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

Assignment 3 - Due date 02/15/21

Benjamin Culberson

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.\

```r
#Load/install required package here
library(forecast)
```

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

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

```
library(readxl)
library(lubridate)
```

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

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

```
library(ggplot2)
library(tseries)
```

##Trend Component

Q1

Create a plot window that has one row and three columns. (using the par command) 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)

```
Renewable.df <- read.csv("/Users/benculberson/Documents/Duke /Spring 2021/Time Series Analysis/ENV790_3

Renewable_date<-Renewable.df[12:585,1]

Renewable_altered.df<-Renewable.df[12:585,4:6]




colnames(Renewable_altered.df)=c("Total Biomass Energy Production","Total Renewable Energy Production",

Renewable_altered.df$Date<-Renewable_date

Renewable_altered.df$`Total Biomass Energy Production`<-as.numeric(Renewable_altered.df$`Total Biomass E
Renewable_altered.df$`Total Renewable Energy Production`<-as.numeric(Renewable_altered.df$`Total Renewab
Renewable_altered.df$`Hydroelectric Power Consumption`<-as.numeric(Renewable_altered.df$`Hydroelectric P


#using package lubridate
my_date <- paste(Renewable_altered.df[,4])
my_date <- ym(my_date)  #function my from package lubridate, my is short for month, year

head(my_date)
```

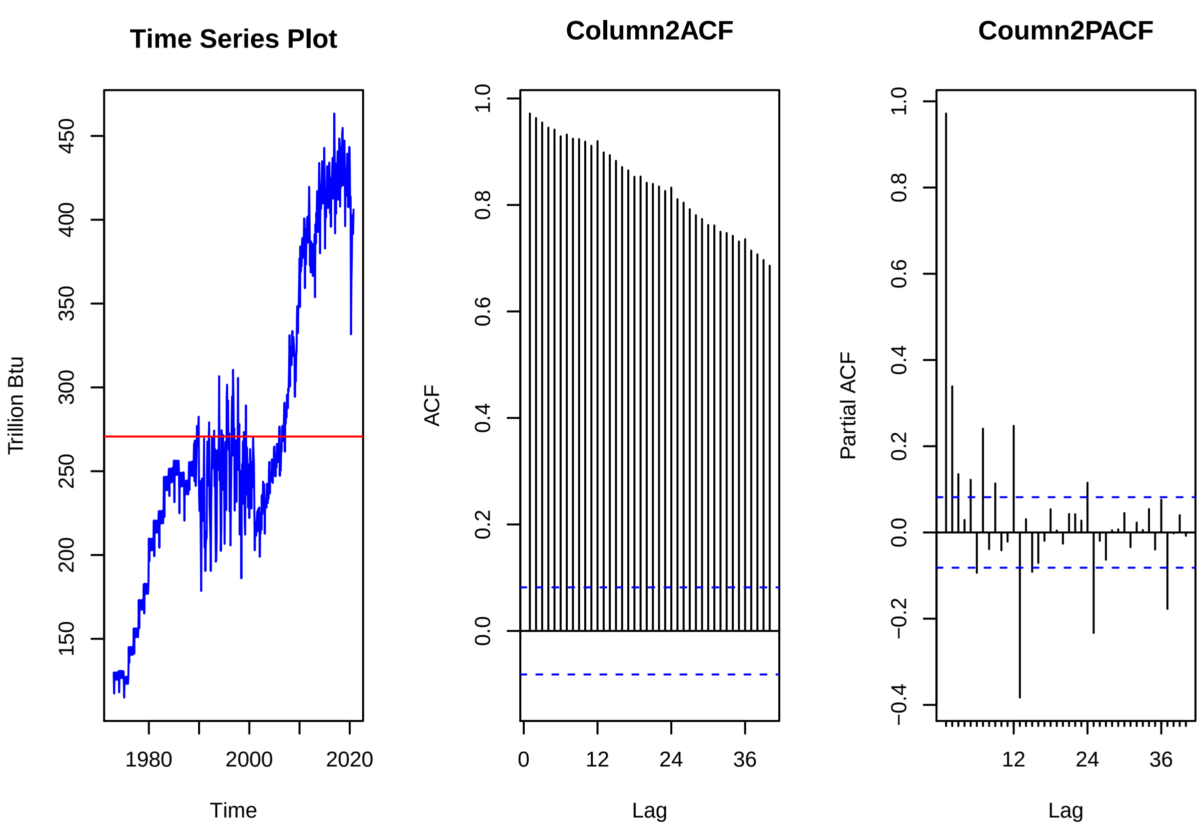
```
## [1] "1973-01-01" "1973-02-01" "1973-03-01" "1973-04-01" "1973-05-01"
## [6] "1973-06-01"
```

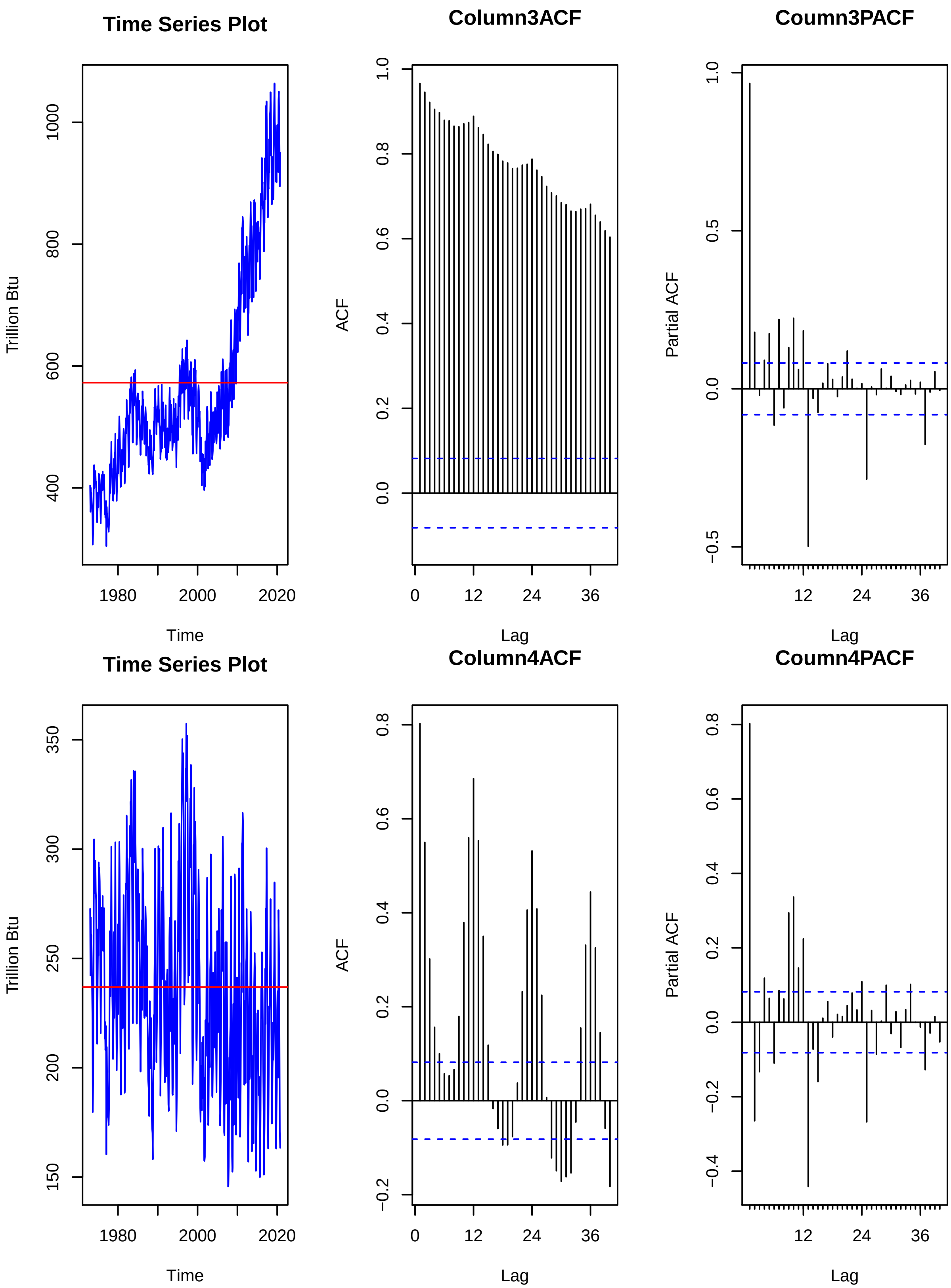
```
#add that to inflow_data and store in a new data frame
Renewable_altered.df <- cbind(my_date,Renewable_altered.df[,1:3])
head(Renewable_altered.df)
```

```
##       my_date Total Biomass Energy Production Total Renewable Energy Production
## 12 1973-01-01                         129.787                           403.981
## 13 1973-02-01                         117.338                           360.900
```

```
## 14 1973-03-01                         129.938                           400.161
## 15 1973-04-01                         125.636                           380.470
## 16 1973-05-01                         129.834                           392.141
## 17 1973-06-01                         125.611                           377.232
##    Hydroelectric Power Consumption
## 12                         272.703
## 13                         242.199
## 14                         268.810
## 15                         253.185
## 16                         260.770
## 17                         249.859
```

```
ncolumns <- ncol(Renewable_altered.df)

nmonths <- nrow(Renewable_altered.df)

#Renewable_altered.df<-lapply(Renewable_altered.df, as.numeric)
Renewable_ts.df<-ts(Renewable_altered.df[,2:4], start=c(1973, 1), end=c(2020,10), frequency=12)
Renewable_ts.df<-cbind(my_date,Renewable_ts.df)

#View(Renewable_ts.df)
for(i in 2:4){
  par(mfrow=c(1,3))  #place plot side by side
  plot(Renewable_ts.df[,i], ylab="Trillion Btu",  col=c("blue"))+abline(h=mean(Renewable_ts.df[,i]), col
  Acf(Renewable_ts.df[,i],lag.max=40,main=paste("Column",i,"ACF",sep=""))
  # because I am not storing Acf() into any object, I don't need to specify plot=TRUE
  Pacf(Renewable_ts.df[,i],lag.max=40,main=paste("Coumn",i,"PACF",sep=""))
}
```





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?

