ENV 790.30 - Time Series Analysis for Energy Data | Spring 2025

Assignment 3 - Due date 02/04/25

Nicole Leines

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_A03_Sp25.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_Consumpti
The data comes from the US Energy Information and Administration and corresponds to the December 2024 **Monthly** Energy Review. Once again you will work only with the following columns: Total Renewable Energy Production and Hydroelectric Power Consumption. Create a data frame structure with these two time series only.

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

```
#Load/install required package here
library(forecast)
library(tseries)
library(tidyverse)
library(readxl)
library(openxlsx)
library(ggplot2)
library(Kendall)
library(patchwork)
library(ggplotify)
library(cowplot)
```

Data

```r
#Importing data set
base_dir <- "D:/Geani/Box/Home Folder gnl13/Private/1 Academics/3 Time series/TSA_Sp25"
data_dir <- file.path(base_dir, "Data")
file_name <- "Table_10.1_Renewable_Energy_Production_and_Consumption_by_Source.xlsx"
file_path <- file.path(data_dir, file_name)

#Importing data set without change the original file using read.xlsx
energy_data1 <- read_excel(path=file_path,
                           skip = 12,
                           sheet="Monthly Data",
                           col_names=FALSE)

#Now let's extract the column names from row 11
read_col_names <- read_excel(path=file_path,
                             skip = 10,n_max = 1,
                             sheet="Monthly Data",
                             col_names=FALSE)

#Assign the column names to the data set
colnames(energy_data1) <- read_col_names

#Visualize the first rows of the data set
head(energy_data1)
```

```
## # A tibble: 6 x 14
##   Month               `Wood Energy Production` `Biofuels Production`
##   <dttm>                                 <dbl> <chr>
## 1 1973-01-01 00:00:00                     130. Not Available
## 2 1973-02-01 00:00:00                     117. Not Available
## 3 1973-03-01 00:00:00                     130. Not Available
## 4 1973-04-01 00:00:00                     125. Not Available
## 5 1973-05-01 00:00:00                     130. Not Available
## 6 1973-06-01 00:00:00                     125. Not Available
## # i 11 more variables: `Total Biomass Energy Production` <dbl>,
## #   `Total Renewable Energy Production` <dbl>,
## #   `Hydroelectric Power Consumption` <dbl>,
## #   `Geothermal Energy Consumption` <dbl>, `Solar Energy Consumption` <chr>,
## #   `Wind Energy Consumption` <chr>, `Wood Energy Consumption` <dbl>,
## #   `Waste Energy Consumption` <dbl>, `Biofuels Consumption` <chr>,
## #   `Total Biomass Energy Consumption` <dbl>, ...
```

```r
tail(energy_data1)
```

```
## # A tibble: 6 x 14
##   Month               `Wood Energy Production` `Biofuels Production`
##   <dttm>                                 <dbl> <chr>
## 1 2024-04-01 00:00:00                     163. 222.079
## 2 2024-05-01 00:00:00                     168. 231.668
## 3 2024-06-01 00:00:00                     160. 237.359
## 4 2024-07-01 00:00:00                     166. 251.826
```

```
## 5 2024-08-01 00:00:00                      172. 249.981
## 6 2024-09-01 00:00:00                      165. 234.89
## # i 11 more variables: `Total Biomass Energy Production` <dbl>,
## #   `Total Renewable Energy Production` <dbl>,
## #   `Hydroelectric Power Consumption` <dbl>,
## #   `Geothermal Energy Consumption` <dbl>, `Solar Energy Consumption` <chr>,
## #   `Wind Energy Consumption` <chr>, `Wood Energy Consumption` <dbl>,
## #   `Waste Energy Consumption` <dbl>, `Biofuels Consumption` <chr>,
## #   `Total Biomass Energy Consumption` <dbl>, ...
```

Time series

```
#Create a data frame structure with Total Renewable Energy Production and Hydroelectric Power Consumpti

column_names <- colnames(energy_data1)
print(column_names)
```

```
##  [1] "Month"                             "Wood Energy Production"
##  [3] "Biofuels Production"               "Total Biomass Energy Production"
##  [5] "Total Renewable Energy Production" "Hydroelectric Power Consumption"
##  [7] "Geothermal Energy Consumption"     "Solar Energy Consumption"
##  [9] "Wind Energy Consumption"           "Wood Energy Consumption"
## [11] "Waste Energy Consumption"          "Biofuels Consumption"
## [13] "Total Biomass Energy Consumption"  "Total Renewable Energy Consumption"
```

```
energy_data2 <- energy_data1 |>
  select("Month",
         "Total Renewable Energy Production",
         "Hydroelectric Power Consumption") |>
  mutate(Month = as.Date(Month, format = "%Y/%m/%d")) |>
  rename(date=Month)

summary(energy_data2)
```

```
##       date            Total Renewable Energy Production
##  Min.   :1973-01-01   Min.   :185.3
##  1st Qu.:1985-12-01   1st Qu.:310.7
##  Median :1998-11-01   Median :347.2
##  Mean   :1998-10-31   Mean   :402.0
##  3rd Qu.:2011-10-01   3rd Qu.:517.8
##  Max.   :2024-09-01   Max.   :771.5
##  Hydroelectric Power Consumption
##  Min.   : 49.02
##  1st Qu.: 68.98
##  Median : 78.59
##  Mean   : 79.55
##  3rd Qu.: 88.93
##  Max.   :119.40
```

```r
year1 <- year(energy_data2$date[1])
month1 <- month(energy_data2$date[1])

ts_energy_data2 <- ts(
  data = energy_data2[,2:3],
  start = c(year1,month1),
  frequency = 12
)

head(ts_energy_data2)
```

```
##          Total Renewable Energy Production Hydroelectric Power Consumption
## Jan 1973                           219.839                          89.562
## Feb 1973                           197.330                          79.544
## Mar 1973                           218.686                          88.284
## Apr 1973                           209.330                          83.152
## May 1973                           215.982                          85.643
## Jun 1973                           208.249                          82.060
```

