ENV 790.30 - Time Series Analysis for Energy Data | Spring 2021

Assignment 3 - Due date 02/12/21

Rajat Khandelwal

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 project open the first thing you will do is change "Student Name" on line 3 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. Rename the pdf file such that it includes your first and last name (e.g., "LuanaLima_TSA_A01_Sp21.Rmd"). 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 January 2021 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(forecast)
library(tseries)
library(dplyr)
library(utils)
library(Kendall)
library(ggplot2)
library(readxl)
library(cowplot)

#Importing data set
#NOTE: Locally changed name of XLSX file to "REP_Data.xlsx" for easier reference.
data <- read_excel("~/Documents/GitHub/ENV790_30_TSA_S2021/Data/REP_Data.xlsx", skip = 10)
data <- data %>% select(1,4:6) %>% slice(2:n())
```

```r
#Converting dataframe columns into numeric values
data$`Total Biomass Energy Production` <- as.numeric(data$`Total Biomass Energy Production`)
data$`Total Renewable Energy Production` <- as.numeric(data$`Total Renewable Energy Production`)
data$`Hydroelectric Power Consumption` <- as.numeric(data$`Hydroelectric Power Consumption`)

#Converting dataframe into time-series
data_ts <- ts(data[2:4], start = c(1973,1), end = c(2020,10), frequency = 12)

#Convert time column to date format
data$Month <- as.Date(data$Month , format = "%m/%d/%y")

#Setting parameters for future use
ncols <- ncol(data)-1
nobs <- nrow(data)
```

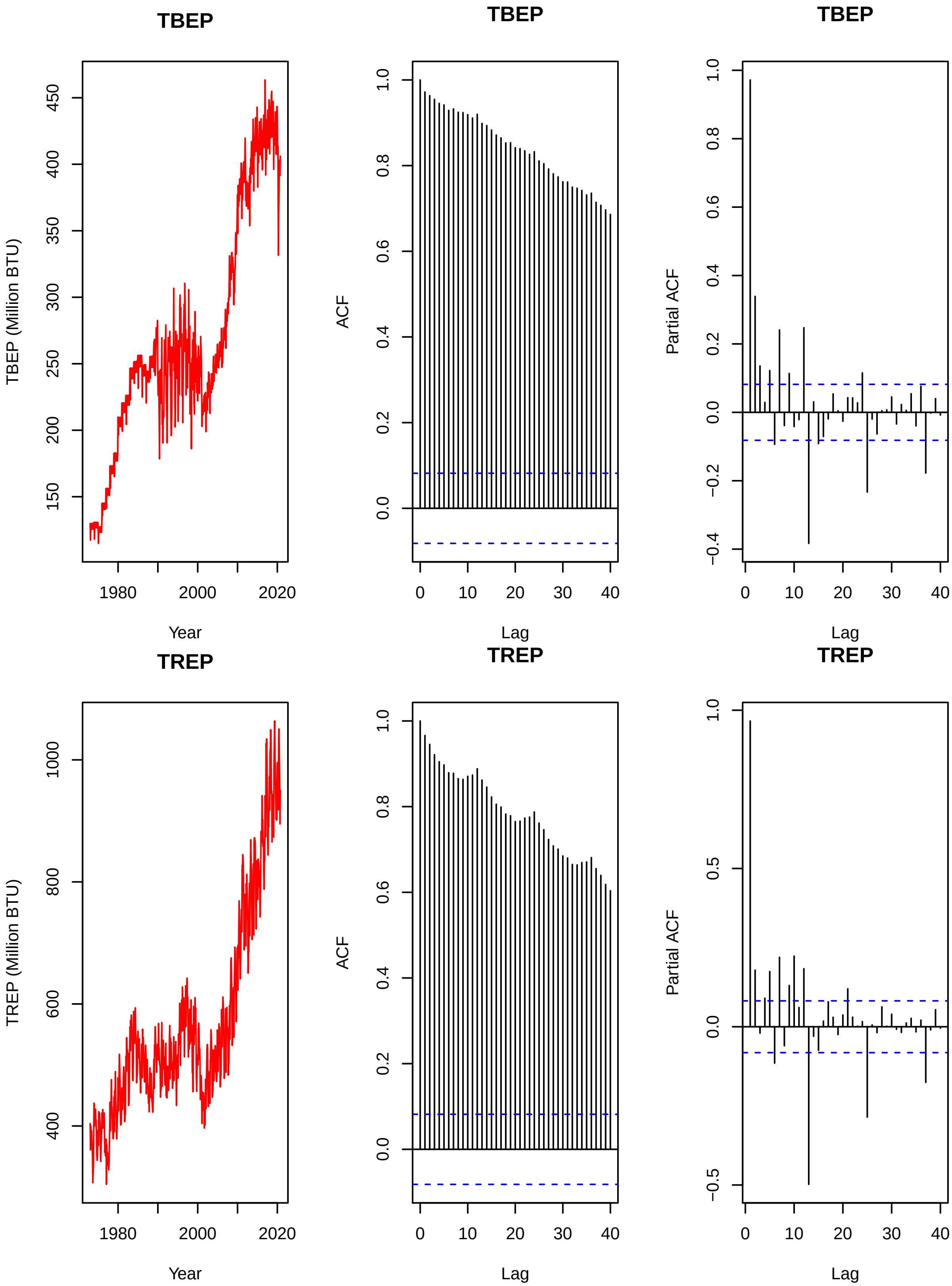
Trend Component

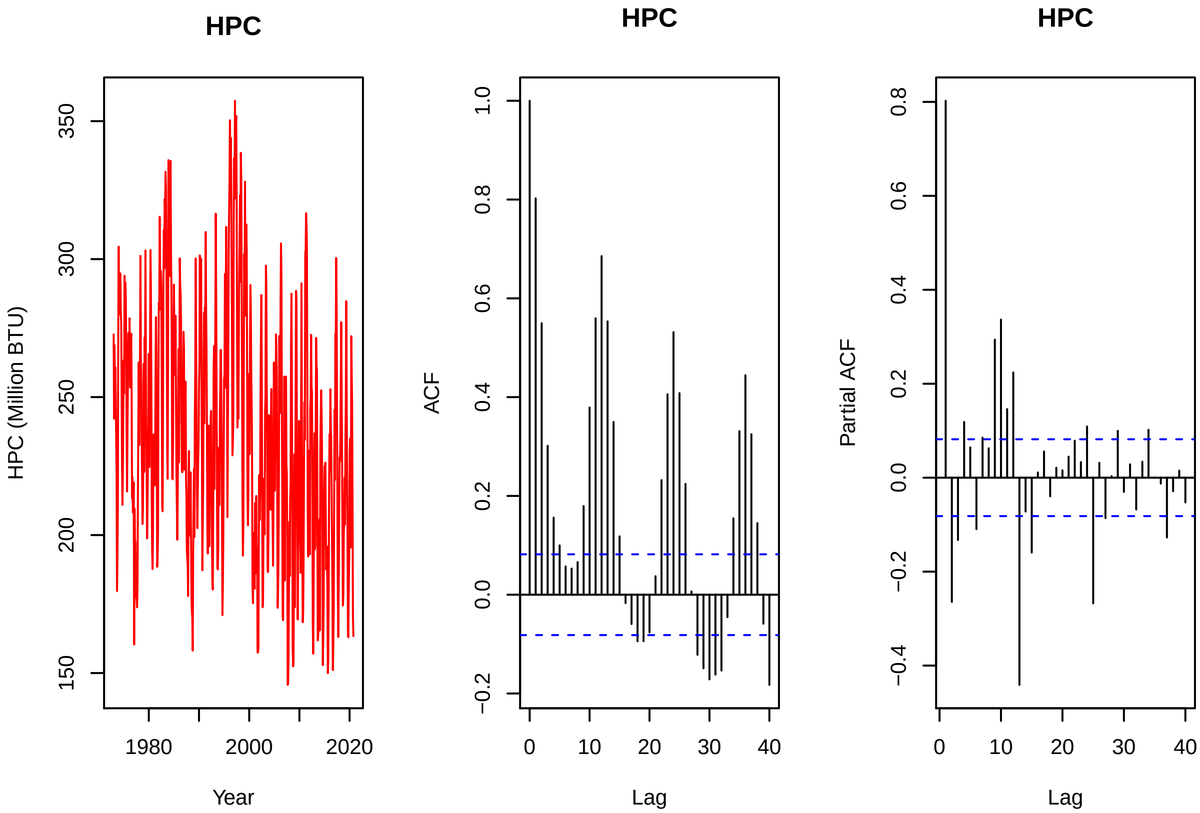
Q1

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: watch videos for M4)

