ENV 790.30 - Time Series Analysis for Energy Data | Spring 2024

Assignment 3 - Due date 02/01/24

David Robinson

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_Sp24.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 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)
```

```
## Warning: package 'forecast' was built under R version 4.3.2
```

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

```r
library(tseries)
library(dplyr)
```

```
##
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':
##
##     filter, lag

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

```r
library(readxl)
library(ggplot2)
library(Kendall)
library(cowplot)
```

```r
#Importing data set

getwd()
```

```
## [1] "C:/Users/dhr20/OneDrive - Duke University/1 - Academics/1 - First Year/2 - Spring 2024/3 - Time
```

```r
raw_energy_data <- read_excel(path="./Data/Table_10.1_Renewable_Energy_Production_and_Consumption_by_Sou
```

```
## New names:
## * `` -> `...1`
## * `` -> `...2`
## * `` -> `...3`
## * `` -> `...4`
## * `` -> `...5`
## * `` -> `...6`
## * `` -> `...7`
## * `` -> `...8`
## * `` -> `...9`
## * `` -> `...10`
## * `` -> `...11`
## * `` -> `...12`
## * `` -> `...13`
## * `` -> `...14`
```

```r
colnames(raw_energy_data)=c("Month",
                             "Wood Energy Production",
                             "Biofuels Production",
                             "Total Biomass Energy Production",
                             "Total Renewable Energy Production",
                             "Hydroelectric Power Consumption",
                             "Geothermal Energy Consumption",
```

```r
                            "Solar Energy Consumption",
                            "Wind Energy Consumption",
                            "Wood Energy Consumption",
                            "Waste Energy Consumption",
                            "Biofuels Consumption",
                            "Total Biomass Energy Consumption",
                            "Total Renewable Energy Consumption")

raw_energy_data <- raw_energy_data[,1:6]
raw_energy_data_dates <- raw_energy_data[,1]
raw_energy_data_others <- raw_energy_data[,4:6]
raw_energy_data <- cbind(raw_energy_data_dates,raw_energy_data_others)

head(raw_energy_data)
```

```
##        Month Total Biomass Energy Production Total Renewable Energy Production
## 1 1973-01-01                         129.787                           219.839
## 2 1973-02-01                         117.338                           197.330
## 3 1973-03-01                         129.938                           218.686
## 4 1973-04-01                         125.636                           209.330
## 5 1973-05-01                         129.834                           215.982
## 6 1973-06-01                         125.611                           208.249
##   Hydroelectric Power Consumption
## 1                          89.562
## 2                          79.544
## 3                          88.284
## 4                          83.152
## 5                          85.643
## 6                          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
#Converting to time series
ts_energy_data <- ts(raw_energy_data[,2:4], start=c(1973,1), frequency=12)

head(ts_energy_data)
```

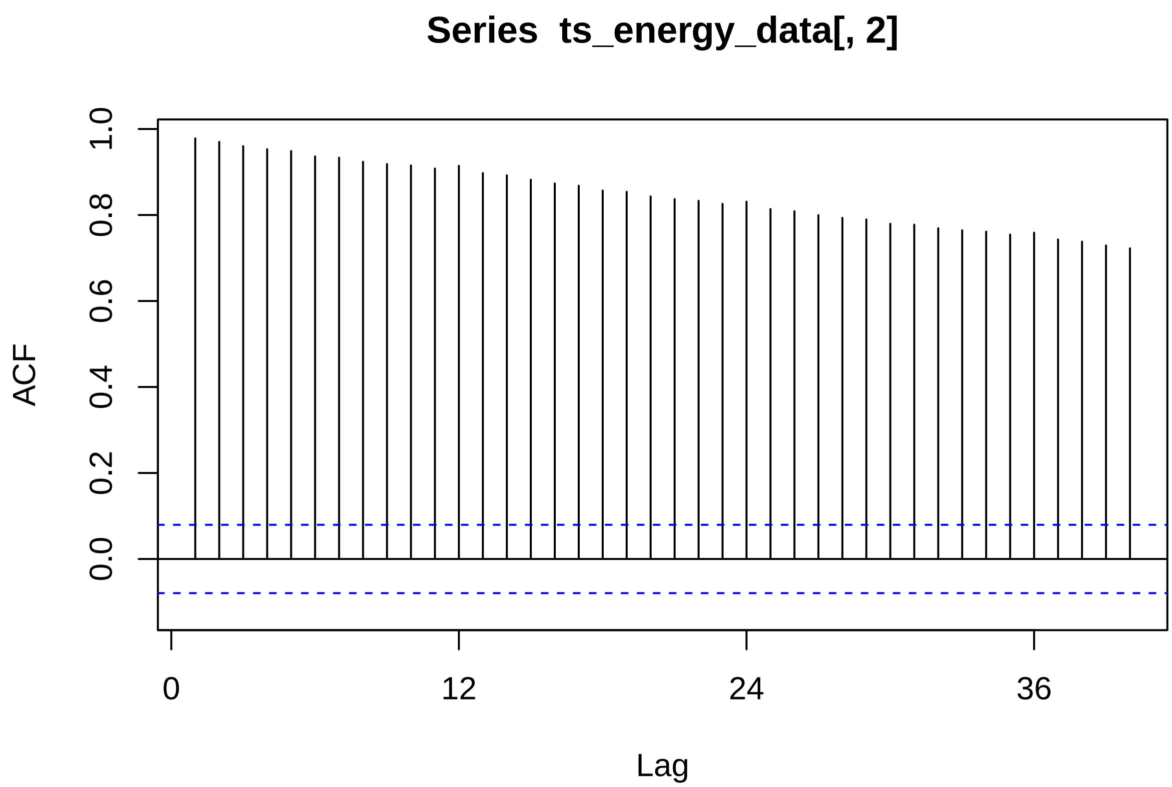
```
##          Total Biomass Energy Production Total Renewable Energy Production
## Jan 1973                         129.787                           219.839
## Feb 1973                         117.338                           197.330
## Mar 1973                         129.938                           218.686
## Apr 1973                         125.636                           209.330
## May 1973                         129.834                           215.982
## Jun 1973                         125.611                           208.249
##          Hydroelectric Power Consumption
## Jan 1973                          89.562
```

```
## Feb 1973                          79.544
## Mar 1973                          88.284
## Apr 1973                          83.152
## May 1973                          85.643
## Jun 1973                          82.060
```

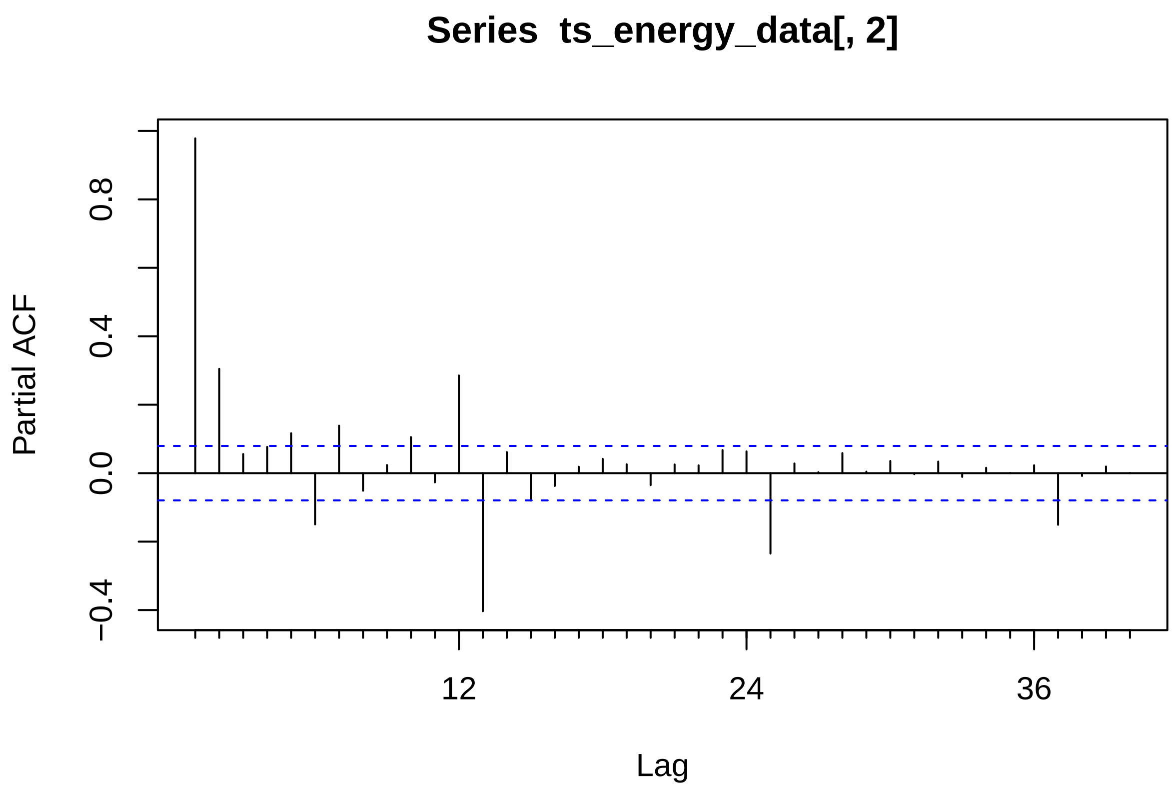
```r
tail(ts_energy_data)
```

```
##          Total Biomass Energy Production Total Renewable Energy Production
## Apr 2023                         404.131                           699.747
## May 2023                         437.506                           740.660
## Jun 2023                         429.839                           691.709
## Jul 2023                         437.109                           711.895
## Aug 2023                         439.521                           711.962
## Sep 2023                         422.351                           666.253
##          Hydroelectric Power Consumption
## Apr 2023                          59.646
## May 2023                          93.759
## Jun 2023                          66.434
## Jul 2023                          72.463
## Aug 2023                          72.150
## Sep 2023                          56.284
```

