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

Assignment 3 - Due date 02/03/26

Aisha Shen

```r
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
```

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 2025 Monthly Energy Review. This time 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.

```
library(readxl)
library(openxlsx)
library(forecast)
```

```
## Warning: package 'forecast' was built under R version 4.5.2

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

```
library(Kendall)
```

```
## Warning: package 'Kendall' was built under R version 4.5.2
```

```
library(tseries)
```

```
## Warning: package 'tseries' was built under R version 4.5.2
```

```
energy_data1 <- read_excel(path="/Users/meilishen/Documents/TimeSeries/TSA_Sp26/Data/Table_10.1_Renewabl
```

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

```
#Now let's extract the column names from row 11
read_col_names <- read_excel(path="/Users/meilishen/Documents/TimeSeries/TSA_Sp26/Data/Table_10.1_Renewa
```

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

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

timeseries_df <- energy_data1 [, c("Total Renewable Energy Production", "Hydroelectric Power Consumptio
head(timeseries_df)
```

```
## # A tibble: 6 x 2
##   `Total Renewable Energy Production` `Hydroelectric Power Consumption`
##                                 <dbl>                             <dbl>
## 1                                220.                              89.6
## 2                                197.                              79.5
## 3                                219.                              88.3
## 4                                209.                              83.2
## 5                                216.                              85.6
## 6                                208.                              82.1
```

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
# install.packages(c("forecast","tseries","Kendall","cowplot"))
#library(forecast)
#library(Kendall)
#library(tseries)

energy_ts <- ts(timeseries_df, start=c(1931,1),frequency=12 ) # TAKEN FROM A02
mean_series<-apply(energy_ts, 2, mean) # TAKEN FROM A02
sd_series<-apply(energy_ts, 2, sd) # TAKEN FROM A02
```

##Trend Component

Q1

For each series (Total Renewable Production and Hydroelectric Consumption) create three plots arranged in a row (side-by-side): (1) time series plot, (2) ACF, (3) PACF. Use cowplot::plot_grid() to place them in a grid.

```
library(ggplot2)
```

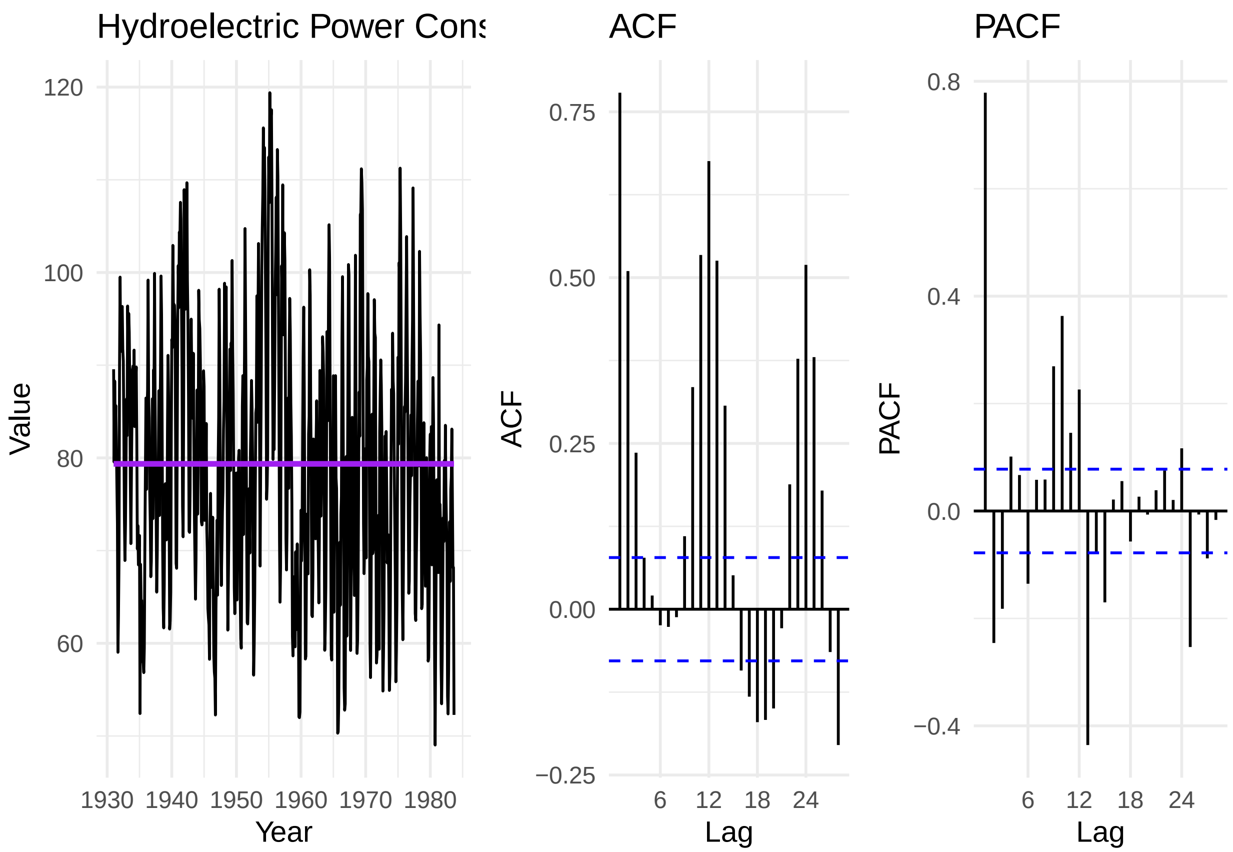
```
## Warning: package 'ggplot2' was built under R version 4.5.2
```

