DaigleInClassLabWk13D2.R

2011home

Wed Apr 18 22:13:41 2018

```r
# Chris Daigle
# Econ5495 - R Programming
# Wk13D2 In Class Lab - Prediction

# Importing & Setup####
setwd(
  "/Users/2011home/Library/Mobile Documents/com~apple~CloudDocs/Education/UConn/Spring 2018/R/DataSets"
)
gdp <- read.csv('GDP.csv', stringsAsFactors = FALSE)
library(tseries)
```

```
## Warning: package 'tseries' was built under R version 3.4.4
```

```r
head(gdp)
```

```
##         DATE     GDP
## 1 1947-01-01 243.080
## 2 1947-04-01 246.267
## 3 1947-07-01 250.115
## 4 1947-10-01 260.309
## 5 1948-01-01 266.173
## 6 1948-04-01 272.897
```

```r
str(gdp)
```

```
## 'data.frame':    284 obs. of  2 variables:
##  $ DATE: chr  "1947-01-01" "1947-04-01" "1947-07-01" "1947-10-01" ...
##  $ GDP : num  243 246 250 260 266 ...
```

```r
gdp$DATE <- as.Date(gdp$DATE)
gdp$Time <- format(gdp$DATE, format = '%y/%m')
gdp <- gdp[, c(3, 2)]

# Time Setting ####
gdp <-
  ts(
    gdp$GDP,
    start = c(1947, 1),
    end = c(2017, 4),
    frequency = 4
  ) # Quarterly data

# Summary Statistics ####
str(gdp)
```

```
##  Time-Series [1:284] from 1947 to 2018: 243 246 250 260 266 ...
```

```r
start(gdp)
```

```
## [1] 1947    1
```

```r
head(gdp)
```

```
## [1] 243.080 246.267 250.115 260.309 266.173 272.897
```

```r
end(gdp)
```

```
## [1] 2017    4
```

```r
tail(gdp)
```

```
## [1] 18729.13 18905.54 19057.71 19250.01 19500.60 19754.10
```

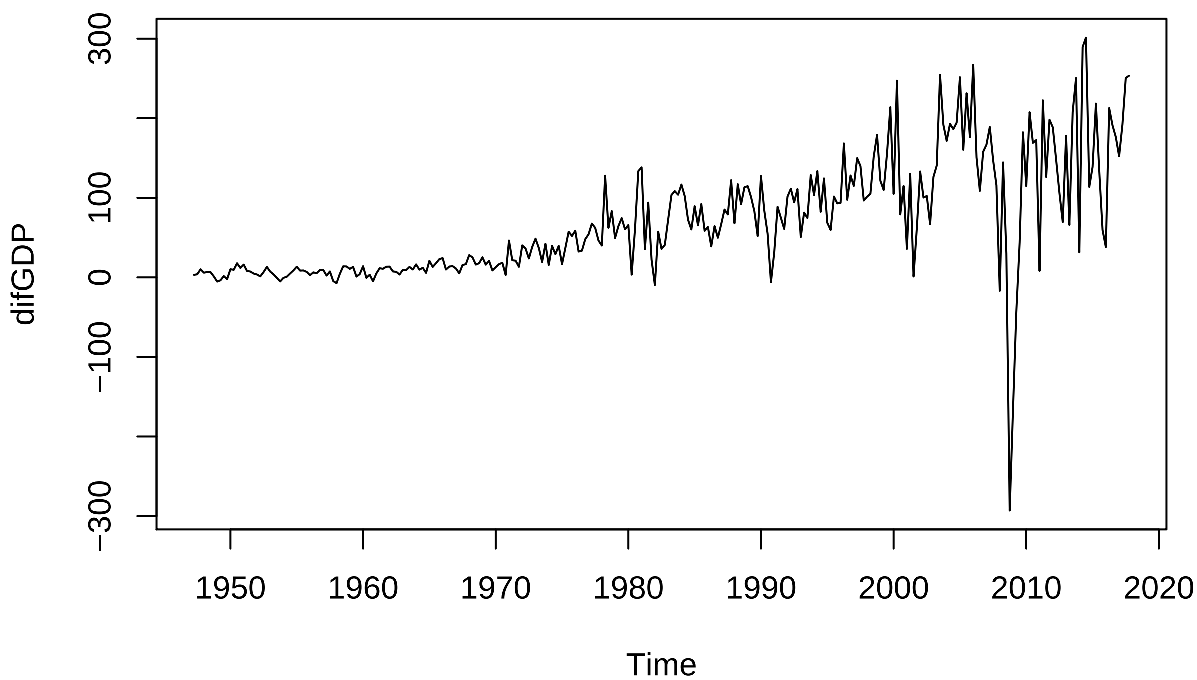
```r
frequency(gdp)
```

```
## [1] 4
```

