Stat 153 Midterm 2

Samba Njie Jr., Veronika Yang

4/7/2017

Report

Appendix : Code

Establishing working directory:

```
setwd("/Users/sambamamba/Documents/Cal Spring 2017/STAT_153/MT_2/GoogleTimeSeries")

wd <- getwd(); items <- dir()
```

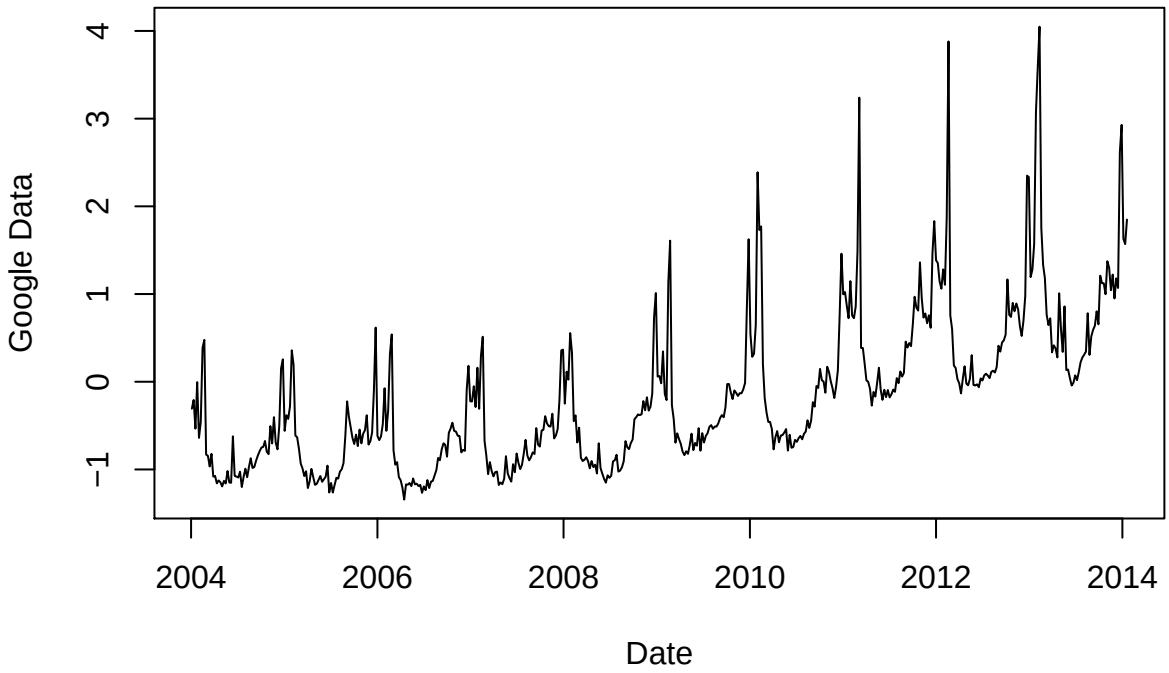
Read in data sets:

```
readData <- function() { # creates a list of the 5 Google data sets
  dtasets <-  items[grepl(".csv", items) == TRUE]
  dataList <- lapply(dtasets, function(dta) read_csv(file.path(wd, dta)))
  names(dataList) <-lapply(1:5, function(x) as.vector(paste0("Q",x,"Train")))
  return(dataList)
}

data <- readData() # where question i can be found by data[[i]] or data$QiTrain
```

Question 1

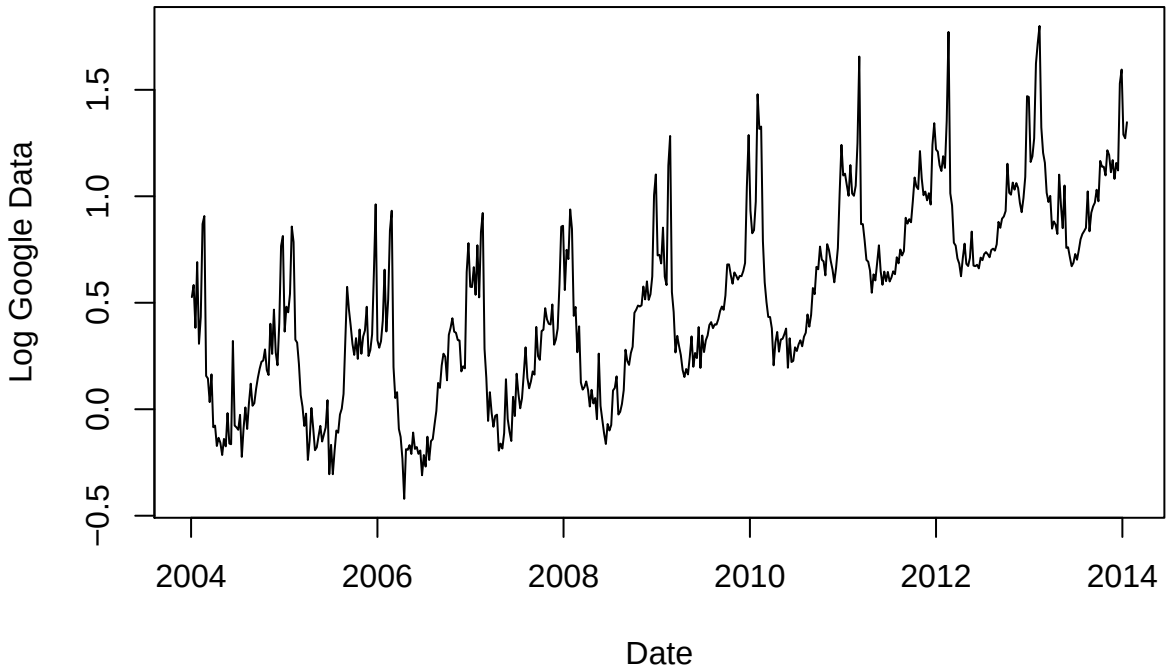
Exploratory Data Analysis

```
Q1Train <- data$Q1Train

plot(Q1Train, type = 'l', xlab = "Date", ylab = "Google Data")
```



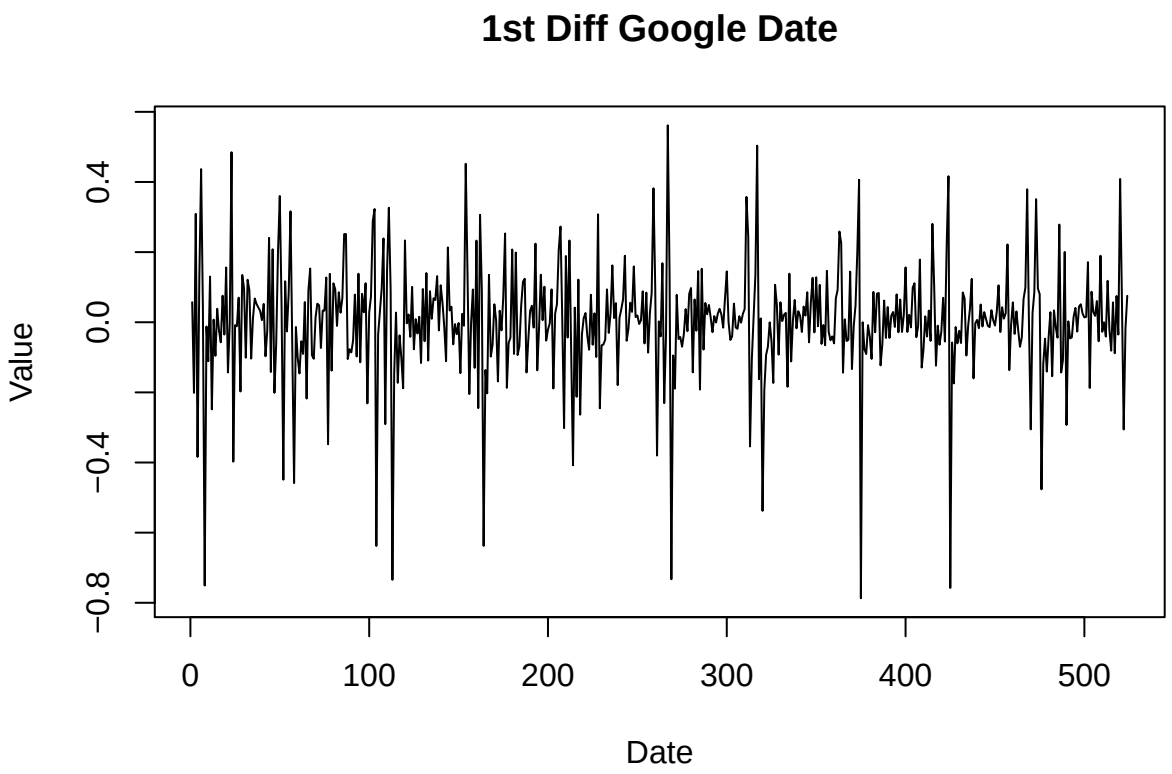
There seems to be an increasing linear trend and a clear seasonality in the data set, with a period of around a year. Homoscedasticity in the data set exists. Meaning, as time increases, there seems to be increasing variance in every period. For more convenient analysis and making variance more consistent, we will implement a log transformation of the data. However while log transformation reduces homoskedasticity, logarithms return `NaN` values with negative data. since the minimum data point in this question is -1.3435, we will shift the data by 2, then perform a log transform, as can be seen in the plot:

```
Q1Train.Log <- data.frame(Date = Q1Train$Date, Activity = log(Q1Train$activity + 2))
plot(Q1Train.Log, type = 'l', xlab = "Date", ylab = "Log Google Data")
```

