R Report

Author

2022-12-04

Introduction

This analysis is conducted on Sustainable Development Goals (SDGs) data obtained from World Bank database https://databank.worldbank.org/source/sustainable-development-goals-(sdgs). We selected the data for 10 countries (5 African and 5 South American Countries) with 3 **indicators**: `Access to clean fuels and technologies for cooking`, `Access to electricity` and `Access to electricity, rural` with a view to see how these indicators are connected and how the two continents perform against each other.

We will load the required packages `tidyvers`, `lubridate`, `readxl`, `moments`, `psych`, `ggplot2`, `ggpubr`, `Hmisc`, `tseries`, `forecast` after installing them using the following code chunk:

```
library(tidyverse)

## ── Attaching packages ─────────────────────────────────────── tidyverse
1.3.2 ──
## ✔ ggplot2 3.4.0      ✔ purrr   0.3.4
## ✔ tibble  3.1.7      ✔ dplyr   1.0.9
## ✔ tidyr   1.2.0      ✔ stringr 1.4.0
## ✔ readr   2.1.2      ✔ forcats 0.5.1
## ── Conflicts ──────────────────────────────────────────
tidyverse_conflicts() ──
## ✖ dplyr::filter() masks stats::filter()
## ✖ dplyr::lag()    masks stats::lag()

library(lubridate)

##
## Attaching package: 'lubridate'
##
## The following objects are masked from 'package:base':
##
##     date, intersect, setdiff, union

library(readxl)
library(moments)
library(psych)

##
## Attaching package: 'psych'
##
## The following objects are masked from 'package:ggplot2':
```

```
##
##     %+%, alpha
```

```
library(ggplot2)
library(ggpubr)
library(Hmisc)
```

```
## Loading required package: lattice
## Loading required package: survival
## Loading required package: Formula
##
## Attaching package: 'Hmisc'
##
## The following object is masked from 'package:psych':
##
##     describe
##
## The following objects are masked from 'package:dplyr':
##
##     src, summarize
##
## The following objects are masked from 'package:base':
##
##     format.pval, units
```

```
library(tseries)
```

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

```
library(forecast)
```

```
##
## Attaching package: 'forecast'
##
## The following object is masked from 'package:ggpubr':
##
##     gghistogram
```

About our Data

After scrapping the data from the data bank of the World Bank, I did some data cleaning in Excel before importing into R.

* I replaced the missing values with the average value for the rows involved.

* Since we're looking to see how the two continents perform against each other, I created a `Continent` column and filled the name of the continent each of the countries in our data set belonged to.

* I created 2 columns `Continent_Code` and `Series_Code` to replace the `Continent` and `Series Name` columns to make it easier to read the data.

* Under `Continent_Code` column, I represented **Africa** with **AFR** and **South America** with **SAR**

* Under the `Series_Code` column, I represented **Access to clean fuels and technologies for cooking (% of population)** with ***CF***, **Access to electricity (% of population)** with ***E*** and **Access to electricity, rural (% of rural population)** with ***ER***

* I deleted the `Country Name`, `Series Name`, `Series Code`, `Continent` columns.

* I renamed the year columns from `2011 [YR2011]` to `2011` for all ten years accordingly.

* I then imported both the **raw data** and the **cleaned data** using the code chunk below

```
SDG_Raw_Data <-
read_xlsx('P_Data_Extract_From_Sustainable_Development_Goals_(SDGs).xlsx')
SDG_Data <- read_xlsx('SDG.xlsx')
```

We take glimpse of our cleaned data using the code chunk below

```
str(SDG_Data)

## tibble [30 × 13] (S3: tbl_df/tbl/data.frame)
##  $ Country_Code  : chr [1:30] "BEN" "BEN" "BEN" "CIV" ...
##  $ Continent_Code: chr [1:30] "AFR" "AFR" "AFR" "AFR" ...
##  $ Series_Code   : chr [1:30] "CF" "E" "ER" "CF" ...
##  $ 2011          : num [1:30] 4.1 36.9 13.6 18.7 55.8 ...
##  $ 2012          : num [1:30] 4.3 38.4 14.6 19.4 55.8 ...
##  $ 2013          : num [1:30] 4.3 34.7 13.4 20.6 61.4 ...
##  $ 2014          : num [1:30] 4.3 34.1 16.2 21.7 61.9 ...
##  $ 2015          : num [1:30] 4.3 29.6 11.1 23.1 62.6 ...
##  $ 2016          : num [1:30] 4.3 37.1 15.3 24.9 64.3 ...
##  $ 2017          : num [1:30] 4.2 34.5 17.2 26.6 65.6 ...
##  $ 2018          : num [1:30] 4.2 39.2 16.7 28.4 67.1 ...
##  $ 2019          : num [1:30] 4 40.3 17.5 30.3 68.5 ...
##  $ 2020          : num [1:30] 4 41.4 18.2 31.8 69.7 ...

head(SDG_Data)

## # A tibble: 6 × 13
##   Country_Code Continent_Code Series_Code `2011` `2012` `2013` `2014`
`2015`
##   <chr>        <chr>          <chr>        <dbl>  <dbl>  <dbl>  <dbl>
<dbl>
## 1 BEN          AFR            CF             4.1    4.3    4.3    4.3
4.3
## 2 BEN          AFR            E             36.9   38.4   34.7   34.1
29.6
## 3 BEN          AFR            ER            13.6   14.6   13.4   16.2
11.1
```

```
## 4 CIV          AFR           CF             18.7   19.4   20.6   21.7
23.1
## 5 CIV          AFR           E              55.8   55.8   61.4   61.9
62.6
## 6 CIV          AFR           ER             26.3   25.8   34.3   34.3
39.2
## # … with 5 more variables: `2016` <dbl>, `2017` <dbl>, `2018` <dbl>,
## #   `2019` <dbl>, `2020` <dbl>
```

```
tail(SDG_Data)
```

```
## # A tibble: 6 × 13
##   Country_Code Continent_Code Series_Code `2011` `2012` `2013` `2014`
`2015`
##   <chr>        <chr>          <chr>        <dbl>  <dbl>  <dbl>  <dbl>
<dbl>
## 1 COL          SAR            CF            87     87.6   88.4   89.1
89.8
## 2 COL          SAR            E             96.7   97.0   97.8   97.8
98.2
## 3 COL          SAR            ER            85.5   87.1   90.1   89.9
91.8
## 4 BRA          SAR            CF            94.3   94.7   95     95.2
95.4
## 5 BRA          SAR            E             99.3   99.5   99.6   99.7
99.7
## 6 BRA          SAR            ER            96.2   97.2   97.5   97.9
98.2
## # … with 5 more variables: `2016` <dbl>, `2017` <dbl>, `2018` <dbl>,
## #   `2019` <dbl>, `2020` <dbl>
```

