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

Assignment 2 - Due date 02/25/24

Vincient Whatley

## Submission Instructions

You should open the .rmd file corresponding to this assignment on RStudio. The file is available on our class repository on Github.

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_A02_Sp24.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

R packages needed for this assignment:"forecast","tseries", and "dplyr". 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(forecast)
```

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

```
library(tseries)
library(dplyr)
```

```
##
## 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
```

## Data set information

Consider the data provided in the spreadsheet "Table_10.1_Renewable_Energy_Production_and_Consumption_by_Source on our **Data** folder. The data comes from the US Energy Information and Administration and corresponds to the December 2023 Monthly Energy Review. The spreadsheet is ready to be used. You will also find a *.csv* version of the data "Table_10.1_Renewable_Energy_Production_and_Consumption_by_Source-Edit.csv". You may use the function *read.table*() to import the *.csv* data in R. Or refer to the file "M2_ImportingData_CSV_XLSX.Rmd" in our Lessons folder for functions that are better suited for importing the *.xlsx*.

```
#Importing data set
getwd()
```

```
## [1] "/Users/vincient/Desktop/Projects/ENV 797"
```

```
energy_data_raw <- read.csv(file="./Data/Table_10.1_Renewable_Energy_Production_and_Consumption_by_Sour
```

## Question 1

You will work only with the following columns: Total Biomass Energy Production, Total Renewable Energy Production, Hydroelectric Power Consumption. Create a data frame structure with these three time series only. Use the command head() to verify your data.

```
#Triming data for specified colmunis
energydata <- energy_data_raw[,1:6]

#Removing unnecessary columns, renaming columns
energydata <- energydata[, -c(2, 3)]
colnames(energydata)=c("Month", "TBEP", "TREP","HEPC")

#convert month day year

mdy_date <- paste(energydata[,1]," 01",sep="")
mdy_date1 <- as.Date(mdy_date, format = "%Y %B %d")  #function my from package lubridate
head(mdy_date1)
```

```
## [1] "1973-01-01" "1973-02-01" "1973-03-01" "1973-04-01" "1973-05-01"
## [6] "1973-06-01"
```

```
#add that to energydata
energydata <- cbind(mdy_date1,energydata[,2:4]) #cbind stands for column bind
head(energydata)
```

```
##    mdy_date1    TBEP    TREP   HEPC
## 1 1973-01-01 129.787 219.839 89.562
## 2 1973-02-01 117.338 197.330 79.544
## 3 1973-03-01 129.938 218.686 88.284
## 4 1973-04-01 125.636 209.330 83.152
## 5 1973-05-01 129.834 215.982 85.643
## 6 1973-06-01 125.611 208.249 82.060
```

## Question 2

Transform your data frame in a time series object and specify the starting point and frequency of the time series using the function ts().

```
ts_TBEP_data <- ts(energydata [,"TBEP"], start= c(1973,1), frequency = 12)
ts_TREP_data <- ts(energydata [,"TREP"], start= c(1973,1), frequency = 12)
ts_HEPC_data <- ts(energydata [,"HEPC"], start= c(1973,1), frequency = 12)

##combined_ts <- cbind(ts_TBEP_data, ts_TREP_data, ts_HEPC_data)
```

## Question 3

Compute mean and standard deviation for these three series.

```
#Finding the mean and SD of Total Biomass Energy Production
mean(ts_TBEP_data)
```

```
## [1] 279.8046
```

```
sd(ts_TBEP_data)
```

```
## [1] 92.66504
```

```
#Finding the mean and SD of Total Renwable Energy Production
mean(ts_TREP_data)
```

```
## [1] 395.7213
```

```
sd(ts_TREP_data)
```

```
## [1] 137.7952
```

```
#Finding the mean and SD of Hydro Eletric Power Consumption
mean(ts_HEPC_data)
```

