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

Assignment 4 - Due date 02/17/22

Student Name

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)
library(ggplot2)
library(forecast)
```

```
## Warning: package 'forecast' was built under R version 4.1.2
```

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

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

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".

```
#Importing data set - using xlsx package
tab <- read_excel("/Users/emreyurtbay/Documents/Duke/env790/ENV790_TimeSeriesAnalysis_Sp2022/Data/Table_

# remove the first row
tab = tab[-1,]
df = tab[,c(1, 5)]
```

```
df$`Total Renewable Energy Production` = as.numeric(df$`Total Renewable Energy Production`)
ts_df <- ts(df[, 2], frequency = 12, start = c(1973, 1))
```

Stochastic Trend and Stationarity Tests

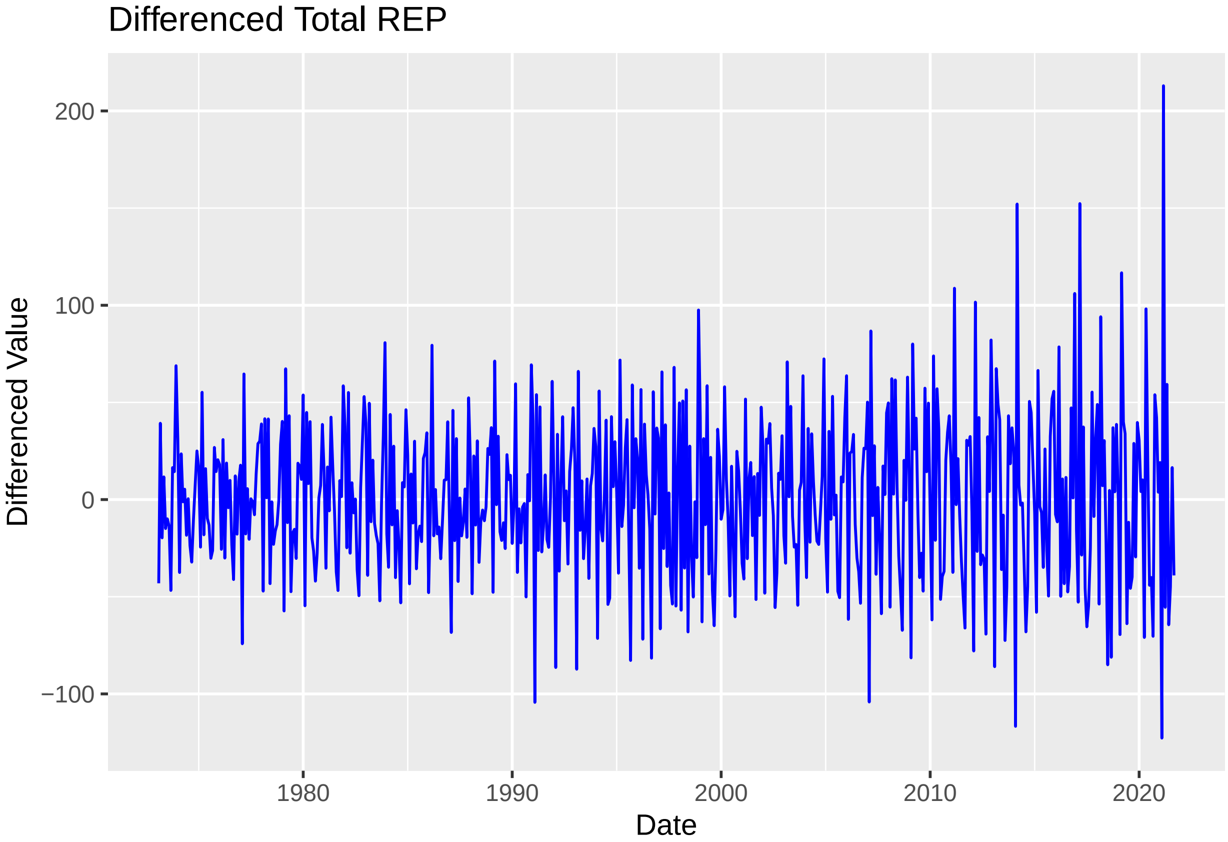
Q1

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. Does the series still seem to have trend?

