ENV 790.30 - Time Series Analysis for Energy Data | Spring 2022

Assignment 4 - Due date 02/17/22

Yu Hai

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 project open the first thing you will do is change "Student Name" on line 3 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. Rename the pdf file such that it includes your first and last name (e.g., "LuanaLima_TSA_A04_Sp21.Rmd"). 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)
```

```
## Warning:  'readxl' R 4.1.2
```

```
library(forecast)
```

```
## Warning:  'forecast' R 4.1.2
```

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

```
library(tseries)
```

```
## Warning:  'tseries' R 4.1.2
```

```
library(Kendall)
```

```
## Warning:  'Kendall' R 4.1.2
```

```
library(lubridate)
```

```
##
##     'lubridate'

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

```
library(ggplot2)
```

Questions

Consider the same data you used for A3 from the spreadsheet "Table_10.1_Renewable_Energy_Production_and_Consumpt
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 xlsx package
raw_RE_data<-read_excel(path="./Data/Table_10.1_Renewable_Energy_Production_and_Consumption_by_Source.x
raw_RE_data
```

```
## # A tibble: 586 x 14
##    Month               `Wood Energy Prod~ `Biofuels Product~ `Total Biomass Ene~
##    <dttm>              <chr>              <chr>              <chr>
##  1 NA                  (Trillion Btu)     (Trillion Btu)     (Trillion Btu)
##  2 1973-01-01 00:00:00 129.63             