ENV 790.30 - Time Series Analysis for Energy Data | Spring 2023

Assignment 3 - Due date 02/10/23

Katherine Burley

Directions

You should open the .rmd file corresponding to this assignment on RStudio. The file is available on our class repository on Github.

Once you have the file open on your local machine the first thing you will do is rename the file such that it includes your first and last name (e.g., "LuanaLima_TSA_A02_Sp23.Rmd"). Then change "Student Name" on line 4 with your name.

Then you will start working through the assignment by **creating code and output** that answer each question. Be sure to use this assignment document. Your report should contain the answer to each question and any plots/tables you obtained (when applicable).

Please keep this R code chunk options for the report. It is easier for us to grade when we can see code and output together. And the tidy.opts will make sure that line breaks on your code chunks are automatically added for better visualization.

When you have completed the assignment, **Knit** the text and code into a single PDF file. Submit this pdf using Sakai.

Questions

Consider the same data you used for A2 from the spreadsheet "Table_10.1_Renewable_Energy_Production_and_Consumption The data comes from the US Energy Information and Administration and corresponds to the December 2022 **Monthly** Energy Review. Once again you will work only with the following columns: Total Biomass Energy Production, Total Renewable Energy Production, Hydroelectric Power Consumption. Create a data frame structure with these three time series only.

R packages needed for this assignment:"forecast","tseries", and "Kendall". Install these packages, if you haven't done yet. Do not forget to load them before running your script, since they are NOT default packages.\

```
#Load/install required package here
library(dplyr)
```

```
##
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':
##
##     filter, lag

## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```r
library(forecast)
```

```
## Registered S3 method overwritten by 'quantmod':
##   method            from
##   as.zoo.data.frame zoo
```

```r
library(tseries)
library(readxl)
library(ggplot2)
library(cowplot)
```

##Trend Component

Bring in data

```r
#Importing data set
energy_data <- as.data.frame(read_excel(path="../Data/Table_10.1_Renewable_Energy_Production_and_Consump
                                        skip = 12, sheet="Monthly Data",col_names=FALSE))
```

```
## New names:
## * `` -> `...1`
## * `` -> `...2`
## * `` -> `...3`
## * `` -> `...4`
## * `` -> `...5`
## * `` -> `...6`
## * `` -> `...7`
## * `` -> `...8`
## * `` -> `...9`
## * `` -> `...10`
## * `` -> `...11`
## * `` -> `...12`
## * `` -> `...13`
## * `` -> `...14`
```

```r
  # Use as.data.frame because lm is not recognizing the df columns as a vector

#Extract the column names from row 11
read_col_names <- read_excel(path="../Data/Table_10.1_Renewable_Energy_Production_and_Consumption_by_Sou
                             skip = 10,n_max = 1, sheet="Monthly Data",col_names=FALSE)
```

```
## New names:
## * `` -> `...1`
## * `` -> `...2`
## * `` -> `...3`
## * `` -> `...4`
## * `` -> `...5`
## * `` -> `...6`
## * `` -> `...7`
## * `` -> `...8`
## * `` -> `...9`
## * `` -> `...10`
## * `` -> `...11`
## * `` -> `...12`
## * `` -> `...13`
```

```
## * `` -> `...14`
```

```r
colnames(energy_data) <- read_col_names

# Subset to Columns We Need
energy_data <- energy_data %>%
  select(c("Month", "Total Biomass Energy Production", "Total Renewable Energy Production", "Hydroelect
head(energy_data)
```

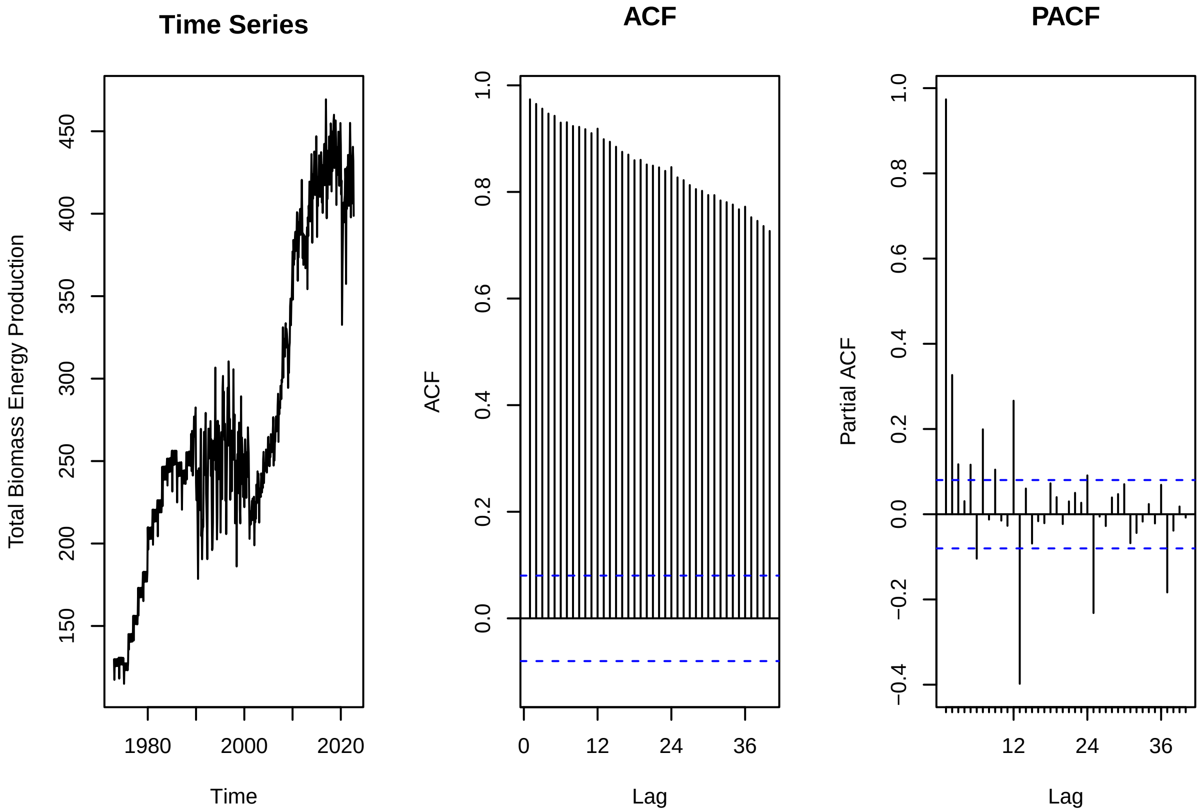
```
##        Month Total Biomass Energy Production Total Renewable Energy Production
## 1 1973-01-01                         129.787                           403.981
## 2 1973-02-01                         117.338                           360.900
## 3 1973-03-01                         129.938                           400.161
## 4 1973-04-01                         125.636                           380.470
## 5 1973-05-01                         129.834                           392.141
## 6 1973-06-01                         125.611                           377.232
##   Hydroelectric Power Consumption
## 1                        272.703
## 2                        242.199
## 3                        268.810
## 4                        253.185
## 5                        260.770
## 6                        249.859
```

```r
# Create time series object
ts_energy <- ts(energy_data, start=1973, frequency=12) # Monthly data, freq=12
```

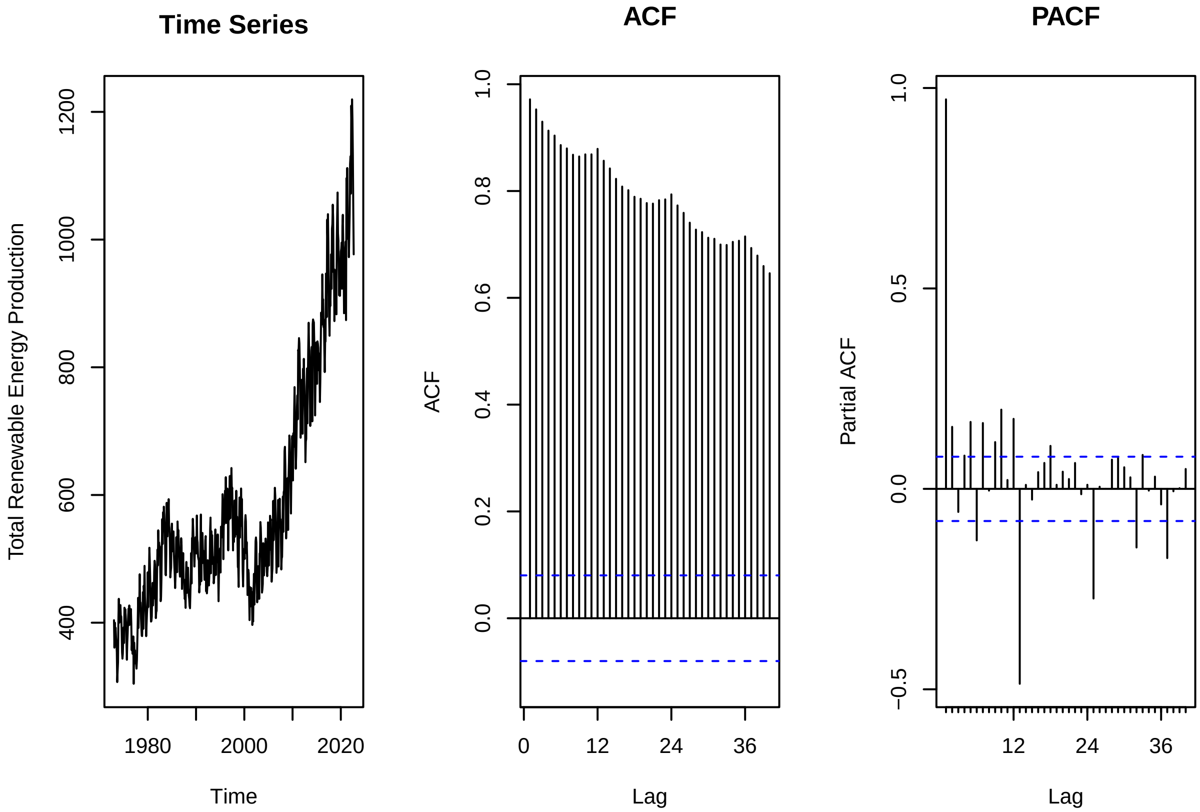
Q1

Create a plot window that has one row and three columns. And then for each object on your data frame, fill the plot window with time series plot, ACF and PACF. You may use the some code form A2, but I want all three plots on the same window this time. (Hint: use par() function)

