ENV 790.30 - Time Series Analysis for Energy Data | Spring 2021

Assignment 3 - Due date 02/15/21

Marie McNamara

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 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).

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. Rename the pdf file such that it includes your first and last name (e.g., "LuanaLima_TSA_A01_Sp21.Rmd"). 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 January 2021 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.\

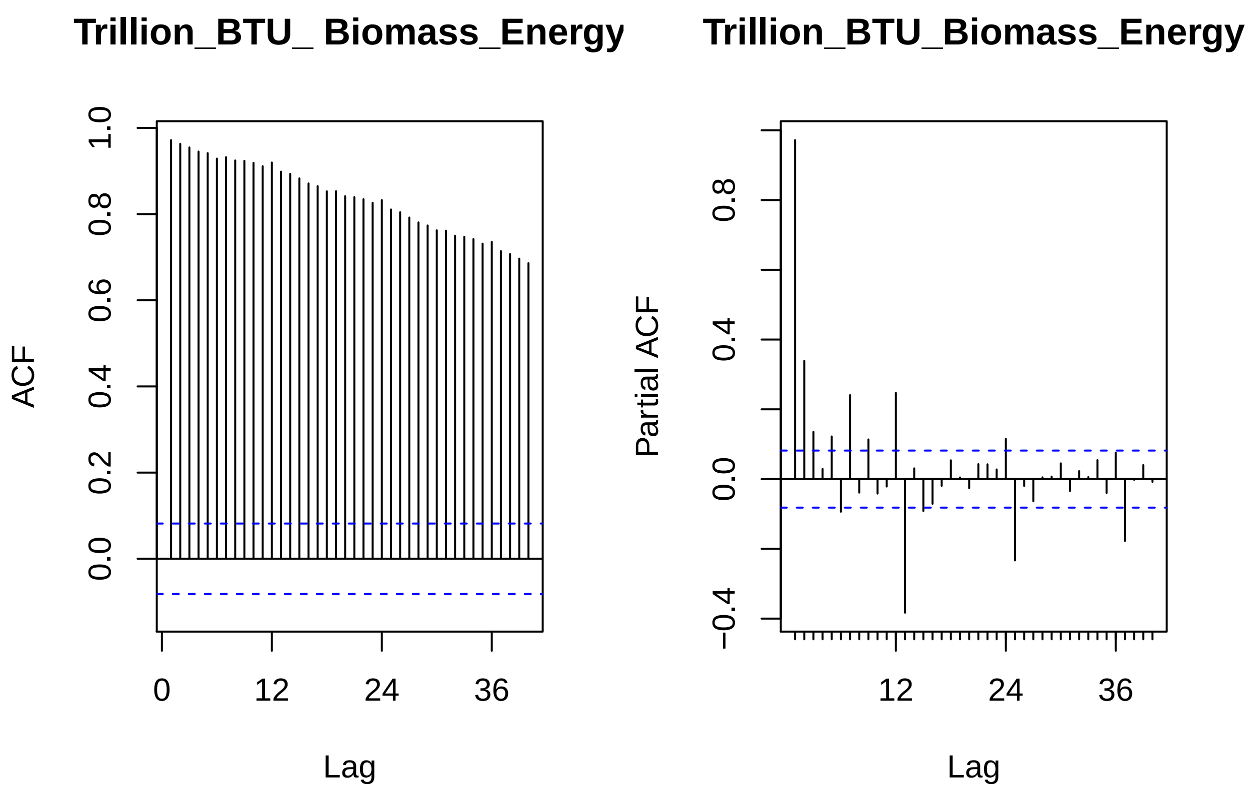
```
#Load/install required package here
library(lubridate)
library(ggplot2)
library(forecast)
library(Kendall)
library(tseries)
```

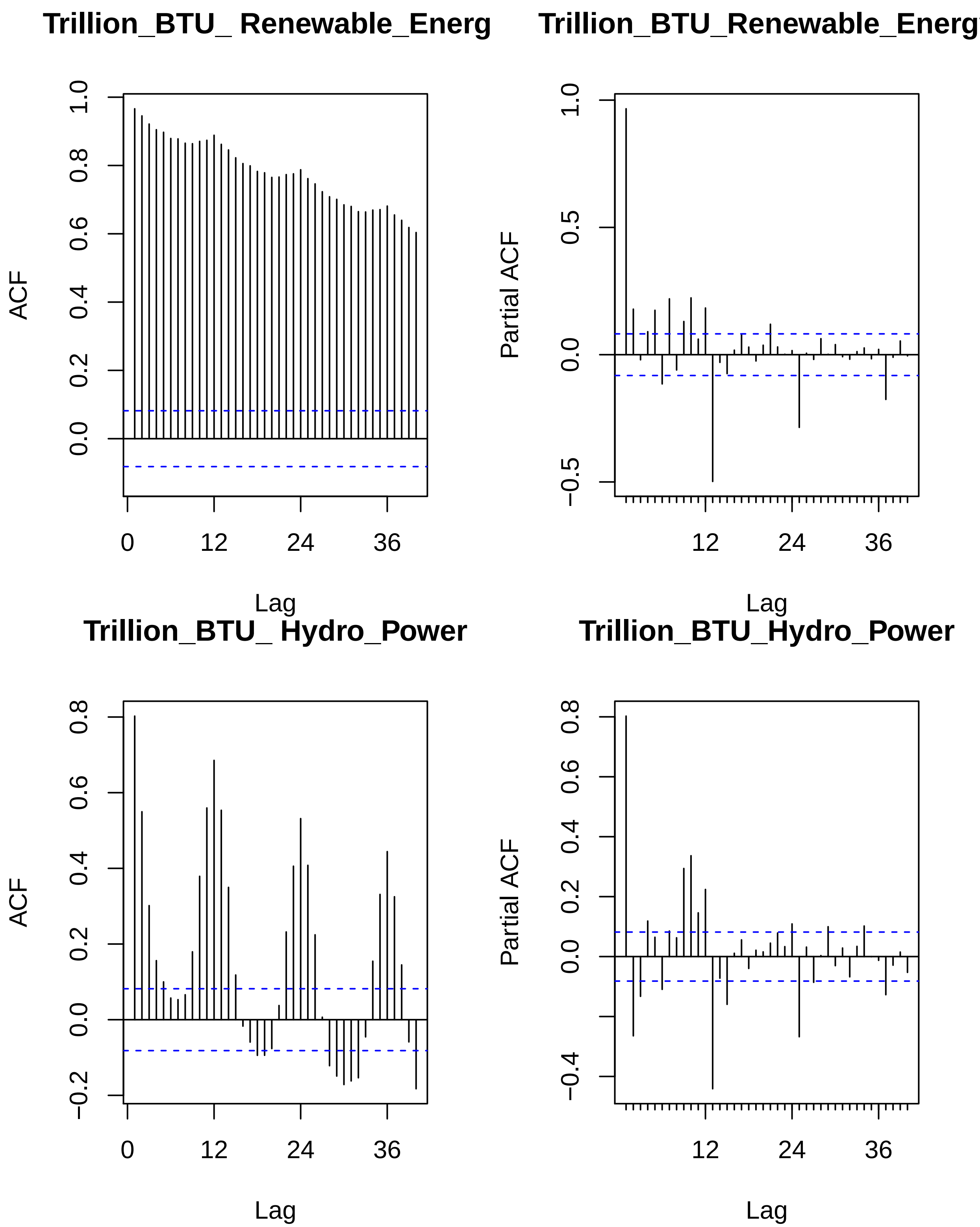
##Trend Component

Q1

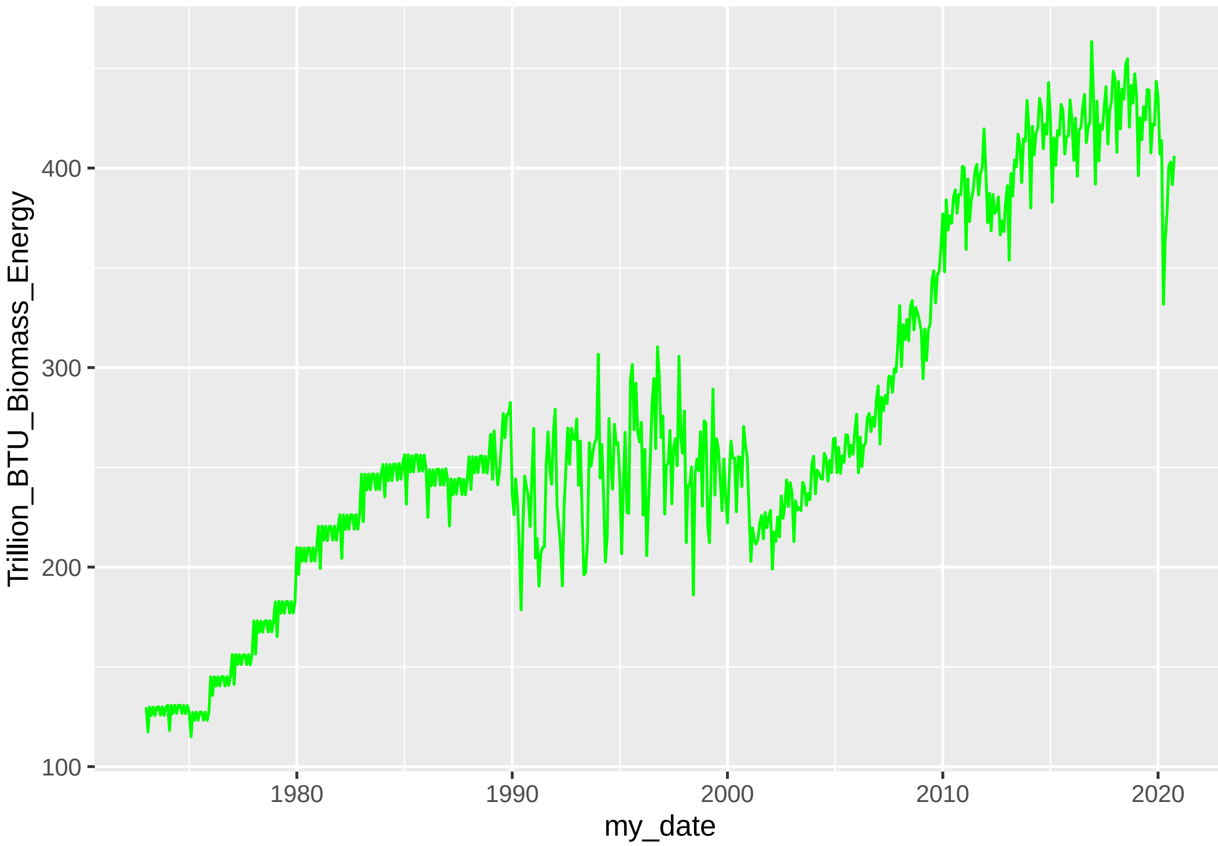
Create a plot window that has one row and three columns. And then for each object on your data frame, fill the plot window with time series plot, ACF and PACF. You may use the some code form A2, but I want all three plots on the same window this time. (Hint: watch videos for M4)

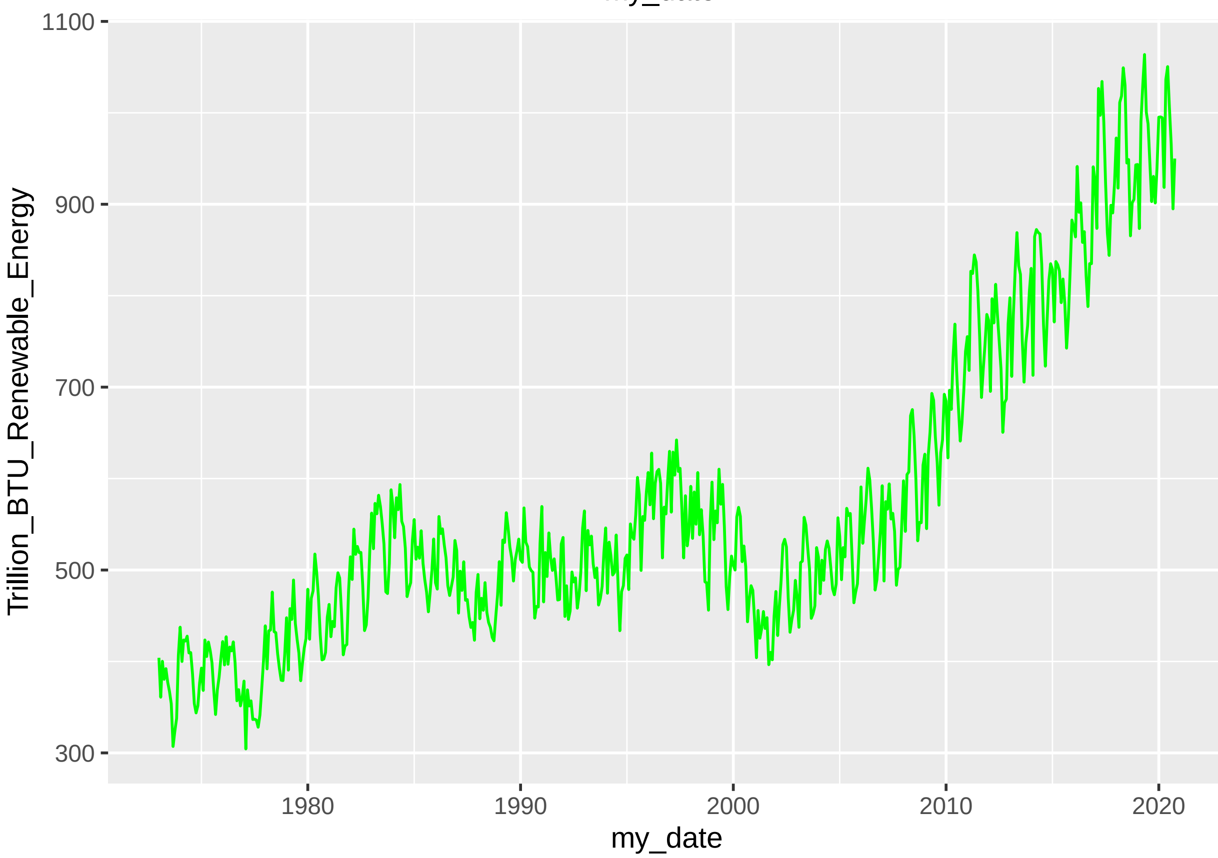
```
Table_new1 <- read.csv("../Data/table102.csv", header=TRUE)
energy <- Table_new1[,c(1,4,5,6)]

colnames(energy)=c("Month","Biomass_Energy", "Renewable_Energy", "Hydro_Power")

my_date <- paste(energy[,1])
my_date <- ym(my_date)

new_energy <- cbind(my_date,energy[,2:4])
ts_energy <- ts(new_energy[,2:4],frequency=12)

nenergy <- ncol(new_energy)-1
nobs <- nrow(new_energy)

for(i in 1:nenergy){
  par(mfrow=c(1,2))
  Acf(ts_energy[,i],lag.max=40,main=paste("Trillion_BTU_ ",colnames(new_energy)[(1+i)],sep=""))
  Pacf(ts_energy[,i],lag.max=40,main=paste("Trillion_BTU_",colnames(new_energy)[(1+i)],sep=""))
}
```