Not Available      129.787
##  3 1973-02-01 00:00:00 117.194            Not Available      117.338
##  4 1973-03-01 00:00:00 129.763            Not Available      129.938
##  5 1973-04-01 00:00:00 125.462            Not Available      125.636
##  6 1973-05-01 00:00:00 129.624            Not Available      129.834
##  7 1973-06-01 00:00:00 125.435            Not Available      125.611
##  8 1973-07-01 00:00:00 129.616            Not Available      129.787
##  9 1973-08-01 00:00:00 129.734            Not Available      129.918
## 10 1973-09-01 00:00:00 125.603            Not Available      125.782
## # ... with 576 more rows, and 10 more variables:
## #   Total Renewable Energy Production <chr>,
## #   Hydroelectric Power Consumption <chr>, Geothermal Energy Consumption <chr>,
## #   Solar Energy Consumption <chr>, Wind Energy Consumption <chr>,
## #   Wood Energy Consumption <chr>, Waste Energy Consumption <chr>,
## #   Biofuels Consumption <chr>, Total Biomass Energy Consumption <chr>,
## #   Total Renewable Energy Consumption <chr>
```

Stochastic Trend and Stationarity Tests

Q1

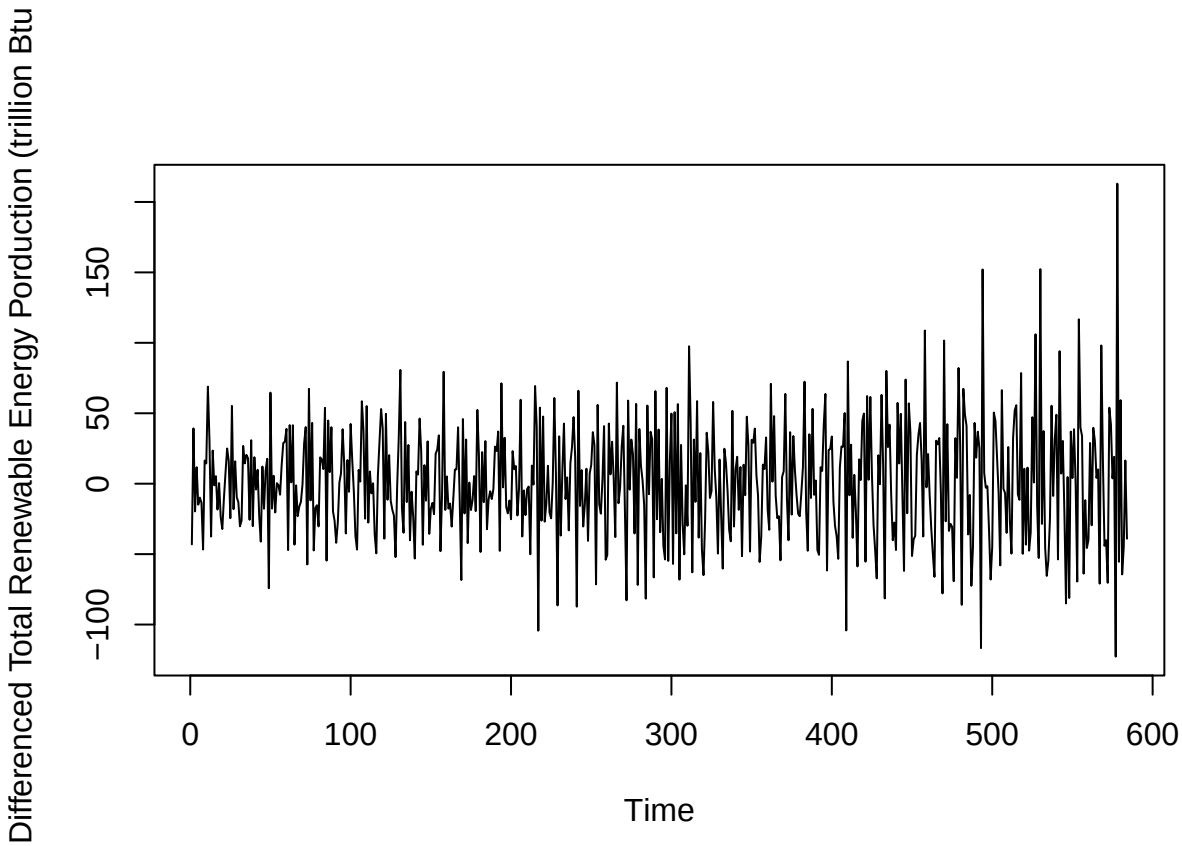
Difference the "Total Renewable Energy Production" series using function diff(). Function diff() is from package base and take three main arguments: * x vector containing values to be differenced; * *lag* integer indicating with lag to use; * *differences* integer indicating how many times series should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