Since our analysis is focused on seeing how the SDGs of Africa does against South America, the code chunk below extracts the data for Africa and splits that further into the data for the three SDGs indicators so as to start our analysis.

```
SDG_Africa <-SDG_Data[SDG_Data$Continent_Code == 'AFR', ]
SDG_Africa_CF <-SDG_Africa[SDG_Africa$Series_Code == 'CF', ]
SDG_Africa_CF <- SDG_Africa_CF[,-1:-3]
SDG_Africa_E <-SDG_Africa[SDG_Africa$Series_Code == 'E', ]
SDG_Africa_E <- SDG_Africa_E[,-1:-3]
SDG_Africa_ER <-SDG_Africa[SDG_Africa$Series_Code == 'ER', ]
SDG_Africa_ER <- SDG_Africa_ER[,-1:-3]
```

The next code chunk does the same for the South America SDGs Data

```
SDG_SAmerica <-SDG_Data[SDG_Data$Continent_Code == 'SAR', ]
SDG_SAmerica_CF <-SDG_SAmerica[SDG_SAmerica$Series_Code == 'CF', ]
SDG_SAmerica_CF <- SDG_SAmerica_CF[,-1:-3]
SDG_SAmerica_E <-SDG_SAmerica[SDG_SAmerica$Series_Code == 'E', ]
SDG_SAmerica_E <- SDG_SAmerica_E[,-1:-3]
```

```
SDG_SAmerica_ER <-SDG_SAmerica[SDG_SAmerica$Series_Code == 'ER', ]
SDG_SAmerica_ER <- SDG_SAmerica_ER[,-1:-3]
```

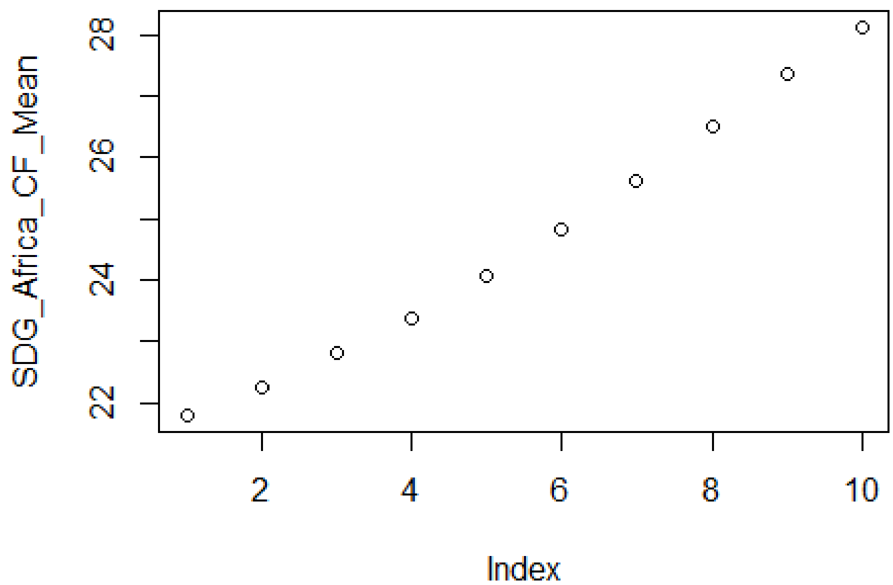
Descriptive Analysis

Mean

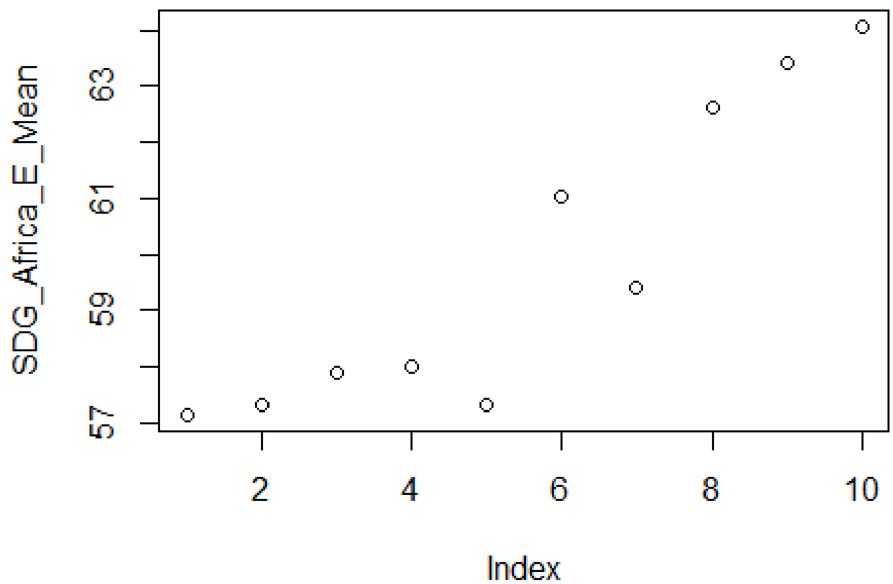
Africa

The code chunk below calculates the **Mean** for each of the **indicators**.

