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

Assignment 3 - Due date 02/10/23

Hugh Cipparone

Directions

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

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

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

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

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

Questions

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

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

```
#Load/install required package here

library(forecast)
```

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

```
library(tseries)
library(Kendall)
library(lubridate)
```

```
## Loading required package: timechange

##
```

```
## Attaching package: 'lubridate'

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

```r
library(tidyverse)
```

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.2 --

## v ggplot2 3.4.0      v purrr   0.3.5
## v tibble  3.1.8      v dplyr   1.0.10
## v tidyr   1.2.1      v stringr 1.5.0
## v readr   2.1.3      v forcats 0.5.2
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x lubridate::as.difftime() masks base::as.difftime()
## x lubridate::date()        masks base::date()
## x dplyr::filter()          masks stats::filter()
## x lubridate::intersect()   masks base::intersect()
## x dplyr::lag()             masks stats::lag()
## x lubridate::setdiff()     masks base::setdiff()
## x lubridate::union()       masks base::union()
```

#load and clean data

```r
data<-read.csv("./Data/Table_10.1_Renewable_Energy_Production_and_Consumption_by_Source-Edit.csv")

date<-ym(data$Month)

data.edit<-cbind(data,date)%>%
  select(date,Total.Biomass.Energy.Production, Total.Renewable.Energy.Production, Hydroelectric.Power.Co

head(data.edit)
```

```
##         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
```

#create time series object

```r
ts_energy_a3 <- ts(data.edit[,(2:4)], frequency=12, start=c(1973,1))
head(ts_energy_a3)
```

```
##          Total.Biomass.Energy.Production Total.Renewable.Energy.Production
## Jan 1973                         129.787                           403.981
## Feb 1973                         117.338                           360.900
## Mar 1973                         129.938                           400.161
```

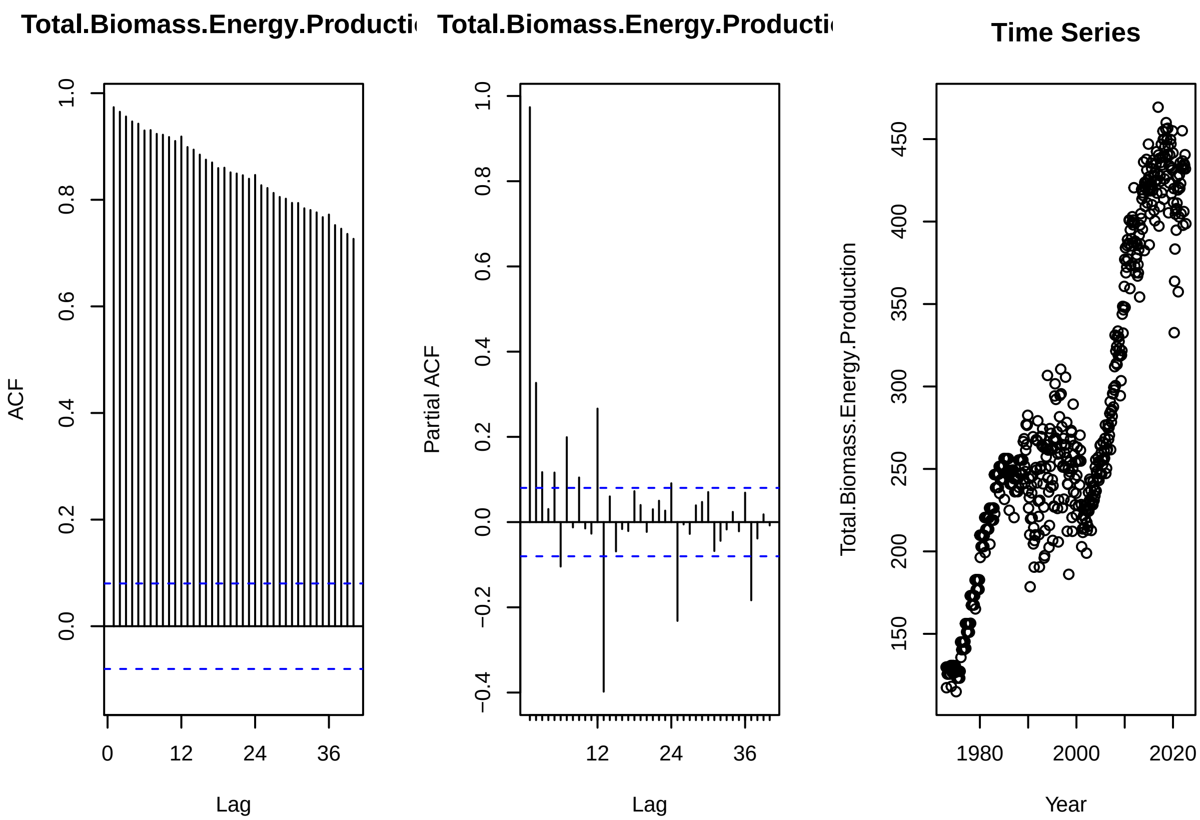
```
## Apr 1973                        125.636                        380.470
## May 1973                        129.834                        392.141
## Jun 1973                        125.611                        377.232
##          Hydroelectric.Power.Consumption
## Jan 1973                        272.703
## Feb 1973                        242.199
## Mar 1973                        268.810
## Apr 1973                        253.185
## May 1973                        260.770
## Jun 1973                        249.859
```

