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

Assignment 4 - Due date 02/12/24

Ina Liao

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.\

```
library(lubridate)
library(ggplot2)
library(forecast)
library(Kendall)
library(tseries)
library(here)
library(tidyr)
library(knitr)
library(kableExtra)
library(ggthemes)
library(cowplot)
library(dplyr)
library(base)
```

Questions

Consider the same data you used for A3 from the spreadsheet "Table_10.1_Renewable_Energy_Production_and_Consumption
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".

```
#check working directory
here()

#import data
```

```r
raw_energy<-read.csv(here("Data/Table_10.1_Renewable_Energy_Production_and_Consumption_by_Source.csv"),s
raw_energy

#transform date format
Date<-ym(raw_energy$Month)
raw_energy<-cbind(Date,raw_energy[,2:14])
raw_energy
```

```r
df_energy<-raw_energy[,c(1,5)]
head(df_energy)
```

```
##         Date Total.Renewable.Energy.Production
## 1 1973-01-01                            219.839
## 2 1973-02-01                            197.330
## 3 1973-03-01                            218.686
## 4 1973-04-01                            209.330
## 5 1973-05-01                            215.982
## 6 1973-06-01                            208.249
```

```r
#rename column names
new_names<-c("Date","Renewable Production")
colnames(df_energy)<-new_names
head(df_energy)
```

```
##         Date Renewable Production
## 1 1973-01-01                219.839
## 2 1973-02-01                197.330
## 3 1973-03-01                218.686
## 4 1973-04-01                209.330
## 5 1973-05-01                215.982
## 6 1973-06-01                208.249
```

```r
#find the start date
year1<-year(df_energy$Date[1])
month1<-month(df_energy$Date[1])

#time series object
ts_energy<-ts(df_energy,start=c(year1,month1),frequency=12)
```

```r
my_plot_theme<- theme(
    #plot title
    plot.title=element_text(color="black",hjust=0.5,vjust=1),

    #axis labels
    axis.title.x = element_text(size = 10),
    axis.title.y = element_text(size = 10,angle = 90, vjust = 0.5, hjust = 0.5),

    #legend
    legend.text = element_text(size = 10),
    legend.position="right"
      )
theme_set(my_plot_theme)
```