It appears that Total Biomass Energy Production, Total Renewable Energy Production, and Hydroelectric

Power Production each have a trend (a long term tendency), although the trend is less apparent in the Hydroelectric Power Consumption series. These trends seem to be deterministic.

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.

```
#Fit a linear trend to TS of Table 10
t <- c(1:nmonths)
linear_trend_model1=lm(Renewable_ts.df[,2]~t)  #left of tilda is dependent, to the right is independent
summary(linear_trend_model1)
```

```
##
## Call:
## lm(formula = Renewable_ts.df[, 2] ~ 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
```

```
linear_trend_model2=lm(Renewable_ts.df[,3]~t)
summary(linear_trend_model2)
```

```
##
## Call:
## lm(formula = Renewable_ts.df[, 3] ~ 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
```

```r
linear_trend_model3=lm(Renewable_ts.df[,4]~t)
summary(linear_trend_model3)
```

```
##
## Call:
## lm(formula = Renewable_ts.df[, 4] ~ 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
```

```r
trend1beta0=as.numeric(linear_trend_model1$coefficients[1])  #first coefficient is the intercept term o
trend1beta1=as.numeric(linear_trend_model1$coefficients[2])  #second coefficient is the slope or beta1

trend2beta0=as.numeric(linear_trend_model2$coefficients[1])  #first coefficient is the intercept term o
trend2beta1=as.numeric(linear_trend_model2$coefficients[2])  #second coefficient is the slope or beta1

trend3beta0=as.numeric(linear_trend_model3$coefficients[1])  #first coefficient is the intercept term o
trend3beta1=as.numeric(linear_trend_model3$coefficients[2])  #second coefficient is the slope or beta1
```

The first linear trend for the plot of Total Biomass Energy Production has an intercept of 135.5249801 which denotes the starting level of energy production in 1973 (in trillion BTU) and a slope of 0.4701605 which is statistically significant with a t-statistic of 47.33. From these values we estimate that for each passing month, Total Biomass Energy Production increases by 4.702e-01 Trillion BTU.

The second linear trend for the plot of Total Renewable Energy Production has an intercept of 330.3715564 which denotes the starting level of energy production (in trillion BTU) in 1973 and a slope of 0.8429932 which is statistically significant with a t-statistic of 35.58. From these values we estimate that for each passing month, Total Renewable Energy Production increases by 0.84299 Trillion BTU.

The third linear trend for the plot of Hydroelectric Power Consumption has an intercept of 258.0562181 which denotes the starting level of consumption in 1973 (in trillion BTU) and a slope of -0.0734076 which is statistically significant with a t-statistic of -6.904. From these values we estimate that for each passing month, Hydroelectric Power Consumption decreases by -0.07341 Trillion BTU.

