ENV 790.30 - Time Series Analysis for Energy Data | Spring 2024

Assignment 4 - Due date 02/12/24

Cara Kuuskvere

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_Sp23.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(ggplot2)
library(forecast)
```

```
## Registered S3 method overwritten by 'quantmod':
##   method            from
##   as.zoo.data.frame zoo
```

```
library(tseries)
library(Kendall)
```

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 - using readxl package
TableRaw <- read_excel(
  "Data/Table_10.1_Renewable_Energy_Production_and_Consumption_by_Source.xlsx",
  skip=12, col_names = FALSE, col_types = c("date","numeric","numeric",
              "numeric","numeric","numeric","numeric","numeric","numeric",
              "numeric","numeric","numeric","numeric","numeric"))
```

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

```
RE_data <- TableRaw[,5] #want all rows
                                  #only RE column
n_obs <- nrow(RE_data) #number observations

#Adding column names
colnames(RE_data)=c("Total Renewable Energy Production (Trillion Btu)")

RE_data <- as.data.frame(RE_data)
head(RE_data)
```

```
##   Total Renewable Energy Production (Trillion Btu)
## 1                                          219.839
## 2                                          197.330
## 3                                          218.686
## 4                                          209.330
## 5                                          215.982
## 6                                          208.249
```

```
ts_RE_data <- ts(RE_data,start=c(1973,1),frequency=12)
```

Stochastic Trend and Stationarity Tests

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?

```
diff <- autoplot(diff(ts_RE_data,lag=1, differences=1))
```

The series does not seem to still have a trend after it is differenced. It is centered at a mean of zero and appears to not be trending either upward or down.

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 orinal 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.

```
library(lubridate)
```

```
##
## Attaching package: 'lubridate'

## The following objects are masked from 'package:base':
##
##     date, intersect, setdiff, union
```

```
raw_energy_data <- read.csv(
  "Data/Table_10.1_Renewable_Energy_Production_and_Consumption_by_Source.csv",
  header=FALSE,skip=12)
ym_date <- paste(raw_energy_data[,1])
ym_date <- ym(ym_date)  #function my from package lubridate

t <- 1:n_obs

regmodel_renewable=lm(ts_RE_data~t,cbind(ts_RE_data,t))
beta0_renewable=regmodel_renewable$coefficients[1]
beta1_renewable=regmodel_renewable$coefficients[2]
print(summary(regmodel_renewable))
```

```
##
## Call:
## lm(formula = ts_RE_data ~ t, data = cbind(ts_RE_data, t))
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -148.27  -35.63   11.58   41.51  144.27
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 180.98940    4.90151   36.92   <2e-16 ***
## t             0.70404    0.01392   50.57   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 60.41 on 607 degrees of freedom
## Multiple R-squared:  0.8081, Adjusted R-squared:  0.8078
## F-statistic:  2557 on 1 and 607 DF,  p-value: < 2.2e-16
```

