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

Assignment 4 - Due date 02/10/26

Lingyue Hao

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.\

```
library(readxl)
library(ggplot2)
library(forecast)
library(tseries)
library(Kendall)
library(cowplot)
```

Questions

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

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

```r
col_line <- readxl::read_excel(
  path = "../Data/Table_10.1_Renewable_Energy_Production_and_Consumption_by_Source.xlsx",
  sheet = "Monthly Data",
  skip = 10,
  n_max = 1,
  col_names = FALSE
)

colnames(energy_data) <- as.character(col_line[1, ])

trp_df <- energy_data[, "Total Renewable Energy Production", drop = FALSE]
trp_df <- as.data.frame(trp_df)

start_year <- 1973
start_month <- 1

trp_ts <- ts(
  as.numeric(trp_df[["Total Renewable Energy Production"]]),
  start = c(start_year, start_month),
  frequency = 12
)

head(trp_ts)
```

```
##          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.

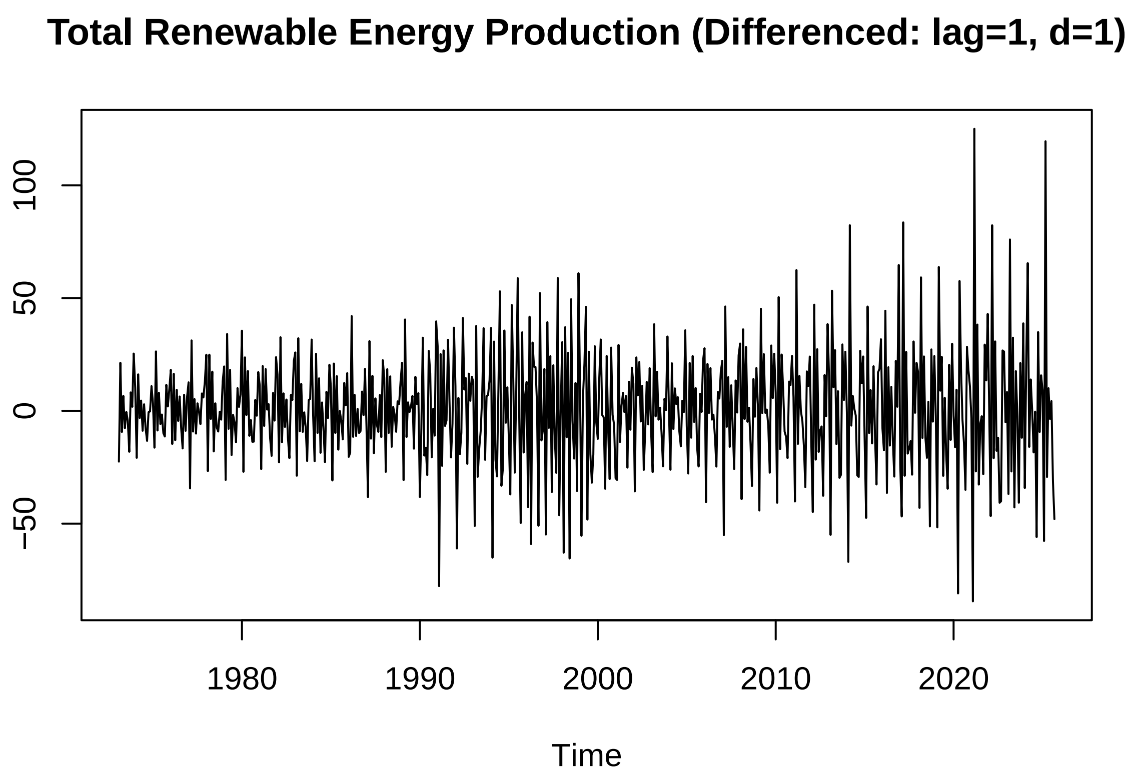
Q1

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?

