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

Assignment 3 - Due date 02/01/24

Faustin Kambale

Directions

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).

Please keep this R code chunk options for the report. It is easier for us to grade when we can see code and output together. And the tidy.opts will make sure that line breaks on your code chunks are automatically added for better visualization.

When you have completed the assignment, **Knit** the text and code into a single PDF file. Submit this pdf using Sakai.

Questions

Consider the same data you used for A2 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 December 2022 **Monthly** Energy Review. Once again 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.

R packages needed for this assignment:"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(cowplot)
library(dplyr)
library(tidyr)
```

Inserting data

```
raw_data <- read.csv(file
="/Users/faustinkambale/Library/CloudStorage/OneDrive-DukeUniversity/Spring 2024 classes/Time Series 4 
```

```
                    header=TRUE,skip=0)

head(raw_data,10)
```

Set Dataset to work with

```
hw_data <- raw_data[, c("Month", "Total.Biomass.Energy.Production",
                       "Total.Renewable.Energy.Production",
                       "Hydroelectric.Power.Consumption")]
colnames(hw_data) <- c("Month", "bioprod", "renen","hydrocon")
nobs <- nrow(hw_data)
head(hw_data,10)
str(hw_data)
```

Transform Dataset in time series

```
ts_renerg <- ts(hw_data$renen, start=c(1980,1),frequency=12) # ts for Energy production
ts_bioenerg <- ts(hw_data$bioprod, start=c(1980,1), frequency=12) # ts for biomass energy production
ts_hydro <- ts(hw_data$hydrocon, start=c(1980,1),frequency=12) # ts for hydroelectric power consumption
```

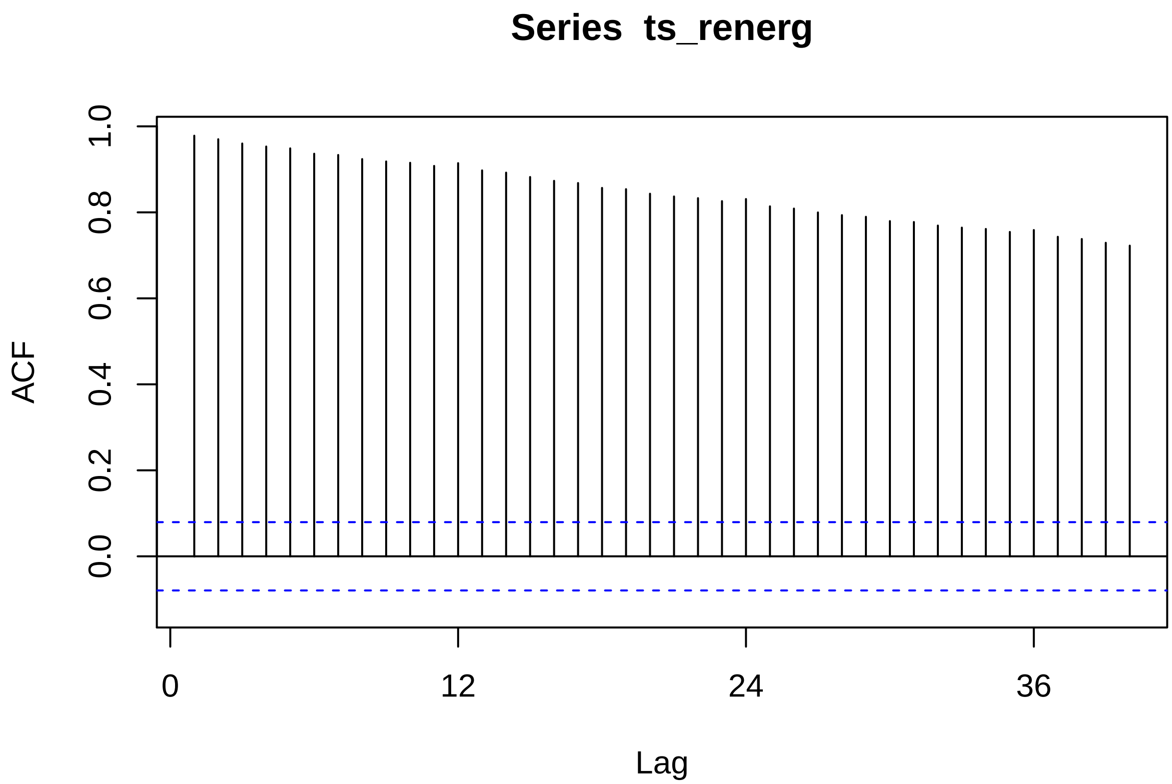
##Trend Component

Q1

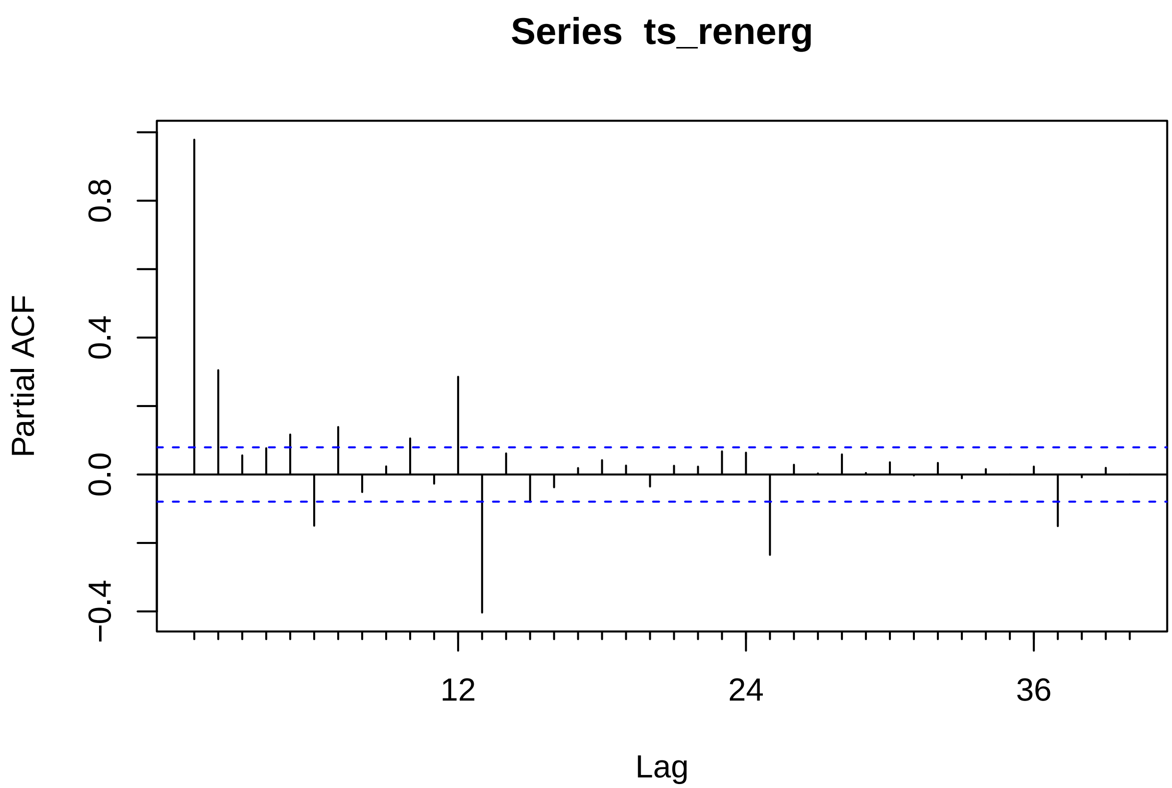
For each time series, i.e., Renewable Energy Production and Hydroelectric Consumption create three plots: one with time series, one with the ACF and with the PACF. You may use the some code form A2, but I want all the three plots side by side as in a grid. (Hint: use function `plot_grid()` from the `cowplot` package)

Creating the plots