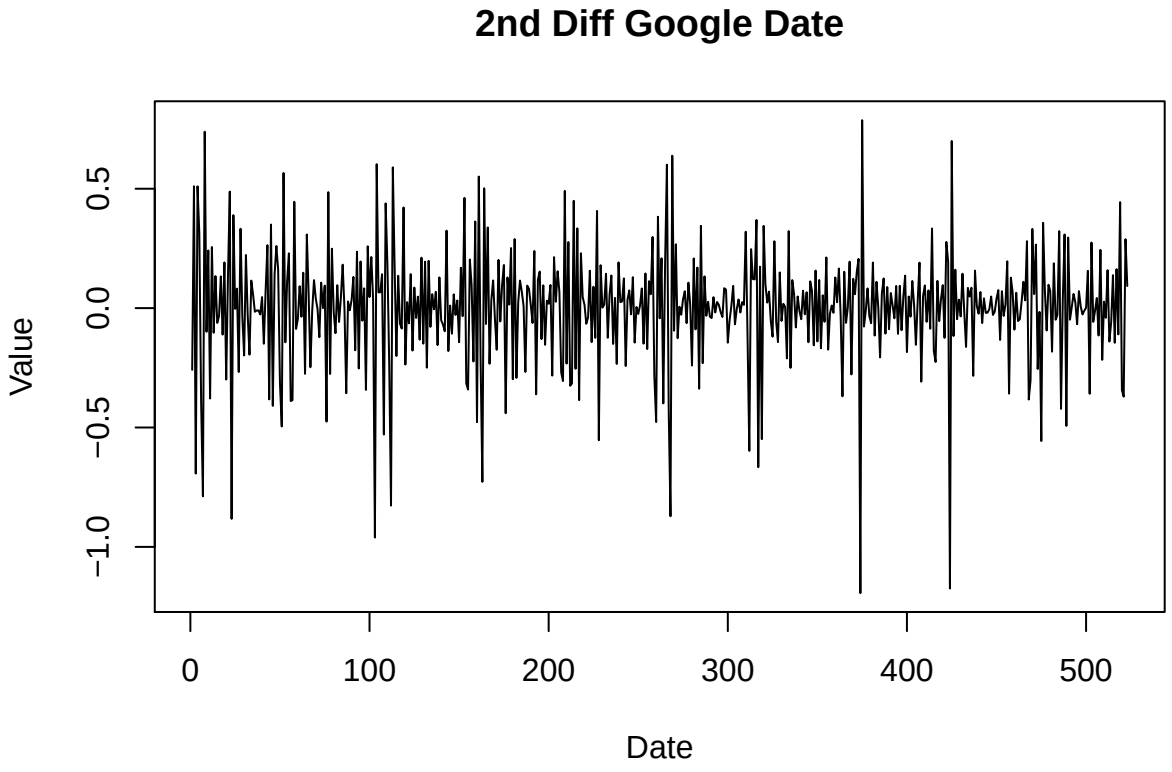


With the shifted log data at hand, we have reduced homoskedasticity extensively. Now, we must remove the trend by using differencing, aspiring to achieve a stationary data set.

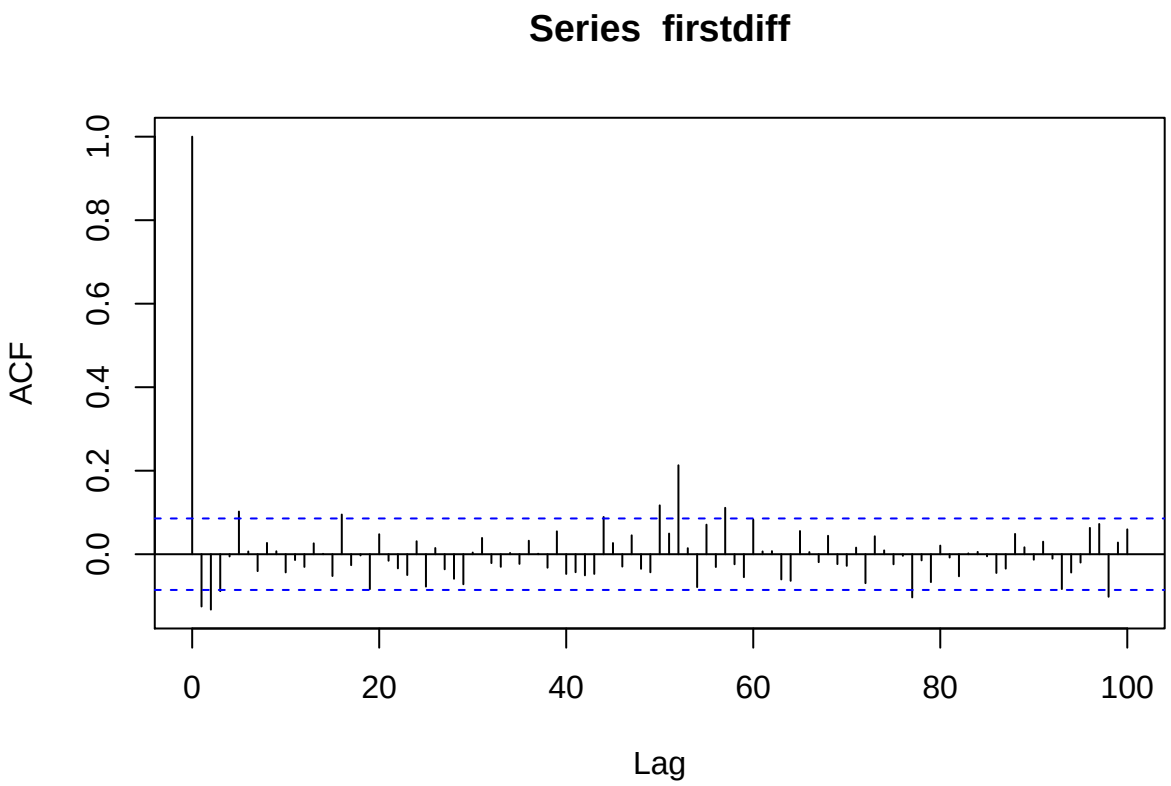
```r
q1train.log <- Q1Train.Log$Activity
#difference <- function(dta, lag.input = 1, order = 1) {
  # Performs differencing for any degree of regular differencing
  #time <- dta[,1]; ts <- dta[,2];
  #ts.out <- diff(ts, lag = lag.input, differences = order)
  #return(data.frame(Date = time[(order + 1):length(time)], Activity = ts.out))
#}

acfIndex <- function(vec, n.max = 1, mod = NA) {
  # input : vector of acf values; output : data frame of the top n.max values and their indices
  stopifnot(n.max <= length(vec));
  val <- rep(NA, n.max); idx <- rep(NA, n.max)
  for (i in 1:n.max) {
    val[i] <- max(vec); idx[i] <- which.max(vec)
    vec <- vec[-idx[i]]
  }
  if (is.na(mod) == FALSE) {
    mod.vec <- idx %% mod
    return(data.frame(index = idx, value = val, remainder = mod.vec))
  }
  return(data.frame(index = idx, value = val))
}

# Observe first and second differenced log data
firstdiff <- diff(q1train.log)
seconddiff <- diff(diff(q1train.log))
thirddiff <- diff(diff(diff(q1train.log)))

# Observe differenced data of orders 1,2
#par(mfrow = c(2,1))
plot(firstdiff, type = 'l', xlab = "Date", ylab = "Value", main =  "1st Diff Google Date");
```



