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

Assignment 3 - Due date 02/03/26

Mingjie Wei

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_Consumption 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.

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(cowplot)
library(ggplot2)
library(readxl)
library(openxlsx)
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
```

```
library(lubridate)
```

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

## The following object is masked from 'package:cowplot':
##
##     stamp

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

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.

```
#Import the dataset
energy_data <- read.xlsx(xlsxFile="../Data/Table_10.1_Renewable_Energy_Production_and_Consumption_by_Sou

#Extract the column names
read_col_names <- read.xlsx(xlsxFile="../Data/Table_10.1_Renewable_Energy_Production_and_Consumption_by_

#Assign the column names to the dataset
colnames(energy_data) <- read_col_names

#Visualize the first rows of the dataset
energy_data$Month <- convertToDate(energy_data$Month)

head(energy_data)
```

```
##        Month Wood Energy Production Biofuels Production
## 1 1973-01-01                129.630       Not Available
## 2 1973-02-01                117.194       Not Available
## 3 1973-03-01                129.763       Not Available
## 4 1973-04-01                125.462       Not Available
## 5 1973-05-01                129.624       Not Available
## 6 1973-06-01                125.435       Not Available
##   Total Biomass Energy Production Total Renewable Energy Production
## 1                         129.787                           219.839
## 2                         117.338                           197.330
## 3                         129.938                           218.686
## 4                         125.636                           209.330
```

```
## 5                      129.834                          215.982
## 6                      125.611                          208.249
##   Hydroelectric Power Consumption Geothermal Energy Consumption
## 1                          89.562                         0.490
## 2                          79.544                         0.448
## 3                          88.284                         0.464
## 4                          83.152                         0.542
## 5                          85.643                         0.505
## 6                          82.060                         0.579
##   Solar Energy Consumption Wind Energy Consumption Wood Energy Consumption
## 1            Not Available           Not Available                 129.630
## 2            Not Available           Not Available                 117.194
## 3            Not Available           Not Available                 129.763
## 4            Not Available           Not Available                 125.462
## 5            Not Available           Not Available                 129.624
## 6            Not Available           Not Available                 125.435
##   Waste Energy Consumption Biofuels Consumption
## 1                    0.157        Not Available
## 2                    0.144        Not Available
## 3                    0.176        Not Available
## 4                    0.174        Not Available
## 5                    0.210        Not Available
## 6                    0.176        Not Available
##   Total Biomass Energy Consumption Total Renewable Energy Consumption
## 1                          129.787                            219.839
## 2                          117.338                            197.330
## 3                          129.938                            218.686
## 4                          125.636                            209.330
## 5                          129.834                            215.982
## 6                          125.611                            208.249
```

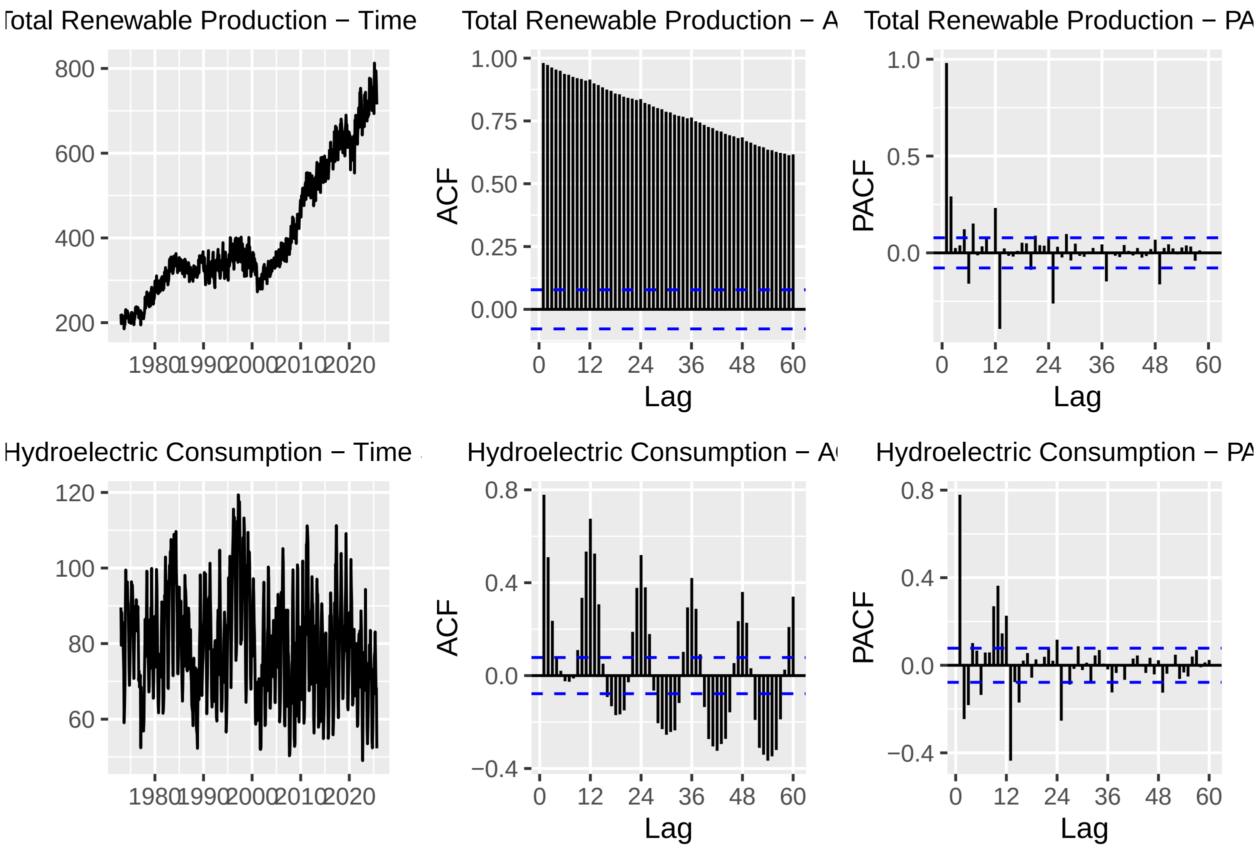
```r
# Select the columns
df2 <- energy_data %>%
  select(
    Month,
    `Total Renewable Energy Production`,
    `Hydroelectric Power Consumption`
  ) %>%
  rename(
    total_renew = `Total Renewable Energy Production`,
    hydro_cons  = `Hydroelectric Power Consumption`
  )