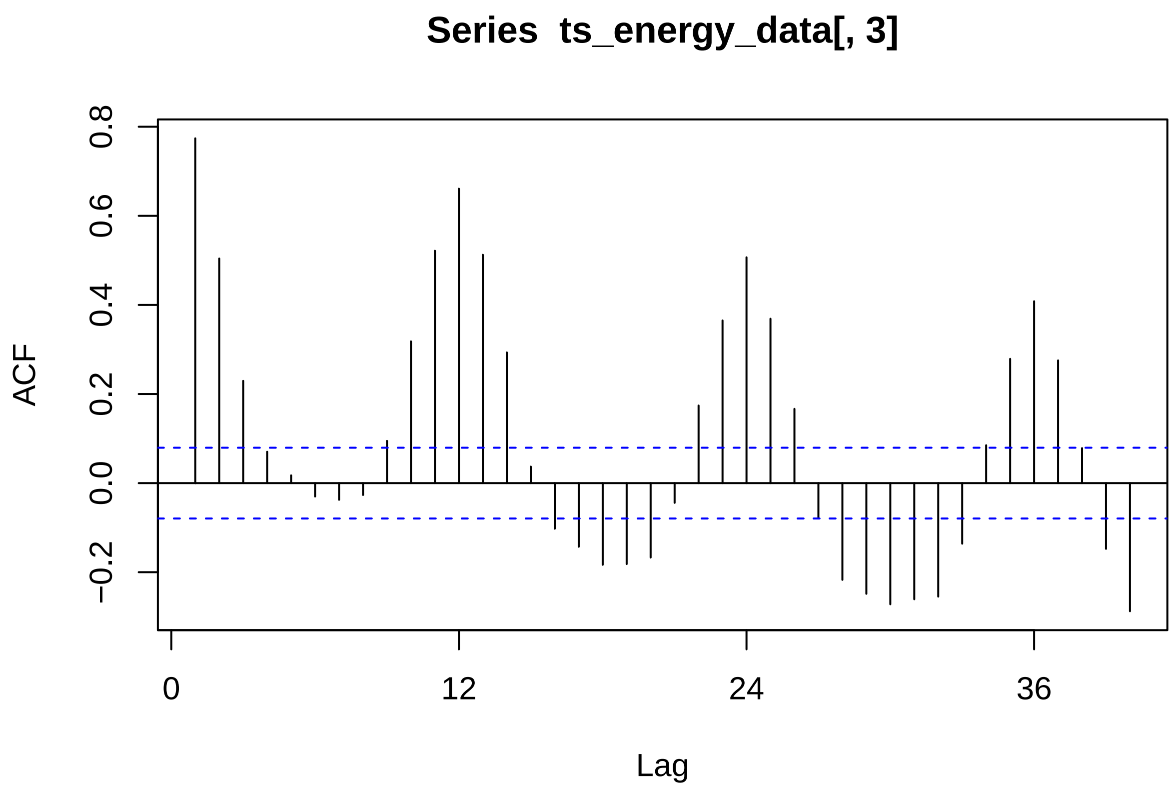
```r
#Preparing Renewable Energy plots
renewable_ts_plot <- autoplot(ts_energy_data[,2]) +
  ggtitle("Renewable Energy Time Series") +
  xlab("Time") +
  ylab("Trillion Btu")

renewable_acf <- Acf(ts_energy_data[,2],lag.max=40,type="correlation",
                    plot=TRUE)
```



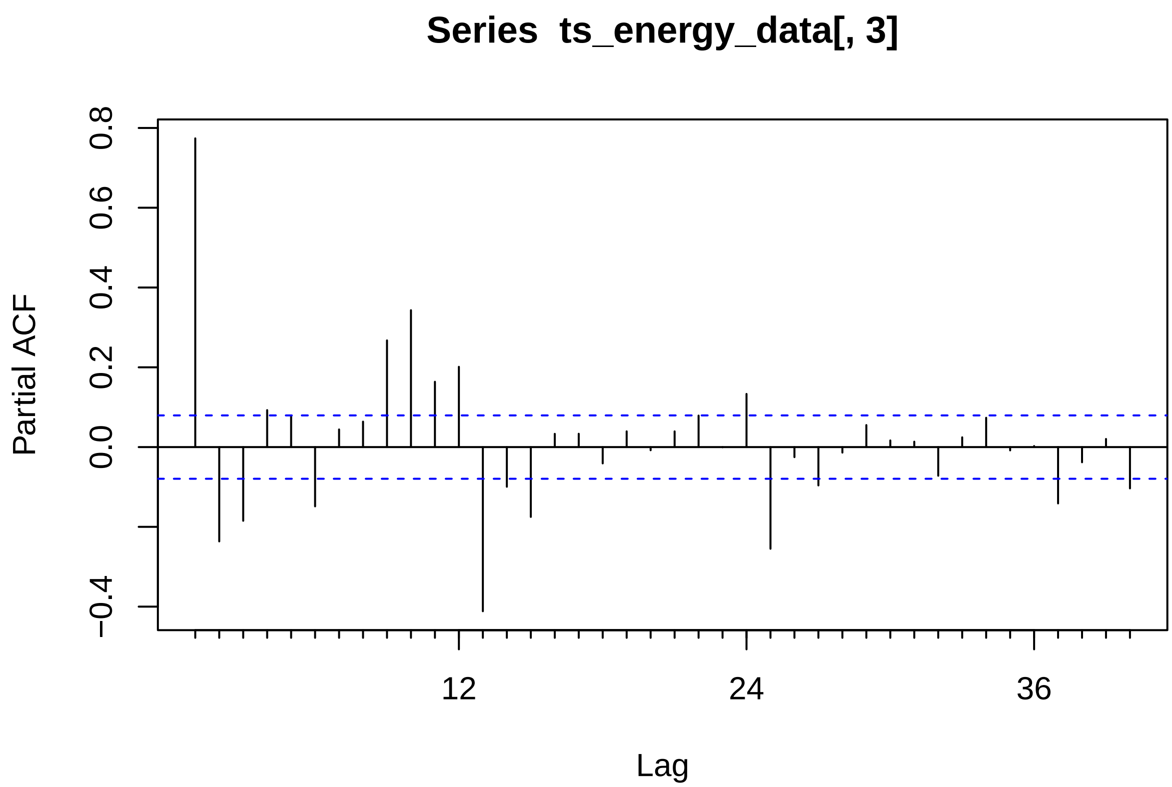

```
renewable_pacf <- Pacf(ts_energy_data[,2],lag.max=40,plot=TRUE)
```



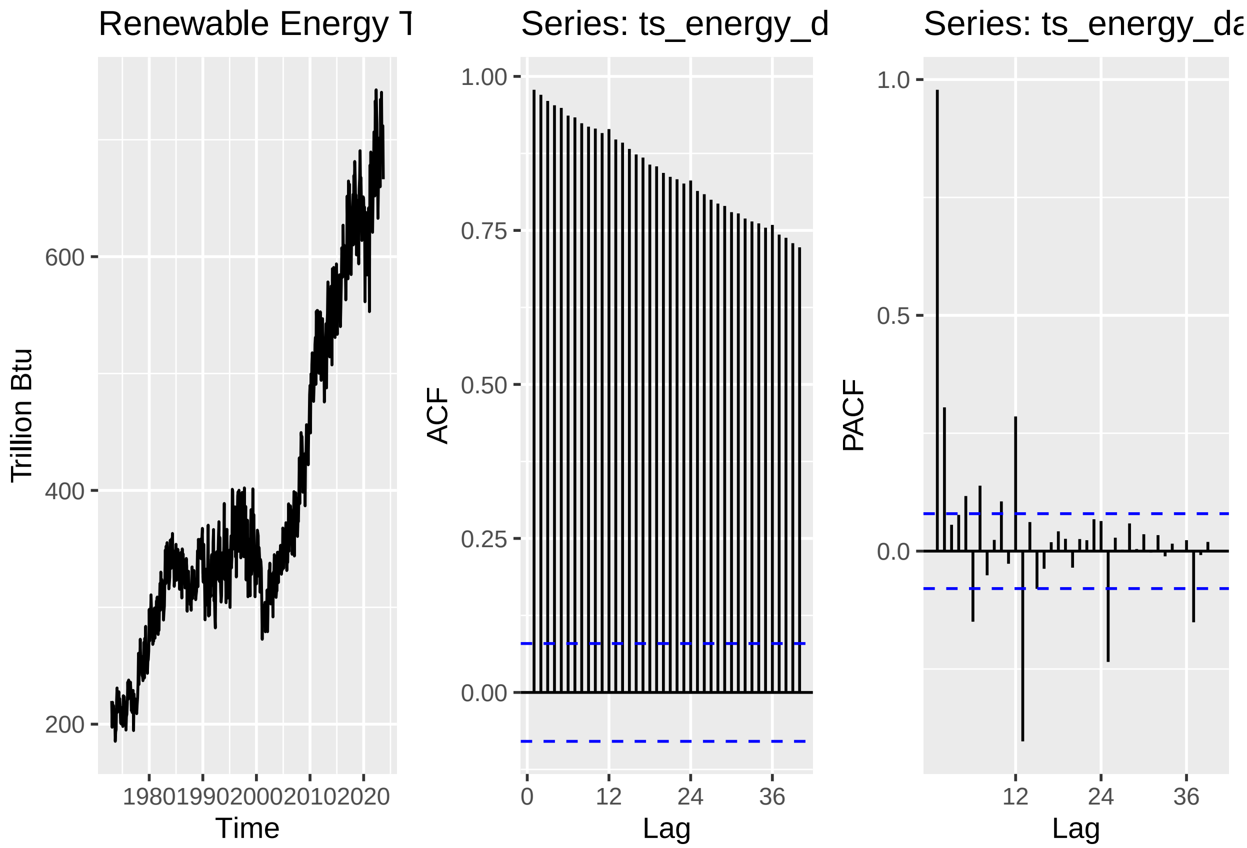

```
#Preparing Hydroelectric Energy plots
hydroelectric_ts_plot <- autoplot(ts_energy_data[,3]) +
  ggtitle("Hydroelectric Energy Time Series") +
  xlab("Time") +
  ylab("Trillion Btu")

hydroelectric_acf <- Acf(ts_energy_data[,3],lag.max=40,type="correlation",
                        plot=TRUE)
```