```r
shortnames <- c("TBEP", "TREP", "HPC")
for (i in 1:ncols){
  par(mfrow = c(1, ncols))
  #print(colnames(data[i+1]))
  #  print(ggplot(data, aes(x = Month, y = data_ts[,i])) +
   # geom_line(color = "red")  +
  #  ylab(colnames(data[i+1])) +
  #  xlab('Year'))
  plot(data_ts[,i], col = "red", main = shortnames[i], xlab = "Year", ylab =
         paste0(shortnames[i], sep = " ", "(Million BTU)"), type = "l")

  acf(data[,i+1], lag.max = 40, main = shortnames[i])
  pacf(data[,i+1], lag.max = 40, main = shortnames[i])
  #plot_grid(time_series, acf, pacf)
}
```


TBEP
TBEP
TBEP
TBEP (Million BTU)
450
400
350
300
250
200
150
1980
2000
2020
Year
ACF
1.0
0.8
0.6
0.4
0.2
0.0
0
10
20
30
40
Lag
Partial ACF
1.0
0.8
0.6
0.4
0.2
0.0
−0.2
−0.4
0
10
20
30
40
Lag
TREP
TREP
TREP
TREP (Million BTU)
1000
800
600
400
1980
2000
2020
Year
ACF
1.0
0.8
0.6
0.4
0.2
0.0
0
10
20
30
40
Lag
Partial ACF
1.0
0.5
0.0
−0.5
0
10
20
30
40
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?

TBEP: Has an increasing trend as the mean appears to increase over time.
TREP: Increasing trend as the mean appears to increase over time.
HPC: Doesn't show any signifcant trend visually. Shows seasonality.

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.

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

lm_list <- vector(mode = "list", length = ncols)
beta0 <- vector(mode = "list", length = ncols)
beta1 <- vector(mode = "list", length = ncols)

for(i in 1:ncols){
  lm_list[[i]] = lm(data_ts[,i]~t)
  print(colnames(data[i+1]))
  print(summary(lm_list[[i]]))
  beta0[[i]] = as.numeric(lm_list[[i]]$coefficients[1])
  beta1[[i]] = as.numeric(lm_list[[i]]$coefficients[2])
}
```

```
## [1] "Total Biomass Energy Production"
##
## Call:
## lm(formula = data_ts[, i] ~ t)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -101.149  -25.456    4.985   33.353   79.634
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 1.355e+02  3.296e+00   41.11   <2e-16 ***
## t          4.702e-01  9.934e-03   47.33   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 39.44 on 572 degrees of freedom
## Multiple R-squared:  0.7966, Adjusted R-squared:  0.7962
## F-statistic:  2240 on 1 and 572 DF,  p-value: < 2.2e-16
##
## [1] "Total Renewable Energy Production"
##
## Call:
## lm(formula = data_ts[, i] ~ t)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -224.735  -55.673    5.418   60.453  263.849
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 330.37156    7.86270   42.02   <2e-16 ***
## t             0.84299    0.02369   35.58   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 94.07 on 572 degrees of freedom
## Multiple R-squared:  0.6887, Adjusted R-squared:  0.6882
## F-statistic:  1266 on 1 and 572 DF,  p-value: < 2.2e-16
##
## [1] "Hydroelectric Power Consumption"
##
## Call:
## lm(formula = data_ts[, i] ~ t)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -94.06 -31.57  -1.63  27.73 120.69
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 258.05622    3.52899  73.125  < 2e-16 ***
## t            -0.07341    0.01063  -6.903 1.36e-11 ***
## ---
```

```
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 42.22 on 572 degrees of freedom
## Multiple R-squared:  0.07689,    Adjusted R-squared:  0.07528
## F-statistic: 47.64 on 1 and 572 DF,  p-value: 1.361e-11
```

TBEP: Slope = 0.47 indicates that TBEP has an increasing trend over time.
TREP: Slope = 0.84 indicates that TREP has a very strong increasing trend over time.
HPC: Slope = -0.07 indicates that HPC has no clear trend over time.

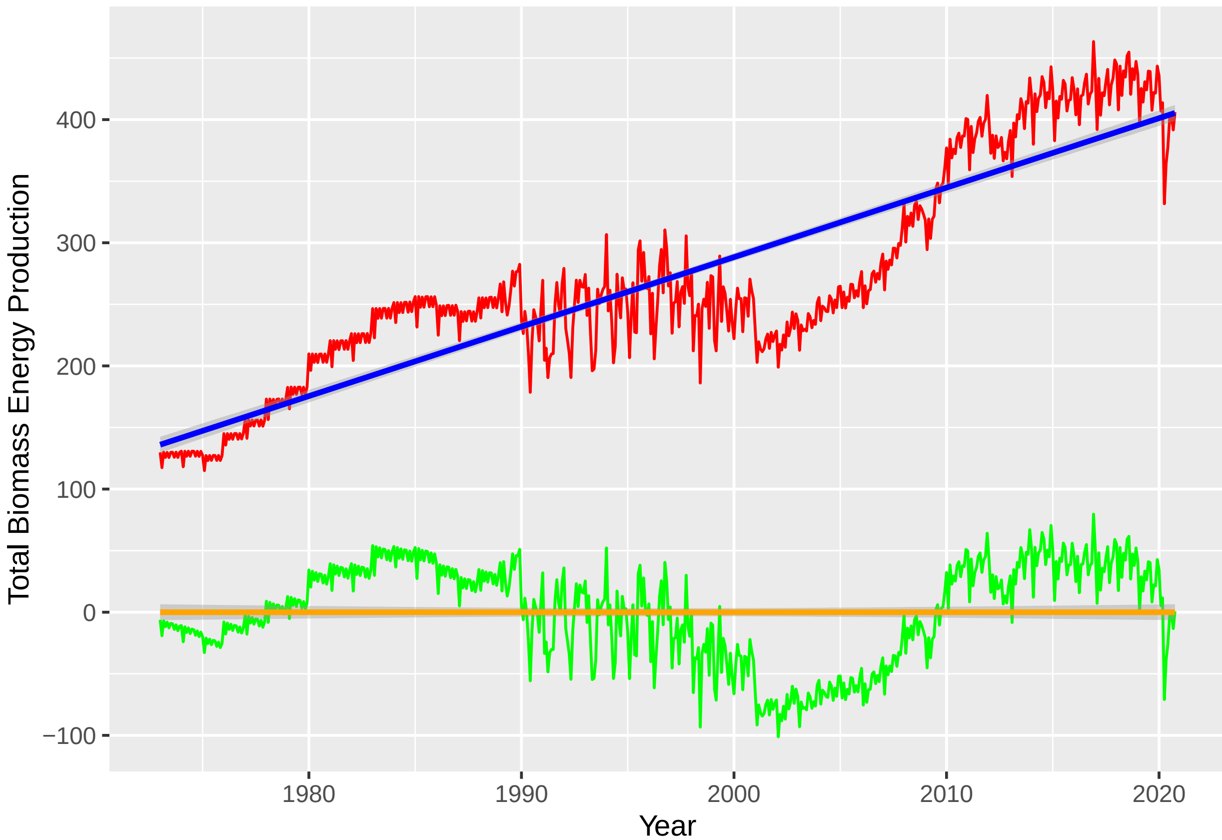
The intercept for all three series shows the value of the repective linear models at the beginning of the time-series.