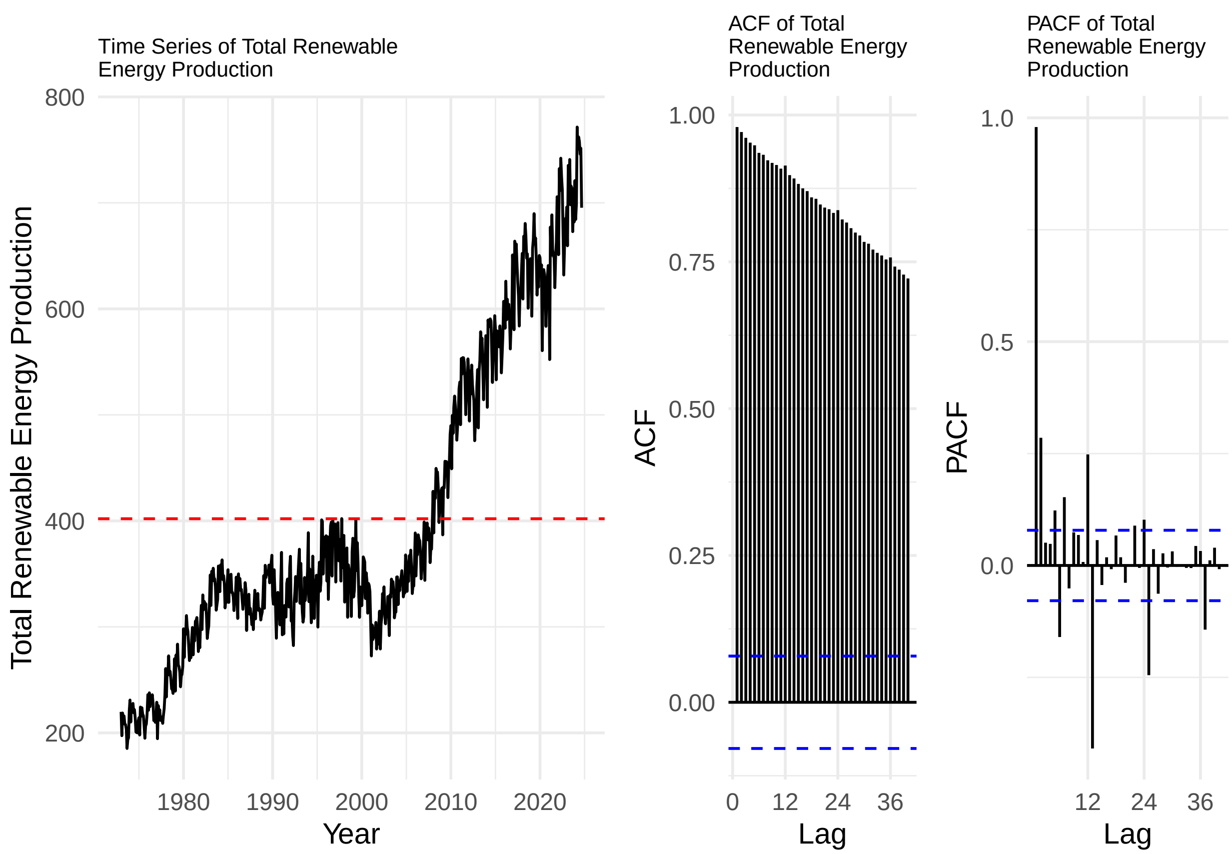
##Trend Component

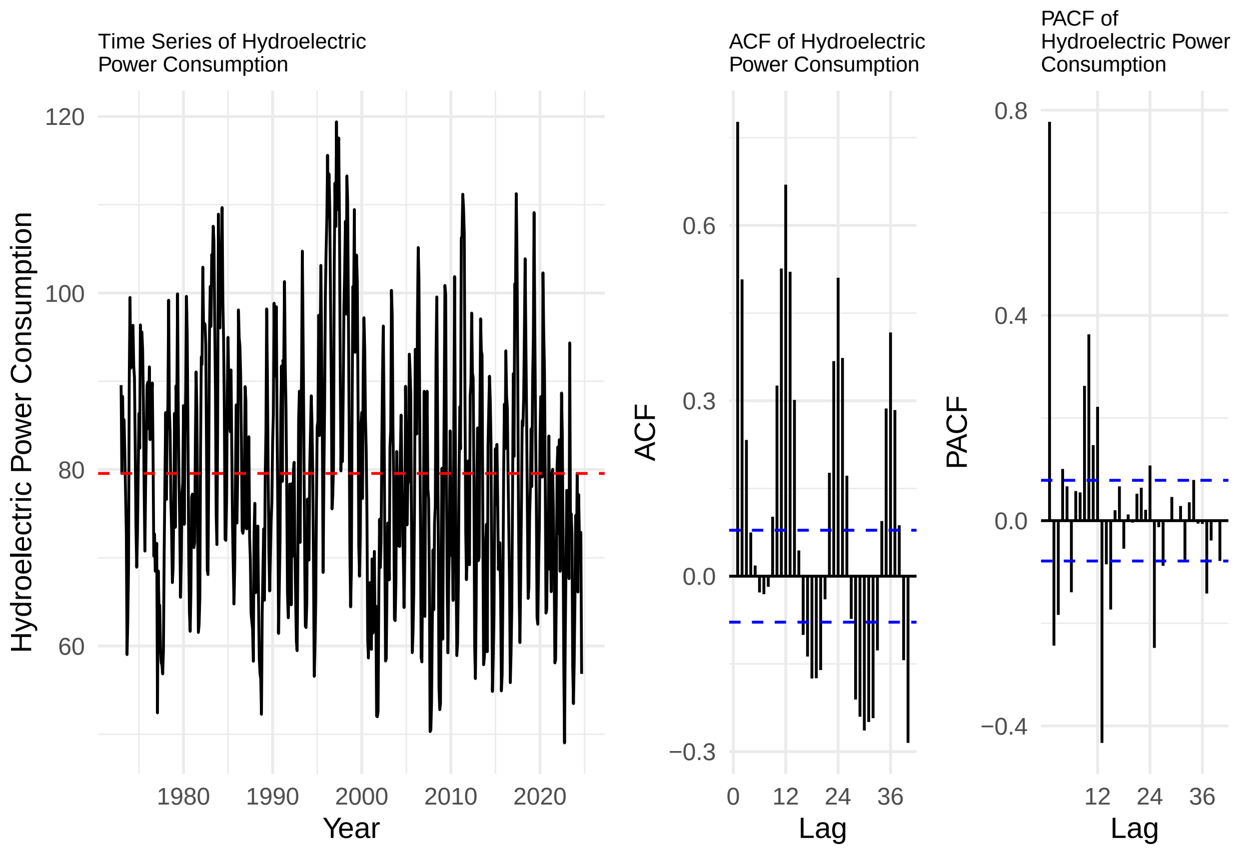
Q1

For each time series, i.e., Renewable Energy Production and Hydroelectric Consumption create three plots: one with time series, one with the ACF and with the PACF. You may use the some code form A2, but I want all the three plots side by side as in a grid. (Hint: use function `plot_grid()` from the `cowplot` package)

```r
nvars <- ncol(ts_energy_data2)

for (i in 1:nvars) {

  # 1. Time Series Plot
  p1 <- autoplot(ts_energy_data2[, i]) +
    geom_hline(yintercept = mean(ts_energy_data2[, i],
                                 na.rm = TRUE),
               color = "red",
               linetype = "dashed") +
    labs(title = str_wrap(paste("Time Series of",
                                colnames(ts_energy_data2)[i]),
                          width = 30),
         x = "Year", y = colnames(ts_energy_data2)[i]) +
    theme_minimal() +
    theme(plot.title = element_text(size = 8))

  # 2. ACF Plot
  p2 <- ggAcf(ts_energy_data2[, i], lag.max = 40) +
    labs(title = str_wrap(paste("ACF of", colnames(ts_energy_data2)[i]), width = 20)) +
    theme_minimal() +
    theme(plot.title = element_text(size = 8))

  # 3. PACF Plot
  p3 <- ggPacf(ts_energy_data2[, i], lag.max = 40) +
```

```r
    labs(title = str_wrap(paste("PACF of", colnames(ts_energy_data2)[i]), width = 20)) +
    theme_minimal() +
    theme(plot.title = element_text(size = 8))

  # Combine plots using plot_grid
  p4 <- plot_grid(p1, p2, p3,
                  ncol = 3,
                  align = "h",
                  axis = "tb",
                  rel_widths = c(2, 1, 1))

  print(p4)
}
```





Q2

From the plot in Q1, do the series Total Renewable Energy Production and Hydroelectric Power Consumption appear to have a trend? If yes, what kind of trend?

The series Renewable Energy Production shows a strong increasing trend but the Hydroelectric Power Consumption fluctuating around a constant mean with no strong upward or downward movement over time so no clear long-term trend.

On the other hand, the ACF for Renewable Energy Production shows very high positive autocorrelations that decay very slowly, starting near 1 and gradually declining but remaining significant even at lag 40 which is a clear indicator of a strong upward trend. While the ACF for Hydroelectric Power Consumption alternates positive and negative correlations that decay more quickly, so no exhibit a clear trend component.