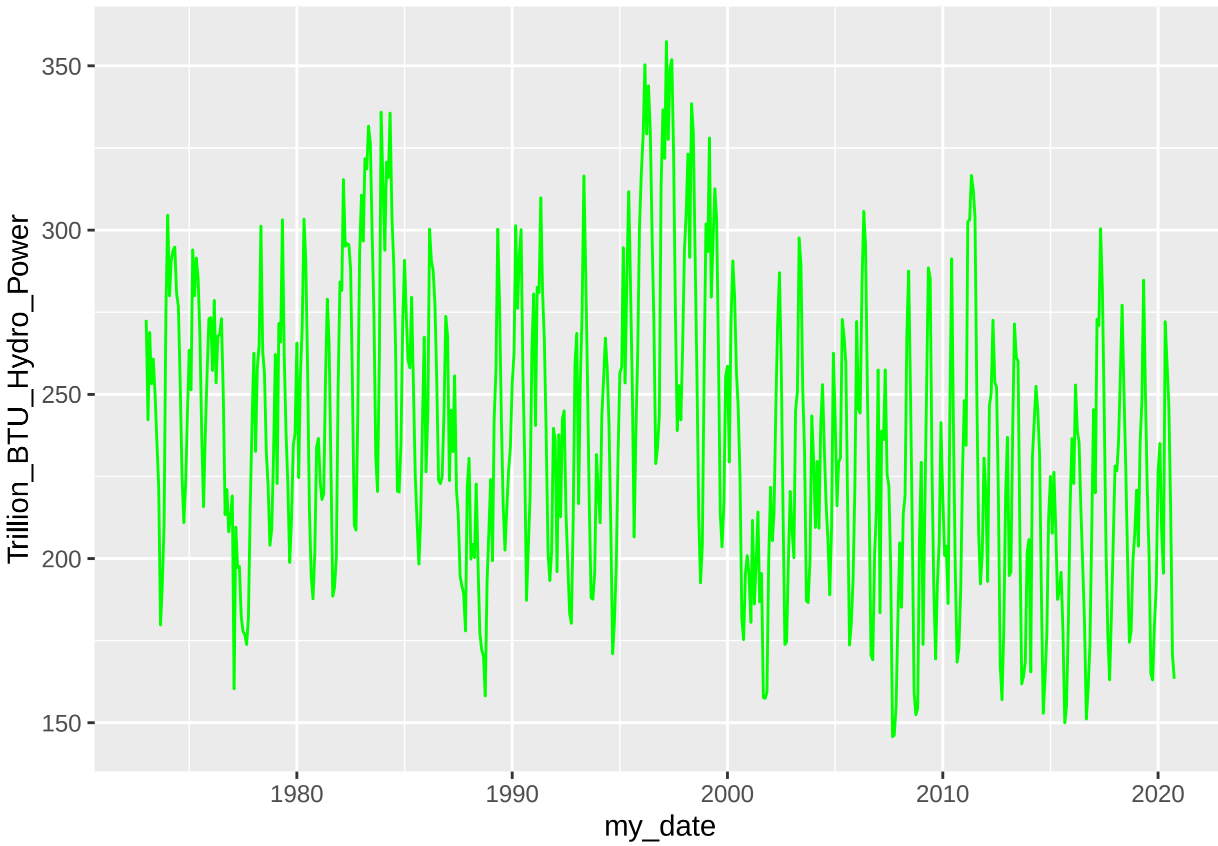





```
for(i in 1:nenergy){
print(ggplot(data=new_energy, aes(x=my_date, y=new_energy[,(1+i)])) + geom_line(color="green")+
            ylab(paste0("Trillion_BTU_",colnames(new_energy)[(1+i)],sep=""))
)
}
```


Trillion_BTU_Biomass_Energy
100
200
300
400
1980
1990
2000
2010
2020
my_date


Trillion_BTU_Renewable_Energy
300
500
700
900
1100
1980
1990
2000
2010
2020
my_date



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?

Biomass and renewable energy production have a decreasing trend component. This can be seen by observing the high correlation component between lags and the decreasing mean over time.

Hydro energy has seasonal trend component. The ACF graph depicts both negative and positive correlations coefficients which is one sign of a seasonal trend. Additionally, there still appears to be seasonal behavior in the PACF with the rise and fall in correlation.

Biomasss Stationary test: After running the Augmented Dickey-fuller test, I found a p-value of 0.7492 and I failed to rejected the null for both the Dickey-fuller thus biomass energy is not stationary, and has a unit root or a ternd.

Renewable energy stationary test: After running the Augmented Dickey-fuller test, I found a p-value of 0.7657,and I failed to rejected the null for both the Dickey-fuller thus biomass energy is not stationary, and has a unit root or a ternd.

Hydro power stationary test: After running the Augmented Dickey-fuller test, I found a p-value of 0.01 which is less than 0.05 and rejected the null that hydro power production does not have a unit root and accepted that the data is stationary. The ADF test just means there is no unit root and it is still possible to have a deterministic trend. I then ran the MannKendall test. The P-value in this tetst was less than 0.05, I failed to reject the null which is that it is stationary, thus the hydro power production is not stationary and has a deterministic trend.

```
# Checking Trends for Biomass Production
```

```r
i=1
print("Results for ADF test/n")
```

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

```r
print(adf.test(ts_energy[,i],alternative = "stationary"))
```

```
##
##  Augmented Dickey-Fuller Test
##
## data:  ts_energy[, i]
## Dickey-Fuller = -1.5962, Lag order = 8, p-value = 0.7492
## alternative hypothesis: stationary
```

```r
i=1
SMKtest <- SeasonalMannKendall(ts_energy[,i])
print(summary(SMKtest))
```

```
## Score =  9874 , Var(Score) = 150368.7
## denominator =  13442
## tau = 0.735, 2-sided pvalue =< 2.22e-16
## NULL
```

```r
#Checking Trends for Renewable production

i=2
print("Results for ADF test/n")
```

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

```r
print(adf.test(ts_energy[,i],alternative = "stationary"))
```

```
##
##  Augmented Dickey-Fuller Test
##
## data:  ts_energy[, i]
## Dickey-Fuller = -1.5574, Lag order = 8, p-value = 0.7657
## alternative hypothesis: stationary
```

```r
i=2
SMKtest <- SeasonalMannKendall(ts_energy[,i])
print(summary(SMKtest))
```

```
## Score =  9476 , Var(Score) = 150368.7
## denominator =  13442
## tau = 0.705, 2-sided pvalue =< 2.22e-16
## NULL
```

```r
#Checking Trends for Hydro Power production
i=3
SMKtest <- SeasonalMannKendall(ts_energy[,i])
print(summary(SMKtest))
```

```
## Score =  -3880 , Var(Score) = 150368.7
## denominator =  13442
## tau = -0.289, 2-sided pvalue =< 2.22e-16
## NULL
```

```
i=3
print("Results for ADF test/n")
```

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

```
print(adf.test(ts_energy[,i],alternative = "stationary"))
```

```
## Warning in adf.test(ts_energy[, i], alternative = "stationary"): p-value smaller
## than printed p-value
```

```
##
##  Augmented Dickey-Fuller Test
##
## data:  ts_energy[, i]
## Dickey-Fuller = -4.9481, Lag order = 8, p-value = 0.01
## alternative hypothesis: stationary
```

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.

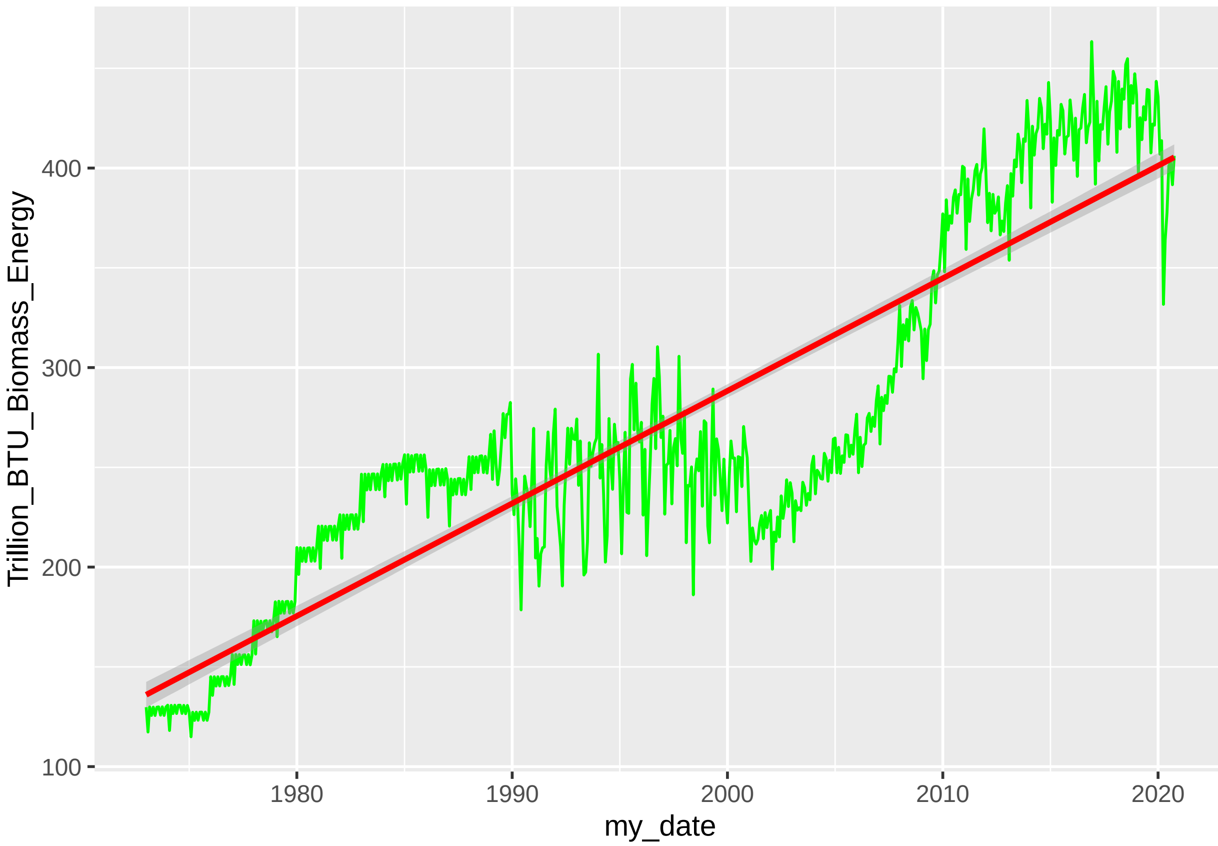
Biomass Energy–Adjusted R-squared: 0.7962, slope= 4.702e-01 intercept=135.5 The r squared is significant and the biomass energy plot has a significant linear trend component. This corresponds with the stationary tests ran in Q2.

Renewable Energy–Adjusted R-squared:0.6887 slope= 0.8429 intercept= 330.37156 The r squared is very close to significant and the biomass energy appears to have a linear trend component. This corresponds with the stationary tests ran in Q2.

