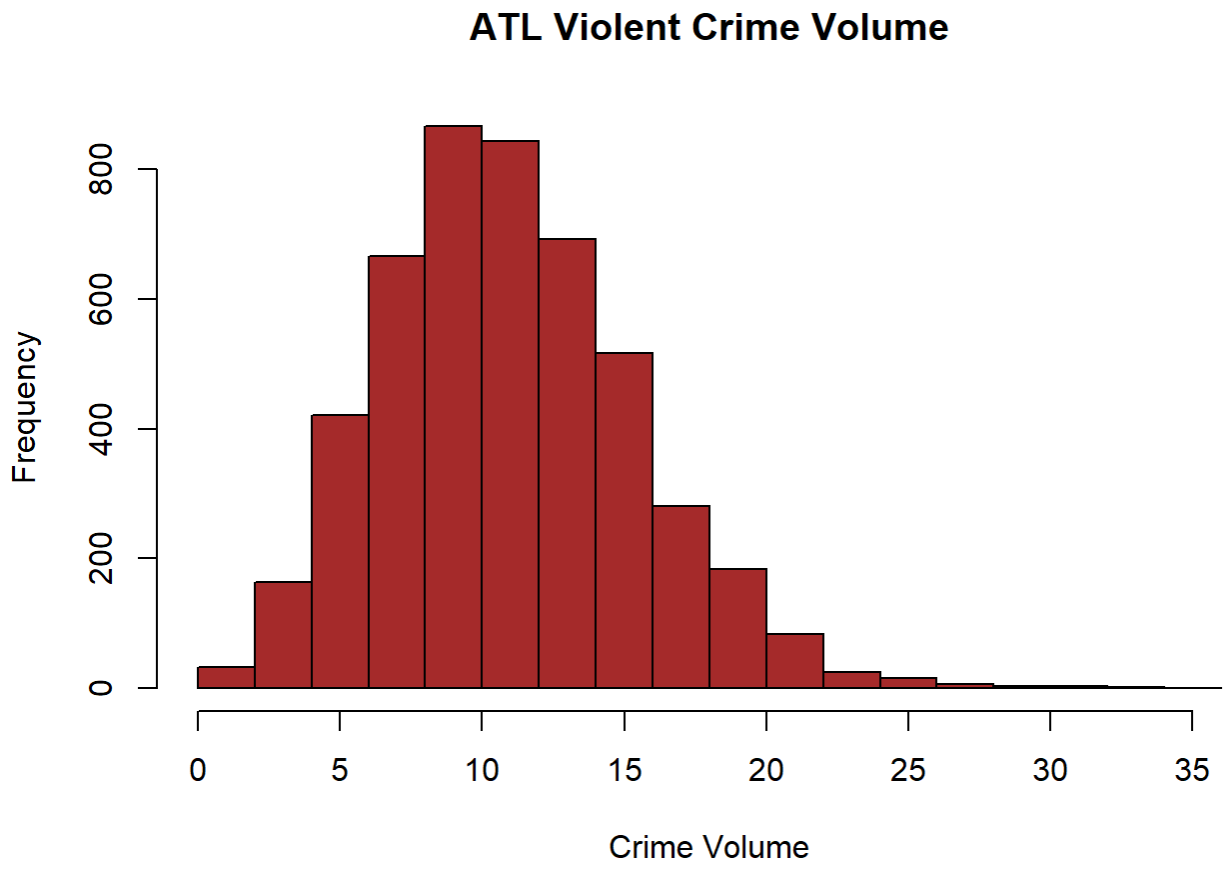
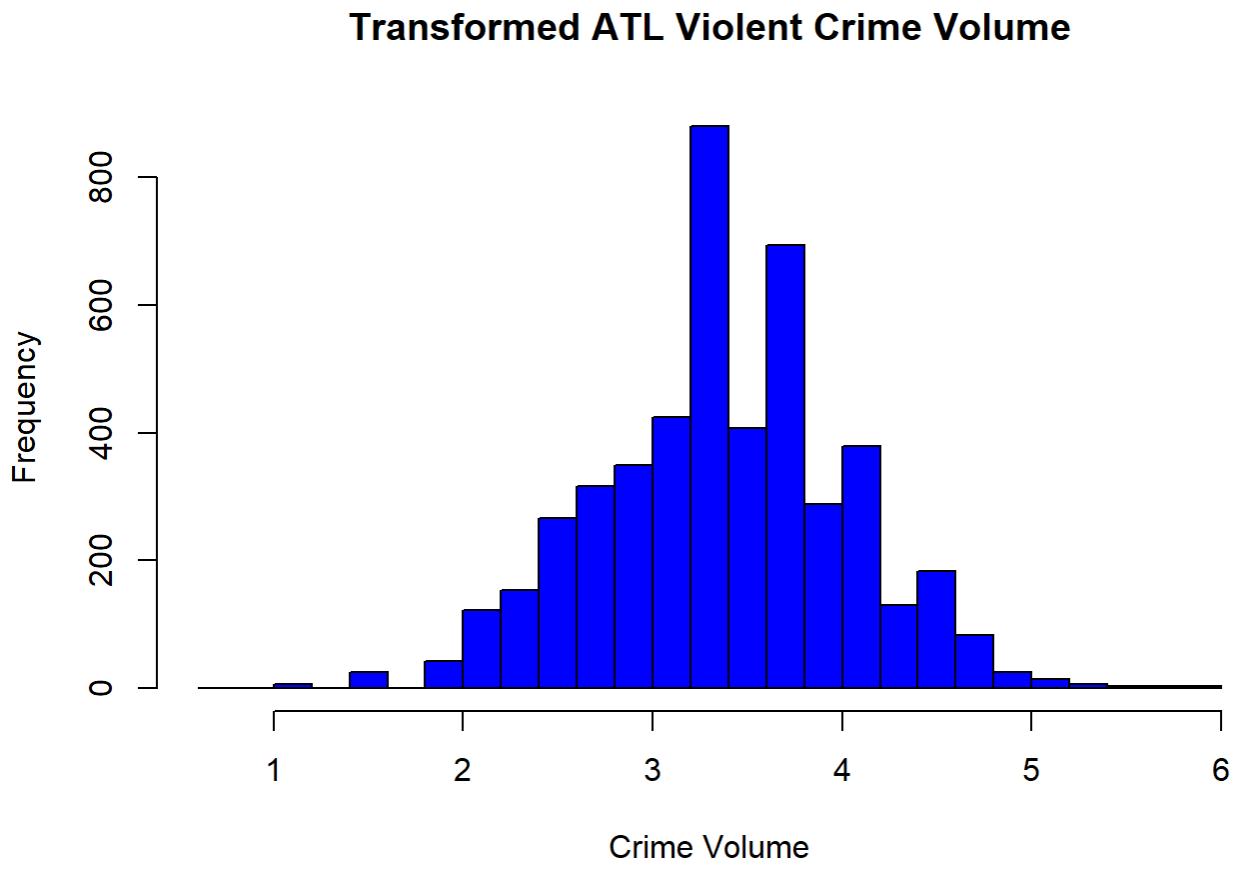
```
###Import File

atl_violent <- read.csv("atl_violent_final.csv", head = TRUE)
names(atl_violent)[1] <- 'Date'
atl_v <- ts(atl_violent$occurance_count, start = 2009, freq = 365)

atl_prop <- read.csv("atl_prop_final.csv", head = TRUE)
names(atl_prop)[1] <- 'Date'
atl_p <- ts(atl_prop$occurance_count, start = 2009, freq = 365)

nyc_violent <- read.csv("nyc_violent_final_avg.csv", head = TRUE)
names(nyc_violent)[1] <- 'Date'
nyc_v <- ts(nyc_violent$occurance_count, start = 2009, freq = 365)

nyc_prop <- read.csv("nyc_prop_final_v2.csv", head = TRUE)
names(nyc_prop)[1] <- 'Date'
nyc_p <- ts(nyc_prop$occurance_count, start = 2009, freq = 365)

nyc_qol <- read.csv("nyc_QOL_final_v2.csv", head = TRUE)
names(nyc_qol)[1] <- 'Date'
nyc_q <- ts(nyc_qol$occurance_count, start = 2009, freq = 365)
```

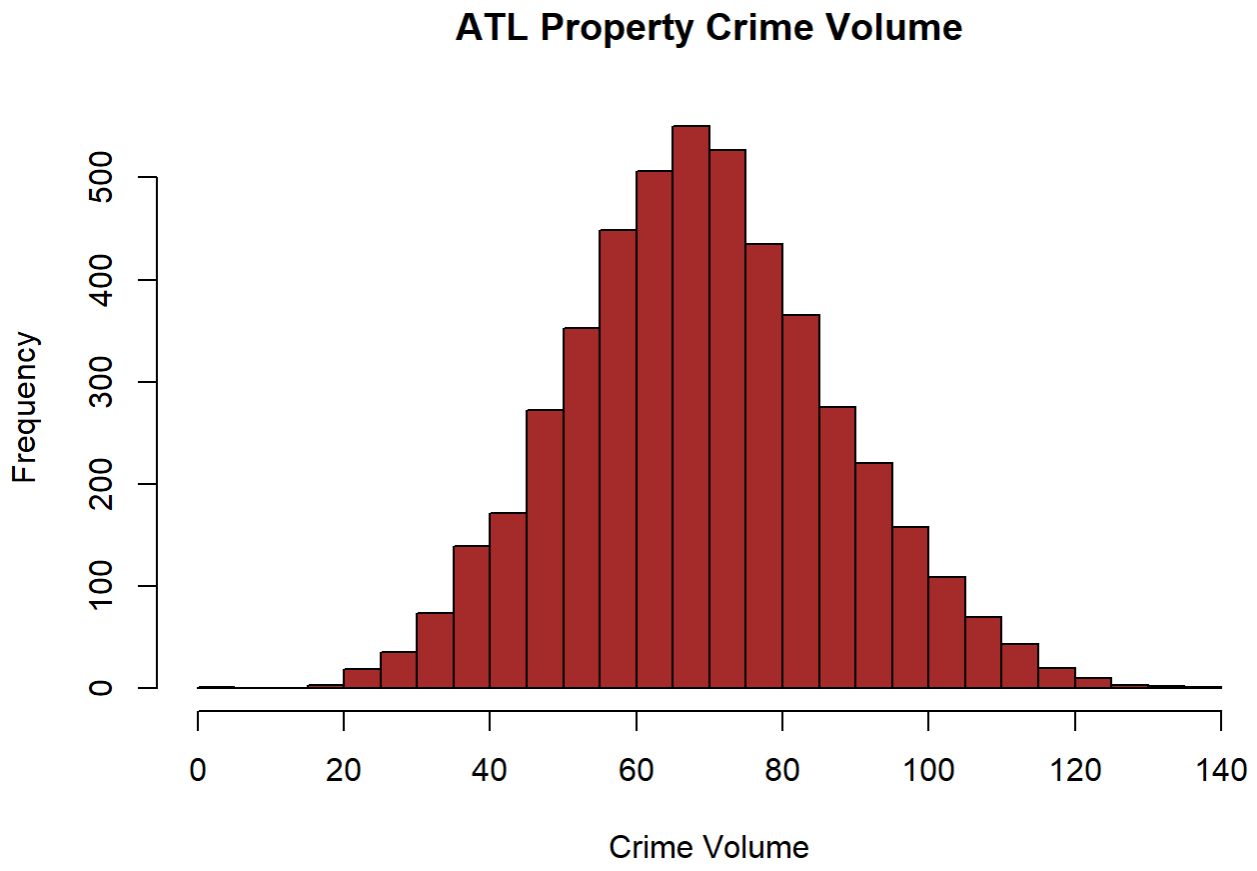
```
###HISTOGRAMS - DISTRIBUTION COMPARISON
atl_violent.tr = sqrt(atl_violent$occurance_count+3/8)
atl_v.tr <- ts(atl_violent.tr, start = 2009, freq = 365)

atl_prop.tr = sqrt(atl_prop$occurance_count+3/8)
atl_p.tr <- ts(atl_prop.tr, start = 2009, freq = 365)

nyc_violent.tr = sqrt(nyc_violent$occurance_count+3/8)
nyc_v.tr <- ts(nyc_violent.tr, start = 2009, freq = 365)

nyc_prop.tr = sqrt(nyc_prop$occurance_count+3/8)
nyc_p.tr <- ts(nyc_prop.tr, start = 2009, freq = 365)

nyc_qol.tr = sqrt(nyc_qol$occurance_count+3/8)
nyc_q.tr <- ts(nyc_qol.tr, start = 2009, freq = 365)

hist(atl_violent$occurance_count,nclass=20,xlab="Crime Volume", main="ATL Violent Crime Volume",
col="brown")
```



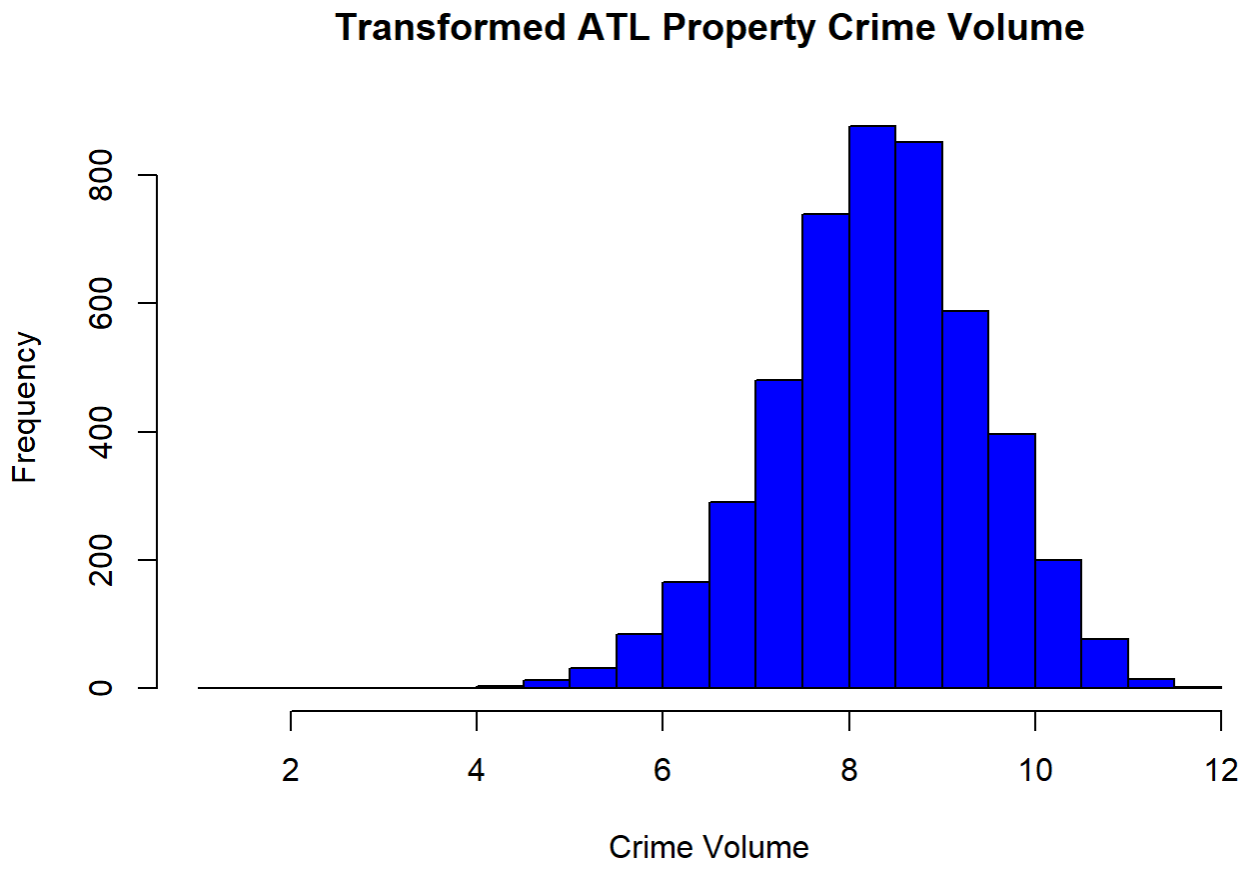

```
hist(atl_violent.tr,nclass=20,xlab="Crime Volume", main="Transformed ATL Violent Crime Volume",
 col="blue")
```



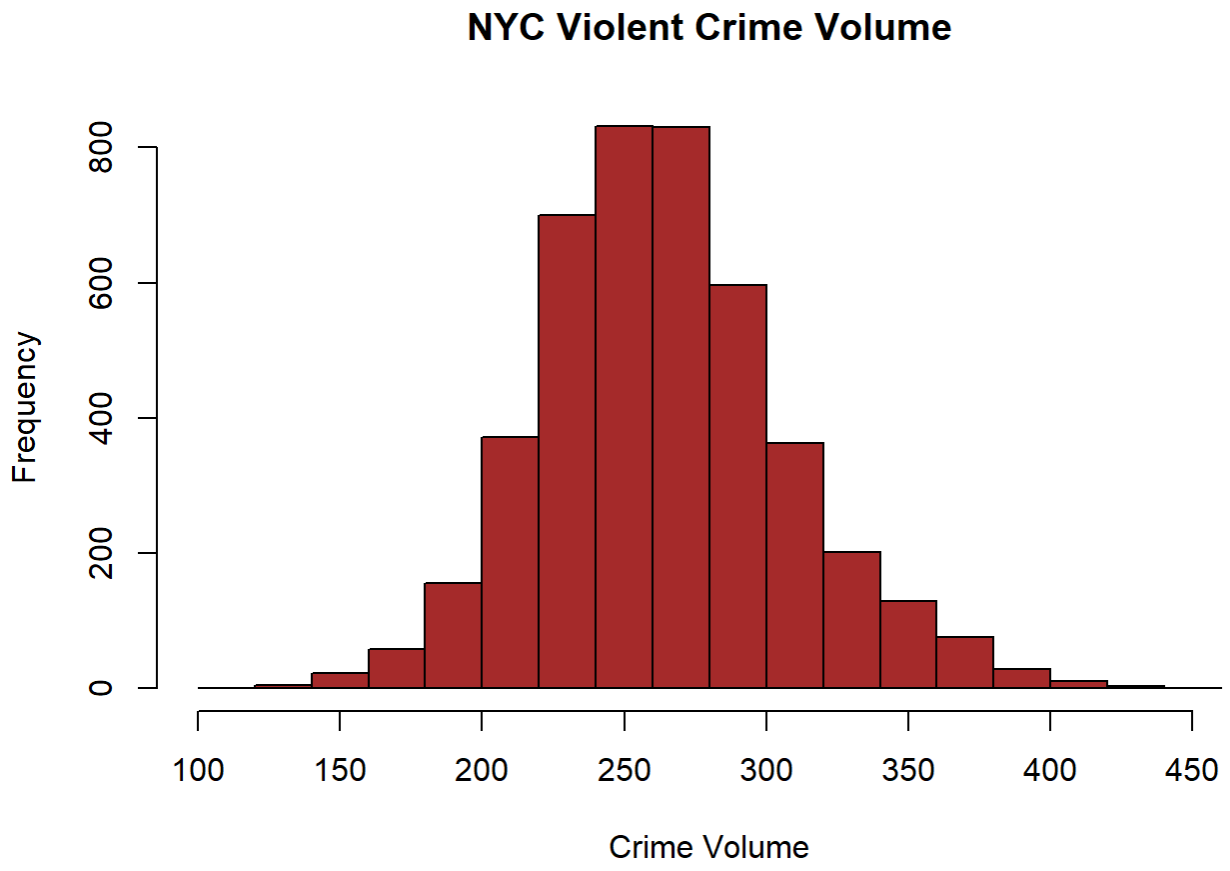

```
hist(atl_prop$occurance_count,nclass=20,xlab="Crime Volume", main="ATL Property Crime Volume",co
l="brown")
```



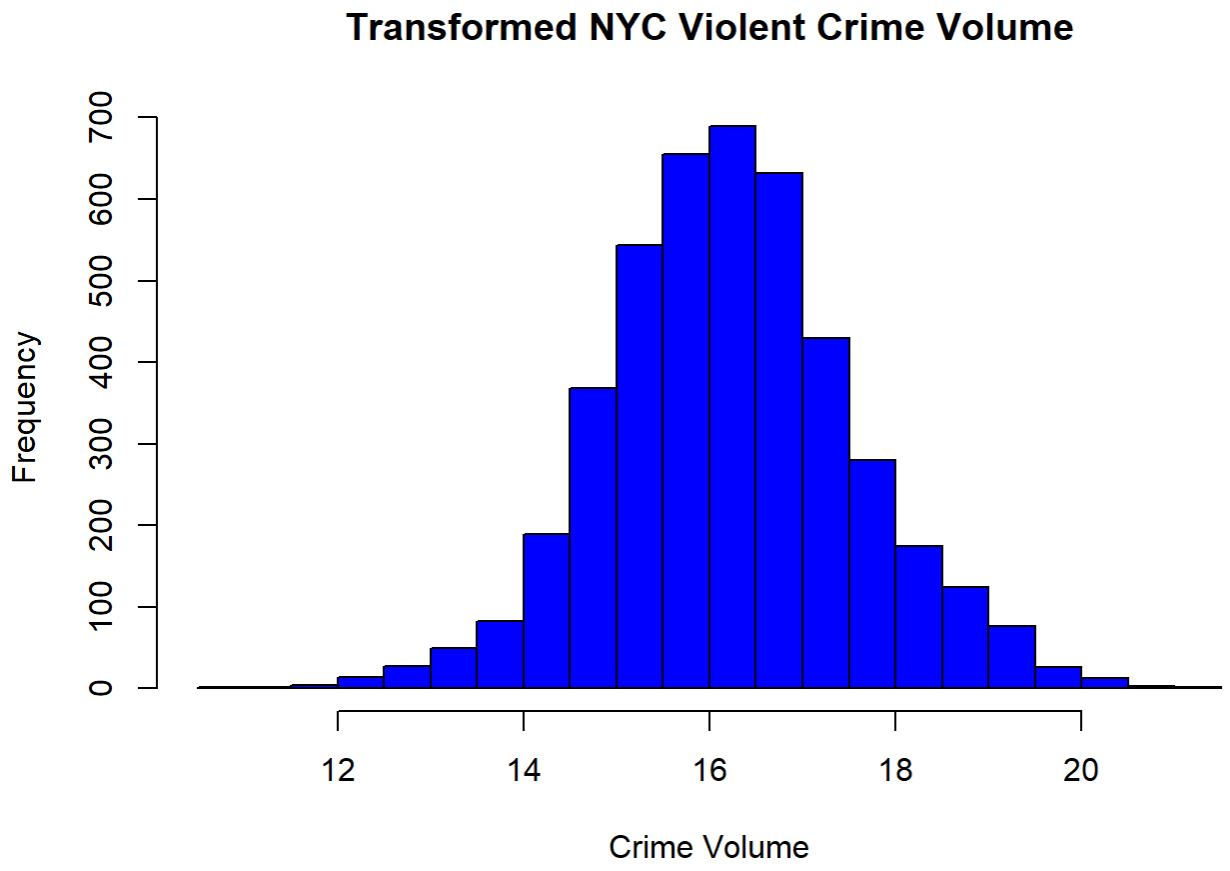

```
hist(atl_prop.tr,nclass=20,xlab="Crime Volume", main="Transformed ATL Property Crime Volume", co
l="blue")
```



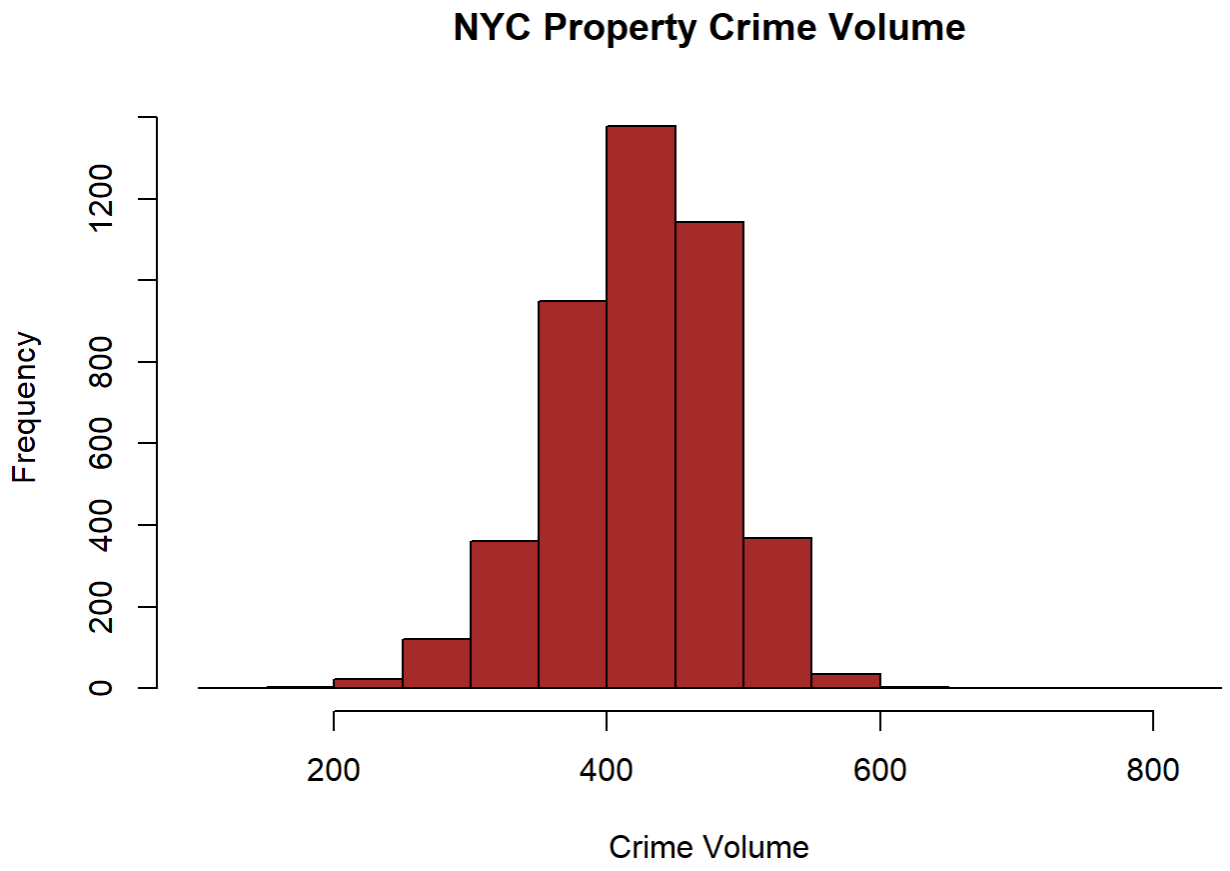

```
hist(nyc_violent$occurance_count,nclass=20,xlab="Crime Volume", main="NYC Violent Crime Volume",
col="brown")
```



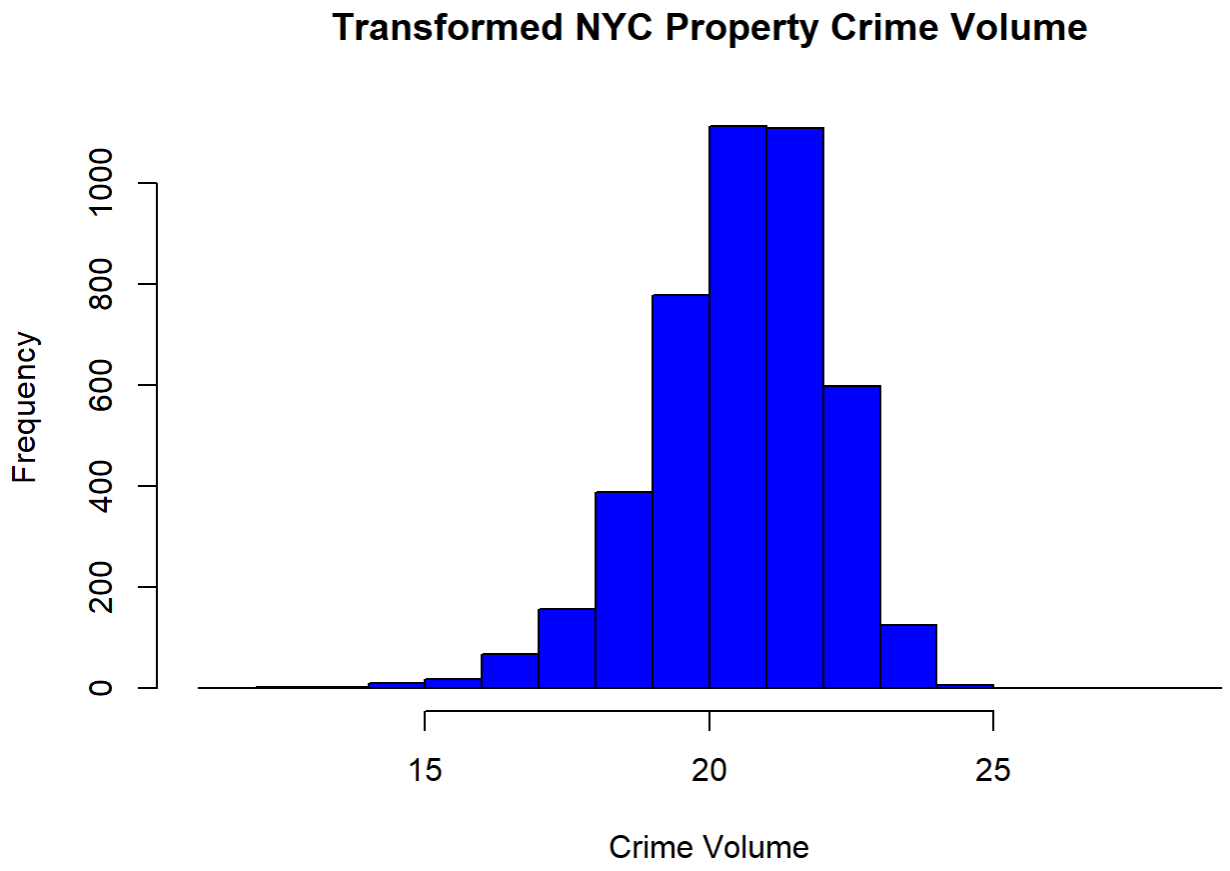

```
hist(nyc_violent.tr,nclass=20,xlab="Crime Volume", main="Transformed NYC Violent Crime Volume",
 col="blue")
```



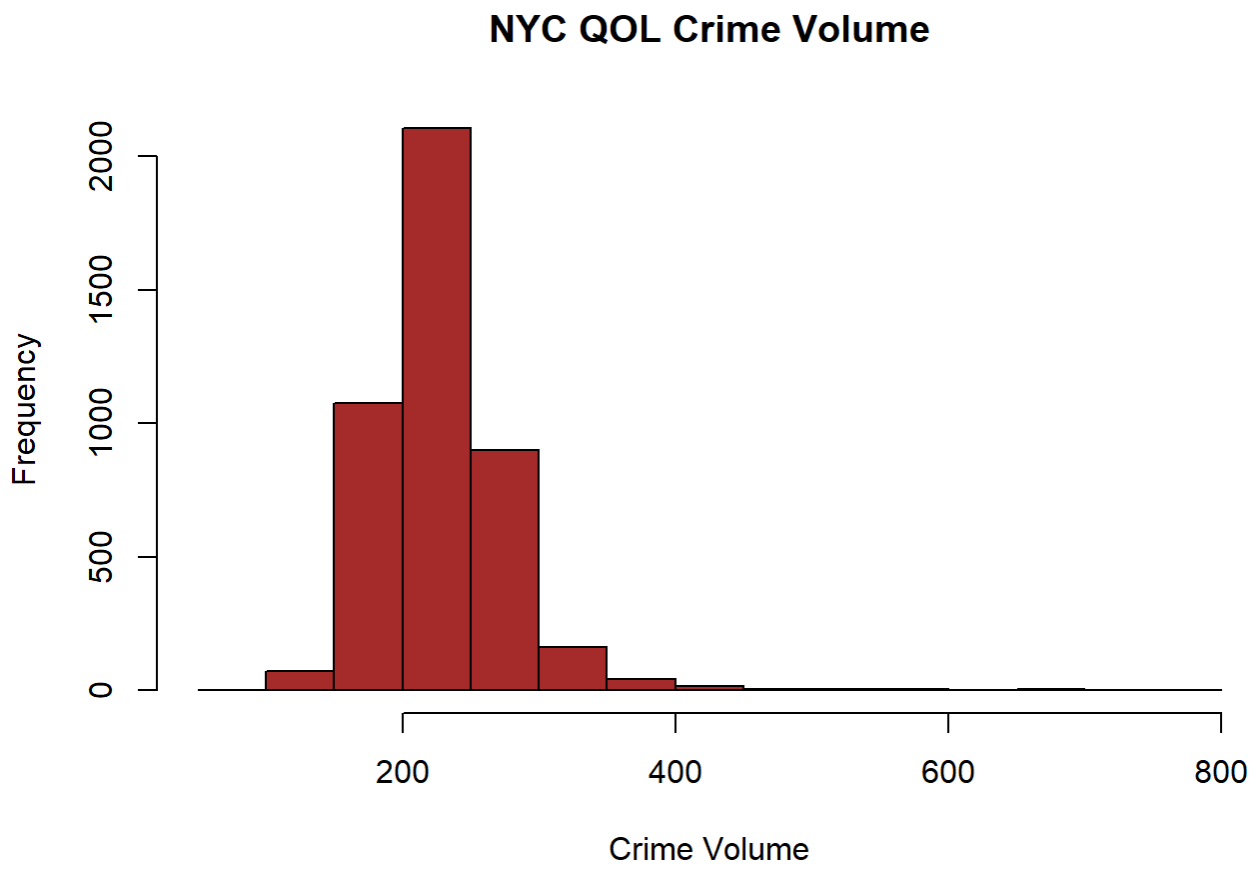

```
hist(nyc_prop$occurance_count,nclass=20,xlab="Crime Volume", main="NYC Property Crime Volume",co
l="brown")
```



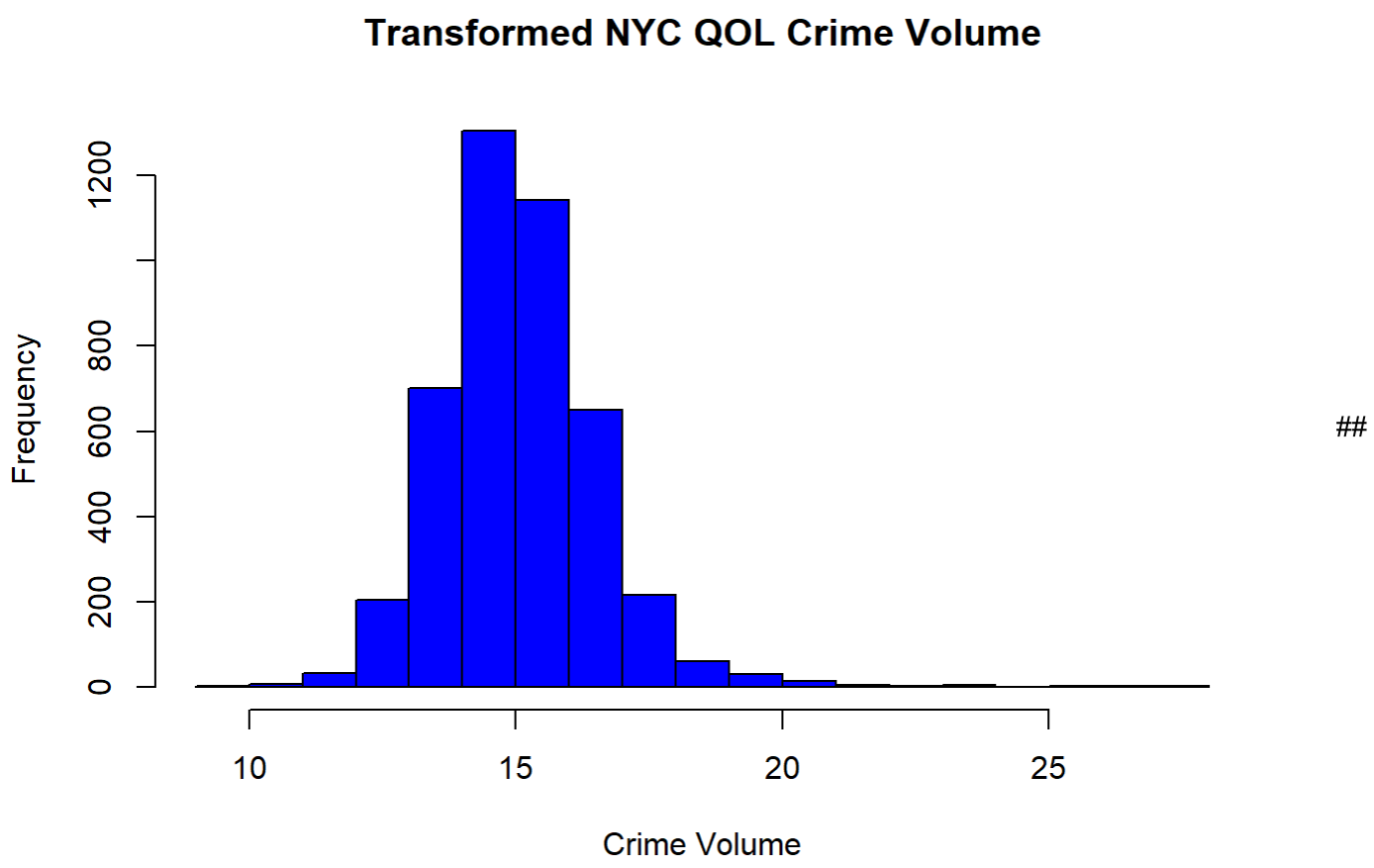

```
hist(nyc_prop.tr,nclass=20,xlab="Crime Volume", main="Transformed NYC Property Crime Volume", co
l="blue")
```




```
hist(nyc_qol$occurance_count,nclass=20,xlab="Crime Volume", main="NYC QOL Crime Volume",col="bro
wn")
```



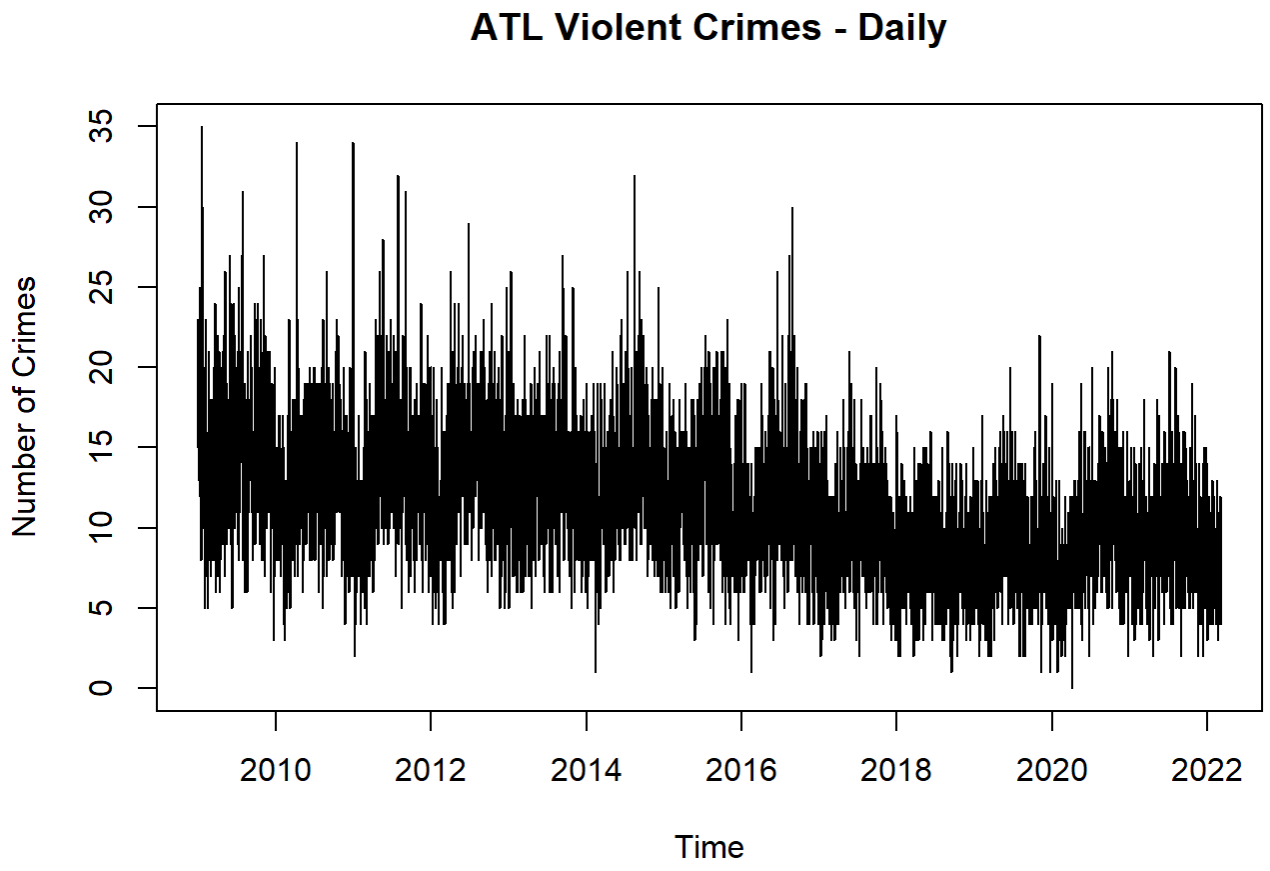

```
hist(nyc_qol.tr,nclass=20,xlab="Crime Volume", main="Transformed NYC QOL Crime Volume", col="blu
e")
```



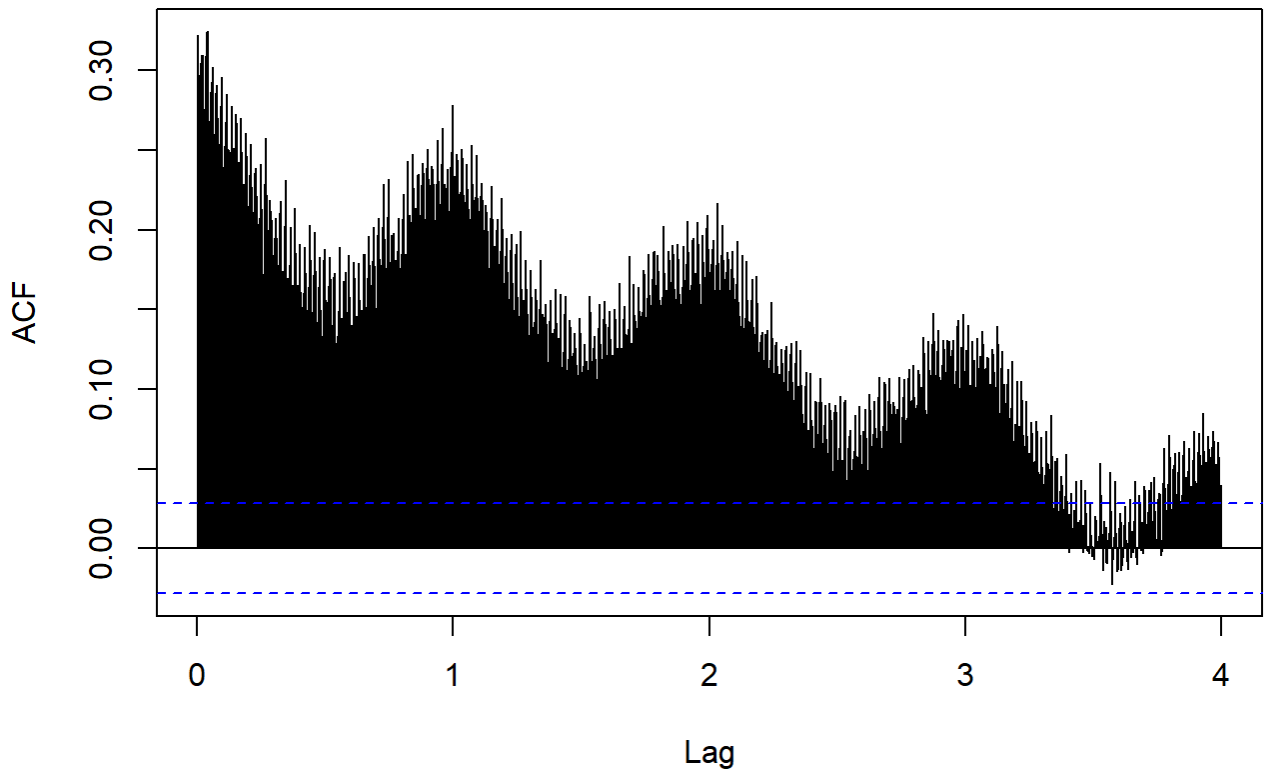
##

ATL Violent Crime

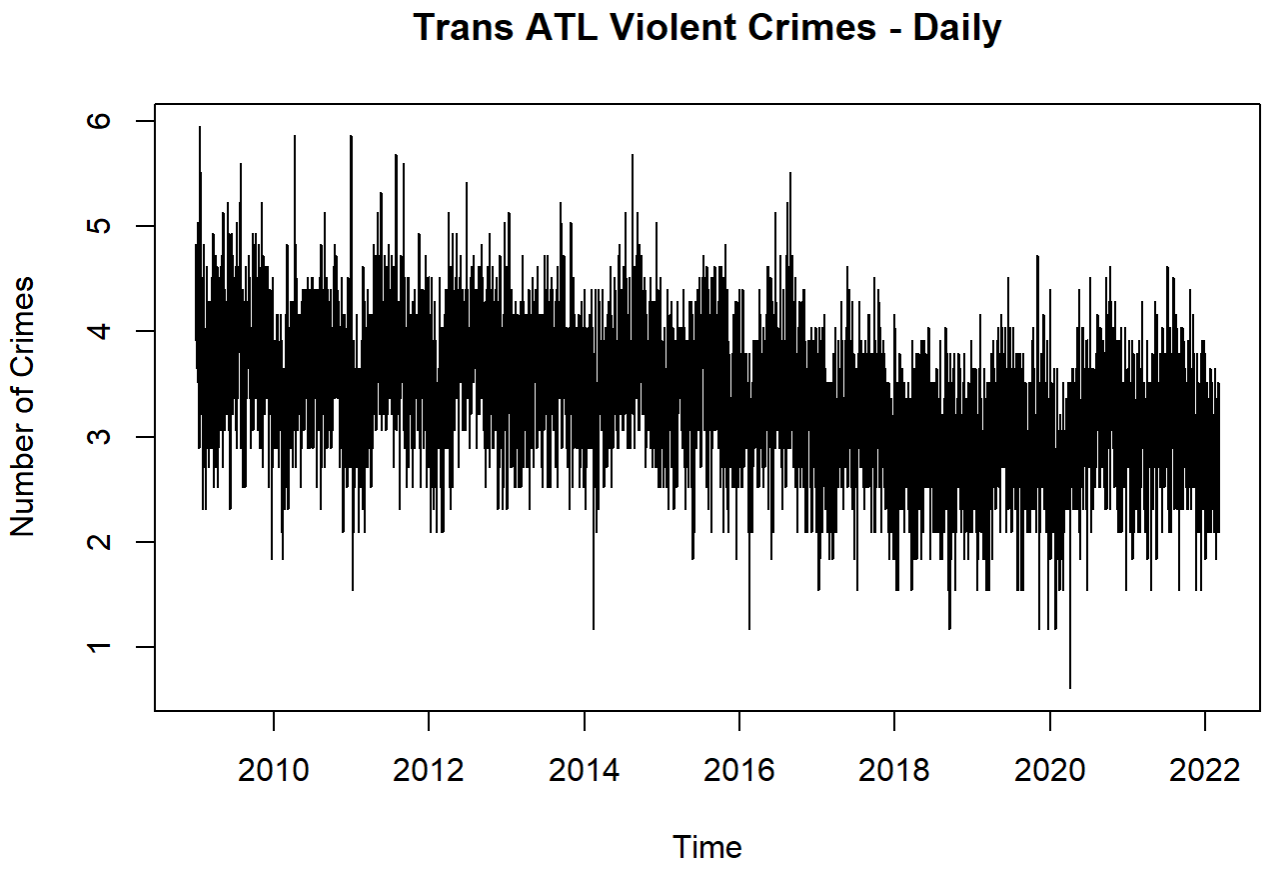
```
###EDA - Time Series / ACF
ts.plot(atl_v,ylab="Number of Crimes",main="ATL Violent Crimes - Daily")
```



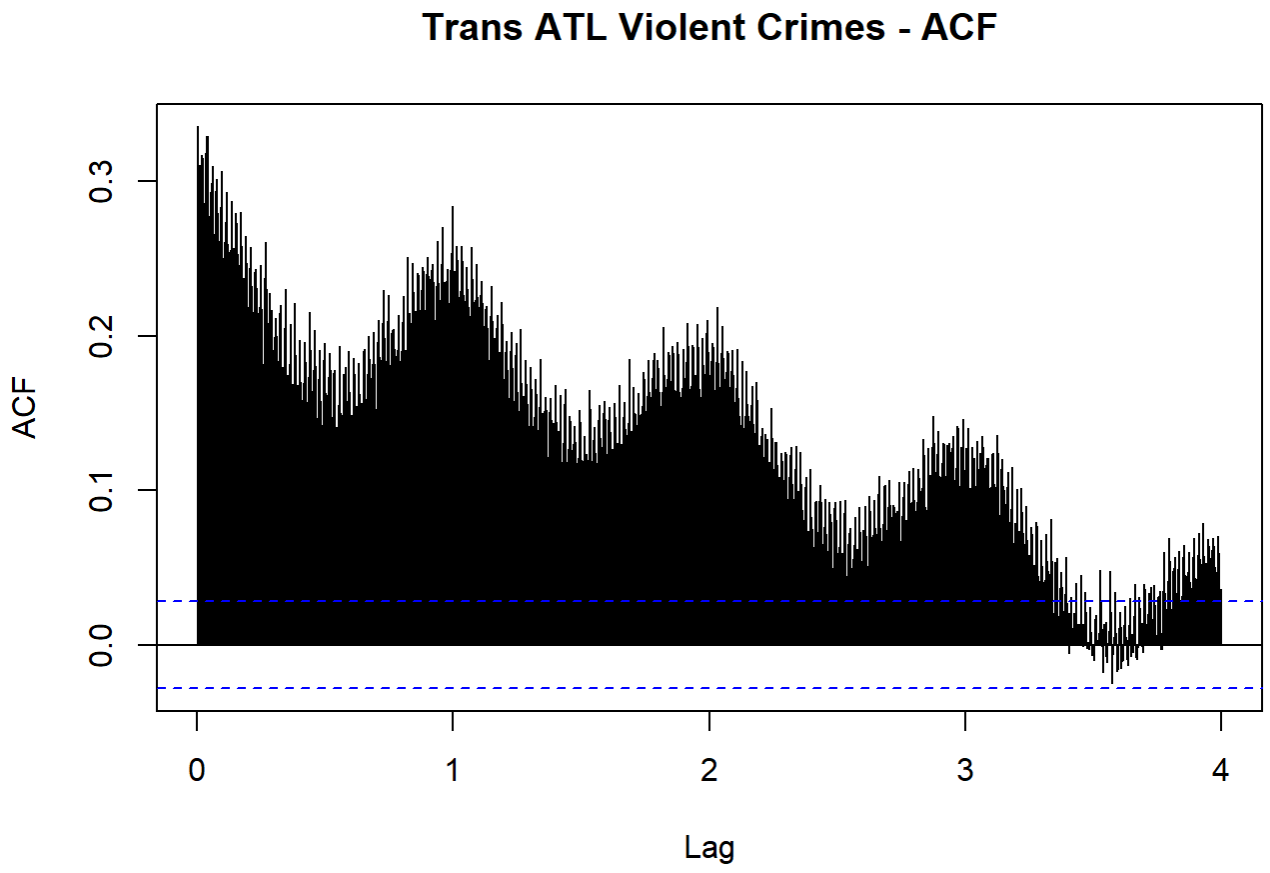

```
acf(atl_v,lag.max=365*4,main="ATL Violent Crimes - ACF")
```



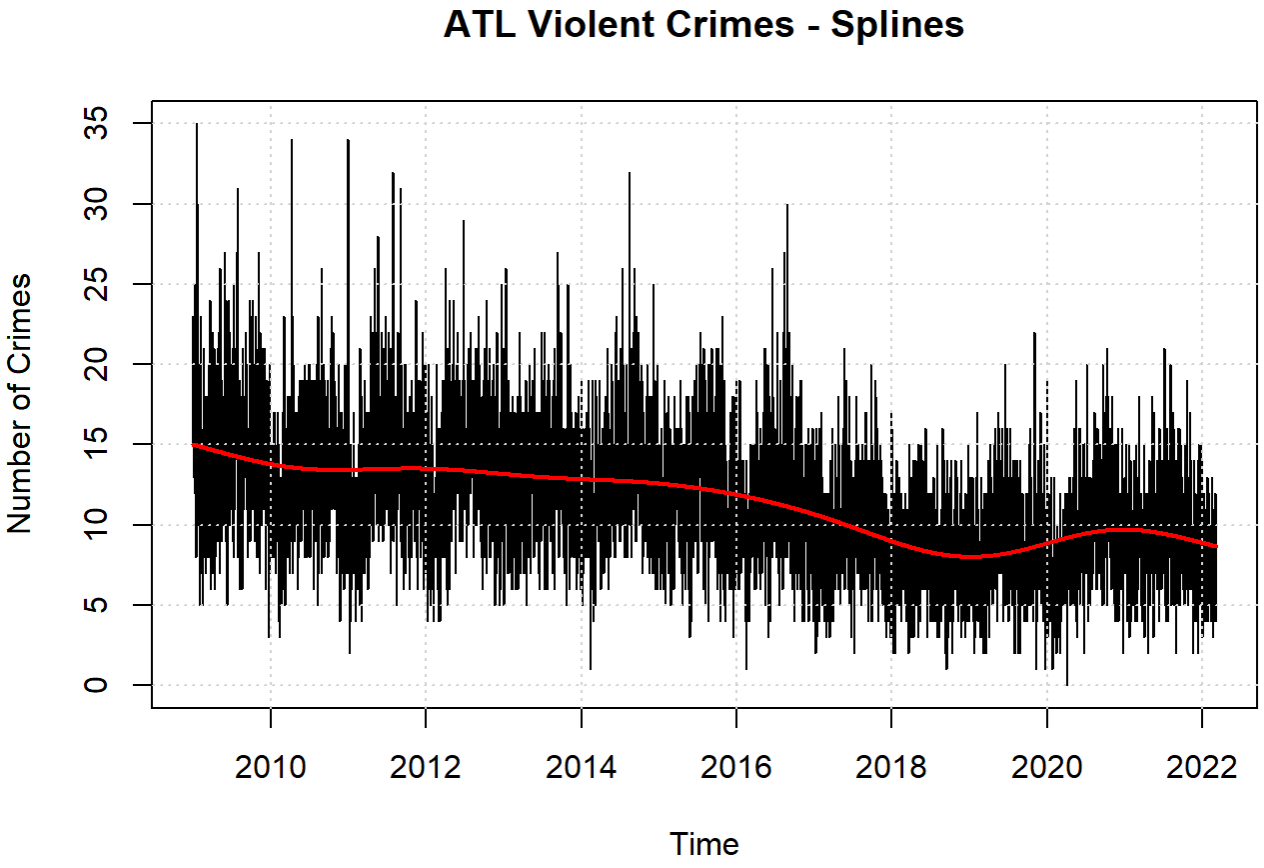

```
ts.plot(atl_v.tr,ylab="Number of Crimes",main="Trans ATL Violent Crimes - Daily")
```



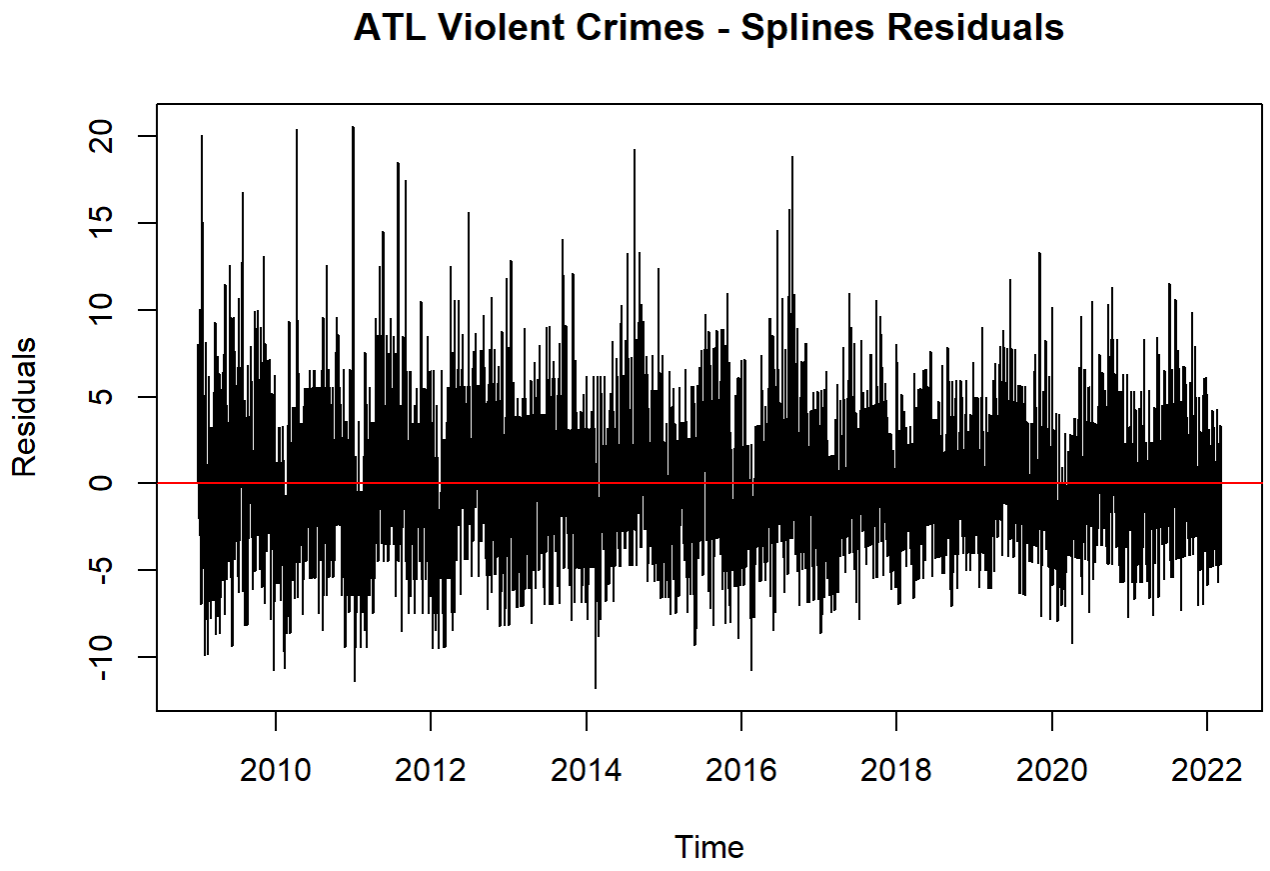

```
acf(atl_v.tr,lag.max=365*4,main="Trans ATL Violent Crimes - ACF")
```



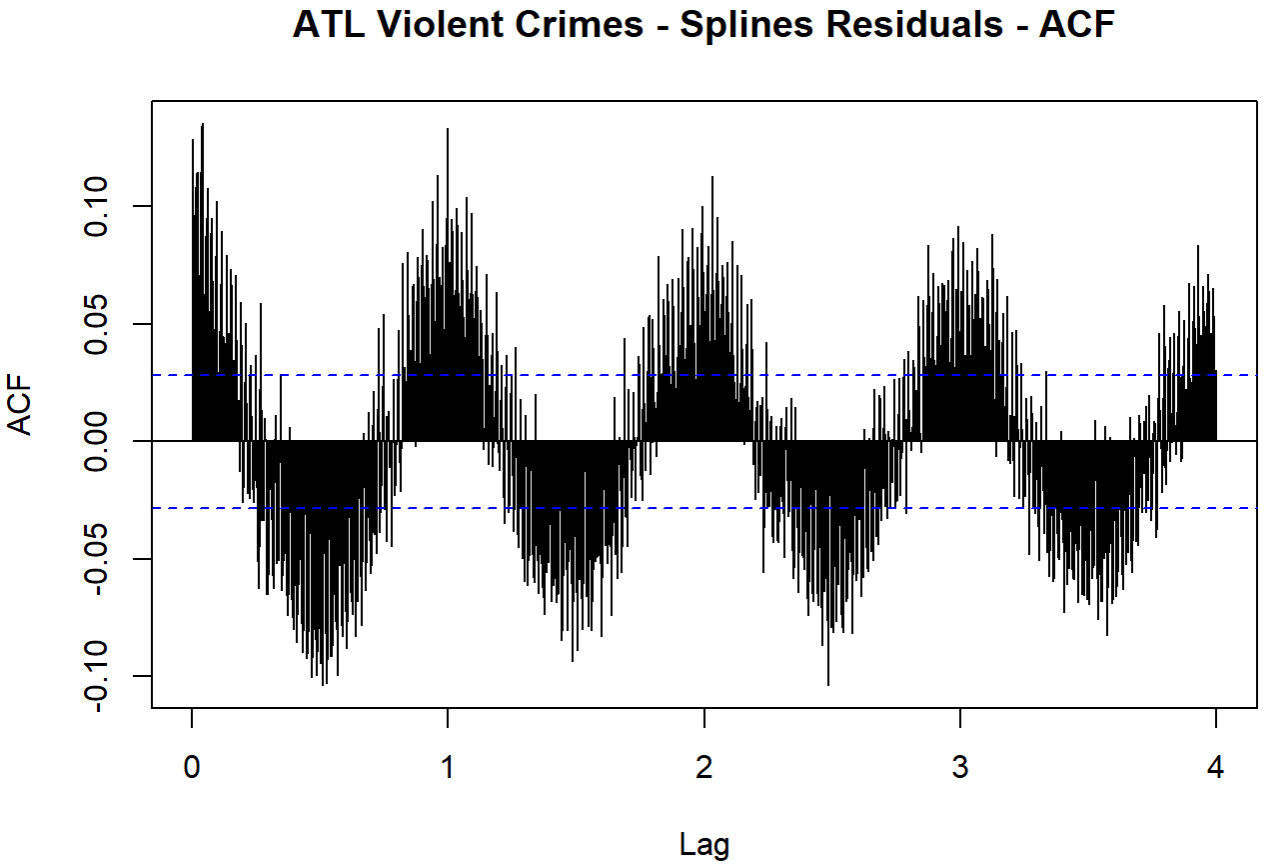

```
###TREND ESTIMATION - SPLINES
# Convert X-Axis to 0-1 Scale
time.pts = c(1:length(atl_v))
time.pts = c(time.pts - min(time.pts))/max(time.pts)

#Splines Trend Estimation
gam.fit = gam(atl_v~s(time.pts))
atl_v.fit.gam = ts(fitted(gam.fit),start=2009,frequency=365)

##Is there a trend?
ts.plot(atl_v,ylab="Number of Crimes", main = "ATL Violent Crimes - Splines")
grid()
lines(atl_v.fit.gam,lwd=2,col="red")
```