```
library(cowplot)

plot_ts_diagnostics <- function(ts_data, col_idx, mean_values, var_name = "Series")
  {
```

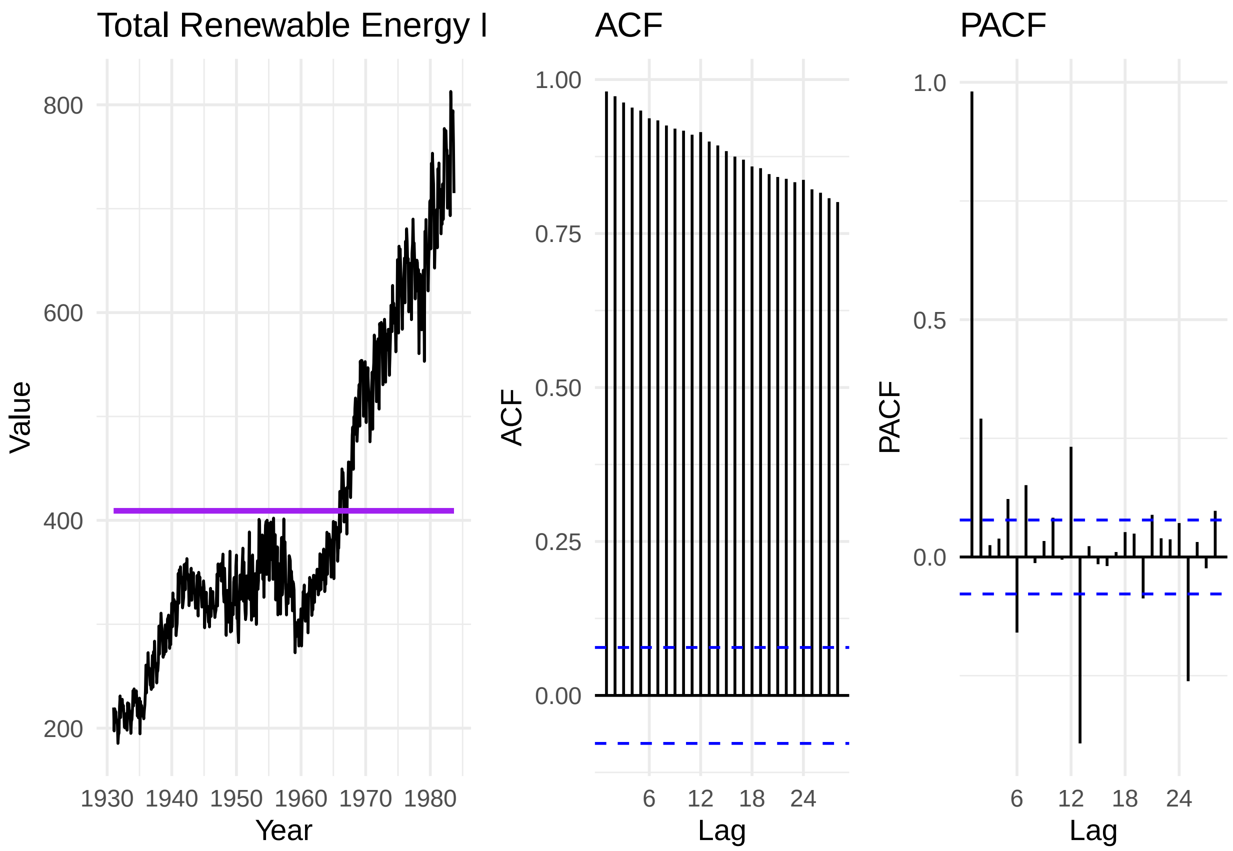
```
  p1 <- autoplot(ts_data[, col_idx]) +
    labs(title = paste(var_name, "Time Series"), x = "Year", y = "Value") +
    geom_line(aes(y = mean_values[col_idx]), color = "purple", size = 1) +
    theme_minimal()
  p2 <- ggAcf(ts_data[, col_idx]) +
    labs(title = "ACF") +
    theme_minimal()
  p3 <- ggPacf(ts_data[, col_idx]) +
    labs(title = "PACF") +
    theme_minimal()
  plot_grid(p1, p2, p3, ncol = 3, align = 'h', rel_widths = c(1.3, 1, 1))
}

row_renewable <- plot_ts_diagnostics(energy_ts, 1, mean_series, "Total Renewable Energy Production")
```

```
## Warning: Using 'size' aesthetic for lines was deprecated in ggplot2 3.4.0.
## i Please use 'linewidth' instead.
## This warning is displayed once per session.
## Call 'lifecycle::last_lifecycle_warnings()' to see where this warning was
## generated.
```

```
row_hydro <- plot_ts_diagnostics(energy_ts, 2, mean_series, "Hydroelectric Power Consumption")
plot(row_hydro)
```



```r
plot(row_renewable)
```



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?

> Answer: Total Renewable Energy Production has an increase over time with some seasonal autocorrelation effects as observed by the significance of partial autocorrelation over a gap of approximately 12 time steps. Hydroelectric Power Consumption appears to be noisily centred around a mean close to 80 but with stronger autocorrelation over shorter time lags of 2 time steps. Additionally there appears to be a stronger cyclicity for Hydroelectric Power consumption when looking at the partial autocorrelation function with positive and negative correlations at periodic intervals showing periodic resurgence and decline.

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.

```r
time_index <- 1:nrow(energy_ts)
model_renewable <- lm(energy_ts[, 1] ~ time_index) # lm(y given x)
```

```r
model_hydro     <- lm(energy_ts[, 2] ~ time_index)
trend_renewable <- fitted(model_renewable)
trend_hydro     <- fitted(model_hydro)
summary(model_renewable)
```

```
##
## Call:
## lm(formula = energy_ts[, 1] ~ time_index)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -154.81  -39.55   12.52   41.49  171.15
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 171.44868    5.11085   33.55   <2e-16 ***
## time_index    0.74999    0.01397   53.69   <2e-16 ***
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
##
## Residual standard error: 64.22 on 631 degrees of freedom
## Multiple R-squared:  0.8204, Adjusted R-squared:  0.8201
## F-statistic:  2883 on 1 and 631 DF,  p-value: < 2.2e-16
```