Q3

Use the *lm()* function to fit a linear trend to the two 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.

```
nobs <- nrow(energy_data2)

# Create vector t
  t <- c(1:nobs)
```

```r
data_re <- data.frame("t"=t,
                      "renewable"=energy_data2$`Total Renewable Energy Production`)

#Fit a linear trend to TS of iHP
linear_trend_model_re=lm(renewable~t,data_re)
summary(linear_trend_model_re)
```

```
##
## Call:
## lm(formula = renewable ~ t, data = data_re)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -151.11  -37.84   13.53   41.76  149.42
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 176.87293    4.96189   35.65   <2e-16 ***
## t             0.72393    0.01382   52.37   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 61.75 on 619 degrees of freedom
## Multiple R-squared:  0.8159, Adjusted R-squared:  0.8156
## F-statistic:  2743 on 1 and 619 DF,  p-value: < 2.2e-16
```

```r
re_beta0=as.numeric(linear_trend_model_re$coefficients[1])
re_beta1=as.numeric(linear_trend_model_re$coefficients[2])
```

The intercept has a value of 176.87 trillion btu at the beginning of the dataset, and it is statistically significant. The slope t is positive with a value lower than 1, specifically 0.72 and it is also statistically significant. In this context, there was a consistent upward trend in renewable energy production over time. So, for each month, the Total Renewable Energy Production increases by approximately 0.72 trillion btu on average.

```r
data_hy <- data.frame("t"=t,"hydroelectric"=energy_data2$`Hydroelectric Power Consumption`)

linear_trend_model_hy=lm(hydroelectric~t,data_hy)
summary(linear_trend_model_hy)
```

```
##
## Call:
## lm(formula = hydroelectric ~ t, data = data_hy)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -29.995 -10.422  -0.720   9.161  39.624
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 82.96766    1.12339  73.855  < 2e-16 ***
## t           -0.01098    0.00313  -3.508 0.000485 ***
## ---
```

```
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 13.98 on 619 degrees of freedom
## Multiple R-squared:  0.01949,    Adjusted R-squared:  0.01791
## F-statistic:  12.3 on 1 and 619 DF,  p-value: 0.0004848
```

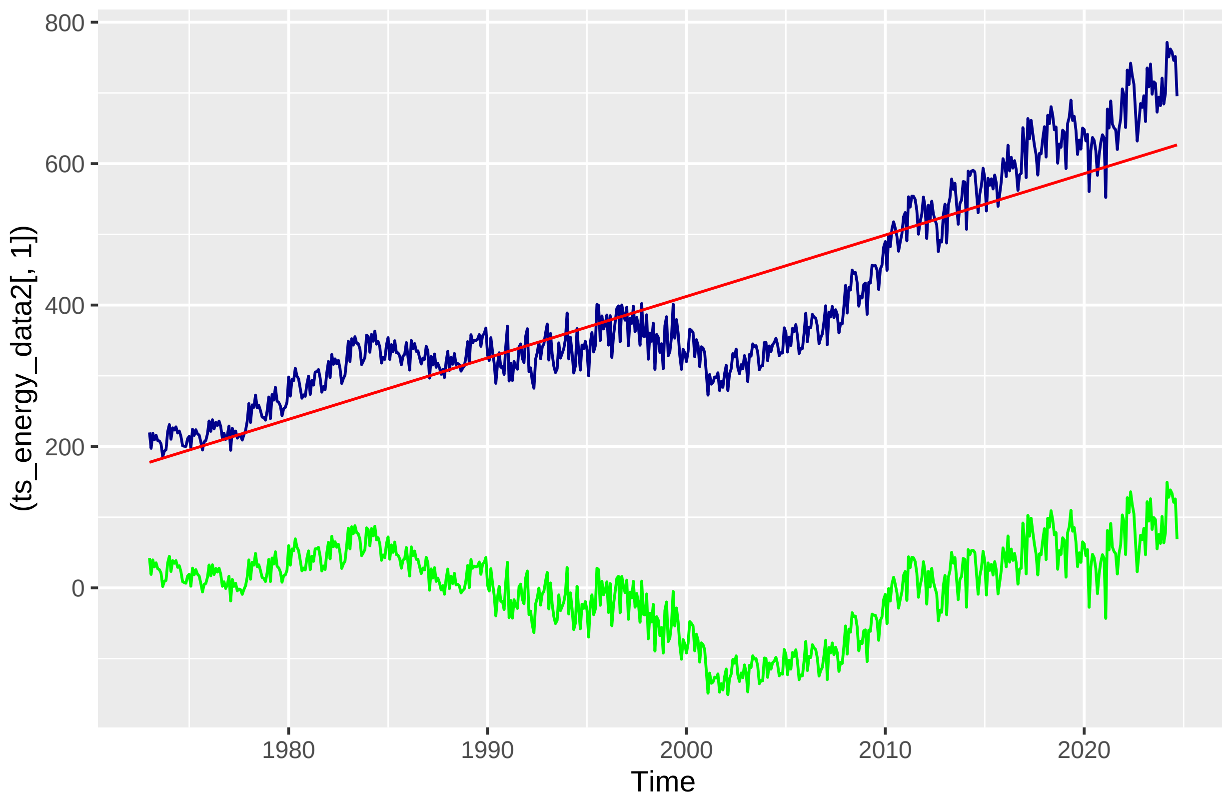
```r
hy_beta0=as.numeric(linear_trend_model_hy$coefficients[1])
hy_beta1=as.numeric(linear_trend_model_hy$coefficients[2])
```

The intercept has a value of 82.96 trillion btu at the beginning of the dataset, and it is statistically significant. The slope t is negative with a value close to 0, specifically -0.01 and it is also statistically significant. In this context, there was a consistent negative trend in Hydroelectric Power Consumption over time. So, for each month, the Hydroelectric Power Consumption decrease by approximately 0.01 trillion btu on average. However, only 1.95% of the variation in Hydroelectric Power Consumption is explained by time, so time alone does not explain most of its fluctuations.

Q4

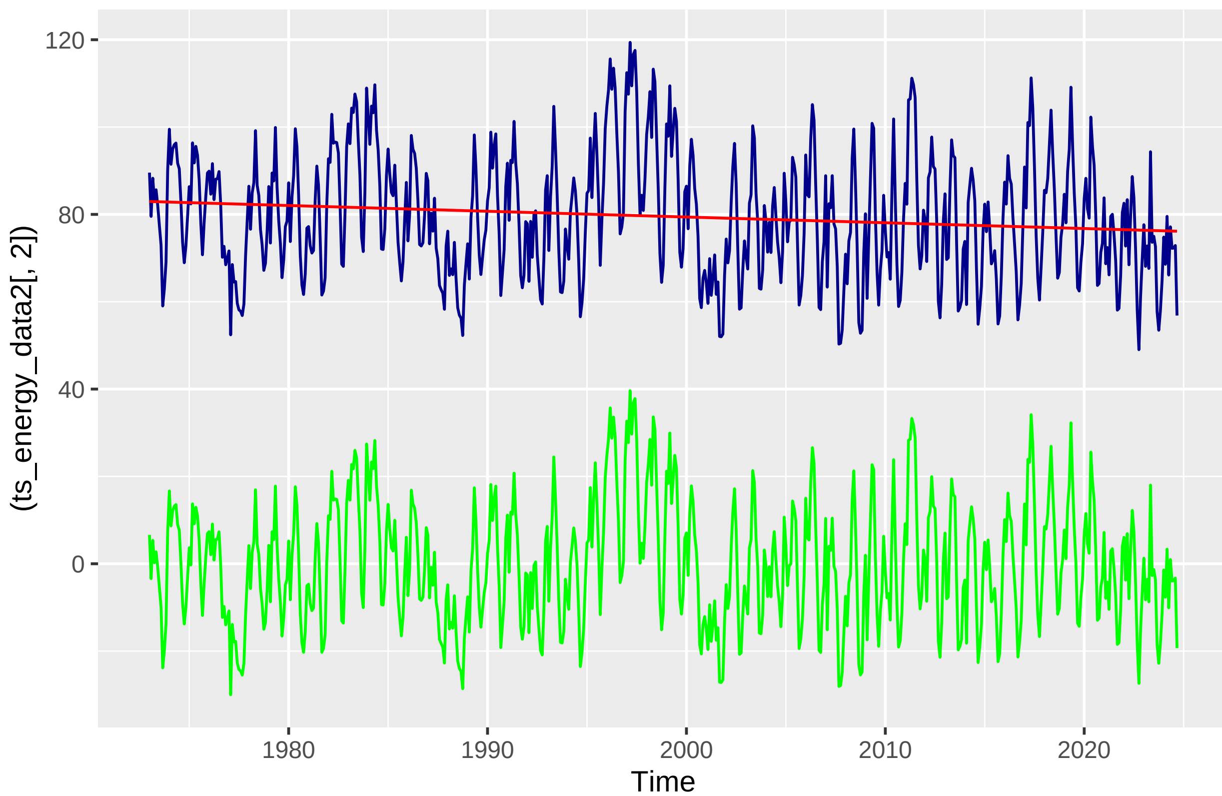
Use the regression coefficients from Q3 to detrend the series. Plot the detrended series and compare with the plots from Q1. What happened? Did anything change?

```r
#Create detrended series from linear trend

linear_trend_re <- re_beta0 + re_beta1*t
ts_linear_trend_re<-ts(linear_trend_re,star=c(year1,month1),frequency=12)

detrend_re<-energy_data2$`Total Renewable Energy Production`- linear_trend_re
ts_detrend_re<-ts(detrend_re, start = c(year1,month1), frequency = 12)

autoplot((ts_energy_data2[,1]), color="darkblue")+
  autolayer(ts_detrend_re, series="Detrended",color="green")+
  autolayer(ts_linear_trend_re,series="Linear Component",color="red")
```



