

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

Assignment 3 - Due date 02/03/26

Si Min (Lucy) Loo

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)
```

```
## Warning: package 'Kendall' was built under R version 4.4.3
```

```
library(readxl)
```

```
energy_data <- read_excel(path="../Data/Table_10.1_Renewable_Energy_Production_and_Consumption_by_Source")
```

```

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

read_col_names <- read_excel(path="../Data/Table_10.1_Renewable_Energy_Production_and_Consumption_by_Sor

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

colnames(energy_data) <- read_col_names

df <- data.frame(
  Month = energy_data$`Month`,
  Renewable = energy_data$`Total Renewable Energy Production`,
  Hydroelectric = energy_data$`Hydroelectric Power Consumption`
)

head(df)

##           Month Renewable Hydroelectric
## 1 1973-01-01    219.839         89.562
## 2 1973-02-01    197.330         79.544
## 3 1973-03-01    218.686         88.284
## 4 1973-04-01    209.330         83.152
## 5 1973-05-01    215.982         85.643
## 6 1973-06-01    208.249         82.060

##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(cowplot)
library(ggplot2)

start_year <- as.numeric(format(df$Month[1], "%Y"))
start_month <- as.numeric(format(df$Month[1], "%m"))

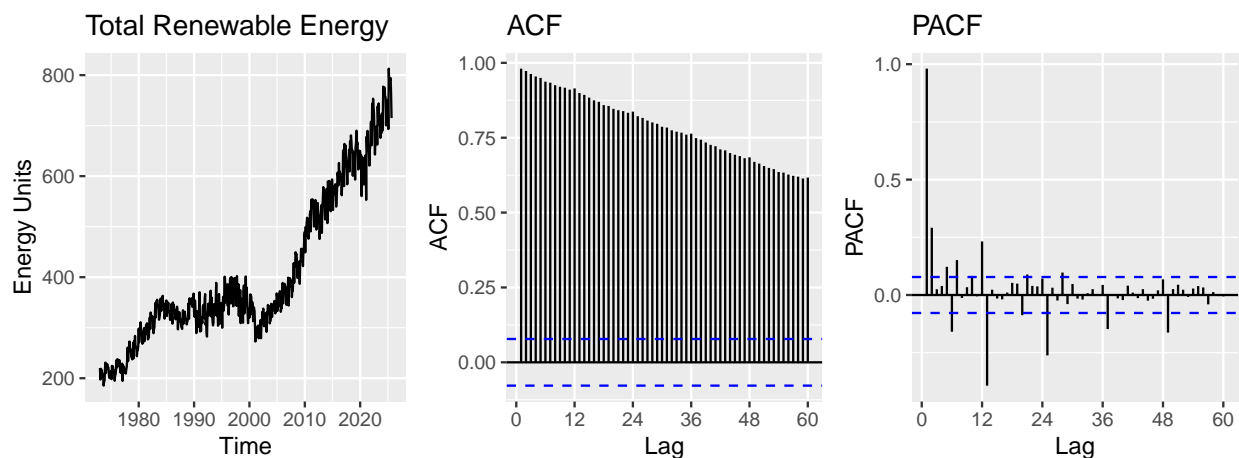
renew_ts <- ts(df$Renewable, frequency = 12, start = c(start_year, start_month))
hydro_ts <- ts(df$Hydroelectric, frequency = 12, start = c(start_year, start_month))

p1_renew <- autoplot(renew_ts) +
  labs(title = "Total Renewable Energy",
       x = "Time", y = "Energy Units")

p2_renew <- autoplot(Acf(renew_ts, lag.max = 60, plot = FALSE)) +
  labs(title = "ACF",
       x = "Lag", y = "ACF")

p3_renew <- autoplot(Pacf(renew_ts, lag.max = 60, plot = FALSE)) +
  labs(title = "PACF",
       x = "Lag", y = "PACF")

cowplot::plot_grid(p1_renew, p2_renew, p3_renew,
  nrow = 1, align = "h")
```



```
p1_hydro <- autoplot(hydro_ts) +
  labs(title = "Hydroelectric Power",
       x = "Time", y = "Energy Units")

p2_hydro <- autoplot(Acf(hydro_ts, lag.max = 60, plot = FALSE)) +
  labs(title = "ACF",
       x = "Lag", y = "ACF")

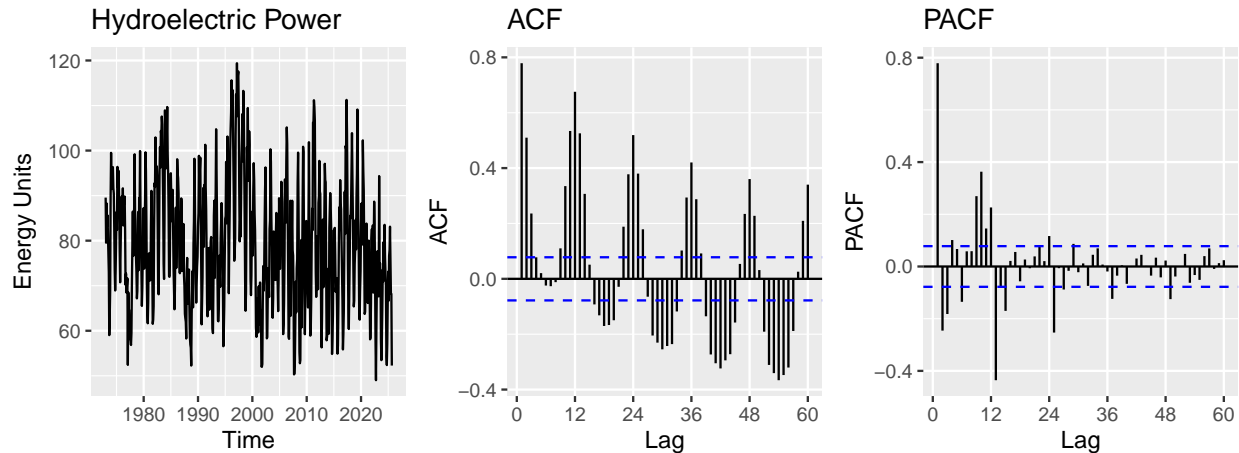
p3_hydro <- autoplot(Pacf(hydro_ts, lag.max = 60, plot = FALSE)) +
```

