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

Assignment 4 - Due date 02/10/26

Jeffrey Chu

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

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"**.

```
#Importing data set - you may copy your code from A3
file_path <- "C:/Users/Jeffrey Chu/OneDrive/Duke MEM coursework/Spring 2026/ENVIRON 797/Week 3/HW 3/Tab

#Importing data set
```

```r
energy_data1 <- read_excel(path=file_path,skip = 12, sheet="Monthly Data",col_names=FALSE)

#Now let's extract the column names from row 11
read_col_names <- read_excel(path=file_path,skip = 10,n_max = 1, sheet="Monthly Data",col_names=FALSE)

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

#Visualize the first rows of the data set
head(energy_data1)
```

```
## # A tibble: 6 x 14
##   Month               `Wood Energy Production` `Biofuels Production`
##   <dttm>                                 <dbl> <chr>
## 1 1973-01-01 00:00:00                     130. Not Available
## 2 1973-02-01 00:00:00                     117. Not Available
## 3 1973-03-01 00:00:00                     130. Not Available
## 4 1973-04-01 00:00:00                     125. Not Available
## 5 1973-05-01 00:00:00                     130. Not Available
## 6 1973-06-01 00:00:00                     125. Not Available
## # i 11 more variables: `Total Biomass Energy Production` <dbl>,
## #   `Total Renewable Energy Production` <dbl>,
## #   `Hydroelectric Power Consumption` <dbl>,
## #   `Geothermal Energy Consumption` <dbl>, `Solar Energy Consumption` <chr>,
## #   `Wind Energy Consumption` <chr>, `Wood Energy Consumption` <dbl>,
## #   `Waste Energy Consumption` <dbl>, `Biofuels Consumption` <chr>,
## #   `Total Biomass Energy Consumption` <dbl>, ...
```

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?

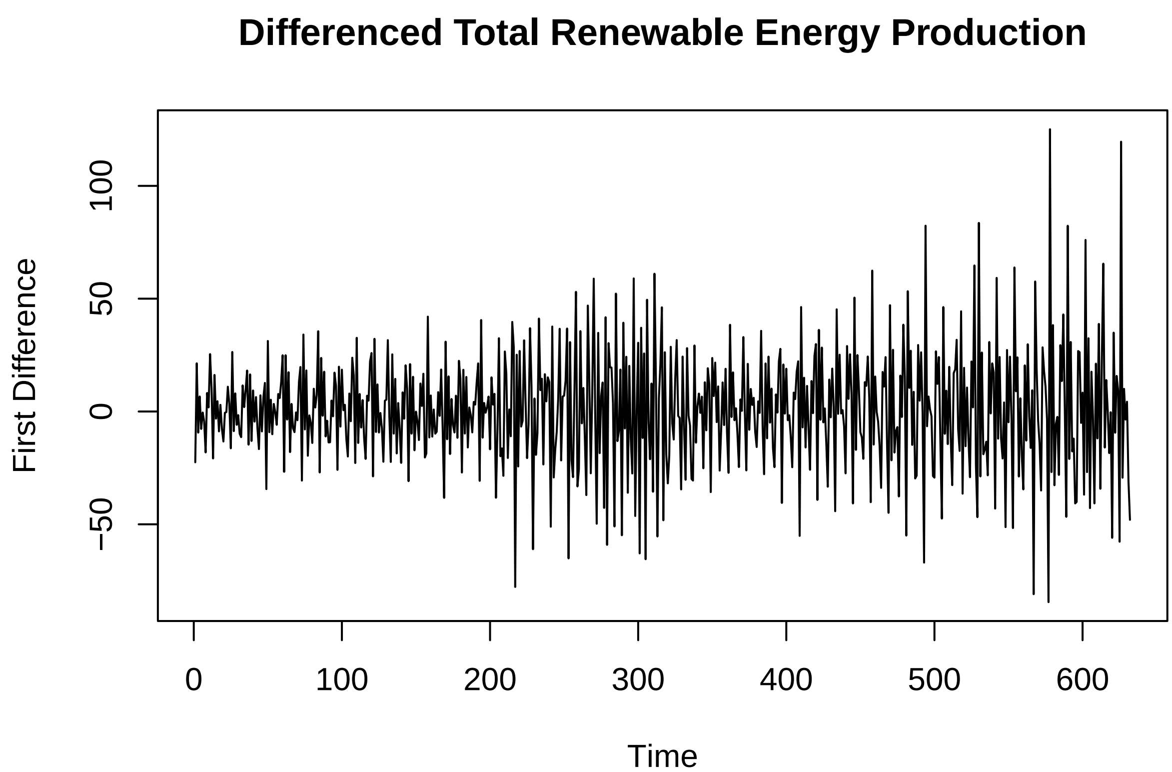
```r
renew_ts <- energy_data1[["Total Renewable Energy Production"]]

head(renew_ts)
```

```
## [1] 219.839 197.330 218.686 209.330 215.982 208.249
```

```r
diff_renew_ts <- diff(
  x = renew_ts,
  lag = 1,
  differences = 1
)
```

```r
plot(
  diff_renew_ts,
  type = "l",
  main = "Differenced Total Renewable Energy Production",
  ylab = "First Difference",
  xlab = "Time"
)
```



