ENV 797 - Time Series Analysis for Energy and Environment Applications | Spring 2026

Assignment 4 - Due date 02/10/26

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. And to do so you will need to fork our repository and link it to your RStudio.

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_A04_Sp26.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).

When you have completed the assignment, **Knit** the text and code into a single PDF file. Submit this pdf using Sakai.

R packages needed for this assignment: "xlsx" or "readxl", "ggplot2", "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(readxl)
library(forecast)
library(dplyr)
library(ggplot2)
library(cowplot)
library(tseries)
library(Kendall)
```

Questions

Consider the same data you used for A3 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. **For this assignment you will work only with the column "Total Renewable Energy Production"**.

```
energy_data <- read_excel(
  path="../Data/Table_10.1_Renewable_Energy_Production_and_Consumption_by_Source.xlsx",
  skip = 12,
  sheet="Monthly Data",
  col_names=FALSE)

read_col_names <- read_excel(
  path="../Data/Table_10.1_Renewable_Energy_Production_and_Consumption_by_Source.xlsx",
  skip = 10,
  n_max = 1,
  sheet="Monthly Data",
  col_names=FALSE)

colnames(energy_data) <- read_col_names

data <- energy_data[,c(1,5)]
nobs <- nrow(data)
t <- 1:nobs

ts_total <- ts(data[,2], frequency=12, start=c(1973,1))

head(ts_total)
```

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

Stochastic Trend and Stationarity Tests

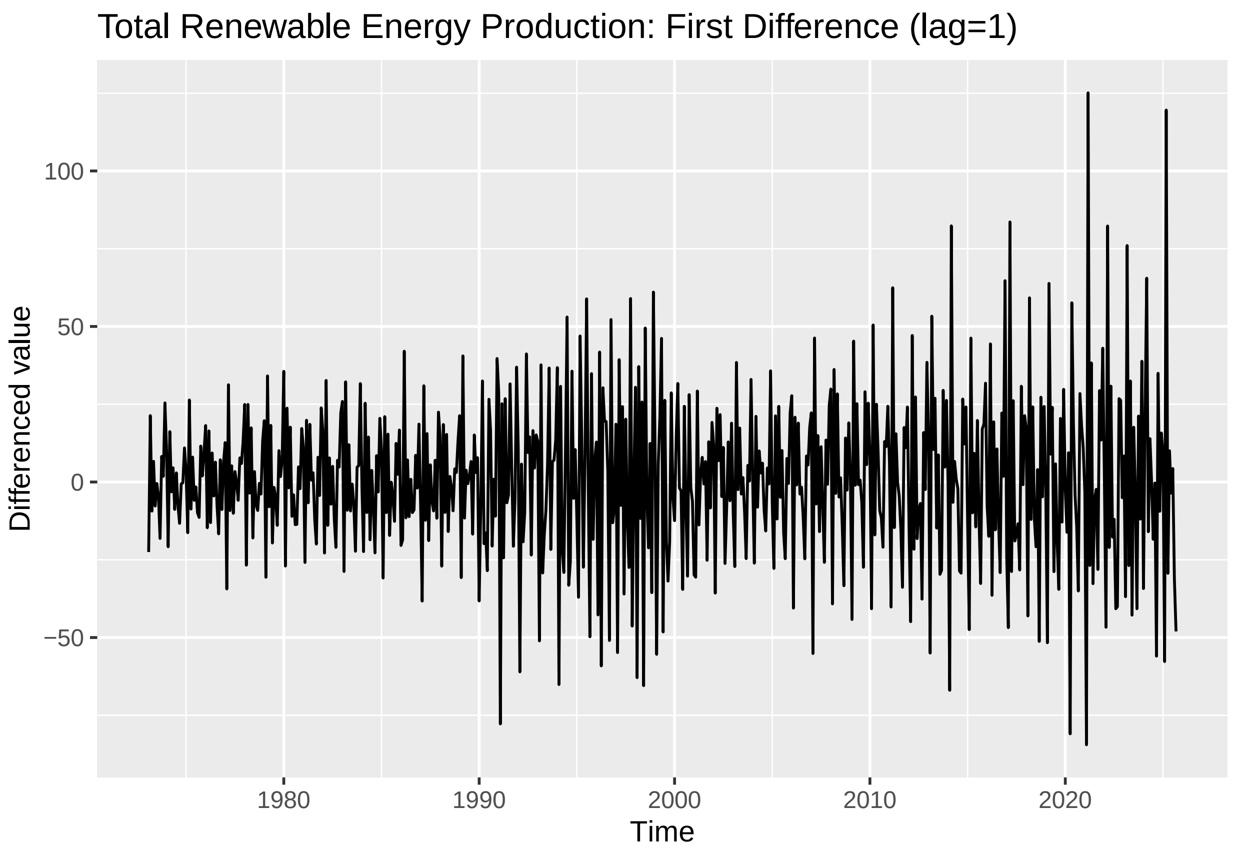
For this part you will work only with the column Total Renewable Energy Production.

Q1

Difference the "Total Renewable Energy Production" series using function diff(). Function diff() is from package base and take three main arguments: * *x* vector containing values to be differenced; * *lag* integer indicating with lag to use; * *differences* integer indicating how many times series should be differenced.

Try differencing at lag 1 only once, i.e., make `lag=1` and `differences=1`. Plot the differenced series. Do the series still seem to have trend?