```r
trp_diff <- diff(trp_ts, lag = 1, differences = 1)

plot(
  trp_diff,
  main = "Total Renewable Energy Production (Differenced: lag=1, d=1)",
  xlab = "Time",
  ylab = ""
)
```



*The series no longer shows a clear trend and the values fluctuate around zero.**

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.

```r
nobs <- length(trp_ts)
t <- 1:nobs

linear_trp <- lm(trp_ts ~ t)
summary(linear_trp)
```

```
##
## Call:
## lm(formula = trp_ts ~ t)
##
## 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             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
trp_trend <- fitted(linear_trp)
trp_detrended <- trp_ts - trp_trend
```

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.

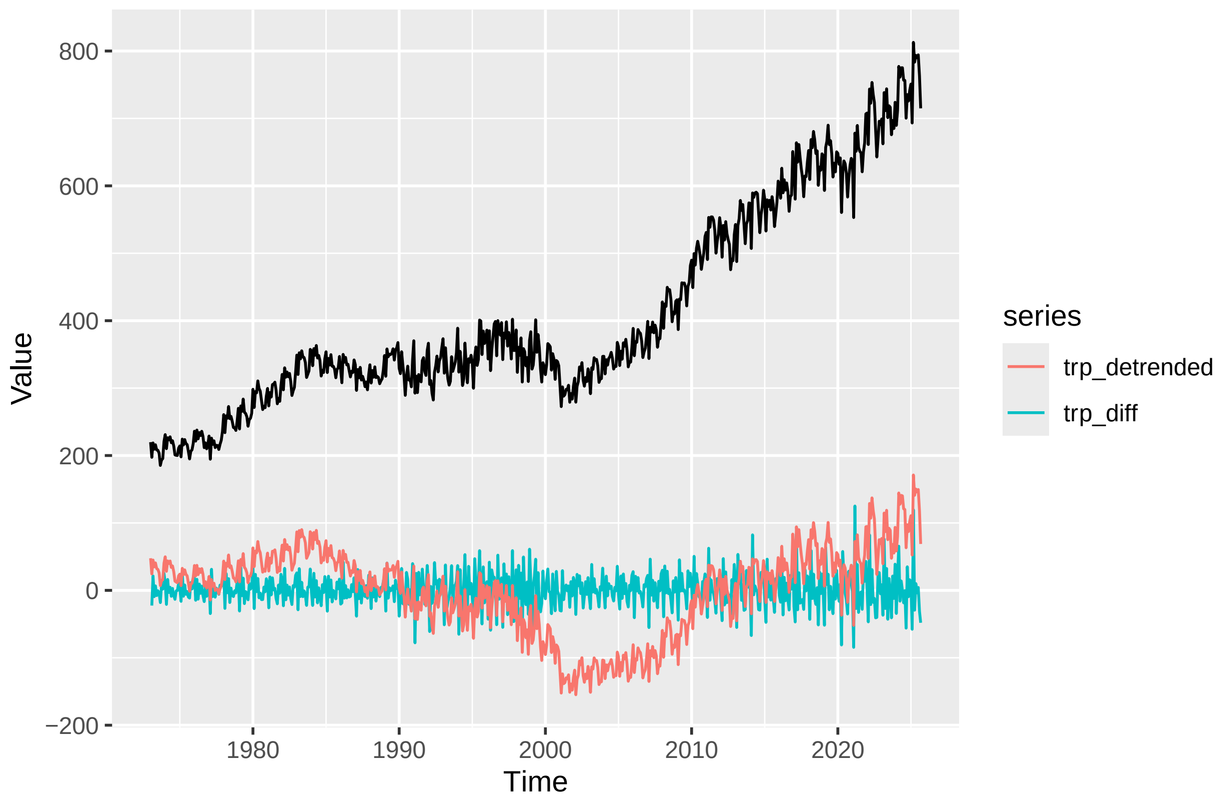
Using autoplot() + autolayer() create a plot that shows the three series together (i.e. "Original", "Differenced", "Detrended lm()"). Make sure your plot has a legend. The easiest way to do it is by adding the `series=` argument to each autoplot and autolayer function. Look at the key for A03 for an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