Comparing with Q1, the detrended series fluctuates around 0, indicating the variations that remain after removing the trend. The detrended series removes the increasing pattern and shows a cyclical behavior. So, it definitely changed because there is no more a steady, consistent increase over time.

```r
#Create detrended series from linear trend

linear_trend_hy <- (hy_beta0+hy_beta1*t)
ts_linear_trend_hy<-ts(linear_trend_hy,star=c(year1,month1),frequency=12)

detrend_hy<-energy_data2$`Hydroelectric Power Consumption`- linear_trend_hy
ts_detrend_hy<-ts(detrend_hy, start = c(year1,month1), frequency = 12)

autoplot((ts_energy_data2[,2]), color="darkblue")+
  autolayer(ts_detrend_hy, series="Detrended",color="green")+
  autolayer(ts_linear_trend_hy,series="Linear Component",color="red")
```



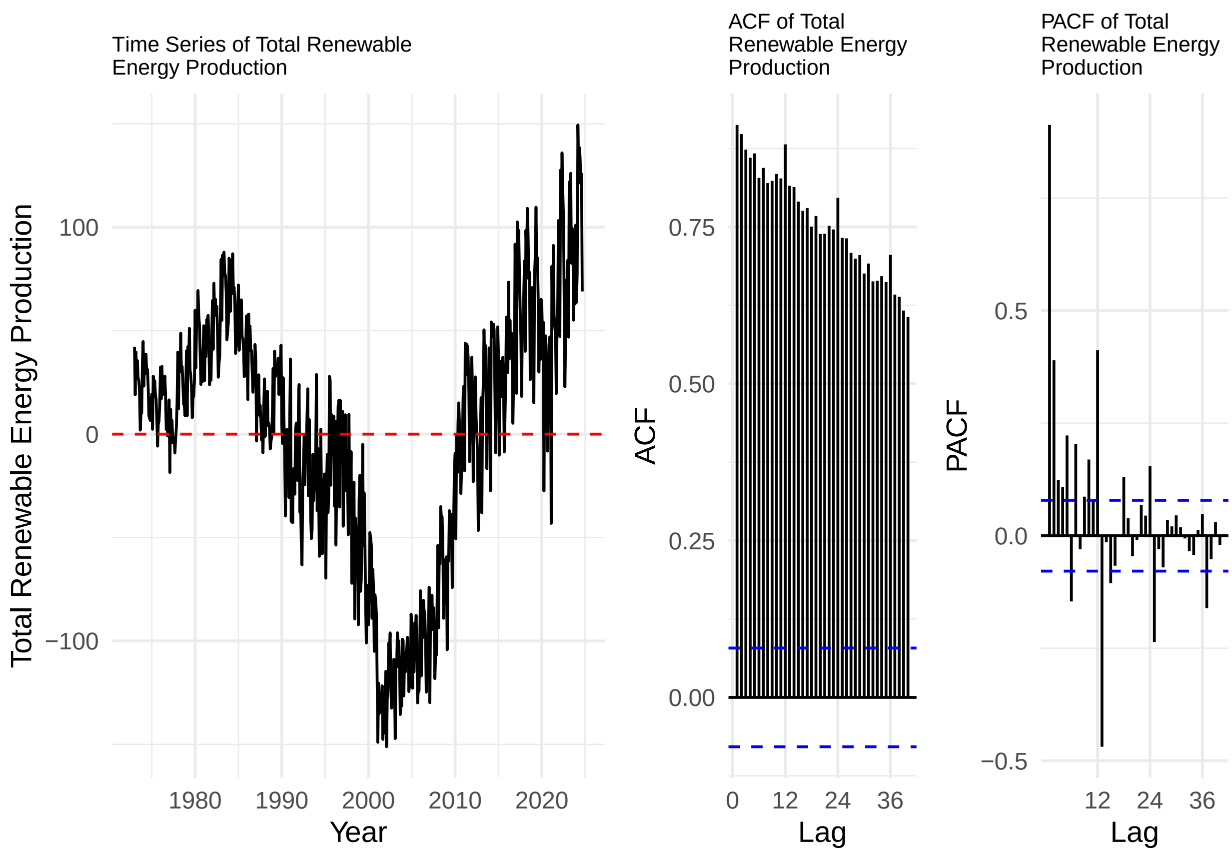
Comparing with Q1, the detrended series fluctuates around 0, indicating the variations that remain after removing the trend. However, it still resembles the original time series reflecting the fact that there was a minimal downward trend. So, in this case, it almost do not change because there was a minimal trend.

Q5

Plot ACF and PACF for the detrended series and compare with the plots from Q1. You may use plot_grid() again to get them side by side, but not mandatory. Did the plots change? How?

```
# 1. Time Series Plot
p5 <- autoplot(ts_detrend_re) +
  geom_hline(yintercept = mean(ts_detrend_re,
                               na.rm = TRUE),
             color = "red",
             linetype = "dashed") +
  labs(title = str_wrap(paste("Time Series of",
                              colnames(ts_energy_data2)[1]),
                        width = 30),
       x = "Year", y = colnames(ts_energy_data2)[1]) +
  theme_minimal() +
  theme(plot.title = element_text(size = 8))

# 2. ACF Plot
p6 <- ggAcf(ts_detrend_re, lag.max = 40) +
  labs(title = str_wrap(paste("ACF of", colnames(ts_energy_data2)[1]), width = 20)) +
  theme_minimal() +
```

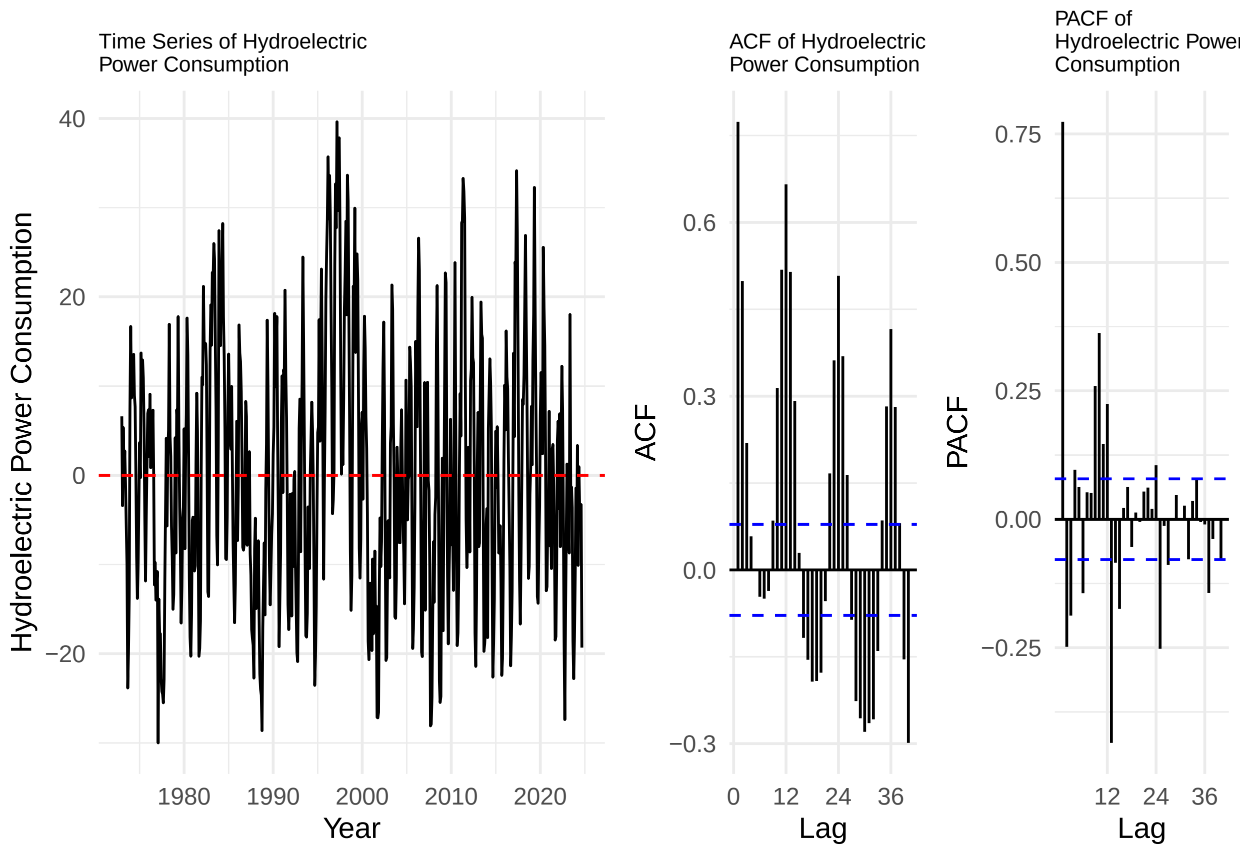
```r
  theme(plot.title = element_text(size = 8))

# 3. PACF Plot
p7 <- ggPacf(ts_detrend_re, lag.max = 40) +
  labs(title = str_wrap(paste("PACF of", colnames(ts_energy_data2)[1]), width = 20)) +
  theme_minimal() +
  theme(plot.title = element_text(size = 8))

# Combine plots using plot_grid
p8 <- plot_grid(p5, p6, p7,
                       ncol = 3,
                       align = "h",
                       axis = "tb",
                       rel_widths = c(2, 1, 1))

print(p8)
```