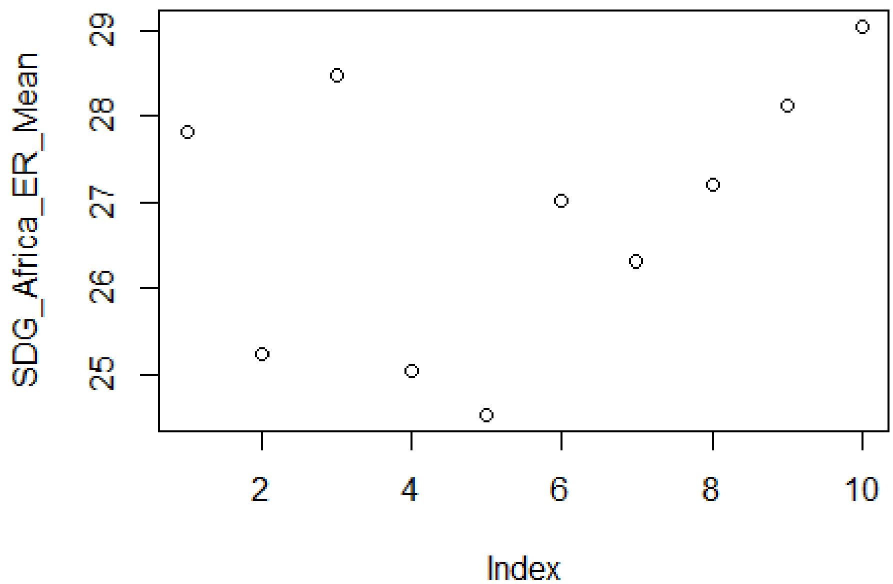
```
SDG_Africa_CF_Mean <-colMeans(SDG_Africa_CF)
SDG_Africa_CF_Mean

##  2011  2012  2013  2014  2015  2016  2017  2018  2019  2020 
## 21.80 22.27 22.82 23.39 24.08 24.83 25.63 26.50 27.36 28.12

plot(SDG_Africa_CF_Mean)
```




```
SDG_Africa_E_Mean <-colMeans(SDG_Africa_E)
SDG_Africa_E_Mean

##     2011     2012     2013     2014     2015     2016     2017     2018 
## 57.14133 57.31184 57.91076 57.98925 57.33177 61.04530 59.39400 62.61516 
##     2019     2020 
## 63.40291 64.06694

plot(SDG_Africa_E_Mean)
```



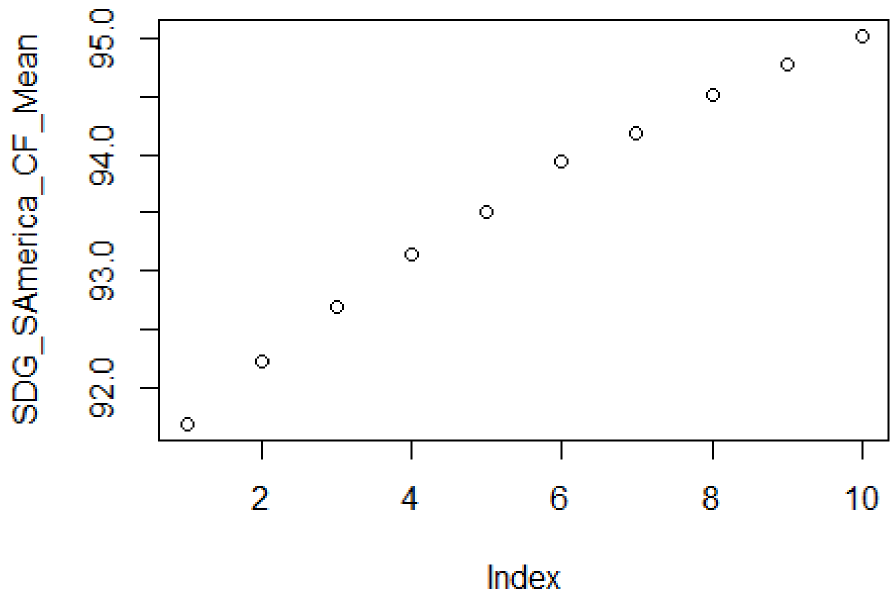

```
SDG_Africa_ER_Mean <-colMeans(SDG_Africa_ER)
SDG_Africa_ER_Mean

##     2011     2012     2013     2014     2015     2016     2017     2018
## 27.82028 25.23894 28.48678 25.05142 24.52781 27.01913 26.32509 27.21997
##     2019     2020
## 28.13831 29.04710

plot(SDG_Africa_ER_Mean)
```