```
total_diff <- diff(ts_total, lag = 1, differences = 1)

# plot
autoplot(total_diff) +
  labs(title = "Total Renewable Energy Production: First Difference (lag=1)",
       x = "Time", y = "Differenced value")
```



After taking the first difference (lag = 1), the series fluctuates around zero and the long-run upward trend observed in the original data is largely removed. However, the differenced series shows larger swings in later years, suggesting increased variability (and possible structural changes) even after differencing.

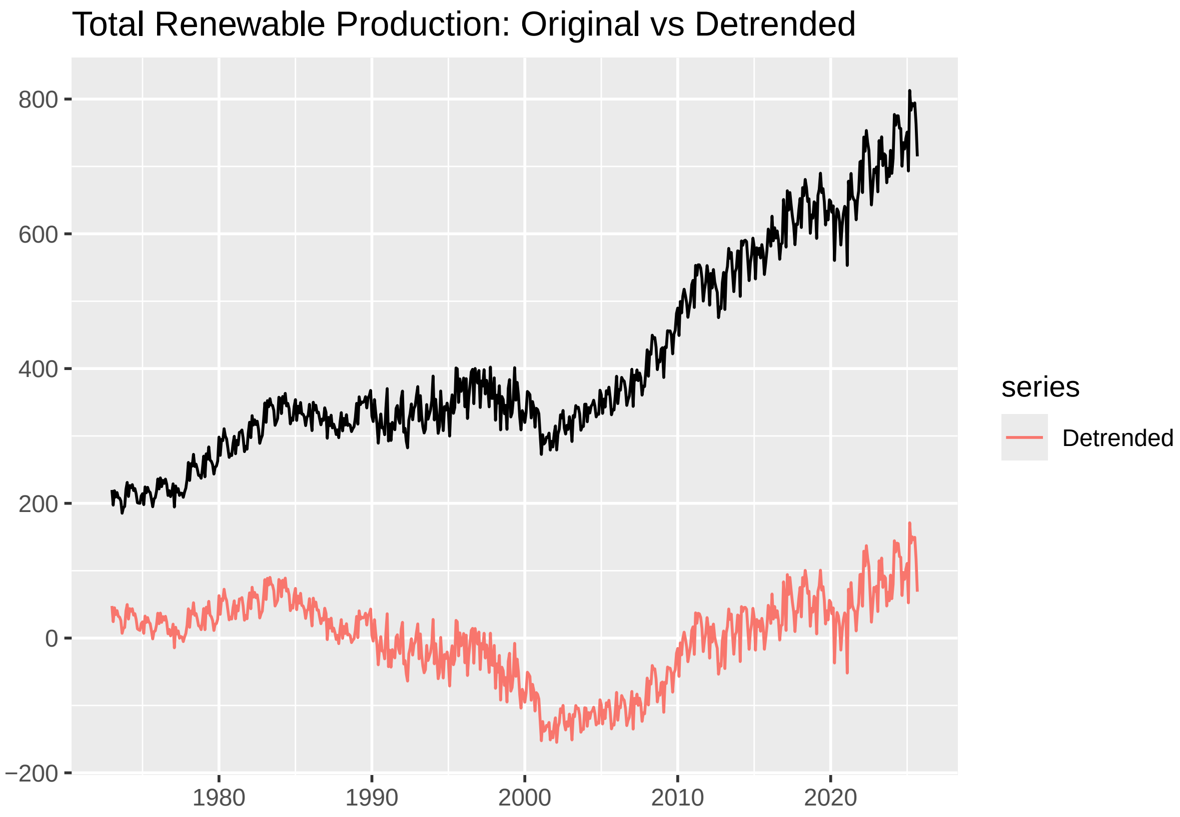
Q2

Copy and paste part of your code for A3 where you run the regression for Total Renewable Energy Production and subtract that from the original series. This should be the code for Q3 and Q4. make sure you use assign same name for the time series object that you had in A3, otherwise the code will not work.

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

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
```

```
# Save coefficients
beta0_total <- coef(lm_total)[1]   # intercept