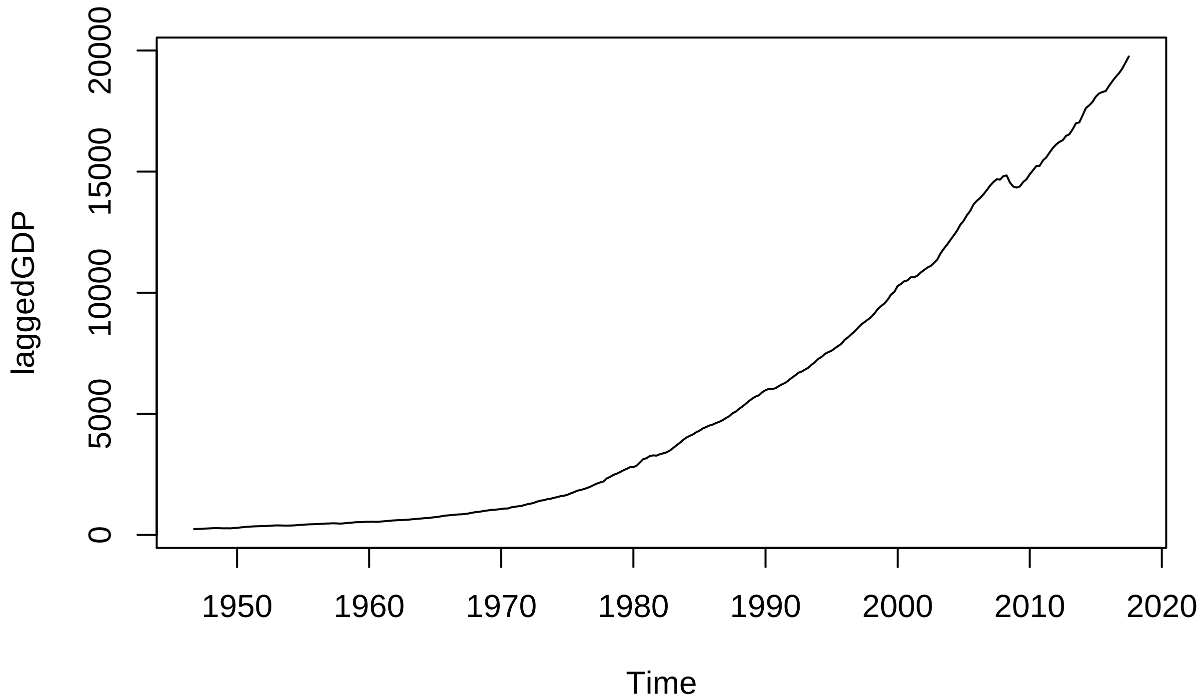
```r
summary(gdp)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   243.1   697.0  3349.2  5781.1 10092.9 19754.1
```

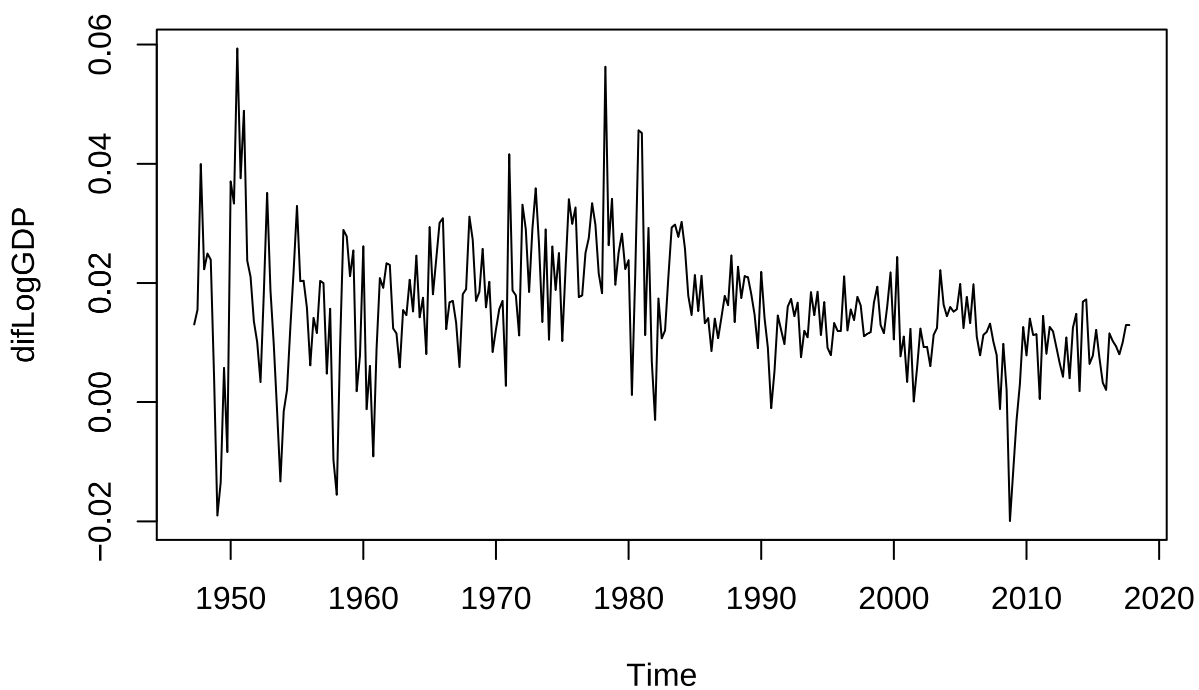
```r
# Plotting ####
laggedGDP <- lag(gdp, 1)
difGDP <- diff(gdp, 1) # Y(t) - Y(t-1)
difLogGDP <- diff(log(gdp))
plot(difGDP)
```