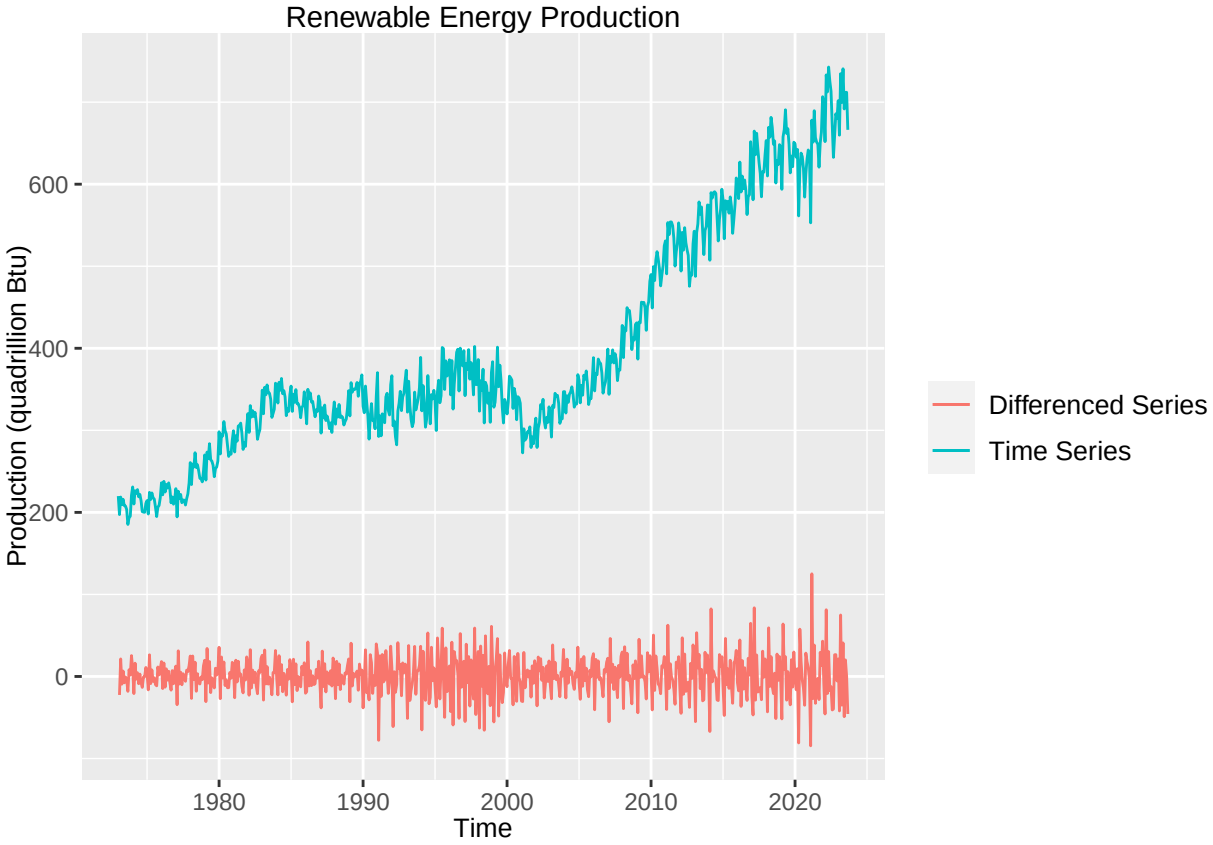
Stochastic Trend and Stationarity Tests

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?

```
ts_energy_diff<-diff(ts_energy[,2],lag = 1,differences = 1)

plot_time_diff <- autoplot(ts_energy[,2],series="Time Series")+
  autolayer(ts_energy_diff, series="Differenced Series")+
  labs(x="Time",y="Production (quadrillion Btu)",title="Renewable Energy Production",color="")
plot_time_diff
```



From the time series plot, it seems that the differenced time series does not have a trend.

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.

```
#create a time vector
num_row<-nrow(df_energy)
t<-c(1:num_row)

#create a new data frame
```

```r
df_renew<-data.frame("time"=t,"renewable"=df_energy$`Renewable Production`)

#run linear regression
linear_renew=lm(renewable~t,df_renew)
summary(linear_renew)
```

```
##
## Call:
## lm(formula = renewable ~ t, data = df_renew)
##
## 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
```

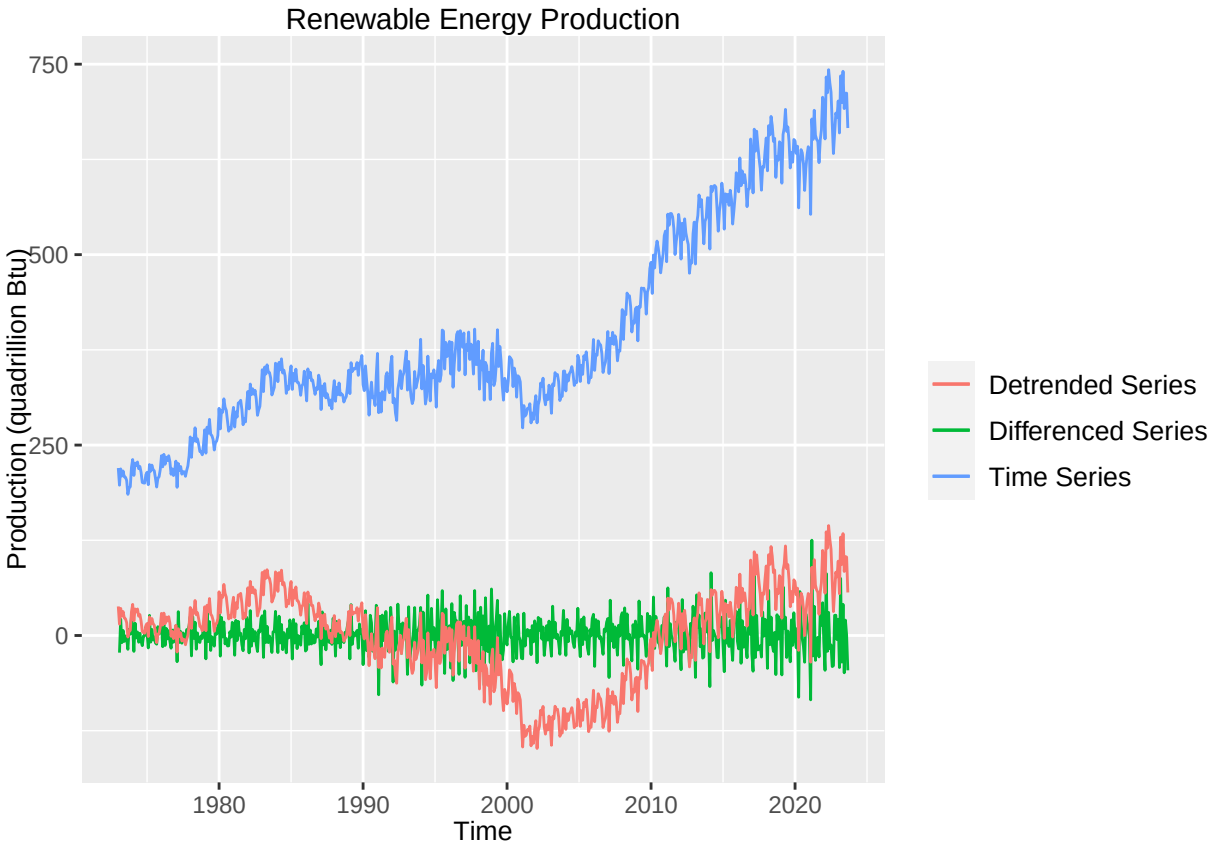
```r
#store the intercept and slope
intercept_renew=as.numeric(linear_renew$coefficients[1])
slope_renew=as.numeric(linear_renew$coefficients[2])

#create the detrended time series from linear trend
detrend_renew<-df_energy$`Renewable Production`-
  (slope_renew*df_renew$time+intercept_renew)

#create detrended time series object
ts_detrend_renew<-ts(detrend_renew,start=c(year1,month1),frequency=12)
```