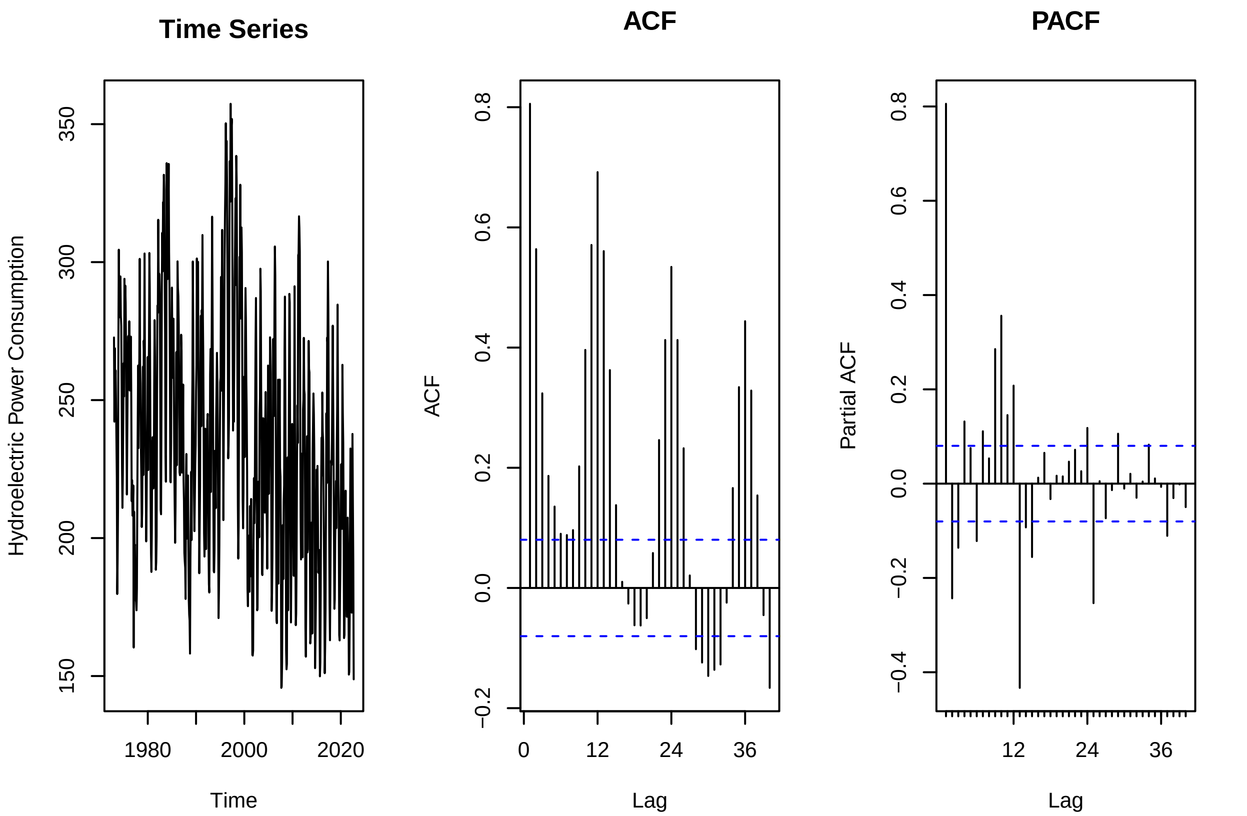
```r
cols <- 4
nobs <- nrow(energy_data)
titles <- colnames(ts_energy)

# Times Series, ACF, PACF Plots
for(i in 2:cols){
  par(mfrow=c(1,3))  #place plot side by side
  plot(ts_energy[,i], main="Time Series", ylab = titles[i])
  Acf(ts_energy[,i],lag.max=40, main="ACF")
  Pacf(ts_energy[,i],lag.max=40, main="PACF")
}
```



Time Series
Total Biomass Energy Production
150
200
250
300
350
400
450
1980
2000
2020
Time
ACF
ACF
0.0
0.2
0.4
0.6
0.8
1.0
0
12
24
36
Lag
PACF
Partial ACF
−0.4
−0.2
0.0
0.2
0.4
0.6
0.8
1.0
12
24
36
Lag



Time Series
Total Renewable Energy Production
400
600
800
1000
1200
1980
2000
2020
Time
ACF
0.0
0.2
0.4
0.6
0.8
1.0
0
12
24
36
Lag
PACF
Partial ACF
−0.5
0.0
0.5
1.0
12
24
36
Lag



Q2

From the plot in Q1, do the series Total Biomass Energy Production, Total Renewable Energy Production, Hydroelectric Power Consumption appear to have a trend? If yes, what kind of trend?

Total Biomass Energy Production appears to have a positive linear trend, with flat period in the middle and at the end of the time series. This could possibly be a polynomial trend. Total Renewable Energy Production appears to have a strong positive trend, which becomes steeper in the early 2000s. Hydroelectric Power Consumption appears to have a slight downward, mostly linear trend.

Q3

Use the *lm()* function to fit a linear trend to the three time series. Ask R to print the summary of the regression. Interpret the regression output, i.e., slope and intercept. Save the regression coefficients for further analysis.

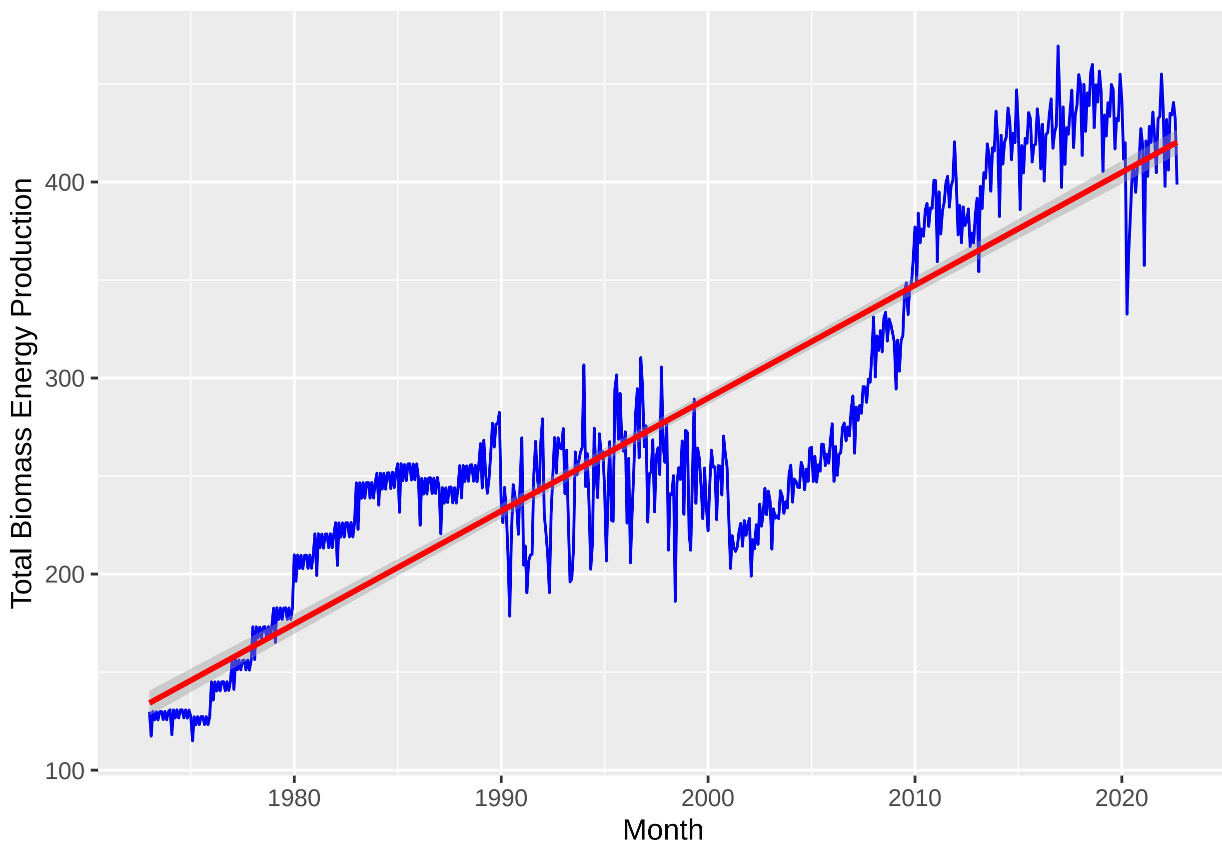
```r
t <- c(1:nobs)

# Total Biomass Energy Production
biomass_trend = lm(energy_data[,2] ~ t)
summary(biomass_trend)
```

```
##
## Call:
## lm(formula = energy_data[, 2] ~ t)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
```

```
## -102.800  -23.994    5.667   32.265   82.192
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 1.337e+02  3.245e+00   41.22   <2e-16 ***
## t           4.800e-01  9.402e-03   51.05   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 39.59 on 595 degrees of freedom
## Multiple R-squared:  0.8142, Adjusted R-squared:  0.8138
## F-statistic:  2607 on 1 and 595 DF,  p-value: < 2.2e-16
```

```r
# Plot series with its trend line
ggplot(energy_data, aes(x=Month, y=energy_data[,2])) +
            geom_line(color="blue") +
            ylab("Total Biomass Energy Production") +
            #geom_abline(intercept = beta0, slope = beta1, color="red")
            geom_smooth(color="red",method="lm")
```

```
## `geom_smooth()` using formula = 'y ~ x'
```



