ENV 797 - Time Series Analysis for Energy and Environment Applications | Spring 2025

Assignment 4 - Due date 02/11/25

Daniel Whitehead

Directions

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.\

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 January 2021 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
library(tseries)
library(Kendall)
library(forecast)
library(tidyverse)
library(dplyr)
library(ggplot2)
library(cowplot)
library(readxl)
library(gridExtra)

Table_10_1_Renewable_Energy_Production_and_Consumption_by_Source <- read_excel(
  "~/ENV797/Data/Table_10.1_Renewable_Energy_Production_and_Consumption_by_Source.xlsx",
  skip = 9)
Energy_df <-
  Table_10_1_Renewable_Energy_Production_and_Consumption_by_Source[-1, ]
```

Stochastic Trend and Stationarity Tests

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

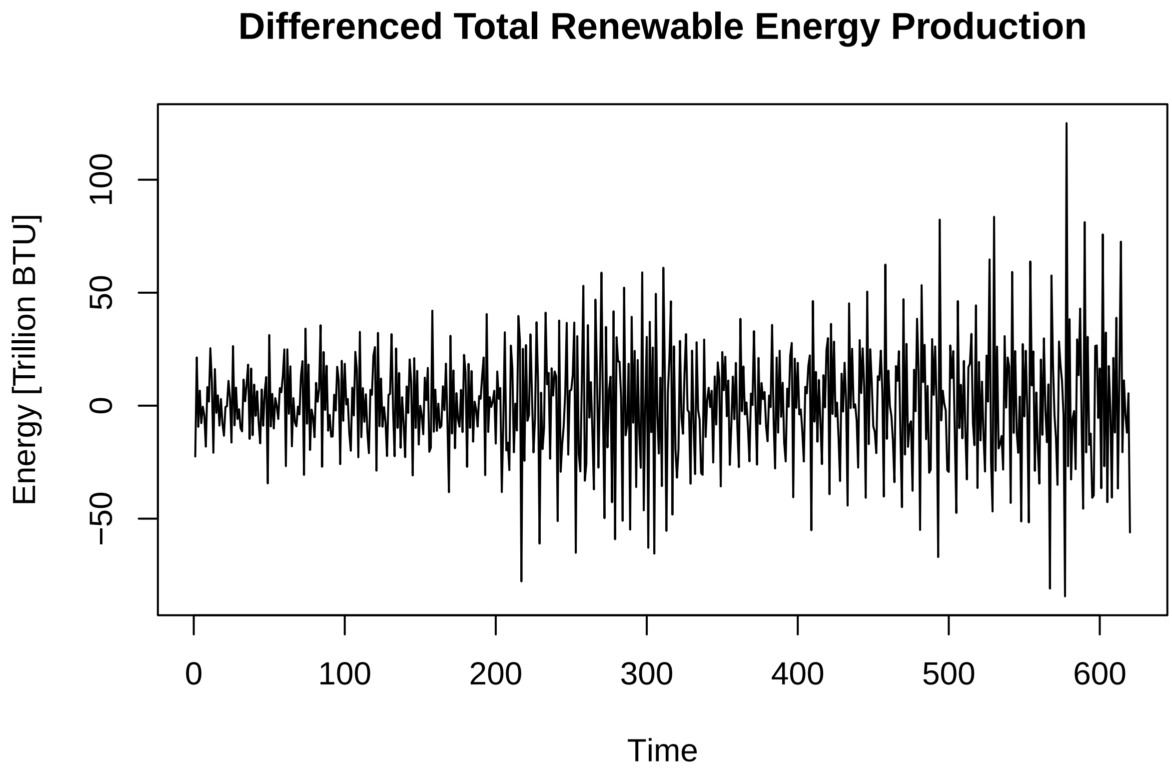
Q1

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.

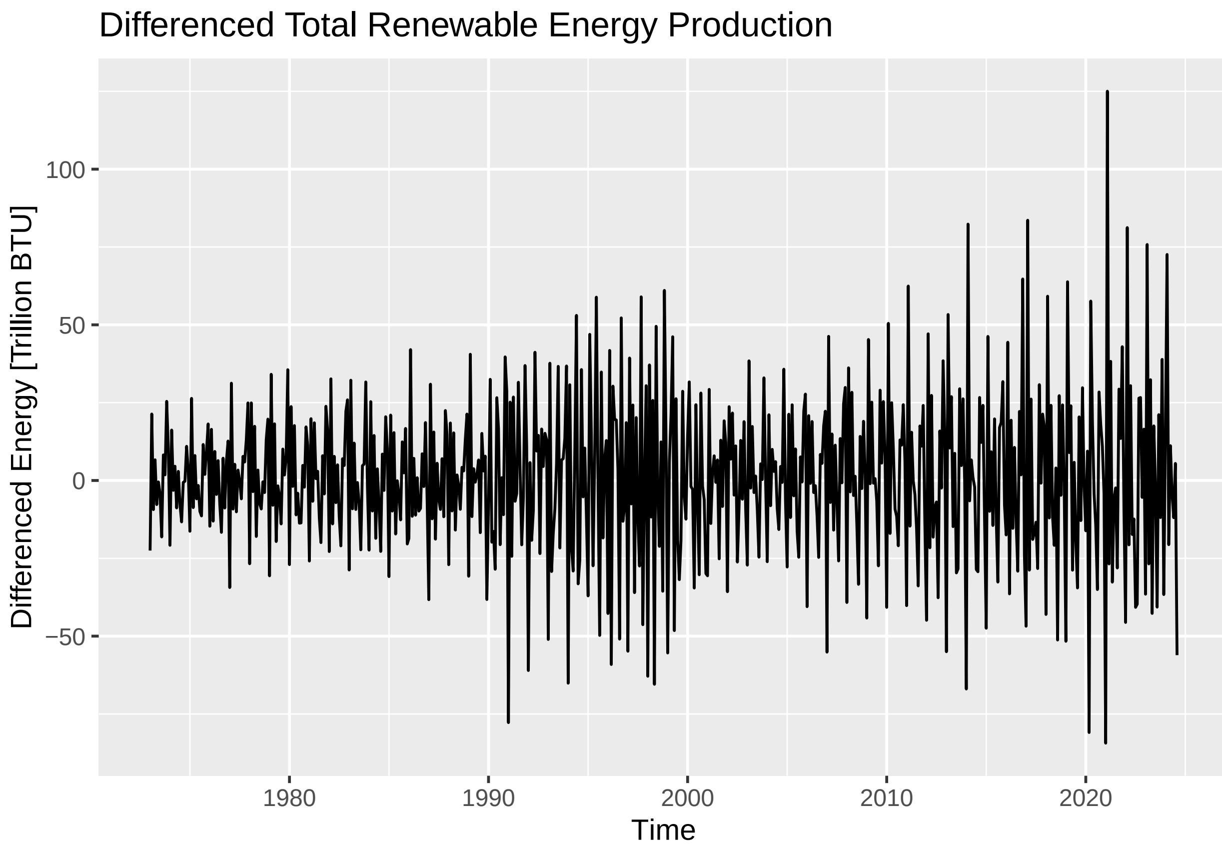
Try differencing at lag 1 only once, i.e., make `lag=1` and `differences=1`. Plot the differenced series. Do the series still seem to have trend?