```
#Plots for Renewable energy production
reneg <- autoplot(ts_renerg, main="Energy Production", xlab = "Years", ylab = "Inflow")# autoplot for e
reneg_acf <- Acf(ts_renerg, lag.max = 40)
```

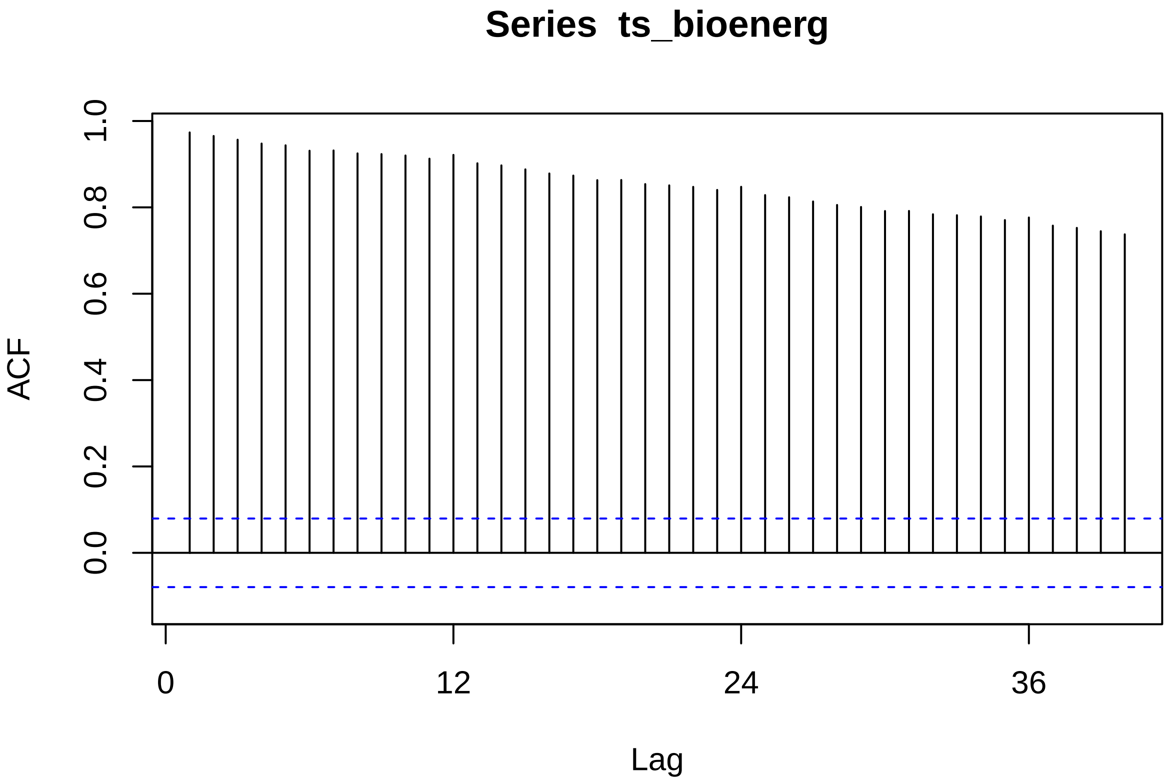



```r
reneg_pacf <- Pacf(ts_renerg, lag.max = 40)
```

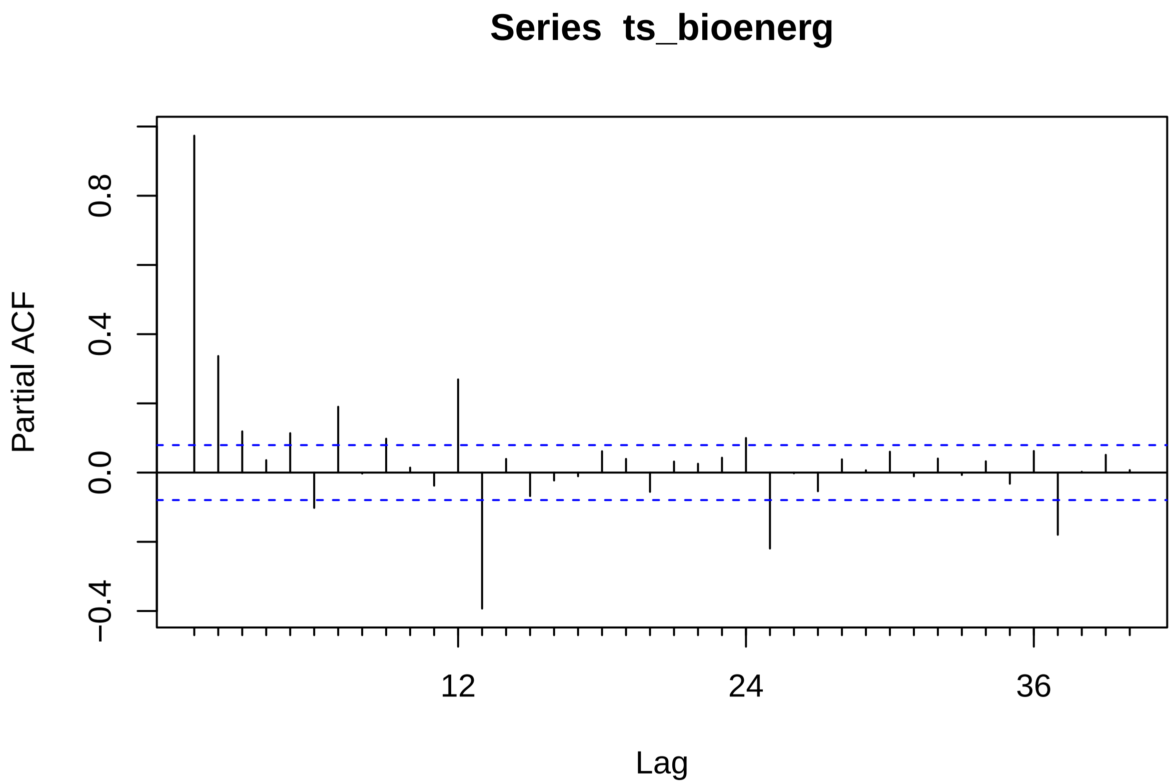



```r
plotenerg_acf <- autoplot(reneg_acf) #plot for Renewable energy production
plotenerg_pacf <- autoplot(reneg_pacf) #plot for Renewable energy production
```

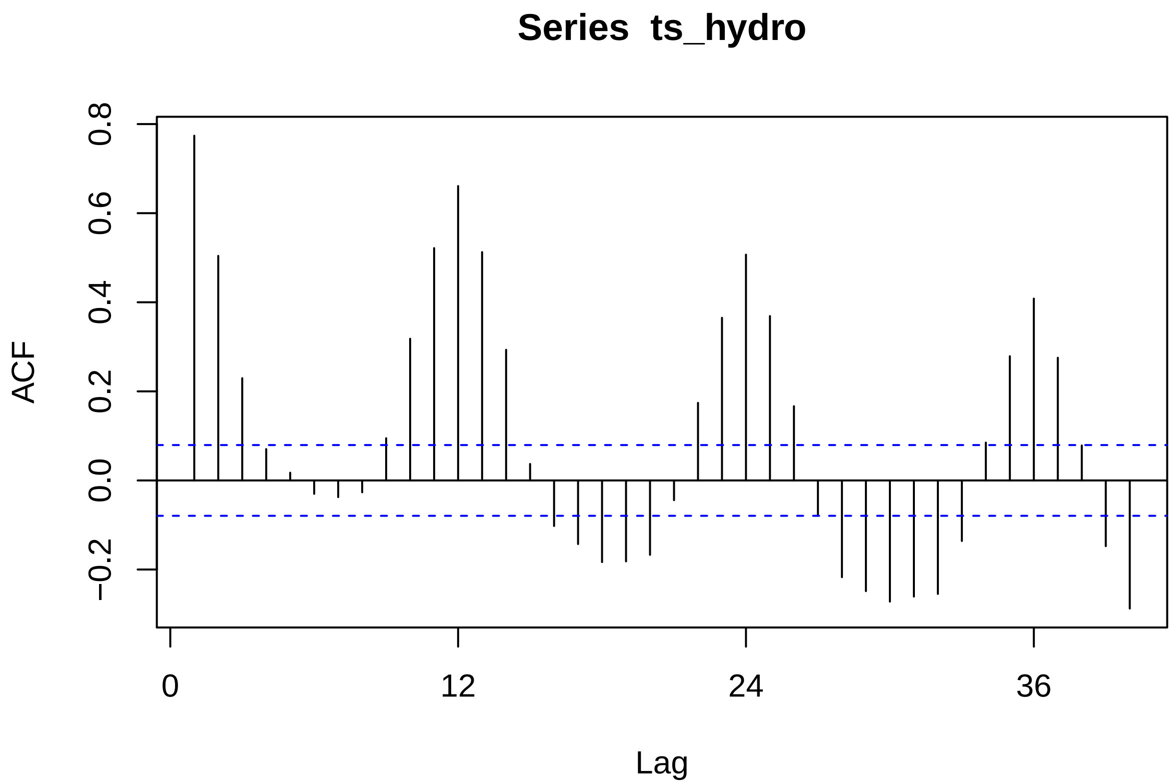
```r
#Plot for Biomass Energy production
biom <- autoplot(ts_bioenerg, main="BiomEn. Production", xlab="Years", ylab="Inflow")
biom_acf <- Acf(ts_bioenerg, lag.max = 40)
```



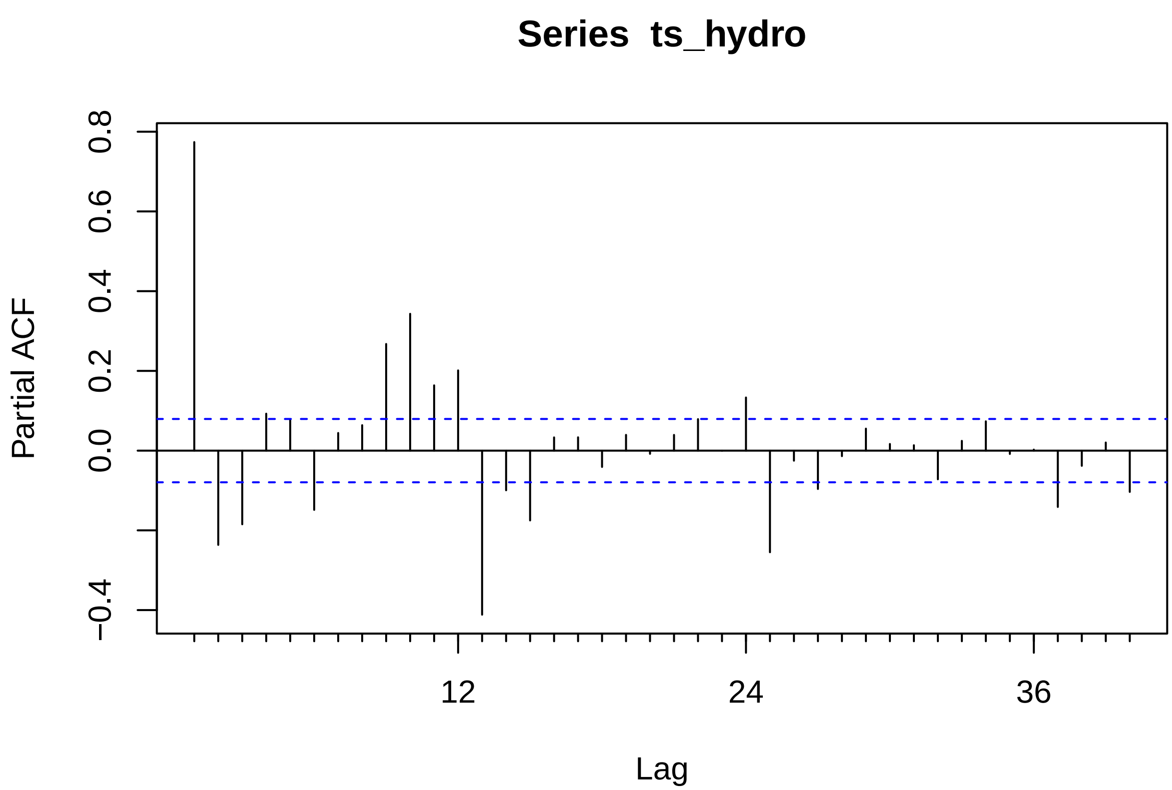