```
## [1] 79.73071
```

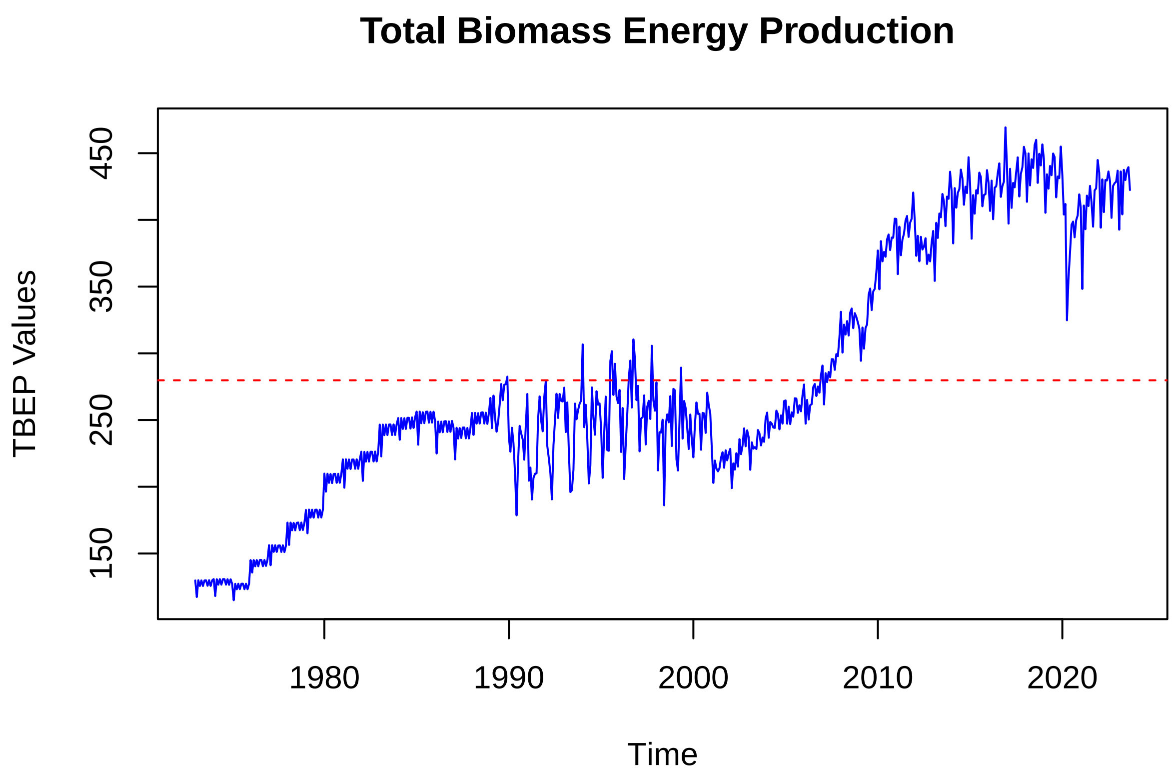
```
sd(ts_HEPC_data)
```

```
## [1] 14.14734
```

## Question 4

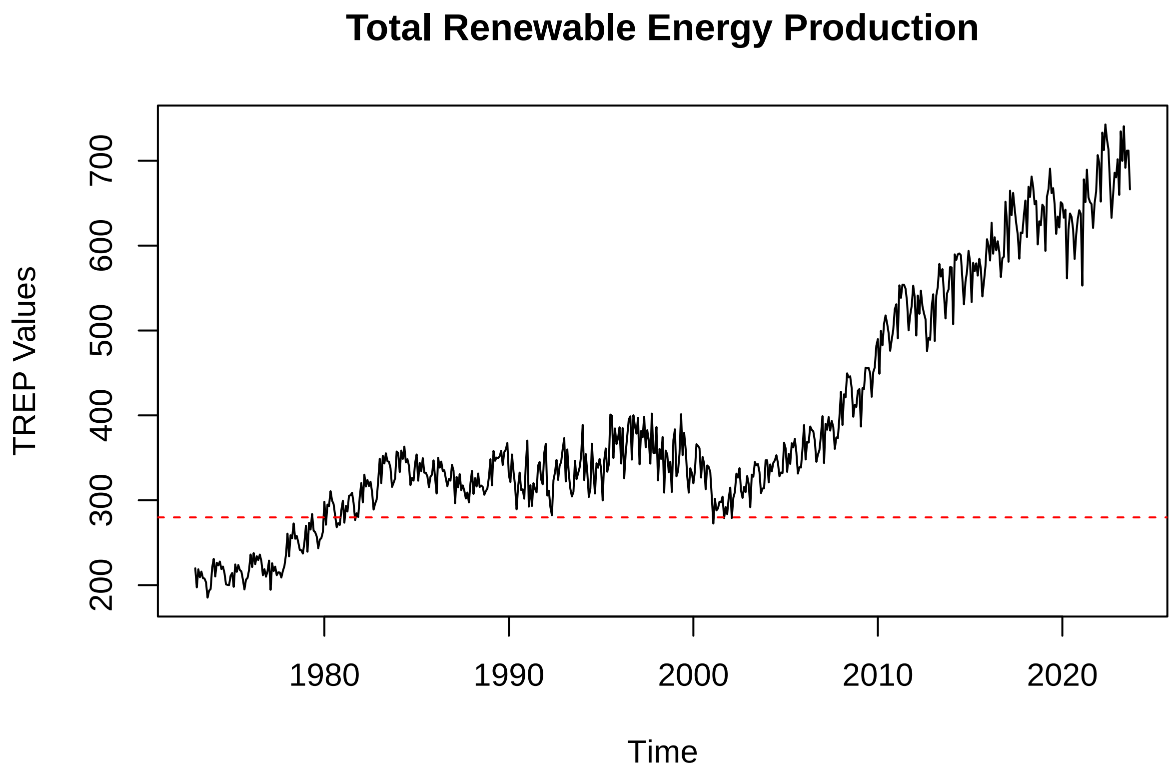
Display and interpret the time series plot for each of these variables. Try to make your plot as informative as possible by writing titles, labels, etc. For each plot add a horizontal line at the mean of each series in a different color.

