ENV 790.30 - Time Series Analysis for Energy Data | Spring 2022

Assignment 3 - Due date 02/08/22

Abbhijith Hari Gopal

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_A03_Sp22.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_Consumpti
The data comes from the US Energy Information and Administration and corresponds to the January 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.\

```
#Loalibrary(lubridate)
library(ggplot2)
```

```
## Warning: package 'ggplot2' was built under R version 4.0.5
```

```
library(forecast)
```

```
## Warning: package 'forecast' was built under R version 4.0.5
```

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

```
library(Kendall)
```

```
## Warning: package 'Kendall' was built under R version 4.0.5
```

```
library(readxl)
```

```
## Warning: package 'readxl' was built under R version 4.0.5
```

```
library(tseries)
```

```
## Warning: package 'tseries' was built under R version 4.0.5
```

```
library(lubridate)
```

```
## Warning: package 'lubridate' was built under R version 4.0.5
```

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

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

```
#Importing data set
getwd()
```

```
## [1] "C:/Users/User/Documents/GithubRepos/ENV790_TimeSeriesAnalysis_Sp2022/Assignments"
```

```
data = read_excel("../Data/Table_10.1_Renewable_Energy_Production_and_Consumption_by_Source.xlsx", skip
```

```
## New names:
## * `` -> ...1
## * `` -> ...2
## * `` -> ...3
## * `` -> ...4
## * `` -> ...5
## * ...
```

```
data_of_interest = data[, 4:6]
date_of_interest = data[, 1]
#date_of_interest = parse_date_time(date_of_interest)

colnames(data_of_interest) = c("Total Biomass Energy Production", "Total Renewable Energy Production",
colnames(date_of_interest) = c("Date")
data_of_interest$`Total Biomass Energy Production` = as.numeric(data_of_interest$`Total Biomass Energy 
data_of_interest$`Total Renewable Energy Production` = as.numeric(data_of_interest$`Total Renewable Ene
data_of_interest$`Hydroelectric Power Consumption` = as.numeric(data_of_interest$`Hydroelectric Power C

new_data = cbind(date_of_interest, data_of_interest[, 1:3])
head(new_data)
```

```
##         Date 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
```

```
ncolumns = ncol(new_data)
nmonths = nrow(new_data)

ts_object = ts(data = new_data[, 2:4], start = c(1973, 1), frequency=12)
ts_object = cbind(date_of_interest, ts_object)
ts_object = ts(ts_object, start = c(1973, 1), frequency=12)

head(ts_object, 15)
```

```
##               Date Total Biomass Energy Production
## Jan 1973  94694400                          129.787
## Feb 1973  97372800                          117.338
## Mar 1973  99792000                          129.938
## Apr 1973 102470400                          125.636
## May 1973 105062400                          129.834
## Jun 1973 107740800                          125.611
## Jul 1973 110332800                          129.787
## Aug 1973 113011200                          129.918
## Sep 1973 115689600                          125.782
## Oct 1973 118281600                          129.970
## Nov 1973 120960000                          125.643
## Dec 1973 123552000                          129.824
## Jan 1974 126230400                          130.807
## Feb 1974 128908800                          118.091
## Mar 1974 131328000                          130.727
##          Total Renewable Energy Production Hydroelectric Power Consumption
## Jan 1973                           403.981                         272.703
## Feb 1973                           360.900                         242.199
## Mar 1973                           400.161                         268.810
## Apr 1973                           380.470                         253.185
## May 1973                           392.141                         260.770
## Jun 1973                           377.232                         249.859
## Jul 1973                           367.325                         235.670
## Aug 1973                           353.757                         222.077
## Sep 1973                           307.006                         179.733
## Oct 1973                           323.453                         191.723
## Nov 1973                           337.817                         210.285
## Dec 1973                           406.694                         274.435
## Jan 1974                           437.467                         304.506
```

```
## Feb 1974                          399.942                          279.950
## Mar 1974                          423.474                          290.582
```

```r
class(date_of_interest)
```

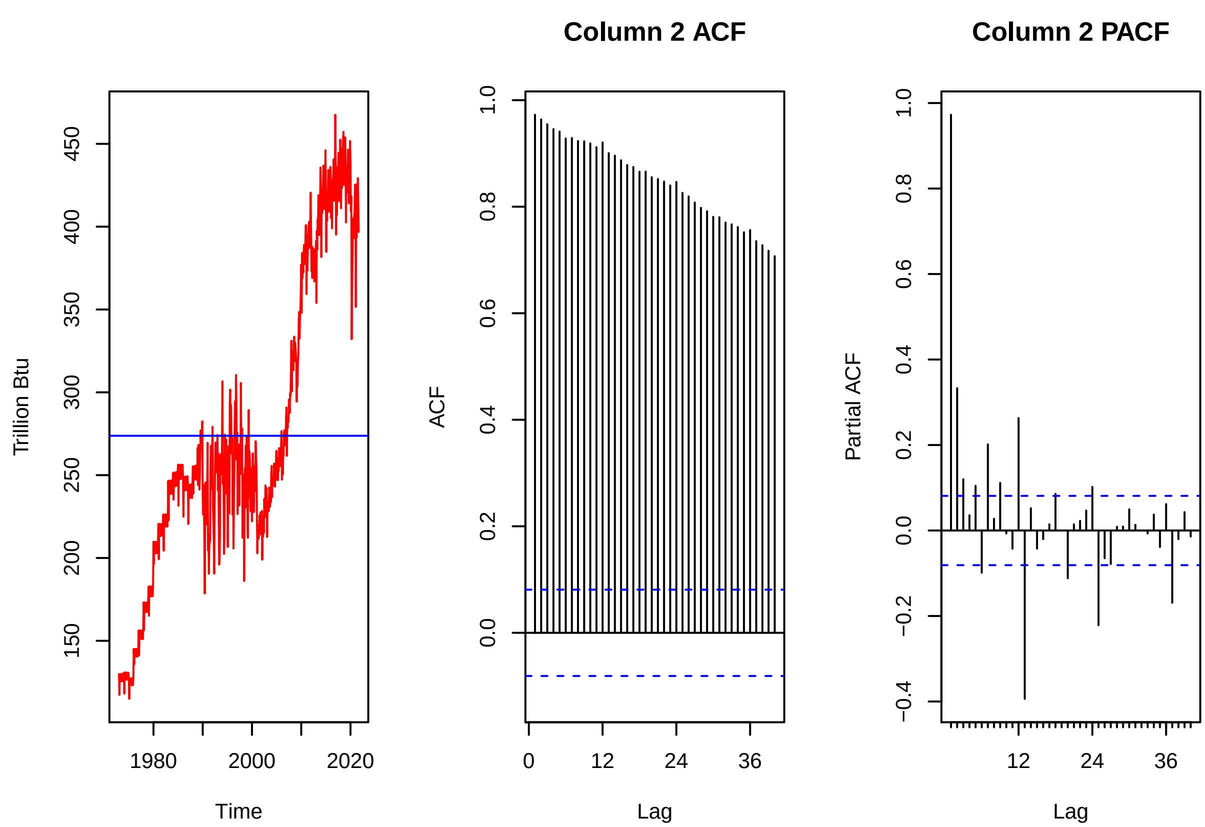
```
## [1] "tbl_df"     "tbl"        "data.frame"
```