```r
biom_pacf <- Pacf(ts_bioenerg, lag.max = 40)
```




```
plotbiomeneg_acf <- autoplot(biom_acf)
plotbiomeneg_pacf <- autoplot(biom_pacf)
#Plot for Hydroelectric consumption
hydro <- autoplot(ts_hydro, main="Hydro Consum", xlab="Years", ylab="Inflow")
hydro_acf <- Acf(ts_hydro, lag.max = 40)
```

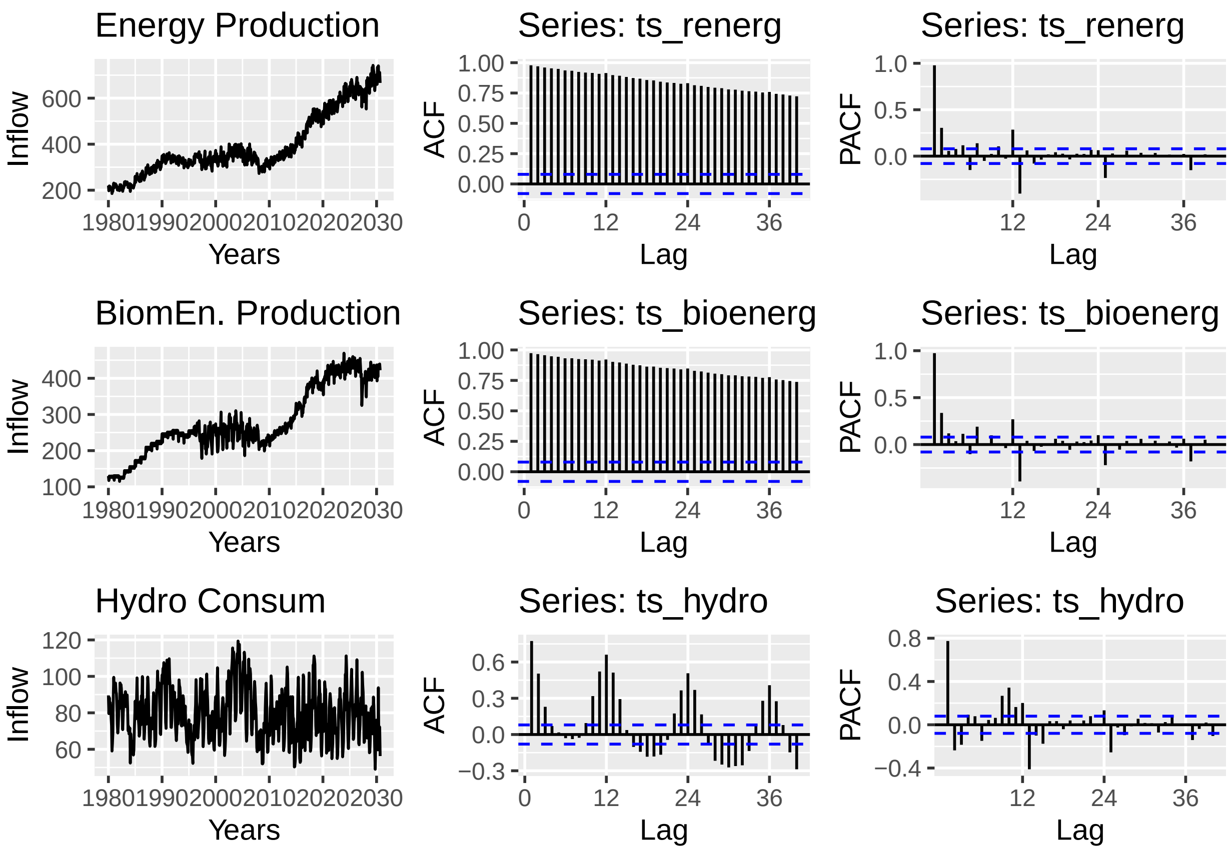



```r
hydro_pacf <- Pacf(ts_hydro, lag.max = 40)
```




```r
plothydro_acf <- autoplot(hydro_acf)
plothydro_pacf <- autoplot(hydro_pacf)
```

```
allplots <- plot_grid(reneg, plotenerg_acf, plotenerg_pacf, biom, plotbiomeneg_acf,
                      plotbiomeneg_pacf, hydro, plothydro_acf, plothydro_pacf,
                      nrow=3,ncol=3)
ggsave("allplots.tiff", width = 6, height = 6)
allplots
```



Q2

From the plot in Q1, do the series Total Biomass Energy Production, Total Renewable Energy Production, Hydroelectric Power Consumption appear to have a trend? If yes, what kind of trend?

From these plots, the Renewable Energy Production plot and the biomass production have a significant decreasing trend (clearly shown through the Acf and Pcf plots). However, the hydroelectric power consumption plot displays an ambiguous trend.

Q3

Use the *lm()* function to fit a linear trend to the three time series. Ask R to print the summary of the regression. Interpret the regression output, i.e., slope and intercept. Save the regression coefficients for further analysis.