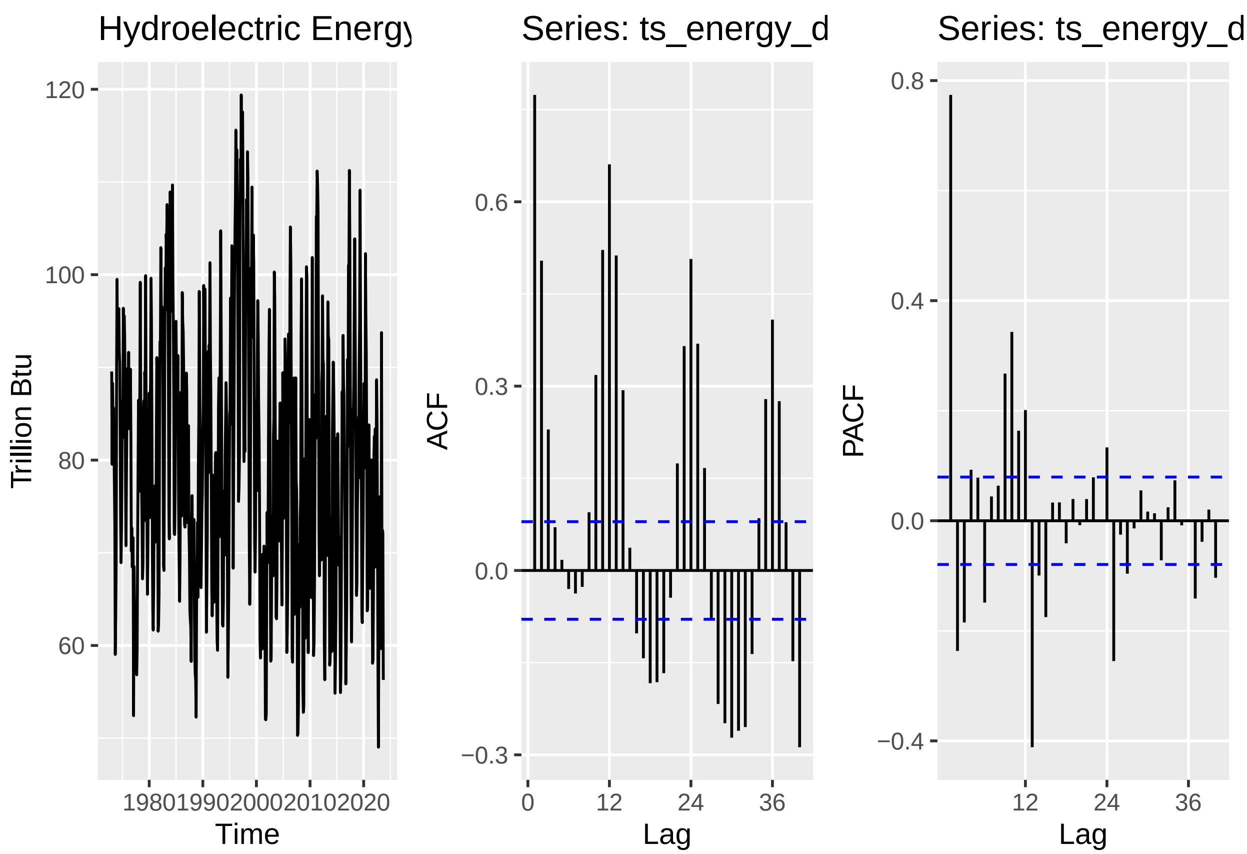

```
hydroelectric_pacf <- Pacf(ts_energy_data[,3],lag.max=40,plot=TRUE)
```




```r
#Combining plots
plot_grid(renewable_ts_plot,
          autoplot(renewable_acf),
          autoplot(renewable_pacf),
          nrow = 1, ncol = 3)
```




```
plot_grid(hydroelectric_ts_plot,
          autoplot(hydroelectric_acf),
          autoplot(hydroelectric_pacf),
          nrow = 1, ncol = 3)
```



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 Renewable Energy Production
#This series does appear to have a trend -- the time series is increasing
#steadily as time goes on. Additionally, we note that the high correlation
#values in the ACF and the spikes in the PACF indicate a trend.

#Hydroelectric Power Consumption
#This series does appear to have a trend -- the time series is increasing
#steadily as time goes on. Additionally, we note that the high correlation
#values in the ACF and the spikes in the PACF indicate a trend.
```

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.

```
nobs <- nrow(raw_energy_data)

t <- 1:nobs
```

```r
renewable_linear_trend <- lm(raw_energy_data[,2]~t)
summary(renewable_linear_trend)
```

```
##
## Call:
## lm(formula = raw_energy_data[, 2] ~ t)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -102.344  -23.754    5.491   31.980   83.154
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 134.27841    3.18601   42.15   <2e-16 ***
## t             0.47713    0.00905   52.72   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 39.26 on 607 degrees of freedom
## Multiple R-squared:  0.8208, Adjusted R-squared:  0.8205
## F-statistic:  2780 on 1 and 607 DF,  p-value: < 2.2e-16
```

```r
renewable_beta0 <- renewable_linear_trend$coefficients[1]
renewable_beta1 <- renewable_linear_trend$coefficients[2]

#For renewable energy, the slope is 0.47713 (Beta 0) and the intercept is
#134.27841 (Beta 1). These values, in addition to the low p-value and
#higher R-squared, indicate a significant relationship between the renewable
#energy data and time.

hydroelectric_linear_trend <- lm(raw_energy_data[,3]~t)
summary(hydroelectric_linear_trend)
```

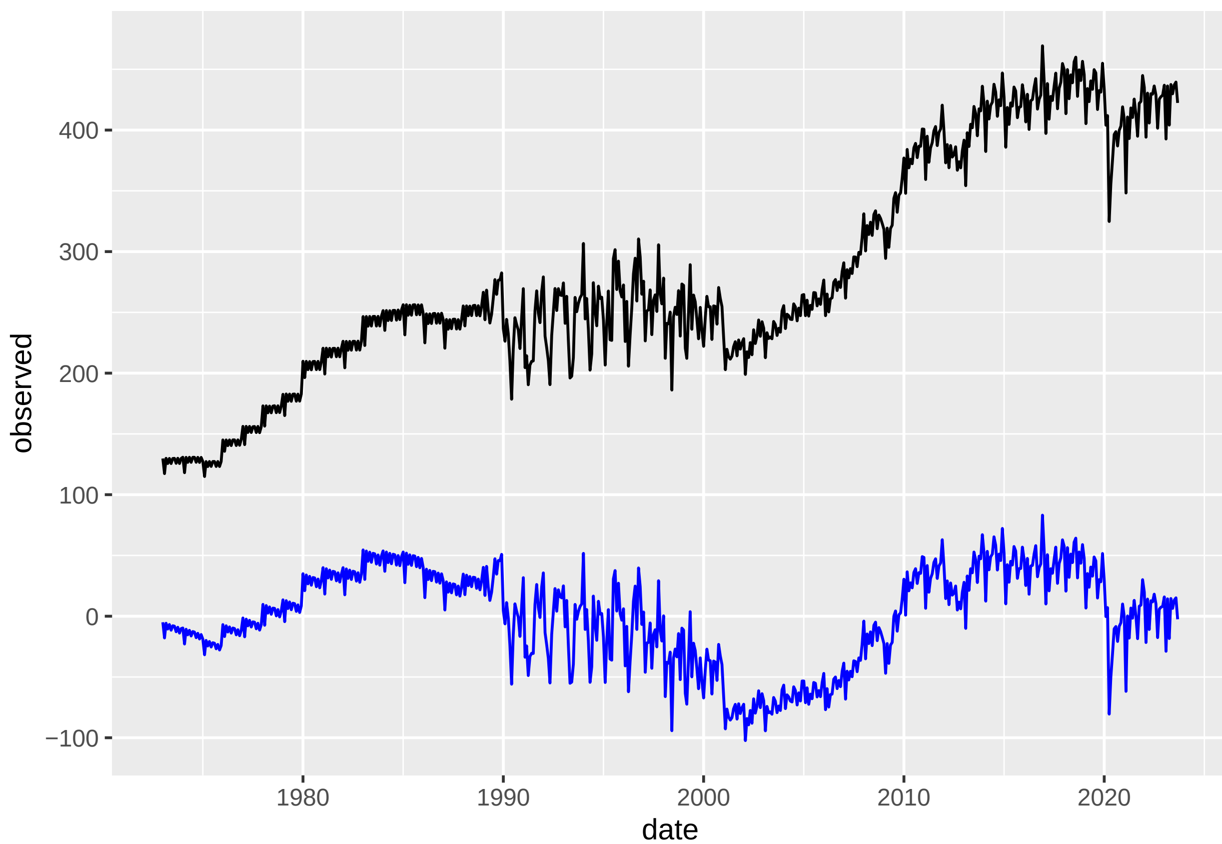
```
##
## Call:
## lm(formula = raw_energy_data[, 3] ~ t)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -148.27  -35.63   11.58   41.51  144.27
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 180.98940    4.90151   36.92   <2e-16 ***
## t             0.70404    0.01392   50.57   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 60.41 on 607 degrees of freedom
## Multiple R-squared:  0.8081, Adjusted R-squared:  0.8078
## F-statistic:  2557 on 1 and 607 DF,  p-value: < 2.2e-16
```