Answer: After first differencing at lag 1, the series no longer exhibits a clear trend. The differenced series fluctuates around zero, suggesting that first differencing effectively removes the trend in the original series.

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
t <- 1:length(renew_ts)

lm_renew <- lm(renew_ts ~ t)

detrend_renew_data <- renew_ts - fitted(lm_renew)
```

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().

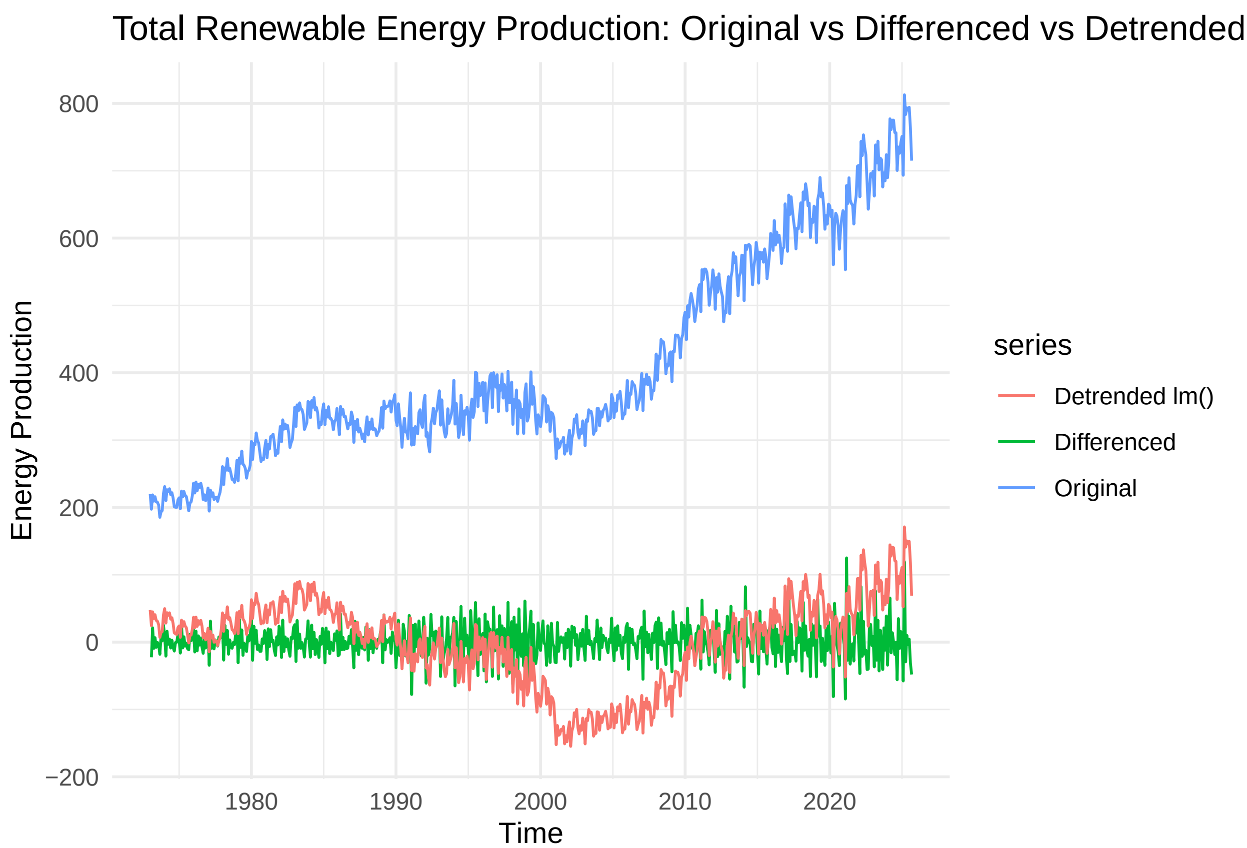
What can you tell from this plot? Which method seems to have been more efficient in removing the trend?