# Create time series
start_year  <- year(min(df2$Month))
start_month <- month(min(df2$Month))

ts_total <- ts(as.numeric(df2$total_renew), start = c(start_year, start_month), frequency = 12)
ts_hydro <- ts(as.numeric(df2$hydro_cons),  start = c(start_year, start_month), frequency = 12)

head(ts_total)
```

```
##          Jan     Feb     Mar     Apr     May     Jun
## 1973 219.839 197.330 218.686 209.330 215.982 208.249
```

```r
head(ts_hydro)
```

```
##         Jan    Feb    Mar    Apr    May    Jun
## 1973 89.562 79.544 88.284 83.152 85.643 82.060
```

```r
# Plots
make_3plots <- function(tsx, series_name = "") {

  p_ts <- autoplot(tsx) +
    ggtitle(paste0(series_name, " - Time Series")) +
    xlab("") + ylab("") +
    theme(plot.title = element_text(size = 10, hjust = 0.5),
          plot.margin = margin(5.5, 12, 5.5, 5.5))

  p_acf <- ggAcf(tsx, lag.max = 60) +
    ggtitle(paste0(series_name, " - ACF")) +
    theme(plot.title = element_text(size = 10, hjust = 0.5),
          plot.margin = margin(5.5, 12, 5.5, 5.5))

  p_pacf <- ggPacf(tsx, lag.max = 60) +
    ggtitle(paste0(series_name, " - PACF")) +
    theme(plot.title = element_text(size = 10, hjust = 0.5),
          plot.margin = margin(5.5, 12, 5.5, 5.5))

  cowplot::plot_grid(p_ts, p_acf, p_pacf, nrow = 1)
}

row1 <- make_3plots(ts_total, "Total Renewable Production")
row2 <- make_3plots(ts_hydro, "Hydroelectric Consumption")

cowplot::plot_grid(row1, row2, ncol = 1)
```



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?

Total Renewable Energy Production:

Yes, it clearly shows a strong upward trend over time. The time series plot rises from around the 1970s to the 2020s, with faster growth after the mid-2000s. The ACF stays very high and decays very slowly across many lags, which is consistent with a non-stationary series driven by a trend.

Hydroelectric Power Consumption:

It does not show a clear long-term increasing or decreasing trend. The series fluctuates around a roughly stable level over the decades. Instead, it shows strong seasonality: the ACF has noticeable spikes at lags 12, 24, 36, etc., indicating a yearly (12-month) seasonal pattern rather than a trend.

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.

```
# ========== Linear Trend: Total Renewable Production ==========
y_total <- as.numeric(ts_total)
t_total <- 1:length(y_total)
```

```r
lm_total <- lm(y_total ~ t_total)
summary(lm_total)
```

```
##
## Call:
## lm(formula = y_total ~ t_total)
##
## 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 ***
## t_total       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
# Save coefficients
beta0_total <- coef(lm_total)[1]   # intercept
beta1_total <- coef(lm_total)[2]   # slope

beta0_total; beta1_total
```

```
## (Intercept)
##    171.4487
```

```
##  t_total
## 0.749989
```

```r
# ========== Linear Trend: Hydroelectric Consumption ==========
y_hydro <- as.numeric(ts_hydro)
t_hydro <- 1:length(y_hydro)

lm_hydro <- lm(y_hydro ~ t_hydro)
summary(lm_hydro)
```

```
##
## Call:
## lm(formula = y_hydro ~ t_hydro)
##
## 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 ***
## t_hydro     -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
```

```
# Save coefficients
beta0_hydro <- coef(lm_hydro)[1]   # intercept
beta1_hydro <- coef(lm_hydro)[2]   # slope