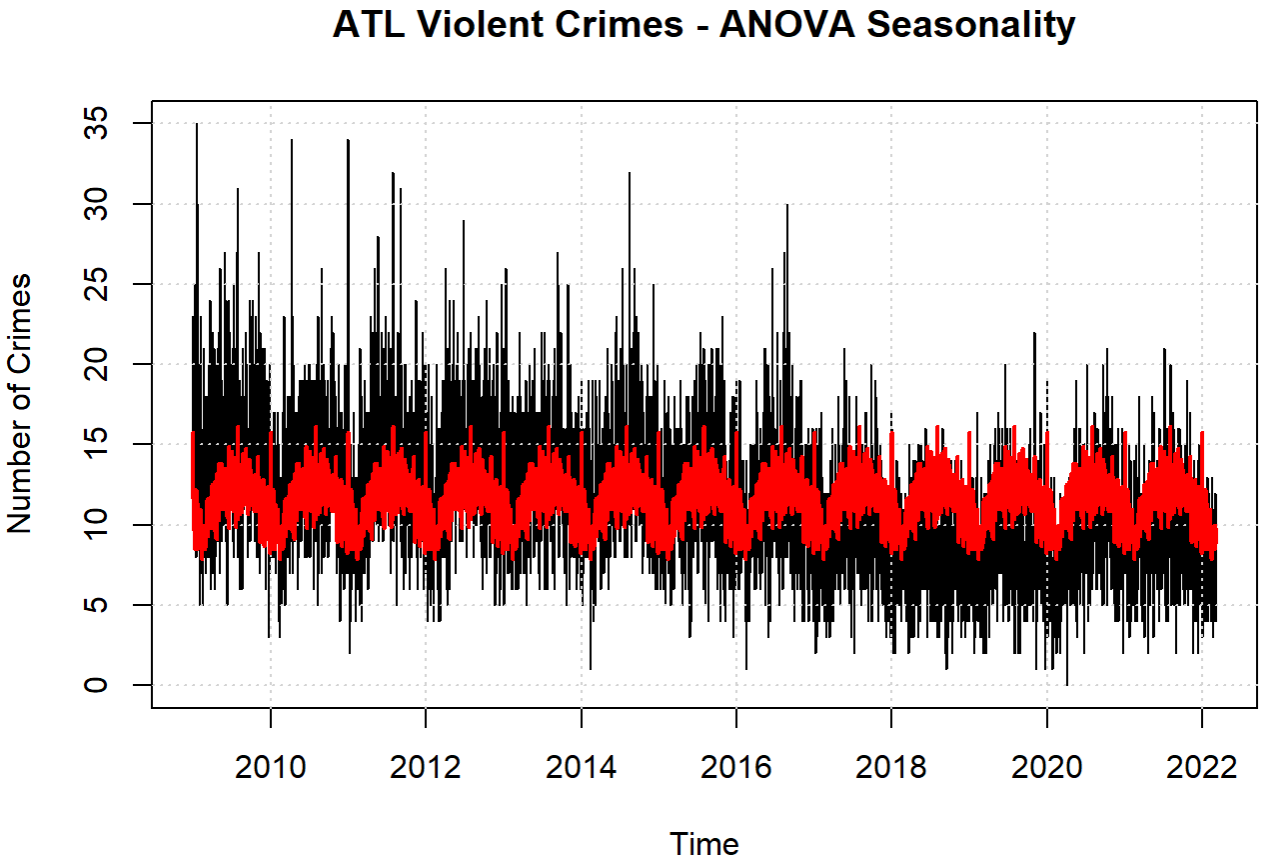

```
# Splines Residuals + Residuals ACF
dif.fit.gam = atl_v - atl_v.fit.gam
ts.plot(dif.fit.gam, ylab = "Residuals", main = "ATL Violent Crimes - Splines Residuals")
abline(h=0, col='red')
```



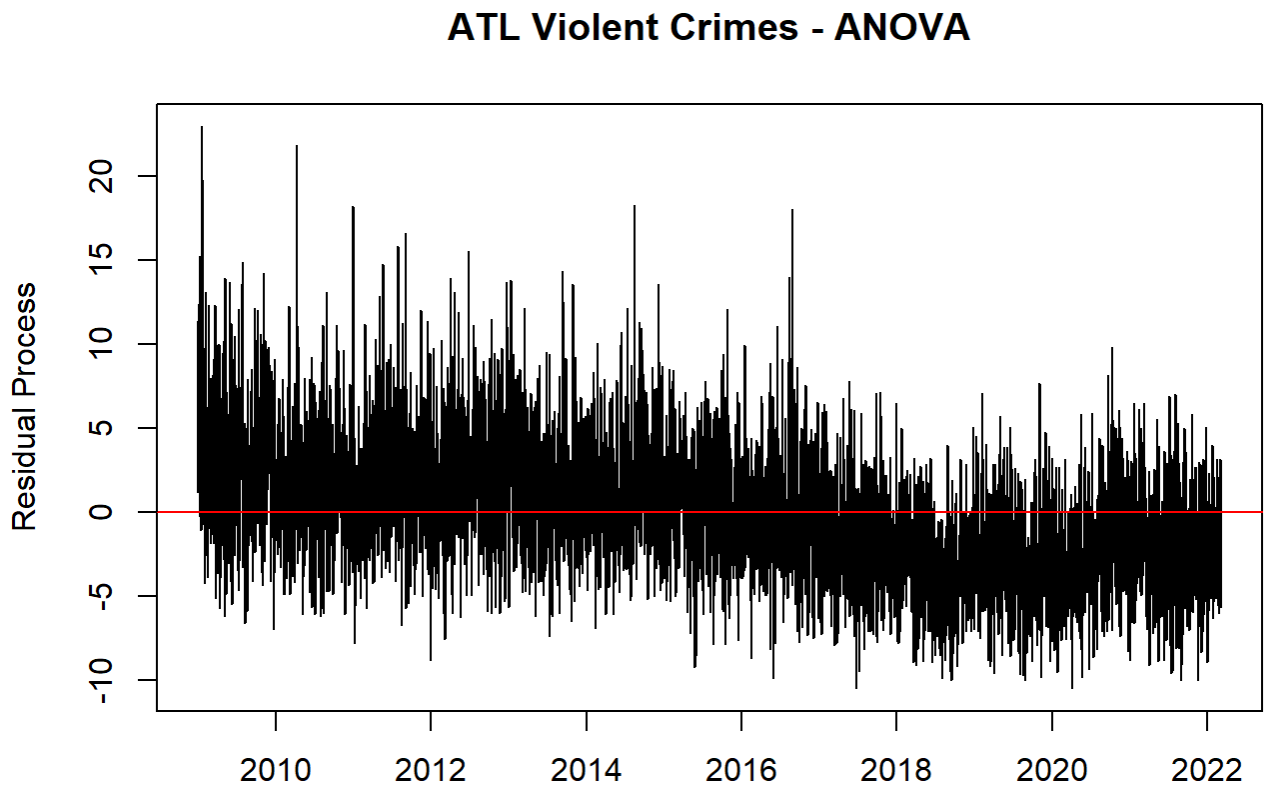

```
acf(dif.fit.gam, lag.max = 365 * 4, main = "ATL Violent Crimes - Splines Residuals - ACF")
```



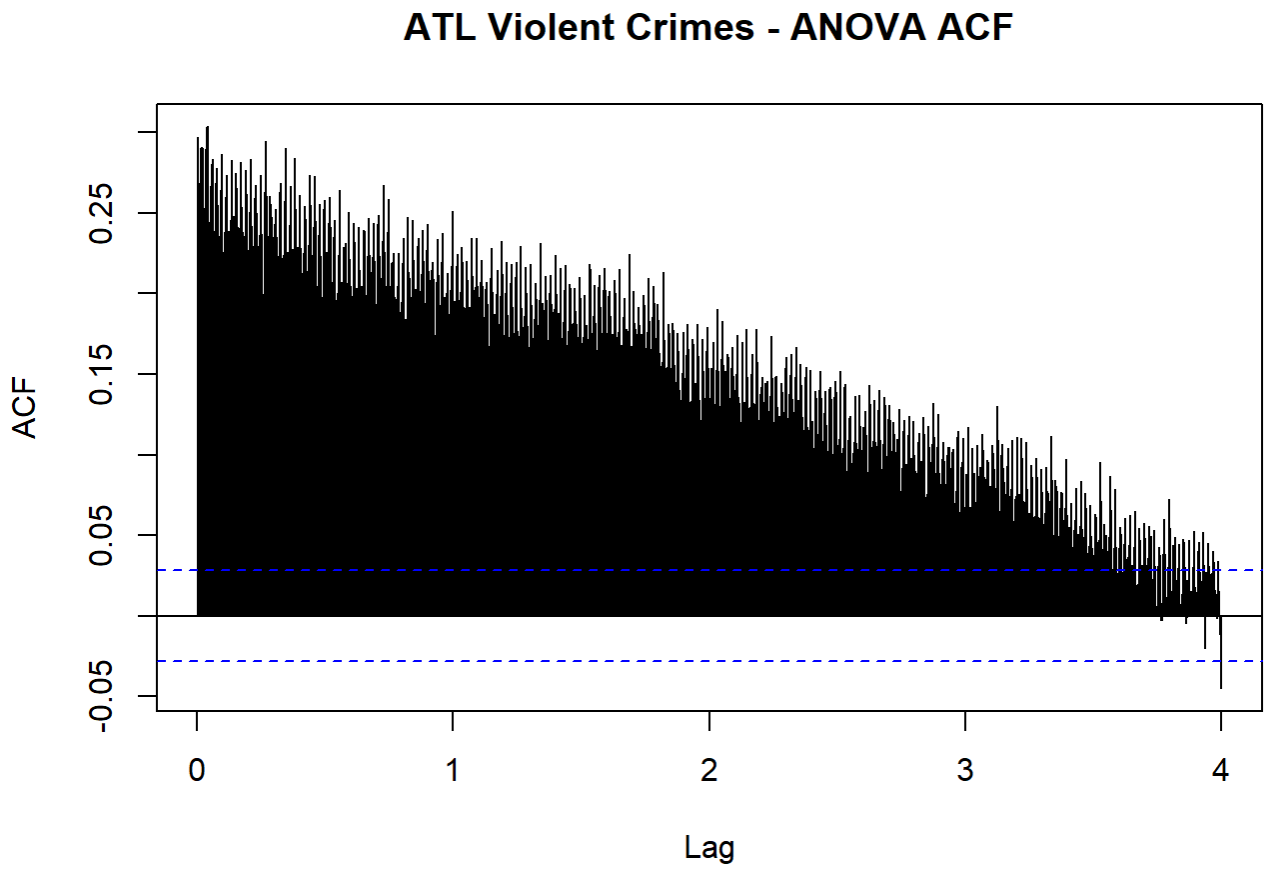

```
###SEASONALITY ANALYSIS - ANOVA

## Estimate seasonality using ANOVA approach
model.anova = dynlm(atl_v~season(atl_v))
#summary(model.anova)

## Plot
ts.plot(atl_v,ylab="Number of Crimes", main = "ATL Violent Crimes - ANOVA Seasonality")
grid()
lines(fitted(model.anova),lwd=2,col="red")
```




```
###Seasonality ANOVA- Residuals + Resid ACF
resid.anova <- residuals(model.anova)

ts.plot(resid.anova, xlab = "", ylab = "Residual Process", main = "ATL Violent Crimes - ANOVA")
abline(h=0, col='red')
```




```
acf(resid.anova, lag.max = 365 * 4, main = "ATL Violent Crimes - ANOVA ACF")
```




```
###SEASONALITY ANALYSIS - HARMONIC
###SUMMARY
harmonic.1 = dynlm(atl_v~harmon(atl_v))
summary(harmonic.1)
```

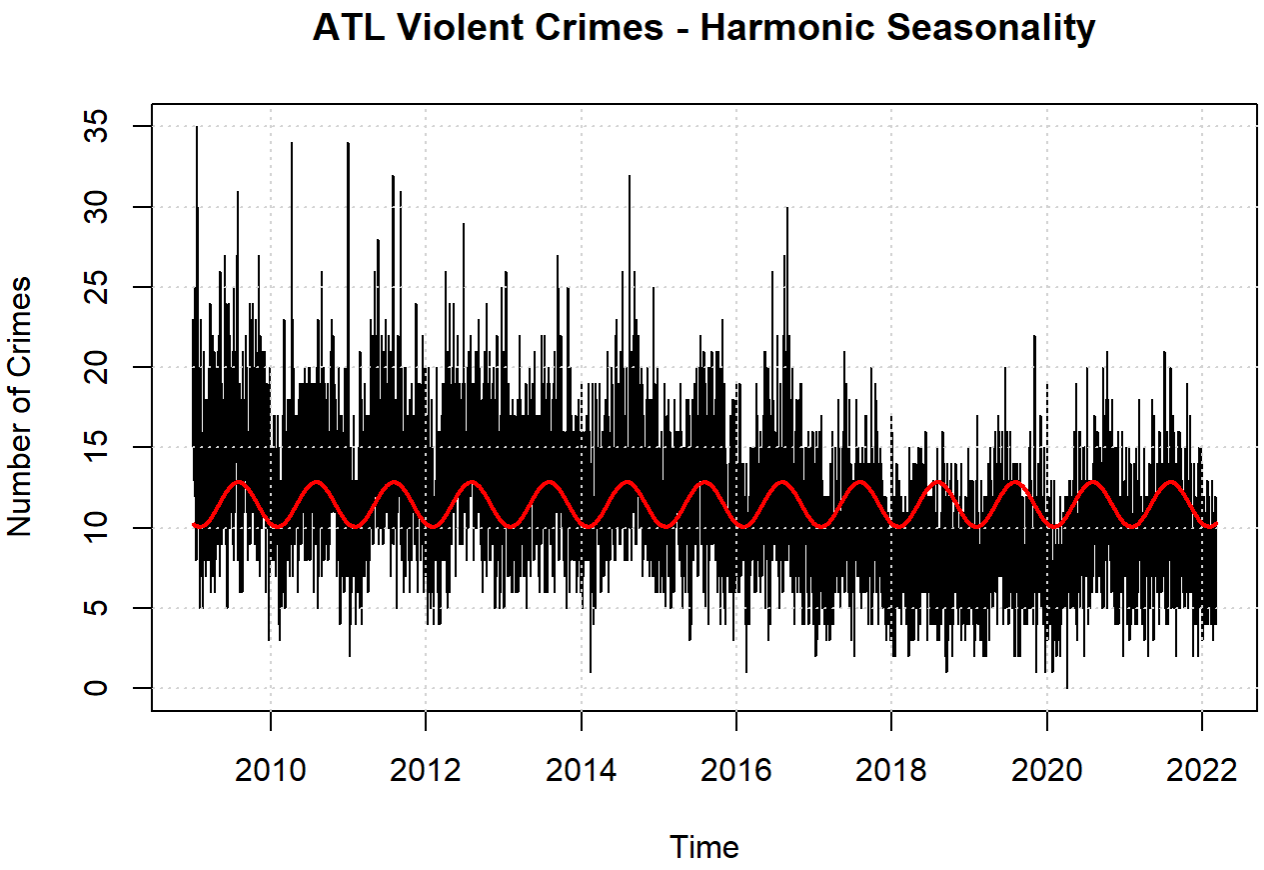
```
##
## Time series regression with "ts" data:
## Start = 2009(1), End = 2022(65)
##
## Call:
## dynlm(formula = atl_v ~ harmon(atl_v))
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -11.456  -3.111  -0.287   2.731  24.894
##
## Coefficients:
##                  Estimate Std. Error t value Pr(>|t|)
## (Intercept)      11.46067    0.06262 183.030  < 2e-16 ***
## harmon(atl_v)cos -1.20444    0.08834 -13.635  < 2e-16 ***
## harmon(atl_v)sin -0.69831    0.08876  -7.867 4.45e-15 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 4.342 on 4807 degrees of freedom
## Multiple R-squared:  0.04941,    Adjusted R-squared:  0.04902
## F-statistic: 124.9 on 2 and 4807 DF,  p-value: < 2.2e-16
```

```
harmonic.2 = dynlm(atl_v~harmon(atl_v,2))
summary(harmonic.2)
```

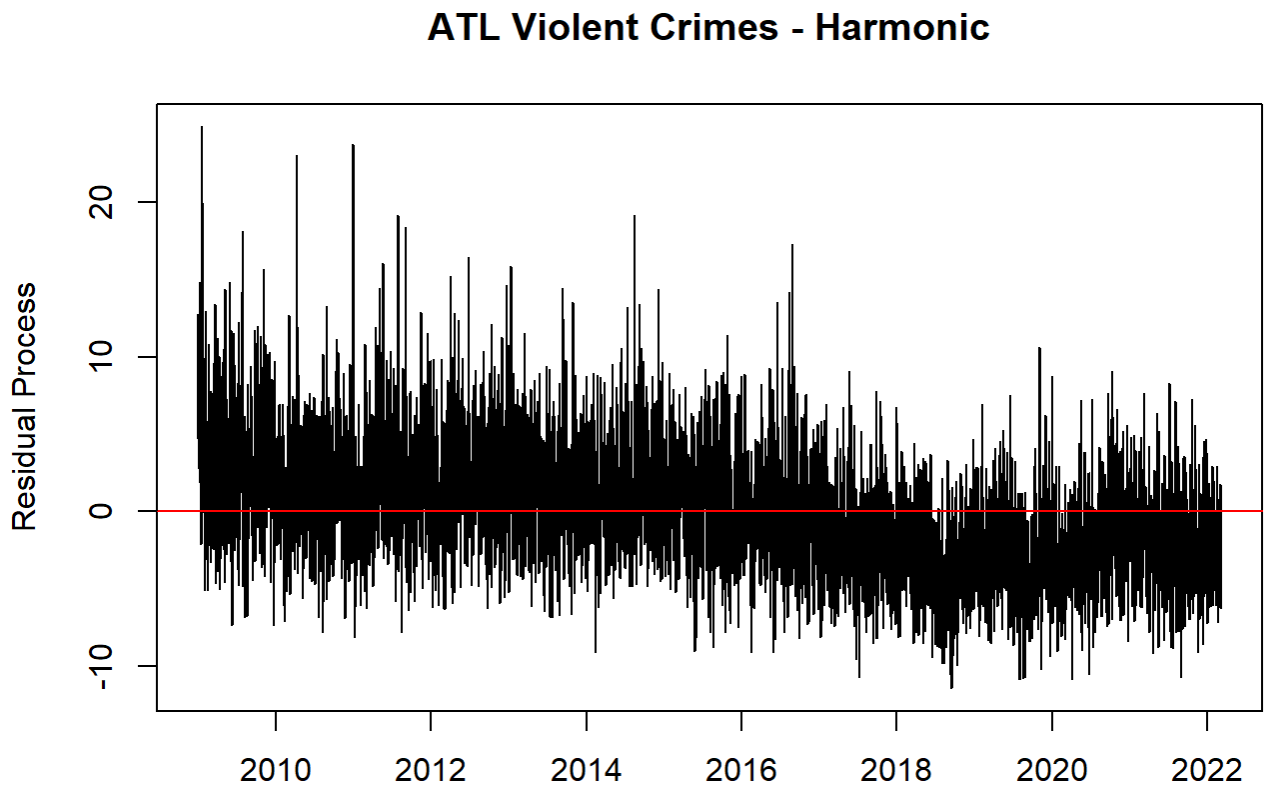
```
##
## Time series regression with "ts" data:
## Start = 2009(1), End = 2022(65)
##
## Call:
## dynlm(formula = atl_v ~ harmon(atl_v, 2))
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -11.2790  -3.1140  -0.3648   2.6857  25.0937
##
## Coefficients:
##                      Estimate Std. Error t value Pr(>|t|)
## (Intercept)          11.46414    0.06253 183.344  < 2e-16 ***
## harmon(atl_v, 2)cos1 -1.19916    0.08822 -13.594  < 2e-16 ***
## harmon(atl_v, 2)cos2 -0.01254    0.08854  -0.142    0.887
## harmon(atl_v, 2)sin1 -0.69421    0.08864  -7.832 5.86e-15 ***
## harmon(atl_v, 2)sin2 -0.36010    0.08830  -4.078 4.62e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 4.335 on 4805 degrees of freedom
## Multiple R-squared:  0.05269,    Adjusted R-squared:  0.05191
## F-statistic: 66.82 on 4 and 4805 DF,  p-value: < 2.2e-16
```

```
###not all 2 harmonic values statistically significant - step down to simpler model
```

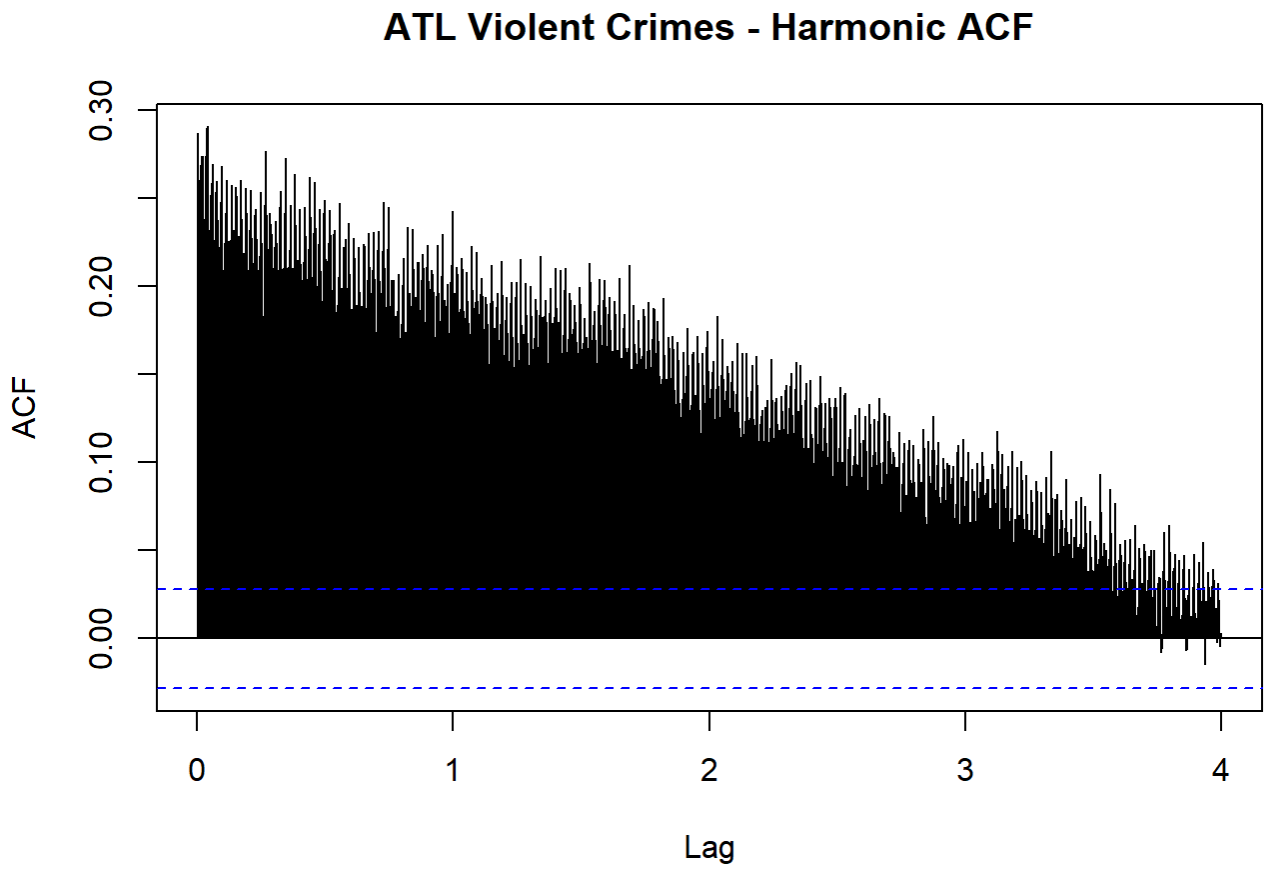
```
###Harmonic Plot
## Plot
ts.plot(atl_v,ylab="Number of Crimes", main = "ATL Violent Crimes - Harmonic Seasonality")
grid()
lines(fitted(harmonic.1),lwd=2,col="red")
```




```
###Seasonality Harmonic - Residuals + Resid ACF

resid.harmonic <- residuals(harmonic.1)

ts.plot(resid.harmonic, xlab = "", ylab = "Residual Process", main = "ATL Violent Crimes - Harmo
nic")
abline(h=0, col='red')
```



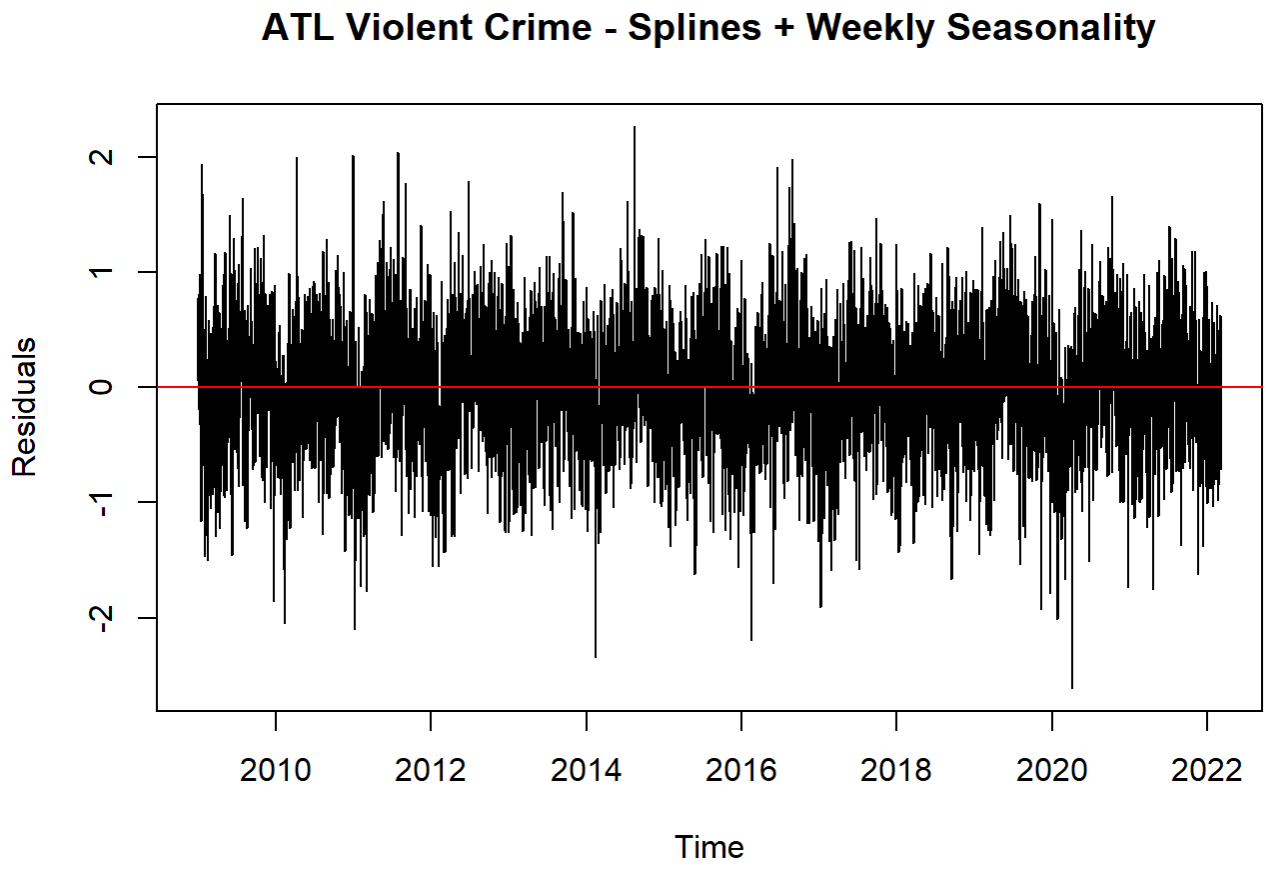

```
acf(resid.harmonic, lag.max = 365 * 4, main = "ATL Violent Crimes - Harmonic ACF")
```



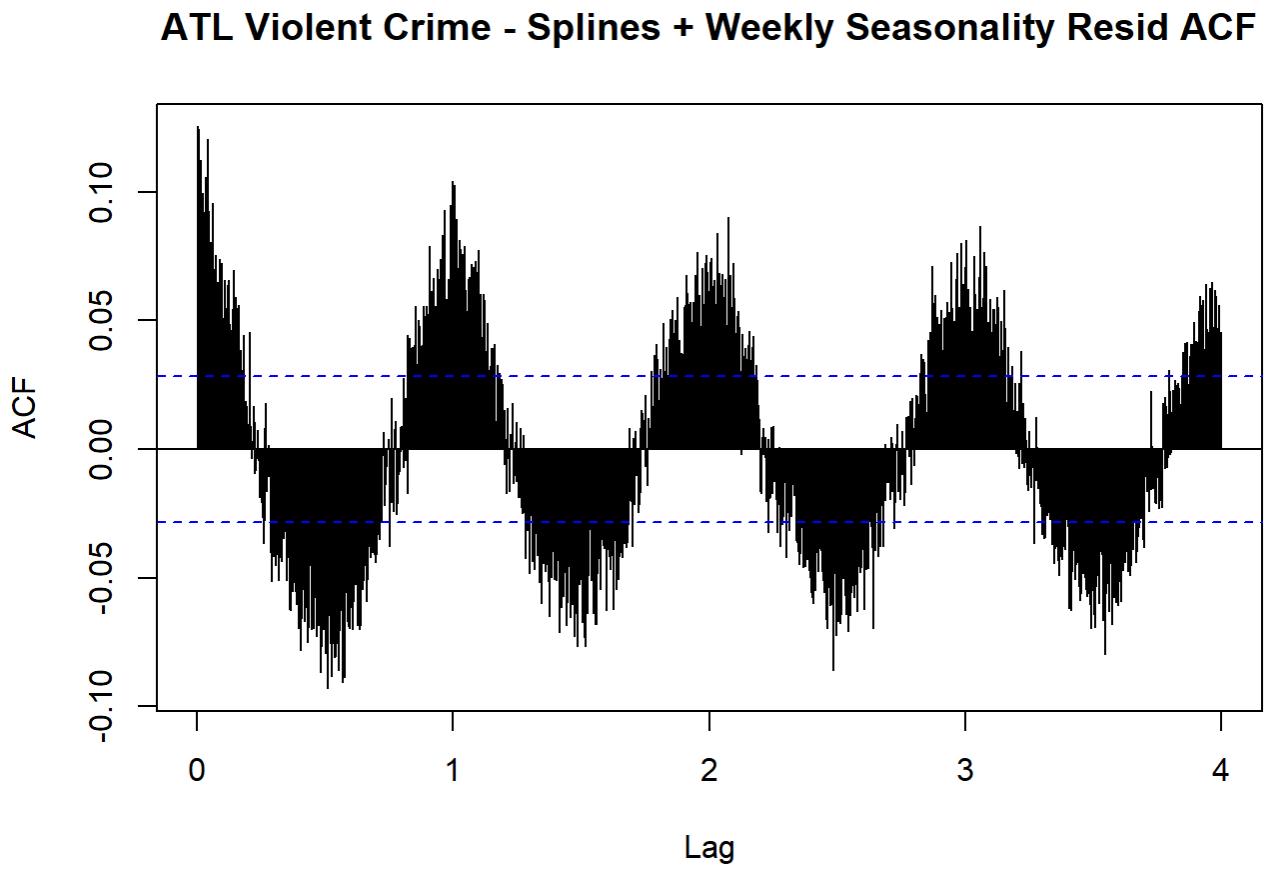

```
###TREND + SEASONALITY - SPLINES + WEEKLY SEASONALITY - TRANSFORMED
atl_violent$Date <- as.Date(atl_violent$Date)
year <- as.factor(format(atl_violent$Date,'%Y'))
month <- as.factor(format(atl_violent$Date, '%b'))
week <- as.factor(weekdays(atl_violent$Date))

gam.fit.seastr = gam(atl_v.tr~s(time.pts)+week-1)
summary(gam.fit.seastr)
```

```
##
## Family: gaussian
## Link function: identity
##
## Formula:
## atl_v.tr ~ s(time.pts) + week - 1
##
## Parametric coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## weekFriday     3.35695    0.02160   155.4   <2e-16 ***
## weekMonday     3.34340    0.02160   154.8   <2e-16 ***
## weekSaturday   3.56113    0.02160   164.8   <2e-16 ***
## weekSunday     3.52354    0.02160   163.1   <2e-16 ***
## weekThursday   3.22643    0.02159   149.5   <2e-16 ***
## weekTuesday    3.31407    0.02160   153.4   <2e-16 ***
## weekWednesday  3.29528    0.02160   152.5   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Approximate significance of smooth terms:
##                edf Ref.df     F p-value
## s(time.pts) 8.609  8.956 165.4  <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## R-sq.(adj) =  0.257   Deviance explained = 97.3%
## GCV = 0.32166  Scale est. = 0.32062   n = 4810
```

```
dif.fit.seastr = ts(atl_v.tr - fitted(gam.fit.seastr),start=2009,frequency=365)

ts.plot(dif.fit.seastr, ylab = "Residuals", main = "ATL Violent Crime - Splines + Weekly Seasona
lity")
abline(h=0, col='red')
```



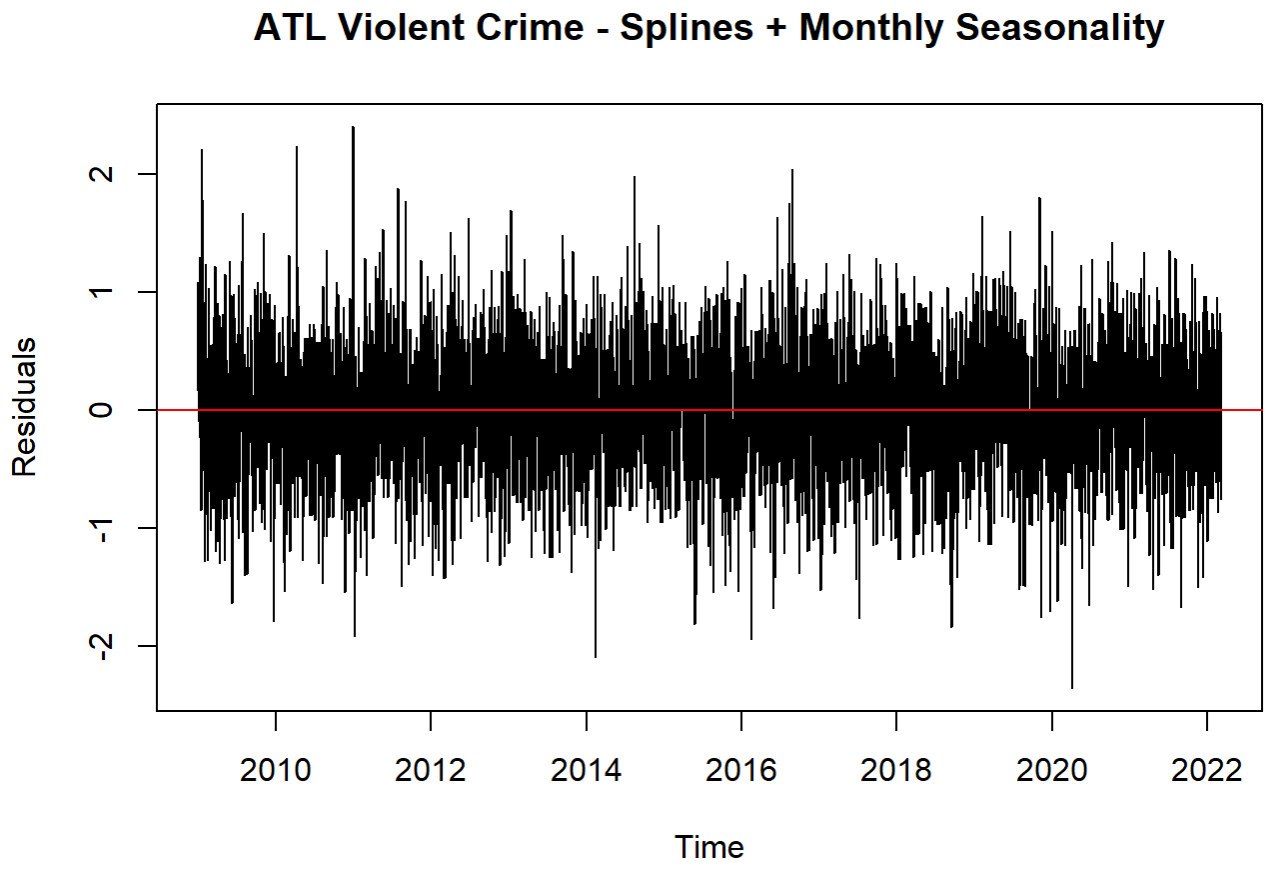

```
acf(dif.fit.seastr, lag.max = 365 * 4, main = "ATL Violent Crime - Splines + Weekly Seasonality
 Resid ACF")
```



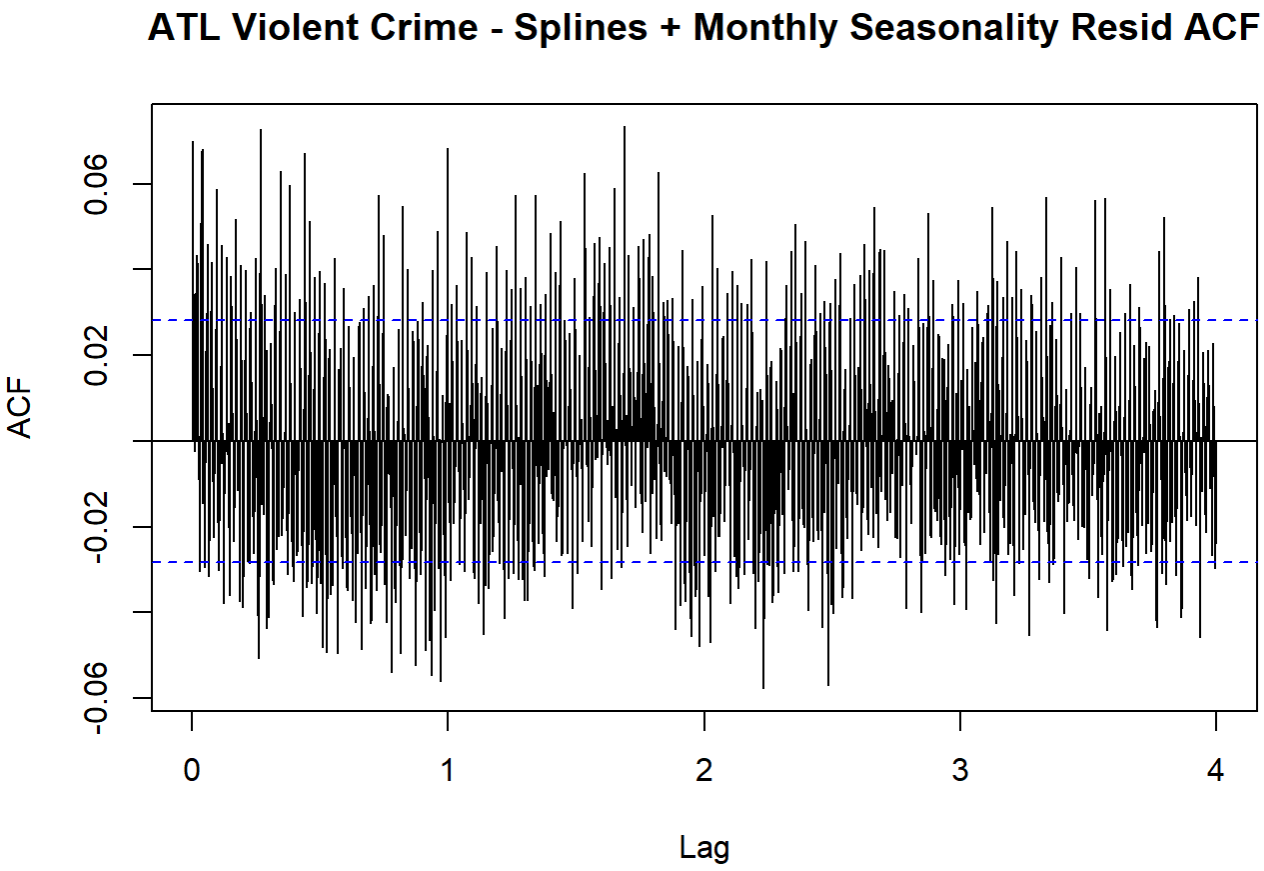

```
###TREND + SEASONALITY - SPLINES + MONTHLY SEASONALITY - TRANSFORMED
atl_violent$Date <- as.Date(atl_violent$Date)
year <- as.factor(format(atl_violent$Date,'%Y'))
month <- as.factor(format(atl_violent$Date, '%b'))
week <- as.factor(weekdays(atl_violent$Date))

gam.fit.seastr = gam(atl_v.tr~s(time.pts)+month-1)
summary(gam.fit.seastr)
```

```
##
## Family: gaussian
## Link function: identity
##
## Formula:
## atl_v.tr ~ s(time.pts) + month - 1
##
## Parametric coefficients:
##          Estimate Std. Error t value Pr(>|t|)
## monthApr  3.32566    0.02823   117.8   <2e-16 ***
## monthAug  3.51006    0.02774   126.6   <2e-16 ***
## monthDec  3.29128    0.02779   118.4   <2e-16 ***
## monthFeb  3.05773    0.02809   108.9   <2e-16 ***
## monthJan  3.18493    0.02681   118.8   <2e-16 ***
## monthJul  3.54659    0.02774   127.9   <2e-16 ***
## monthJun  3.50562    0.02820   124.3   <2e-16 ***
## monthMar  3.20994    0.02767   116.0   <2e-16 ***
## monthMay  3.50743    0.02775   126.4   <2e-16 ***
## monthNov  3.36569    0.02822   119.3   <2e-16 ***
## monthOct  3.47668    0.02775   125.3   <2e-16 ***
## monthSep  3.52768    0.02820   125.1   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Approximate significance of smooth terms:
##               edf Ref.df   F p-value
## s(time.pts) 8.583   8.95 172  <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## R-sq.(adj) =  0.282   Deviance explained = 97.4%
## GCV = 0.31117  Scale est. = 0.30984   n = 4810
```

```
dif.fit.seastr = ts(atl_v.tr - fitted(gam.fit.seastr),start=2009,frequency=365)

ts.plot(dif.fit.seastr, ylab = "Residuals", main = "ATL Violent Crime - Splines + Monthly Season
ality")
abline(h=0, col='red')
```



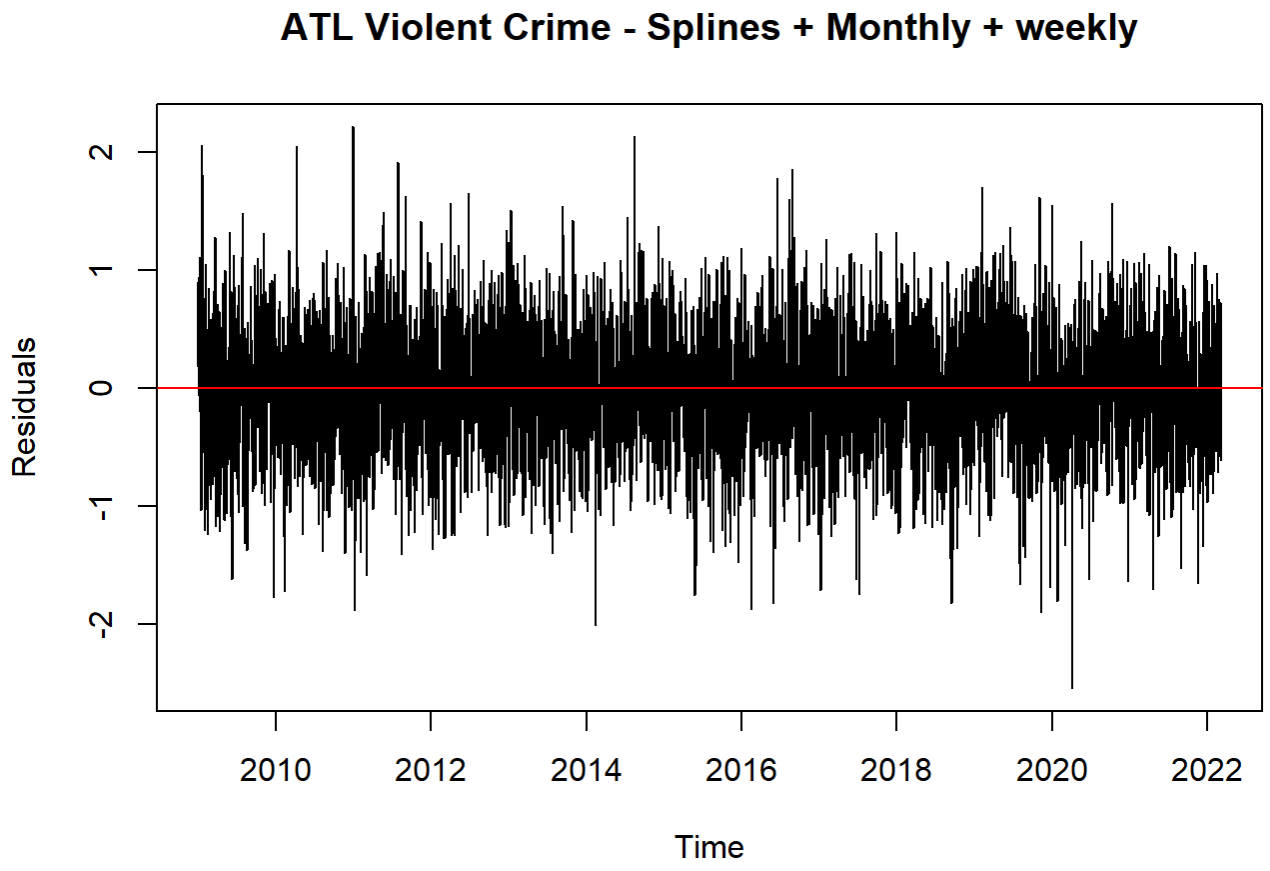

```
acf(dif.fit.seastr, lag.max = 365 * 4, main = "ATL Violent Crime - Splines + Monthly Seasonality
Resid ACF")
```



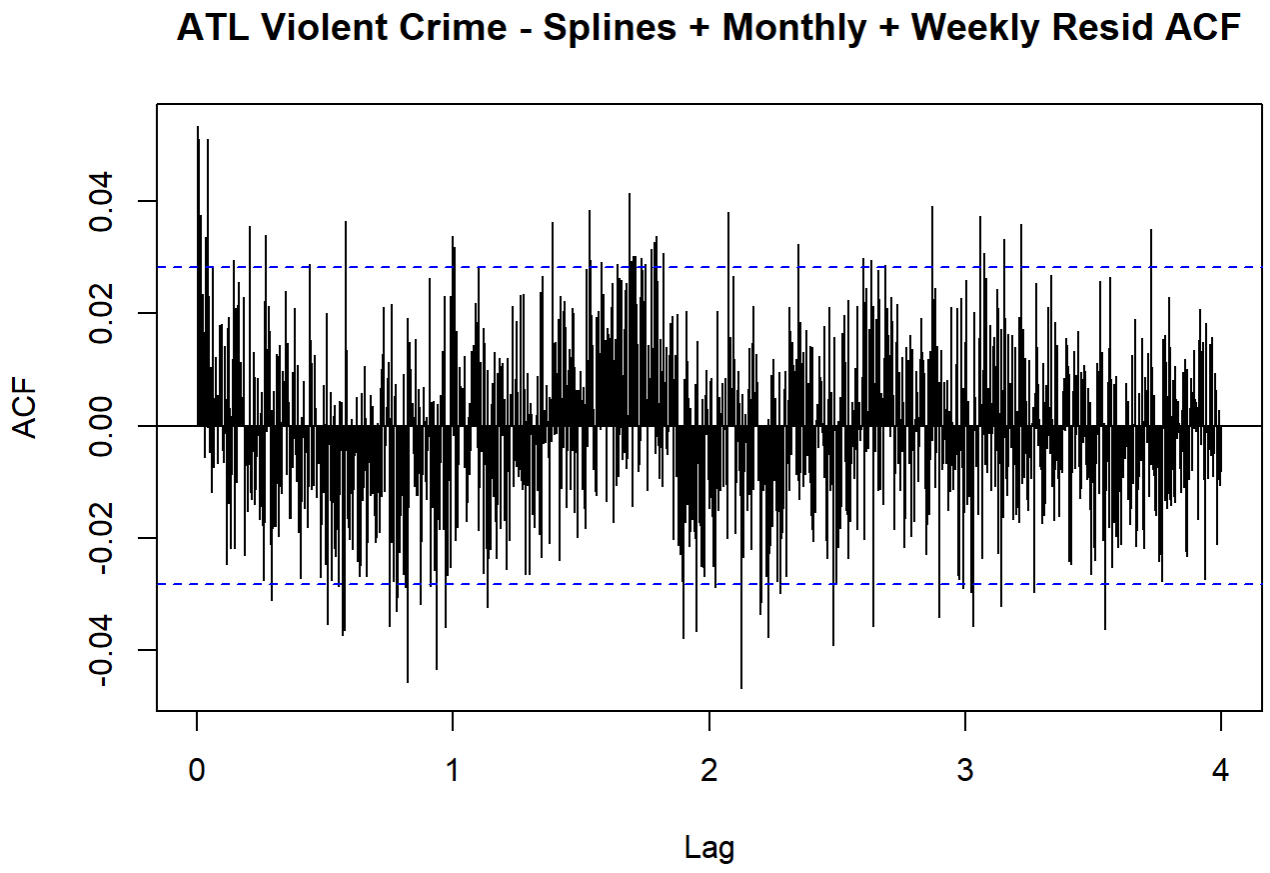

```
###TREND + SEASONALITY - SPLINES + MONTHLY + WEEKLY SEASONALITY
atl_violent$Date <- as.Date(atl_violent$Date)
year <- as.factor(format(atl_violent$Date,'%Y'))
month <- as.factor(format(atl_violent$Date, '%b'))
week <- as.factor(weekdays(atl_violent$Date))

gam.fit.seastr.1 = gam(atl_v.tr~s(time.pts)+month+week-1)
summary(gam.fit.seastr.1)
```

```
##
## Family: gaussian
## Link function: identity
##
## Formula:
## atl_v.tr ~ s(time.pts) + month + week - 1
##
## Parametric coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## monthApr       3.30902    0.03369  98.231  < 2e-16 ***
## monthAug       3.49165    0.03332 104.792  < 2e-16 ***
## monthDec       3.27462    0.03336  98.162  < 2e-16 ***
## monthFeb       3.03976    0.03360  90.481  < 2e-16 ***
## monthJan       3.16603    0.03253  97.321  < 2e-16 ***
## monthJul       3.52951    0.03326 106.132  < 2e-16 ***
## monthJun       3.48801    0.03370 103.510  < 2e-16 ***
## monthMar       3.19292    0.03328  95.933  < 2e-16 ***
## monthMay       3.48854    0.03330 104.756  < 2e-16 ***
## monthNov       3.34793    0.03368  99.399  < 2e-16 ***
## monthOct       3.45843    0.03330 103.861  < 2e-16 ***
## monthSep       3.51050    0.03370 104.183  < 2e-16 ***
## weekMonday    -0.01313    0.02942  -0.446   0.6554
## weekSaturday   0.20452    0.02942   6.952 4.08e-12 ***
## weekSunday     0.16680    0.02942   5.670 1.51e-08 ***
## weekThursday  -0.13013    0.02941  -4.425 9.85e-06 ***
## weekTuesday   -0.04243    0.02942  -1.442   0.1493
## weekWednesday -0.06146    0.02942  -2.089   0.0367 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Approximate significance of smooth terms:
##               edf Ref.df     F p-value
## s(time.pts) 8.598  8.954 179.1  <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## R-sq.(adj) =  0.311   Deviance explained = 97.5%
## GCV = 0.29893  Scale est. = 0.29728   n = 4810
```

```
dif.fit.seastr.1 = ts(atl_v.tr - fitted(gam.fit.seastr.1),start=2009,frequency=365)

ts.plot(dif.fit.seastr.1, ylab = "Residuals", main = "ATL Violent Crime - Splines + Monthly + we
ekly")
abline(h=0, col='red')
```



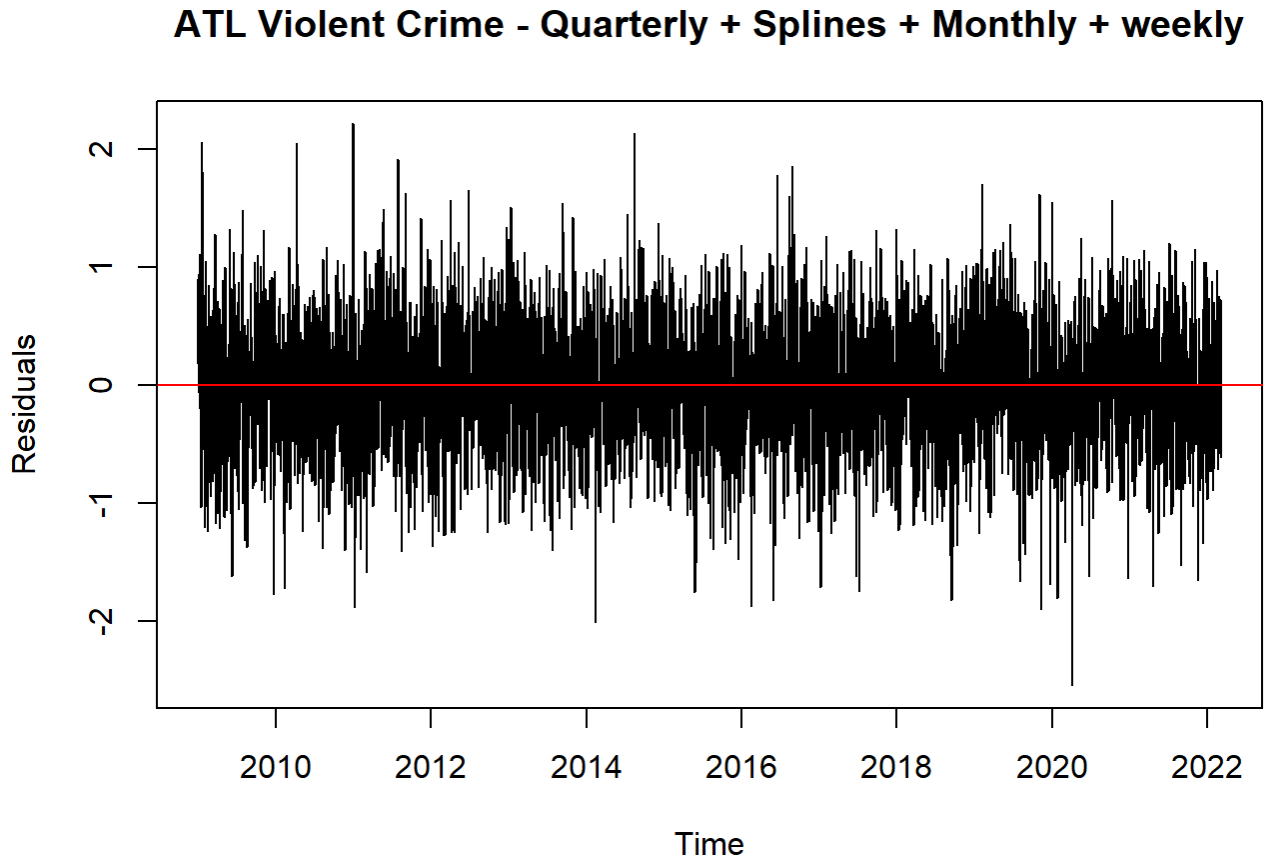

```
acf(dif.fit.seastr.1, lag.max = 365 * 4, main = "ATL Violent Crime - Splines + Monthly + Weekly
 Resid ACF")
```



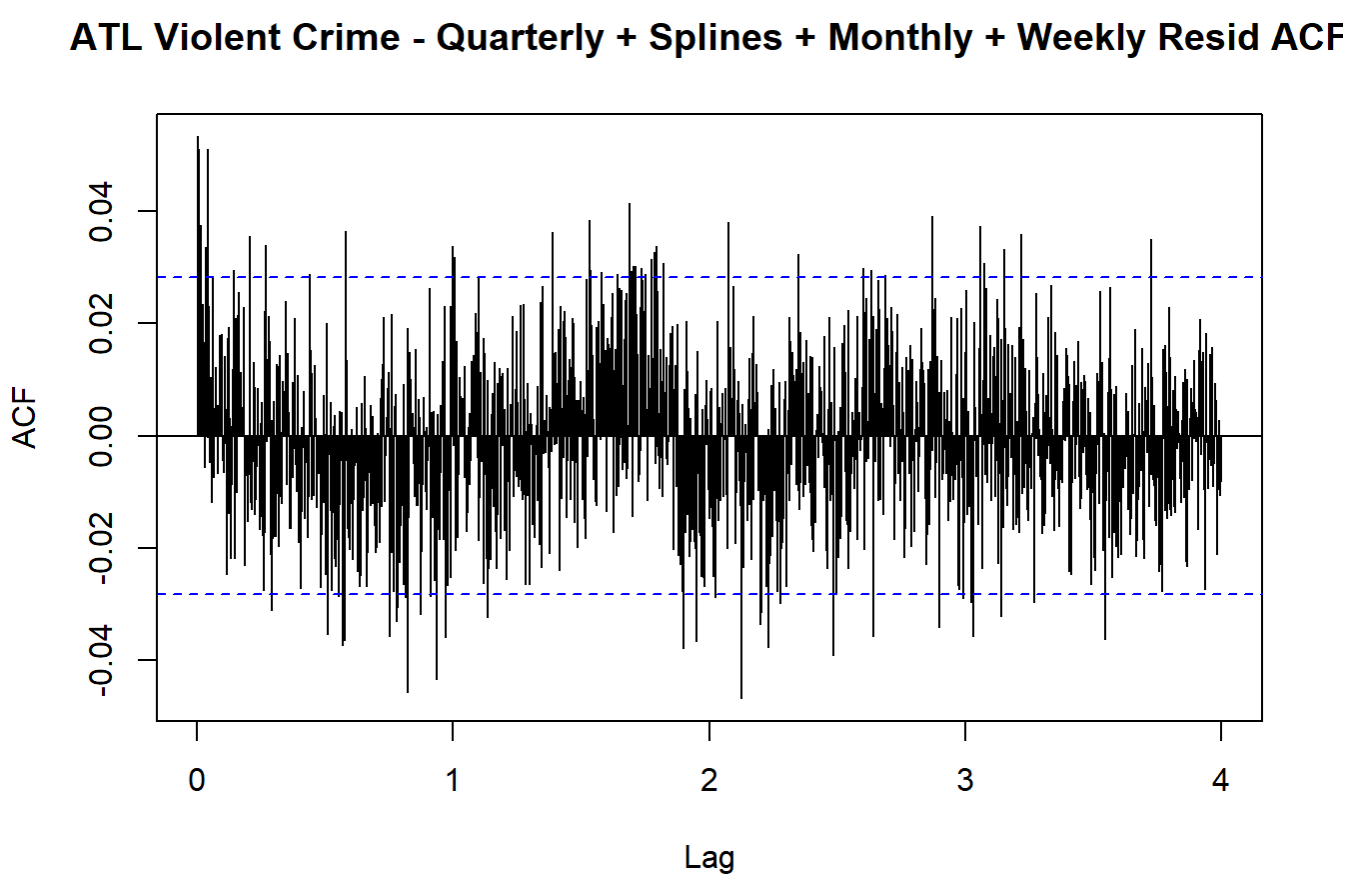

```
###TREND + SEASONALITY - SPLINES + QUARTERLY + MONTHLY + WEEKLY SEASONALITY
atl_violent$Date <- as.Date(atl_violent$Date)
year <- as.factor(format(atl_violent$Date,'%Y'))
month <- as.factor(format(atl_violent$Date, '%b'))
week <- as.factor(weekdays(atl_violent$Date))
quarterly <- as.factor(quarter(atl_violent$Date))

gam.fit.seastr.1 = gam(atl_v.tr~s(time.pts)+quarterly+month+week-1)
summary(gam.fit.seastr.1)
```

```
##
## Family: gaussian
## Link function: identity
##
## Formula:
## atl_v.tr ~ s(time.pts) + quarterly + month + week - 1
##
## Parametric coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## quarterly1     2.34968    0.01859 126.377  < 2e-16 ***
## quarterly2     3.30902    0.03369  98.231  < 2e-16 ***
## quarterly3     2.63292    0.01867 141.007  < 2e-16 ***
## quarterly4     2.52024    0.01869 134.808  < 2e-16 ***
## monthAug       0.85874    0.02307  37.223  < 2e-16 ***
## monthDec       0.75438    0.02308  32.684  < 2e-16 ***
## monthFeb       0.69008    0.02317  29.787  < 2e-16 ***
## monthJan       0.81635    0.02240  36.441  < 2e-16 ***
## monthJul       0.89659    0.02305  38.899  < 2e-16 ***
## monthJun       0.17898    0.03905   4.583 4.70e-06 ***
## monthMar       0.84325    0.02297  36.706  < 2e-16 ***
## monthMay       0.17952    0.03873   4.635 3.66e-06 ***
## monthNov       0.82769    0.02333  35.482  < 2e-16 ***
## monthOct       0.93818    0.02306  40.686  < 2e-16 ***
## monthSep       0.87759    0.02334  37.599  < 2e-16 ***
## weekMonday    -0.01313    0.02942  -0.446   0.6554
## weekSaturday   0.20452    0.02942   6.952 4.08e-12 ***
## weekSunday     0.16680    0.02942   5.670 1.51e-08 ***
## weekThursday  -0.13013    0.02941  -4.425 9.85e-06 ***
## weekTuesday   -0.04243    0.02942  -1.442   0.1493
## weekWednesday -0.06146    0.02942  -2.089   0.0367 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Approximate significance of smooth terms:
##               edf Ref.df     F p-value
## s(time.pts) 8.598  8.954 179.1  <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Rank: 27/30
## R-sq.(adj) =  0.311   Deviance explained = 97.5%
## GCV = 0.29893  Scale est. = 0.29728   n = 4810
```