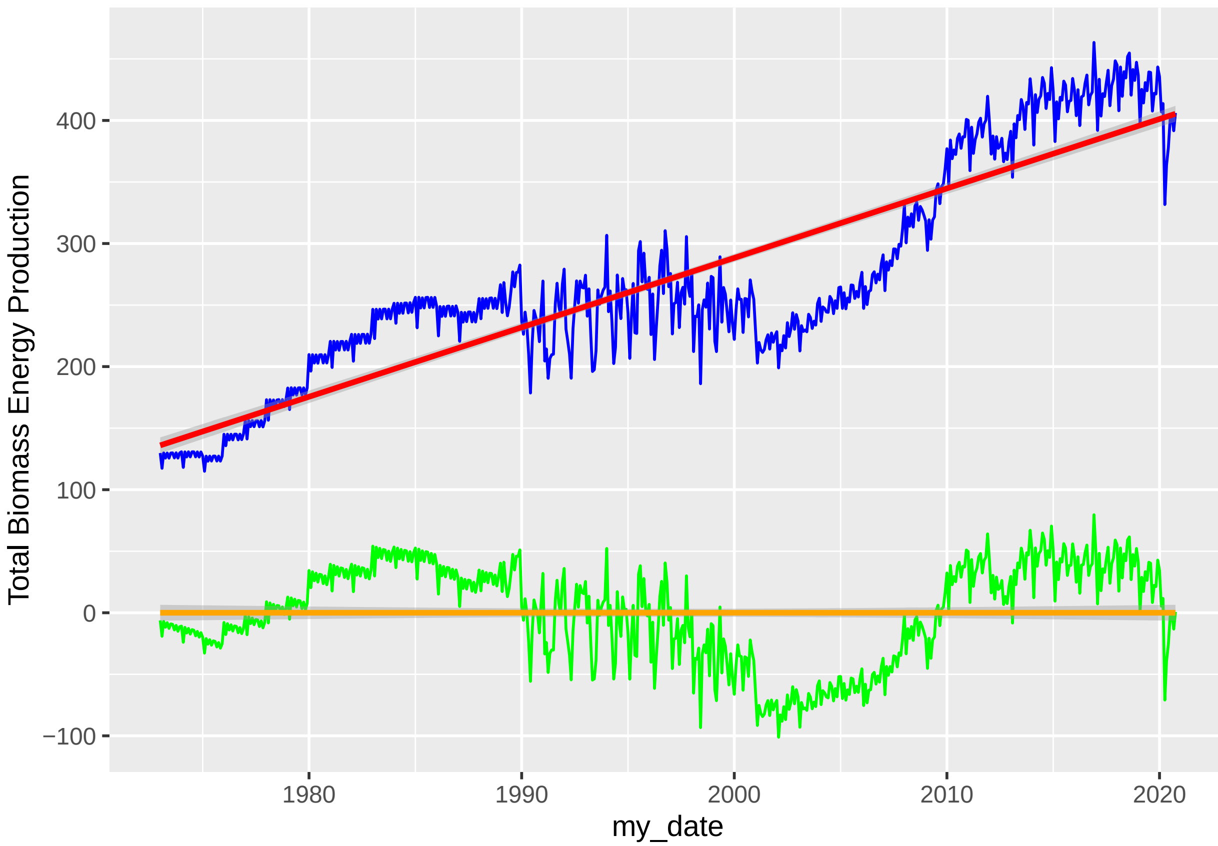
Q4

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
t <- c(1:nmonths)
detrend_renewable_biomass <- Renewable_altered.df[,2]-(trend1beta0+trend1beta1*t)
detrend_renewable_energy <- Renewable_altered.df[,3]-(trend2beta0+trend2beta1*t)
detrend_renewable_hydro <- Renewable_altered.df[,4]-(trend3beta0+trend3beta1*t)
```

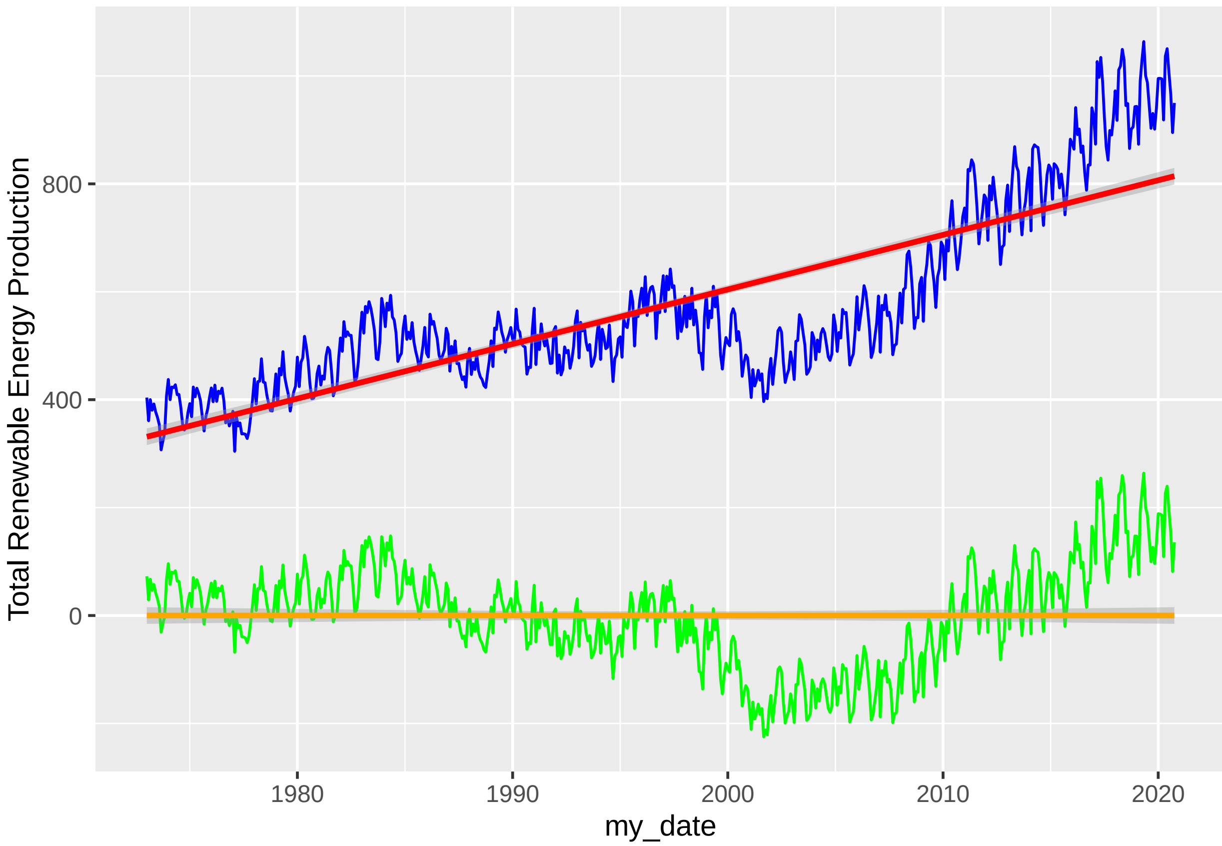
```r
ggplot(Renewable_altered.df, aes(x=my_date, y=Renewable_altered.df[,2])) +
          geom_line(color="blue") +
          ylab(paste0(colnames(Renewable_altered.df)[(2)],sep="")) +
          #geom_abline(intercept = beta0, slope = beta1, color="red") +
          geom_smooth(color="red",method="lm") +
          geom_line(aes(y=detrend_renewable_biomass), col="green")+
          geom_smooth(aes(y=detrend_renewable_biomass),color="orange",method="lm")
```

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



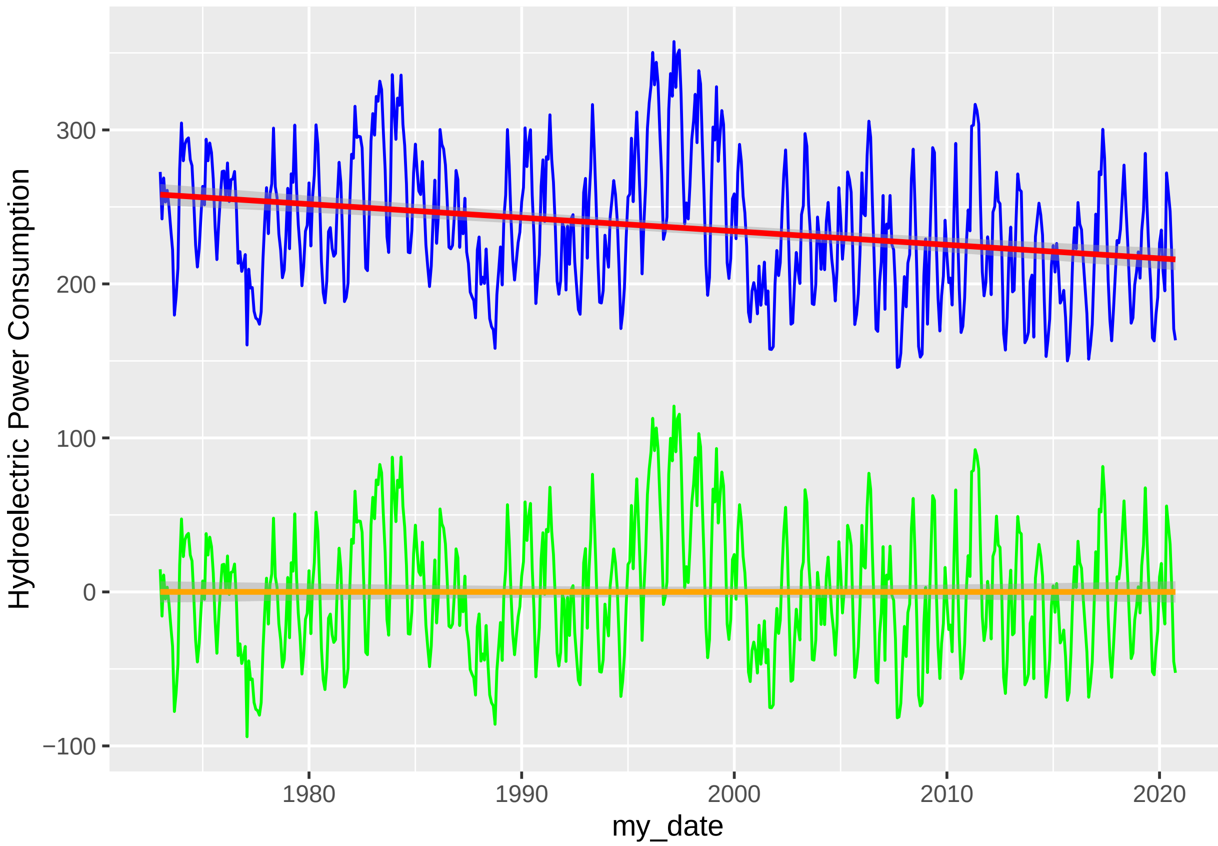

```r
ggplot(Renewable_altered.df, aes(x=my_date, y=Renewable_altered.df[,3])) +
          geom_line(color="blue") +
          ylab(paste0(colnames(Renewable_altered.df)[(3)],sep="")) +
          #geom_abline(intercept = beta0, slope = beta1, color="red")
          geom_smooth(color="red",method="lm") +
          geom_line(aes(y=detrend_renewable_energy), col="green")+
          geom_smooth(aes(y=detrend_renewable_energy),color="orange",method="lm")
```

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