beta0_hydro; beta1_hydro
```

```
## (Intercept)
##     83.2238

##     t_hydro
## -0.01219868
```

Total Renewable Energy Production:

- Slope (0.749989): The estimated slope is about 0.75 units per month, meaning that, on average, Total Renewable Energy Production increases by roughly 0.75 units each month under a linear trend model. Converting to a yearly rate, this is about $0.749989 \times 12 \approx 9.00$ units per year. The slope is highly statistically significant (p-value < 2e-16), so we have strong evidence of an upward linear trend.
- Intercept (171.4487): The intercept is the fitted value when t=0. Since our time index starts at t=1, the intercept mainly sets the baseline level of the fitted line and is not directly meaningful by itself, but together with the slope it defines the linear trend line.

Hydroelectric Power Consumption:

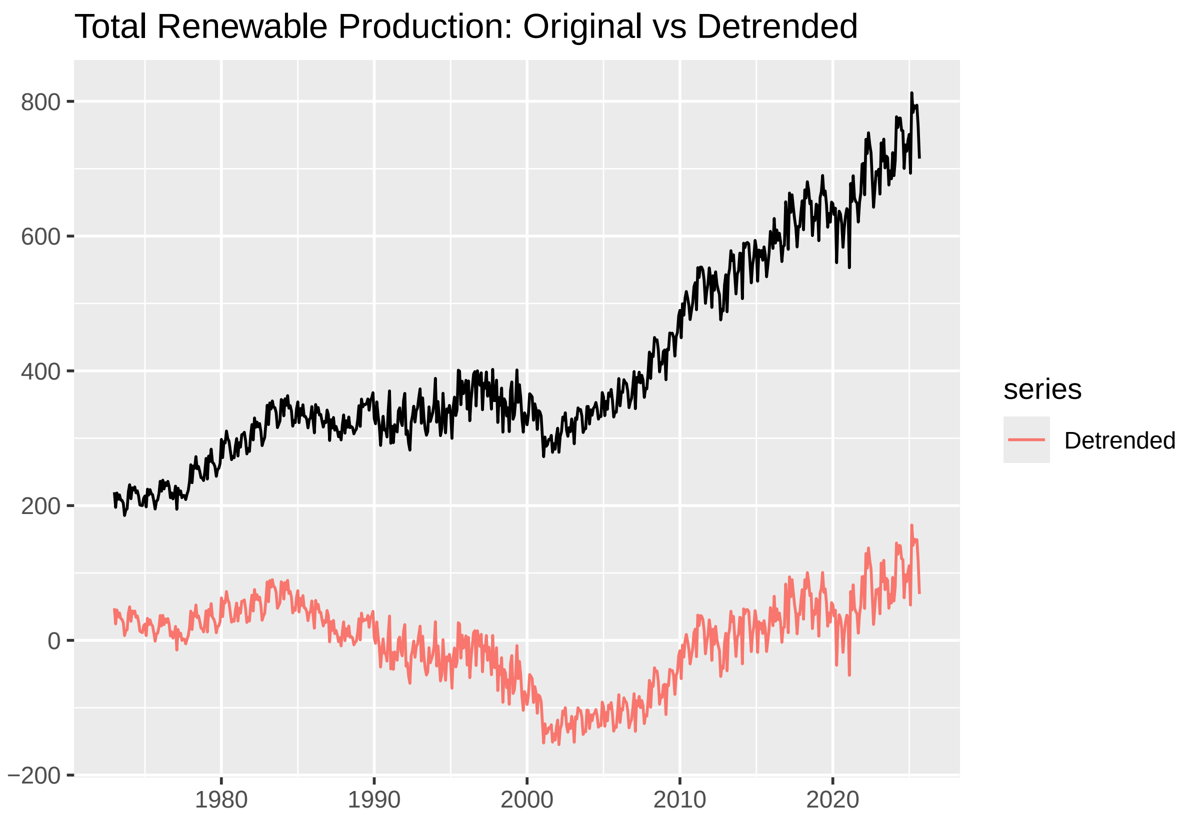
- Slope (-0.0122): The estimated slope is about -0.012 units per month, which suggests a very slight downward linear trend. In yearly terms, this is $-0.01219868 \times 12 \approx -0.15$ units per year, which is extremely small compared with the month-to-month variability seen in the plot. The slope is statistically significant (p-value = 6.55e-05), so the decrease is unlikely to be due to random sampling variation alone.
- Intercept (83.2238): This is the fitted value at t=0. As above, it mainly provides the baseline level for the fitted line and is not directly interpretable on its own.

Q4

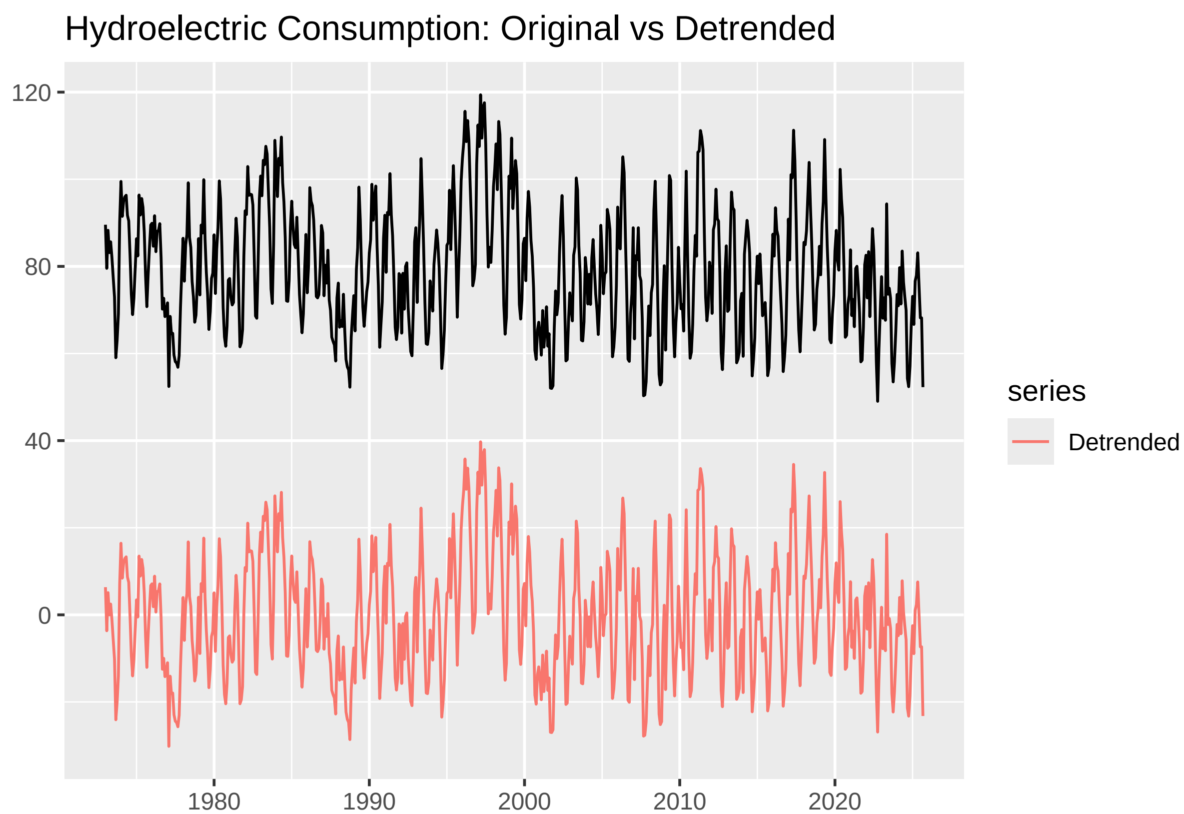
Use the regression coefficients to detrend each series (subtract fitted linear trend). Plot detrended series and compare with the original time series from Q1. Describe what changed.

```
# ---------- Detrend Total Renewable Production ----------
t_total <- 1:length(ts_total)
trend_total <- beta0_total + beta1_total * t_total

ts_total_detr <- ts(as.numeric(ts_total) - trend_total,
                    start = start(ts_total),
                    frequency = frequency(ts_total))

# ---------- Detrend Hydroelectric Consumption ----------
t_hydro <- 1:length(ts_hydro)
trend_hydro <- beta0_hydro + beta1_hydro * t_hydro

ts_hydro_detr <- ts(as.numeric(ts_hydro) - trend_hydro,
                    start = start(ts_hydro),
                    frequency = frequency(ts_hydro))
```

```r
# Total: Original vs Detrended
autoplot(ts_total) +
  autolayer(ts_total_detr, series = "Detrended") +
  ggtitle("Total Renewable Production: Original vs Detrended") +
  xlab("") + ylab("")
```