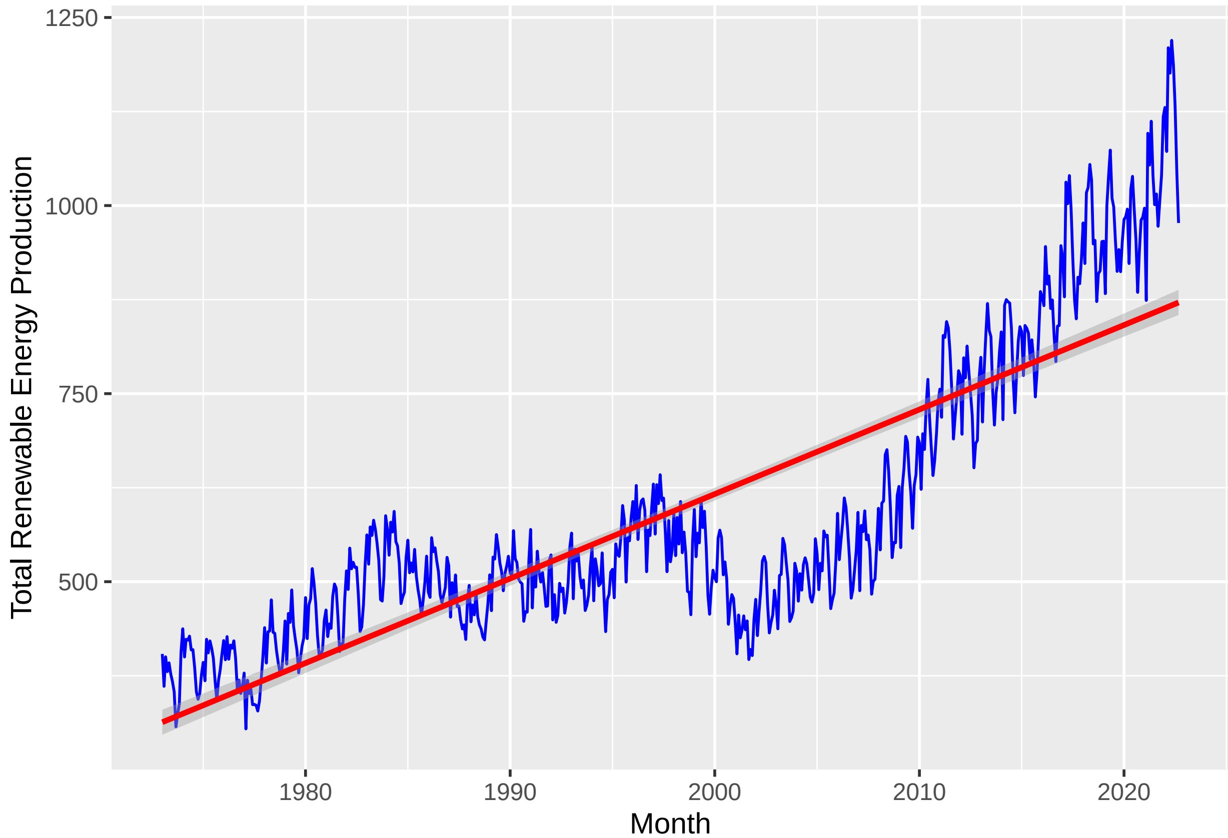
In January 1973, expected total biomass energy production is 133.74 trillion BTU. Each additional month is associated with a 0.48 trillion BTU increase in total biomass energy production on average. In other words, the linear trend that best fits monthly total biomass energy production from 1973 to 2022 begins at 133.74 trillion BTU and increases by 0.48 trillion BTU per month.

```r
# Total Renewable Energy Production
re_trend <- lm(energy_data[,3] ~ t)
summary(re_trend)
```

```
##
## Call:
## lm(formula = energy_data[, 3] ~ t)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -238.75  -61.85    8.59   64.48  352.27
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 312.2475     8.4902   36.78   <2e-16 ***
## t             0.9362     0.0246   38.05   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 103.6 on 595 degrees of freedom
## Multiple R-squared:  0.7088, Adjusted R-squared:  0.7083
## F-statistic:  1448 on 1 and 595 DF,  p-value: < 2.2e-16
```

```r
# Plot series with its trend line
ggplot(energy_data, aes(x=Month, y=energy_data[,3])) +
            geom_line(color="blue") +
            ylab("Total Renewable Energy Production") +
            #geom_abline(intercept = beta0, slope = beta1, color="red")
            geom_smooth(color="red",method="lm")
```

```
## `geom_smooth()` using formula = 'y ~ x'
```



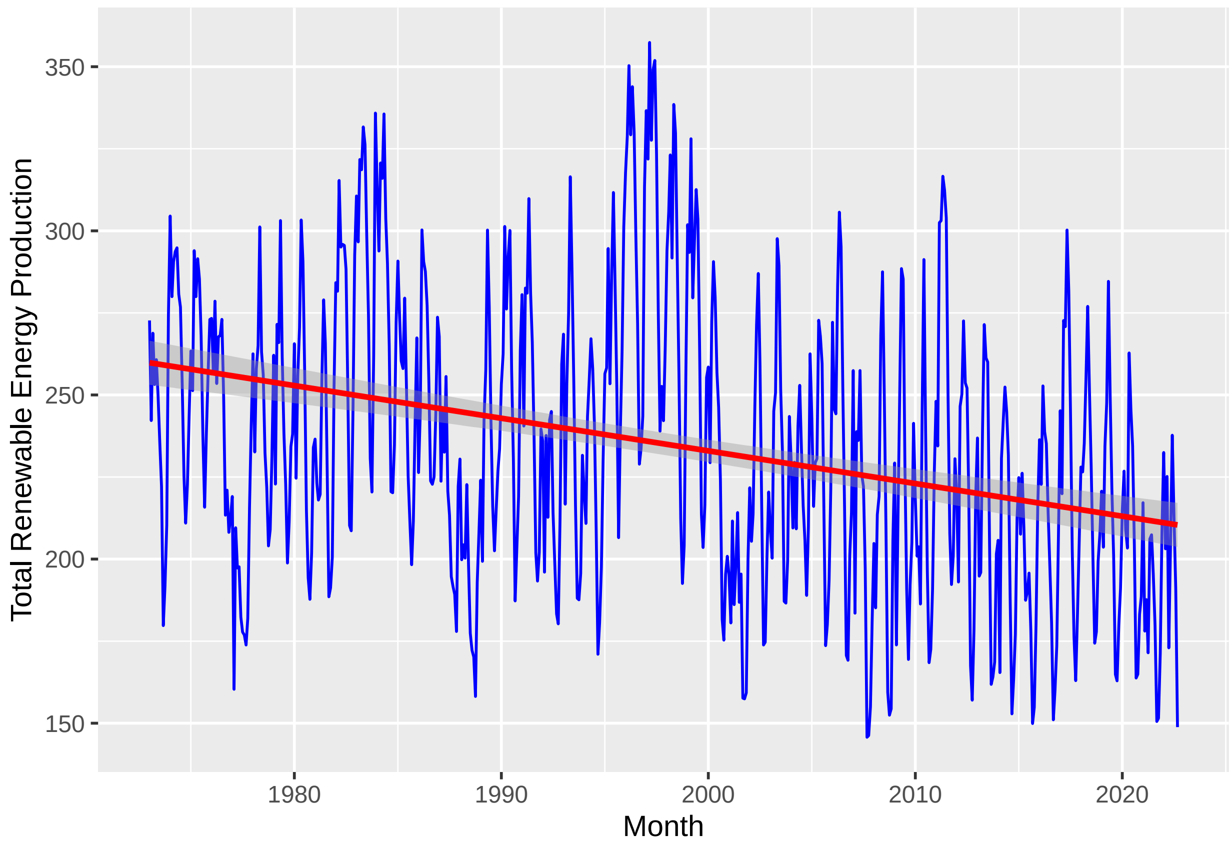
In January 1973, expected total renewable energy production is 312.25 trillion BTU. Each additional month is associated with a 0.9362 trillion BTU increase in total renewable energy production on average. In other words, the linear trend that best fits monthly total renewable energy production from 1973 to 2022 begins at 312.25 trillion BTU and increases by 0.9362 trillion BTU per month.

```r
# Hydroelectric Power Consumption
hydro_trend <- lm(energy_data[,4] ~ t)
summary(hydro_trend)
```

```
##
## Call:
## lm(formula = energy_data[, 4] ~ t)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -95.42 -31.20  -2.56  27.32 121.61
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) 259.898013   3.427300  75.832  < 2e-16 ***
## t            -0.082888   0.009931  -8.346 4.94e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 41.82 on 595 degrees of freedom
## Multiple R-squared:  0.1048, Adjusted R-squared:  0.1033
## F-statistic: 69.66 on 1 and 595 DF,  p-value: 4.937e-16
```

```r
# Plot series with its trend line
ggplot(energy_data, aes(x=Month, y=energy_data[,4])) +
            geom_line(color="blue") +
            ylab("Total Renewable Energy Production") +
            #geom_abline(intercept = beta0, slope = beta1, color="red")
            geom_smooth(color="red",method="lm")
```

```
## `geom_smooth()` using formula = 'y ~ x'
```