```
sub_RE_data <- raw_RE_data[-c(1),4:6]
head(sub_RE_data)
```

```
## # A tibble: 6 x 3
##   `Total Biomass Energy Production` `Total Renewable Ener~ `Hydroelectric Power~
##   <chr>                             <chr>                   <chr>
## 1 129.787                           403.981                 272.703
## 2 117.338                           360.9                   242.199
## 3 129.938                           400.161                 268.81
## 4 125.636                           380.47                  253.185
## 5 129.834                           392.141                 260.77
## 6 125.611                           377.232                 249.859
```

```
RE_data <- cbind(raw_RE_data[-c(1),1],sub_RE_data[,])
RE_data$`Total Renewable Energy Production`<- as.numeric(RE_data$`Total Renewable Energy
 ↪  Production`)
diff_RE_1=diff(x=RE_data$`Total Renewable Energy Production`,lag=1,differences=1)
plot(diff_RE_1,type="l",ylab="Differenced Total Renewable Energy Porduction (trillion
 ↪  Btu)",xlab="Time")
```



The differenced series doesn't have trend (the mean is always around zero).

Q2

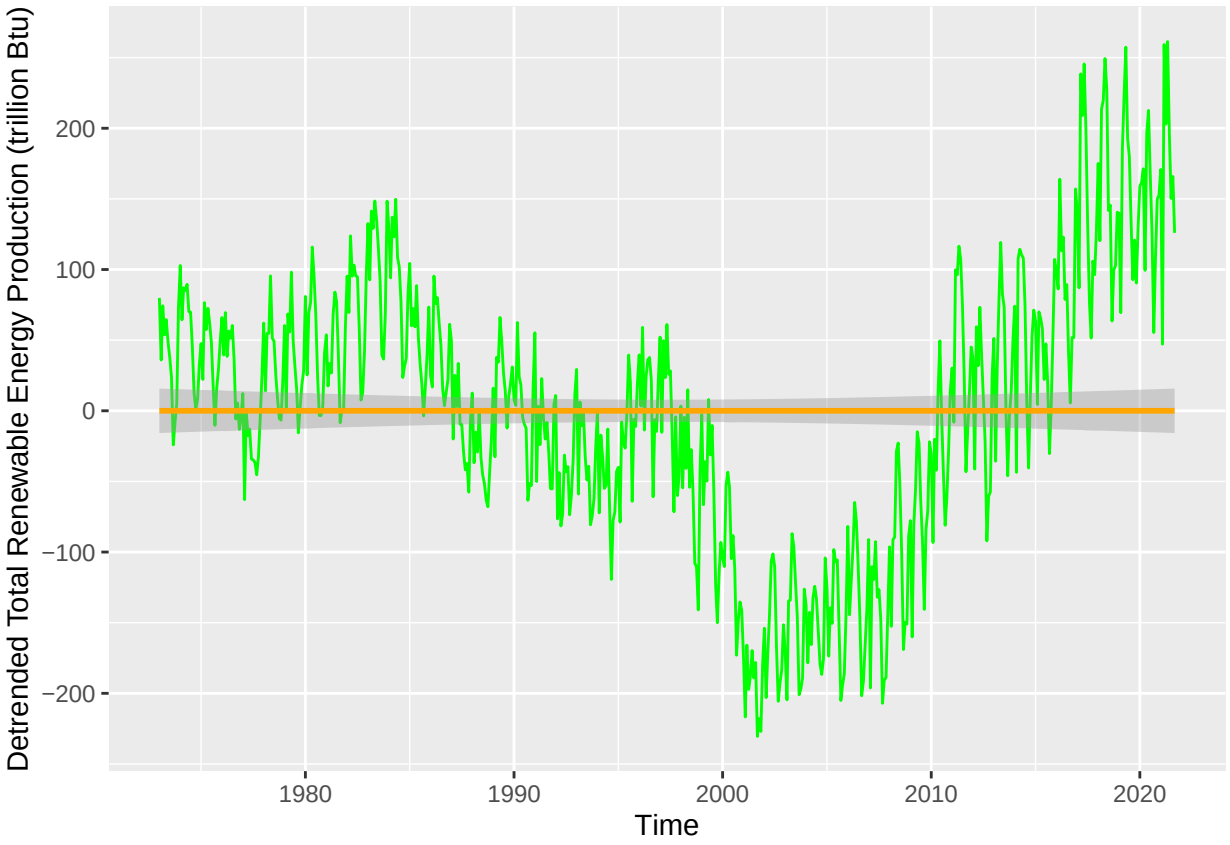
Now let's compare the differenced series with the detrended series you calculated on A3. In other words, for the "Total Renewable Energy Production" compare the differenced series from Q1 with the series you detrended in A3 using linear regression. (Hint: Just copy and paste part of your code for A3)

Copy and paste part of your code for A3 where you compute regression for Total Energy Production and the detrended Total Energy Production

```
nobs <- nrow(RE_data)
t <- c(1:nobs)
ts_RE_data <- ts(RE_data[,2:4], start=c(1973, 1), end=c(2021, 09), frequency=12)
lm2=lm(RE_data[,3]~t)
beta0_renew=as.numeric(lm2$coefficients[1])
beta1_renew=as.numeric(lm2$coefficients[2])