```r
hydroelectric_beta0 <- hydroelectric_linear_trend$coefficients[1]
hydroelectric_beta1 <- hydroelectric_linear_trend$coefficients[2]

#For hydroelectric power consumption, the slope is 0.70404 (Beta 0) and the
#intercept is 180.98940 (Beta 1). These values, in addition to the low p-value
#and higher R-squared, indicate a significant relationship between the
#hydroelectric data and time.
```

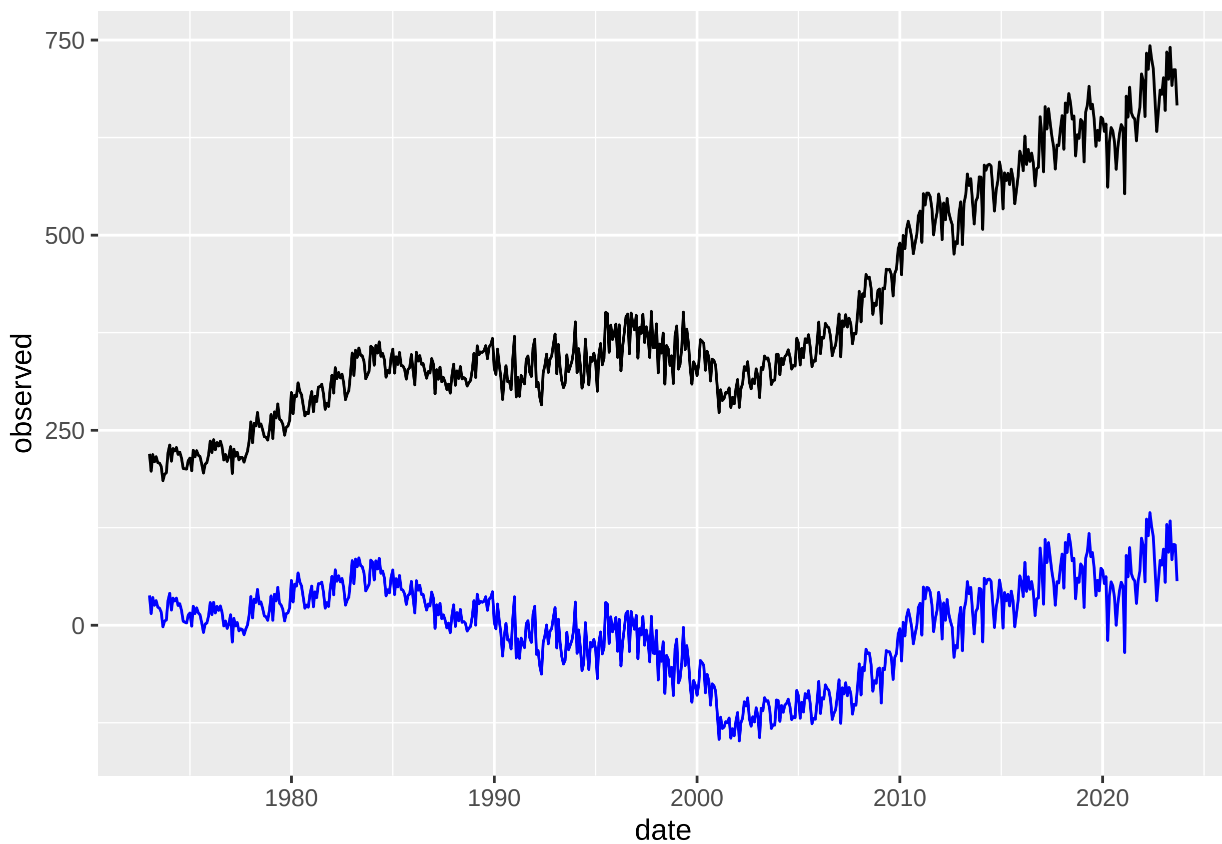
Q4

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
renewable_y_detrend <- raw_energy_data[,2] - (renewable_beta0 + renewable_beta1*t)

renewable_df_detrend <- data.frame("date"=raw_energy_data[,1],
                                   "observed"=raw_energy_data[,2],
                                   "detrend"=renewable_y_detrend)

ggplot(renewable_df_detrend,aes(x=date))+
  geom_line(aes(y=observed),color="black")+
  geom_line(aes(y=detrend),color="blue")
```



```r
hydroelectric_y_detrend <- raw_energy_data[,3] -(hydroelectric_beta0 + hydroelectric_beta1*t)

hydroelectric_df_detrend <- data.frame("date"=raw_energy_data[,1],
                                       "observed"=raw_energy_data[,3],
                                       "detrend"=hydroelectric_y_detrend)

ggplot(hydroelectric_df_detrend,aes(x=date))+
  geom_line(aes(y=observed),color="black")+
  geom_line(aes(y=detrend),color="blue")
```

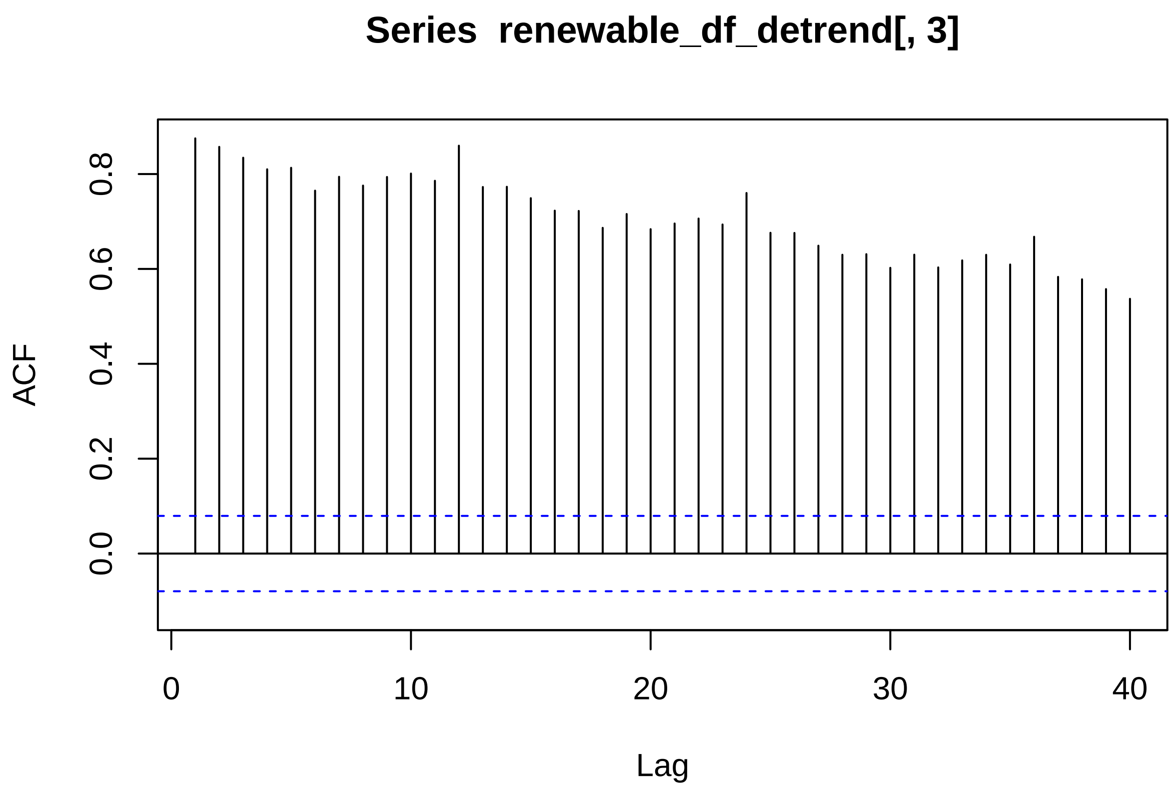



```r
#As compared to the plots from Q1, the detrended lines (shown in blue on both
#plots) have a mean of 0. These are significantly changed from the plots in Q1
#as the data has now been "de-trended". That being said, because a linear model
#was used to de-trend the data, we still see that there is some trend present.
```