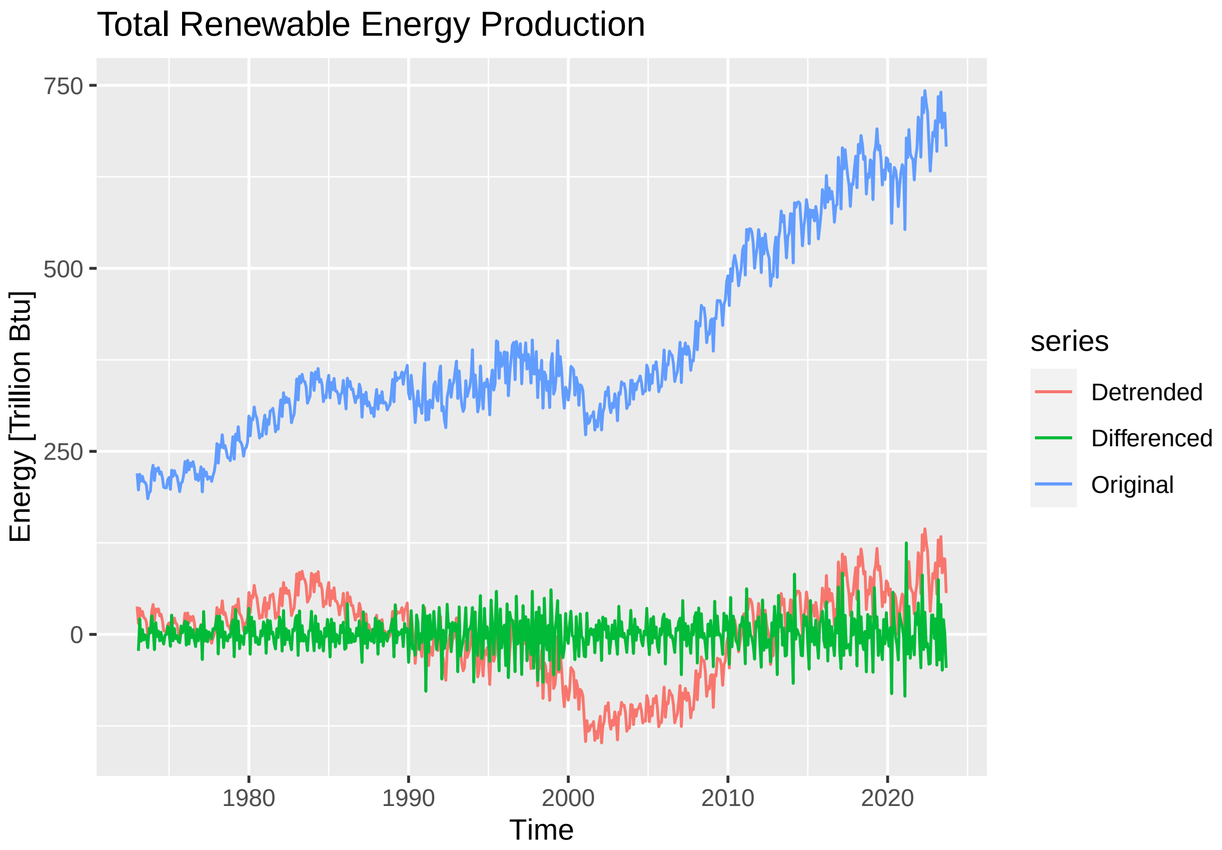
```r
renewable_detrend <- ts_RE_data - (beta0_renewable+ beta1_renewable*t)
renewable_detrend=ts(renewable_detrend, frequency=12,start=c(1973,1))
#the plot from part (d)
```

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.