detrend_renew <- RE_data[,3]-(beta0_renew+beta1_renew*t)
 ggplot(RE_data, aes(x=Month, y=RE_data[,3])) +
            ylab("Detrended Total Renewable Energy Production (trillion Btu)")+
            xlab("Time")+
            geom_line(aes(y=detrend_renew), col="green")+
            geom_smooth(aes(y=detrend_renew),color="orange",method="lm")
```

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



Q3

Create a data frame with 4 columns: month, original series, detrended by Regression Series and differenced series. Make sure you properly name all columns. Also note that the differenced series will have only 584 rows because you loose the first observation when differencing. Therefore, you need to remove the first observations for the original series and the detrended by regression series to build the new data frame.

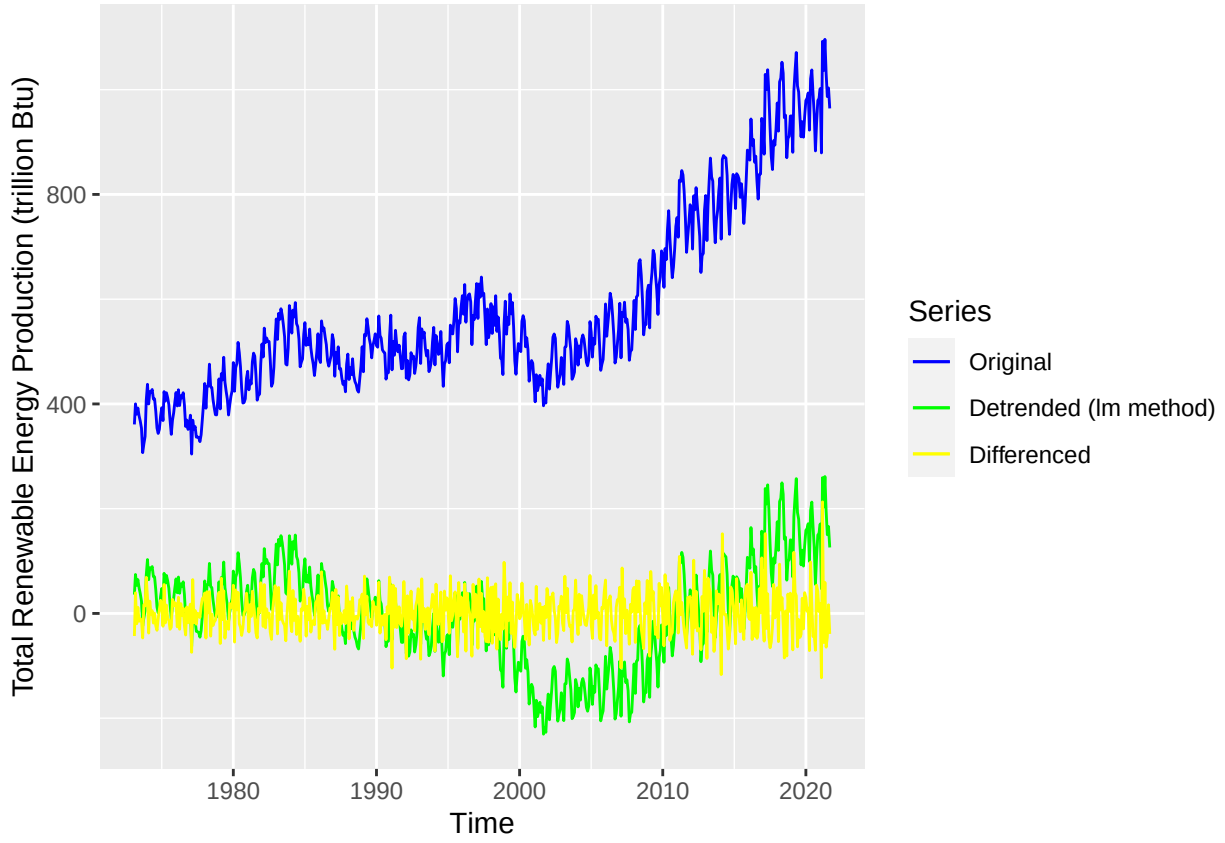
```
#Data frame - remember to note include January 1973
RE_data<-RE_data[-c(1),]
New_renew <-data.frame(Month=(RE_data$Month), Original=RE_data$`Total Renewable Energy
 ↪  Production`, Detrended=detrend_renew[-1],Differenced=diff_RE_1)
head(RE_data)
```