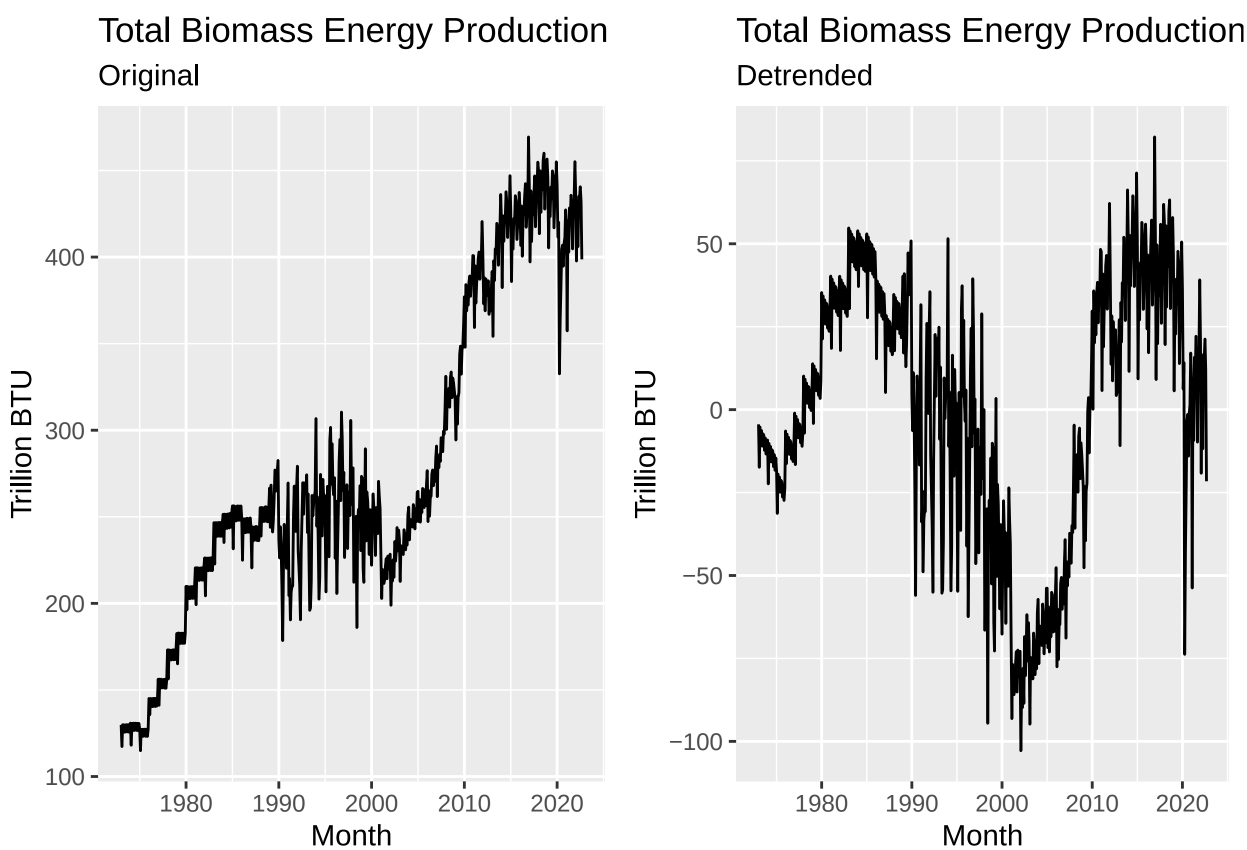
In January 1973, expected hydroelectric power consumption is 259.9 trillion BTU. Each additional month is associated with a 0.083 trillion BTU decrease in hydroelectric power consumption on average. In other words, the linear trend that best fits monthly hydroelectric power consumption from 1973 to 2022 begins at 259.9 trillion BTU and decreases by 0.083 trillion BTU per month.

Q4

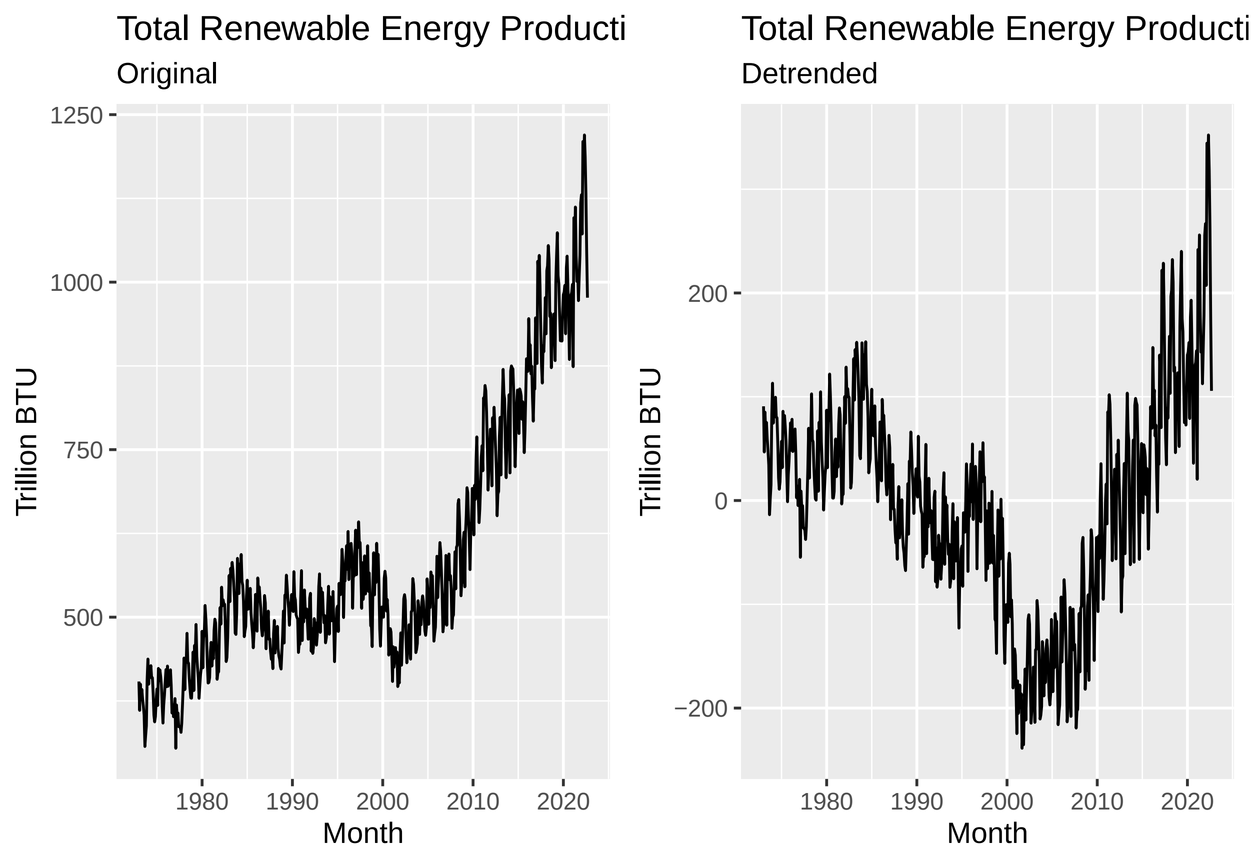
Use the regression coefficients from Q3 to detrend the series. Plot the detrended series and compare with the plots from Q1. What happened? Did anything change?

```r
# Detrend Each Series
detrend_biomass <- energy_data[,2]-(as.numeric(biomass_trend$coefficients[1])+as.numeric(biomass_trend$c
detrend_re <- energy_data[,3]-(as.numeric(re_trend$coefficients[1])+as.numeric(re_trend$coefficients[2]
detrend_hydro <- energy_data[,4]-(as.numeric(hydro_trend$coefficients[1])+as.numeric(hydro_trend$coeffi

# Biomass Detrended Plot
biomass_orig <- ggplot(data=energy_data, aes(x=Month, y=energy_data[,2])) +
  geom_line() +
  ylab("Trillion BTU") +
```

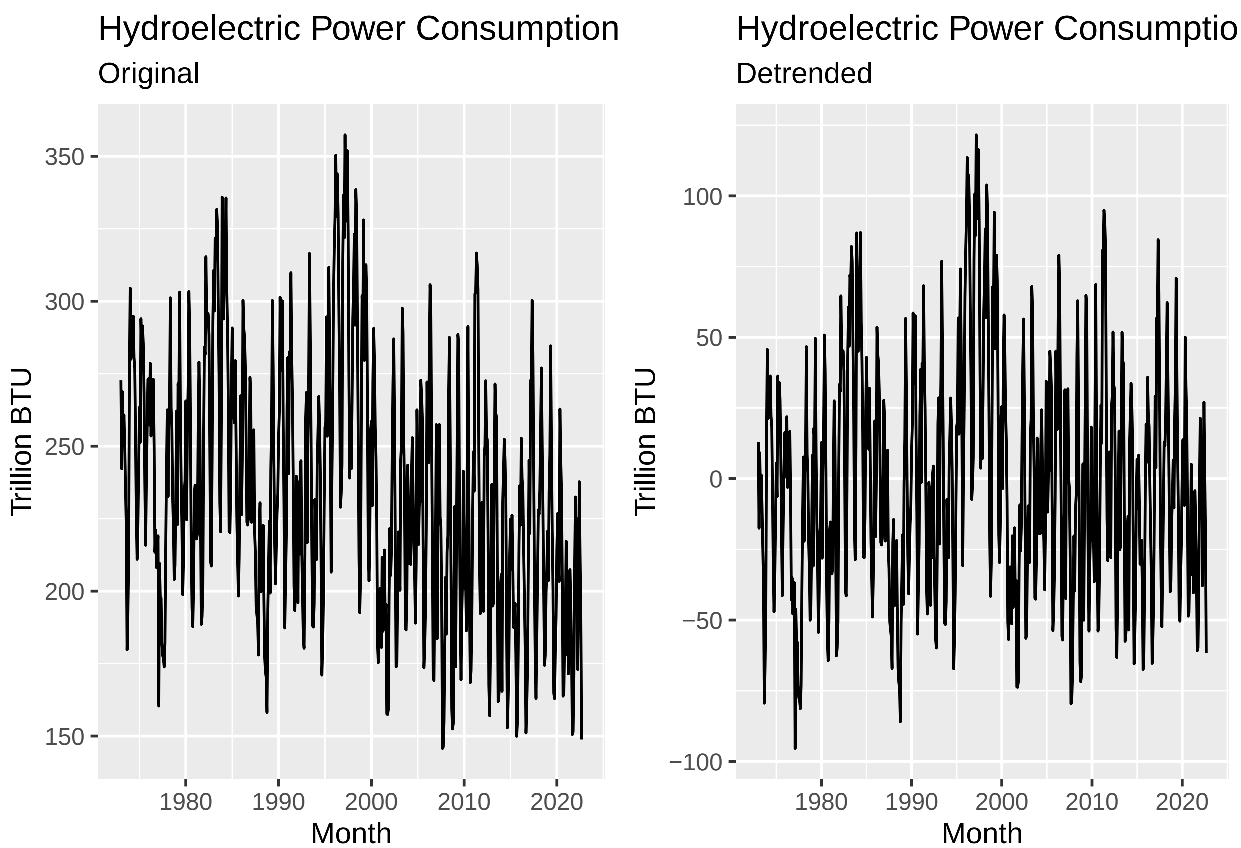
```
  xlab("Month") +
  labs(title="Total Biomass Energy Production", subtitle="Original")

biomass_detrend <- ggplot(data=energy_data, aes(x=Month, y=detrend_biomass)) +
  geom_line() +
  ylab("Trillion BTU") +
  xlab("Month") +
  labs(title="Total Biomass Energy Production", subtitle="Detrended")

plot_grid(biomass_orig, biomass_detrend)
```




```
# RE Detrended Plot
re_orig <- ggplot(data=energy_data, aes(x=Month, y=energy_data[,3])) +
  geom_line() +
  ylab("Trillion BTU") +
  xlab("Month") +
  labs(title="Total Renewable Energy Production", subtitle="Original")

re_detrend <- ggplot(data=energy_data, aes(x=Month, y=detrend_re)) +
  geom_line() +
  ylab("Trillion BTU") +
  xlab("Month") +
  labs(title="Total Renewable Energy Production", subtitle="Detrended")

plot_grid(re_orig, re_detrend)
```




```r
# Hydro Detrended Plot
hydro_orig <- ggplot(data=energy_data, aes(x=Month, y=energy_data[,4])) +
  geom_line() +
  ylab("Trillion BTU") +
  xlab("Month") +
  labs(title="Hydroelectric Power Consumption", subtitle="Original")

hydro_detrend <- ggplot(data=energy_data, aes(x=Month, y=detrend_hydro)) +
  geom_line() +
  ylab("Trillion BTU") +
  xlab("Month") +
  labs(title="Hydroelectric Power Consumption", subtitle="Detrended")

plot_grid(hydro_orig, hydro_detrend)
```