```r
summary(model_hydro)
```

```
##
## Call:
## lm(formula = energy_ts[, 2] ~ time_index)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -30.190 -10.214  -0.715   8.909  39.723
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 83.223802   1.110552  74.939  < 2e-16 ***
## time_index  -0.012199   0.003035  -4.019 6.55e-05 ***
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
##
## Residual standard error: 13.95 on 631 degrees of freedom
## Multiple R-squared:  0.02496,    Adjusted R-squared:  0.02342
## F-statistic: 16.15 on 1 and 631 DF,  p-value: 6.547e-05
```

```r
hydro_coeffs<-coefficients(model_hydro)
renew_coeffs<-coefficients(model_renewable)
print(hydro_coeffs)
```

```
## (Intercept)  time_index
## 83.22380232 -0.01219868
```

```
print(renew_coeffs)
```

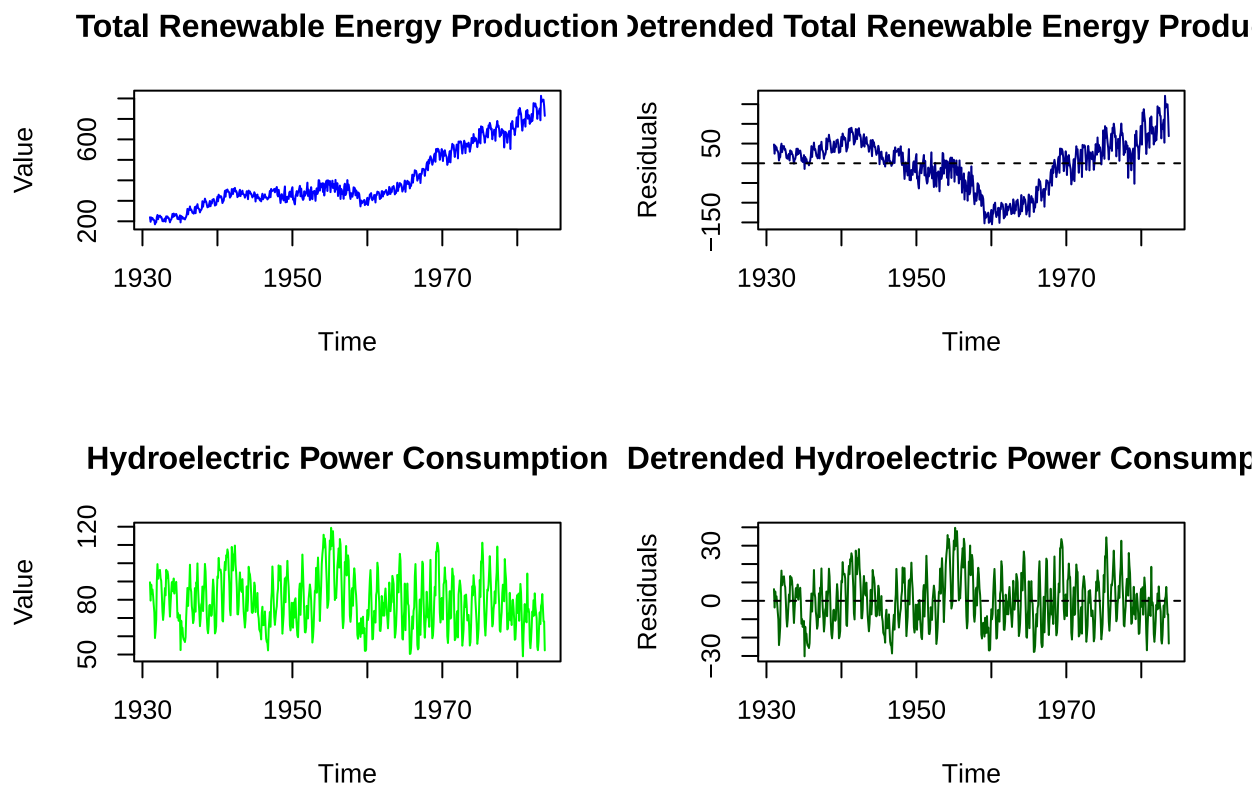
```
## (Intercept)  time_index
##  171.448682    0.749989
```

> Answer: The slope for renewable energy is positive (0.75) suggesting that there is a 0.75 unit change in the conditional mean of total renewable energy for a unit change in time. Conversely the slope for hydropower is close to zero (slighly negative) suggesting that there is no effect of time on the uptake of hydropower. The intercepts are somewhat less meaningful but suggest that based on the given data it is safe to assume that the baseline rate at the beginning of this time series for renewable energy would be about 171 while for hydropower it would be about 83.

Q4

Use the regression coefficients to detrend each series (subtract fitted linear trend). Plot detrended series and compare with the original time series from Q1. Describe what changed.

```
detrended_renewable <- energy_ts[, 1] - trend_renewable
detrended_hydro     <- energy_ts[, 2] - trend_hydro


par(mfrow = c(2, 2))
plot(energy_ts[, 1], main = "Total Renewable Energy Production", ylab = "Value", col = "blue", type = "
plot(detrended_renewable, main = "Detrended Total Renewable Energy Production", ylab = "Residuals", col
abline(h = 0, lty = 2)

plot(energy_ts[, 2], main = "Hydroelectric Power Consumption", ylab = "Value", col = "green", type = "l
plot(detrended_hydro, main = "Detrended Hydroelectric Power Consumption", ylab = "Residuals", col = "da
abline(h = 0, lty = 2)
```