```
# Had some errors, so made sure data was numeric and checked na
Energy_df$`Total Renewable Energy Production` <- as.numeric(Energy_df$`Total Renewable Energy Productio
Energy_df <- Energy_df[!is.na(Energy_df$`Total Renewable Energy Production`), ]
#selected for just renewable
total_renewable_energy <- Energy_df$`Total Renewable Energy Production`
# use diff function
diff_series <- diff(total_renewable_energy, lag = 1, differences = 1)
diff_ts <- ts(diff_series, frequency = 12, start = c(1973, 1))
# plot diff series
plot(diff_series, type = "line", main = "Differenced Total Renewable Energy Production",
     ylab = "Energy [Trillion BTU]", xlab = "Time")
```

```
## Warning in plot.xy(xy, type, ...): plot type 'line' will be truncated to first
## character
```



```r
#trying my usual autoplot style
autoplot(diff_ts)+
  ylab("Differenced Energy [Trillion BTU]") +
  xlab("Time") +
  ggtitle("Differenced Total Renewable Energy Production")
```




```r
#bringing in ts from last time

selected_energy <- Energy_df %>%
  select(4,5,6)
selected_energy <- as.data.frame(lapply(selected_energy, as.numeric))
# Create the time series object
energy_ts <- ts(selected_energy[,2:3], start = c(1973, 1), frequency = 12)

# Time series plots
ts_renewable <- autoplot(energy_ts[,1], main = "Renewable") +
  ylab("Energy [Trillion BTU]")
```

<The differencing eliminates the upward trend we saw in the original time series plot, though there is still a little bit of volatility in the data.

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 the same name for you time series object that you had in A3, otherwise the code will not work.

```
nobs <- length(energy_ts[,1])
t <- 1:nobs

# Fit linear trend to renewable
linear_model_renewable <- lm(energy_ts[,1] ~ t)
summary(linear_model_renewable)
```

```
##
## Call:
## lm(formula = energy_ts[, 1] ~ t)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -151.11  -37.84   13.53   41.76  149.42
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 176.87293    4.96189   35.65   <2e-16 ***
## t             0.72393    0.01382   52.37   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 61.75 on 619 degrees of freedom
## Multiple R-squared:  0.8159, Adjusted R-squared:  0.8156
## F-statistic:  2743 on 1 and 619 DF,  p-value: < 2.2e-16
```