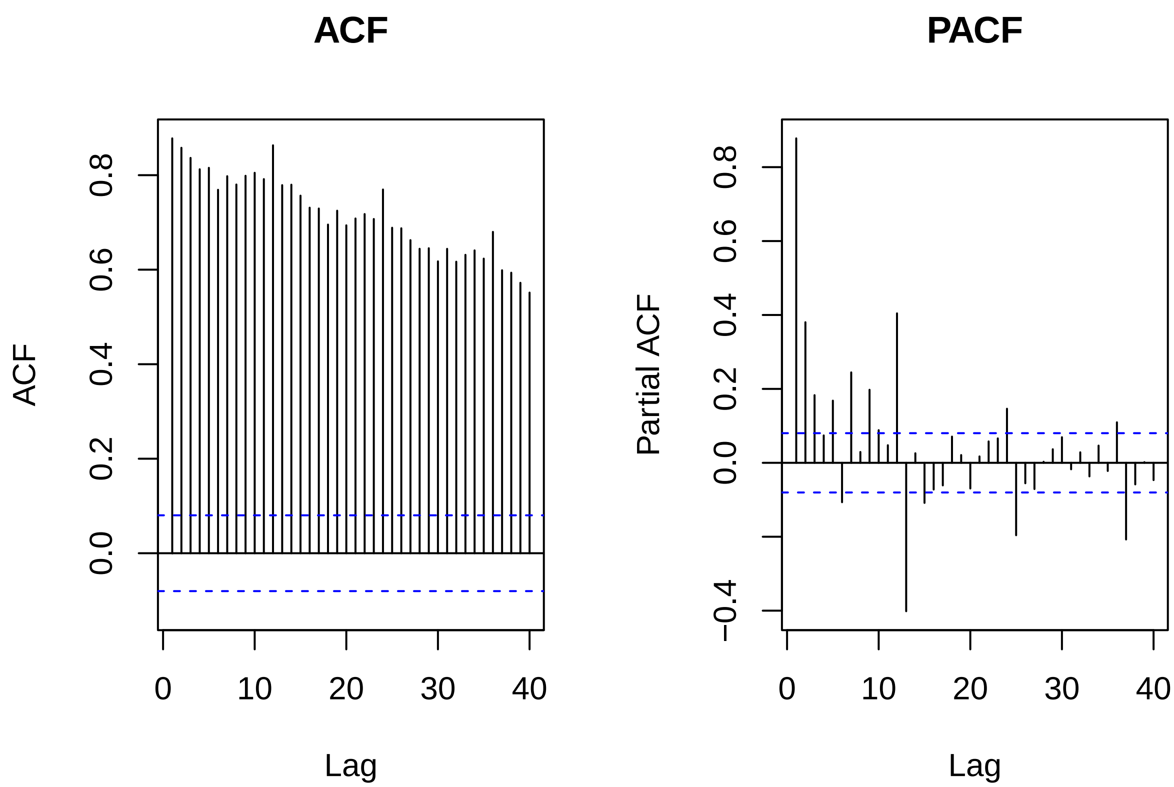


For total biomass energy production and total renewable energy production, the portions of the time series that previously showed a positive trend are now much flatter. For hydroelectric power consumption, the detrended time series plot looks very similar to the original, but a bit flatter now that the slightly negative trend has been removed.

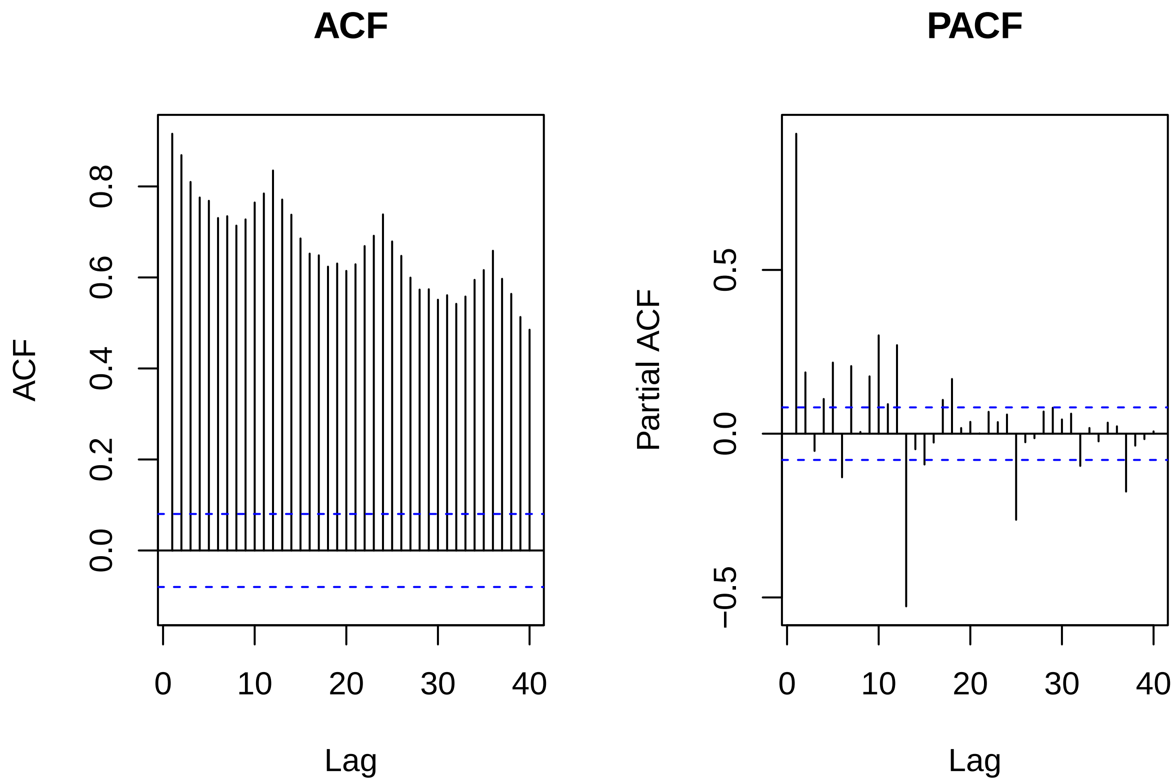
Q5

Plot ACF and PACF for the detrended series and compare with the plots from Q1. Did the plots change? How?

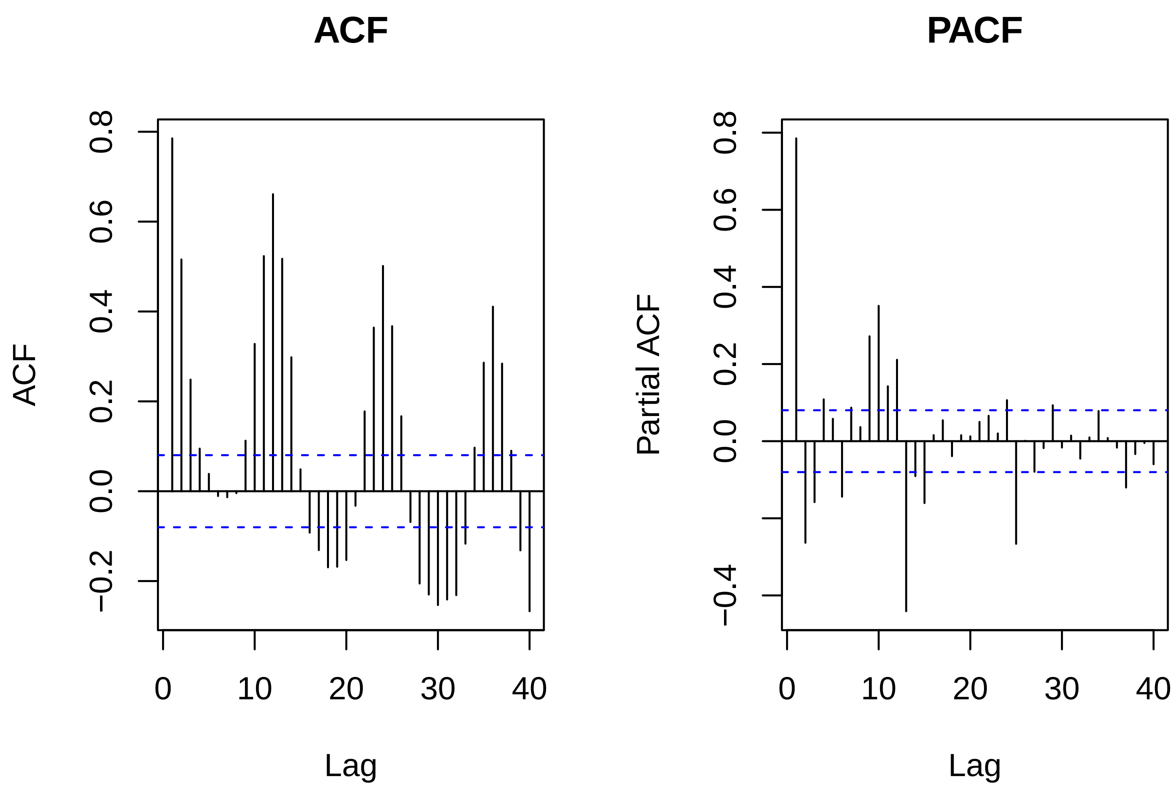
```
# Biomass Detrended ACF, PACF
  par(mfrow=c(1,2))  #place plot side by side
  Acf(detrend_biomass,lag.max=40, main="ACF")
  Pacf(detrend_biomass,lag.max=40, main="PACF")
```




```
# RE Detrended ACF, PACF
  par(mfrow=c(1,2))  #place plot side by side
  Acf(detrend_re,lag.max=40, main="ACF")
  Pacf(detrend_re,lag.max=40, main="PACF")
```




```
# RE Detrended ACF, PACF
  par(mfrow=c(1,2))  #place plot side by side
  Acf(detrend_hydro,lag.max=40, main="ACF")
  Pacf(detrend_hydro,lag.max=40, main="PACF")
```