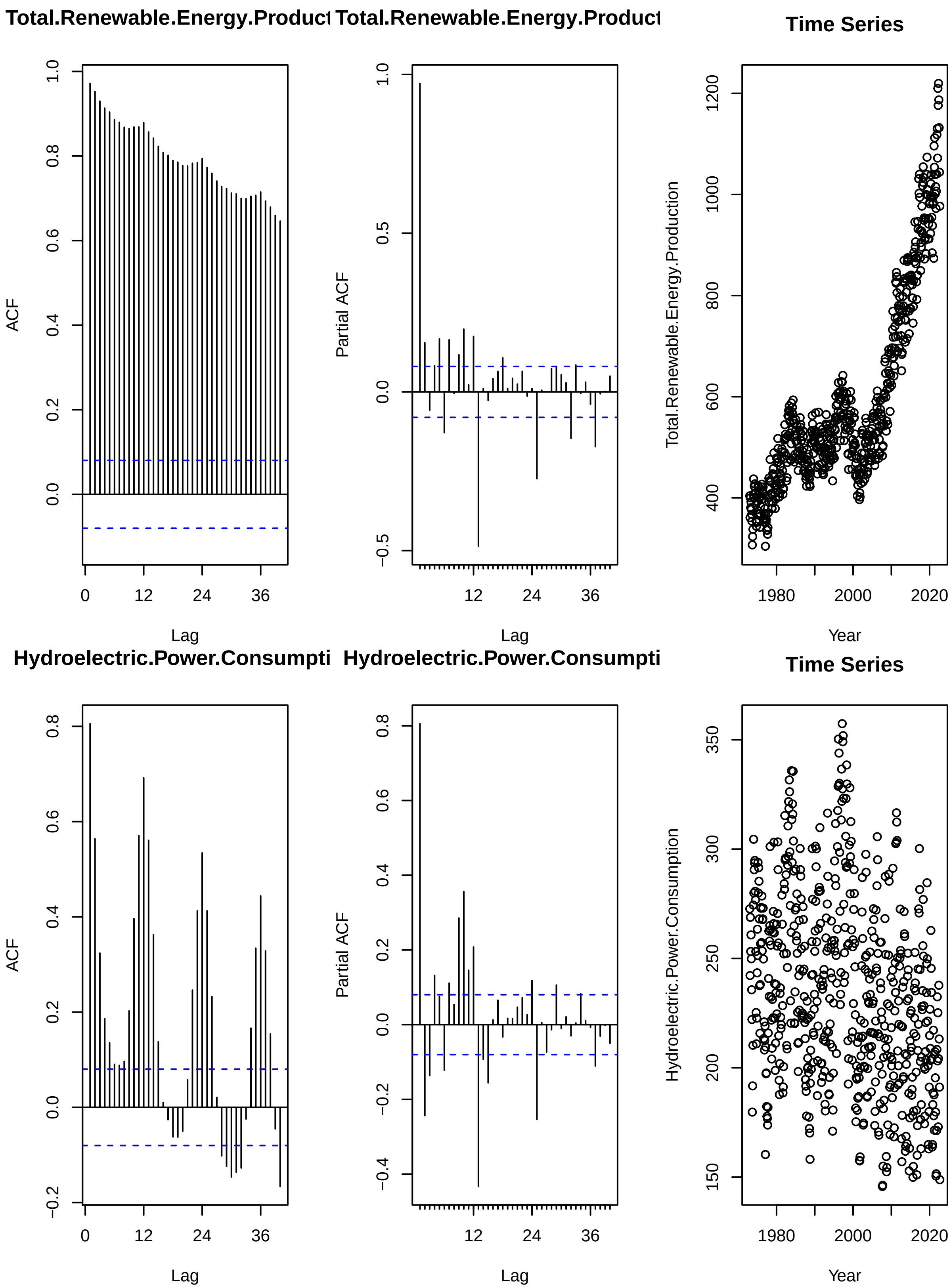
##Trend Component

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)

```
column.name=colnames(ts_energy_a3)


par(mfrow=c(1,3))  #place plot side by side
for(i in 1:3){
  Acf(ts_energy_a3[,i],lag.max=40, main=column.name[i])
  Pacf(ts_energy_a3[,i],lag.max=40, main=column.name[i])
  plot(data.edit[,1], data.edit[,i+1], main=paste("Time Series"), xlab="Year", ylab=column.name[i])
}
```




Total.Renewable.Energy.Product
Total.Renewable.Energy.Product
Time Series
ACF
Partial ACF
Total.Renewable.Energy.Production
Lag
Lag
Year
Hydroelectric.Power.Consumpti
Hydroelectric.Power.Consumpti
Time Series
ACF
Partial ACF
Hydroelectric.Power.Consumption
Lag
Lag
Year

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?

Answer: Total Biomass and Renewable both appear to have a trend upwards over time (although it appears not perfectly linear, particularly in the early years). Hydroelectric consumption however does not appear to have a trend over time.

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:nrow(data.edit))
linear_trend_model_1=lm(data.edit[,2]~t)
summary(linear_trend_model_1)
```

```
##
## Call:
## lm(formula = data.edit[, 2] ~ t)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -102.800  -23.994    5.667   32.265   82.192
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 1.337e+02  3.245e+00   41.22   <2e-16 ***
## t           4.800e-01  9.402e-03   51.05   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 39.59 on 595 degrees of freedom
## Multiple R-squared:  0.8142, Adjusted R-squared:  0.8138
## F-statistic:  2607 on 1 and 595 DF,  p-value: < 2.2e-16
```

```
beta0_1=as.numeric(linear_trend_model_1$coefficients[1])
beta1_1=as.numeric(linear_trend_model_1$coefficients[2])

linear_trend_model_2=lm(data.edit[,3]~t)
summary(linear_trend_model_2)
```

```
##
## Call:
## lm(formula = data.edit[, 3] ~ t)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -238.75  -61.85    8.59   64.48  352.27
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 312.2475     8.4902   36.78   <2e-16 ***
## t             0.9362     0.0246   38.05   <2e-16 ***
## ---
```

```
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 103.6 on 595 degrees of freedom
## Multiple R-squared:  0.7088, Adjusted R-squared:  0.7083
## F-statistic:  1448 on 1 and 595 DF,  p-value: < 2.2e-16
```

```r
beta0_2=as.numeric(linear_trend_model_2$coefficients[1])
beta1_2=as.numeric(linear_trend_model_2$coefficients[2])

linear_trend_model_3=lm(data.edit[,4]~t)
summary(linear_trend_model_3)
```

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

```r
beta0_3=as.numeric(linear_trend_model_3$coefficients[1])
beta1_3=as.numeric(linear_trend_model_3$coefficients[2])
```

Interpretation: For total biomass and total renewables, the slope is significantly positive ($p<0.05$, coeff > 0). For hydro power consumption, the slope is significantly negative ($p<0.05$, coeff <0). This means that, over time (and assuming that the linear regression assumptions hold - unclear if they do), total biomass energy production and renewables energy production has increased but hydro power consumption has decreased. It is important to note that the biomass and renewables coefficients are quite different - renewables being around 0.9 while biomass is about half of that rate - around 0.4. This suggests that renewables have seen a larger growth over time than biomass.

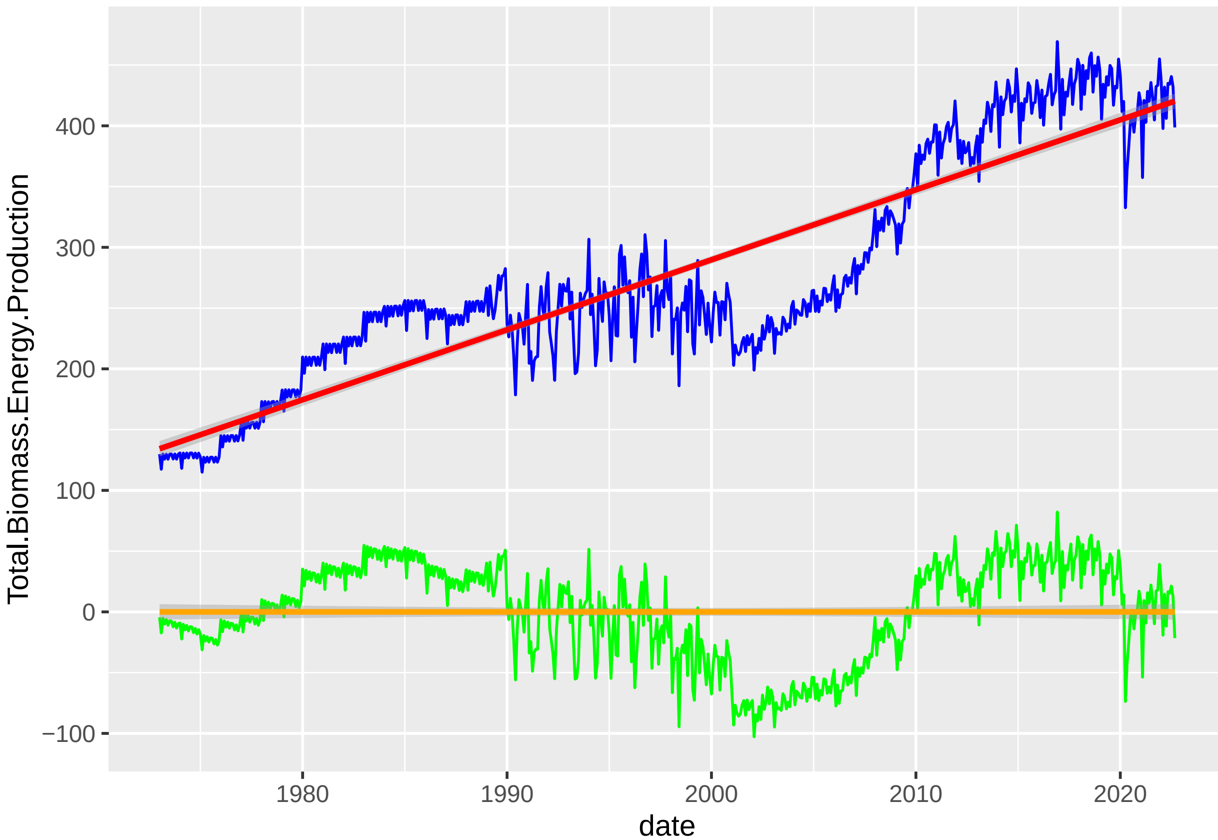
Q4

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
#Biomass
detrend_data_1 <- data.edit[,2]-(beta0_1+beta1_1*t)


ggplot(data.edit, aes(x=date, y=data.edit[,2])) +
            geom_line(color="blue") +
            ylab(column.name[1]) +
            #geom_abline(intercept = beta0, slope = beta1, color="red")
            geom_smooth(color="red",method="lm") +
```