```
# Plot for TBEP
plot(ts_TBEP_data, main ="Total Biomass Energy Production ", ylab = "TBEP Values", col = "blue")+
  abline(h = mean(ts_TBEP_data), col = "red", lty = 2)
```



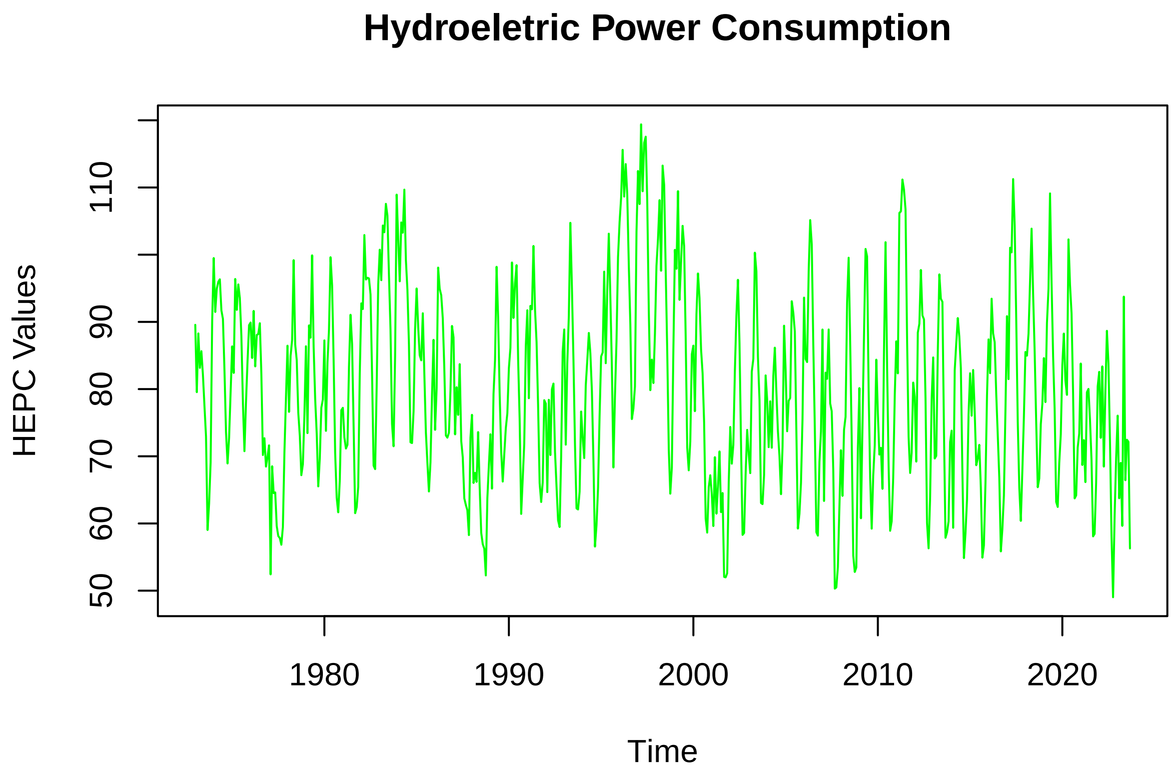

