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

Assignment 3 - Due date 02/15/21

Aria Cai

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(lubridate)
```

```
##
## Attaching package: 'lubridate'

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

```
library(readxl)
library(forecast)
```

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

```r
library(tseries)
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(ggplot2)
library(Kendall)
```

##Trend Component

```r
#Importing data set
raw_data <- read_excel("../Data/Table_10.1_Renewable_Energy_Production_and_Consumption_by_Source.xlsx",s

spec_data <- raw_data[,4:6]
head(spec_data)
```

```
## # A tibble: 6 x 3
##   `Total Biomass Energy Pr~ `Total Renewable Energy Pr~ `Hydroelectric Power Co~
##   <chr>                     <chr>                       <chr>
## 1 (Trillion Btu)            (Trillion Btu)              (Trillion Btu)
## 2 129.787                   403.981                     272.703
## 3 117.338                   360.9                       242.199
## 4 129.938                   400.161                     268.81
## 5 125.636                   380.47                      253.185
## 6 129.834                   392.141                     260.77
```

```r
my_date <- raw_data[,1]
head(my_date)
```

```
## # A tibble: 6 x 1
##   Month
##   <dttm>
## 1 NA
## 2 1973-01-01 00:00:00
## 3 1973-02-01 00:00:00
## 4 1973-03-01 00:00:00
## 5 1973-04-01 00:00:00
## 6 1973-05-01 00:00:00
```

```r
spec_data <- cbind(my_date, spec_data)
colnames(spec_data)=c("Date", "TBEP", "TREP","HPC")
spec_data <- spec_data[-c(1),]
cols.num <- c("TBEP","TREP","HPC")
spec_data[cols.num] <- sapply(spec_data[cols.num],as.numeric)
head(spec_data)
```

```
##         Date    TBEP    TREP     HPC
## 2 1973-01-01 129.787 403.981 272.703
## 3 1973-02-01 117.338 360.900 242.199
```

```
## 4 1973-03-01 129.938 400.161 268.810
## 5 1973-04-01 125.636 380.470 253.185
## 6 1973-05-01 129.834 392.141 260.770
## 7 1973-06-01 125.611 377.232 249.859
```

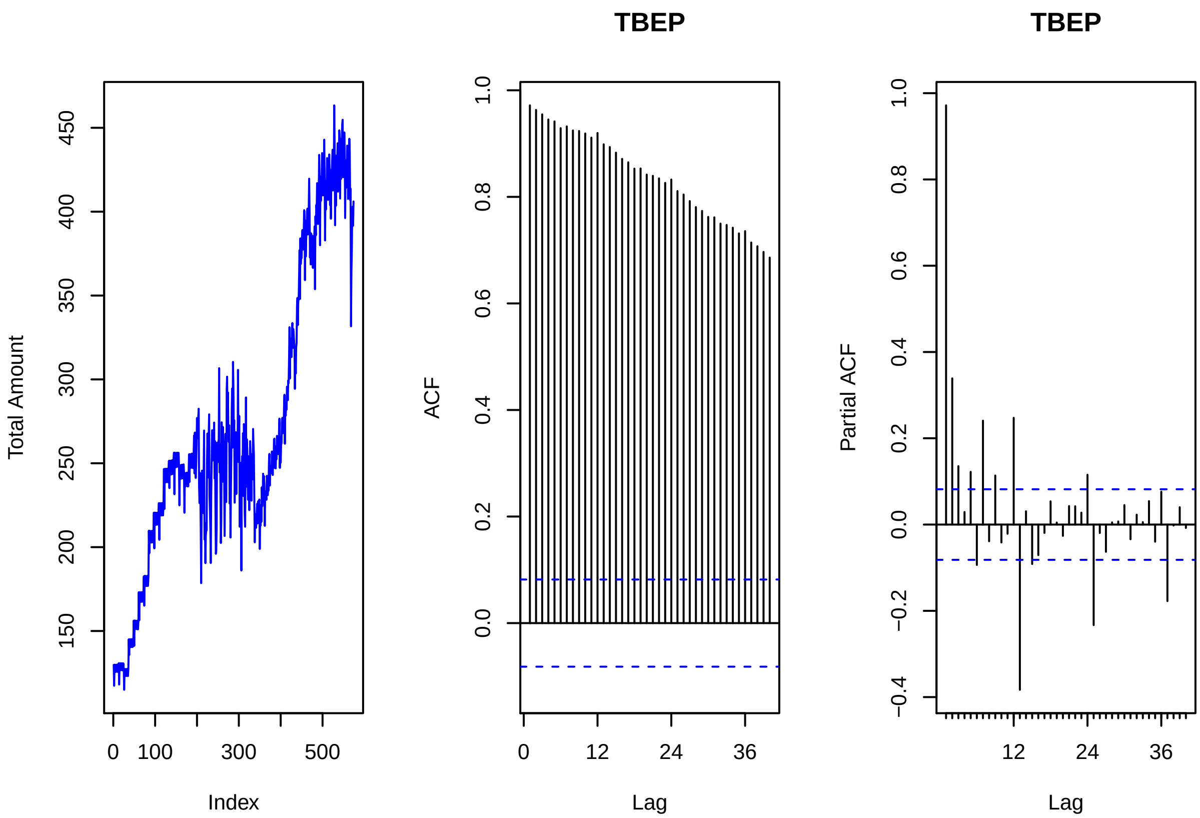
```
nobs <- nrow(spec_data)

ts_data <- ts(spec_data[,2:4],frequency = 12)
head(ts_data,15)
```

```
##          TBEP    TREP     HPC
## Jan 1 129.787 403.981 272.703
## Feb 1 117.338 360.900 242.199
## Mar 1 129.938 400.161 268.810
## Apr 1 125.636 380.470 253.185
## May 1 129.834 392.141 260.770
## Jun 1 125.611 377.232 249.859
## Jul 1 129.787 367.325 235.670
## Aug 1 129.918 353.757 222.077
## Sep 1 125.782 307.006 179.733
## Oct 1 129.970 323.453 191.723
## Nov 1 125.643 337.817 210.285
## Dec 1 129.824 406.694 274.435
## Jan 2 130.807 437.467 304.506
## Feb 2 118.091 399.942 279.950
## Mar 2 130.727 423.474 290.582
```