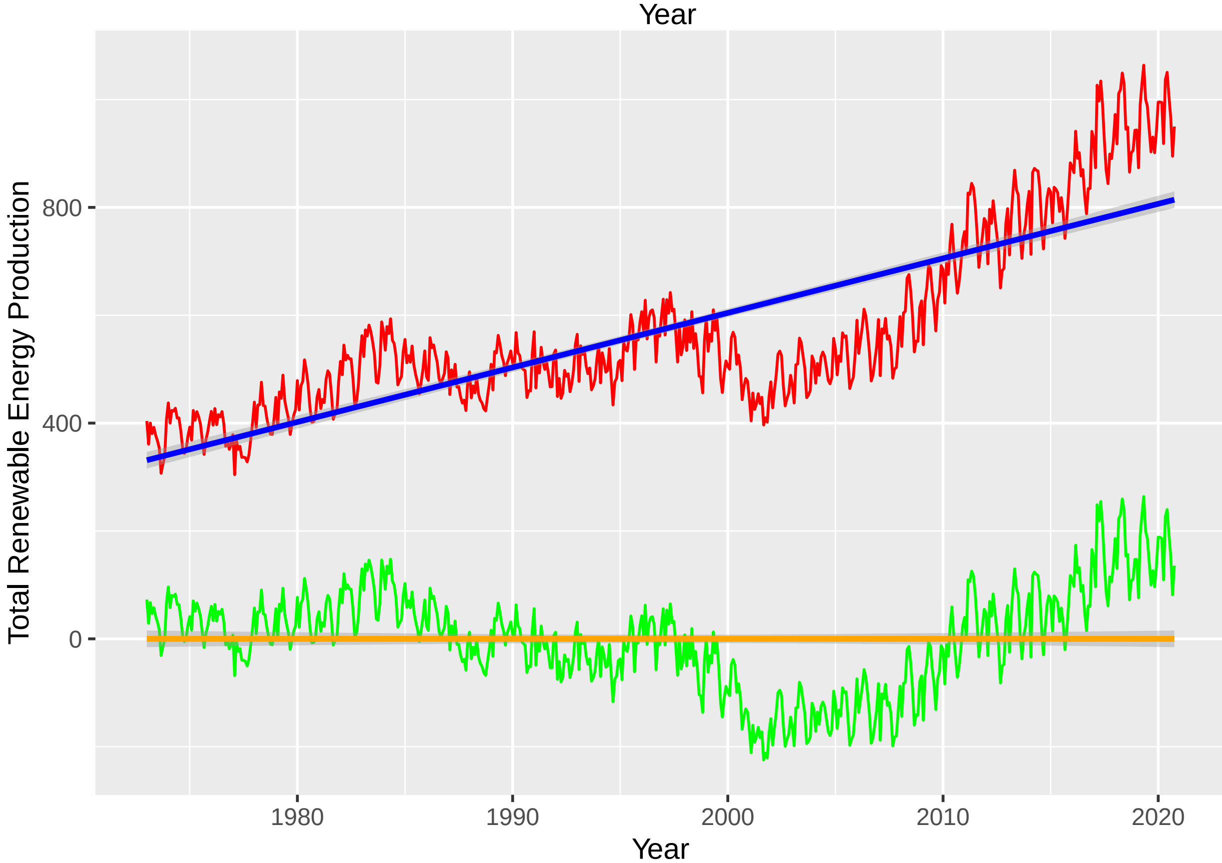
Q4

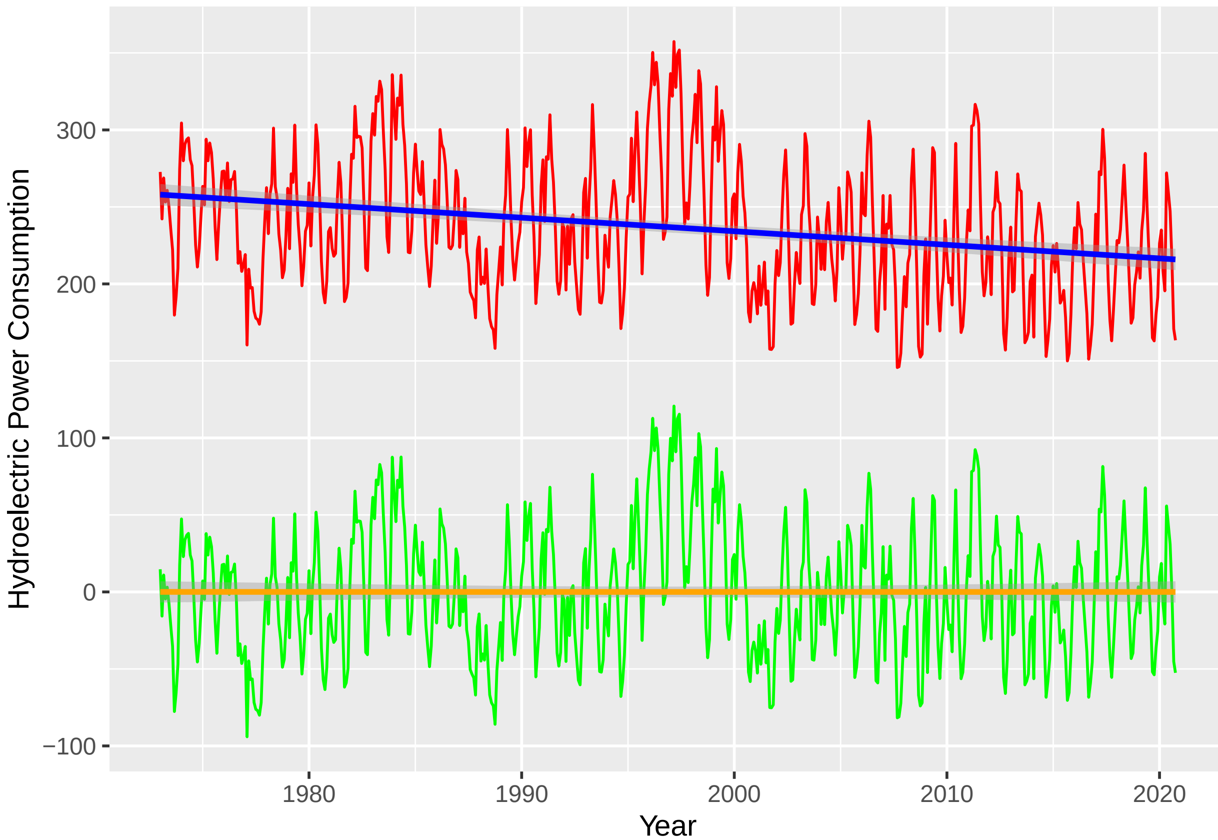
Use the regression coefficients from Q3 to detrend the series. Plot the detrended series and compare with the plots from Q1. What happened? Did anything change?

Original series in red. Detrended series in green.

```
detrended_data <- data.frame(matrix(nrow = nobs,ncol = ncols))
data <- as.data.frame(data)
for (i in 1:ncols){
    detrended_data[,i] <- data[,i+1] - (beta0[[i]] + beta1[[i]]*t)
    print(
       ggplot(detrended_data, aes(x = data$Month, y = detrended_data[,i]))+
            ylab(colnames(data[i+1]))+
            xlab("Year")+
            geom_line(color = "green") +
            geom_smooth(color = "orange", method = "lm")+
            geom_line(aes(y = data[,i+1]), color = "red") +
            geom_smooth(aes(y = data[,i+1]), color = "blue", method = "lm")
    )
}
```


Total Biomass Energy Production
400
300
200
100
0
−100
1980
1990
2000
2010
2020
Year