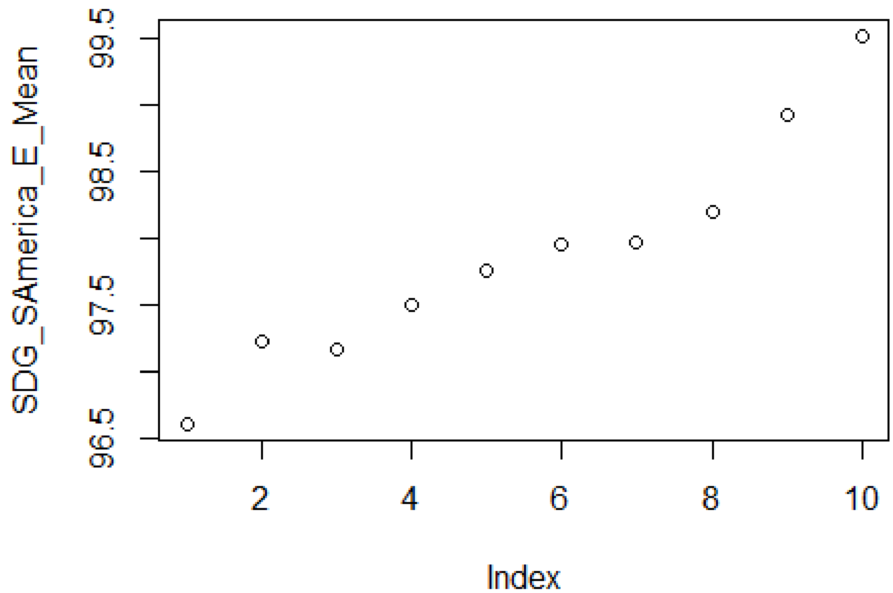
South America

The following code chunk calculates the **Mean** for each **indicators**.

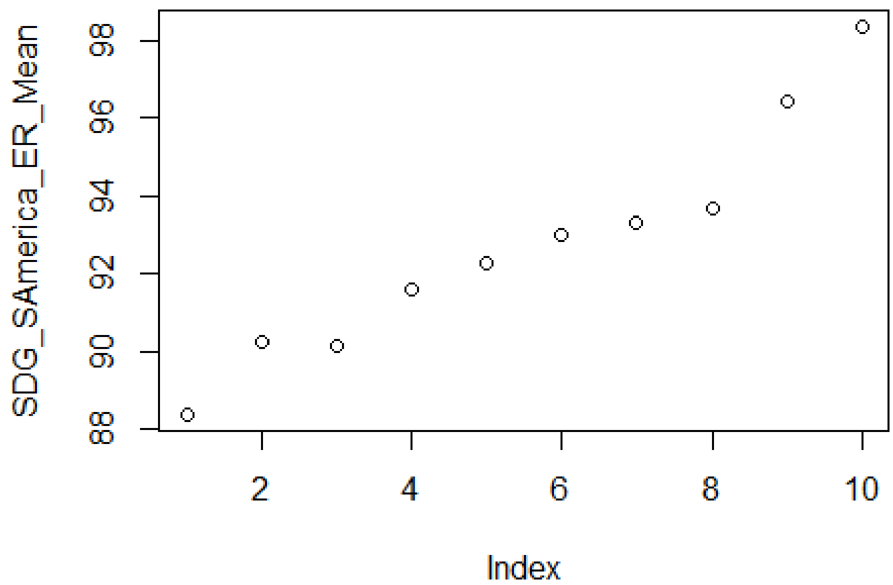
```
SDG_SAmerica_CF_Mean <-colMeans(SDG_SAmerica_CF)
SDG_SAmerica_CF_Mean

##  2011  2012  2013  2014  2015  2016  2017  2018  2019  2020
## 91.68 92.22 92.69 93.14 93.51 93.94 94.18 94.52 94.78 95.02

plot(SDG_SAmerica_CF_Mean)
```




```
SDG_SAmerica_E_Mean <-colMeans(SDG_SAmerica_E)
SDG_SAmerica_E_Mean

##     2011     2012     2013     2014     2015     2016     2017     2018
## 96.60836 97.23380 97.16059 97.49598 97.75217 97.94992 97.96000 98.19792
##     2019     2020
## 98.92719 99.51082

plot(SDG_SAmerica_E_Mean)
```




```
SDG_SAmerica_ER_Mean <-colMeans(SDG_SAmerica_ER)
SDG_SAmerica_ER_Mean

##     2011     2012     2013     2014     2015     2016     2017     2018
## 88.39217 90.23192 90.15595 91.61902 92.29053 92.98707 93.34608 93.69721
##     2019     2020
## 96.42277 98.36270

plot(SDG_SAmerica_ER_Mean)
```



Median

Africa

The code chunk below calculates the **Median** for each of the **indicators**.

```
SDG_Africa_CF_Median <-median(SDG_Africa_CF_Mean)
SDG_Africa_CF_Median

## [1] 24.455

SDG_Africa_E_Median <-median(SDG_Africa_E_Mean)
SDG_Africa_E_Median

## [1] 58.69163

SDG_Africa_ER_Median <-median(SDG_Africa_ER_Mean)
SDG_Africa_ER_Median

## [1] 27.11955
```

South America

The following code chunk calculates the **Median** for each **indicators**.

```
SDG_SAmerica_CF_Median <-median(SDG_SAmerica_CF_Mean)
SDG_SAmerica_CF_Median

## [1] 93.725
```

```
SDG_SAmerica_E_Median <-median(SDG_SAmerica_E_Mean)
SDG_SAmerica_E_Median

## [1] 97.85104

SDG_SAmerica_ER_Median <-median(SDG_SAmerica_ER_Mean)
SDG_SAmerica_ER_Median

## [1] 92.6388
```

Standard Deviation

Africa

The code chunk below calculates the **Standard Deviation** for each of the **indicators**.

```
SDG_Africa_CF_SD <- sd(SDG_Africa_CF_Mean)
SDG_Africa_CF_SD

## [1] 2.181732

SDG_Africa_E_SD <- sd(SDG_Africa_E_Mean)
SDG_Africa_E_SD

## [1] 2.728935

SDG_Africa_ER_SD <- sd(SDG_Africa_ER_Mean)
SDG_Africa_ER_SD

## [1] 1.554829
```

South America

The following code chunk calculates the **Standard Deviation** for each **indicators**.