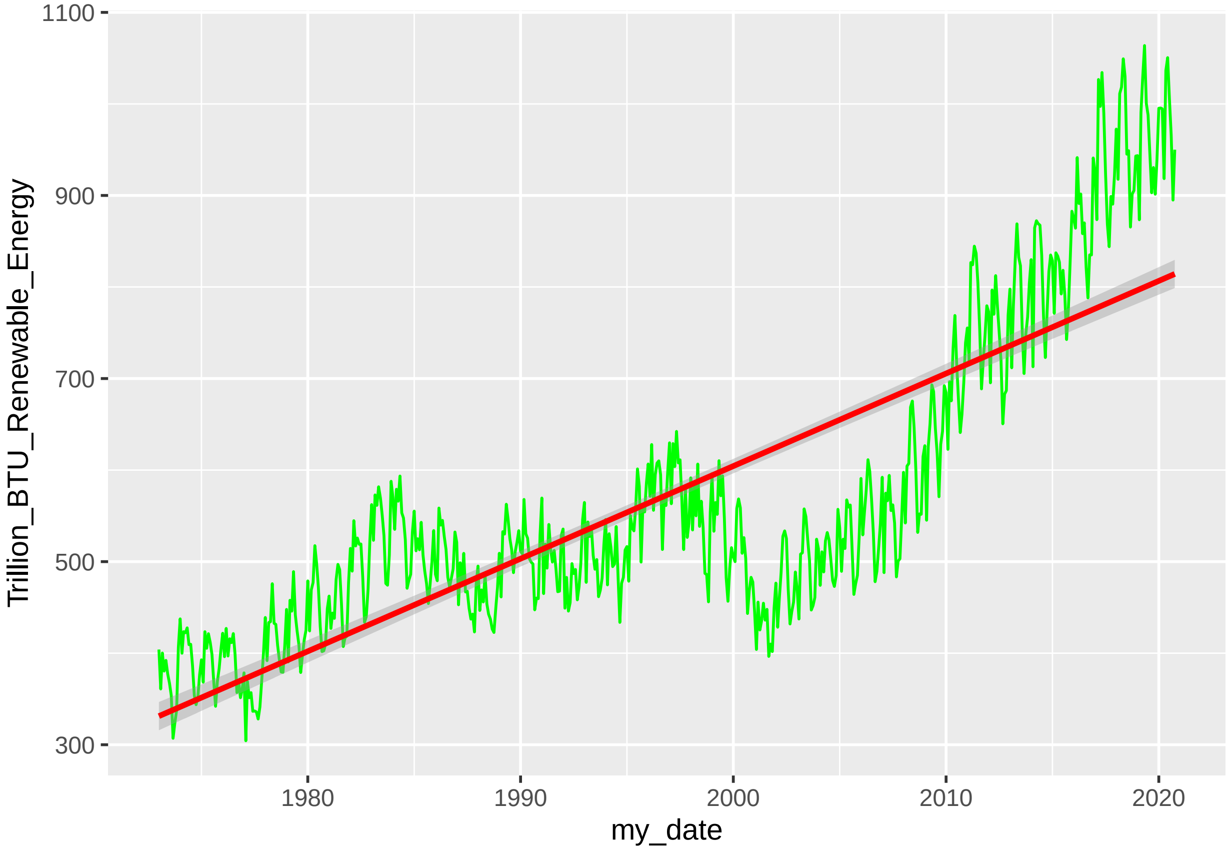
Hydro Power Production –Adjusted R-squared:0.07528 slope=-0.07341 intercept= 258.05622. The r squared is insignificant and there is little if linear trend in the hydro power production series.

```
for(i in 1:nenergy){
t <- c(1:nobs)

linear_trend_model=lm(new_energy[,i+1]~t)
summary(linear_trend_model)
beta0=as.numeric(linear_trend_model$coefficients[1])
beta1=as.numeric(linear_trend_model$coefficients[2]) }

for(i in 1:nenergy){
print(ggplot(data=new_energy, aes(x=my_date, y=new_energy[,(1+i)])) + geom_line(color="green")+
            ylab(paste0("Trillion_BTU_",colnames(new_energy)[(1+i)],sep="")) + geom_smooth(color="red",
)
}
```

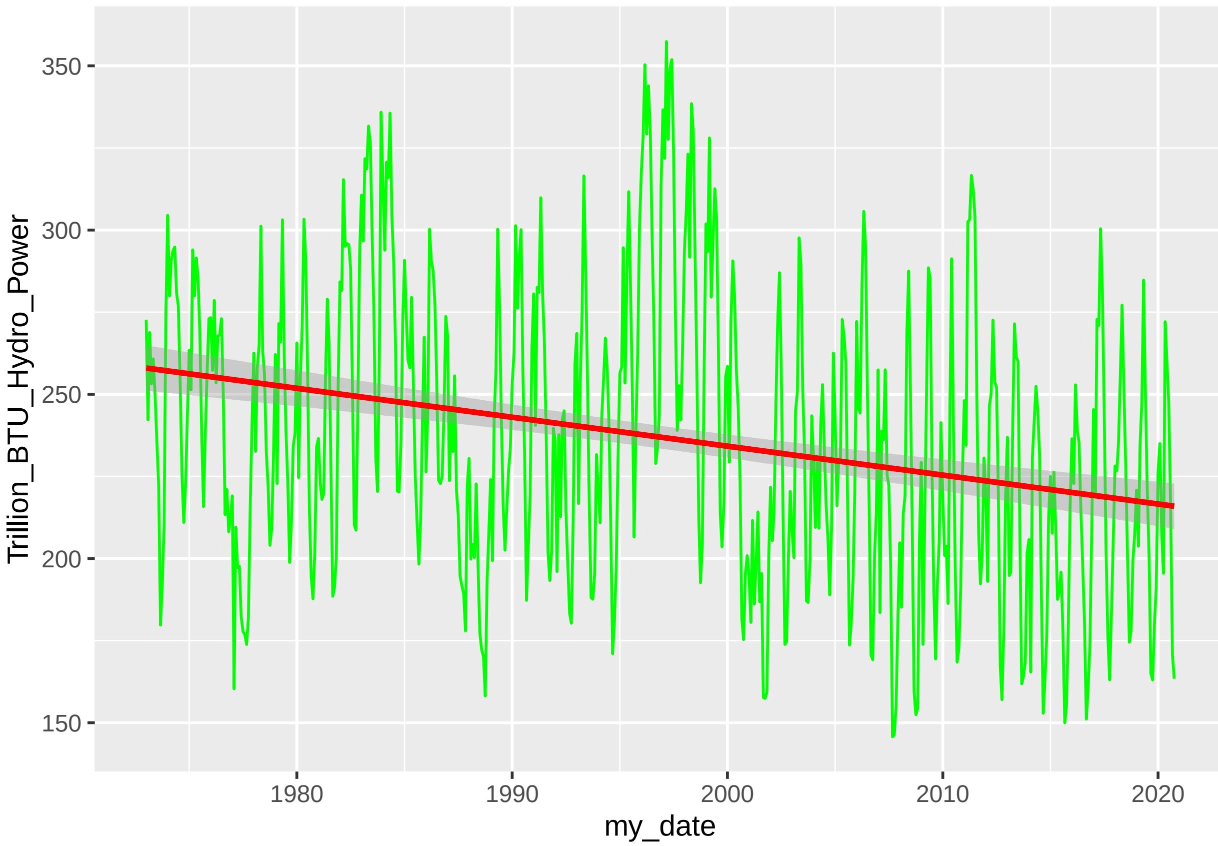
```
## `geom_smooth()` using formula 'y ~ x'
```




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




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




```
t <- c(1:nobs)
i=1

linear_trend_model=lm(new_energy[,i+1]~t)
summary(linear_trend_model)
```

```
##
## Call:
## lm(formula = new_energy[, i + 1] ~ t)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -101.149  -25.456    4.985   33.353   79.634
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 1.355e+02  3.296e+00   41.11   <2e-16 ***
## t           4.702e-01  9.934e-03   47.33   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 39.44 on 572 degrees of freedom
## Multiple R-squared:  0.7966, Adjusted R-squared:  0.7962
## F-statistic:  2240 on 1 and 572 DF,  p-value: < 2.2e-16
```

```
beta0=as.numeric(linear_trend_model$coefficients[1])
beta1=as.numeric(linear_trend_model$coefficients[2])
```

```
i=2

linear_trend_model=lm(new_energy[,i+1]~t)
summary(linear_trend_model)
```

```
##
## Call:
## lm(formula = new_energy[, i + 1] ~ t)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -224.735  -55.673    5.418   60.453  263.849
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 330.37156    7.86270   42.02   <2e-16 ***
## t             0.84299    0.02369   35.58   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 94.07 on 572 degrees of freedom
## Multiple R-squared:  0.6887, Adjusted R-squared:  0.6882
## F-statistic:  1266 on 1 and 572 DF,  p-value: < 2.2e-16
```

```
beta0=as.numeric(linear_trend_model$coefficients[1])
beta1=as.numeric(linear_trend_model$coefficients[2])


i=3

linear_trend_model=lm(new_energy[,i+1]~t)
summary(linear_trend_model)
```

```
##
## Call:
## lm(formula = new_energy[, i + 1] ~ t)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -94.06 -31.57  -1.63  27.73 120.69
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 258.05622    3.52899  73.125  < 2e-16 ***
## t            -0.07341    0.01063  -6.903 1.36e-11 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 42.22 on 572 degrees of freedom
## Multiple R-squared:  0.07689,    Adjusted R-squared:  0.07528
## F-statistic: 47.64 on 1 and 572 DF,  p-value: 1.361e-11
```

```
beta0=as.numeric(linear_trend_model$coefficients[1])
beta1=as.numeric(linear_trend_model$coefficients[2])
```

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?

The detrend plots for renewable energy and biomass energy no longer have an increasing mean. The linear fit has removed the trend component in both the biomass energy and renewable energy plots.

The hydro power series changed very little but the small decreasing mean has been removed from the hydro detrend series.

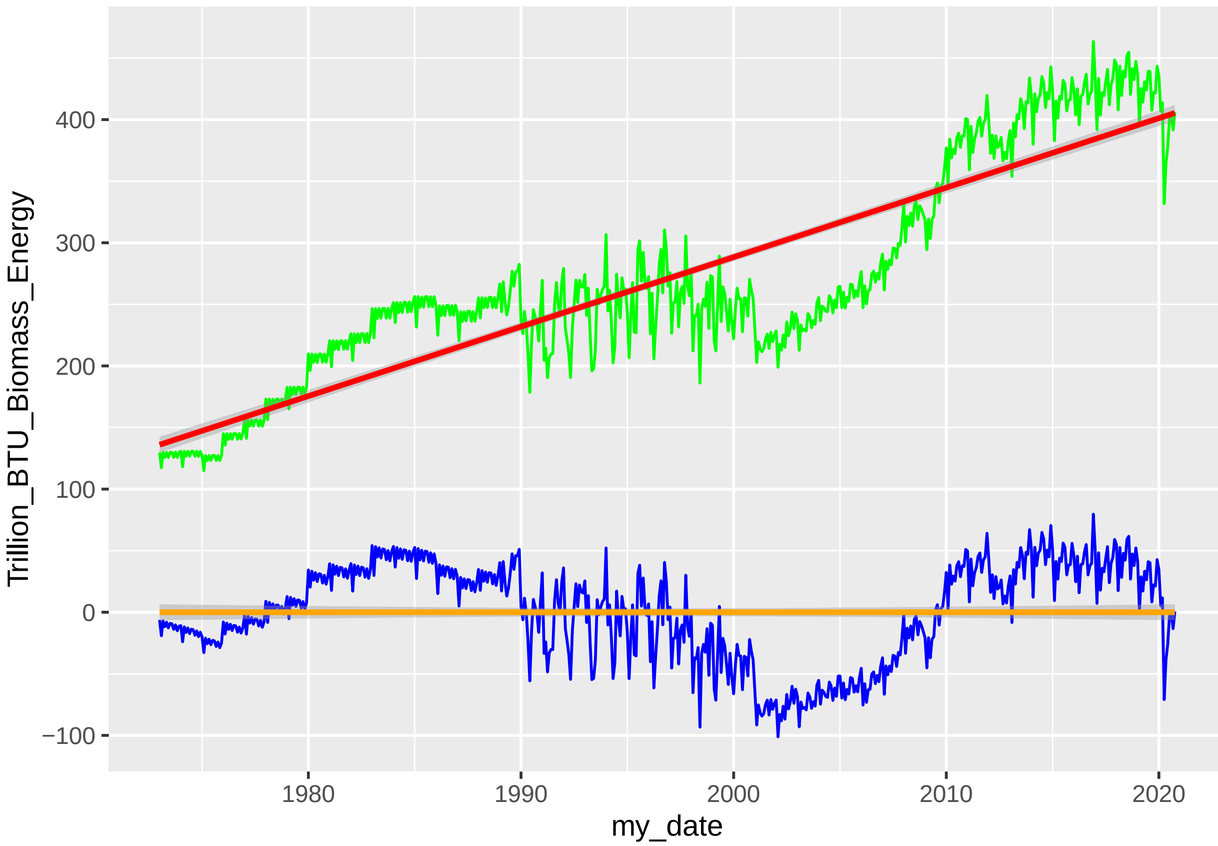
The detrend series is now flat and oscillates about zero in all three series. The slope of the detrend linear fit line is at zero in all three of the series.

```
#Biomass Energy Production Detrend
t <- c(1:nobs)
i=1

linear_trend_model=lm(new_energy[,i+1]~t)
summary(linear_trend_model)
```

```
##
## Call:
## lm(formula = new_energy[, i + 1] ~ t)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -101.149  -25.456    4.985   33.353   79.634
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 1.355e+02  3.296e+00   41.11   <2e-16 ***
## t           4.702e-01  9.934e-03   47.33   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 39.44 on 572 degrees of freedom
## Multiple R-squared:  0.7966, Adjusted R-squared:  0.7962
## F-statistic:  2240 on 1 and 572 DF,  p-value: < 2.2e-16
```

```
beta0=as.numeric(linear_trend_model$coefficients[1])
beta1=as.numeric(linear_trend_model$coefficients[2])


#linear trend for biomass energy
detrend_bio_energy <- new_energy[,(i+1)]-(beta0+beta1*t)


ggplot(data=new_energy, aes(x=my_date, y=new_energy[,(1+i)])) + geom_line(color="green")+
            ylab(paste0("Trillion_BTU_",colnames(new_energy)[(1+i)],sep="")) + geom_smooth(color="red",
```