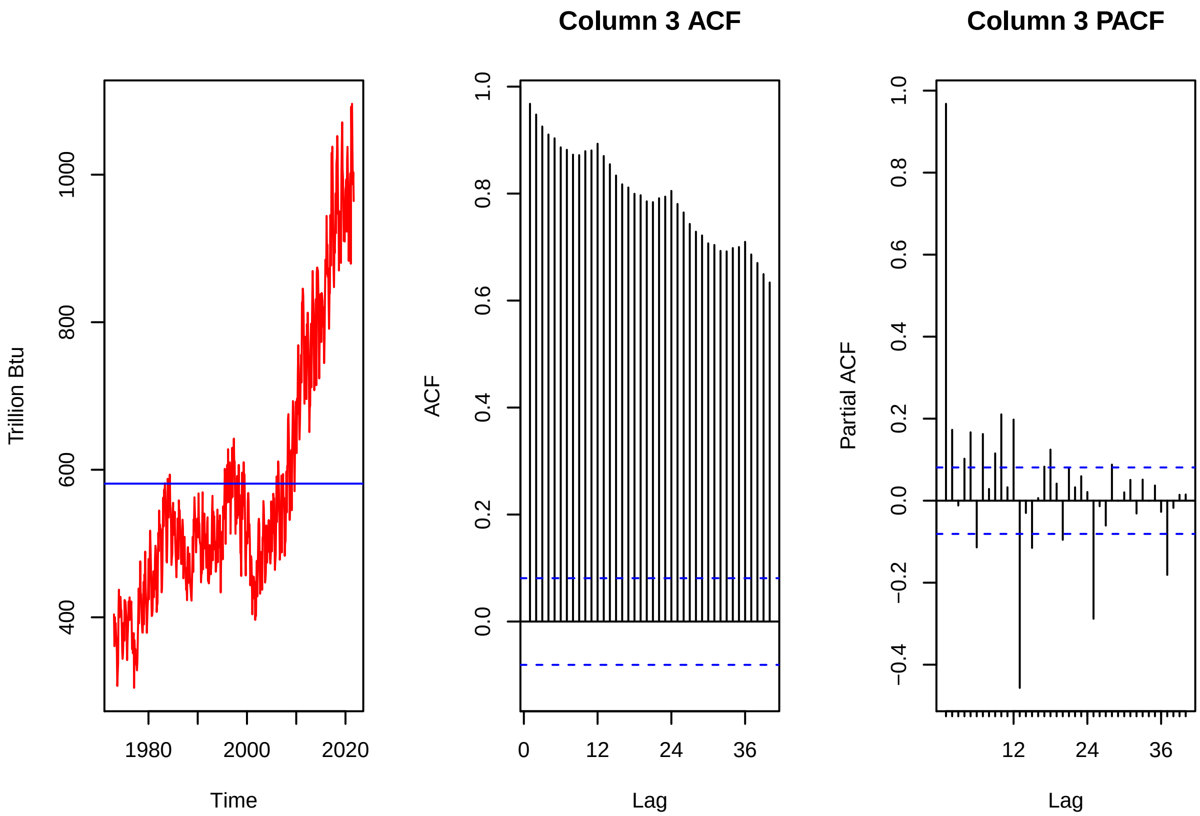
##Trend Component

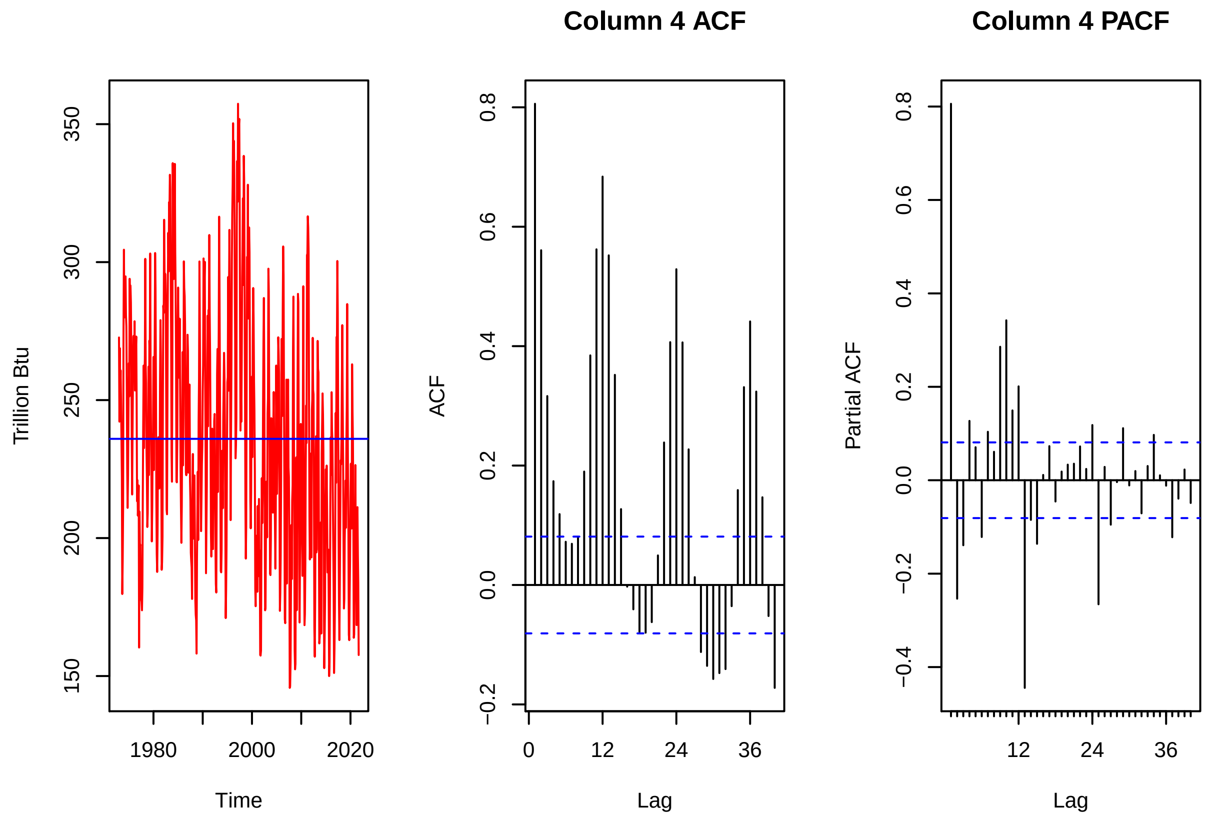
Q1

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: use par() function)

```r
for (i in 2:4)
{
  par(mfrow=c(1,3))
  plot(ts_object[, i], type="l",col="red",ylab="Trillion Btu" )
  abline(h=mean(ts_object[, i]), col="blue")
  Acf(ts_object[, i], lag.max=40,main=paste("Column",i,"ACF",sep=" "))
  Pacf(ts_object[, i], lag.max=40,main=paste("Column",i,"PACF",sep=" "))

}
```




Column 3 ACF
Column 3 PACF
Trillion Btu
Time
ACF
Lag
Partial ACF
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 and Total Renewable Energy Production seem to have a positive trend whereas, Hydroelectric Power Consumption doesn't appear to have a trend/the trend is not so apparent

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: nmonths)

linear_trend_model1=lm(ts_object[,2]~t)
summary(linear_trend_model1)
```

```
##
## Call:
## lm(formula = ts_object[, 2] ~ t)
##
```

```
## Residuals:
##      Min       1Q   Median       3Q      Max
## -101.892  -24.306    4.932   33.103   82.292
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 1.348e+02  3.282e+00   41.07   <2e-16 ***
## t           4.744e-01  9.705e-03   48.88   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 39.64 on 583 degrees of freedom
## Multiple R-squared:  0.8039, Adjusted R-squared:  0.8035
## F-statistic:  2389 on 1 and 583 DF,  p-value: < 2.2e-16
```

```
print(paste("The slope is", linear_trend_model1$coefficients[2], "and intercept is", linear_trend_model1
```

```
## [1] "The slope is 0.474382900448469 and intercept is 134.789734954923"
```

```
linear_trend1_beta0 = as.numeric(linear_trend_model1$coefficients[1])
linear_trend1_beta1 = as.numeric(linear_trend_model1$coefficients[2])

linear_trend_model2=lm(ts_object[,3]~t)
summary(linear_trend_model2)
```

```
##
## Call:
## lm(formula = ts_object[, 3] ~ t)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -230.488  -57.869    5.595   62.090  261.349
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 323.18243    8.02555   40.27   <2e-16 ***
## t             0.88051    0.02373   37.10   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 96.93 on 583 degrees of freedom
## Multiple R-squared:  0.7025, Adjusted R-squared:  0.702
## F-statistic:  1377 on 1 and 583 DF,  p-value: < 2.2e-16
```

```
print(paste("The slope is", linear_trend_model2$coefficients[2], "and intercept is", linear_trend_model2
```

```
## [1] "The slope is 0.880506471155194 and intercept is 323.182434720759"
```

```
linear_trend2_beta0 = as.numeric(linear_trend_model2$coefficients[1])
linear_trend2_beta1 = as.numeric(linear_trend_model2$coefficients[2])

linear_trend_model3=lm(ts_object[,4]~t)
summary(linear_trend_model3)
```

```
##
## Call:
## lm(formula = ts_object[, 4] ~ t)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -94.892 -31.300  -2.414  27.876 121.263
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 259.18303    3.47464  74.593  < 2e-16 ***
## t            -0.07924    0.01027  -7.712 5.36e-14 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 41.97 on 583 degrees of freedom
## Multiple R-squared:  0.09258,    Adjusted R-squared:  0.09103
## F-statistic: 59.48 on 1 and 583 DF,  p-value: 5.364e-14
```

```
print(paste("The slope is", linear_trend_model3$coefficients[2], "and intercept is", linear_trend_model3
```

```
## [1] "The slope is -0.0792415350689486 and intercept is 259.183029604262"
```