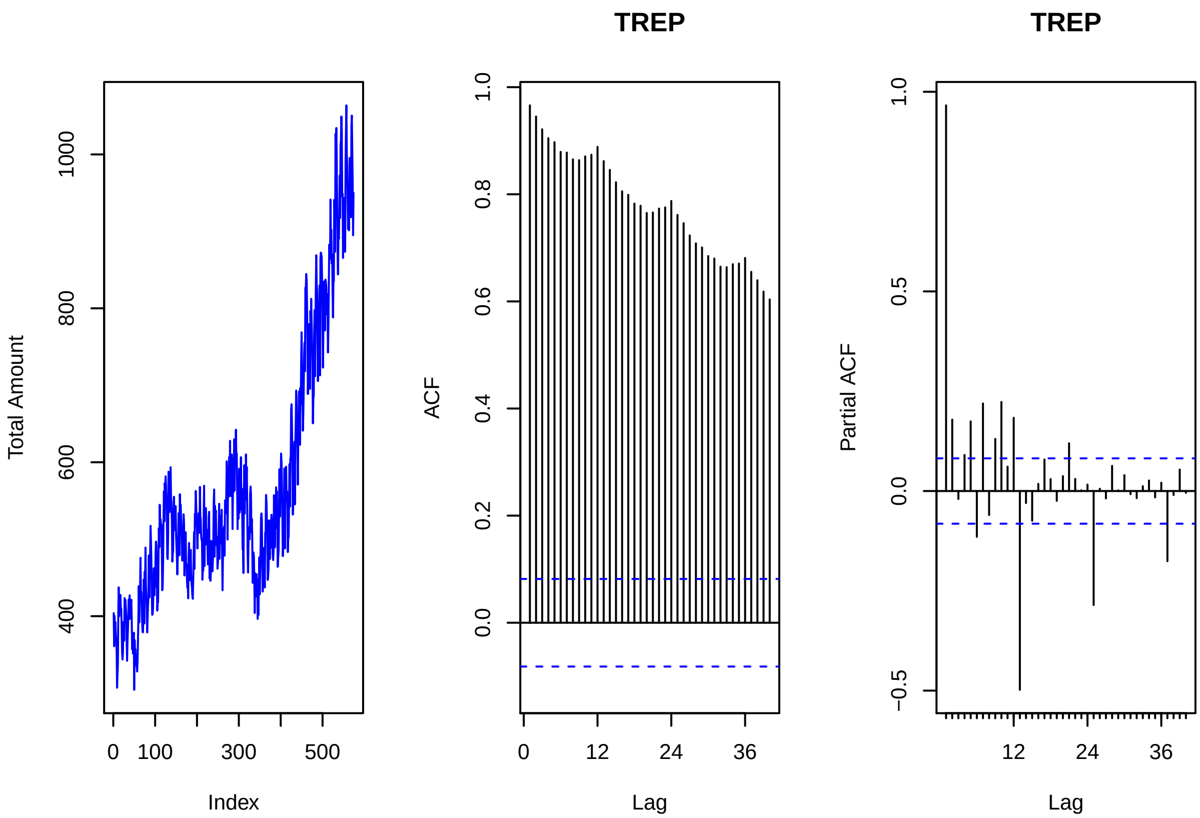
Q1

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)

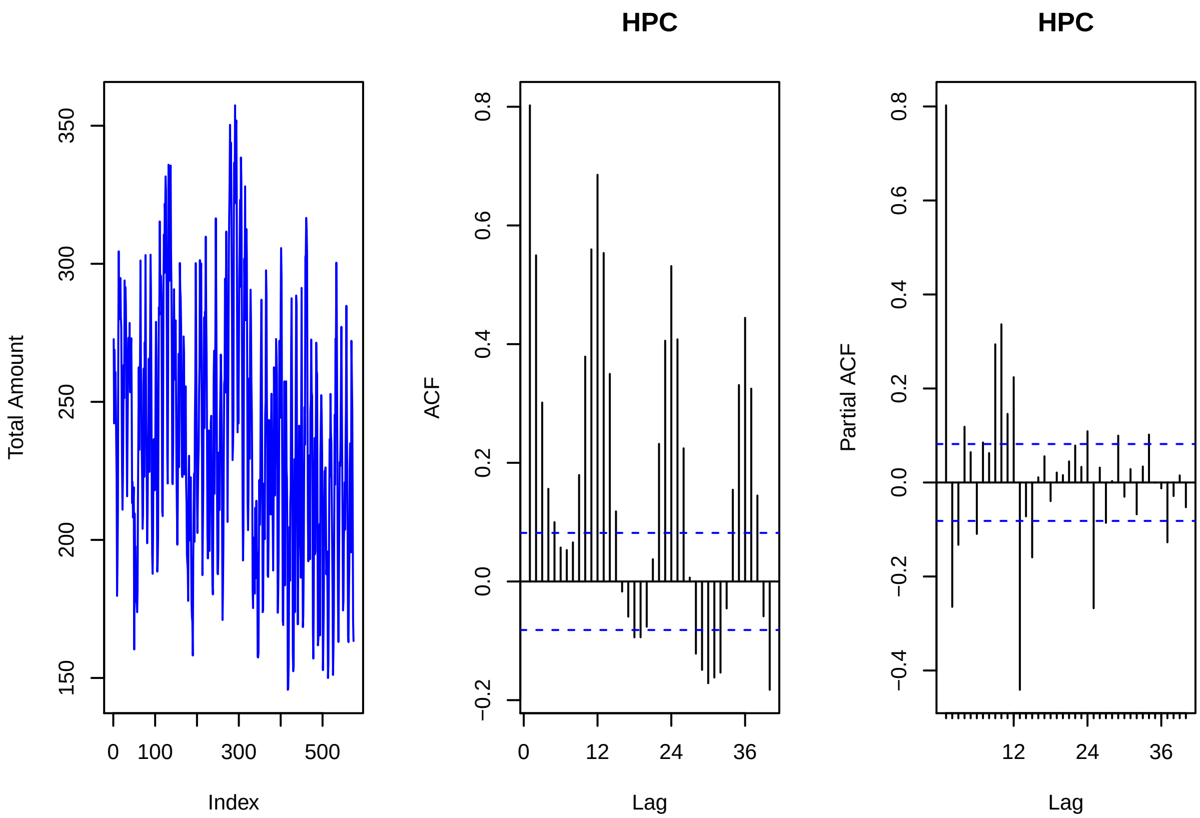
```
par(mfrow=c(1,3)) #place plot side by side
plot(spec_data[,"TBEP"],type="l",col="blue",ylab="Total Amount")
Acf(ts_data[,1],lag.max=40,main=paste("TBEP"))
Pacf(ts_data[,1],lag.max=40,main=paste("TBEP"))
```




```
par(mfrow=c(1,3)) #place plot side by side
plot(spec_data[,"TREP"],type="l",col="blue",ylab="Total Amount")
Acf(ts_data[,2],lag.max=40,main=paste("TREP"))
Pacf(ts_data[,2],lag.max=40,main=paste("TREP"))
```




```
par(mfrow=c(1,3)) #place plot side by side
plot(spec_data[,"HPC"],type="l",col="blue",ylab="Total Amount")
Acf(ts_data[,3],lag.max=40,main=paste("HPC"))
Pacf(ts_data[,3],lag.max=40,main=paste("HPC"))
```



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?