```r
# Hydro: Original vs Detrended
autoplot(ts_hydro) +
  autolayer(ts_hydro_detr, series = "Detrended") +
  ggtitle("Hydroelectric Consumption: Original vs Detrended") +
  xlab("") + ylab("")
```



Total Renewable Production:

After detrending (subtracting the fitted linear trend), the strong upward growth is removed and the detrended series fluctuates around zero. However, the detrended series still shows long-run swings: it is mostly positive in the earlier years, becomes strongly negative around the early 2000s, and turns positive again after about 2010 with larger positive deviations in recent years. This suggests that the original trend is not perfectly linear; instead, the growth rate changes over time (i.e., there are nonlinear/structural trend components that remain after removing a single linear trend).

Hydroelectric Consumption:

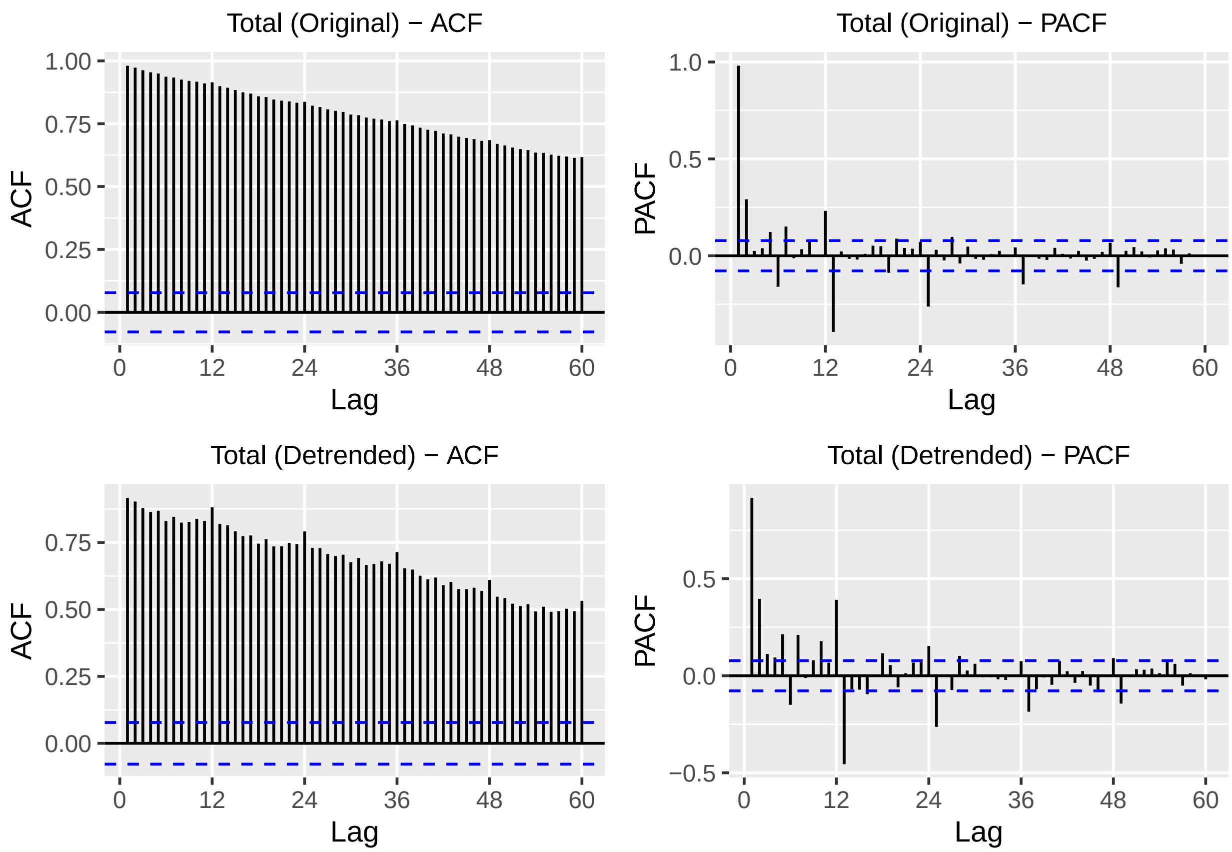
Detrending has only a minor effect. The detrended series looks very similar to the original series, except it is centered closer to zero. This is because the fitted linear trend is very small compared to the large seasonal and irregular fluctuations. Therefore, most of the variability remains after detrending, indicating that this series is dominated by seasonality rather than a strong long-term trend.

Q5

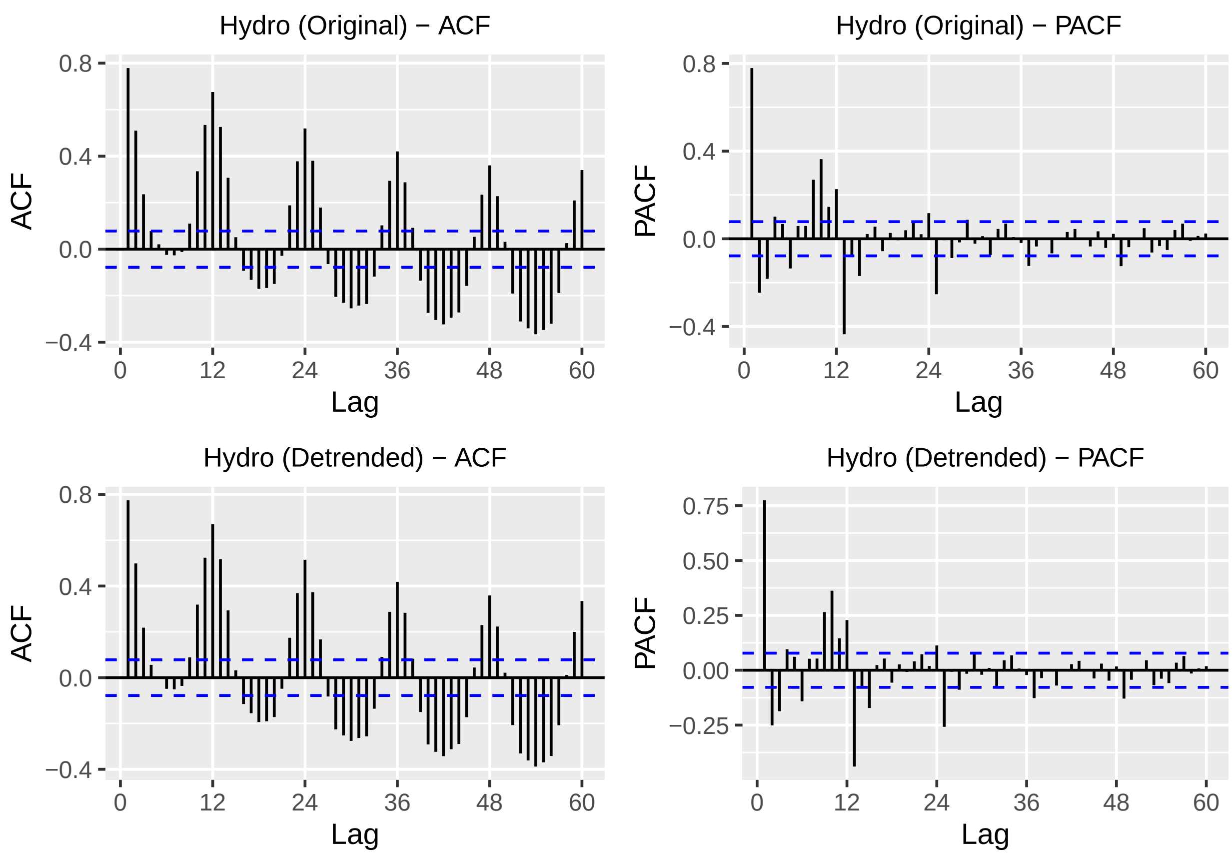
Plot ACF and PACF for the detrended series and compare with the plots from Q1. You may use plot_grid() again to get them side by side to make it easier to compare. Did the plots change? How?