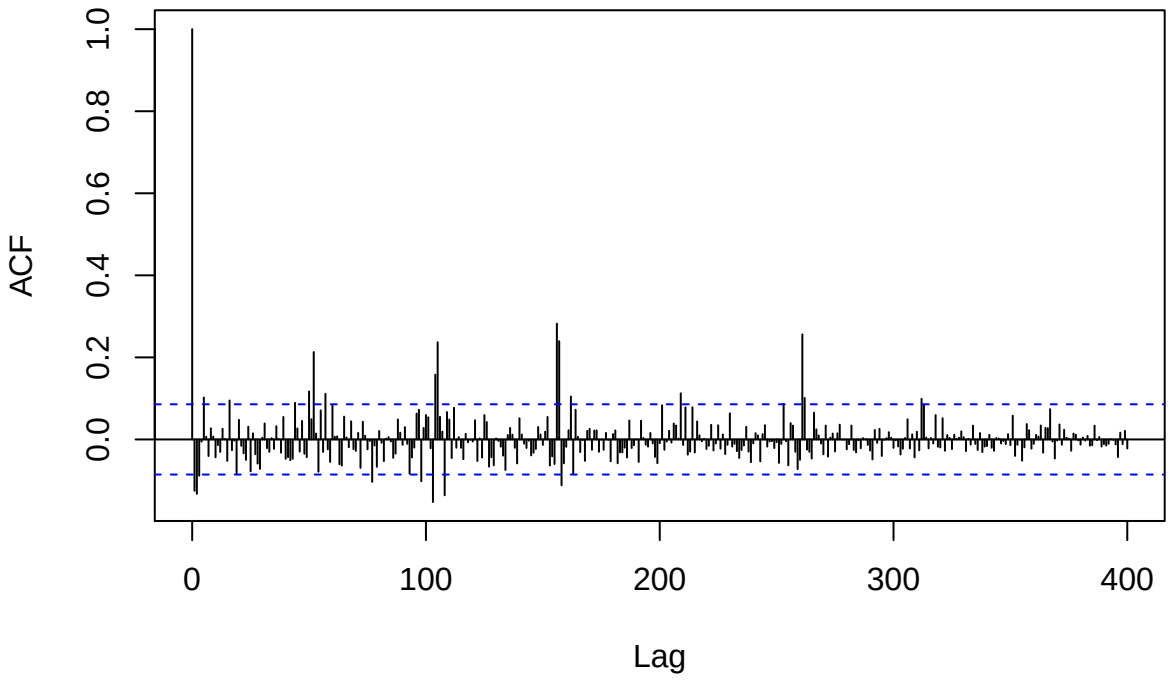

```
plot(seconddiff, type = 'l', xlab = "Date", ylab = "Value", main =  "2nd Diff Google Date")
```




```
# Observe acf of data of orders 1, 2
acf(firstdiff, lag.max = 100)
```

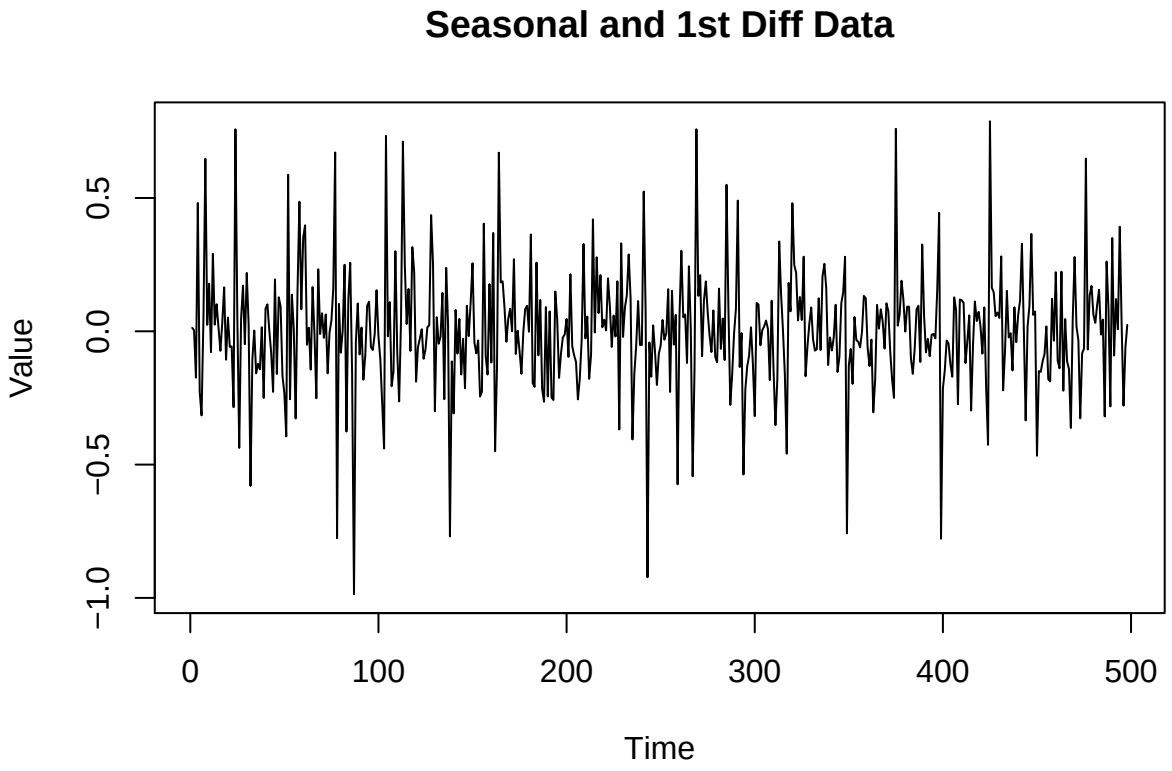



```
checkSeas <- acfIndex(acf(firstdiff, lag.max = 400)$acf, n.max = 10, mod = 52) # check if yearly
```

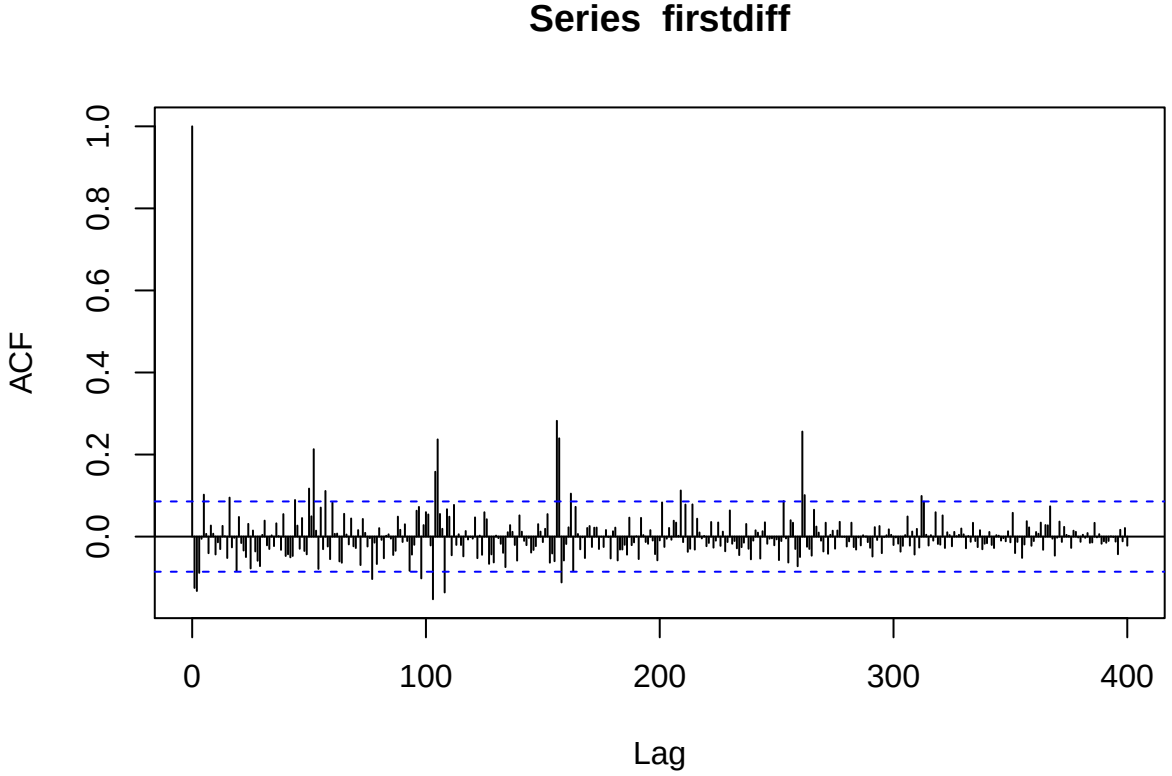



```
# the acfIndex indicates indices and values of the n highest acf values, and found a lot of them are ye

firstdiff.52 <- diff(firstdiff, lag = 26)
```