```
ggplot(Renewable_altered.df, aes(x=my_date, y=Renewable_altered.df[,4])) +
            geom_line(color="blue") +
            ylab(paste0(colnames(Renewable_altered.df)[(4)],sep="")) +
            #geom_abline(intercept = beta0, slope = beta1, color="red")
            geom_smooth(color="red",method="lm") +
            geom_line(aes(y=detrend_renewable_hydro), col="green")+
            geom_smooth(aes(y=detrend_renewable_hydro),color="orange",method="lm")
```

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

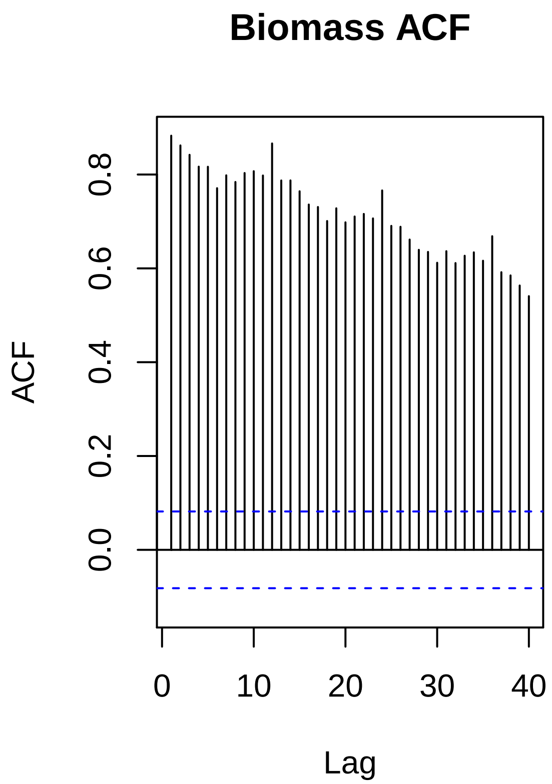


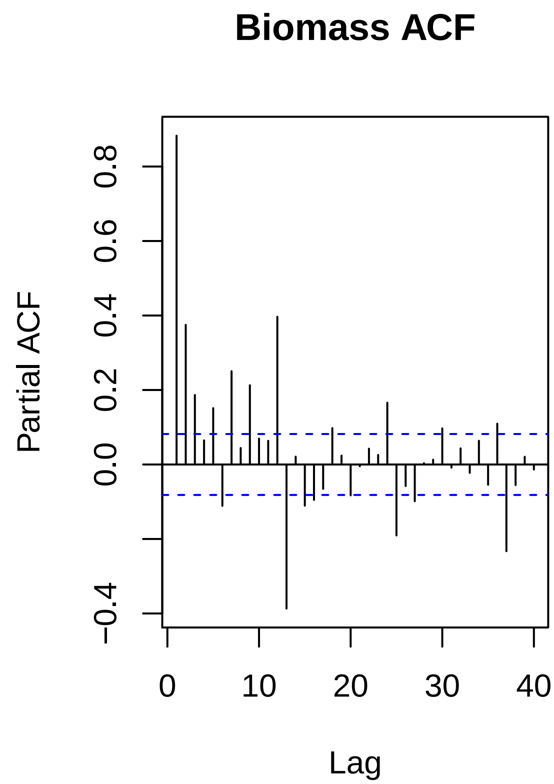
The plots in Q3 are significantly altered compared to the Q1 plots. The slope of the linear trend in Q3 is 0 for all 3 plots and so are all 3 intercepts. Now, the data varies around y=0 as opposed to in Q1 where it varied around 3 different y=mx+b linear trends.

Q5

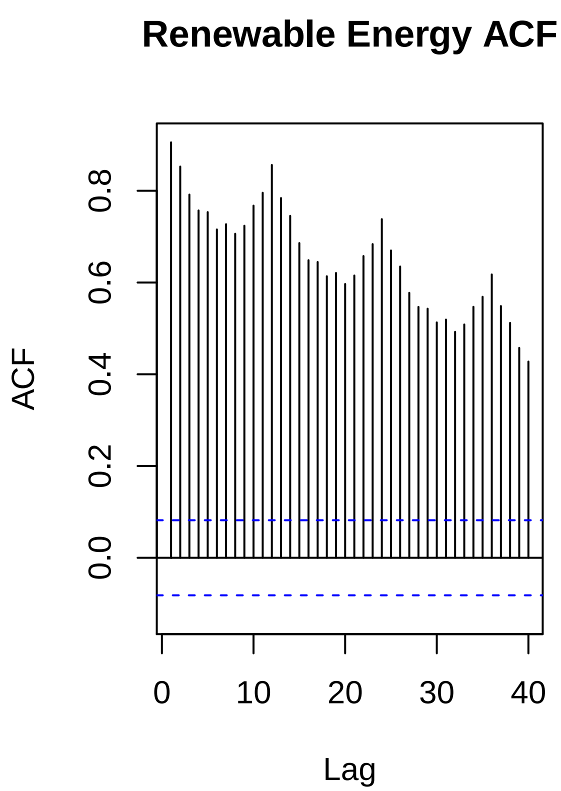
Plot ACF and PACF for the detrended series and compare with the plots from Q1. Did the plots change? How?

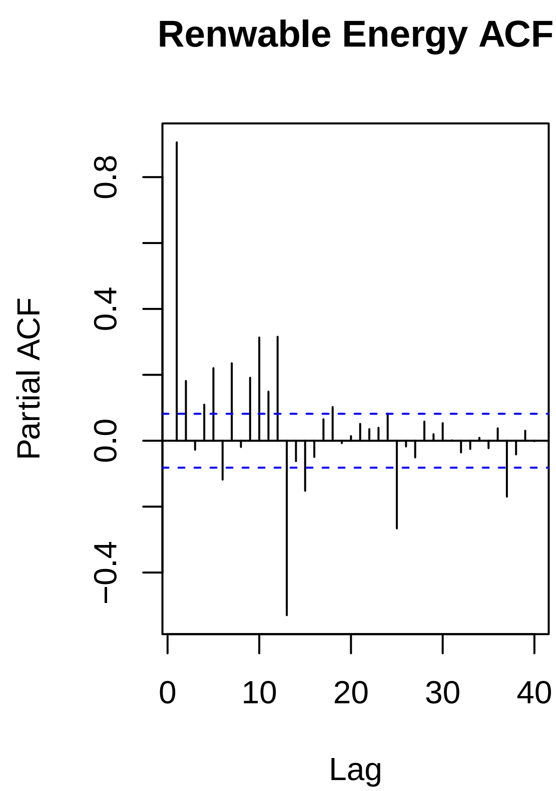
```
par(mfrow=c(1,2))  #place plot side by side
Acf(detrend_renewable_biomass,lag.max=40,main=paste("Biomass ACF",sep=""))
# because I am not storing Acf() into any object, I don't need to specify plot=TRUE
Pacf(detrend_renewable_biomass,lag.max=40,main=paste("Biomass ACF",sep=""))
```



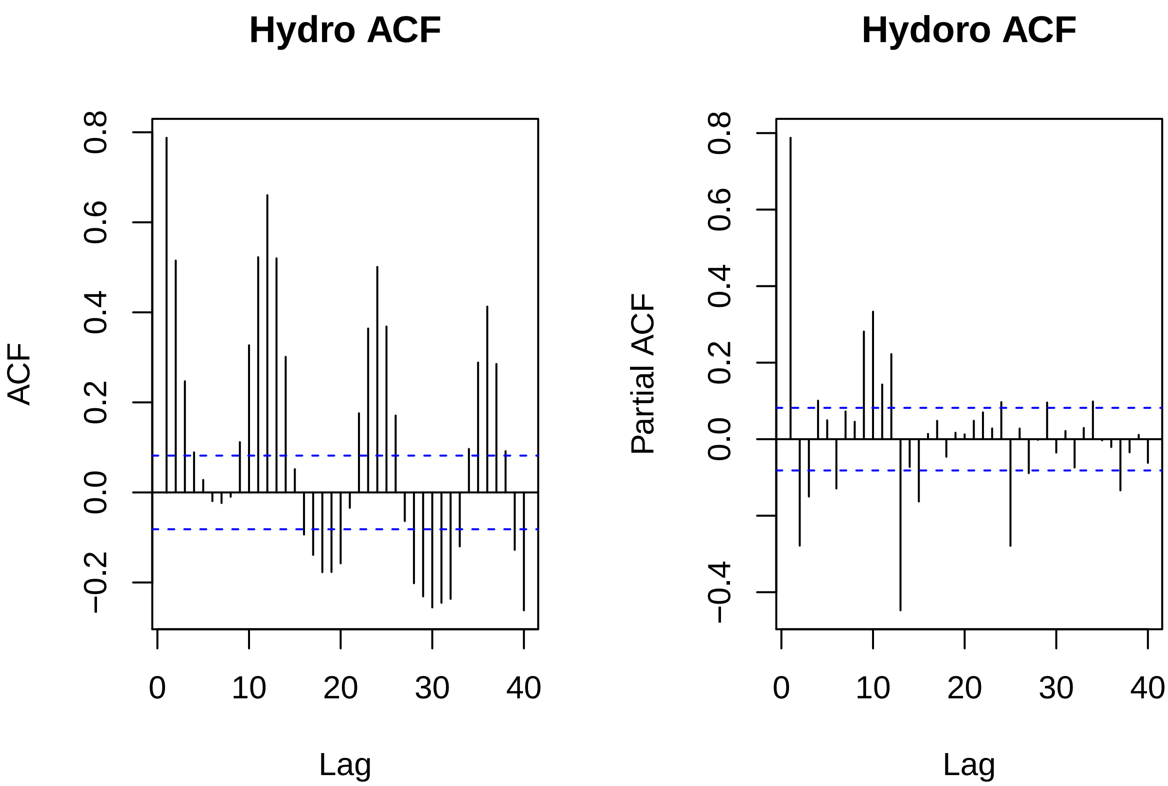



```
par(mfrow=c(1,2))  #place plot side by side
Acf(detrend_renewable_energy,lag.max=40,main=paste("Renewable Energy ACF",sep=""))
# because I am not storing Acf() into any object, I don't need to specify plot=TRUE
Pacf(detrend_renewable_energy,lag.max=40,main=paste("Renwable Energy ACF",sep=""))
```





```r
par(mfrow=c(1,2))  #place plot side by side
Acf(detrend_renewable_hydro,lag.max=40,main=paste("Hydro ACF",sep=""))
# because I am not storing Acf() into any object, I don't need to specify plot=TRUE
Pacf(detrend_renewable_hydro,lag.max=40,main=paste("Hydoro ACF",sep=""))
```



The ACF and PACF seems to have changed somewhat slightly after detrending, with the main difference being magnitude. The ACF and PCF of the Total Biomass Produced and Total Renewable Energy Production decreased somewhat in magnitude across most of the plot but the shapes are similar, as is the shape of Total Hydroelectric Power Consumed's ACF. However, both the ACF and PACF of Total Hydroelectric Power Consumption look very similar even after detrending.

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.