Total Renewable Energy Production
800
400
0
1980
1990
2000
2010
2020
Year



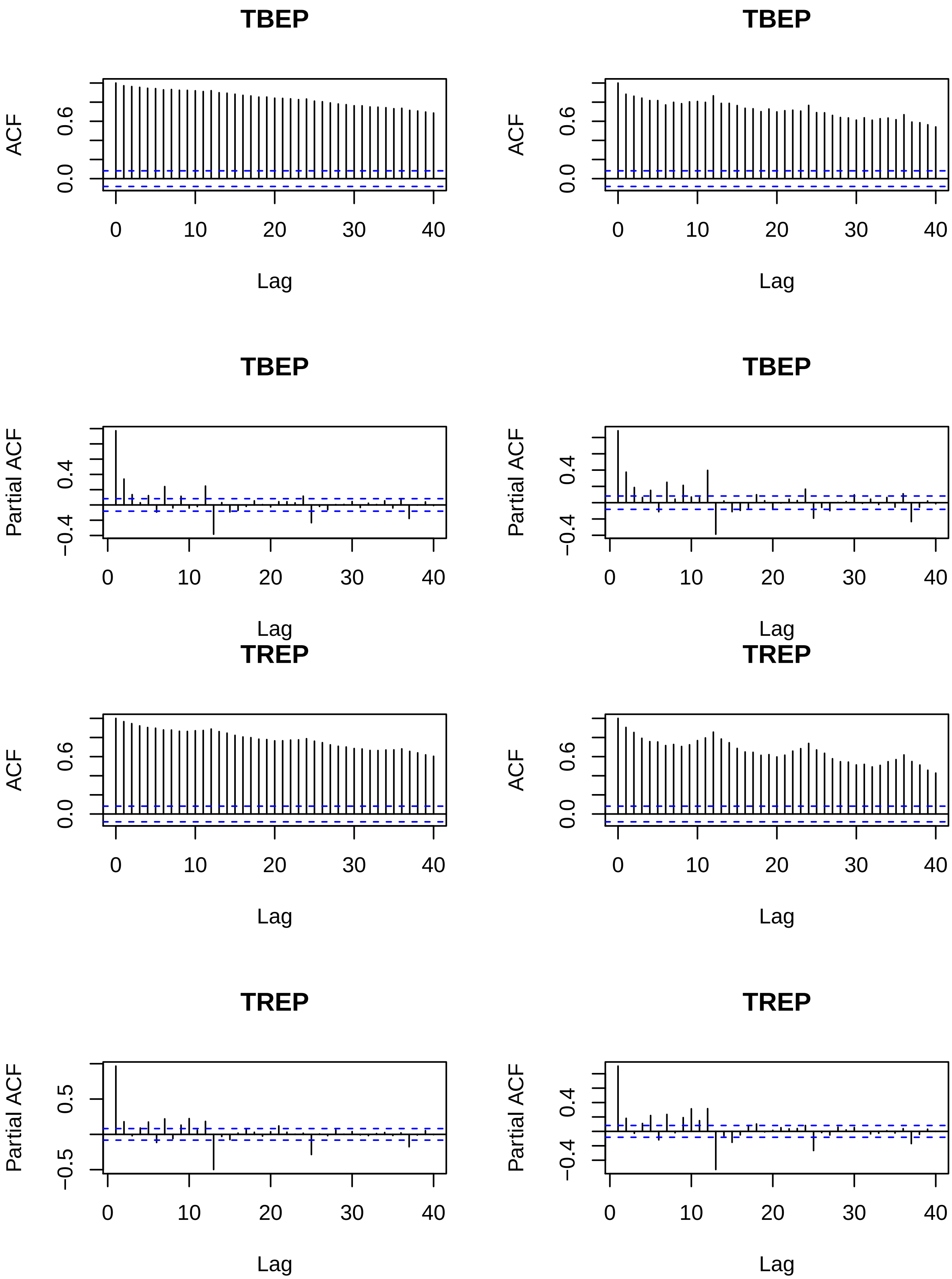
Yes, the linear models for all three series are effectively a flat y = 0 line. After removing the trend by subtracting the linear model from each series, we effectively get a down-shift for each series as we obtain a detrended series. This detrended series has no trend but only seasonality and random variation.

Q5

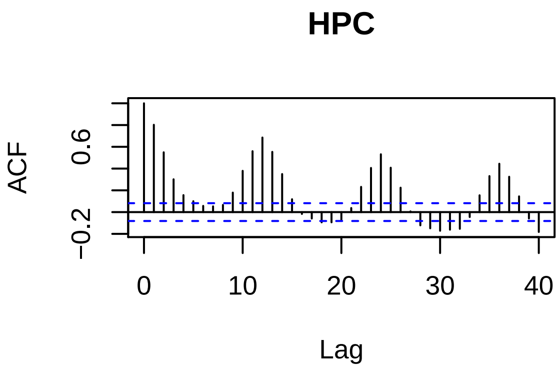
Plot ACF and PACF for the detrended series and compare with the plots from Q1. Did the plots change? How?

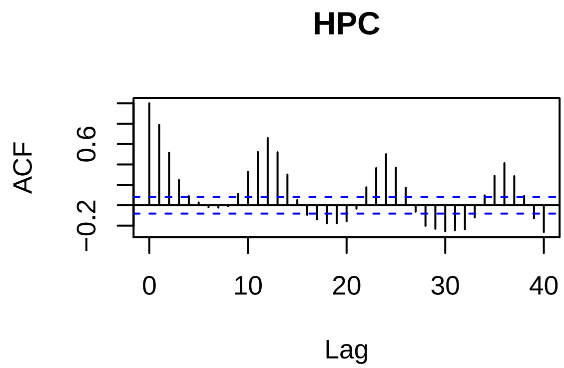
Plots on the left are the original series. Plots on the right are the detrended series.

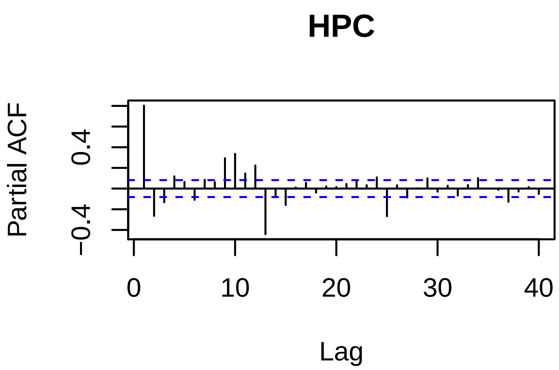
```
for (i in 1:ncols){
  par(mfrow = c(2,2))
  acf(data[,i+1], lag = 40, main = shortnames[i])
  acf(detrended_data[,i], lag = 40, main = shortnames[i])
  pacf(data[,i+1], lag = 40, main = shortnames[i])
  pacf(detrended_data[,i], lag = 40, main = shortnames[i])
}
```

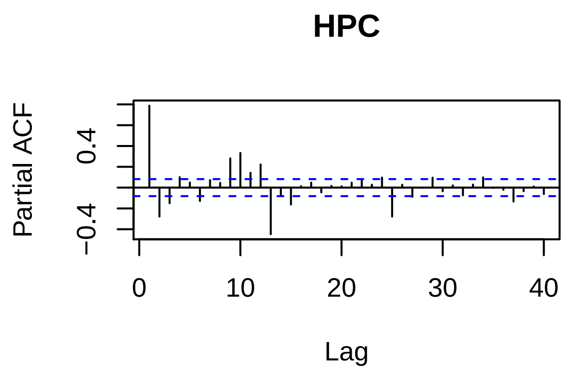


TBEP
ACF
Lag
TBEP
ACF
Lag
TBEP
Partial ACF
Lag
TBEP
Partial ACF
Lag
TREP
ACF
Lag
TREP
ACF
Lag
TREP
Partial ACF
Lag
TREP
Partial ACF
Lag









The ACF and PACF plots for the detrended lines, when compared to that of the original series, clearly show that trend has been removed from the detrended series. After the removal of the trend, we can observe that the seasonality is more prominent in the ACF and PACF on the right. We observe more pronounced peaks at lag $= 13, 25, 37$ as would be expected in a series with frequency $= 12$.