According to the ACF plot, it seems that the hydroelectric power consumption is correlated to seasonality. And the total biomass energy and total renewable energy production shows a high coefficient at lag1 and continue decreasing as the lag increase. it shows these two series have linear trends and it's not correalted to 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)
#TBEP
i=1
linear_trend_model_TBEP=lm(spec_data[,i+1]~t)
summary(linear_trend_model_TBEP)
```

```
##
## Call:
## lm(formula = spec_data[, i + 1] ~ 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
```

```
#TREP
i=2
linear_trend_model_TREP=lm(spec_data[,i+1]~t)
summary(linear_trend_model_TREP)
```

```
##
## Call:
## lm(formula = spec_data[, i + 1] ~ 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
```

```
#HPC
i=3
linear_trend_model_HPC=lm(spec_data[,i+1]~t)
summary(linear_trend_model_HPC)
```

```
##
## Call:
## lm(formula = spec_data[, i + 1] ~ 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
```

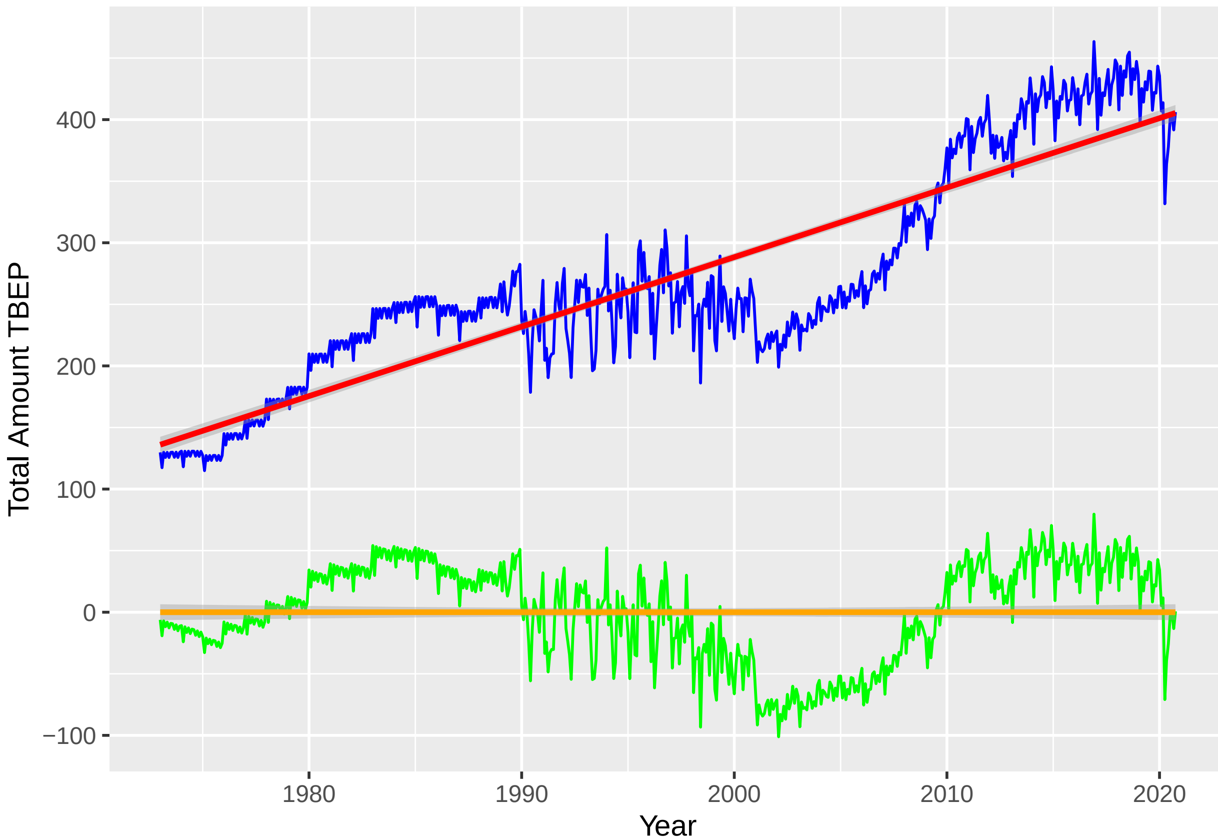
Q4

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?

For the total biomass energy production and total renewable energy production, the detrended series does not shows the strong positive increase with time. And it also does not show correlation with seasonality. The total amount of these two production is negative during 2000-2010, and the rest of them are positive. For hydroelectric power consumption, there are smaller changes because the original trend is negative with a small slope.

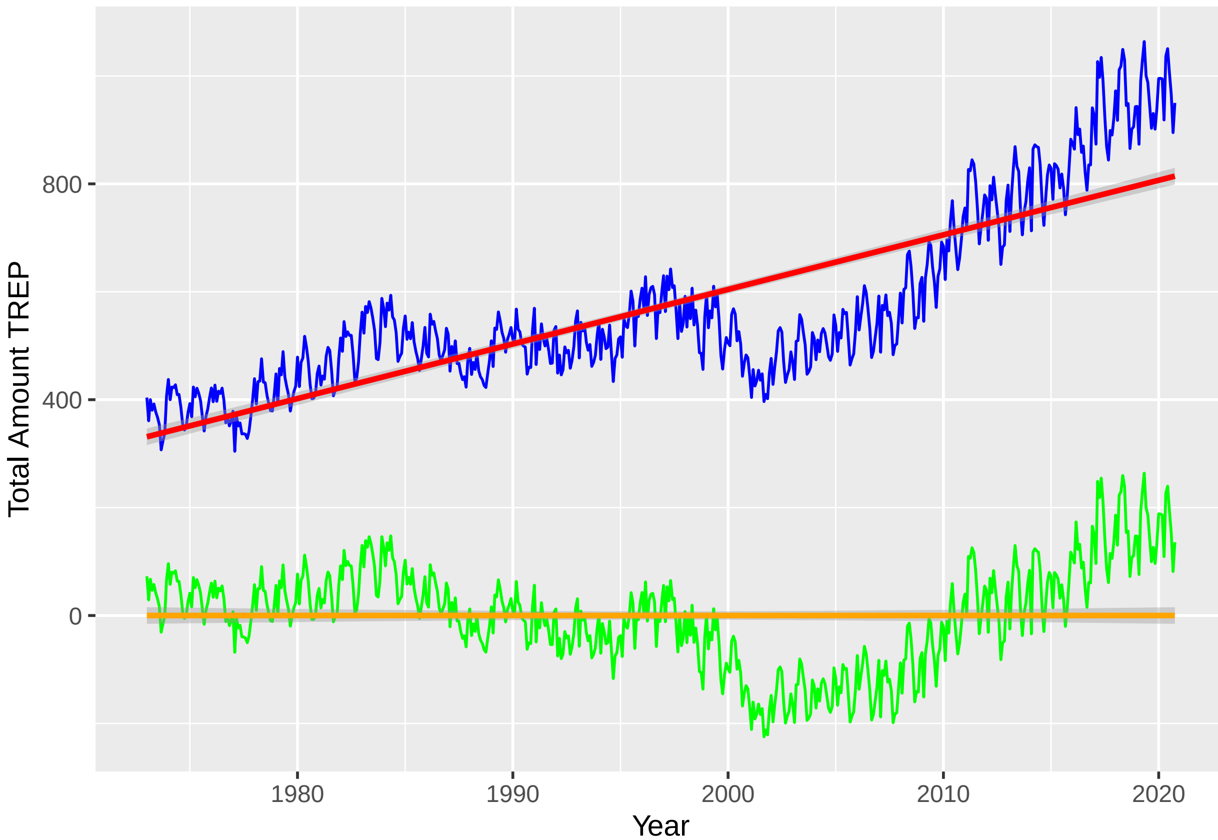
```
i = 1
beta0=as.numeric(linear_trend_model_TBEP$coefficients[1])  #first coefficient is the intercept term or
beta1=as.numeric(linear_trend_model_TBEP$coefficients[2])  #second coefficient is the slope or beta1
detrend_data_TBEP<- spec_data[,(i+1)]-(beta0+beta1*t)