```r
# ACF/PACF plotting
acf_pacf_pair <- function(tsx, title_prefix="", lag_max=60) {
  p_acf <- ggAcf(tsx, lag.max = lag_max) +
    ggtitle(paste0(title_prefix, "ACF")) +
    theme(plot.title = element_text(size = 10, hjust = 0.5))
```

```
  p_pacf <- ggPacf(tsx, lag.max = lag_max) +
    ggtitle(paste0(title_prefix, "PACF")) +
    theme(plot.title = element_text(size = 10, hjust = 0.5))

  plot_grid(p_acf, p_pacf, nrow = 1)
}

# ---- Total Renewable: Original vs Detrended ----
row_total_orig <- acf_pacf_pair(ts_total, "Total (Original) - ")
row_total_detr <- acf_pacf_pair(ts_total_detr, "Total (Detrended) - ")

plot_grid(row_total_orig, row_total_detr, ncol = 1)
```




```
# ---- Hydroelectric: Original vs Detrended ----
row_hydro_orig <- acf_pacf_pair(ts_hydro, "Hydro (Original) - ")
row_hydro_detr <- acf_pacf_pair(ts_hydro_detr, "Hydro (Detrended) - ")

plot_grid(row_hydro_orig, row_hydro_detr, ncol = 1)
```



Total Renewable Production:

Yes, the plots change, but the change is moderate rather than dramatic.

- In the original series, the ACF is very high and decays slowly across many lags, consistent with strong persistence caused by trend/non-stationarity.
- After detrending, the ACF is still positive and decays slowly, but its overall magnitude is lower (weaker long-run dependence), indicating that removing the linear trend reduced some of the persistence.
- The PACF in both cases has a large spike at lag 1 and then mostly small values, but the detrended series shows a few more noticeable spikes at specific lags, suggesting that some non-linear/structural components remain even after removing a simple linear trend.

Hydroelectric Consumption:

- The seasonal pattern (spikes at lags 12, 24, 36, . . .) remains almost identical in the ACF.
- This indicates that the series is dominated by seasonality, and subtracting a small linear trend does not materially change the correlation structure.

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:

Total Renewable Production:

Seasonality does **not** appear to be strong. The time series plot is dominated by a clear long-term upward trend (especially after the mid-2000s). In the ACF, correlations remain very high and decay slowly across many lags, which is typical of a trend-driven (non-stationary) series rather than a series with a clear repeating seasonal cycle (e.g., strong spikes at lags 12, 24, 36, . . .).

Hydroelectric Consumption:

Seasonality appears to be **strong**, with an annual (12-month) cycle. The time series shows repeating up-and-down patterns over time, and the ACF has clear significant spikes at lags 12, 24, 36, 48, 60, which is a classic signature of yearly seasonality in monthly data.

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?

```r
# ===== Seasonal Means Model (Total) =====
dum_total <- seasonaldummy(ts_total)
lm_season_total <- lm(as.numeric(ts_total) ~ dum_total)
summary(lm_season_total)
```

```
##
## Call:
## lm(formula = as.numeric(ts_total) ~ dum_total)
##
## 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 ***
## dum_totalJan    2.090     29.693   0.070    0.944
## dum_totalFeb  -34.524     29.693  -1.163    0.245
## dum_totalMar    5.956     29.693   0.201    0.841
## dum_totalApr   -6.900     29.693  -0.232    0.816
## dum_totalMay    8.162     29.693   0.275    0.784
## dum_totalJun   -2.231     29.693  -0.075    0.940
## dum_totalJul    3.864     29.693   0.130    0.897
## dum_totalAug   -3.978     29.693  -0.134    0.893
## dum_totalSep  -29.033     29.693  -0.978    0.329
## dum_totalOct  -19.937     29.834  -0.668    0.504
## dum_totalNov  -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
```

```r
# Save coefficients
coef_season_total <- coef(lm_season_total)
```

```r
# ===== Seasonal Means Model (Hydro) =====
dum_hydro <- seasonaldummy(ts_hydro)
lm_season_hydro <- lm(as.numeric(ts_hydro) ~ dum_hydro)
summary(lm_season_hydro)
```

```
##
## Call:
## lm(formula = as.numeric(ts_hydro) ~ dum_hydro)
##
## 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)    79.724      1.436  55.511  < 2e-16 ***
## dum_hydroJan    4.951      2.021   2.449 0.014591 *
## dum_hydroFeb   -2.415      2.021  -1.195 0.232608
## dum_hydroMar    7.116      2.021   3.520 0.000463 ***
## dum_hydroApr    5.614      2.021   2.777 0.005649 **
## dum_hydroMay   14.080      2.021   6.965 8.38e-12 ***
## dum_hydroJun   10.780      2.021   5.333 1.36e-07 ***
## dum_hydroJul    4.003      2.021   1.980 0.048091 *
## dum_hydroAug   -5.320      2.021  -2.632 0.008710 **
## dum_hydroSep  -16.598      2.021  -8.211 1.28e-15 ***
## dum_hydroOct  -16.329      2.031  -8.040 4.56e-15 ***
## dum_hydroNov  -10.782      2.031  -5.308 1.54e-07 ***
## ---
## 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
```