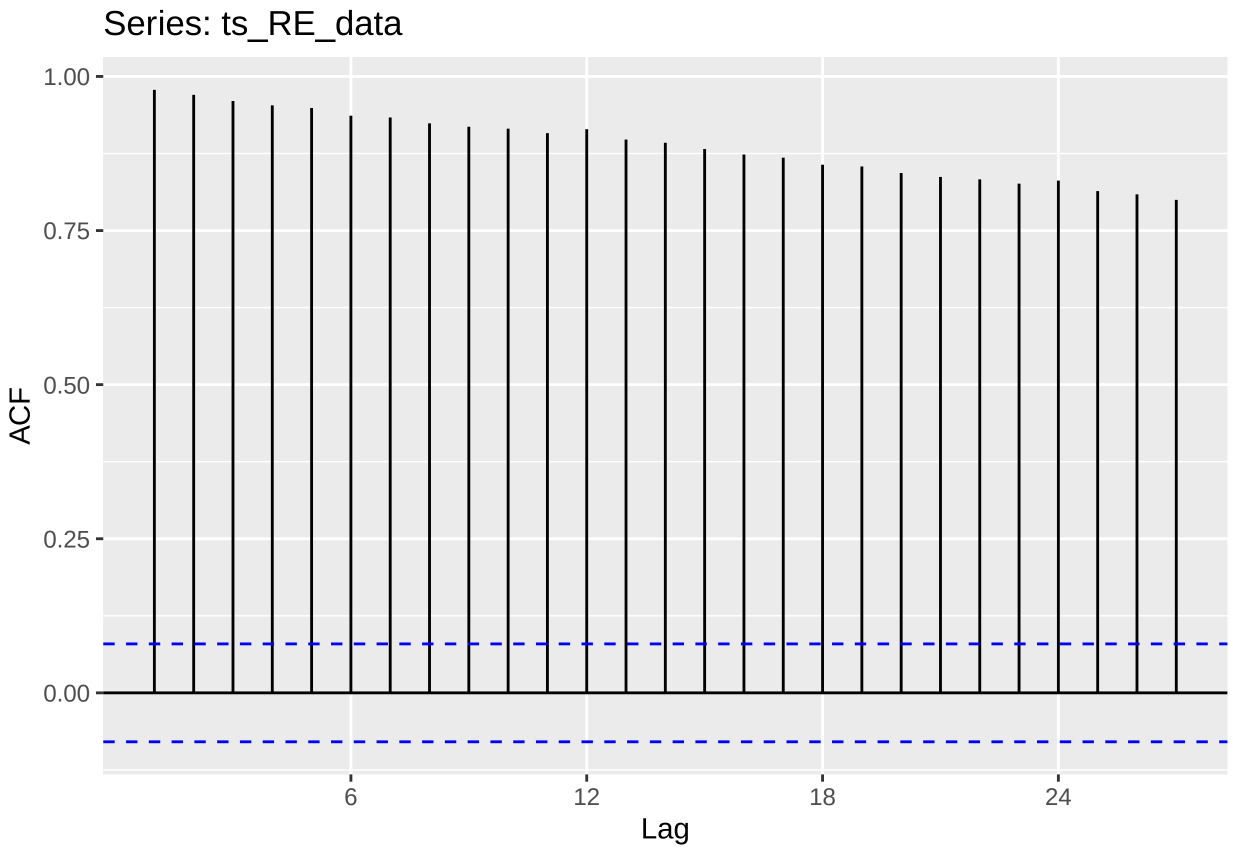
```r
autoplot(ts_RE_data,series="Original") + autolayer(renewable_detrend,series="Detrended") + ylab("Energy
ggtitle("Total Renewable Energy Production")
```