ggplot(spec_data, aes(x=spec_data[,1], y=spec_data[,(1+i)])) +
            geom_line(color="blue") +
            ylab(paste0("Total Amount ",colnames(spec_data)[(1+i)],sep="")) +
            xlab("Year")+
            geom_abline(intercept = beta0, slope = beta1, color="red")+
            geom_smooth(color="red",method="lm")+
            geom_line(aes(y=detrend_data_TBEP), col="green")+
            geom_smooth(aes(y=detrend_data_TBEP),color="orange",method="lm")
```

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



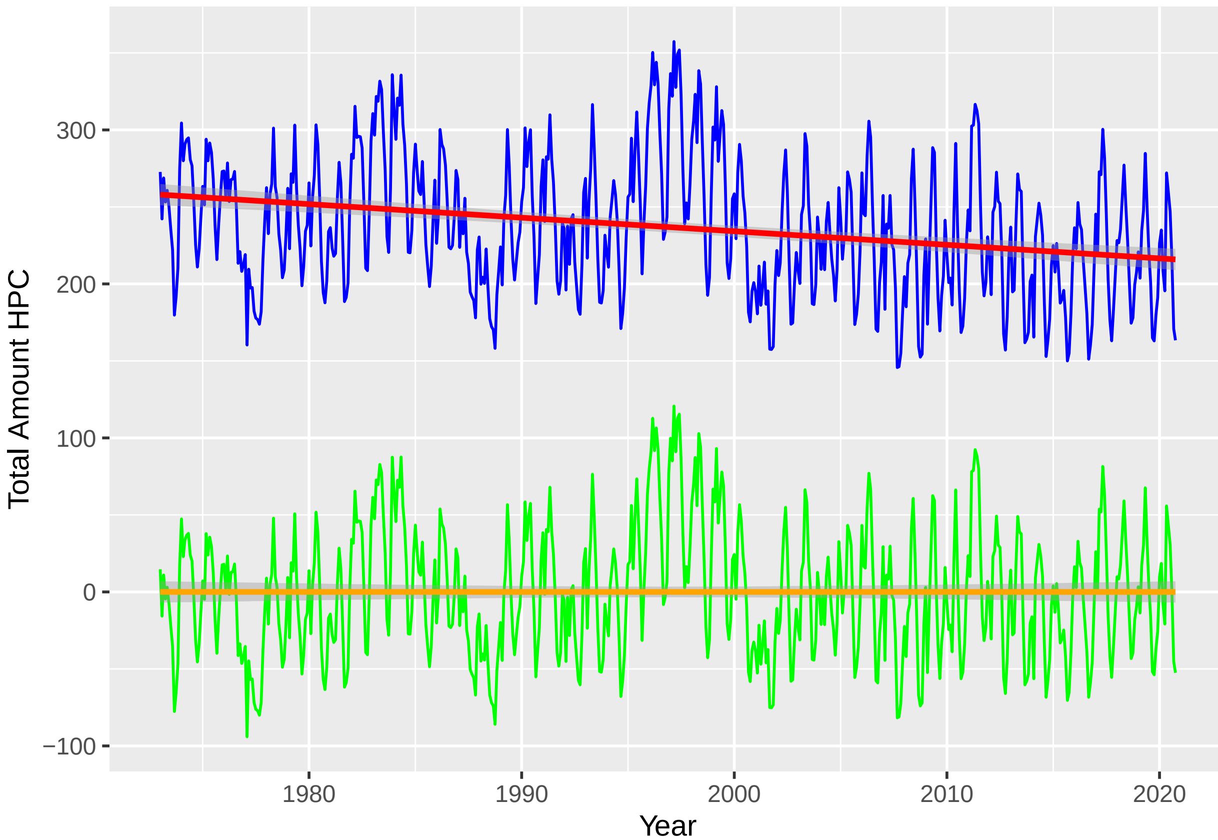

```r
i = 2
beta0=as.numeric(linear_trend_model_TREP$coefficients[1])  #first coefficient is the intercept term or
beta1=as.numeric(linear_trend_model_TREP$coefficients[2])  #second coefficient is the slope or beta1

detrend_data_TREP<- spec_data[,(i+1)]-(beta0+beta1*t)

ggplot(spec_data, aes(x=spec_data[,1], y=spec_data[,(1+i)])) +
            geom_line(color="blue") +
            ylab(paste0("Total Amount ",colnames(spec_data)[(1+i)],sep="")) +
            xlab("Year")+
            geom_abline(intercept = beta0, slope = beta1, color="red")+
            geom_smooth(color="red",method="lm")+
            geom_line(aes(y=detrend_data_TREP), col="green")+
            geom_smooth(aes(y=detrend_data_TREP),color="orange",method="lm")
```

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




```
i = 3
beta0=as.numeric(linear_trend_model_HPC$coefficients[1])  #first coefficient is the intercept term or b
beta1=as.numeric(linear_trend_model_HPC$coefficients[2])  #second coefficient is the slope or beta1

detrend_data_HPC<- spec_data[,(i+1)]-(beta0+beta1*t)