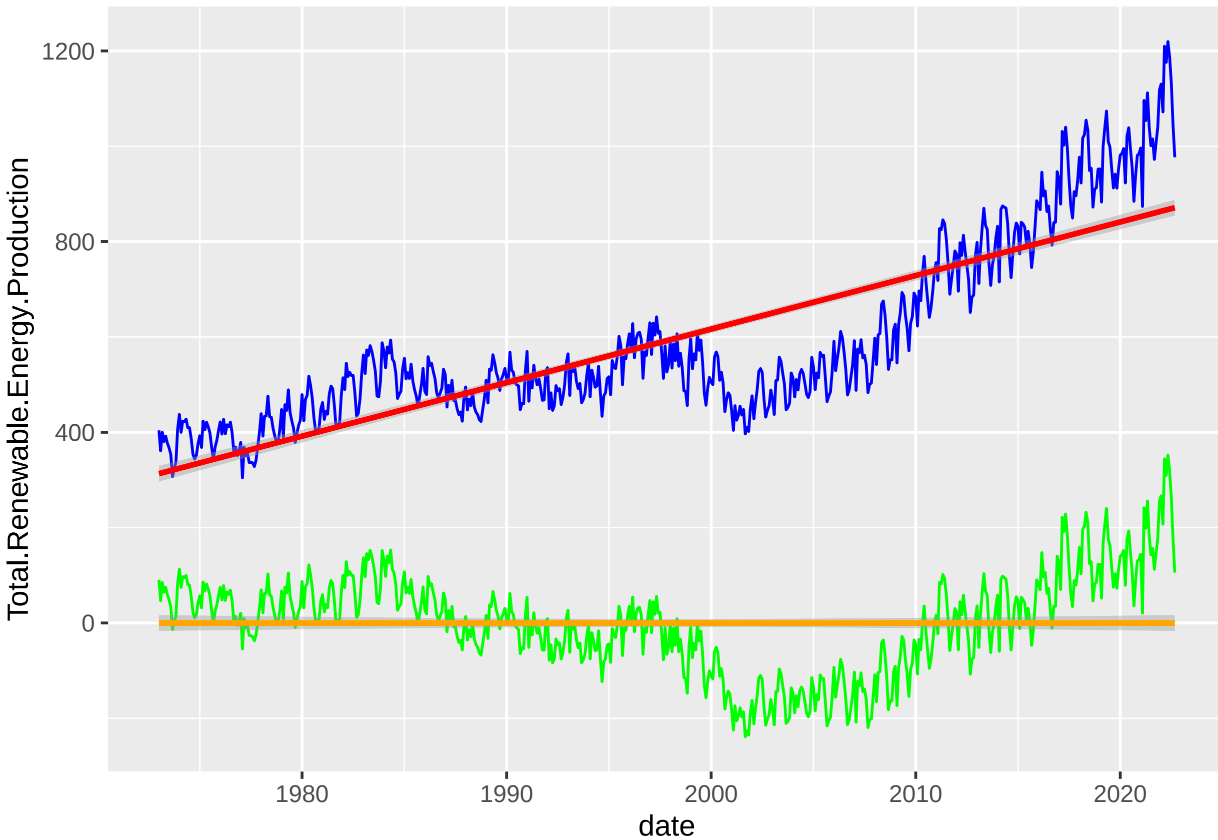
```r
            geom_line(aes(y=detrend_data_1), col="green")+
            geom_smooth(aes(y=detrend_data_1),color="orange",method="lm")
```

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



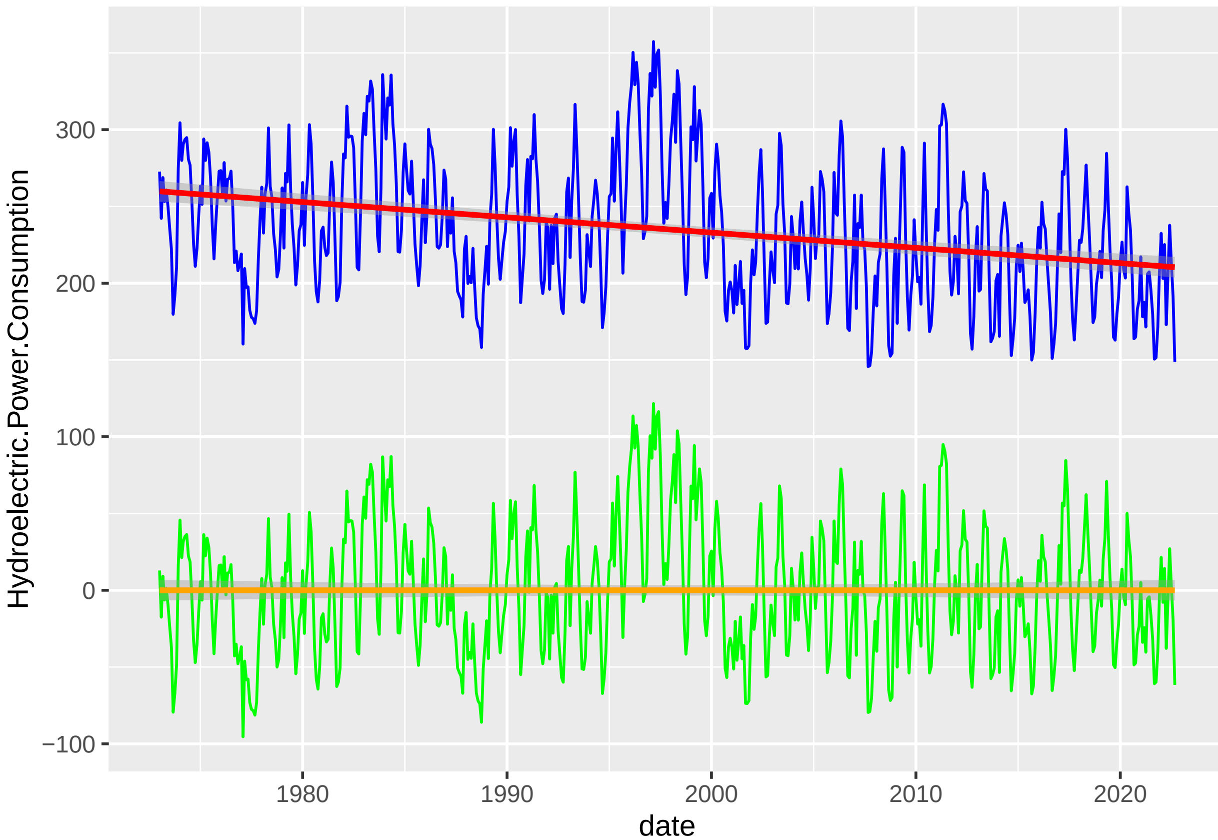

```r
#Renewables
detrend_data_2 <- data.edit[,3]-(beta0_2+beta1_2*t)


ggplot(data.edit, aes(x=date, y=data.edit[,3])) +
            geom_line(color="blue") +
            ylab(column.name[2]) +
            #geom_abline(intercept = beta0, slope = beta1, color="red")
            geom_smooth(color="red",method="lm") +
            geom_line(aes(y=detrend_data_2), col="green")+
            geom_smooth(aes(y=detrend_data_2),color="orange",method="lm")
```