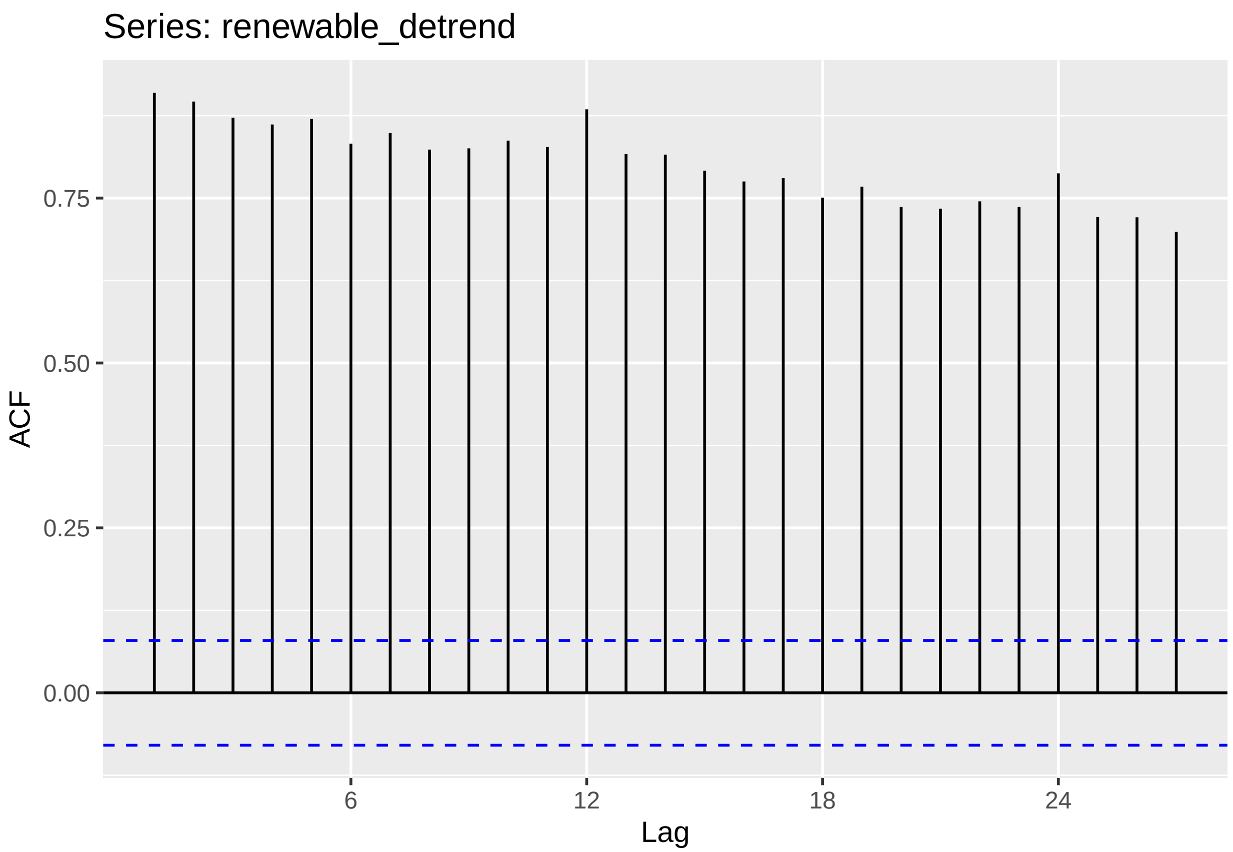
Q4

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. Which method do you think was more efficient in eliminating the trend? The linear regression or differencing?