Comparing with Q1, the ACF for Renewable Energy Production still shows a positive autocorrelation but it decays quicker, remaining significant but a lower level around 0.6 at lag 40 which is a clear indicator of a successfully detrending process. The detrended PACF still shows correlations after detrending, specially at 13,25 and 37 that can suggest seasonal patterns.

```r
# 1. Time Series Plot
p9 <- autoplot(ts_detrend_hy) +
  geom_hline(yintercept = mean(ts_detrend_hy,
                               na.rm = TRUE),
```

```
             color = "red",
             linetype = "dashed") +
  labs(title = str_wrap(paste("Time Series of",
                              colnames(ts_energy_data2)[2]),
                        width = 30),
       x = "Year", y = colnames(ts_energy_data2)[2]) +
  theme_minimal() +
  theme(plot.title = element_text(size = 8))

# 2. ACF Plot
p10 <- ggAcf(ts_detrend_hy, lag.max = 40) +
  labs(title = str_wrap(paste("ACF of", colnames(ts_energy_data2)[2]), width = 20)) +
  theme_minimal() +
  theme(plot.title = element_text(size = 8))

# 3. PACF Plot
p11 <- ggPacf(ts_detrend_hy, lag.max = 40) +
  labs(title = str_wrap(paste("PACF of", colnames(ts_energy_data2)[2]), width = 20)) +
  theme_minimal() +
  theme(plot.title = element_text(size = 8))

# Combine plots using plot_grid
p12 <- plot_grid(p9, p10, p11,
                       ncol = 3,
                       align = "h",
                       axis = "tb",
                       rel_widths = c(2, 1, 1))

print(p12)
```



Comparing with Q1, the main change seems to be the scale that changes from 60-120 units in Q1 to -20 to 40 units in the detrended series. However, the detrending process has not changed significantly the ACF pattern for Hydroelectric Power Consumption. Same for the PACF which still shows correlations after detrending that can suggest seasonal patterns.

Seasonal Component

Set aside the detrended series and consider the original series again from Q1 to answer Q6 to Q8.

Q6

Just by looking at the time series and the acf plots, do the series seem to have a seasonal trend? No need to run any code to answer your question. Just type in you answer below.

By only looking at the time series and the acf plots, the Renewable Energy Production doesn't seem to have a seasonal trend, but the Hydroelectric Power Consumption does. Specifically, the ACF plot for the Hydroelectric Power Consumption shows peaks around every 12 lags which suggests a strong seasonal component. The seasonal component for the Renewable Energy Production seems noticeable in the PACF but not in the ACF.

Q7

Use function *lm()* to fit a seasonal means model (i.e. using the seasonal dummies) the two time series. Ask R to print the summary of the regression. Interpret the regression output. From the results which series have a seasonal trend? Do the results match you answer to Q6?

```r
#First create the seasonal dummies
dummies_re <- seasonaldummy(ts_detrend_re) #this function only accepts ts object

#Then fit a linear model to the seasonal dummies
seas_means_model_re=lm(detrend_re~dummies_re)
summary(seas_means_model_re)
```

```
##
## Call:
## lm(formula = detrend_re ~ dummies_re)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -149.18  -38.16   14.42   41.50  134.67
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)      7.858      8.504   0.924  0.35584
## dummies_reJan    5.592     11.968   0.467  0.64048
## dummies_reFeb  -31.452     11.968  -2.628  0.00881 **
## dummies_reMar    6.892     11.968   0.576  0.56491
## dummies_reApr   -6.449     11.968  -0.539  0.59023
## dummies_reMay    7.923     11.968   0.662  0.50822
## dummies_reJun   -3.394     11.968  -0.284  0.77682
## dummies_reJul    2.126     11.968   0.178  0.85906
## dummies_reAug   -5.878     11.968  -0.491  0.62351
## dummies_reSep  -31.209     11.968  -2.608  0.00934 **
## dummies_reOct  -18.757     12.026  -1.560  0.11937
## dummies_reNov  -19.982     12.026  -1.661  0.09713 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 60.73 on 609 degrees of freedom
## Multiple R-squared:  0.04839,    Adjusted R-squared:  0.0312
## F-statistic: 2.815 on 11 and 609 DF,  p-value: 0.001358
```

```r
#Store regression coefficients
beta_int_re=seas_means_model_re$coefficients[1]
beta_coeff_re=seas_means_model_re$coefficients[2:12]
```

For the Renewable Energy Production, only February and September show statistically significant seasonality and it has a low R-squared that indicates seasonal dummies explain only about 4.8% of variation.

```r
#First create the seasonal dummies
dummies_hy <- seasonaldummy(ts_detrend_hy) #this function only accepts ts object

#Then fit a linear model to the seasonal dummies
seas_means_model_hy=lm(detrend_hy~dummies_hy)
summary(seas_means_model_hy)
```

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

```
## lm(formula = detrend_hy ~ dummies_hy)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -33.933  -5.798  -0.531   5.721  32.166
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)     0.4379     1.4258   0.307 0.758849
## dummies_hyJan   4.8863     2.0067   2.435 0.015177 *
## dummies_hyFeb  -2.5567     2.0067  -1.274 0.203116
## dummies_hyMar   7.0202     2.0067   3.498 0.000502 ***
## dummies_hyApr   5.3770     2.0067   2.680 0.007572 **
## dummies_hyMay  13.8957     2.0067   6.925 1.11e-11 ***
## dummies_hyJun  10.7293     2.0067   5.347 1.27e-07 ***
## dummies_hyJul   4.0439     2.0067   2.015 0.044320 *
## dummies_hyAug  -5.3775     2.0067  -2.680 0.007566 **
## dummies_hySep -16.5635     2.0067  -8.254 9.51e-16 ***
## dummies_hyOct -16.3915     2.0164  -8.129 2.43e-15 ***
## dummies_hyNov -10.8163     2.0164  -5.364 1.16e-07 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 10.18 on 609 degrees of freedom
## Multiple R-squared:  0.4781, Adjusted R-squared:  0.4687
## F-statistic: 50.72 on 11 and 609 DF,  p-value: < 2.2e-16
```

```
#Store regression coefficients
beta_int_hy=seas_means_model_hy$coefficients[1]
beta_coeff_hy=seas_means_model_hy$coefficients[2:12]
```

On the other hand, for the Hydroelectric Power Consumption, 6 months are highly significant: March, May, June, September, October, November. It also has a R-squared that indicates seasonal dummies explain about 48% of variation.