```r
# Save coefficients
coef_season_hydro <- coef(lm_season_hydro)
```

Total Renewable Production:

From the regression output:

- Overall seasonality test: The overall F-test for the 11 monthly dummies is not significant ($F = 0.4692$, $p = 0.9223$). This indicates there is no evidence of seasonality in Total Renewable Energy Production.
- Individual months: None of the monthly dummy coefficients are statistically significant (all p-values are large), meaning monthly means are not meaningfully different from the baseline month.

Hydroelectric Consumption:

Using the same seasonal means model:

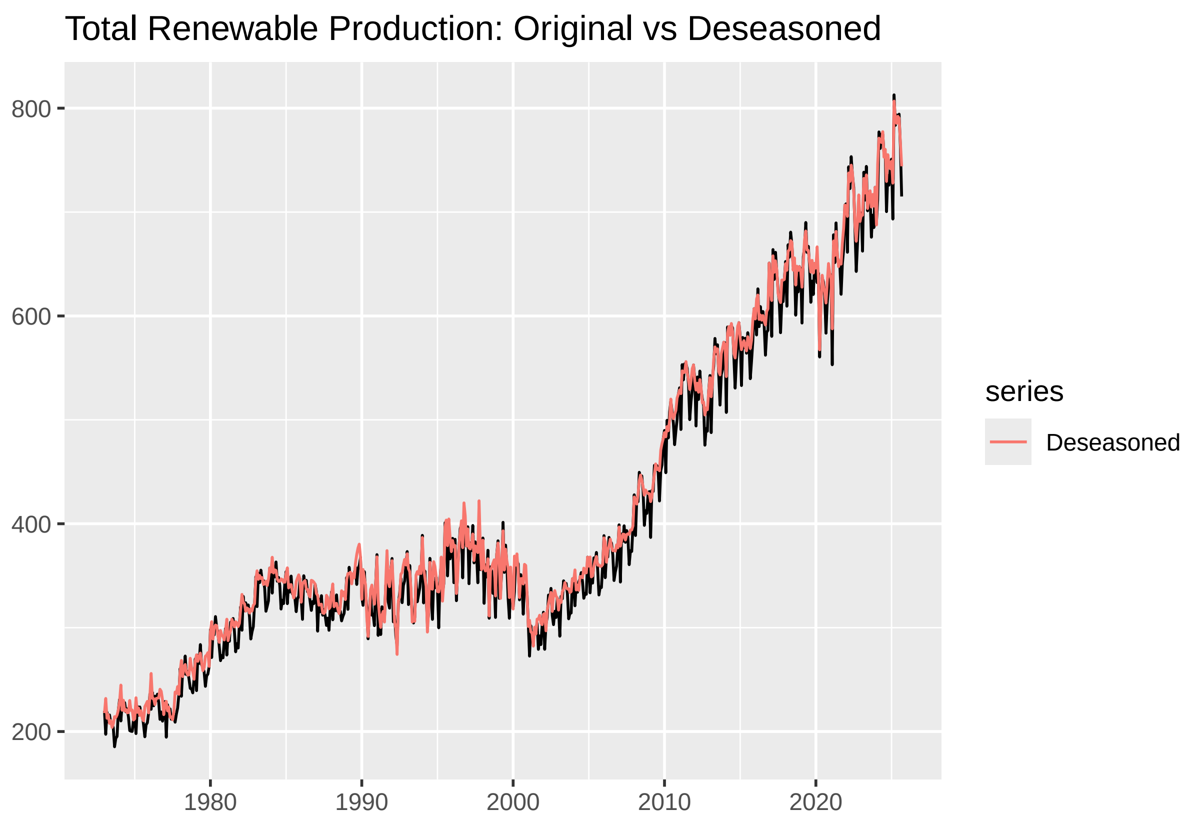
- Overall seasonality test: The overall F-test is highly significant ($F = 50.35$, $p < 2.2e\text{-}16$), providing strong evidence that monthly means differ (i.e., the series has strong seasonality).
- Which months differ: Many months have significant coefficients relative to the baseline month. For example, May (+14.08) and June (+10.78) are much higher than baseline, while September (-16.60) and

October (-16.33) are much lower, indicating a clear seasonal cycle with high values in late spring/early summer and low values in early fall.

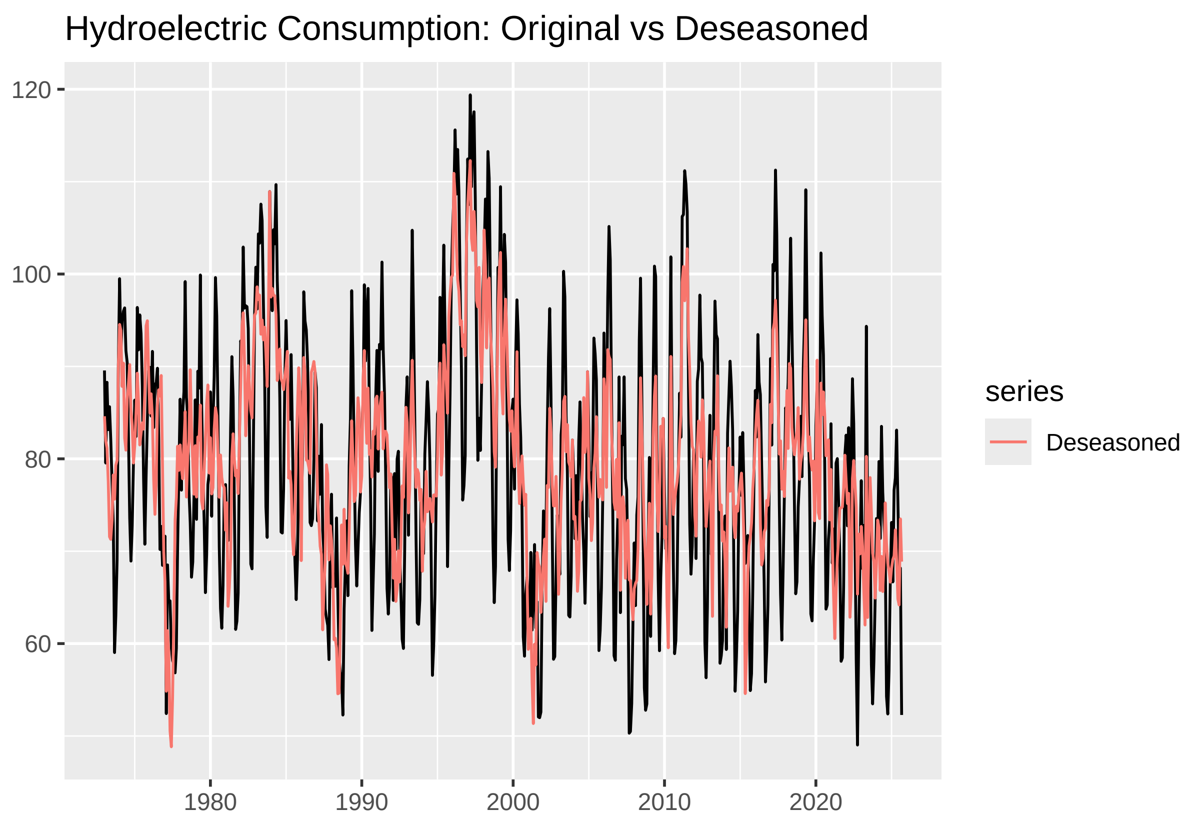
Q8

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?