```
linear_trend3_beta0 = as.numeric(linear_trend_model3$coefficients[1])
linear_trend3_beta1 = as.numeric(linear_trend_model3$coefficients[2])
```

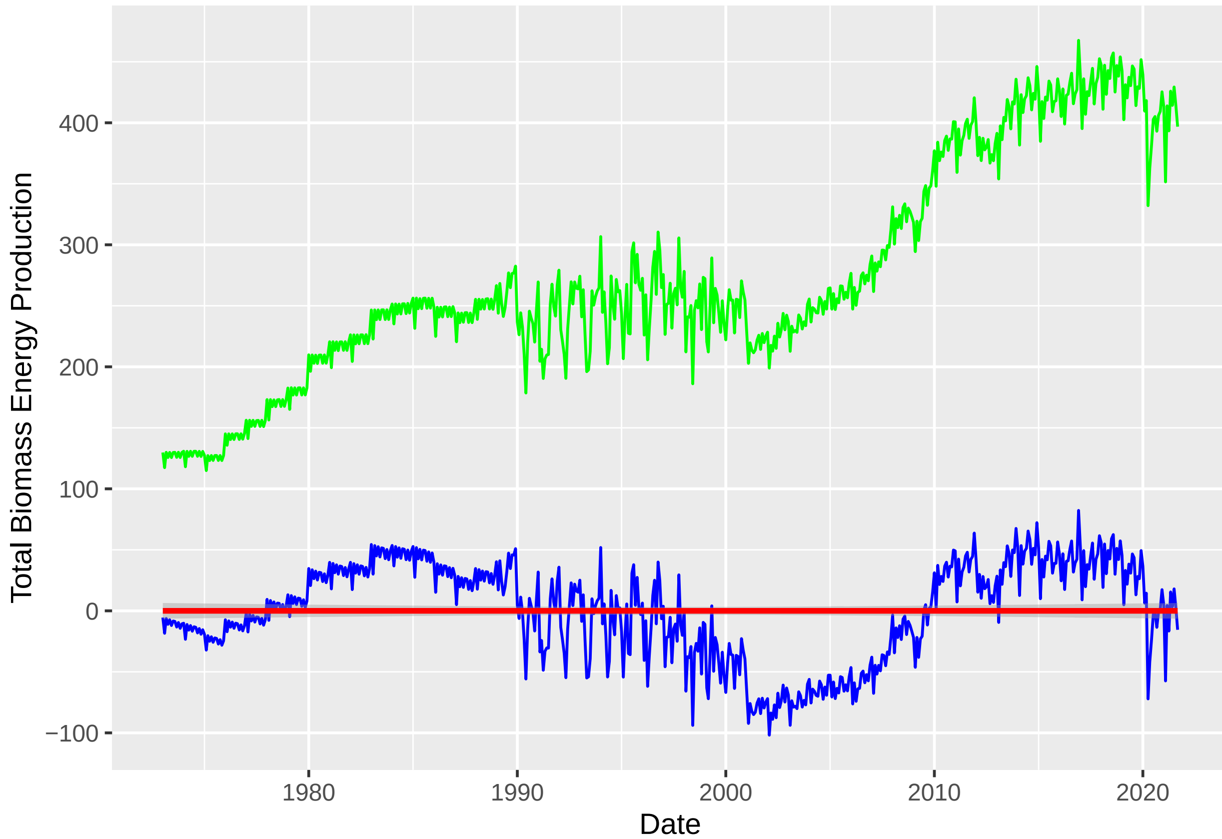
The slope and intercept for Total Biomass Energy production are 0.4743829 and 134.789735 respectively.The slope and intercept for Total Renewable Energy Production are 0.8805065 and 323.1824347 respectively. The slope and intercept for Hydroelectric Power Consumption are -0.0792415 and 259.1830296 respectively.

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?

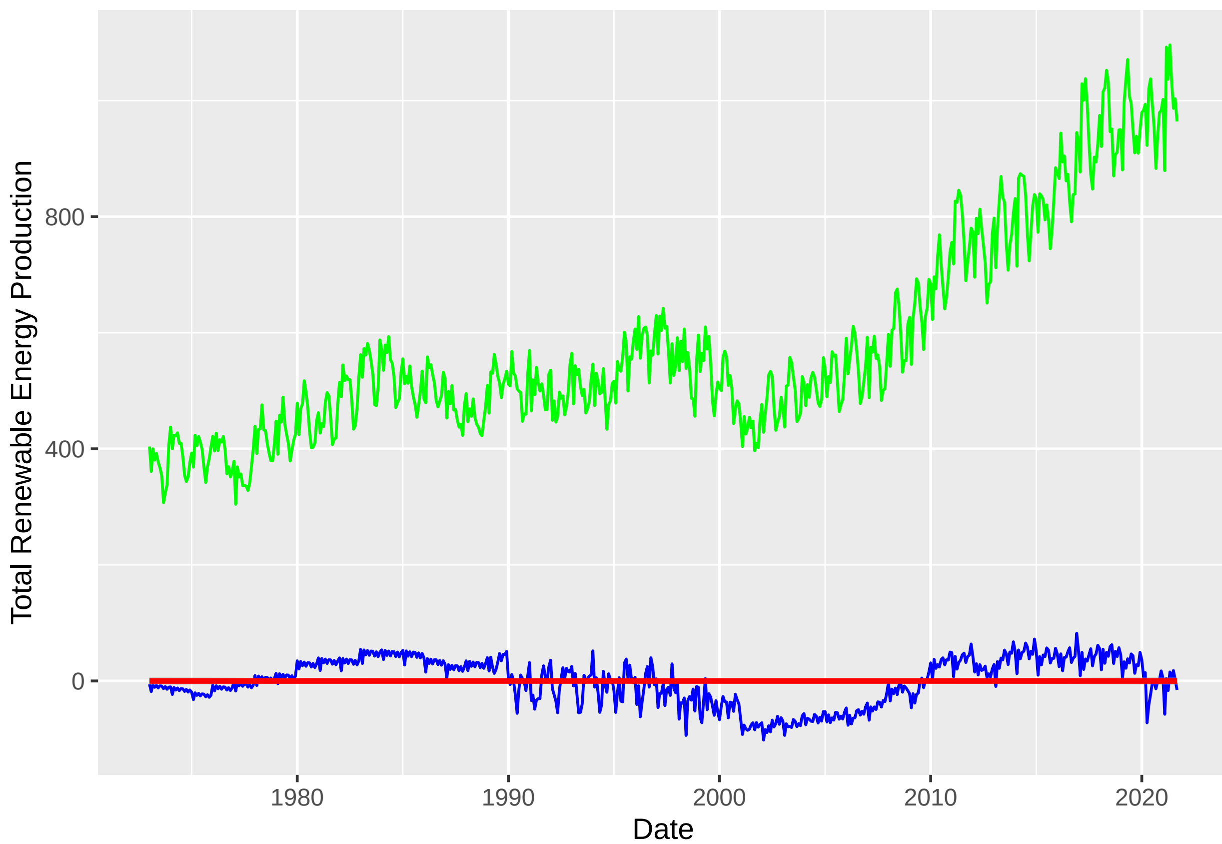
```
t = c(1: nmonths)

biomass_detrend = new_data[, 2] - (linear_trend1_beta0 + linear_trend1_beta1*t)
renewable_detrend = new_data[, 3] - (linear_trend2_beta0 + linear_trend2_beta1*t)
hydro_detrend = new_data[, 4] - (linear_trend3_beta0 + linear_trend3_beta1*t)

ggplot(data = new_data, aes(x = Date, y = new_data[, 2])) +
             geom_line(color="green") +
             ylab(paste0(colnames(new_data)[(2)],sep="")) +

            geom_line(aes(y=biomass_detrend), col="blue")+
            geom_smooth(aes(y=biomass_detrend),color="red",method="lm")
```

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



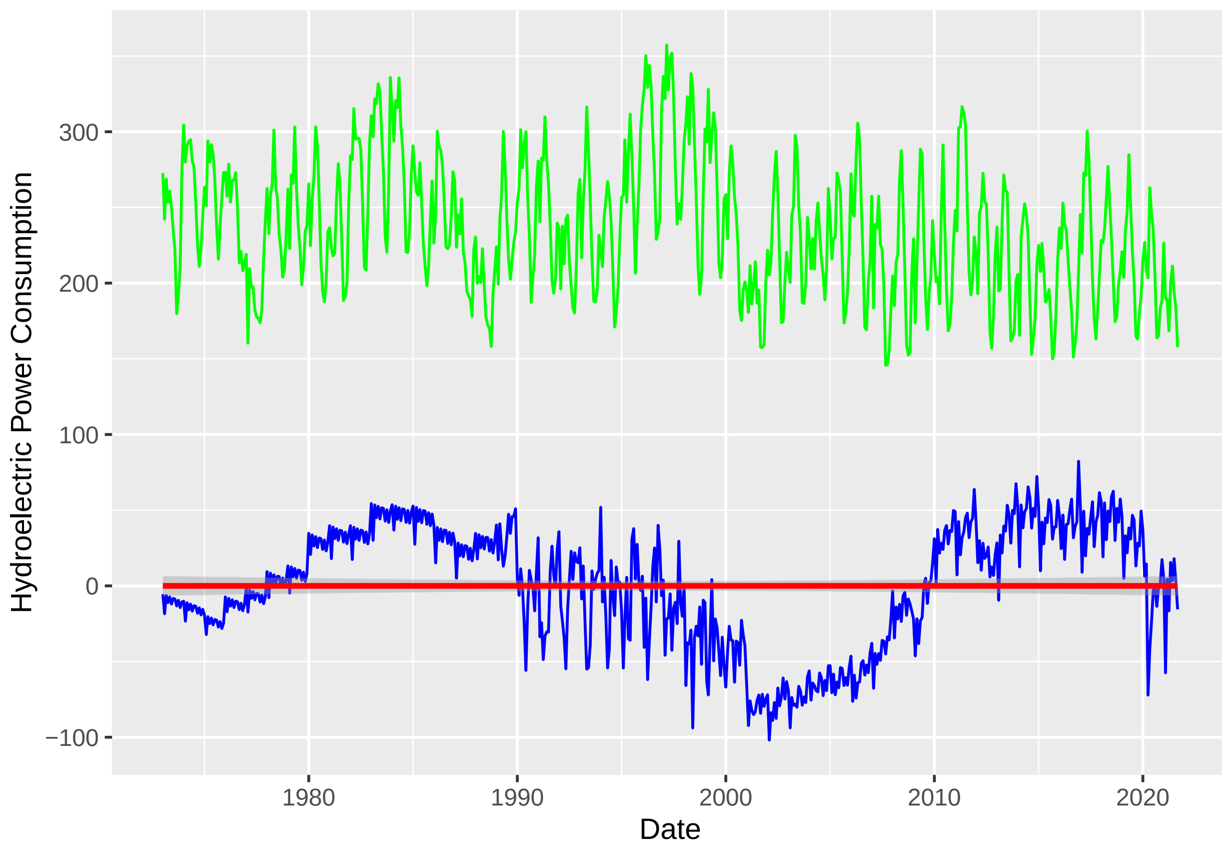