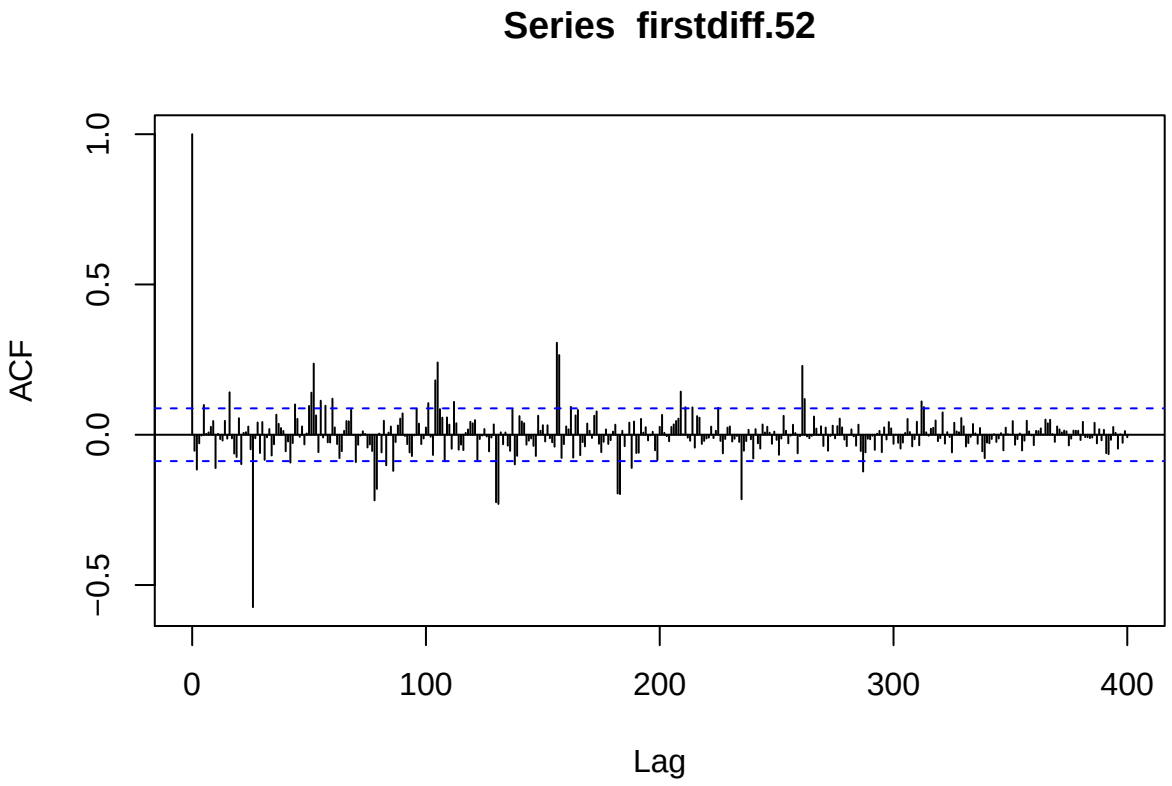
```r
plot(firstdiff.52, type = 'l', xlab = "Time", ylab = "Value", main = "Seasonal and 1st Diff Data")
```



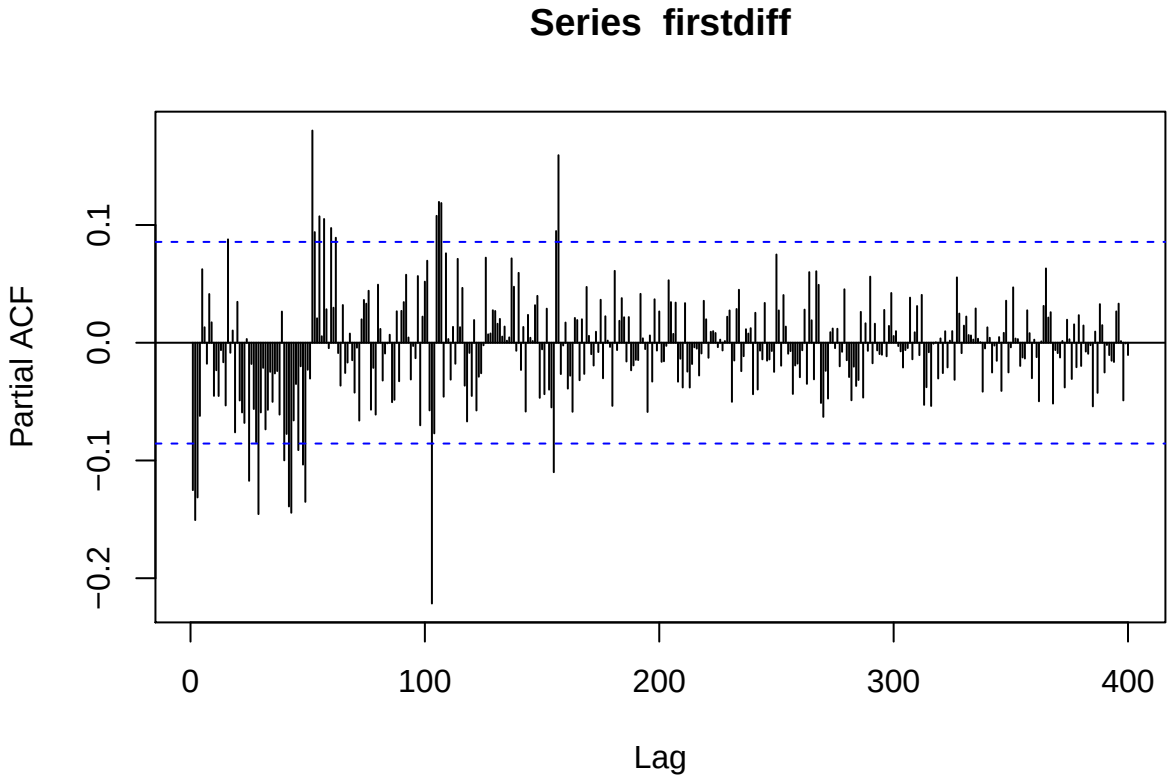

```r
#par(mfrow = c(2,1))
acf(firstdiff, lag.max = 400)
```



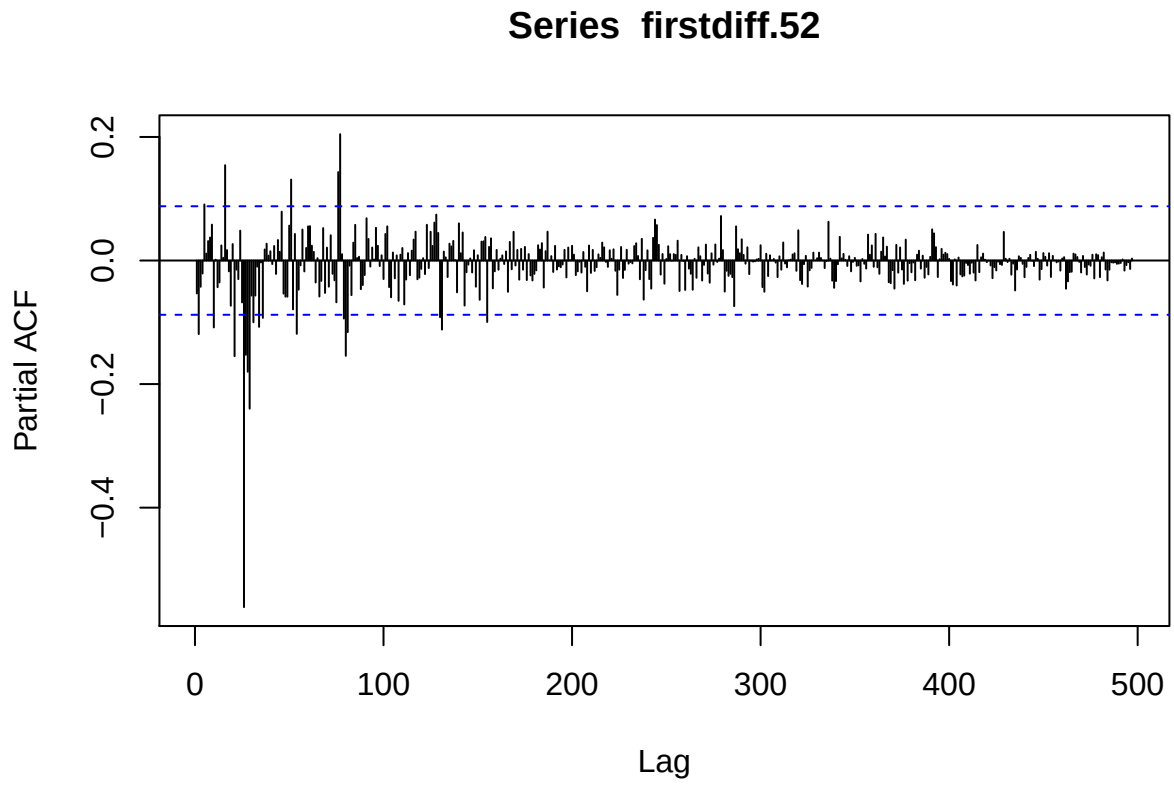
```r
acf(firstdiff.52, lag.max = 400)
```