```
par(mfrow = c(1, 1))
```

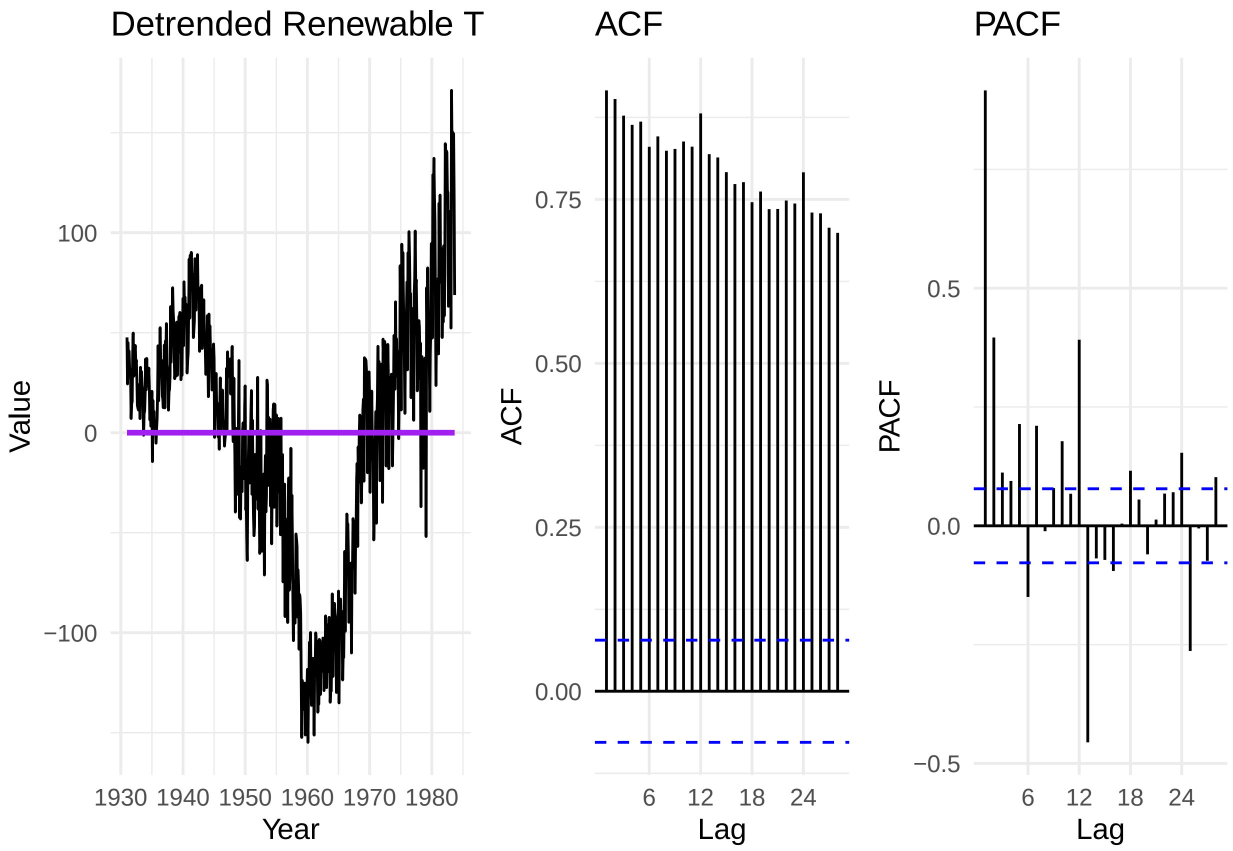
Answer: Since we remove the trend, we have effectively removed the positive drift observed in Renewable Energy, and are instead plotting the residuals on the y axis. This allows us to more prominently observe the second order effects for total renewable energy, where we can see that while between mid 1930's to 1940's and after 1970's the rate of growth was faster than the overall trendline(positive residuals), the growth was sluggish and below the trendline rate in the period between 1950 and 1970. Looking at the detrended hydroelectric consumption, it looks almost identical, validating the fact that the original data was likely stationary.

Q5

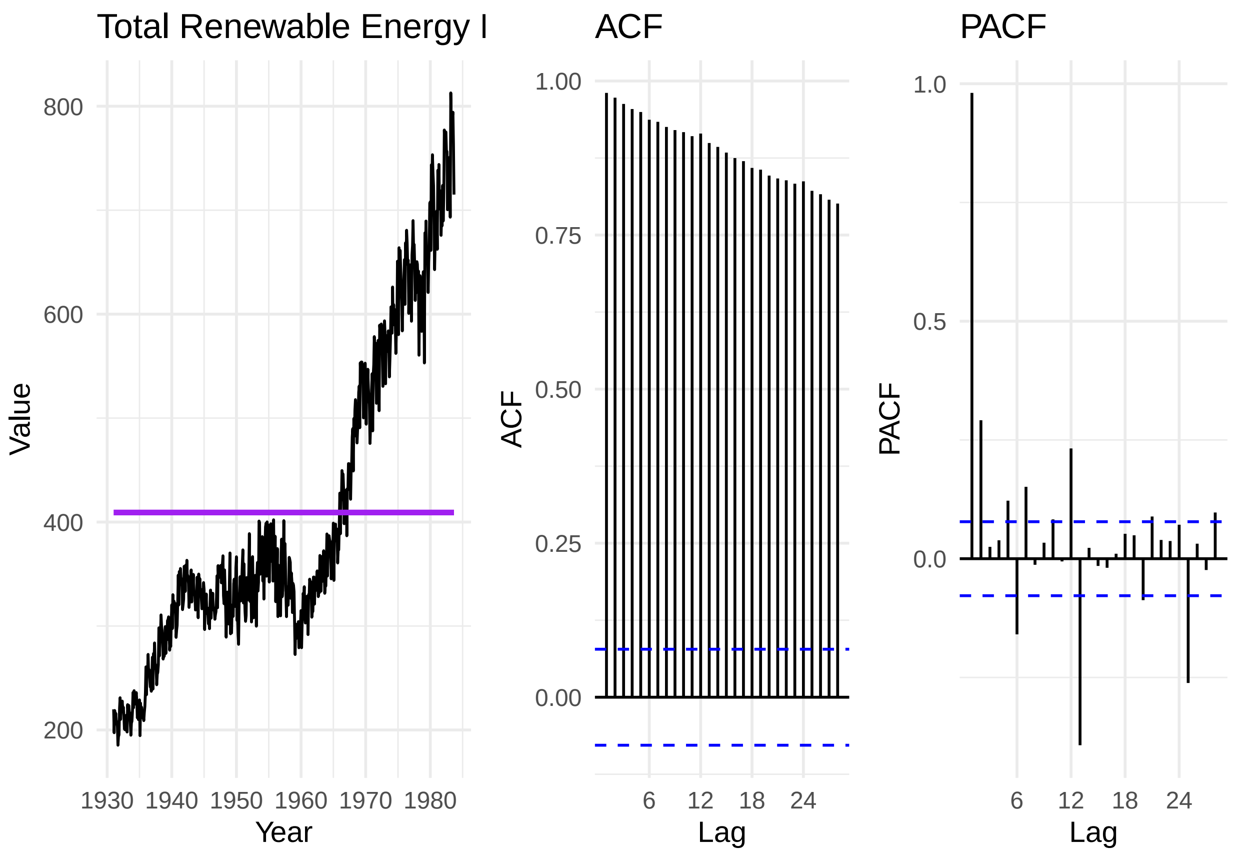
Plot ACF and PACF for the detrended series and compare with the plots from Q1. You may use plot_grid() again to get them side by side to make it easier to compare. Did the plots change? How?