```

labs(title = "PACF",
     x = "Lag", y = "PACF")

plot_grid(p1_hydro, p2_hydro, p3_hydro,
         nrow = 1, align = "v")

```



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 seems to display an increasing trend over time. This is confirmed by the ACF, which remains very high and decays slowly across many lags rather than cutting off quickly. The PACF shows a very large and significant spike at lag 1, followed by some but mostly insignificant values at higher lags.

Meanwhile, hydroelectric power consumption fluctuates around a relatively stable mean with no obvious long-term trend. The ACF displays a clear seasonal pattern. The PACF also shows a dominant spike at lag 1 and additional significant seasonal effects around lag 12, followed by rapid decay.

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.

```

lm_renew <- lm(formula = Renewable ~ Month, data = df)
summary(lm_renew)

```

```

##
## Call:
## lm(formula = Renewable ~ Month, data = df)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -154.84  -39.52   12.50   41.50  171.17
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  1.452e+02  5.540e+00  26.21  <2e-16 ***
## Month        2.852e-07  5.312e-09   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

beta0_renew <- as.numeric(lm_renew$coefficients[1]) #intercept
beta1_renew <- as.numeric(lm_renew$coefficients[2]) #slope

beta0_renew
```

```
## [1] 145.2216
```

```
beta1_renew
```

```
## [1] 2.851861e-07
```

The slope coefficient represents the average monthly increase in renewable energy production and is positive and statistically significant at 2.85×10^{-7} . The high statistical significance of the slope confirms the strong upward trend observed in the time series plot. The intercept (145.22) corresponds to the estimated production level at the beginning of the sample period (when the time index equals 1). The model fits the data very well, with an R^2 of approximately 0.82, meaning that over 80% of the variation in renewable energy production is explained by the linear time trend alone.

```
lm_hydro <- lm(Hydroelectric ~ Month, data = df)
summary(lm_hydro)
```

```
##
## Call:
## lm(formula = Hydroelectric ~ Month, data = df)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -30.190 -10.214  -0.714   8.907  39.723
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  8.365e+01  1.204e+00   69.50 < 2e-16 ***
## Month       -4.640e-09  1.154e-09   -4.02 6.52e-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:  0.02497,    Adjusted R-squared:  0.02343
## F-statistic: 16.16 on 1 and 631 DF,  p-value: 6.515e-05

beta0_hydro<- as.numeric(lm_hydro$coefficients[1]) #intercept
beta1_hydro <- as.numeric(lm_hydro$coefficients[2]) #slope

beta0_hydro
```

```
## [1] 83.65166
```

```
beta1_hydro
```

```
## [1] -4.639963e-09
```

The estimated regression shows a statistically significant but small negative trend over time. The slope

on Month is -4.64×10^{-9} and is statistically significant at the 1% level ($p = 6.5 \times 10^{-5}$), indicating a very slight decline in hydroelectric consumption over the sample period. The intercept (83.65) represents the estimated average hydroelectric power consumption at the beginning of the sample period. Despite the statistical significance of the slope, the model explains very little of the overall variation in the data ($R^2 = 0.023$).

Q4

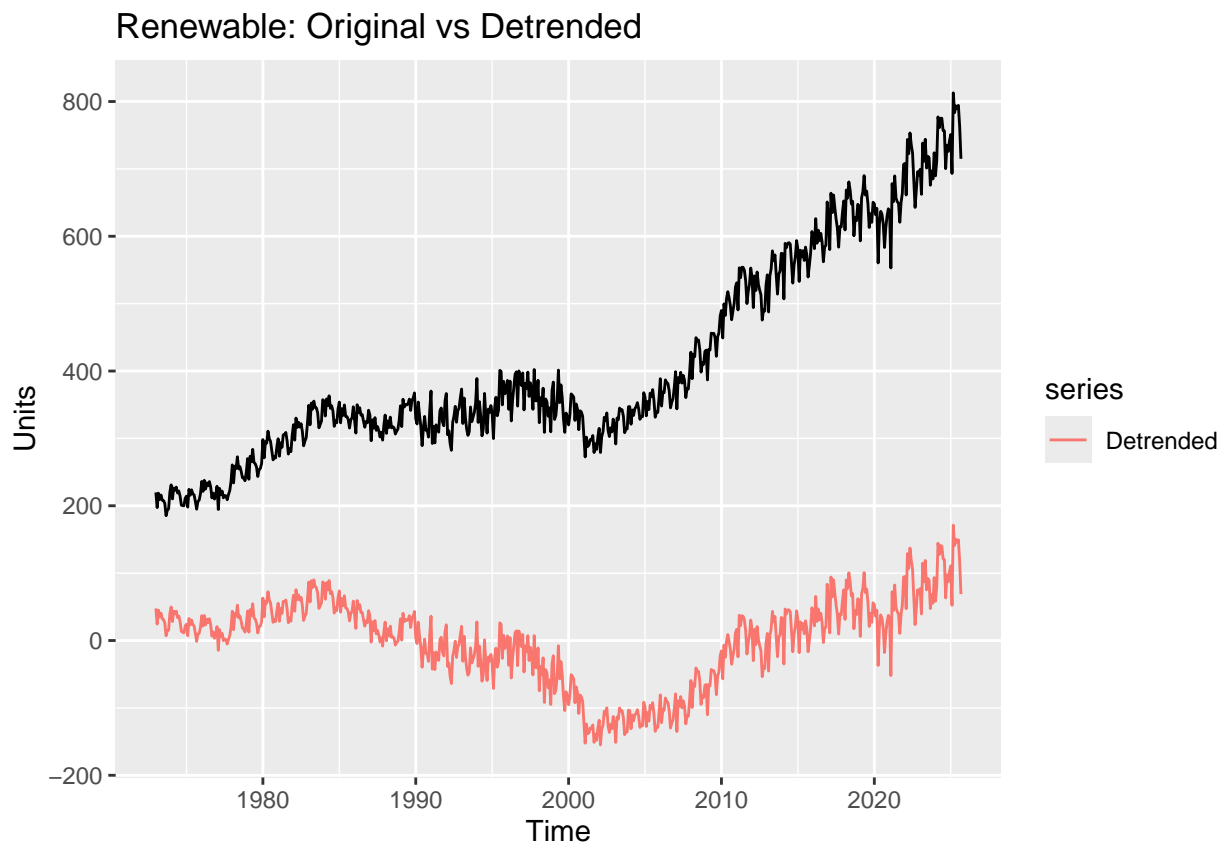
Use the regression coefficients to detrend each series (subtract fitted linear trend). Plot detrended series and compare with the original time series from Q1. Describe what changed.