```
beta0_renewable <- as.numeric(linear_model_renewable$coefficients[1])
beta1_renewable <- as.numeric(linear_model_renewable$coefficients[2])

linear_trend_renewable <- beta0_renewable + beta1_renewable * t
detrended_renewable <- energy_ts[,1] - linear_trend_renewable
ts_detrended_renewable <- ts(detrended_renewable, start = c(1973,1), frequency = 12)
plot_detrended_renewable <- autoplot(ts_detrended_renewable,
                                     main = "Detrended Renewable")+
                                     ylab("Energy [Trillion BTU]")
```

Q3

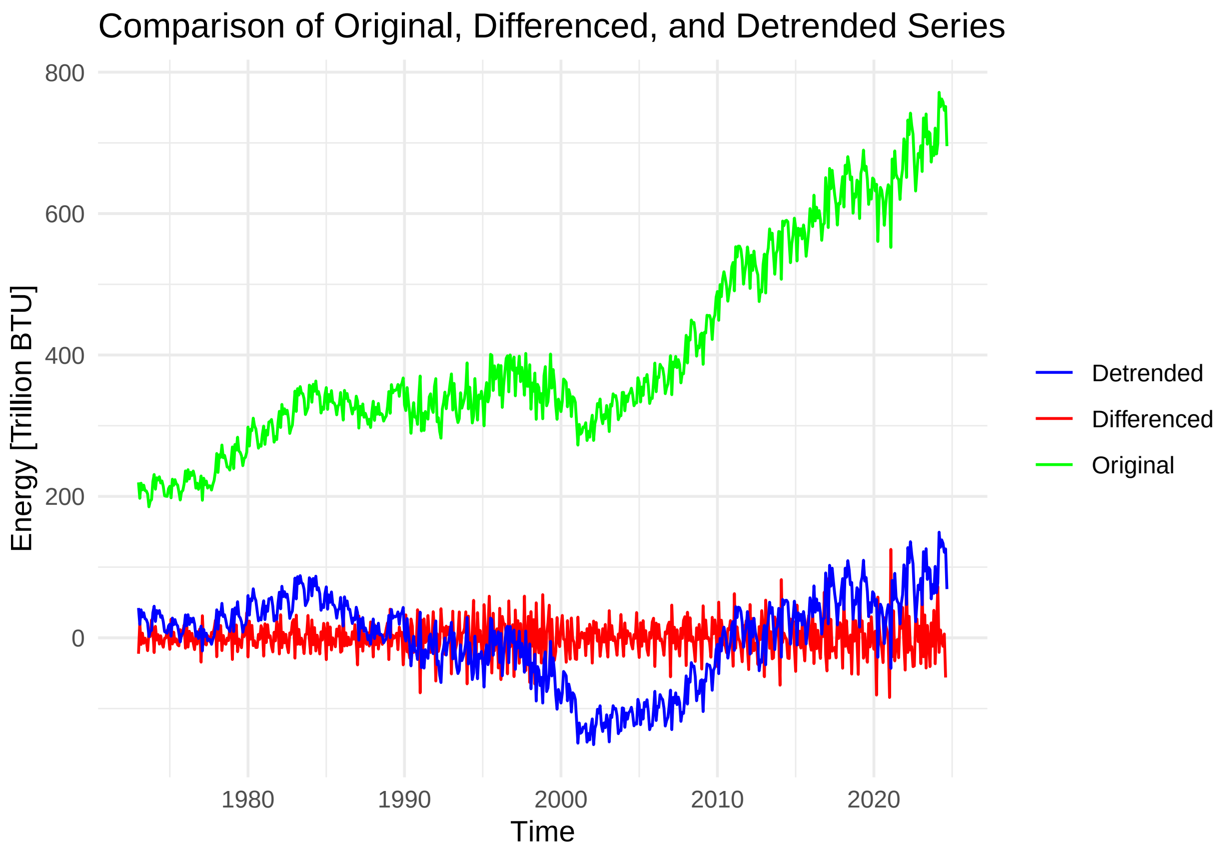
Now let's compare the differenced series with the detrended series you calculated on A3. In other words, for the "Total Renewable Energy Production" compare the differenced series from Q1 with the series you detrended in Q2 using linear regression.

Using autoplot() + autolayer() create a plot that shows the three series together. 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?