```r
# Original series
ts_renew <- ts(renew_ts,
               frequency = 12,
               start = c(1973, 1))

# Differenced series (Q1)
ts_diff_renew <- ts(diff_renew_ts,
                    frequency = 12,
                    start = c(1973, 2))

# Detrended series (Q2 / A3)
ts_detrend_renew <- ts(detrend_renew_data,
                       frequency = 12,
                       start = c(1973, 1))

autoplot(ts_renew, series = "Original") +
  autolayer(ts_diff_renew, series = "Differenced") +
  autolayer(ts_detrend_renew, series = "Detrended lm()") +
  labs(
    title = "Total Renewable Energy Production: Original vs Differenced vs Detrended",
    y = "Energy Production",
    x = "Time"
  ) +
  theme_minimal()
```



Answer: The original series shows a strong upward trend over time. Linear detrending removes the deterministic linear trend but the detrended series still exhibits persistent long-term fluctuations. In contrast, the differenced series fluctuates closely around zero with no visible trend, suggesting that differencing is more effective 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?

```r
# ACF of original series
p1 <- autoplot(
  Acf(ts_renew, plot = FALSE),
  ylim = c(-0.5, 1)
) +
  labs(title = "ACF: Original Series")
```

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

```r
# ACF of differenced series
p2 <- autoplot(
```

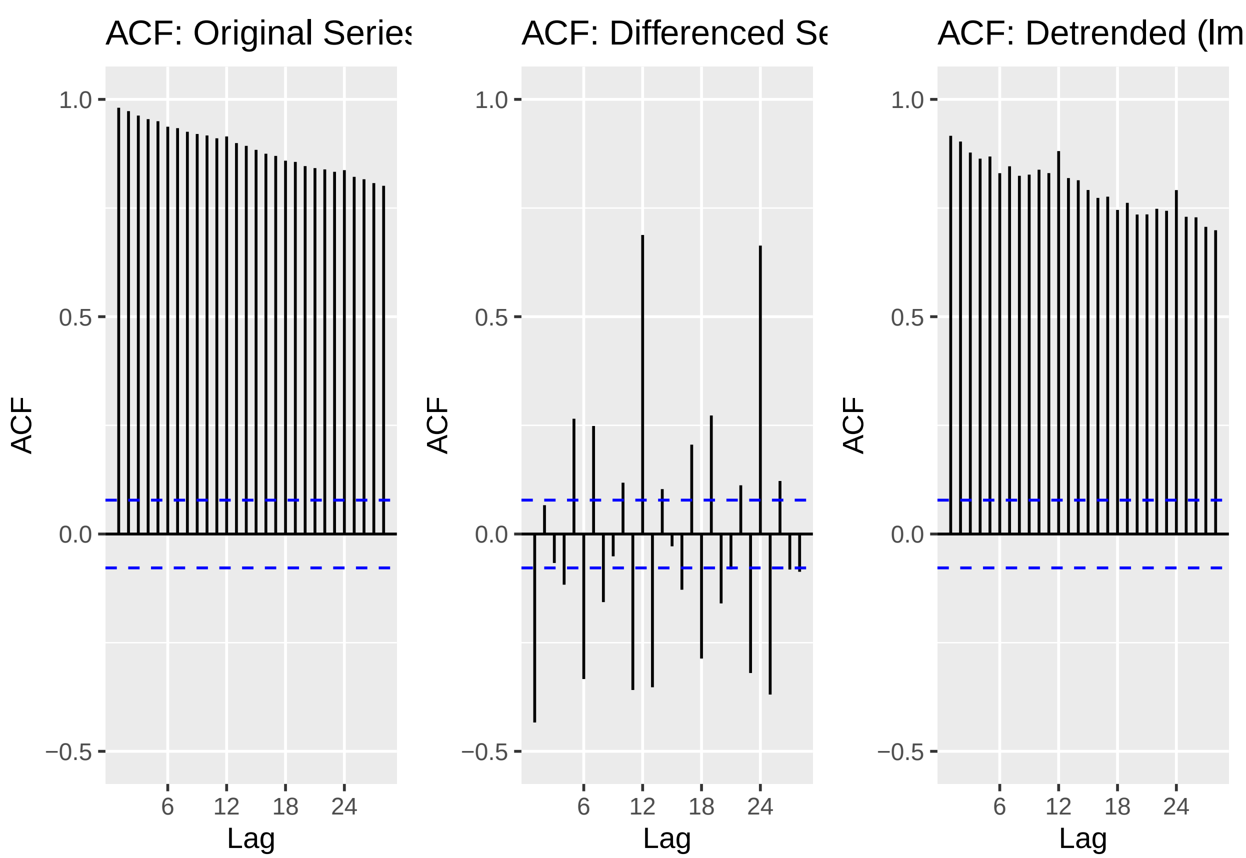
```r
  Acf(ts_diff_renew, plot = FALSE),
  ylim = c(-0.5, 1)
) +
  labs(title = "ACF: Differenced Series")
```

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