```
t <- 1:nobs #I create vector t

#combining *t* and each variable created into one data frame for nobs = 609
databio <- data.frame("t"=t[1:609],"original"=hw_data$bioprod)
dataenerg <- data.frame("t"=t[1:609],"original"=hw_data$renen)
datahydro <- data.frame("t"=t[1:609],"original"=hw_data$hydrocon)
```

```r
#Running a multiple regression model
lintrend <- lm(ts_bioenerg ~ t + ts_renerg + ts_hydro, data = databio) #
summary(lintrend)
```

```
##
## Call:
## lm(formula = ts_bioenerg ~ t + ts_renerg + ts_hydro, data = databio)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -81.670 -10.845   3.344  14.464  39.952
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 72.94619    5.18599  14.066  < 2e-16 ***
## t            0.06484    0.01090   5.946 4.66e-09 ***
## ts_renerg    0.57829    0.01382  41.847  < 2e-16 ***
## ts_hydro    -0.52375    0.05940  -8.817  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 19.92 on 605 degrees of freedom
## Multiple R-squared:  0.954,  Adjusted R-squared:  0.9538
## F-statistic:  4185 on 3 and 605 DF,  p-value: < 2.2e-16
```

This regression model is statistically significant given a high T-value for all the variables. The model, as well, is a good fit based on the high R-squared value of 95%.

Saving regression coefficients

```r
beta0=as.numeric(lintrend$coefficients[1])
beta1=as.numeric(lintrend$coefficients[2:4])
```

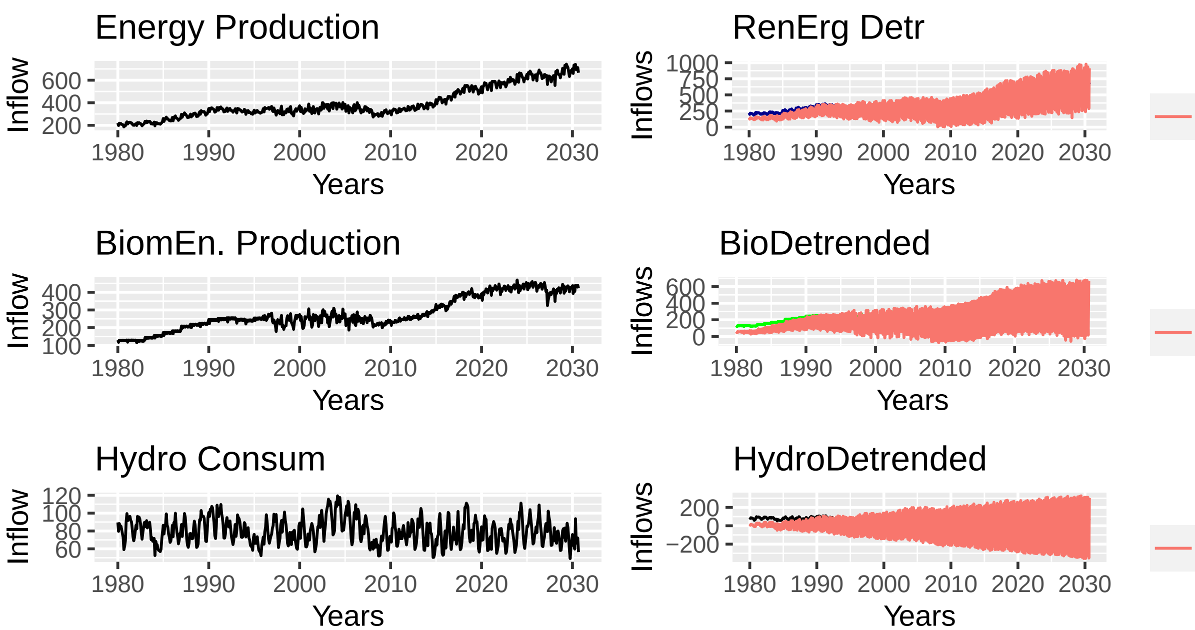
Q4

Use the regression coefficients from Q3 to detrend the series. Plot the detrended series and compare with the plots from Q1. What happened? Did anything change?

Detrending

```r
#Create detrended series from linear trend
detrenerg <- hw_data$renen - (beta0+beta1*t[1:609])
detrendbio <- hw_data$bioprod - (beta0+beta1*t[1:609])
detrendhydro <-hw_data$hydrocon - (beta0+beta1*t[1:609])

ts_detrenerg <- ts(detrenerg, start= c(1980,1),frequency=12) #for Renewable Energy
#visualize the trend on observed data
plotdetener <- autoplot(ts_renerg, color="darkblue") +
  autolayer(ts_detrenerg,series="") +
  ylab("Inflows") +
  xlab("Years") +
  labs(color="", title = "RenErg Detr")

ts_detrebio <- ts(detrendbio, start= c(1980,1),frequency=12) #for Biomass Energy
#visualize the trend on observed data
```

```r
plotdetbio <- autoplot(ts_bioenerg, color="green") +
  autolayer(ts_detrebio,series="") +
  ylab("Inflows") +
  xlab("Years") +
  labs(color="", title = "BioDetrended")


ts_dethydro <- ts(detrendhydro, start= c(1980,1),frequency=12) #for Hydro Energy
#visualize the trend on observed data
plotdethydro <- autoplot(ts_hydro, color="black") +
  autolayer(ts_dethydro,series="") +
  ylab("Inflows") +
  xlab("Years") +
  labs(color="", title = "HydroDetrended")

allplots1 <- plot_grid(reneg, plotdetener, biom, plotdetbio, hydro,
                       plotdethydro, #ploteenerg_acf, #ploteenerg_pcf, #plotbiomeneg_acf,
                      #plotbiomeneg_pcf, #plothydro_acf, #plothydro_pcf,
                      nrow=4,ncol=2)
ggsave("allplots1.tiff", width = 6, height = 6)
allplots1
```

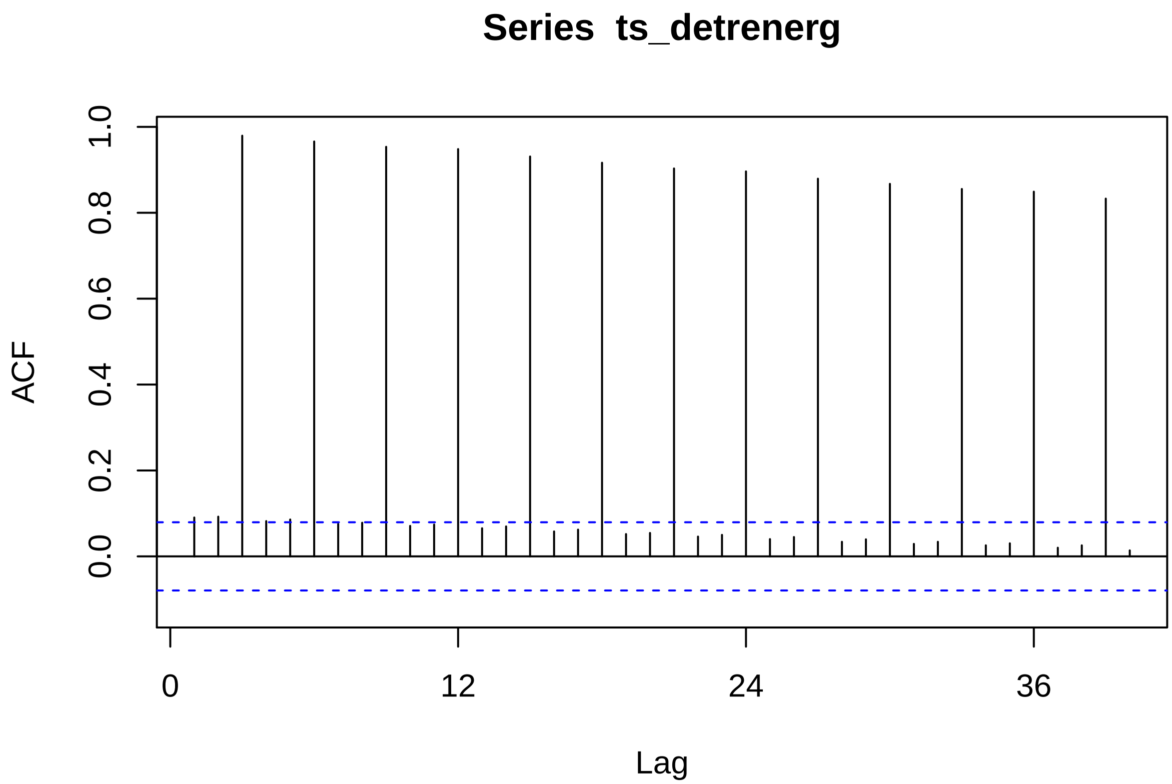


As from these plots, the increase pattern in the data is clearly visible based on the plot with detrended series. For instance, the variability from the Hydroelectric consumption series was very fluctuating, but seems to be stabilized after detrending, displaying the increase over time. Same can be noticed with other variables as well.

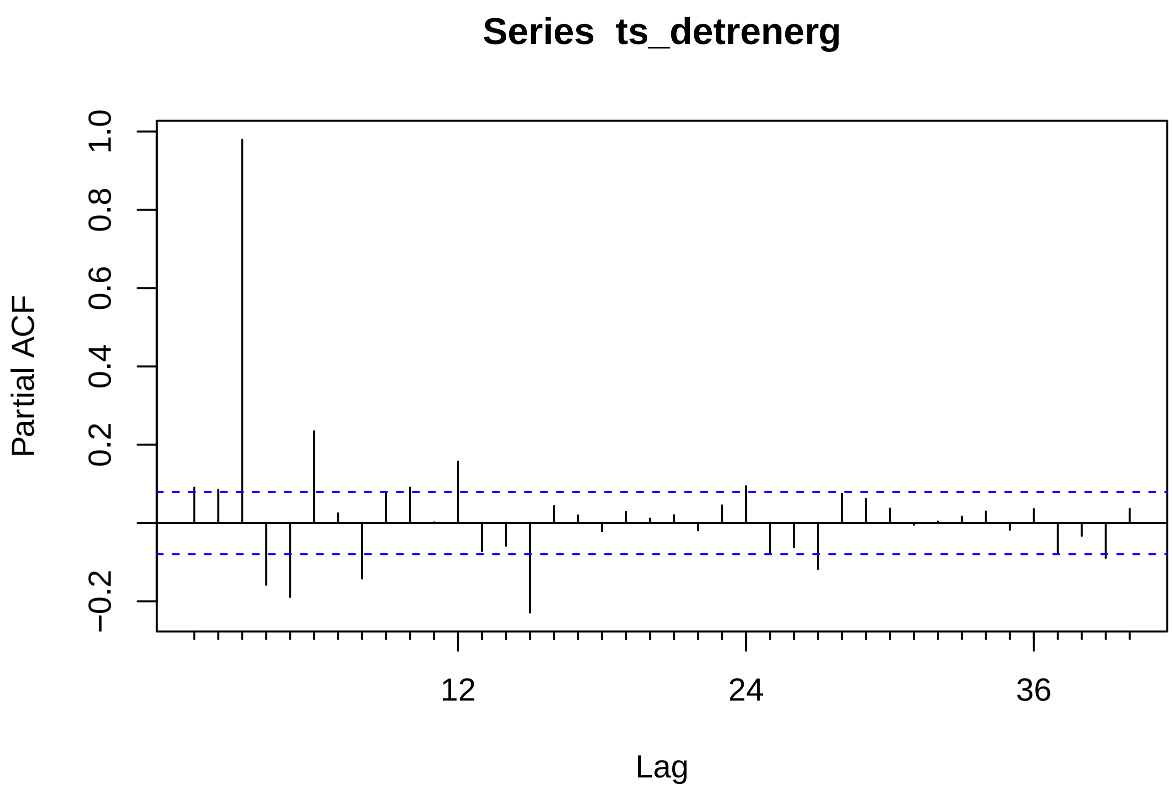
Q5

Plot ACF and PACF for the detrended series and compare with the plots from Q1. You may use plot_grid() again to get them side by side, but mot mandatory. Did the plots change? How?