```
##        Month Total Biomass Energy Production Total Renewable Energy Production
## 2 1973-02-01                          117.338                             360.900
## 3 1973-03-01                          129.938                             400.161
## 4 1973-04-01                          125.636                             380.470
## 5 1973-05-01                          129.834                             392.141
## 6 1973-06-01                          125.611                             377.232
## 7 1973-07-01                          129.787                             367.325
##   Hydroelectric Power Consumption
## 2                         242.199
## 3                          268.81
## 4                         253.185
## 5                          260.77
## 6                         249.859
## 7                          235.67
```

Q4

Using ggplot() create a line plot that shows the three series together. Make sure you add a legend to the plot.

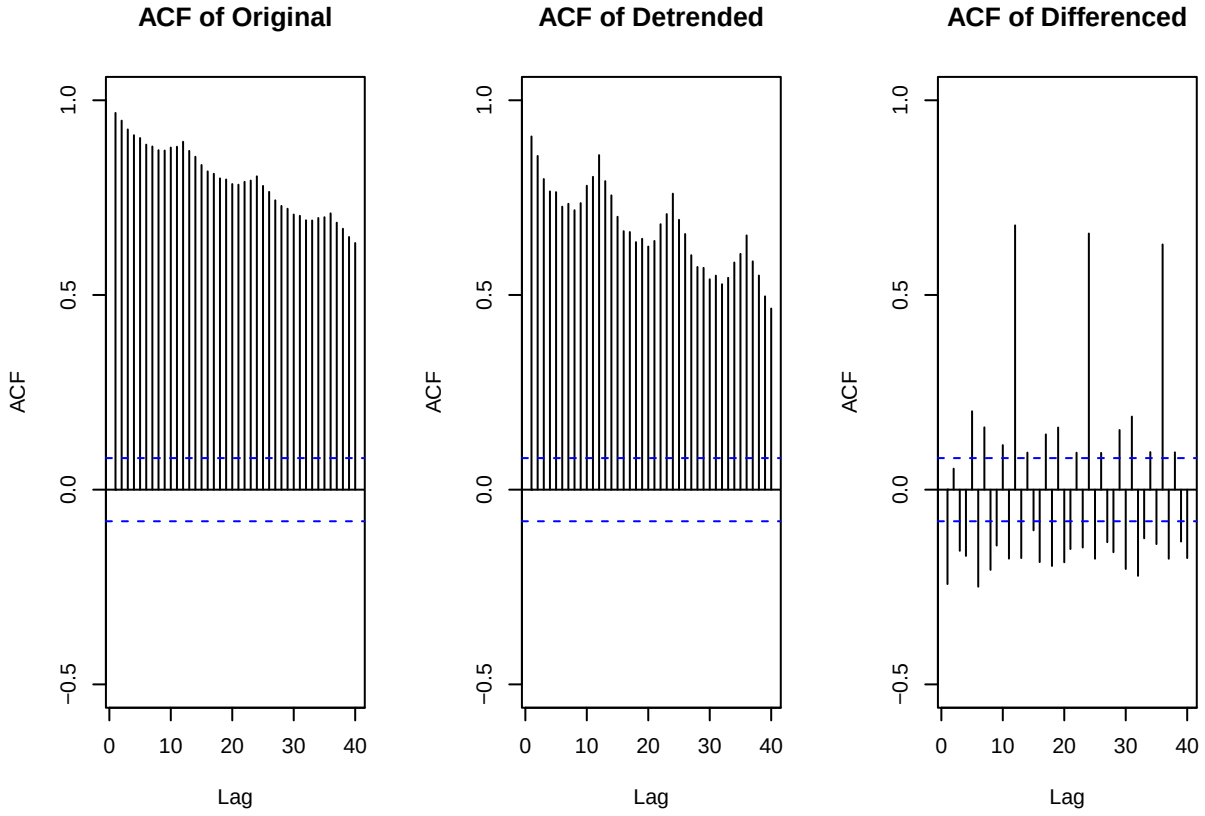
```
#Use ggplot
ggplot(New_renew, aes(x=Month, y=New_renew[,2],show.legend=TRUE)) +
            geom_line(color="blue") +
            ylab("Total Renewable Energy Production (trillion Btu)") +
            xlab("Time")+
            geom_line(aes(y=New_renew[,3],color="Detrended (lm method)"))+
            geom_line(aes(y=New_renew[,4],color="Differenced"))+
  scale_color_manual(name = "Series", breaks=c("Original","Detrended (lm
   ↪  method)","Differenced"),
                     values = c("Original" = "blue",
                                "Detrended (lm method)" = "green",
                                "Differenced" = "yellow"))
```



Q5

Plot the ACF for the three series and compare the plots. Add the argument `ylim=c(-0.5,1)` to the Acf() function 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?