```
detrended_renewable_ts <- ts(detrended_renewable,
                             start = start(energy_ts),
                             frequency = frequency(energy_ts))

detrended_hydro_ts     <- ts(detrended_hydro,
                             start = start(energy_ts),
                             frequency = frequency(energy_ts))

detrended_mts <- cbind(detrended_renewable_ts, detrended_hydro_ts)
```

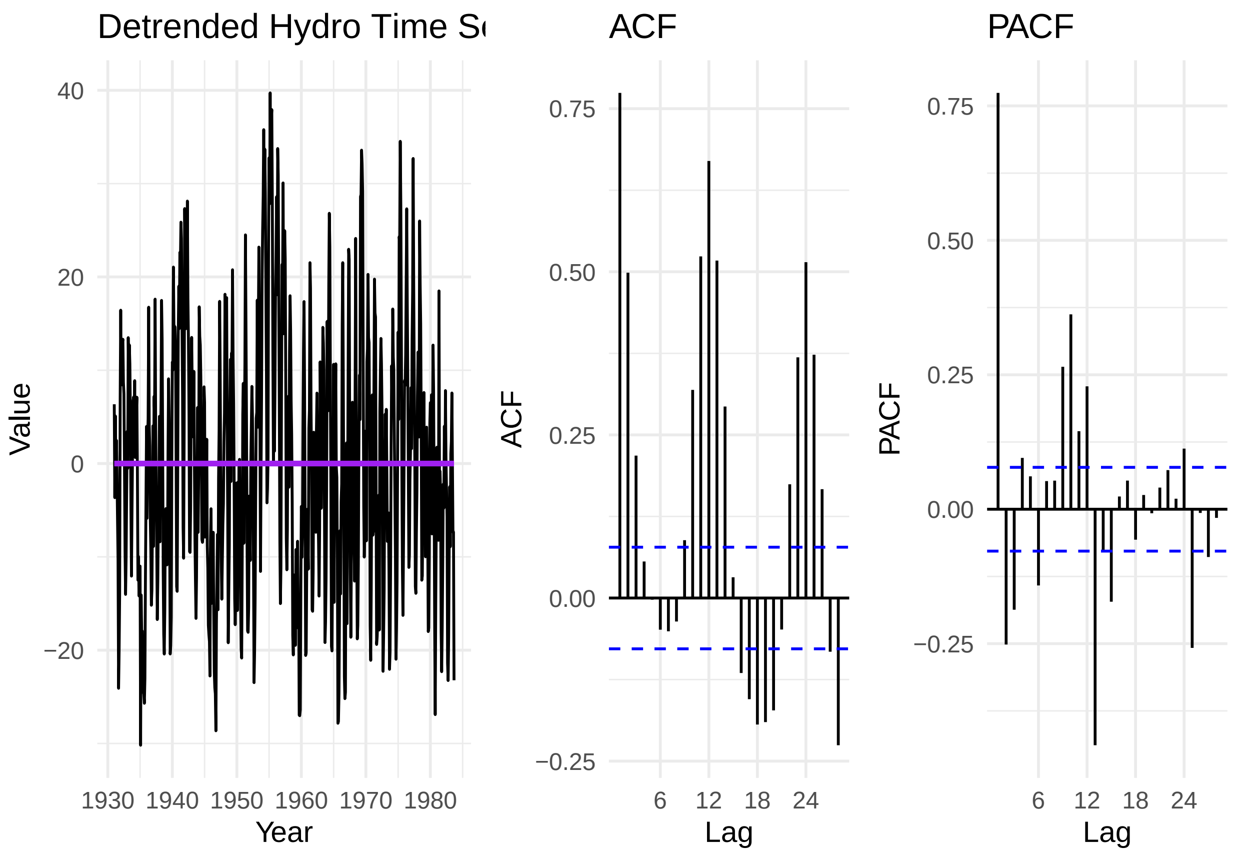
```
plot_ts_diagnostics(detrended_mts, 1, mean_values = c(0, 0), var_name = "Detrended Renewable")
```



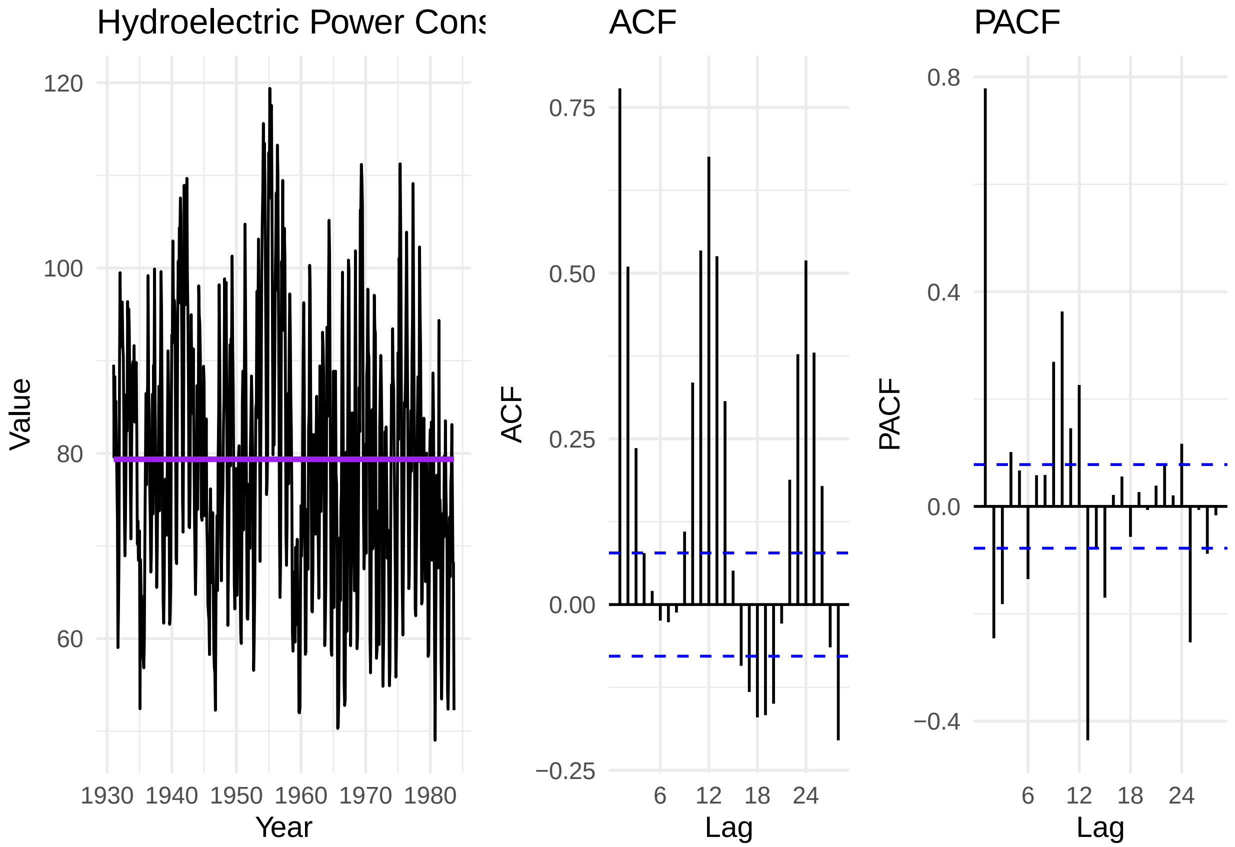
```
plot(row_renewable)
```