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

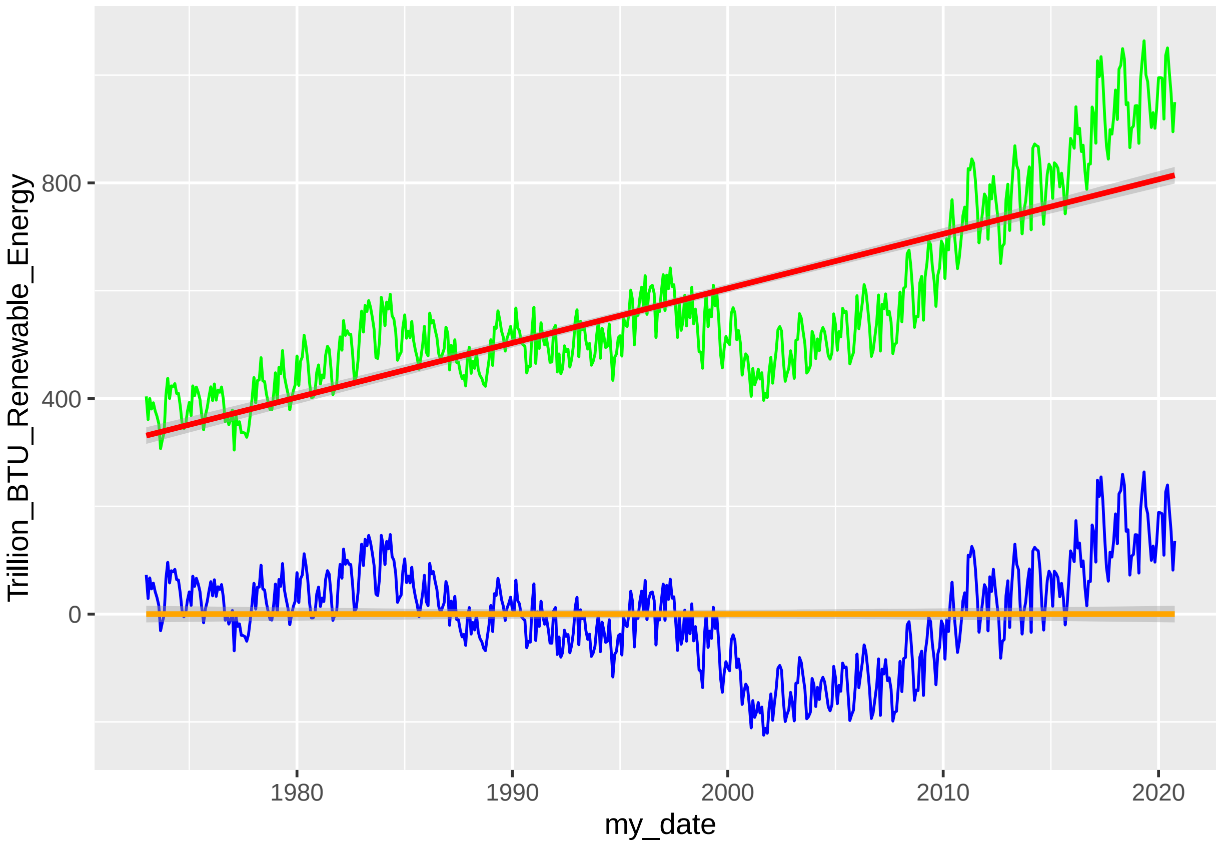



```
#Renewable Energy Production Detrend

t <- c(1:nobs)
i=2

linear_trend_model=lm(new_energy[,i+1]~t)
summary(linear_trend_model)
```

```
##
## Call:
## lm(formula = new_energy[, i + 1] ~ t)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -224.735  -55.673    5.418   60.453  263.849
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 330.37156    7.86270   42.02   <2e-16 ***
## t             0.84299    0.02369   35.58   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 94.07 on 572 degrees of freedom
## Multiple R-squared:  0.6887, Adjusted R-squared:  0.6882
## F-statistic:  1266 on 1 and 572 DF,  p-value: < 2.2e-16
```

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


detrend_renew_energy <- new_energy[,(i+1)]-(beta0+beta1*t)

ggplot(data=new_energy, aes(x=my_date, y=new_energy[,(1+i)])) + geom_line(color="green")+
            ylab(paste0("Trillion_BTU_",colnames(new_energy)[(1+i)],sep="")) + geom_smooth(color="red",m
```

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



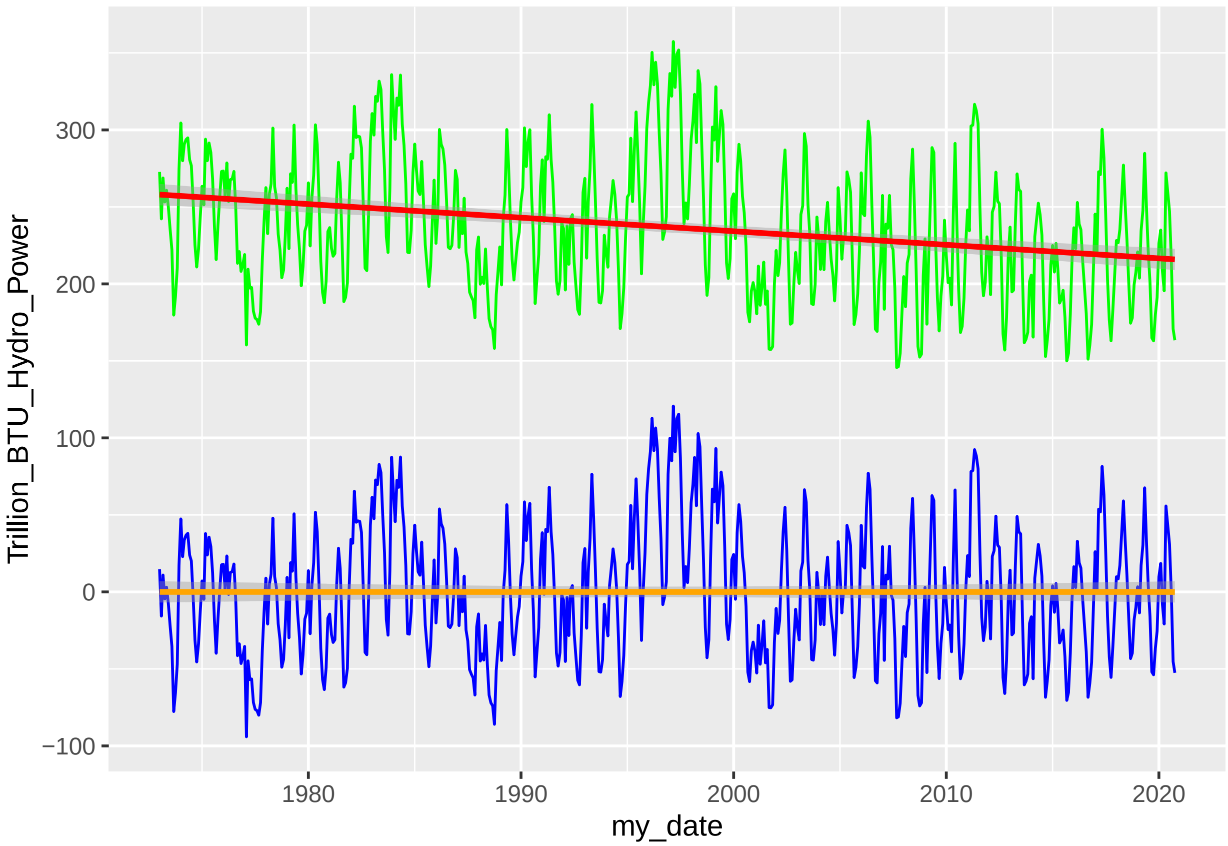

```r
#Hydro Power Production Detrend

t <- c(1:nobs)
i=3

linear_trend_model=lm(new_energy[,i+1]~t)
summary(linear_trend_model)
```

```
##
## Call:
## lm(formula = new_energy[, i + 1] ~ t)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -94.06 -31.57  -1.63  27.73 120.69
```

```
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 258.05622    3.52899  73.125  < 2e-16 ***
## t            -0.07341    0.01063  -6.903 1.36e-11 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 42.22 on 572 degrees of freedom
## Multiple R-squared:  0.07689,    Adjusted R-squared:  0.07528
## F-statistic: 47.64 on 1 and 572 DF,  p-value: 1.361e-11
```

```
beta0=as.numeric(linear_trend_model$coefficients[1])
beta1=as.numeric(linear_trend_model$coefficients[2])


detrend_hydro_energy <- new_energy[,(i+1)]-(beta0+beta1*t)

ggplot(data=new_energy, aes(x=my_date, y=new_energy[,(1+i)])) + geom_line(color="green")+
            ylab(paste0("Trillion_BTU_",colnames(new_energy)[(1+i)],sep="")) + geom_smooth(color="red",m
```

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

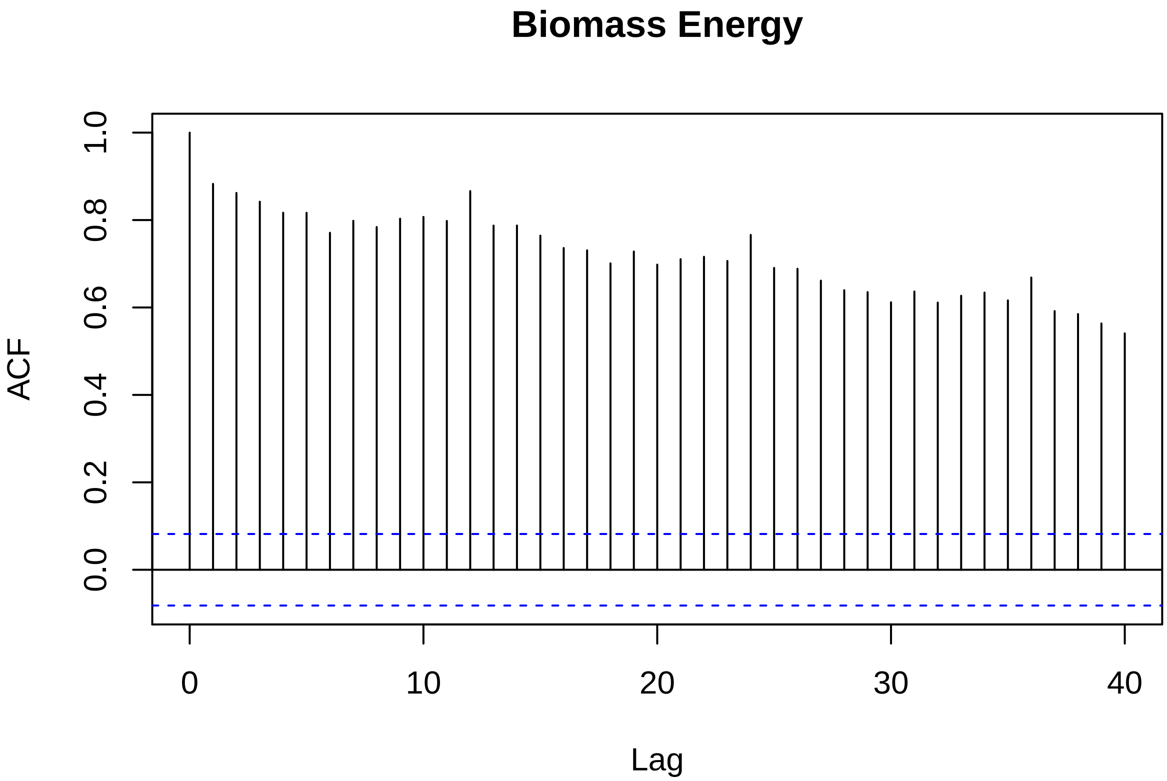