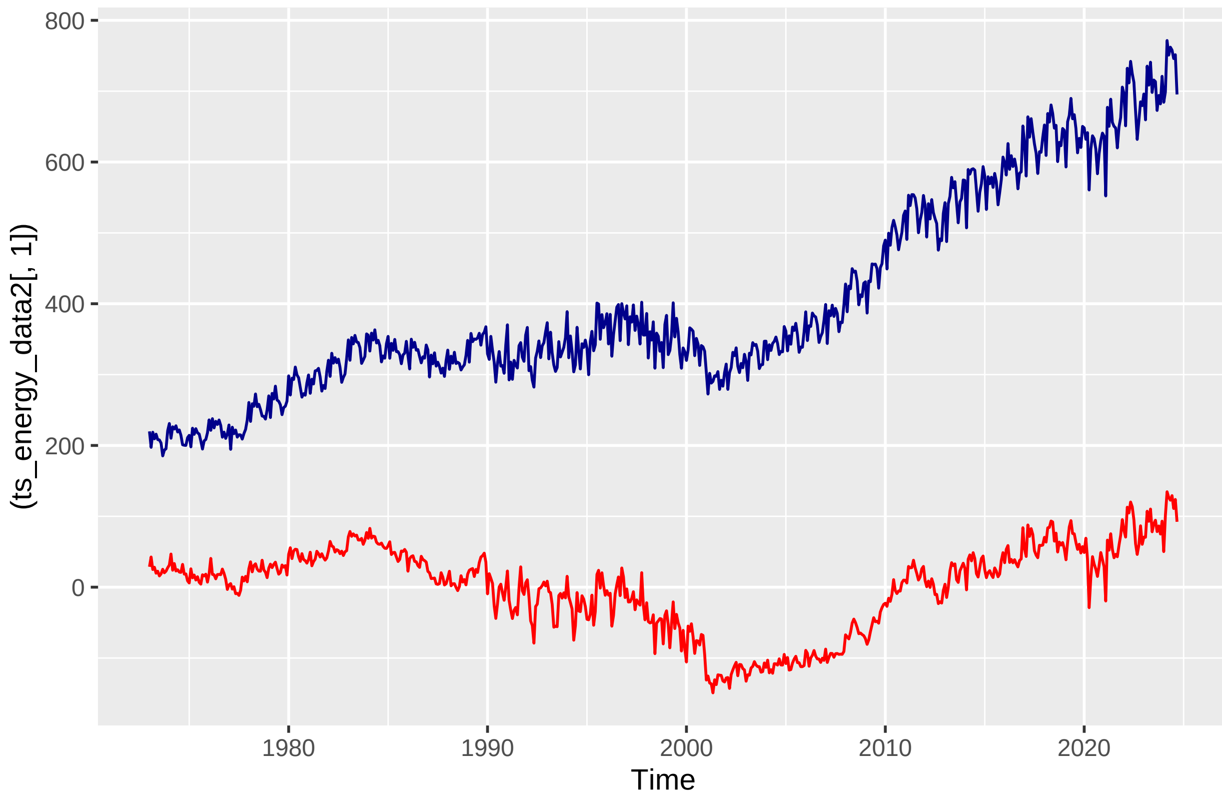
Based on those results and comparing with Q6, Hydroelectric Power Consumption clearly shows a much stronger seasonal trend than Total Renewable Energy Production as was established in Q6.

Q8

Use the regression coefficients from Q7 to deseason the series. Plot the deseason series and compare with the plots from part Q1. Did anything change?

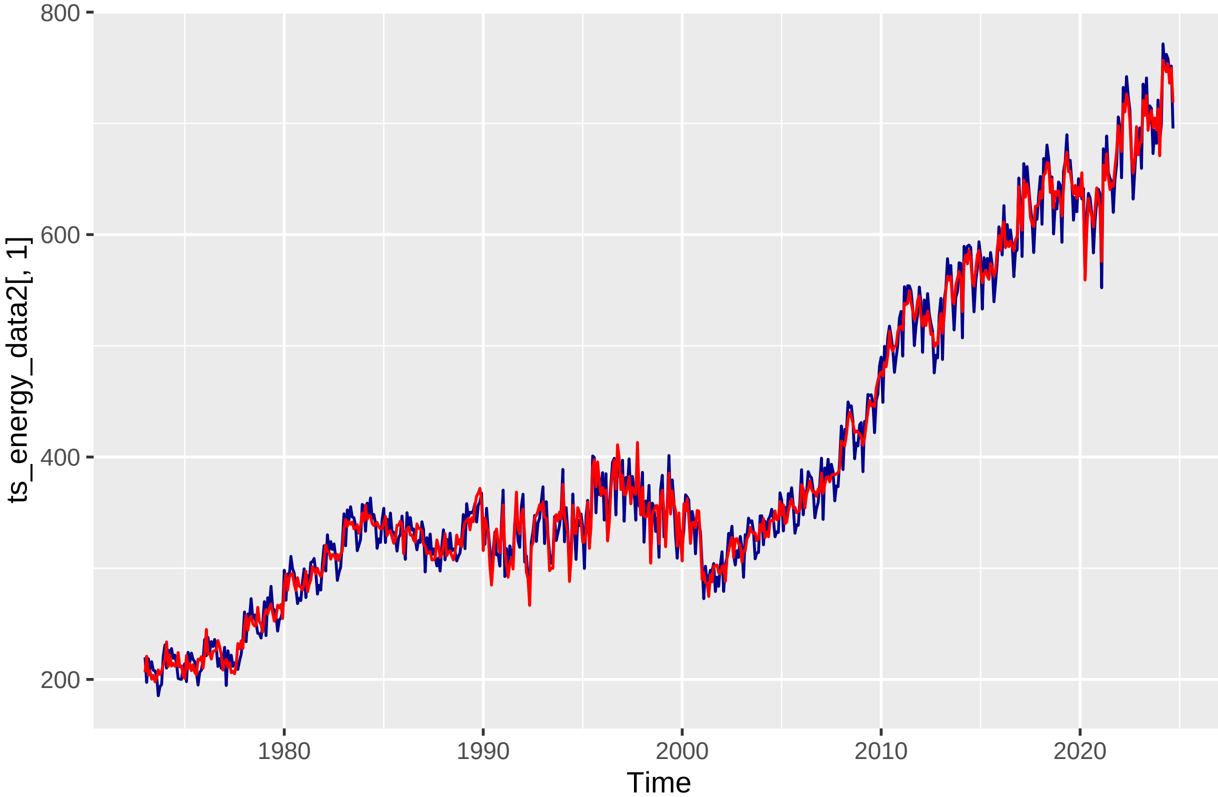
```
#compute seasonal component
seas_comp_re=array(0,nobs)

for(i in 1:nobs){
seas_comp_re[i]=(beta_int_re + beta_coeff_re %*% dummies_re[i,])
}


deseason_re <- detrend_re - seas_comp_re
```

```r
#Transform into a ts object
ts_deseason_re <- ts(deseason_re,start=c(year1,month1),frequency=12)

#Plot
autoplot((ts_energy_data2[,1]), color="darkblue") +
  autolayer(ts_deseason_re,series="Seasonal Component",color="red")
```