```
ts_renewables <- ts(energy_ts[,1], start = c(1973, 1), frequency = 12)

autoplot(ts_renewables, series = "Original") +
```

```r
autolayer(diff_ts, series = "Differenced") +
autolayer(ts_detrended_renewable, series = "Detrended") +
ggtitle("Comparison of Original, Differenced, and Detrended Series") +
ylab("Energy [Trillion BTU]") +
xlab("Time") +
scale_color_manual(values = c("blue", "red", "green")) +
theme_minimal() +
theme(legend.title = element_blank())
```

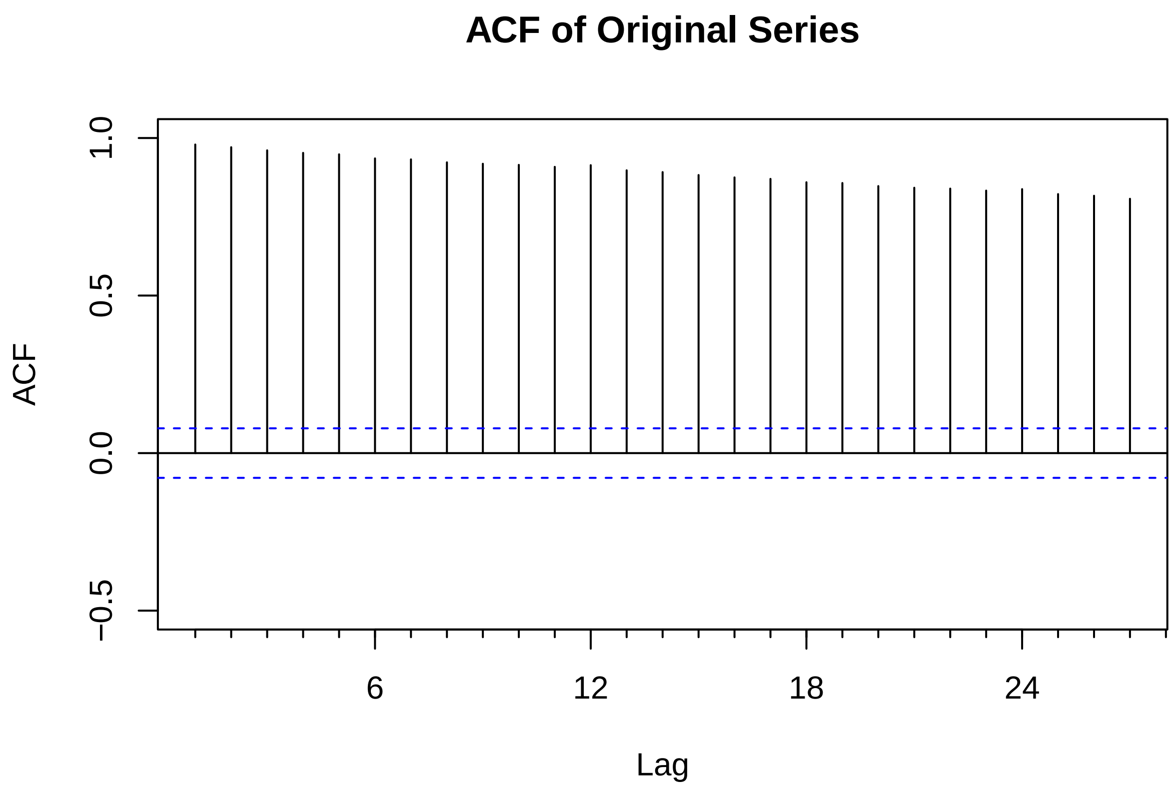


Answer: From this plot, we can tell that the original data has an upward trend, there is more production of renewable energy over time. With the detrended plot, we can see that certain periods of time have similar shape