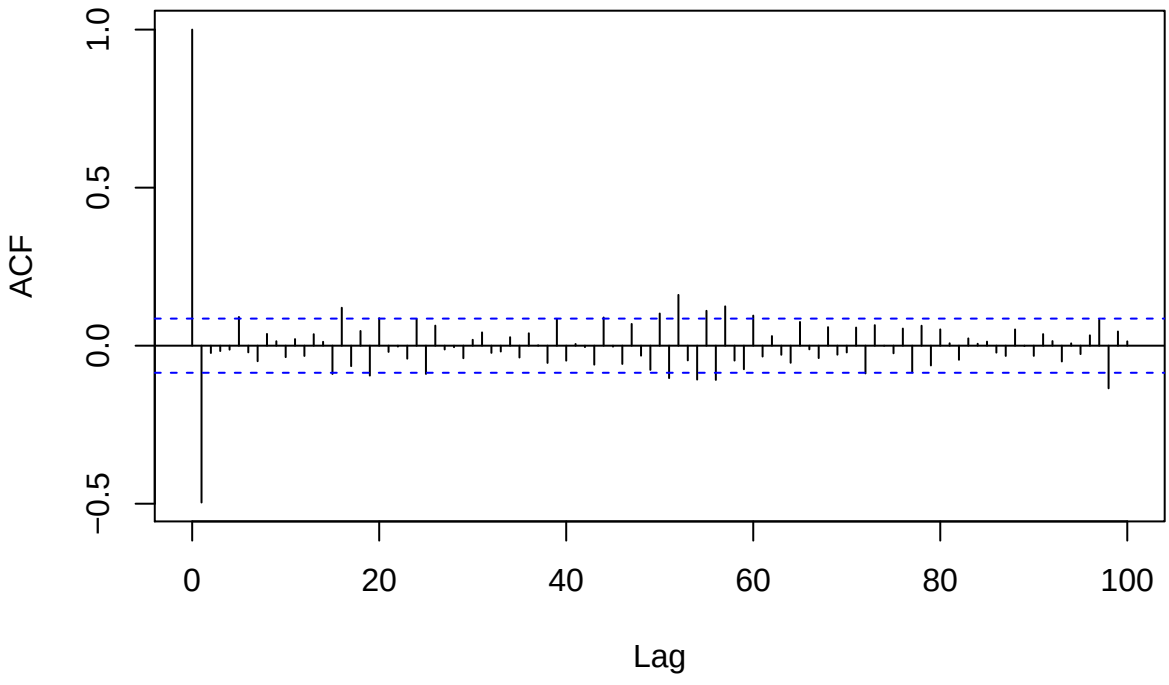

```r
#par(mfrow = c(2,1))
pacf(firstdiff, lag.max = 400)
```



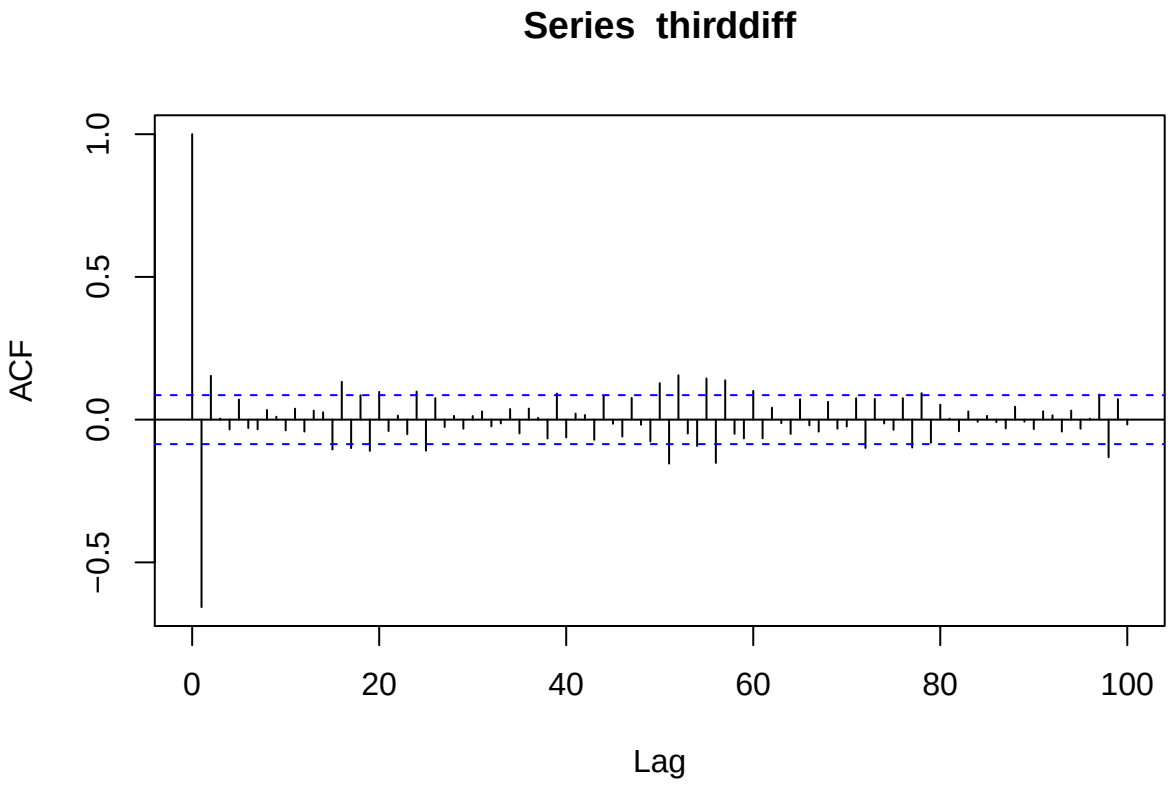
```r
pacf(firstdiff.52, lag.max = 500)
```



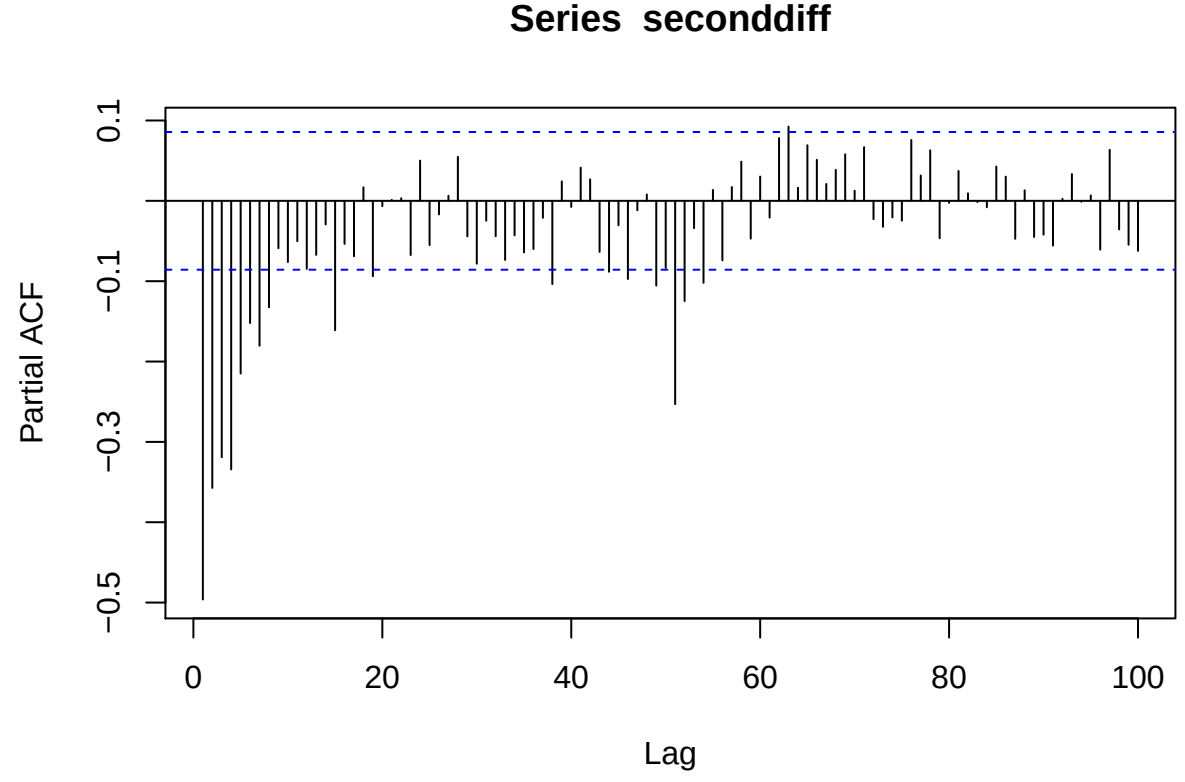

```r
#par(mfrow = c(2,1))
acf(seconddiff, lag.max = 100)
```



