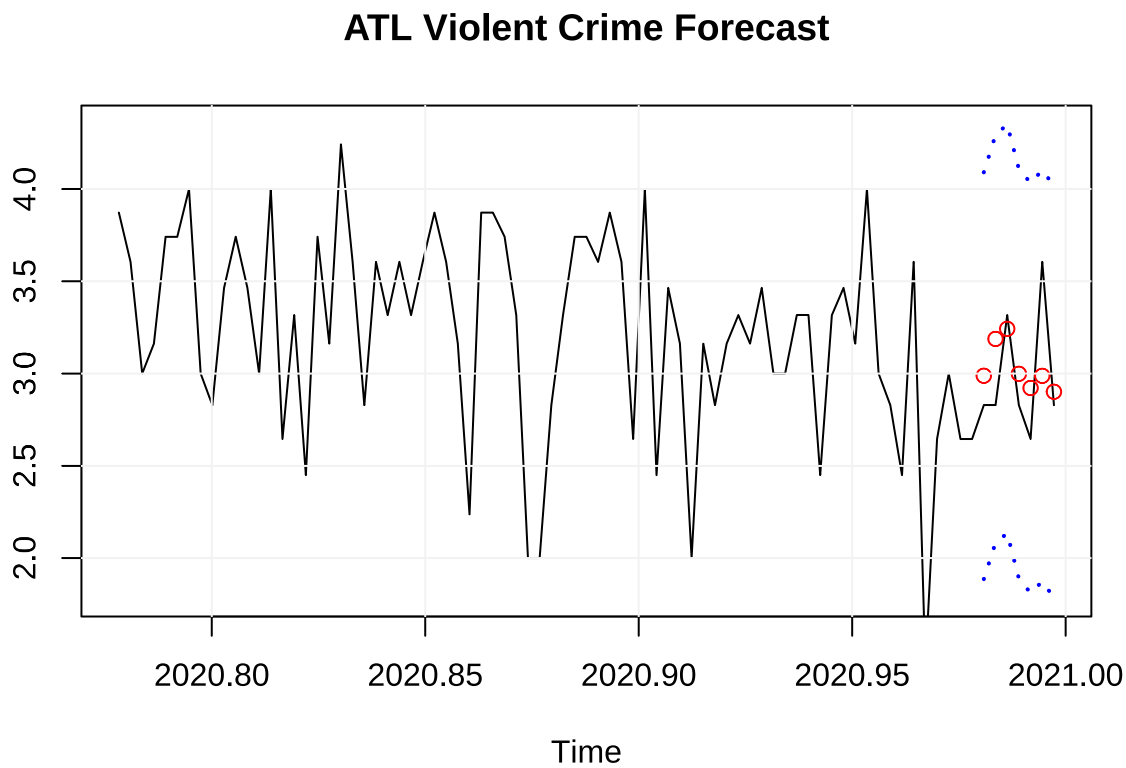


```
acf(residuals(nyc.v.sarima), lag.max = 365.25*4, main="ACF of NYC Violent Crime SARIMA Residuals")
```

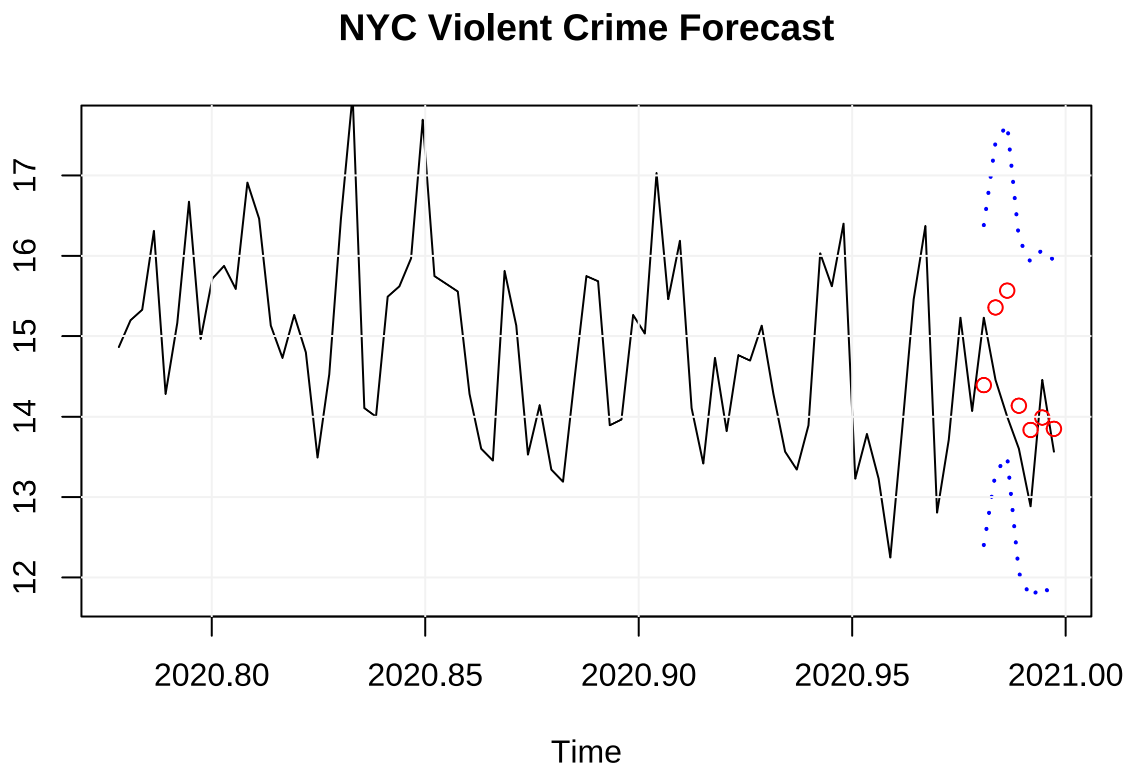


Forecasting Analysis

```
## Forecast ATL Violent Crime
atl.v.spred <- as.vector(predict(atl.v.sarima, n.ahead=n.ahead))
plot_forecast(atl.v.ts, atl.v.spred, n.ahead, conf=1.96, plot_title = "ATL Violent Crime Forecast")
grid(lty=1, col=gray(0.95))
```




```r
## Forecast NYC Violent Crime
nyc.v.spred <- as.vector(predict(nyc.v.sarima, n.ahead=n.ahead))
plot_forecast(nyc.v.ts, nyc.v.spred, n.ahead, conf=1.96, plot_title = "NYC Violent Crime Forecast")
grid(lty=1, col=gray(0.95))
```



Evaluation

```
atl.v.smape <- mape(atl.test$violentCrime, atl.v.spred$pred)
atl.v.spm <- pm(atl.test$violentCrime, atl.v.spred$pred)

nyc.v.smape <- mape(nyc.test$violentCrime, nyc.v.spred$pred)
nyc.v.spm <- pm(nyc.test$violentCrime, nyc.v.spred$pred)

cat("ATL Violent:\nMAPE =", atl.v.smape, "\nPM =", atl.v.spm,
    "\n\nNYC Violent:\nMAPE =", nyc.v.smape, "\nPM =", nyc.v.spm)
```

```
## ATL Violent:
## MAPE = 0.08107274
## PM = 0.9211366
##
## NYC Violent:
## MAPE = 0.05655839
## PM = 1.551702
```