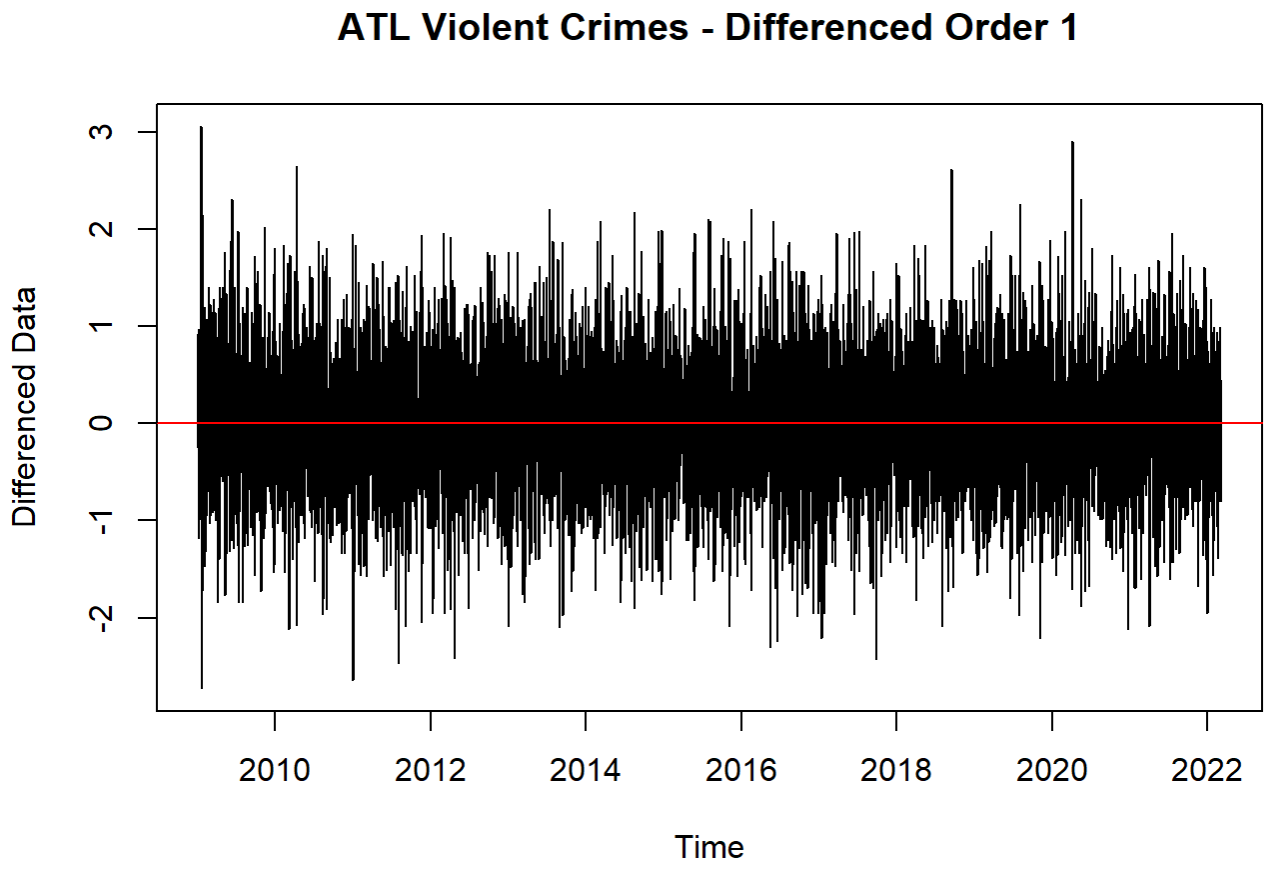
```
dif.fit.seastr.1 = ts(atl_v.tr - fitted(gam.fit.seastr.1),start=2009,frequency=365)

ts.plot(dif.fit.seastr.1, ylab = "Residuals", main = "ATL Violent Crime - Quarterly + Splines +
 Monthly + weekly")
abline(h=0, col='red')
```



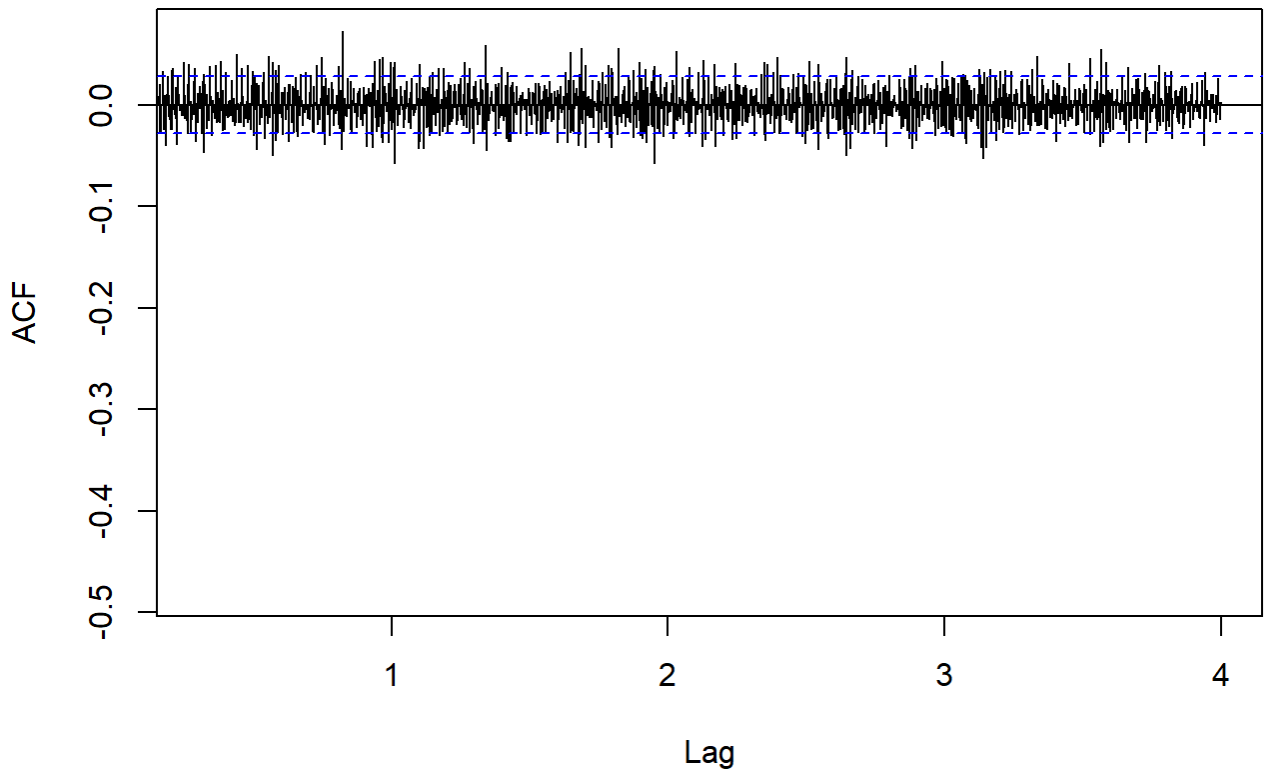

```
acf(dif.fit.seastr.1, lag.max = 365 * 4, main = "ATL Violent Crime - Quarterly + Splines + Month
ly + Weekly Resid ACF")
```




```
###DIFFERENCED DATA - ORDER 1 - TRANSFORMED
diff.atl_v = diff(atl_v.tr)
ts.plot(diff.atl_v, ylab = "Differenced Data", main = "ATL Violent Crimes - Differenced Order 1"
)
abline(h=0, col='red')
```

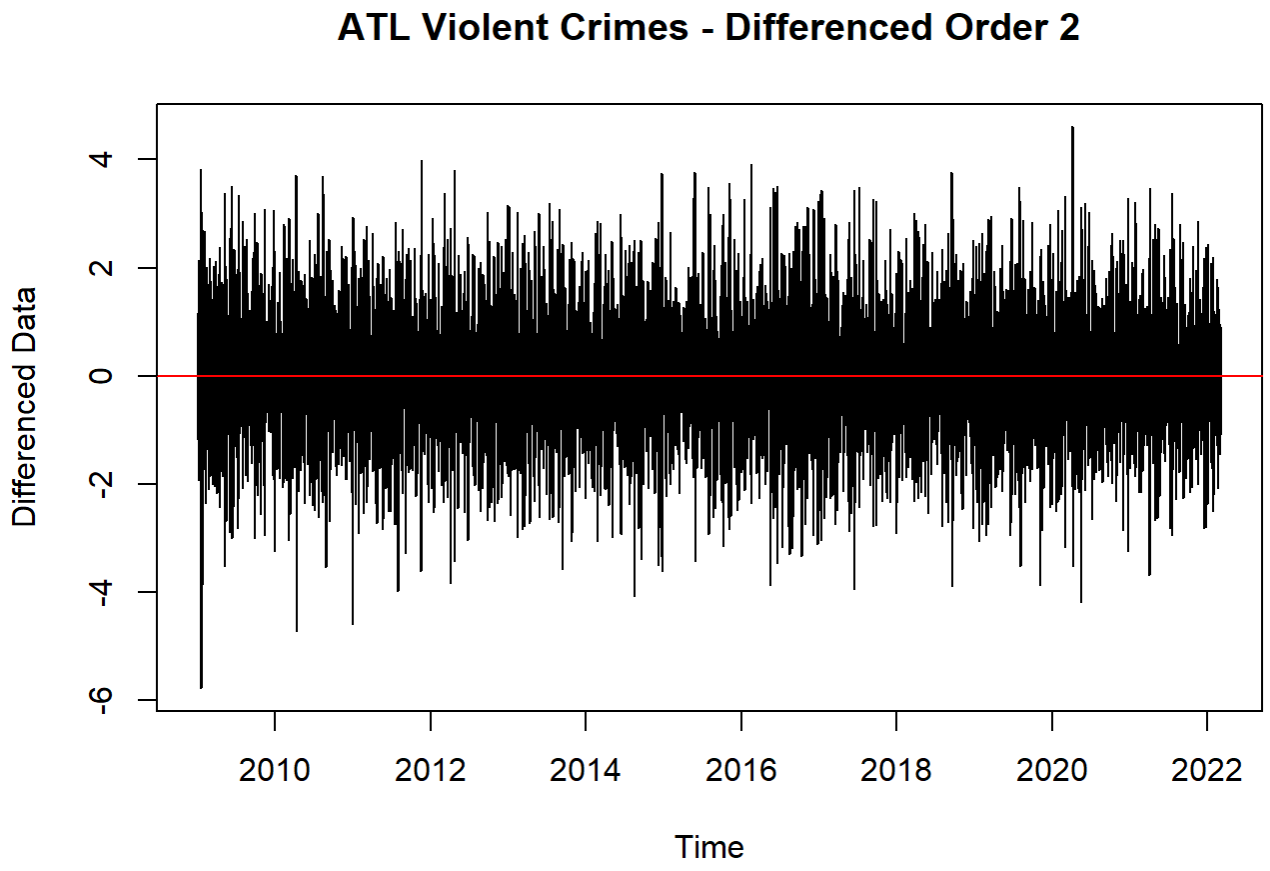



```
acf(diff.atl_v, xlim=c(0.3,4), lag.max = 365 * 4, main = "ATL Violent Crimes - Differenced Order
1 ACF")
```

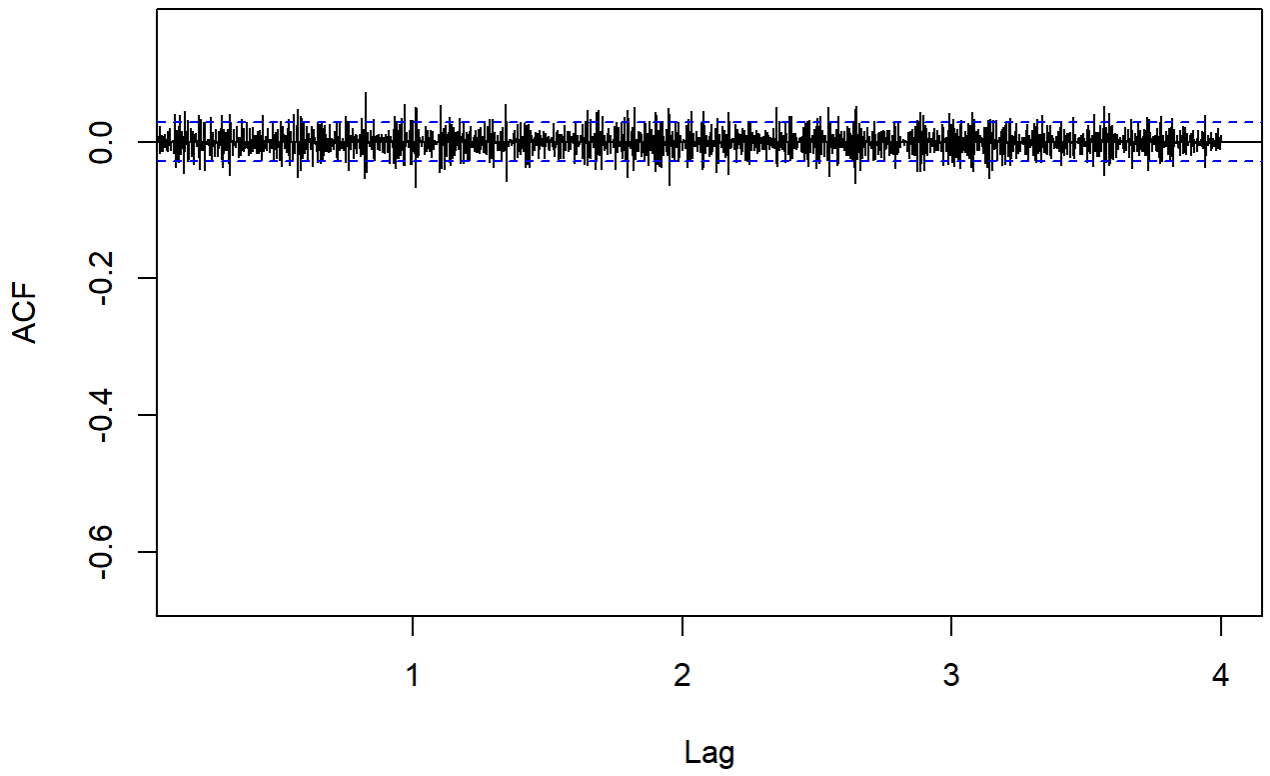



```
### SEE HW2 - Q2 FOR EXPLANATION
```

```
###DIFFERENCED DATA - ORDER 2 - TRANSFORMED
diff.atl_v = diff(atl_v.tr, differences = 2)
ts.plot(diff.atl_v, ylab = "Differenced Data", main = "ATL Violent Crimes - Differenced Order 2"
)
abline(h=0, col='red')
```

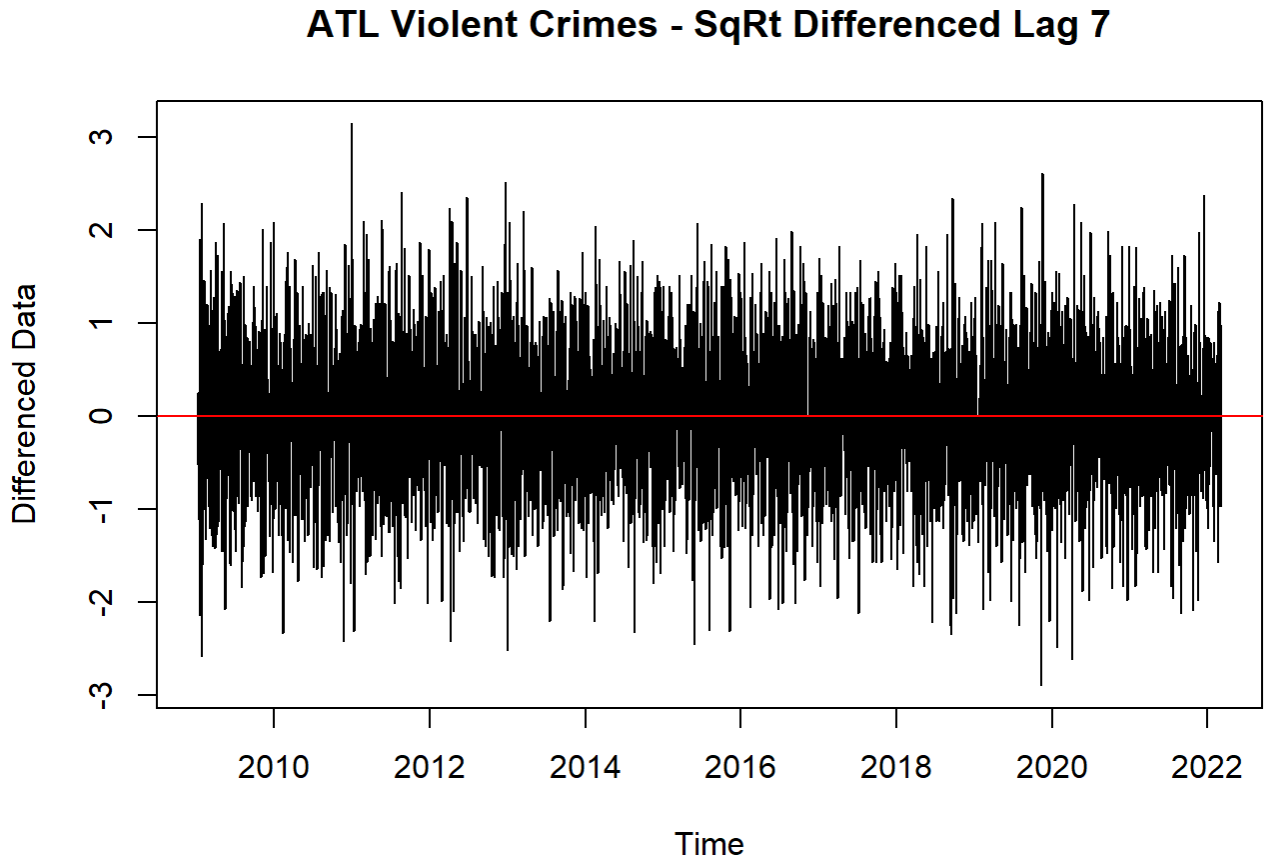



```
acf(diff.atl_v, xlim=c(0.2,4), lag.max = 365 * 4, main = "ATL Violent Crimes - Differenced Order
2 ACF")
```

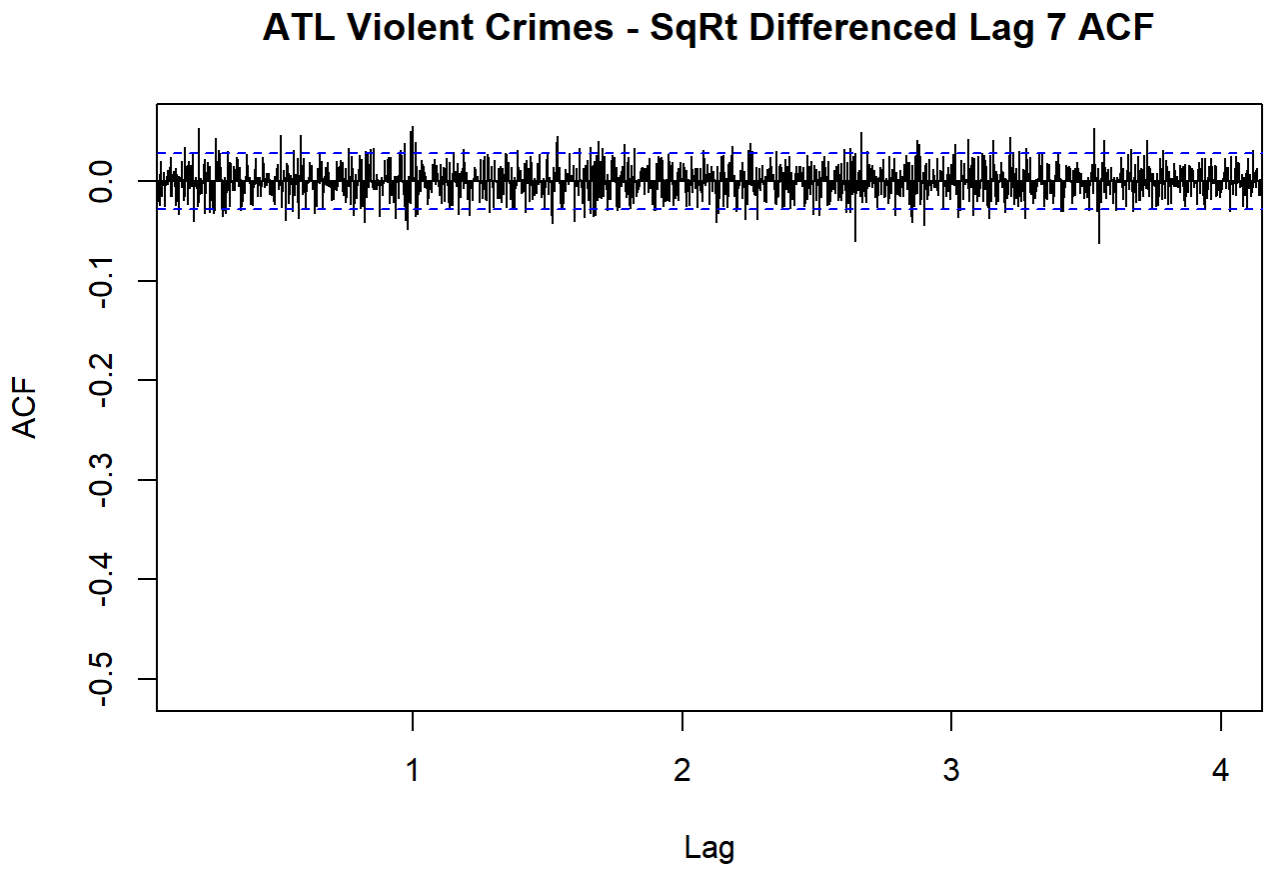



```
### SEE HW2 - Q2 FOR EXPLANATION
```

```
###DIFFERENCED DATA - TRANSFORMED - LAG 7
diff.atl_v = diff(atl_v.tr, 7)
ts.plot(diff.atl_v, ylab = "Differenced Data", main = "ATL Violent Crimes - SqRt Differenced Lag
7")
abline(h=0, col='red')
```

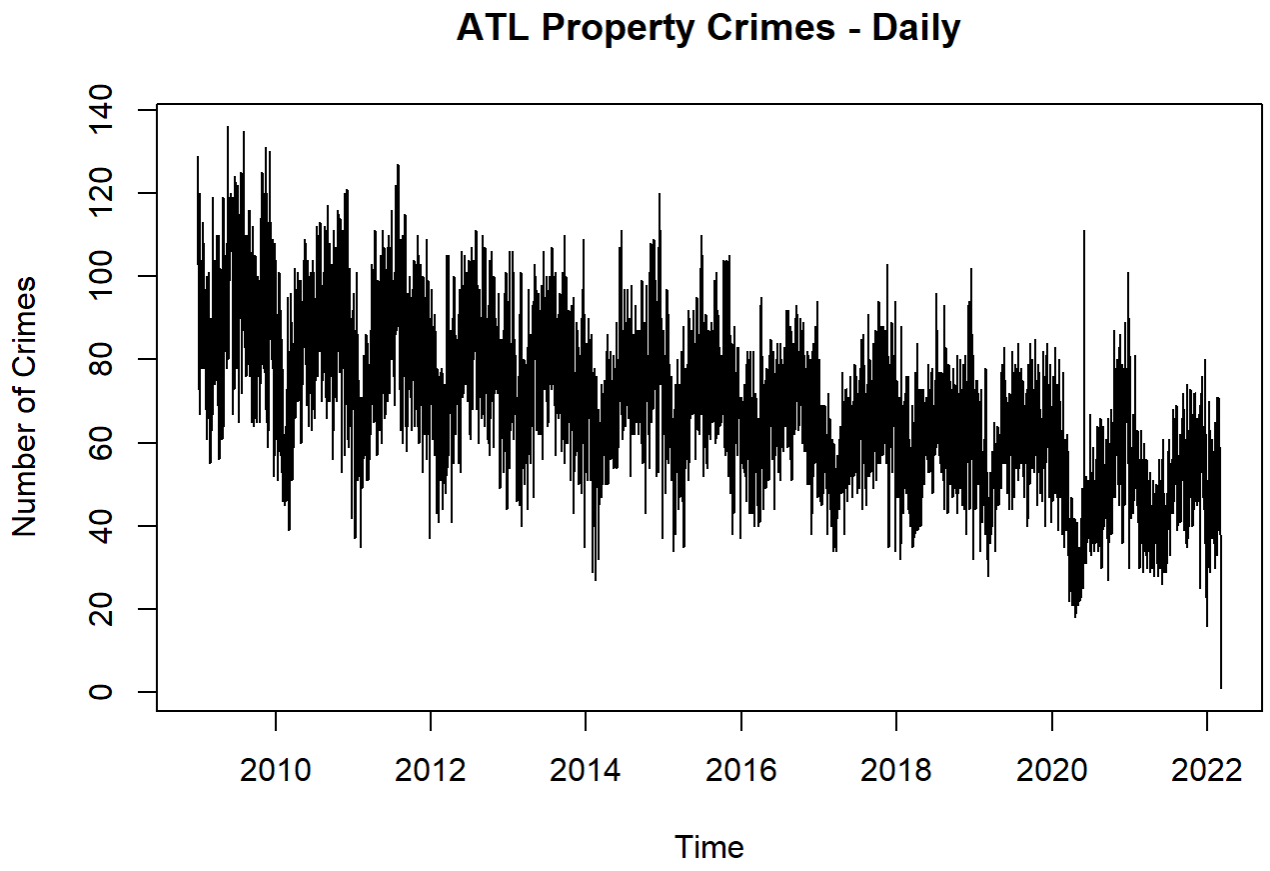



```
acf(diff.atl_v, xlim=c(0.2,4),lag.max = 365 * 5, main = "ATL Violent Crimes - SqRt Differenced L
ag 7 ACF")
```

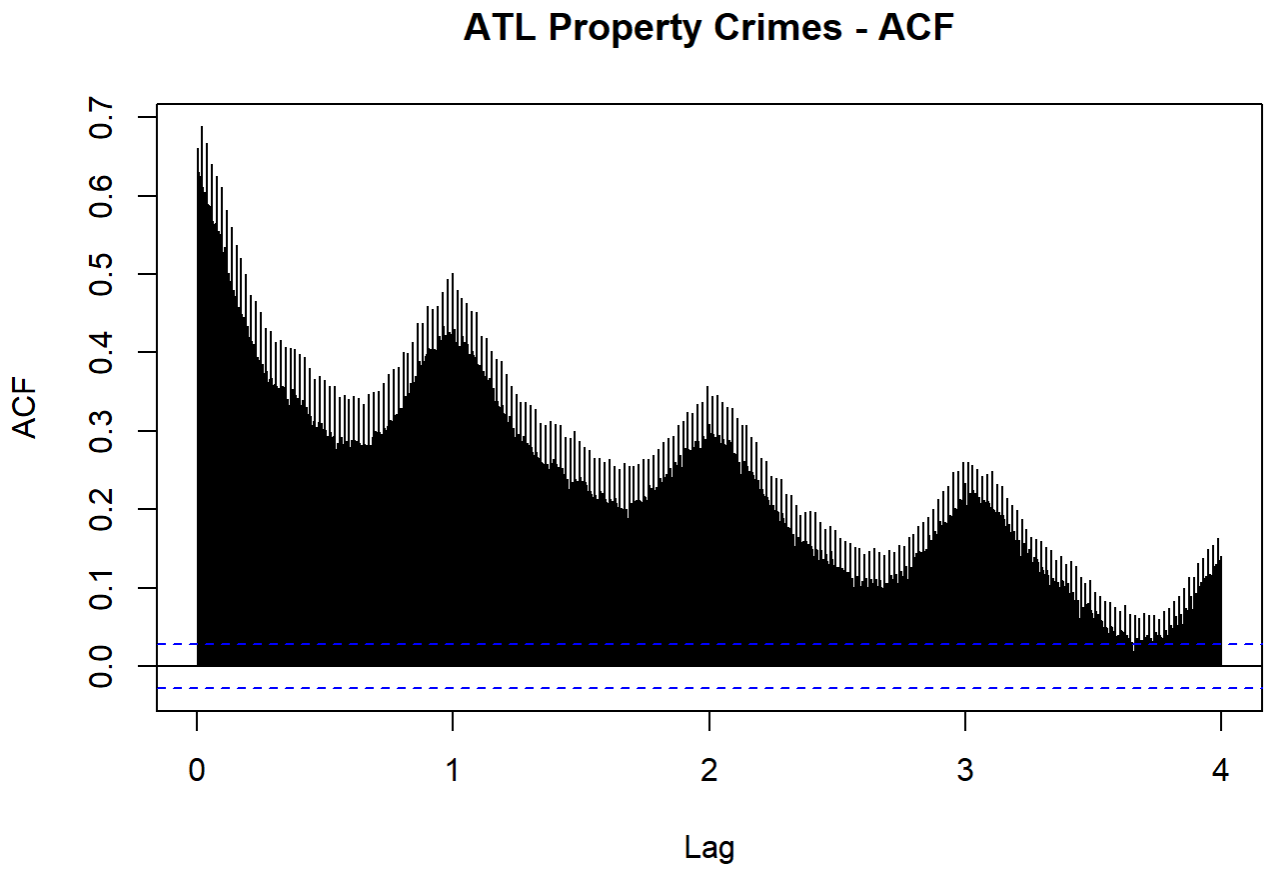


ATL Property Crime

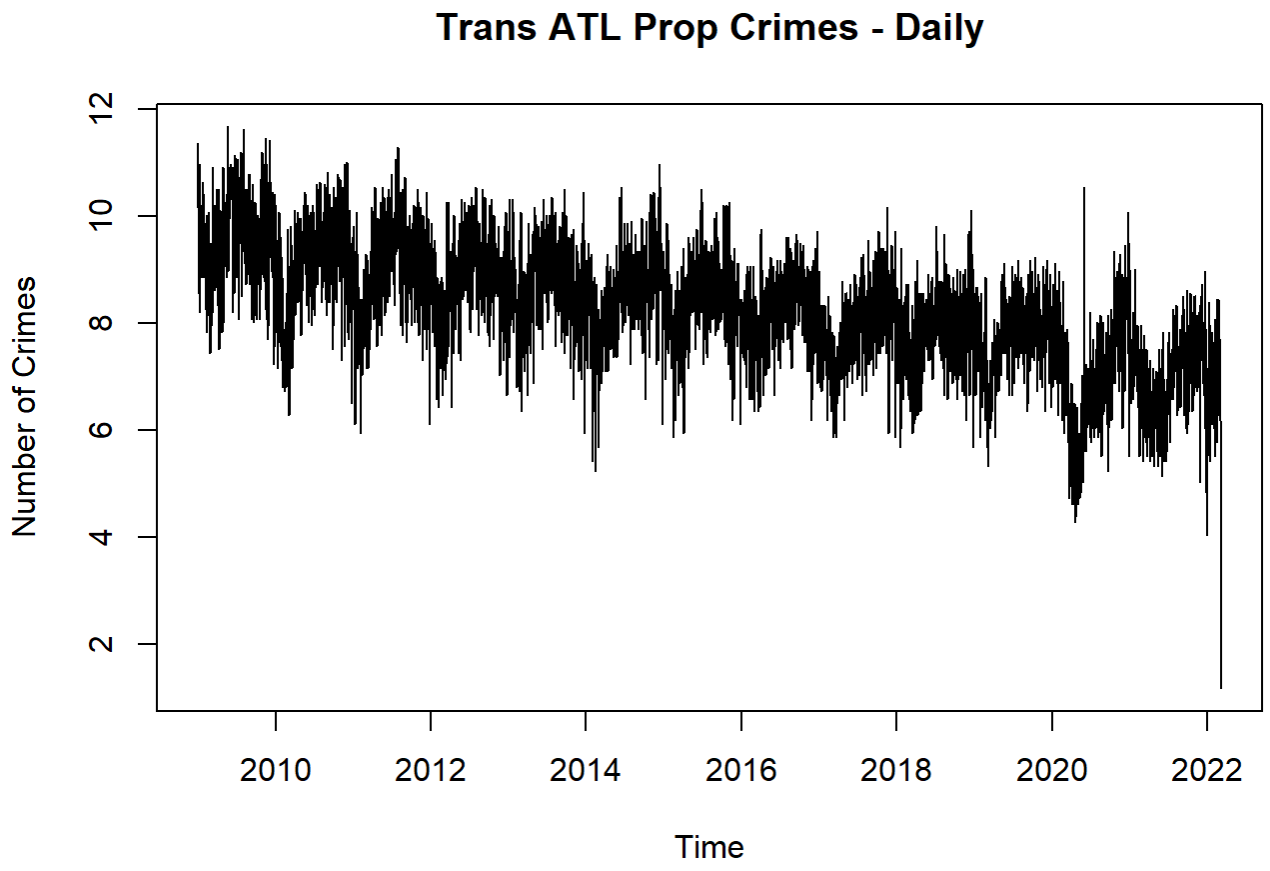
```
###EDA - Time Series / ACF
ts.plot(atl_p,ylab="Number of Crimes",main="ATL Property Crimes - Daily")
```



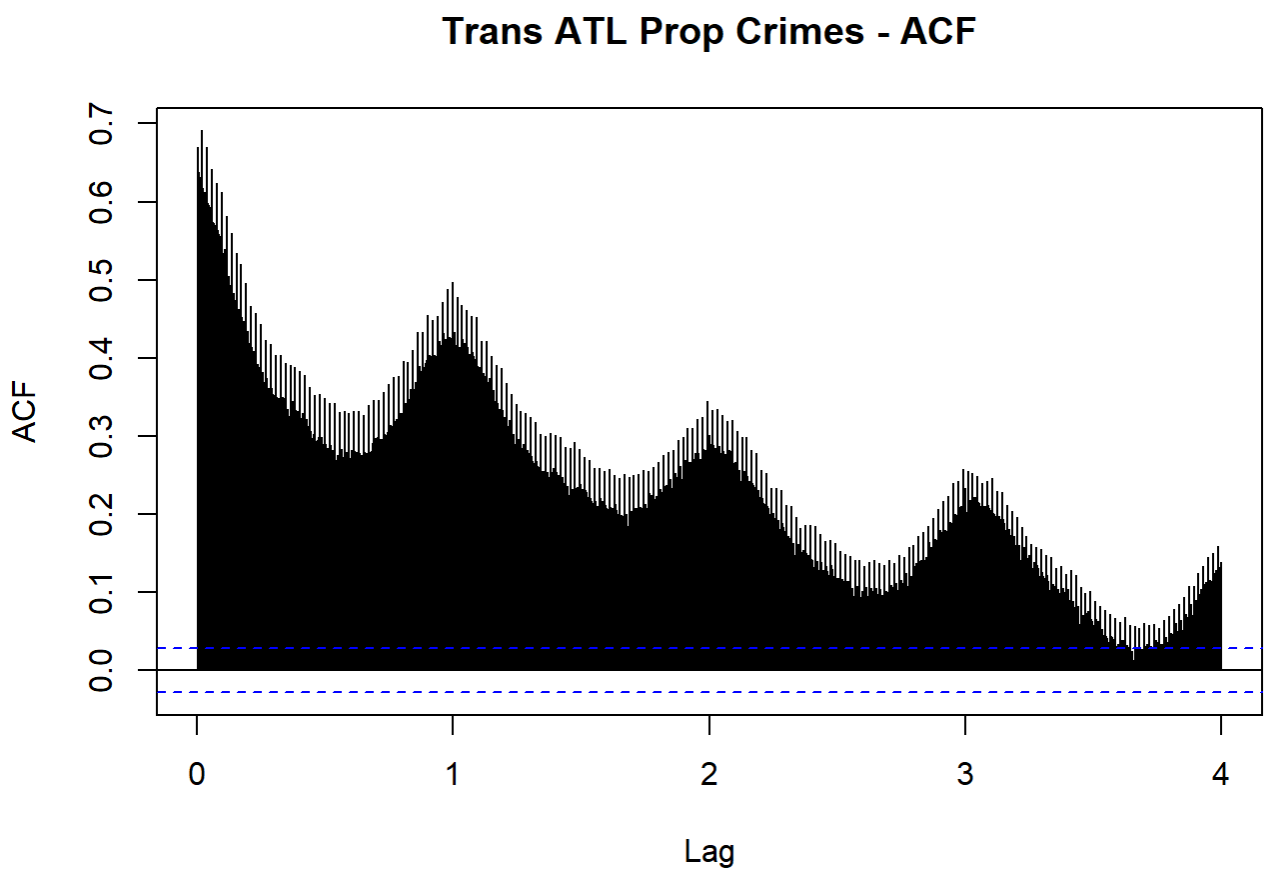

```
acf(atl_p,lag.max=365*4,main="ATL Property Crimes - ACF")
```



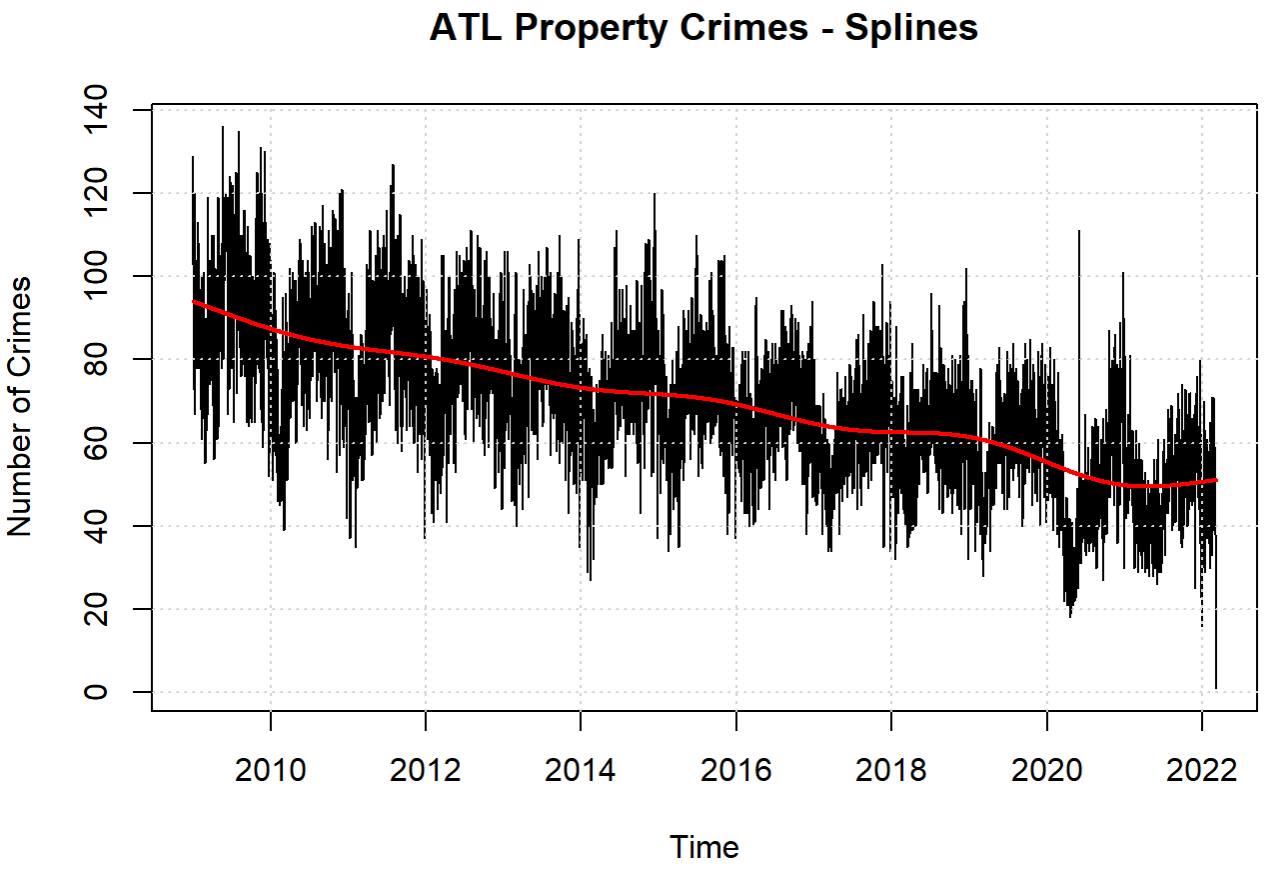

```
ts.plot(atl_p.tr,ylab="Number of Crimes",main="Trans ATL Prop Crimes - Daily")
```



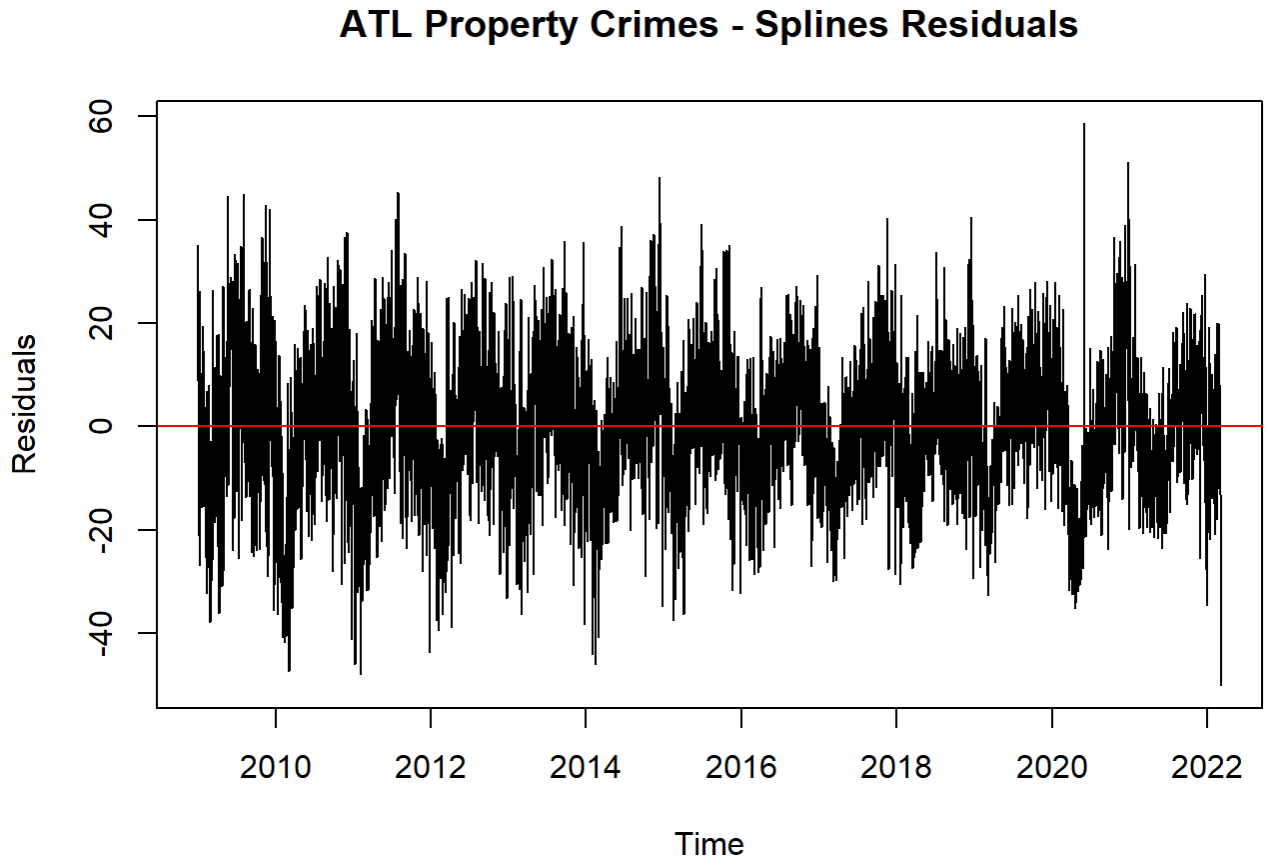

```
acf(atl_p.tr,lag.max=365*4,main="Trans ATL Prop Crimes - ACF")
```



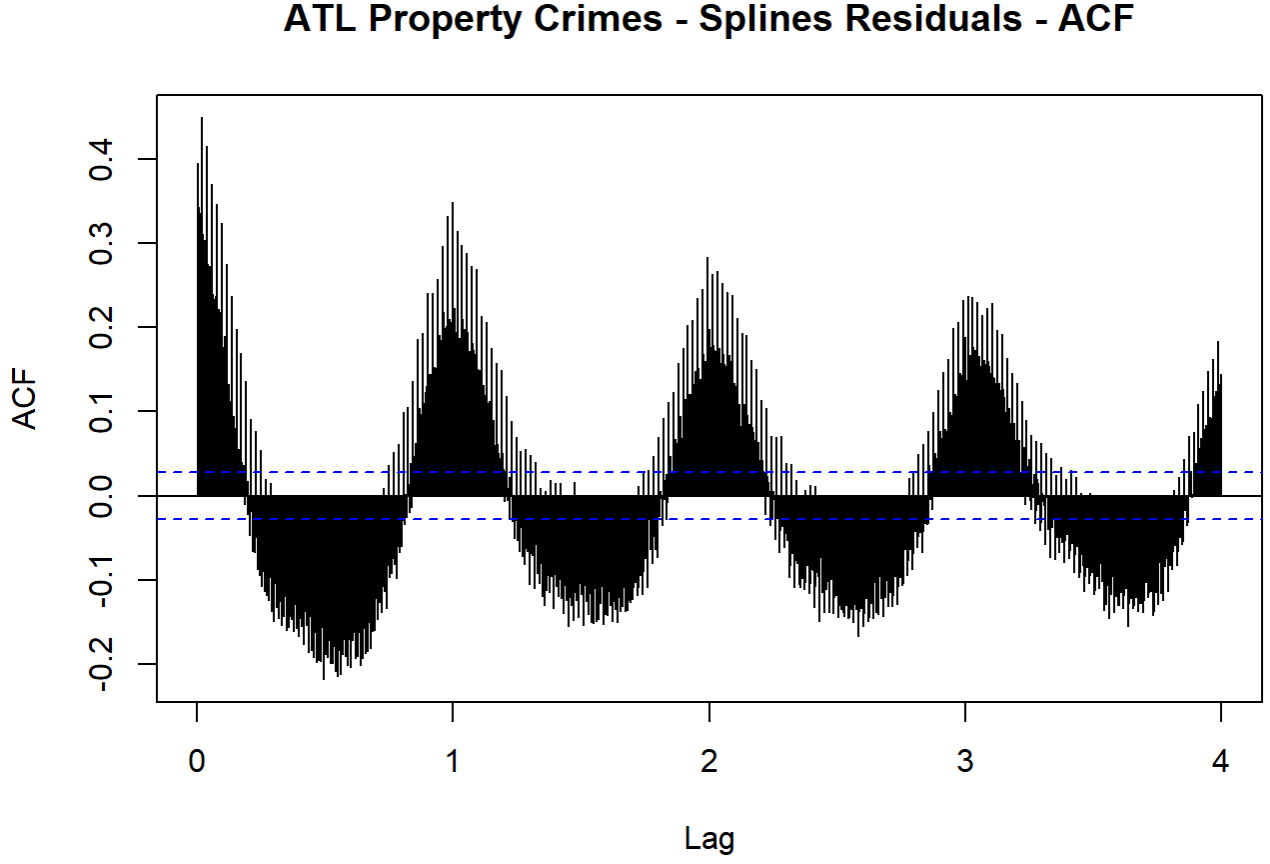

```
###TREND ESTIMATION - SPLINES
# Convert X-Axis to 0-1 Scale
time.pts = c(1:length(atl_p))
time.pts = c(time.pts - min(time.pts))/max(time.pts)

#Splines Trend Estimation
gam.fit.3 = gam(atl_p~s(time.pts))
atl_p.fit.gam = ts(fitted(gam.fit.3),start=2009,frequency=365)

##Is there a trend?
ts.plot(atl_p,ylab="Number of Crimes", main = "ATL Property Crimes - Splines")
grid()
lines(atl_p.fit.gam,lwd=2,col="red")
```



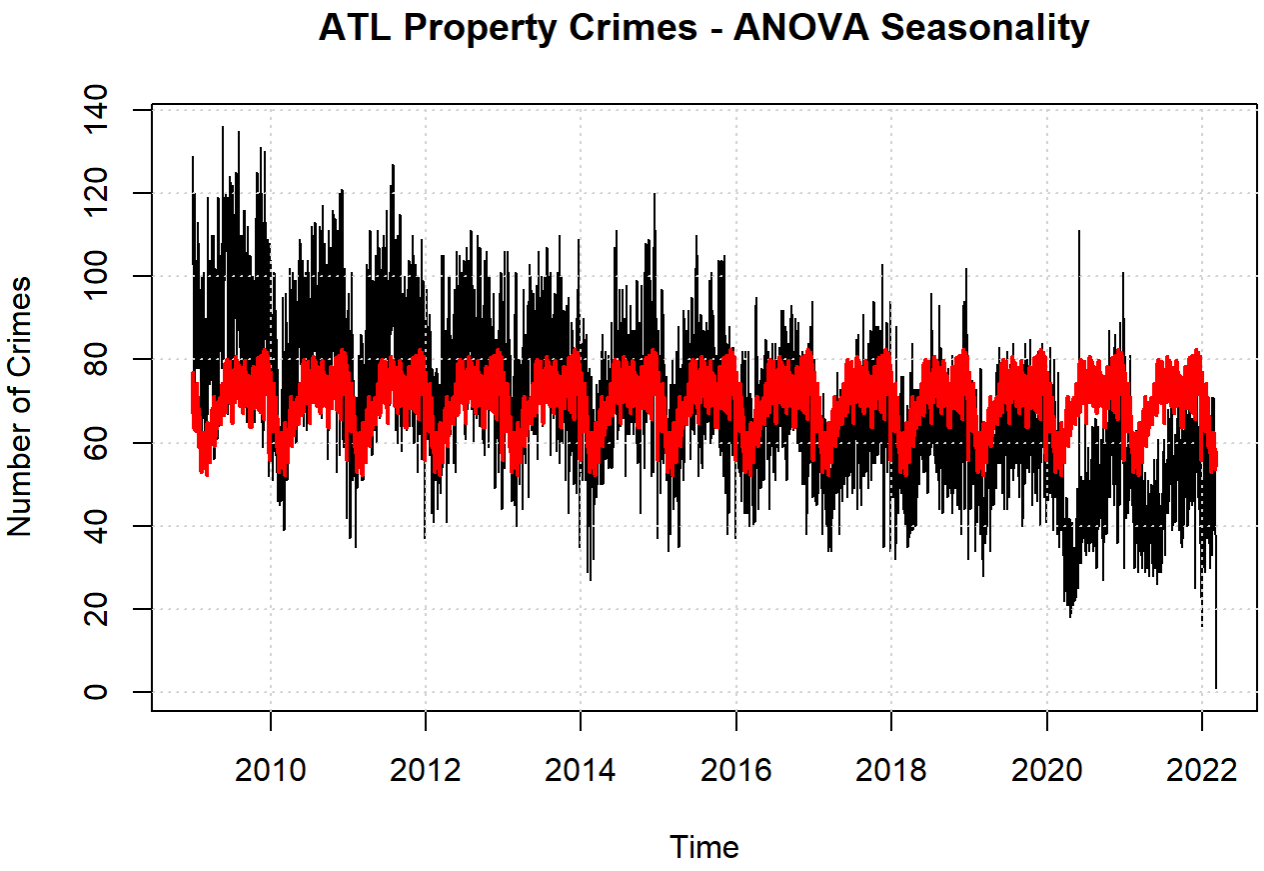

```
# Splines Residuals + Residuals ACF

dif.fit.gam4 = atl_p - atl_p.fit.gam
ts.plot(dif.fit.gam4, ylab = "Residuals", main = "ATL Property Crimes - Splines Residuals")
abline(h=0, col='red')
```



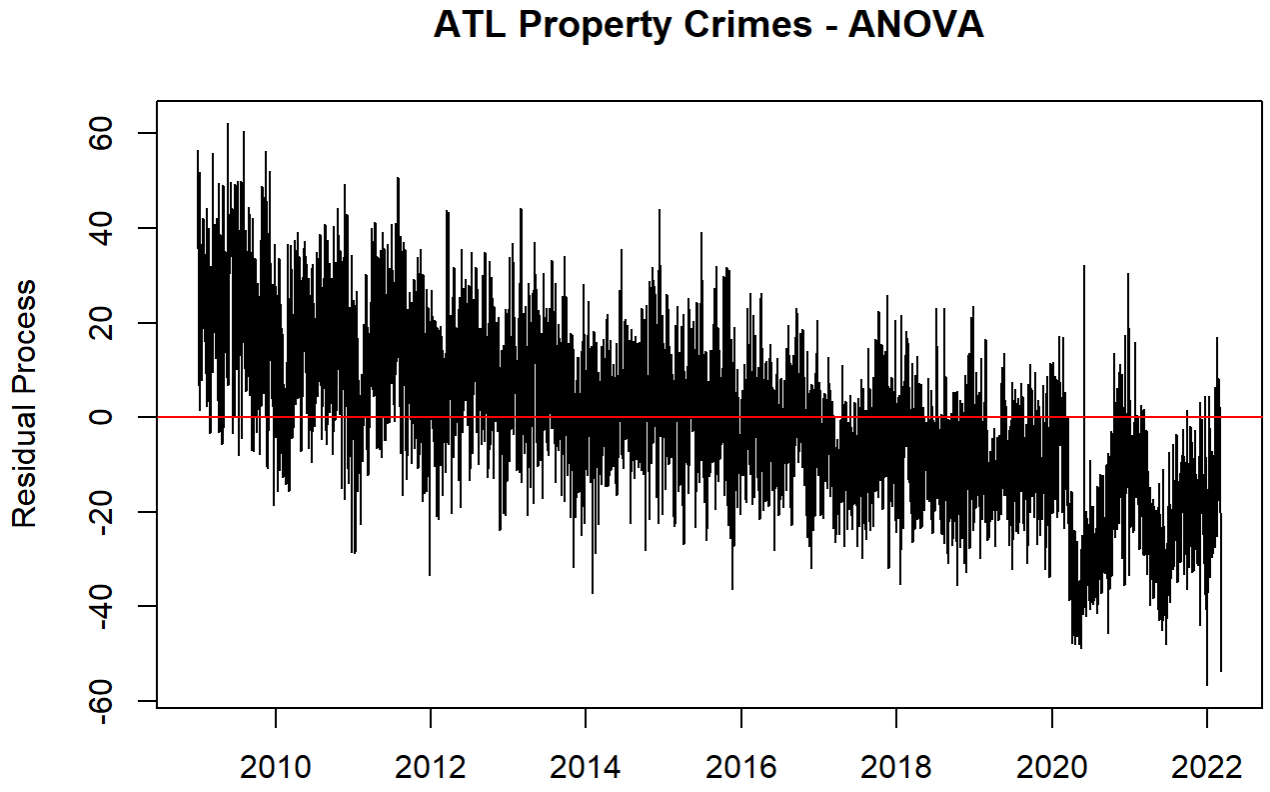

```
acf(dif.fit.gam4, lag.max = 365 * 4, main = "ATL Property Crimes - Splines Residuals - ACF")
```



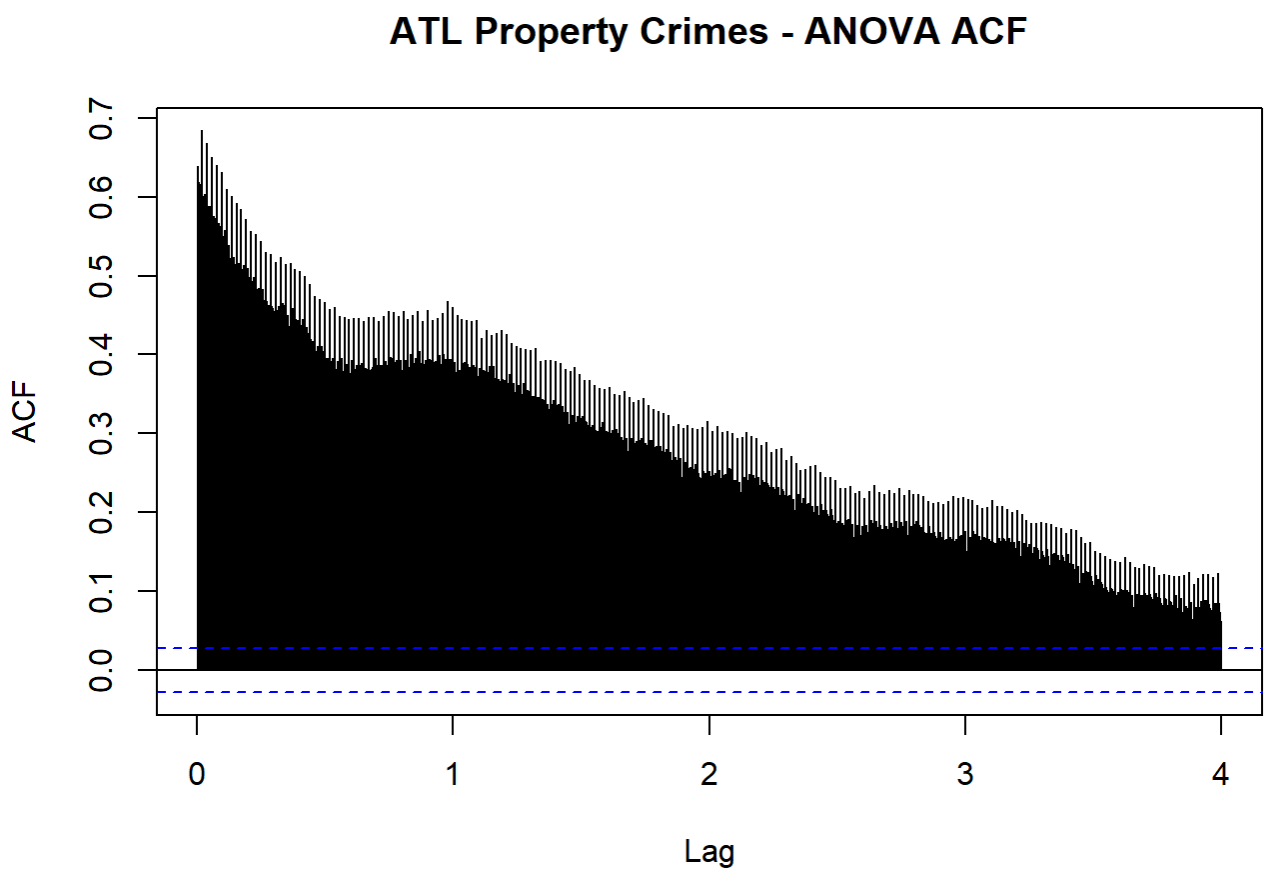

```
###SEASONALITY ANALYSIS - ANOVA

## Estimate seasonality using ANOVA approach
model.anova2 = dynlm(atl_p~season(atl_p))
#summary(model.anova)

## Plot
ts.plot(atl_p,ylab="Number of Crimes", main = "ATL Property Crimes - ANOVA Seasonality")
grid()
lines(fitted(model.anova2),lwd=2,col="red")
```




```
###Seasonality ANOVA- Residuals + Resid ACF
resid.anova2 <- residuals(model.anova2)

ts.plot(resid.anova2, xlab = "", ylab = "Residual Process", main = "ATL Property Crimes - ANOVA"
)
abline(h=0, col='red')
```




```
acf(resid.anova2, lag.max = 365 * 4, main = "ATL Property Crimes - ANOVA ACF")
```




```
###SEASONALITY ANALYSIS - HARMONIC
###SUMMARY
harmonic.3 = dynlm(atl_p~harmon(atl_p))
summary(harmonic.3)
```