```r
#plot for Detrended Renewable energy production
detreneg_acf <- Acf(ts_detrenerg, lag.max = 40)
```

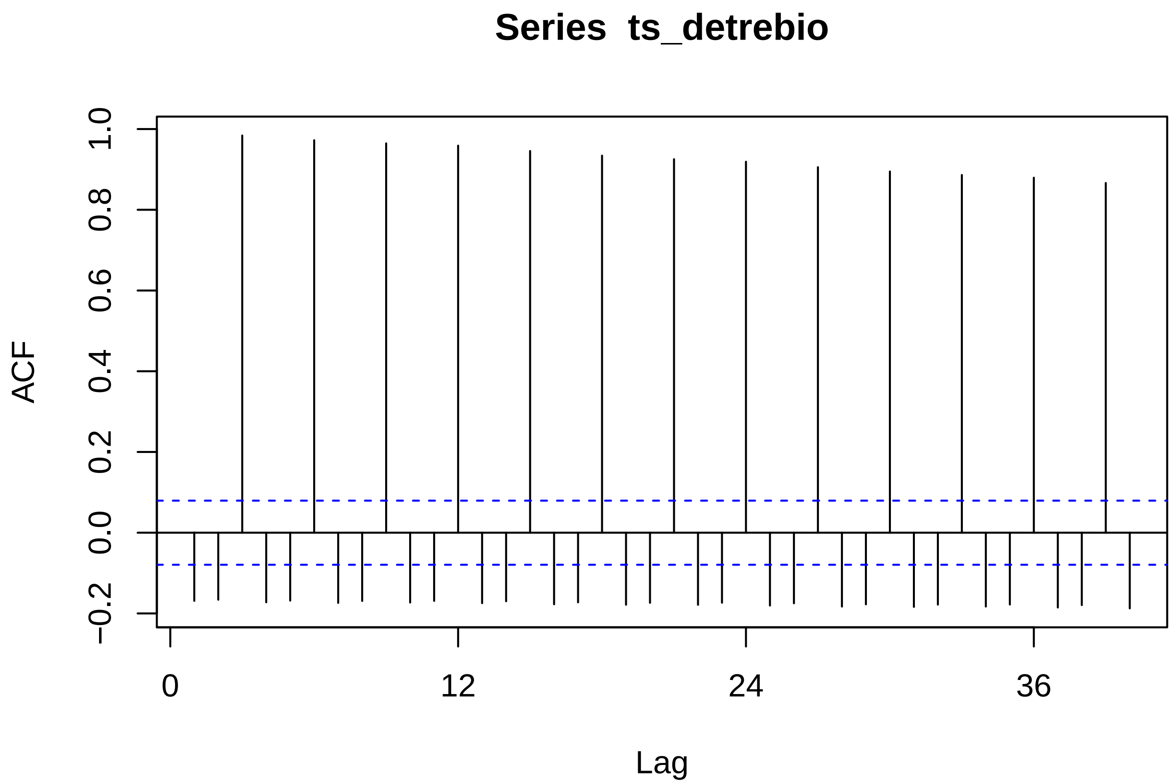



```
detreneg_pacf <- Pacf(ts_detrenerg, lag.max = 40)
```

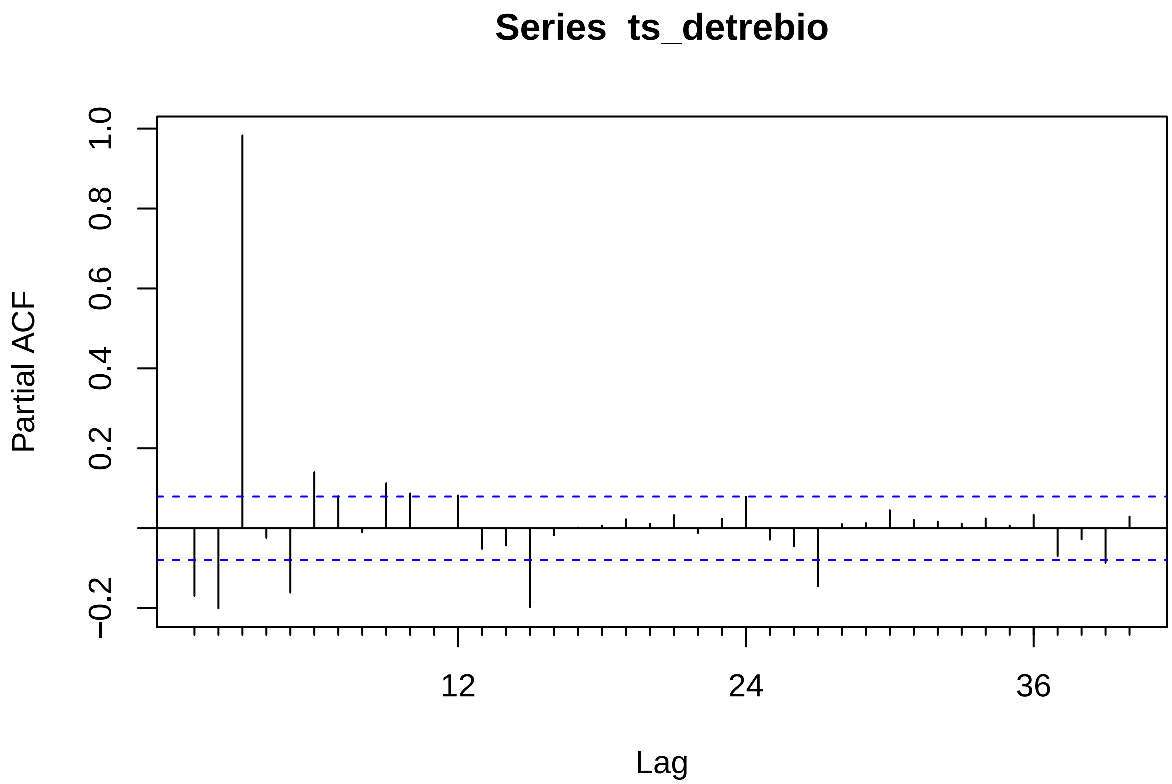