```
diffed <- diff(df$`Total Renewable Energy Production`, lag = 1, differences = 1)
```

```
ggplot(data.frame(Date = df$Month[-1], Value = diffed),
       aes(x = Date, y = Value))+
  geom_line(color="blue") +
  labs(
    x = "Date", y = "Differenced Value",
    title = "Differenced Total REP"
  )
```



This differencing has seemed to remove the trend; the values oscillate around 0, without any discernable systematic increase or decrease.

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

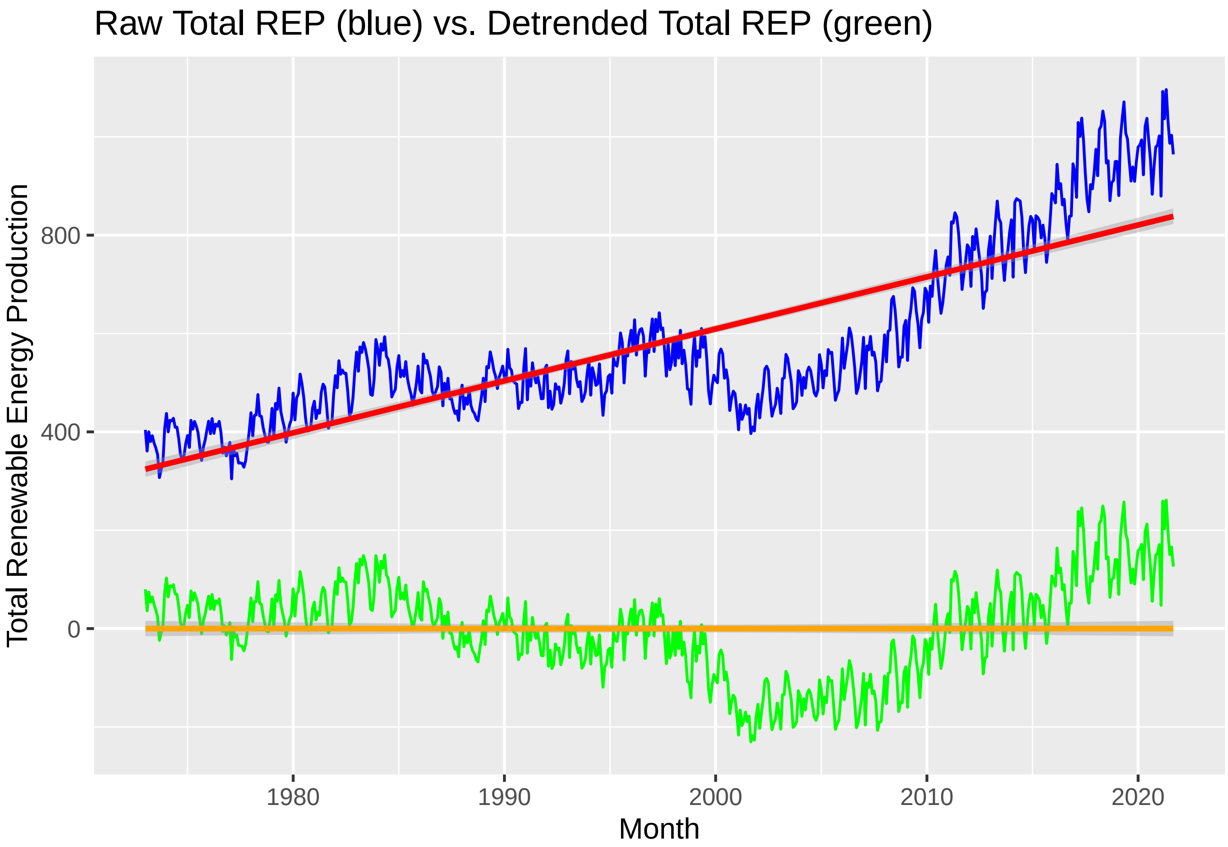
```
nobs <- nrow(df)
t <- c(1:nobs)

reg_r=lm(df$`Total Renewable Energy Production`~t)
#summary(reg_r)

beta0_r=as.numeric(reg_r$coefficients[1])
beta1_r=as.numeric(reg_r$coefficients[2])
#remove the trend from series
detrend_r <- df$`Total Renewable Energy Production`-(beta0_r+beta1_r*t)