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




```
#Hydro
detrend_data_3 <- data.edit[,4]-(beta0_3+beta1_3*t)


ggplot(data.edit, aes(x=date, y=data.edit[,4])) +
            geom_line(color="blue") +
            ylab(column.name[3]) +
            #geom_abline(intercept = beta0, slope = beta1, color="red")
            geom_smooth(color="red",method="lm") +
            geom_line(aes(y=detrend_data_3), col="green")+
            geom_smooth(aes(y=detrend_data_3),color="orange",method="lm")
```

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




```
detrend_data<-cbind(date,detrend_data_1,detrend_data_2,detrend_data_3)
```

Answer: We can see that for each of the three series the trend was removed - or at least a linear regression now returns no significant variation over the course of the series.

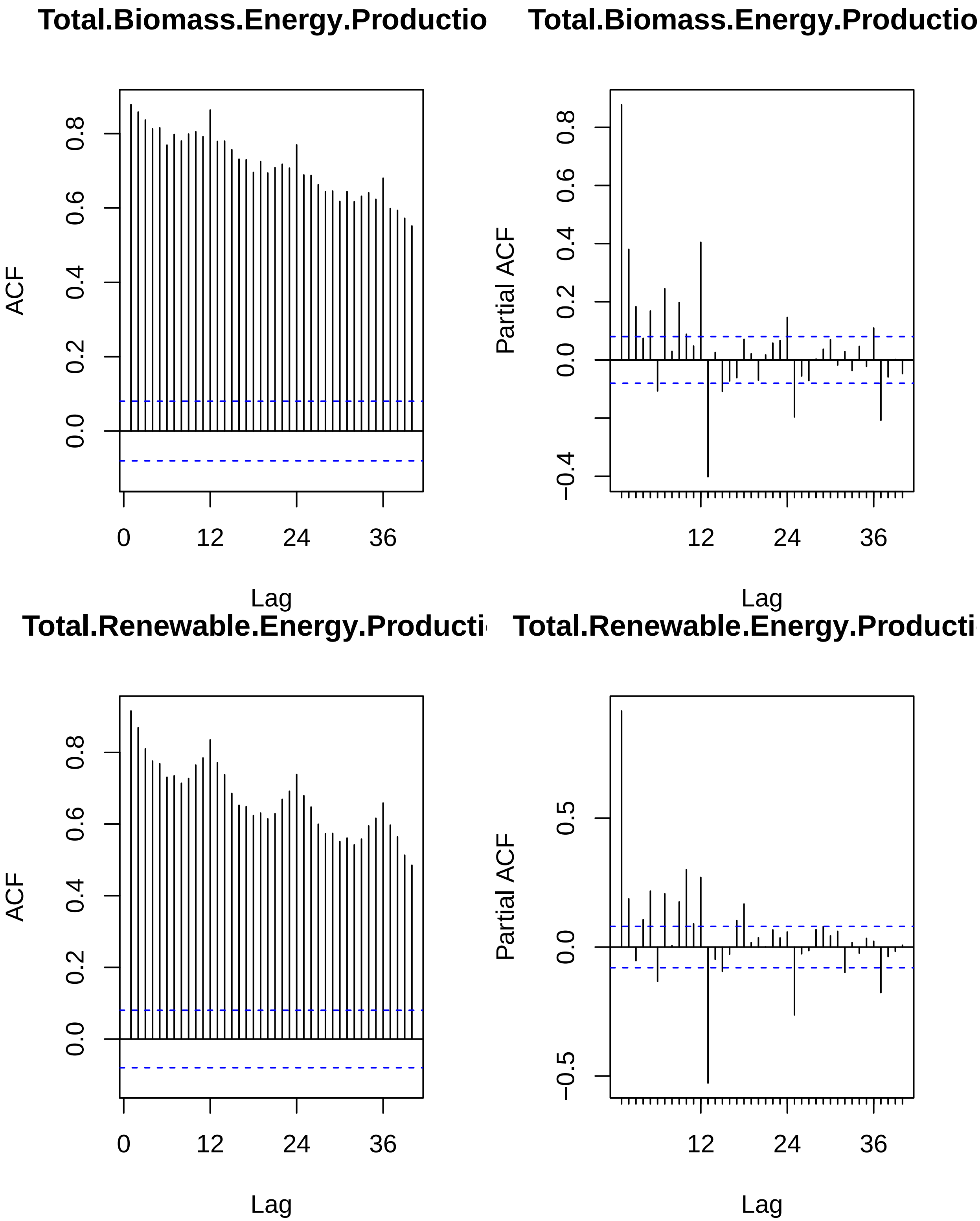
Q5

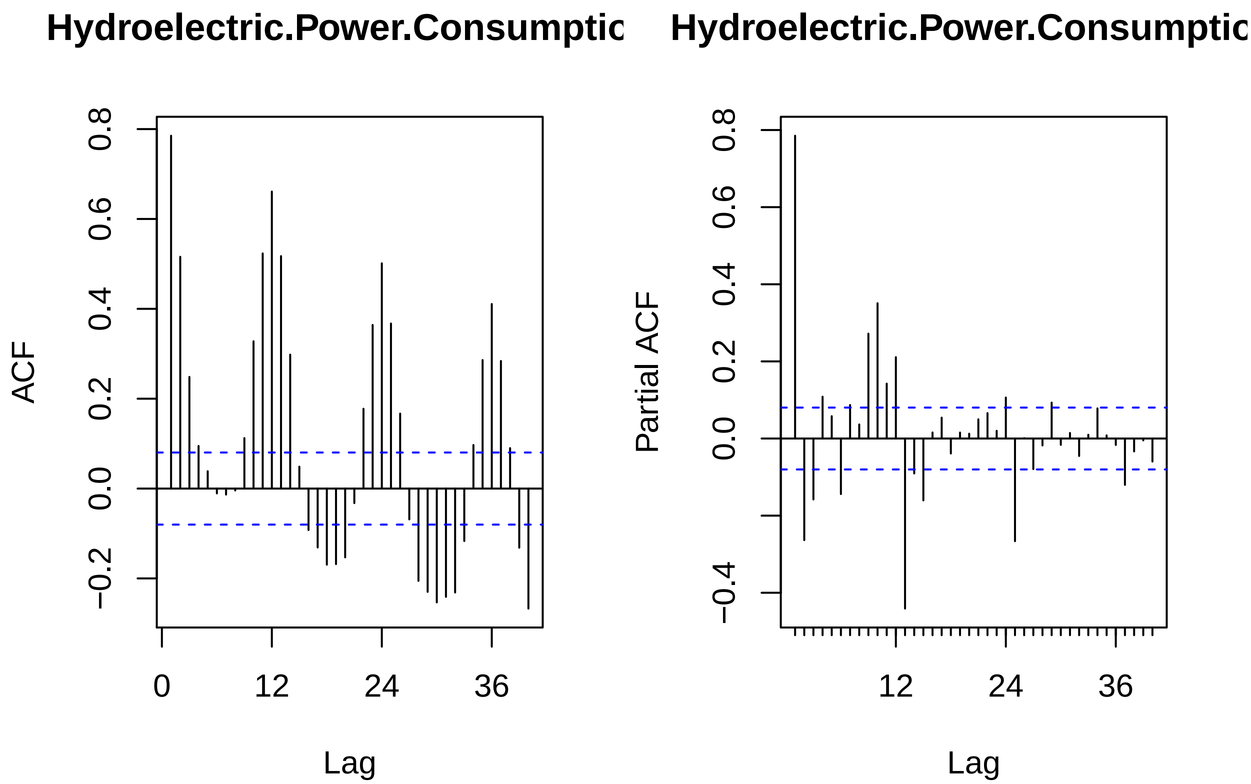
Plot ACF and PACF for the detrended series and compare with the plots from Q1. Did the plots change? How?

```
ts_energy_a3_detrend <- ts(detrend_data[,(2:4)], frequency=12, start=c(1973,1))
head(ts_energy_a3_detrend)
```

```
##          detrend_data_1 detrend_data_2 detrend_data_3
## Jan 1973      -4.429614       90.79731      12.887875
## Feb 1973     -17.358600       46.78016     -17.533238
## Mar 1973      -5.238586       85.10500       9.160650
## Apr 1973     -10.020573       64.47784      -6.381463
## May 1973      -6.302559       75.21269       1.286425
## Jun 1973     -11.005545       59.36753      -9.541688
```

```
par(mfrow=c(1,2))  #place plot side by side
for(i in 1:3){
  Acf(ts_energy_a3_detrend[,i],lag.max=40, main=column.name[i])
  Pacf(ts_energy_a3_detrend[,i],lag.max=40, main=column.name[i])
}
```


Total.Biomass.Energy.Productio
Total.Biomass.Energy.Productio
ACF
Partial ACF
Lag
Lag
Total.Renewable.Energy.Producti
Total.Renewable.Energy.Producti
ACF
Partial ACF
Lag
Lag



Answer: The hydro plot didn't appear to change much, but the biomass and renewables plots both appeared to have lower ACF values and increased the apparent seasonality (the wave-like motion) across the lags.

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.

Answer: The original series' renewables (to some degree) and hydro (to a large degree) have some seasonal trend. Looking at this regression output, it's clear to me that only the december values (the intercept) increases over time ($p<0.05$), or changes over time in any statistically significant manner. This, along with a cumulative p-value much larger than 0.05 for the test, suggests that there isn't seasonality on the biomass series. In the second series (renewables) we once again only see significance in the intercept over time, which - along with a non-significant overall p-value ($p=0.1$), suggests no seasonality. There may be more seasonality here, however - the p-value is close to a 10% confidence level, where for biomass the p-value was 0.8. In the final series multiple months have specific significant changes and the overall p-value is less than 0.05, which collectively indicates clear seasonality for the hydro data.

```
dummies_1 <- seasonaldummy(ts_energy_a3[,1])
dummies_2 <- seasonaldummy(ts_energy_a3[,2])
dummies_3 <- seasonaldummy(ts_energy_a3[,3])


#Then fit a linear model to the seasonal dummies
seas_means_model_1=lm(data.edit[,(2)]~dummies_1)
```

```r
summary(seas_means_model_1)
```

```
##
## Call:
## lm(formula = data.edit[, (2)] ~ dummies_1)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -160.74  -53.67  -24.36   90.73  181.34
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)   288.020     13.163  21.881   <2e-16 ***