```r
dummies<-seasonaldummy(Renewable_ts.df[,1])
seas_means_model_biomass=lm(Renewable_ts.df[,2]~dummies)
seas_means_model_energy=lm(Renewable_ts.df[,3]~dummies)
seas_means_model_hydro=lm(Renewable_ts.df[,4]~dummies)
summary(seas_means_model_biomass)
```

```
##
## Call:
## lm(formula = Renewable_ts.df[, 2] ~ dummies)
##
```

```
## 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 ***
## dummiesJan   -1.0039    18.0009  -0.056    0.956
## dummiesFeb  -29.3891    18.0009  -1.633    0.103
## dummiesMar   -8.6090    18.0009  -0.478    0.633
## dummiesApr  -20.5046    18.0009  -1.139    0.255
## dummiesMay  -14.0960    18.0009  -0.783    0.434
## dummiesJun  -19.5548    18.0009  -1.086    0.278
## dummiesJul   -3.4306    18.0009  -0.191    0.849
## dummiesAug    0.2220    18.0009   0.012    0.990
## dummiesSep  -11.9821    18.0009  -0.666    0.506
## dummiesOct   -0.5379    18.0009  -0.030    0.976
## dummiesNov   -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
summary(seas_means_model_energy)
```

```
##
## Call:
## lm(formula = Renewable_ts.df[, 3] ~ dummies)
##
## 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 ***
## dummiesJan    12.451     34.335   0.363   0.7170
## dummiesFeb   -38.964     34.335  -1.135   0.2569
## dummiesMar    20.515     34.335   0.597   0.5504
## dummiesApr     8.294     34.335   0.242   0.8092
## dummiesMay    36.628     34.335   1.067   0.2865
## dummiesJun    19.560     34.335   0.570   0.5691
## dummiesJul     8.863     34.335   0.258   0.7964
## dummiesAug   -18.480     34.335  -0.538   0.5906
## dummiesSep   -62.410     34.335  -1.818   0.0696 .
## dummiesOct   -42.649     34.335  -1.242   0.2147
## dummiesNov   -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
summary(seas_means_model_hydro)
```

```
##
## Call:
## lm(formula = Renewable_ts.df[, 4] ~ dummies)
##
## 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 ***
## dummiesJan    13.270      6.841   1.940  0.05291 .
## dummiesFeb    -8.133      6.841  -1.189  0.23499
## dummiesMar    20.442      6.841   2.988  0.00293 **
## dummiesApr    17.199      6.841   2.514  0.01221 *
## dummiesMay    40.726      6.841   5.953 4.64e-09 ***
## dummiesJun    31.764      6.841   4.643 4.28e-06 ***
## dummiesJul    10.858      6.841   1.587  0.11306
## dummiesAug   -17.907      6.841  -2.618  0.00909 **
## dummiesSep   -50.121      6.841  -7.326 8.26e-13 ***
## dummiesOct   -49.165      6.841  -7.187 2.12e-12 ***
## dummiesNov   -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_biomass=seas_means_model_biomass$coefficients[1]
beta_coeff_biomass=seas_means_model_biomass$coefficients[2:12]
beta_int_energy=seas_means_model_energy$coefficients[1]
beta_coeff_energy=seas_means_model_energy$coefficients[2:12]
beta_int_hydro=seas_means_model_hydro$coefficients[1]
beta_coeff_hydro=seas_means_model_hydro$coefficients[2:12]
```

After running these three seasonal regressions, it seems that my interpretation of the detrended PACFs was correct. Both the Total Biomass Energy Production and Total Renewable Energy Production seasonal regressions show insignificant seasonal dummies pointing toward no seasonality, however some of the Hydroelectric Power Consumption seasonal dummies are significant indicating a seasonal trend.

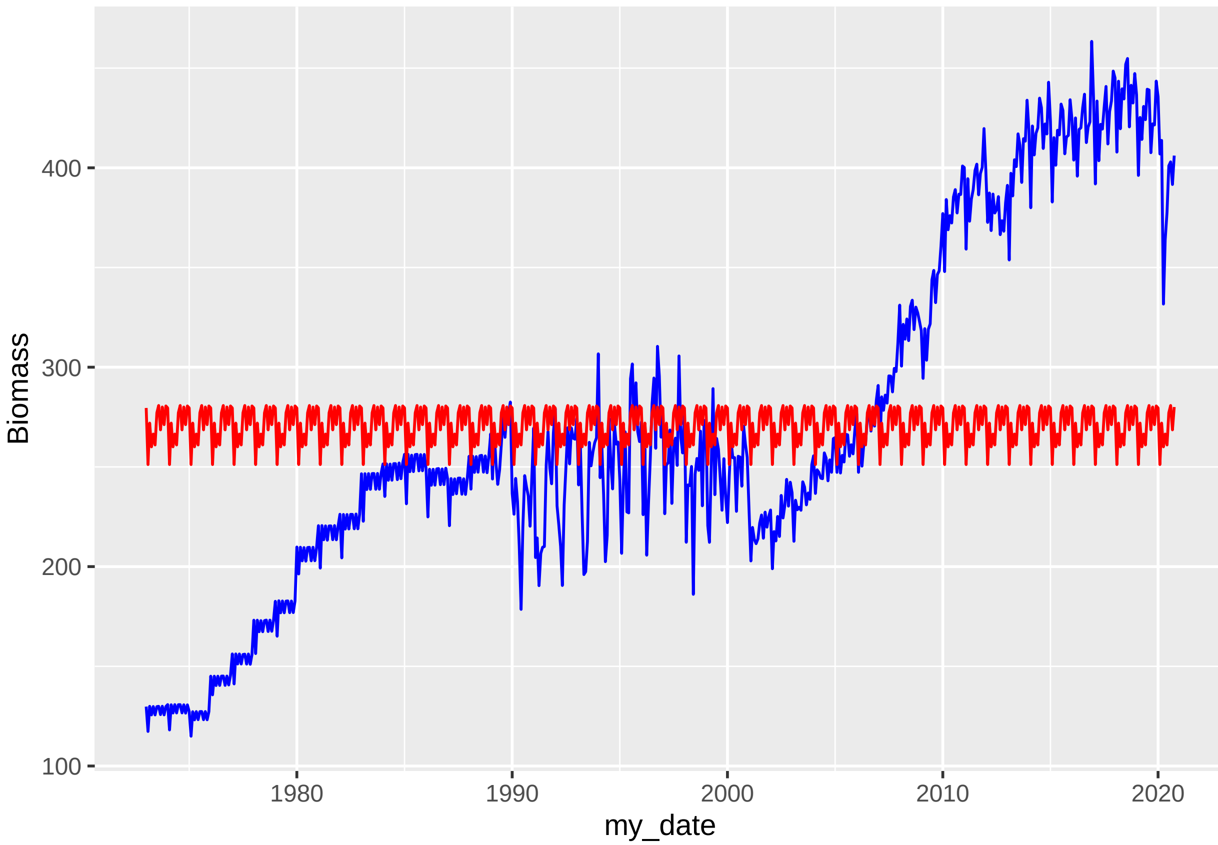
The intercepts for the 3 series again show where each series begins prior to estimating the seasonal trends. Total Biomass Energy Production starts at 280.5692553 Trillion BTU, Total Renewable Energy Production starts at 580.9124681 Trillion BTU, and Hydroelectric Power Consumption starts at 238.8868085 Trillion BTU in 1973.

Q7

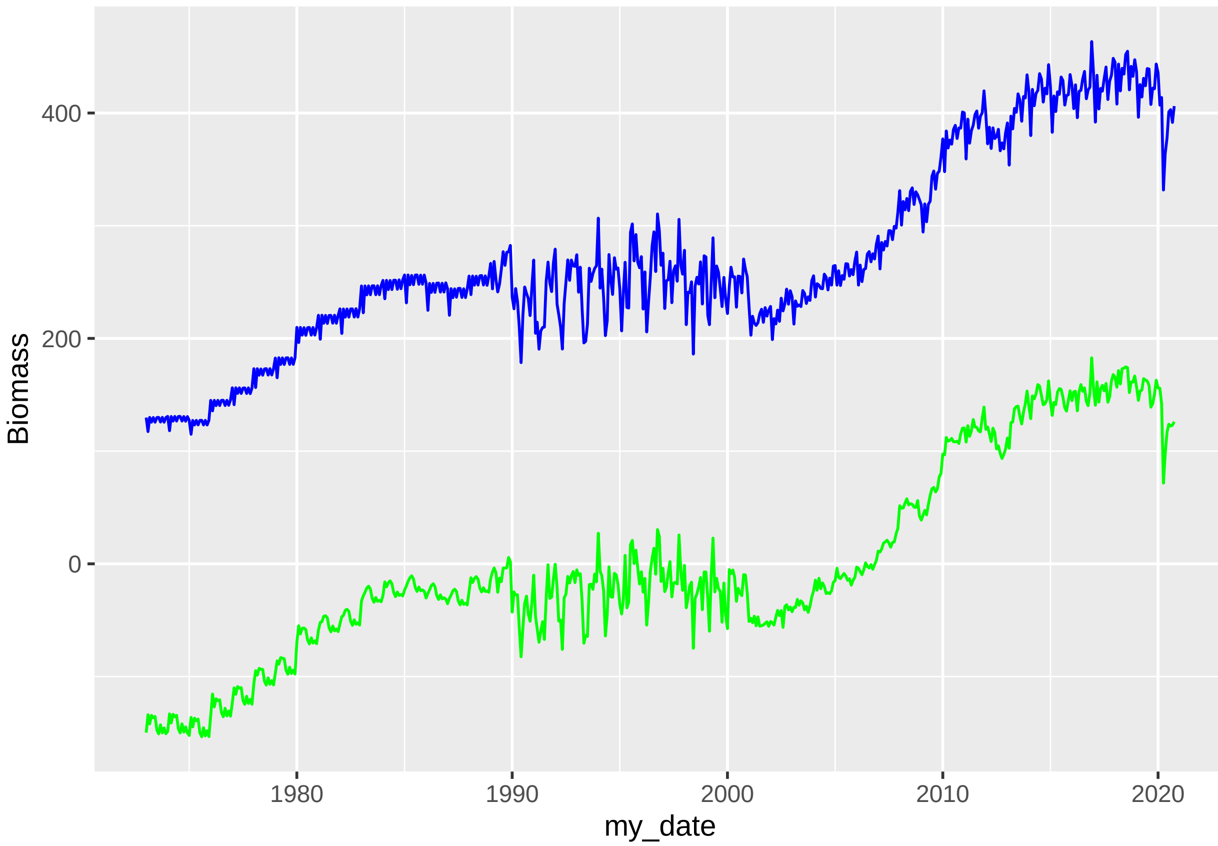
Use the regression coefficients from Q6 to deseason the series. Plot the deseason series and compare with the plots from part Q1. Did anything change?