```
ggplot(data = new_data, aes(x = Date, y = new_data[, 3])) +
             geom_line(color="green") +
             ylab(paste0(colnames(new_data)[(3)],sep="")) +

            geom_line(aes(y=biomass_detrend), col="blue")+
            geom_smooth(aes(y=biomass_detrend),color="red",method="lm")
```

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




```
ggplot(data = new_data, aes(x = Date, y = new_data[, 4])) +
            geom_line(color="green") +
            ylab(paste0(colnames(new_data)[(4)],sep="")) +

           geom_line(aes(y=biomass_detrend), col="blue")+
           geom_smooth(aes(y=biomass_detrend),color="red",method="lm")
```

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

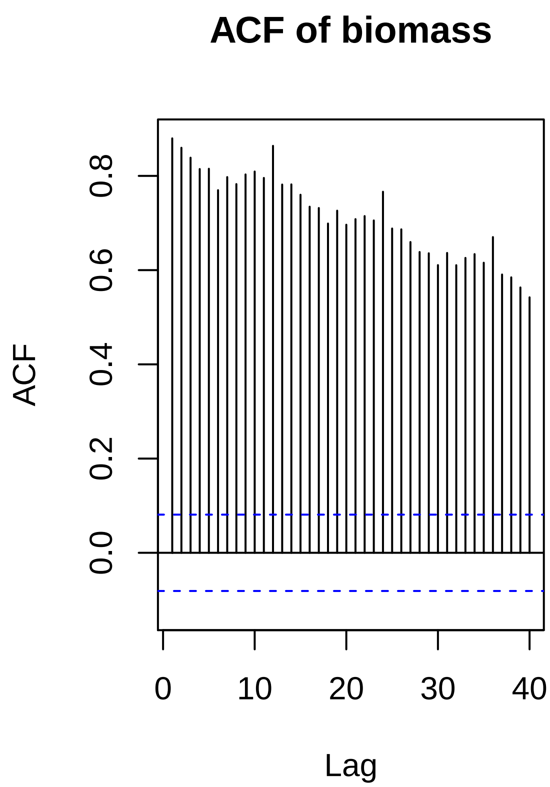


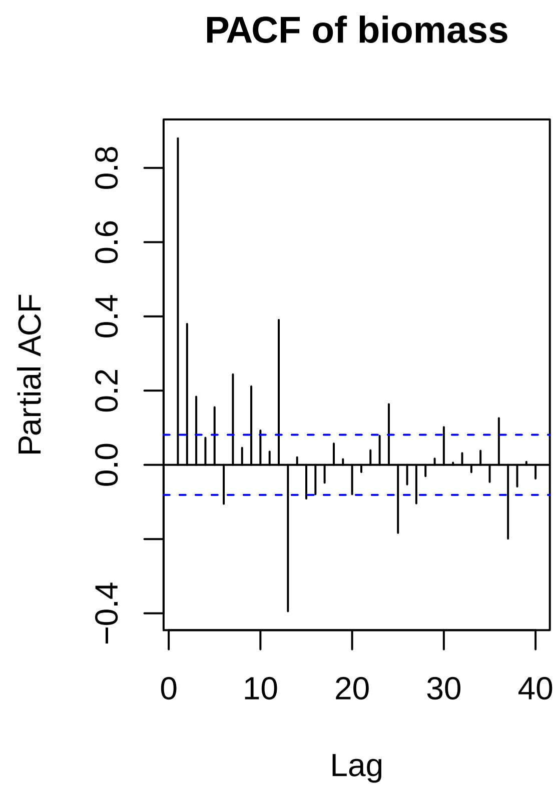
Yes, all three plots had their trend lines changed. All the trend lines became horizontal meaning slope = 0. Also, from the plots, we can observe that all 3 intercepts became 0

Q5

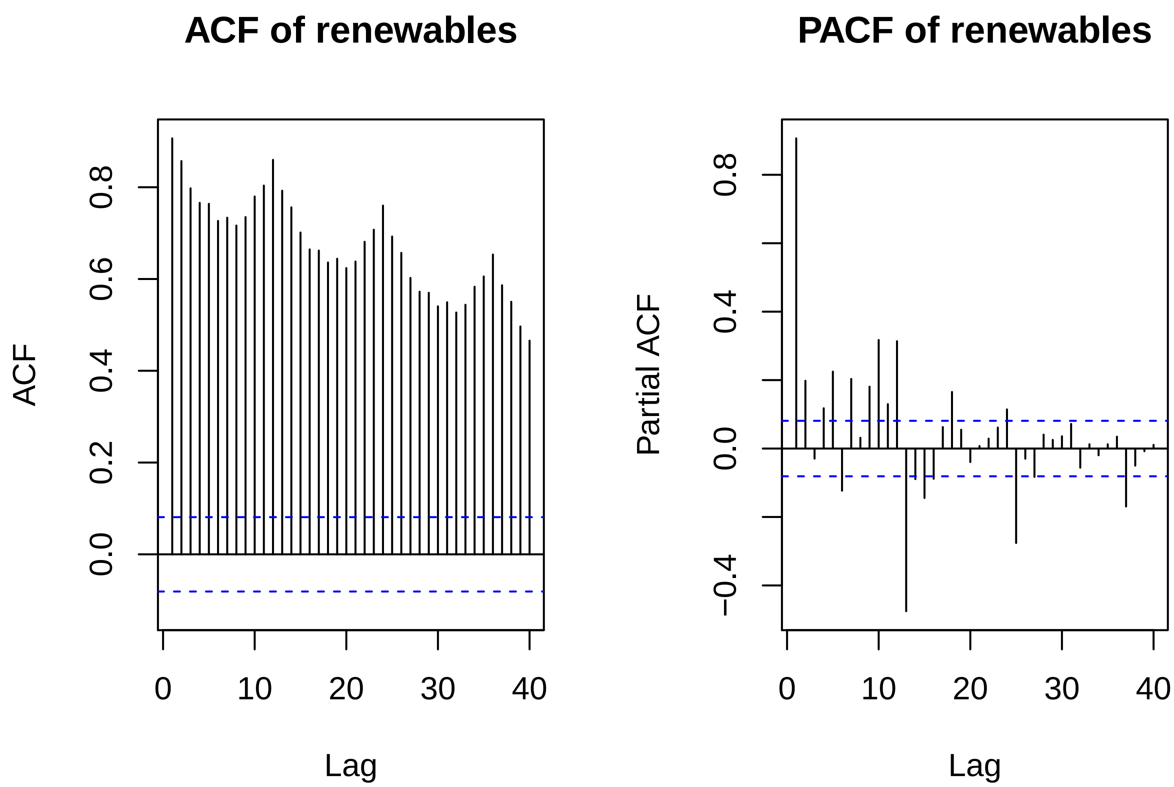
Plot ACF and PACF for the detrended series and compare with the plots from Q1. Did the plots change? How?

```
par(mfrow=c(1,2))
Acf(biomass_detrend,lag.max=40,main=paste("ACF of biomass",sep=" "))
Pacf(biomass_detrend,lag.max=40,main=paste("PACF of biomass",sep=" "))
```

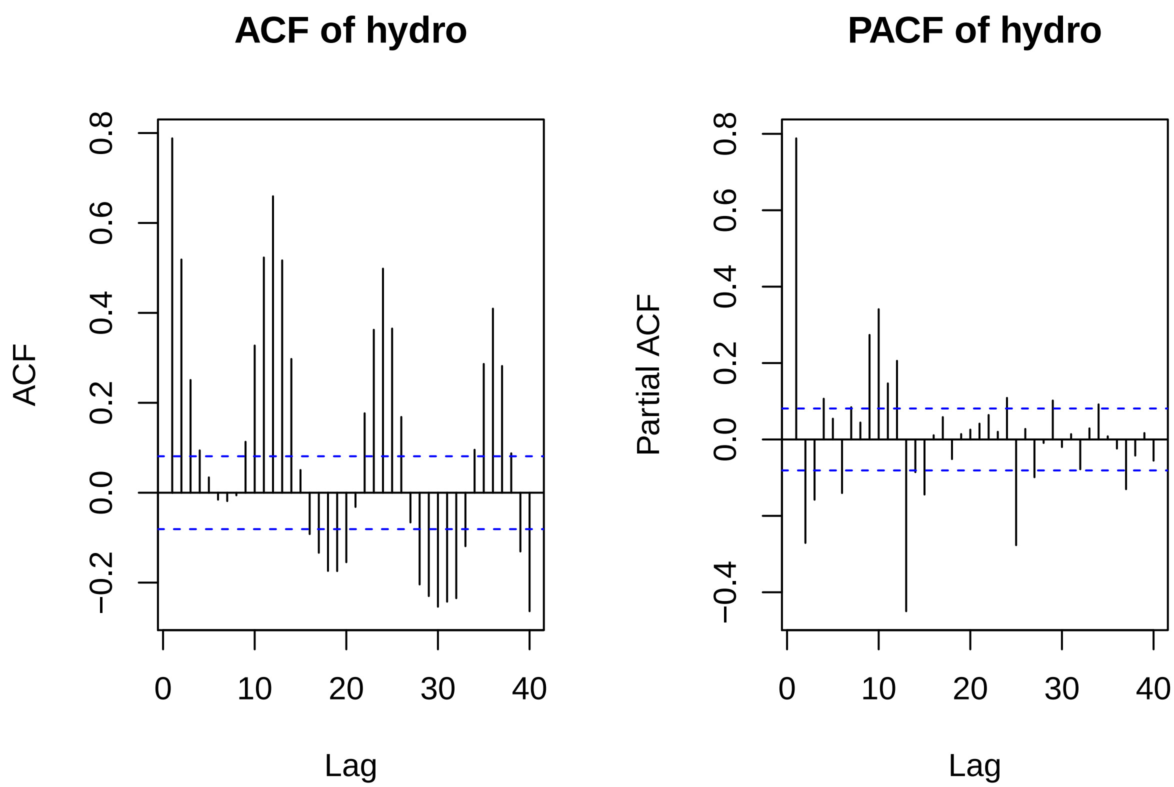





```
Acf(renewable_detrend,lag.max=40,main=paste("ACF of renewables",sep=" "))
Pacf(renewable_detrend,lag.max=40,main=paste("PACF of renewables",sep=" "))
```




```
Acf(hydro_detrend,lag.max=40,main=paste("ACF of hydro",sep=" "))
Pacf(hydro_detrend,lag.max=40,main=paste("PACF of hydro",sep=" "))
```



With regard to ACF plot, there isn't much of a change - the ACF gradually decreases as lag increases for Total Biomass Energy Production and Total Renewable Energy Production whereas for Hydro Power Consumption, the ACF plot suggests seasonality in the data. Therefore, in addition to detrending, we'd also need to remove seasonality. Same goes with PACF plots too, marginal change, if any, on the magnitude of the correlation function. Have to explore de-seasoning to know more!

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.