ggplot(spec_data, aes(x=spec_data[,1], y=spec_data[,(1+i)])) +
            geom_line(color="blue") +
            ylab(paste0("Total Amount ",colnames(spec_data)[(1+i)],sep="")) +
            xlab("Year")+
            geom_abline(intercept = beta0, slope = beta1, color="red")+
            geom_smooth(color="red",method="lm")+
            geom_line(aes(y=detrend_data_HPC), col="green")+
            geom_smooth(aes(y=detrend_data_HPC),color="orange",method="lm")
```

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

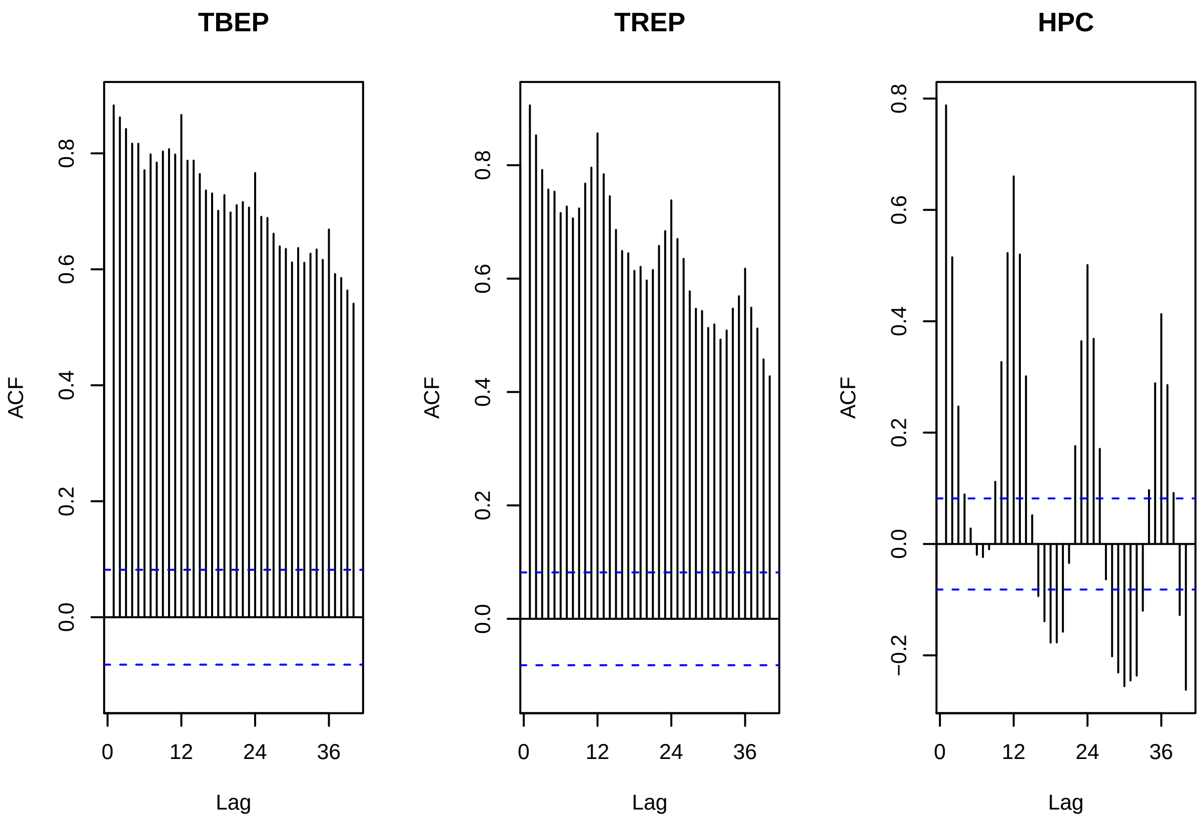


Q5

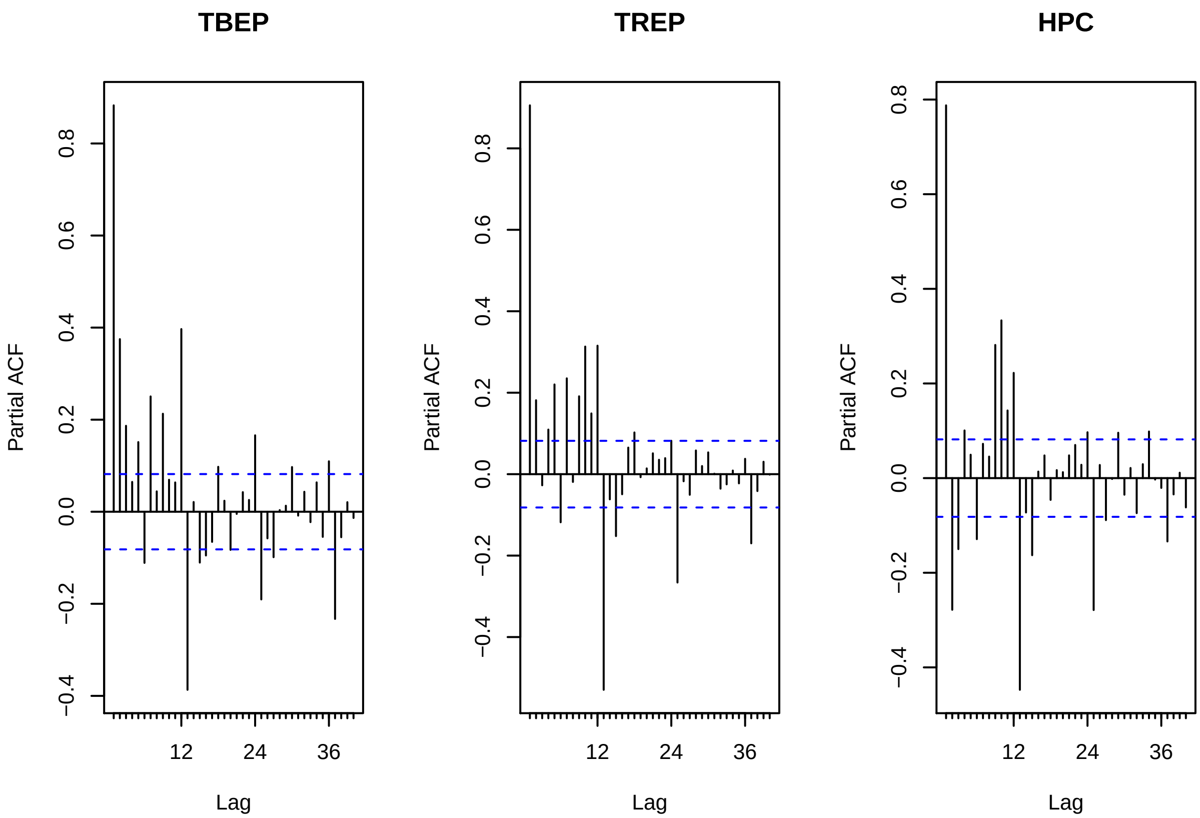
Plot ACF and PACF for the detrended series and compare with the plots from Q1. Did the plots change? How? Yes, the ACF has larger change than the PACF plots. The ACF plots shows higher seasonality in the correlation. However, the shape and amplitude of the PACF plot does not change a lot.

```
ts_TBEP <- ts(data = detrend_data_TBEP, frequency = 12)
ts_TREP <- ts(data = detrend_data_TREP, frequency = 12)
ts_HPC <- ts(data = detrend_data_HPC, frequency = 12)
```

```
par(mfrow=c(1,3)) #place plot side by side
Acf(ts_TBEP,lag.max=40,main=paste("TBEP"))
Acf(ts_TREP,lag.max=40,main=paste("TREP"))
Acf(ts_HPC,lag.max=40,main=paste("HPC"))
```




```
par(mfrow=c(1,3)) #place plot side by side
Pacf(ts_TBEP,lag.max=40,main=paste("TBEP"))
Pacf(ts_TREP,lag.max=40,main=paste("TREP"))
Pacf(ts_HPC,lag.max=40,main=paste("HPC"))
```



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.

```
i = 1
dummies_TBEP <- seasonaldummy(ts_data[,i])
seas_means_model_TBEP=lm(spec_data[,(i+1)]~dummies_TBEP)
summary(seas_means_model_TBEP)
```

```
##
## Call:
## lm(formula = spec_data[, (i + 1)] ~ dummies_TBEP)
##
## 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_TBEPJan  -1.0039    18.0009  -0.056    0.956
## dummies_TBEPFeb -29.3891    18.0009  -1.633    0.103
## dummies_TBEPMar  -8.6090    18.0009  -0.478    0.633
```

```
## dummies_TBEPApr -20.5046    18.0009  -1.139    0.255
## dummies_TBEPMay -14.0960    18.0009  -0.783    0.434
## dummies_TBEPJun -19.5548    18.0009  -1.086    0.278
## dummies_TBEPJul  -3.4306    18.0009  -0.191    0.849
## dummies_TBEPAug   0.2220    18.0009   0.012    0.990
## dummies_TBEPSep -11.9821    18.0009  -0.666    0.506
## dummies_TBEPOct  -0.5379    18.0009  -0.030    0.976
## dummies_TBEPNov  -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
```

```r
beta_int_TBEP=seas_means_model_TBEP$coefficients[1]
beta_coeff_TBEP=seas_means_model_TBEP$coefficients[2:12]
```

```r
i = 2
dummies_TREP <- seasonaldummy(ts_data[,i])
seas_means_model_TREP=lm(spec_data[,(i+1)]~dummies_TREP)
summary(seas_means_model_TREP)
```