For Total Biomass Energy Production and Total Renewable Energy Production, the ACF and PACF patterns looks fairly similar, but the magnitude of ACF and PACF values across 40 lags are consistently lower. In each case, the ACF on the detrended series appears to have a stronger cyclical pattern over the lags. For examples, on total RE production, the ACF value appears to spike every 10-15 lags (around 15, 25, 35), as the number of lags increases. Since this is month data, this could be picking up a seasonal pattern.

For Hydroelectric Power Consumption, the magnitude and pattern of ACF and PACF values across 40 lags remain fairly similar in the original and detrended data. However, there is a slight change in the pattern - ACF values that were close to zero in the original data appear to have flipped negative or become more strongly negative in the detrended data.

Seasonal Component

Set aside the detrended series and consider the original series again from Q1 to answer Q6 to Q8.

Q6

Do the series seem to have a seasonal trend? Which serie/series? Use function *lm()* to fit a seasonal means model (i.e. using the seasonal dummies) to this/these time series. Ask R to print the summary of the regression. Interpret the regression output. Save the regression coefficients for further analysis.

```
# Visually, it appears that all three series might have a seasonal component.

# Create Dummies
dummies <- seasonaldummy(ts_energy[,2])
```

```
# Estimate Seasonal Means Model and Save Coefficients
```

```r
# Biomass
biomass_seas <- lm(energy_data[,2]~dummies)
beta_int_biomass = biomass_seas$coefficients[1]
beta_coeff_biomass = biomass_seas$coefficients[2:12]
summary(biomass_seas)
```

```
##
## Call:
## lm(formula = energy_data[, 2] ~ dummies)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -160.74  -53.67  -24.36   90.73  181.34
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  288.020     13.163  21.881   <2e-16 ***
## dummiesJan    -1.793     18.522  -0.097   0.9229
## dummiesFeb   -31.102     18.522  -1.679   0.0936 .
## dummiesMar    -9.104     18.522  -0.492   0.6232
## dummiesApr   -21.502     18.522  -1.161   0.2462
## dummiesMay   -14.238     18.522  -0.769   0.4424
## dummiesJun   -19.602     18.522  -1.058   0.2904
## dummiesJul    -3.674     18.522  -0.198   0.8428
## dummiesAug    -0.612     18.522  -0.033   0.9737
## dummiesSep   -13.335     18.522  -0.720   0.4718
## dummiesOct    -4.030     18.615  -0.216   0.8287
## dummiesNov    -9.849     18.615  -0.529   0.5970
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 92.14 on 585 degrees of freedom
## Multiple R-squared:  0.01018,    Adjusted R-squared:  -0.008437
## F-statistic: 0.5467 on 11 and 585 DF,  p-value: 0.8714
```

February is the only month whose average total biomass energy production is statistically significantly different from December, which is the baseline month/intercept (at the 90% confidence level). Thus, total biomass energy production does not demonstrate a strong seasonal pattern.

```r
# RE
re_seas <- lm(energy_data[,3]~dummies)
beta_int_re = re_seas$coefficients[1]
beta_coeff_re = re_seas$coefficients[2:12]
summary(re_seas)
```

```
##
## Call:
## lm(formula = energy_data[, 3] ~ dummies)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -284.92 -122.23  -68.42   91.22  585.68
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
```

```
## (Intercept)  601.022     27.260  22.048   <2e-16 ***
## dummiesJan    11.468     38.358   0.299    0.765
## dummiesFeb   -41.456     38.358  -1.081    0.280
## dummiesMar    23.130     38.358   0.603    0.547
## dummiesApr     9.959     38.358   0.260    0.795
## dummiesMay    38.853     38.358   1.013    0.312
## dummiesJun    20.378     38.358   0.531    0.595
## dummiesJul     8.298     38.358   0.216    0.829
## dummiesAug   -19.450     38.358  -0.507    0.612
## dummiesSep   -63.770     38.358  -1.662    0.097 .
## dummiesOct   -52.612     38.551  -1.365    0.173
## dummiesNov   -42.537     38.551  -1.103    0.270
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 190.8 on 585 degrees of freedom
## Multiple R-squared:  0.02844,    Adjusted R-squared:  0.01017
## F-statistic: 1.557 on 11 and 585 DF,  p-value: 0.1076
```