beta1_total <- coef(lm_total)[2]   # slope

# ---------- Detrend Total Renewable Production ----------
trend_total <- beta0_total + beta1_total * t_total

ts_total_detr <- ts(y_total - trend_total,
                    start = start(ts_total),
                    frequency = frequency(ts_total))

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



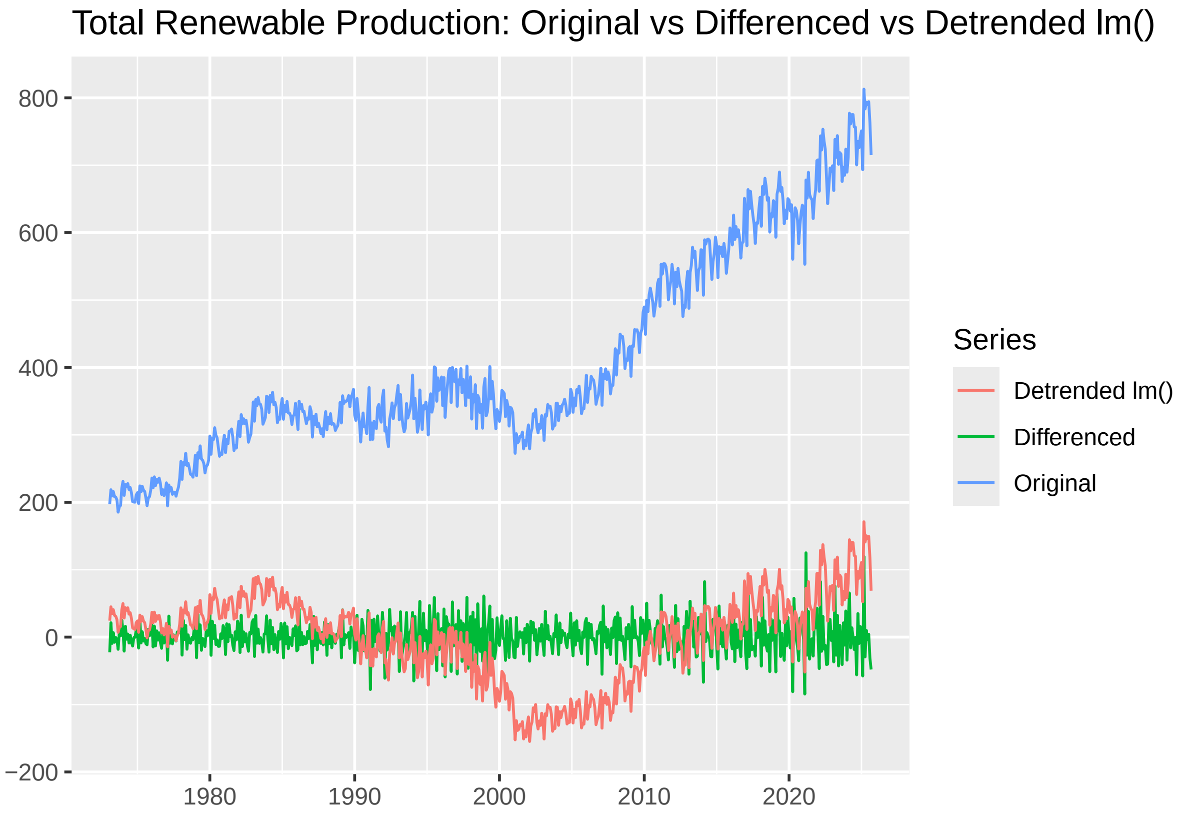
Q3

Now let's compare the differenced series with the detrended series you calculated on A3. In other words, for the "Total Renewable Energy Production" compare the differenced series from Q1 with the series you detrended in Q2 using linear regression.

Using autoplot() + autolayer() create a plot that shows the three series together (i.e. "Original", "Differenced", "Detrended lm()"). Make sure your plot has a legend. The easiest way to do it is by adding the `series=` argument to each autoplot and autolayer function. Look at the key for A03 for an example on how to use autoplot() and autolayer().

What can you tell from this plot? Which method seems to have been more efficient in removing the trend?