```
#Compare ACFs
par(mfrow=c(1,3))
for(i in 2:4){
  Acf(New_renew[,i],lag.max=40,ylim=c(-0.5,1),main=paste("ACF of
    ↪  ",colnames(New_renew)[(i)],sep=""))
}
```



The differencing is the most efficient method to remove the trend, because in its ACF there is no decay pattern with the increase in lags, but only the significant seasonal pattern is left.

Q6

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. Whats 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.

```
SMKtest <- SeasonalMannKendall(ts_RE_data[,2])
print("Results for Seasonal Mann Kendall /n")
```

```
## [1] "Results for Seasonal Mann Kendall /n"
```

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

```
## Score =  9984 , Var(Score) = 159104
## denominator =  13968
## tau = 0.715, 2-sided pvalue =< 2.22e-16
## NULL
```

The seasonal Mann Kendall test indicates there is a trend ($p<0.05$).

```
print("Results for ADF test")
```

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

```
print(adf.test(ts_RE_data[,2],alternative = "stationary"))
```

```
##
##  Augmented Dickey-Fuller Test
##
## data:  ts_RE_data[, 2]
## Dickey-Fuller = -1.4383, Lag order = 8, p-value = 0.8161
## alternative hypothesis: stationary
```

The ADF test indicates that the series contain a unit root (p>0.05), so there is a stochastic trend in the total renewable energy production from 1973 to 2021. The stochastic trend indicated by the results matches what we observed in Q2 and Q3, where the plot shows that using linear regression coefficients to remove the trend is not an appropriate method, rather, differencing will eliminate the trend component better(less variation around zero). ### Q7

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.

```
ts_RE_data_new <- as.ts(ts_RE_data[1:576,2])
RE_data_matrix <- matrix(ts_RE_data_new,byrow=FALSE,nrow=12)
RE_data_yearly <- colMeans(RE_data_matrix)