```r
biomass_seas_comp=array(0,nmonths)
for(i in 1:nmonths){
  biomass_seas_comp[i]=(beta_int_biomass+beta_coeff_biomass%*%dummies[i,])
```

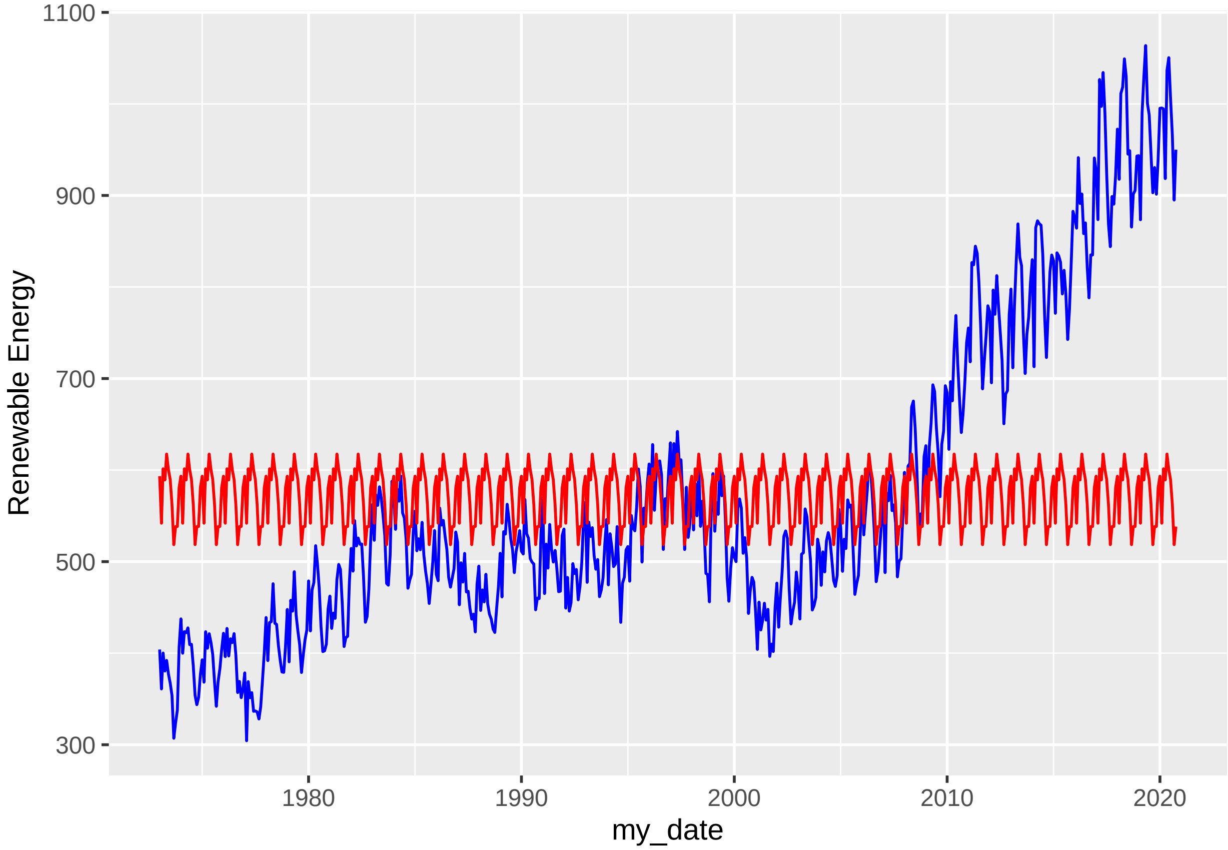
```
}

ggplot(Renewable_altered.df, aes(x=my_date, y=Renewable_altered.df[,2])) +
            geom_line(color="blue") +
            ylab(paste0("Biomass")) +
            geom_line(aes(y=biomass_seas_comp), col="red")
```



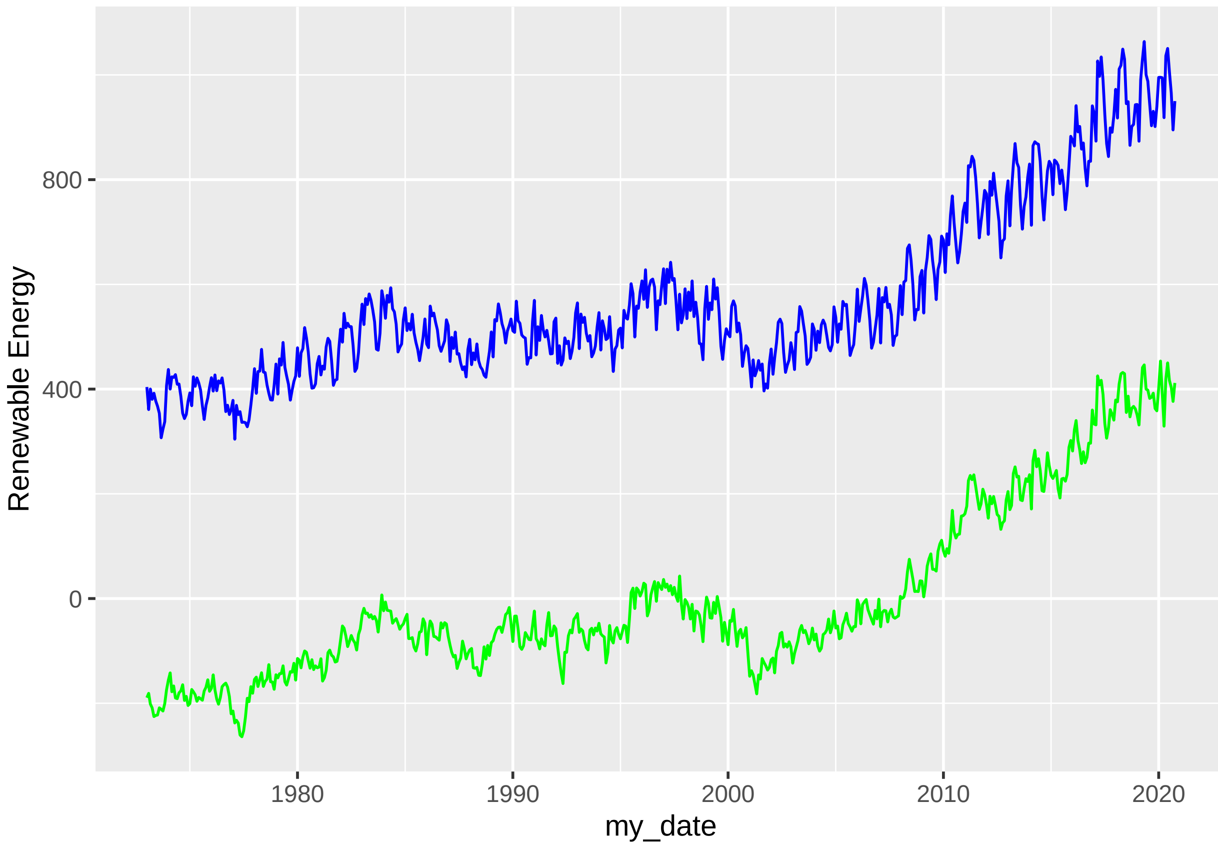