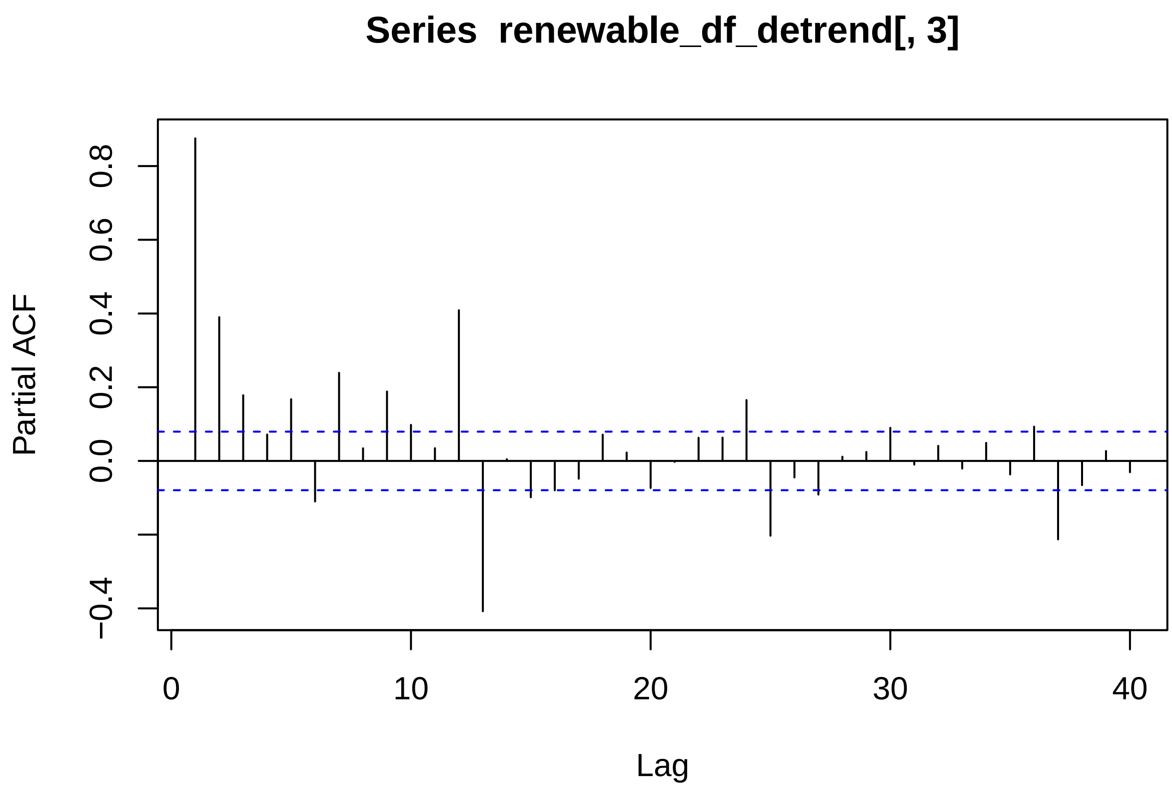
Q5

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?

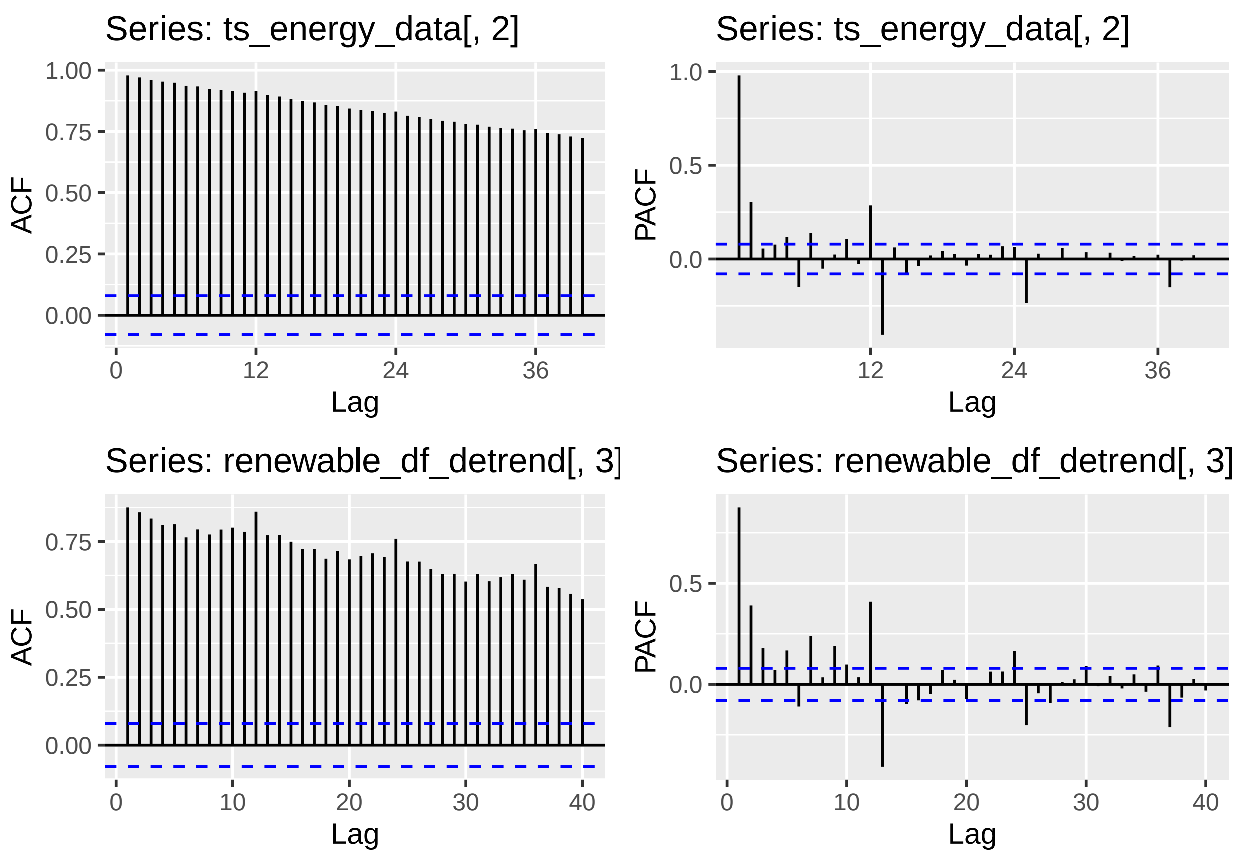
```r
renewable_acf_detrend <- Acf(renewable_df_detrend[,3],lag.max=40,
                             type="correlation",plot=TRUE)
```



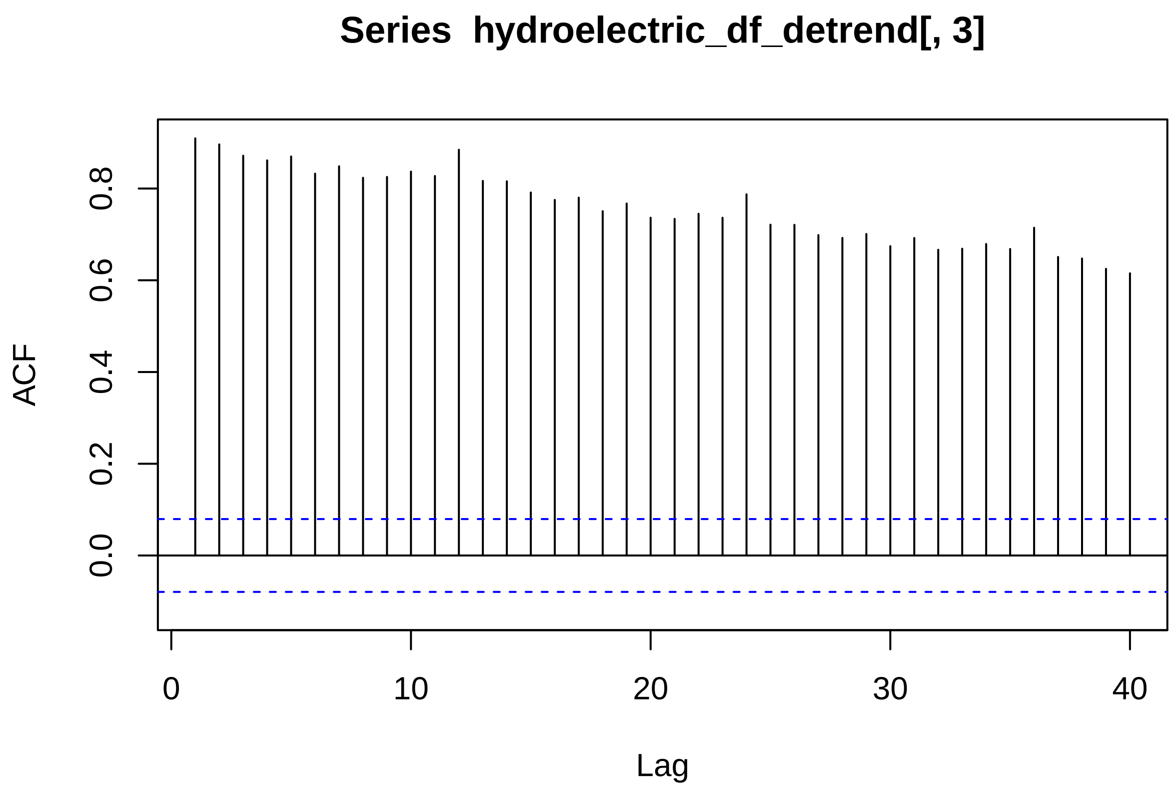

```
renewable_pacf_detrend <- Pacf(renewable_df_detrend[,3],lag.max=40,plot=TRUE)
```