```r
# Q3: align lengths (diff drops the first observation)
ts_total_align      <- window(ts_total, start = time(ts_total)[2])
ts_total_detr_align <- window(ts_total_detr, start = time(ts_total_detr)[2])

autoplot(ts_total_align, series = "Original") +
  autolayer(total_diff, series = "Differenced") +
  autolayer(ts_total_detr_align, series = "Detrended lm()") +
  ggtitle("Total Renewable Production: Original vs Differenced vs Detrended lm()") +
  xlab("") + ylab("") +
  guides(colour = guide_legend(title = "Series"))
```



Answer:

The original series shows a strong upward trend.

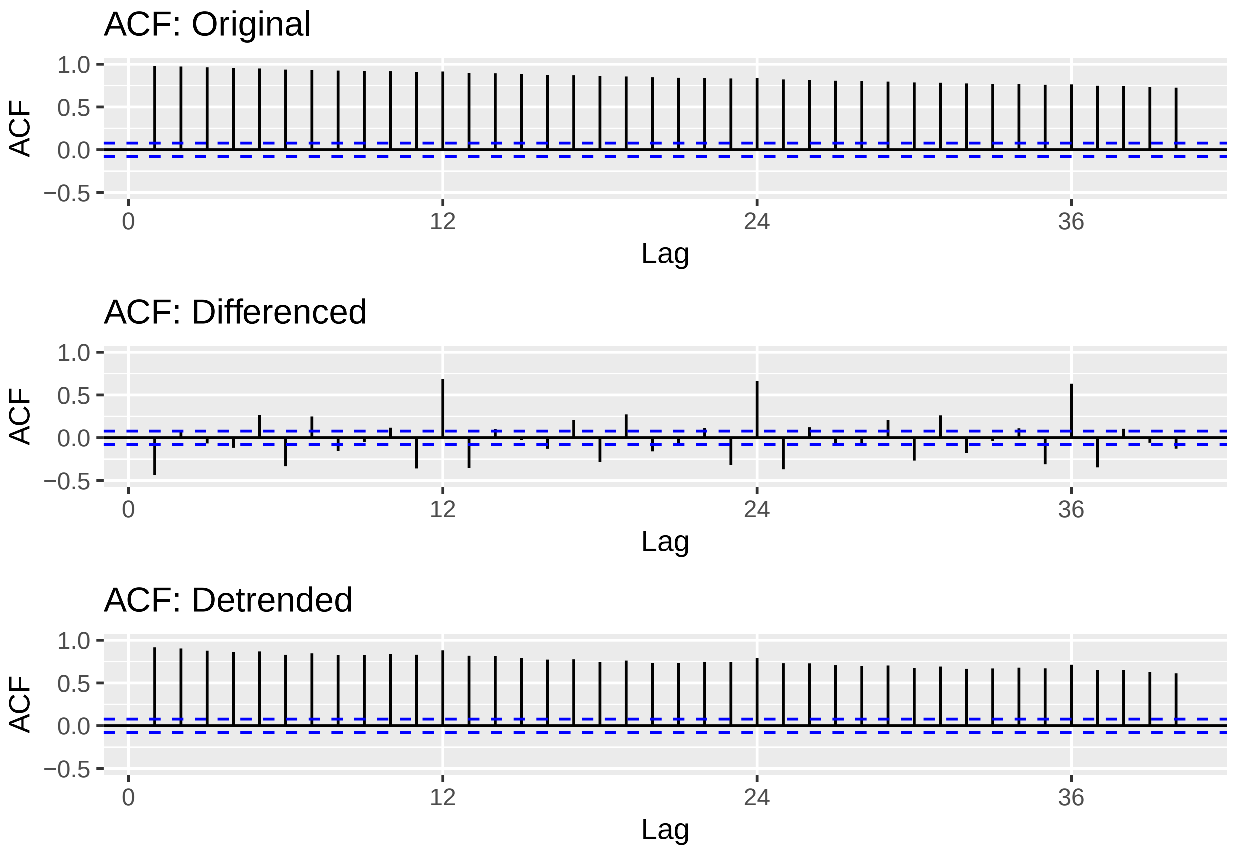
The detrended series removes the fitted linear component, but it still displays large long-run movements (a clear dip around the early 2000s and a rise afterward), suggesting that the trend is not purely linear.

In contrast, the differenced series fluctuates around zero with much less persistent drift. Therefore, differencing appears more efficient than linear detrending in removing the trend.

Q4

Plot the ACF for the three series and compare the plots. Add the argument `ylim=c(-0.5,1)` to the autoplot() or Acf() function - whichever you are using to generate the plots - to make sure all three y axis have the same limits. Looking at the ACF which method do you think was more efficient in eliminating the trend? The linear regression or differencing?

```
p1 <- autoplot(Acf(ts_total, lag.max=40, plot=FALSE)) +