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 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.

```
dummies <- vector(mode = "list", length = ncols)
seasonal_model <- vector(mode = "list", length = ncols)
beta_int <- vector(mode = "list", length = ncols)
beta_coeff <- vector(mode = "list", length = ncols)

for (i in 1:ncols){
  dummies[[i]] <- seasonaldummy(data_ts[,i])
  seasonal_model[[i]] = lm(data[,(i+1)]~dummies[[i]])
  beta_int[[i]] <- seasonal_model[[i]]$coefficients[1]
  beta_coeff[[i]] <- seasonal_model[[i]]$coefficients[2:12]
   print(colnames(data[i+1]))
   print(summary(seasonal_model[[i]]))
}
```

```
## [1] "Total Biomass Energy Production"
##
## Call:
## lm(formula = data[, (i + 1)] ~ dummies[[i]])
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -153.47  -50.56  -20.25   52.13  182.84
##
## Coefficients:
##                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)     280.5693    12.7954  21.927   <2e-16 ***
## dummies[[i]]Jan  -1.0039    18.0009  -0.056    0.956
## dummies[[i]]Feb -29.3891    18.0009  -1.633    0.103
## dummies[[i]]Mar  -8.6090    18.0009  -0.478    0.633
## dummies[[i]]Apr -20.5046    18.0009  -1.139    0.255
## dummies[[i]]May -14.0960    18.0009  -0.783    0.434
## dummies[[i]]Jun -19.5548    18.0009  -1.086    0.278
## dummies[[i]]Jul  -3.4306    18.0009  -0.191    0.849
## dummies[[i]]Aug   0.2220    18.0009   0.012    0.990
## dummies[[i]]Sep -11.9821    18.0009  -0.666    0.506
## dummies[[i]]Oct  -0.5379    18.0009  -0.030    0.976
## dummies[[i]]Nov  -9.3753    18.0954  -0.518    0.605
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 87.72 on 562 degrees of freedom
## Multiple R-squared:  0.01116,    Adjusted R-squared:  -0.008199
## F-statistic: 0.5764 on 11 and 562 DF,  p-value: 0.8486
##
## [1] "Total Renewable Energy Production"
##
## Call:
## lm(formula = data[, (i + 1)] ~ dummies[[i]])
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -263.99 -102.98  -52.33   36.68  453.58
##
## Coefficients:
##                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)      580.912     24.406  23.802   <2e-16 ***
## dummies[[i]]Jan   12.451     34.335   0.363   0.7170
## dummies[[i]]Feb  -38.964     34.335  -1.135   0.2569
## dummies[[i]]Mar   20.515     34.335   0.597   0.5504
## dummies[[i]]Apr    8.294     34.335   0.242   0.8092
## dummies[[i]]May   36.628     34.335   1.067   0.2865
## dummies[[i]]Jun   19.560     34.335   0.570   0.5691
## dummies[[i]]Jul    8.863     34.335   0.258   0.7964
## dummies[[i]]Aug  -18.480     34.335  -0.538   0.5906
## dummies[[i]]Sep  -62.410     34.335  -1.818   0.0696 .
## dummies[[i]]Oct  -42.649     34.335  -1.242   0.2147
## dummies[[i]]Nov  -42.516     34.515  -1.232   0.2185
## ---
```

```
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 167.3 on 562 degrees of freedom
## Multiple R-squared:  0.03244,    Adjusted R-squared:  0.01351
## F-statistic: 1.713 on 11 and 562 DF,  p-value: 0.06702
##
## [1] "Hydroelectric Power Consumption"
##
## Call:
## lm(formula = data[, (i + 1)] ~ dummies[[i]])
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -92.064 -22.897  -2.654  20.642  98.058
##
## Coefficients:
##                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)      238.887      4.863  49.125  < 2e-16 ***
## dummies[[i]]Jan   13.270      6.841   1.940  0.05291 .
## dummies[[i]]Feb   -8.133      6.841  -1.189  0.23499
## dummies[[i]]Mar   20.442      6.841   2.988  0.00293 **
## dummies[[i]]Apr   17.199      6.841   2.514  0.01221 *
## dummies[[i]]May   40.726      6.841   5.953 4.64e-09 ***
## dummies[[i]]Jun   31.764      6.841   4.643 4.28e-06 ***
## dummies[[i]]Jul   10.858      6.841   1.587  0.11306
## dummies[[i]]Aug  -17.907      6.841  -2.618  0.00909 **
## dummies[[i]]Sep  -50.121      6.841  -7.326 8.26e-13 ***
## dummies[[i]]Oct  -49.165      6.841  -7.187 2.12e-12 ***
## dummies[[i]]Nov  -32.757      6.877  -4.763 2.43e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 33.34 on 562 degrees of freedom
## Multiple R-squared:  0.4345, Adjusted R-squared:  0.4234
## F-statistic: 39.25 on 11 and 562 DF,  p-value: < 2.2e-16
```

All three series show seasonal behaviour as can be seen from the coefficients b1 to b12 for each series. All the coefficients are significantly non-zero and hence can't be ignored.