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
plot_all<-autoplot(ts_energy[,2],series="Time Series")+
  autolayer(ts_energy_diff,series="Differenced Series")+
  autolayer(ts_detrend_renew,series="Detrended Series")+
  labs(x="Time",y="Production (quadrillion Btu)",title="Renewable Energy Production",color="")
plot_all
```



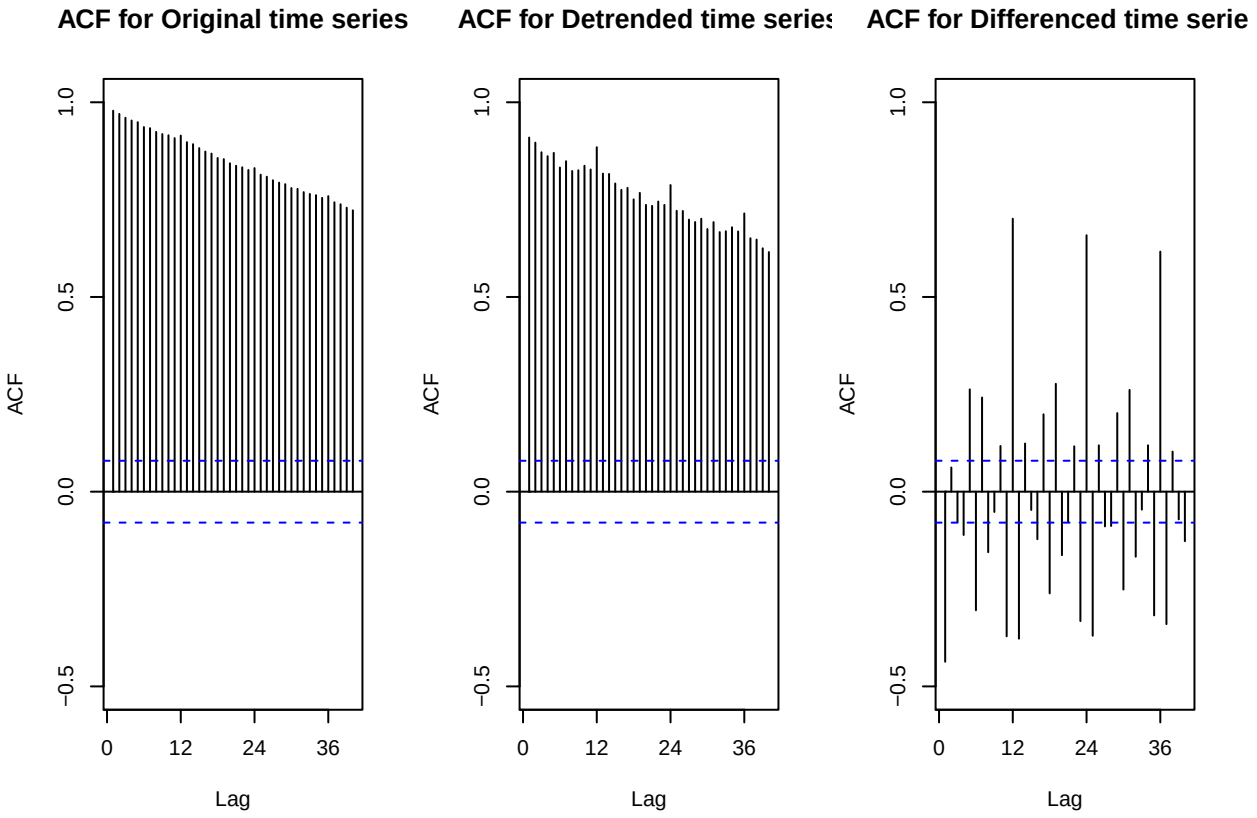
Q4

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?