```
nobs <- nrow(df)
t <- 1:nobs

lm_renew <- lm(Renewable ~ t, data = df)
beta0_renew <- coef(lm_renew)[1]
beta1_renew <- coef(lm_renew)[2]

renew_detrend <- df$Renewable - (beta0_renew + beta1_renew * t)
renew_det_ts <- ts(renew_detrend, frequency = 12, start = c(start_year, start_month))

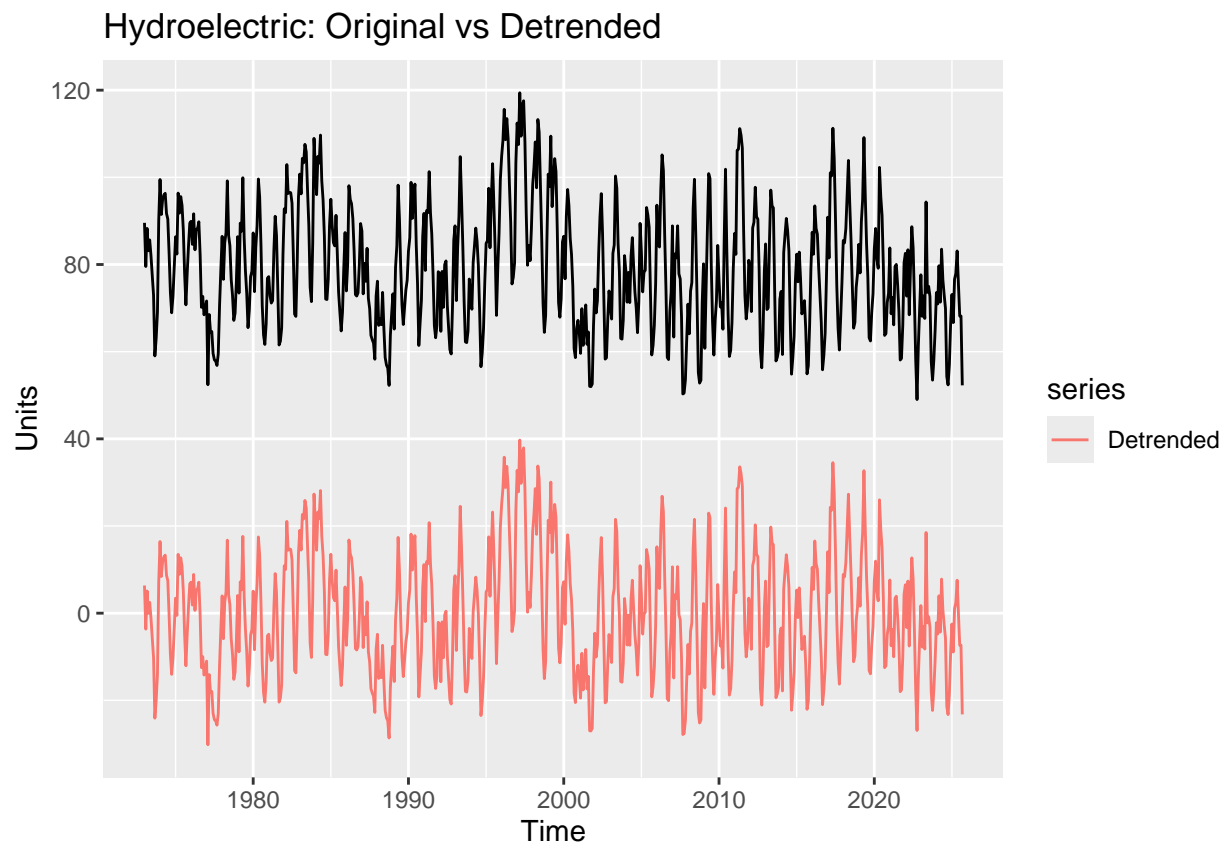
autoplot(renew_ts) +
  autolayer(renew_det_ts, series = "Detrended") +
  labs(title = "Renewable: Original vs Detrended",
       x = "Time", y = "Units")
```



```
lm_hydro <- lm(Hydroelectric ~ t, data = df)
beta0_hydro <- coef(lm_hydro)[1]
beta1_hydro <- coef(lm_hydro)[2]

hydro_detrend <- df$Hydroelectric - (beta0_hydro + beta1_hydro * t)
hydro_det_ts <- ts(hydro_detrend, frequency = 12, start = c(start_year, start_month))

autoplot(hydro_ts) +
  autolayer(hydro_det_ts, series = "Detrended") +
  labs(title = "Hydroelectric: Original vs Detrended",
        x = "Time", y = "Units")
```



The original renewable energy series exhibits a strong upward trend over time, with production levels increasing substantially across the sample period. After detrending, this long-run growth is removed, and the series now fluctuates around a zero mean. The detrended series highlights medium- and short-term deviations from the linear trend, including periods of below-trend production in the late 1990s and early 2000s and above-trend production in more recent years.

For hydroelectric power consumption, the original series already fluctuates around a relatively stable level with no clear long-term trend. After detrending, the series appears almost unchanged aside from a recentering around zero.

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?