```
## integer(0)
```

```
# Plot for TREP
plot(ts_TREP_data, main = "Total Renewable Energy Production", ylab = "TREP Values", col = "Black")+
  abline(h = mean(ts_TBEP_data), col = "red", lty = 2)
```




```
## integer(0)
```

```r
# Plot for HEPC
plot(ts_HEPC_data, main = "Hydroeletric Power Consumption ", ylab = "HEPC Values", col = "Green")+
  abline(h = mean(ts_TBEP_data), col = "red", lty = 2)
```




```
## integer(0)
```

### Question 5

Compute the correlation between these three series. Are they significantly correlated? Explain your answer.

```
# Using cbind function I tool the  ts_TBEP_data, ts_TREP_data, and
#ts_HEPC_data time series objects and created a matrix to look at all
#correlation without doing each one separately.

correlation_matrix <- cor(cbind(ts_TBEP_data, ts_TREP_data, ts_HEPC_data))
correlation_matrix
```

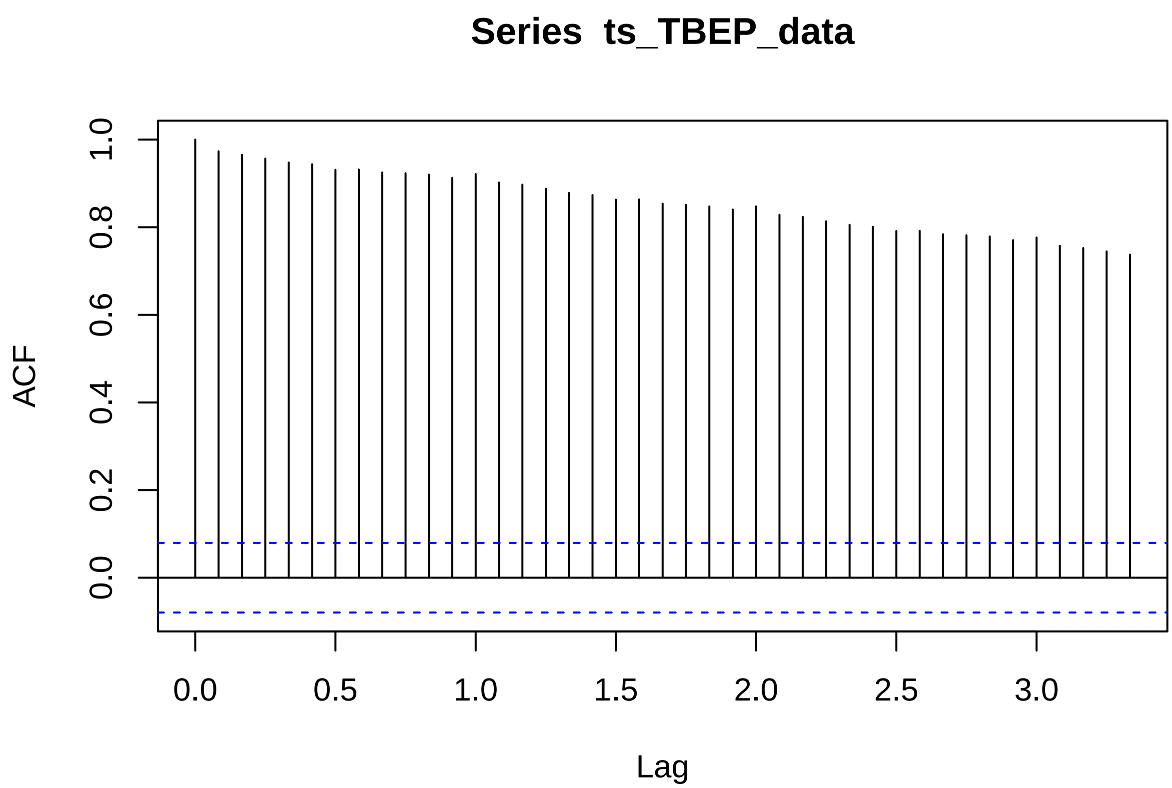
```
##              ts_TBEP_data ts_TREP_data ts_HEPC_data
## ts_TBEP_data   1.00000000  0.970746212 -0.096563177
## ts_TREP_data   0.97074621  1.000000000 -0.001768629
## ts_HEPC_data  -0.09656318 -0.001768629  1.000000000
```