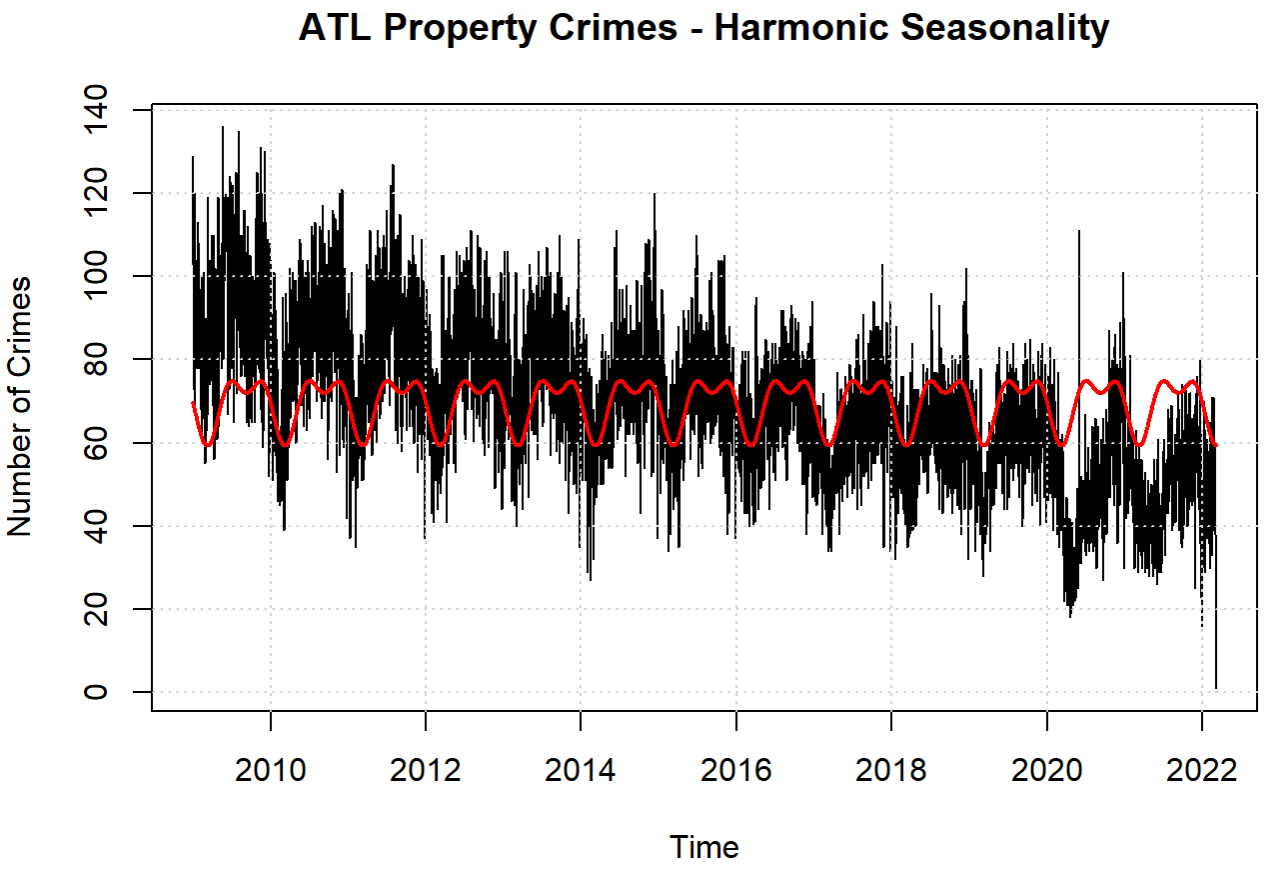
```
##
## Time series regression with "ts" data:
## Start = 2009(1), End = 2022(66)
##
## Call:
## dynlm(formula = atl_p ~ harmon(atl_p))
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -62.221 -11.879  -0.908  11.489  68.240
##
## Coefficients:
##                   Estimate Std. Error t value Pr(>|t|)
## (Intercept)       69.6073     0.2563 271.628  < 2e-16 ***
## harmon(atl_p)cos  -2.6123     0.3615  -7.225 5.78e-13 ***
## harmon(atl_p)sin  -5.8314     0.3632 -16.053  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 17.77 on 4808 degrees of freedom
## Multiple R-squared:  0.06097,    Adjusted R-squared:  0.06058
## F-statistic: 156.1 on 2 and 4808 DF,  p-value: < 2.2e-16
```

```
harmonic.4 = dynlm(atl_p~harmon(atl_p,2))
summary(harmonic.4)
```

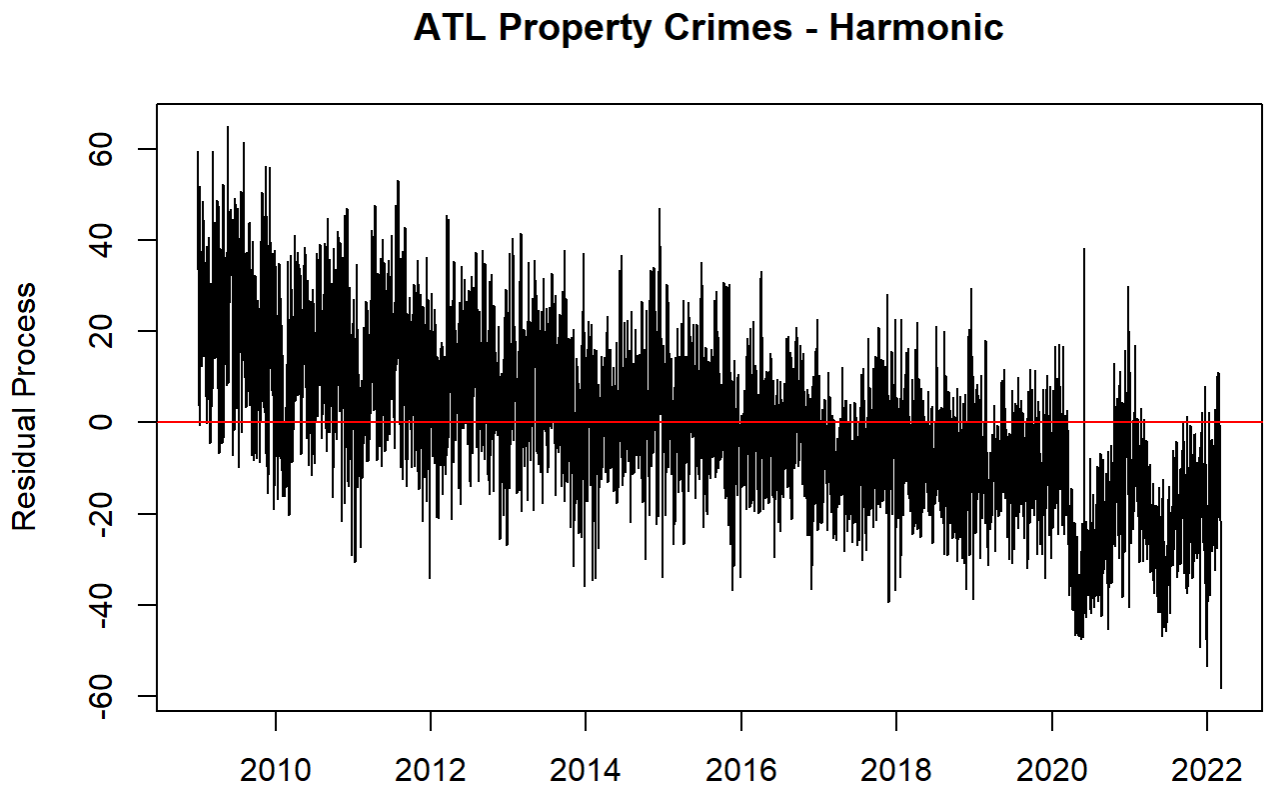
```
##
## Time series regression with "ts" data:
## Start = 2009(1), End = 2022(66)
##
## Call:
## dynlm(formula = atl_p ~ harmon(atl_p, 2))
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -58.340 -11.410  -0.762  11.398  64.982
##
## Coefficients:
##                      Estimate Std. Error t value Pr(>|t|)
## (Intercept)           69.6225     0.2532 274.998  < 2e-16 ***
## harmon(atl_p, 2)cos1  -2.5976     0.3572  -7.272 4.11e-13 ***
## harmon(atl_p, 2)cos2   2.6942     0.3585   7.516 6.72e-14 ***
## harmon(atl_p, 2)sin1  -5.8002     0.3589 -16.163  < 2e-16 ***
## harmon(atl_p, 2)sin2  -2.8808     0.3575  -8.057 9.74e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 17.56 on 4806 degrees of freedom
## Multiple R-squared:  0.08404,    Adjusted R-squared:  0.08327
## F-statistic: 110.2 on 4 and 4806 DF,  p-value: < 2.2e-16
```

```
###all 4 harmonic values statistically significant - use more complex model
```

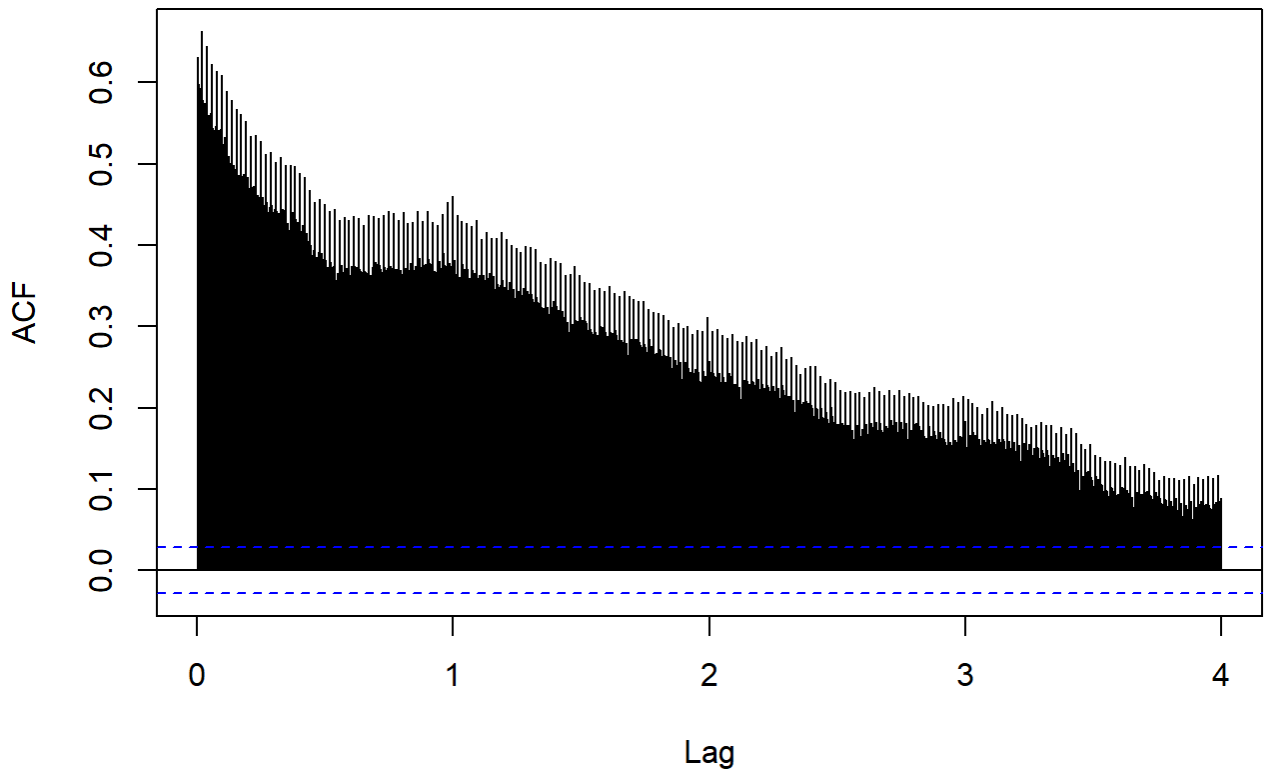
```
###Harmonic Plot
## Plot
ts.plot(atl_p,ylab="Number of Crimes", main = "ATL Property Crimes - Harmonic Seasonality")
grid()
lines(fitted(harmonic.4),lwd=2,col="red")
```




```
###Seasonality Harmonic - Residuals + Resid ACF

resid.harmonic4 <- residuals(harmonic.4)

ts.plot(resid.harmonic4, xlab = "", ylab = "Residual Process", main = "ATL Property Crimes - Har
monic")
abline(h=0, col='red')
```



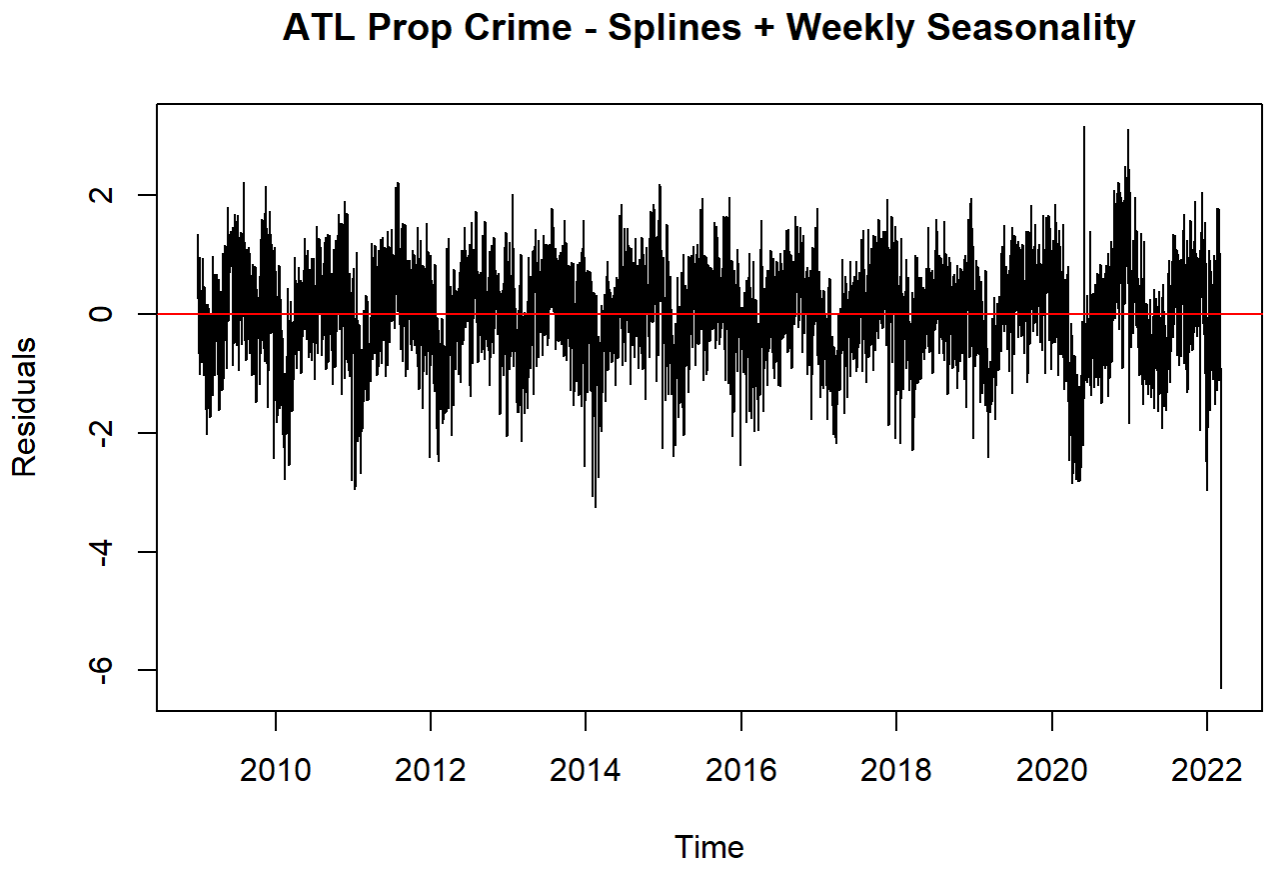

```
acf(resid.harmonic4, lag.max = 365 * 4, main = "ATL Property Crimes - Harmonic ACF")
```



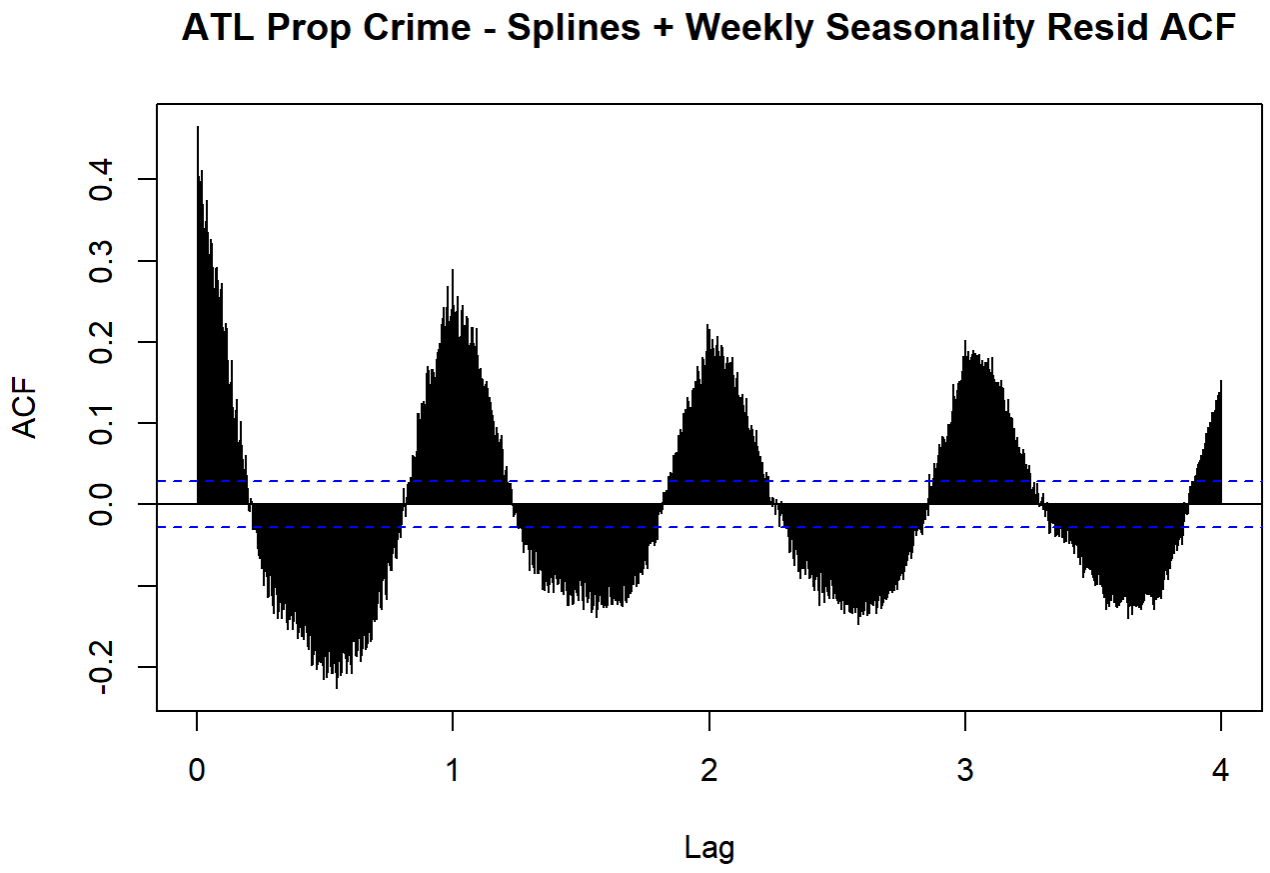

```
###TREND + SEASONALITY - SPLINES + WEEKLY SEASONALITY - TRANSFORMED
atl_prop$Date <- as.Date(atl_prop$Date)
year <- as.factor(format(atl_prop$Date,'%Y'))
month <- as.factor(format(atl_prop$Date, '%b'))
week <- as.factor(weekdays(atl_prop$Date))

gam.fit.seastr = gam(atl_p.tr~s(time.pts)+week-1)
summary(gam.fit.seastr)
```

```
##
## Family: gaussian
## Link function: identity
##
## Formula:
## atl_p.tr ~ s(time.pts) + week - 1
##
## Parametric coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## weekFriday     8.62724    0.03082   280.0   <2e-16 ***
## weekMonday     8.34488    0.03084   270.6   <2e-16 ***
## weekSaturday   8.50804    0.03084   275.9   <2e-16 ***
## weekSunday     7.83626    0.03084   254.1   <2e-16 ***
## weekThursday   8.20206    0.03082   266.2   <2e-16 ***
## weekTuesday    8.23269    0.03084   267.0   <2e-16 ***
## weekWednesday  8.24954    0.03084   267.5   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Approximate significance of smooth terms:
##              edf Ref.df     F p-value
## s(time.pts) 8.72  8.977 452.6  <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## R-sq.(adj) =  0.481   Deviance explained = 99.1%
## GCV = 0.65544  Scale est. = 0.6533    n = 4811
```

```
dif.fit.seastr = ts(atl_p.tr - fitted(gam.fit.seastr),start=2009,frequency=365)

ts.plot(dif.fit.seastr, ylab = "Residuals", main = "ATL Prop Crime - Splines + Weekly Seasonalit
y")
abline(h=0, col='red')
```



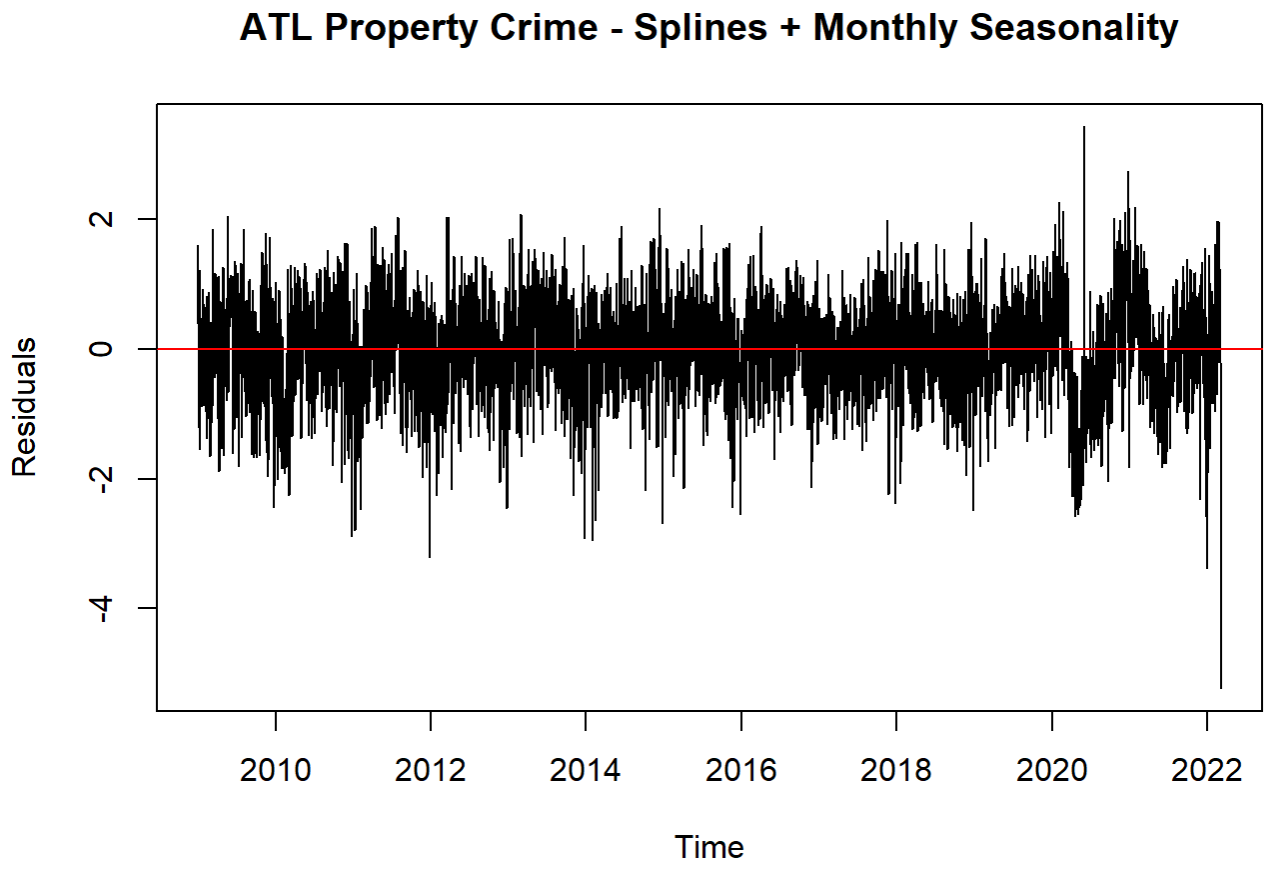

```
acf(dif.fit.seastr, lag.max = 365 * 4, main = "ATL Prop Crime - Splines + Weekly Seasonality Res
id ACF")
```



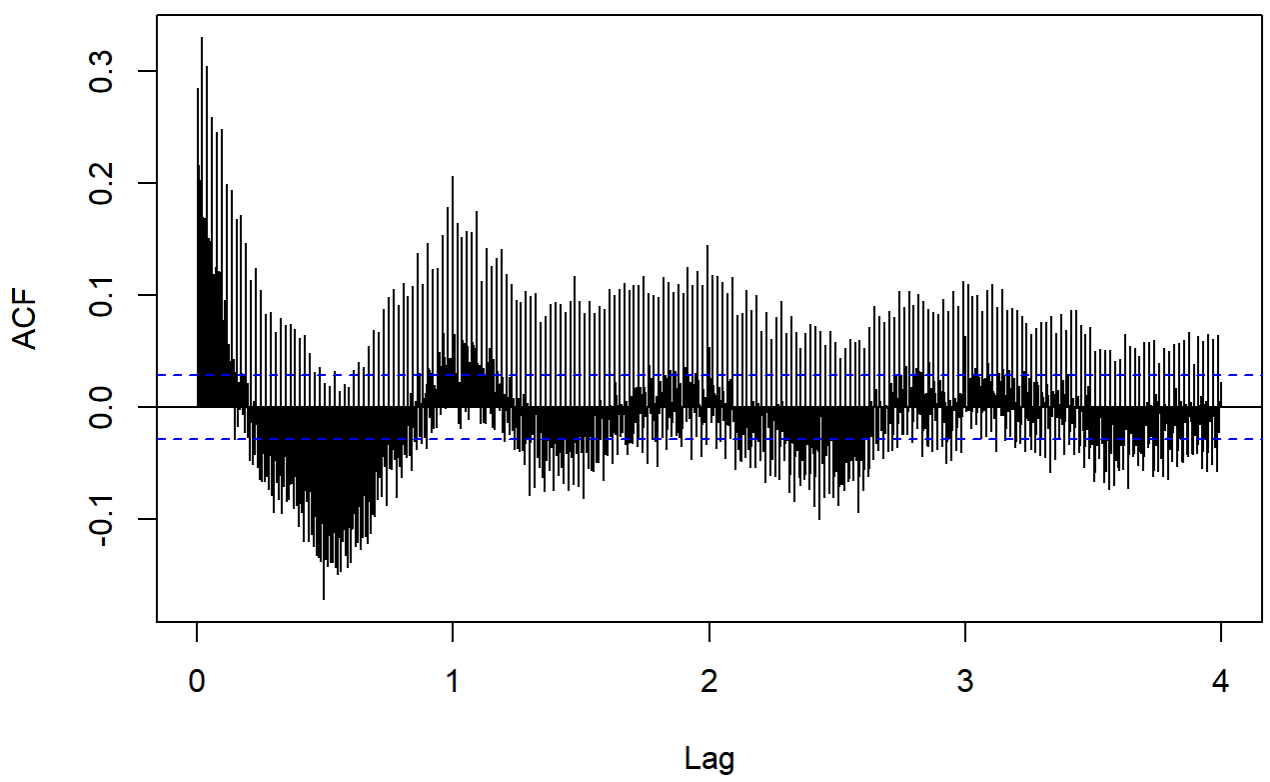

```
###TREND + SEASONALITY - SPLINES + MONTHLY SEASONALITY
atl_prop$Date <- as.Date(atl_prop$Date)
year <- as.factor(format(atl_prop$Date,'%Y'))
month <- as.factor(format(atl_prop$Date, '%b'))
week <- as.factor(weekdays(atl_prop$Date))

gam.fit.seastr = gam(atl_p.tr~s(time.pts)+month-1)
summary(gam.fit.seastr)
```

```
##
## Family: gaussian
## Link function: identity
##
## Formula:
## atl_p.tr ~ s(time.pts) + month - 1
##
## Parametric coefficients:
##          Estimate Std. Error t value Pr(>|t|)
## monthApr  7.93272    0.03858   205.6   <2e-16 ***
## monthAug  8.52900    0.03791   225.0   <2e-16 ***
## monthDec  8.62759    0.03798   227.1   <2e-16 ***
## monthFeb  7.65957    0.03839   199.5   <2e-16 ***
## monthJan  8.14560    0.03664   222.3   <2e-16 ***
## monthJul  8.59189    0.03791   226.6   <2e-16 ***
## monthJun  8.47398    0.03855   219.8   <2e-16 ***
## monthMar  7.57280    0.03777   200.5   <2e-16 ***
## monthMay  8.24899    0.03793   217.5   <2e-16 ***
## monthNov  8.57720    0.03858   222.3   <2e-16 ***
## monthOct  8.58332    0.03793   226.3   <2e-16 ***
## monthSep  8.50434    0.03854   220.6   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Approximate significance of smooth terms:
##               edf Ref.df     F p-value
## s(time.pts) 8.791  8.987 528.5  <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## R-sq.(adj) =   0.54   Deviance explained = 99.2%
## GCV = 0.58142  Scale est. = 0.57891   n = 4811
```

```
dif.fit.seastr = ts(atl_p.tr - fitted(gam.fit.seastr),start=2009,frequency=365)

ts.plot(dif.fit.seastr, ylab = "Residuals", main = "ATL Property Crime - Splines + Monthly Seaso
nality")
abline(h=0, col='red')
```



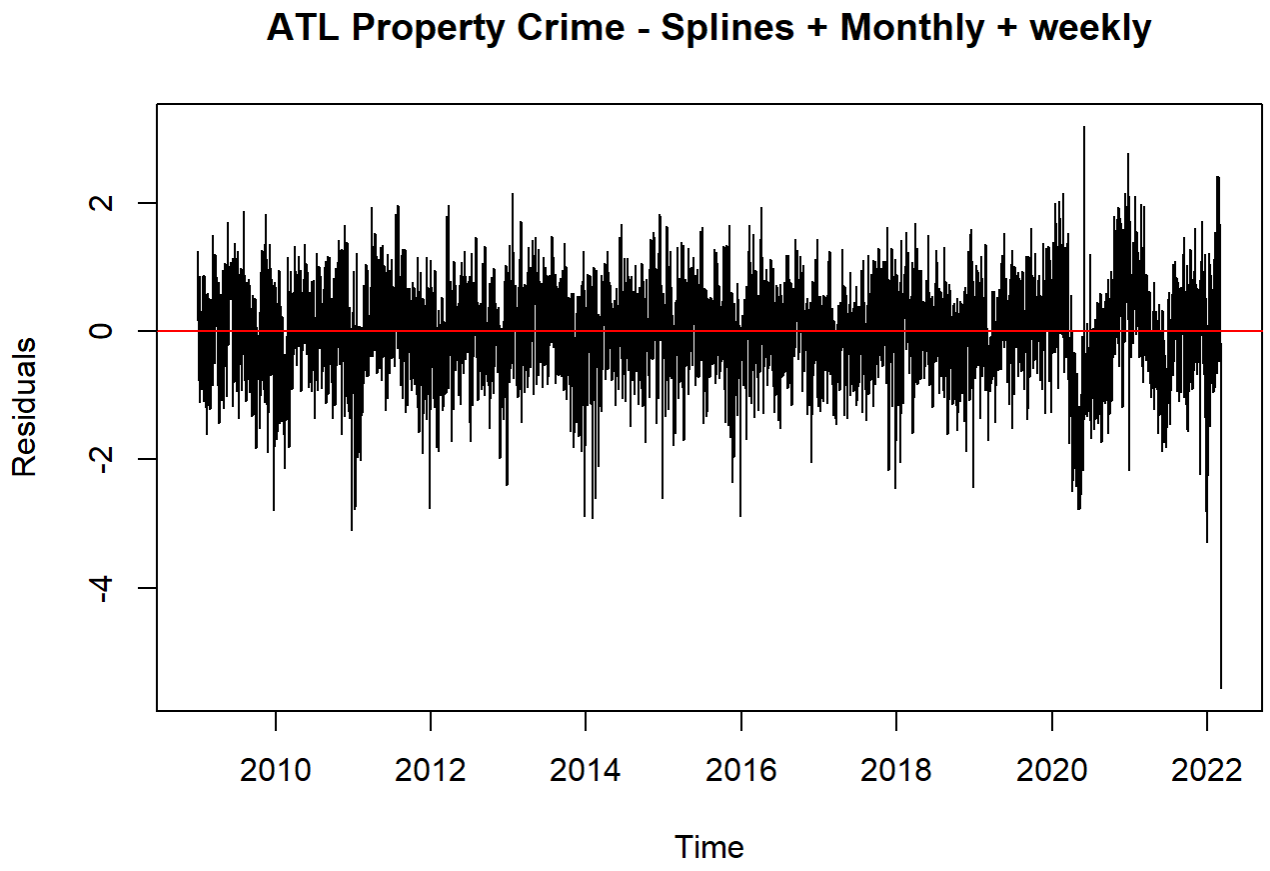

```
acf(dif.fit.seastr, lag.max = 365 * 4, main = "ATL Property Crime - Splines + Monthly Seasonalit
y Resid ACF")
```



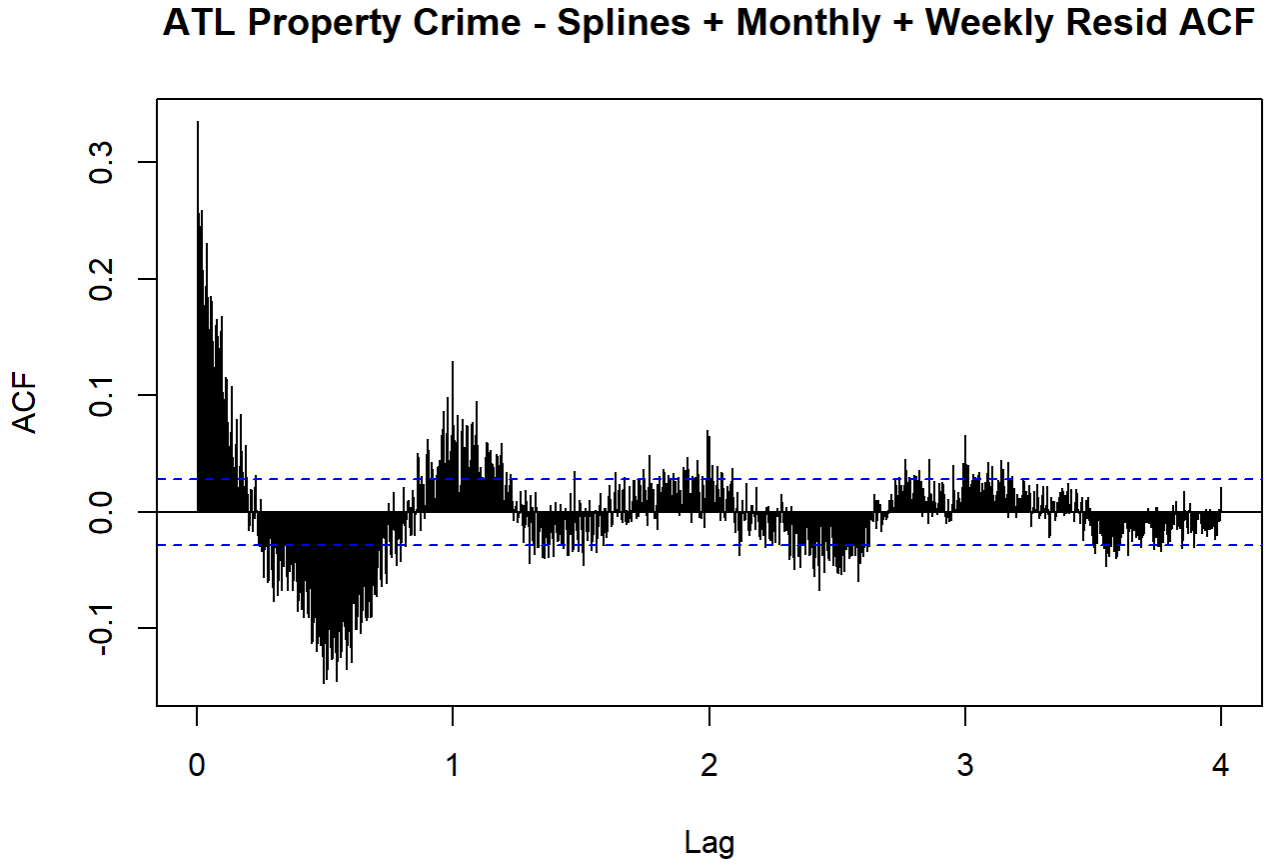

```
###TREND + SEASONALITY - SPLINES + MONTHLY + WEEKLY SEASONALITY
atl_prop$Date <- as.Date(atl_prop$Date)
year <- as.factor(format(atl_prop$Date,'%Y'))
month <- as.factor(format(atl_prop$Date, '%b'))
week <- as.factor(weekdays(atl_prop$Date))

gam.fit.seastr.1 = gam(atl_p.tr~s(time.pts)+month+week-1)
summary(gam.fit.seastr.1)
```

```
##
## Family: gaussian
## Link function: identity
##
## Formula:
## atl_p.tr ~ s(time.pts) + month + week - 1
##
## Parametric coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## monthApr       8.27338    0.04474 184.910  < 2e-16 ***
## monthAug       8.87043    0.04426 200.427  < 2e-16 ***
## monthDec       8.96936    0.04431 202.404  < 2e-16 ***
## monthFeb       8.00007    0.04463 179.257  < 2e-16 ***
## monthJan       8.48640    0.04322 196.375  < 2e-16 ***
## monthJul       8.93166    0.04417 202.201  < 2e-16 ***
## monthJun       8.81478    0.04476 196.943  < 2e-16 ***
## monthMar       7.91475    0.04413 179.351  < 2e-16 ***
## monthMay       8.59098    0.04423 194.225  < 2e-16 ***
## monthNov       8.91845    0.04474 199.333  < 2e-16 ***
## monthOct       8.92273    0.04423 201.729  < 2e-16 ***
## monthSep       8.84719    0.04476 197.669  < 2e-16 ***
## weekMonday    -0.28187    0.03907  -7.214 6.27e-13 ***
## weekSaturday  -0.11956    0.03907  -3.060  0.00222 **
## weekSunday    -0.79073    0.03907 -20.239  < 2e-16 ***
## weekThursday  -0.42542    0.03906 -10.893  < 2e-16 ***
## weekTuesday   -0.39333    0.03907 -10.067  < 2e-16 ***
## weekWednesday -0.37714    0.03907  -9.653  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Approximate significance of smooth terms:
##              edf Ref.df   F p-value
## s(time.pts) 8.81  8.989 583  <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## R-sq.(adj) =  0.583   Deviance explained = 99.3%
## GCV = 0.52766  Scale est. = 0.52472   n = 4811
```

```
dif.fit.seastr.1 = ts(atl_p.tr - fitted(gam.fit.seastr.1),start=2009,frequency=365)

ts.plot(dif.fit.seastr.1, ylab = "Residuals", main = "ATL Property Crime - Splines + Monthly + w
eekly")
abline(h=0, col='red')
```



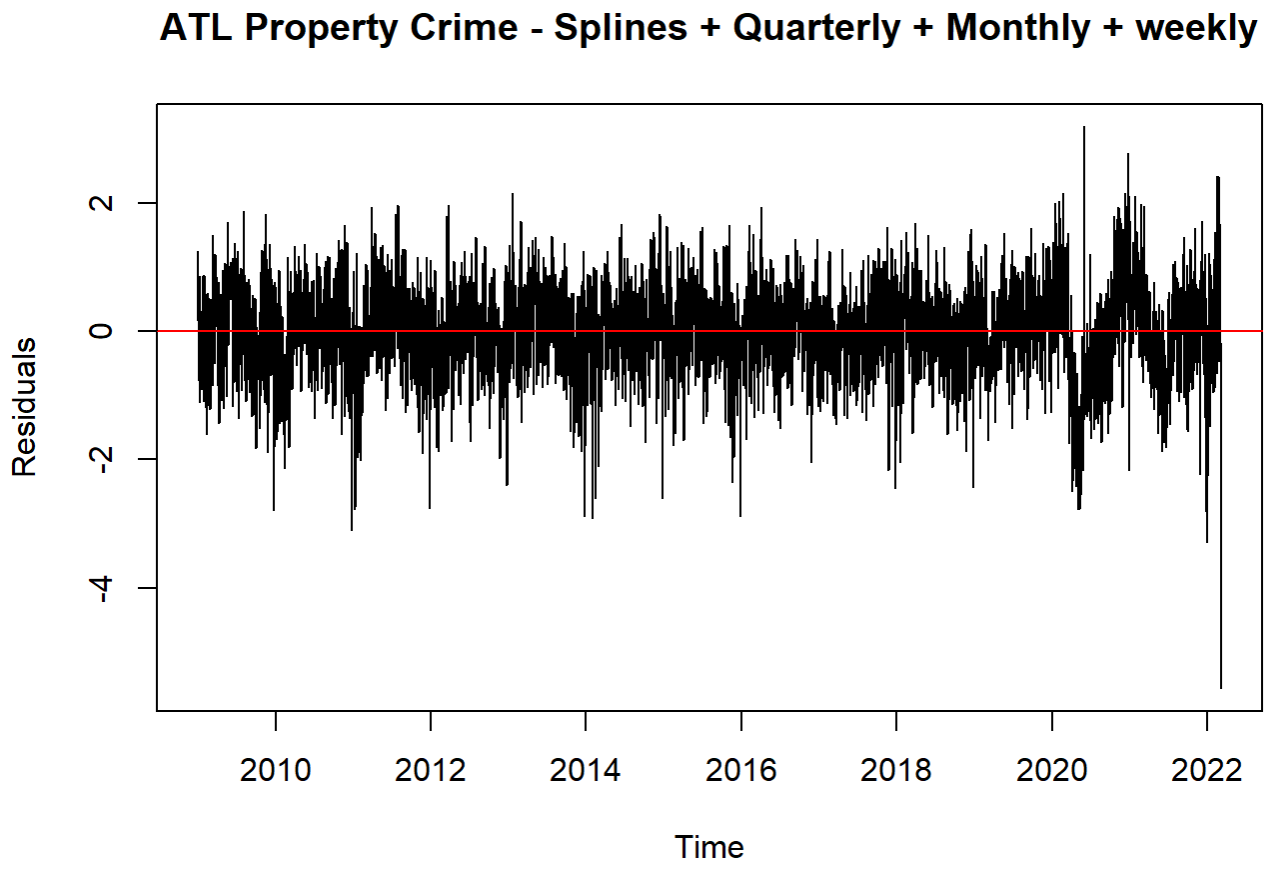

```
acf(dif.fit.seastr.1, lag.max = 365 * 4, main = "ATL Property Crime - Splines + Monthly + Weekly
Resid ACF")
```



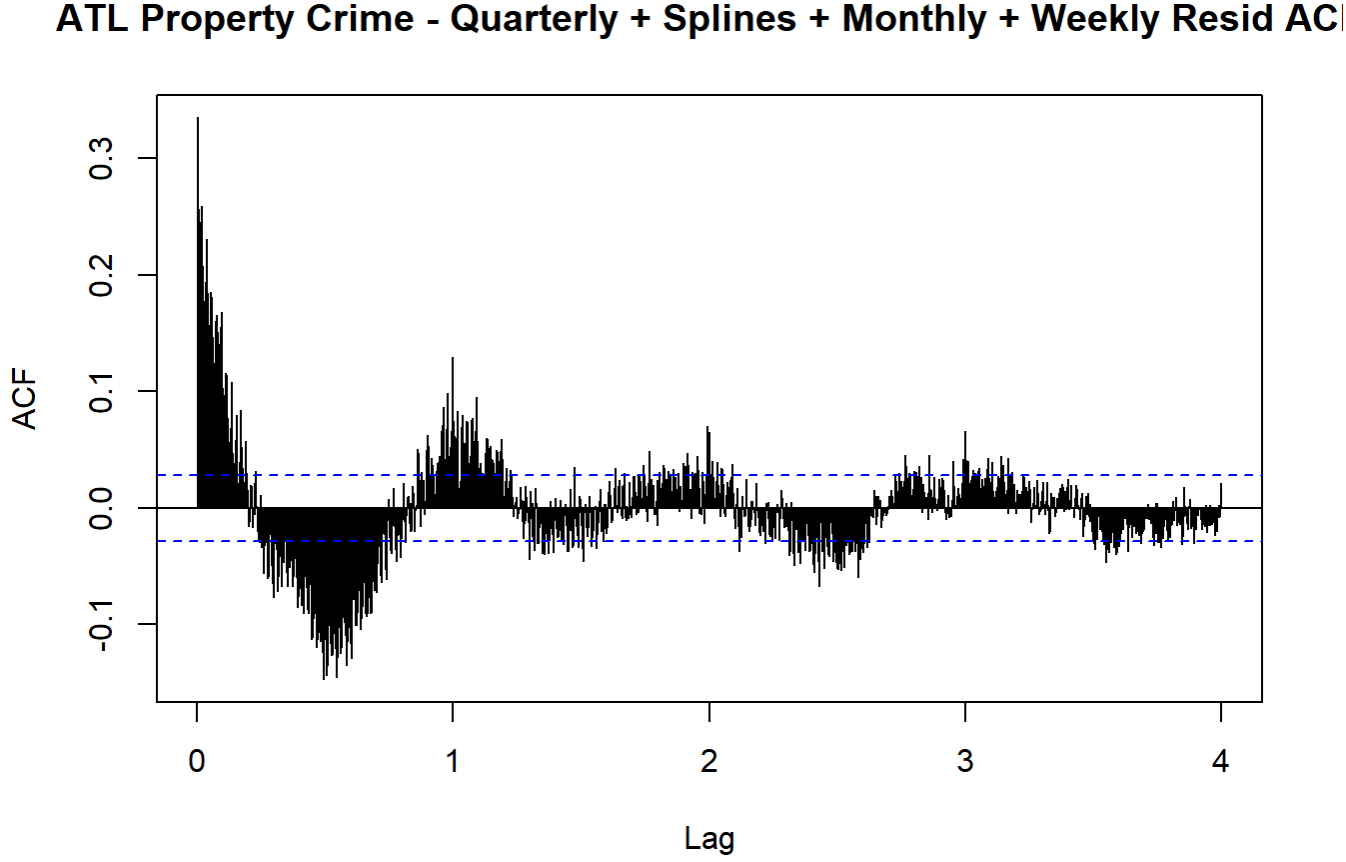

```
###TREND + SEASONALITY - SPLINES + QUARTERLY + MONTHLY + WEEKLY SEASONALITY
atl_prop$Date <- as.Date(atl_prop$Date)
year <- as.factor(format(atl_prop$Date,'%Y'))
month <- as.factor(format(atl_prop$Date, '%b'))
week <- as.factor(weekdays(atl_prop$Date))
quarterly <- as.factor(quarter(atl_prop$Date))

gam.fit.seastr.1 = gam(atl_p.tr~s(time.pts)+quarterly+month+week-1)
summary(gam.fit.seastr.1)
```

```
##
## Family: gaussian
## Link function: identity
##
## Formula:
## atl_p.tr ~ s(time.pts) + quarterly + month + week - 1
##
## Parametric coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## quarterly1     6.10031    0.02468 247.188  < 2e-16 ***
## quarterly2     8.27338    0.04474 184.910  < 2e-16 ***
## quarterly3     6.66232    0.02480 268.669  < 2e-16 ***
## quarterly4     6.70264    0.02483 269.936  < 2e-16 ***
## monthAug       2.20811    0.03065  72.045  < 2e-16 ***
## monthDec       2.26673    0.03066  73.921  < 2e-16 ***
## monthFeb       1.89977    0.03078  61.726  < 2e-16 ***
## monthJan       2.38610    0.02976  80.175  < 2e-16 ***
## monthJul       2.26934    0.03062  74.109  < 2e-16 ***
## monthJun       0.54140    0.05189  10.435  < 2e-16 ***
## monthMar       1.81444    0.03047  59.542  < 2e-16 ***
## monthMay       0.31761    0.05146   6.172 7.29e-10 ***
## monthNov       2.21581    0.03099  71.499  < 2e-16 ***
## monthOct       2.22010    0.03063  72.471  < 2e-16 ***
## monthSep       2.18487    0.03101  70.460  < 2e-16 ***
## weekMonday    -0.28187    0.03907  -7.214 6.27e-13 ***
## weekSaturday  -0.11956    0.03907  -3.060  0.00222 **
## weekSunday    -0.79073    0.03907 -20.239  < 2e-16 ***
## weekThursday  -0.42542    0.03906 -10.893  < 2e-16 ***
## weekTuesday   -0.39333    0.03907 -10.067  < 2e-16 ***
## weekWednesday -0.37714    0.03907  -9.653  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Approximate significance of smooth terms:
##              edf Ref.df   F p-value
## s(time.pts) 8.81  8.989 583  <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Rank: 27/30
## R-sq.(adj) =  0.583   Deviance explained = 99.3%
## GCV = 0.52766  Scale est. = 0.52472   n = 4811
```

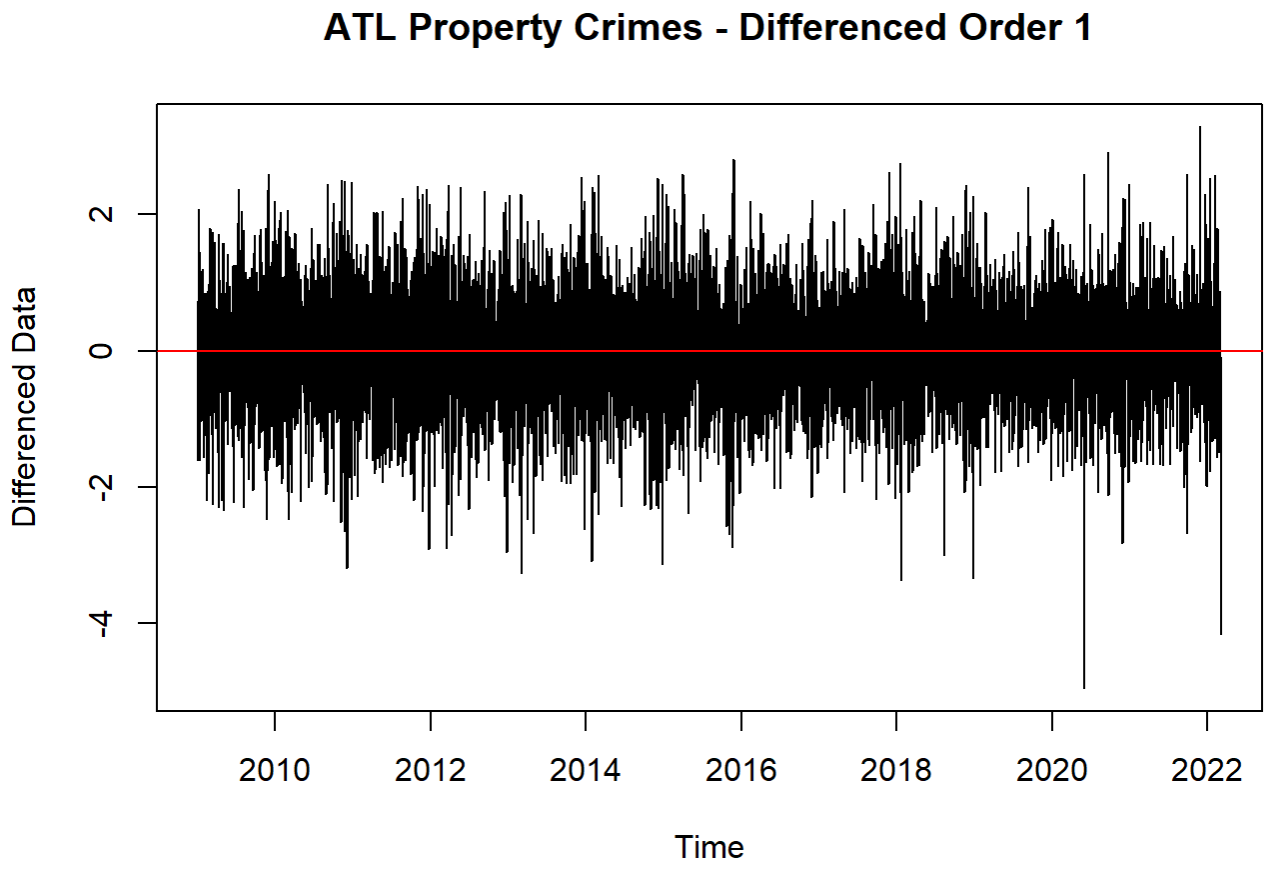
```
dif.fit.seastr.1 = ts(atl_p.tr - fitted(gam.fit.seastr.1),start=2009,frequency=365)

ts.plot(dif.fit.seastr.1, ylab = "Residuals", main = "ATL Property Crime - Splines + Quarterly +
Monthly + weekly")
abline(h=0, col='red')
```



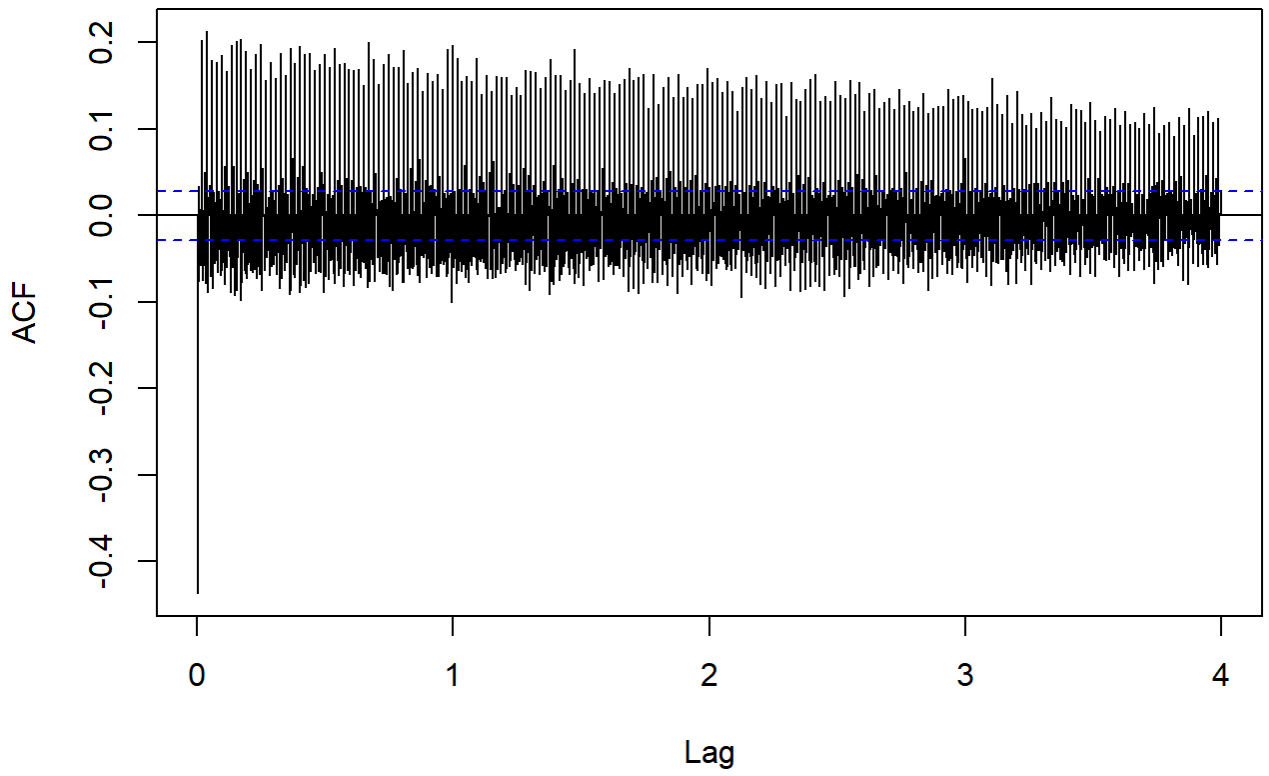

```
acf(dif.fit.seastr.1, lag.max = 365 * 4, main = "ATL Property Crime - Quarterly + Splines + Mont
hly + Weekly Resid ACF")
```




```
###DIFFERENCED DATA - ORDER 1
diff.atl_p = diff(atl_p.tr)
ts.plot(diff.atl_p, ylab = "Differenced Data", main = "ATL Property Crimes - Differenced Order
 1")
abline(h=0, col='red')
```

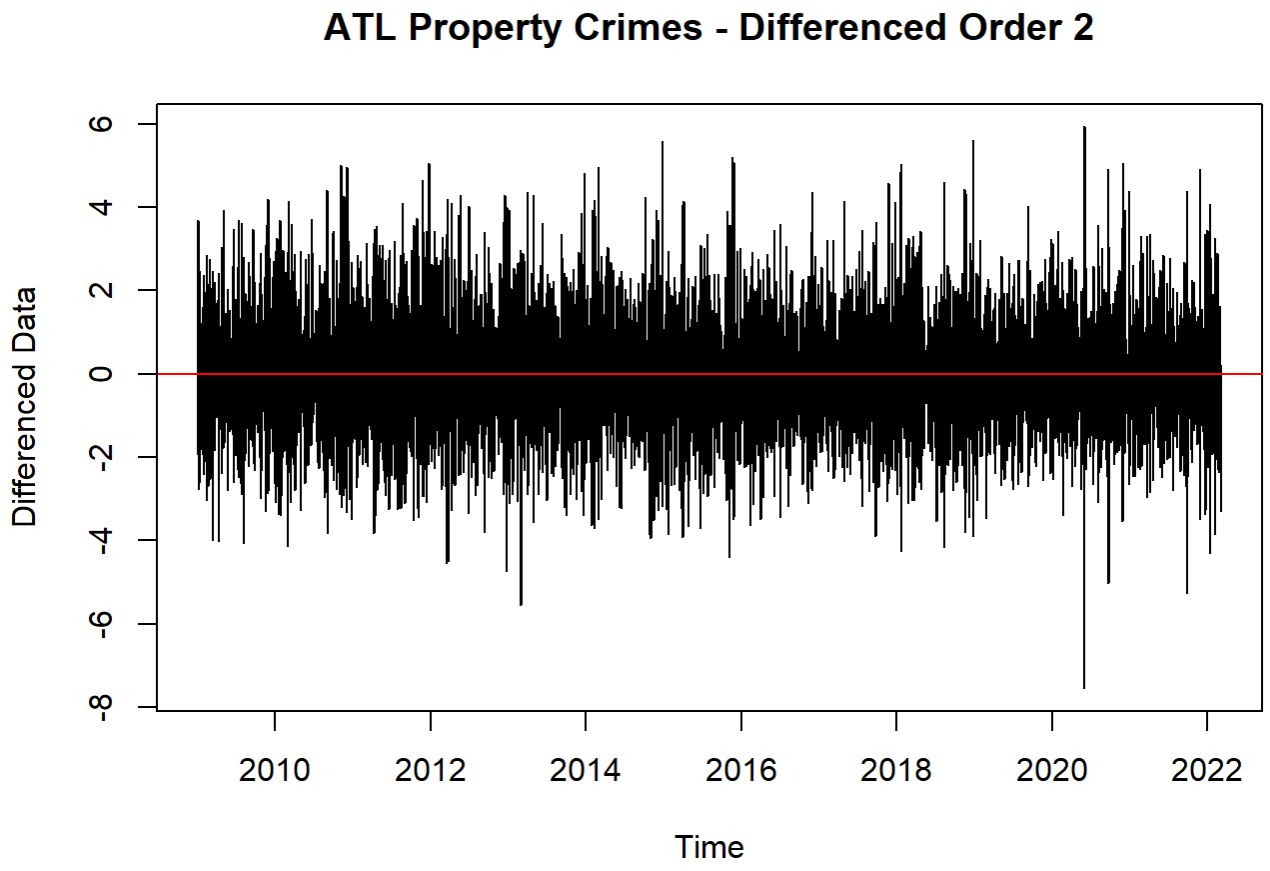



```
acf(diff.atl_p, lag.max = 365 * 4, main = "ATL Property Crimes - Differenced Order 1 ACF")
```

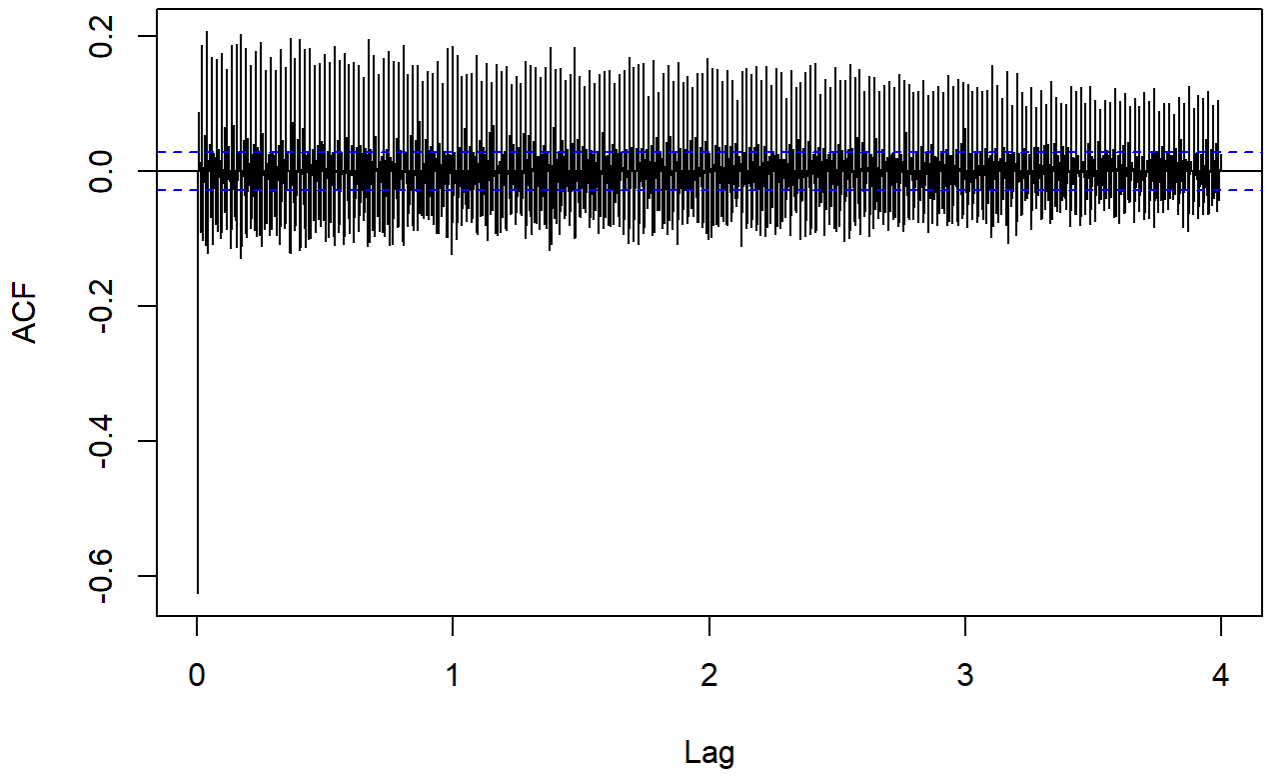



```
### SEE HW2 - Q2 FOR EXPLANATION
```