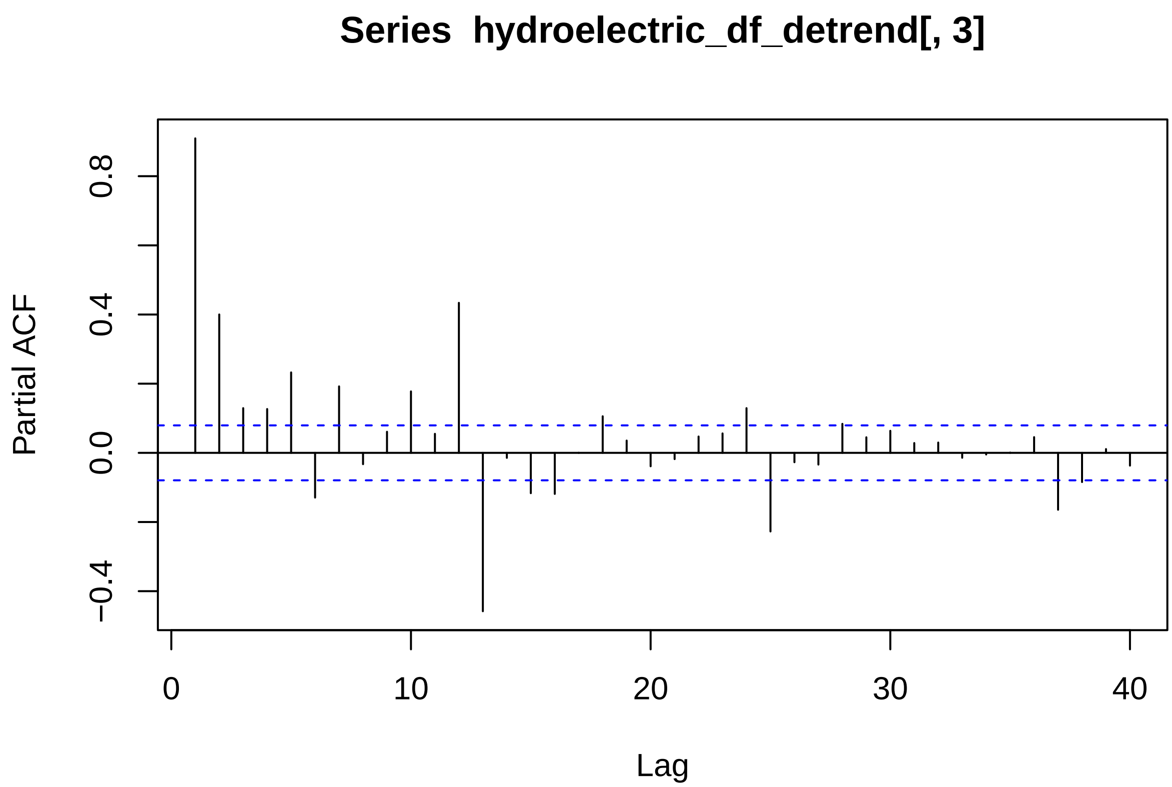

```
plot_grid(autoplot(renewable_acf),
          autoplot(renewable_pacf),
          autoplot(renewable_acf_detrend),
          autoplot(renewable_pacf_detrend))
```



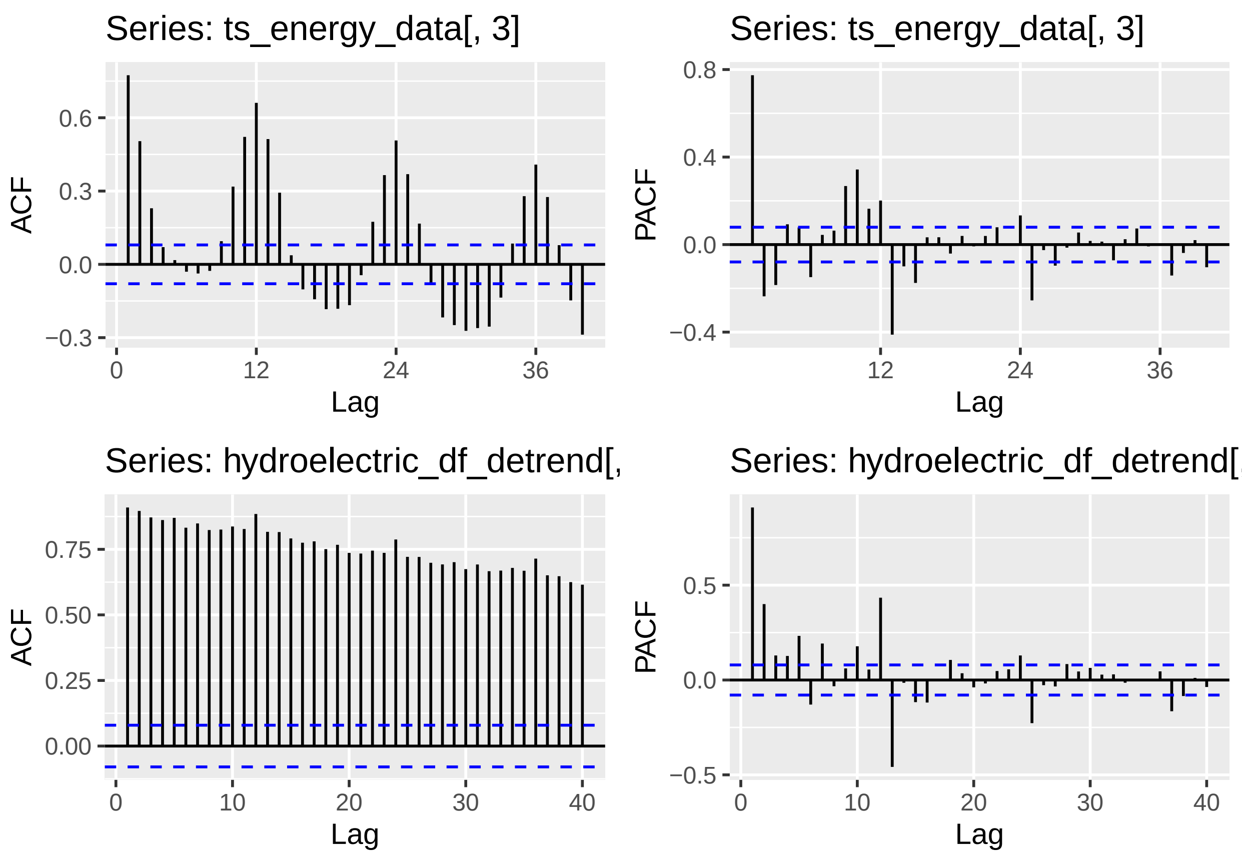

```
hydroelectric_acf_detrend <- Acf(hydroelectric_df_detrend[,3],lag.max=40,
                                 type="correlation",plot=TRUE)
```




```
hydroelectric_pacf_detrend <- Pacf(hydroelectric_df_detrend[,3],lag.max=40,
                                   plot=TRUE)
```




```
plot_grid(autoplot(hydroelectric_acf),
          autoplot(hydroelectric_pacf),
          autoplot(hydroelectric_acf_detrend),
          autoplot(hydroelectric_pacf_detrend))
```




```
#For renewable energy, the plots changed slightly -- for the ACF, in the
#detrended data, there are still very large correlation coefficients indicating
#that the trends were not totally removed. That said, the detrending had some
#effect as we can see a few spikes appeaing at lag 12, lag 24, and lag 3 that
#indicate seasonality. The PACF, in the detrended data, has similar spikes but
#they are more pronounced as they "blossom" with less trend elsewhere in the
#data.

#For hydroelectric power consumption, the plots changed more dramatically --
#for the ACF, in the detrended data, the correlation coefficients are still
#very high and we can see that the sinusoidal pattern is gone and the ACF has a
#more consistent pattern with a few spikes at lag 12, 24, and 36. The PACF, in
#the detrended data, has similar spikes but they are less pronounced.
```

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.

```
#For renewable energy, the time series does indicate a seasonal trend; the ACF
#does not indicate a seasonal trend.

#For hydroelectric power consumption, the time series does indicate a seasonal
#trend given the spikes that occur annually; the ACF does indicate a seasonal
#trend -- the spikes decrease around lags 6, 18, and 30 while they increase
#around 12, 24, and 36.
```

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?

```
renewable_dummies <- seasonaldummy(ts_energy_data[,2])
renewable_seas_means_model <- lm(raw_energy_data[,3]~renewable_dummies)
summary(renewable_seas_means_model)
```

```
##
## Call:
## lm(formula = raw_energy_data[, 3] ~ renewable_dummies)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -199.19  -86.35  -48.84  113.18  331.58
##
## Coefficients:
##                      Estimate Std. Error t value Pr(>|t|)
## (Intercept)           404.526     19.574  20.666   <2e-16 ***
## renewable_dummiesJan    2.962     27.546   0.108    0.914
## renewable_dummiesFeb  -34.476     27.546  -1.252    0.211
## renewable_dummiesMar    3.929     27.546   0.143    0.887
## renewable_dummiesApr   -8.695     27.546  -0.316    0.752
## renewable_dummiesMay    6.645     27.546   0.241    0.809
## renewable_dummiesJun   -4.198     27.546  -0.152    0.879
## renewable_dummiesJul    2.460     27.546   0.089    0.929
## renewable_dummiesAug   -5.026     27.546  -0.182    0.855
## renewable_dummiesSep  -29.119     27.546  -1.057    0.291
## renewable_dummiesOct  -20.068     27.682  -0.725    0.469
## renewable_dummiesNov  -20.346     27.682  -0.735    0.463
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 138.4 on 597 degrees of freedom
## Multiple R-squared:  0.009296,   Adjusted R-squared:  -0.008958
## F-statistic: 0.5093 on 11 and 597 DF,  p-value: 0.8976
```

```
#For renewable energy, the high p-value and the low R-squared value indicate
#that the seasonal means model is not a good way to remove seasonality.

hydroelectric_dummies <- seasonaldummy(ts_energy_data[,3])
```

```r
hydroelectric_seas_means_model <- lm(raw_energy_data[,4]~hydroelectric_dummies)
summary(hydroelectric_seas_means_model)
```

```
##
## Call:
## lm(formula = raw_energy_data[, 4] ~ hydroelectric_dummies)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -31.323  -5.849  -0.468   6.243  32.290
##
## Coefficients:
##                          Estimate Std. Error t value Pr(>|t|)
## (Intercept)                80.282      1.470  54.601  < 2e-16 ***
## hydroelectric_dummiesJan    4.807      2.069   2.323  0.02050 *
## hydroelectric_dummiesFeb   -2.725      2.069  -1.317  0.18831
## hydroelectric_dummiesMar    6.825      2.069   3.298  0.00103 **
## hydroelectric_dummiesApr    5.319      2.069   2.571  0.01039 *
## hydroelectric_dummiesMay   13.922      2.069   6.729 4.02e-11 ***
## hydroelectric_dummiesJun   10.650      2.069   5.147 3.60e-07 ***
## hydroelectric_dummiesJul    3.912      2.069   1.891  0.05914 .
## hydroelectric_dummiesAug   -5.677      2.069  -2.744  0.00626 **
## hydroelectric_dummiesSep  -16.797      2.069  -8.118 2.72e-15 ***
## hydroelectric_dummiesOct  -16.468      2.079  -7.920 1.17e-14 ***
## hydroelectric_dummiesNov  -10.885      2.079  -5.235 2.29e-07 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 10.4 on 597 degrees of freedom
## Multiple R-squared:  0.4697, Adjusted R-squared:  0.4599
## F-statistic: 48.07 on 11 and 597 DF,  p-value: < 2.2e-16
```