```
#From this we are able to figure out a couple of things, that the
#correlation(s) between the three series are all significant
#except for the relationship between ts_TREP_data & ts_HEPC_data.
#Also vice versa between ts_HEPC_data & ts_TREP_data.
```

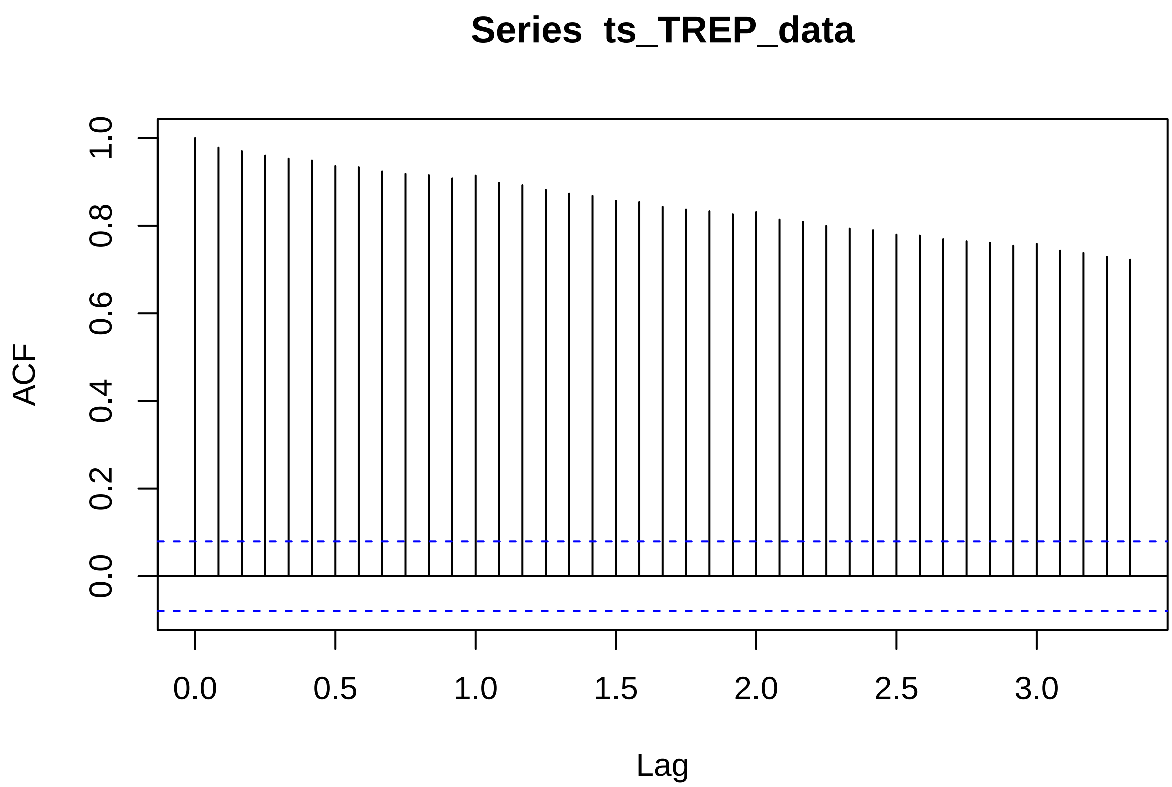
### Question 6

Compute the autocorrelation function from lag 1 up to lag 40 for these three variables. What can you say about these plots? Do the three of them have the same behavior?

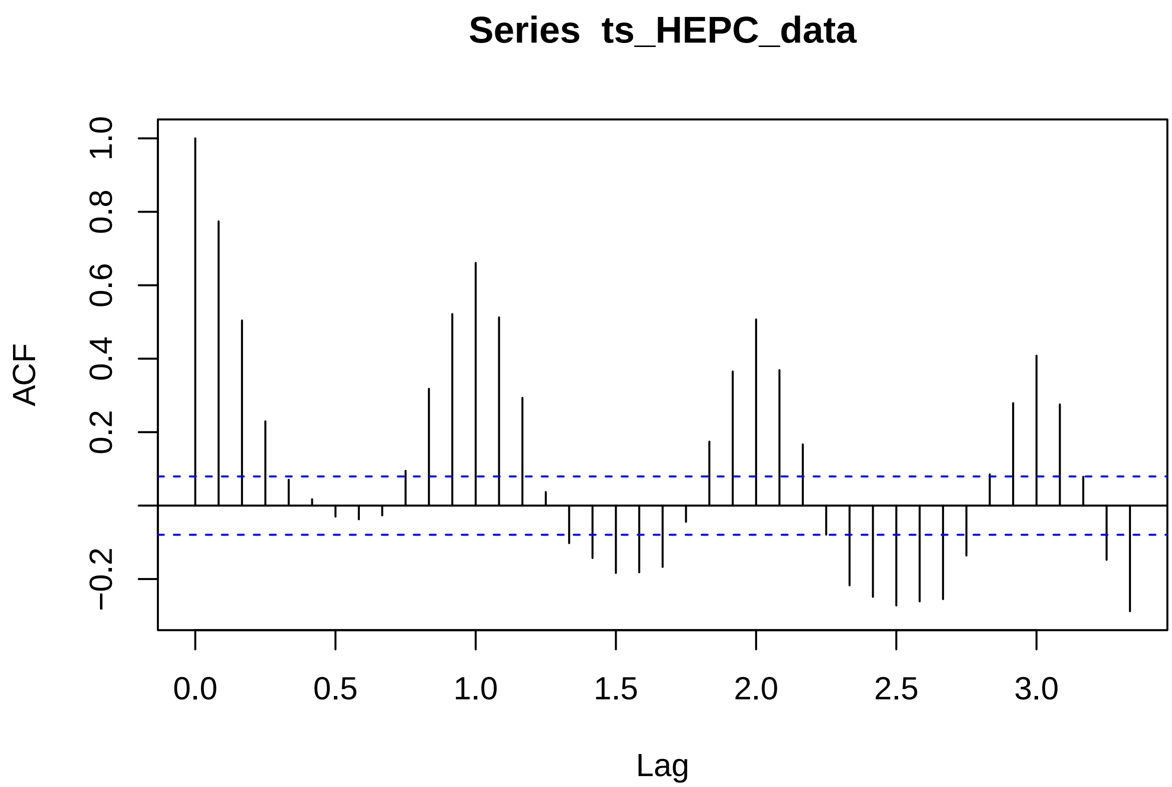
```
acf_TBEP <-acf(ts_TBEP_data,lag.max =40)
```