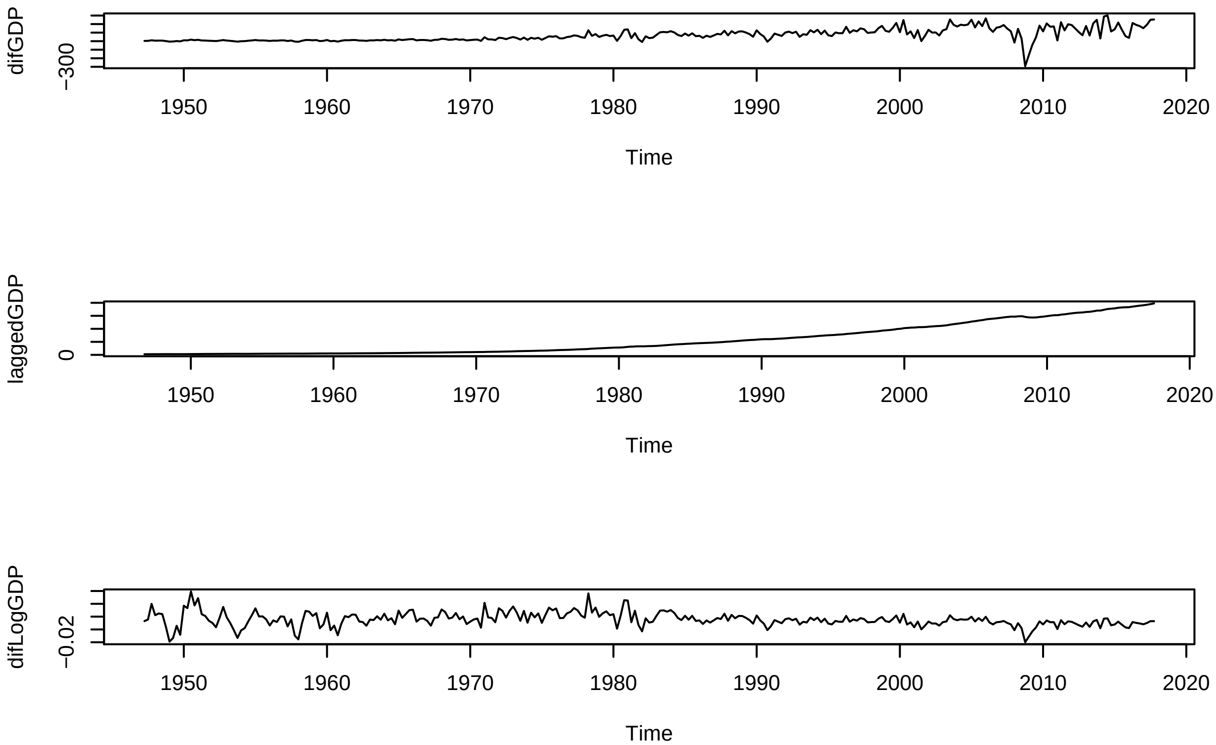

```r
plot(laggedGDP)
```



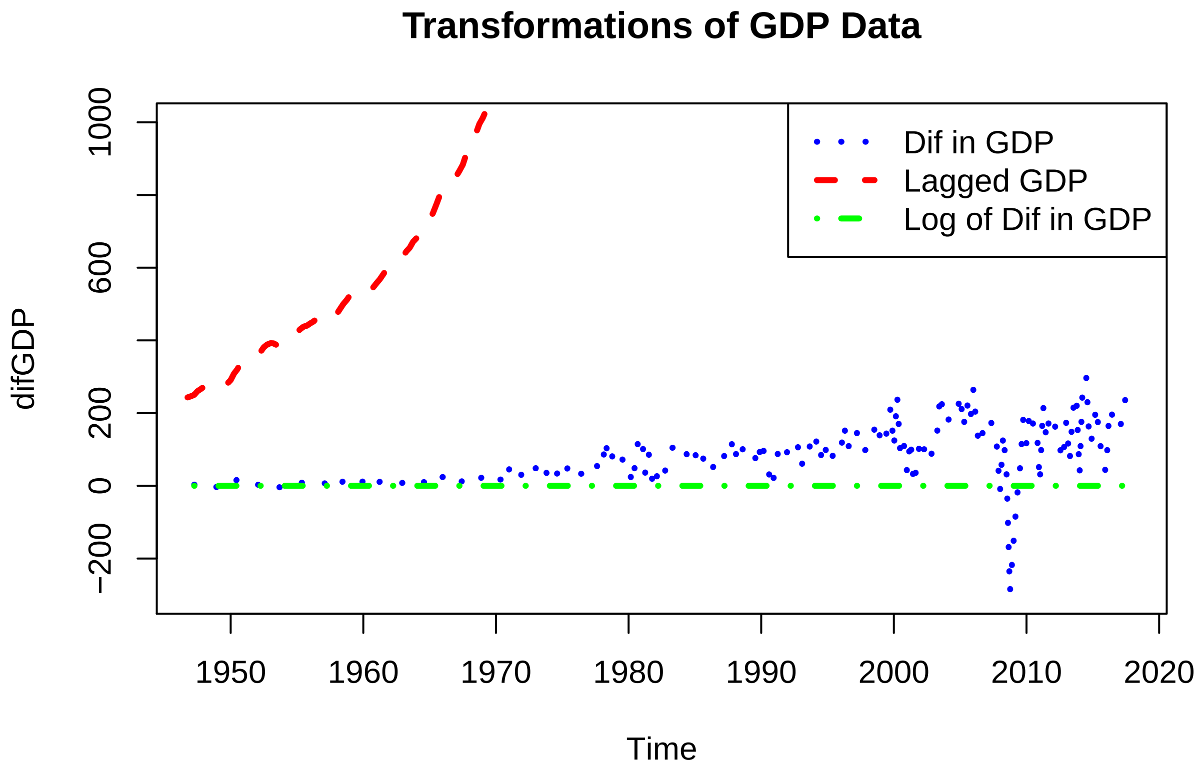

```
plot(difLogGDP)
```



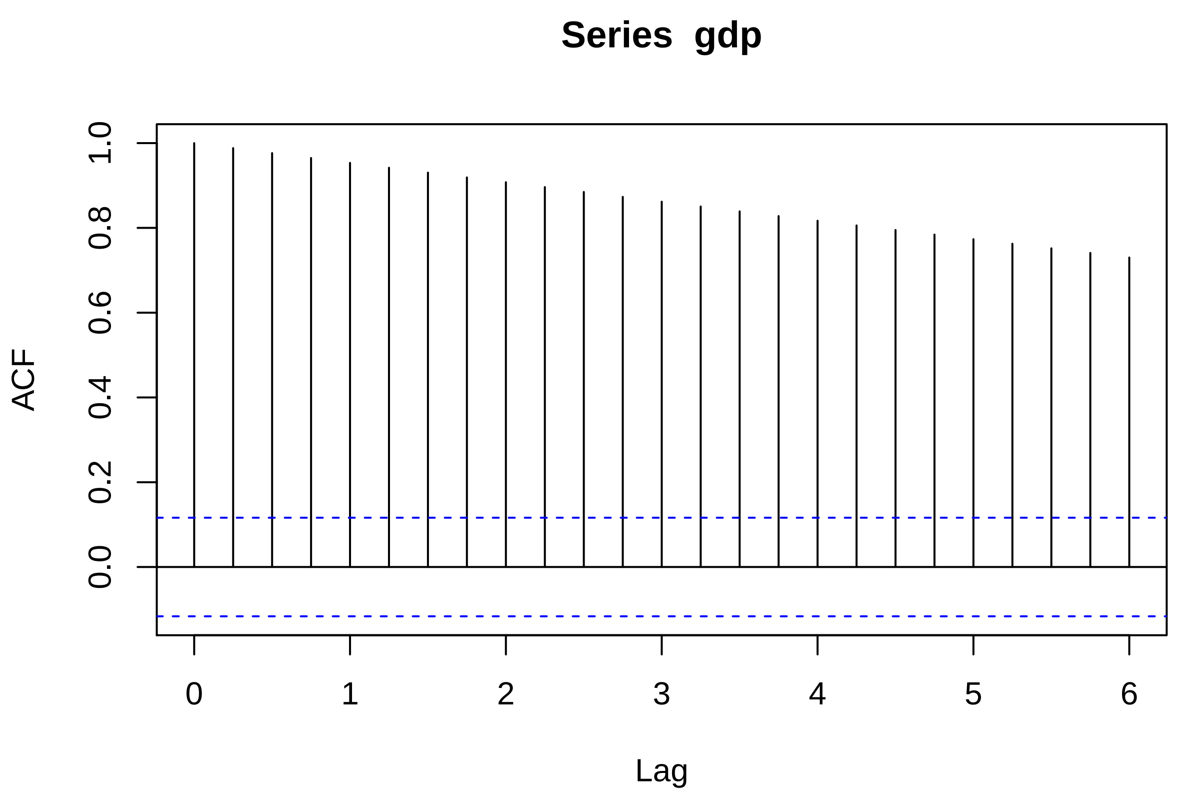

```
par(mfrow = c(3,1))
plot(difGDP)
plot(laggedGDP)
plot(difLogGDP)
```



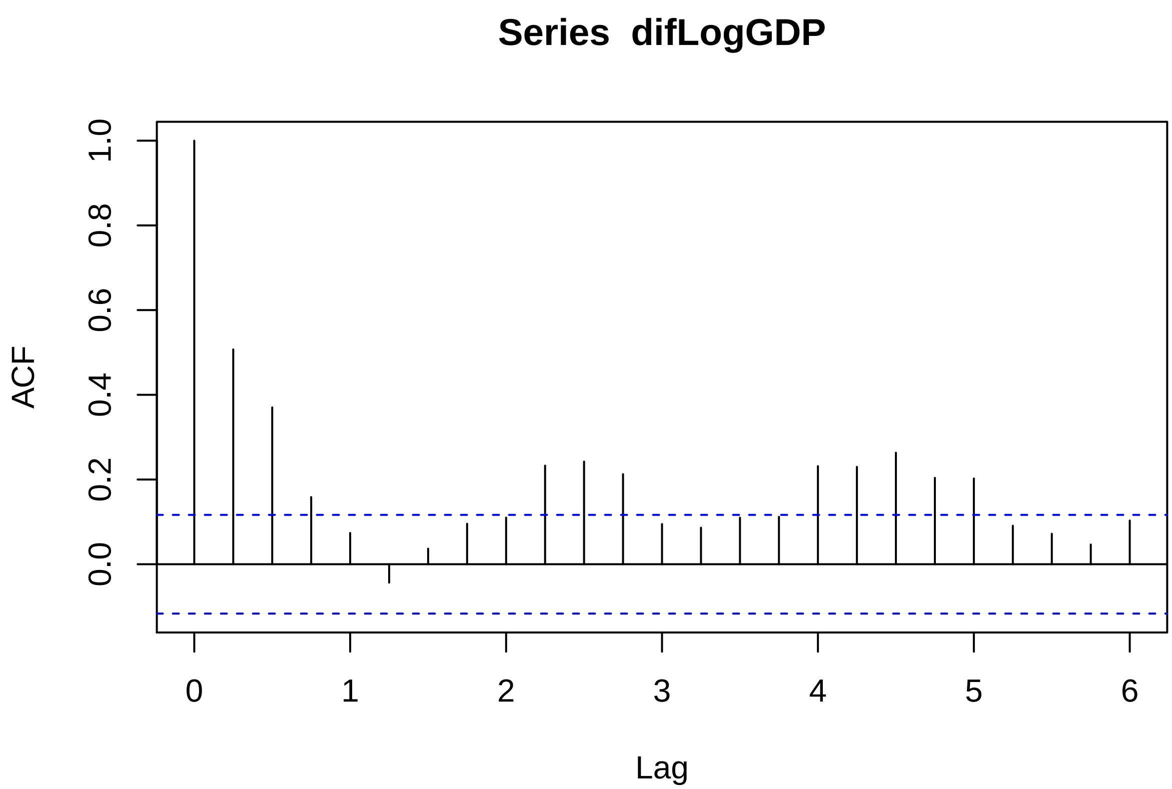

```
par(mfrow = c(1,1))

plot(difGDP, col = 'blue', ylim = c(-300, 1000), lty = 3, lwd = 3)
lines(laggedGDP, col = 'red', lty = 2, lwd = 3)
lines(difLogGDP, col = 'green', lty = 4, lwd = 3)
legend('topright', legend = c('Dif in GDP', 'Lagged GDP', 'Log of Dif in GDP'), col = c('blue', 'red',
title(main = 'Transformations of GDP Data')
```