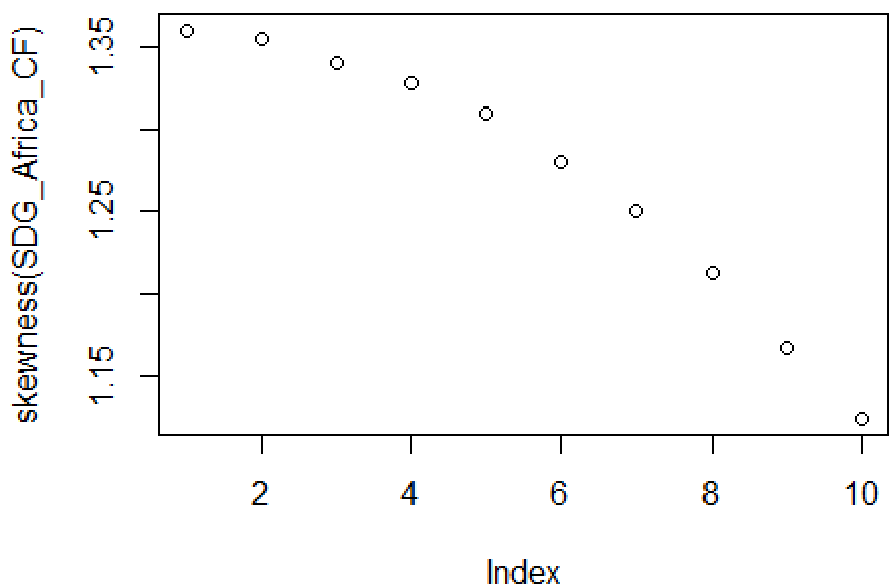
```
SDG_SAmerica_CF_SD <- sd(SDG_SAmerica_CF_Mean)
SDG_SAmerica_CF_SD

## [1] 1.122277

SDG_SAmerica_E_SD <- sd(SDG_SAmerica_E_Mean)
SDG_SAmerica_E_SD

## [1] 0.8553463

SDG_SAmerica_ER_SD <- sd(SDG_SAmerica_ER_Mean)
SDG_SAmerica_ER_SD

## [1] 2.978649
```

Skewness

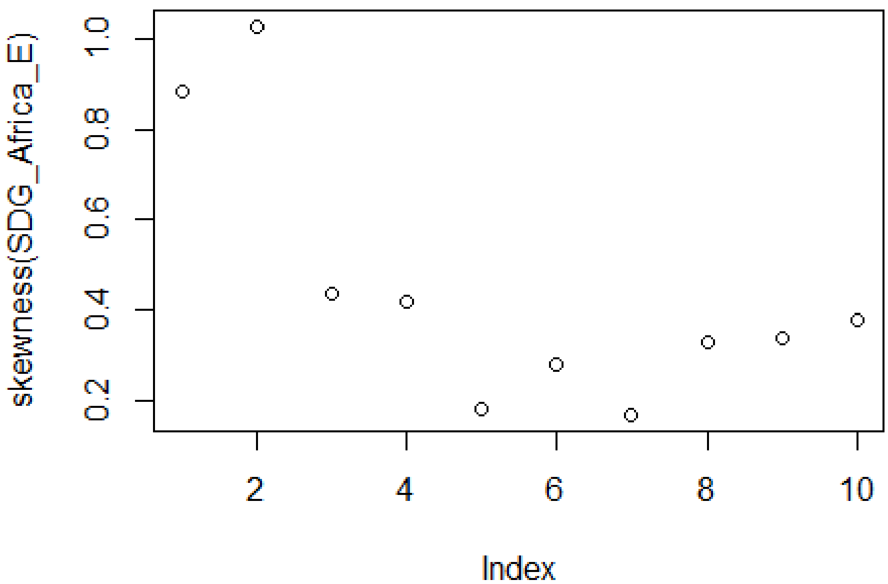
Africa

The code chunk below calculates the **Skewness** for each of the **indicators**.

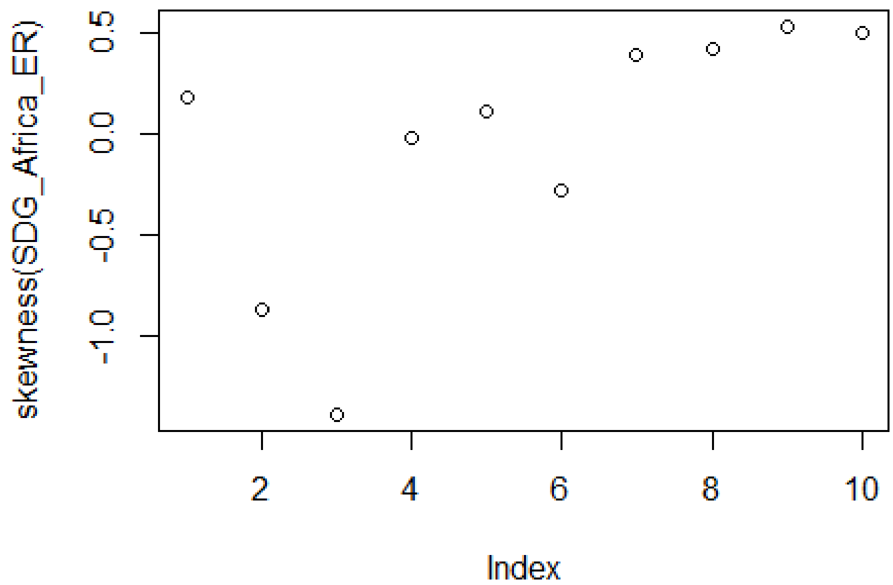
```
skewness(SDG_Africa_CF_Mean)

## [1] 0.2273959

plot(skewness(SDG_Africa_CF))
```




```
skewness(SDG_Africa_E_Mean)

## [1] 0.4755113

plot(skewness(SDG_Africa_E))
```