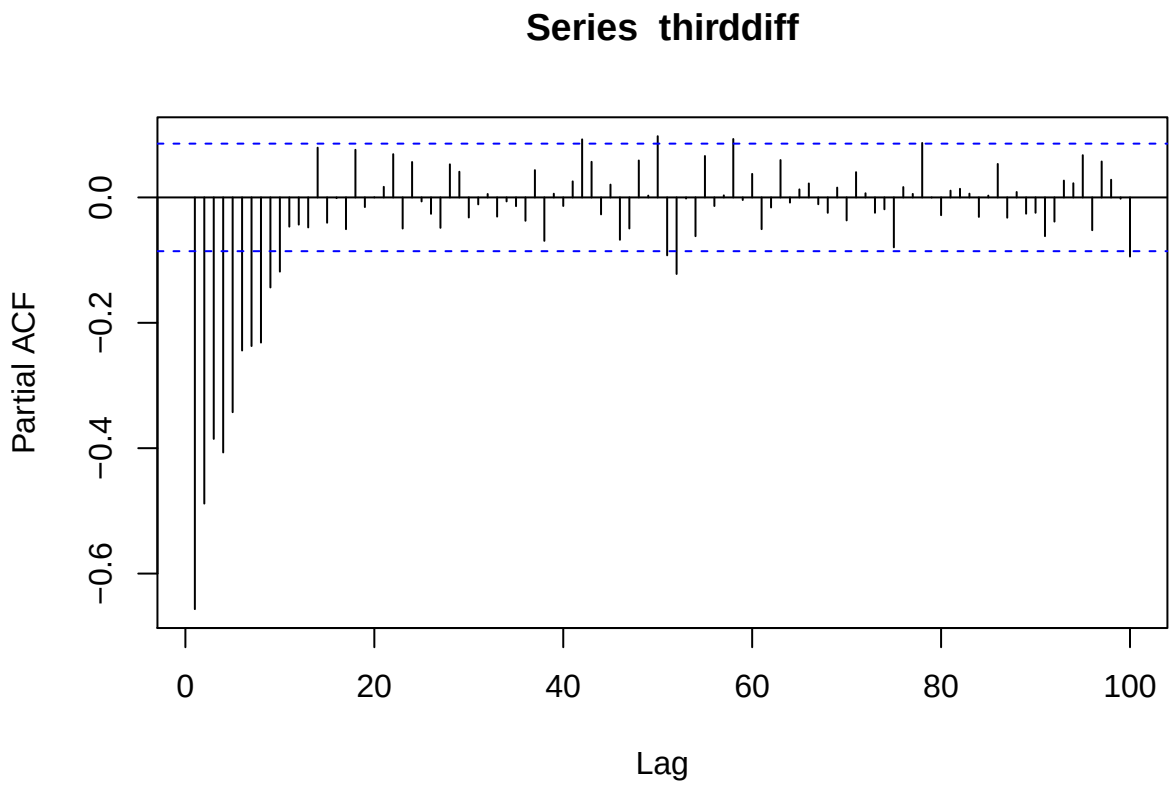
```r
acf(thirddiff, lag.max = 100)
```




```r
#par(mfrow = c(2,1))
pacf(seconddiff, lag.max = 100)
```



```r
pacf(thirddiff, lag.max = 100)
```

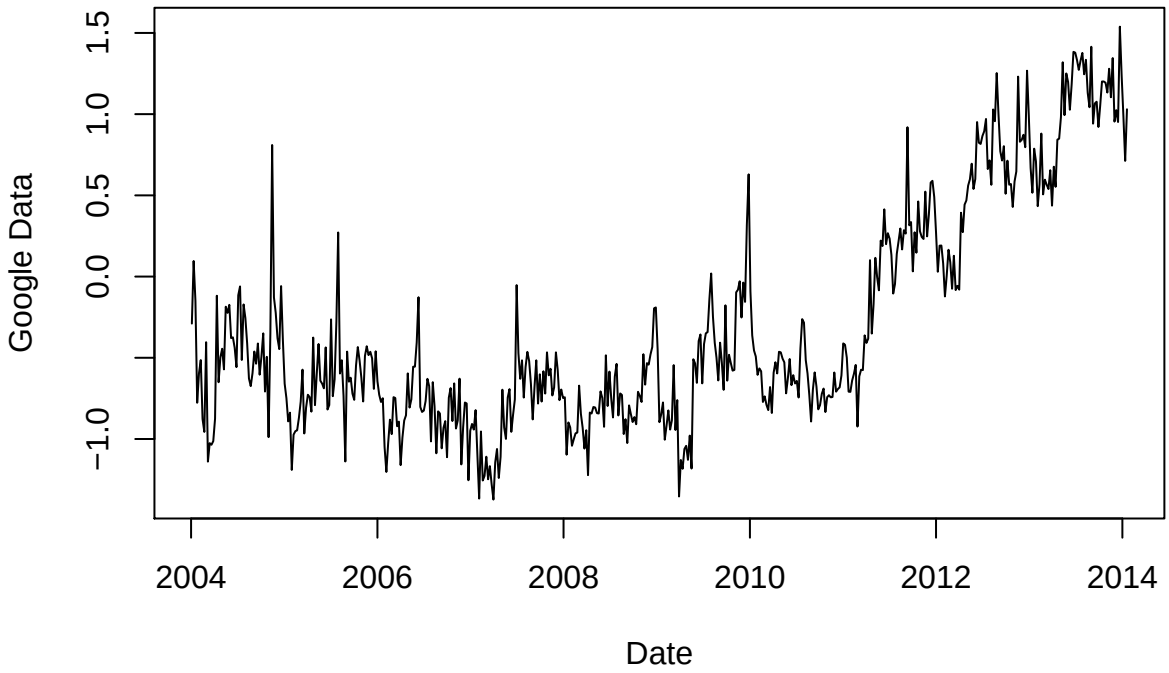



```r
# Next step : see if there is seasonality in differencing
```

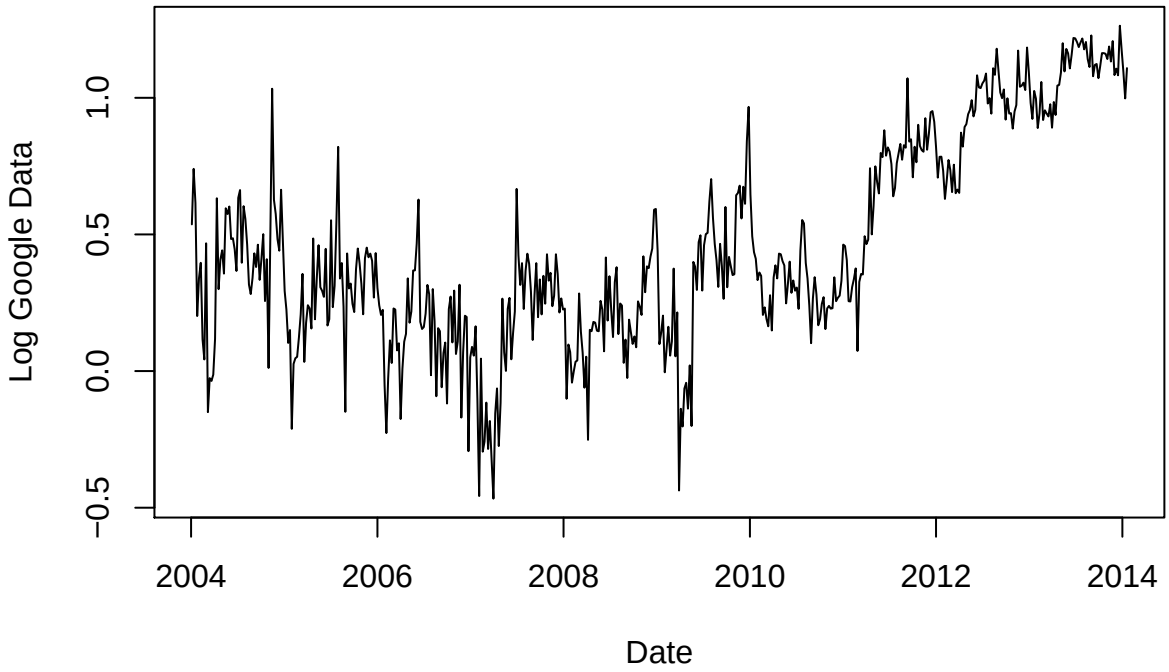
- residuals of plot
- doing adjusted R^2, AIC, BIC, differencing
- polynomial, exponential fit

Question 4

```r
Q4Train <- data$Q4Train

plot(Q4Train, type = 'l', xlab = "Date", ylab = "Google Data")
```