```

acf_renew_orig <- autoplot(Acf(renew_ts, lag.max = 40, plot = FALSE)) +
  labs(title = "Renewable (Original) - ACF", x = "Lag", y = "ACF")

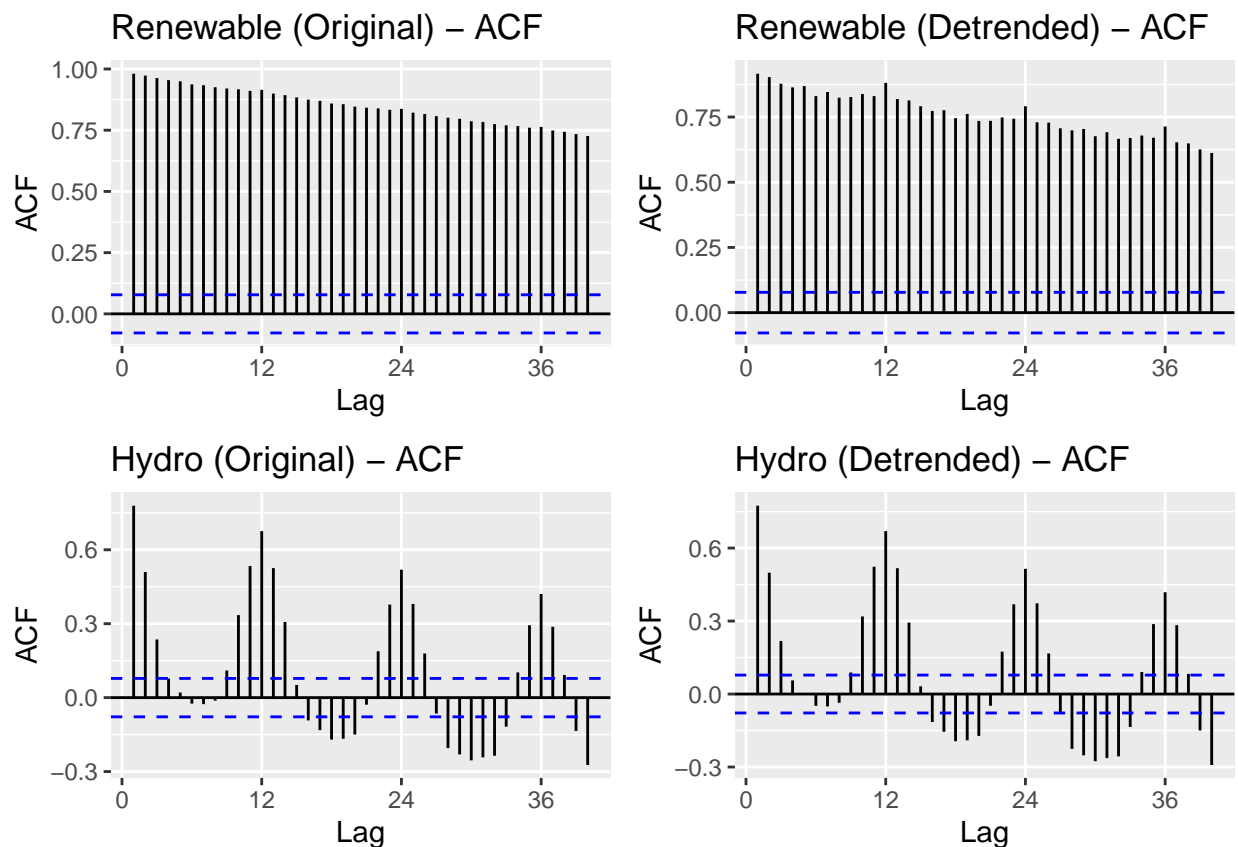
acf_renew_det <- autoplot(Acf(renew_det_ts, lag.max = 40, plot = FALSE)) +
  labs(title = "Renewable (Detrended) - ACF", x = "Lag", y = "ACF")

acf_hydro_orig <- autoplot(Acf(hydro_ts, lag.max = 40, plot = FALSE)) +
  labs(title = "Hydro (Original) - ACF", x = "Lag", y = "ACF")

acf_hydro_det <- autoplot(Acf(hydro_det_ts, lag.max = 40, plot = FALSE)) +
  labs(title = "Hydro (Detrended) - ACF", x = "Lag", y = "ACF")

plot_grid(acf_renew_orig, acf_renew_det,
          acf_hydro_orig, acf_hydro_det,
          nrow = 2, align = "v")

```



Compared to Q1, the detrended renewable series shows some change but not a complete transformation. The ACF of the detrended series still remains very high and decays slowly across many lags, indicating that strong persistence remains even after detrending. The PACF continues to show a very strong spike at lag 1, with several additional significant spikes at short and seasonal lags.

In contrast, the ACF and PACF of the detrended hydroelectric series are almost unchanged relative to Q1. The ACF continues to display strong seasonal autocorrelation at lags 12, 24, and their multiples, while the PACF shows a dominant spike at lag 1 and seasonal effects around lag 12.

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 your answer below.

Answer: Based on visual inspection of the time series plots and the ACFs from Q1, hydroelectric power consumption clearly exhibits a seasonal pattern, while total renewable energy production does not show a strong seasonal trend. The hydroelectric series displays regular, repeating fluctuations over time, and its ACF shows prominent spikes at lags of 12 and its multiples. In contrast, the renewable energy production series is dominated by a long-term upward trend, and although some short-term seasonality fluctuations are present, it is not strong.

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 your answer to Q6?