Similar to monthly biomass production, only one month has statistically significantly different total renewable energy production compared to the baseline month, December. The coefficient on September is statistically significant at the 90% confidence level. Average total renewable energy consumption in December is expected to be 63.8 trillion BTU lower than total RE in December, which is 601.02 trillion BTU on average. Thus, total renewable energy production also does not demonstrate a strong seasonal pattern.

```
# Hydro
hydro_seas <- lm(energy_data[,4]~dummies)
beta_int_hydro = hydro_seas$coefficients[1]
beta_coeff_hydro = hydro_seas$coefficients[2:12]
summary(hydro_seas)
```

```
##
## Call:
## lm(formula = energy_data[, 4] ~ dummies)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -88.99 -23.47  -2.81  21.99 100.18
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  237.225      4.878  48.634  < 2e-16 ***
## dummiesJan    13.594      6.864   1.981  0.04811 *
## dummiesFeb    -8.254      6.864  -1.203  0.22964
## dummiesMar    19.980      6.864   2.911  0.00374 **
## dummiesApr    15.649      6.864   2.280  0.02297 *
## dummiesMay    39.210      6.864   5.713 1.77e-08 ***
## dummiesJun    31.209      6.864   4.547 6.61e-06 ***
## dummiesJul    10.436      6.864   1.520  0.12895
## dummiesAug   -17.909      6.864  -2.609  0.00931 **
## dummiesSep   -50.173      6.864  -7.310 8.82e-13 ***
## dummiesOct   -48.262      6.898  -6.996 7.22e-12 ***
## dummiesNov   -32.285      6.898  -4.680 3.56e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

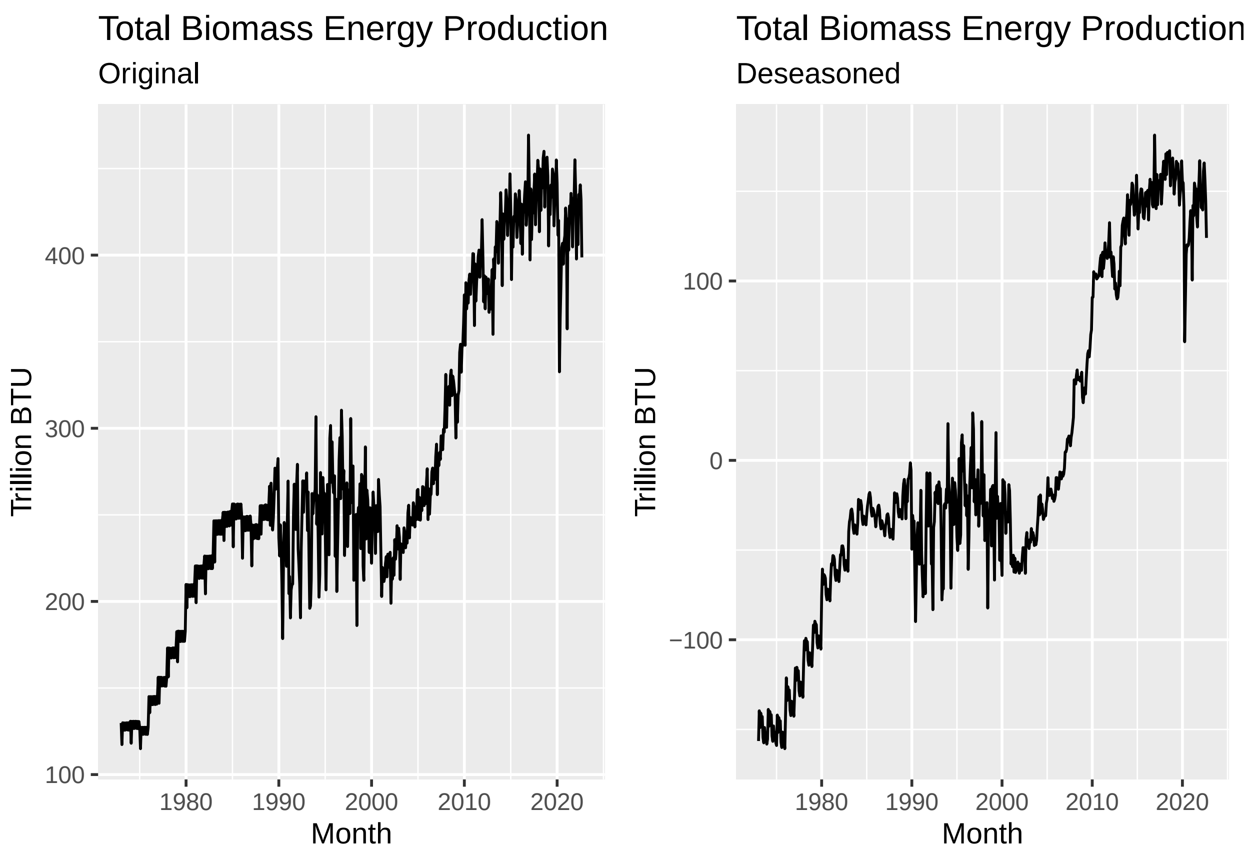
```
##
## Residual standard error: 34.14 on 585 degrees of freedom
## Multiple R-squared:  0.4132, Adjusted R-squared:  0.4022
## F-statistic: 37.45 on 11 and 585 DF,  p-value: < 2.2e-16
```

Unlike biomass and renewable energy production, for total hydroelectric power consumption, nine months have statistically significant coefficients at least at the 90% confidence level, which suggests that hydro consumption exhibits seasonality. Average total hydroelectric power consumption is expect to be 237.2 trillion BTU in December. On average, hydroelectric power consumption is higher in March through June and lower in August through November compared to December.

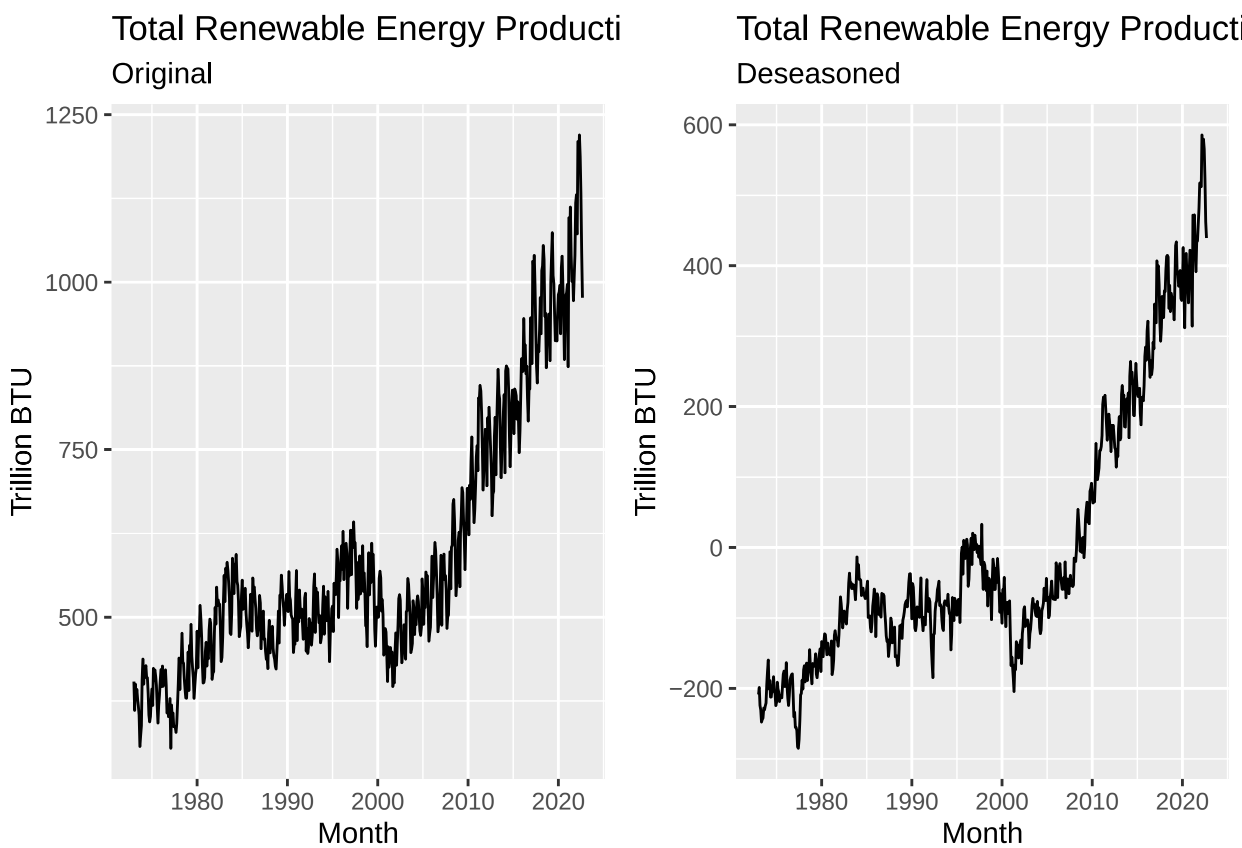
Q7

Use the regression coefficients from Q6 to deseason the series. Plot the deseason series and compare with the plots from part Q1. Did anything change?

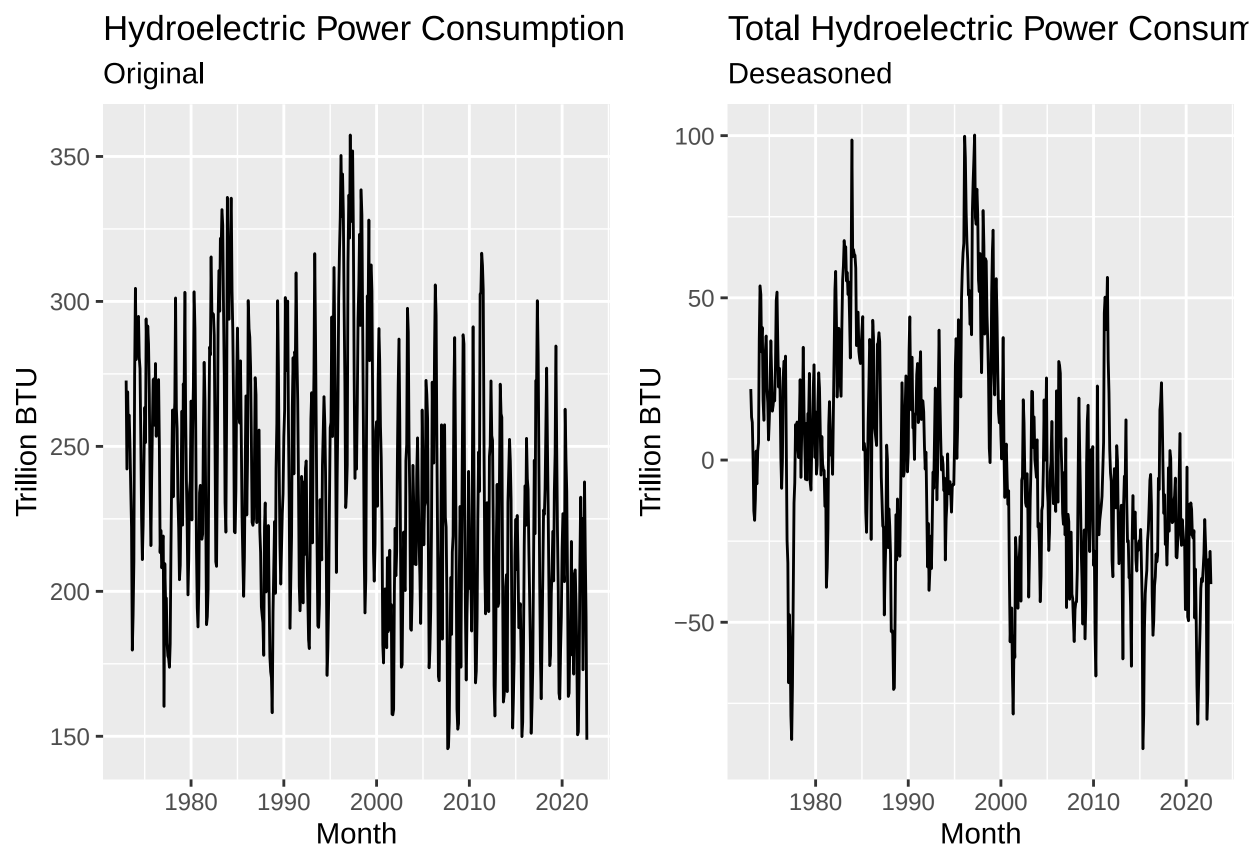
```r
# Biomass
biomass_seas_est=array(0,nobs)
for(i in 1:nobs){
  biomass_seas_est[i]=(beta_int_biomass+beta_coeff_biomass%*%dummies[i,])
}
deseason_biomass <- energy_data[,2]-biomass_seas_est

biomass_deseas <- ggplot(data=energy_data, aes(x=Month, y=deseason_biomass)) +
  geom_line() +
  ylab("Trillion BTU") +
  xlab("Month") +
  labs(title="Total Biomass Energy Production", subtitle="Deseasoned")

plot_grid(biomass_orig, biomass_deseas)
```




```
# RE
re_seas_est=array(0,nobs)
for(i in 1:nobs){
  re_seas_est[i]=(beta_int_re+beta_coeff_re%*%dummies[i,])
}
deseason_re <- energy_data[,3]-re_seas_est

re_deseas <- ggplot(data=energy_data, aes(x=Month, y=deseason_re)) +
  geom_line() +
  ylab("Trillion BTU") +
  xlab("Month") +
  labs(title="Total Renewable Energy Production", subtitle="Deseasoned")

plot_grid(re_orig, re_deseas)
```




```
# Hydro
hydro_seas_est=array(0,nobs)
for(i in 1:nobs){
  hydro_seas_est[i]=(beta_int_hydro+beta_coeff_hydro%*%dummies[i,])
}
deseason_hydro <- energy_data[,4]-hydro_seas_est

hydro_deseas <- ggplot(data=energy_data, aes(x=Month, y=deseason_hydro)) +
  geom_line() +
  ylab("Trillion BTU") +
  xlab("Month") +
  labs(title="Total Hydroelectric Power Consumption", subtitle="Deseasoned")