```
dummies = seasonaldummy(ts_object[,2])

seas_means_model_biomass=lm(new_data[,2]~dummies)
summary(seas_means_model_biomass)
```

```
##
## Call:
```

```
## lm(formula = new_data[, 2] ~ dummies)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -156.96  -51.40  -22.15   60.65  183.31
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  284.241     12.962  21.928   <2e-16 ***
## dummiesJan    -1.498     18.238  -0.082   0.9346
## dummiesFeb   -30.582     18.238  -1.677   0.0941 .
## dummiesMar    -8.873     18.238  -0.486   0.6268
## dummiesApr   -21.009     18.238  -1.152   0.2498
## dummiesMay   -14.065     18.238  -0.771   0.4409
## dummiesJun   -19.601     18.238  -1.075   0.2829
## dummiesJul    -3.499     18.238  -0.192   0.8479
## dummiesAug    -0.252     18.238  -0.014   0.9890
## dummiesSep   -12.518     18.238  -0.686   0.4928
## dummiesOct    -3.629     18.331  -0.198   0.8432
## dummiesNov    -9.592     18.331  -0.523   0.6010
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 89.81 on 573 degrees of freedom
## Multiple R-squared:  0.01056,    Adjusted R-squared:  -0.008439
## F-statistic: 0.5557 on 11 and 573 DF,  p-value: 0.8647
```

```
seas_means_model_renewable=lm(new_data[,3]~dummies)
summary(seas_means_model_renewable)
```

```
##
## Call:
## lm(formula = new_data[, 3] ~ dummies)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -272.95 -111.55  -59.35   65.68  480.41
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  589.971     25.464  23.169   <2e-16 ***
## dummiesJan    11.793     35.828   0.329   0.7422
## dummiesFeb   -40.992     35.828  -1.144   0.2530
## dummiesMar    21.892     35.828   0.611   0.5414
## dummiesApr     8.908     35.828   0.249   0.8037
## dummiesMay    37.500     35.828   1.047   0.2957
## dummiesJun    19.465     35.828   0.543   0.5871
## dummiesJul     8.115     35.828   0.227   0.8209
## dummiesAug   -18.359     35.828  -0.512   0.6086
## dummiesSep   -62.115     35.828  -1.734   0.0835 .
## dummiesOct   -51.377     36.012  -1.427   0.1542
## dummiesNov   -41.789     36.012  -1.160   0.2464
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
##
## Residual standard error: 176.4 on 573 degrees of freedom
## Multiple R-squared:  0.03139,    Adjusted R-squared:  0.0128
## F-statistic: 1.688 on 11 and 573 DF,  p-value: 0.07235
```

```
seas_means_model_hydro=lm(new_data[,4]~dummies)
summary(seas_means_model_hydro)
```

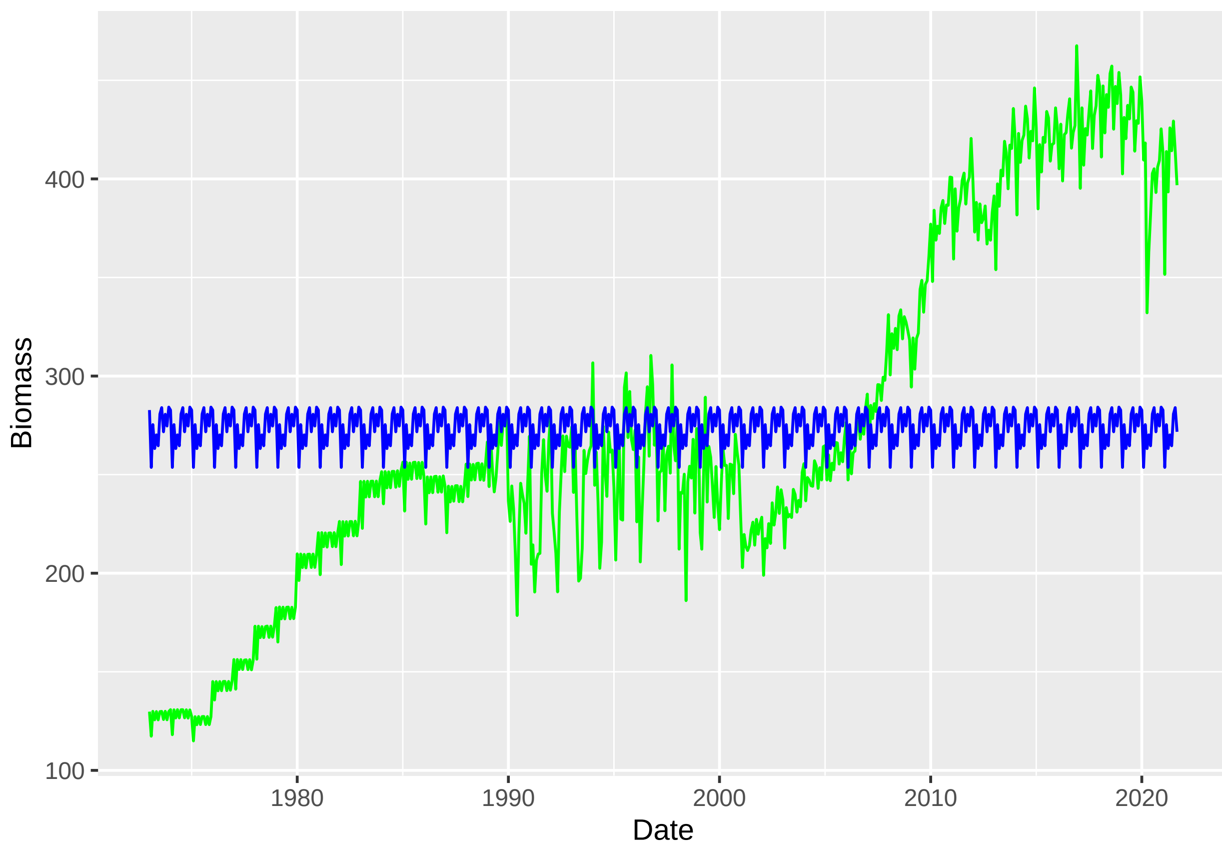
```
##
## Call:
## lm(formula = new_data[, 4] ~ dummies)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -90.253 -23.017  -3.042  21.487  99.478
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  237.841      4.892  48.616  < 2e-16 ***
## dummiesJan    13.558      6.883   1.970  0.04936 *
## dummiesFeb    -8.090      6.883  -1.175  0.24037
## dummiesMar    20.067      6.883   2.915  0.00369 **
## dummiesApr    16.619      6.883   2.414  0.01607 *
## dummiesMay    39.961      6.883   5.805 1.06e-08 ***
## dummiesJun    31.315      6.883   4.549 6.57e-06 ***
## dummiesJul    10.511      6.883   1.527  0.12732
## dummiesAug   -17.853      6.883  -2.594  0.00974 **
## dummiesSep   -49.852      6.883  -7.242 1.43e-12 ***
## dummiesOct   -48.086      6.919  -6.950 9.96e-12 ***
## dummiesNov   -32.187      6.919  -4.652 4.08e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 33.89 on 573 degrees of freedom
## Multiple R-squared:  0.4182, Adjusted R-squared:  0.4071
## F-statistic: 37.45 on 11 and 573 DF,  p-value: < 2.2e-16
```