```
plotdetreneg_acf <- autoplot(detreneg_acf)
plotdetreneg_pacf <- autoplot(detreneg_pacf) #plot for Renewable energy production
```

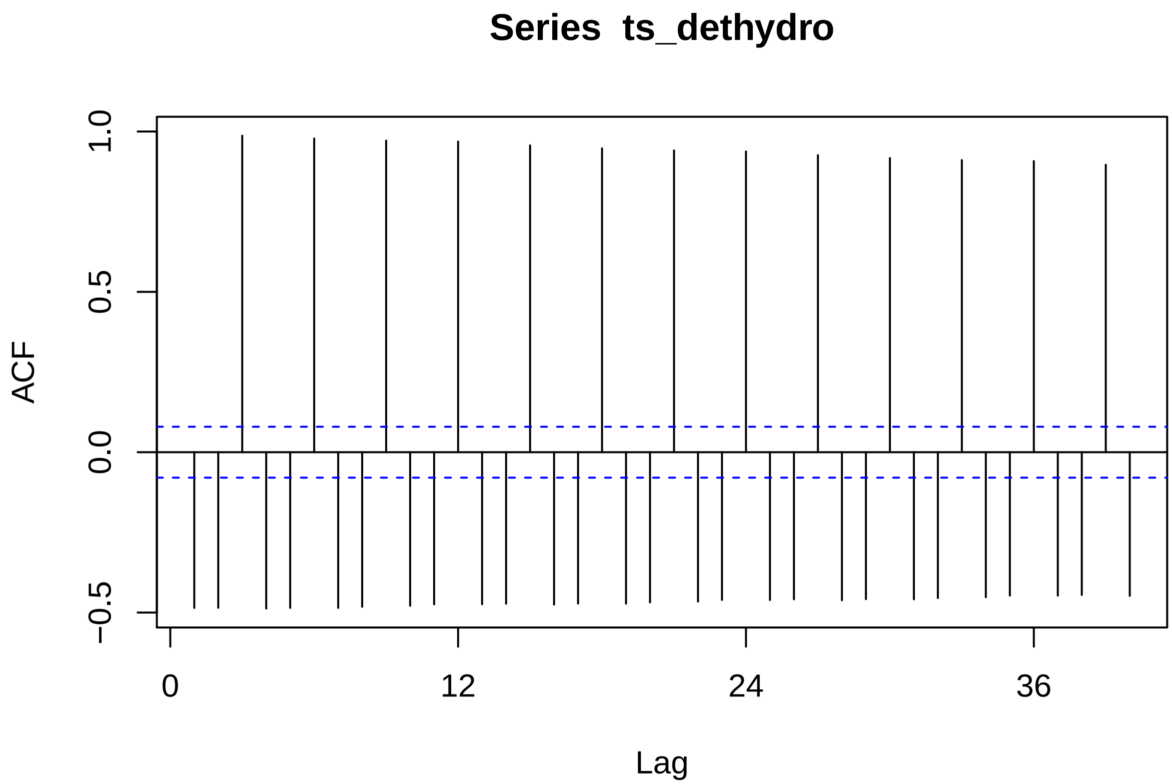
```r
#Plot for detrended Biomass energy production
detrbiom_acf <- Acf(ts_detrebio, lag.max = 40)
```



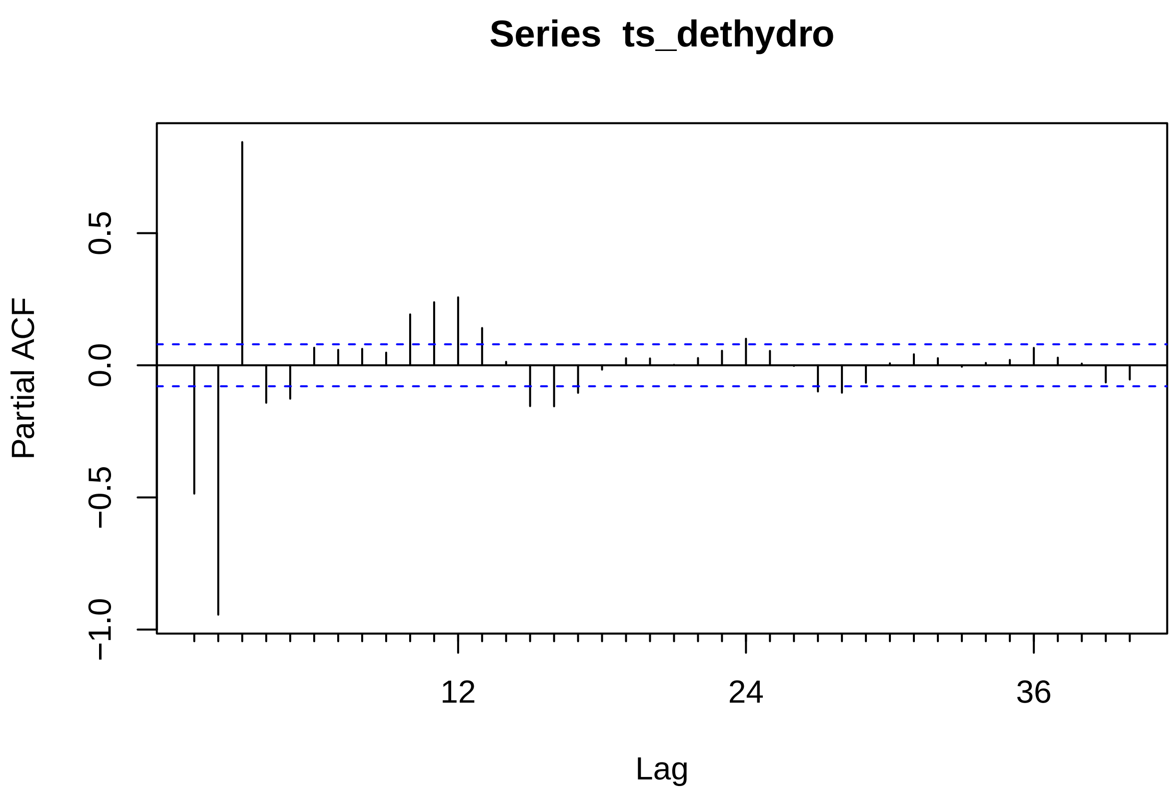

```r
detrbiom_pacf <- Pacf(ts_detrebio, lag.max = 40)
```




```
plotdetrbiom_acf <- autoplot(detrbiom_acf)
plotdetrbiom_pacf <- autoplot(detrbiom_pacf)

#Plot for Hydroelectric consumption

detrhydro_acf <- Acf(ts_dethydro, lag.max = 40)
```

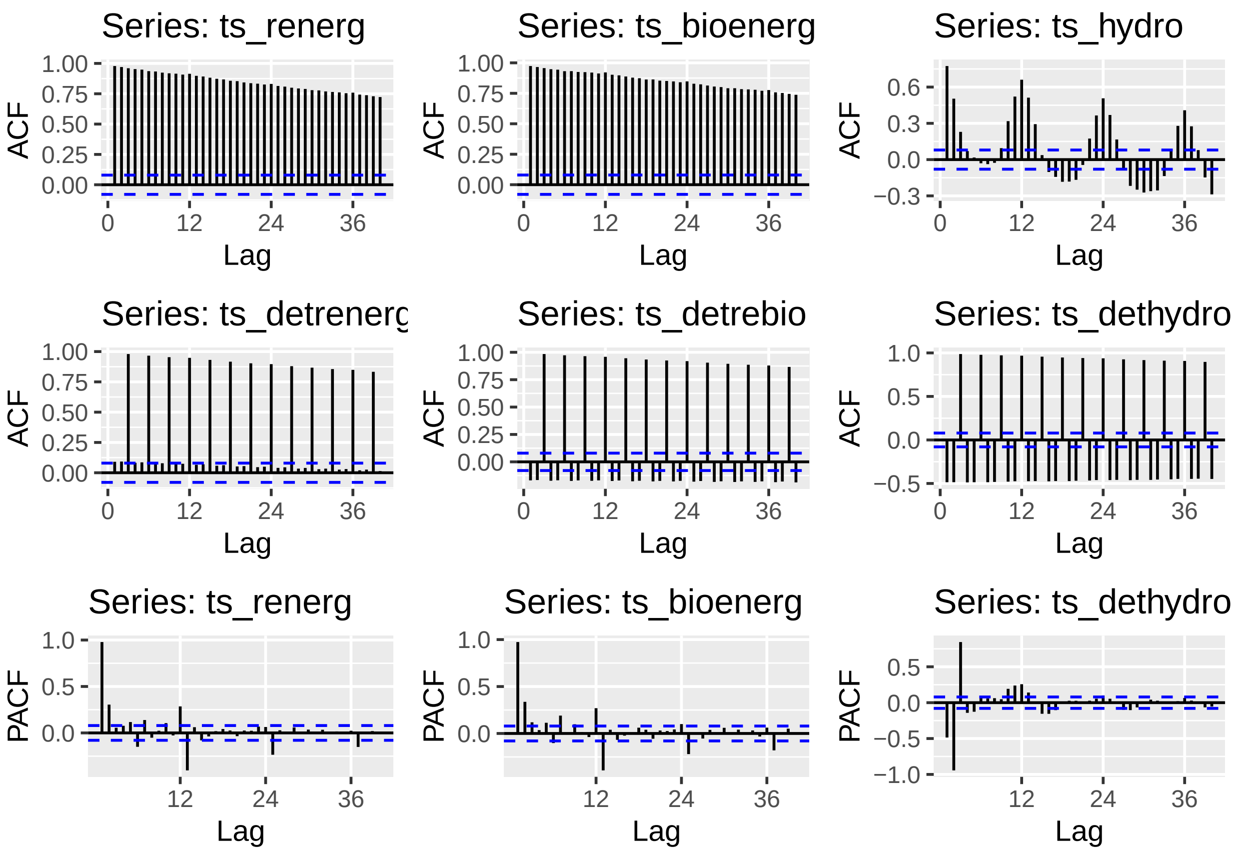