```
plot_ts_diagnostics(detrended_mts, 2, mean_values = c(0, 0), var_name = "Detrended Hydro")
```



```
plot(row_hydro)
```



>Answer: Compared to Q1, Detrended Total Renewable Energy Production also has an increase over time with some seasonal autocorrelation effects as observed by the significance of partial autocorrelation over a gap of approximately 12 time steps, however the degree of partial autocorrelation is higher and closer to 0.5 and -0.5. This helps us to see stronger correlation at different lags due to the detrending. The detrended Hydroelectric Power Consumption appear to be the same compared to Q1 with stronger autocorrelation over shorter time lags of 2 time steps, however, there is an even stronger cyclicity for Hydroelectric Power consumption with higher positive and negative correlations at periodic intervals showing periodic resurgence and decline.

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.

> Answer: It does have a seasonal trend as you can see autocorrelation for lag of multiples of 12 months (yearly seasonality).

Q7

Use function *lm()* to fit a seasonal means model (i.e. using the seasonal dummies) to 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?

```
dummies<-seasonaldummy(energy_ts[,1])

model_seas_renewable <- lm(energy_ts[, 1] ~ dummies) # using the original data and not the detrended da
dummies<-seasonaldummy(energy_ts[,2])

model_seas_hydro     <- tslm(energy_ts[, 2] ~ season)# using the original data and not the detrended da

seasonal_comp_renewable <- fitted(model_seas_renewable)
seasonal_comp_hydro     <- fitted(model_seas_hydro)

summary(model_seas_renewable)
```

```
##
## Call:
## lm(formula = energy_ts[, 1] ~ dummies)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -213.33  -97.36  -59.88  121.55  389.62
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  417.265     21.096  19.779   <2e-16 ***
## dummiesJan     2.090     29.693   0.070    0.944
## dummiesFeb   -34.524     29.693  -1.163    0.245
## dummiesMar     5.956     29.693   0.201    0.841
## dummiesApr    -6.900     29.693  -0.232    0.816
## dummiesMay     8.162     29.693   0.275    0.784
## dummiesJun    -2.231     29.693  -0.075    0.940
## dummiesJul     3.864     29.693   0.130    0.897
## dummiesAug    -3.978     29.693  -0.134    0.893
## dummiesSep   -29.033     29.693  -0.978    0.329
## dummiesOct   -19.937     29.834  -0.668    0.504
## dummiesNov   -20.617     29.834  -0.691    0.490
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 152.1 on 621 degrees of freedom
## Multiple R-squared:  0.008243,   Adjusted R-squared:  -0.009324
## F-statistic: 0.4692 on 11 and 621 DF,  p-value: 0.9223
```