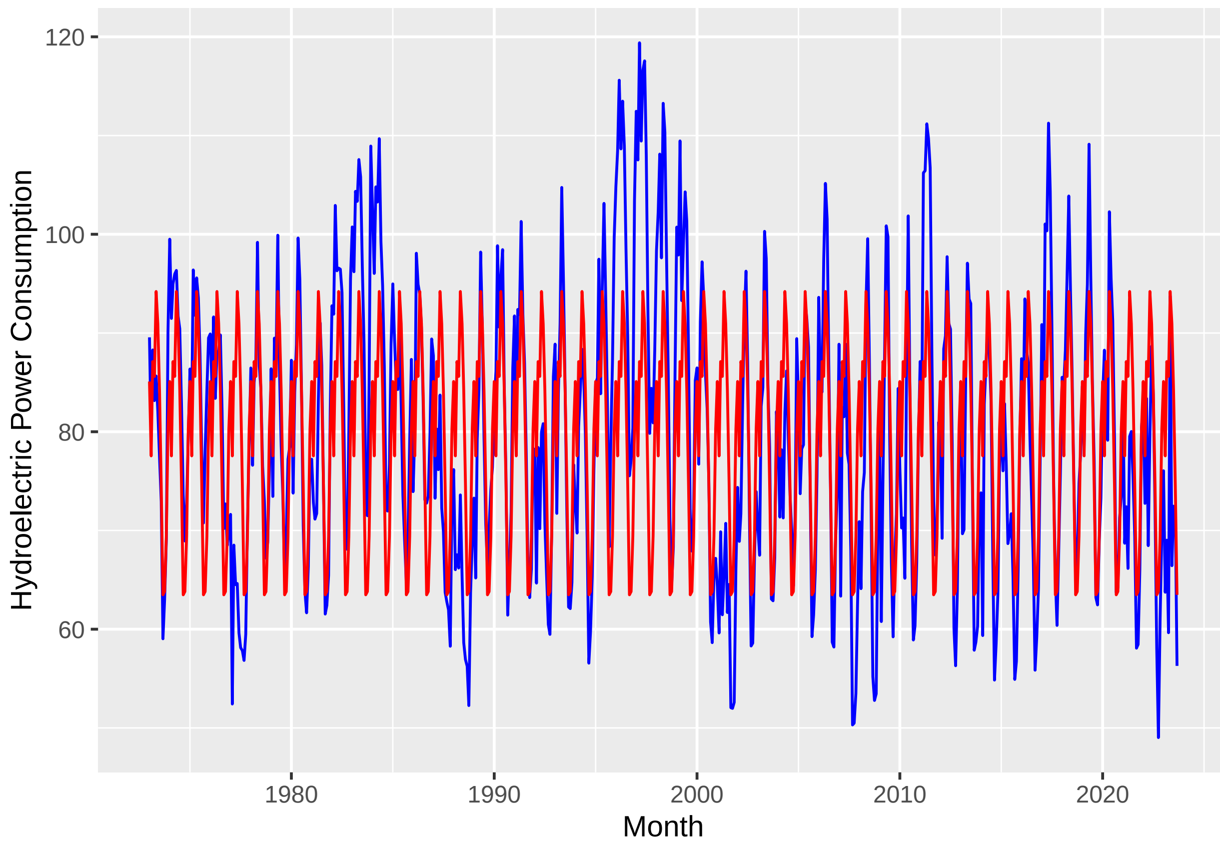
```r
#For hydroelectric, the low p-value and the higher R-squared value indicate
#that the seasonal means model is a good fit for this data.

#In the case of renewable energy, it's not that there is no seasonality, it's
#just that a seasonal means model is not a good representation of the data. For
#hydroelectric, the results do match the answer to Q6.
```

Q8

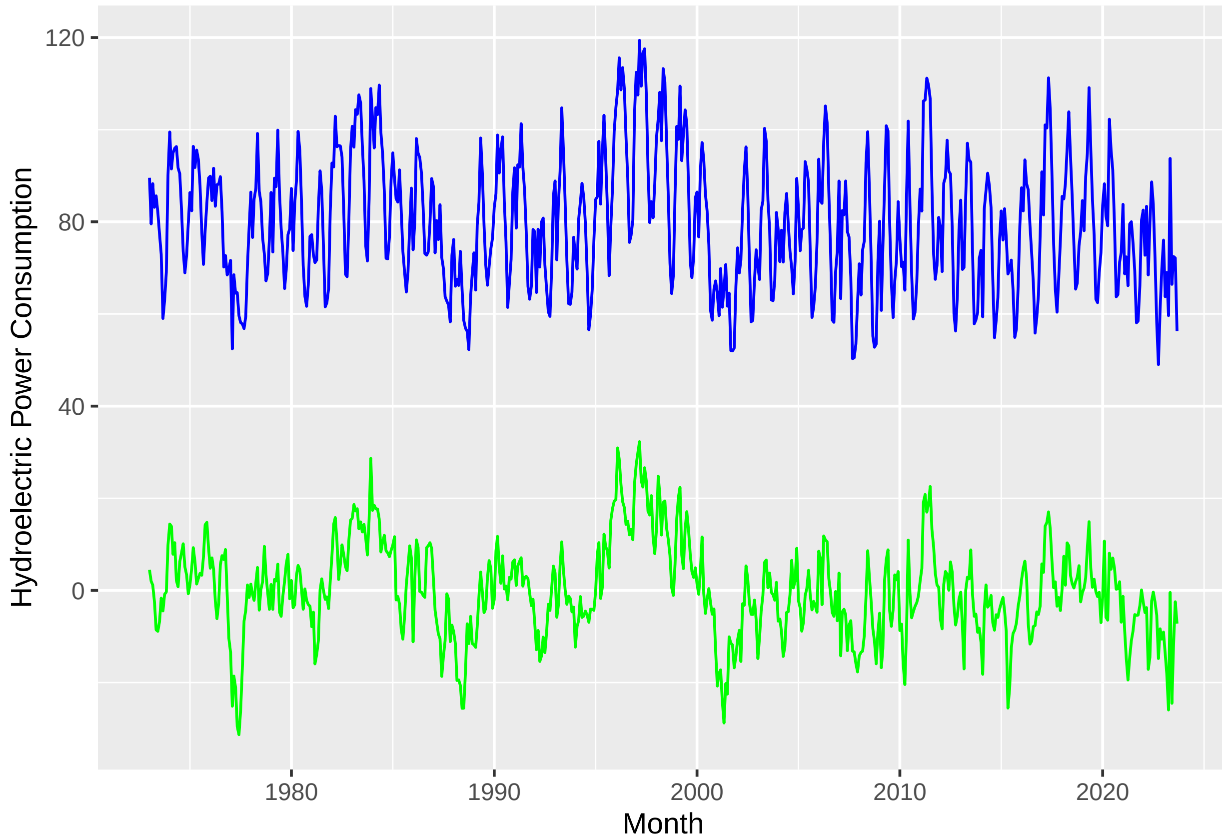
Use the regression coefficients from Q7 to deseason the series. Plot the deseason series and compare with the plots from part Q1. Did anything change?

```r
#Not running this for renewable energy since there was no seasonal trend

#Hydroelectric
#Look at the regression coefficient. These will be the values of Beta
#Store regression coefficients
hydroelectric_beta_int <- hydroelectric_seas_means_model$coefficients[1]
hydroelectric_beta_coeff <- hydroelectric_seas_means_model$coefficients[2:12]
```

```r
#compute seasonal component
hydroelectric_seas_comp <- array(0,nobs)
for(i in 1:nobs){
  hydroelectric_seas_comp[i] <- (hydroelectric_beta_int+hydroelectric_beta_coeff
                                 %*% hydroelectric_dummies[i,])
}

#Understanding what we did
ggplot(raw_energy_data, aes(x=Month, y=raw_energy_data[,4])) +
            geom_line(color="blue") +
            ylab(paste0(colnames(raw_energy_data)[4],sep="")) +
            geom_line(aes(y=hydroelectric_seas_comp), col="red")
```




```r
#Removing seasonal component
deseason_hydroelectric_data <- raw_energy_data[,4]-hydroelectric_seas_comp

#Understanding what we did
ggplot(raw_energy_data, aes(x=Month, y=raw_energy_data[,4])) +
            geom_line(color="blue") +
            ylab(paste0(colnames(raw_energy_data)[4],sep="")) +
            geom_line(aes(y=deseason_hydroelectric_data), col="green")
```