```
## Warning in ggplot2::geom_line(na.rm = TRUE, ...): Ignoring unknown parameters:
## `Series`

## Warning in ggplot2::geom_line(ggplot2::aes(x = .data[["timeVal"]], y = .data[["seriesVal"]], : Ignori
## Ignoring unknown parameters: `Series`
```



Answer: The original series shows a clear upward trend over time. After detrending using linear regression, the series no longer has a strong upward slope, but it still shows some slow, long-term fluctuations. In contrast, the differenced series fluctuates closely around zero and appears much more stable over time. In this case, differencing is more efficient at 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 <- ggAcf(trp_ts, ylim = c(-0.5, 1)) + ggtitle("ACF: Original")
```

```
## Warning in ggplot2::geom_segment(lineend = "butt", ...): Ignoring unknown
## parameters: ‘ylim‘
```

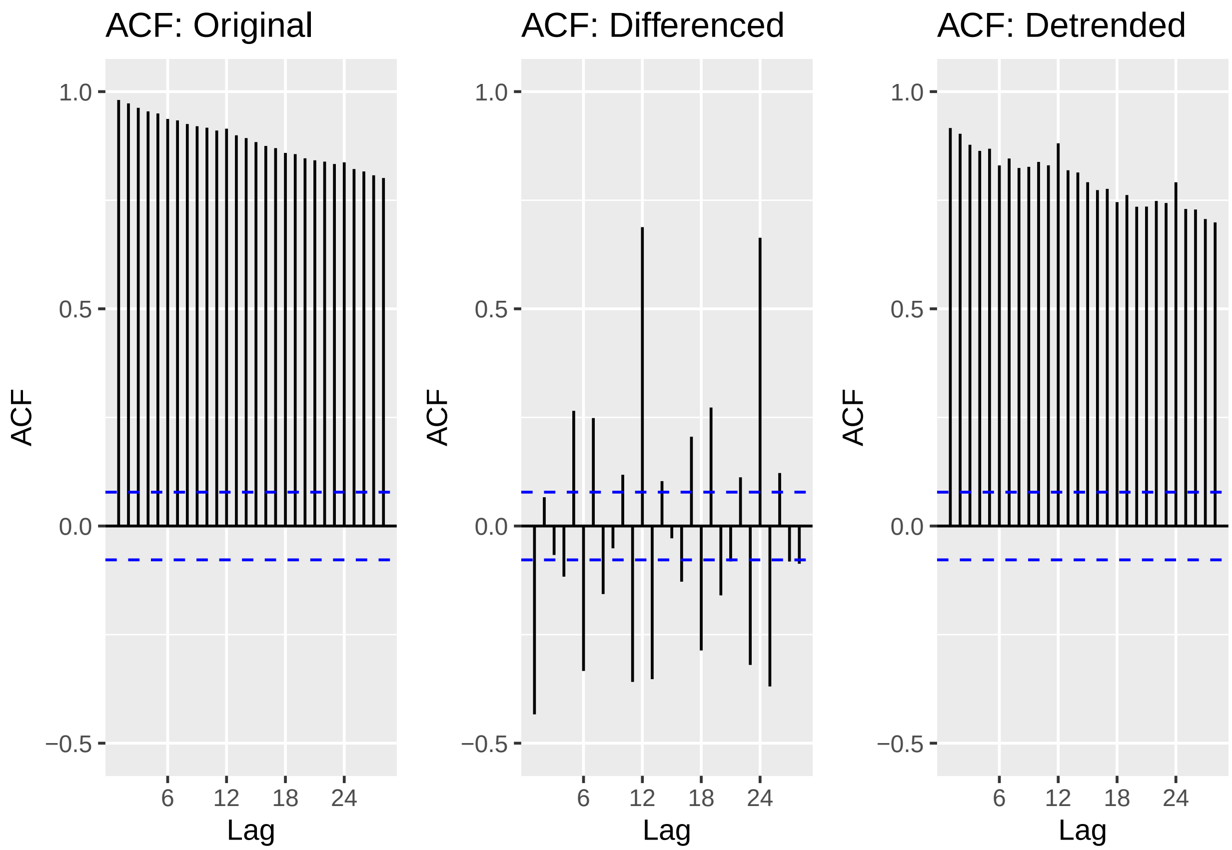
```
p2 <- ggAcf(trp_diff, ylim = c(-0.5, 1)) + ggtitle("ACF: Differenced")
```

```
## Warning in ggplot2::geom_segment(lineend = "butt", ...): Ignoring unknown
## parameters: ‘ylim‘
```

```r
p3 <- ggAcf(trp_detrended, ylim = c(-0.5, 1)) + ggtitle("ACF: Detrended")
```

```
## Warning in ggplot2::geom_segment(lineend = "butt", ...): Ignoring unknown
## parameters: ‘ylim‘
```

```r
plot_grid(p1, p2, p3, ncol = 3)
```