```
##
## Call:
## lm(formula = spec_data[, (i + 1)] ~ dummies_TREP)
##
## 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_TREPJan    12.451     34.335   0.363   0.7170
## dummies_TREPFeb   -38.964     34.335  -1.135   0.2569
## dummies_TREPMar    20.515     34.335   0.597   0.5504
## dummies_TREPApr     8.294     34.335   0.242   0.8092
## dummies_TREPMay    36.628     34.335   1.067   0.2865
## dummies_TREPJun    19.560     34.335   0.570   0.5691
## dummies_TREPJul     8.863     34.335   0.258   0.7964
## dummies_TREPAug   -18.480     34.335  -0.538   0.5906
## dummies_TREPSep   -62.410     34.335  -1.818   0.0696 .
## dummies_TREPOct   -42.649     34.335  -1.242   0.2147
## dummies_TREPNov   -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
```

```r
beta_int_TREP=seas_means_model_TREP$coefficients[1]
beta_coeff_TREP=seas_means_model_TREP$coefficients[2:12]
```

```r
i = 3
dummies_HPC <- seasonaldummy(ts_data[,i])
seas_means_model_HPC=lm(spec_data[,(i+1)]~dummies_HPC)
summary(seas_means_model_HPC)
```

```
##
## Call:
## lm(formula = spec_data[, (i + 1)] ~ dummies_HPC)
##
## 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_HPCJan    13.270      6.841   1.940  0.05291 .
## dummies_HPCFeb    -8.133      6.841  -1.189  0.23499
## dummies_HPCMar    20.442      6.841   2.988  0.00293 **
## dummies_HPCApr    17.199      6.841   2.514  0.01221 *
## dummies_HPCMay    40.726      6.841   5.953 4.64e-09 ***
## dummies_HPCJun    31.764      6.841   4.643 4.28e-06 ***
## dummies_HPCJul    10.858      6.841   1.587  0.11306
## dummies_HPCAug   -17.907      6.841  -2.618  0.00909 **
## dummies_HPCSep   -50.121      6.841  -7.326 8.26e-13 ***
## dummies_HPCOct   -49.165      6.841  -7.187 2.12e-12 ***
## dummies_HPCNov   -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
```

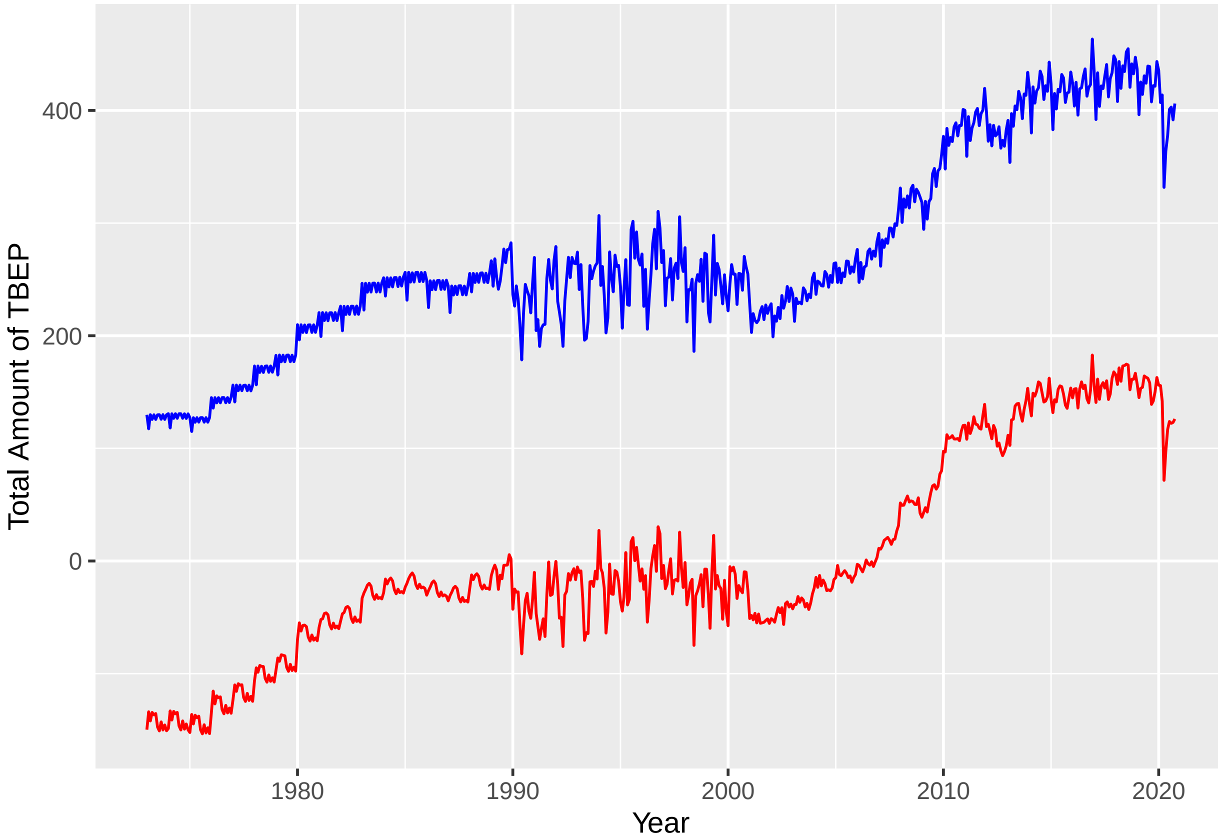
```r
beta_int_HPC=seas_means_model_HPC$coefficients[1]
beta_coeff_HPC=seas_means_model_HPC$coefficients[2:12]
```

Q7

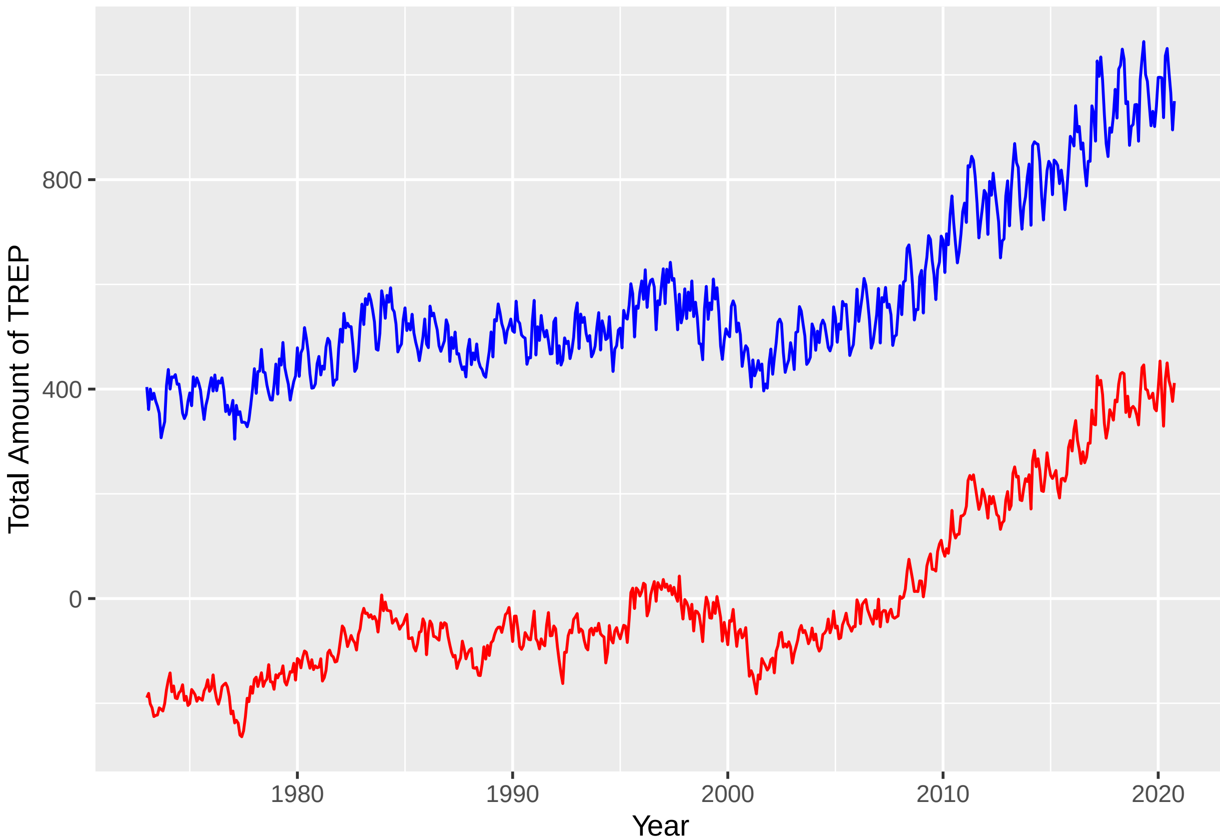
Use the regression coefficients from Q6 to deseason the series. Plot the deseason series and compare with the plots from part Q1. Did anything change? Nothing has been changed for total biomass energy production and total renewable energy production. But the trends for hydroelectric power consumption is more clear, and the change between the neigboring data is smaller.