compared to the original data, but this data is centered around zero, meaning the detrending had a major effect. The differenced plot is different than the other two plots- what I mean by this is that this plot does not show any upward or downward trends. The differenced plot looks to be the most effective in removing the trend, as the data remains very close to zero and does not have the same rise and fall over time as the other plots exhibit.

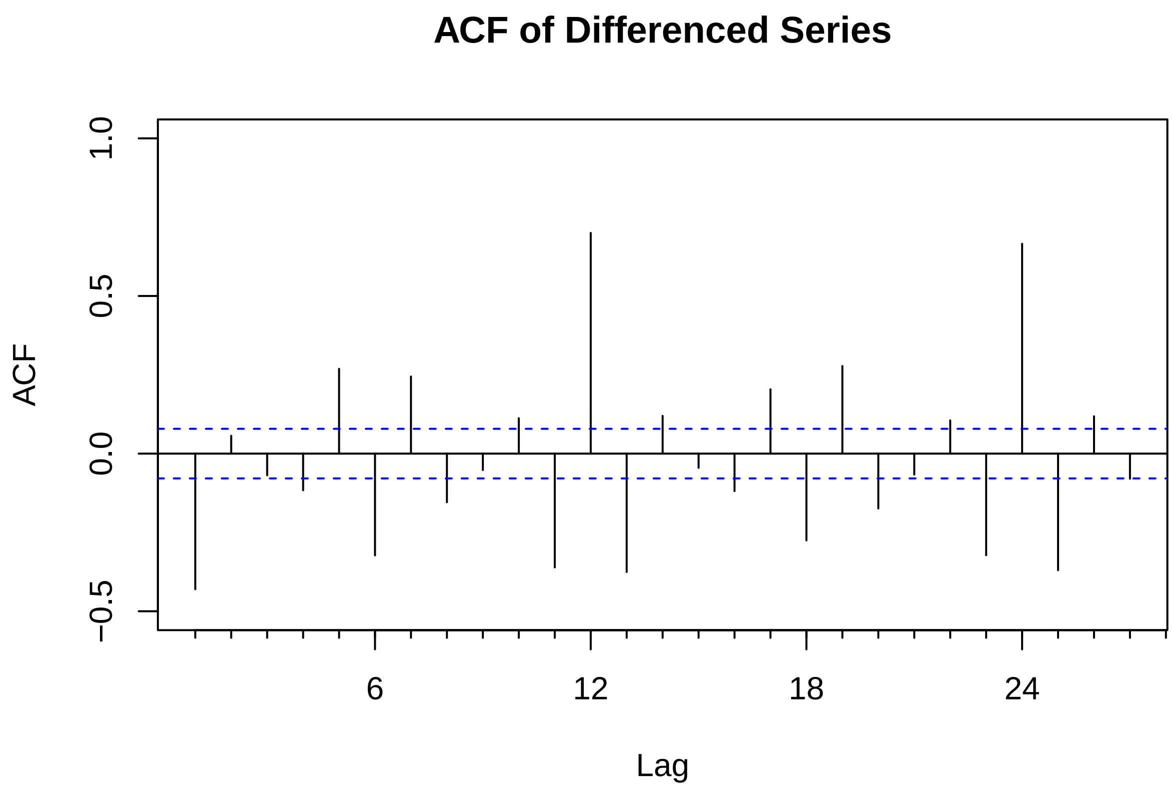
Q4

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?

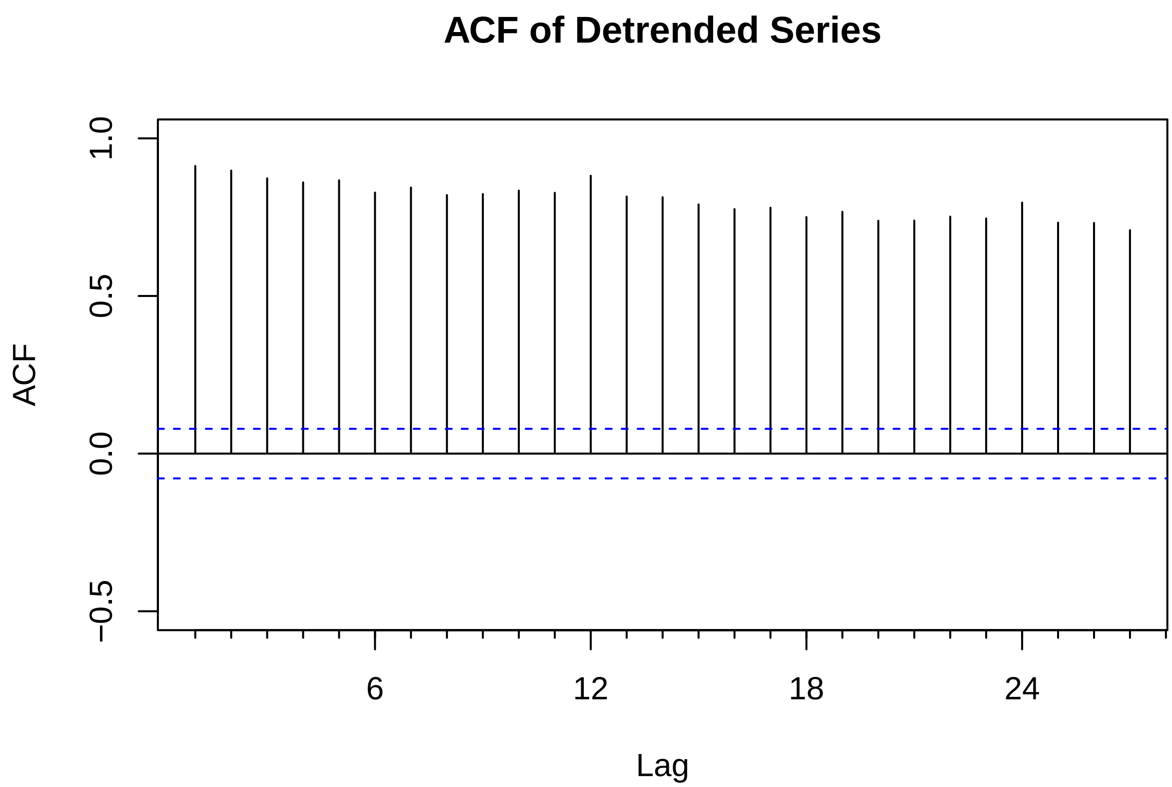
```
acf_original <- Acf(ts_renewables, main = "ACF of Original Series", ylim = c(-0.5, 1))
```



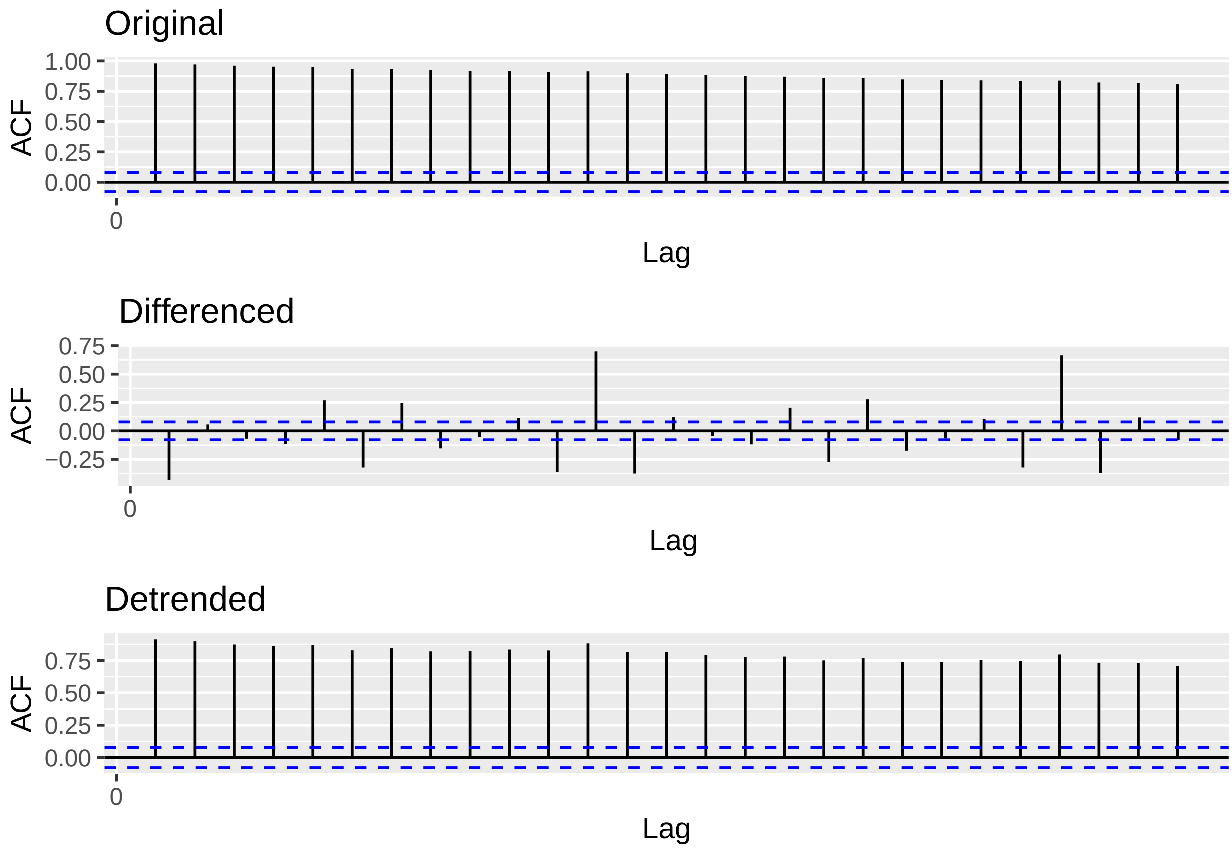