```
### Q5
Plot ACF and PACF for the detrended series and compare with the plots from Q1. Did the plots change? How
```

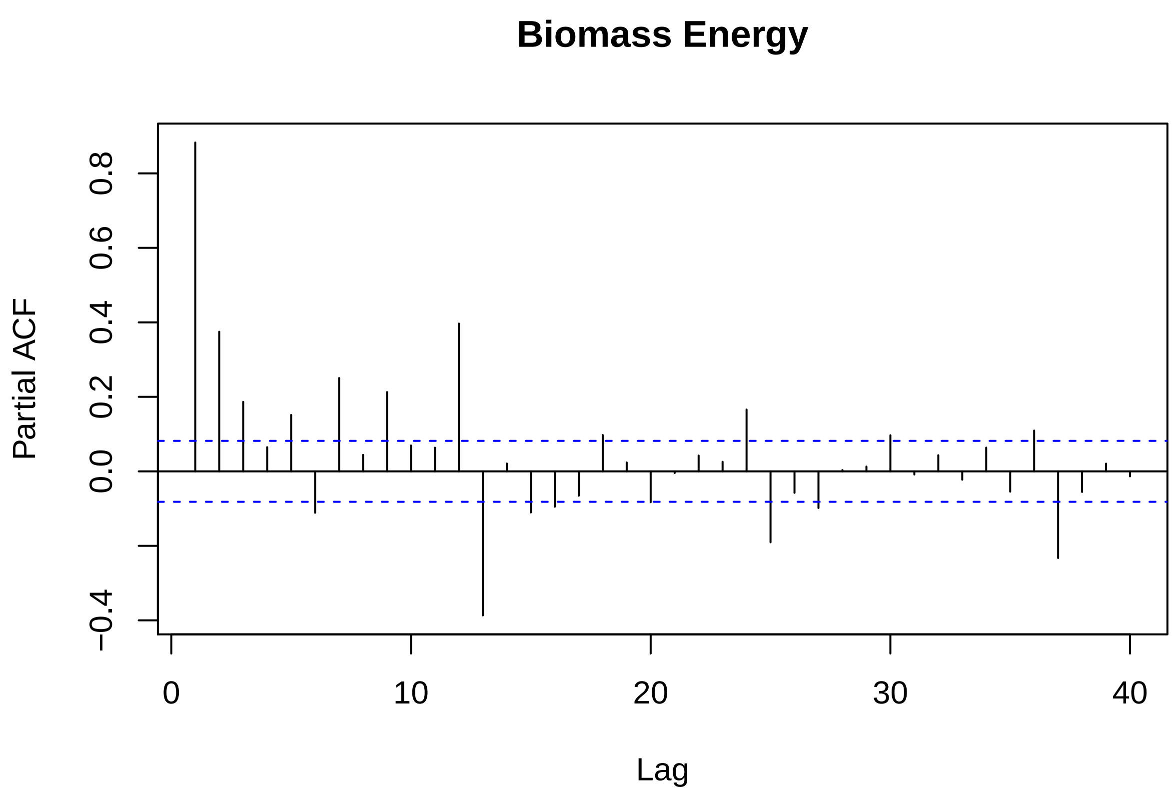
The ACF charts for renewable energy and biomass energy have a stronger correlation across lags. While th

The detrend series for hydro power remains unchanged from Q1 and the detrend series seems to have very s

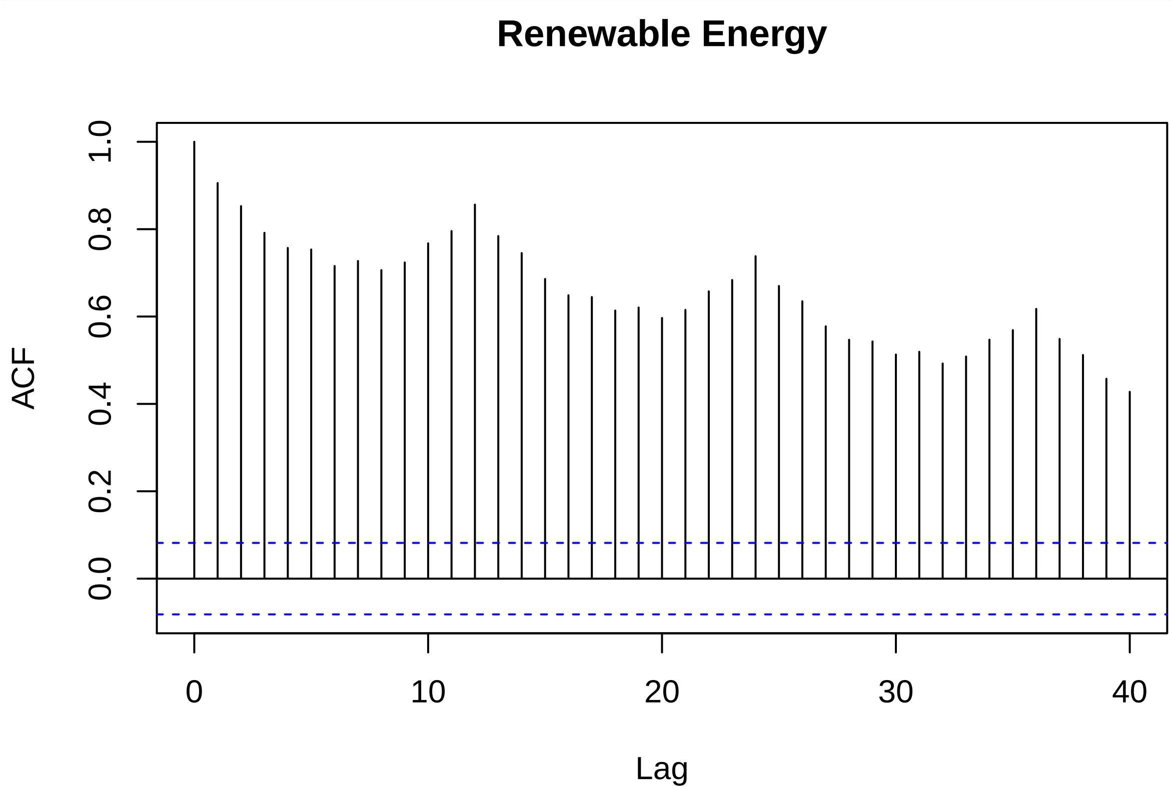
```r
ts_acf_bio=acf(detrend_bio_energy,lag.max=40, plot = TRUE, main="Biomass Energy")
```



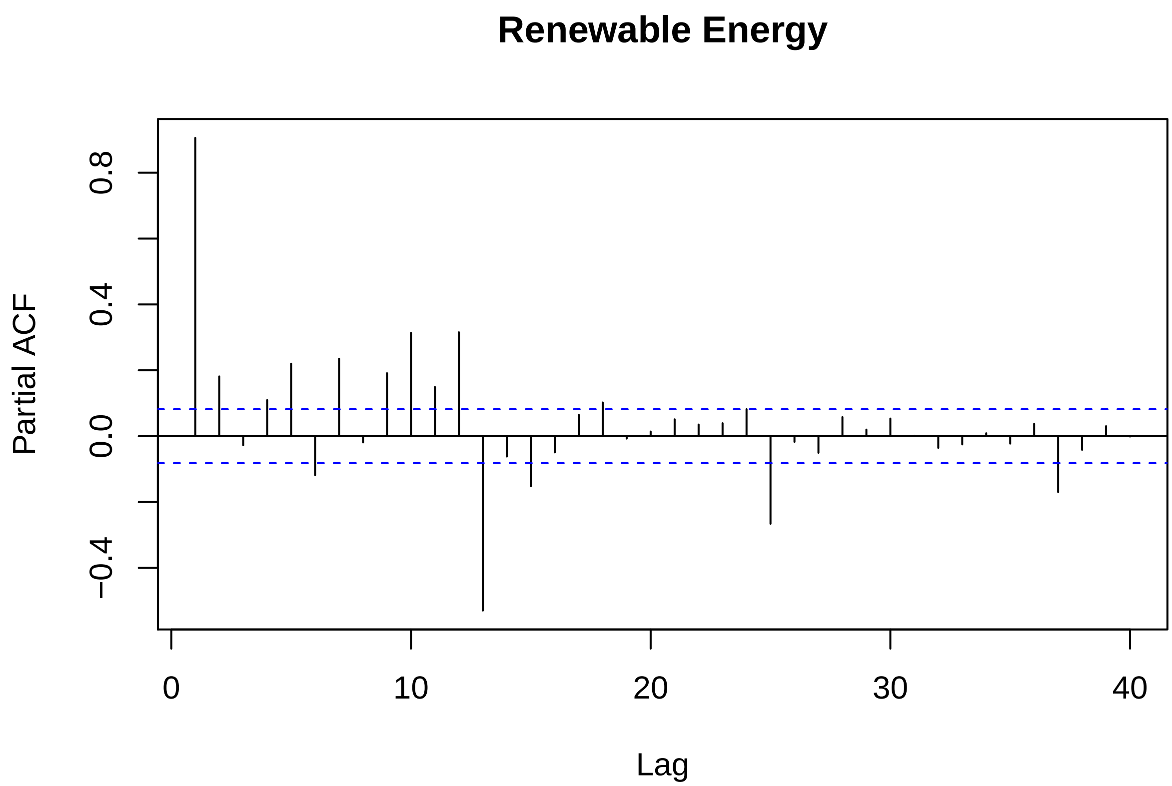

```r
ts_pacf_bio=pacf(detrend_bio_energy,lag.max=40, plot = TRUE, main="Biomass Energy")
```



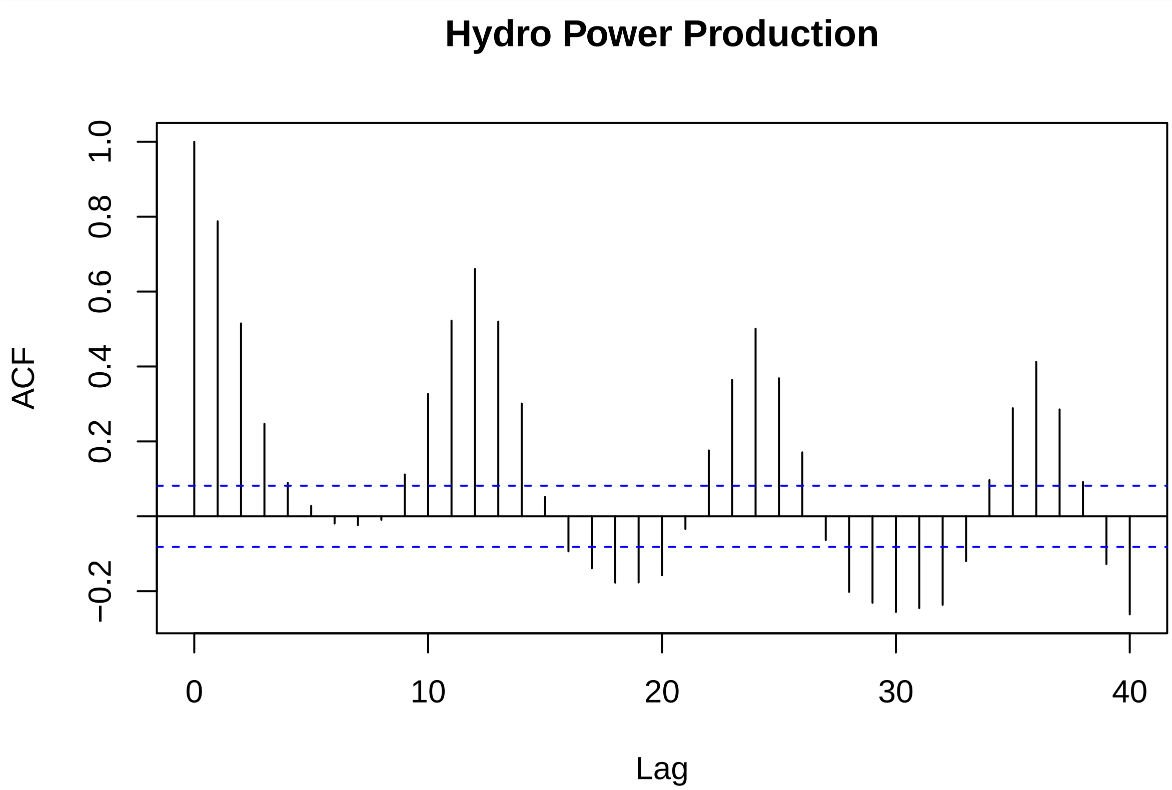

```
ts_acf_renew=acf(detrend_renew_energy,lag.max=40, plot = TRUE, main="Renewable Energy")
```



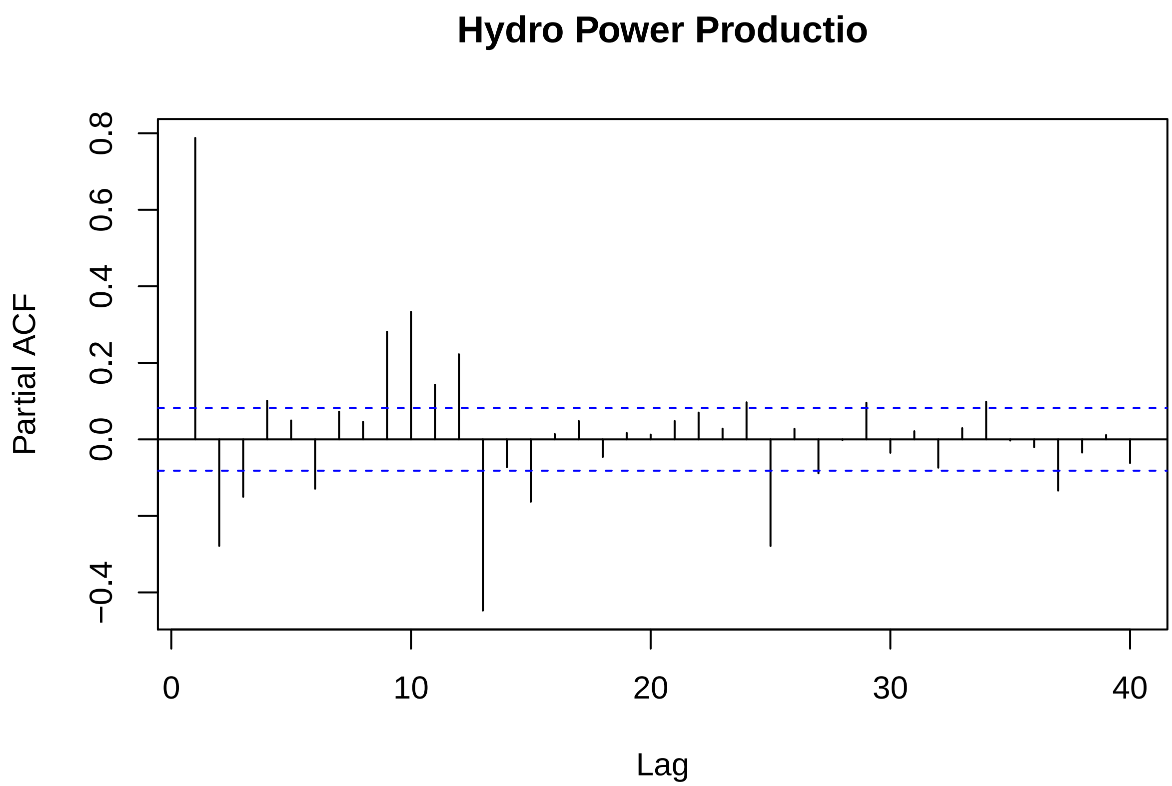

```
ts_pacf_renew=pacf(detrend_renew_energy,lag.max=40, plot = TRUE, main="Renewable Energy")
```




```
ts_acf_hydro=acf(detrend_hydro_energy,lag.max=40, plot = TRUE, main="Hydro Power Production")
```




```
ts_pacf_hydro=pacf(detrend_hydro_energy,lag.max=40, plot = TRUE, main="Hydro Power Productio")
```



Seasonal Component

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

Q6

Do the series seem to have a seasonal trend? Which serie/series? Use function *lm()* to fit a seasonal means model to this/these time series. Ask R to print the summary of the regression. Interpret the regression output. Save the regression coefficients for further analysis.

The series that has a seasonal trend is hydro power production. In looking at the graphs it is clear that the seasonal component which is the mean value for each month fits right over the hydro power production values, which leds me to conclude that the seasonal component exists for hydro power production. The R square value is 0.4234, while is is less than 0.7 a signifcant correlation factor it is still high, and P value was 2.2e -16.

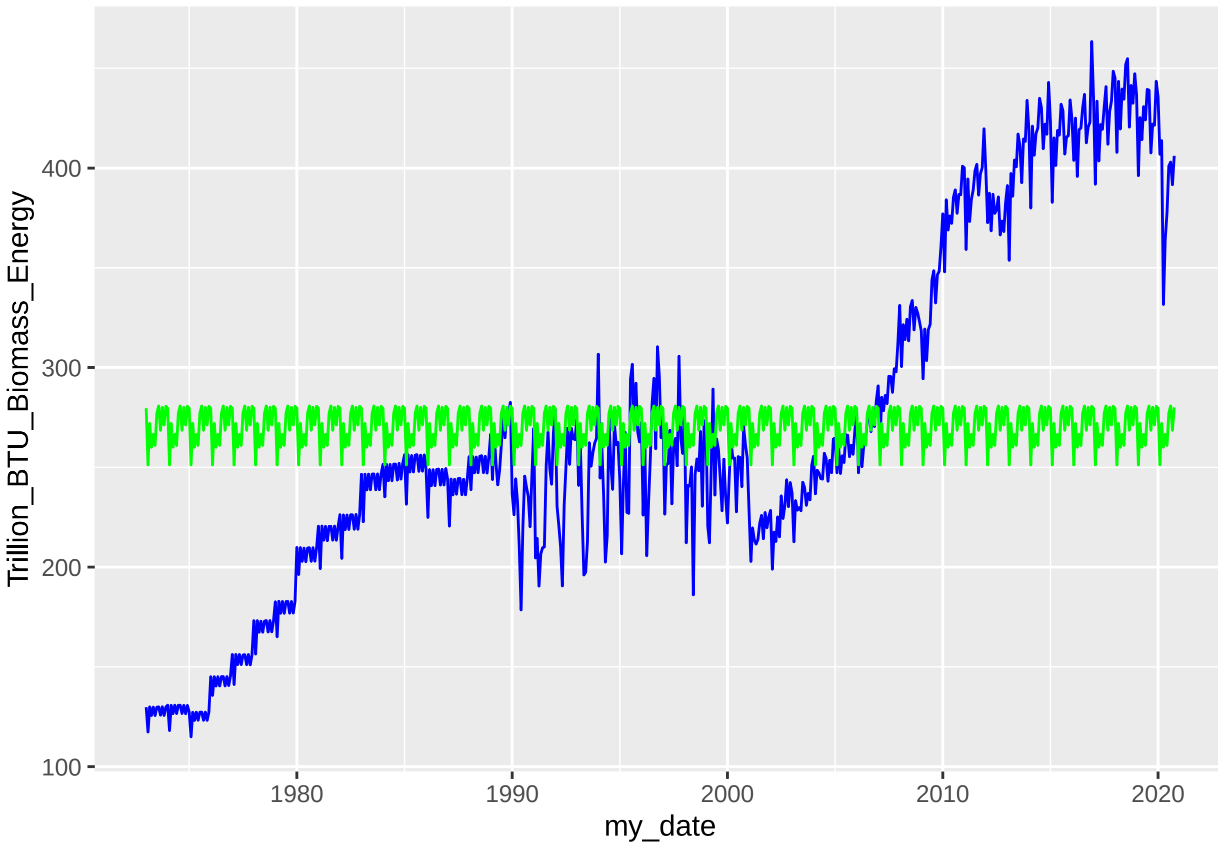
The biomass energy production and renewable energy production seem to have very little seasonal trend. The regression output for biomass production was -0.008199 and for renewable energy production it was 0.1351.