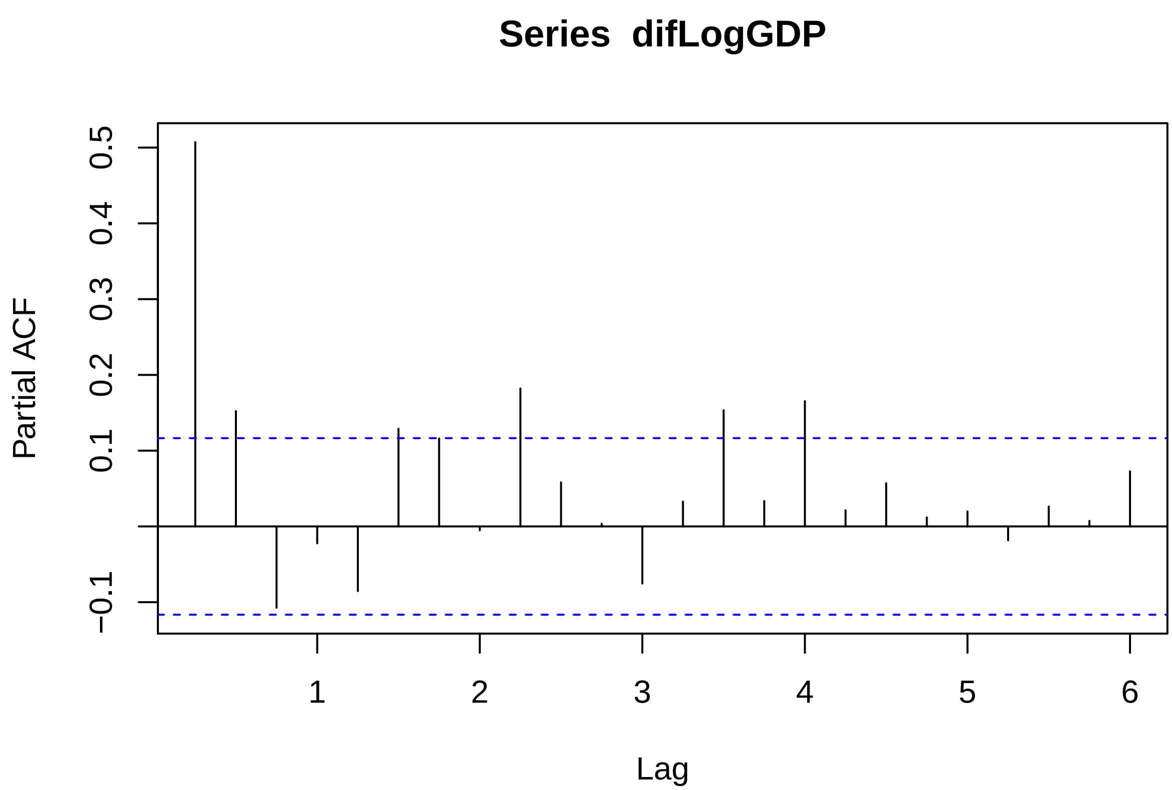

```
acf(gdp) # high serial correlation
```




```
acf(difLogGDP) # explanatory value of the first, second, and third term is very small (not useful)
```