```
# Seasonal dummies (11 columns; one month is baseline via intercept)
# renew_ts <- ts(df$Renewable, frequency=12, start=c(start_year, start_month))
# hydro_ts <- ts(df$Hydroelectric, frequency=12, start=c(start_year, start_month))
```

```
dummies_renew <- seasonaldummy(renew_ts)
dummies_hydro <- seasonaldummy(hydro_ts)
```

```
# Seasonal means models
```

```
seas_model_renew <- lm(as.numeric(renew_ts) ~ dummies_renew)
seas_model_hydro <- lm(as.numeric(hydro_ts) ~ dummies_hydro)
```

```
# Print summaries
```

```
summary(seas_model_renew)
```

```
##
```

```
## Call:
```

```
## lm(formula = as.numeric(renew_ts) ~ dummies_renew)
```

```
##
```

```
## 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 ***
## dummies_renewJan     2.090     29.693    0.070    0.944
## dummies_renewFeb   -34.524     29.693   -1.163    0.245
## dummies_renewMar     5.956     29.693    0.201    0.841
## dummies_renewApr    -6.900     29.693   -0.232    0.816
## dummies_renewMay     8.162     29.693    0.275    0.784
## dummies_renewJun    -2.231     29.693   -0.075    0.940
## dummies_renewJul     3.864     29.693    0.130    0.897
## dummies_renewAug    -3.978     29.693   -0.134    0.893
```

```
## dummies_renewSep -29.033      29.693 -0.978      0.329
## dummies_renewOct -19.937      29.834 -0.668      0.504
## dummies_renewNov -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(seas_model_hydro)
```

```
##
## Call:
## lm(formula = as.numeric(hydro_ts) ~ dummies_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 ***
## dummies_hydroJan    4.951      2.021   2.449  0.014591 *
## dummies_hydroFeb   -2.415      2.021  -1.195  0.232608
## dummies_hydroMar    7.116      2.021   3.520  0.000463 ***
## dummies_hydroApr    5.614      2.021   2.777  0.005649 **
## dummies_hydroMay   14.080      2.021   6.965  8.38e-12 ***
## dummies_hydroJun   10.780      2.021   5.333  1.36e-07 ***
## dummies_hydroJul    4.003      2.021   1.980  0.048091 *
## dummies_hydroAug   -5.320      2.021  -2.632  0.008710 **
## dummies_hydroSep  -16.598      2.021  -8.211  1.28e-15 ***
## dummies_hydroOct  -16.329      2.031  -8.040  4.56e-15 ***
## dummies_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
```

The seasonal means regression for total renewable energy production shows that none of the monthly dummy variables are statistically significant at conventional levels. The overall model fit is extremely weak, with an R^2 of approximately -0.009324. This indicates that average renewable energy production does not differ systematically across months. In other words, there is no evidence of a meaningful seasonal pattern in renewable energy production once month-to-month variation is accounted for.

In contrast, the seasonal means regression for hydroelectric power consumption reveals strong and statistically significant seasonal effects. Many monthly dummy coefficients are significant at the 1% or 0.1% level, with sizable magnitudes. For example, hydroelectric consumption is significantly higher in late spring and early summer (e.g., May and June) and significantly lower in late summer and fall (e.g., August through November) relative to the reference month. The model explains a substantial portion of the variation in the data, with an R^2 of approximately 0.46.

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?

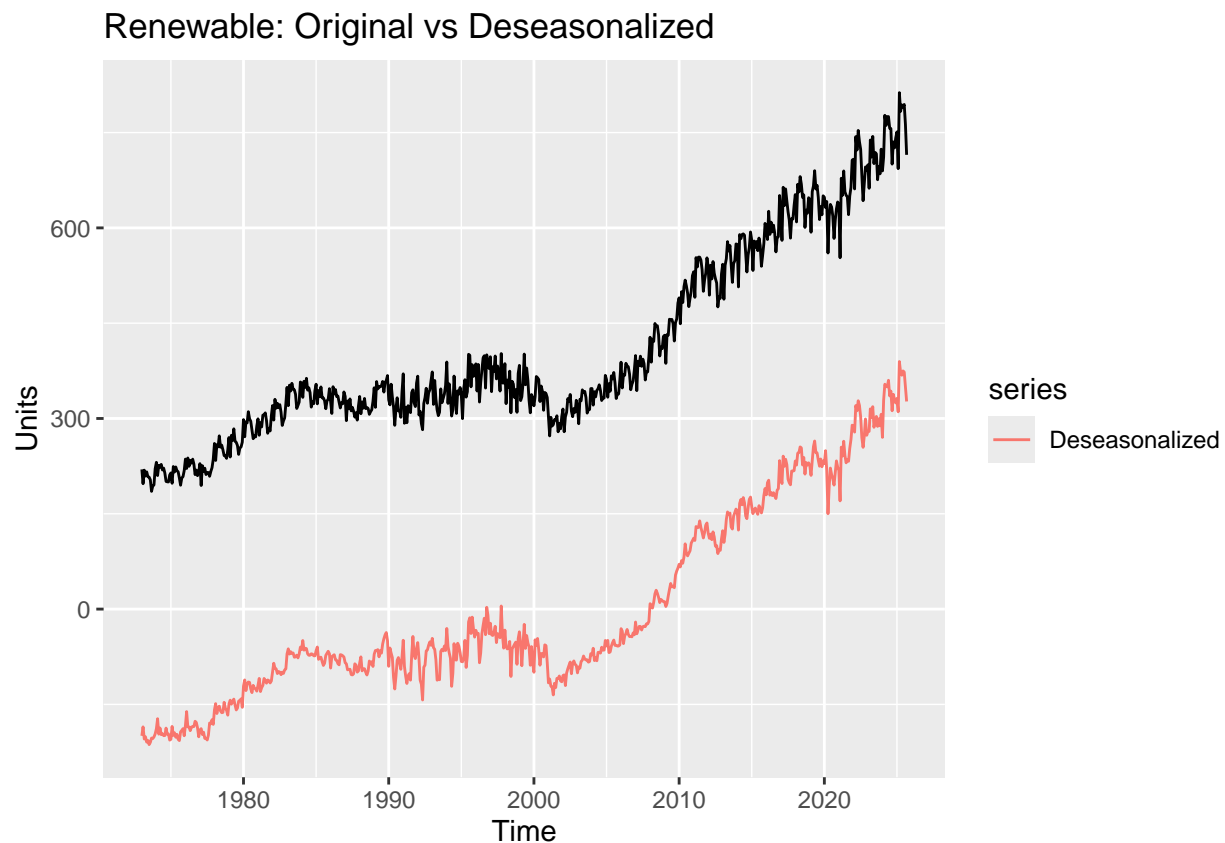
```
nobs_renew <- length(renew_ts)