```
# ===== Deseason: remove only seasonal effect =====

# Total
fit_total <- fitted(lm_season_total)
b0_total_season <- coef(lm_season_total)[1]
season_total <- fit_total - b0_total_season
ts_total_deseason <- ts(as.numeric(ts_total) - season_total,
                        start = start(ts_total), frequency = frequency(ts_total))

# Hydro
fit_hydro <- fitted(lm_season_hydro)
b0_hydro_season <- coef(lm_season_hydro)[1]
season_hydro <- fit_hydro - b0_hydro_season
ts_hydro_deseason <- ts(as.numeric(ts_hydro) - season_hydro,
                        start = start(ts_hydro), frequency = frequency(ts_hydro))

# ===== Plots =====

autoplot(ts_total) +
  autolayer(ts_total_deseason, series="Deseasoned") +
  ggtitle("Total Renewable Production: Original vs Deseasoned") +
  xlab("") + ylab("")
```




```r
autoplot(ts_hydro) +
  autolayer(ts_hydro_deseason, series="Deseasoned") +
  ggtitle("Hydroelectric Consumption: Original vs Deseasoned") +
  xlab("") + ylab("")
```



Total Renewable Production:

Very little changed after deseasoning. The deseasoned series almost overlaps with the original series, which is consistent with the seasonal means model results from Q7 (overall F-test p = 0.9223 and $R^2 = 0.008$). This indicates that Total Renewable Production has negligible monthly seasonality.

Hydroelectric Consumption:

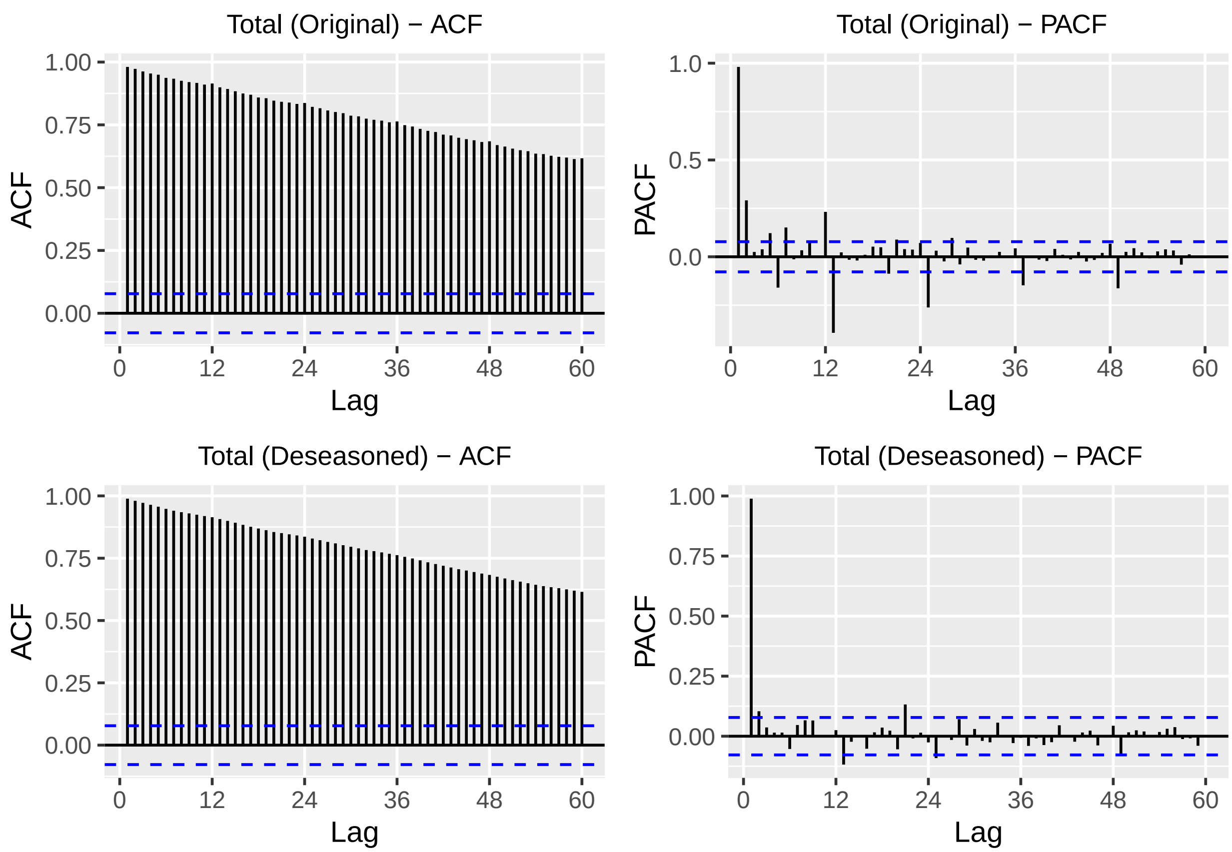
Yes, the series changed noticeably after deseasoning. The repeating within-year seasonal pattern is much weaker in the deseasoned series, and the series becomes smoother around a more stable level. This matches the strong seasonality found in Q7 (overall F-test $p < 2.2e\text{-}16$ and $R^2 = 0.47$), meaning monthly seasonality explains a large portion of the variability in Hydroelectric Consumption.

Q9

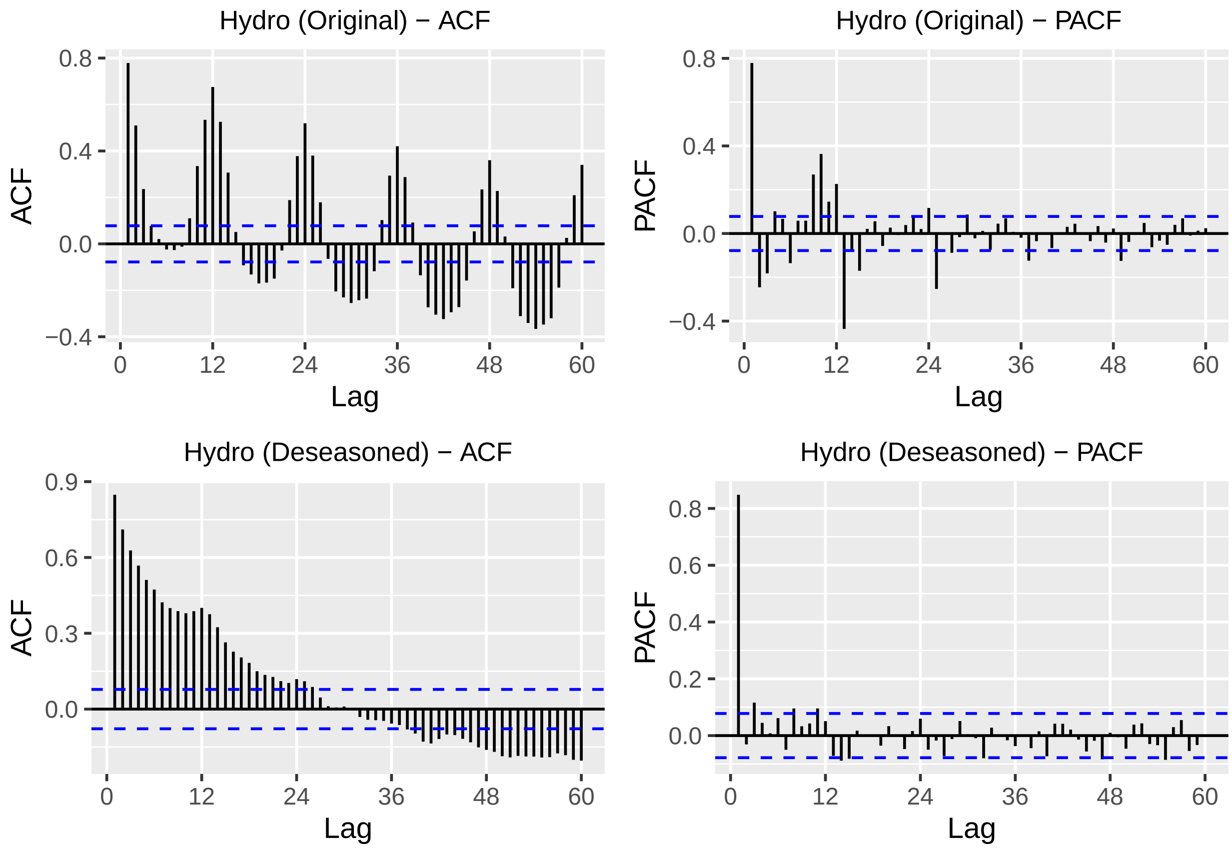
Plot ACF and PACF for the deseason series and compare with the plots from Q1. You may use plot_grid() again to get them side by side. Did the plots change? How?

```