```r
# ACF of detrended series
p3 <- autoplot(
  Acf(ts_detrend_renew, plot = FALSE),
  ylim = c(-0.5, 1)
) +
  labs(title = "ACF: Detrended (lm)")
```

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

```r
plot_grid(p1,p2,p3, nrow = 1)
```



Answer: The ACF of the differenced series drops rapidly toward zero, indicating that differencing is more effective 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 test
SMKtest <- SeasonalMannKendall(ts_renew)
print("Results for Seasonal Mann-Kendall test")
```

```
## [1] "Results for Seasonal Mann-Kendall test"
```

```r
print(summary(SMKtest))
```

```
## Score =  13083 , Var(Score) = 201135
## denominator =  16379.5
## tau = 0.799, 2-sided pvalue =< 2.22e-16
## NULL
```

```r
# Augmented Dickey-Fuller test
print("Results for ADF test")
```

```
## [1] "Results for ADF test"
```

```r
print(adf.test(ts_renew, alternative = "stationary"))
```

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

> Answer: The Seasonal Mann-Kendall test result rejects the null hypothesis, indicating a statistically significant increasing trend in Total Renewable Energy Production. The ADF test fails to reject the null hypothesis, suggesting that the series is non-stationary and exhibits a stochastic trend. These results are consistent with the strong upward trend observed in the Q3 plot. Since the series contains a stochastic trend, differencing is required to remove the trend and achieve stationarity.

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
frequency(ts_renew)
```

```
## [1] 12
```

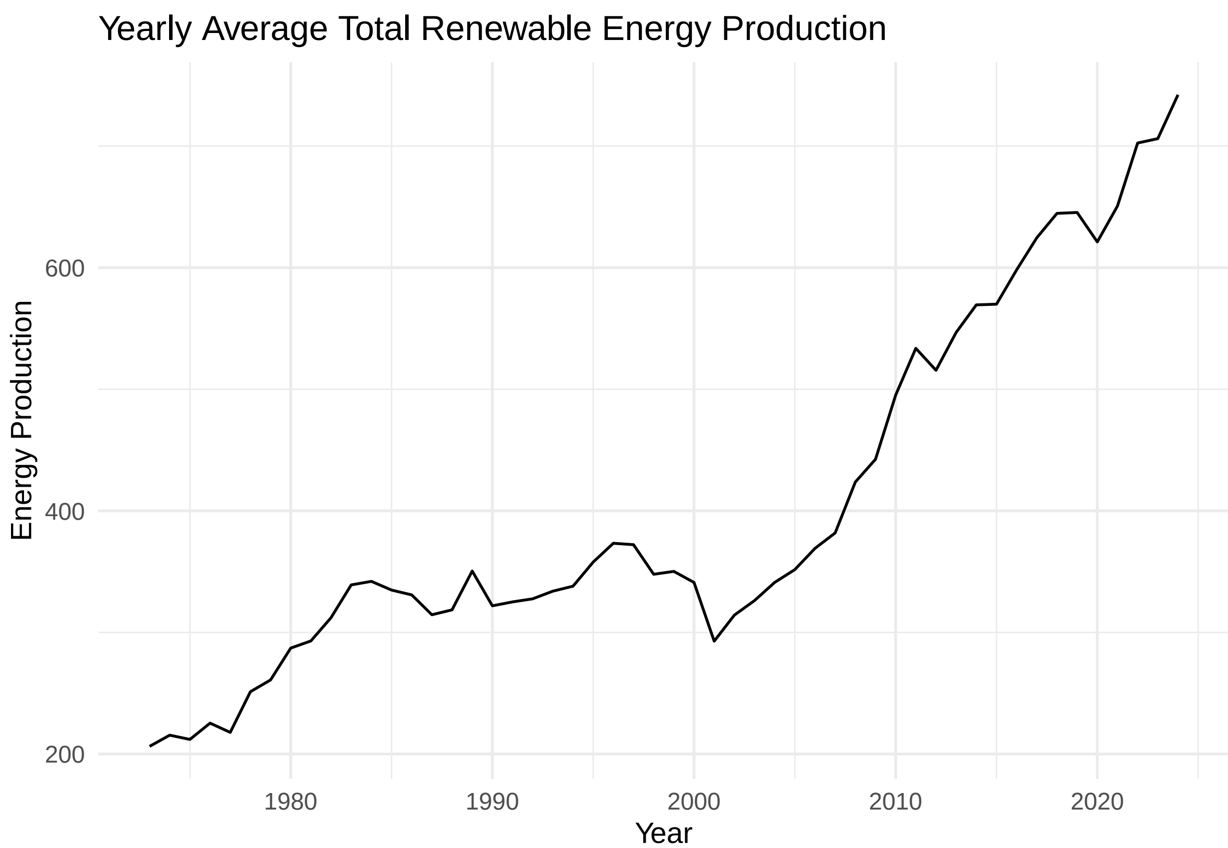
```r
# Number of complete years
n_years <- length(ts_renew) / 12