```
# ACF for Differenced Series
acf_diff <- Acf(diff_ts, main = "ACF of Differenced Series", ylim = c(-0.5, 1))
```



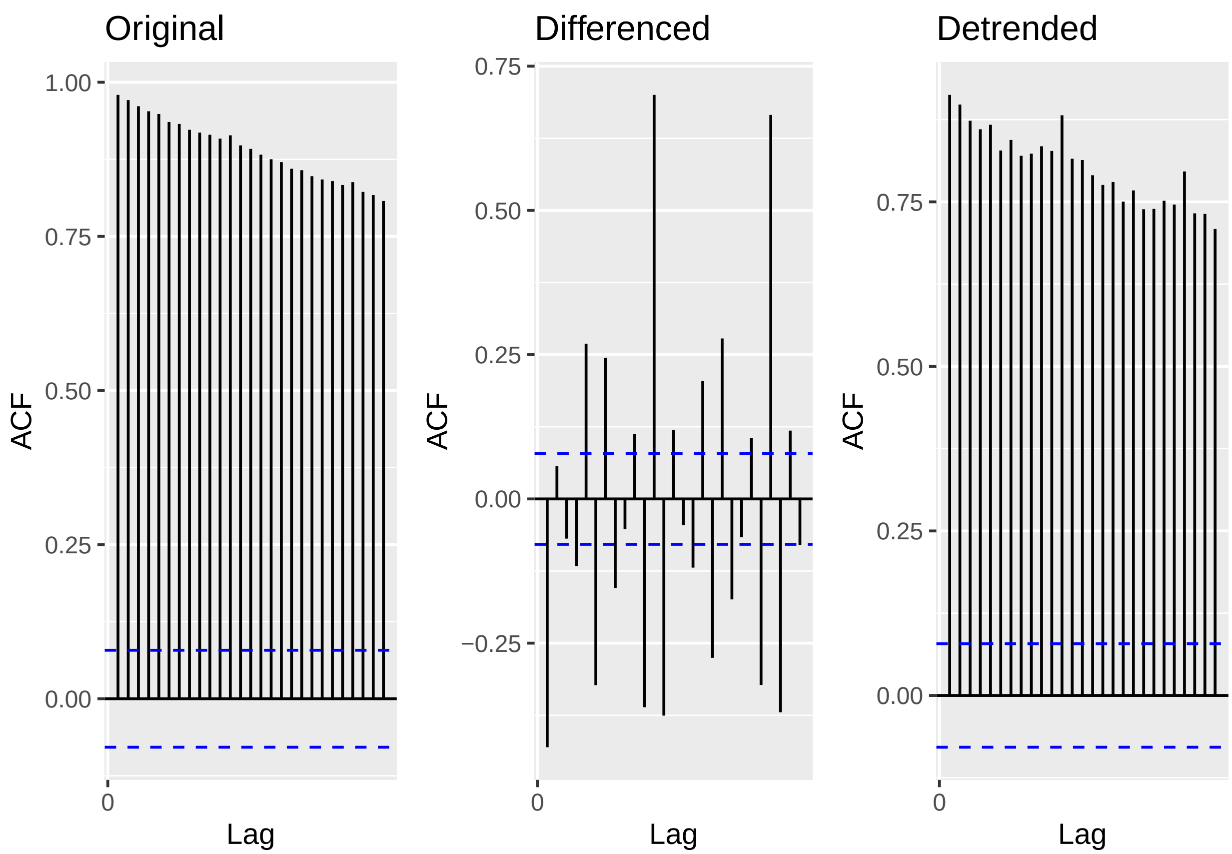