  labs(title="ACF: Original") +
  coord_cartesian(ylim=c(-0.5,1))

p2 <- autoplot(Acf(total_diff, lag.max=40, plot=FALSE)) +
  labs(title="ACF: Differenced") +
  coord_cartesian(ylim=c(-0.5,1))

p3 <- autoplot(Acf(ts_total_detr, lag.max=40, plot=FALSE)) +
  labs(title="ACF: Detrended") +
  coord_cartesian(ylim=c(-0.5,1))

plot_grid(p1, p2, p3, nrow=3)
```



Answer:

The ACF of the original series stays very high and decays slowly across many lags, which is characteristic of a nonstationary series with trend.

After linear detrending, the ACF still shows a slow decay and remains strongly positive, indicating that removing only a linear trend is not sufficient.

In contrast, the differenced series has much smaller autocorrelations, and the long slow decay largely disappears. The remaining significant spikes at lags 12, 24, and 36 suggest seasonal dependence.

Overall, differencing appears more efficient than linear detrending in removing the trend.

Q5

Compute the Seasonal Mann-Kendall and ADF Test for the original "Total Renewable Energy Production" series. Ask R to print the results. Interpret the results for both test. What is the conclusion from the Seasonal Mann Kendall test? What's the conclusion for the ADF test? Do they match what you observed in Q3 plot? Recall that having a unit root means the series has a stochastic trend. And when a series has stochastic trend we need to use differencing to remove the trend.