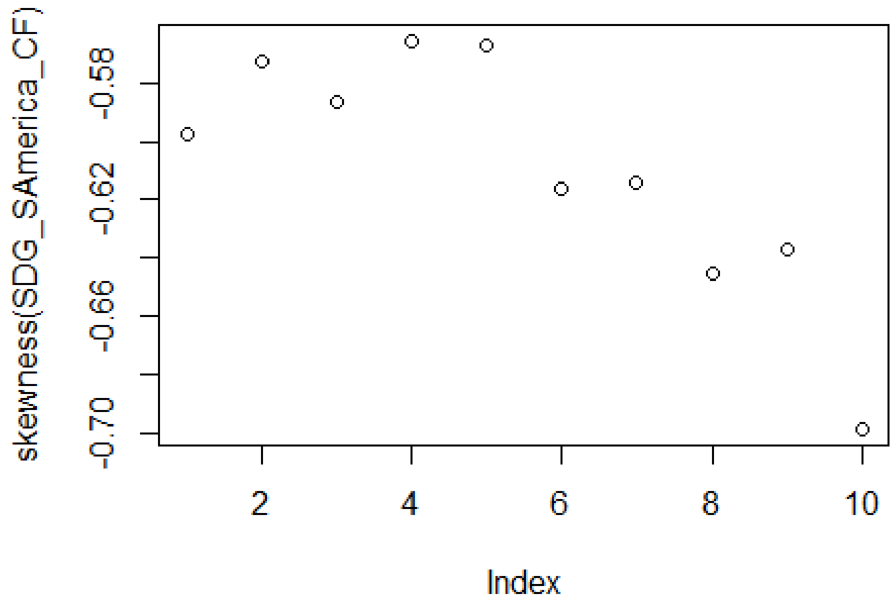

```
skewness(SDG_Africa_ER_Mean)

## [1] -0.2186113

plot(skewness(SDG_Africa_ER))
```



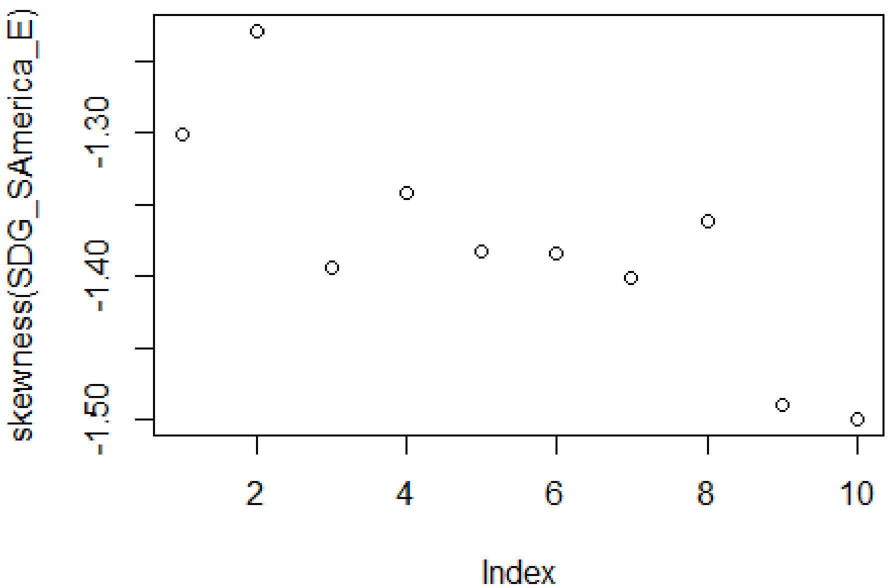
South America

The following code chunk calculates the **Skewness** for each **indicators**.

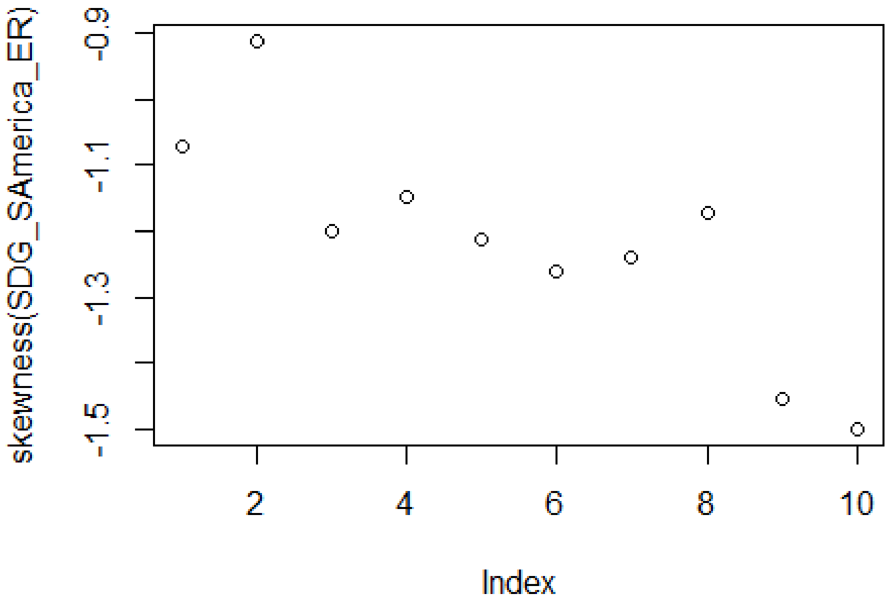
```
skewness(SDG_SAmerica_CF_Mean)

## [1] -0.3272593

plot(skewness(SDG_SAmerica_CF))
```




```
skewness(SDG_SAmerica_E_Mean)

## [1] 0.5180284

plot(skewness(SDG_SAmerica_E))
```




```
skewness(SDG_SAmerica_ER_Mean)

## [1] 0.4856253

plot(skewness(SDG_SAmerica_ER))
```