```r
# BioMass Energy
i=1
dummies <- seasonaldummy(ts_energy[,i])

seas_means_model=lm(new_energy[,(i+1)]~dummies)
summary(seas_means_model)
```

```
##
## Call:
## lm(formula = new_energy[, (i + 1)] ~ dummies)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
```

```
## -153.47  -50.56  -20.25   52.13  182.84
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 280.5693    12.7954  21.927   <2e-16 ***
## dummiesJan   -1.0039    18.0009  -0.056    0.956
## dummiesFeb  -29.3891    18.0009  -1.633    0.103
## dummiesMar   -8.6090    18.0009  -0.478    0.633
## dummiesApr  -20.5046    18.0009  -1.139    0.255
## dummiesMay  -14.0960    18.0009  -0.783    0.434
## dummiesJun  -19.5548    18.0009  -1.086    0.278
## dummiesJul   -3.4306    18.0009  -0.191    0.849
## dummiesAug    0.2220    18.0009   0.012    0.990
## dummiesSep  -11.9821    18.0009  -0.666    0.506
## dummiesOct   -0.5379    18.0009  -0.030    0.976
## dummiesNov   -9.3753    18.0954  -0.518    0.605
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 87.72 on 562 degrees of freedom
## Multiple R-squared:  0.01116,    Adjusted R-squared:  -0.008199
## F-statistic: 0.5764 on 11 and 562 DF,  p-value: 0.8486
```

```
beta_int=seas_means_model$coefficients[1]
beta_coeff=seas_means_model$coefficients[2:12]

#compute seasonal component
bio_seas_comp=array(0,nobs)
for(i in 1:nobs){
  bio_seas_comp[i]=(beta_int+beta_coeff%*%dummies[i,])
}

ggplot(data=new_energy, aes(x=my_date, y=new_energy[,2])) + geom_line(color="blue")+
            ylab(paste0("Trillion_BTU_",colnames(new_energy)[(2)],sep="")) +  geom_line(aes(y=bio_seas_c
```

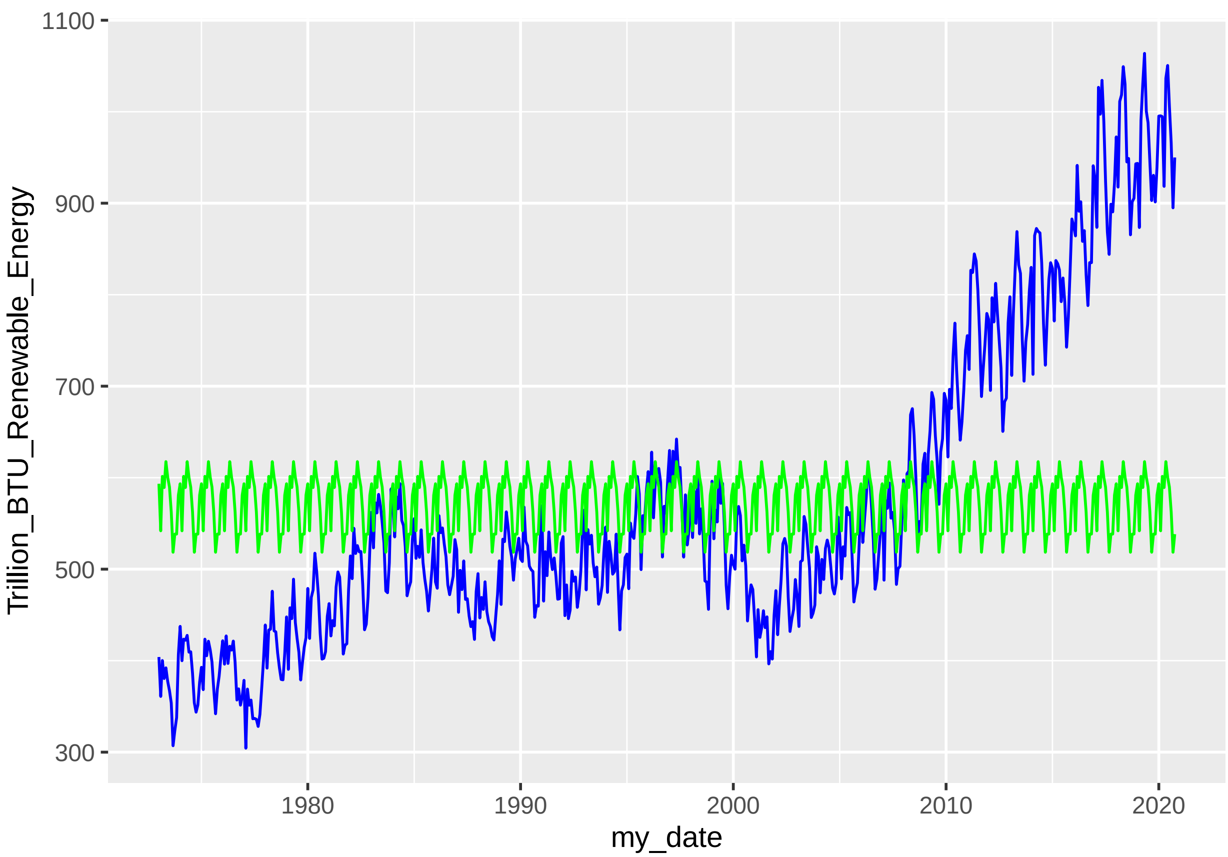



```r
# Renewable Energy
i=2
dummies <- seasonaldummy(ts_energy[,i])

seas_means_model=lm(new_energy[,(i+1)]~dummies)
summary(seas_means_model)
```

```
##
## Call:
## lm(formula = new_energy[, (i + 1)] ~ dummies)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -263.99 -102.98  -52.33   36.68  453.58
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  580.912     24.406  23.802   <2e-16 ***
## dummiesJan    12.451     34.335   0.363   0.7170
## dummiesFeb   -38.964     34.335  -1.135   0.2569
## dummiesMar    20.515     34.335   0.597   0.5504
## dummiesApr     8.294     34.335   0.242   0.8092
## dummiesMay    36.628     34.335   1.067   0.2865
## dummiesJun    19.560     34.335   0.570   0.5691
## dummiesJul     8.863     34.335   0.258   0.7964
## dummiesAug   -18.480     34.335  -0.538   0.5906
## dummiesSep   -62.410     34.335  -1.818   0.0696 .
```

```
## dummiesOct    -42.649     34.335  -1.242   0.2147
## dummiesNov    -42.516     34.515  -1.232   0.2185
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 167.3 on 562 degrees of freedom
## Multiple R-squared:  0.03244,    Adjusted R-squared:  0.01351
## F-statistic: 1.713 on 11 and 562 DF,  p-value: 0.06702
```

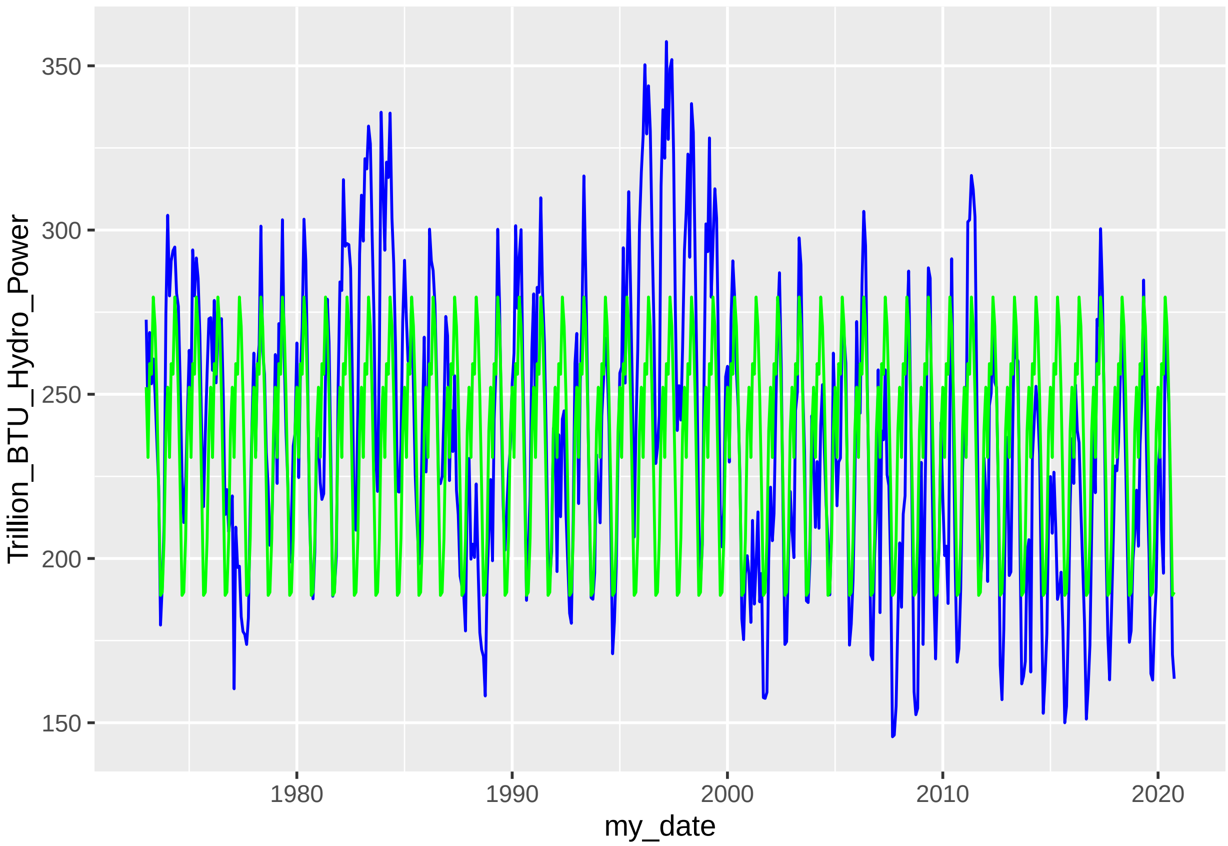
```
beta_int=seas_means_model$coefficients[1]
beta_coeff=seas_means_model$coefficients[2:12]

#compute seasonal component
renew_seas_comp=array(0,nobs)
for(i in 1:nobs){
  renew_seas_comp[i]=(beta_int+beta_coeff%*%dummies[i,])
}

ggplot(data=new_energy, aes(x=my_date, y=new_energy[,3])) + geom_line(color="blue")+
            ylab(paste0("Trillion_BTU_",colnames(new_energy)[(3)],sep="")) +  geom_line(aes(y=renew_seas
```




```
# Hydro Power Production
i=3

dummies <- seasonaldummy(ts_energy[,i])

seas_means_model=lm(new_energy[,(i+1)]~dummies)
summary(seas_means_model)
```

```
##
## Call:
## lm(formula = new_energy[, (i + 1)] ~ dummies)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -92.064 -22.897  -2.654  20.642  98.058
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  238.887      4.863  49.125  < 2e-16 ***
## dummiesJan    13.270      6.841   1.940  0.05291 .
## dummiesFeb    -8.133      6.841  -1.189  0.23499
## dummiesMar    20.442      6.841   2.988  0.00293 **
## dummiesApr    17.199      6.841   2.514  0.01221 *
## dummiesMay    40.726      6.841   5.953 4.64e-09 ***
## dummiesJun    31.764      6.841   4.643 4.28e-06 ***
## dummiesJul    10.858      6.841   1.587  0.11306
## dummiesAug   -17.907      6.841  -2.618  0.00909 **
## dummiesSep   -50.121      6.841  -7.326 8.26e-13 ***
## dummiesOct   -49.165      6.841  -7.187 2.12e-12 ***
## dummiesNov   -32.757      6.877  -4.763 2.43e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 33.34 on 562 degrees of freedom
## Multiple R-squared:  0.4345, Adjusted R-squared:  0.4234
## F-statistic: 39.25 on 11 and 562 DF,  p-value: < 2.2e-16
```

```
beta_int=seas_means_model$coefficients[1]
beta_coeff=seas_means_model$coefficients[2:12]