```
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_renewable$coefficients[1]
beta_coeff_energy=seas_means_model_renewable$coefficients[2:12]
beta_int_hydro=seas_means_model_hydro$coefficients[1]
beta_coeff_hydro=seas_means_model_hydro$coefficients[2:12]
```

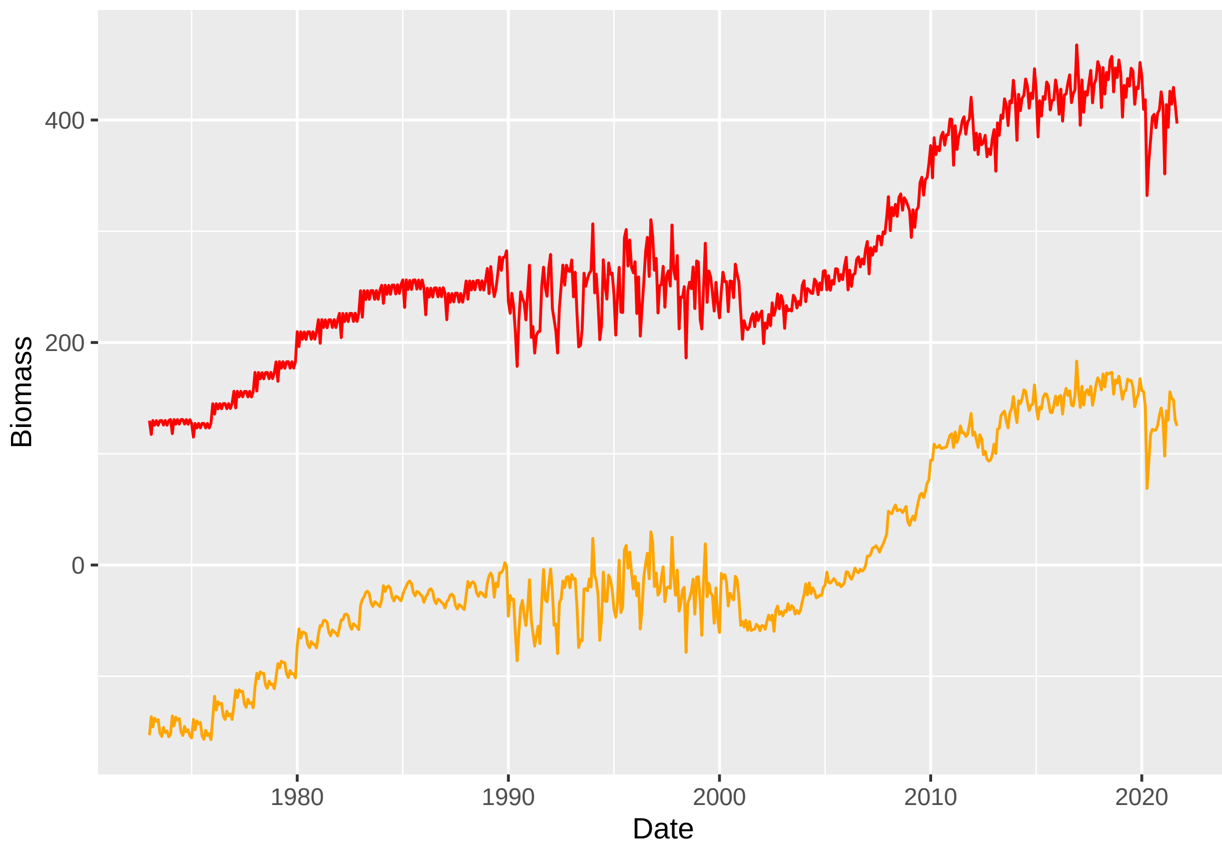
From Seasonal means model for biomass and renewable, the aren't significant dummy variables. For hydro power consumption, there are many significant dummy variables (May, June, Sep, Oct, and Nov). This bolsters the previous argument - the prersence of seasonality in hydro power consumption data ### Q7 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?

```
biomass_seas_comp=array(0,nmonths)
```

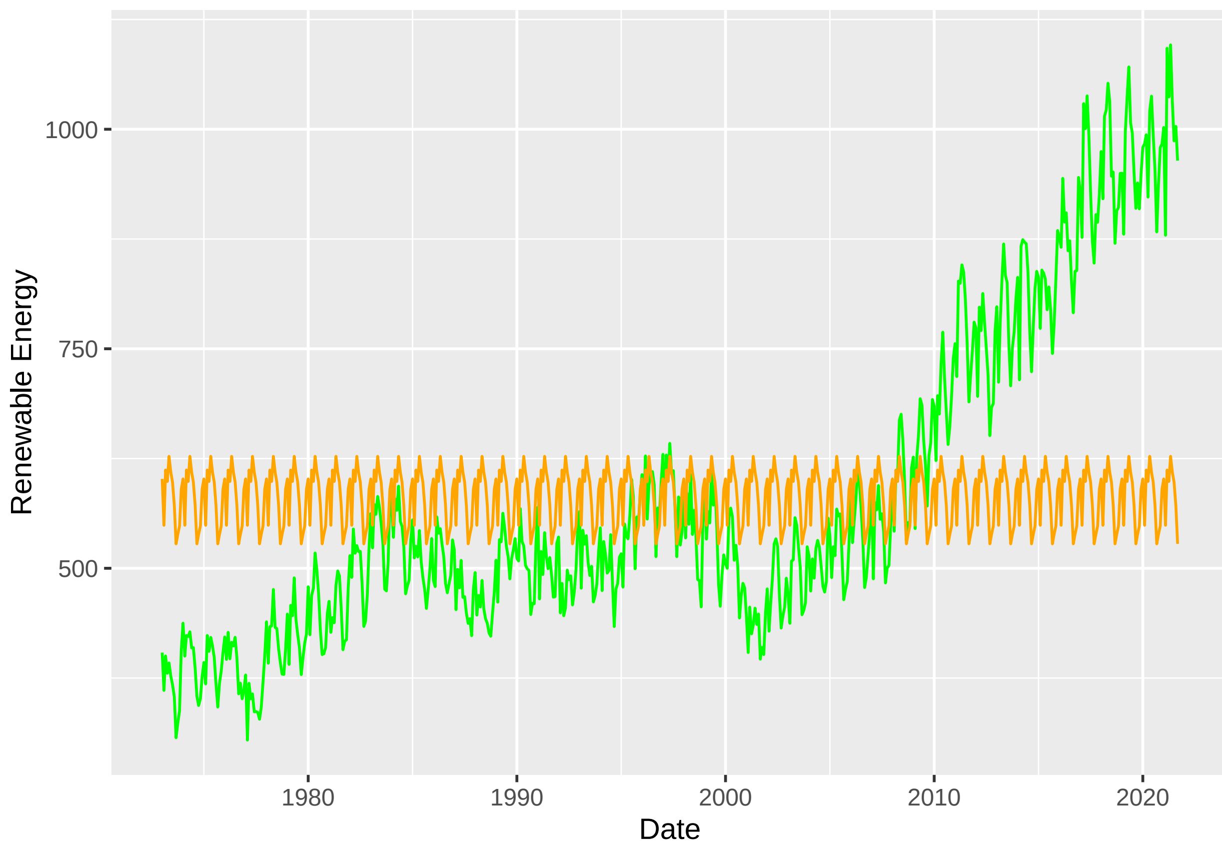
```
for(i in 1:nmonths){
  biomass_seas_comp[i]=(beta_int_biomass+beta_coeff_biomass%*%dummies[i,])

}

ggplot(new_data, aes(x=Date, y=new_data[,2])) +
            geom_line(color="green") +
            ylab(paste0("Biomass")) +
            geom_line(aes(y=biomass_seas_comp), col="blue")
```



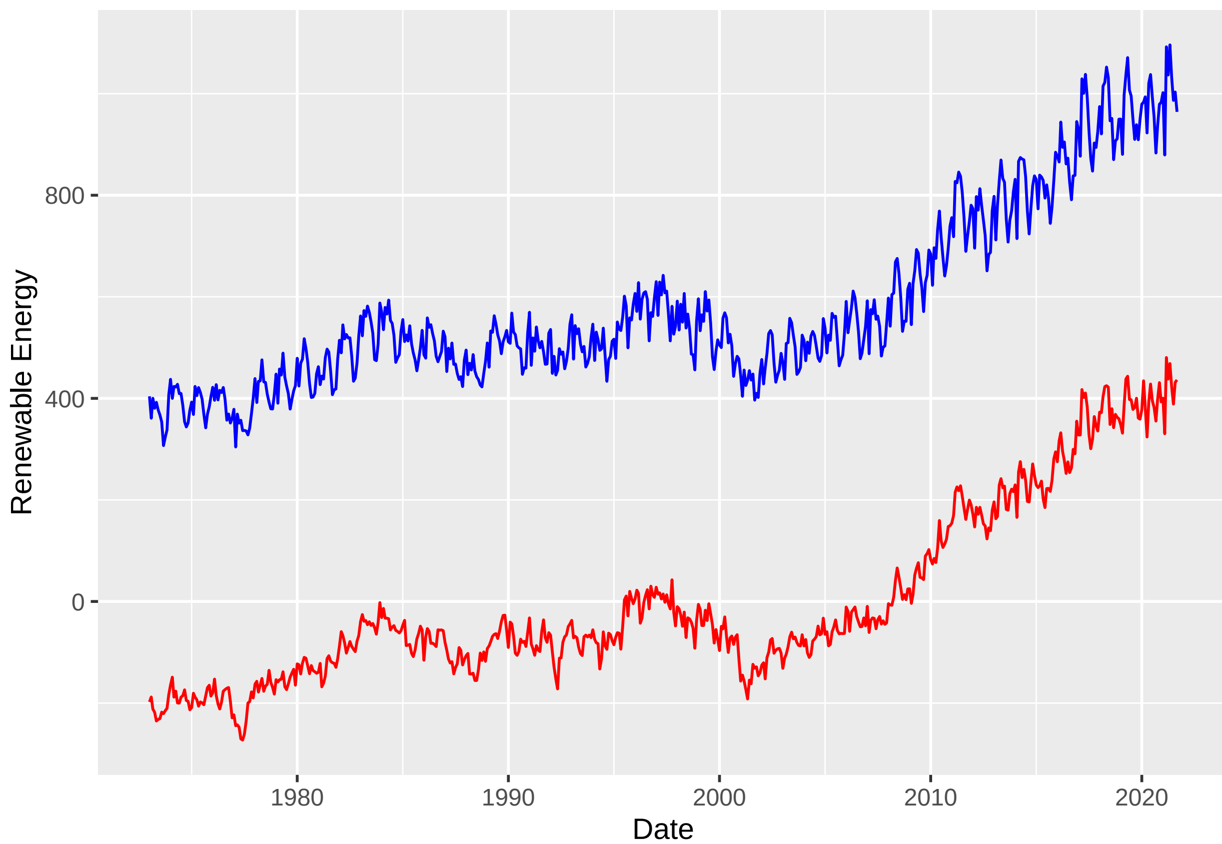