## dummies_1Jan   -1.793     18.522  -0.097   0.9229
## dummies_1Feb  -31.102     18.522  -1.679   0.0936 .
## dummies_1Mar   -9.104     18.522  -0.492   0.6232
## dummies_1Apr  -21.502     18.522  -1.161   0.2462
## dummies_1May  -14.238     18.522  -0.769   0.4424
## dummies_1Jun  -19.602     18.522  -1.058   0.2904
## dummies_1Jul   -3.674     18.522  -0.198   0.8428
## dummies_1Aug   -0.612     18.522  -0.033   0.9737
## dummies_1Sep  -13.335     18.522  -0.720   0.4718
## dummies_1Oct   -4.030     18.615  -0.216   0.8287
## dummies_1Nov   -9.849     18.615  -0.529   0.5970
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 92.14 on 585 degrees of freedom
## Multiple R-squared:  0.01018,    Adjusted R-squared:  -0.008437
## F-statistic: 0.5467 on 11 and 585 DF,  p-value: 0.8714
```

```r
seas_means_model_2=lm(data.edit[,(3)]~dummies_2)
summary(seas_means_model_2)
```

```
##
## Call:
## lm(formula = data.edit[, (3)] ~ dummies_2)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -284.92 -122.23  -68.42   91.22  585.68
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)   601.022     27.260  22.048   <2e-16 ***
## dummies_2Jan   11.468     38.358   0.299    0.765
## dummies_2Feb  -41.456     38.358  -1.081    0.280
## dummies_2Mar   23.130     38.358   0.603    0.547
## dummies_2Apr    9.959     38.358   0.260    0.795
## dummies_2May   38.853     38.358   1.013    0.312
## dummies_2Jun   20.378     38.358   0.531    0.595
## dummies_2Jul    8.298     38.358   0.216    0.829
## dummies_2Aug  -19.450     38.358  -0.507    0.612
## dummies_2Sep  -63.770     38.358  -1.662    0.097 .
```

```
## dummies_2Oct  -52.612     38.551  -1.365    0.173
## dummies_2Nov  -42.537     38.551  -1.103    0.270
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 190.8 on 585 degrees of freedom
## Multiple R-squared:  0.02844,    Adjusted R-squared:  0.01017
## F-statistic: 1.557 on 11 and 585 DF,  p-value: 0.1076
```

```
seas_means_model_3=lm(data.edit[,(4)]~dummies_3)
summary(seas_means_model_3)
```

```
##
## Call:
## lm(formula = data.edit[, (4)] ~ dummies_3)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -88.99 -23.47  -2.81  21.99 100.18
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)   237.225      4.878  48.634  < 2e-16 ***
## dummies_3Jan   13.594      6.864   1.981  0.04811 *
## dummies_3Feb   -8.254      6.864  -1.203  0.22964
## dummies_3Mar   19.980      6.864   2.911  0.00374 **
## dummies_3Apr   15.649      6.864   2.280  0.02297 *
## dummies_3May   39.210      6.864   5.713 1.77e-08 ***
## dummies_3Jun   31.209      6.864   4.547 6.61e-06 ***
## dummies_3Jul   10.436      6.864   1.520  0.12895
## dummies_3Aug  -17.909      6.864  -2.609  0.00931 **
## dummies_3Sep  -50.173      6.864  -7.310 8.82e-13 ***
## dummies_3Oct  -48.262      6.898  -6.996 7.22e-12 ***
## dummies_3Nov  -32.285      6.898  -4.680 3.56e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 34.14 on 585 degrees of freedom
## Multiple R-squared:  0.4132, Adjusted R-squared:  0.4022
## F-statistic: 37.45 on 11 and 585 DF,  p-value: < 2.2e-16
```

```
#Store regression coefficients
beta_int_1=seas_means_model_1$coefficients[1]
beta_coeff_1=seas_means_model_1$coefficients[2:12]

beta_int_2=seas_means_model_2$coefficients[1]
beta_coeff_2=seas_means_model_2$coefficients[2:12]

beta_int_3=seas_means_model_3$coefficients[1]
beta_coeff_3=seas_means_model_3$coefficients[2:12]
```