```
summary(model_seas_hydro)
```

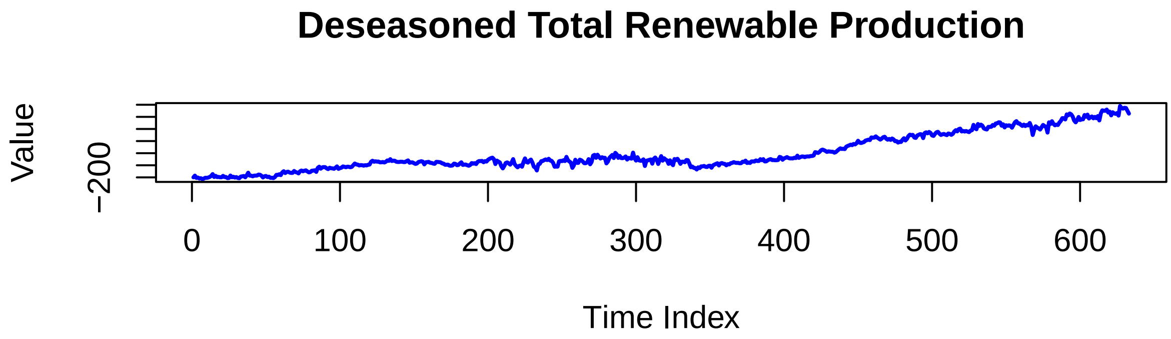
```
##
## Call:
## tslm(formula = energy_ts[, 2] ~ season)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -30.895  -6.368  -0.595   6.213  32.557
##
```

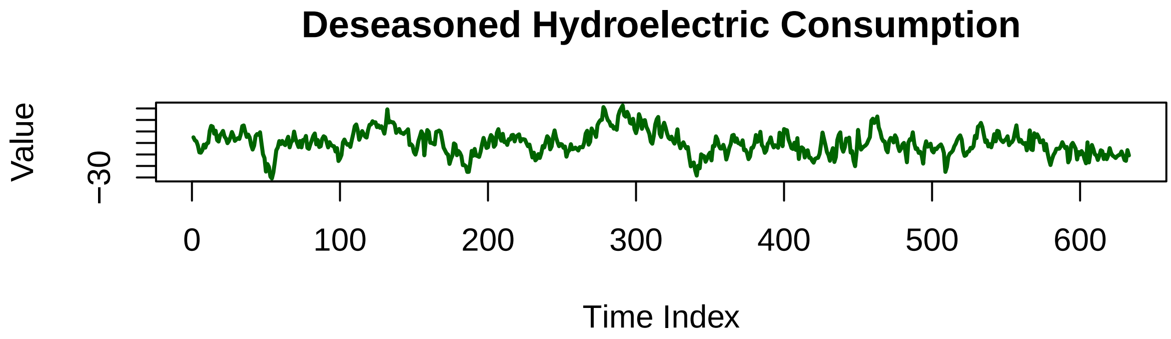
```
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  84.6753     1.4226  59.523  < 2e-16 ***
## season2      -7.3663     2.0118  -3.662 0.000272 ***
## season3       2.1649     2.0118   1.076 0.282292
## season4       0.6628     2.0118   0.329 0.741911
## season5       9.1288     2.0118   4.538 6.83e-06 ***
## season6       5.8288     2.0118   2.897 0.003896 **
## season7      -0.9476     2.0118  -0.471 0.637798
## season8     -10.2707     2.0118  -5.105 4.40e-07 ***
## season9     -21.5488     2.0118 -10.711  < 2e-16 ***
## season10    -21.2801     2.0215 -10.527  < 2e-16 ***
## season11    -15.7329     2.0215  -7.783 2.97e-14 ***
## season12     -4.9510     2.0215  -2.449 0.014591 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 10.36 on 621 degrees of freedom
## Multiple R-squared:  0.4714, Adjusted R-squared:  0.4621
## F-statistic: 50.35 on 11 and 621 DF,  p-value: < 2.2e-16
```

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?

```
y_renew_orig <- as.numeric(energy_ts[, 1])
y_hydro_orig <- as.numeric(energy_ts[, 2])

deseason_renewable <- y_renew_orig - as.numeric(fitted(model_seas_renewable))
deseason_hydro     <- y_hydro_orig - as.numeric(fitted(model_seas_hydro))

par(mfrow = c(2, 1))

plot(deseason_renewable, type = "l", col = "blue", lwd = 2,
     main = "Deseasoned Total Renewable Production",
     ylab = "Value", xlab = "Time Index")


plot(deseason_hydro, type = "l", col = "darkgreen", lwd = 2,
     main = "Deseasoned Hydroelectric Consumption",
     ylab = "Value", xlab = "Time Index")
```



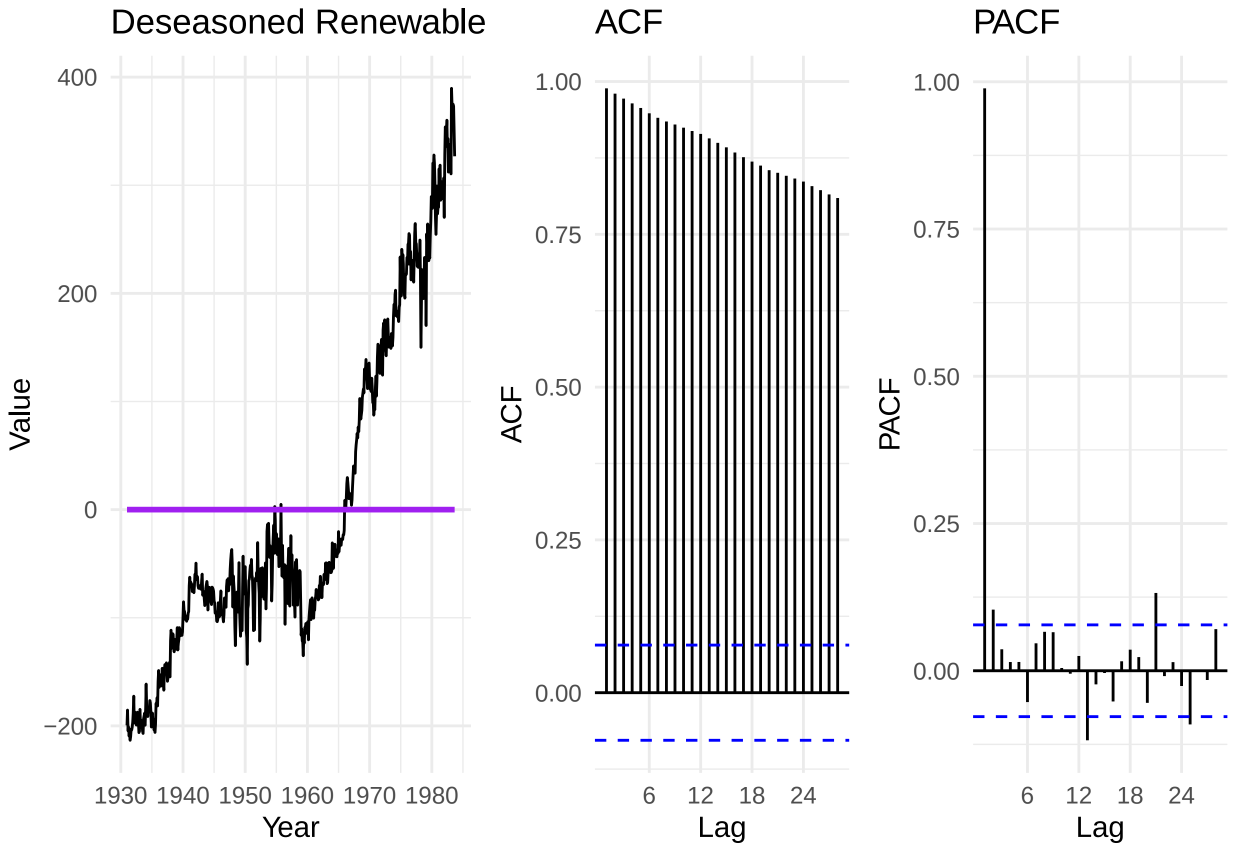