```
acf_TREP <-acf(ts_TREP_data, lag.max =40)
```




```r
acf_HEPC <-acf(ts_HEPC_data,lag.max=40)
```

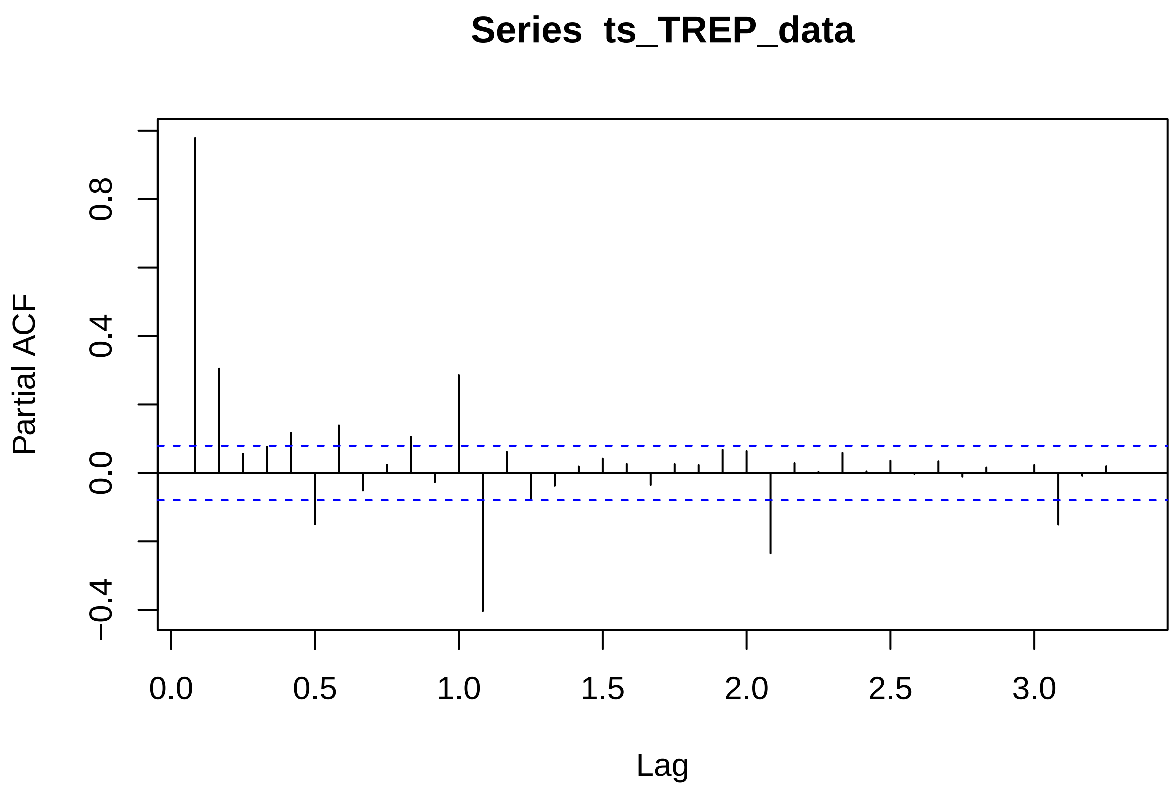