beta_int_renew <- seas_model_renew$coefficients[1]
beta_coeff_renew <- seas_model_renew$coefficients[2:12]

season_renew <- array(0, nobs_renew)
for (i in 1:nobs_renew) {
  season_renew[i] <- beta_int_renew + sum(beta_coeff_renew * dummies_renew[i, ])
}

deseason_renew <- as.numeric(renew_ts) - season_renew
renew_deseason_ts <- ts(deseason_renew,
  start = c(start_year, start_month),
  frequency = 12)

autoplot(renew_ts) +
  autolayer(renew_deseason_ts, series = "Deseasonalized") +
  labs(title = "Renewable: Original vs Deseasonalized",
    x = "Time", y = "Units")
```



```
nobs_hydro <- length(hydro_ts)
```

```

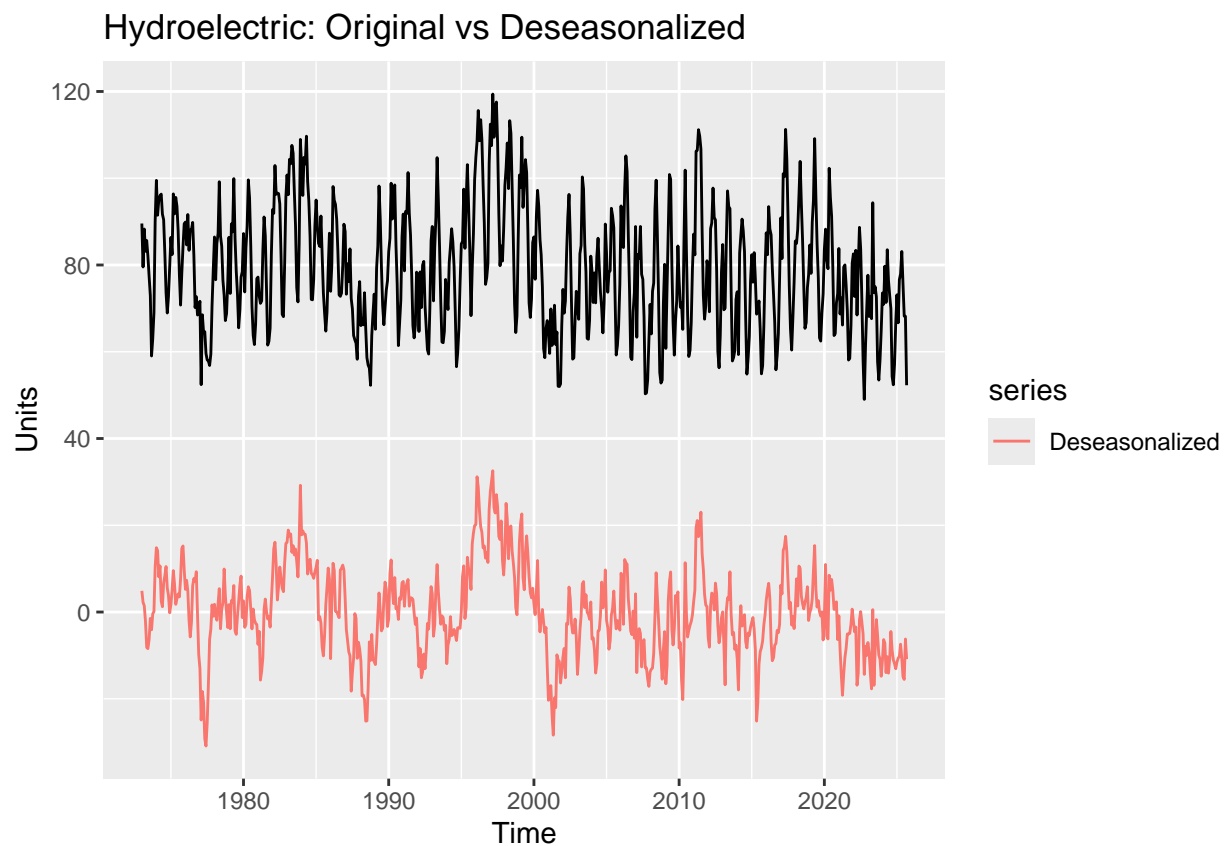
beta_int_hydro <- seas_model_hydro$coefficients[1]
beta_coeff_hydro <- seas_model_hydro$coefficients[2:12]