#compute seasonal component
hydro_seas_comp=array(0,nobs)
for(i in 1:nobs){
  hydro_seas_comp[i]=(beta_int+beta_coeff%*%dummies[i,])
}

ggplot(data=new_energy, aes(x=my_date, y=new_energy[,4])) + geom_line(color="blue")+
            ylab(paste0("Trillion_BTU_",colnames(new_energy)[(4)],sep="")) +  geom_line(aes(y=hydro_seas
```



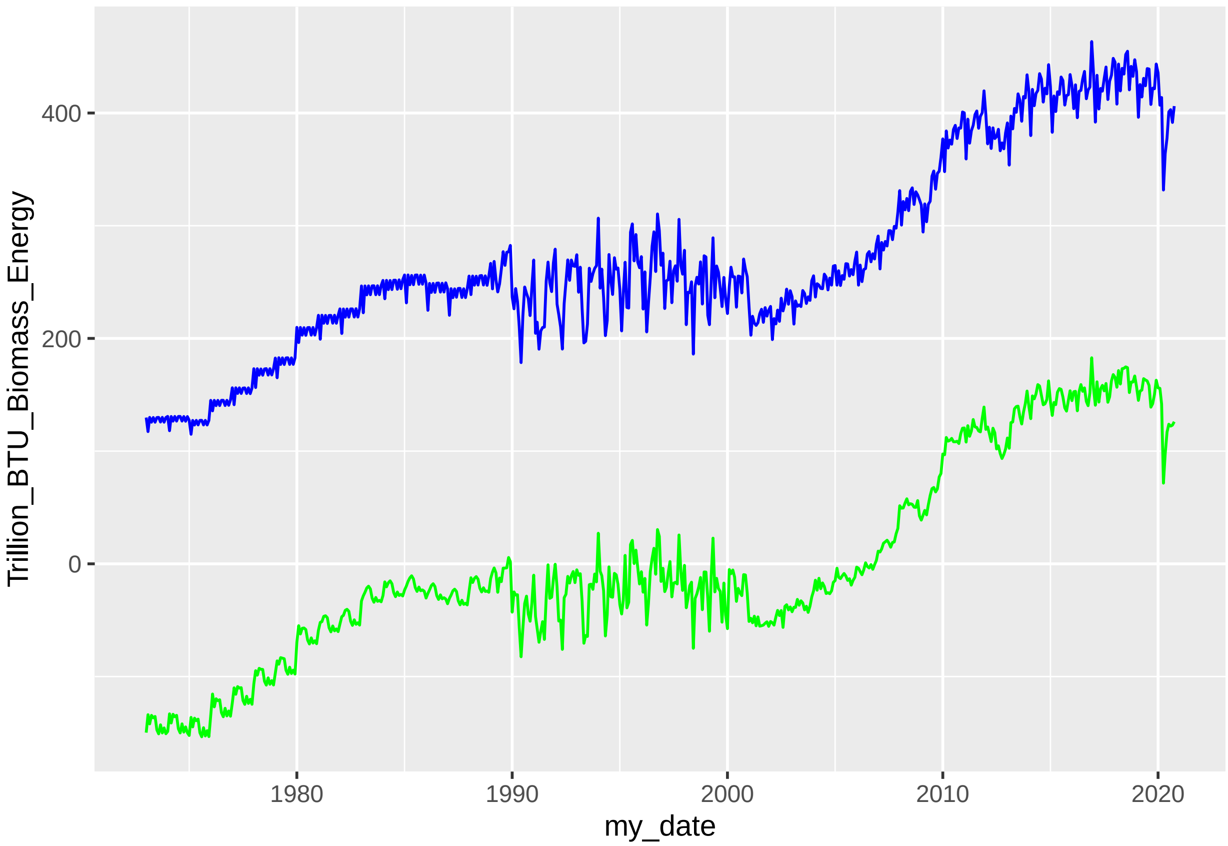
Q7

Use the regression coefficients from Q6 to deseason the series. Plot the deseason series and compare with the plots from part Q1. Did anything change?

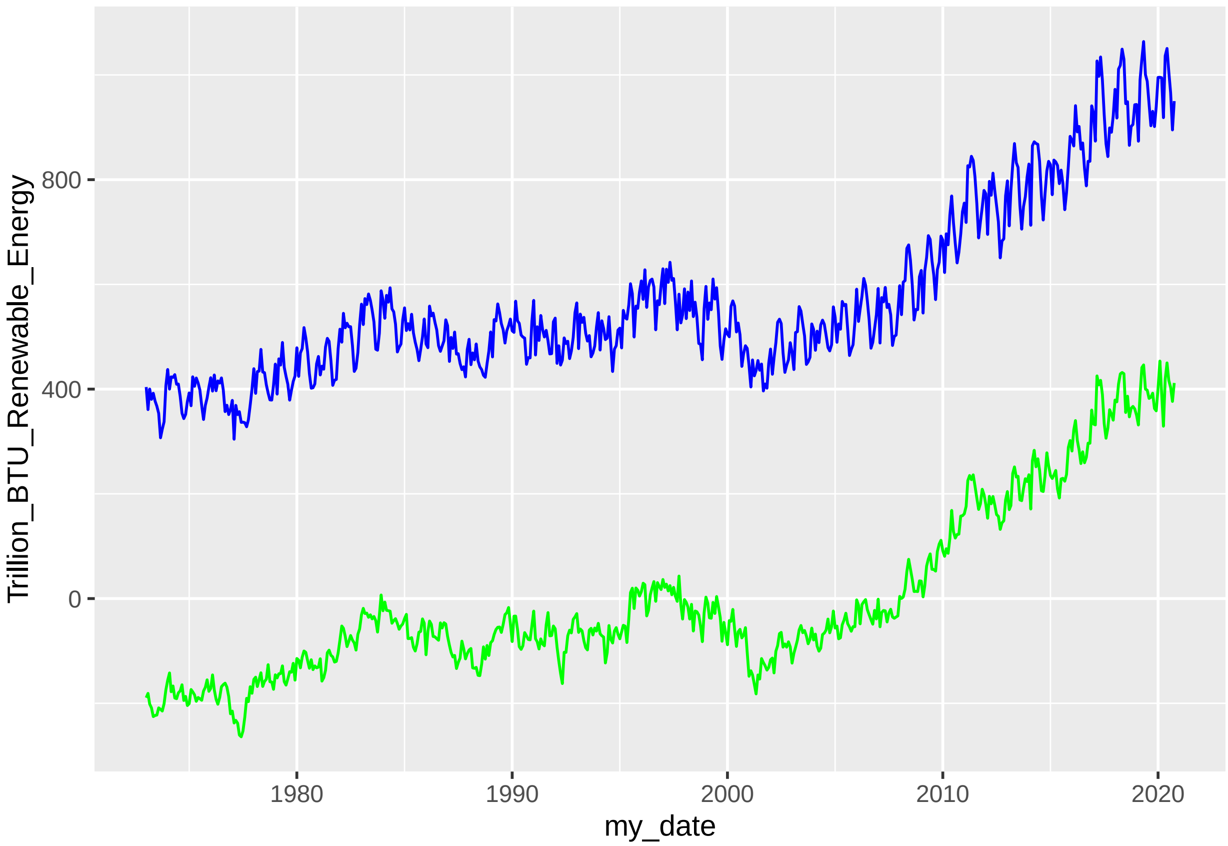
The seasonal means model depicted in green is shifted below the data sets. The seasonal means model is a calculated by subtracting the monthly means, and that is why the mean is closer to zero.For hydro power production in particular which had the greatest seasonal variability the deseason series is less variable than the original data set as the seasonality is removed and it is just random variability.

To confirm this I ran the Mann Kendall for a deterministic trend, to check for seasonal trend. The score is very negative. The P-value is less than 0.05 which led me to reject the null which is that it is stationary, thus the hydro power production is not stationary and follows a seasonal trend.

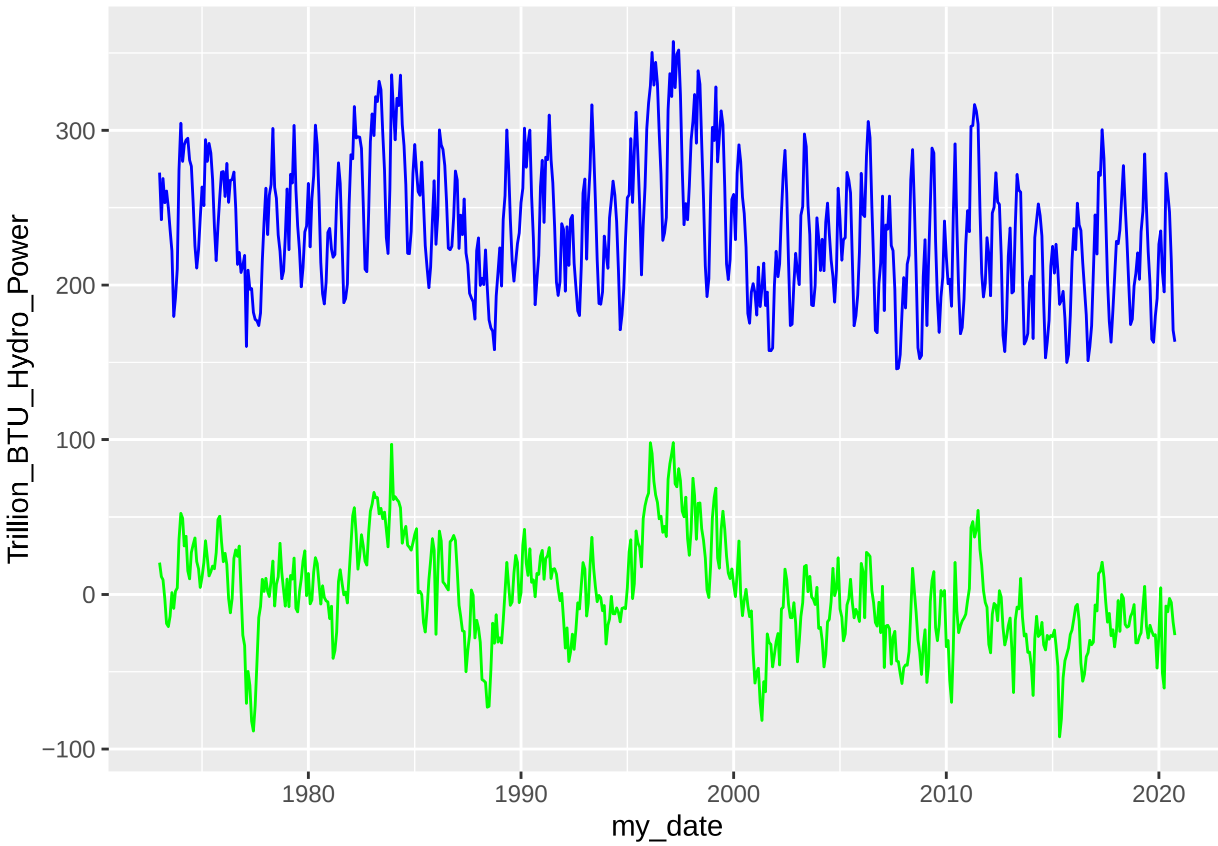
```
#Biomass Energy Deseason Graph
i=1

bio_deseason_energy <- new_energy[,(1+i)]-bio_seas_comp

ggplot(data=new_energy, aes(x=my_date, y=new_energy[,(1+i)])) + geom_line(color="blue")+
            ylab(paste0("Trillion_BTU_",colnames(new_energy)[(1+i)],sep="")) +  geom_line(aes(y=bio_des
```




```
#Renewable Energy Deseason Graph
i=2

renew_deseason_energy <- new_energy[,(1+i)]-renew_seas_comp

ggplot(data=new_energy, aes(x=my_date, y=new_energy[,(1+i)])) + geom_line(color="blue")+
            ylab(paste0("Trillion_BTU_",colnames(new_energy)[(1+i)],sep="")) +  geom_line(aes(y=renew_de
```