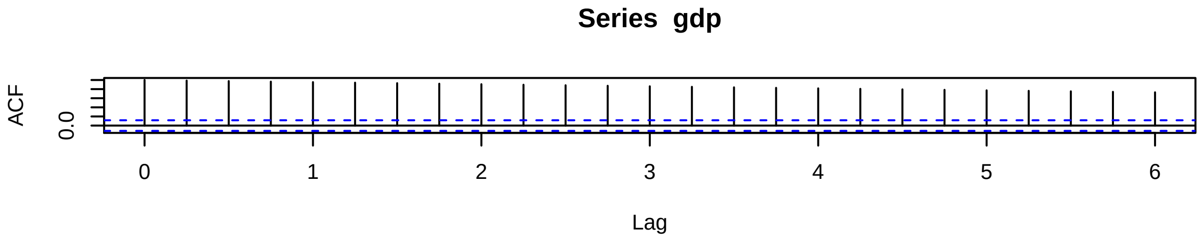


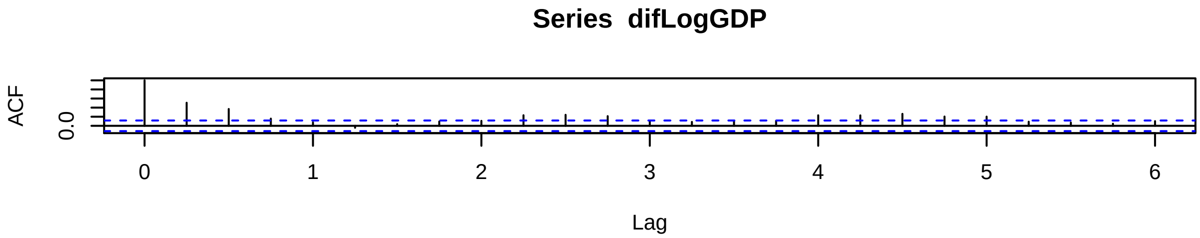

```r
pacf(difLogGDP) # explanatory power of first isn't great, but the following are better
```

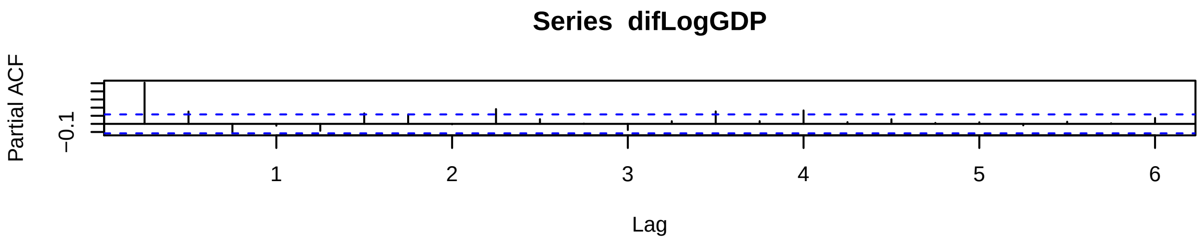



```r
par(mfrow = c(3,1))
acf(gdp)
```