```
#place the plot side by side
par(mfrow=c(1,3))
df_all<-cbind(ts_energy[,2],ts_detrend_renew,ts_energy_diff)

#rename column names
new_names<-c("Original time series","Detrended time series","Differenced time series")
colnames(df_all)<-new_names
head(df_all)
```

```
##          Original time series Detrended time series Differenced time series
## Jan 1973              219.839              38.14556                      NA
## Feb 1973              197.330              14.93252                 -22.509
## Mar 1973              218.686              35.58448                  21.356
## Apr 1973              209.330              25.52444                  -9.356
## May 1973              215.982              31.47240                   6.652
## Jun 1973              208.249              23.03536                  -7.733
```

```
for(i in 1:3){
  Acf(df_all[,i],lag.max=40,main=paste("ACF for",new_names[i]),
      ylim=c(-0.5,1))
}
```




```
#df_all[i] will not print the column names
```

The differencing method appears to be a more effective way of removing the trend. (1) After using linear regression to detrend, the autocorrelation remains high, which indicates that the trend may not be linear and still exists in the detrended time series. (2) After taking the difference, we can see that there are several spikes in the differenced time series. With this information, we can further identify months with high autocorrelation.

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.

```
#Seasonal Mann-Kendall
SMK_original_ts<-SeasonalMannKendall(ts_energy[,2])
print(summary(SMK_original_ts))
```

```
## Score =  11865 , Var(Score) = 179299
## denominator =  15149.5
## tau = 0.783, 2-sided pvalue =< 2.22e-16
## NULL
```

```
#ADF test
ADF_original_ts<-adf.test(ts_energy[,2],alternative = "stationary")
print(ADF_original_ts)
```

```
##
```

```
##  Augmented Dickey-Fuller Test
##
## data:  ts_energy[, 2]
## Dickey-Fuller = -1.24, Lag order = 8, p-value = 0.9
## alternative hypothesis: stationary
```

(1) We have sufficient evidence to reject the null hypothesis that renewable energy is stationary (p-value $<$ 0.05). The renewable energy production has a strong time dependence (tau = 0.783) and might have an increasing trend over time (S score $>$ 0).
(2) We do not have sufficient evidence to reject the null hypothesis that there is a unit root in renewable energy production time series (p-value $>$ 0.05). The result from the Augmented Dickey-Fuller Test suggests that there might be a stochastic trend in the renewable energy production.
(3) To sum up, there is a non-linear trend in renewable energy production, consistent with previous observations.

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().

```
#store series in a matrix
#row: month, column: year
energy_matrix<-matrix(ts_energy[,2],byrow=FALSE,nrow=12)
```

```
## Warning in matrix(ts_energy[, 2], byrow = FALSE, nrow = 12): data length [609]
## is not a sub-multiple or multiple of the number of rows [12]
```

```
#calculate production annual mean
energy_yearly<-colMeans(energy_matrix)

#create another column for year
date_new<-ymd(df_energy$Date)
year<-c(year(first(date_new)):year(last(date_new)))