```r
i=1
seas_comp_TBEP=array(0,nobs)
for(n in 1:nobs){
  seas_comp_TBEP[n]=(beta_int_TBEP+beta_coeff_TBEP%*%dummies_TBEP[n,])
}
deseason_data_TBEP <- spec_data[,(1+i)]-seas_comp_TBEP

ggplot(spec_data, aes(x=spec_data[,1], y=spec_data[,(1+i)])) +
            geom_line(color="blue") +
            xlab("Year")+
            ylab(paste0("Total Amount of ",colnames(spec_data)[(1+i)],sep="")) +
```

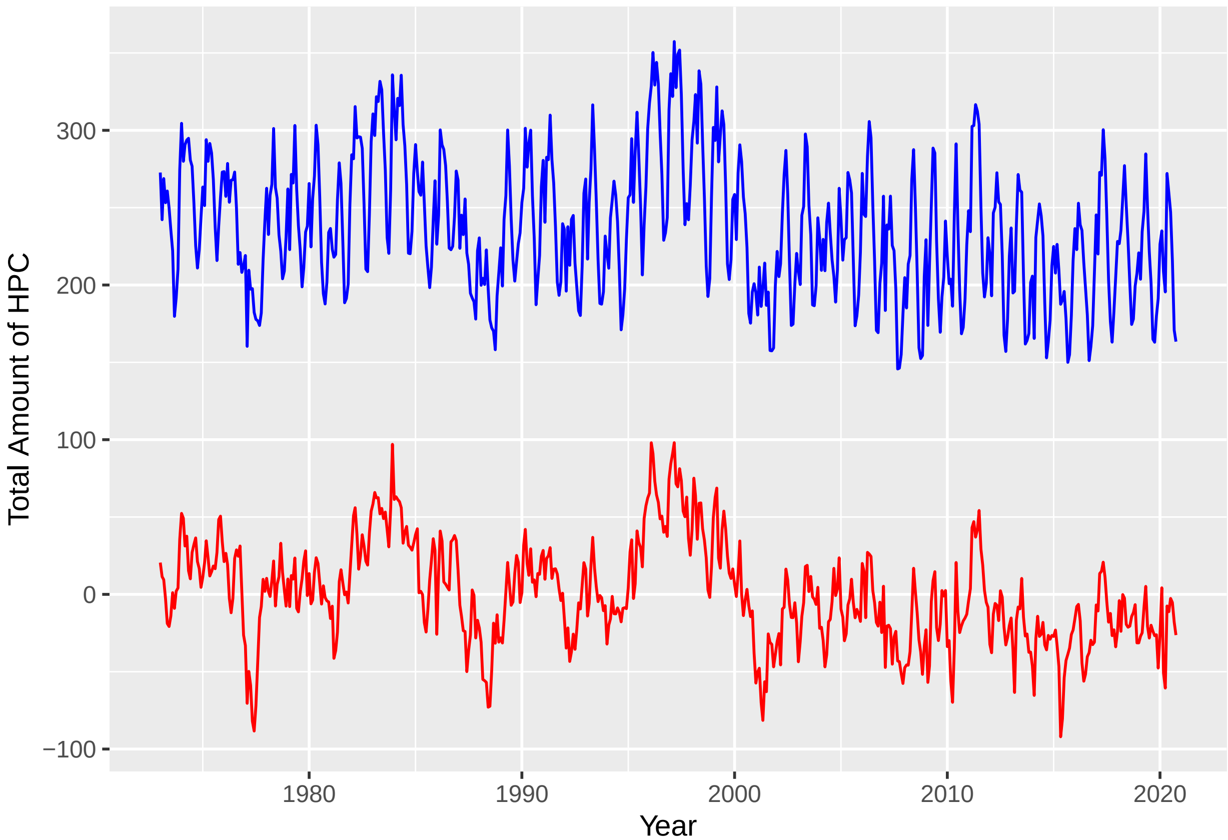
```
           geom_line(aes(y=deseason_data_TBEP), col="red")
```




```
i=2
seas_comp_TREP=array(0,nobs)
for(n in 1:nobs){
  seas_comp_TREP[n]=(beta_int_TREP+beta_coeff_TREP%*%dummies_TREP[n,])
}
deseason_data_TREP <- spec_data[,(1+i)]-seas_comp_TREP

ggplot(spec_data, aes(x=spec_data[,1], y=spec_data[,(1+i)])) +
           geom_line(color="blue") +
           xlab("Year")+
           ylab(paste0("Total Amount of ",colnames(spec_data)[(1+i)],sep="")) +
           geom_line(aes(y=deseason_data_TREP), col="red")
```