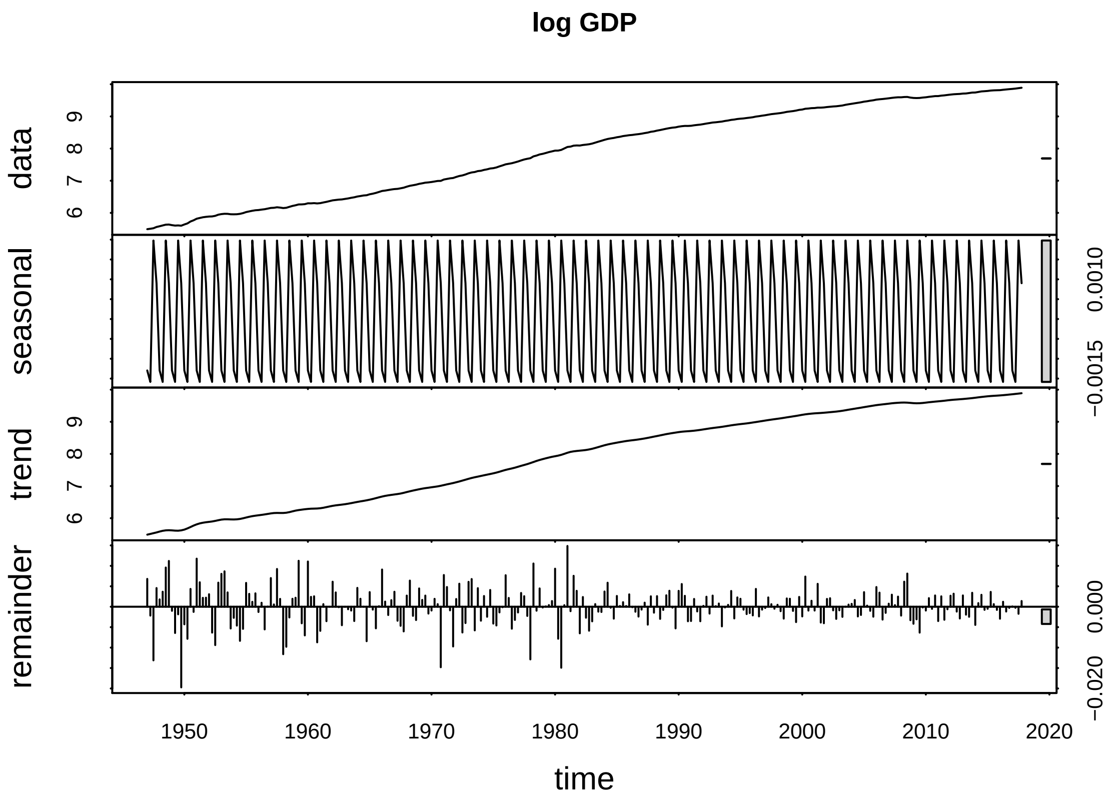
```
acf(difLogGDP)
pacf(difLogGDP)
```



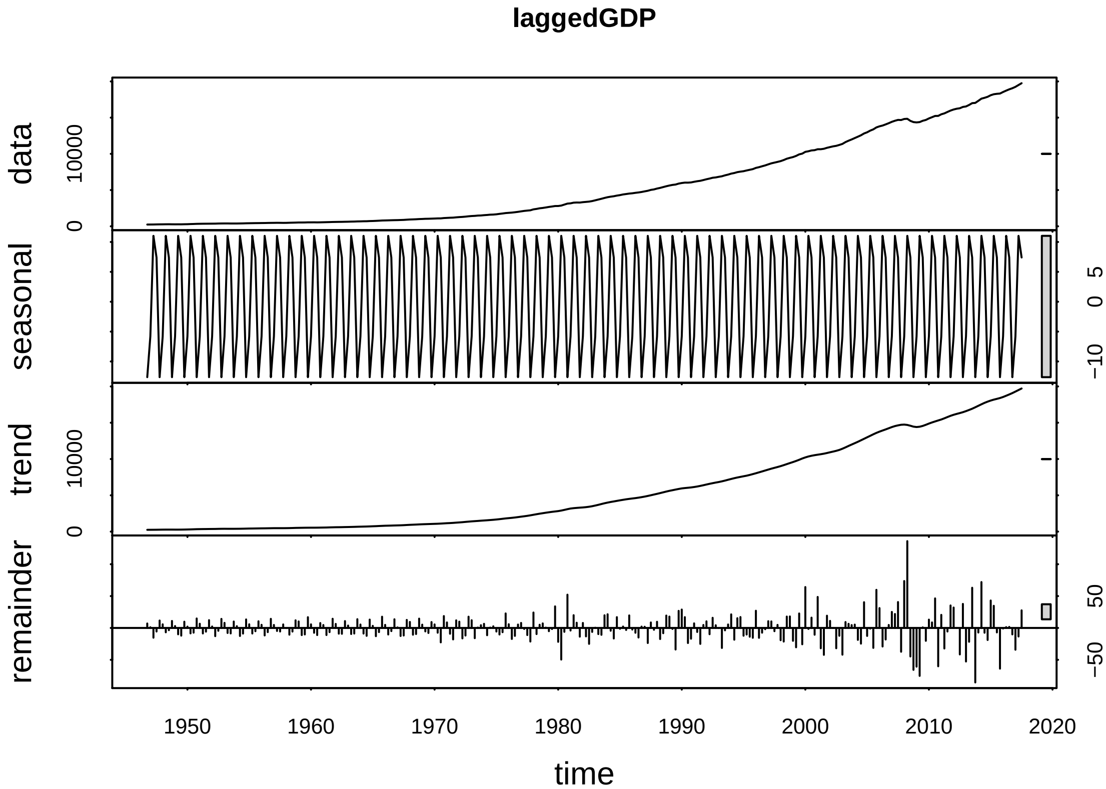




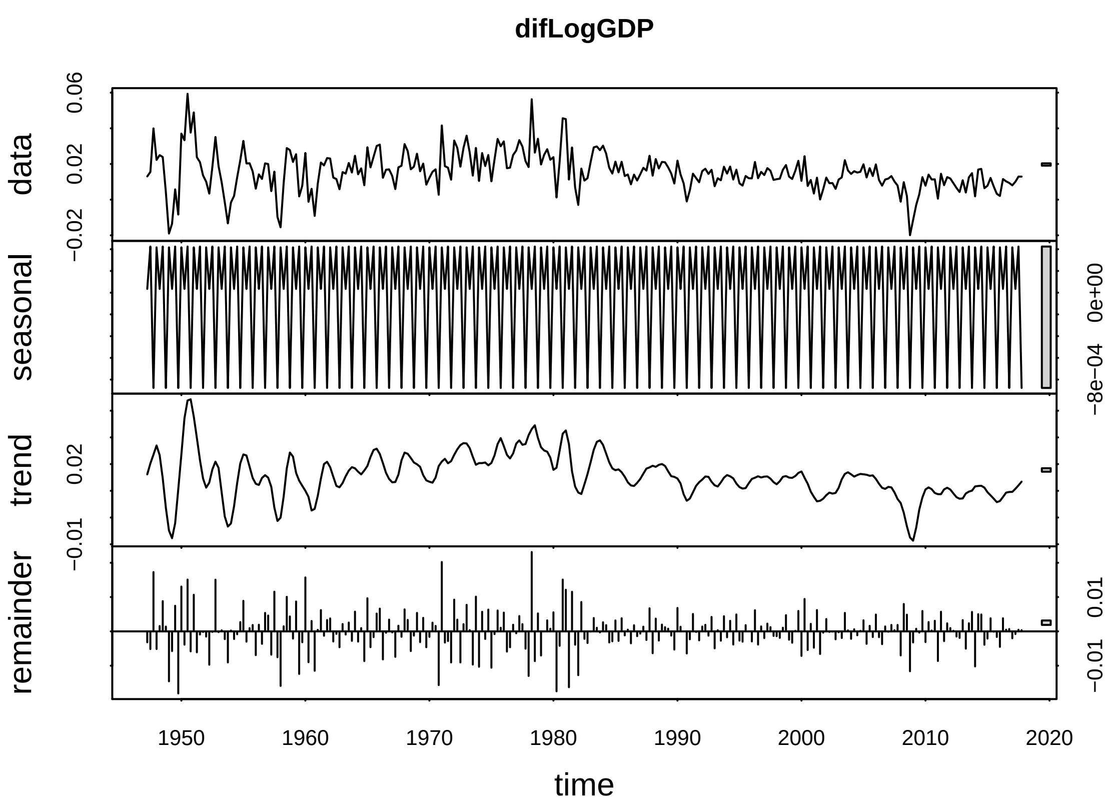

```
seasDecom <- stl(log(gdp), s.window = 'period')
plot(seasDecom, lwd = 1, main = 'log GDP')
```




```r
# Because there is a trend line, we can immediately see the data is
# non-stationary (dynamic?). Further, we can perform a hypothesis test such that
# the H0: data is non-stationary (ß = 1)

seasDecom1 <- stl(laggedGDP, s.window = 'period')
plot(seasDecom1, lwd = 1, main = 'laggedGDP')
```




```r
seasDecom2 <- stl(difLogGDP, s.window = 'period')
plot(seasDecom2, lwd = 1, main = 'difLogGDP')
```



```r
adf.test(gdp, alternative = 'stationary')
```

```
## Warning in adf.test(gdp, alternative = "stationary"): p-value greater than
## printed p-value
```