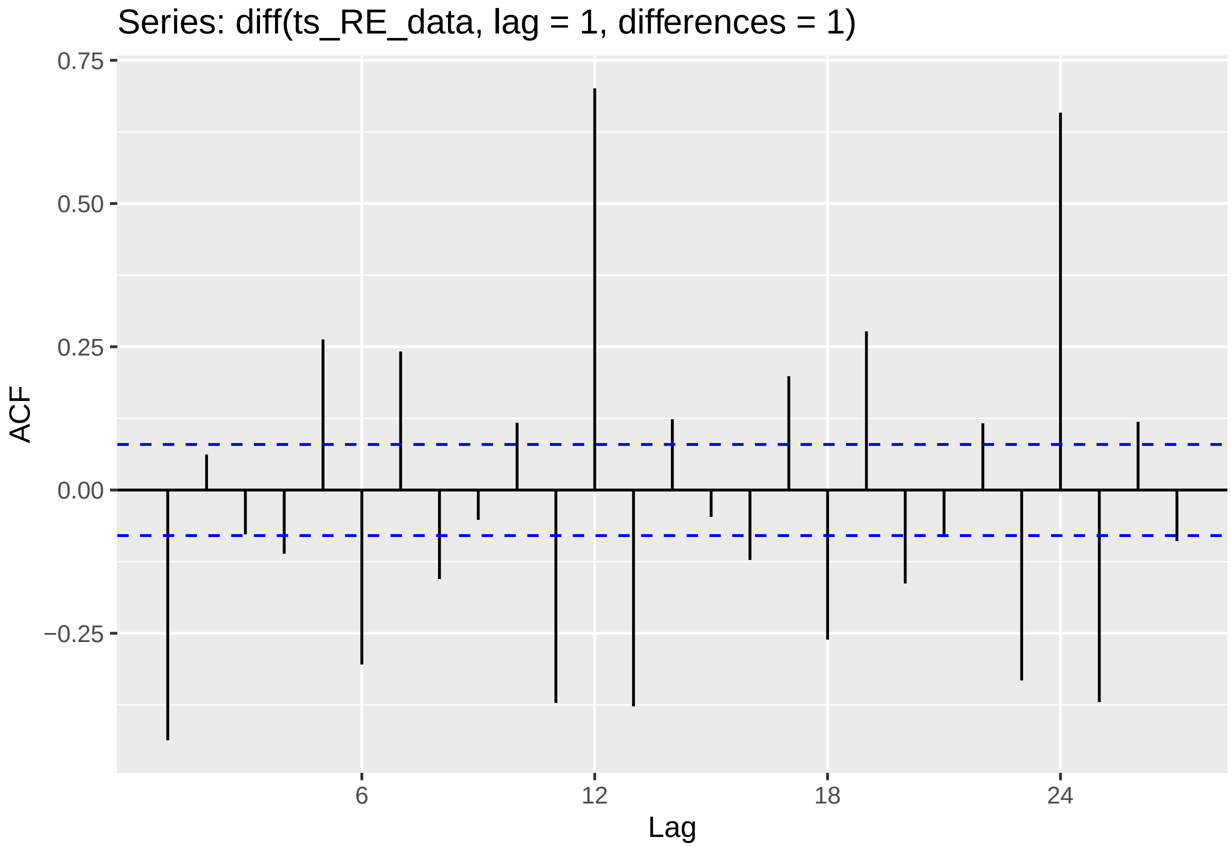
```
library(ggplot2)
ggAcf(ts_RE_data)
```



```
ggAcf(renewable_detrend)
```



```
ggAcf(diff(ts_RE_data,lag=1, differences=1))
```