```
#looking at the significance threshold we can interpret that data from TBEP
#and TREP both as each lag is far above the significance threshold.
#This data is not stationary and suggest a positive autocorrelation.
#Whereas with the HEPC instead demonstrates seasonality.
```

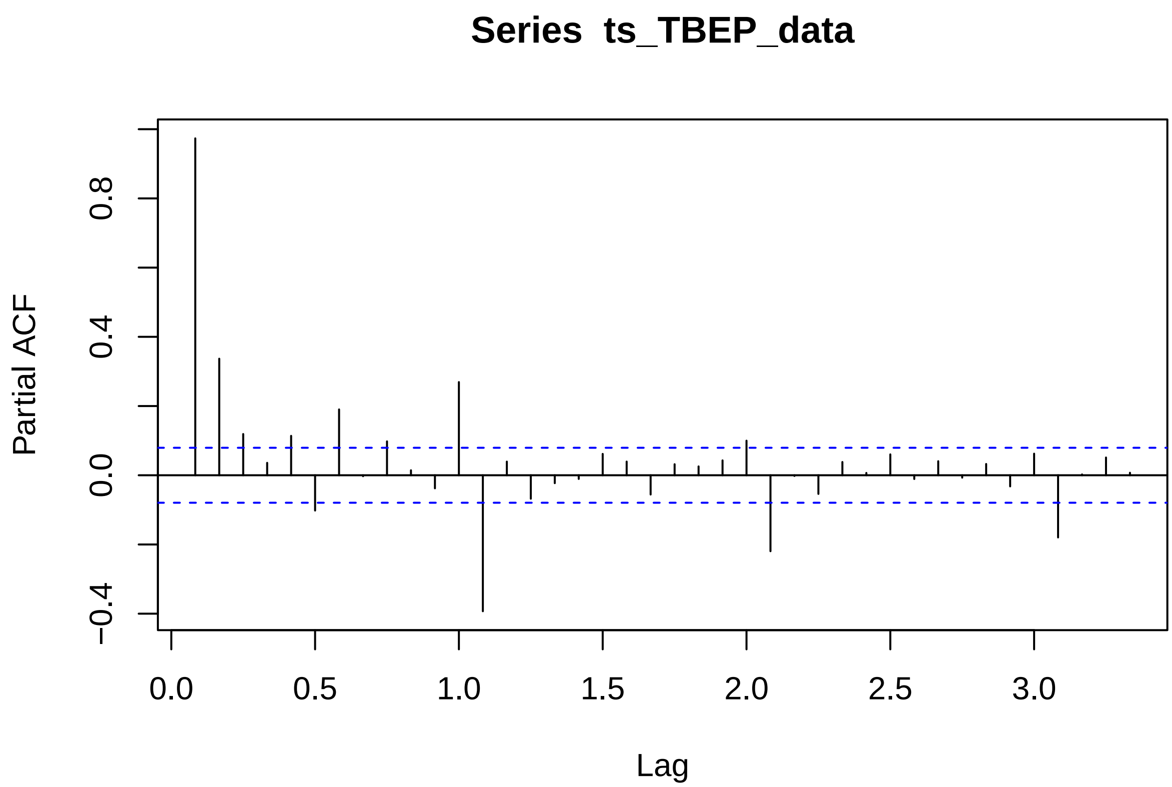
### Question 7

Compute the partial autocorrelation function from lag 1 to lag 40 for these three variables. How these plots differ from the ones in Q6?