acf_pacf_pair <- function(tsx, title_prefix="", lag_max=60) {
  p_acf <- ggAcf(tsx, lag.max = lag_max) +
    ggtitle(paste0(title_prefix, "ACF")) +
    theme(plot.title = element_text(size = 10, hjust = 0.5))

  p_pacf <- ggPacf(tsx, lag.max = lag_max) +
    ggtitle(paste0(title_prefix, "PACF")) +
    theme(plot.title = element_text(size = 10, hjust = 0.5))

  plot_grid(p_acf, p_pacf, nrow = 1)
```

```r
}

# ---- Total: Original vs Deseasoned ----
row_total_orig <- acf_pacf_pair(ts_total, "Total (Original) - ")
row_total_des  <- acf_pacf_pair(ts_total_deseason, "Total (Deseasoned) - ")
plot_grid(row_total_orig, row_total_des, ncol = 1)
```




```r
# ---- Hydro: Original vs Deseasoned ----
row_hydro_orig <- acf_pacf_pair(ts_hydro, "Hydro (Original) - ")
row_hydro_des  <- acf_pacf_pair(ts_hydro_deseason, "Hydro (Deseasoned) - ")
plot_grid(row_hydro_orig, row_hydro_des, ncol = 1)
```



Total Renewable Production:

The ACF and PACF plots changed very little after deseasoning. In both the original and the deseasoned series, the ACF remains very high and decays slowly across many lags, and the PACF shows a large spike at lag 1 with most other lags close to zero. This is consistent with Q7, where the seasonal means model for Total was not significant ($p = 0.9223$, $R^2 = 0.008$), so removing seasonality has almost no effect on the correlation structure.

Hydroelectric Consumption:

Yes, the ACF and PACF plots changed noticeably after deseasoning. In the original series, the ACF shows clear seasonal behavior with strong spikes at lags 12, 24, 36, … . After deseasoning, the ACF becomes dominated by strong positive short-lag autocorrelation that decays gradually, and the seasonal pattern is less pronounced in the PACF (most spikes beyond the first few lags are smaller). Overall, removing the monthly mean seasonal component changes the dependence structure, although some seasonal correlation can still remain because deseasoning removes monthly mean differences but does not necessarily eliminate seasonal dynamics.