library(dplyr)
```

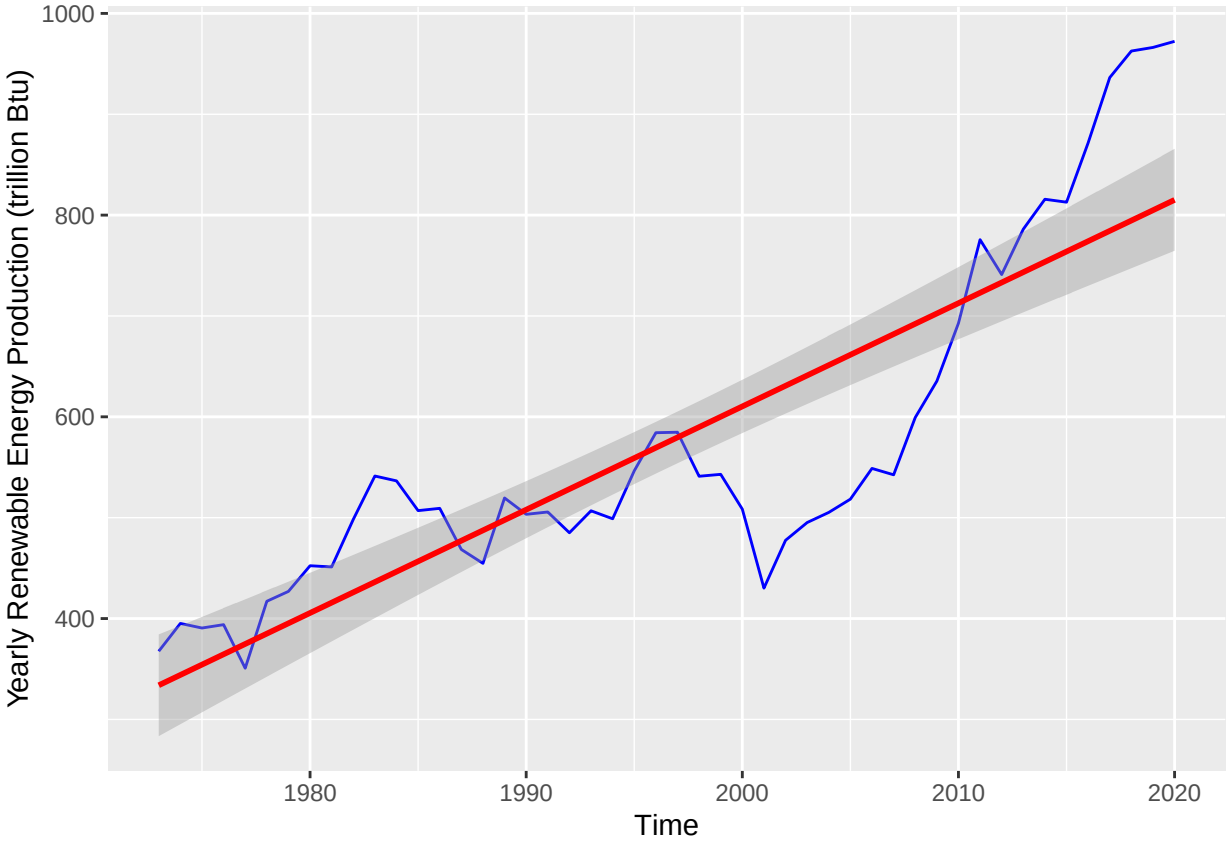
```
##
##    'dplyr'
```

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

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

```
my_year <- c(1973:2020)
RE_data_new_yearly <- data.frame(my_year, RE_data_yearly)
ggplot(RE_data_new_yearly, aes(x=my_year, y=RE_data_yearly)) +
            ylab("Yearly Renewable Energy Production (trillion Btu)") +
            xlab("Time")+
            geom_line(color="blue") +
            geom_smooth(color="red",method="lm")
```

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



Q8

Apply the Mann Kendal, Spearman correlation rank test and ADF. Are the results from the test in agreement with the test results for the non-aggregated series, i.e., results for Q6?

```
print("Results of Mann Kendall on average yearly series")
```

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

```
print(summary(MannKendall(RE_data_yearly)))
```

```
## Score =  816 , Var(Score) = 12658.67
## denominator =  1128
## tau = 0.723, 2-sided pvalue =< 2.22e-16
## NULL
```

The result of Mann Kendall test indicates that there is a trend in the time series ($p<0.05$).

```
print("Results from Spearman Correlation")
```

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

```
sp_rho=cor(RE_data_yearly,my_year,method="spearman")
print(sp_rho)
```

```
## [1] 0.8617021
```

```
sp_rho=cor.test(RE_data_yearly,my_year,method="spearman")
print(sp_rho)
```

```
##
##  Spearman's rank correlation rho
##
## data:  RE_data_yearly and my_year
## S = 2548, p-value < 2.2e-16
## alternative hypothesis: true rho is not equal to 0
## sample estimates:
##       rho
## 0.8617021
```

The Spearman correlation also indicates that there is a trend in the aggregated series ($p<0.05$).

```
print("Results for ADF test on yearly data")
```

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

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

```
##
##  Augmented Dickey-Fuller Test
##
## data:  RE_data_yearly
## Dickey-Fuller = -1.0426, Lag order = 3, p-value = 0.9219
## alternative hypothesis: stationary
```

The ADF test indicates that the yearly aggregated series contain a unit root (i.e. there is a stochastic trend)($p>0.05$).

The results from tests on the yearly-aggregated series are in agreement with the test results for the non-aggregated series.