The differenced method was more efficiency in eliminating the trend. This is because the linear regression shows some changes in the temporal dependency, however this is most heavily eliminated in the differenced ACF as seen by the much lower values for correlation.

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 Q2? Recall that having a unit root means the series has a stochastic trend. And when a series has stochastic trend we need to use a different procedure to remove the trend.

```
print(SeasonalMannKendall(ts_RE_data))
```

```
## tau = 0.783, 2-sided pvalue =< 2.22e-16
```

```
print(adf.test(ts_RE_data))
```

```
##
##  Augmented Dickey-Fuller Test
##
## data:  ts_RE_data
## Dickey-Fuller = -1.24, Lag order = 8, p-value = 0.9
## alternative hypothesis: stationary
```

The seasonal Mann Kendall has a pvalue less than 0.05, which means that we have a trend. The positive value of S means that this is an increasing trend, and the magnitude of S is small due to running the seasonal test.

The ADF shows us that the p value is very high (0.9) so we cannot reject the null hypothesis that the trend is stationary. The ADF tests for stationarity. The results show us that there is a trend in the data.

These findings do reflect what we see in Q2

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().

```
#Group data in yearly steps instances
RE_data_matrix <- matrix(ts_RE_data,byrow=FALSE,nrow=12)
```

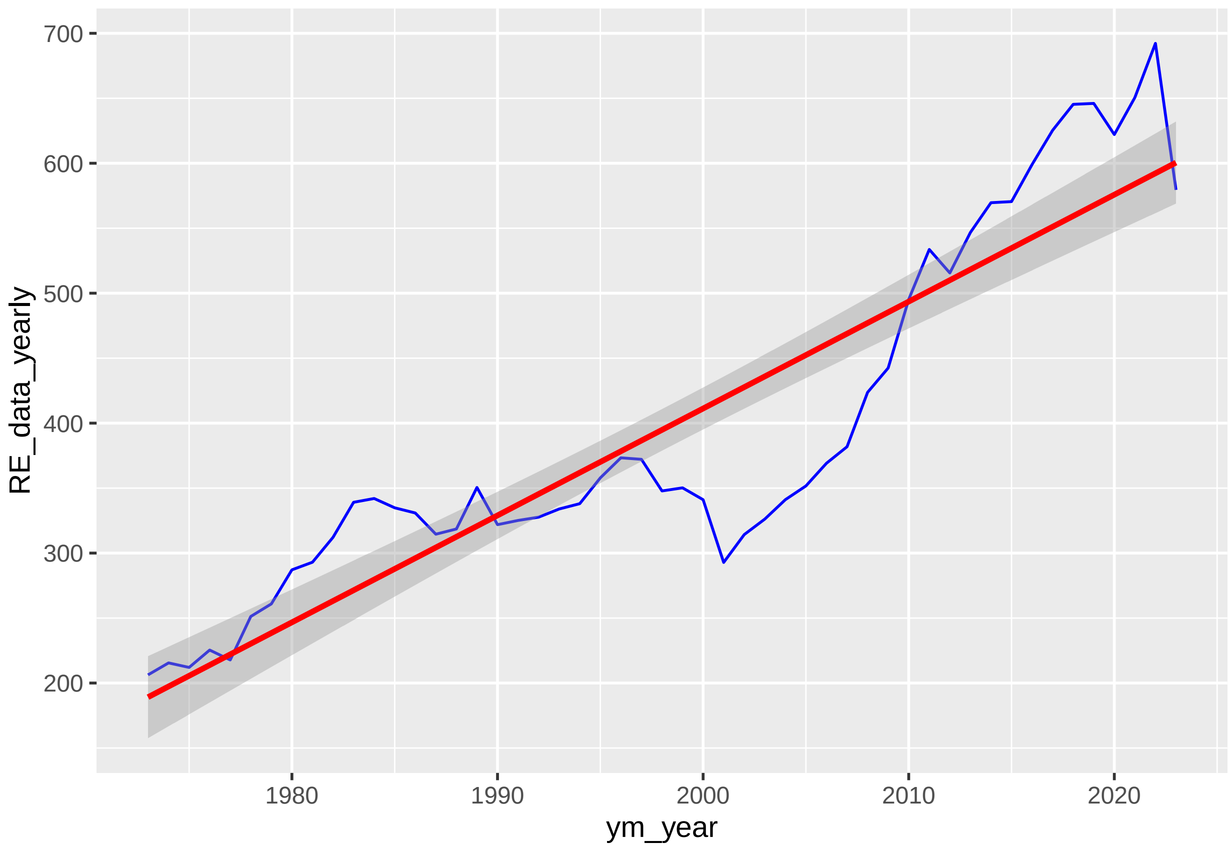
```
## Warning in matrix(ts_RE_data, byrow = FALSE, nrow = 12): data length [609] is
## not a sub-multiple or multiple of the number of rows [12]
```

```
RE_data_yearly <- colMeans(RE_data_matrix)

library(dplyr)  #move this to package chunk later
```

```
##
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':
##
##     filter, lag

## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
ym_year <- c(year(first(ym_date)):year(last(ym_date)))

RE_data_new_yearly <- data.frame(ym_year, RE_data_yearly)

ggplot(RE_data_new_yearly, aes(x=ym_year, y=RE_data_yearly)) +
            geom_line(color="blue") +
            geom_smooth(color="red",method="lm")
```

```
## `geom_smooth()` using formula = 'y ~ x'
```



```
ts_RE_data_new_yearly <- ts(RE_data_new_yearly,start=c(1973),frequency=1)
```

Q7

Apply the Mann Kendal, 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?

```
print(MannKendall(ts_RE_data_new_yearly[,2]))
```

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

```
print(adf.test(ts_RE_data_new_yearly[,2]))
```

```
##
##  Augmented Dickey-Fuller Test
##
## data:  ts_RE_data_new_yearly[, 2]
## Dickey-Fuller = -2.0953, Lag order = 3, p-value = 0.5361
## alternative hypothesis: stationary
```

```
print(cor.test(x=ts_RE_data_new_yearly[,1],y=ts_RE_data_new_yearly[,2]))
```

```
##
##  Pearson's product-moment correlation
##
## data:  ts_RE_data_new_yearly[, 1] and ts_RE_data_new_yearly[, 2]
## t = 15.204, df = 49, p-value < 2.2e-16
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  0.8440559 0.9468997
## sample estimates:
##       cor
## 0.9083501
```

The Mann Kendall now shows that there is a deterministic trend, and that this time series is not stationary as the P value is small.

The ADF shows that the p value is also large, which means that we cannot reject the null that the trend is stationary compared to a unit root, not that there is no trend.

The Spearman correlation rank shows that there is a true correlation not equal to zero in the data between RE output and time.

All three tests show that there is a trend in the data.