```r
par(mfrow = c(1, 1))
```

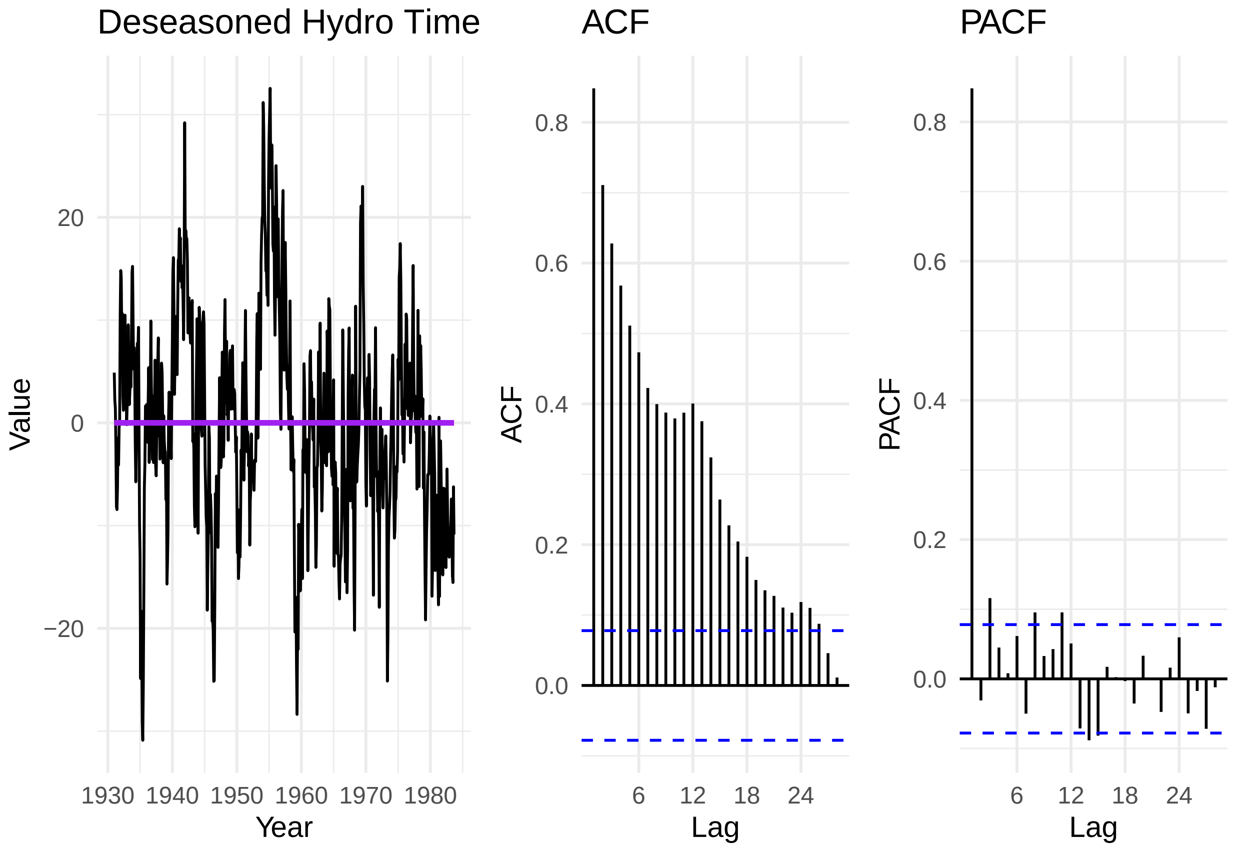
Q9

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. Did the plots change? How?

```r
deseason_renewable_ts <- ts(deseason_renewable,
                            start = start(energy_ts),
                            frequency = frequency(energy_ts))

deseason_hydro_ts     <- ts(deseason_hydro,
                            start = start(energy_ts),
                            frequency = frequency(energy_ts))
deseason_mts <- cbind(deseason_renewable_ts, deseason_hydro_ts)

means_orig <- c(0, 0)

plot_ts_diagnostics(deseason_mts, 1, mean_values = means_orig, var_name = "Deseasoned Renewable")
```



```r
plot_ts_diagnostics(deseason_mts, 2, mean_values = means_orig, var_name = "Deseasoned Hydro")
```



> Answer: Most of the effects we saw before were because of seasonality and now we observe that for renewable, the PACF is much smaller, though still significant for a lag of 12 months in both the time series, whereas for hydro, the PACF is smaller, but still slightly above the significant line at multiple different lags (2,8, etc.)