# Store data in a matrix: rows = months, columns = years
renew_matrix <- matrix(ts_renew,
                       nrow = 12,
                       ncol = n_years)
```

```
## Warning in matrix(ts_renew, nrow = 12, ncol = n_years): data length [633] is
## not a sub-multiple or multiple of the number of rows [12]
```

```r
# Take column means to remove seasonal variation
renew_yearly <- colMeans(renew_matrix, na.rm = TRUE)
# Convert to yearly time series
ts_renew_yearly <- ts(renew_yearly,
                      start = 1973,
                      frequency = 1)

autoplot(ts_renew_yearly) +
  labs(
    title = "Yearly Average Total Renewable Energy Production",
    x = "Year",
    y = "Energy Production"
  ) +
  theme_minimal()
```



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?

```
#Use yearly date to run Mann Kendall
print("Results of Mann Kendall on average yearly series")
```

```
## [1] "Results of Mann Kendall on average yearly series"
```

```
print(summary(MannKendall(renew_yearly)))
```

```
## Score =  1084 , Var(Score) = 16059.33
## denominator =  1326
## tau = 0.817, 2-sided pvalue =< 2.22e-16
## NULL
```

```
# Time index for yearly data
t_year <- 1:length(renew_yearly)

# Spearman correlation (deterministic / monotonic trend)
print("Results from Spearman Correlation")
```

```
## [1] "Results from Spearman Correlation"
```

```r
sp_rho <- cor(renew_yearly, t_year, method = "spearman")
print(sp_rho)
```

```
## [1] 0.9209425
```

```r
# Spearman test with statistics
sp_test <- cor.test(renew_yearly, t_year, method = "spearman")
print(sp_test)
```

```
##
##  Spearman's rank correlation rho
##
## data:  renew_yearly and t_year
## S = 1852, p-value < 2.2e-16
## alternative hypothesis: true rho is not equal to 0
## sample estimates:
##       rho
## 0.9209425
```

```r
#Now let's try the yearly data
print("Results for ADF test on yearly data/n")
```

```
## [1] "Results for ADF test on yearly data/n"
```

```r
print(adf.test(renew_yearly, alternative = "stationary"))
```

```
##
##  Augmented Dickey-Fuller Test
##
## data:  renew_yearly
## Dickey-Fuller = -0.85301, Lag order = 3, p-value = 0.9515
## alternative hypothesis: stationary
```

> Answer: The Mann-Kendall test strongly rejects the null hypothesis of no trend, indicating a significant increasing trend in the yearly average. The Spearman correlation test confirms a strong deterministic monotonic relationship between the series and time. However, the ADF test fails to reject the null hypothesis of a unit root, suggesting that the series is non-stationary and exhibits a stochastic trend. These results are consistent with previous visual and ACF-based analyses. Since the trend is stochastic, differencing is required to achieve stationarity.