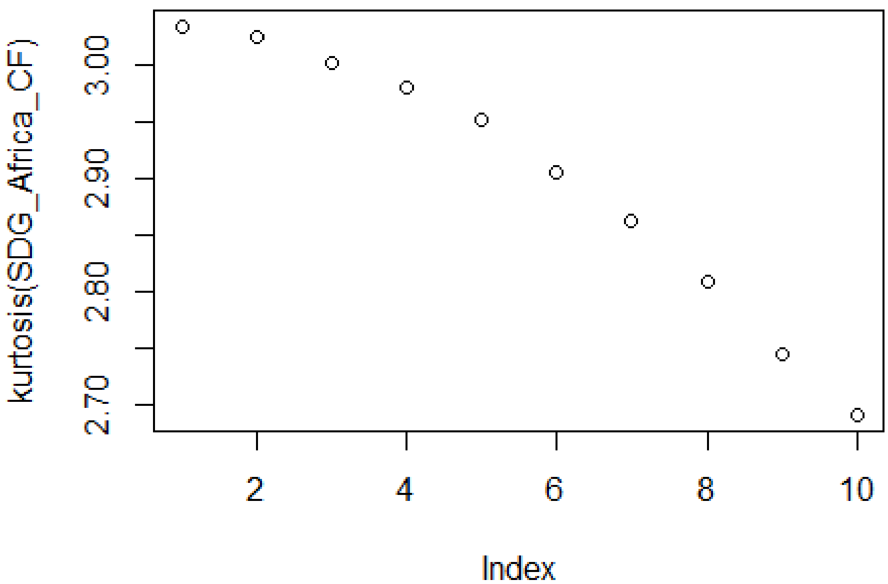


Kurtosis

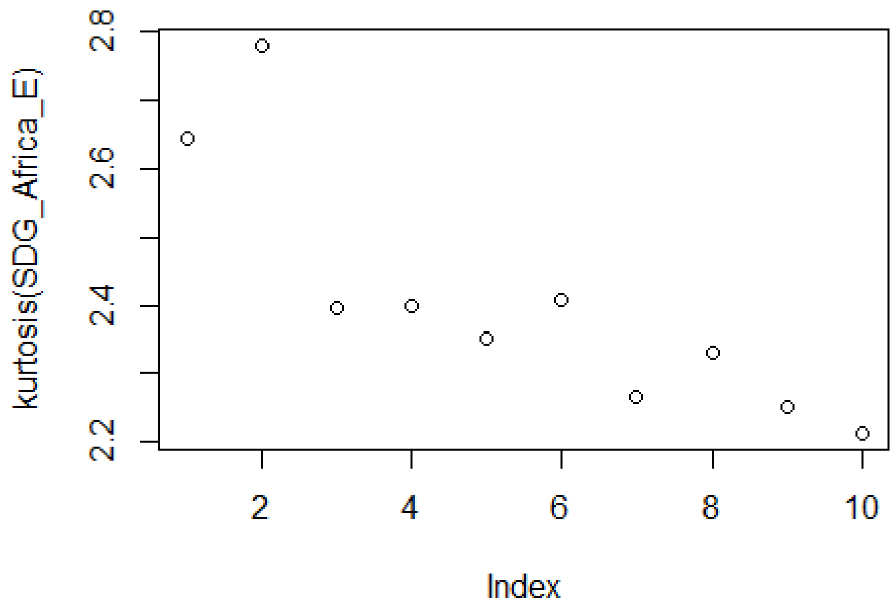
Africa

The code chunk below calculates the **Kurtosis** for each of the **indicators**.

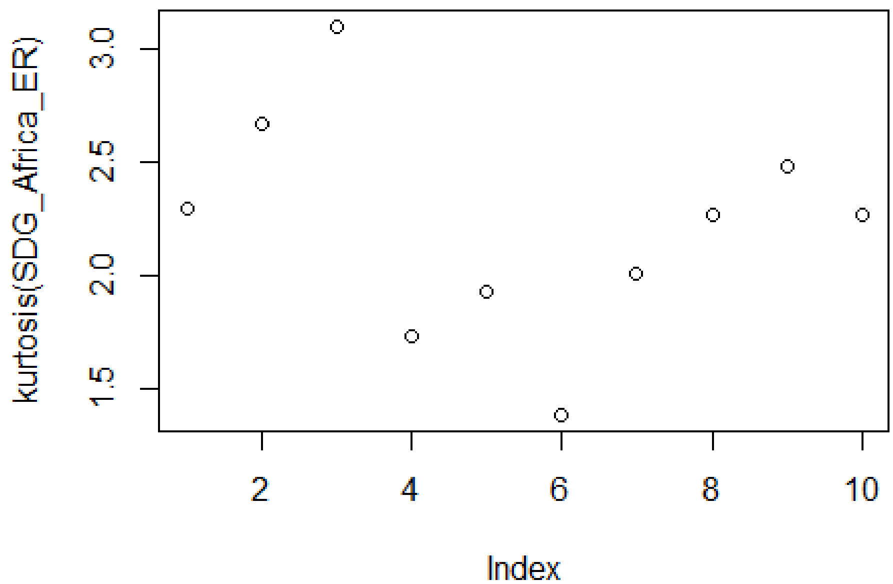
```
kurtosis(SDG_Africa_CF_Mean)

## [1] 1.748045

plot(kurtosis(SDG_Africa_CF))
```




```
kurtosis(SDG_Africa_E_Mean)

## [1] 1.573946

plot(kurtosis(SDG_Africa_E))
```