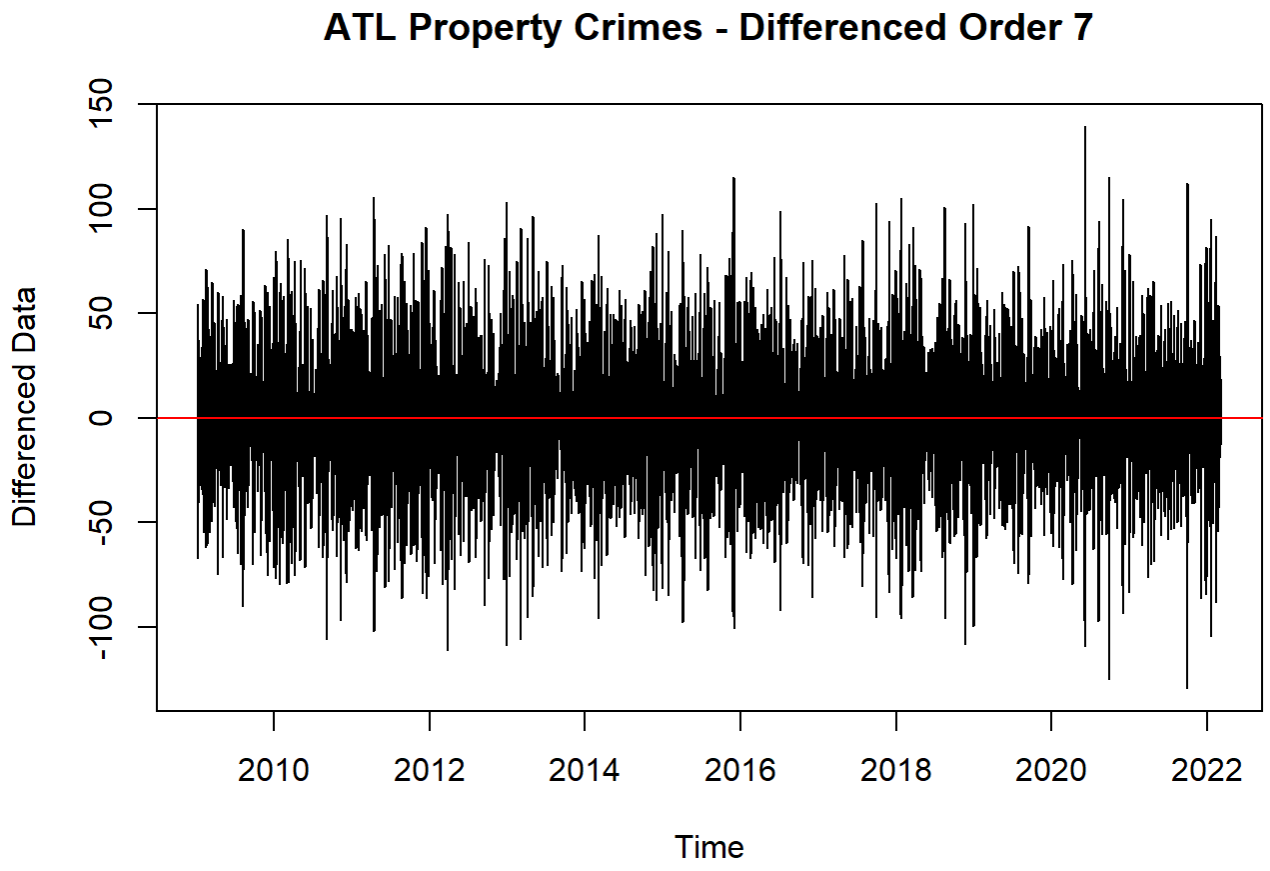
```
###DIFFERENCED DATA - ORDER 2
diff.atl_p3 = diff(atl_p.tr, differences = 2)
ts.plot(diff.atl_p3, ylab = "Differenced Data", main = "ATL Property Crimes - Differenced Order
 2")
abline(h=0, col='red')
```



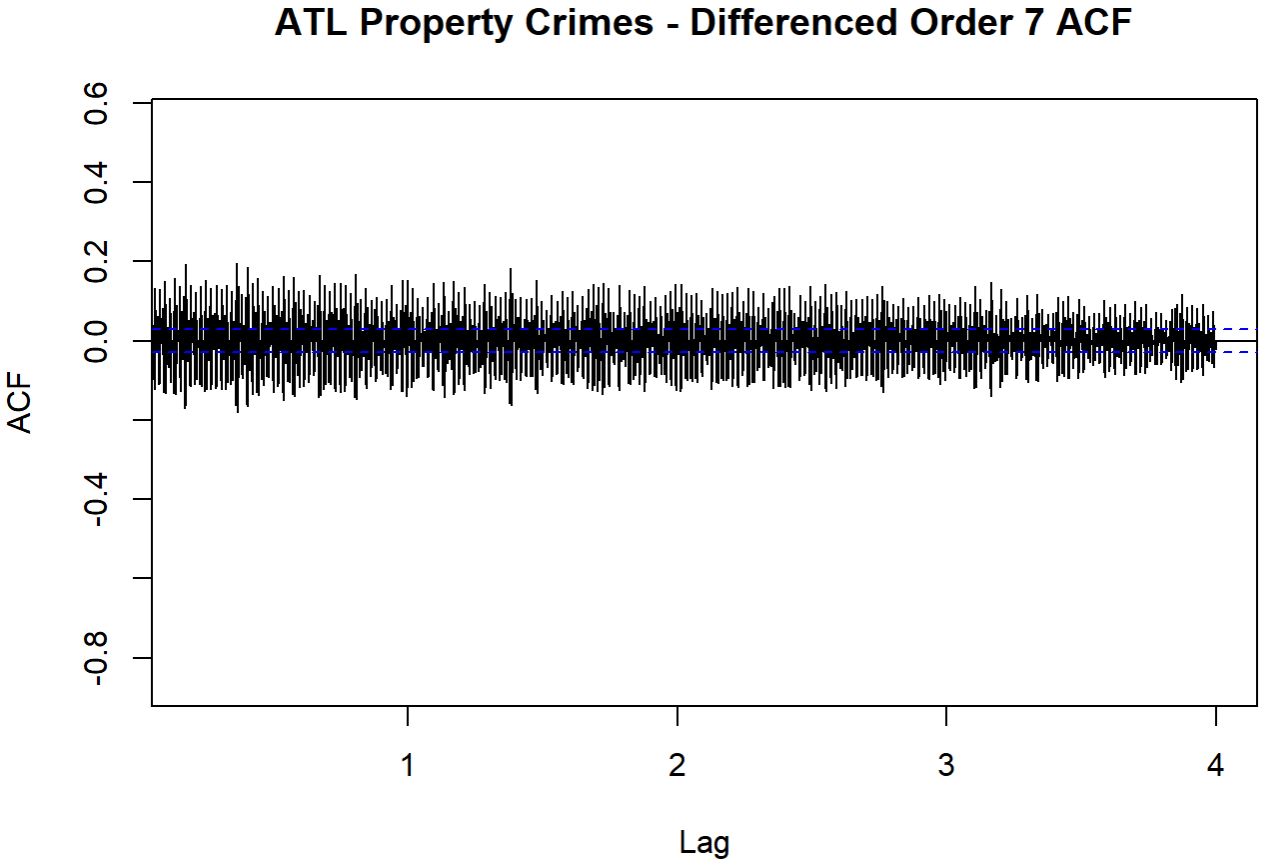

```
acf(diff.atl_p3, lag.max = 365 * 4, main = "ATL Property Crimes - Differenced Order 2 ACF")
```



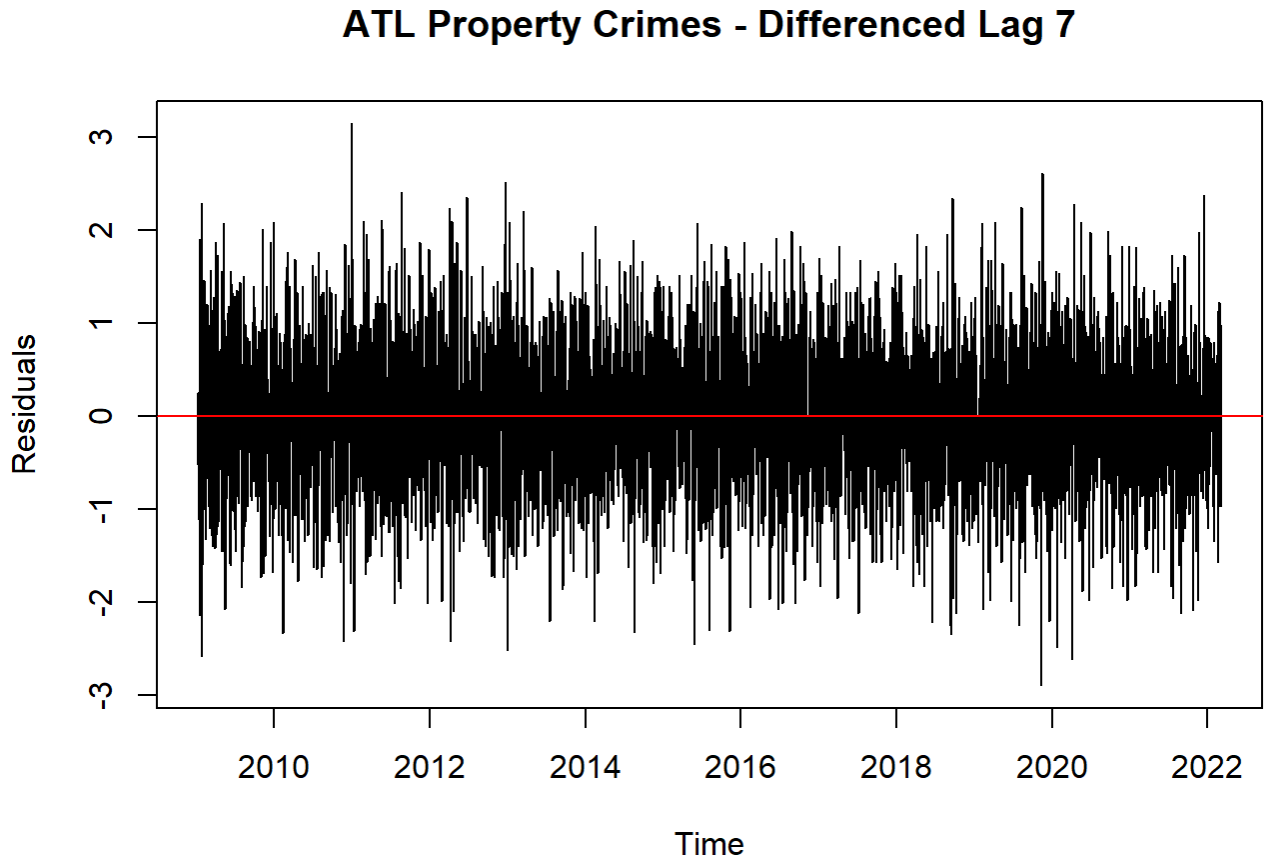

```
###DIFFERENCED DATA - ORDER 7
diff.atl_p3 = diff(atl_p.tr, differences = 7)
ts.plot(diff.atl_p3, ylab = "Differenced Data", main = "ATL Property Crimes - Differenced Order
 7")
abline(h=0, col='red')
```



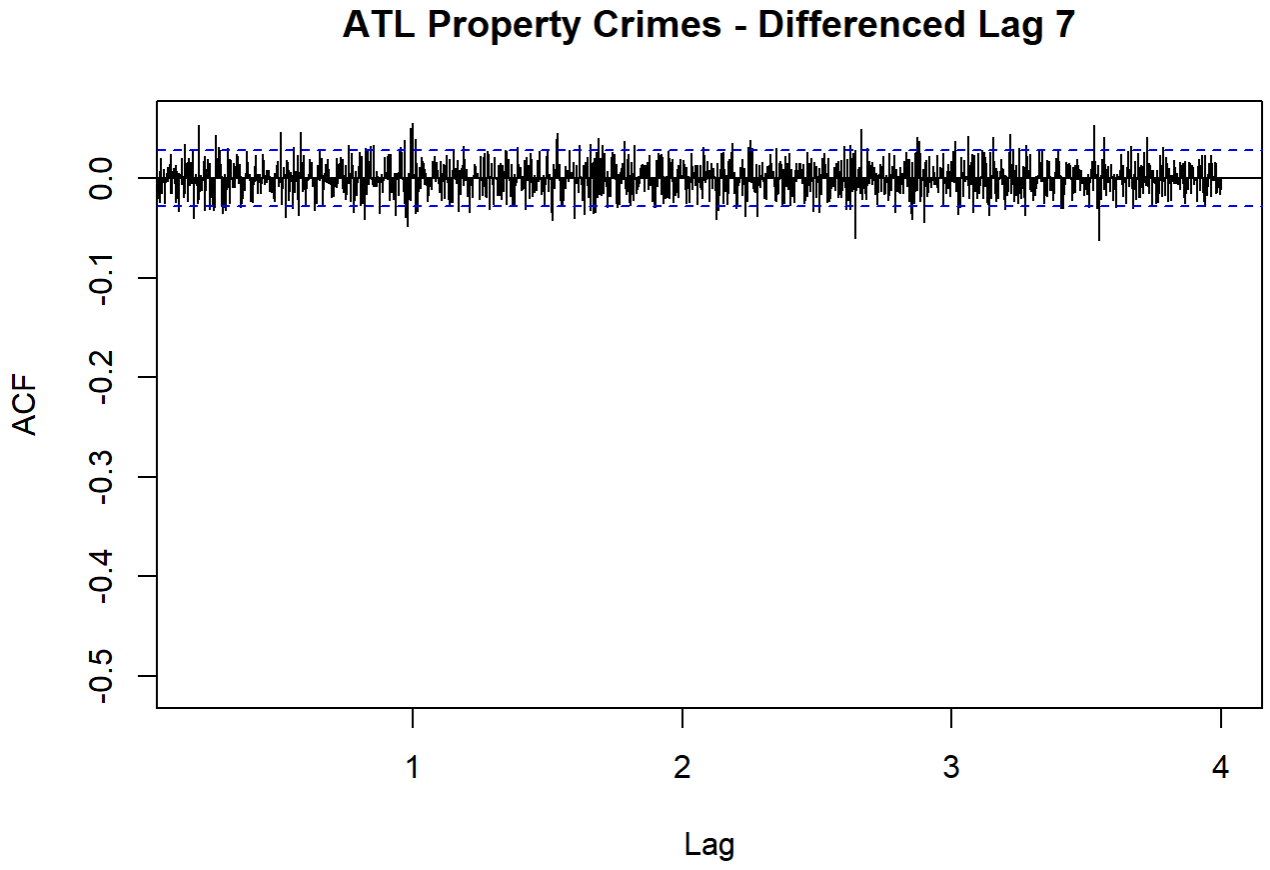

```
acf(diff.atl_p3, xlim=c(0.2,4), lag.max = 365 * 4, main = "ATL Property Crimes - Differenced Ord
er 7 ACF")
```




```
###DIFFERENCED DATA - LAG 365
diff.atl_p2 = diff(atl_v.tr, lag = 7)
ts.plot(diff.atl_p2, ylab = "Residuals", main = "ATL Property Crimes - Differenced Lag 7")
abline(h=0, col='red')
```

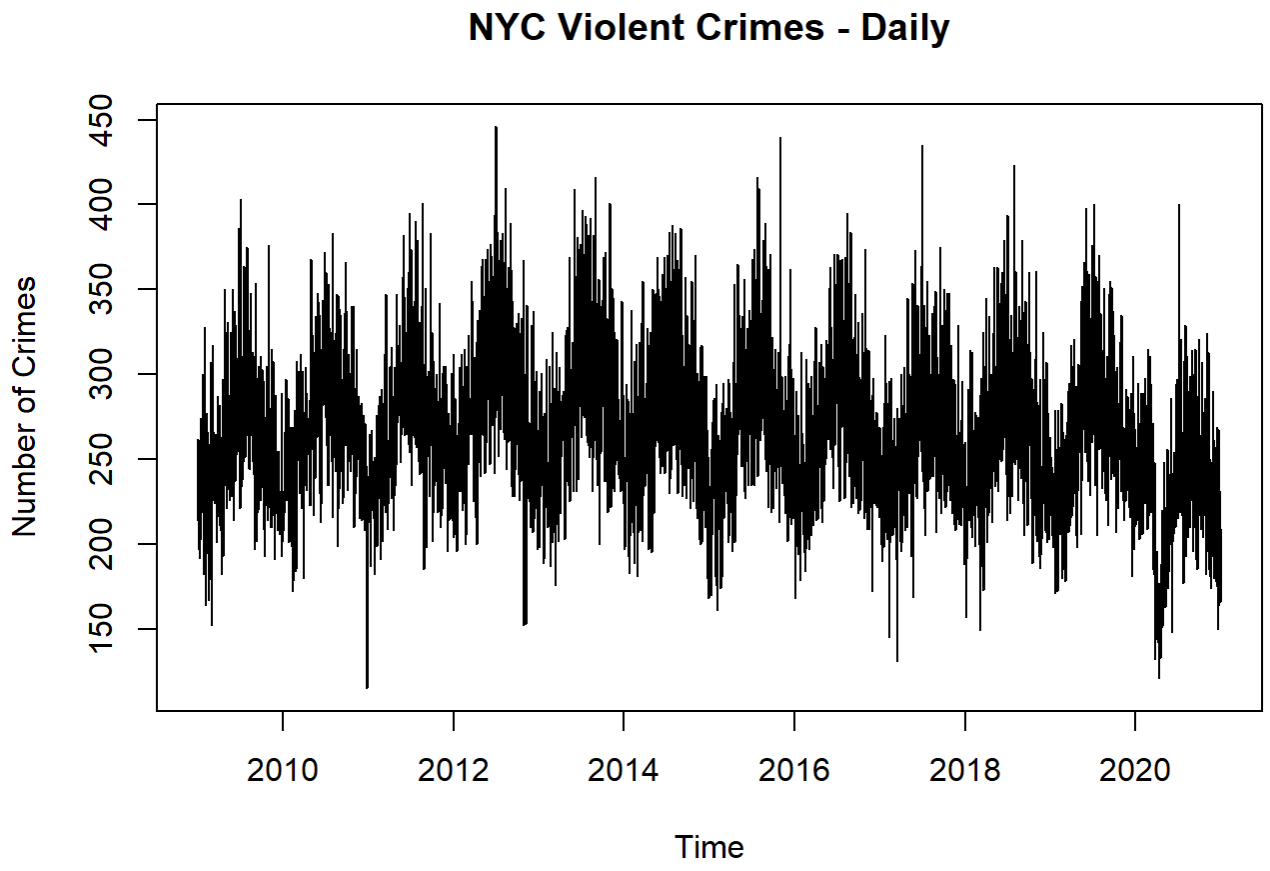



```
acf(diff.atl_p2, xlim=c(0.2,4), lag.max = 365 * 4, main = "ATL Property Crimes - Differenced Lag
7")
```

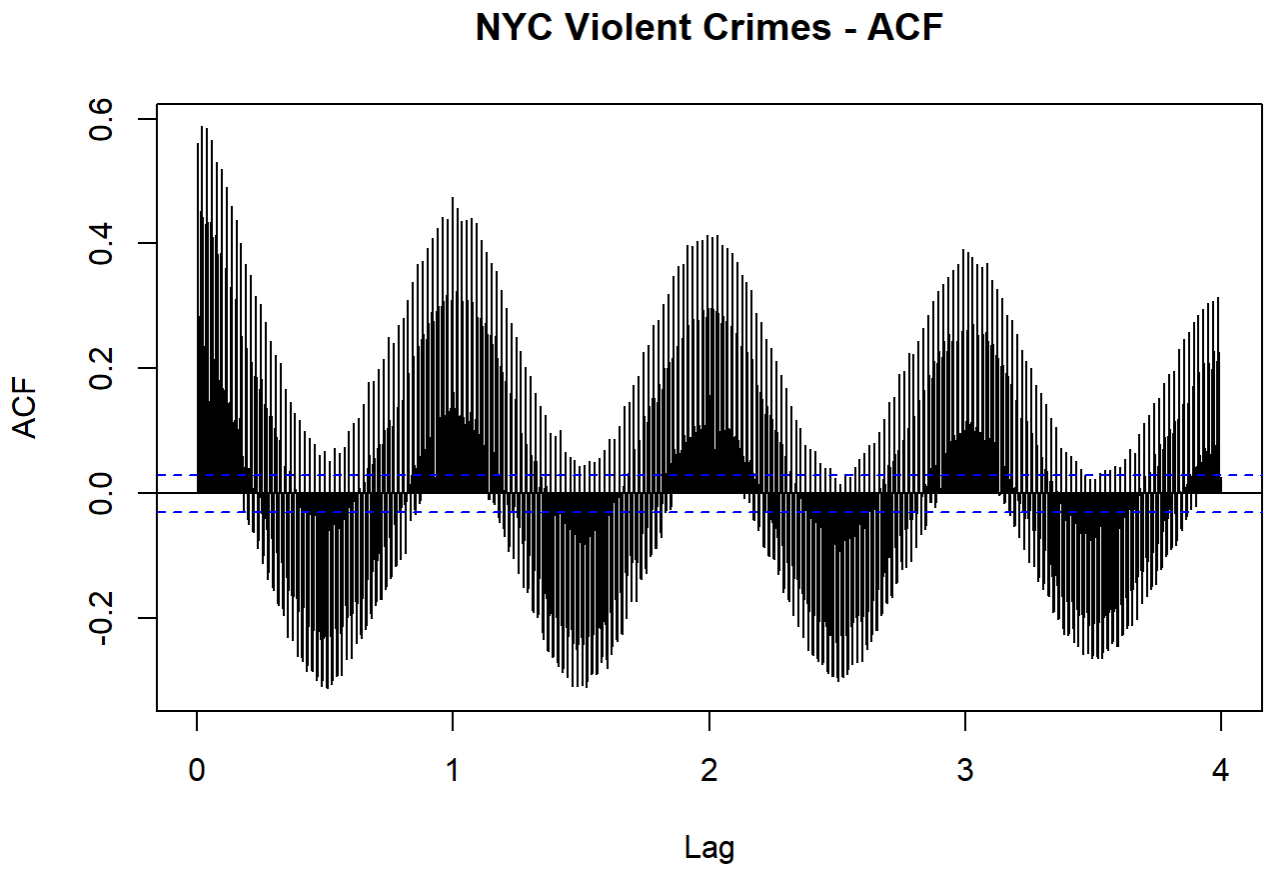


NYC Violent Crime

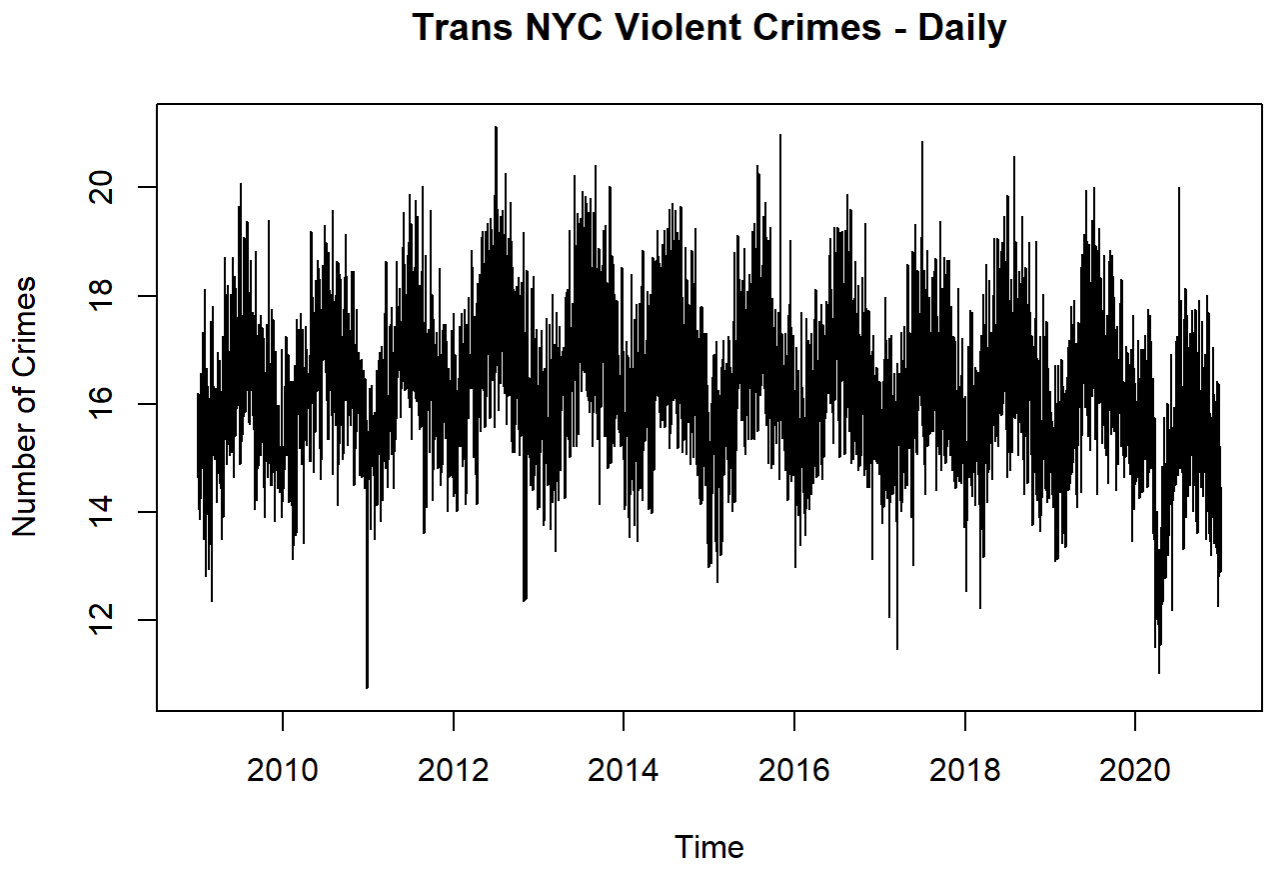
```
###EDA - Time Series / ACF

ts.plot(nyc_v,ylab="Number of Crimes",main="NYC Violent Crimes - Daily")
```



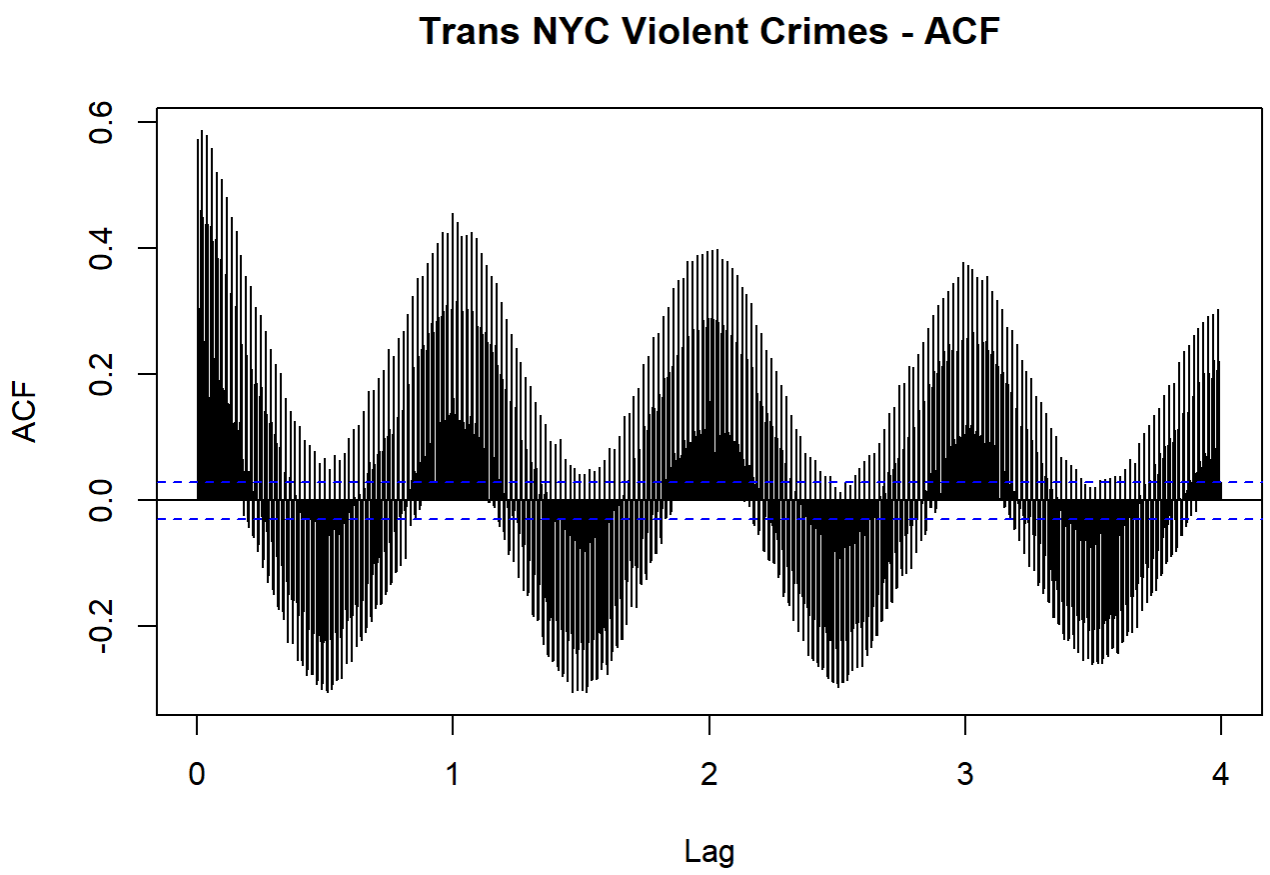

```
acf(nyc_v,lag.max=365*4,main="NYC Violent Crimes - ACF")
```



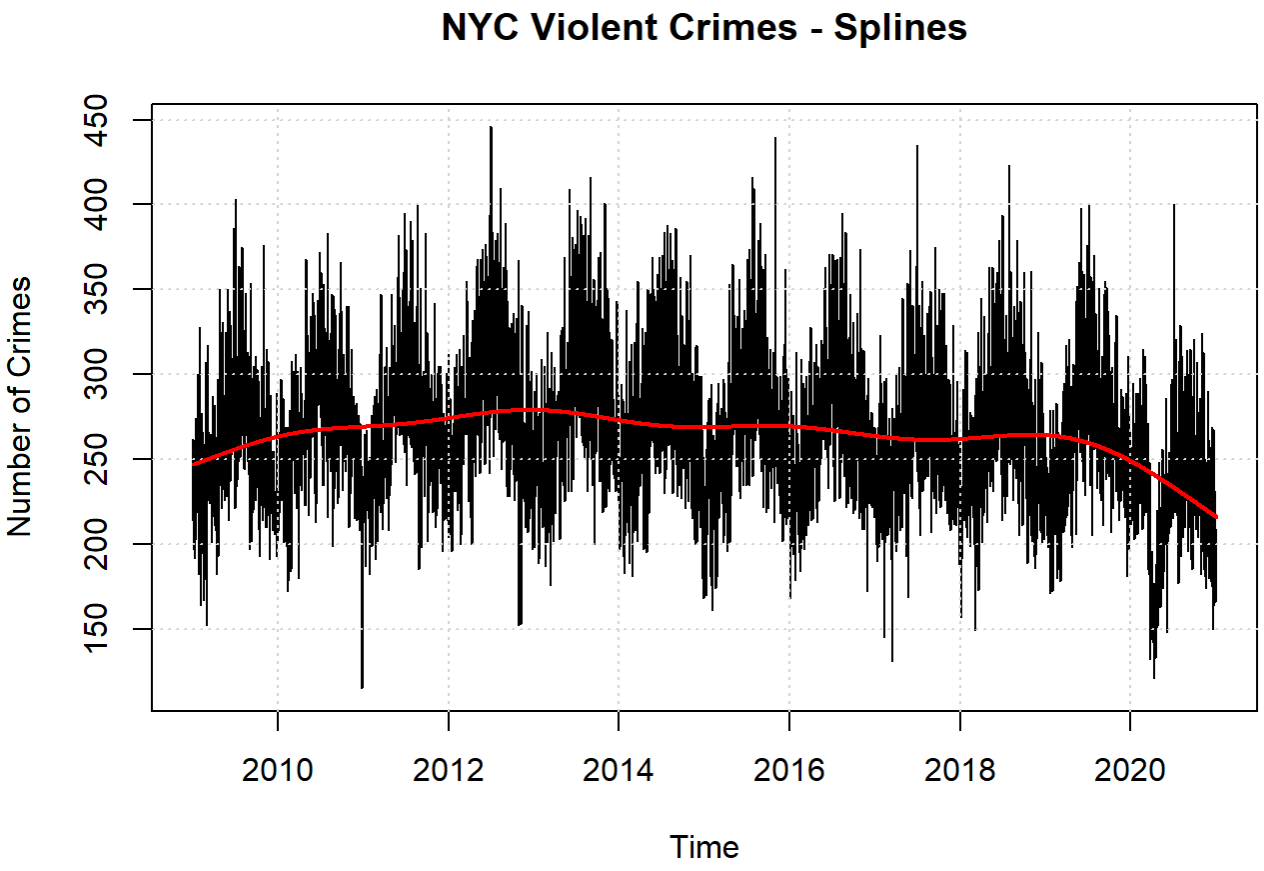

```
ts.plot(nyc_v.tr,ylab="Number of Crimes",main="Trans NYC Violent Crimes - Daily")
```



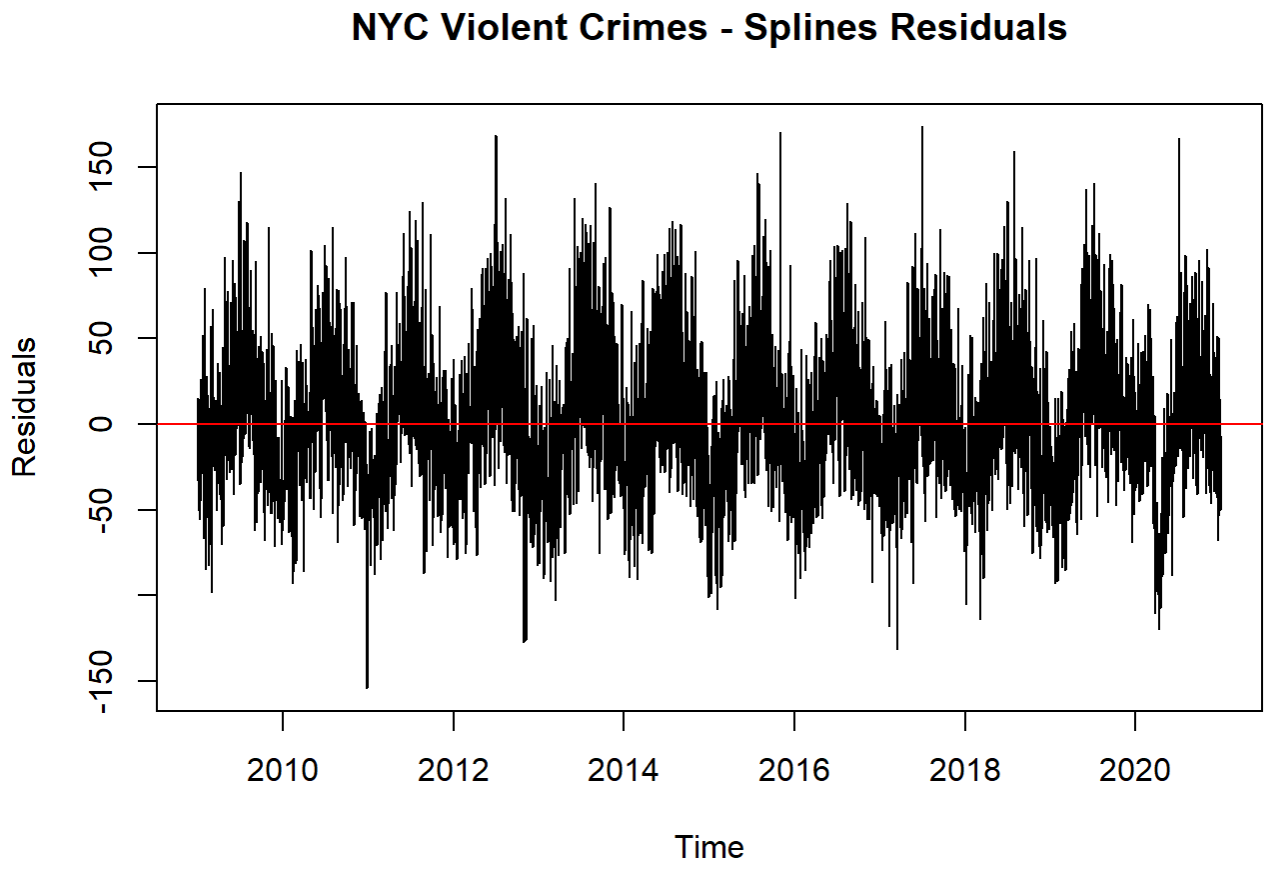

```
acf(nyc_v.tr,lag.max=365*4,main="Trans NYC Violent Crimes - ACF")
```



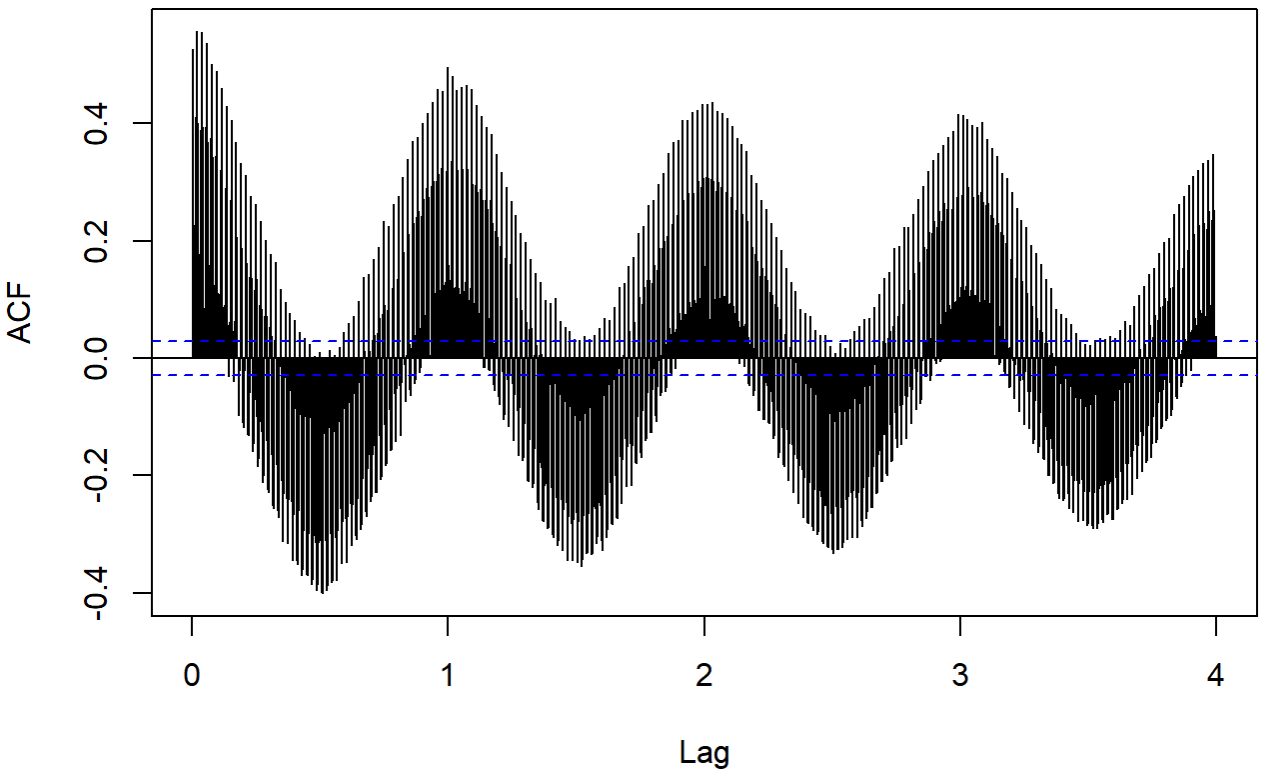

```
###TREND ESTIMATION - SPLINES
# Convert X-Axis to 0-1 Scale
time.pts = c(1:length(nyc_v))
time.pts = c(time.pts - min(time.pts))/max(time.pts)

#Splines Trend Estimation
gam.fit = gam(nyc_v~s(time.pts))
nyc_v.fit.gam = ts(fitted(gam.fit),start=2009,frequency=365)

##Is there a trend?
ts.plot(nyc_v,ylab="Number of Crimes", main = "NYC Violent Crimes - Splines")
grid()
lines(nyc_v.fit.gam,lwd=2,col="red")
```



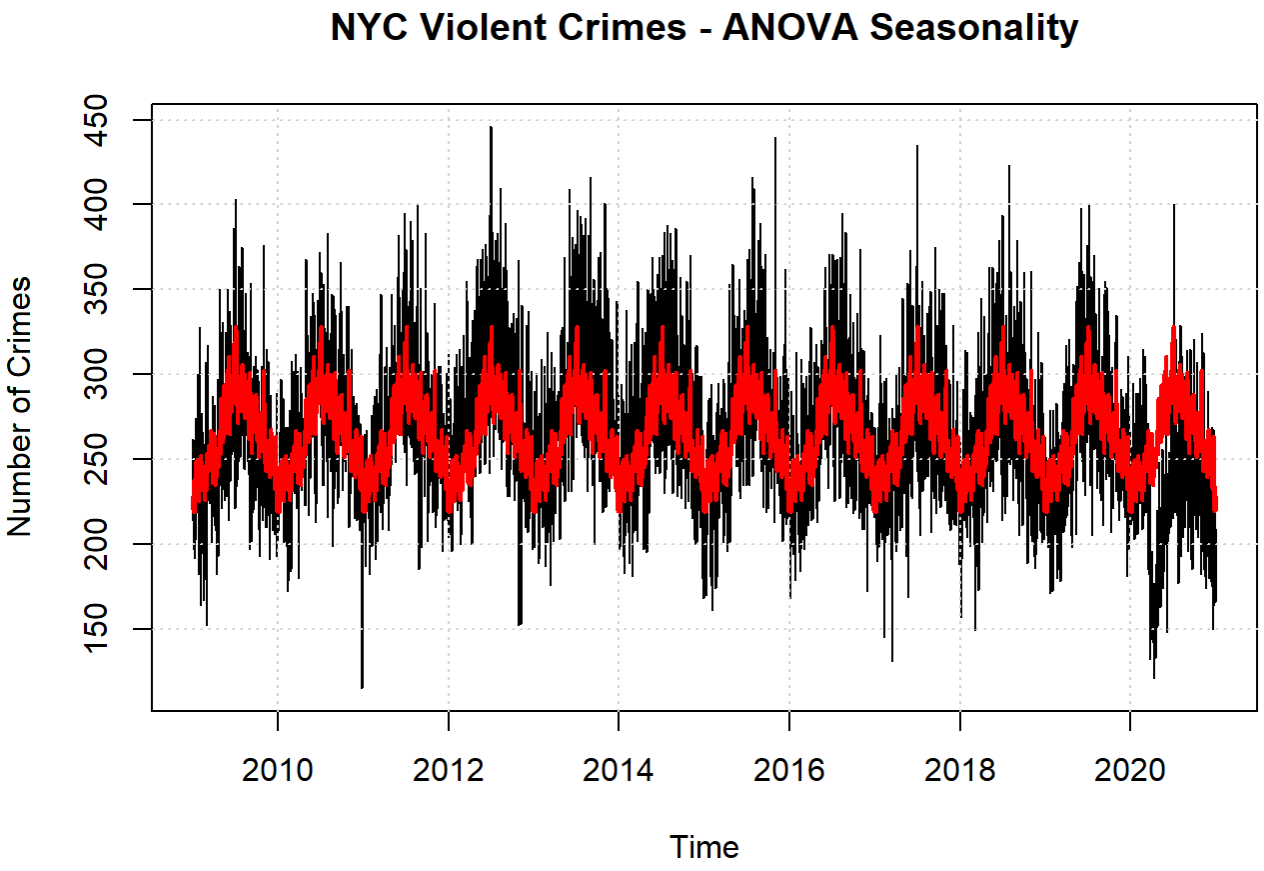

```
# Splines Residuals + Residuals ACF
dif.fit.gam = nyc_v - nyc_v.fit.gam
ts.plot(dif.fit.gam, ylab = "Residuals", main = "NYC Violent Crimes - Splines Residuals")
abline(h=0, col='red')
```



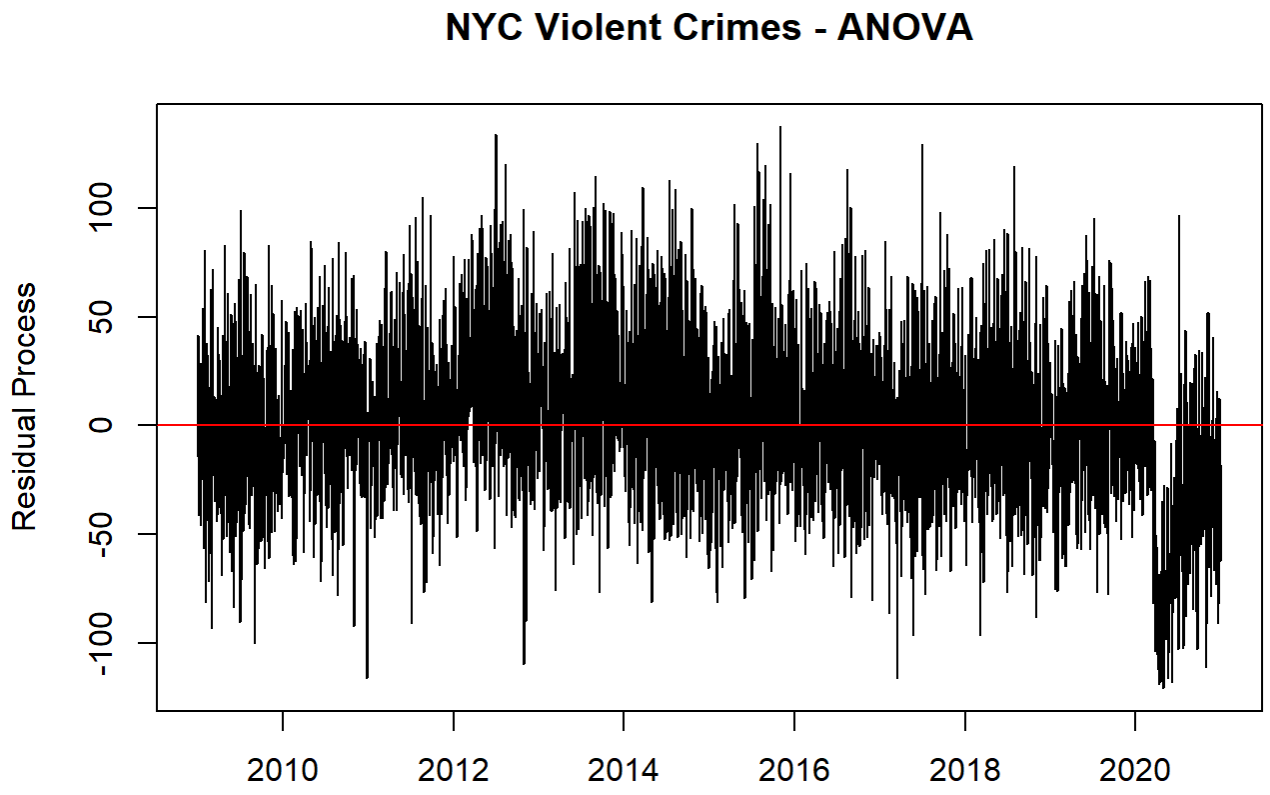

```
acf(dif.fit.gam, lag.max = 365 * 4, main = "NYC Violent Crimes - Splines Residuals - ACF")
```



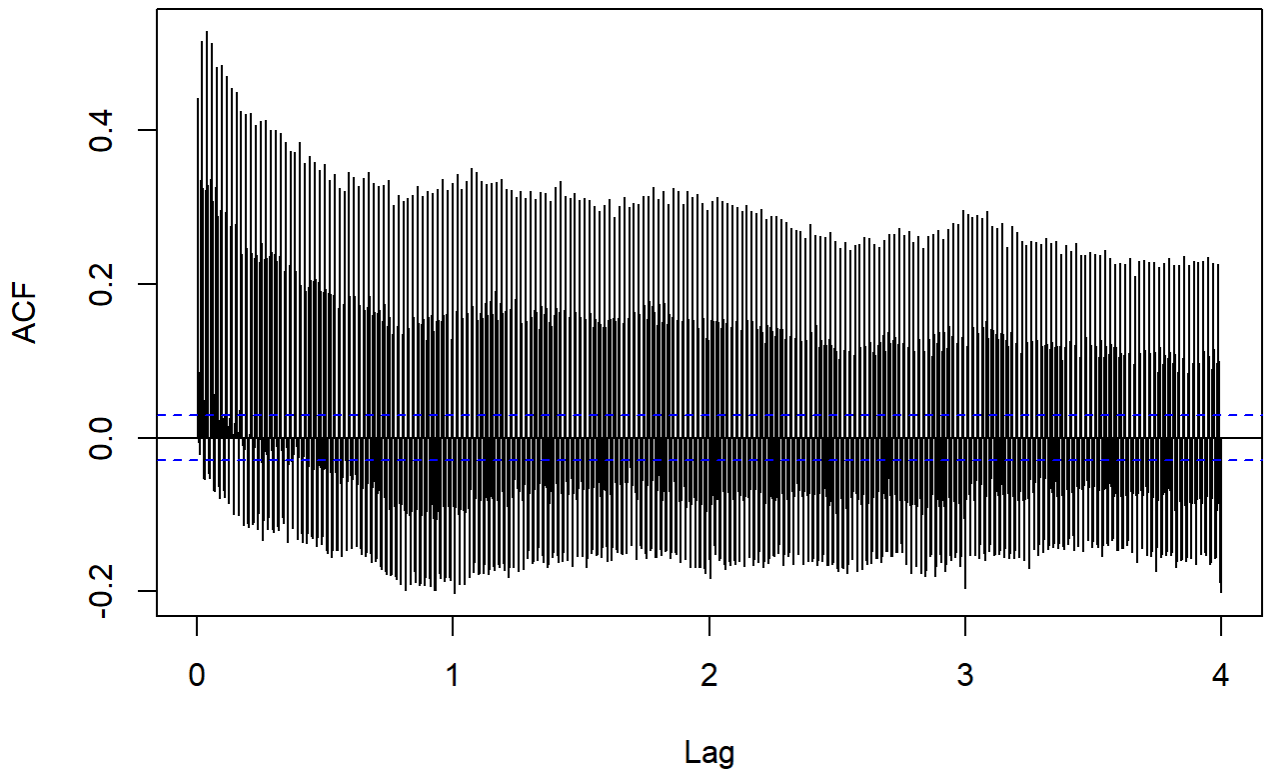

```
###SEASONALITY ANALYSIS - ANOVA

## Estimate seasonality using ANOVA approach
model.anova = dynlm(nyc_v~season(nyc_v))
#summary(model.anova)

## Plot
ts.plot(nyc_v,ylab="Number of Crimes", main = "NYC Violent Crimes - ANOVA Seasonality")
grid()
lines(fitted(model.anova),lwd=2,col="red")
```




```
###Seasonality ANOVA- Residuals + Resid ACF
resid.anova <- residuals(model.anova)

ts.plot(resid.anova, xlab = "", ylab = "Residual Process", main = "NYC Violent Crimes - ANOVA")
abline(h=0, col='red')
```




```
acf(resid.anova, lag.max = 365 * 4, main = "NYC Violent Crimes - ANOVA ACF")
```




```
###SEASONALITY ANALYSIS - HARMONIC
###SUMMARY
harmonic.1 = dynlm(nyc_v~harmon(nyc_v))
summary(harmonic.1)
```

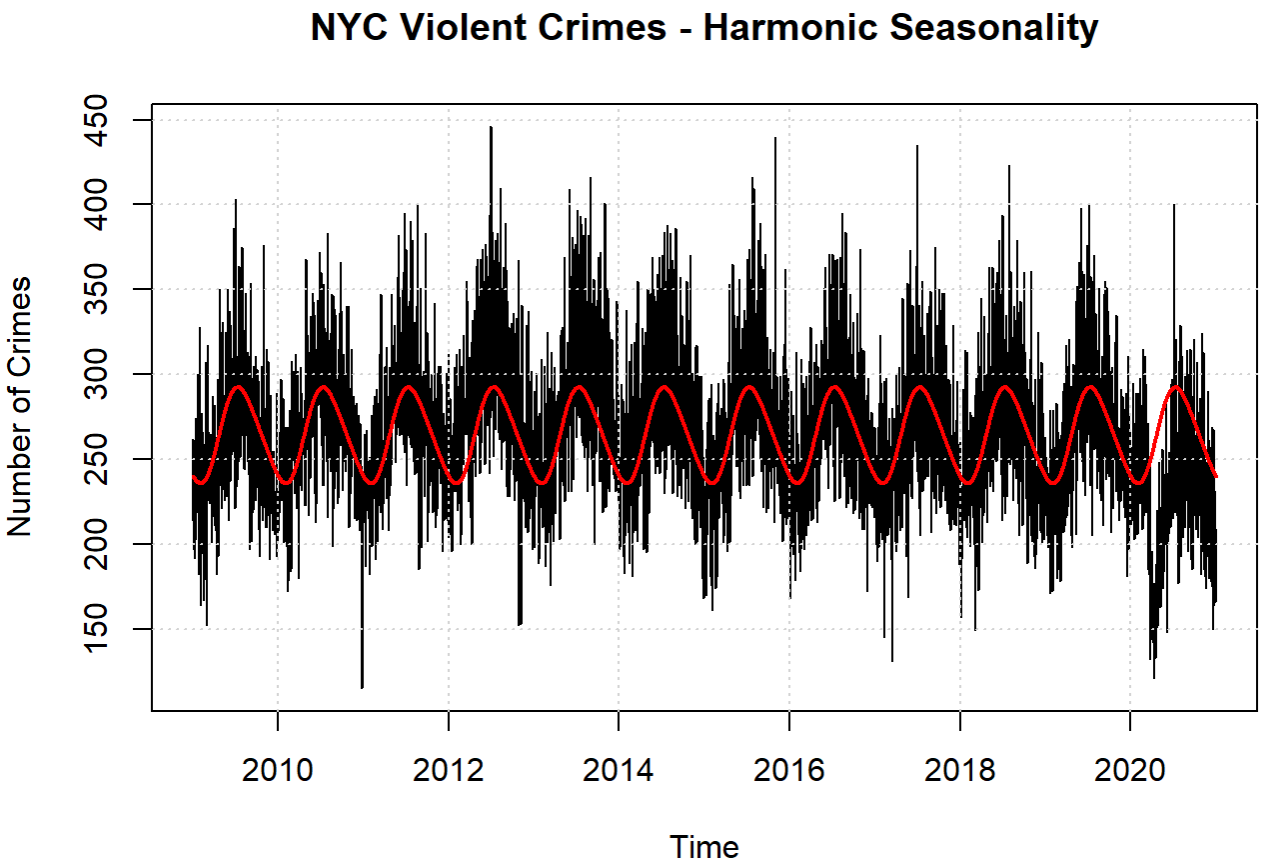
```
##
## Time series regression with "ts" data:
## Start = 2009(1), End = 2021(3)
##
## Call:
## dynlm(formula = nyc_v ~ harmon(nyc_v))
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -138.666  -26.055   -2.335   24.292  180.829
##
## Coefficients:
##                   Estimate Std. Error t value Pr(>|t|)
## (Intercept)      264.2730     0.5908  447.34   <2e-16 ***
## harmon(nyc_v)cos -25.9563     0.8352  -31.08   <2e-16 ***
## harmon(nyc_v)sin  -9.5456     0.8357  -11.42   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 39.11 on 4380 degrees of freedom
## Multiple R-squared:  0.2002, Adjusted R-squared:  0.1998
## F-statistic: 548.2 on 2 and 4380 DF,  p-value: < 2.2e-16
```

```
harmonic.2 = dynlm(nyc_v~harmon(nyc_v,2))
summary(harmonic.2)
```

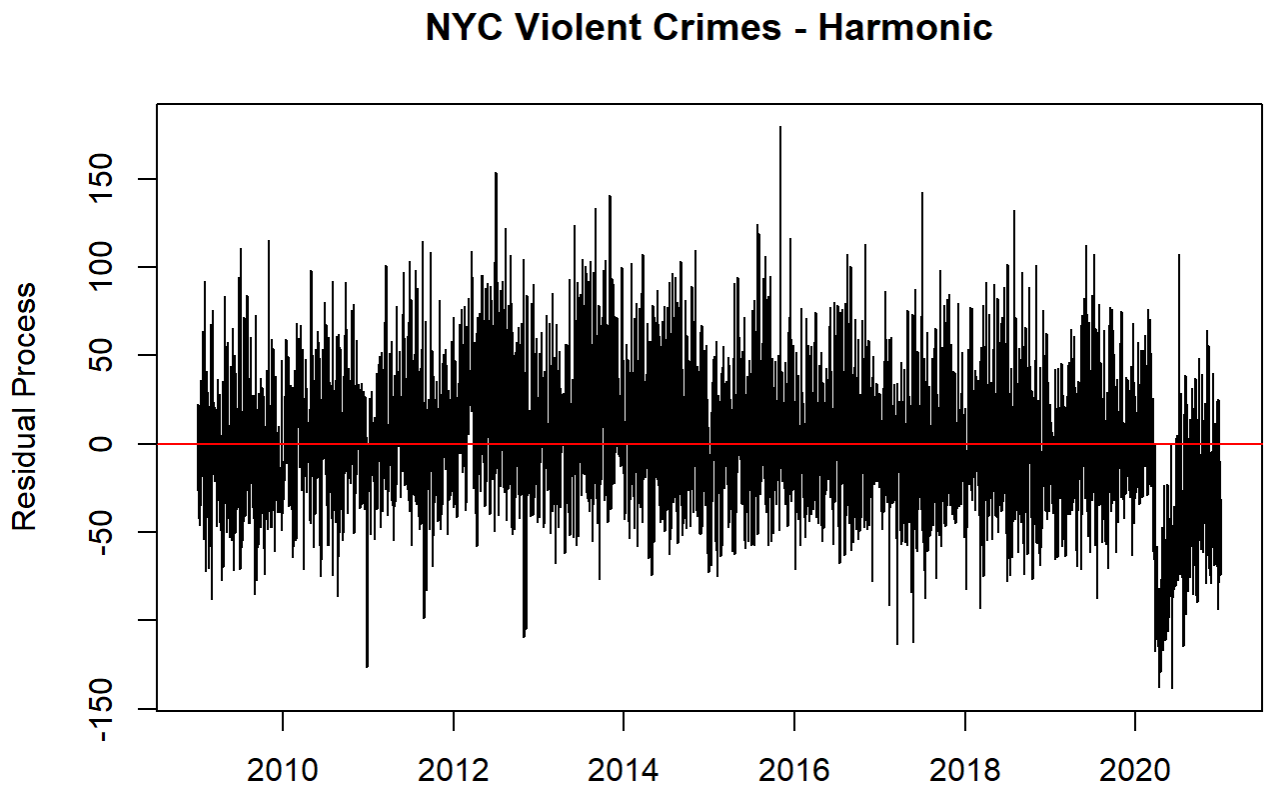
```
##
## Time series regression with "ts" data:
## Start = 2009(1), End = 2021(3)
##
## Call:
## dynlm(formula = nyc_v ~ harmon(nyc_v, 2))
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -138.423  -26.118   -2.339   24.184  179.688
##
## Coefficients:
##                       Estimate Std. Error t value Pr(>|t|)
## (Intercept)          264.2717     0.5900 447.922  < 2e-16 ***
## harmon(nyc_v, 2)cos1 -25.9589     0.8341 -31.122  < 2e-16 ***
## harmon(nyc_v, 2)cos2   1.9480     0.8341   2.335  0.01956 *
## harmon(nyc_v, 2)sin1  -9.5457     0.8347 -11.437  < 2e-16 ***
## harmon(nyc_v, 2)sin2  -2.3512     0.8347  -2.817  0.00487 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 39.06 on 4378 degrees of freedom
## Multiple R-squared:  0.2026, Adjusted R-squared:  0.2019
## F-statistic: 278.2 on 4 and 4378 DF,  p-value: < 2.2e-16
```

```
###use more complex model. all values stat sig at 95% conf level.
```