```
# ACF for Detrended Series
acf_detrended <- Acf(ts_detrended_renewable, main = "ACF of Detrended Series", ylim = c(-0.5, 1))
```




```
acf_original_plot <- autoplot(acf(ts_renewables,
                                   main = "ACF of Original Series",
                                   ylim = c(-0.5, 1),
                                   plot = FALSE))+
  ggtitle("Original")


acf_diff_plot <- autoplot(acf(diff_ts,
                                   main = "ACF of Differenced Series",
                                   ylim = c(-0.5, 1),
                                   plot = FALSE))+
  ggtitle("Differenced")


acf_detrended_plot <- autoplot(acf(ts_detrended_renewable,
                                   main = "ACF of Detrended Series",
                                  ylim = c(-0.5, 1),
                                   plot = FALSE))+
  ggtitle("Detrended")

renewable_acf_series <- plot_grid(acf_original_plot, acf_diff_plot,
                        acf_detrended_plot,
                        nrow = 3, ncol = 1)
renewable_acf_series
```




```
grid.arrange(acf_original_plot, acf_diff_plot, acf_detrended_plot, ncol = 3)
```



Answer: The differenced acf plot looks more efficient in eliminating the trend. Instead of the data having a gradual slope downwards in the original acf, the differenced acf is centered around 0 and does not have the slope that the original or detrended exhibit.

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_renewable <- SeasonalMannKendall(ts_renewables)

# Print the results
print(smk_renewable)
```

```
## tau = 0.791, 2-sided pvalue =< 2.22e-16
```

```r
adf_renewable <- adf.test(ts_renewables)

# Print the results
print(adf_renewable)
```

```
##
##  Augmented Dickey-Fuller Test
##
## data:  ts_renewables
## Dickey-Fuller = -1.0898, Lag order = 8, p-value = 0.9242
## alternative hypothesis: stationary
```

> Answer: The SMK has a tau of 0.79 which shows a strong positive trend in the data. This makes sense when loking at the original time series, as we can see the stron upward trend. The pvalue is also very small, meaning we would reject the null hypothesis (that the data is stationary). The ADF results show that there is evidence for a unit root and that the series is non-stationary. the -1.09 is not quite negative enough for us to look at the possibility of the data being stationary, and the very high 0.92 p-value says we should not reject the null that the data has a unit root. This makes sense when looking at the Q3 plot because the differencing eliminates a lot of this stochastic trend..