```
deseason_renewable_biomass <- Renewable_altered.df[,2]-biomass_seas_comp

ggplot(Renewable_altered.df, aes(x=my_date, y=Renewable_altered.df[,2])) +
            geom_line(color="blue") +
            ylab(paste0("Biomass")) +
            geom_line(aes(y=deseason_renewable_biomass), col="green")
```



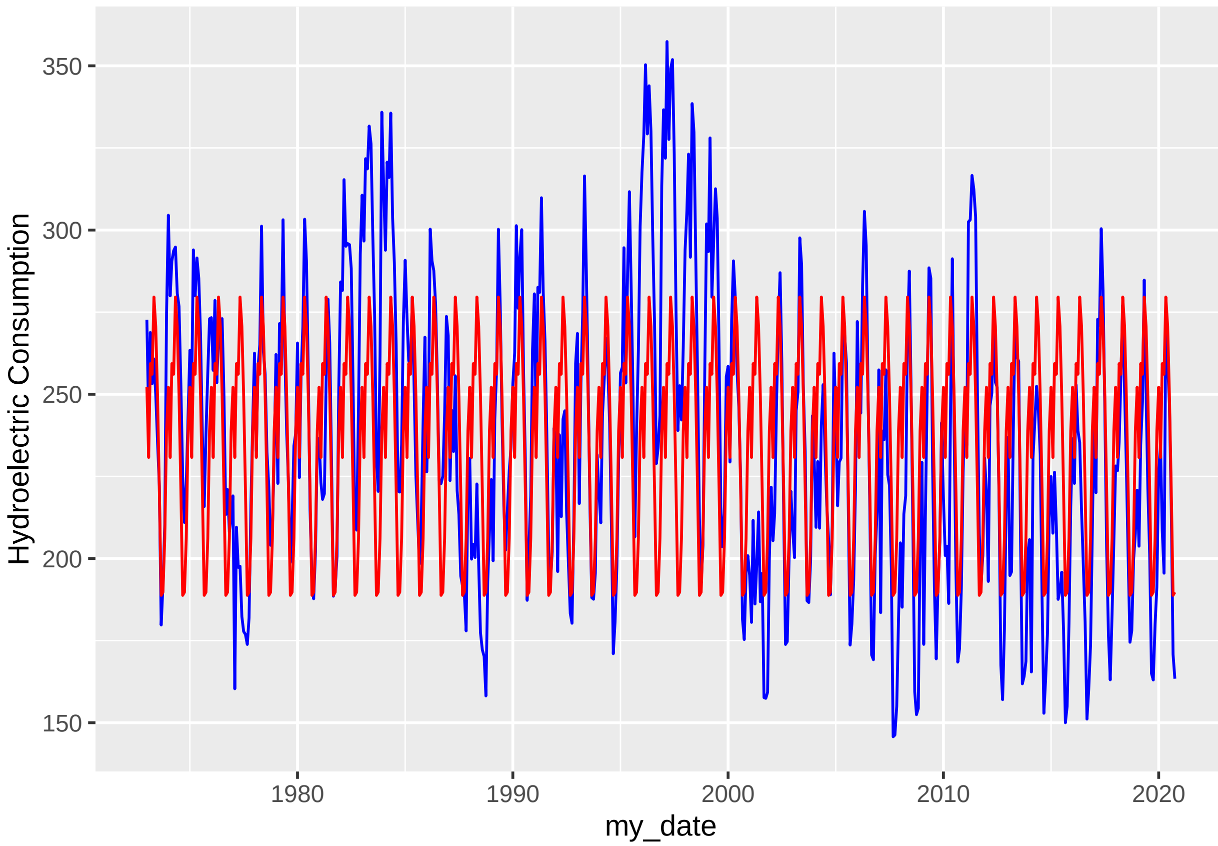

```
energy_seas_comp=array(0,nmonths)
for(i in 1:nmonths){
  energy_seas_comp[i]=(beta_int_energy+beta_coeff_energy%*%dummies[i,])

}

ggplot(Renewable_altered.df, aes(x=my_date, y=Renewable_altered.df[,3])) +
            geom_line(color="blue") +
            ylab(paste0("Renewable Energy")) +
            geom_line(aes(y=energy_seas_comp), col="red")
```



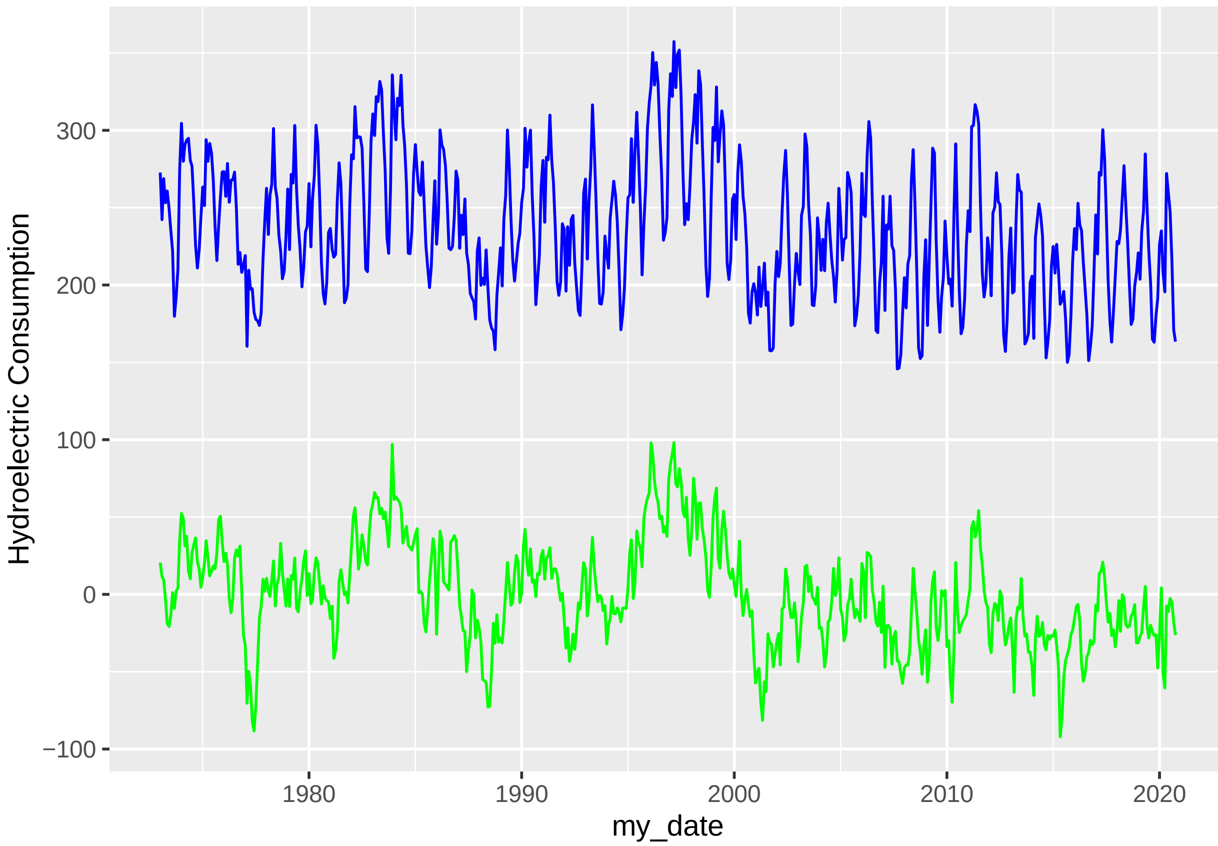

```
deseason_renewable_energy <- Renewable_altered.df[,3]-energy_seas_comp

ggplot(Renewable_altered.df, aes(x=my_date, y=Renewable_altered.df[,3])) +
            geom_line(color="blue") +
            ylab(paste0("Renewable Energy")) +
            geom_line(aes(y=deseason_renewable_energy), col="green")
```




```
hydro_seas_comp=array(0,nmonths)
for(i in 1:nmonths){
  hydro_seas_comp[i]=(beta_int_hydro+beta_coeff_hydro%*%dummies[i,])

}

ggplot(Renewable_altered.df, aes(x=my_date, y=Renewable_altered.df[,4])) +
            geom_line(color="blue") +
            ylab(paste0("Hydroelectric Consumption")) +
            geom_line(aes(y=hydro_seas_comp), col="red")
```