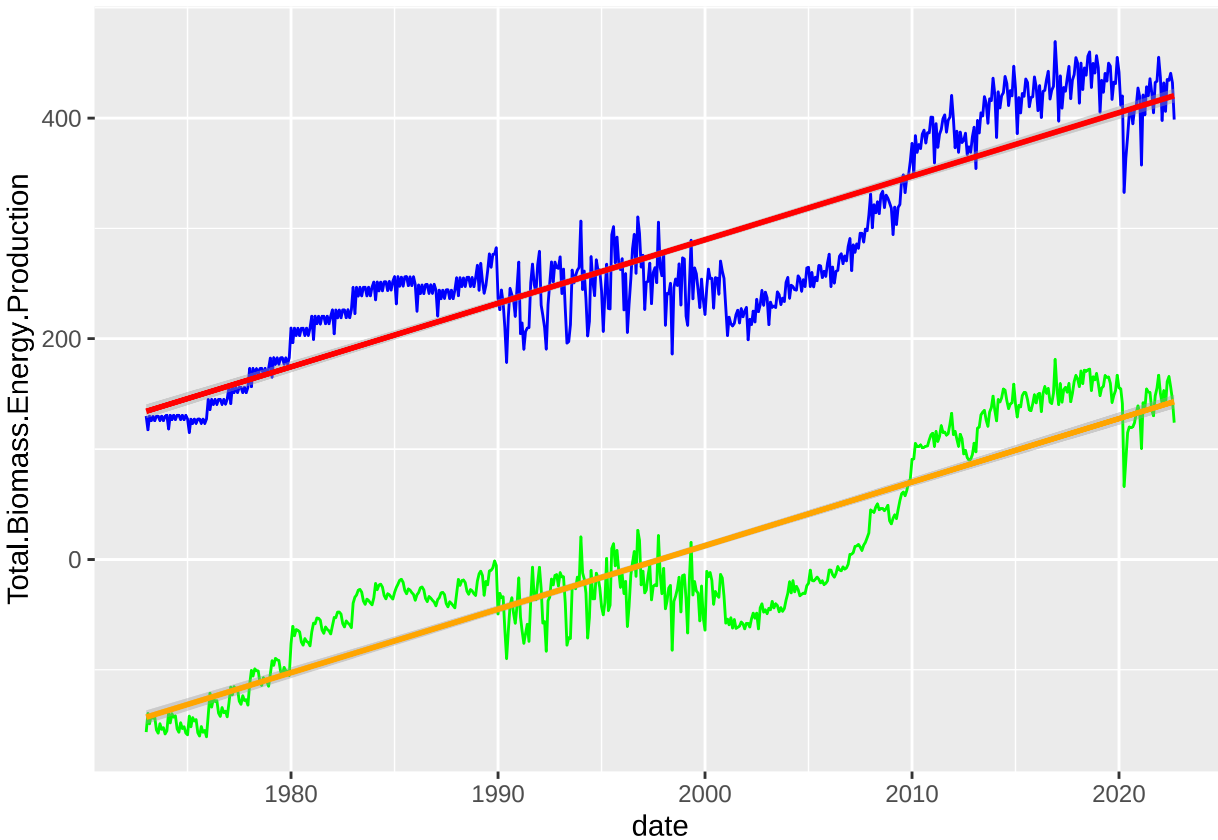
Q7

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?

Answer: I noticed a lessening of the severeity of each of the waves across all the series - but particularly the most seasonal series - the hydro consumption. Essentially it's mitigating the variance of each of the seasons! It also moves all of the results down by a number of values - likely the size of the intercept.

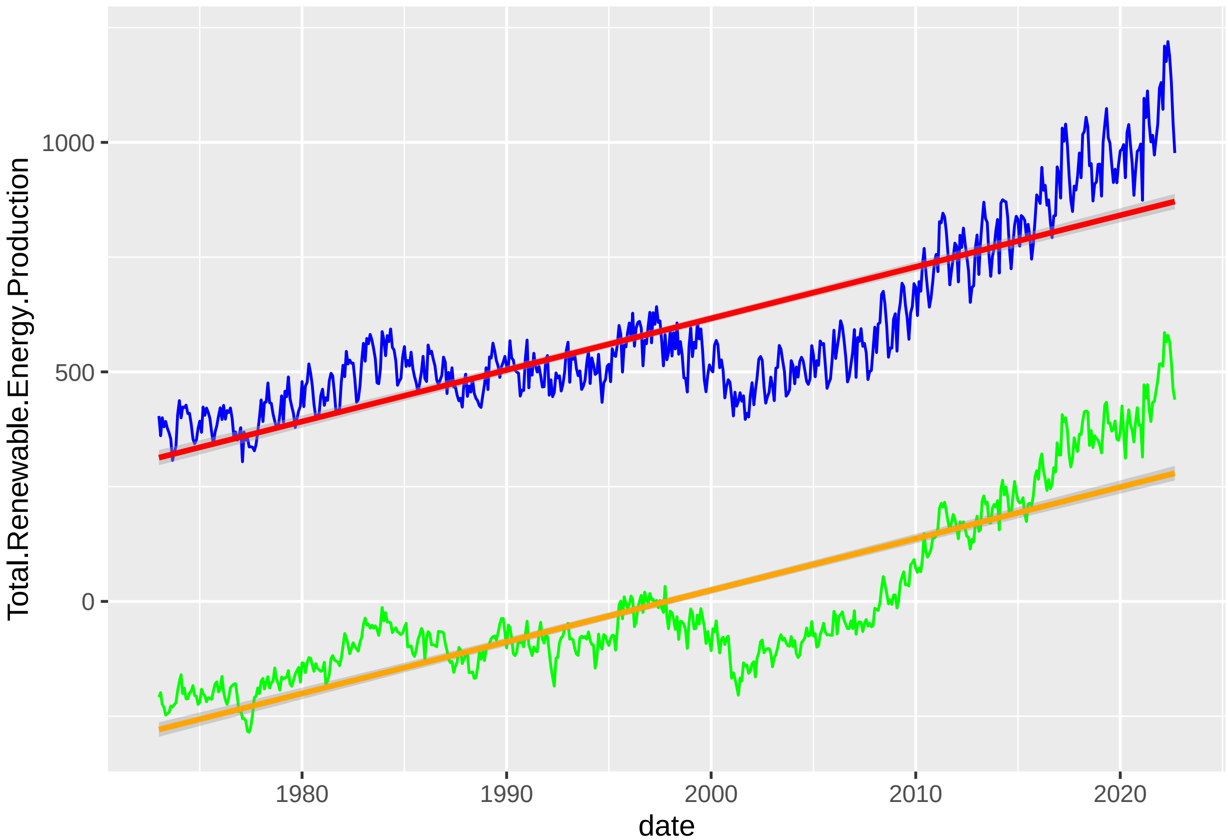
```
#compute seasonal component
energy_seas_comp_1=array(0,nrow(data.edit))
for(i in 1:nrow(data.edit)){
  energy_seas_comp_1[i]=(beta_int_1+beta_coeff_1%*%dummies_1[i,])
}

energy_seas_comp_2=array(0,nrow(data.edit))
for(i in 1:nrow(data.edit)){
  energy_seas_comp_2[i]=(beta_int_2+beta_coeff_2%*%dummies_2[i,])
}

energy_seas_comp_3=array(0,nrow(data.edit))
for(i in 1:nrow(data.edit)){
  energy_seas_comp_3[i]=(beta_int_3+beta_coeff_3%*%dummies_3[i,])
}

#Removing seasonal component
deseason_energy_data_1 <- data.edit[,2]-energy_seas_comp_1
deseason_energy_data_2 <- data.edit[,3]-energy_seas_comp_2
deseason_energy_data_3 <- data.edit[,4]-energy_seas_comp_3

deseason_data<-cbind(date,deseason_energy_data_1, deseason_energy_data_2, deseason_energy_data_3)


#Plot the deseasoned series'

ggplot(data.edit, aes(x=date, y=data.edit[,2])) +
            geom_line(color="blue") +
            ylab(column.name[1]) +
            #geom_abline(intercept = beta0, slope = beta1, color="red")
            geom_smooth(color="red",method="lm") +
            geom_line(aes(y=deseason_energy_data_1), col="green")+
            geom_smooth(aes(y=deseason_energy_data_1),color="orange",method="lm")
```