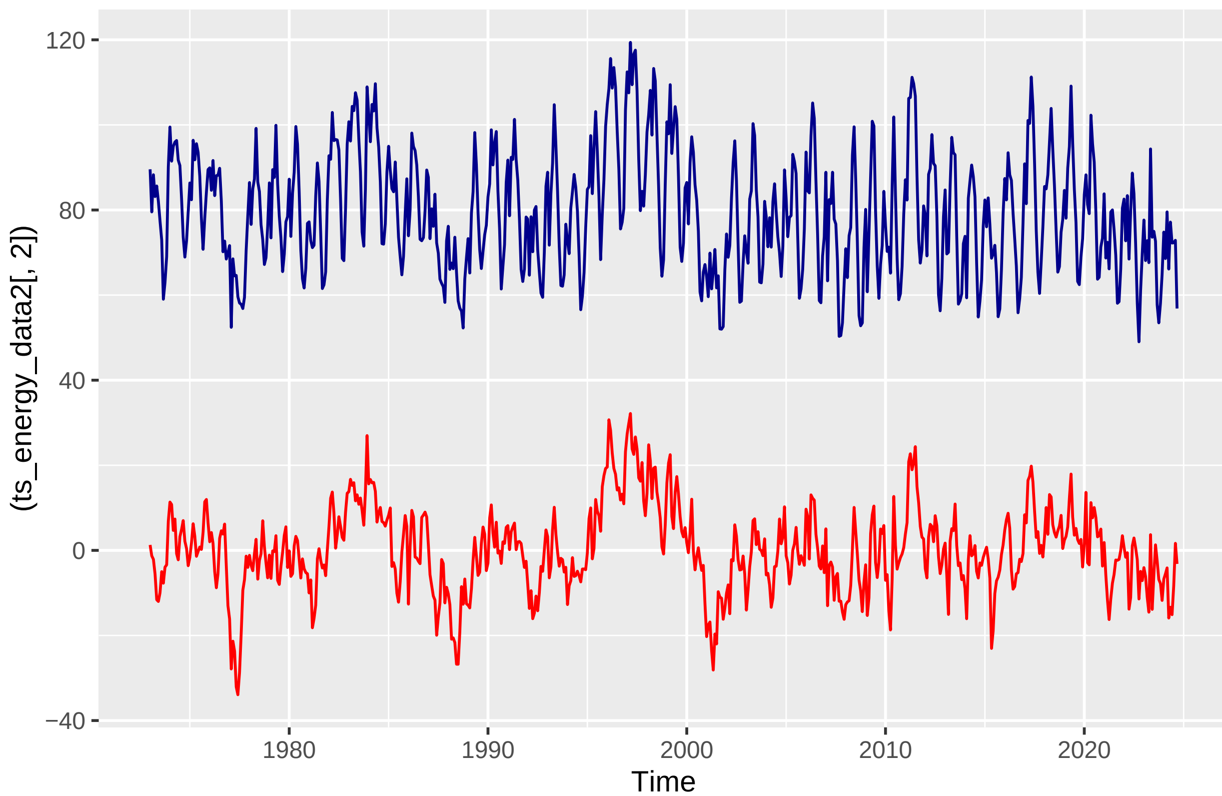


It seems that nothing changed.

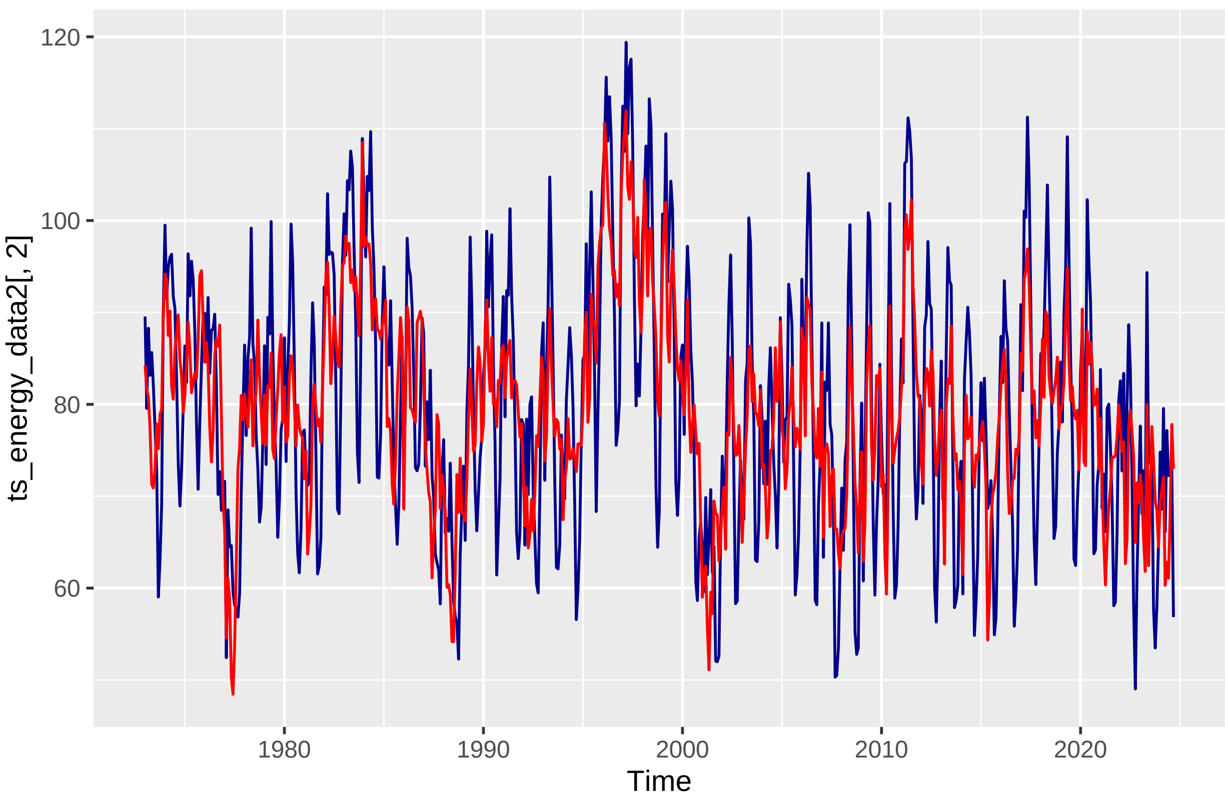
```r
deseason_linear_re <- energy_data2$`Total Renewable Energy Production`-seas_comp_re
ts_deseason_linear_re <- ts(deseason_linear_re,start=c(year1,month1),frequency=12)

autoplot(ts_energy_data2[,1], col="darkblue") +
autolayer(ts_deseason_linear_re,series="Deseason Series",color="red")
```




```r
#compute seasonal component
seas_comp_hy=array(0,nobs)

for(i in 1:nobs){
seas_comp_hy[i]=(beta_int_hy + beta_coeff_hy %*% dummies_hy[i,])
}

deseason_hy <- detrend_hy - seas_comp_hy

#Transform into a ts object
ts_deseason_hy <- ts(deseason_hy,start=c(year1,month1),frequency=12)


#Plot
autoplot((ts_energy_data2[,2]), color="darkblue") +
  autolayer(ts_deseason_hy,series="Seasonal Component",color="red")
```




```
deseason_linear_hy <- energy_data2$`Hydroelectric Power Consumption`-seas_comp_hy
ts_deseason_linear_hy <- ts(deseason_linear_hy,start=c(year1,month1),frequency=12)

autoplot(ts_energy_data2[,2], col="darkblue") +
autolayer(ts_deseason_linear_hy,series="Deseason Series",color="red")
```