```
#Renewable Energy Deseason Graph
i=3

hydro_deseason_energy <- new_energy[,(1+i)]-hydro_seas_comp

ggplot(data=new_energy, aes(x=my_date, y=new_energy[,(1+i)])) + geom_line(color="blue")+
            ylab(paste0("Trillion_BTU_",colnames(new_energy)[(1+i)],sep="")) +  geom_line(aes(y=hydro_de
```



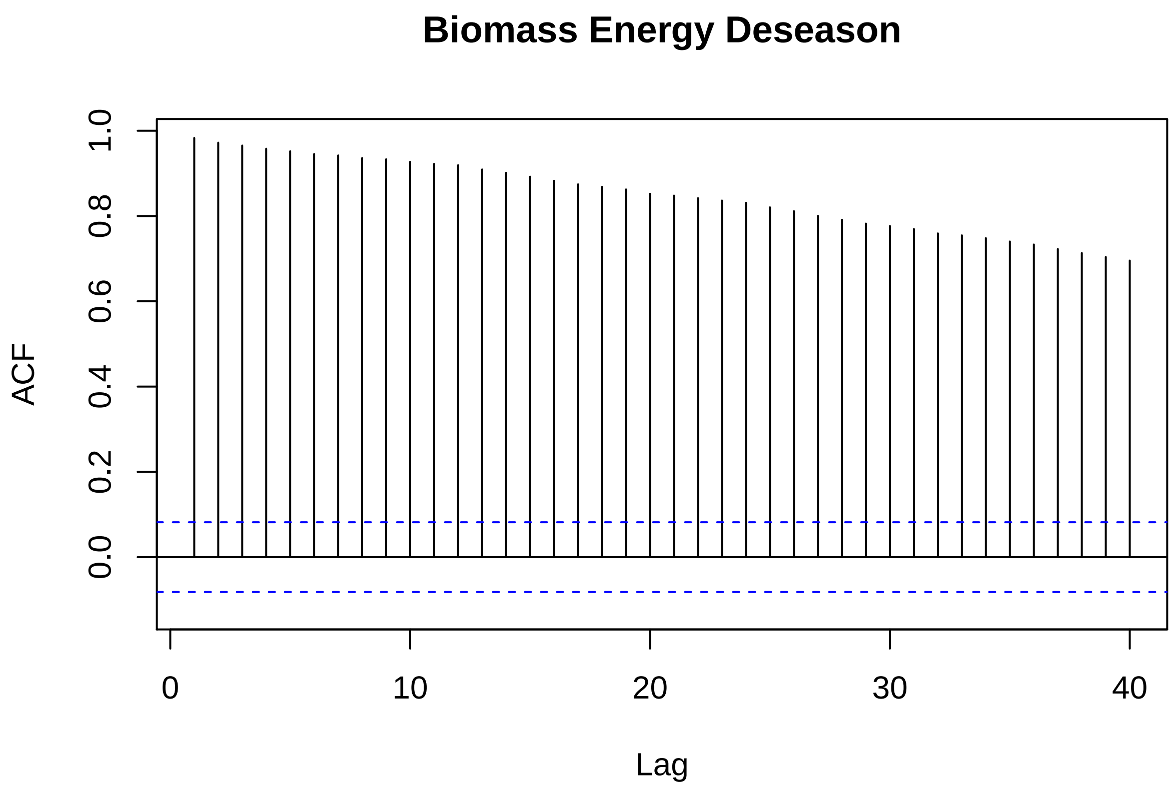
Q8

Plot ACF and PACF for the deseason series and compare with the plots from Q1. Did the plots change? How?

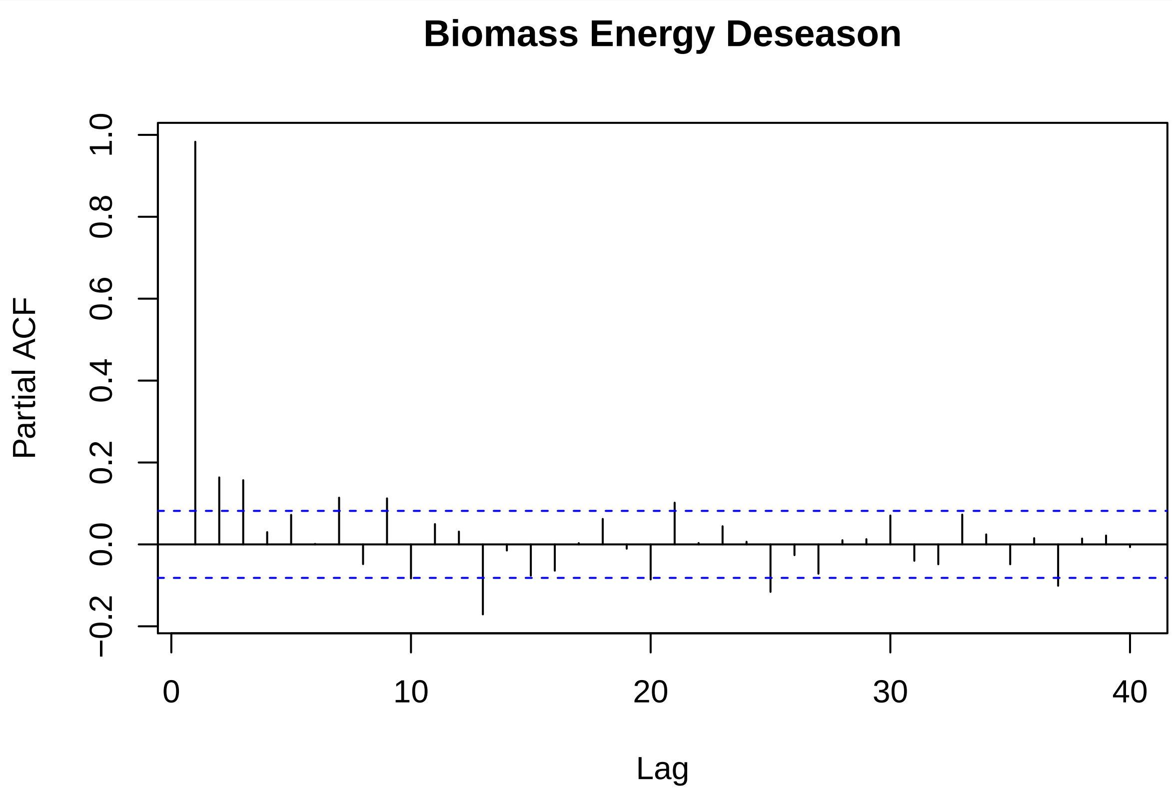
The ACF and PACF charts for hydro power production are significantly different. There is no longer a positive and negative correlation in the hydro power production ACF chart. There is also a weaker correlation coefficent between lags in the PACF chart.

For biomass and renewable energy there is less vairability in the partical ACF charts as there is no longer lags that have a high negative correlation. The ACF charts though look fairly similar to those produced in Q1.

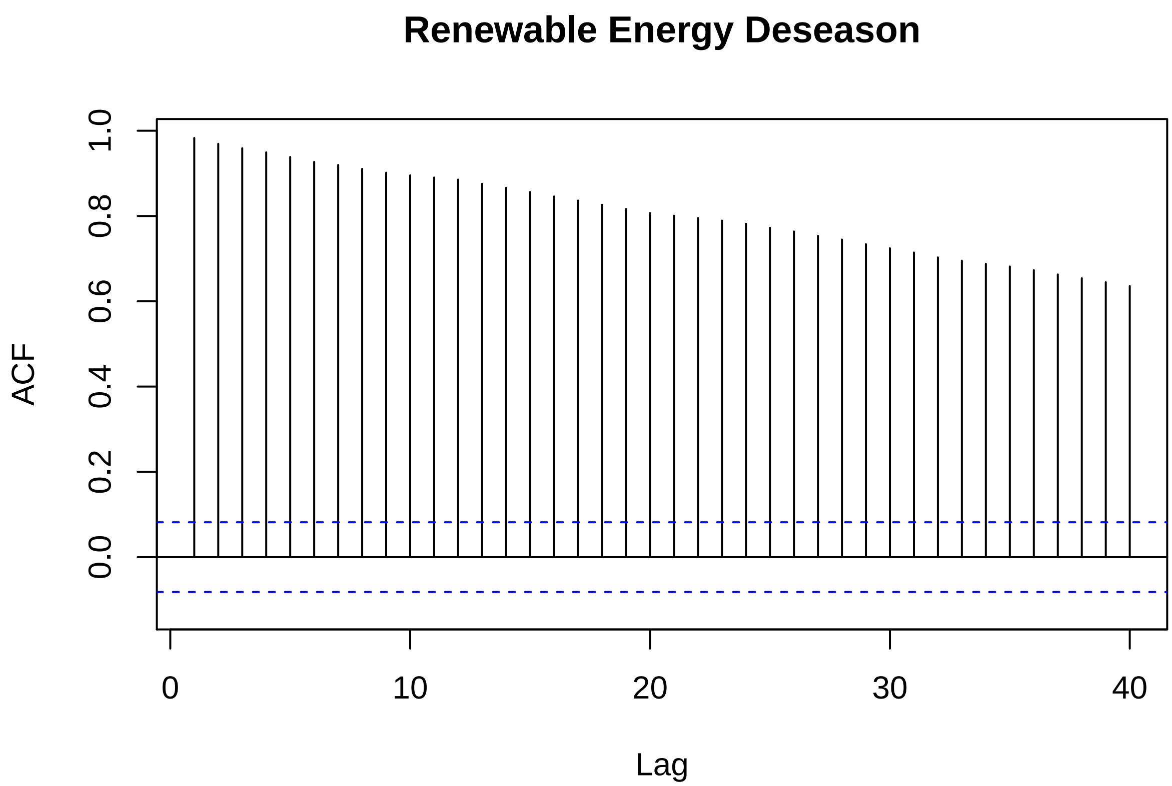
```
Acf(bio_deseason_energy,lag.max=40,main= "Biomass Energy Deseason")
```



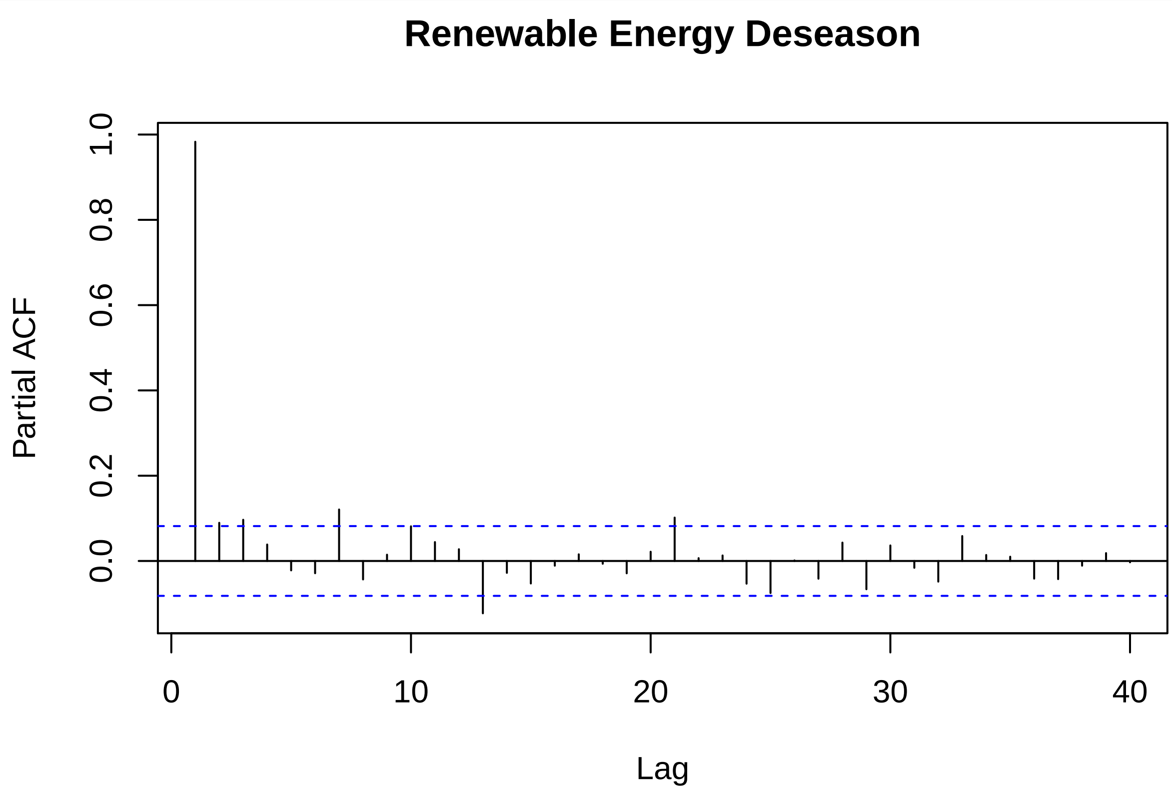

```
Pacf(bio_deseason_energy,lag.max=40,main= "Biomass Energy Deseason")
```



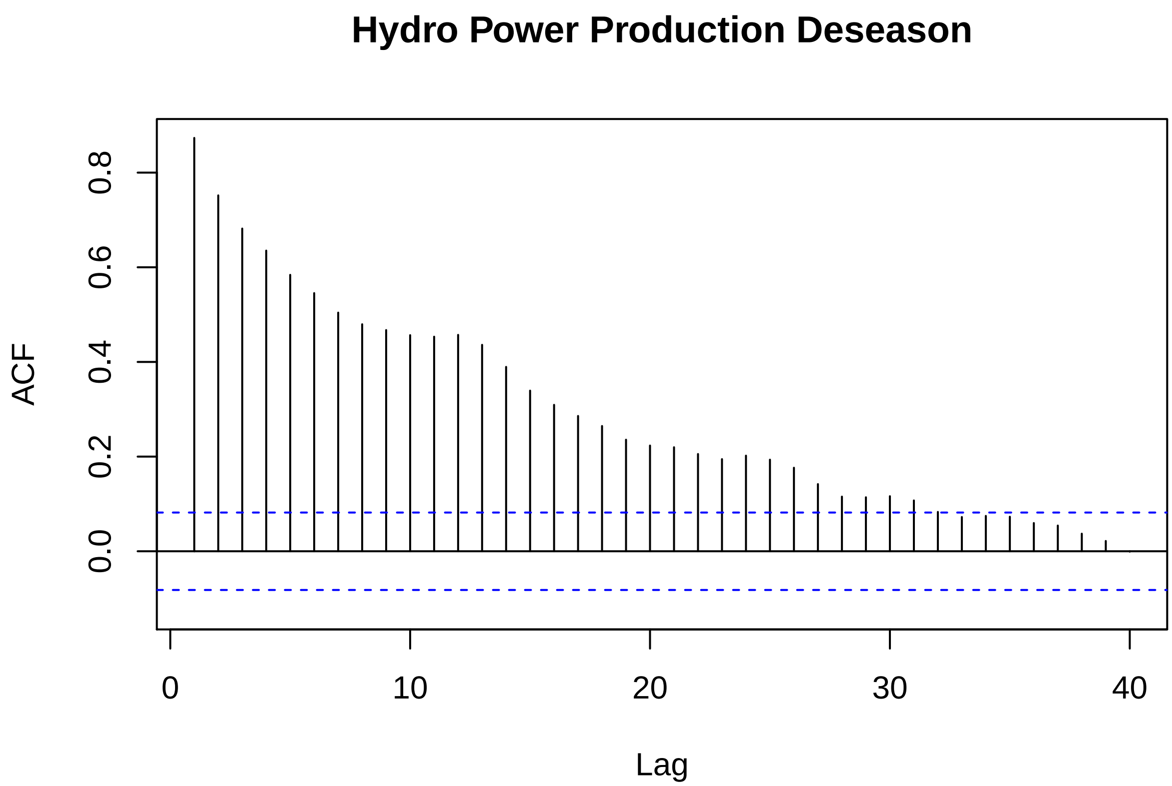

```
Acf(renew_deseason_energy,lag.max=40,main= "Renewable Energy Deseason")
```




```
Pacf(renew_deseason_energy,lag.max=40,main= "Renewable Energy Deseason")
```




```
Acf(hydro_deseason_energy,lag.max=40,main= "Hydro Power Production Deseason")
```




```
Pacf(hydro_deseason_energy,lag.max=40,main= "Hydro Power Production Deseason")
```