```r
detrhydro_pacf <- Pacf(ts_dethydro, lag.max = 40)
```




```r
plotdetrhydro_acf <- autoplot(detrhydro_acf)
plotdetrhydro_pacf <- autoplot(detrhydro_pacf)
```

```r
allplots2 <- plot_grid(ploteneg_acf, plotbiomeneg_acf, plothydro_acf,
                       plotdetreneg_acf, plotdetrbiom_acf, plotdetrhydro_acf,
                       ploteneg_pacf, plotbiomeneg_pacf, plotdetrhydro_pacf,
                       plotdetreneg_pacf, plotdetrbiom_pacf,
                       nrow=3,ncol=3)
ggsave("allplots2.tiff", width = 6, height = 6)
allplots2
```



The most important change one can notice on the plots is that the trend is now stable, and one can easily identify the trend (either increase or decrease). Also, the detrendred variables variations are now close to zero while previously it was spread away from zero.

Seasonal Component

Set aside the detrended series and consider the original series again from Q1 to answer Q6 to Q8.

Q6

Just by looking at the time series and the acf plots, do the series seem to have a seasonal trend? No need to run any code to answer your question. Just type in you answer below.

Time series and Acf plot display different view of the data. For the variables Renewable Energy production and biomass energy production, ts display seasonal trend, but the seasonal trends variability is more visible through the acf than the time series plots.

Q7

Use function *lm()* to fit a seasonal means model (i.e. using the seasonal dummies) the two time series. Ask R to print the summary of the regression. Interpret the regression output. From the results which series have a seasonal trend? Do the results match you answer to Q6?

```
#for detrendred Renewable Energy
dummiesenerg <- seasonaldummy(ts_detrenerg)
#Then fit a linear model to the seasonal dummies
seas_means_model=lm(detrenerg~dummiesenerg)
summary(seas_means_model)
```

```
##
## Call:
## lm(formula = detrenerg ~ dummiesenerg)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -349.21  -85.00  -28.33   57.53  483.33
##
## Coefficients:
##                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)      491.849     21.895  22.464  < 2e-16 ***
## dummiesenergJan -176.822     30.812  -5.739 1.52e-08 ***
## dummiesenergFeb -369.389     30.812 -11.988  < 2e-16 ***
## dummiesenergMar    2.358     30.812   0.077    0.939
## dummiesenergApr -188.674     30.812  -6.123 1.66e-09 ***
## dummiesenergMay -330.003     30.812 -10.710  < 2e-16 ***
## dummiesenergJun   -4.198     30.812  -0.136    0.892
## dummiesenergJul -177.714     30.812  -5.768 1.29e-08 ***
## dummiesenergAug -343.409     30.812 -11.145  < 2e-16 ***
## dummiesenergSep  -27.547     30.812  -0.894    0.372
## dummiesenergOct -200.047     30.964  -6.461 2.17e-10 ***
## dummiesenergNov -356.994     30.964 -11.529  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 154.8 on 597 degrees of freedom
## Multiple R-squared:  0.4562, Adjusted R-squared:  0.4462
## F-statistic: 45.54 on 11 and 597 DF,  p-value: < 2.2e-16
```

```
#for detrendred Biomass Production
dummiesbiom <- seasonaldummy(ts_detrebio)
#Then fit a linear model to the seasonal dummies
seas_means_model1=lm(detrendbio~dummiesbiom)
summary(seas_means_model1)
```

```
##
## Call:
## lm(formula = detrendbio ~ dummiesbiom)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -314.63  -60.49  -14.65   45.71  315.82
##
## Coefficients:
```

```
##                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)       377.79      16.74  22.571  < 2e-16 ***
## dummiesbiomJan  -181.46      23.55  -7.704 5.53e-14 ***
## dummiesbiomFeb  -366.20      23.55 -15.547  < 2e-16 ***
## dummiesbiomMar   -10.42      23.55  -0.443    0.658
## dummiesbiomApr  -201.58      23.55  -8.558  < 2e-16 ***
## dummiesbiomMay  -350.58      23.55 -14.884  < 2e-16 ***
## dummiesbiomJun   -19.32      23.55  -0.820    0.412
## dummiesbiomJul  -183.74      23.55  -7.801 2.76e-14 ***
## dummiesbiomAug  -338.88      23.55 -14.387  < 2e-16 ***
## dummiesbiomSep   -11.61      23.55  -0.493    0.622
## dummiesbiomOct  -183.99      23.67  -7.773 3.37e-14 ***
## dummiesbiomNov  -346.31      23.67 -14.630  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 118.4 on 597 degrees of freedom
## Multiple R-squared:  0.5851, Adjusted R-squared:  0.5774
## F-statistic: 76.53 on 11 and 597 DF,  p-value: < 2.2e-16
```

```r
#for detrendred Hydroenergy Consumption
dummieshydro <- seasonaldummy(ts_dethydro)
#Then fit a linear model to the seasonal dummies
seas_means_model2=lm(detrendhydro~dummieshydro)
summary(seas_means_model2)
```

```
##
## Call:
## lm(formula = detrendhydro ~ dummieshydro)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -187.299  -43.067    1.775   41.795  180.468
##
## Coefficients:
##                  Estimate Std. Error t value Pr(>|t|)
## (Intercept)       167.604     11.434  14.658   <2e-16 ***
## dummieshydroJan -174.977     16.091 -10.874   <2e-16 ***
## dummieshydroFeb -337.639     16.091 -20.983   <2e-16 ***
## dummieshydroMar    5.254     16.091   0.327    0.744
## dummieshydroApr -174.660     16.091 -10.855   <2e-16 ***
## dummieshydroMay -322.726     16.091 -20.057   <2e-16 ***
## dummieshydroJun   10.650     16.091   0.662    0.508
## dummieshydroJul -176.261     16.091 -10.954   <2e-16 ***
## dummieshydroAug -344.060     16.091 -21.382   <2e-16 ***
## dummieshydroSep  -15.226     16.091  -0.946    0.344
## dummieshydroOct -196.447     16.170 -12.149   <2e-16 ***
## dummieshydroNov -347.533     16.170 -21.492   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 80.85 on 597 degrees of freedom
## Multiple R-squared:  0.7495, Adjusted R-squared:  0.7449
## F-statistic: 162.4 on 11 and 597 DF,  p-value: < 2.2e-16
```

From this linear regression, we can assert that the biomass production variable and the hydro energy consumption variables have a seasonal variability given the significance of the regression model. Nevertheless, the renewable energy detrended regression output appears to be not statistically significant (R-square less that 50%).

Q8

Use the regression coefficients from Q7 to deseason the series. Plot the deseason series and compare with the plots from part Q1. Did anything change?