#Understanding what we did
ggplot(df, aes(x=Month, y=`Total Renewable Energy Production`)) +
            geom_line(color="blue") +
            geom_smooth(color="red",method="lm") +
            geom_line(aes(y=detrend_r), col="green")+
            geom_smooth(aes(y=detrend_r),color="orange",method="lm")+
  ggtitle("Raw Total REP (blue) vs. Detrended Total REP (green)")
```

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



The differenced series and the detrended series look different. For the differenced series, the values oscillate around 0, without any discernable systematic increase or decrease. For the detrended series, there is more drift and "stickiness", as it were, in our series.

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.

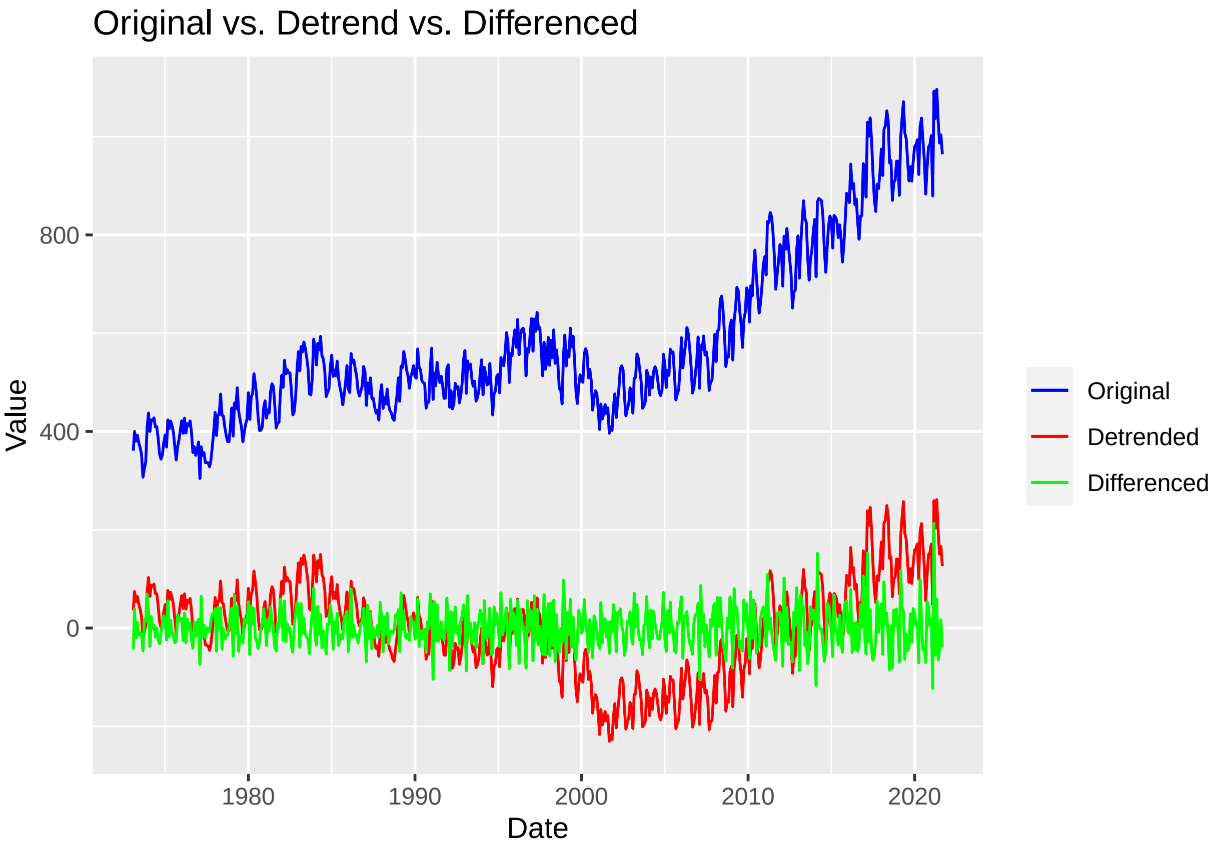
```
#Data frame - remember to not include January 1973
new_df <- data.frame(Month = df$Month[-1],
           Original = df$`Total Renewable Energy Production`[-1],
           Detrended = detrend_r[-1],
           Differenced = diffed)
head(new_df)
```