season_hydro <- array(0, nobs_hydro)
for (i in 1:nobs_hydro) {
  season_hydro[i] <- beta_int_hydro + sum(beta_coeff_hydro * dummies_hydro[i, ])
}

deseason_hydro <- as.numeric(hydro_ts) - season_hydro
hydro_deseason_ts <- ts(deseason_hydro,
  start = c(start_year, start_month),
  frequency = 12)

autoplot(hydro_ts) +
  autolayer(hydro_deseason_ts, series = "Deseasonalized") +
  labs(title = "Hydroelectric: Original vs Deseasonalized",
    x = "Time", y = "Units")

```



Deseasonalizing the renewable energy series produces very little change compared to the original series shown in Q1. The strong upward trend remains clearly visible, and the overall shape and variability of the series are essentially unchanged aside from a vertical shift. This confirms that renewable energy production does not have a meaningful seasonal component, and that its dynamics are dominated by long-term growth rather than recurring monthly effects.

In contrast, deseasonalizing the hydroelectric series leads to a substantial change in the time series. The regular, repeating monthly oscillations observed in the original series are largely removed, resulting in a

much smoother series centered around zero. While short-term fluctuations remain, the pronounced seasonal pattern evident in Q1 is no longer present. This indicates that seasonality is a dominant component of hydroelectric power consumption and that removing it significantly alters the structure of the series.

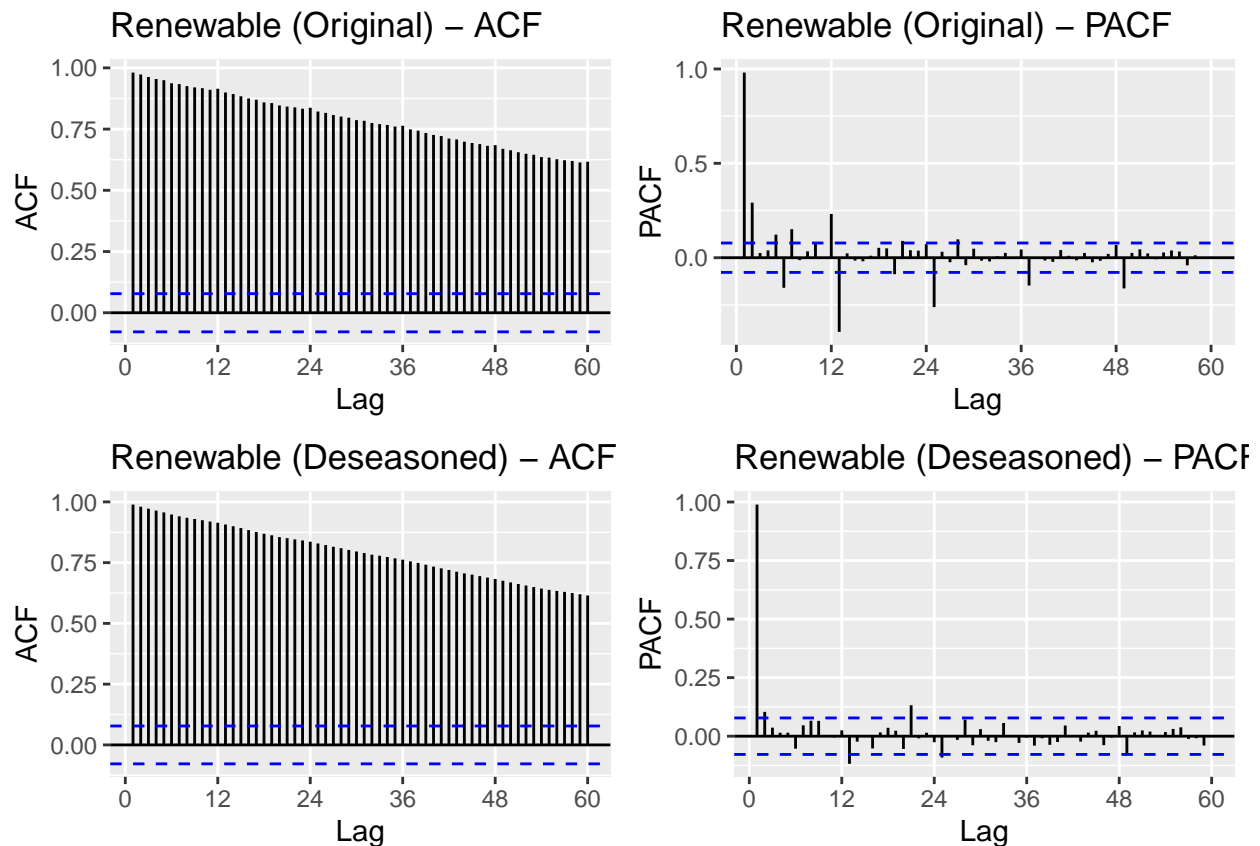
Q9

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?