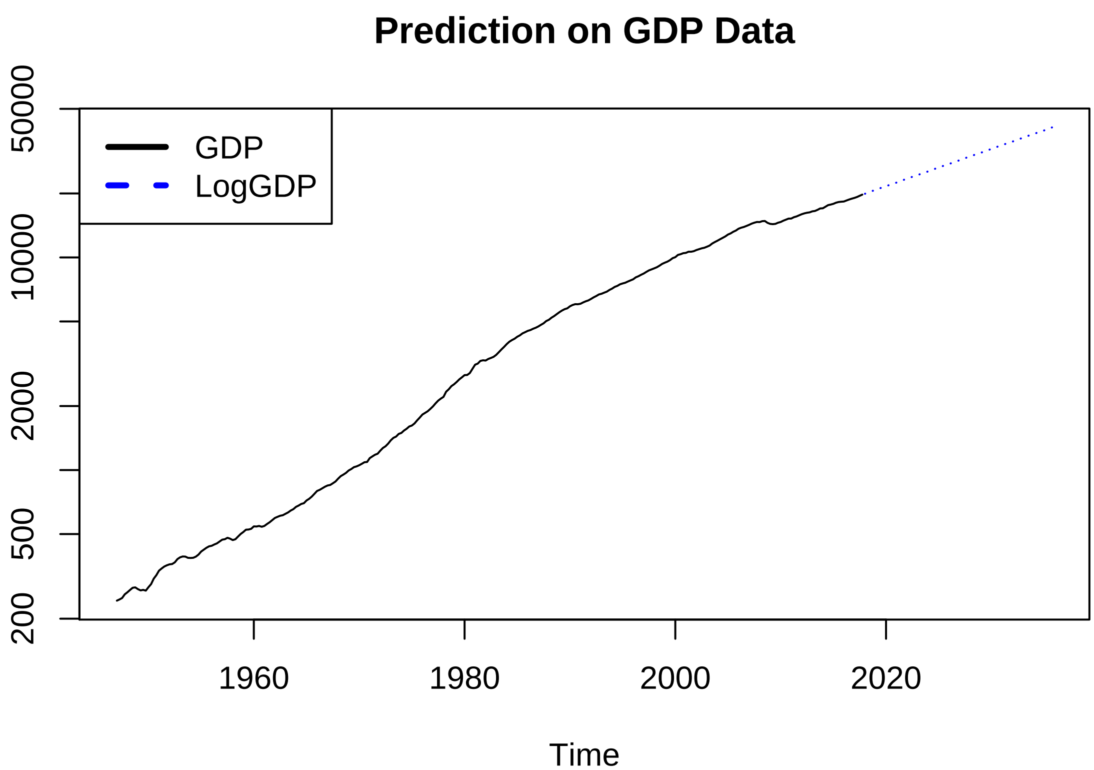
```
##
##  Augmented Dickey-Fuller Test
##
## data:  gdp
## Dickey-Fuller = 0.43299, Lag order = 6, p-value = 0.99
## alternative hypothesis: stationary
```

```r
# Fail to reject the null (p > 0.05), so it's likely non-stationary The value is
# so large, it's essentially 1 for p, so it is, with near certainty,
# non-stationary
adf.test(difLogGDP, alternative = 'stationary')
```

```
## Warning in adf.test(difLogGDP, alternative = "stationary"): p-value smaller
## than printed p-value
```

```
##
##  Augmented Dickey-Fuller Test
##
## data:  difLogGDP
## Dickey-Fuller = -5.0524, Lag order = 6, p-value = 0.01
## alternative hypothesis: stationary
```

```r
# Reject the null (p <0.05), so the data is stationary. In this case, the value
# is so small, it is with near certainty, stationary.

par(mfrow = c(1,1))
fit <- arima(log(gdp), c(0,1,1), seasonal = list(order = c(0,1,1), period = 4)) # first argument is the
pred <- predict(fit, n.ahead = 18*4) # two years by 12 months
ts.plot(gdp, exp(pred$pred), log = 'y', lty = c(1,3), col = c('black', 'blue'), lwd = 1)
legend('topleft', legend = c('GDP', 'LogGDP'), col = c('black', 'blue'), lty = c(1, 2), lwd = 3)
title(main = 'Prediction on GDP Data')
```




```
fit <- arima(difLogGDP, c(0,1,1), seasonal = list(order = c(0,1,1), period = 4)) # first argument is th
pred <- predict(fit, n.ahead = 18*4) # two years by 12 months
ts.plot(difLogGDP, pred$pred, lty = c(1,3), col = c('black', 'blue'), lwd = 1)
legend('topright', legend = c('difLogGDP', 'LogGDP'), col = c('black', 'blue'), lty = c(1, 2), lwd = 3)
title(main = 'Prediction on GDP Data')
```