```
deseason_renewable_hydro <- Renewable_altered.df[,4]-hydro_seas_comp

ggplot(Renewable_altered.df, aes(x=my_date, y=Renewable_altered.df[,4])) +
           geom_line(color="blue") +
           ylab(paste0("Hydroelectric Consumption")) +
           geom_line(aes(y=deseason_renewable_hydro), col="green")
```

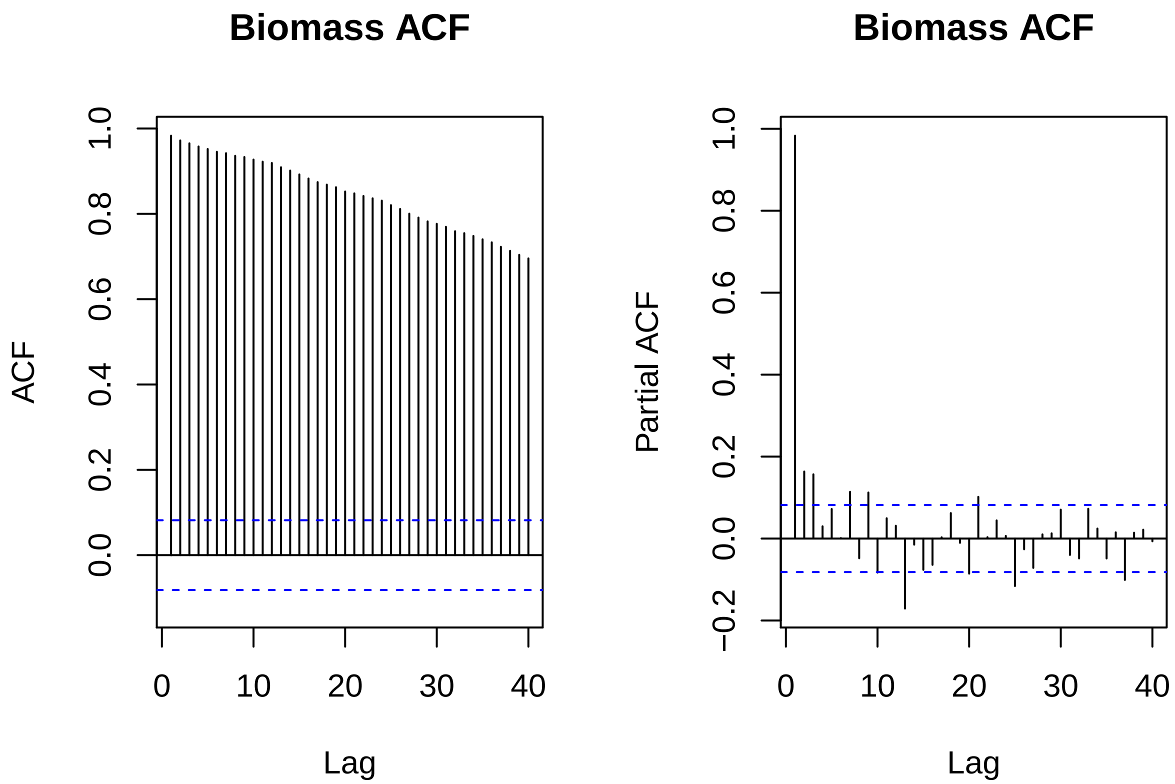


After deseasoning the series and comparing these plots to Q1, all 3 plots have obviously changed. The first two plots, Total Biomass Produced and Total Renewable Energy Produced had their intercepts dropped down to 0, but the overall shape of their plots did not change very much. This is likely because they are not seasonally varying series (as we concluded in Q6). However, the Hydroelectric Consumption series appeared to not only have its intercept drop to 0, but it also changed shape somewhat after subtracting the calculated seasonal variation. This too fits with out conclusion in Q6, which stipulates that the Hydroelectric Consumption series is seasonally variant. If the deseasoned series is plotted next to the unaltered series, there should be a significant reduction in variation over the course of time.

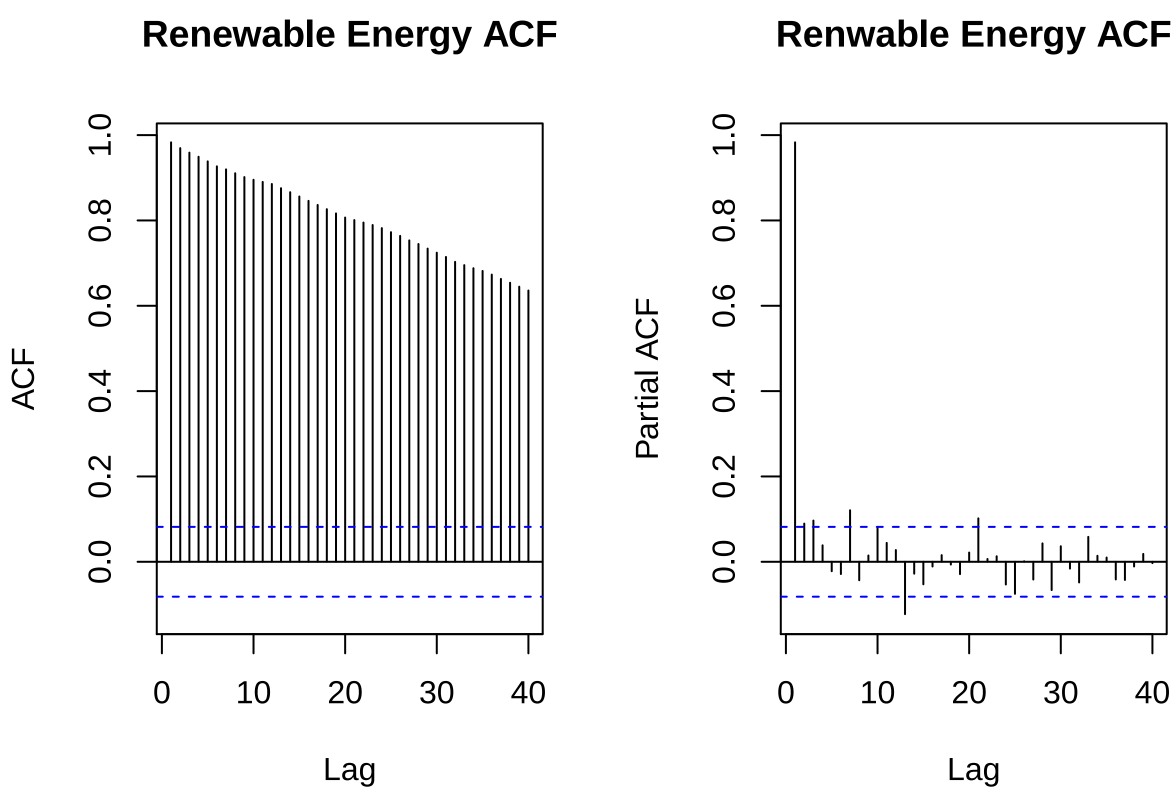
Q8

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

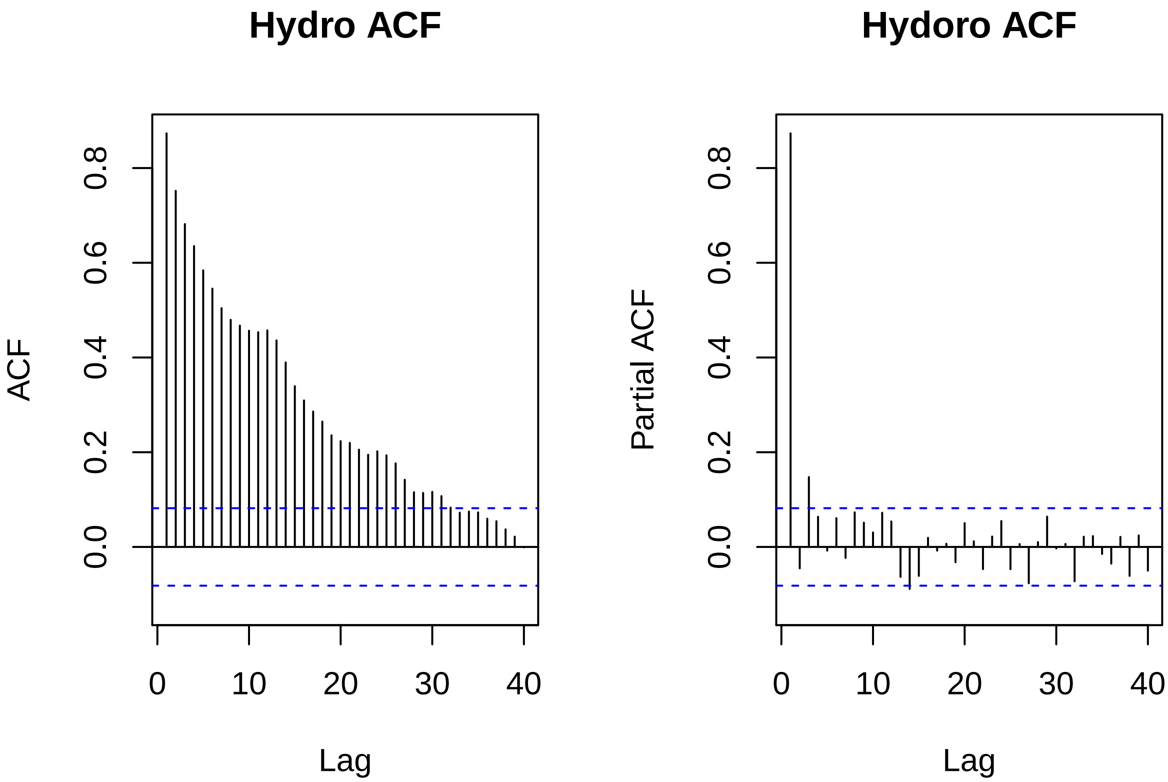
```
par(mfrow=c(1,2))  #place plot side by side
Acf(deseason_renewable_biomass,lag.max=40,main=paste("Biomass ACF",sep=""))
# because I am not storing Acf() into any object, I don't need to specify plot=TRUE
Pacf(deseason_renewable_biomass,lag.max=40,main=paste("Biomass ACF",sep=""))
```




```r
par(mfrow=c(1,2))  #place plot side by side
Acf(deseason_renewable_energy,lag.max=40,main=paste("Renewable Energy ACF",sep=""))
# because I am not storing Acf() into any object, I don't need to specify plot=TRUE
Pacf(deseason_renewable_energy,lag.max=40,main=paste("Renwable Energy ACF",sep=""))
```



```r
par(mfrow=c(1,2))  #place plot side by side
Acf(deseason_renewable_hydro,lag.max=40,main=paste("Hydro ACF",sep=""))
# because I am not storing Acf() into any object, I don't need to specify plot=TRUE
Pacf(deseason_renewable_hydro,lag.max=40,main=paste("Hydoro ACF",sep=""))
```



The ACFs of Total Biomass Produced and Total Renewable Energy Produced appear similar, however their PACFs appear far less seasonal than before (it's mostly 0). The ACF of Hydroelectric Power Consumed has significantly changed its shape after deseasoning, moving from an obvious seasonal variation to a gradual decrease in autocorrelation as lag increases. PACF of Hydroelectric Power Consumed has also gone from perhaps a weak seasonal variation to almost statistically zero after deseasoning.