```
# --- Renewable: Original vs Deseasoned ACF/PACF ---
acf_renew_orig <- autoplot(Acf(renew_ts, lag.max = 60, plot = FALSE)) +
  labs(title = "Renewable (Original) - ACF", x = "Lag", y = "ACF")
pacf_renew_orig <- autoplot(Pacf(renew_ts, lag.max = 60, plot = FALSE)) +
  labs(title = "Renewable (Original) - PACF", x = "Lag", y = "PACF")

acf_renew_des <- autoplot(Acf(renew_deseason_ts, lag.max = 60, plot = FALSE)) +
  labs(title = "Renewable (Deseasoned) - ACF", x = "Lag", y = "ACF")
pacf_renew_des <- autoplot(Pacf(renew_deseason_ts, lag.max = 60, plot = FALSE)) +
  labs(title = "Renewable (Deseasoned) - PACF", x = "Lag", y = "PACF")

plot_grid(acf_renew_orig, pacf_renew_orig,
          acf_renew_des, pacf_renew_des,
          nrow = 2)
```



```
# --- Hydroelectric: Original vs Deseasoned ACF/PACF ---
acf_hydro_orig <- autoplot(Acf(hydro_ts, lag.max = 60, plot = FALSE)) +
  labs(title = "Hydro (Original) - ACF", x = "Lag", y = "ACF")
```

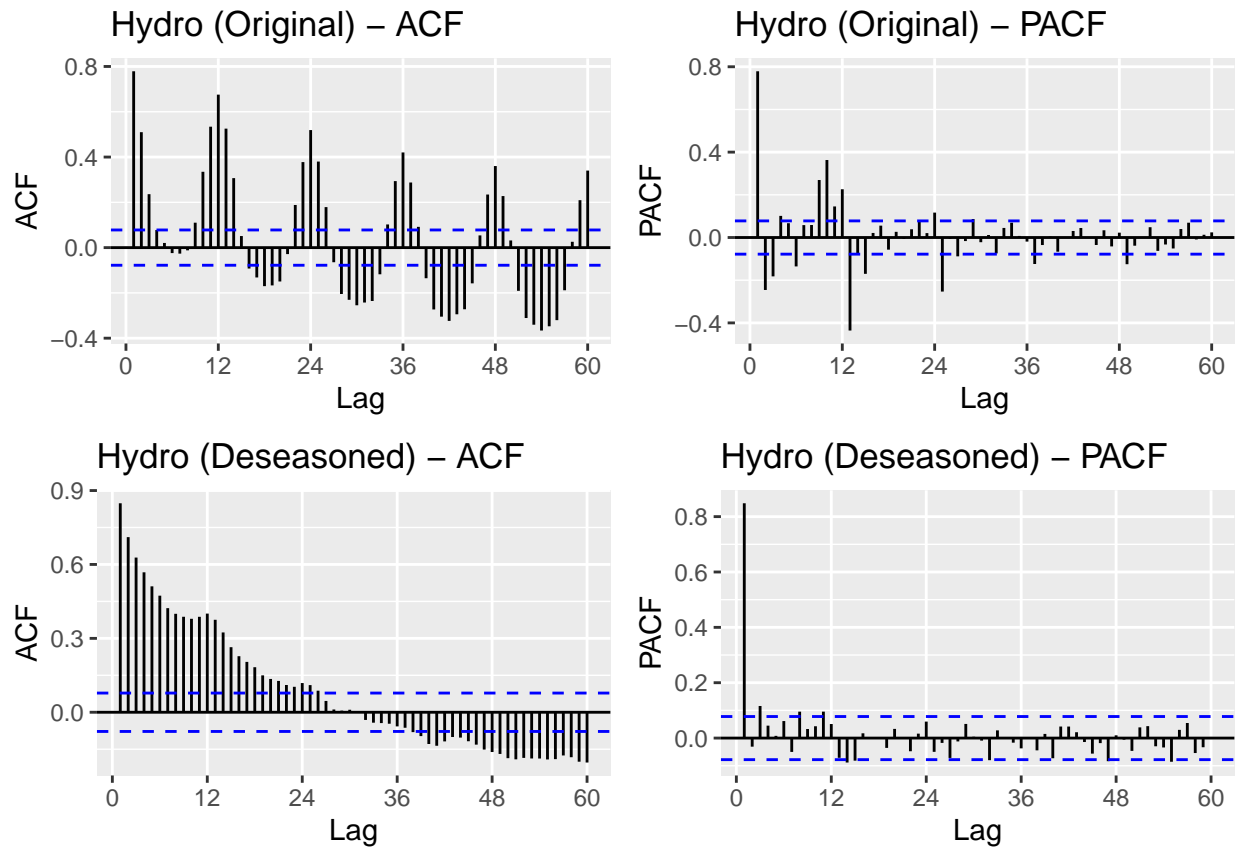
```

pacf_hydro_orig <- autoplot(Pacf(hydro_ts, lag.max = 60, plot = FALSE)) +
  labs(title = "Hydro (Original) - PACF", x = "Lag", y = "PACF")

acf_hydro_des <- autoplot(Acf(hydro_deseason_ts, lag.max = 60, plot = FALSE)) +
  labs(title = "Hydro (Deseasoned) - ACF", x = "Lag", y = "ACF")
pacf_hydro_des <- autoplot(Pacf(hydro_deseason_ts, lag.max = 60, plot = FALSE)) +
  labs(title = "Hydro (Deseasoned) - PACF", x = "Lag", y = "PACF")

plot_grid(acf_hydro_orig, pacf_hydro_orig,
          acf_hydro_des, pacf_hydro_des,
          nrow = 2)

```



Comparing the deseasonalized ACF and PACF with those from Q1, there is essentially no change in the correlation structure. The ACF remains very high at short lags and decays slowly across many lags, indicating strong persistence and non-stationarity. The PACF continues to show a dominant spike at lag 1, but the higher lags are comparatively more insignificant.

In contrast, the deseasonalized hydroelectric series shows a substantial change relative to Q1. In the original series, the ACF exhibited strong seasonal spikes at lags 12, 24, and their multiples. After deseasonalization, these seasonal spikes are largely removed. Similarly, the PACF no longer shows pronounced seasonal effects and is dominated by a strong spike at lag 1, followed by mostly insignificant higher-order lags.