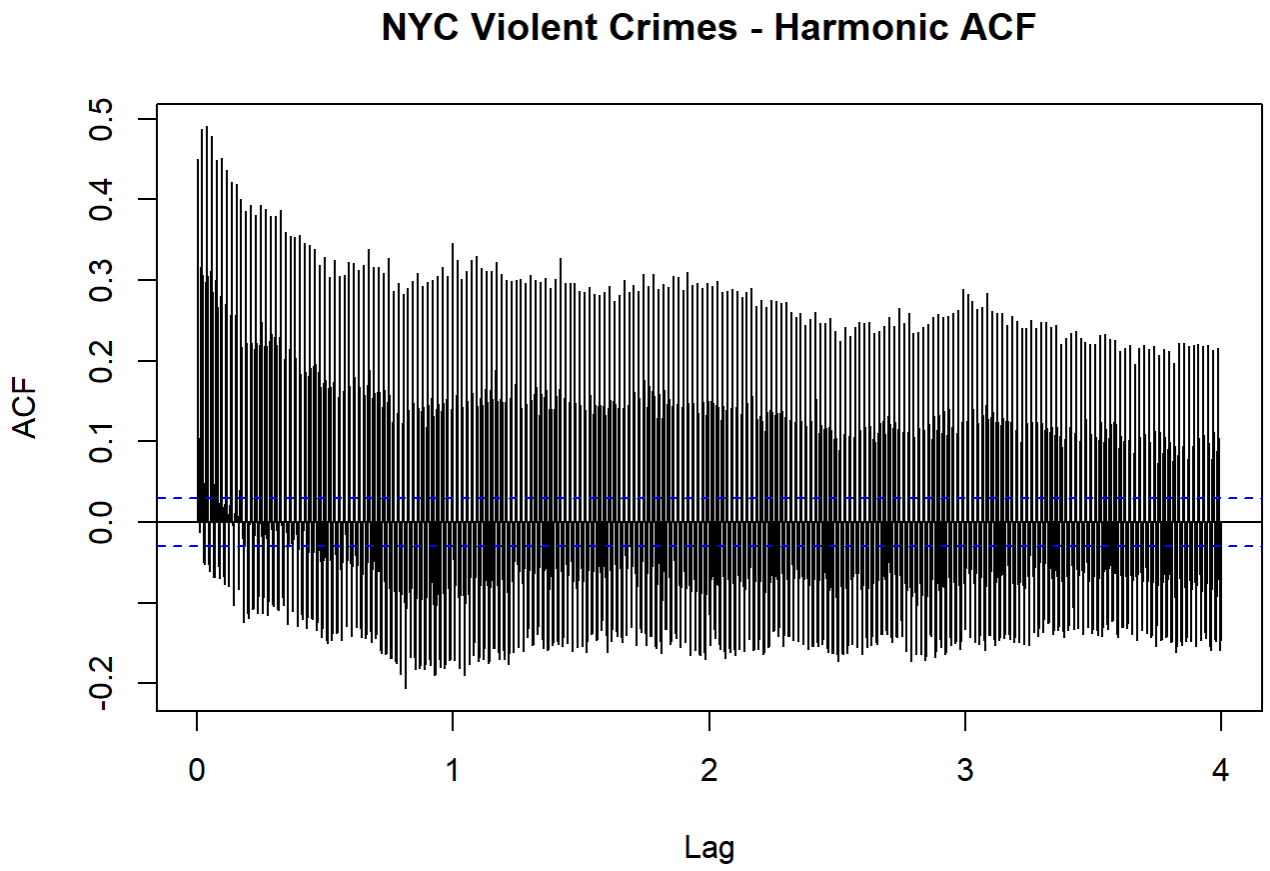
```
###Harmonic Plot
## Plot
ts.plot(nyc_v,ylab="Number of Crimes", main = "NYC Violent Crimes - Harmonic Seasonality")
grid()
lines(fitted(harmonic.2),lwd=2,col="red")
```




```
###Seasonality Harmonic - Residuals + Resid ACF

resid.harmonic <- residuals(harmonic.2)

ts.plot(resid.harmonic, xlab = "", ylab = "Residual Process", main = "NYC Violent Crimes - Harmo
nic")
abline(h=0, col='red')
```




```
acf(resid.harmonic, lag.max = 365 * 4, main = "NYC Violent Crimes - Harmonic ACF")
```



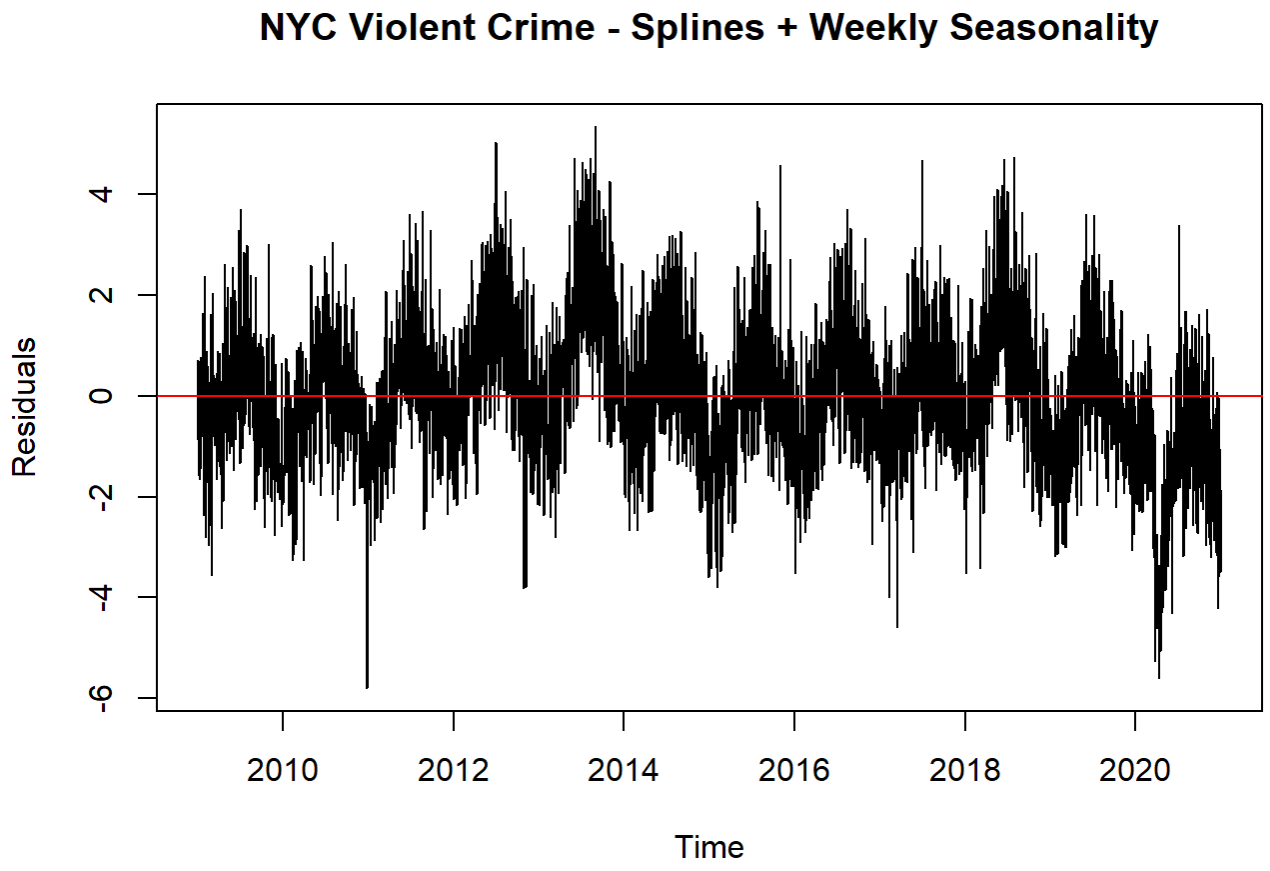

```
###TREND + SEASONALITY - SPLINES + weekLY SEASONALITY
nyc_violent$Date <- as.Date(nyc_violent$Date)
year <- as.factor(format(nyc_violent$Date,'%Y'))
month <- as.factor(format(nyc_violent$Date, '%b'))
week <- as.factor(weekdays(nyc_violent$Date))

gam.fit.seastr = gam(nyc_v.tr~s(time.pts)+week-1)
summary(gam.fit.seastr)
```

```
##
## Family: gaussian
## Link function: identity
##
## Formula:
## nyc_v.tr ~ s(time.pts) + week - 1
##
## Parametric coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## weekFriday    16.24462    0.08278   196.2   <2e-16 ***
## weekMonday    16.18856    0.08703   186.0   <2e-16 ***
## weekSaturday  16.13507    0.08482   190.2   <2e-16 ***
## weekSunday    16.38653    0.08278   198.0   <2e-16 ***
## weekThursday  16.31560    0.08278   197.1   <2e-16 ***
## weekTuesday   16.13603    0.07910   204.0   <2e-16 ***
## weekWednesday 16.26188    0.08703   186.9   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Approximate significance of smooth terms:
##               edf Ref.df     F p-value
## s(time.pts) 7.655  8.543 14.49  <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## R-sq.(adj) =  0.0665   Deviance explained = 99.4%
## GCV = 1.7415  Scale est. = 1.7267    n = 1728
```

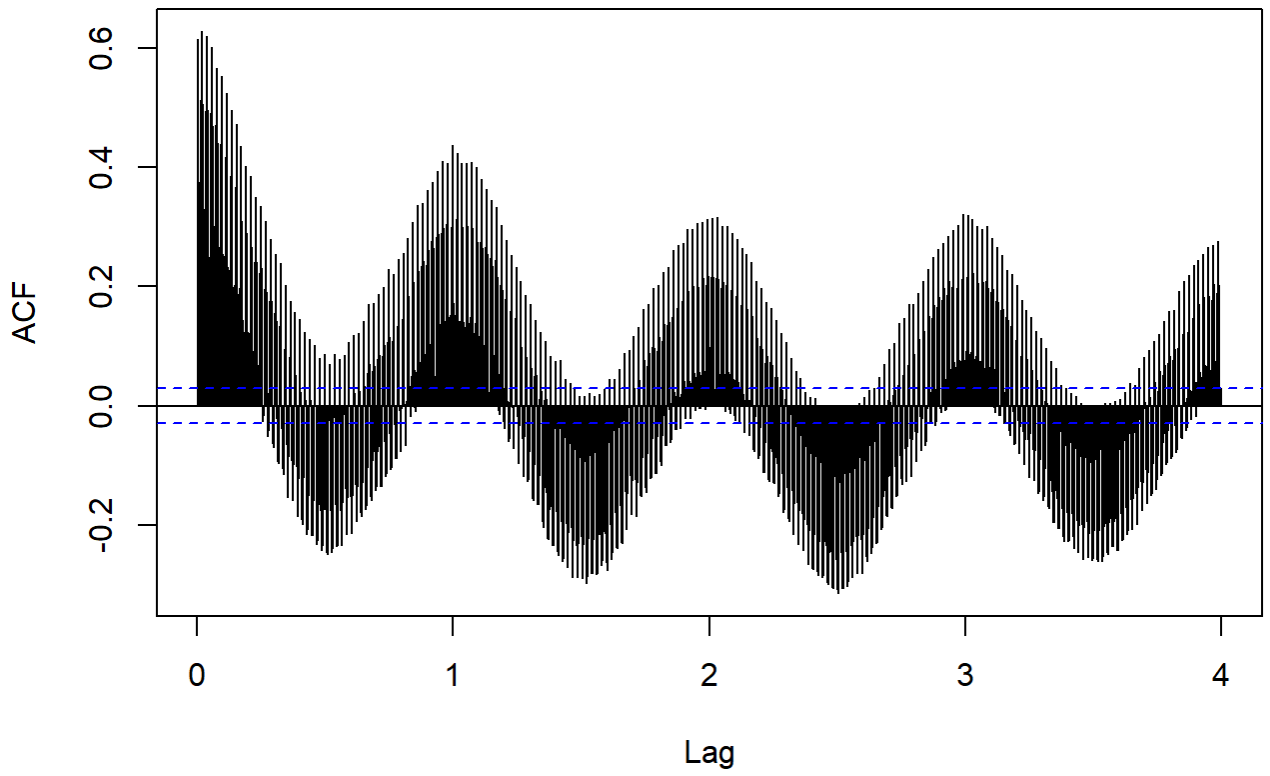
```
dif.fit.seastr = ts(nyc_v.tr - fitted(gam.fit.seastr),start=2009,frequency=365)
```

```
## Warning in `-.default`(nyc_v.tr, fitted(gam.fit.seastr)): longer object length
## is not a multiple of shorter object length
```

```
ts.plot(dif.fit.seastr, ylab = "Residuals", main = "NYC Violent Crime - Splines + Weekly Seasona
lity")
abline(h=0, col='red')
```




```
acf(dif.fit.seastr, lag.max = 365 * 4, main = "NYC Violent Crime - Splines + Weekly Seasonality
 Resid ACF")
```



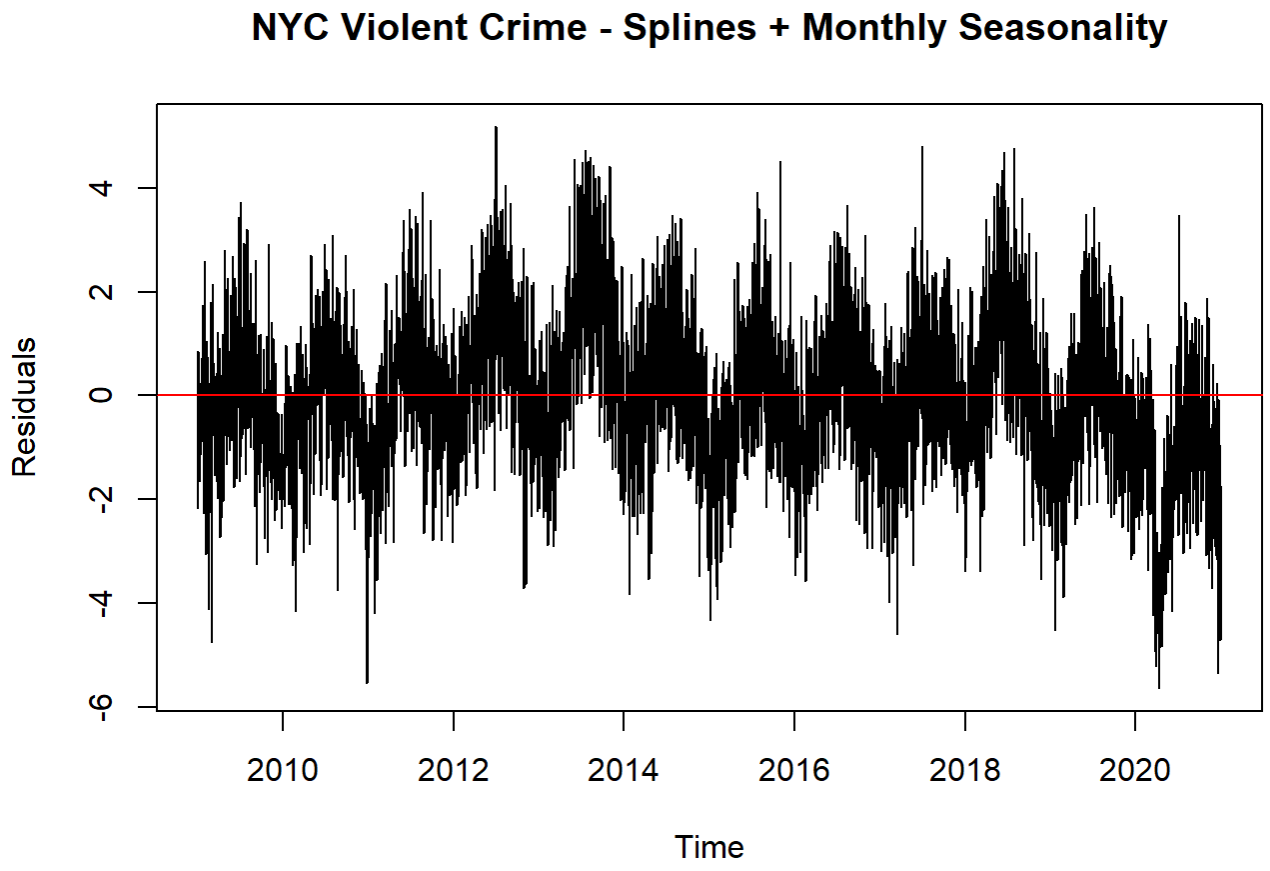

```
###TREND + SEASONALITY - SPLINES + MONTHLY SEASONALITY
nyc_violent$Date <- as.Date(nyc_violent$Date)
year <- as.factor(format(nyc_violent$Date,'%Y'))
month <- as.factor(format(nyc_violent$Date, '%b'))
week <- as.factor(weekdays(nyc_violent$Date))

gam.fit.seastr = gam(nyc_v.tr~s(time.pts)+month-1)
summary(gam.fit.seastr)
```

```
##
## Family: gaussian
## Link function: identity
##
## Formula:
## nyc_v.tr ~ s(time.pts) + month - 1
##
## Parametric coefficients:
##          Estimate Std. Error t value Pr(>|t|)
## monthApr  16.2948     0.1051   155.0   <2e-16 ***
## monthAug  16.0414     0.1051   152.6   <2e-16 ***
## monthDec  16.2429     0.1051   154.5   <2e-16 ***
## monthFeb  16.0795     0.1051   153.0   <2e-16 ***
## monthJan  17.4661     0.1051   166.2   <2e-16 ***
## monthJul  16.1408     0.1051   153.6   <2e-16 ***
## monthJun  16.0182     0.1051   152.4   <2e-16 ***
## monthMar  15.9787     0.1051   152.0   <2e-16 ***
## monthMay  16.2755     0.1051   154.8   <2e-16 ***
## monthNov  16.1498     0.1051   153.7   <2e-16 ***
## monthOct  16.1357     0.1051   153.5   <2e-16 ***
## monthSep  16.0325     0.1051   152.5   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Approximate significance of smooth terms:
##               edf Ref.df     F p-value
## s(time.pts) 7.777  8.615 15.62  <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## R-sq.(adj) =   0.14   Deviance explained = 99.4%
## GCV = 1.6093  Scale est. = 1.5908    n = 1728
```

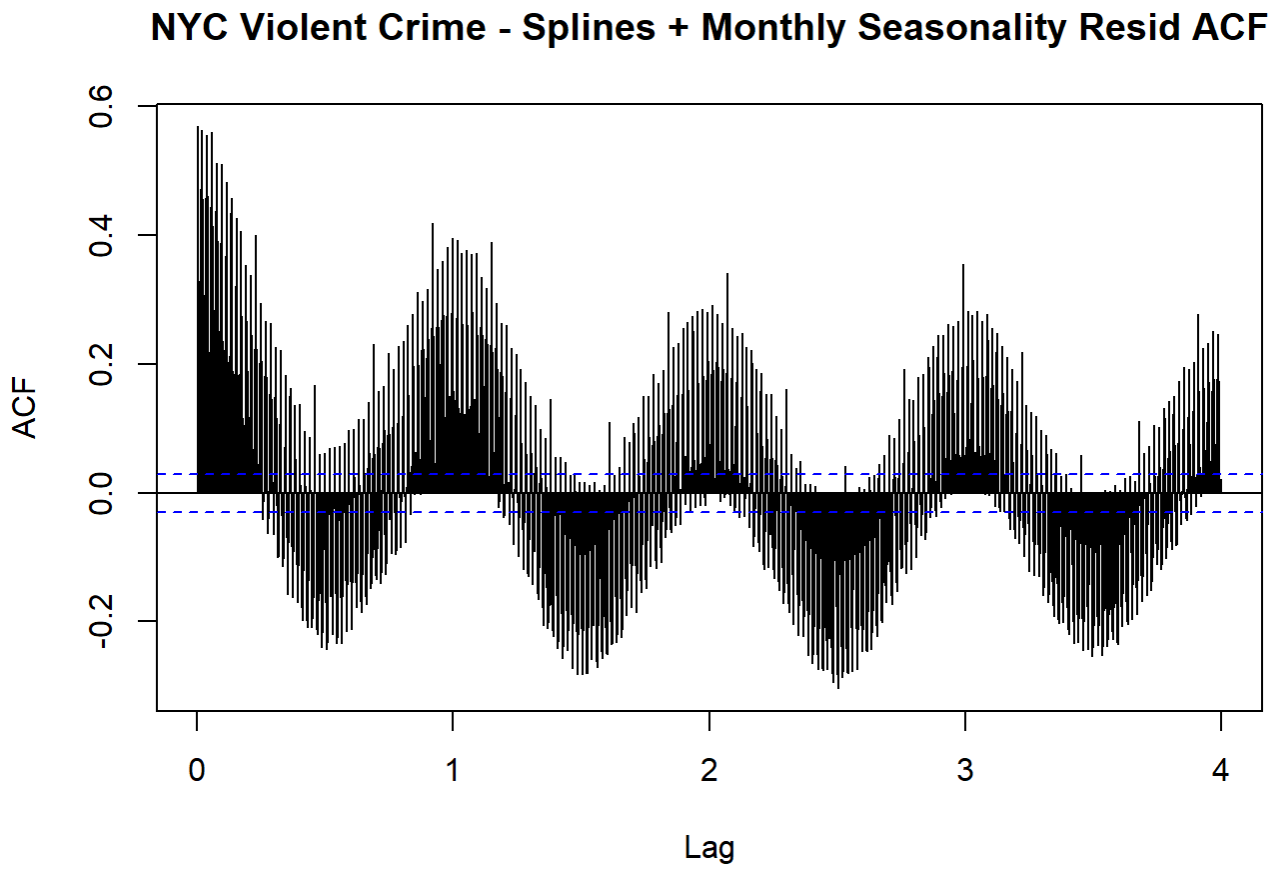
```
dif.fit.seastr = ts(nyc_v.tr - fitted(gam.fit.seastr),start=2009,frequency=365)
```

```
## Warning in `-.default`(nyc_v.tr, fitted(gam.fit.seastr)): longer object length
## is not a multiple of shorter object length
```

```
ts.plot(dif.fit.seastr, ylab = "Residuals", main = "NYC Violent Crime - Splines + Monthly Season
ality")
abline(h=0, col='red')
```




```
acf(dif.fit.seastr, lag.max = 365 * 4, main = "NYC Violent Crime - Splines + Monthly Seasonality
Resid ACF")
```



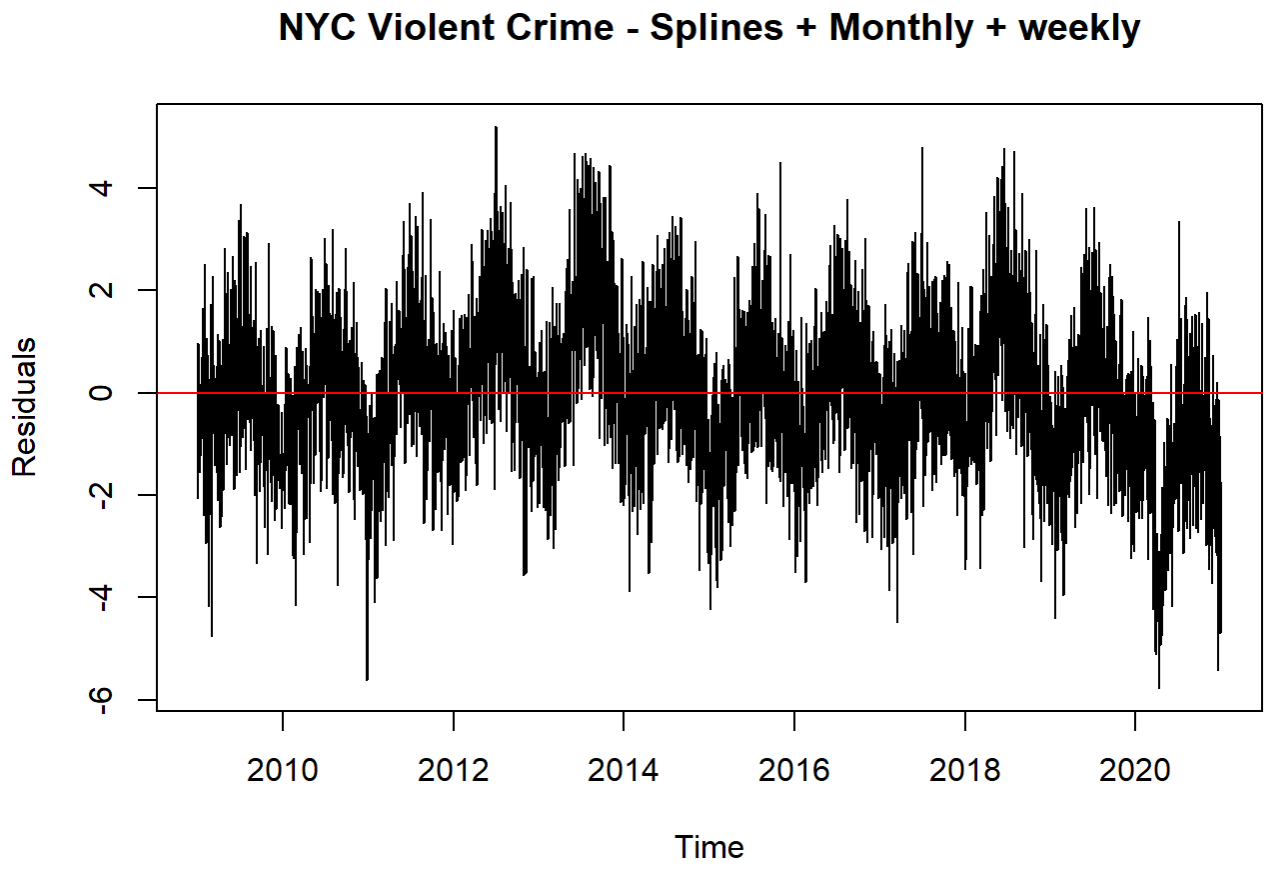

```
###TREND + SEASONALITY - SPLINES + MONTHLY + WEEKLY SEASONALITY
nyc_violent$Date <- as.Date(nyc_violent$Date)
year <- as.factor(format(nyc_violent$Date,'%Y'))
month <- as.factor(format(nyc_violent$Date, '%b'))
week <- as.factor(weekdays(nyc_violent$Date))

gam.fit.seastr.1 = gam(nyc_v.tr~s(time.pts)+month+week-1)
summary(gam.fit.seastr.1)
```

```
##
## Family: gaussian
## Link function: identity
##
## Formula:
## nyc_v.tr ~ s(time.pts) + month + week - 1
##
## Parametric coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## monthApr      16.26989    0.12389 131.324   <2e-16 ***
## monthAug      16.02488    0.12853 124.682   <2e-16 ***
## monthDec      16.23173    0.13269 122.332   <2e-16 ***
## monthFeb      16.05234    0.12788 125.523   <2e-16 ***
## monthJan      17.45136    0.12798 136.359   <2e-16 ***
## monthJul      16.11590    0.12389 130.081   <2e-16 ***
## monthJun      15.97551    0.12832 124.501   <2e-16 ***
## monthMar      15.97154    0.13242 120.609   <2e-16 ***
## monthMay      16.24034    0.12820 126.682   <2e-16 ***
## monthNov      16.14260    0.13243 121.900   <2e-16 ***
## monthOct      16.12352    0.13274 121.469   <2e-16 ***
## monthSep      16.02131    0.13269 120.747   <2e-16 ***
## weekMonday     0.02670    0.11692   0.228    0.819
## weekSaturday  -0.10049    0.11600  -0.866    0.386
## weekSunday     0.14457    0.11299   1.280    0.201
## weekThursday   0.07483    0.11490   0.651    0.515
## weekTuesday   -0.09144    0.11135  -0.821    0.412
## weekWednesday  0.09565    0.11591   0.825    0.409
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Approximate significance of smooth terms:
##               edf Ref.df     F p-value
## s(time.pts) 7.774  8.614 15.65  <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## R-sq.(adj) =  0.141   Deviance explained = 99.4%
## GCV =  1.613  Scale est. = 1.5889    n = 1728
```

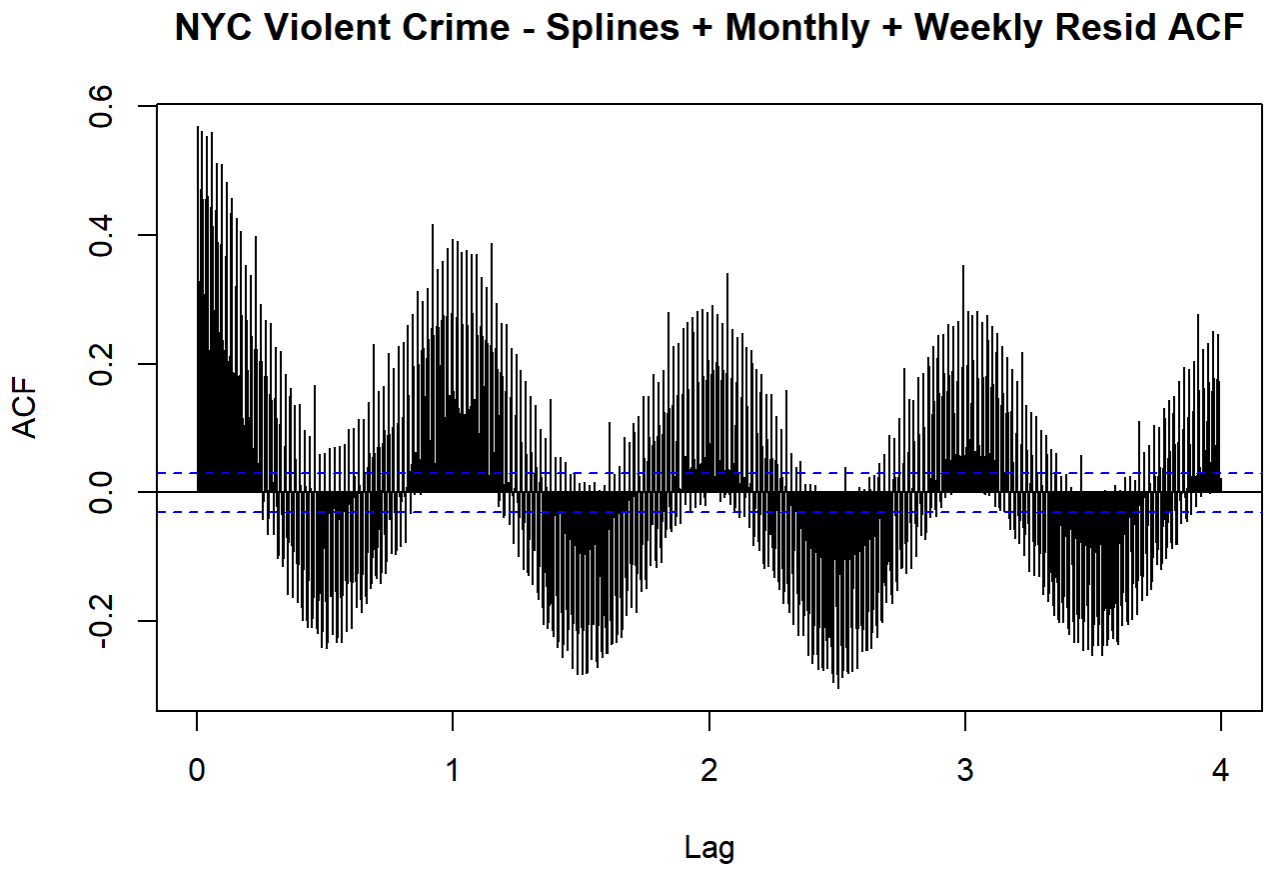
```
dif.fit.seastr.1 = ts(nyc_v.tr - fitted(gam.fit.seastr.1),start=2009,frequency=365)
```

```
## Warning in `-.default`(nyc_v.tr, fitted(gam.fit.seastr.1)): longer object length
## is not a multiple of shorter object length
```

```
ts.plot(dif.fit.seastr.1, ylab = "Residuals", main = "NYC Violent Crime - Splines + Monthly + we
ekly")
abline(h=0, col='red')
```




```
acf(dif.fit.seastr.1, lag.max = 365 * 4, main = "NYC Violent Crime - Splines + Monthly + Weekly
 Resid ACF")
```



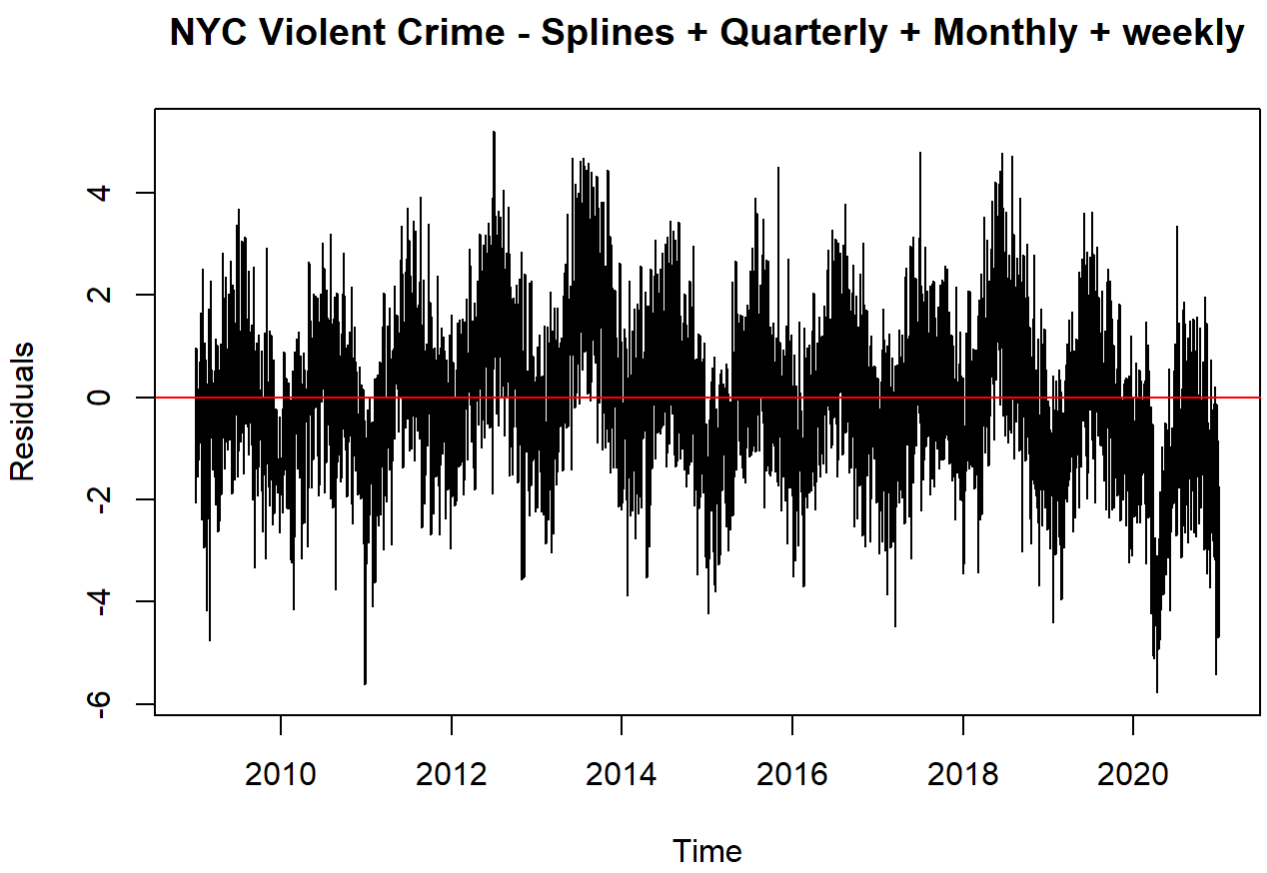

```
###TREND + SEASONALITY - SPLINES + QUARTERLY + MONTHLY + WEEKLY SEASONALITY
nyc_violent$Date <- as.Date(nyc_violent$Date)
year <- as.factor(format(nyc_violent$Date,'%Y'))
month <- as.factor(format(nyc_violent$Date, '%b'))
week <- as.factor(weekdays(nyc_violent$Date))
quarterly <- as.factor(quarter(nyc_violent$Date))

gam.fit.seastr.1 = gam(nyc_v.tr~s(time.pts)+quarterly+month+week-1)
summary(gam.fit.seastr.1)
```

```
##
## Family: gaussian
## Link function: identity
##
## Formula:
## nyc_v.tr ~ s(time.pts) + quarterly + month + week - 1
##
## Parametric coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## quarterly1    12.36881    0.07247 170.671   <2e-16 ***
## quarterly2    16.26989    0.12389 131.324   <2e-16 ***
## quarterly3    12.04052    0.07110 169.354   <2e-16 ***
## quarterly4    12.12446    0.07556 160.462   <2e-16 ***
## monthAug       3.98436    0.08967  44.432   <2e-16 ***
## monthDec       4.10727    0.08970  45.788   <2e-16 ***
## monthFeb       3.68353    0.08904  41.369   <2e-16 ***
## monthJan       5.08255    0.08879  57.241   <2e-16 ***
## monthJul       4.07538    0.08831  46.150   <2e-16 ***
## monthJun      -0.29439    0.14918  -1.973   0.0486 *
## monthMar       3.60273    0.09053  39.794   <2e-16 ***
## monthMay      -0.02956    0.14918  -0.198   0.8430
## monthNov       4.01814    0.08985  44.718   <2e-16 ***
## monthOct       3.99906    0.08997  44.448   <2e-16 ***
## monthSep       3.98079    0.09143  43.537   <2e-16 ***
## weekMonday     0.02670    0.11692   0.228   0.8194
## weekSaturday  -0.10049    0.11600  -0.866   0.3865
## weekSunday     0.14457    0.11299   1.280   0.2009
## weekThursday   0.07483    0.11490   0.651   0.5150
## weekTuesday   -0.09144    0.11135  -0.821   0.4116
## weekWednesday  0.09565    0.11591   0.825   0.4094
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Approximate significance of smooth terms:
##               edf Ref.df     F p-value
## s(time.pts) 7.774  8.614 15.65  <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Rank: 27/30
## R-sq.(adj) =  0.141   Deviance explained = 99.4%
## GCV =  1.613  Scale est. = 1.5889    n = 1728
```

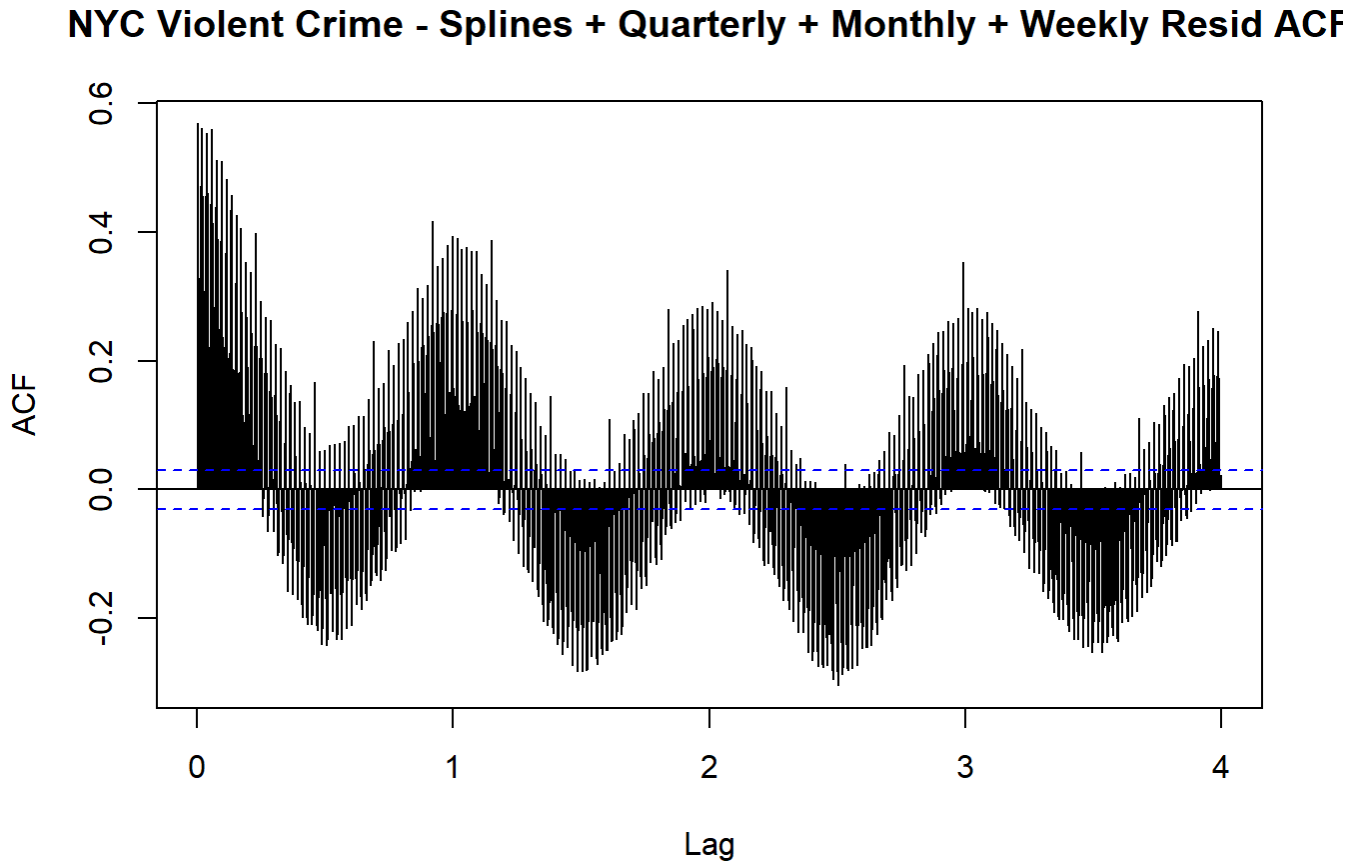
```
dif.fit.seastr.1 = ts(nyc_v.tr - fitted(gam.fit.seastr.1),start=2009,frequency=365)
```

```
## Warning in `-.default`(nyc_v.tr, fitted(gam.fit.seastr.1)): longer object length
## is not a multiple of shorter object length
```

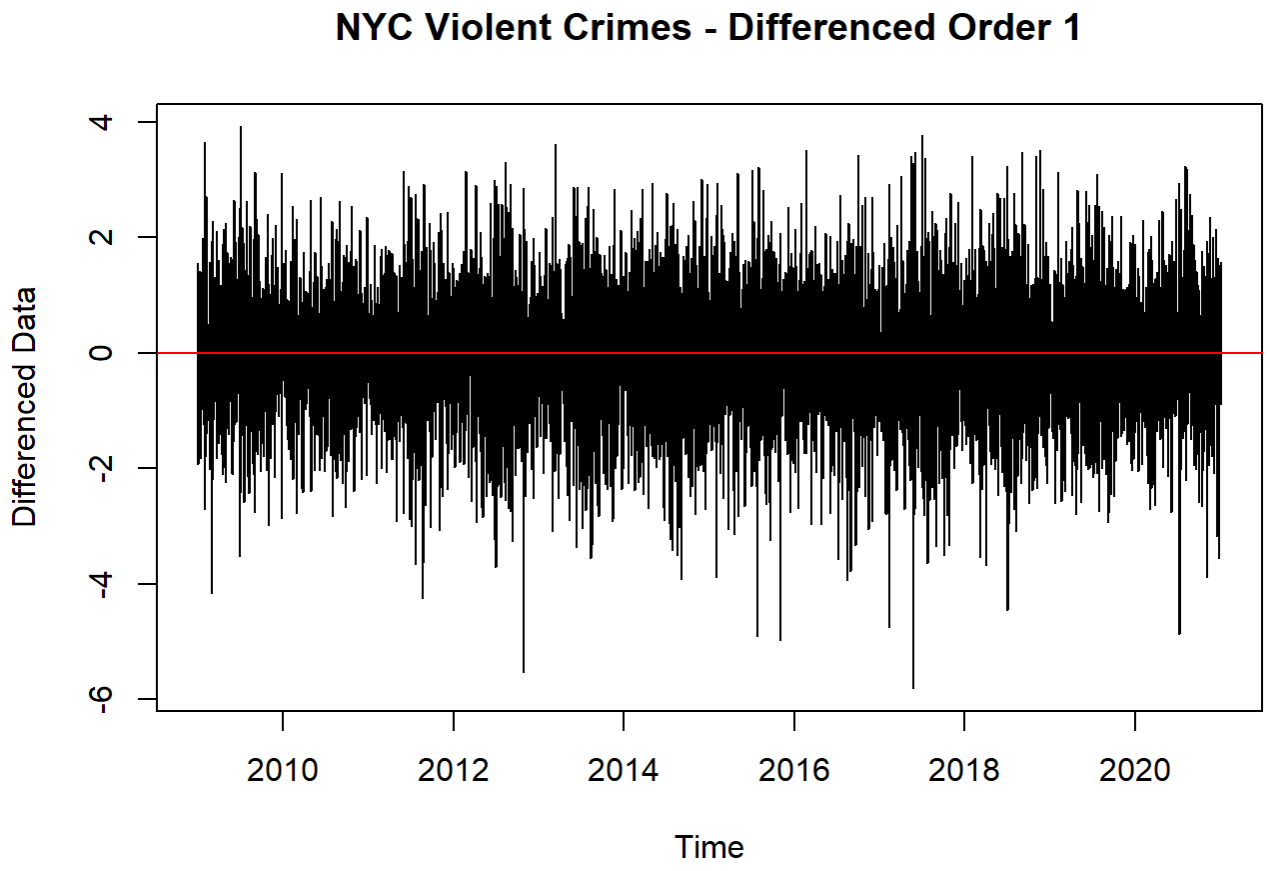
```
ts.plot(dif.fit.seastr.1, ylab = "Residuals", main = "NYC Violent Crime - Splines + Quarterly +
 Monthly + weekly")
abline(h=0, col='red')
```



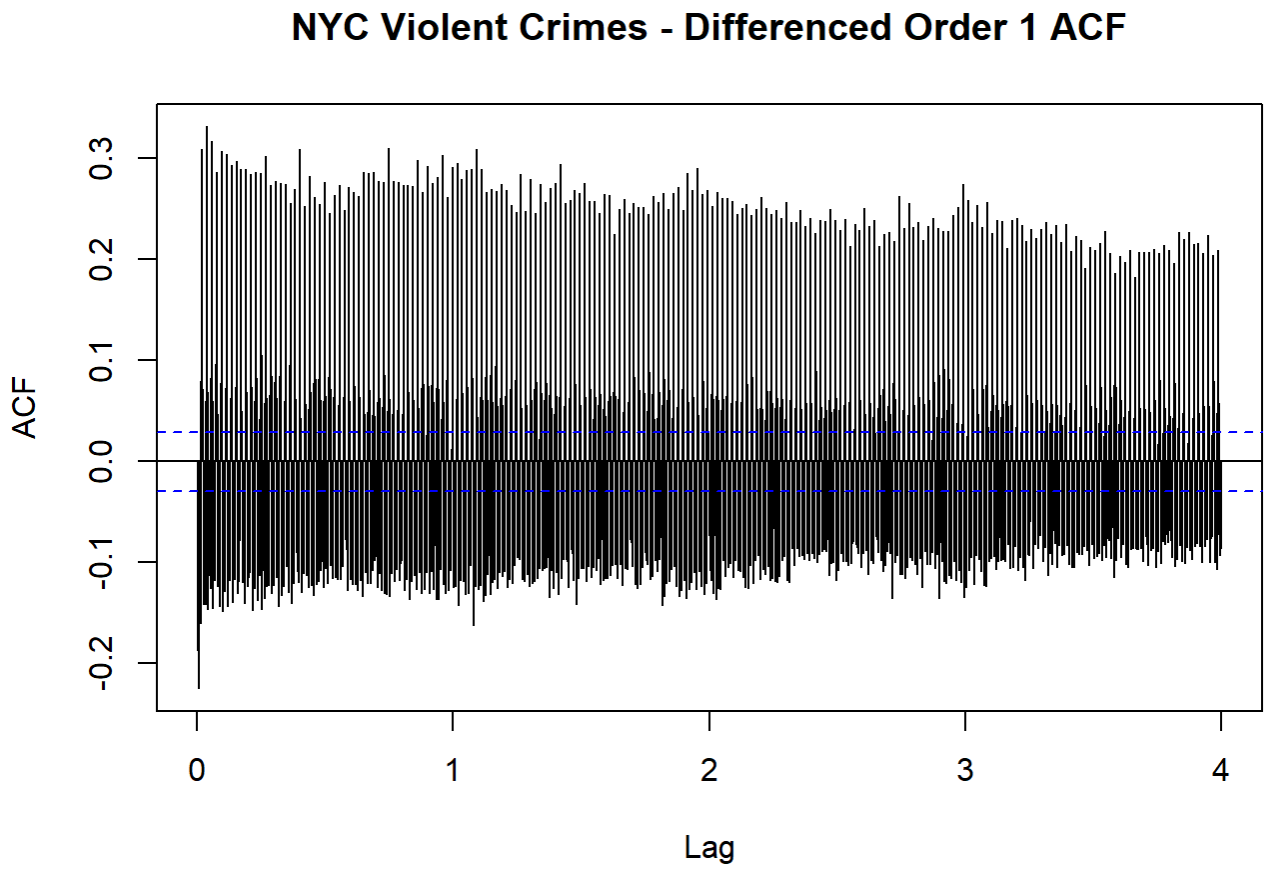

```
acf(dif.fit.seastr.1, lag.max = 365 * 4, main = "NYC Violent Crime - Splines + Quarterly + Month
ly + Weekly Resid ACF")
```



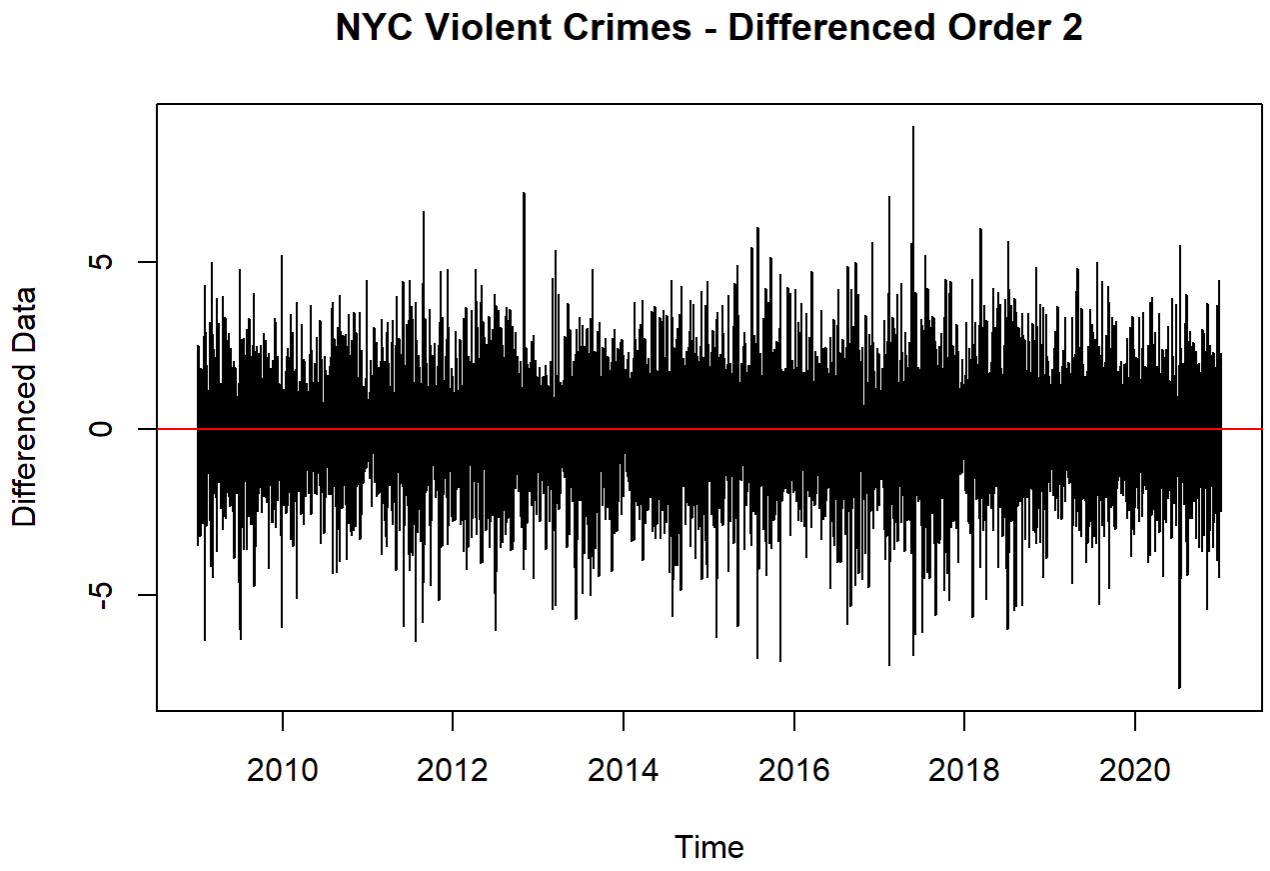

```
###DIFFERENCED DATA - ORDER 1
diff.nyc_v = diff(nyc_v.tr)
ts.plot(diff.nyc_v, ylab = "Differenced Data", main = "NYC Violent Crimes - Differenced Order 1"
)
abline(h=0, col='red')
```



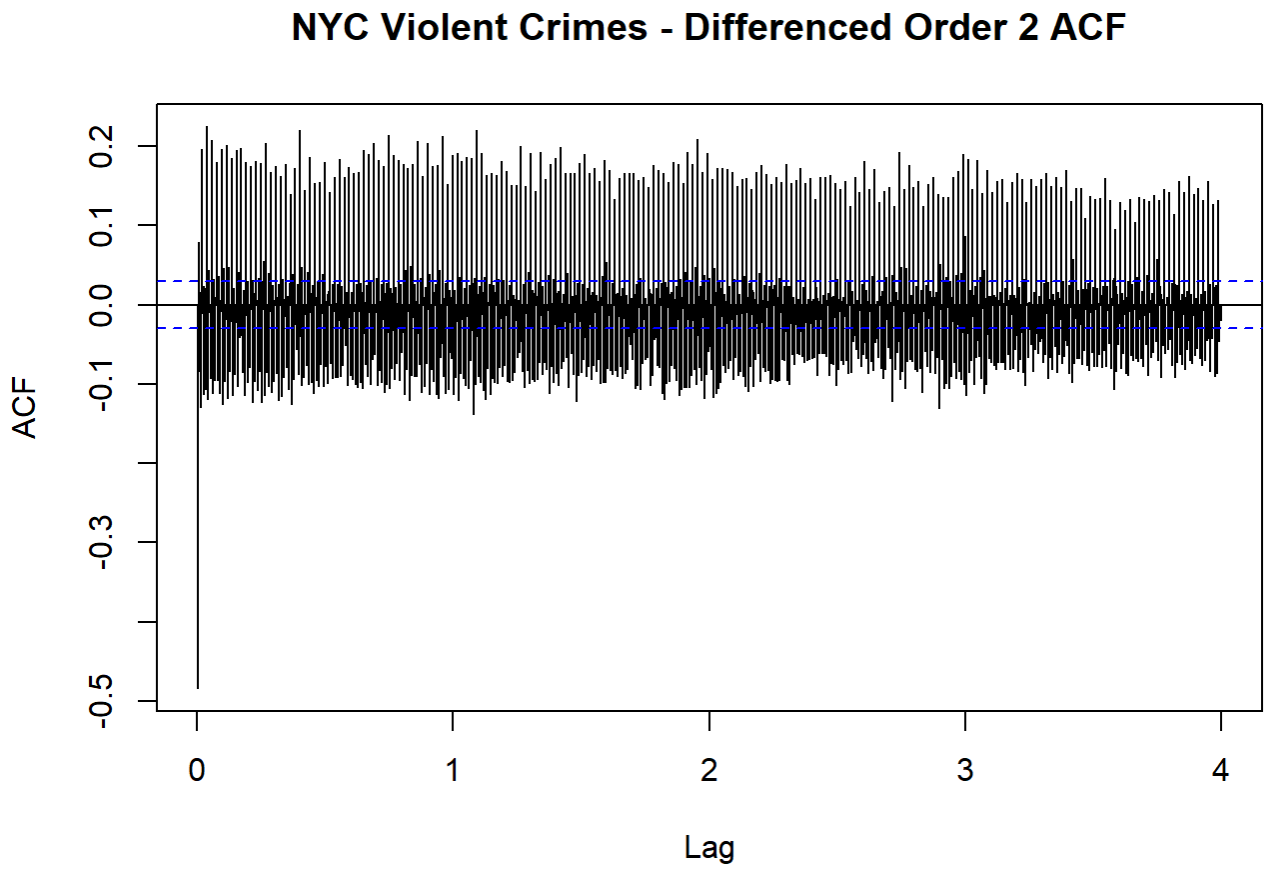

```
acf(diff.nyc_v, lag.max = 365 * 4, main = "NYC Violent Crimes - Differenced Order 1 ACF")
```



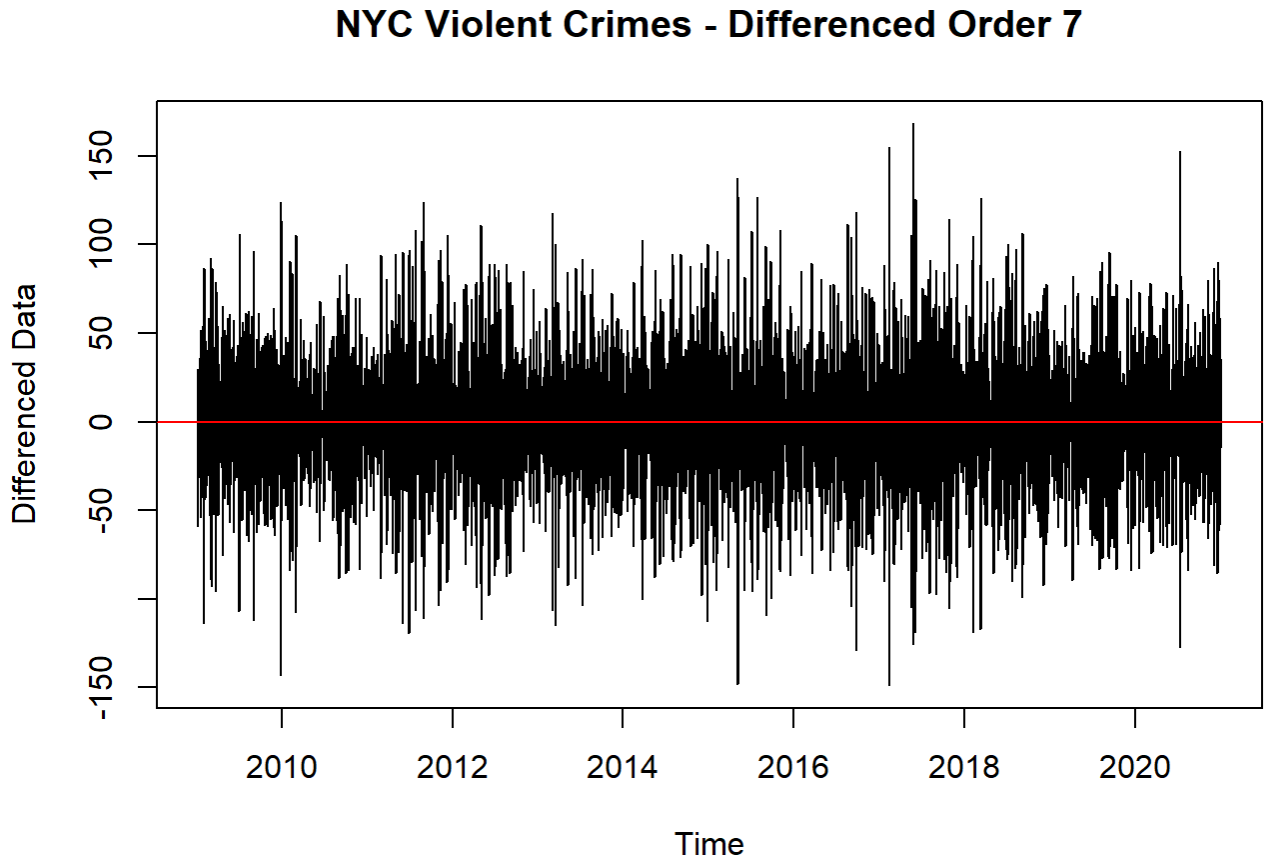

```
###DIFFERENCED DATA - ORDER 2
diff.nyc_v = diff(nyc_v.tr, differences = 2)
ts.plot(diff.nyc_v, ylab = "Differenced Data", main = "NYC Violent Crimes - Differenced Order 2"
)
abline(h=0, col='red')
```



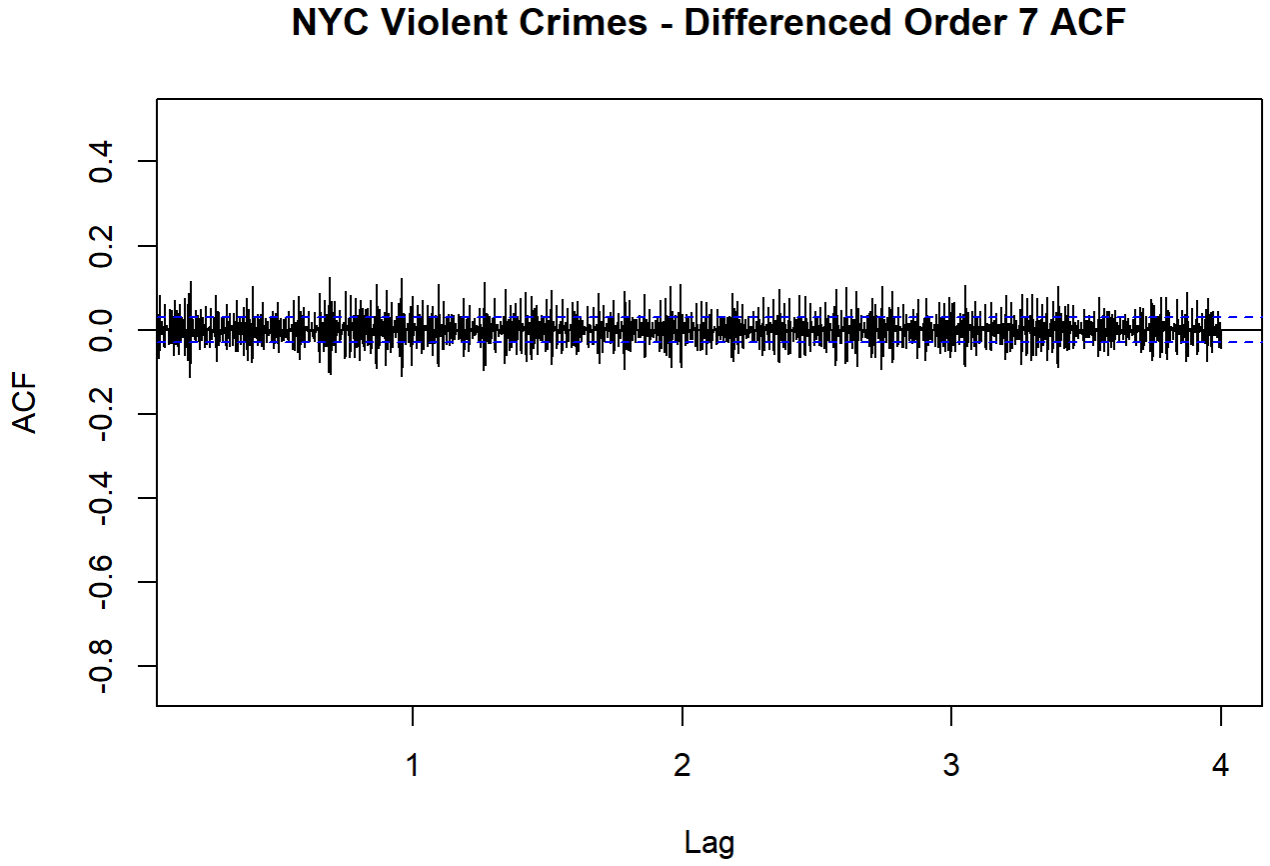

```
acf(diff.nyc_v, lag.max = 365 * 4, main = "NYC Violent Crimes - Differenced Order 2 ACF")
```