```
deseason_biomass = new_data[,2]-biomass_seas_comp

ggplot(new_data, aes(x=Date, y=new_data[,2])) +
            geom_line(color="red") +
            ylab(paste0("Biomass")) +
            geom_line(aes(y=deseason_biomass), col="orange")
```



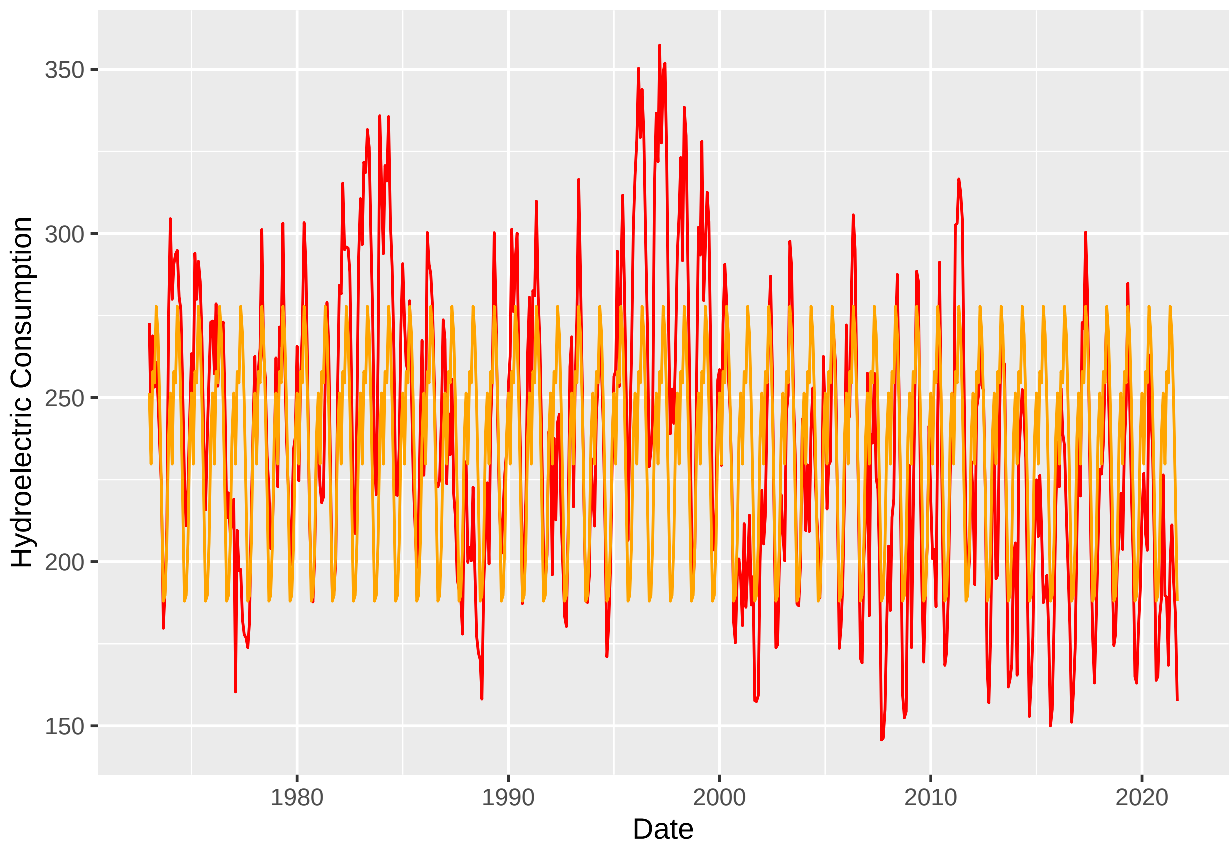

```
energy_seas_comp=array(0,nmonths)
for(i in 1:nmonths){
  energy_seas_comp[i]=(beta_int_energy+beta_coeff_energy%*%dummies[i,])

}
ggplot(new_data, aes(x=Date, y=new_data[,3])) +
            geom_line(color="green") +
            ylab(paste0("Renewable Energy")) +
            geom_line(aes(y=energy_seas_comp), col="orange")
```



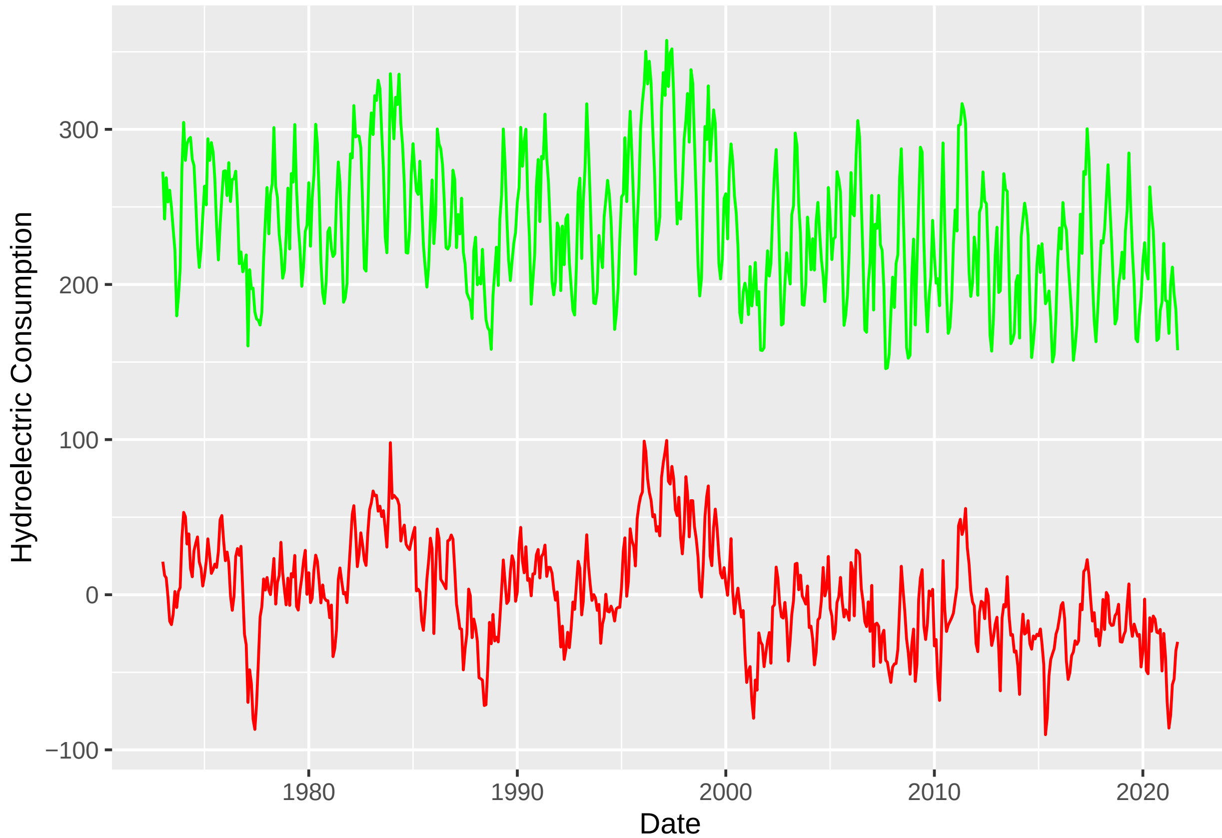

```r
deseason_renewable_energy <- new_data[,3]-energy_seas_comp
ggplot(new_data, aes(x=Date, y=new_data[,3])) +
            geom_line(color="blue") +
            ylab(paste0("Renewable Energy")) +
            geom_line(aes(y=deseason_renewable_energy), col="red")
```




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

}
ggplot(new_data, aes(x=Date, y=new_data[,4])) +
            geom_line(color="red") +
            ylab(paste0("Hydroelectric Consumption")) +
            geom_line(aes(y=hydro_seas_comp), col="orange")
```




```
deseason_hydro = new_data[,4]-hydro_seas_comp
ggplot(new_data, aes(x=Date, y=new_data[,4])) +
            geom_line(color="green") +
            ylab(paste0("Hydroelectric Consumption")) +
            geom_line(aes(y=deseason_hydro), col="red")
```