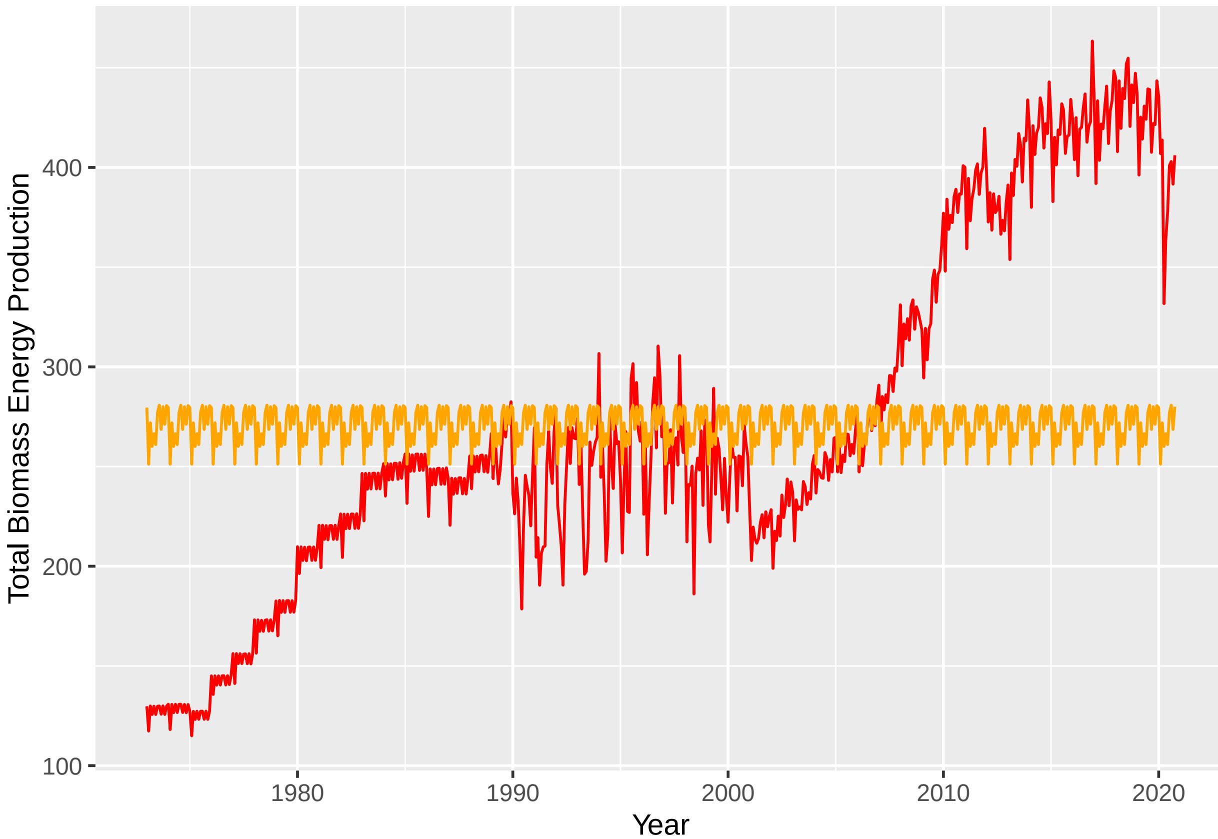
Q7

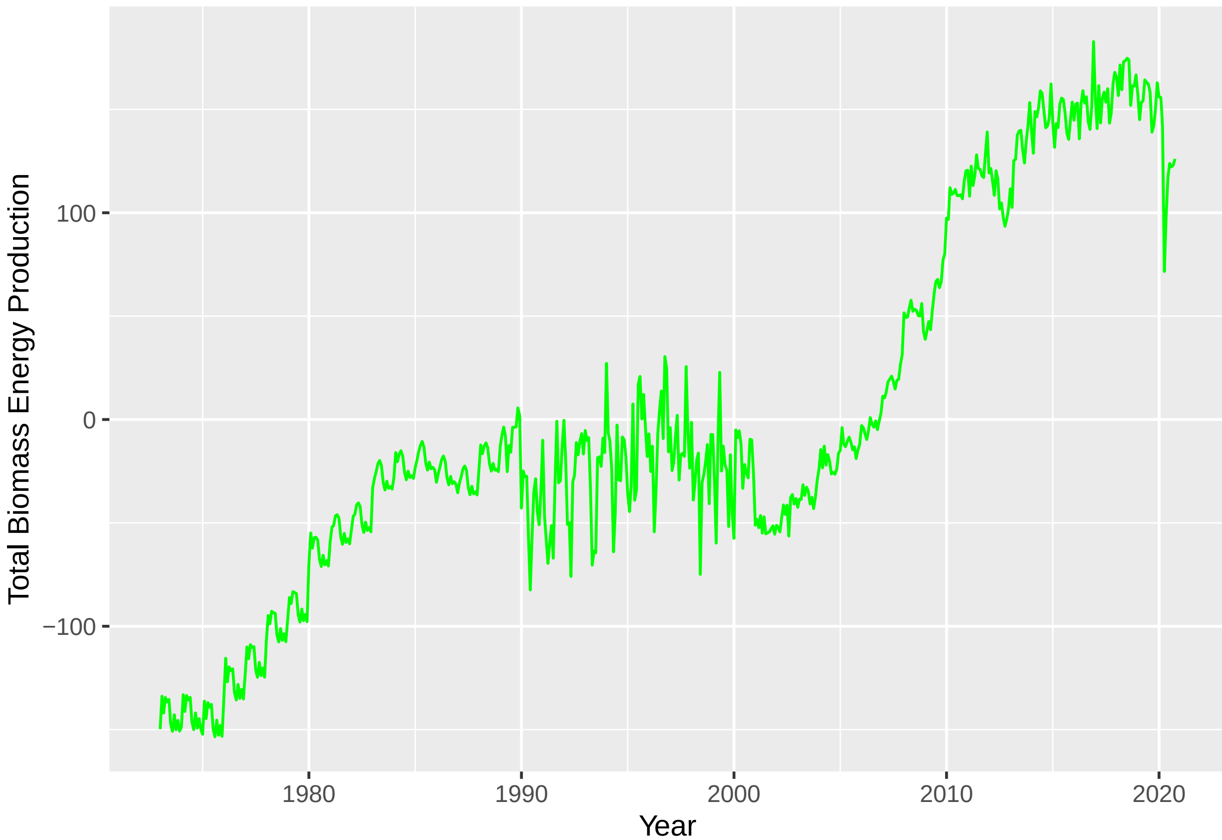
Use the regression coefficients from Q6 to deseason the series. Plot the deseason series and compare with the plots from part Q1. Did anything change?

Original series in red. Deseasoned series in green.

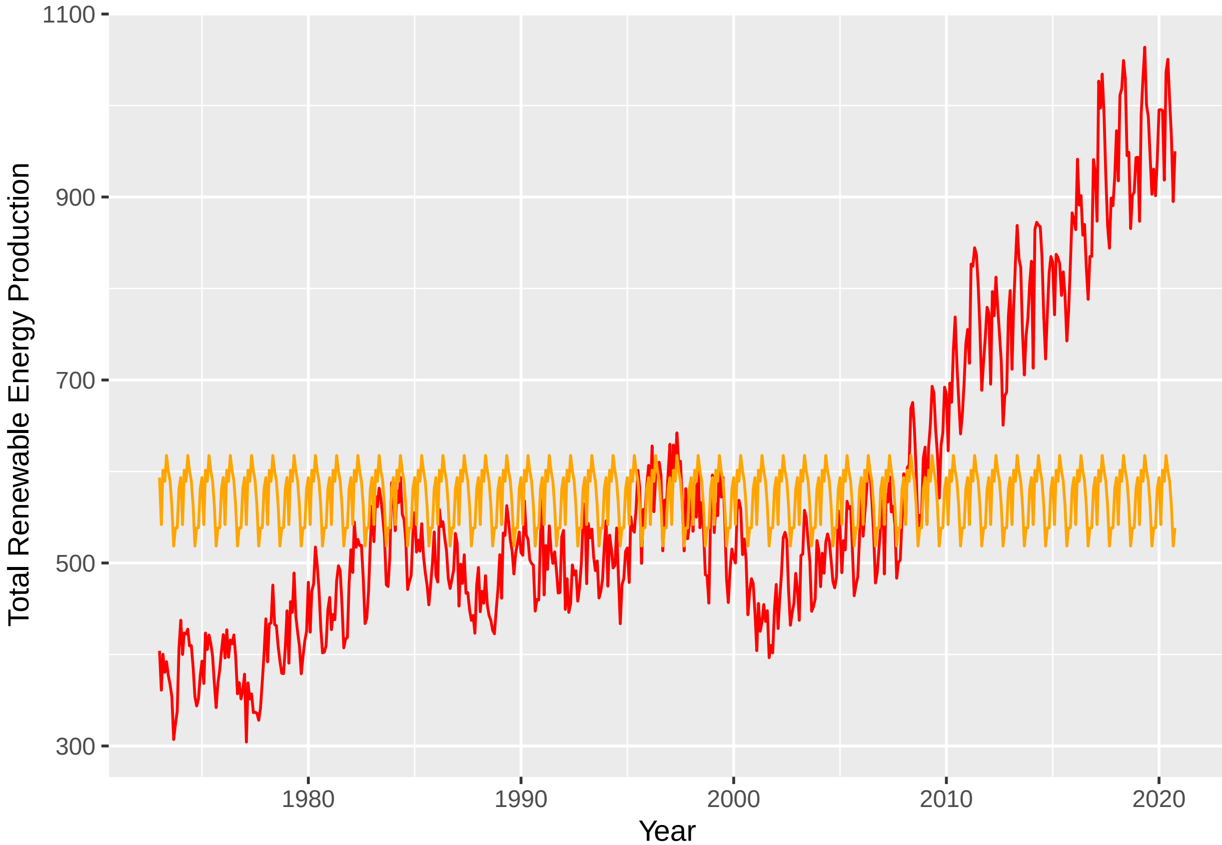
```r
seasonal_comp = vector(mode = "list", length = ncols)
deseasoned_data <- data.frame(matrix(nrow = nobs,ncol = ncols))
for (i in 1:ncols){
  seasonal_comp[i] = array(0,nobs)
  for (j in 1:nobs){
  seasonal_comp[[i]][j] = beta_int[[i]] + beta_coeff[[i]]%*%dummies[[i]][j,]
  }
deseasoned_data[i] <- data[i+1] - seasonal_comp[[i]]
par(mfrow =  c(1,2))
```

```r
print(ggplot(data, aes(x = data$Month, y = data[,i+1])) +
        geom_line(color = "red") +
        xlab("Year") +
        ylab(colnames(data[i+1])) +
        geom_line(aes(y = seasonal_comp[[i]]), color = "orange")
      )
print(ggplot(deseasoned_data, aes(x = data$Month, y = deseasoned_data[,i])) +
        geom_line(color = "green") +
        xlab("Year") +
        ylab(colnames(data[i+1]))
)
}
```