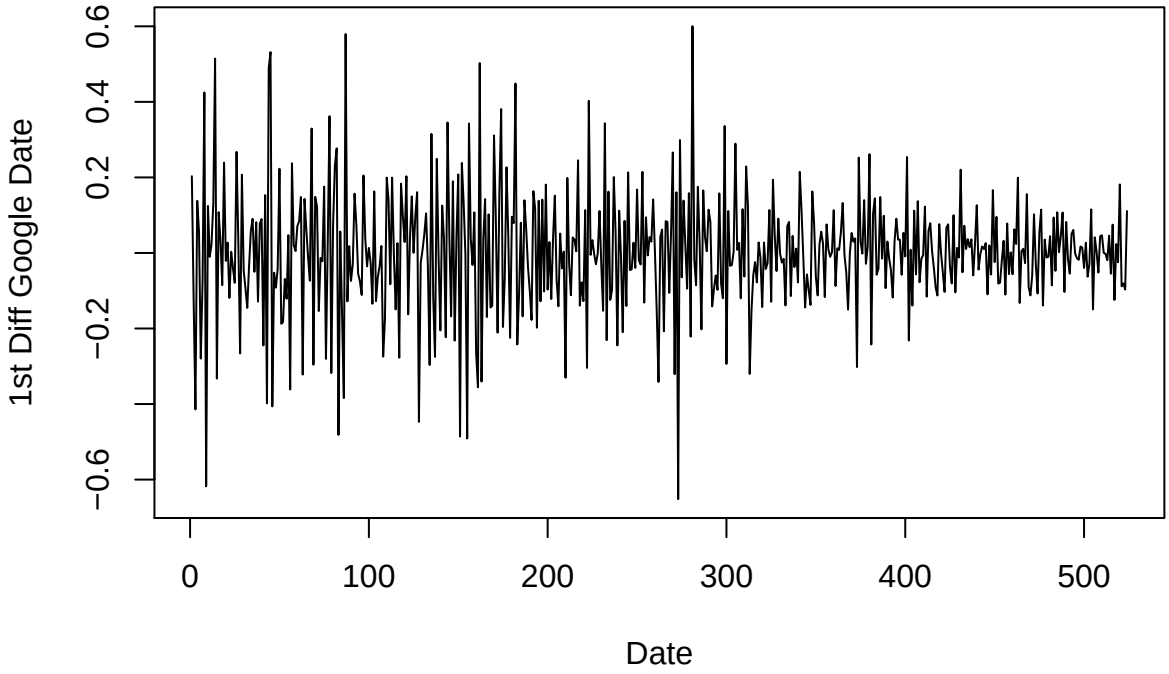



```
Q4Train.Log <- data.frame(Date = Q4Train$Date, Activity = log(Q4Train$activity + 2))
plot(Q4Train.Log, type = 'l', xlab = "Date", ylab = "Log Google Data")
```

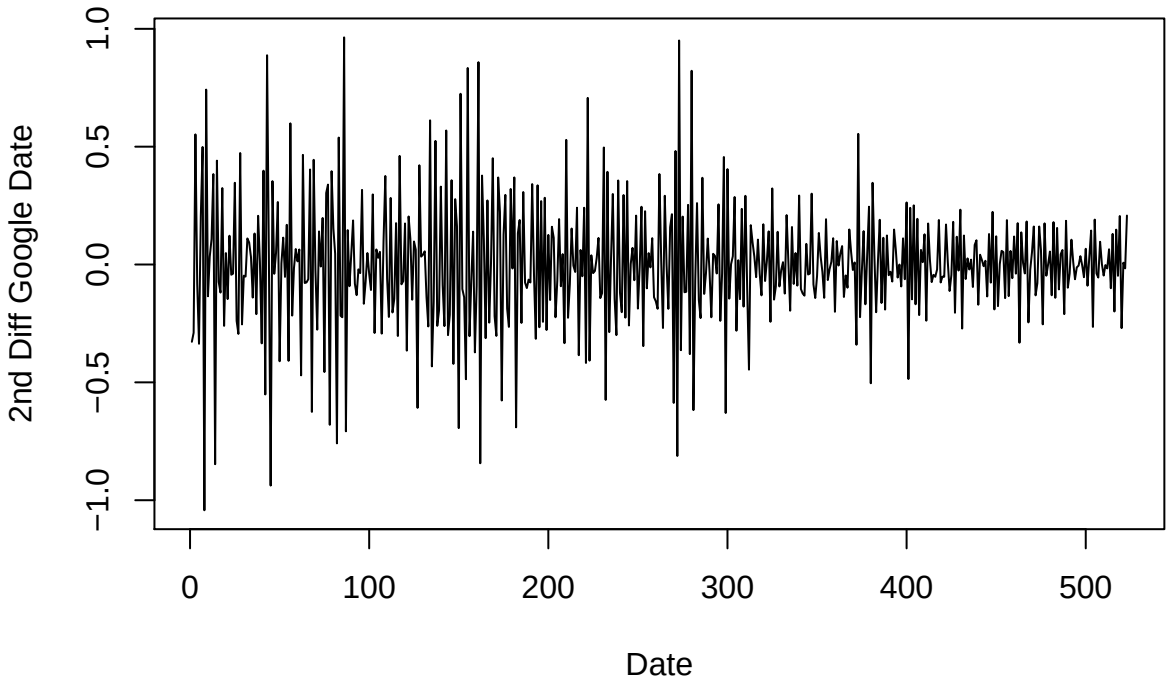



```
q4train.log <- Q4Train.Log$Activity

# Observe first and second differenced log data
q4firstdiff <- diff(q4train.log)
q4seconddiff <- diff(diff(q4train.log))

# Observe differenced data of orders 1,2
#par(mfrow = c(2,1))
```

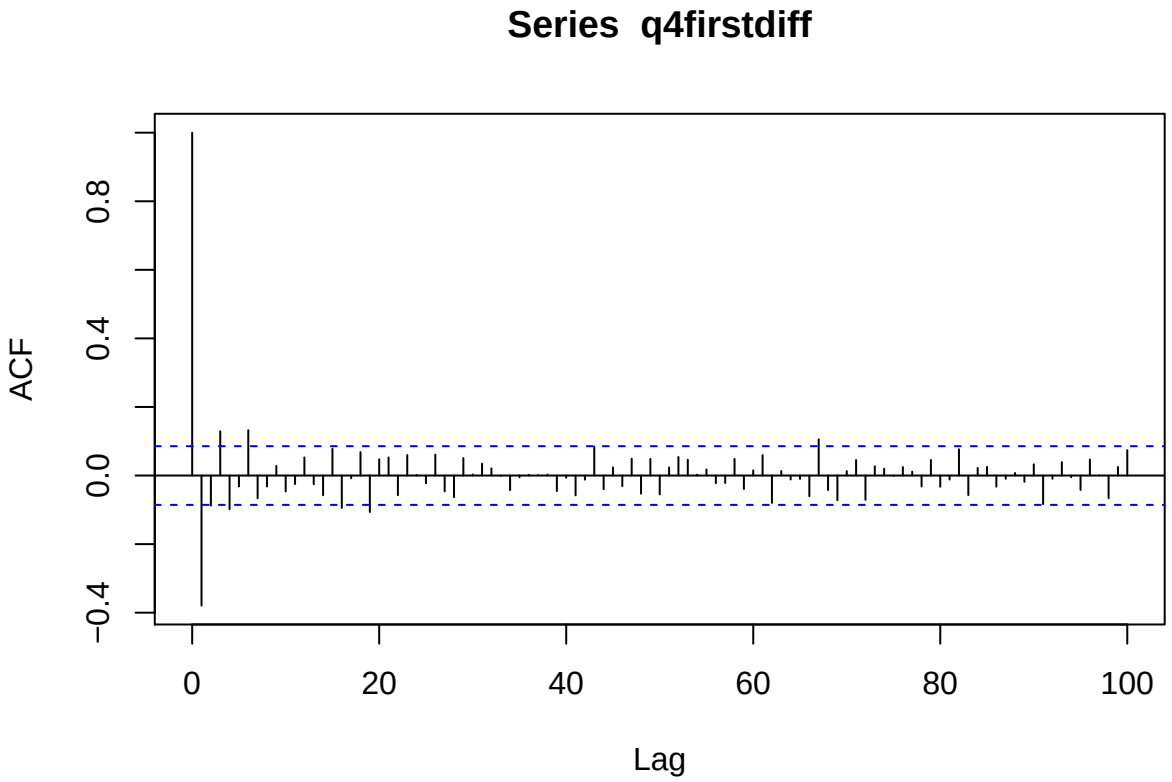
```r
plot(q4firstdiff, type = 'l', xlab = "Date", ylab = "1st Diff Google Date");
```