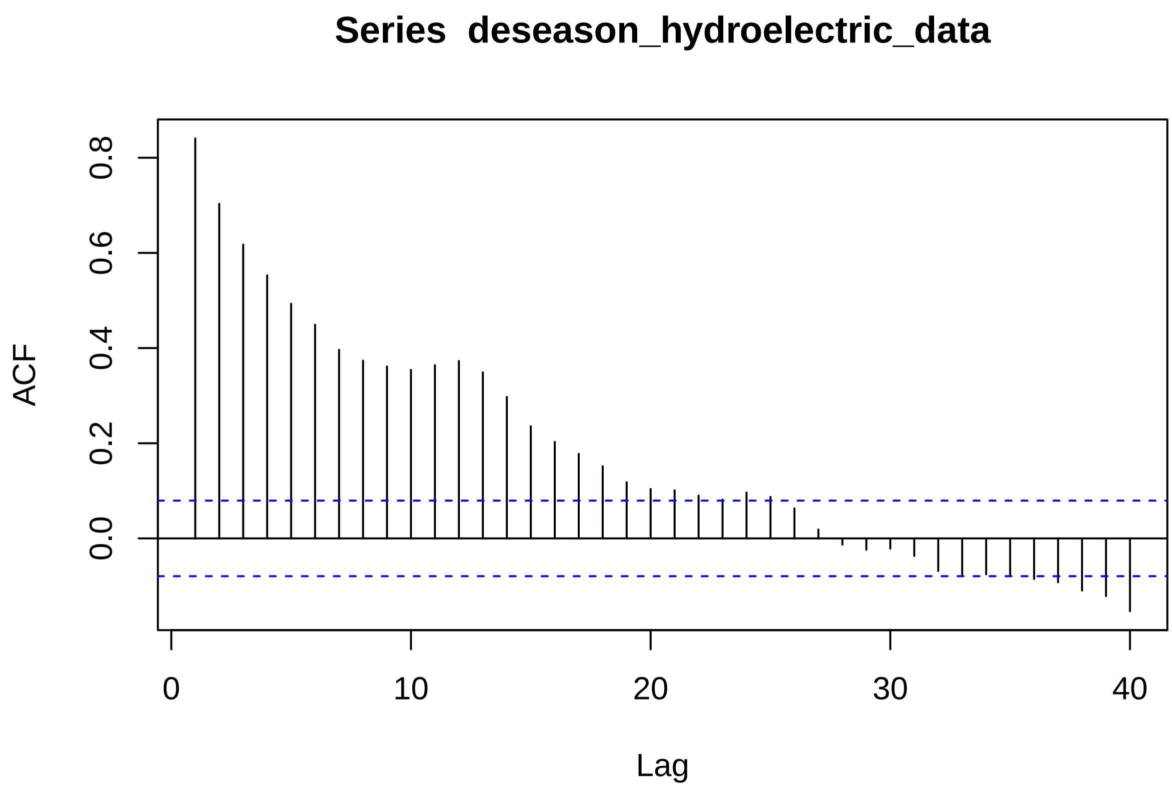



```r
#Compared to Q1, the plots move the mean to 0 and the regular, wave-like
#pattern is gone. The seasonal component has been removed.
```

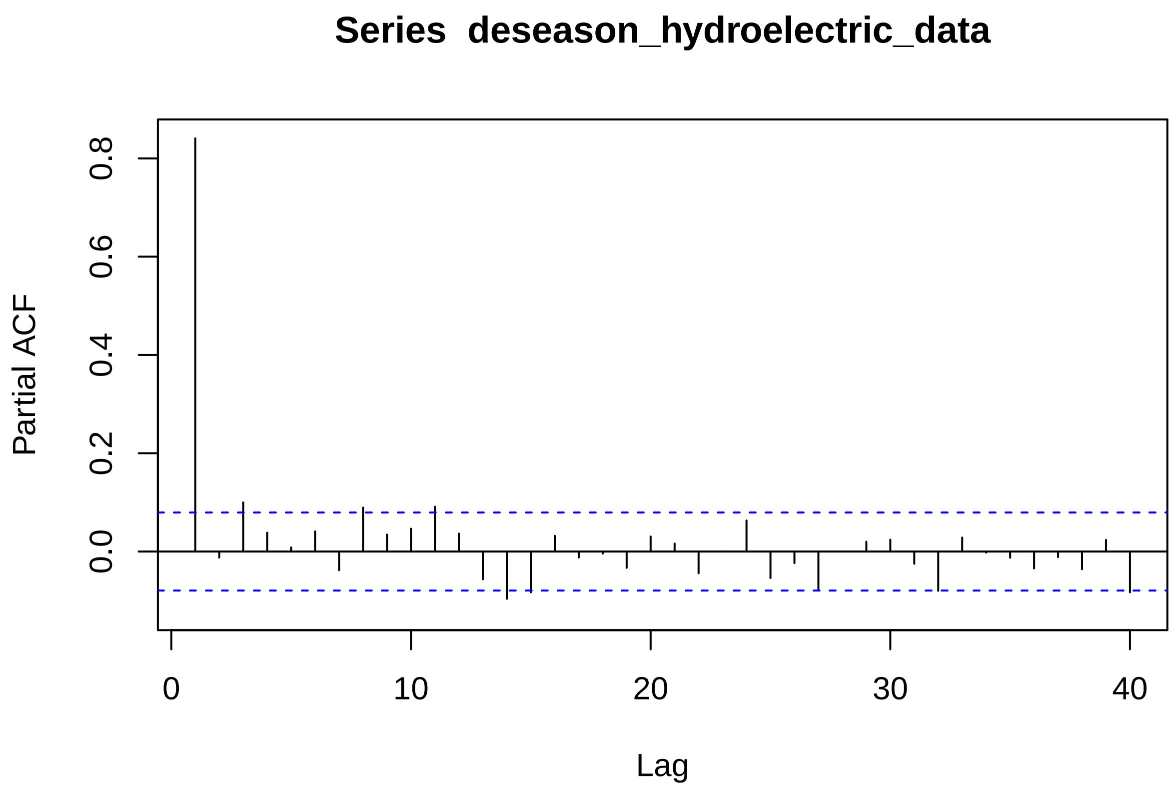
Q9

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?

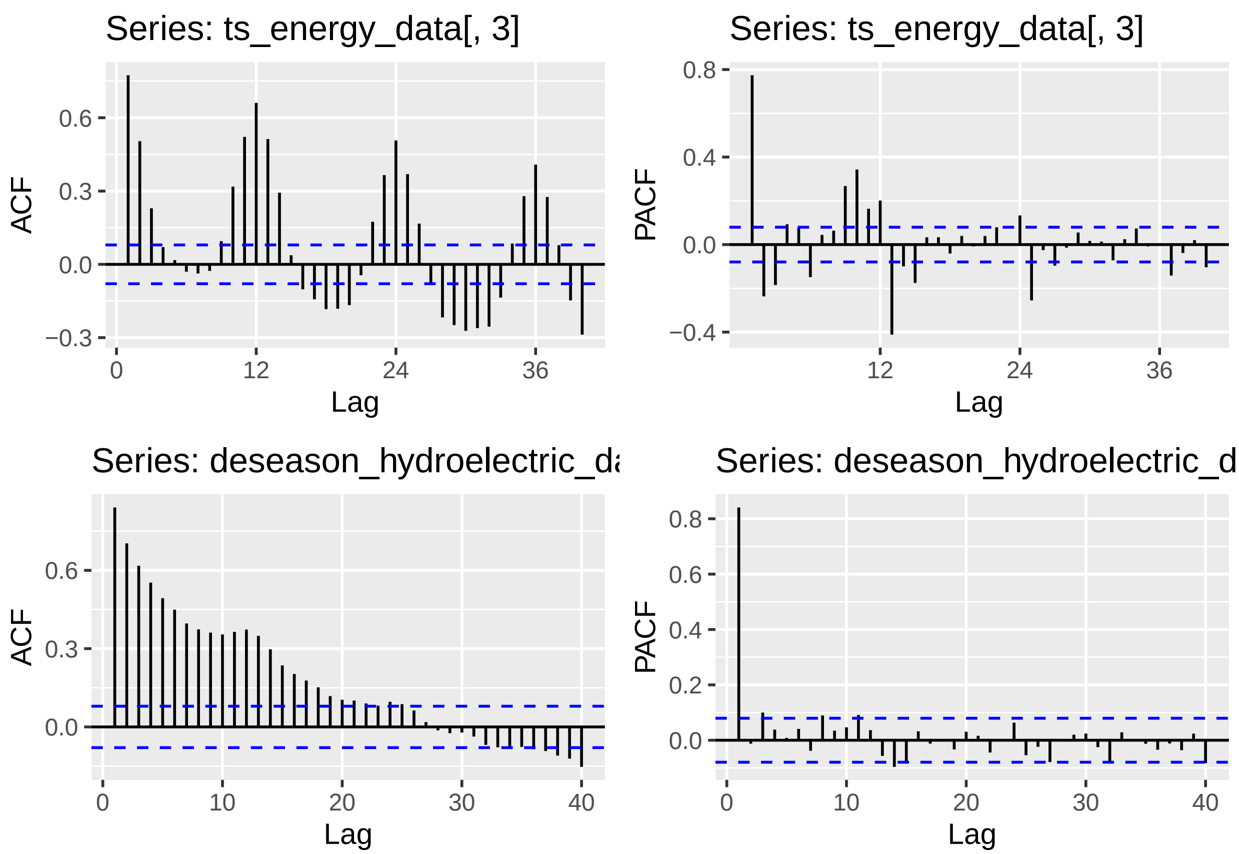
```r
#Not running this for renewable energy since there was no seasonal trend

hydroelectric_acf_deseason <- Acf(deseason_hydroelectric_data,lag.max=40,
                                  type="correlation", plot=TRUE)
```




```
hydroelectric_pacf_deseason <- Pacf(deseason_hydroelectric_data,lag.max=40,
                                    plot=TRUE)
```




```r
plot_grid(autoplot(hydroelectric_acf),
          autoplot(hydroelectric_pacf),
          autoplot(hydroelectric_acf_deseason),
          autoplot(hydroelectric_pacf_deseason))
```




```
#For hydroelectric power consumption, the plots did change. Notably, the ACF
#moves from a sinusoidal pattern to a decreasing trend. Around lag 20, the
#correlation becomes insignificant. The PACF is telling us that we only need to
#add t - 1 to our data set. In other words, significant de-trending has
#occurred here.
```