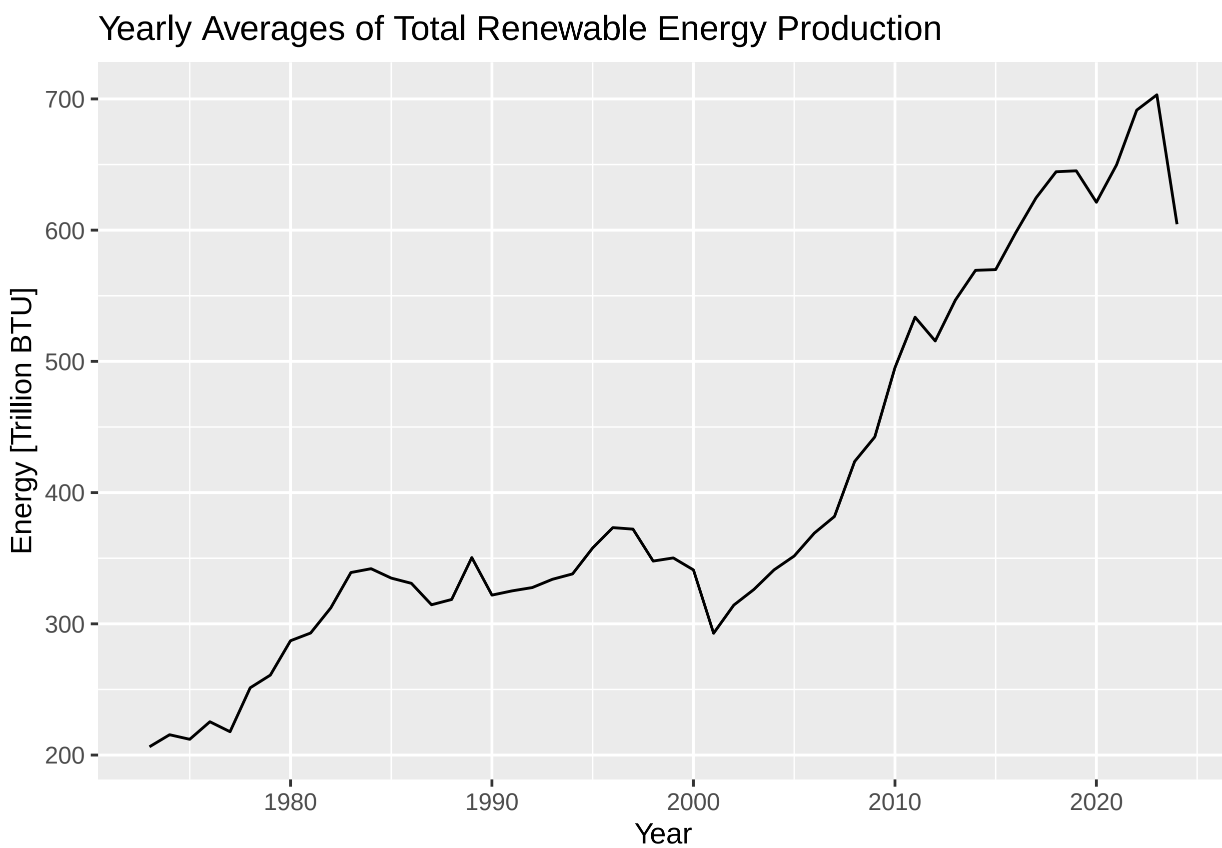
Q6

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
renewable_matrix <- matrix(ts_renewables, nrow = 12, byrow = FALSE)
```

```
## Warning in matrix(ts_renewables, nrow = 12, byrow = FALSE): data length [621]
## is not a sub-multiple or multiple of the number of rows [12]
```

```r
renewable_yearly <- colMeans(renewable_matrix)

my_years <- c(1973:2024)
renewable_yearly_df <- data.frame("year" = my_years, "renewable_data" = renewable_yearly)
renewable_yearly_ts <- ts(renewable_yearly, start = c(1973), frequency = 1)

autoplot(renewable_yearly_ts) +
  ggtitle("Yearly Averages of Total Renewable Energy Production") +
  ylab("Energy [Trillion BTU]") +
  xlab("Year")
```



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 Q6?

```r
# Mann Kendall
mk_renewable <- MannKendall(renewable_yearly_ts)
print(mk_renewable)
```

```
## tau = 0.807, 2-sided pvalue =< 2.22e-16
```

```r
# Spearman's rank correlation test
spearman_renewable <- cor.test(1:length(renewable_yearly_ts), renewable_yearly_ts, method = "spearman")
print(spearman_renewable)
```

```
##
##  Spearman's rank correlation rho
##
## data:  1:length(renewable_yearly_ts) and renewable_yearly_ts
## S = 1908, p-value < 2.2e-16
## alternative hypothesis: true rho is not equal to 0
## sample estimates:
##      rho
## 0.918552
```

```r
#  ADF test
adf_renewable <- adf.test(renewable_yearly_ts)
print(adf_renewable)
```

```
##
##  Augmented Dickey-Fuller Test
##
## data:  renewable_yearly_ts
## Dickey-Fuller = -1.6634, Lag order = 3, p-value = 0.7098
## alternative hypothesis: stationary
```

> Answer: These results are in agreement with the monthly series, as the MK and Spearman tests both have very small pvalues, so we would reject the null that the data is stationary. The ADF has a pvalue that is > 0.05 again, so we would retain the null that the data does have a unit root.