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



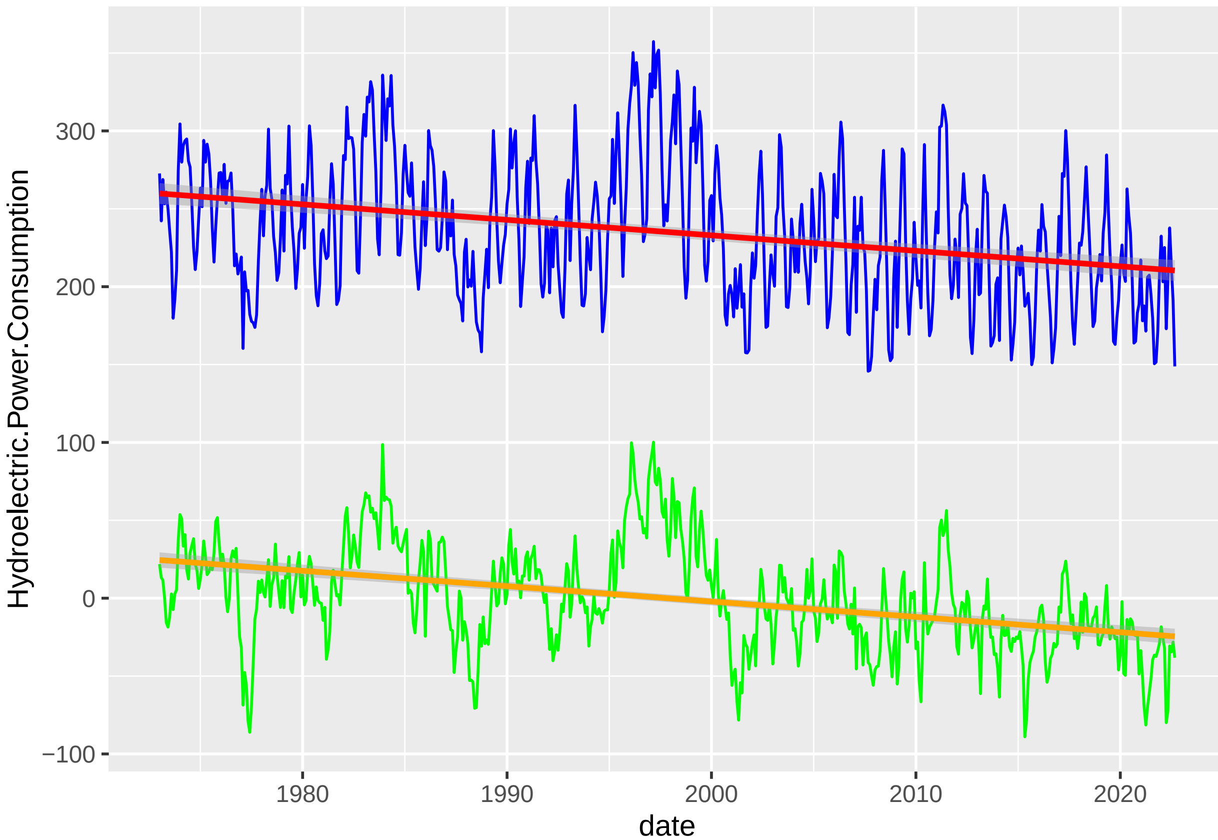

```
ggplot(data.edit, aes(x=date, y=data.edit[,3])) +
            geom_line(color="blue") +
            ylab(column.name[2]) +
            #geom_abline(intercept = beta0, slope = beta1, color="red")
            geom_smooth(color="red",method="lm") +
            geom_line(aes(y=deseason_energy_data_2), col="green")+
            geom_smooth(aes(y=deseason_energy_data_2),color="orange",method="lm")
```

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




```r
ggplot(data.edit, aes(x=date, y=data.edit[,4])) +
            geom_line(color="blue") +
            ylab(column.name[3]) +
            #geom_abline(intercept = beta0, slope = beta1, color="red")
            geom_smooth(color="red",method="lm") +
            geom_line(aes(y=deseason_energy_data_3), col="green")+
            geom_smooth(aes(y=deseason_energy_data_3),color="orange",method="lm")
```