```r
# Seasonal Mann-Kendall
smk_total <- SeasonalMannKendall(ts_total)
smk_total
```

```
## tau = 0.799, 2-sided pvalue =< 2.22e-16
```

```r
# Augmented Dickey-Fuller test
adf_total <- adf.test(as.numeric(ts_total))
adf_total
```

```
##
##  Augmented Dickey-Fuller Test
##
## data:  as.numeric(ts_total)
## Dickey-Fuller = -1.0247, Lag order = 8, p-value = 0.9347
## alternative hypothesis: stationary
```

Answer:

The Seasonal Mann–Kendall test indicates a highly significant upward trend in the monthly series even after accounting for seasonality (tau = 0.799, p-value < 2.22e-16).

In contrast, the ADF test fails to reject the null of a unit root (p-value = 0.9347), suggesting the series is nonstationary and contains a stochastic trend.

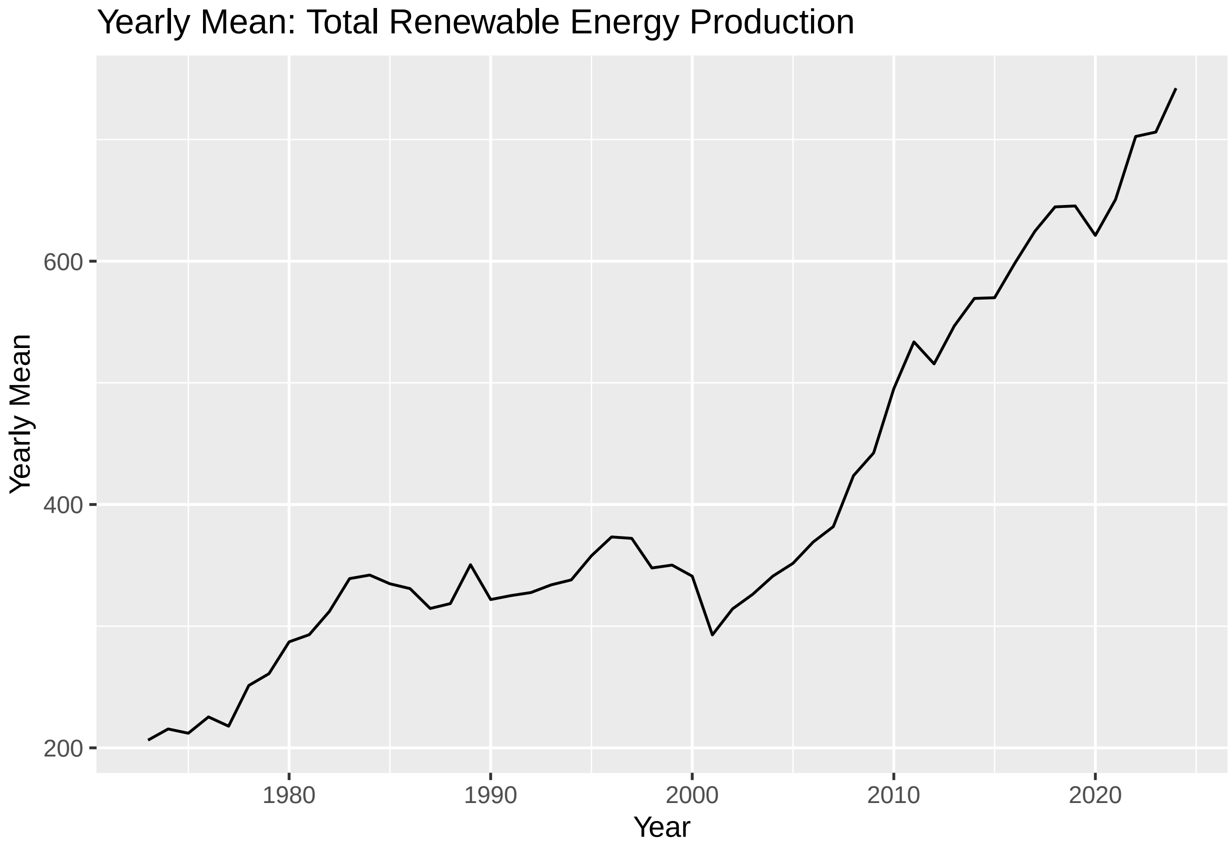
This is consistent with Q3 and Q4: linear detrending removes only the fitted linear component, while differencing more effectively eliminates the persistent trend behavior (the ACF no longer shows a slow decay).

Q6

Aggregate the original "Total Renewable Energy Production" series by year. You can use the same procedure we used in class. Store series in a matrix where rows represent months and columns represent years. And then take the columns mean using function colMeans(). Recall the goal is the remove the seasonal variation from the series to check for trend. Convert the accumulates yearly series into a time series object and plot the series using autoplot().