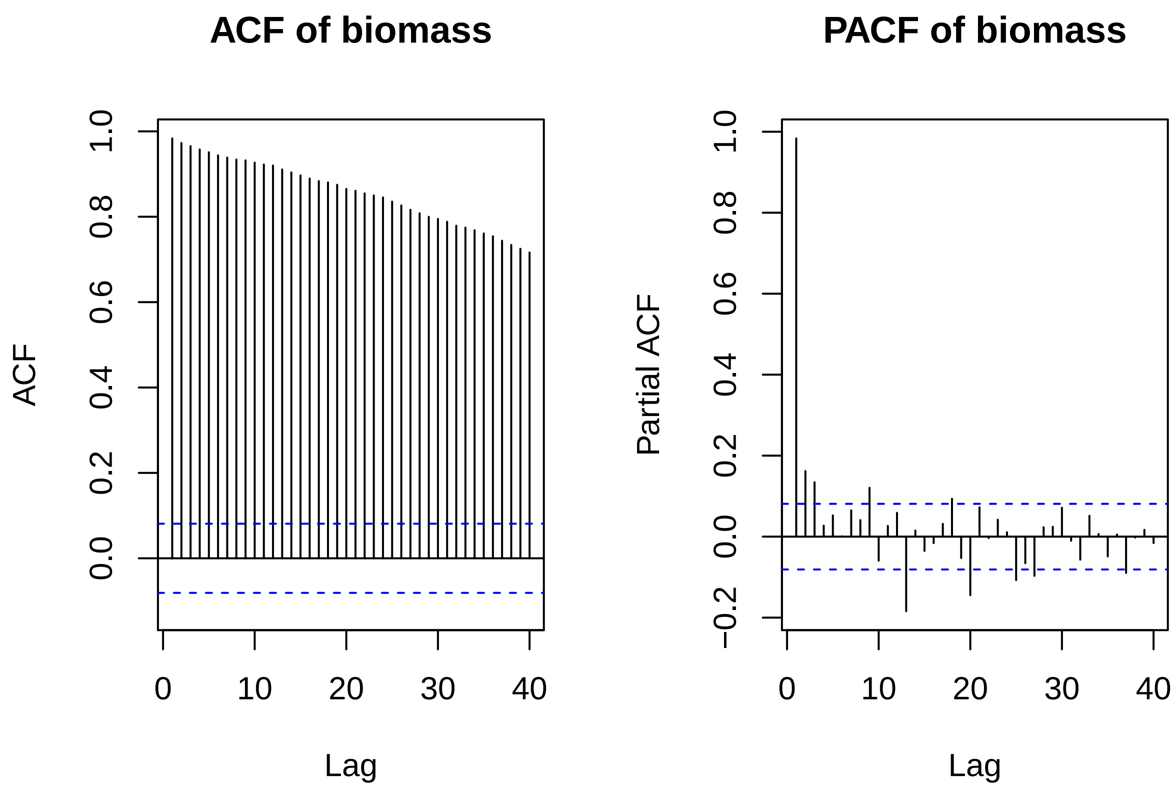


After de-seasoning, all 3 plots changed. The overall shape of the plots, especially for biomass and renewables, did not change much. For hydropower the de-seasoned plot is very evident in terms of increases and decreases of data. All conclusions, once again, fall in line with what we observed with Question 6

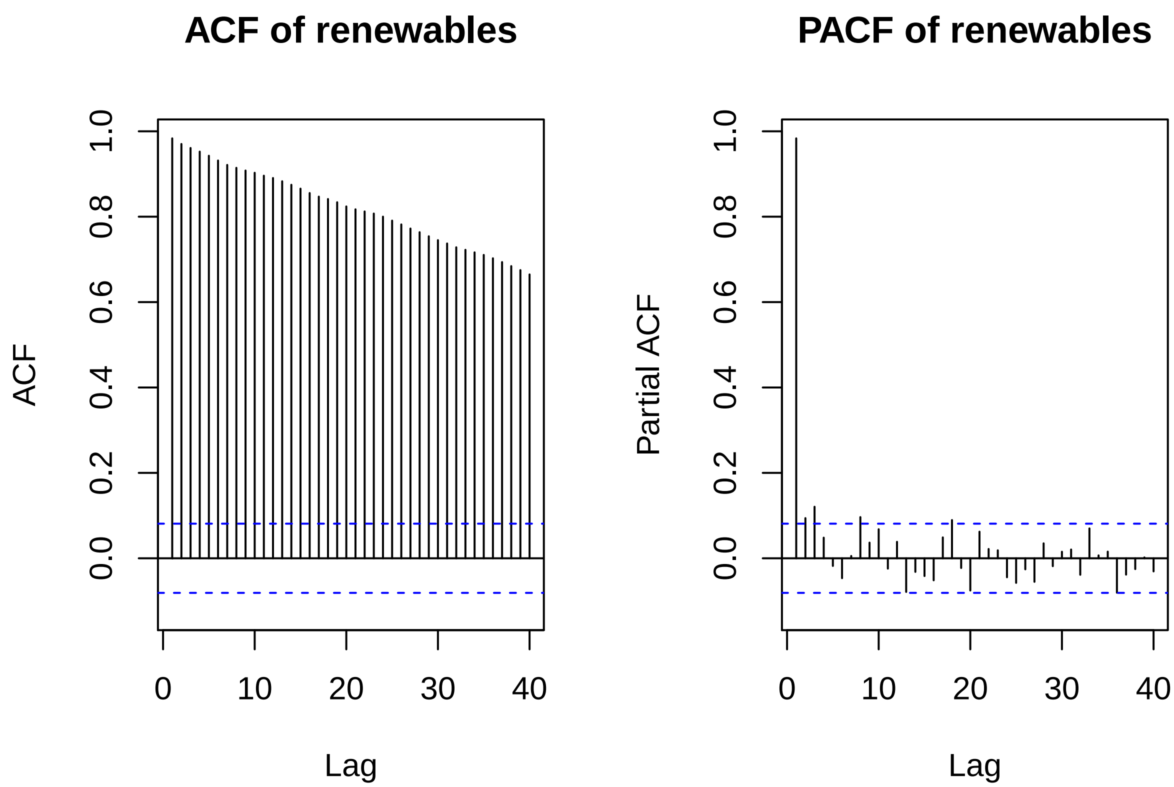
Q8

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

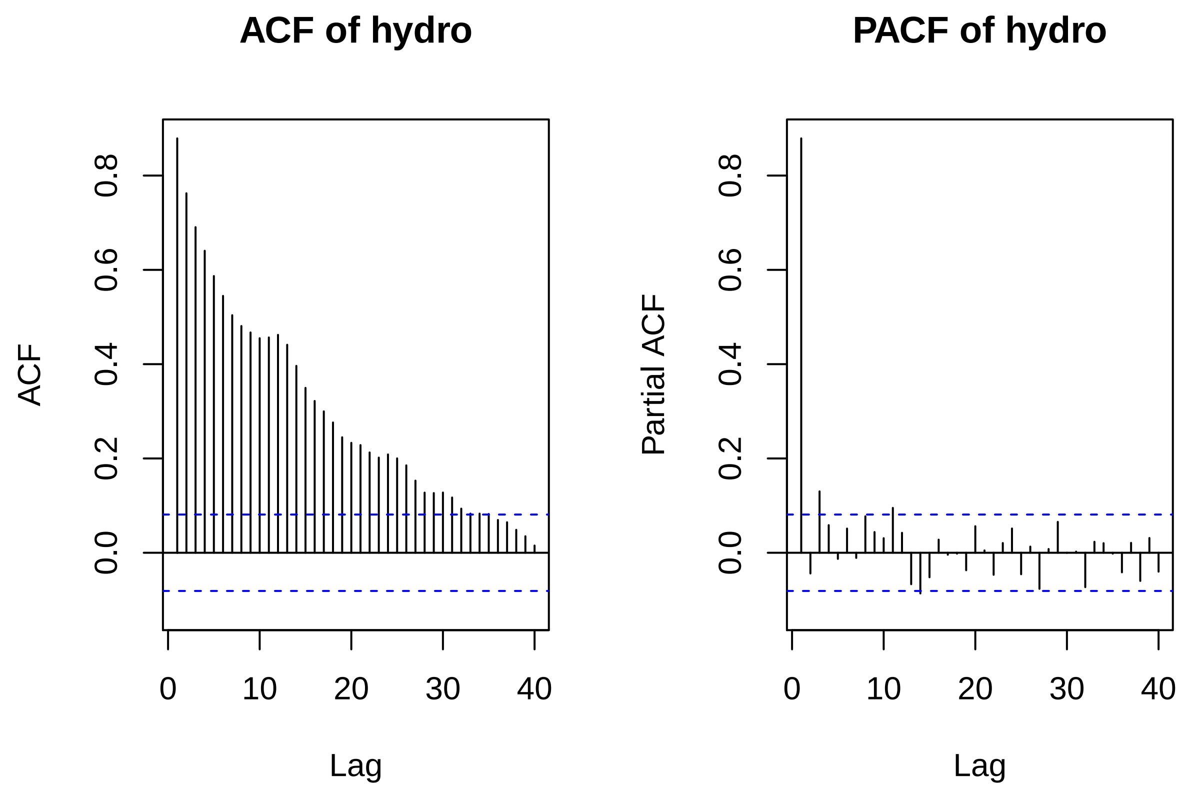
```
par(mfrow=c(1,2))
Acf(deseason_biomass,lag.max=40,main=paste("ACF of biomass",sep=" "))
Pacf(deseason_biomass,lag.max=40,main=paste("PACF of biomass",sep=" "))
```




```
Acf(deseason_renewable_energy,lag.max=40,main=paste("ACF of renewables",sep=" "))
Pacf(deseason_renewable_energy,lag.max=40,main=paste("PACF of renewables",sep=" "))
```




```
Acf(deseason_hydro,lag.max=40,main=paste("ACF of hydro",sep=" "))
Pacf(deseason_hydro,lag.max=40,main=paste("PACF of hydro",sep=" "))
```



The ACF of Hydro Power Consumed, after deseasoning, shows an obvious seasonal variation to a gradual decrease in autocorrelation with increase in lag. The PACF of Hydro Power Consumed shows a weak seasonal variation with almost zero variation after deseasoning. The ACF plots for biomass and renewables were pretty much the same but the PACF plots were showed a decrease in seasonality