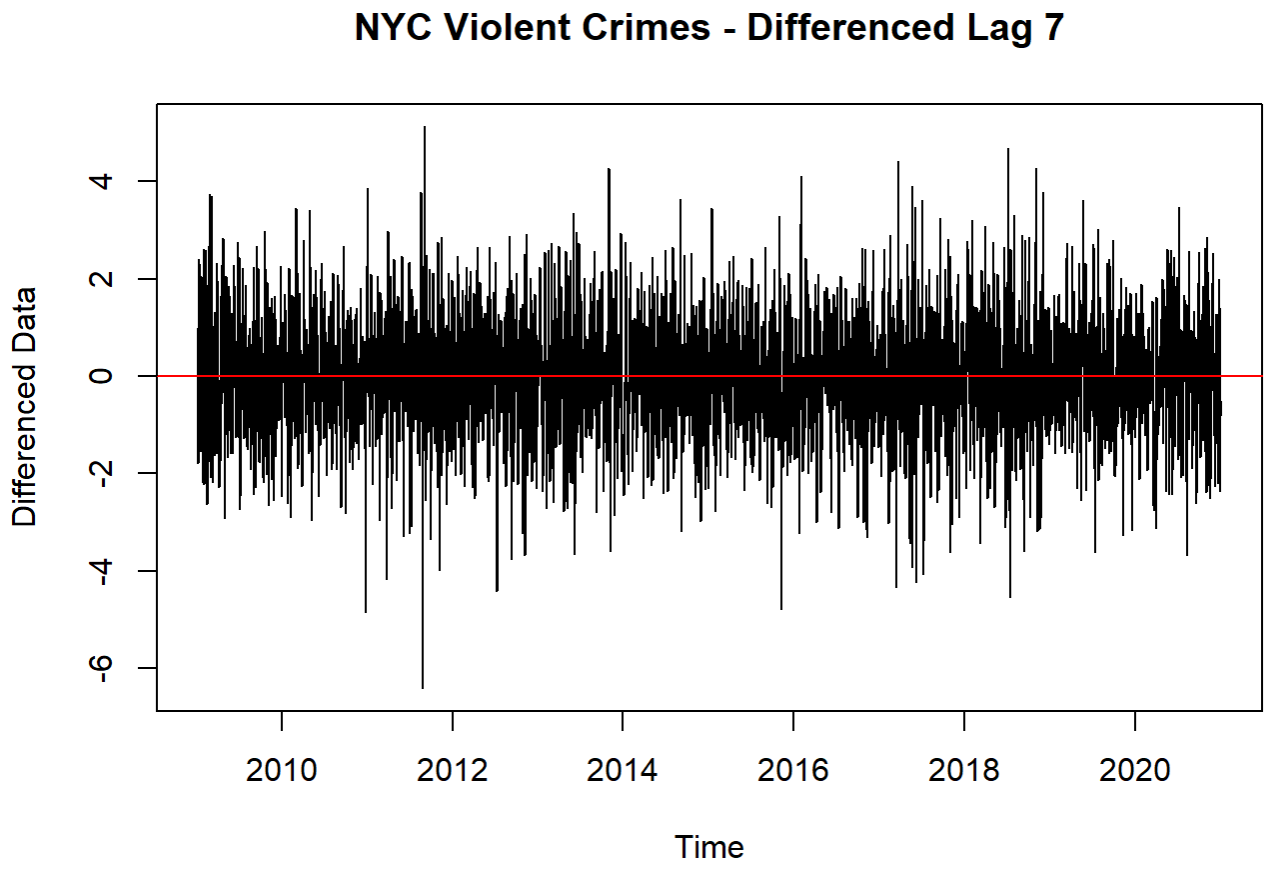

```
###DIFFERENCED DATA - ORDER 7
diff.nyc_v = diff(nyc_v.tr, differences = 7)
ts.plot(diff.nyc_v, ylab = "Differenced Data", main = "NYC Violent Crimes - Differenced Order 7"
)
abline(h=0, col='red')
```



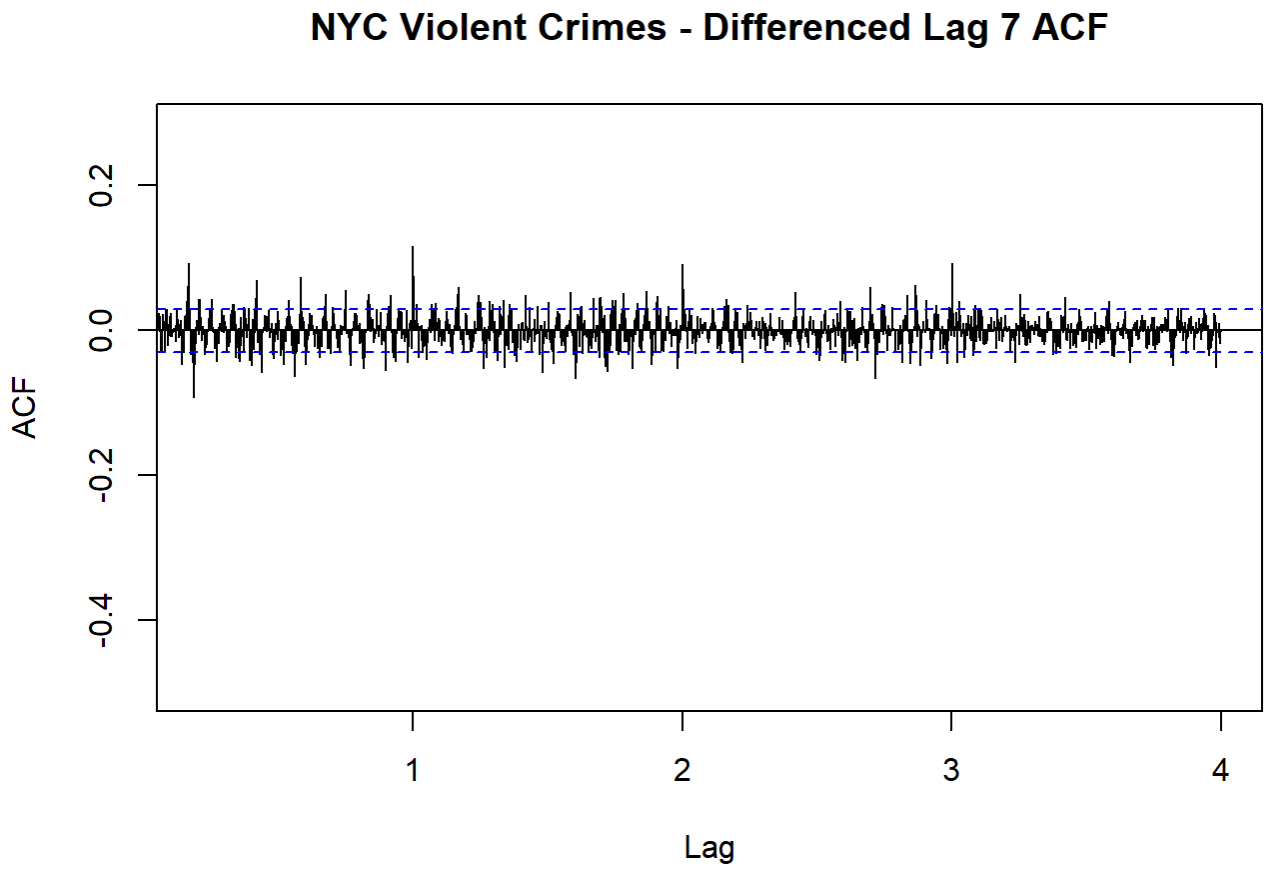

```
acf(diff.nyc_v, xlim=c(0.2,4), lag.max = 365 * 4, main = "NYC Violent Crimes - Differenced Order
7 ACF")
```




```
###DIFFERENCED DATA - LAG 7
diff.crime = diff(nyc_v.tr, 7)
ts.plot(diff.crime, ylab = "Differenced Data", main = "NYC Violent Crimes - Differenced Lag 7")
abline(h=0, col='red')
```

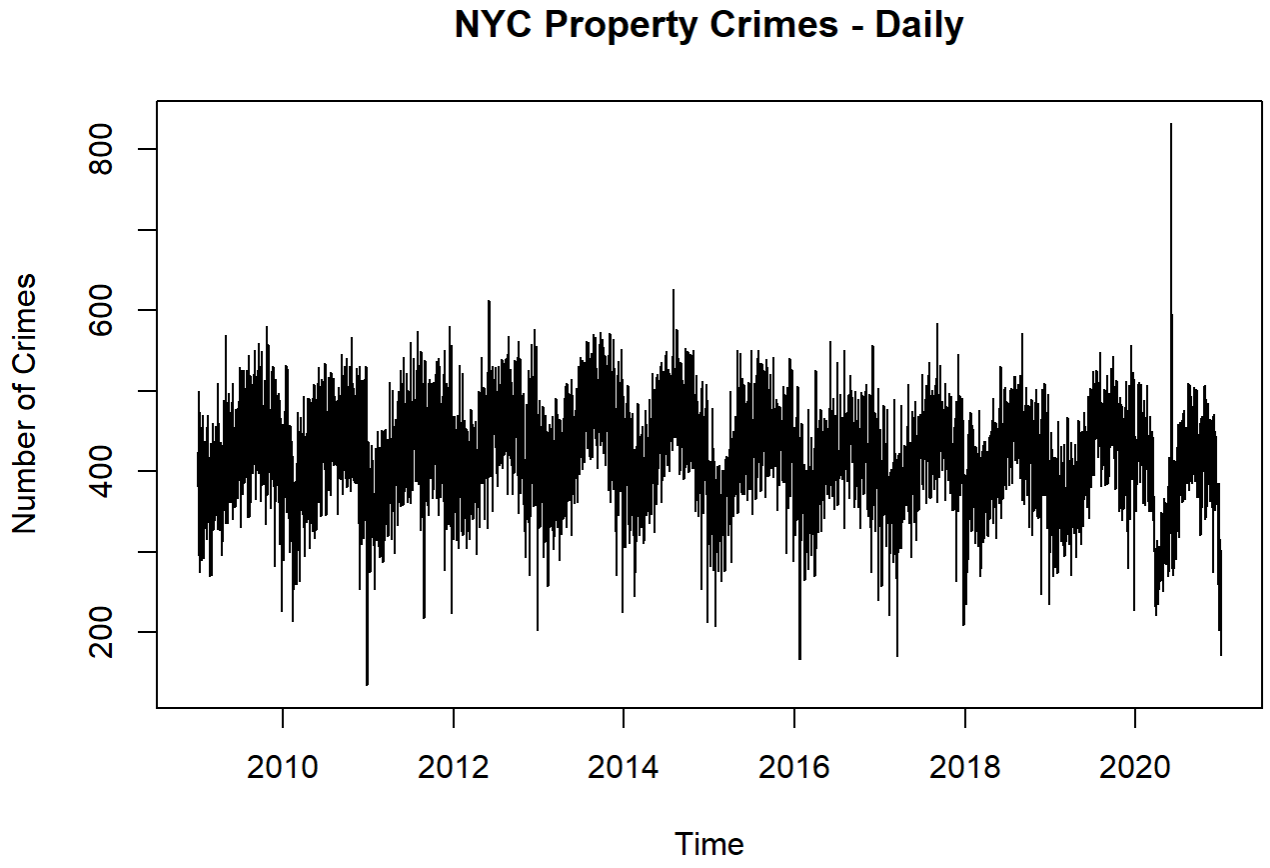



```
acf(diff.crime, xlim=c(0.2,4),lag.max = 365 * 4, main = "NYC Violent Crimes - Differenced Lag 7
 ACF")
```

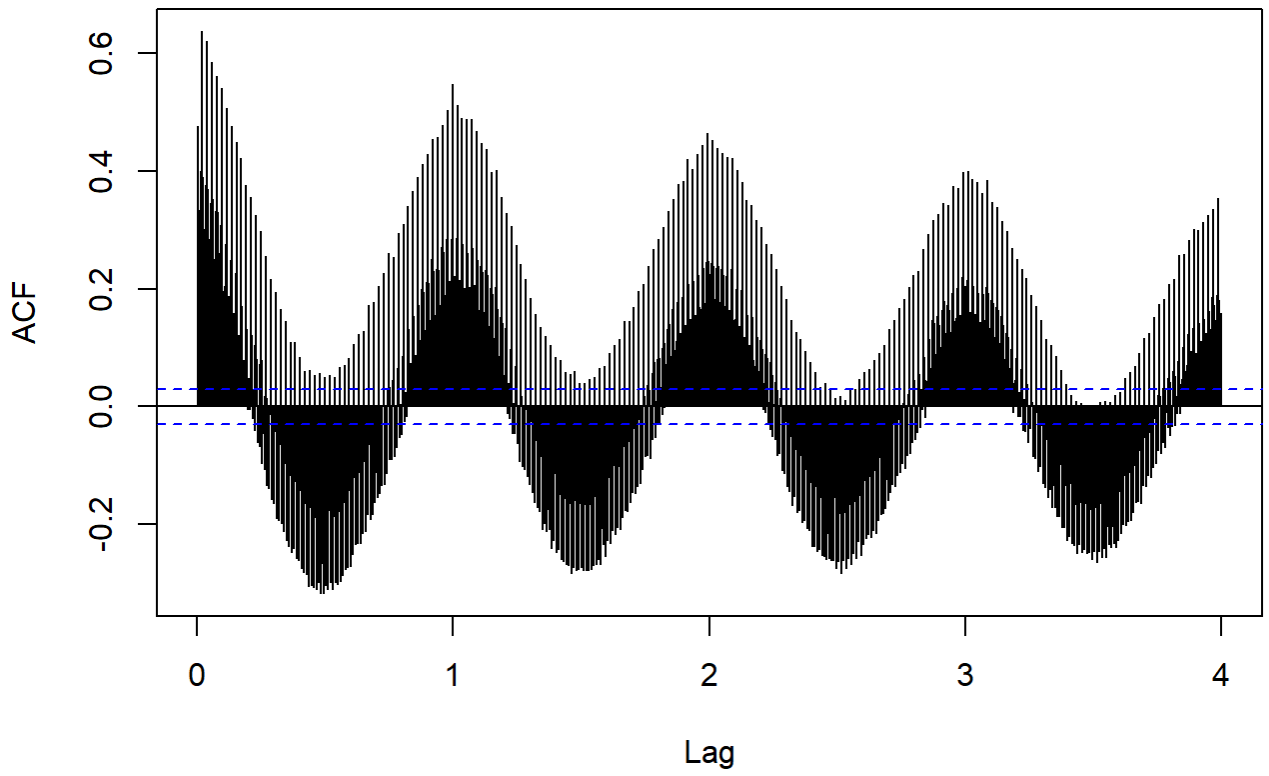


NYC Property Crime

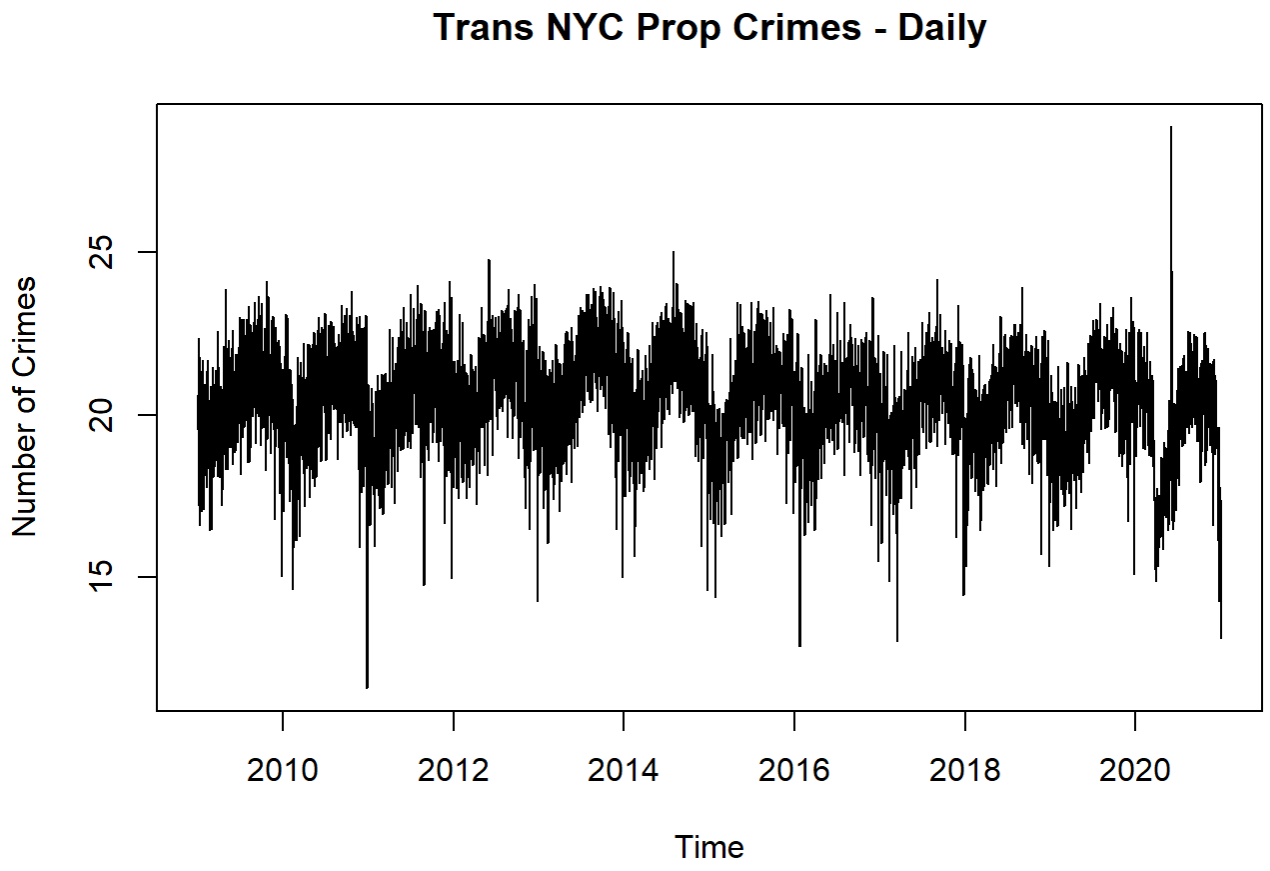
```
###EDA - Time Series / ACF
ts.plot(nyc_p,ylab="Number of Crimes",main="NYC Property Crimes - Daily")
```



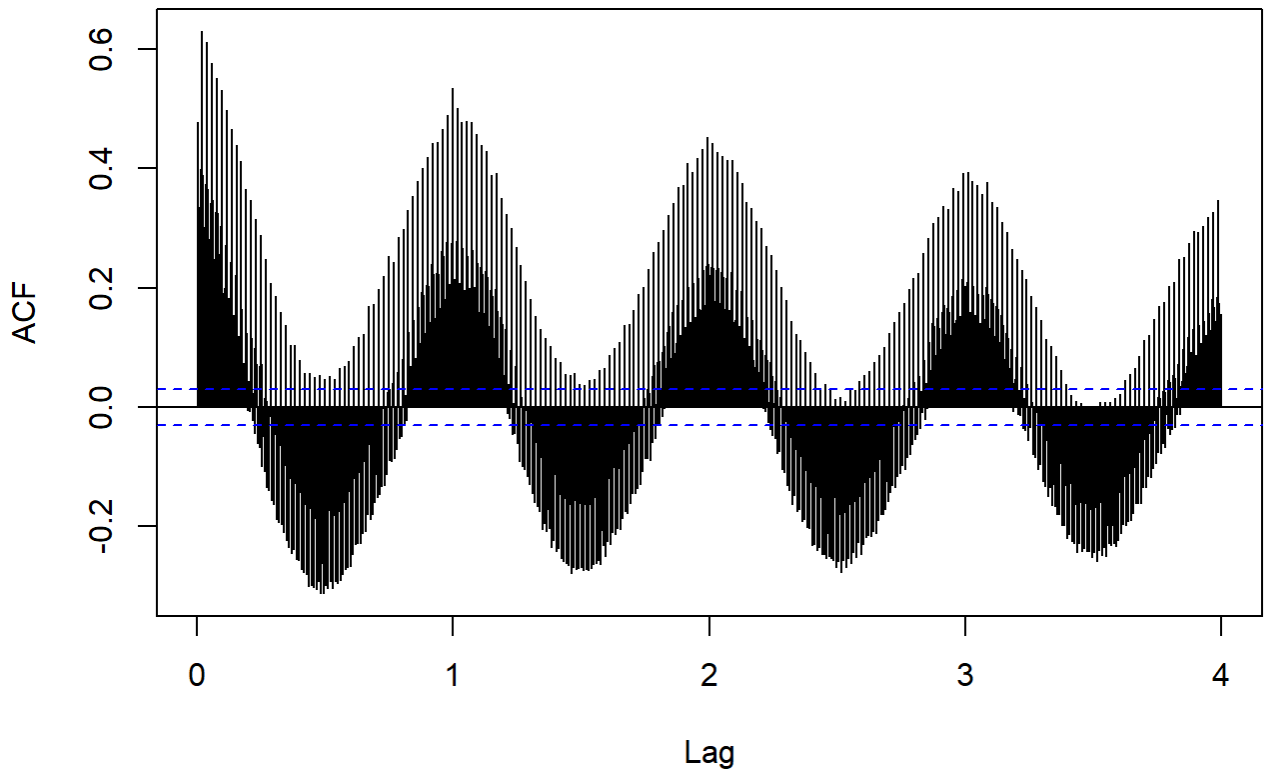

```
acf(nyc_p,lag.max=365*4,main="NYC Property Crimes - ACF")
```



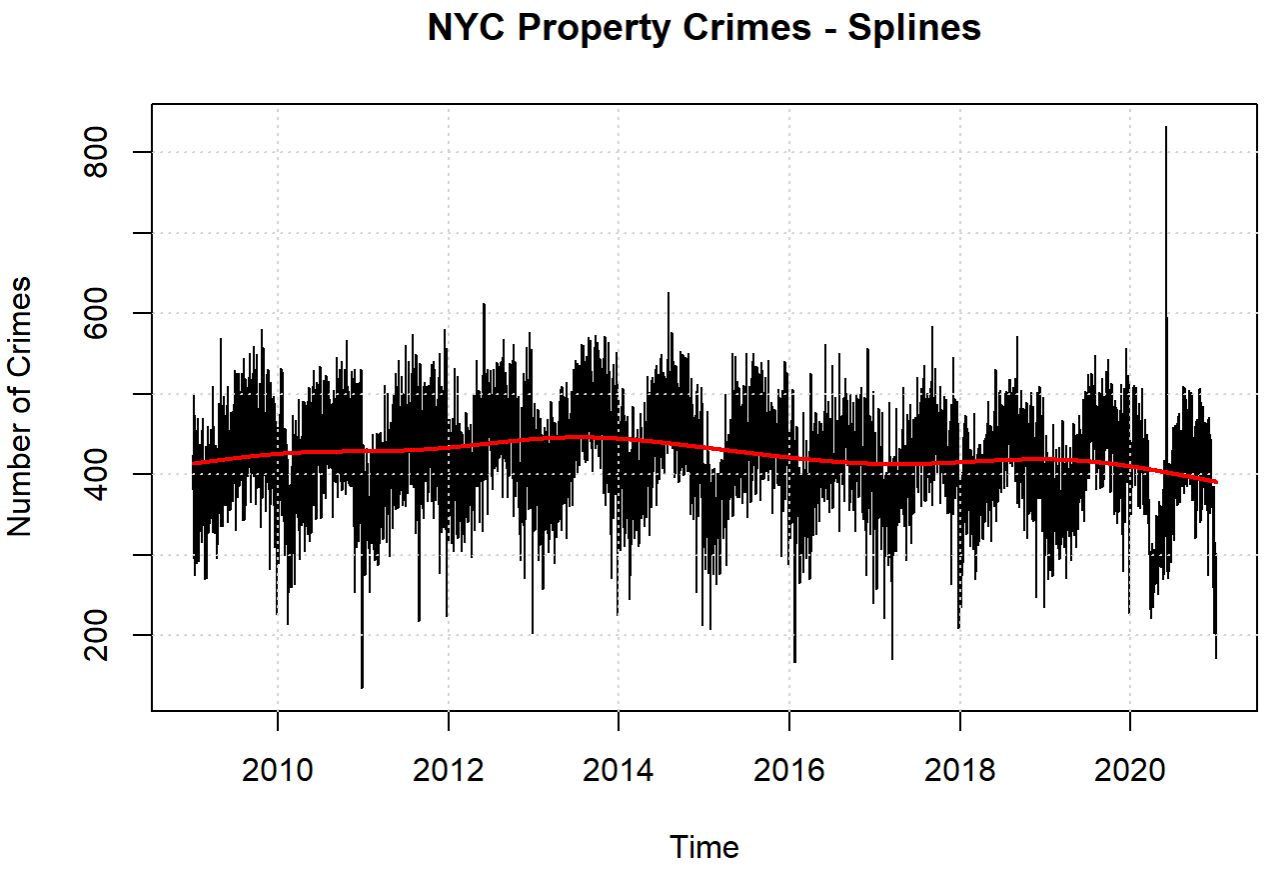

```
ts.plot(nyc_p.tr,ylab="Number of Crimes",main="Trans NYC Prop Crimes - Daily")
```



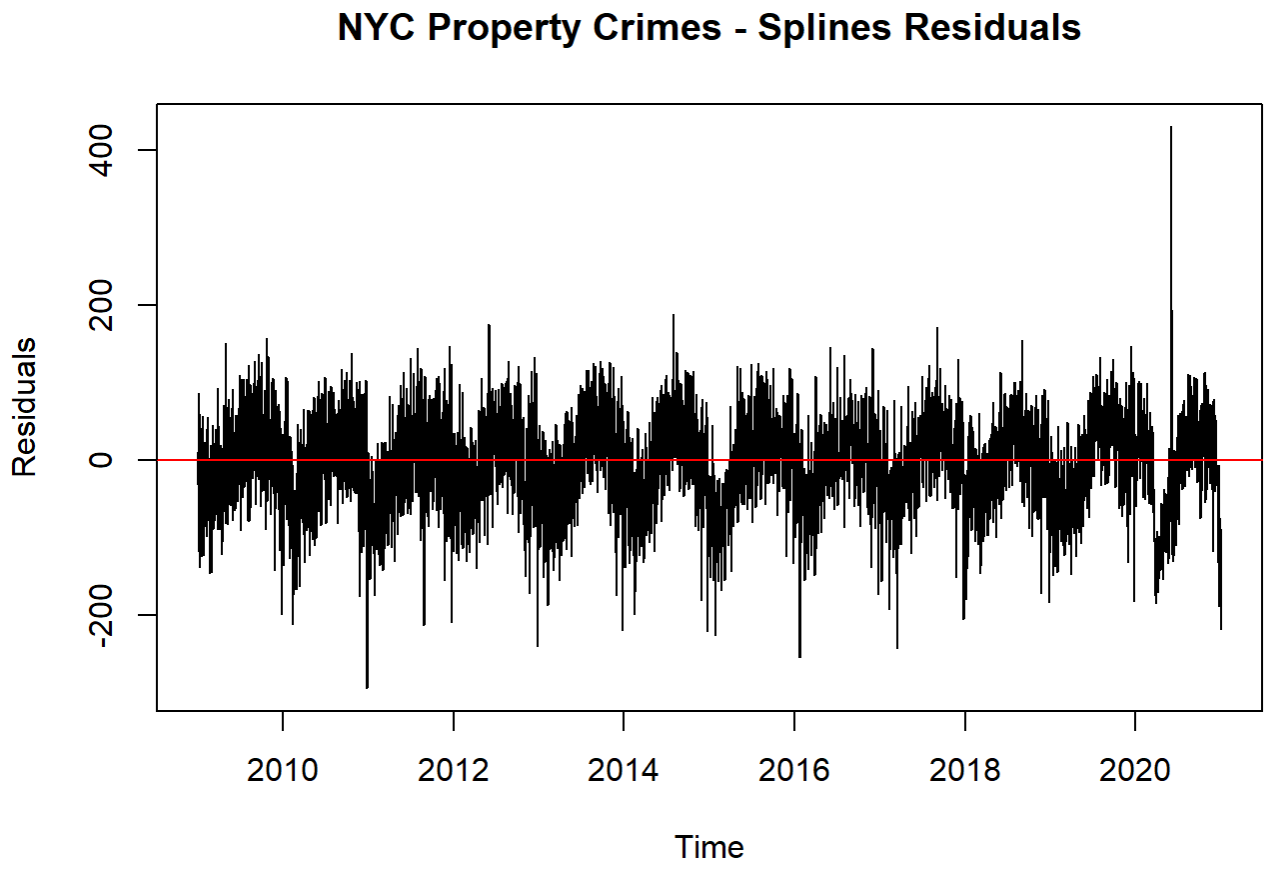

```
acf(nyc_p.tr,lag.max=365*4,main="Trans NYC Prop Crimes - ACF")
```



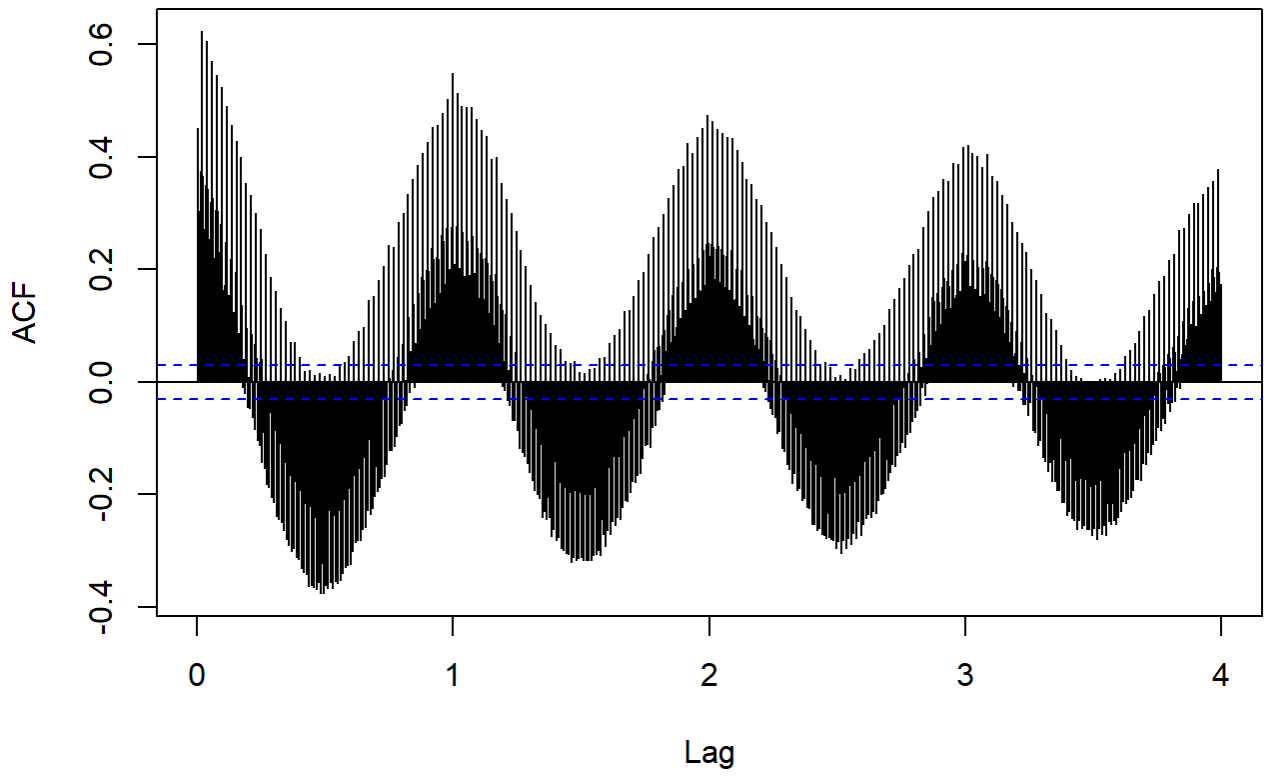

```
###TREND ESTIMATION - SPLINES
# Convert X-Axis to 0-1 Scale
time.pts = c(1:length(nyc_p))
time.pts = c(time.pts - min(time.pts))/max(time.pts)

#Splines Trend Estimation
gam.fit = gam(nyc_p~s(time.pts))
nyc_p.fit.gam = ts(fitted(gam.fit),start=2009,frequency=365)

##Is there a trend?
ts.plot(nyc_p,ylab="Number of Crimes", main = "NYC Property Crimes - Splines")
grid()
lines(nyc_p.fit.gam,lwd=2,col="red")
```



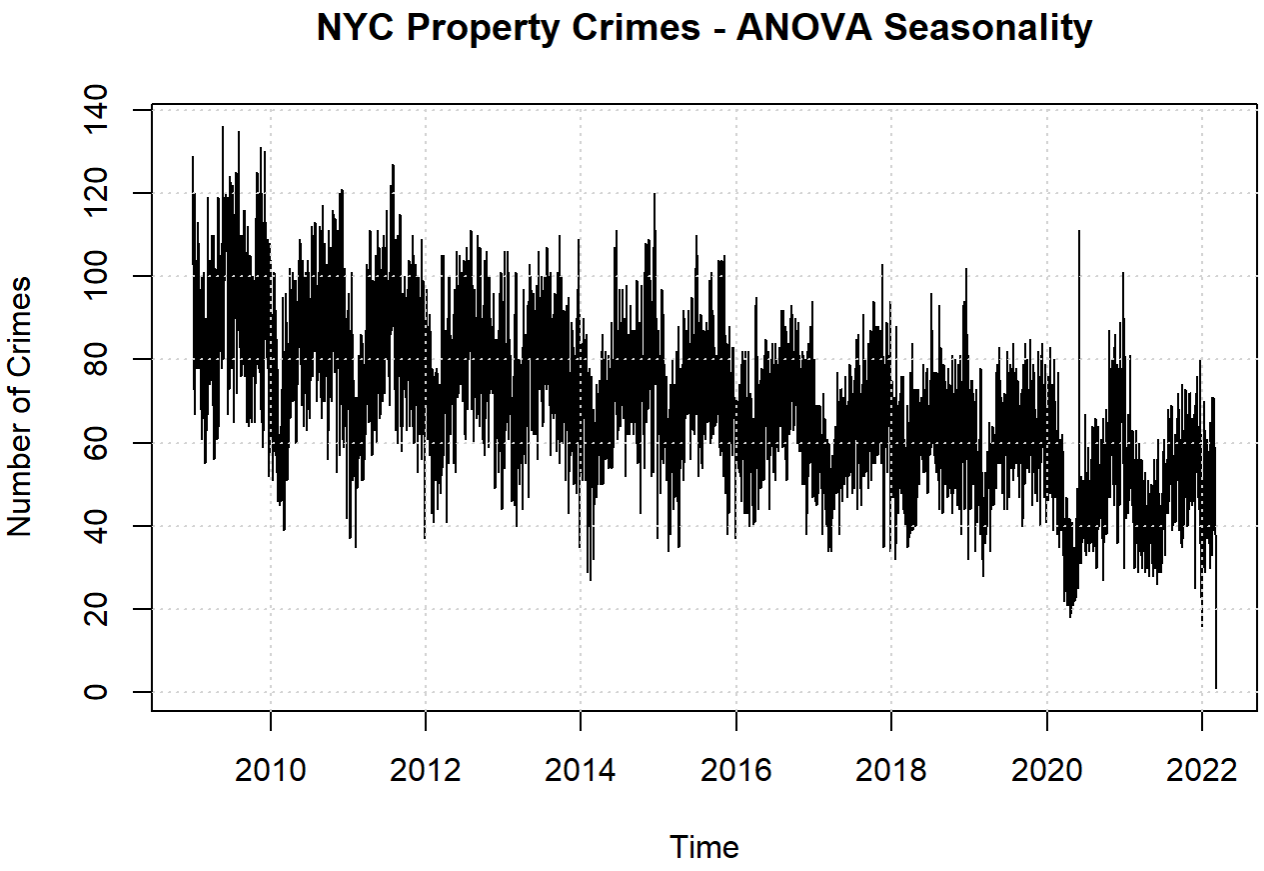

```
# Splines Residuals + Residuals ACF
dif.fit.gam = nyc_p - nyc_p.fit.gam
ts.plot(dif.fit.gam, ylab = "Residuals", main = "NYC Property Crimes - Splines Residuals")
abline(h=0, col='red')
```



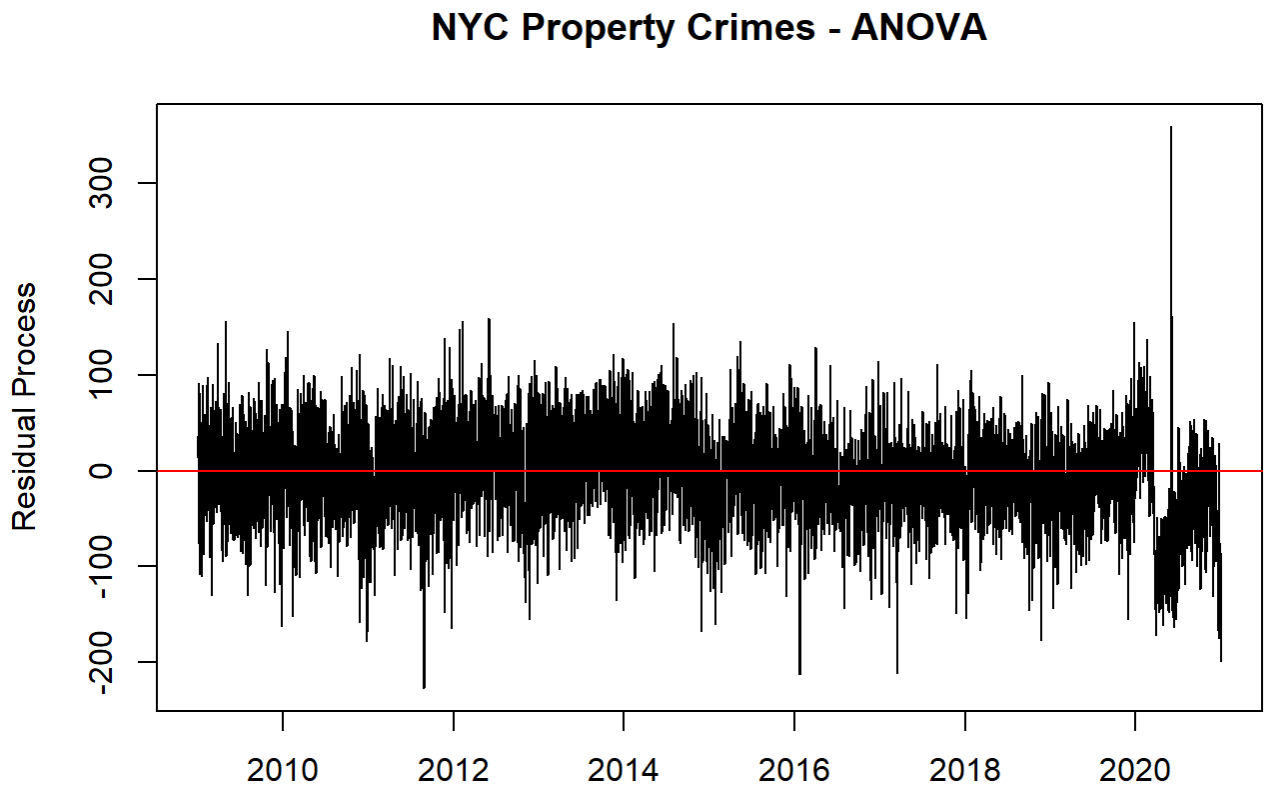

```
acf(dif.fit.gam, lag.max = 365 * 4, main = "NYC Property Crimes - Splines Residuals - ACF")
```



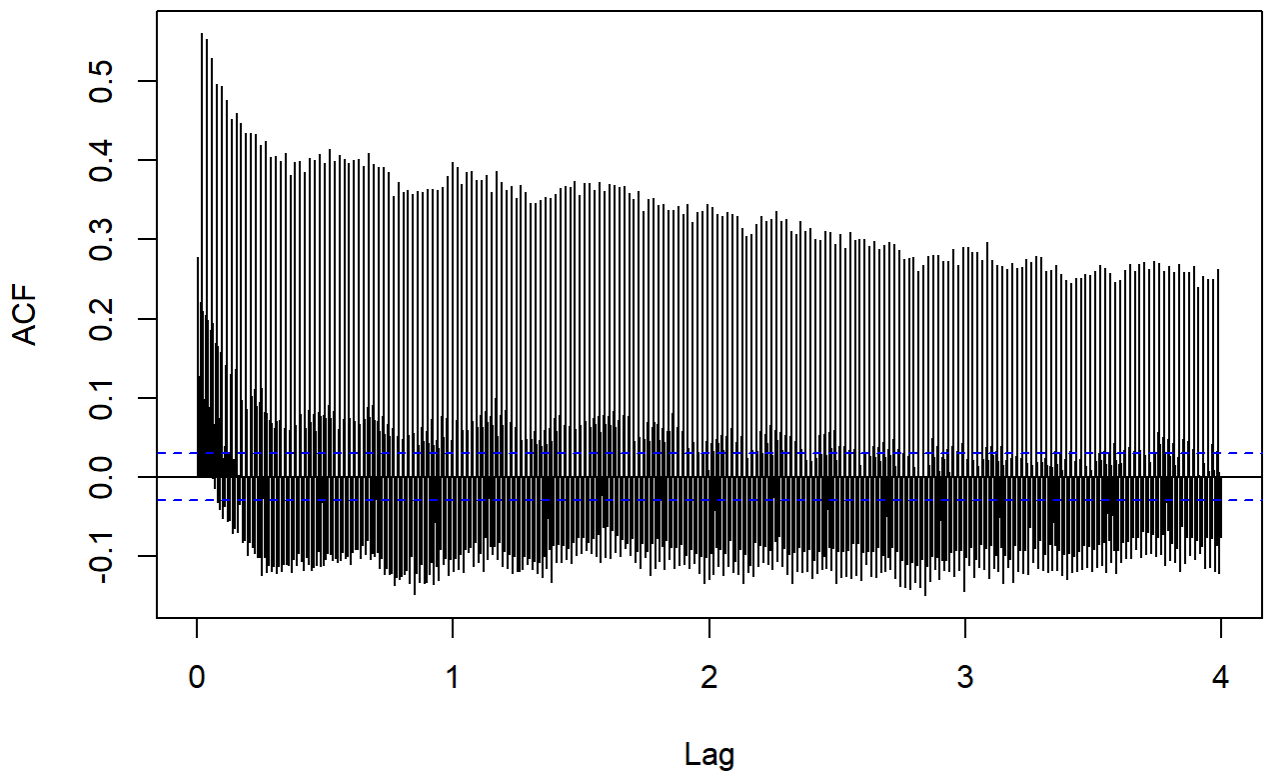

```
###SEASONALITY ANALYSIS - ANOVA

## Estimate seasonality using ANOVA approach
model.anova = dynlm(nyc_p~season(nyc_p))
#summary(model.anova)

## Plot
ts.plot(atl_p,ylab="Number of Crimes", main = "NYC Property Crimes - ANOVA Seasonality")
grid()
lines(fitted(model.anova),lwd=2,col="red")
```




```
###Seasonality ANOVA- Residuals + Resid ACF
resid.anova <- residuals(model.anova)

ts.plot(resid.anova, xlab = "", ylab = "Residual Process", main = "NYC Property Crimes - ANOVA")
abline(h=0, col='red')
```




```
acf(resid.anova, lag.max = 365 * 4, main = "NYC Property Crimes - ANOVA ACF")
```




```
###SEASONALITY ANALYSIS - HARMONIC
###SUMMARY
harmonic.1 = dynlm(nyc_p~harmon(nyc_p))
summary(harmonic.1)
```

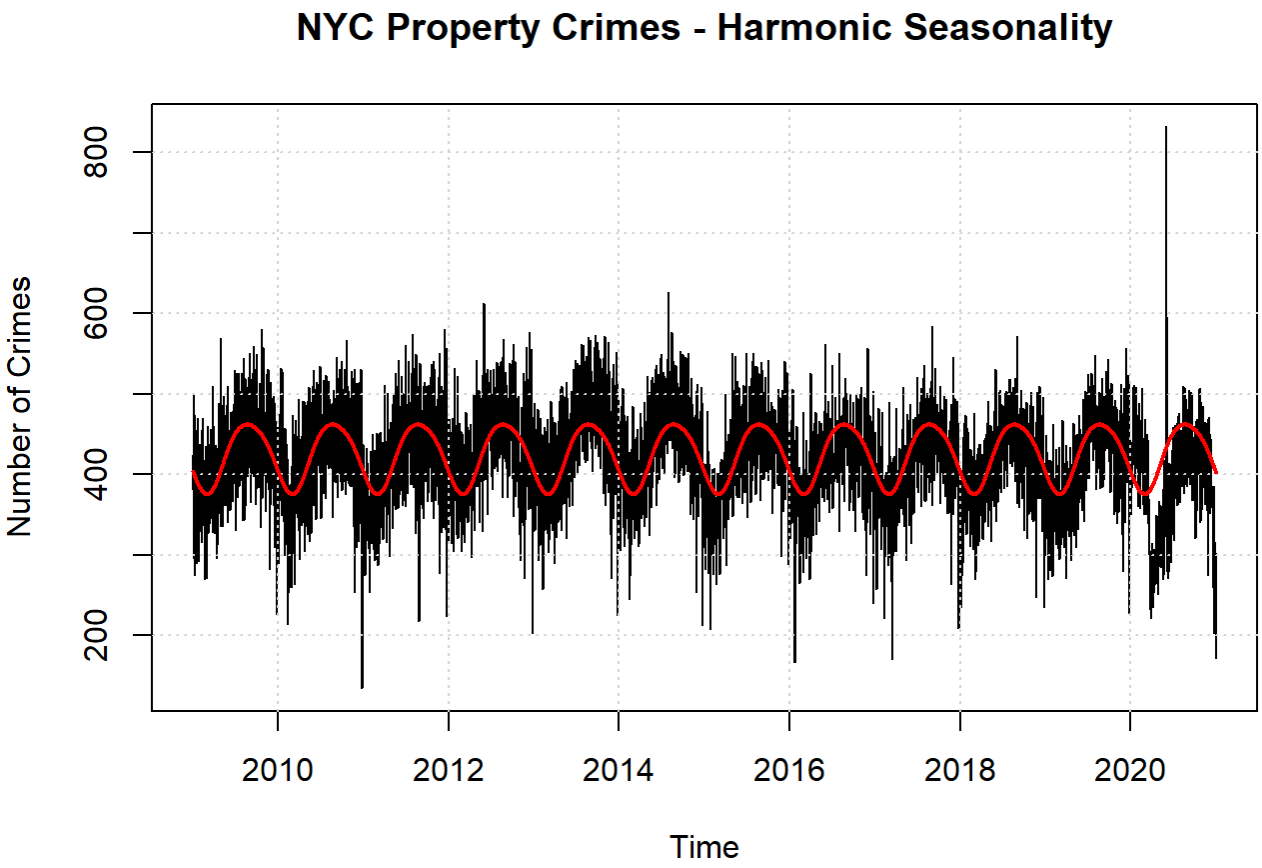
```
##
## Time series regression with "ts" data:
## Start = 2009(1), End = 2021(3)
##
## Call:
## dynlm(formula = nyc_p ~ harmon(nyc_p))
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -269.11  -32.12    5.11   34.30  403.52
##
## Coefficients:
##                   Estimate Std. Error t value Pr(>|t|)
## (Intercept)      424.0262     0.8133  521.39   <2e-16 ***
## harmon(nyc_p)cos -24.0703     1.1497  -20.94   <2e-16 ***
## harmon(nyc_p)sin -35.6132     1.1505  -30.95   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 53.84 on 4380 degrees of freedom
## Multiple R-squared:  0.2418, Adjusted R-squared:  0.2414
## F-statistic: 698.2 on 2 and 4380 DF,  p-value: < 2.2e-16
```

```
harmonic.2 = dynlm(nyc_p~harmon(nyc_p,2))
summary(harmonic.2)
```

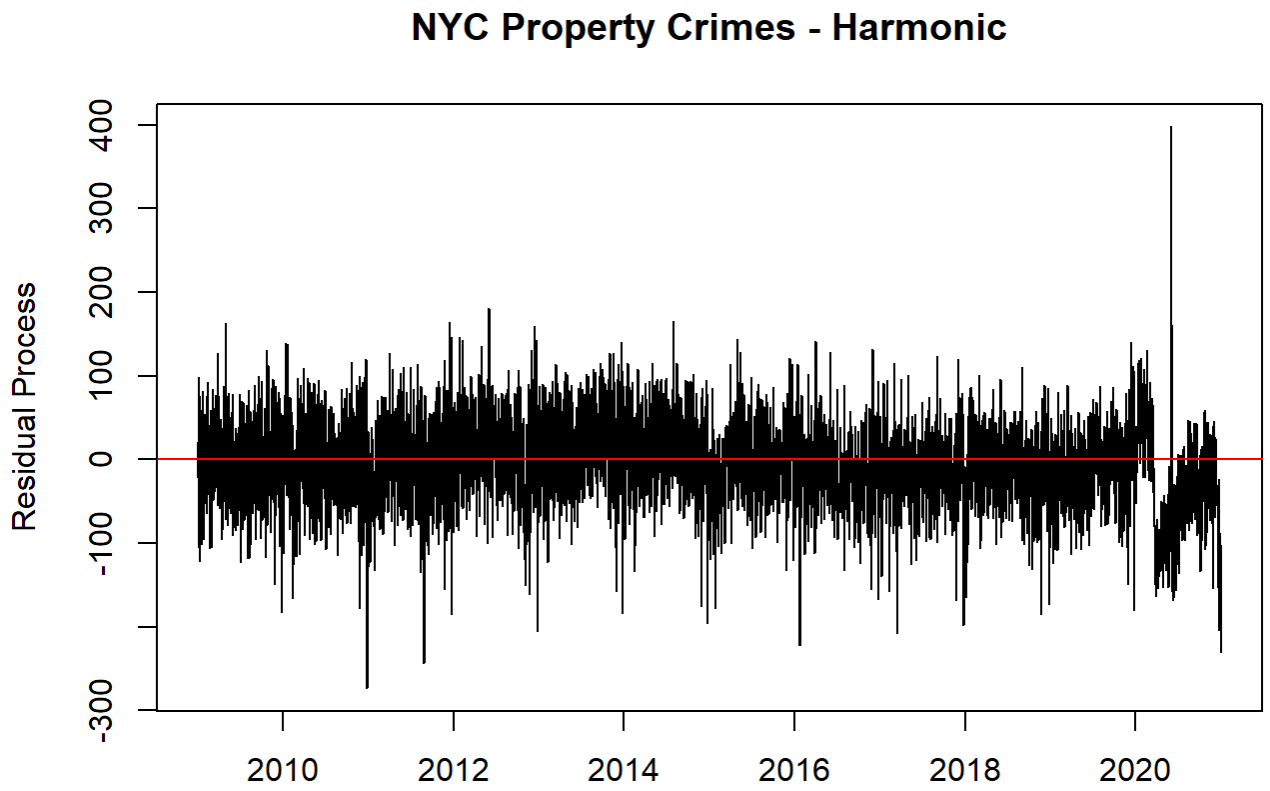
```
##
## Time series regression with "ts" data:
## Start = 2009(1), End = 2021(3)
##
## Call:
## dynlm(formula = nyc_p ~ harmon(nyc_p, 2))
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -273.48  -31.47    5.34   34.86  397.81
##
## Coefficients:
##                       Estimate Std. Error t value Pr(>|t|)
## (Intercept)          424.0238     0.8111 522.757  < 2e-16 ***
## harmon(nyc_p, 2)cos1 -24.0751     1.1467 -20.995  < 2e-16 ***
## harmon(nyc_p, 2)cos2   3.6843     1.1467   3.213 0.001324 **
## harmon(nyc_p, 2)sin1 -35.6132     1.1475 -31.035  < 2e-16 ***
## harmon(nyc_p, 2)sin2  -4.4042     1.1475  -3.838 0.000126 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 53.7 on 4378 degrees of freedom
## Multiple R-squared:  0.2461, Adjusted R-squared:  0.2454
## F-statistic: 357.2 on 4 and 4378 DF,  p-value: < 2.2e-16
```

```
###keep more complex model. all results stat sig at 99% conf level
```

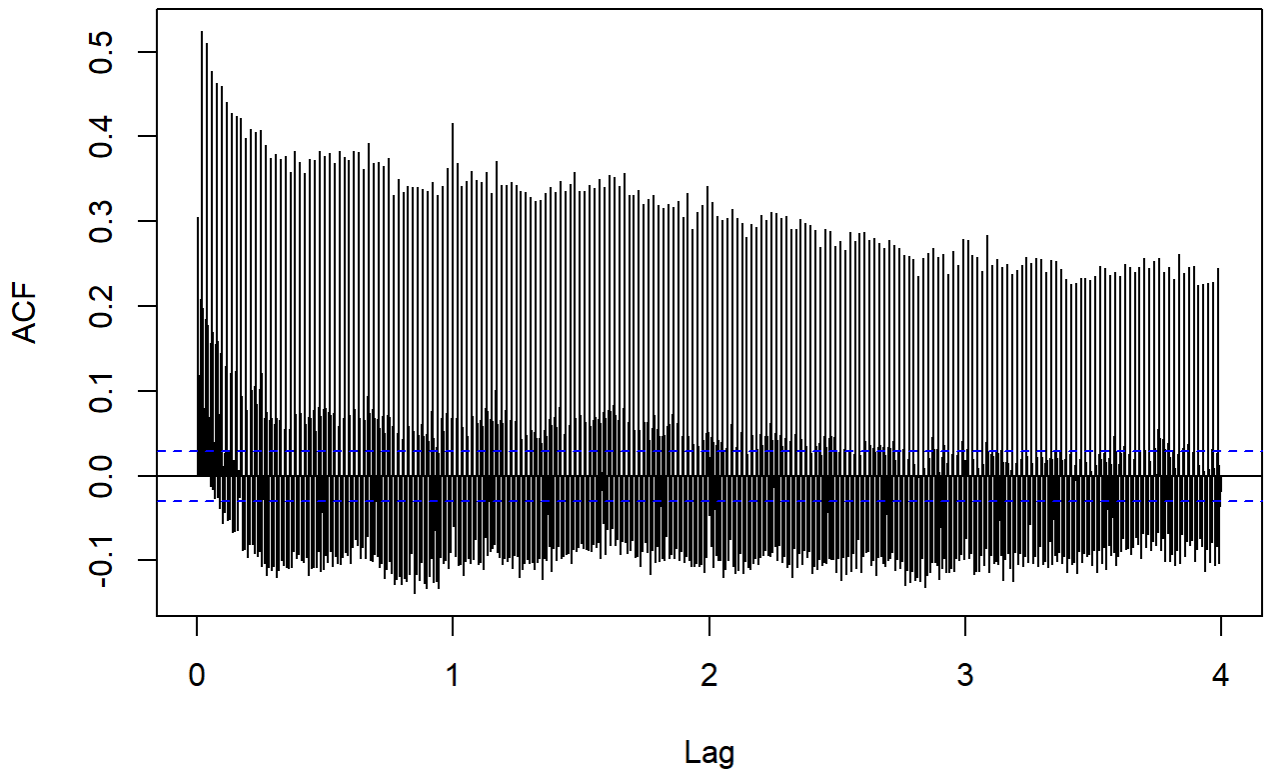
```
###Harmonic Plot
## Plot
ts.plot(nyc_p,ylab="Number of Crimes", main = "NYC Property Crimes - Harmonic Seasonality")
grid()
lines(fitted(harmonic.2),lwd=2,col="red")
```




```
###Seasonality Harmonic - Residuals + Resid ACF

resid.harmonic <- residuals(harmonic.2)

ts.plot(resid.harmonic, xlab = "", ylab = "Residual Process", main = "NYC Property Crimes - Harm
onic")
abline(h=0, col='red')
```




```
acf(resid.harmonic, lag.max = 365 * 4, main = "NYC Property Crimes - Harmonic ACF")
```



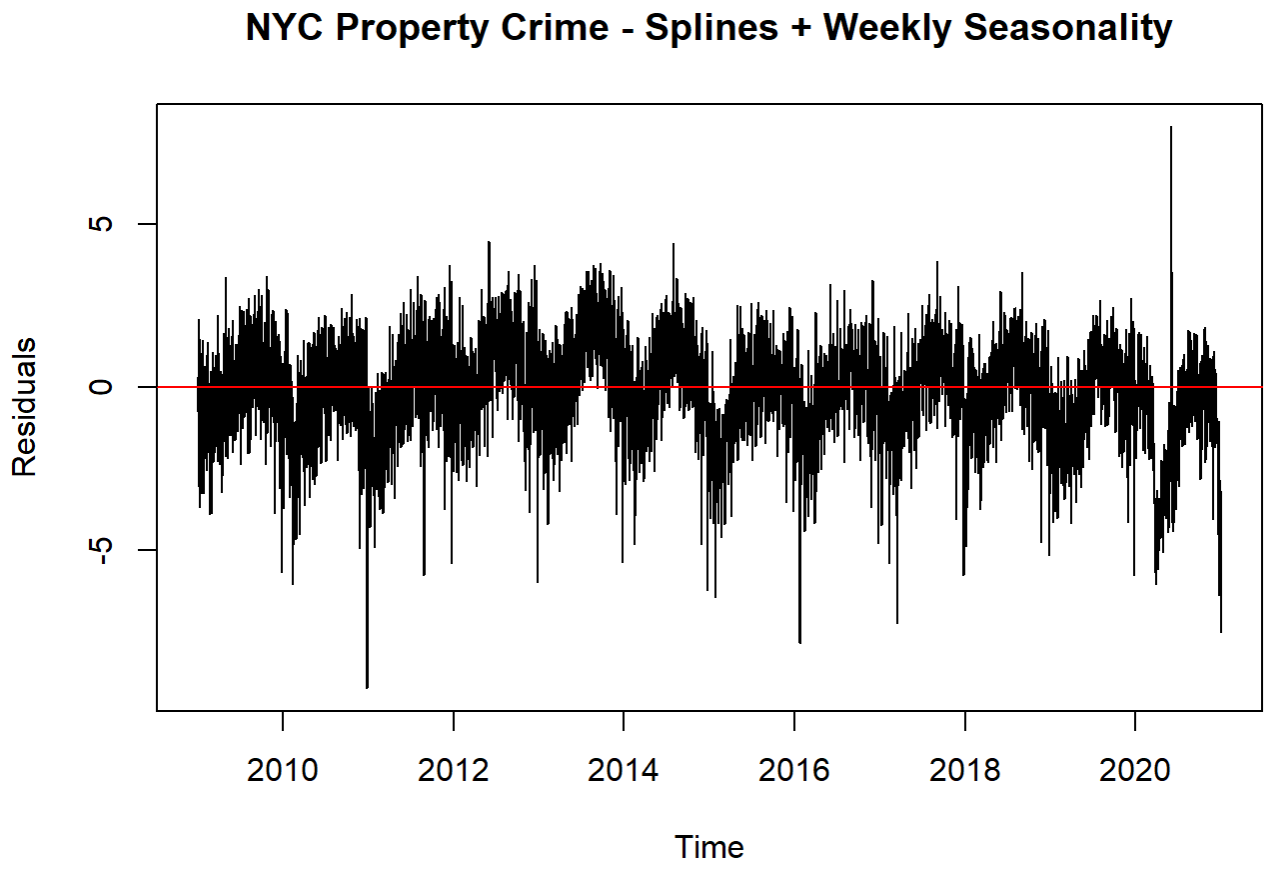

```
###TREND + SEASONALITY - SPLINES + WEEKLY SEASONALITY
nyc_prop$Date <- as.Date(nyc_prop$Date)
year <- as.factor(format(nyc_prop$Date,'%Y'))
month <- as.factor(format(nyc_prop$Date, '%b'))
week <- as.factor(weekdays(nyc_prop$Date))

gam.fit.seastr = gam(nyc_p.tr~s(time.pts)+week-1)
summary(gam.fit.seastr)
```

```
##
## Family: gaussian
## Link function: identity
##
## Formula:
## nyc_p.tr ~ s(time.pts) + week - 1
##
## Parametric coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## weekFriday    20.49064    0.09200   222.7   <2e-16 ***
## weekMonday    20.55536    0.09672   212.5   <2e-16 ***
## weekSaturday  20.52619    0.09427   217.7   <2e-16 ***
## weekSunday    20.50564    0.09200   222.9   <2e-16 ***
## weekThursday  20.52882    0.09200   223.1   <2e-16 ***
## weekTuesday   20.52858    0.08791   233.5   <2e-16 ***
## weekWednesday 20.47315    0.09672   211.7   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Approximate significance of smooth terms:
##               edf Ref.df     F p-value
## s(time.pts) 5.153  6.265 8.402  <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## R-sq.(adj) =  0.0257   Deviance explained = 99.5%
## GCV = 2.1479  Scale est. = 2.1328    n = 1728
```