#merge as a dataframe
df_energy_yearly<-data.frame(year, energy_yearly)
head(df_energy_yearly)
```

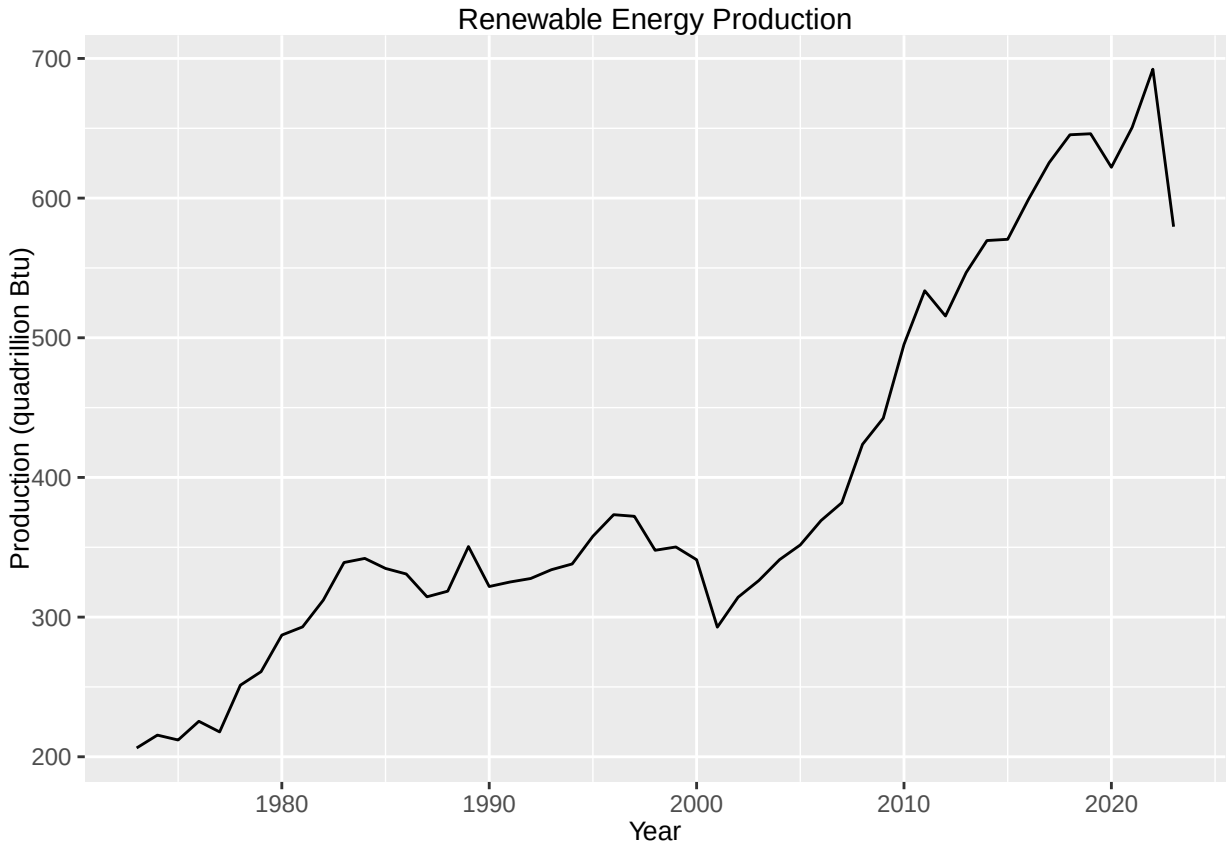
```
##   year energy_yearly
## 1 1973      206.2953
## 2 1974      215.5001
## 3 1975      212.0139
## 4 1976      225.3914
## 5 1977      217.7895
## 6 1978      251.2457
```

```
#create time series object
ts_energy_yearly<-ts(df_energy_yearly[,2],start=1973,end=2023,frequency=1)
head(ts_energy_yearly)
```

```
## Time Series:
## Start = 1973
## End = 1978
## Frequency = 1
```

```
## [1] 206.2953 215.5001 212.0139 225.3914 217.7895 251.2457
```

```
#plot
plot_yearly<-autoplot(ts_energy_yearly)+
labs(x="Year",y="Production (quadrillion Btu)",title="Renewable Energy Production",color="")
plot_yearly
```



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?

```
#Mann-Kendall
MK_yearly_ts<-MannKendall(ts_energy_yearly)
print(summary(MK_yearly_ts))
```

```
## Score =  1019 , Var(Score) = 15158.33
## denominator =  1275
## tau = 0.799, 2-sided pvalue =< 2.22e-16
## NULL
```

```
#Spearman correlation rank test
Spearman_yearly_ts<-cor.test(ts_energy_yearly,year,method="spearman")
print(Spearman_yearly_ts)
```

```
##
##  Spearman's rank correlation rho
##
## data:  ts_energy_yearly and year
```

```
## S = 1908, p-value < 2.2e-16
## alternative hypothesis: true rho is not equal to 0
## sample estimates:
##       rho
## 0.9136652
```

```
#ADF test
ADF_yearly_ts<-adf.test(ts_energy_yearly,alternative = "stationary")
print(ADF_yearly_ts)
```

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

(1) We have sufficient evidence to reject the null hypothesis that renewable energy is stationary (p-value $<$ 0.05). The renewable energy production has a strong time dependence (tau = 0.799) and might have an increasing trend over time (S score $>$ 0).
(2) We have sufficient evidence to reject the null hypothesis that renewable energy is stationary (p-value $<$ 0.05).The renewable energy production has a strong time dependence (rho = 0.913).
(3) We do not have sufficient evidence to reject the null hypothesis that there is a unit root in renewable energy production time series (p-value $>$ 0.05). The result from the Augmented Dickey-Fuller Test suggests that there might be a stochastic trend in the renewable energy production. The test results align with the test results from the previous questions.