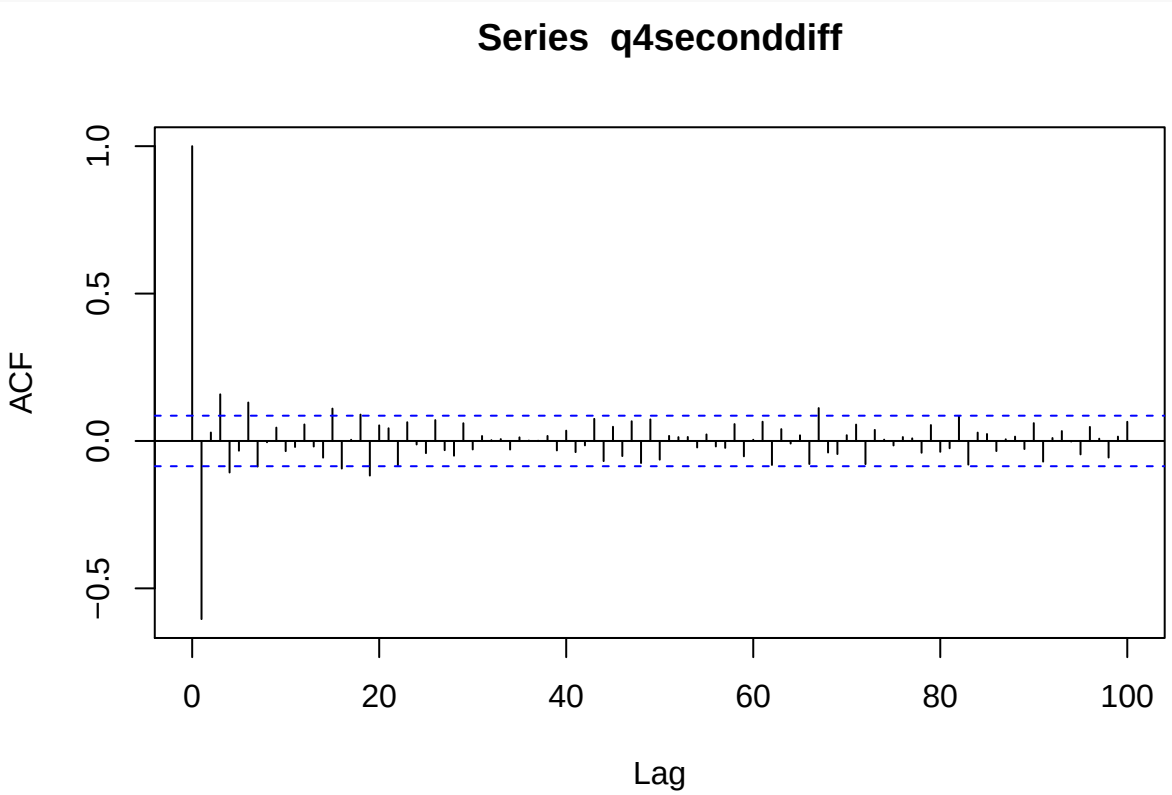

```r
plot(q4seconddiff, type = 'l', xlab = "Date", ylab = "2nd Diff Google Date")
```



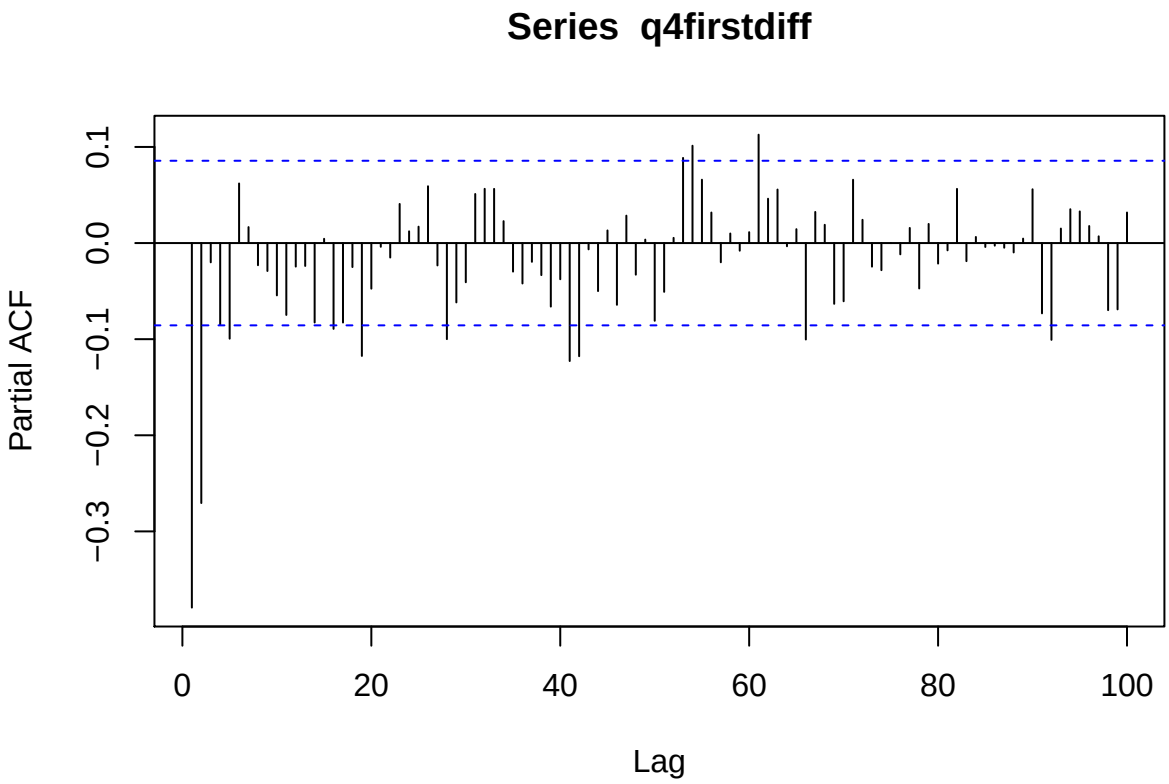

```r
# Observe acf of data of orders 1, 2
#par(mfrow = c(2,1))
acf(q4firstdiff, lag.max = 100)
```



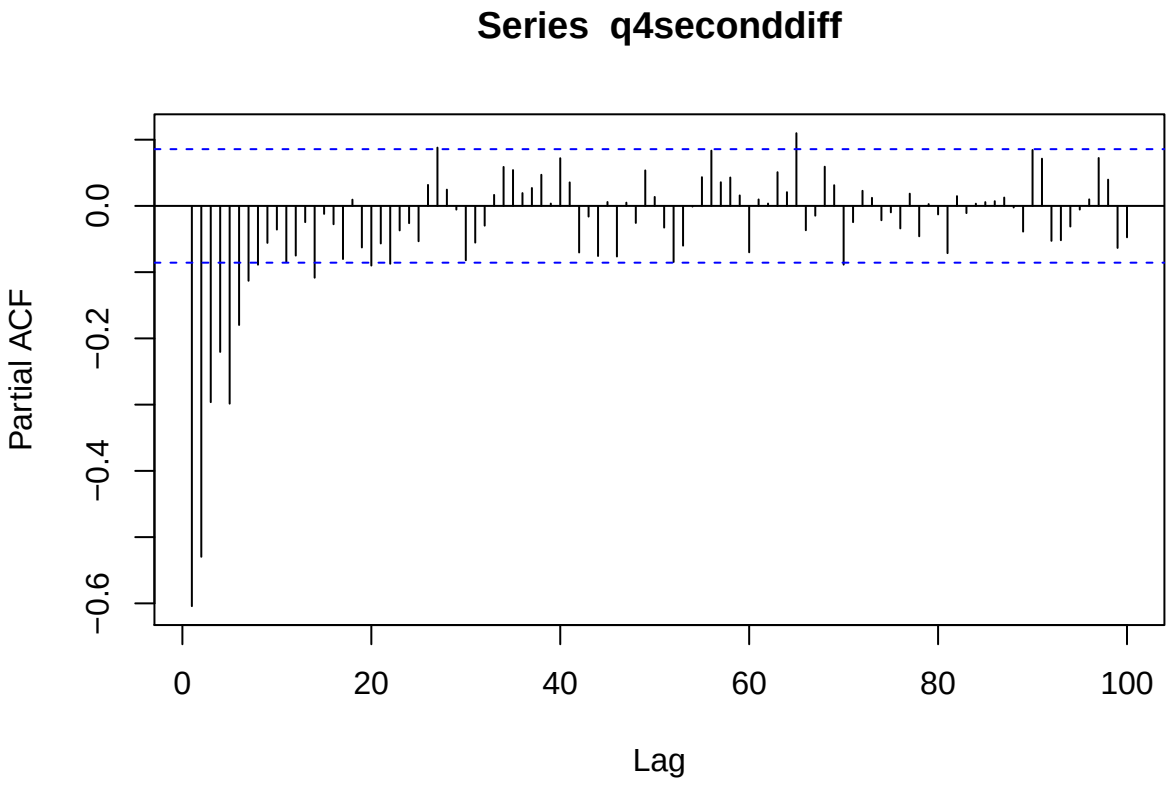

```
acf(q4seconddiff, lag.max = 100)
```




```
#par(mfrow = c(2,1))
pacf(q4firstdiff, lag.max = 100)
```




```
pacf(q4seconddiff, lag.max = 100)
```



Creating the Submission File

```
writeData <- function(dataset, q.num, firstname, lastname, SID) {
  output <- write.table(dataset,
            sep = ",",
            col.names = FALSE,
            row.names = FALSE,
            file = paste0("Q",q.num,"_",
                          firstname,"_",
                          lastname,"_",SID))
  return(output)
}
```