```r
## Saving regression coefficients for model1
beta01=as.numeric(seas_means_model1$coefficients[1])
beta11=as.numeric(seas_means_model1$coefficients[2:12])

## Saving regression coefficients for model2
beta02=as.numeric(seas_means_model2$coefficients[1])
beta12=as.numeric(seas_means_model2$coefficients[2:12])

#Deseason series for model1
seas_component=array(0,nobs)
for(i in 1:nobs){
seas_component[i]=(beta01+beta11%*%dummiesbiom[i,])
}
#Transform into a ts object
ts_deseasbiom <- ts(seas_component,start=c(1980,1),frequency=12)
#Understanding what we did
deseasbiom <-autoplot(ts_detrebio,series="Observed - Trend") +
  autolayer(ts_deseasbiom,series="seasonal_component",alpha=0.5) +
  ylab("Biomass Production") +
  xlab("Years") +
  labs(color="", title = "Seasonal Component")

#Deseason series for model2
seas_component=array(0,nobs)
for(i in 1:nobs){
seas_component[i]=(beta02+beta12%*%dummieshydro[i,])
}
#Transform into a ts object
ts_deseashydro <- ts(seas_component,start=c(1980,1),frequency=12)
#Understanding what we did
deseashydro <- autoplot(ts_dethydro,series="Observed - Trend") +
  autolayer(ts_deseashydro,series="seas_component",alpha=0.5) +
  ylab("Hydro Energy") +
  xlab("Years") +
  labs(color="", title = "Seasonal Component")
```

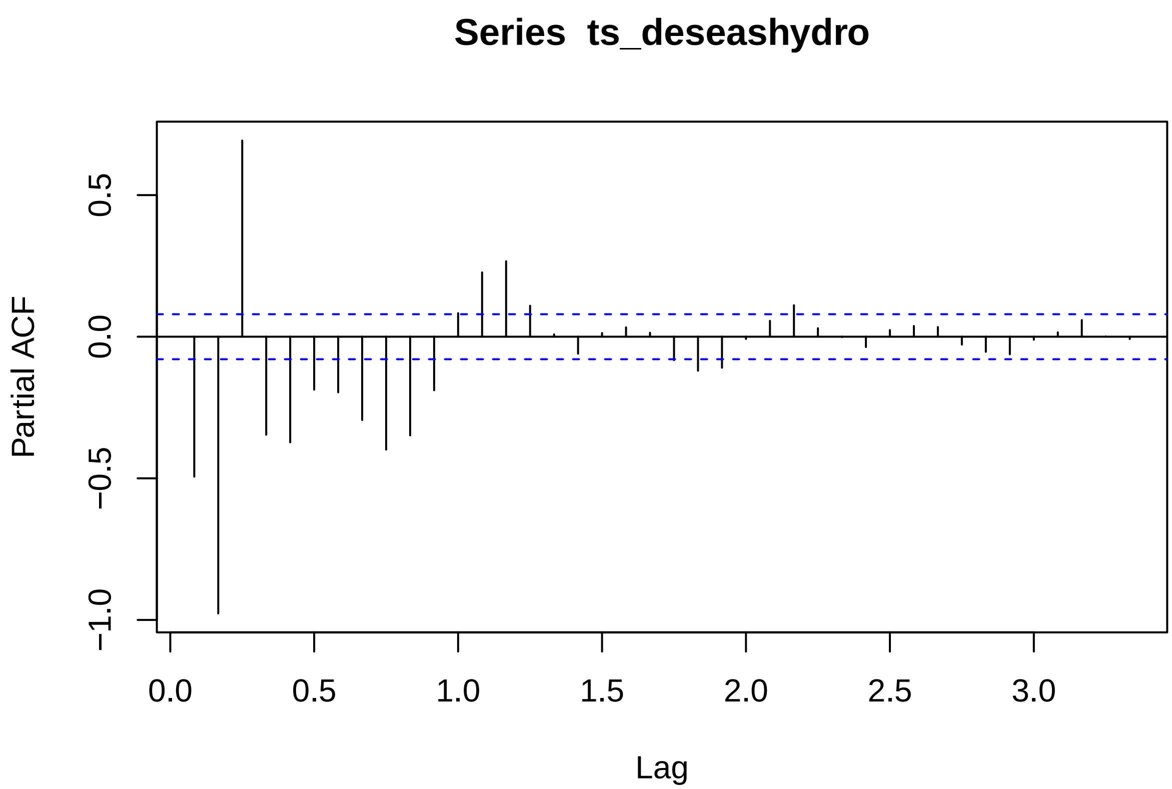
Compared to the Plots from Q1, it is clear now that the trend in these detrended plots are now stabilized. The seasonal component, for sure, is clearly visible.

Q9

Plot ACF and PACF for the deseason series and compare with the plots from Q1. You may use plot_grid() again to get them side by side. but mot mandatory. Did the plots change? How?

```r
deseashydro_acf <- Acf(ts_deseashydro, lag.max = 40)
```

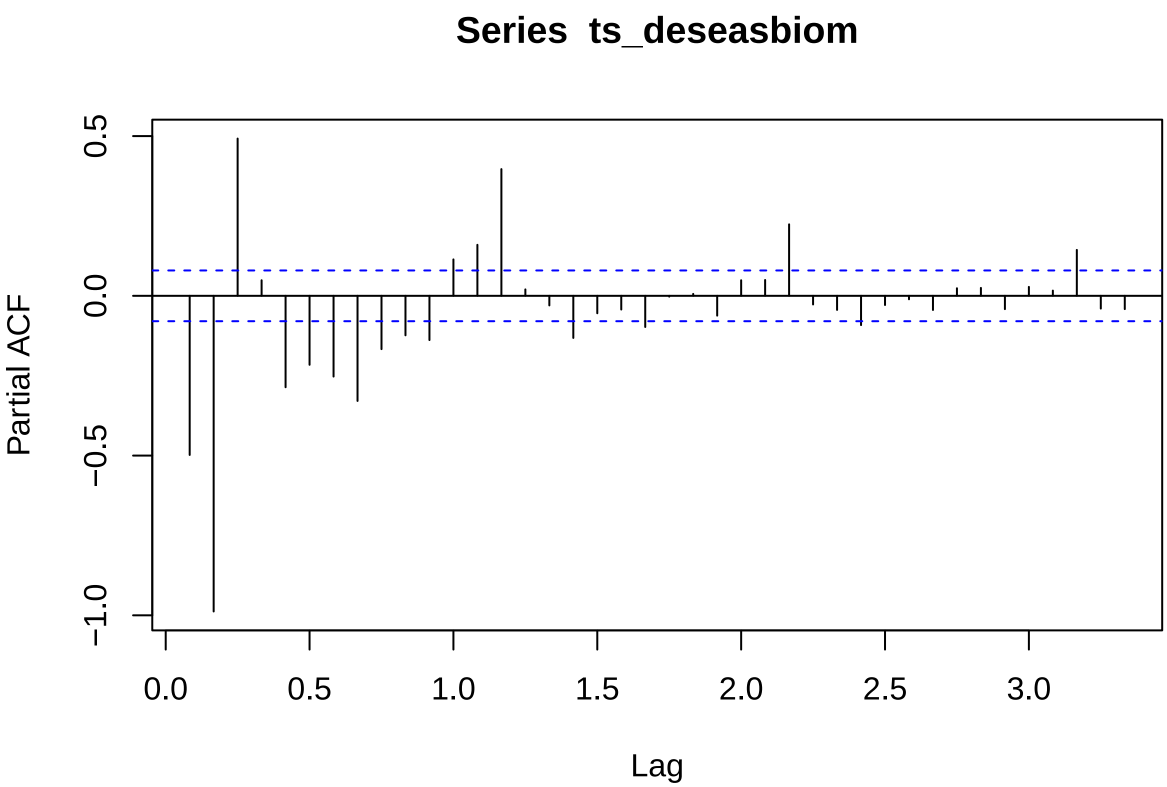

```
plotdeseashydro_acf <- autoplot(deseashydro_acf)
deseashydro_Pcf <- pacf(ts_deseashydro, lag.max = 40)
```




```
plotdeseashydro_Pcf <- autoplot(deseashydro_Pcf)
```

```r
deseasbiom_acf <- Acf(ts_deseasbiom, lag.max = 40)
```


```r
plotdeseasbiom_acf <- autoplot(deseasbiom_acf)
deseasbiom_pacf <- pacf(ts_deseasbiom, lag.max = 40)
```



```
plotdeseasbiom_pacf <- autoplot(deseasbiom_pacf)

allplots3 <- plot_grid(plotbiomeneg_acf, plotbiomeneg_pacf, plotdeseasbiom_acf,
                       plotdeseasbiom_pacf, plothydro_acf, plotdetrhydro_pacf,
                       plotdeseashydro_acf, plotdeseashydro_Pcf,
                       nrow=4,ncol=2)
ggsave("allplots3.tiff", width = 6, height = 6)
allplots3
```