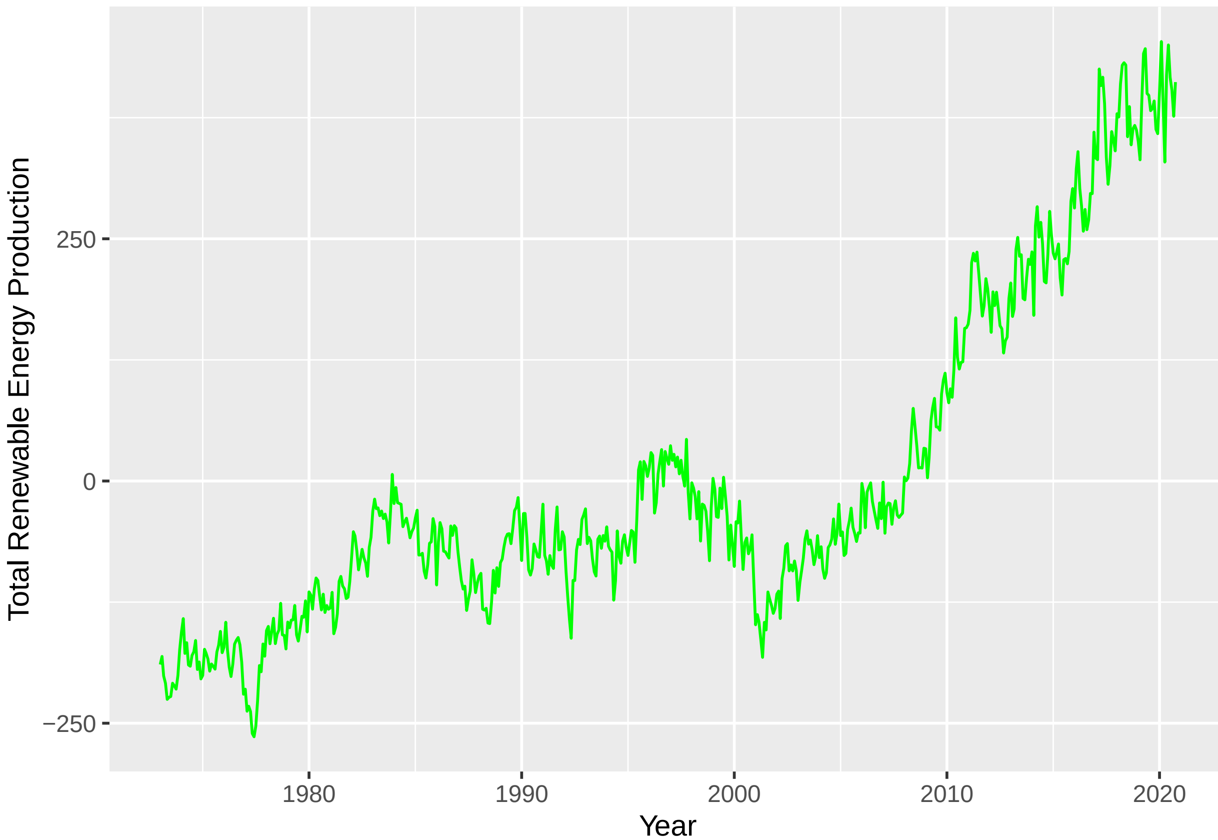




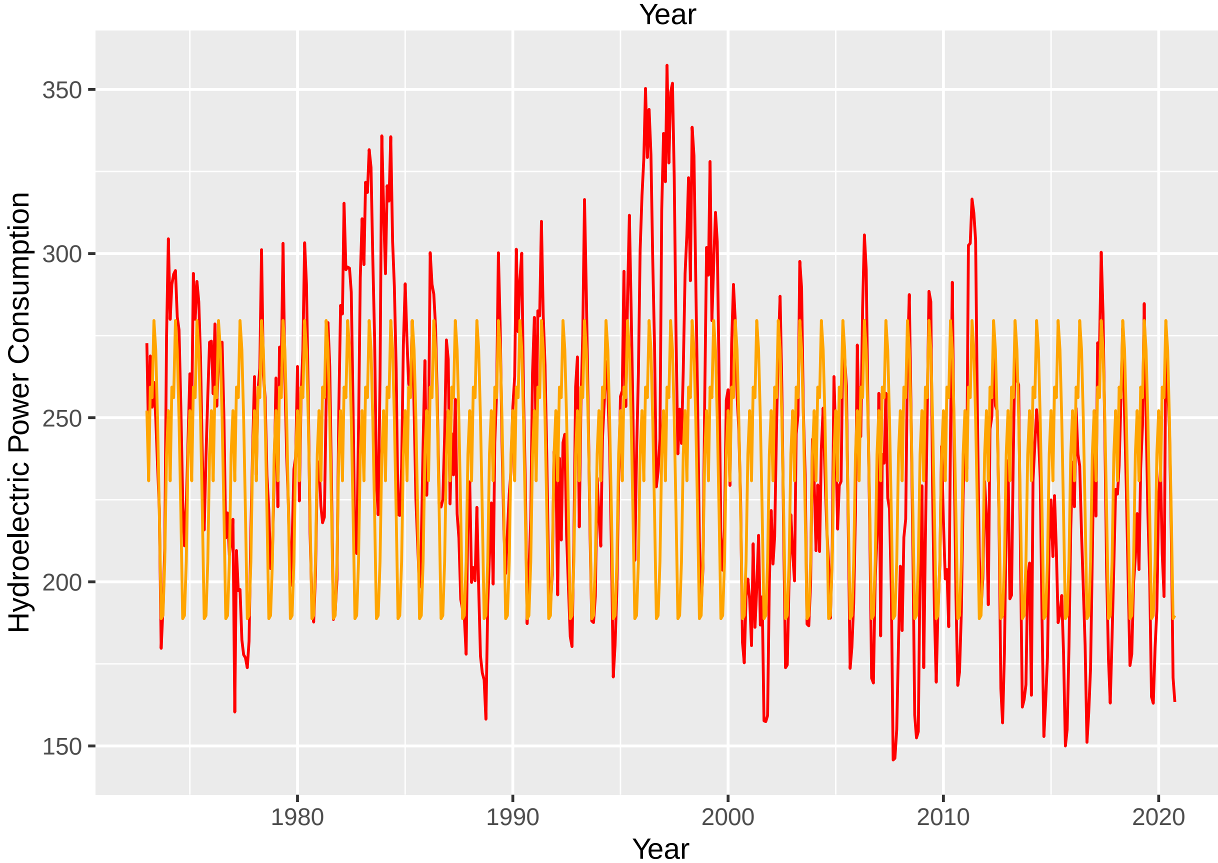
Total Biomass Energy Production
100
0
−100
1980
1990
2000
2010
2020
Year



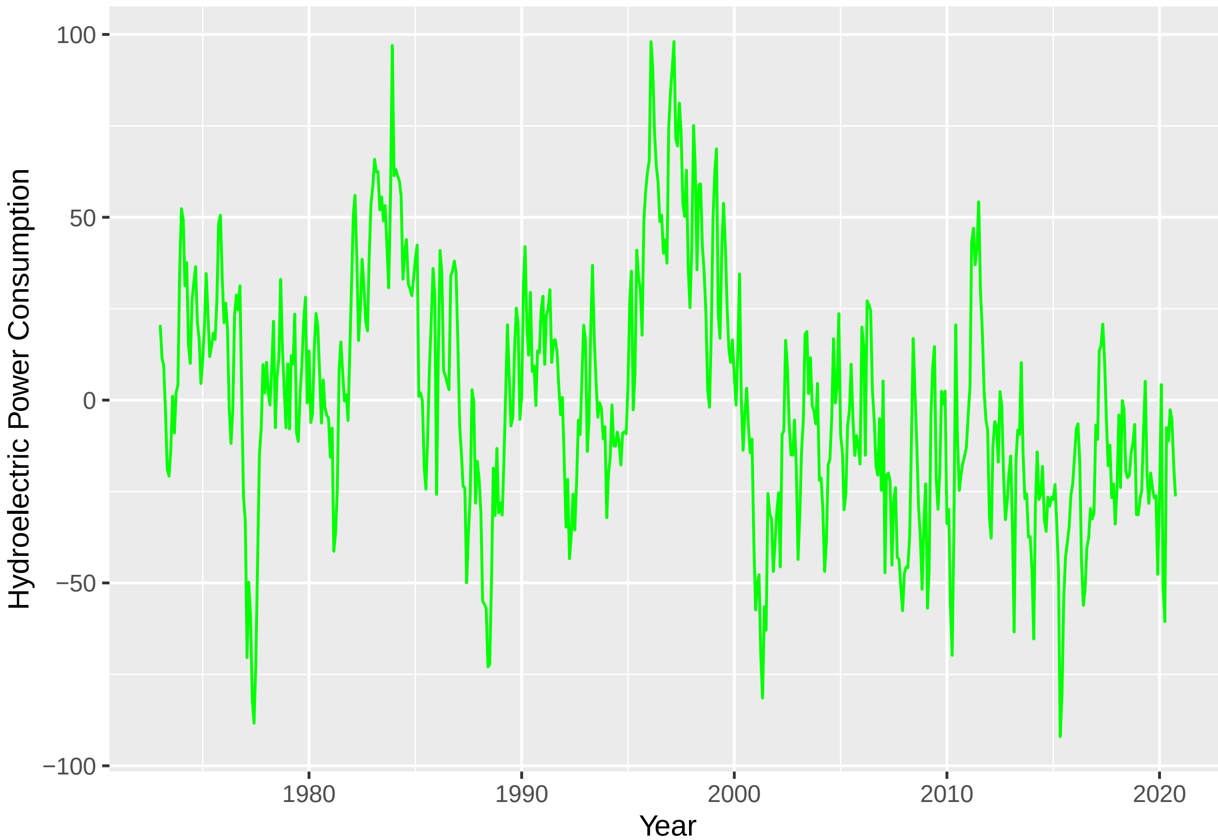
Total Renewable Energy Production
1100
900
700
500
300
1980
1990
2000
2010
2020
Year



Total Renewable Energy Production
250
0
−250
1980
1990
2000
2010
2020
Year



Hydroelectric Power Consumption
350
300
250
200
150
1980
1990
2000
2010
2020
Year