```r
y <- as.numeric(ts_total)

# number of full years
```

```
nyears <- floor(length(y) / 12)

# cut to full years only (avoid partial last year)
y_cut <- y[1:(nyears * 12)]

# reshape: 12 rows (months) x nyears columns (years)
mat <- matrix(y_cut, nrow = 12, ncol = nyears, byrow = FALSE)

# yearly mean (mean across 12 months for each year)
year_mean <- colMeans(mat)

# convert to yearly ts
start_year <- start(ts_total)[1]
ts_year_mean <- ts(year_mean, start = start_year, frequency = 1)

# plot
autoplot(ts_year_mean) +
  labs(title = "Yearly Mean: Total Renewable Energy Production",
       x = "Year", y = "Yearly Mean")
```



After aggregating the monthly data into yearly means, the seasonal fluctuations are removed and the long-term pattern becomes clearer. The yearly series still shows an overall upward trend, with a noticeable dip around the early 2000s and a strong increase after about 2008–2010.

Q7

Apply the Mann Kendall, Spearman correlation rank test and ADF. Are the results from the test in agreement with the test results for the monthly series, i.e., results for Q5?

```r
# Mann-Kendall trend test (yearly series)
mk_year <- MannKendall(as.numeric(ts_year_mean))
mk_year
```

```
## tau = 0.817, 2-sided pvalue =< 2.22e-16
```

```r
# Spearman rank correlation between series and time
sp_year <- cor.test(as.numeric(ts_year_mean), 1:length(ts_year_mean), method = "spearman")
sp_year
```

```
##
##  Spearman's rank correlation rho
##
## data:  as.numeric(ts_year_mean) and 1:length(ts_year_mean)
## S = 1852, p-value < 2.2e-16
## alternative hypothesis: true rho is not equal to 0
## sample estimates:
##       rho
## 0.9209425
```

```r
# ADF on yearly series
adf_year <- adf.test(as.numeric(ts_year_mean))
adf_year
```

```
##
##  Augmented Dickey-Fuller Test
##
## data:  as.numeric(ts_year_mean)
## Dickey-Fuller = -0.85301, Lag order = 3, p-value = 0.9515
## alternative hypothesis: stationary
```

Answer:

For the yearly mean series, both the Mann–Kendall test and the Spearman rank test indicate a highly significant increasing trend (tau = 0.817, p-value < 2.22e-16; rho = 0.921, p-value < 2.2e-16).

However, the ADF test still fails to reject the null of a unit root (p-value = 0.9515), suggesting that the yearly series remains nonstationary and exhibits a stochastic trend.

Compared to the monthly results in Q5, the trend conclusions are consistent: both monthly and yearly data show a strong upward trend. The ADF results are also consistent, supporting the idea that differencing is more appropriate than linear detrending for removing the trend behavior.