Answer: Differencing seems to be more effective at eliminating the trend than linear regression detrending. The original series shows very high and slowly decaying autocorrelations. After detrending with linear regression, the ACF is still large and persistent across many lags, so some trend-like structure remains. In contrast, the differenced series has much smaller autocorrelations overall, with most lags closer to zero, which shows that the trend has been removed more successfully.

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
smk_result <- SeasonalMannKendall(trp_ts)
print(smk_result)
```

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

```r
adf_result <- adf.test(trp_ts)
print(adf_result)
```

```
##
##  Augmented Dickey-Fuller Test
##
## data:  trp_ts
## Dickey-Fuller = -1.0247, Lag order = 8, p-value = 0.9347
## alternative hypothesis: stationary
```

> Answer: The Seasonal Mann–Kendall test indicates a strong and significant increasing trend in Total Renewable Energy Production, with a very small p-value. The ADF test has a large p-value (0.9347), so we fail to reject the null hypothesis of a unit root, which means the series is non-stationary and has a stochastic trend. This is consistent with what we saw in the Q3 plot, where the original series clearly trends upward and the linear detrending does not fully remove the trend.

Q6

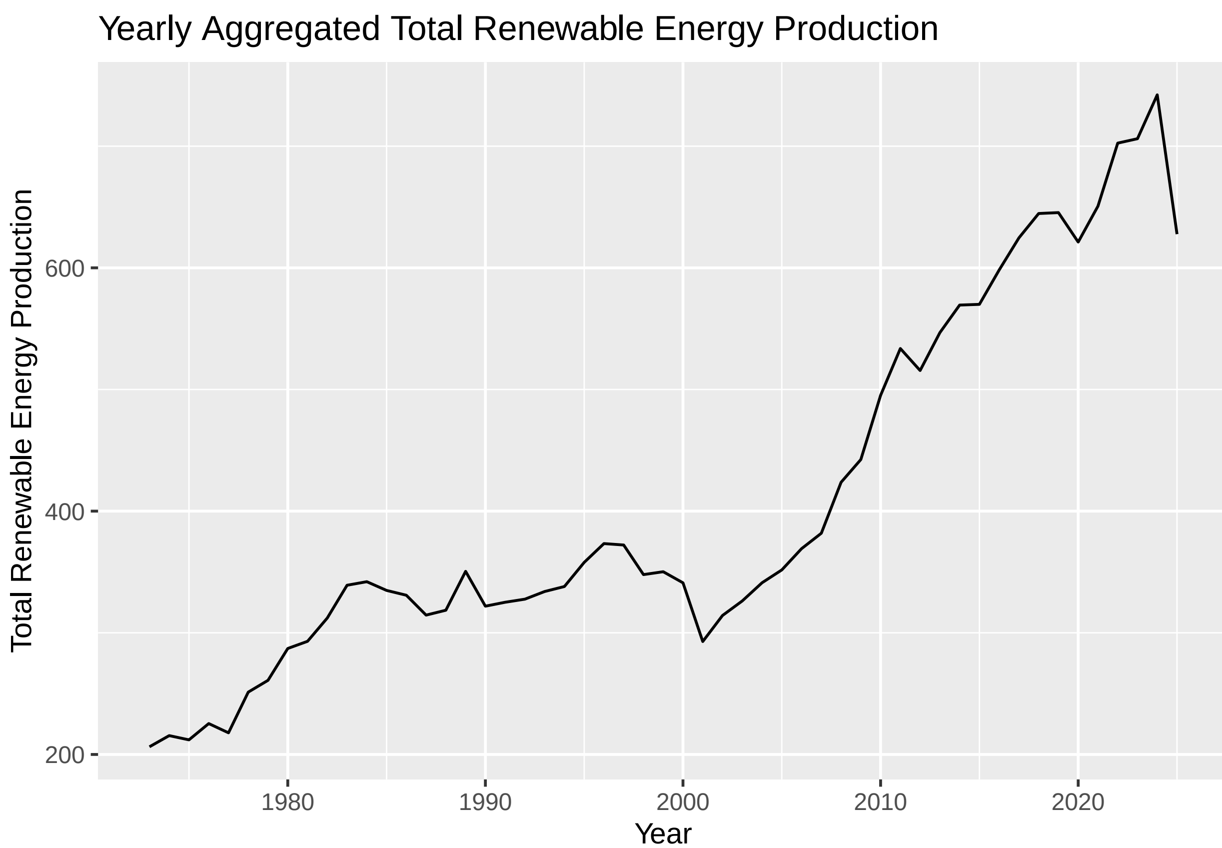
Aggregate the original "Total Renewable Energy Production" series by year. You can use the same procedure we used in class. Store series in a matrix where rows represent months and columns represent years. And then take the columns mean using function colMeans(). Recall the goal is the remove the seasonal 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
trp_mat <- matrix(