```
##        Month Original Detrended Differenced
## 1 1973-02-01  360.900  35.95655     -43.081
## 2 1973-03-01  400.161  74.33705      39.261
## 3 1973-04-01  380.470  53.76554     -19.691
## 4 1973-05-01  392.141  64.55603      11.671
## 5 1973-06-01  377.232  48.76653     -14.909
## 6 1973-07-01  367.325  37.97902      -9.907
```

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_df) +
  geom_line(aes(x = Month, y = Original, color = "Original")) +
  geom_line(aes(x = Month, y = Detrended, color = "Detrended")) +
  geom_line(aes(x = Month, y = Differenced, color = "Differenced")) +
  scale_color_manual("",
                     breaks = c("Original", "Detrended", "Differenced"),
                     values = c("blue", "red", "green")) +
  labs(
    x = "Date", y = "Value",
    title = "Original vs. Detrend vs. Differenced"
  )
```

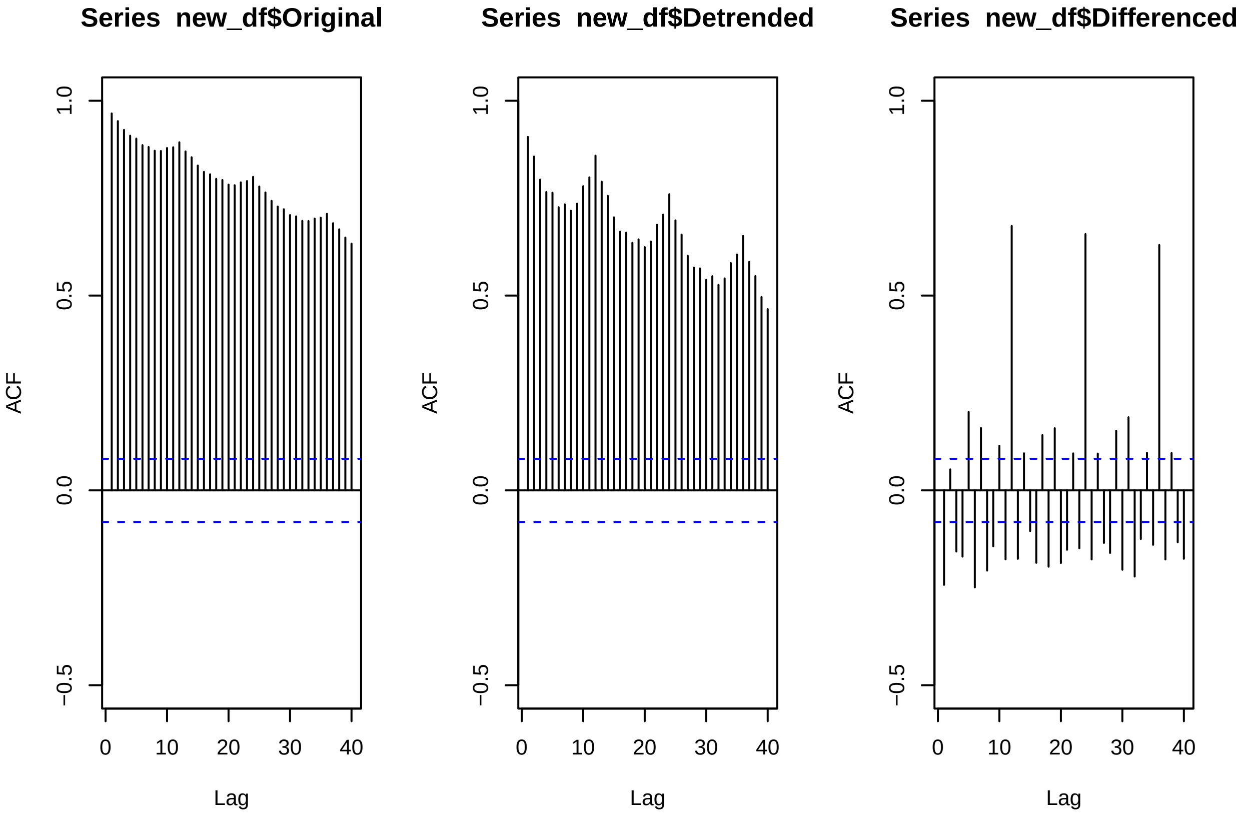


Q5

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?

```
par(mfrow = c(1, 3))

#Compare ACFs
Acf(new_df$Original,lag.max=40, type="correlation", plot=TRUE, ylim=c(-0.5,1))
Acf(new_df$Detrended,lag.max=40, type="correlation", plot=TRUE, ylim=c(-0.5,1))
Acf(new_df$Differenced,lag.max=40, type="correlation", plot=TRUE, ylim=c(-0.5,1))
```



Based on the ACF plots, differencing does a better job than linear regression of eliminating the trend. The ACF plot after differencing does not show the tell-tale signs of trend that even the detrended series still seems to show - we see still see strong autocorrelation for large lags in the regression detrended series.