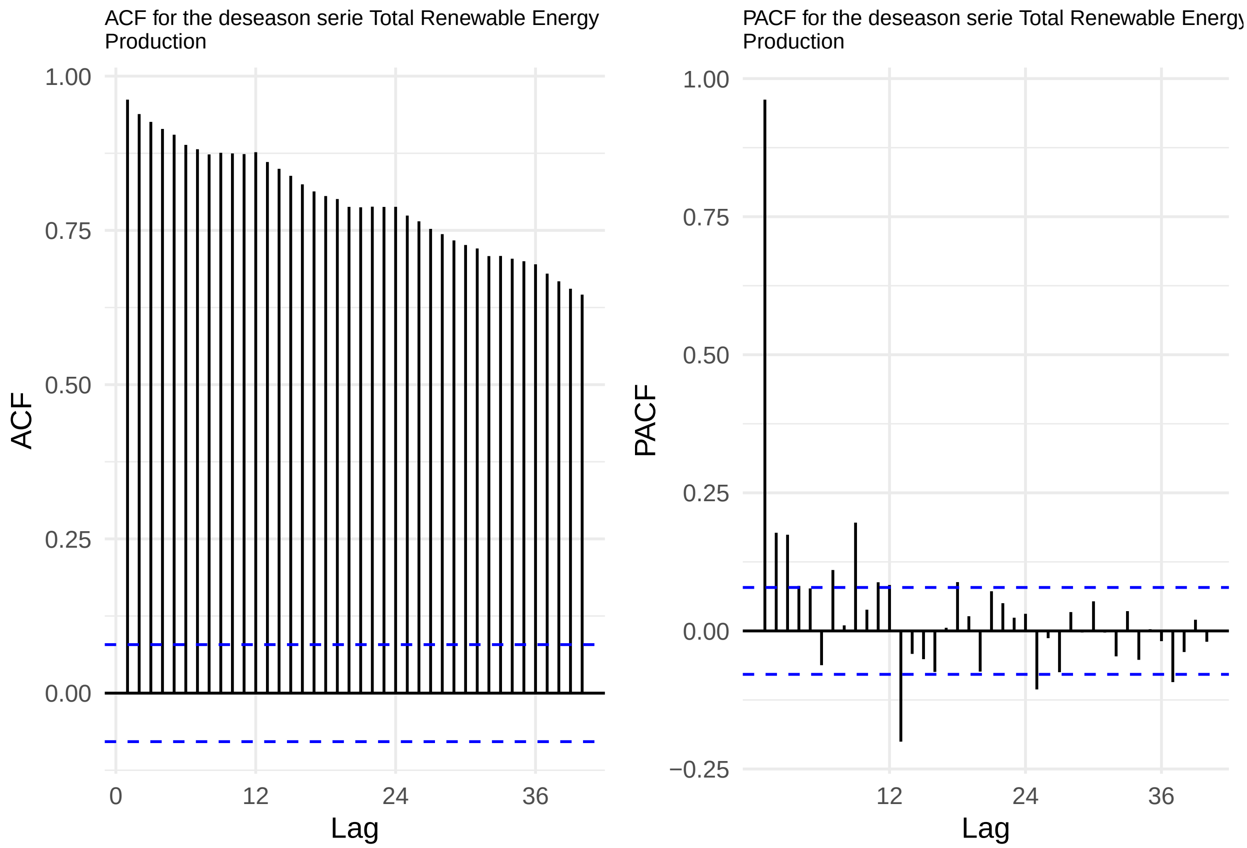
Comparing with Q1, deseasoning had minimal impact on Total Renewable Energy Production, confirming the weak seasonality but deseasoning the Hydroelectric Power Consumption reduced the distance of fluctuations which confirm strong seasonal patterns.

Q9

Plot ACF and PACF for the deseason series and compare with the plots from Q1. You may use plot_grid() again to get them side by side, but not mandatory. Did the plots change? How?

```
# ACF Plot
p13 <- ggAcf(ts_deseason_re, lag.max = 40) +
  labs(title = str_wrap(paste("ACF for the deseason serie", colnames(ts_energy_data2)[1]), width = 50
  theme_minimal() +
  theme(plot.title = element_text(size = 8))

# 3. PACF Plot
p14 <- ggPacf(ts_deseason_re, lag.max = 40) +
  labs(title = str_wrap(paste("PACF for the deseason serie", colnames(ts_energy_data2)[1]), width = 5
  theme_minimal() +
  theme(plot.title = element_text(size = 8))

# Combine plots using plot_grid
p15 <- plot_grid(p13, p14,
                       ncol = 2,
                       align = "h",
                       axis = "tb",
```

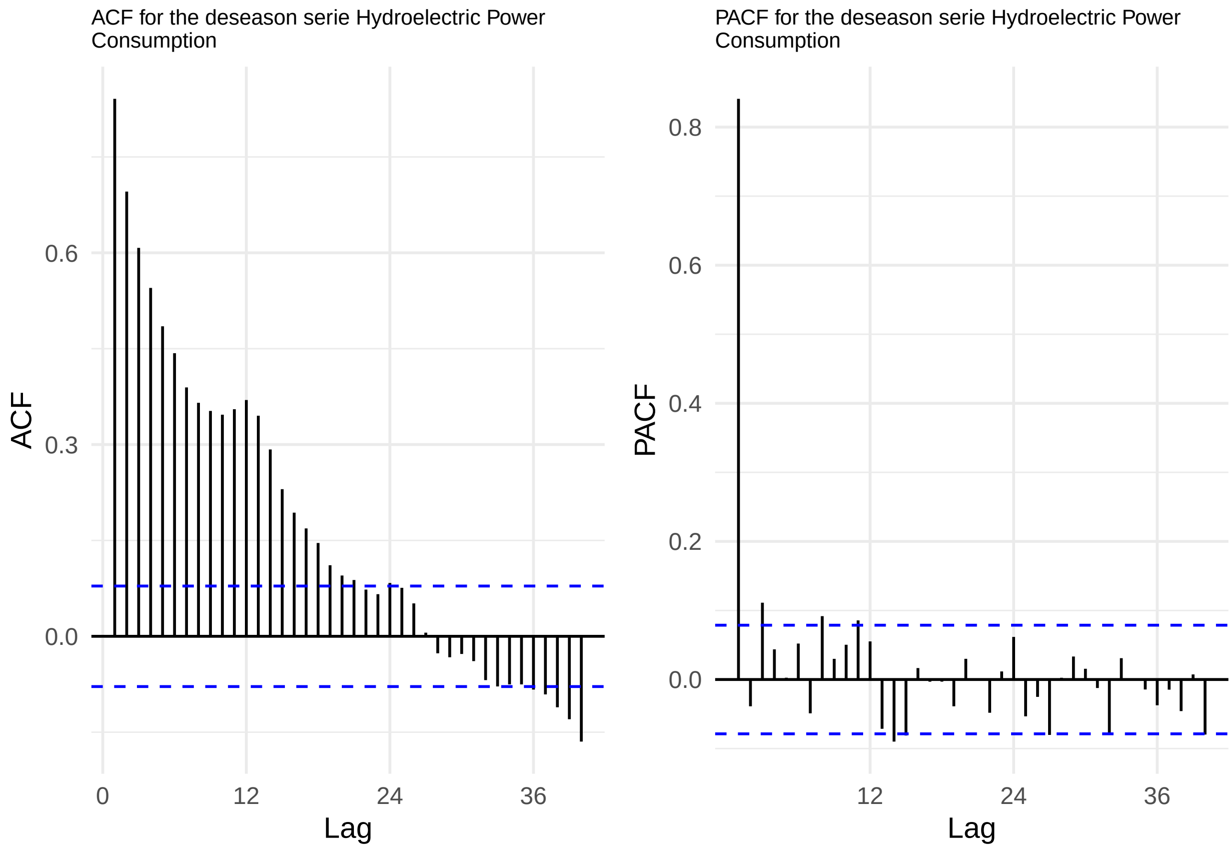
```r
                         rel_widths = c(1, 1))

print(p15)
```



For the Renewal Energy Production, the ACF still shows very slow decay pattern, indicating the trend component remains strong even after deseasoning. However, it is noticeable that PACF shows fewer significant spikes compared to Q1. So, now the most highly correlated in the PACF is the first lag, and other lags show much smaller correlations.

```r
  # ACF Plot
  p16 <- ggAcf(ts_deseason_hy, lag.max = 40) +
  labs(title = str_wrap(paste("ACF for the deseason serie",
                              colnames(ts_energy_data2)[2]), width = 50)) +
    theme_minimal() +
    theme(plot.title = element_text(size = 8))

  # 3. PACF Plot
  p17 <- ggPacf(ts_deseason_hy, lag.max = 40) +
    labs(title = str_wrap(paste("PACF for the deseason serie", colnames(ts_energy_data2)[2]), width = 5
    theme_minimal() +
    theme(plot.title = element_text(size = 8))

  # Combine plots using plot_grid
  p18 <- plot_grid(p16, p17,
                         ncol = 2,
                         align = "h",
```

```r
                          axis = "tb",
                          rel_widths = c(1, 1))

print(p18)
```



For the Hydroelectric Power Consumption, ACF shows much faster decay compared to Q1 and the regular seasonal spikes at lag 12 and multiples are now gone. It is also noticeable that PACF shows fewer significant spikes compared to Q1. So, now the most highly correlated in the PACF is the first lag, and other lags show very small or insignificant correlations.