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



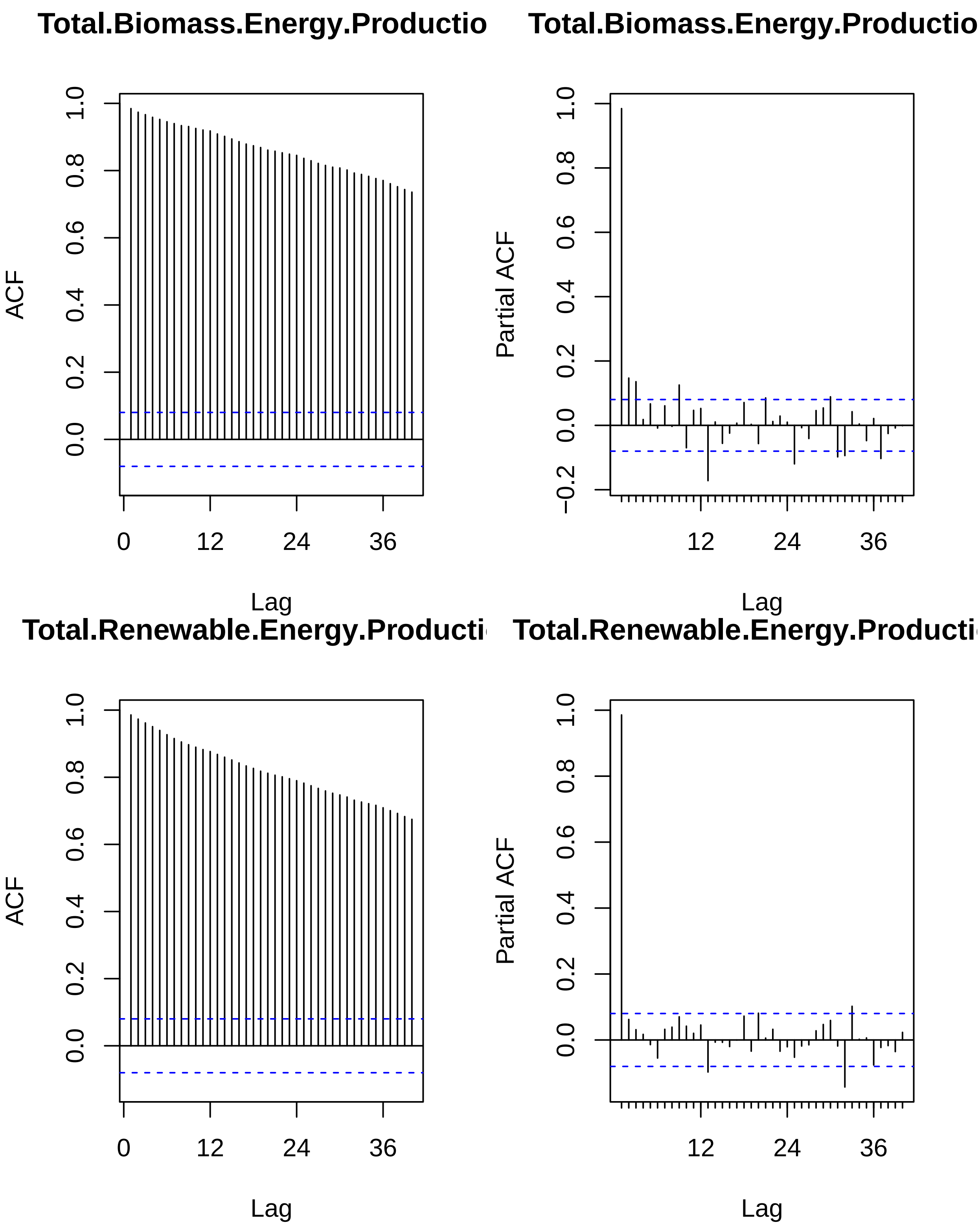
Q8

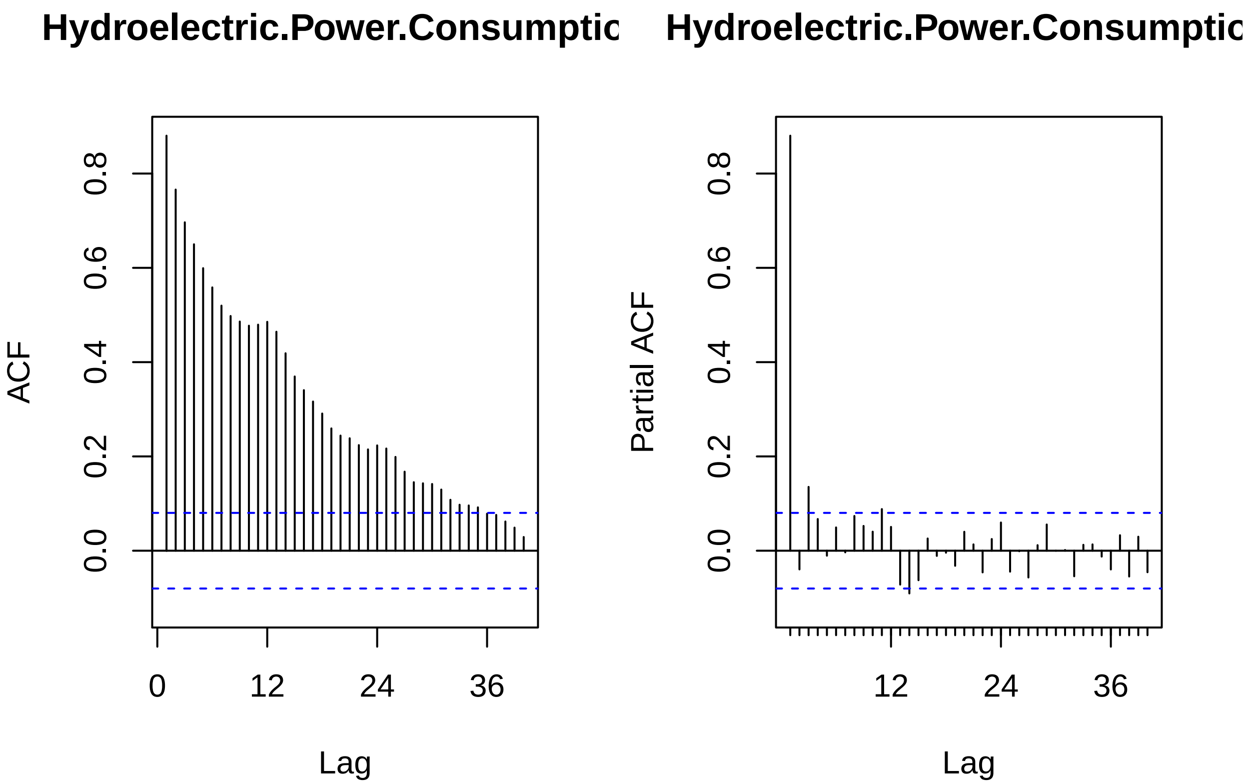
Plot ACF and PACF for the deseason series and compare with the plots from Q1. Did the plots change? How?

```
ts_energy_a3_deseason <- ts(deseason_data[,(2:4)], frequency=12, start=c(1973,1))
head(ts_energy_a3_deseason)
```

```
##          deseason_energy_data_1 deseason_energy_data_2 deseason_energy_data_3
## Jan 1973              -156.4400              -208.5094               21.88460
## Feb 1973              -139.5802              -198.6666               13.22808
## Mar 1973              -148.9780              -223.9915               11.60550
## Apr 1973              -140.8819              -230.5115                0.31160
## May 1973              -143.9480              -247.7338              -15.66446
## Jun 1973              -142.8072              -244.1686              -18.57516
```

```
par(mfrow=c(1,2))  #place plot side by side
for(i in 1:3){
  Acf(ts_energy_a3_deseason[,i],lag.max=40, main=column.name[i])
  Pacf(ts_energy_a3_deseason[,i],lag.max=40, main=column.name[i])
}
```


Total.Biomass.Energy.Productio
Total.Biomass.Energy.Productio
ACF
Partial ACF
Lag
Lag
Total.Renewable.Energy.Producti
Total.Renewable.Energy.Producti
ACF
Partial ACF
Lag
Lag



Answer: The first two plots - biomass and renewable energy - lost what little seasonality they had (shown in the ACF). The Hydro graph changed completed - now there's almost no seasonality at all! At least as displayed in the ACF graph. The PCF also - across all three - displayed fewer time points where there was significant auto-correlation.