```
i=3
seas_comp_HPC=array(0,nobs)
for(n in 1:nobs){
  seas_comp_HPC[n]=(beta_int_HPC+beta_coeff_HPC%*%dummies_HPC[n,])
}
deseason_data_HPC <- spec_data[,(1+i)]-seas_comp_HPC

ggplot(spec_data, aes(x=spec_data[,1], y=spec_data[,(1+i)])) +
            geom_line(color="blue") +
            xlab("Year")+
            ylab(paste0("Total Amount of ",colnames(spec_data)[(1+i)],sep="")) +
            geom_line(aes(y=deseason_data_HPC), col="red")
```



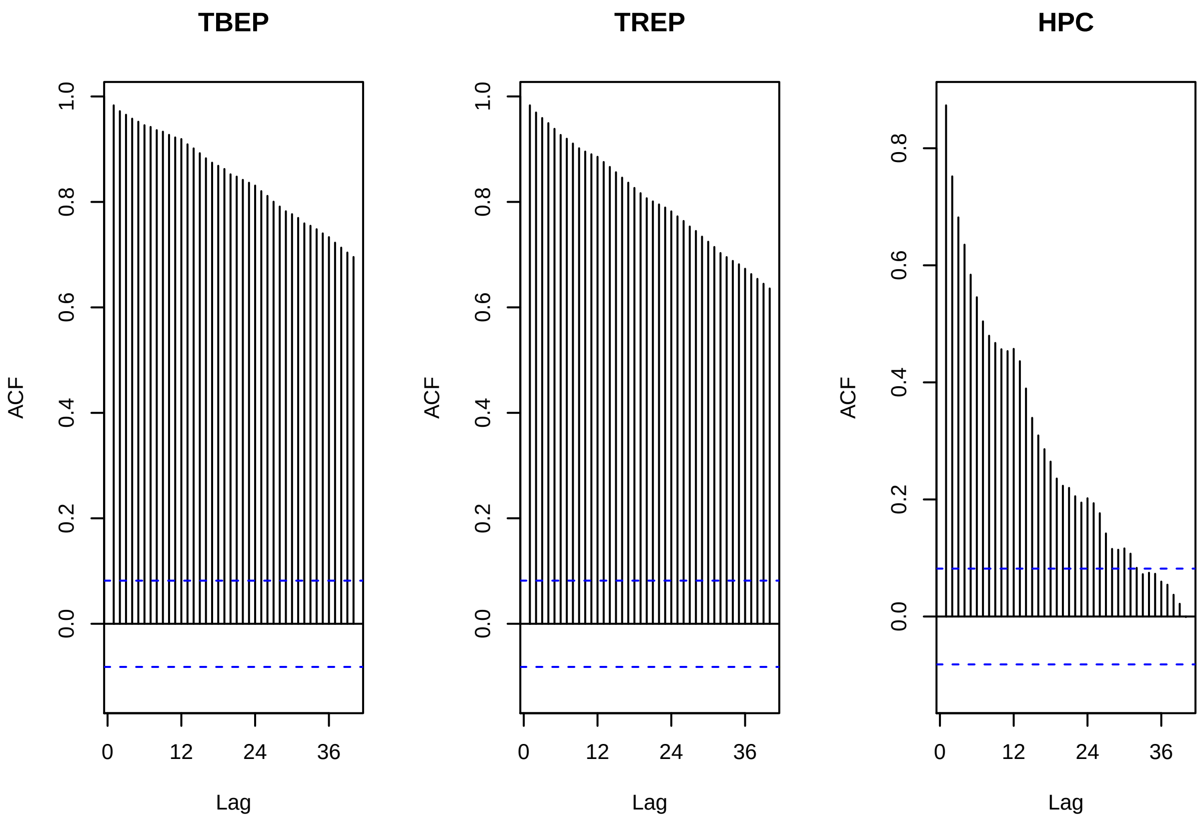
Q8

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

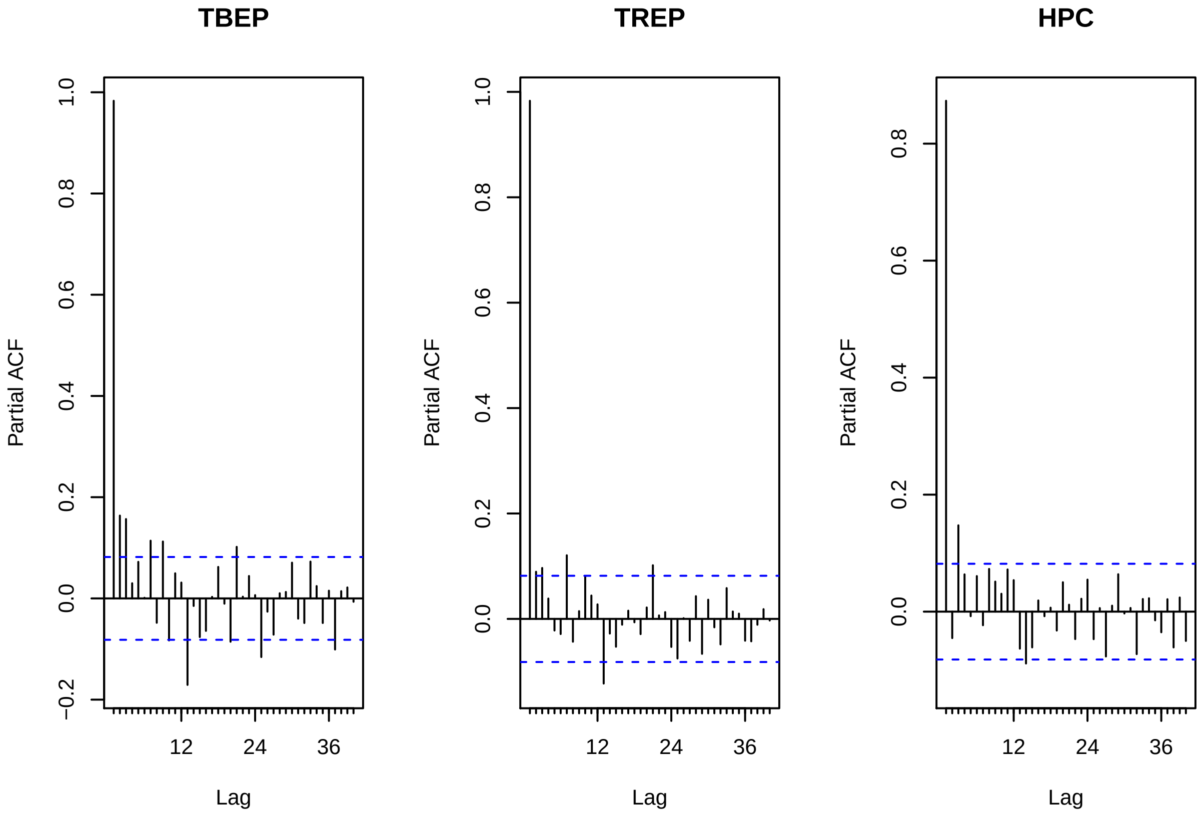
The PACF plots has smaller change than the change in ACF. For Acf plots, the plots for total biomass and renewable energy production does not change. But the ACF plots for Hydroelectric power consumption changes a lot. After deseasoning, the ACF shows that the HPC data has clear decreasing trends over time.

```
ts_TBEP_sea <- ts(data = deseason_data_TBEP, frequency = 12)
ts_TREP_sea <- ts(data = deseason_data_TREP, frequency = 12)
ts_HPC_sea <- ts(data = deseason_data_HPC, frequency = 12)
```

```
par(mfrow=c(1,3)) #place plot side by side
Acf(ts_TBEP_sea,lag.max=40,main=paste("TBEP"))
Acf(ts_TREP_sea,lag.max=40,main=paste("TREP"))
Acf(ts_HPC_sea,lag.max=40,main=paste("HPC"))
```




```
par(mfrow=c(1,3)) #place plot side by side
Pacf(ts_TBEP_sea,lag.max=40,main=paste("TBEP"))
Pacf(ts_TREP_sea,lag.max=40,main=paste("TREP"))
Pacf(ts_HPC_sea,lag.max=40,main=paste("HPC"))
```


TBEP
TREP
HPC
Partial ACF
Lag
12
24
36
1.0
0.8
0.6
0.4
0.2
0.0
−0.2