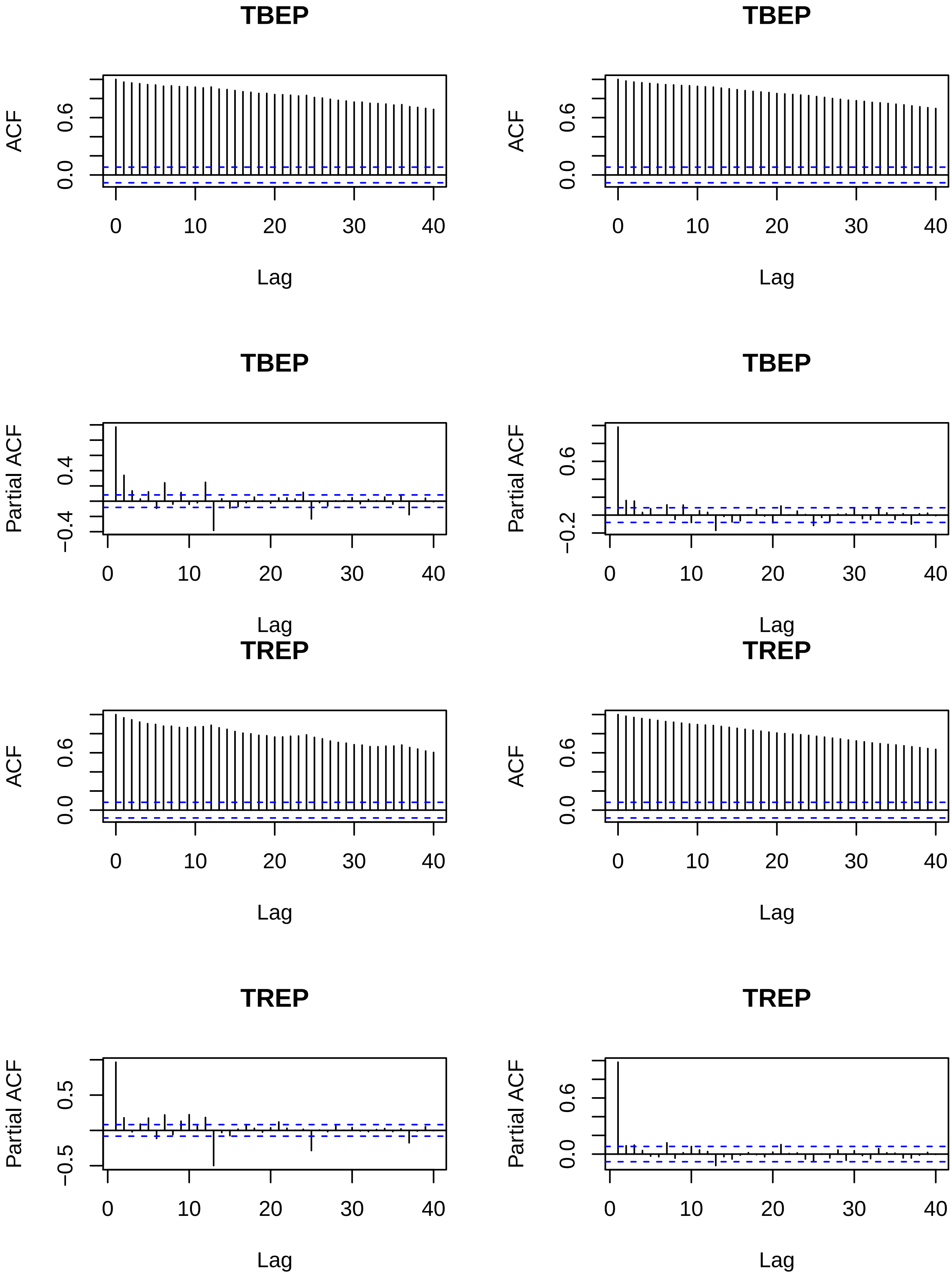
Yes, after removing the seasonality of the series, we observe a down-shift in the data points. The series now only has a trend along with random variation.

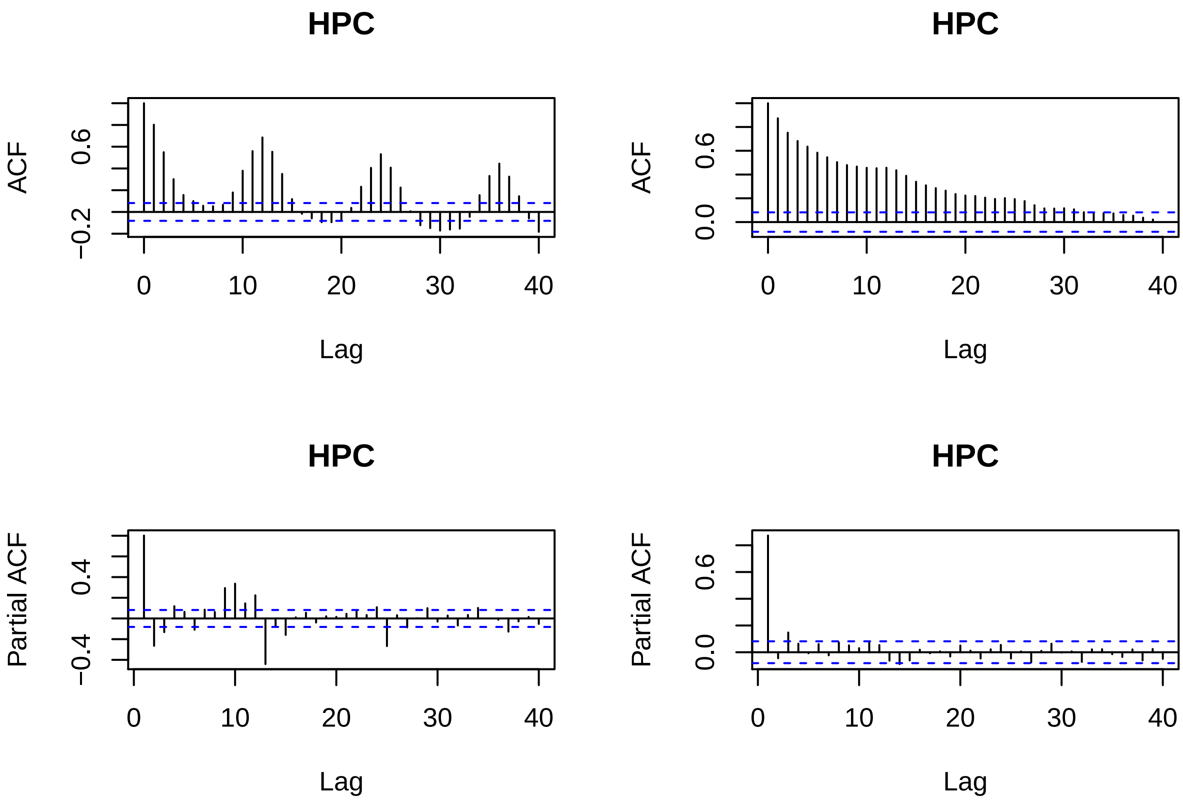
Q8

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

Plots on the left are the original series. Plots on the right are the detrended series.

```
for (i in 1:ncols){
  par(mfrow = c(2,2))
  acf(data[,i+1], lag = 40, main = shortnames[i])
  acf(deseasoned_data[,i], lag = 40, main = shortnames[i])
  pacf(data[,i+1], lag = 40, main = shortnames[i])
  pacf(deseasoned_data[,i], lag = 40, main = shortnames[i])
}
```


TBEP
ACF
0.0
0.6
0
10
20
30
40
Lag
TBEP
ACF
0.0
0.6
0
10
20
30
40
Lag
TBEP
Partial ACF
−0.4
0.4
0
10
20
30
40
Lag
TBEP
Partial ACF
−0.2
0.6
0
10
20
30
40
Lag
TREP
ACF
0.0
0.6
0
10
20
30
40
Lag
TREP
ACF
0.0
0.6
0
10
20
30
40
Lag
TREP
Partial ACF
−0.5
0.5
0
10
20
30
40
Lag
TREP
Partial ACF
0.0
0.6
0
10
20
30
40
Lag



Yes, the plots have now changed. The ACF and PACF plots on the right clearly show no seasonality. After the removal of the seasonality, we can observe that the trend is more prominent in the ACF and PACF on the right. Peaks at lag = 13, 25, 37 are now smaller.