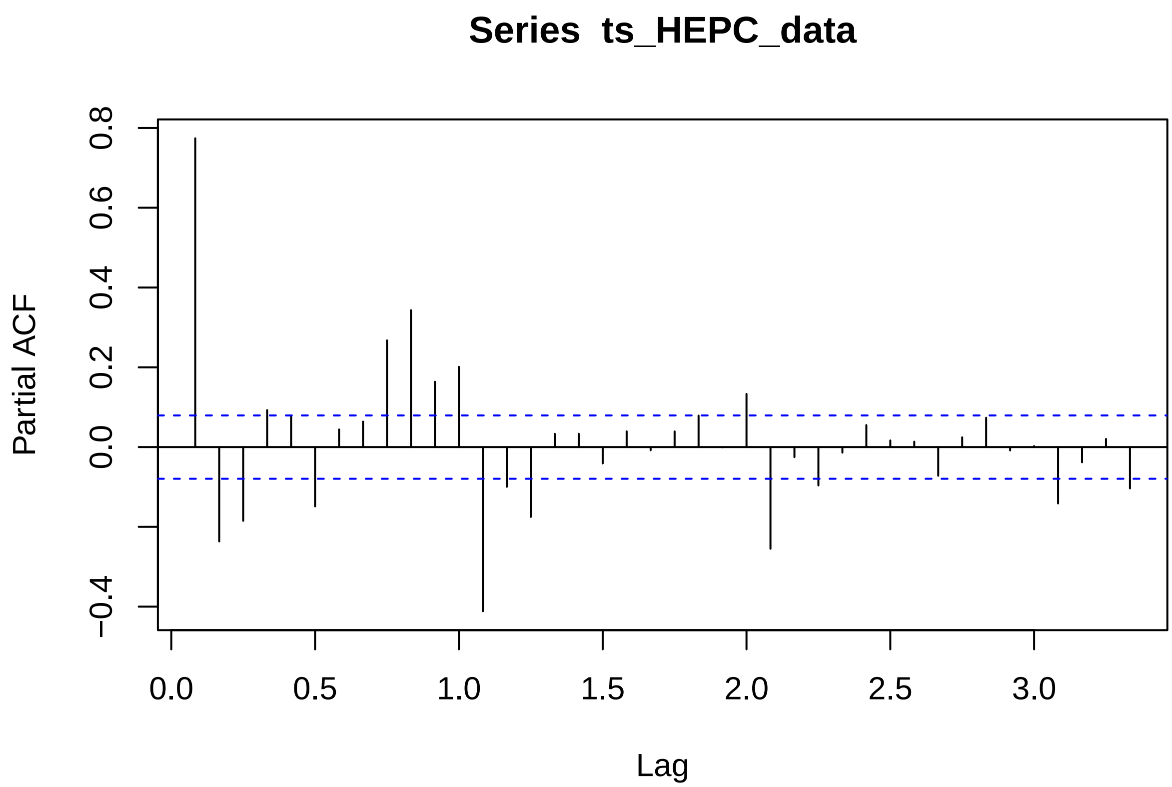
```
pacf(ts_TREP_data, lag.max = 40)
```




```
pacf(ts_TBEP_data, lag.max = 40)
```




```
pacf(ts_HEPC_data, lag.max = 40)
```




```
#With the PCF plots for TREP and TBEP data it is easier to see the peaks at
#lag 1.0, 2.0, and 3.0, suggesting that their might also be some sort of
#seasonality associated with the data.
#For HEPC data you can still see the seasonality but the decay becomes
#more prominent with each succeeding lag.
```