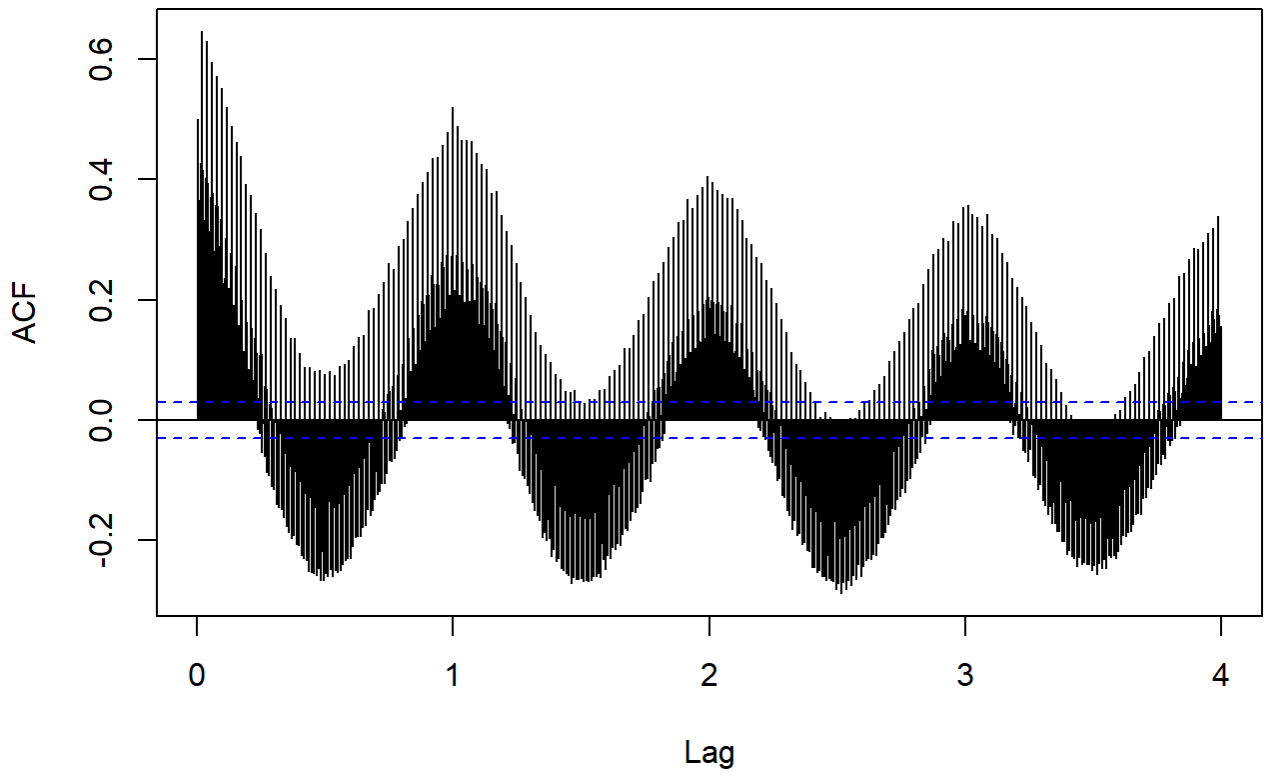
```
dif.fit.seastr = ts(nyc_p.tr - fitted(gam.fit.seastr),start=2009,frequency=365)
```

```
## Warning in `-.default`(nyc_p.tr, fitted(gam.fit.seastr)): longer object length
## is not a multiple of shorter object length
```

```
ts.plot(dif.fit.seastr, ylab = "Residuals", main = "NYC Property Crime - Splines + Weekly Season
ality")
abline(h=0, col='red')
```




```
acf(dif.fit.seastr, lag.max = 365 * 4, main = "NYC Property Crime - Splines + Weekly Seasonality
Resid ACF")
```



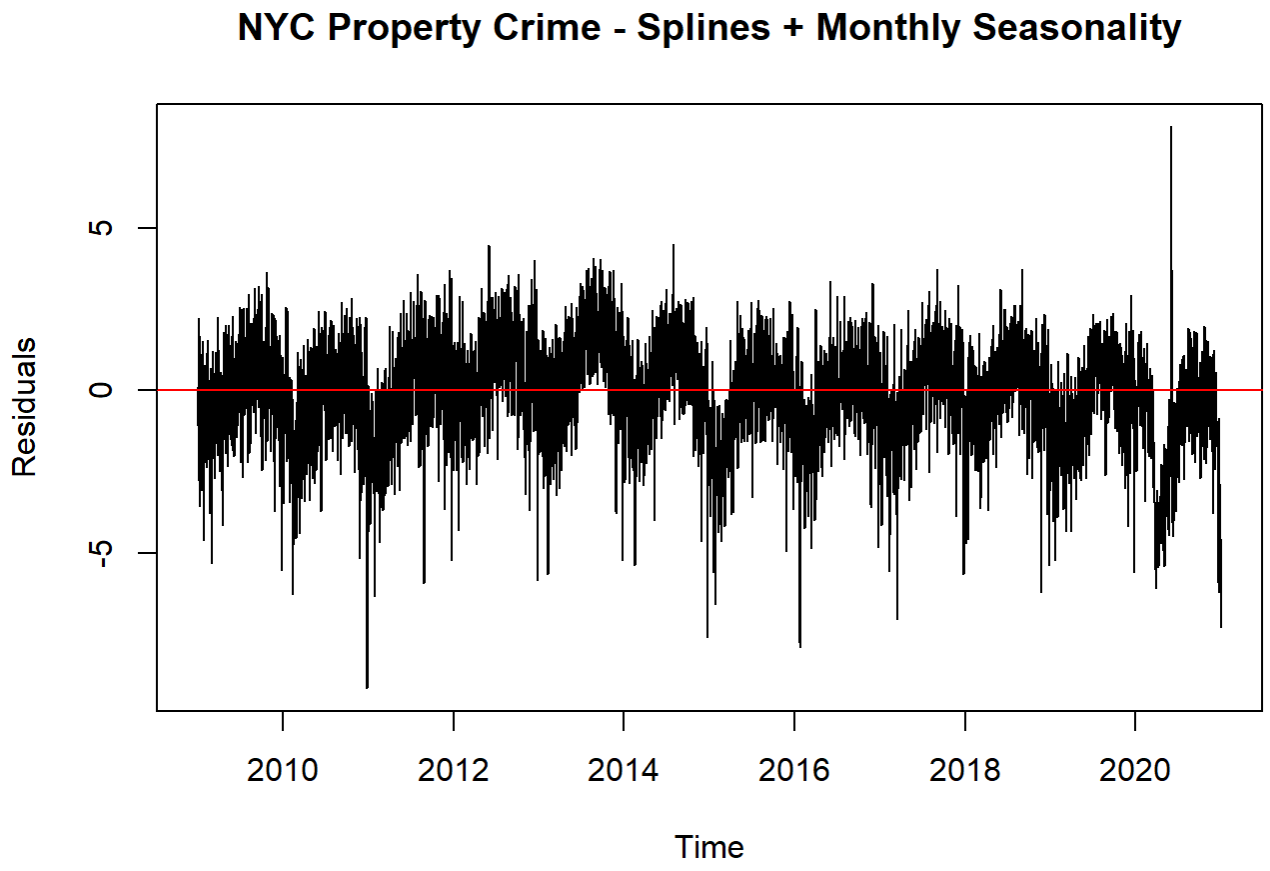

```
###TREND + SEASONALITY - SPLINES + MONTHLY SEASONALITY
nyc_prop$Date <- as.Date(nyc_prop$Date)
year <- as.factor(format(nyc_prop$Date,'%Y'))
month <- as.factor(format(nyc_prop$Date, '%b'))
week <- as.factor(weekdays(nyc_prop$Date))

gam.fit.seastr = gam(nyc_p.tr~s(time.pts)+month-1)
summary(gam.fit.seastr)
```

```
##
## Family: gaussian
## Link function: identity
##
## Formula:
## nyc_p.tr ~ s(time.pts) + month - 1
##
## Parametric coefficients:
##          Estimate Std. Error t value Pr(>|t|)
## monthApr  20.2171     0.1163   173.8   <2e-16 ***
## monthAug  20.3314     0.1163   174.8   <2e-16 ***
## monthDec  20.5423     0.1163   176.6   <2e-16 ***
## monthFeb  20.4184     0.1163   175.6   <2e-16 ***
## monthJan  21.9127     0.1163   188.4   <2e-16 ***
## monthJul  20.3197     0.1163   174.7   <2e-16 ***
## monthJun  20.3569     0.1163   175.0   <2e-16 ***
## monthMar  20.3118     0.1163   174.7   <2e-16 ***
## monthMay  20.3867     0.1163   175.3   <2e-16 ***
## monthNov  20.3853     0.1163   175.3   <2e-16 ***
## monthOct  20.6650     0.1163   177.7   <2e-16 ***
## monthSep  20.3403     0.1163   174.9   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Approximate significance of smooth terms:
##               edf Ref.df     F p-value
## s(time.pts) 5.474  6.615 8.817  <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## R-sq.(adj) =   0.11   Deviance explained = 99.5%
## GCV = 1.9676  Scale est. = 1.9477    n = 1728
```

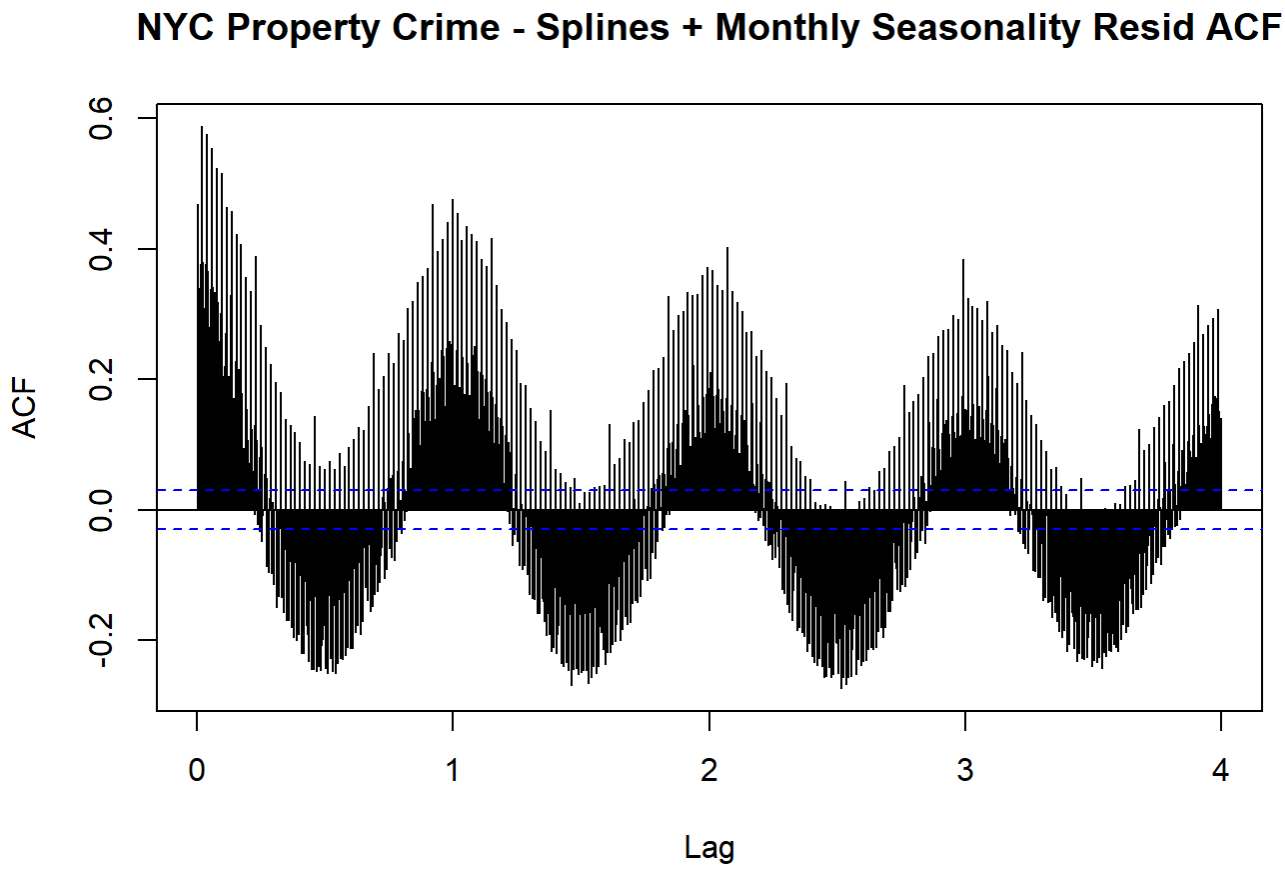
```
dif.fit.seastr = ts(nyc_p.tr - fitted(gam.fit.seastr),start=2009,frequency=365)
```

```
## Warning in `-.default`(nyc_p.tr, fitted(gam.fit.seastr)): longer object length
## is not a multiple of shorter object length
```

```
ts.plot(dif.fit.seastr, ylab = "Residuals", main = "NYC Property Crime - Splines + Monthly Seaso
nality")
abline(h=0, col='red')
```




```
acf(dif.fit.seastr, lag.max = 365 * 4, main = "NYC Property Crime - Splines + Monthly Seasonalit
y Resid ACF")
```



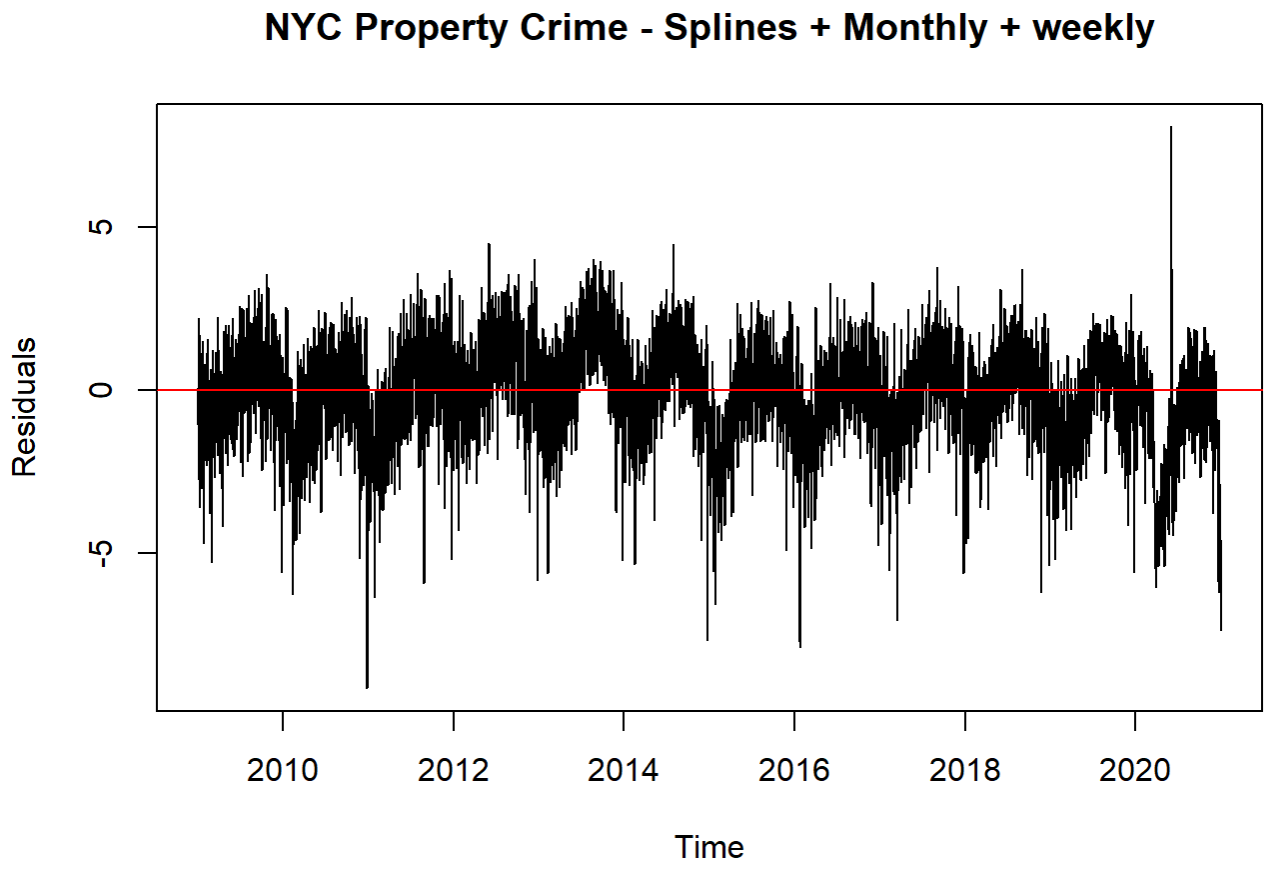

```
###TREND + SEASONALITY - SPLINES + MONTHLY + WEEKLY SEASONALITY
nyc_prop$Date <- as.Date(nyc_prop$Date)
year <- as.factor(format(nyc_prop$Date,'%Y'))
month <- as.factor(format(nyc_prop$Date, '%b'))
week <- as.factor(weekdays(nyc_prop$Date))

gam.fit.seastr.1 = gam(nyc_p.tr~s(time.pts)+month+week-1)
summary(gam.fit.seastr.1)
```

```
##
## Family: gaussian
## Link function: identity
##
## Formula:
## nyc_p.tr ~ s(time.pts) + month + week - 1
##
## Parametric coefficients:
##                Estimate Std. Error t value Pr(>|t|)
## monthApr      20.201424   0.137366 147.063   <2e-16 ***
## monthAug      20.305759   0.142505 142.491   <2e-16 ***
## monthDec      20.524825   0.147117 139.514   <2e-16 ***
## monthFeb      20.395079   0.141793 143.837   <2e-16 ***
## monthJan      21.901737   0.141901 154.346   <2e-16 ***
## monthJul      20.303947   0.137366 147.809   <2e-16 ***
## monthJun      20.334415   0.142272 142.926   <2e-16 ***
## monthMar      20.297967   0.146827 138.244   <2e-16 ***
## monthMay      20.373396   0.142141 143.333   <2e-16 ***
## monthNov      20.371384   0.146828 138.743   <2e-16 ***
## monthOct      20.649671   0.147174 140.307   <2e-16 ***
## monthSep      20.322768   0.147116 138.141   <2e-16 ***
## weekMonday     0.091104   0.129641   0.703    0.482
## weekSaturday  -0.024109   0.128613  -0.187    0.851
## weekSunday     0.003961   0.125277   0.032    0.975
## weekThursday  -0.007923   0.127392  -0.062    0.950
## weekTuesday    0.035354   0.123456   0.286    0.775
## weekWednesday  0.025411   0.128511   0.198    0.843
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Approximate significance of smooth terms:
##               edf Ref.df     F p-value
## s(time.pts) 5.461  6.601 8.808  <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## R-sq.(adj) =  0.108   Deviance explained = 99.5%
## GCV = 1.9803  Scale est. = 1.9534    n = 1728
```

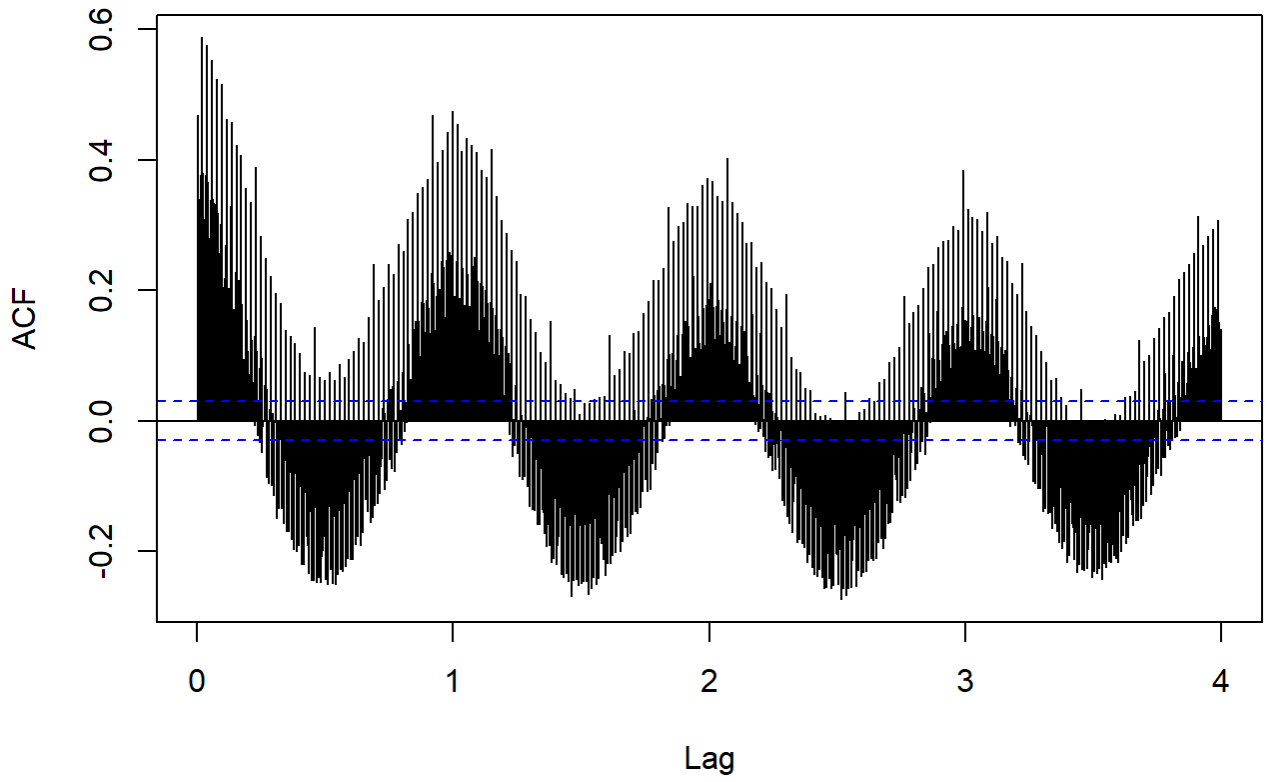
```
dif.fit.seastr.1 = ts(nyc_p.tr - fitted(gam.fit.seastr.1),start=2009,frequency=365)
```

```
## Warning in `-.default`(nyc_p.tr, fitted(gam.fit.seastr.1)): longer object length
## is not a multiple of shorter object length
```

```
ts.plot(dif.fit.seastr.1, ylab = "Residuals", main = "NYC Property Crime - Splines + Monthly + w
eekly")
abline(h=0, col='red')
```




```
acf(dif.fit.seastr.1, lag.max = 365 * 4, main = "NYC Property Crime - Splines + Monthly + Weekly
Resid ACF")
```



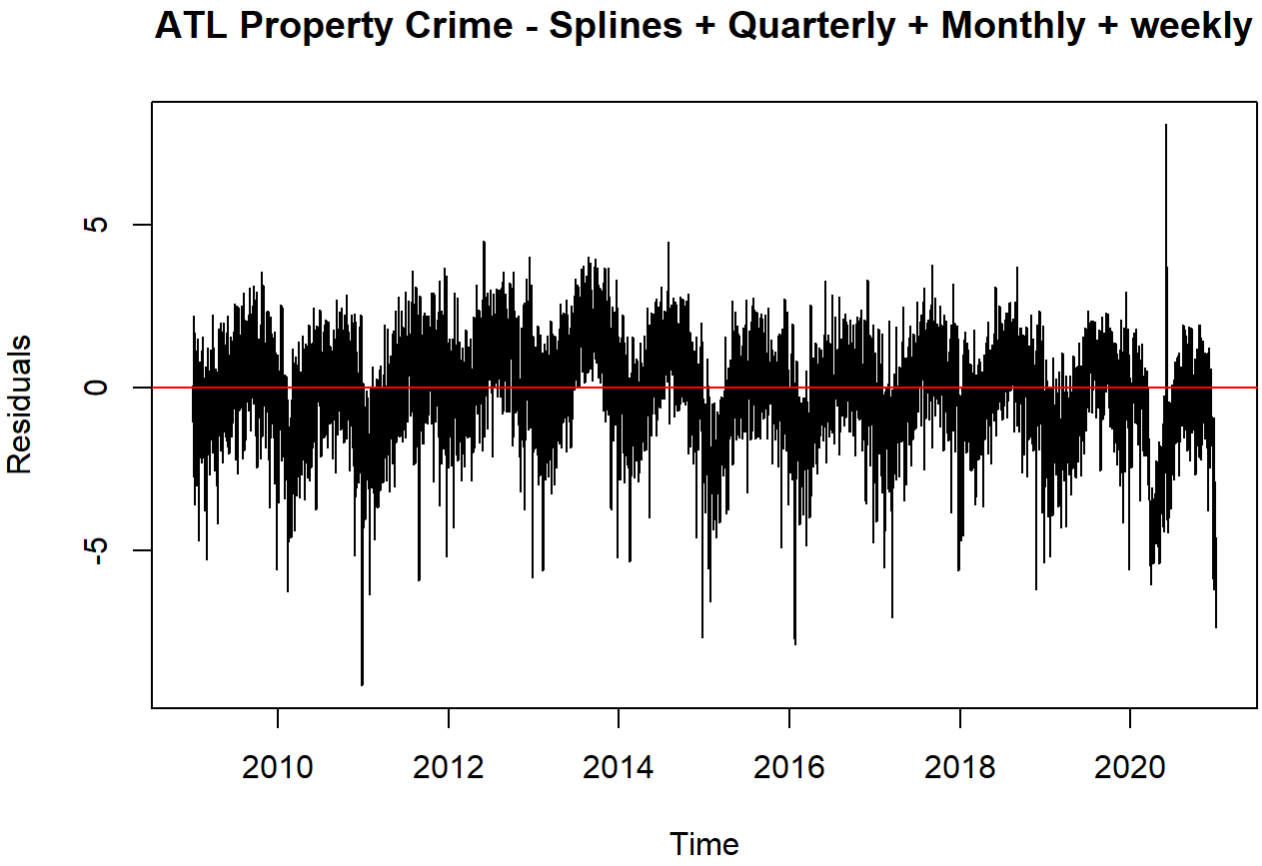

```
###TREND + SEASONALITY - SPLINES + QUARTERLY + MONTHLY + WEEKLY SEASONALITY
nyc_prop$Date <- as.Date(nyc_prop$Date)
year <- as.factor(format(nyc_prop$Date,'%Y'))
month <- as.factor(format(nyc_prop$Date, '%b'))
week <- as.factor(weekdays(nyc_prop$Date))
quarterly <- as.factor(quarter(nyc_prop$Date))

gam.fit.seastr.1 = gam(nyc_p.tr~s(time.pts)+quarterly+month+week-1)
summary(gam.fit.seastr.1)
```

```
##
## Family: gaussian
## Link function: identity
##
## Formula:
## nyc_p.tr ~ s(time.pts) + quarterly + month + week - 1
##
## Parametric coefficients:
##                Estimate Std. Error t value Pr(>|t|)
## quarterly1    15.648696   0.080354 194.748   <2e-16 ***
## quarterly2    20.201424   0.137366 147.063   <2e-16 ***
## quarterly3    15.233118   0.078829 193.242   <2e-16 ***
## quarterly4    15.386470   0.083778 183.659   <2e-16 ***
## monthAug       5.072641   0.099427  51.019   <2e-16 ***
## monthDec       5.138355   0.099457  51.664   <2e-16 ***
## monthFeb       4.746383   0.098725  48.077   <2e-16 ***
## monthJan       6.253041   0.098449  63.515   <2e-16 ***
## monthJul       5.070828   0.097913  51.789   <2e-16 ***
## monthJun       0.132991   0.165408   0.804    0.422
## monthMar       4.649271   0.100382  46.316   <2e-16 ***
## monthMay       0.171972   0.165409   1.040    0.299
## monthNov       4.984914   0.099627  50.036   <2e-16 ***
## monthOct       5.263201   0.099757  52.760   <2e-16 ***
## monthSep       5.089649   0.101378  50.204   <2e-16 ***
## weekMonday     0.091104   0.129641   0.703    0.482
## weekSaturday  -0.024109   0.128613  -0.187    0.851
## weekSunday     0.003961   0.125277   0.032    0.975
## weekThursday  -0.007923   0.127392  -0.062    0.950
## weekTuesday    0.035354   0.123456   0.286    0.775
## weekWednesday  0.025411   0.128511   0.198    0.843
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Approximate significance of smooth terms:
##               edf Ref.df     F p-value
## s(time.pts) 5.461  6.601 8.808  <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Rank: 27/30
## R-sq.(adj) =  0.108   Deviance explained = 99.5%
## GCV = 1.9803  Scale est. = 1.9534    n = 1728
```

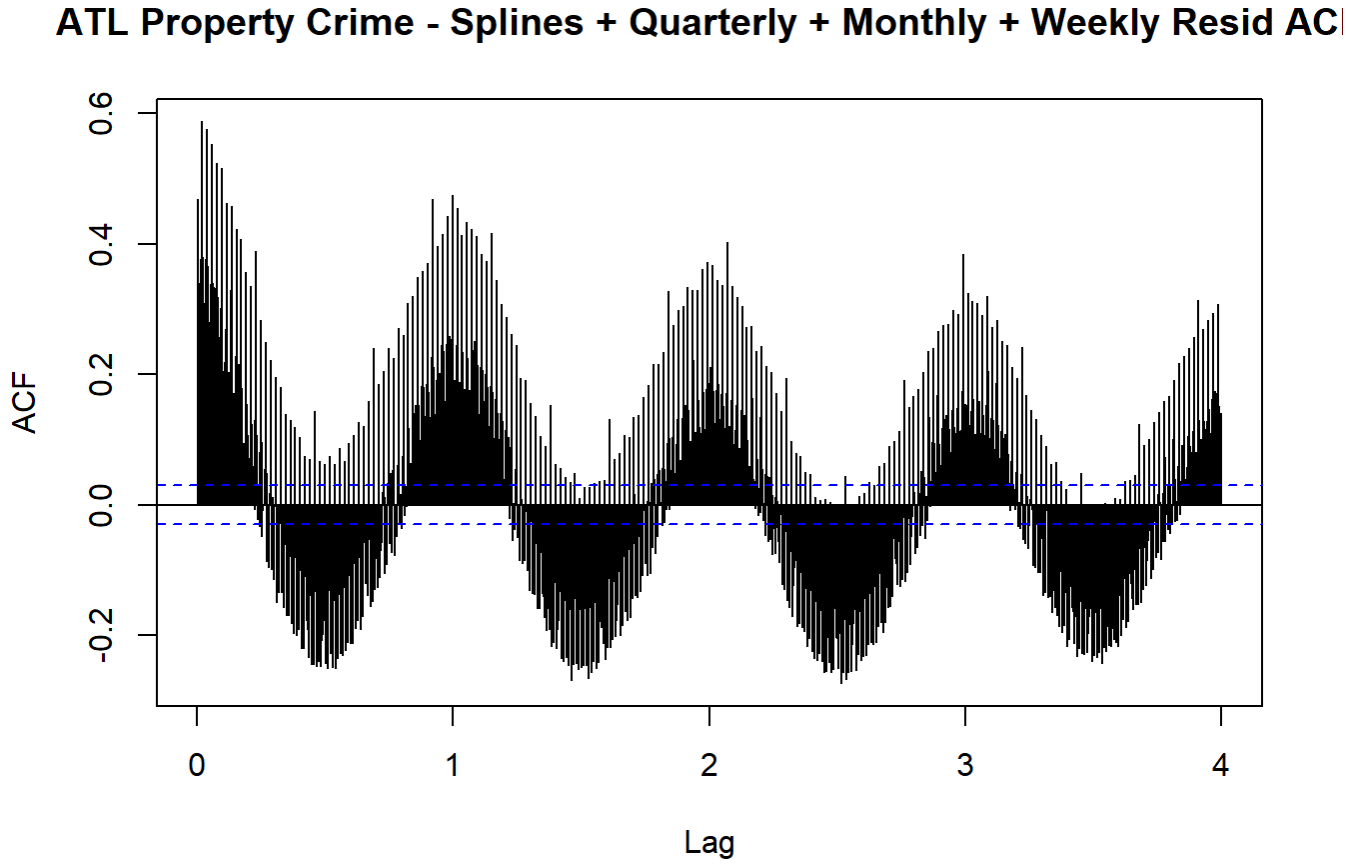
```
dif.fit.seastr.1 = ts(nyc_p.tr - fitted(gam.fit.seastr.1),start=2009,frequency=365)
```

```
## Warning in `-.default`(nyc_p.tr, fitted(gam.fit.seastr.1)): longer object length
## is not a multiple of shorter object length
```

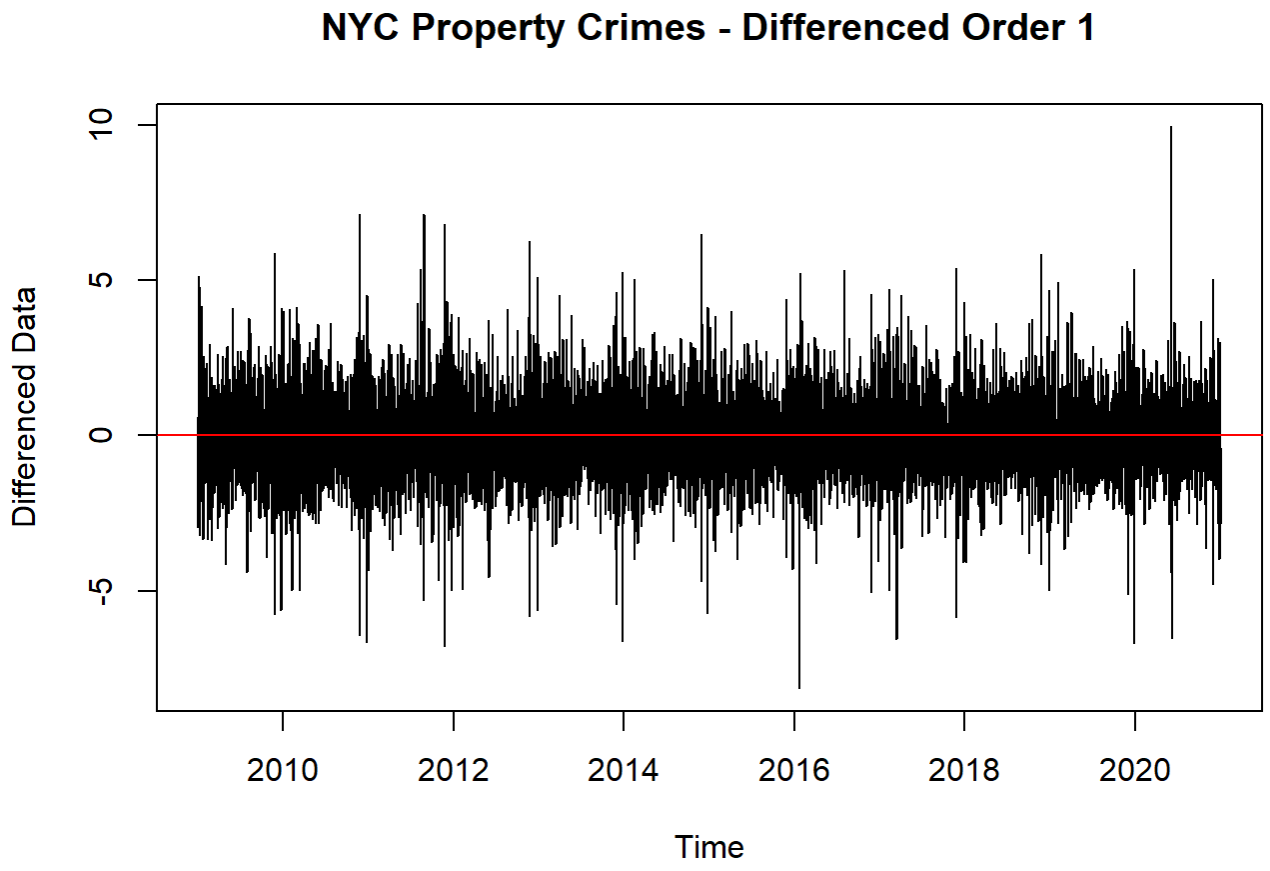
```
ts.plot(dif.fit.seastr.1, ylab = "Residuals", main = "ATL Property Crime - Splines + Quarterly + Monthly + weekly")
abline(h=0, col='red')
```



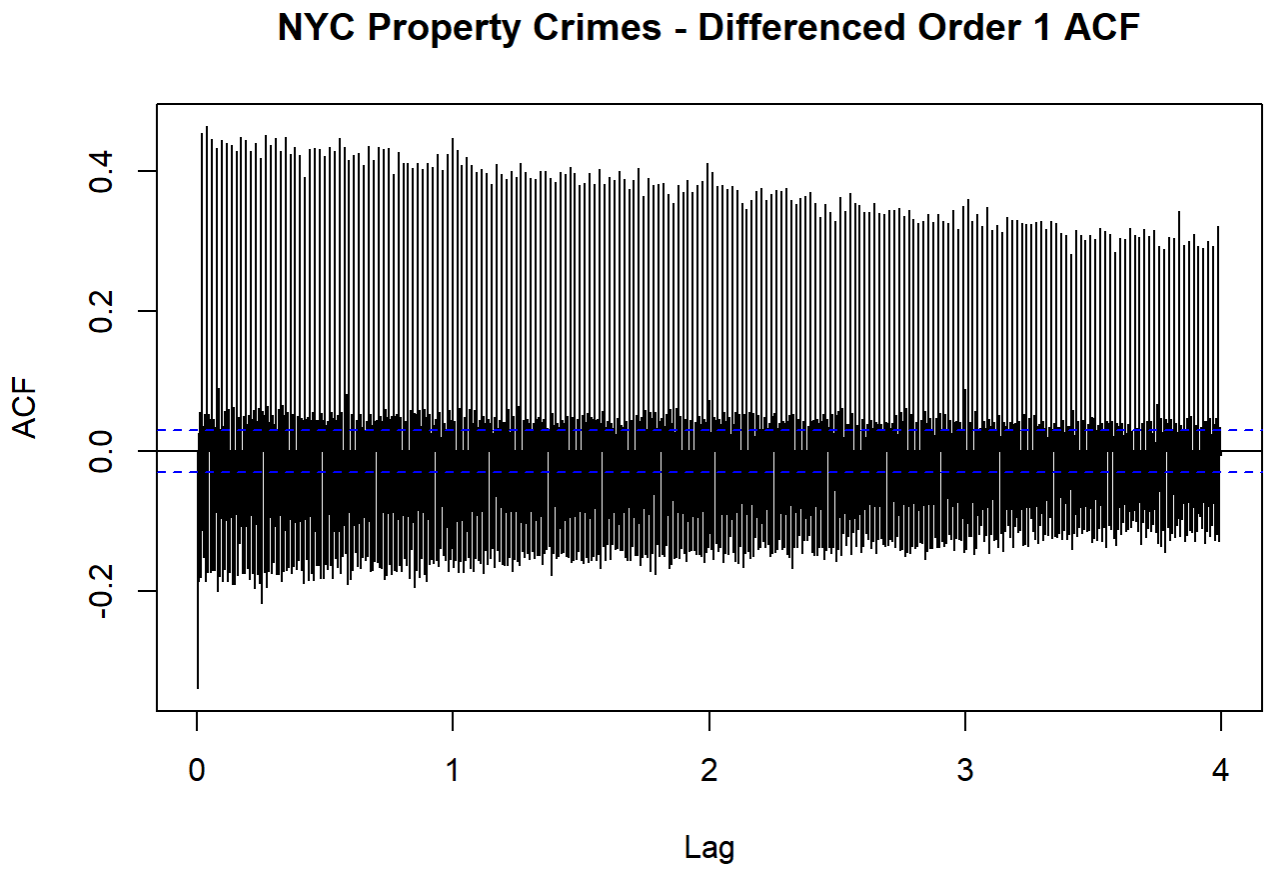

```
acf(dif.fit.seastr.1, lag.max = 365 * 4, main = "ATL Property Crime - Splines + Quarterly + Monthly + Weekly Resid ACF")
```




```
###DIFFERENCED DATA - ORDER 1
diff.nyc_p = diff(nyc_p.tr)
ts.plot(diff.nyc_p, ylab = "Differenced Data", main = "NYC Property Crimes - Differenced Order
 1")
abline(h=0, col='red')
```

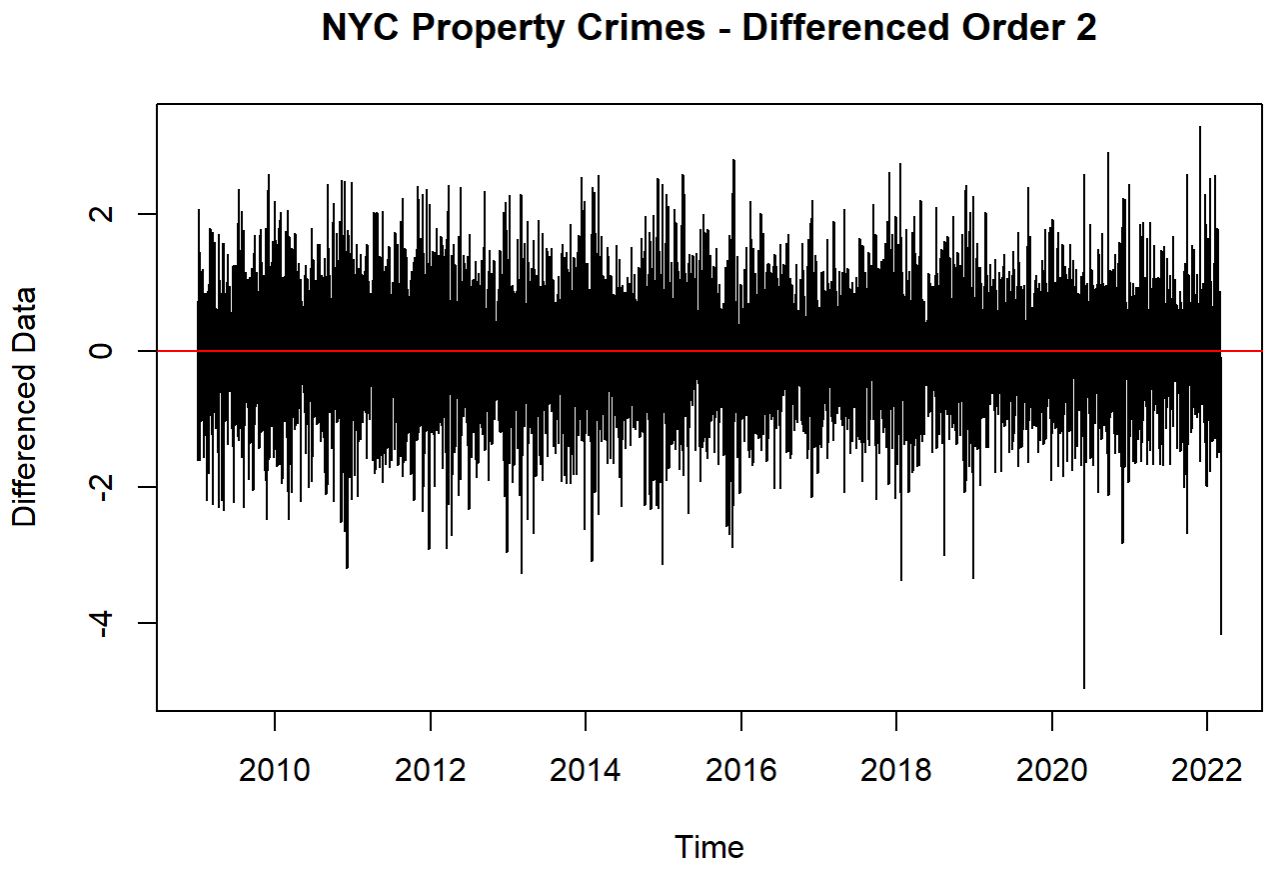



```
acf(diff.nyc_p, lag.max = 365 * 4, main = "NYC Property Crimes - Differenced Order 1 ACF")
```

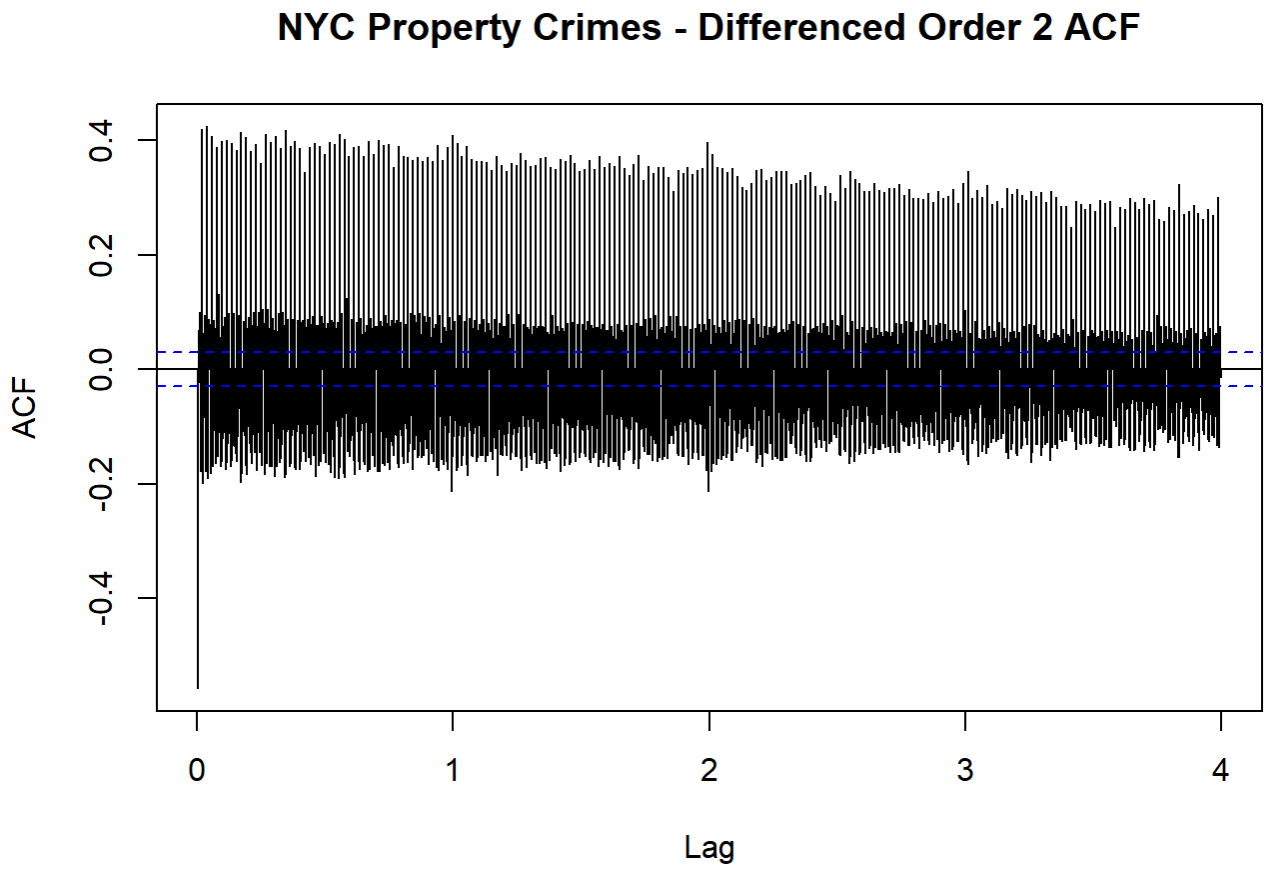



```
### SEE HW2 - Q2 FOR EXPLANATION
```

```
###DIFFERENCED DATA - ORDER 2
diff.nyc_p = diff(nyc_p.tr, differences = 2)
ts.plot(diff.atl_p, ylab = "Differenced Data", main = "NYC Property Crimes - Differenced Order
 2")
abline(h=0, col='red')
```

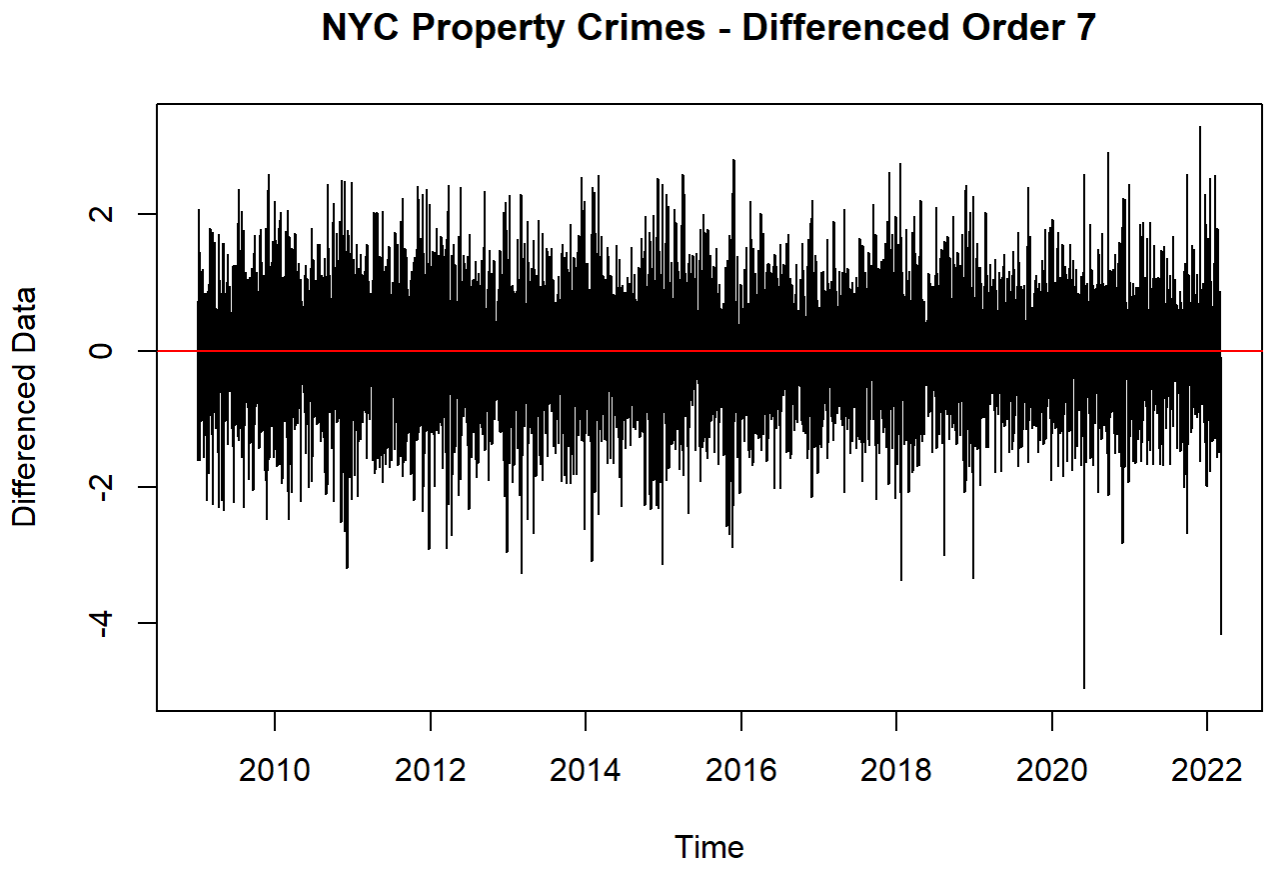



```
acf(diff.nyc_p, lag.max = 365 * 4, main = "NYC Property Crimes - Differenced Order 2 ACF")
```

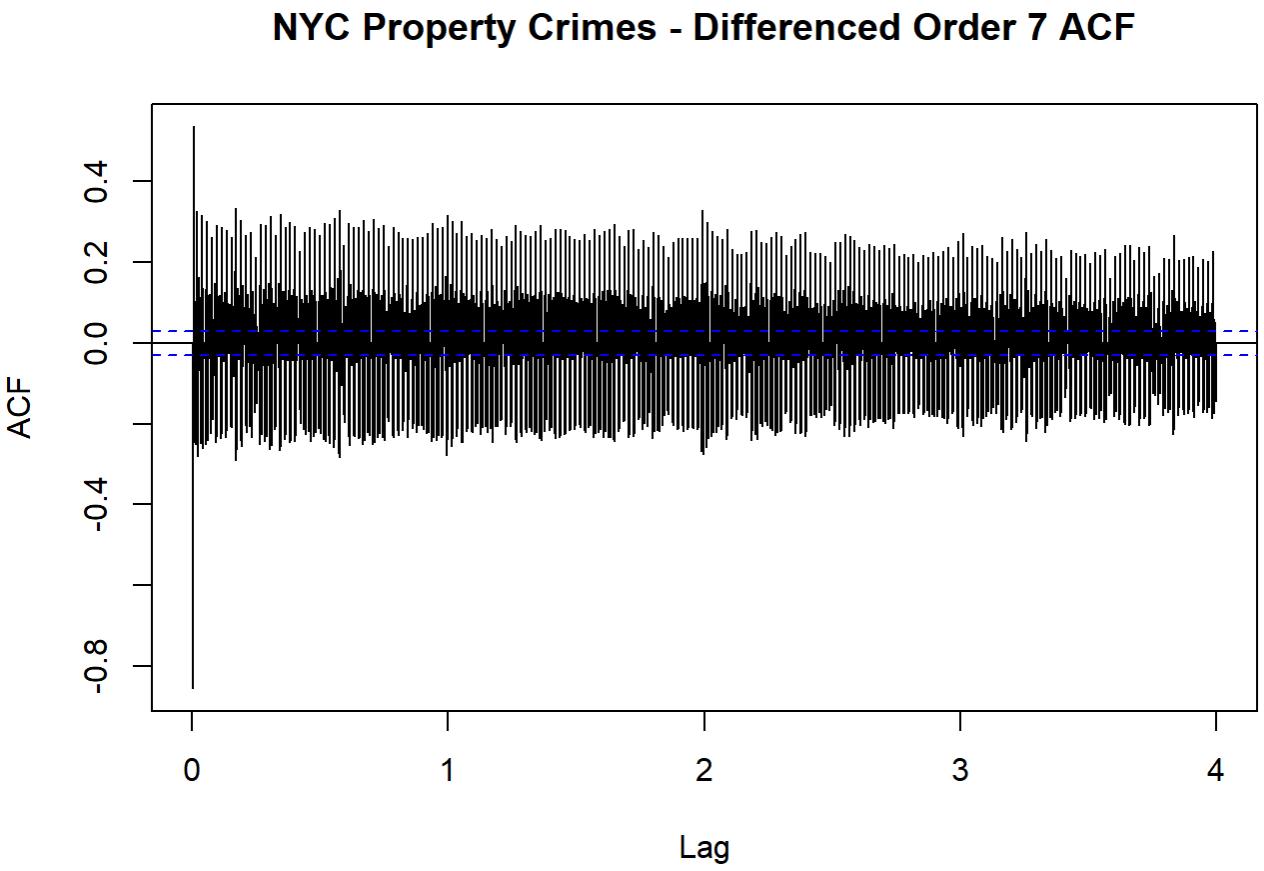



```
### SEE HW2 - Q2 FOR EXPLANATION
```

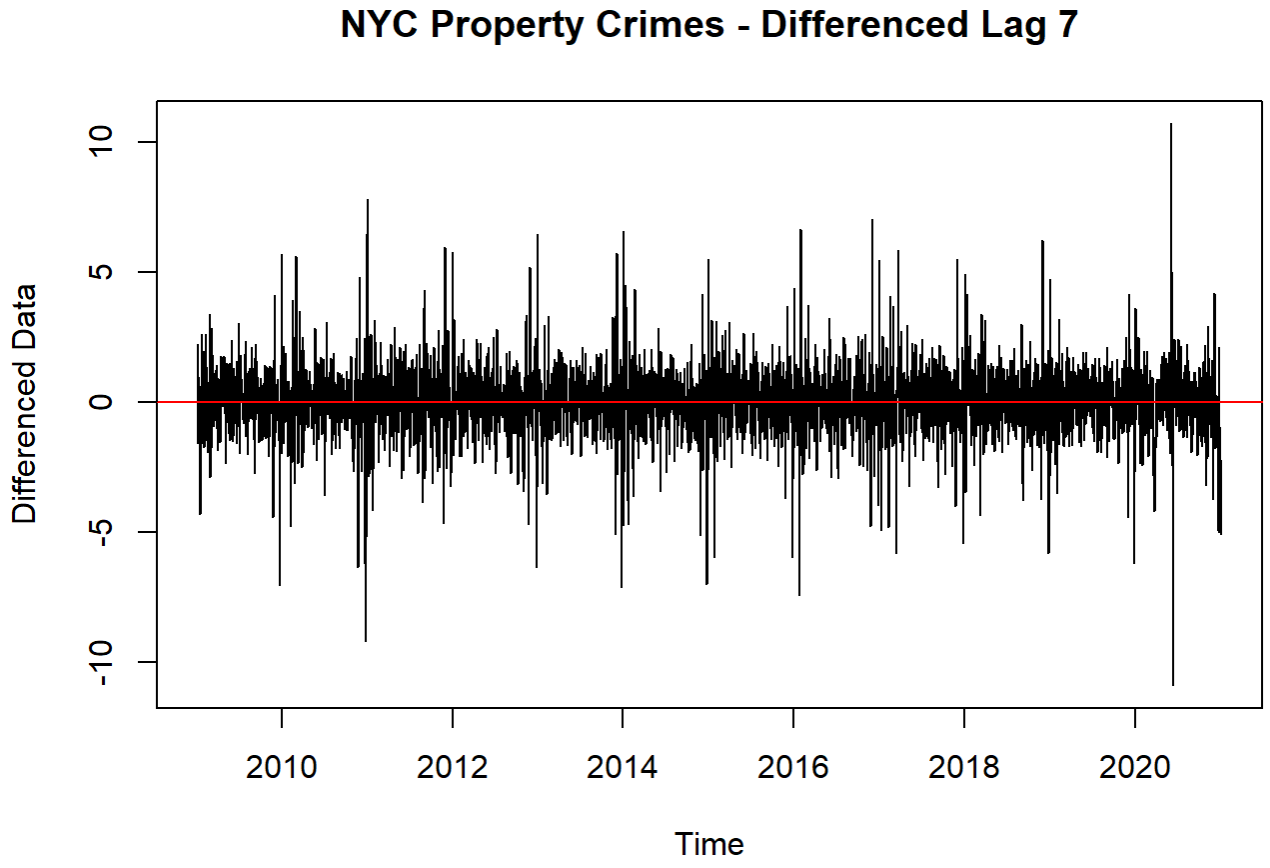
```
###DIFFERENCED DATA - ORDER 7
diff.nyc_p = diff(nyc_p.tr, differences = 7)
ts.plot(diff.atl_p, ylab = "Differenced Data", main = "NYC Property Crimes - Differenced Order
 7")
abline(h=0, col='red')
```



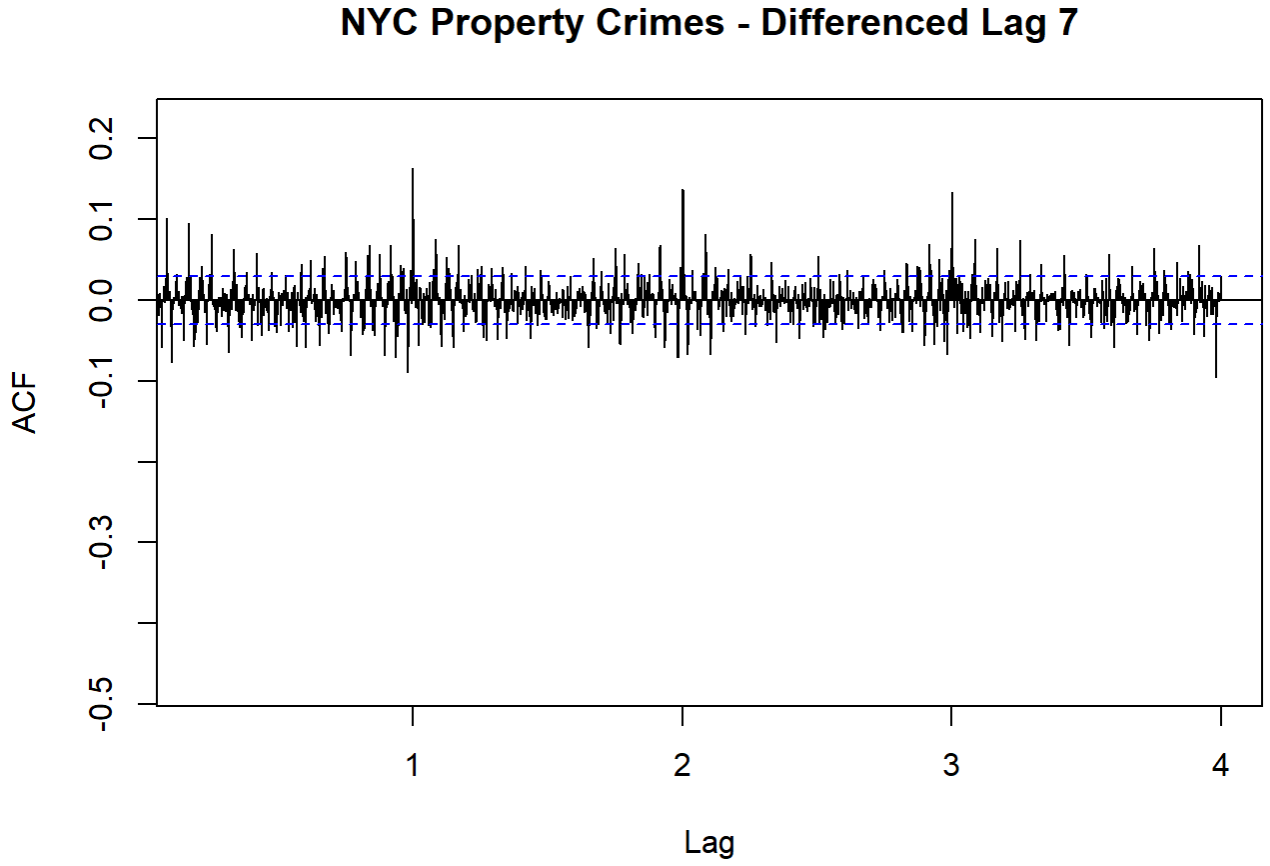

```
acf(diff.nyc_p, lag.max = 365 * 4, main = "NYC Property Crimes - Differenced Order 7 ACF")
```




```
###DIFFERENCED DATA - LAG 7
diff.nyc_p = diff(nyc_p.tr, 7)
ts.plot(diff.nyc_p, ylab = "Differenced Data", main = "NYC Property Crimes - Differenced Lag 7")
abline(h=0, col='red')
```




```
acf(diff.nyc_p, xlim=c(0.2,4), lag.max = 365 * 4, main = "NYC Property Crimes - Differenced Lag
 7")
```




```
###HW2 - Q2 FOR EXPLANATION
```