  as.numeric(trp_ts),
  nrow = 12,
  byrow = FALSE
)
```

```
## Warning in matrix(as.numeric(trp_ts), nrow = 12, byrow = FALSE): data length
## [633] is not a sub-multiple or multiple of the number of rows [12]
```

```r
trp_yearly <- colMeans(trp_mat, na.rm = TRUE)

start_year <- start(trp_ts)[1]
trp_yearly_ts <- ts(trp_yearly, start = start_year, frequency = 1)
autoplot(trp_yearly_ts) +
  xlab("Year") +
  ylab("Total Renewable Energy Production") +
  ggtitle("Yearly Aggregated Total Renewable Energy Production")
```



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?

```
mk_year <- MannKendall(trp_yearly_ts)
print(mk_year)
```

```
## tau = 0.816, 2-sided pvalue =< 2.22e-16
```

```
t_year <- time(trp_yearly_ts)
spearman_year <- cor.test(t_year, as.numeric(trp_yearly_ts), method = "spearman")
print(spearman_year)
```

```
##
##  Spearman's rank correlation rho
##
## data:  t_year and as.numeric(trp_yearly_ts)
## S = 1898, p-value < 2.2e-16
## alternative hypothesis: true rho is not equal to 0
## sample estimates:
##       rho
## 0.9234801
```

```r
adf_year <- adf.test(trp_yearly_ts)
print(adf_year)
```

```
##
##  Augmented Dickey-Fuller Test
##
## data:  trp_yearly_ts
## Dickey-Fuller = -1.6789, Lag order = 3, p-value = 0.7037
## alternative hypothesis: stationary
```

> Answer: Yes, the results are basically the same as what we saw in Q5. For the yearly series, both the Mann–Kendall test and the Spearman rank correlation test show a very strong and significant upward trend, which has very large tau/rho and extremely small p-values. This is consistent with the monthly results. On the other hand, the ADF test still has a large p-value, so we fail to reject the null hypothesis of a unit root. That means the yearly series is also non-stationary and has a stochastic trend.