plot_grid(hydro_orig, hydro_deseas)
```



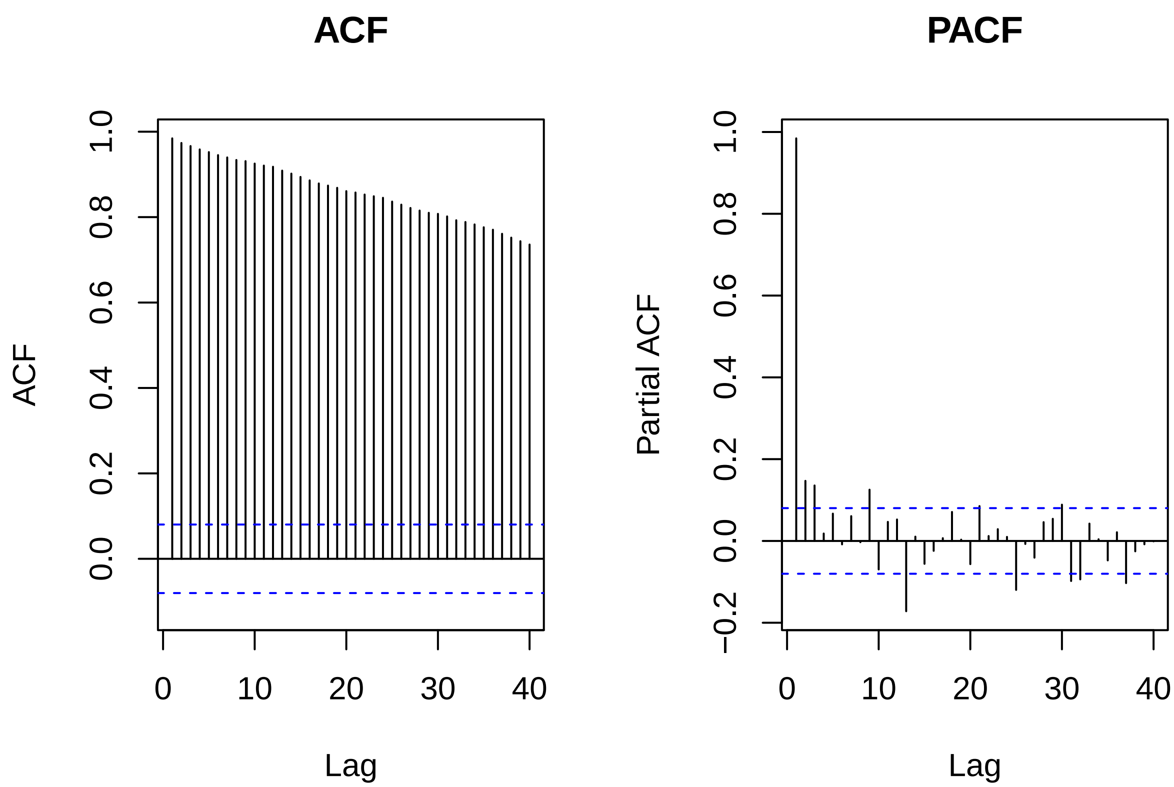
For Total Biomass Energy Production and Total Renewable Energy Production, the time series plots still look very similar to the original ones, with some variation slightly smoothed out. For both of these variables, the seasonal models did not reveal significant seasonality, since most month coefficients were not statistically significant. Therefore, it might not be ideal to use those coefficients to deseason the data.

On the other hand, the seasonal model revealed seasonality in the hydroelectric power consumption variable. The deseased time series plot for this variable looks much smoother compared to the original times series plot.

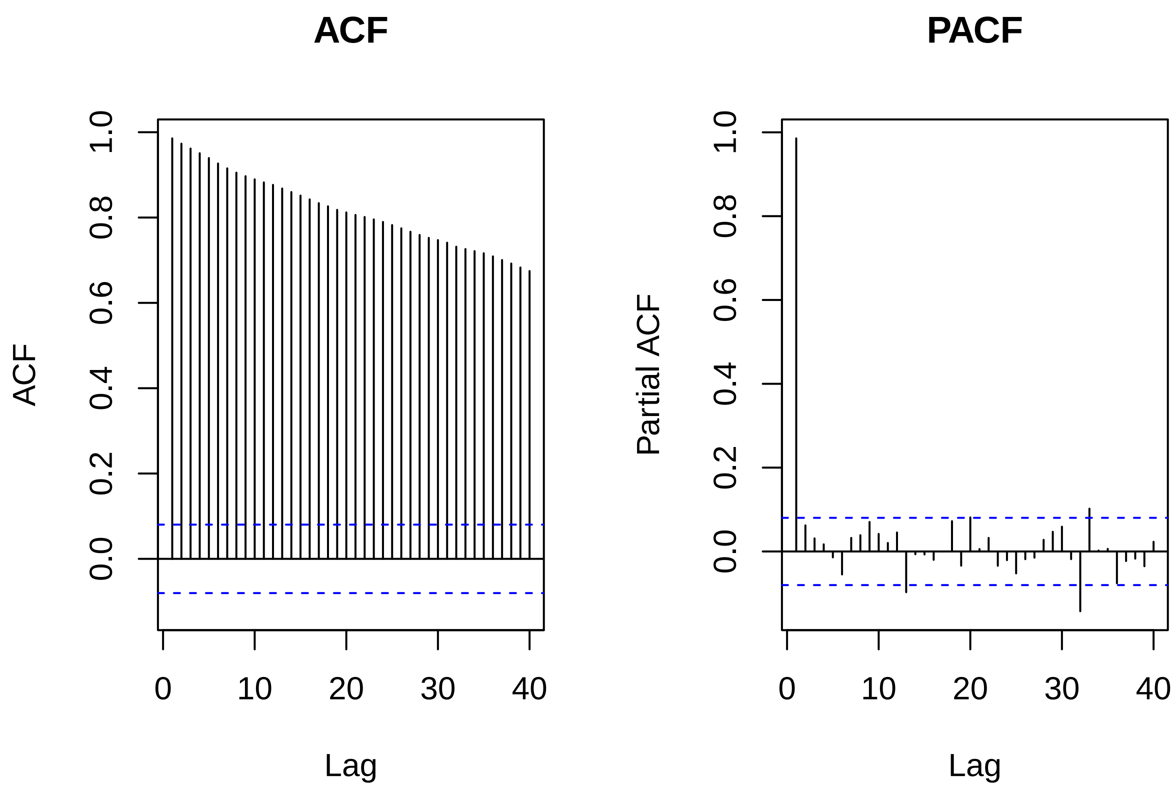
Q8

Plot ACF and PACF for the deseason series and compare with the plots from Q1. Did the plots change? How?

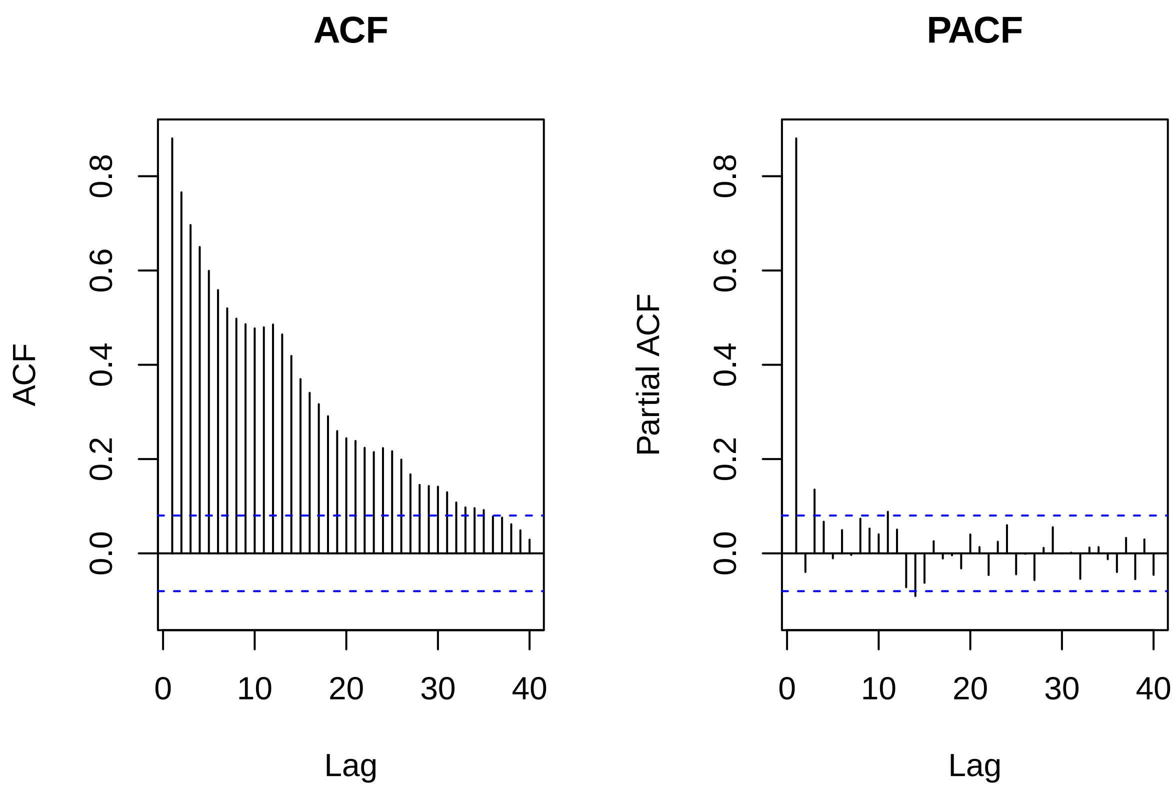
```
# Biomass Deseasoned ACF, PACF
  par(mfrow=c(1,2))  #place plot side by side
  Acf(deseason_biomass,lag.max=40, main="ACF")
  Pacf(deseason_biomass,lag.max=40, main="PACF")
```




```
# RE Deseasoned ACF, PACF
  par(mfrow=c(1,2))  #place plot side by side
  Acf(deseason_re,lag.max=40, main="ACF")
  Pacf(deseason_re,lag.max=40, main="PACF")
```




```
# Hydro Deseasoned ACF, PACF
  par(mfrow=c(1,2))  #place plot side by side
  Acf(deseason_hydro,lag.max=40, main="ACF")
  Pacf(deseason_hydro,lag.max=40, main="PACF")
```



For total biomass energy production and total renewable energy production, the magnitude and overall pattern of ACF values across 40 lags is very similar, but in the deaseasoned version, the periodic spikes in ACF have now been smoothed out and the PACF values after the first lag appear to have reduced in magnitude.

For total hydroelectric power consumption, the pattern of ACF values over 40 lags has change a lot. In the original series, the ACF value fluctuated from positive to negative over the number of lags, while in the deseasoned version, the values are all positive and decreasing with the number of lags, which looks more similar to the biomass and renewable energy series. For the PACF, the change is similar to that of the other series, where the PACF values after the first lag are smaller in magnitude in the deseasoned version.