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(df$`Total Renewable Energy Production`))
print("Results for Seasonal Mann Kendall /n")
```

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

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

```
## Score =  113424 , Var(Score) = 22301500
## denominator =  170820
## tau = 0.664, 2-sided pvalue =< 2.22e-16
## NULL
```

For the Mann-Kendal Test, our null hypothesis is that our data are stationary. Using an alpha level of 0.05, we can reject the null hypothesis with a p-value of near 0. Thus we have evidence that the Total REP series has a deterministic trend.

```r
#Null hypothesis is that data has a unit root
print("Results for ADF test/n")
```

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

```r
print(adf.test(ts(df$`Total Renewable Energy Production`),
               alternative = "stationary"))
```

```
##
##  Augmented Dickey-Fuller Test
##
## data:  ts(df$`Total Renewable Energy Production`)
## Dickey-Fuller = -1.4383, Lag order = 8, p-value = 0.8161
## alternative hypothesis: stationary
```

For the ADF test, our null hypothesis is that the data has a unit root. Using an alpha level of 0.05, we fail to reject the null hypothesis with a p-value of 0.82. Therefore, we conclude that our data are not stationary and that the Total REP series has a stochastic trend.

Both tests determine that the Total REP series has a trend.

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.

```r
#Group data in yearly steps instances
rep_data_matrix <- matrix(df$`Total Renewable Energy Production`,
                          byrow=FALSE,nrow=12)
```

```
## Warning in matrix(df$`Total Renewable Energy Production`, byrow = FALSE, : data
## length [585] is not a sub-multiple or multiple of the number of rows [12]
```

```r
rep_data_yearly <- colMeans(rep_data_matrix)
rep_data_yearly
```

```
##  [1] 367.5781 395.1543 390.5934 393.9292 350.7473 417.1201 426.9045 452.3618
##  [9] 451.1407 498.3031 541.3011 536.4885 507.0013 509.2615 468.4839 454.7295
## [17] 519.5548 503.3353 505.6488 485.0463 506.8257 498.9286 546.4422 584.2408
## [25] 584.7342 541.0612 542.9657 508.4722 430.1476 477.5752 495.2055 505.2226
## [33] 518.4010 548.8537 542.5307 599.2957 635.4112 692.8135 775.6386 741.0728
## [41] 786.0602 815.7308 812.8228 871.6138 936.3855 962.6813 966.2615 972.2888
## [49] 854.7981
```

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?

```r
SMKtest2 <- SeasonalMannKendall(ts(rep_data_yearly))
print("Results for Seasonal Mann Kendall /n")
```

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

```
print(summary(SMKtest2))
```

```
## Score =  854 , Var(Score) = 13458.67
## denominator =  1176
## tau = 0.726, 2-sided pvalue =1.8208e-13
## NULL
```

```
years <- c(1973:2021)
sp_rho=cor.test(ts(rep_data_yearly),years,method="spearman")
print(sp_rho)
```

```
##
##  Spearman's rank correlation rho
##
## data:  ts(rep_data_yearly) and years
## S = 2578, p-value < 2.2e-16
## alternative hypothesis: true rho is not equal to 0
## sample estimates:
##       rho
## 0.8684694
```

```
#Null hypothesis is that data has a unit root
print("Results for ADF test/n")
```

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

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

```
##
##  Augmented Dickey-Fuller Test
##
## data:  ts(rep_data_yearly)
## Dickey-Fuller = -2.2085, Lag order = 3, p-value = 0.4907
## alternative hypothesis: stationary
```

The Seasonal Mann-Kendall test results in a p-value near 0, so we reach the same conclusion as we did in question 6; there is evidence of a deterministic trend. For the spearman correlation test, our null hypothesis is that our data is stationary. Using an alpha level of 0.05, we can reject the null hypothesis with a p-value of near 0. Thus we have evidence that the Total REP series has a deterministic trend based on the correlation test. For the ADF test, using an alpha level of 0.05, we fail to reject the null hypothesis with a p-value of 0.49 . Therefore, we conclude that our data are not stationary and that the Total REP series has a stochastic trend, which is the same result we had in question 6.