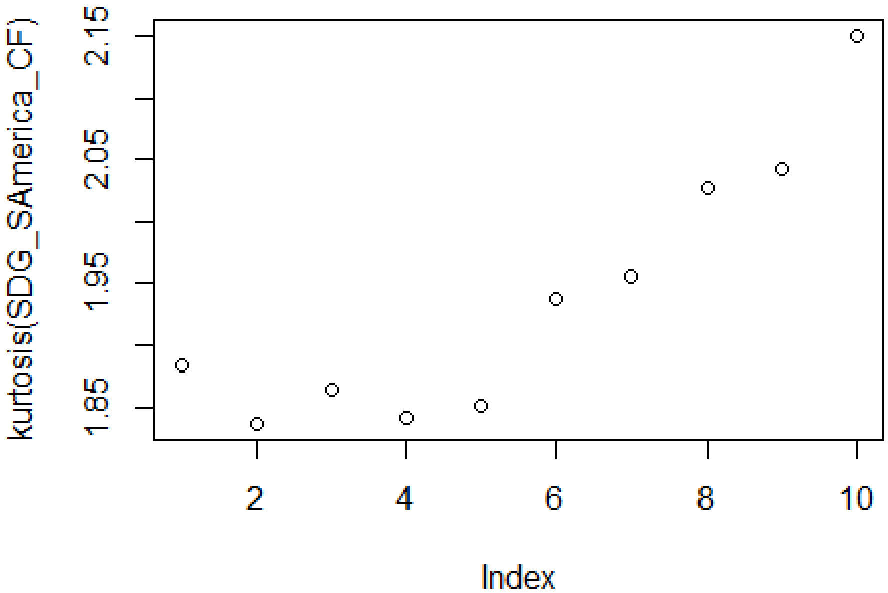

```
kurtosis(SDG_Africa_ER_Mean)

## [1] 1.718805

plot(kurtosis(SDG_Africa_ER))
```



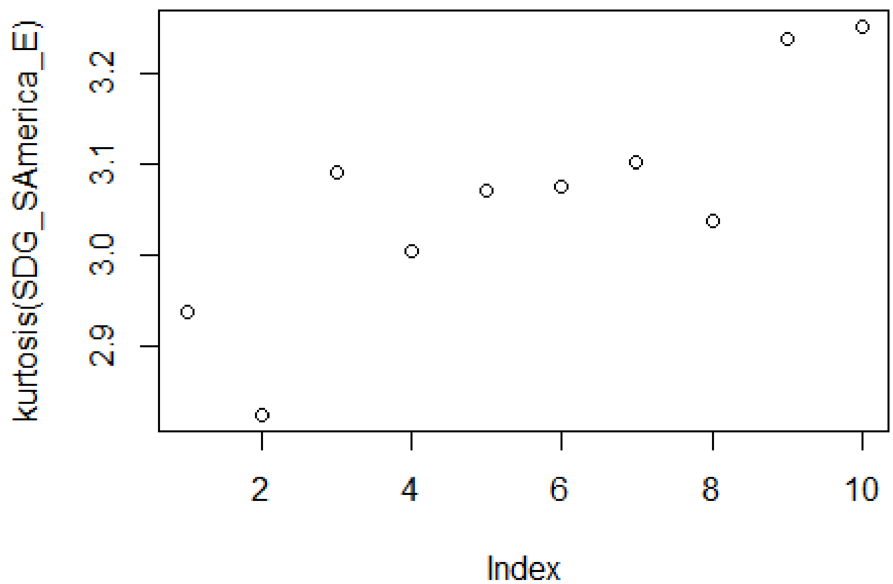
South America

The following code chunk calculates the **Kurtosis** for each **indicators**.

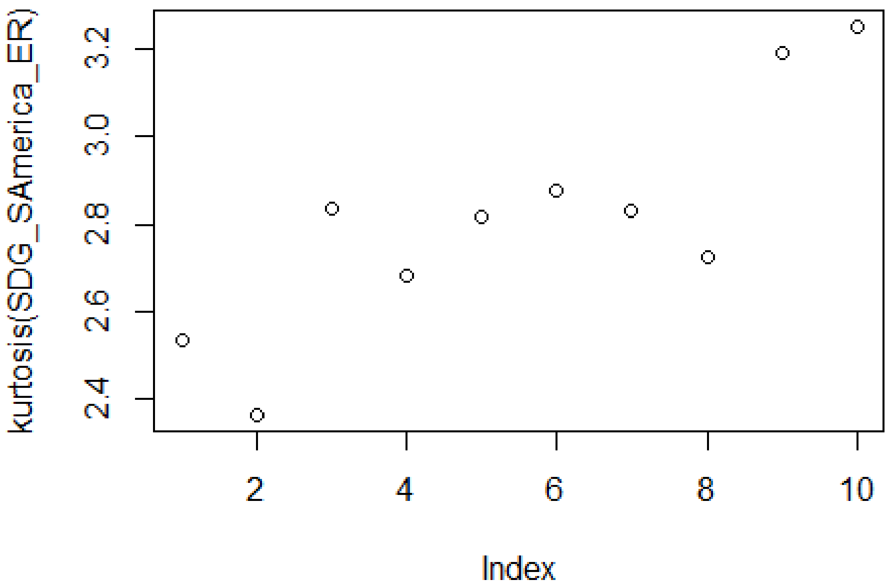
```
kurtosis(SDG_SAmerica_CF_Mean)

## [1] 1.88484

plot(kurtosis(SDG_SAmerica_CF))
```




```
kurtosis(SDG_SAmerica_E_Mean)

## [1] 2.622221

plot(kurtosis(SDG_SAmerica_E))
```




```
kurtosis(SDG_SAmerica_ER_Mean)

## [1] 2.5452

plot(kurtosis(SDG_SAmerica_ER))
```

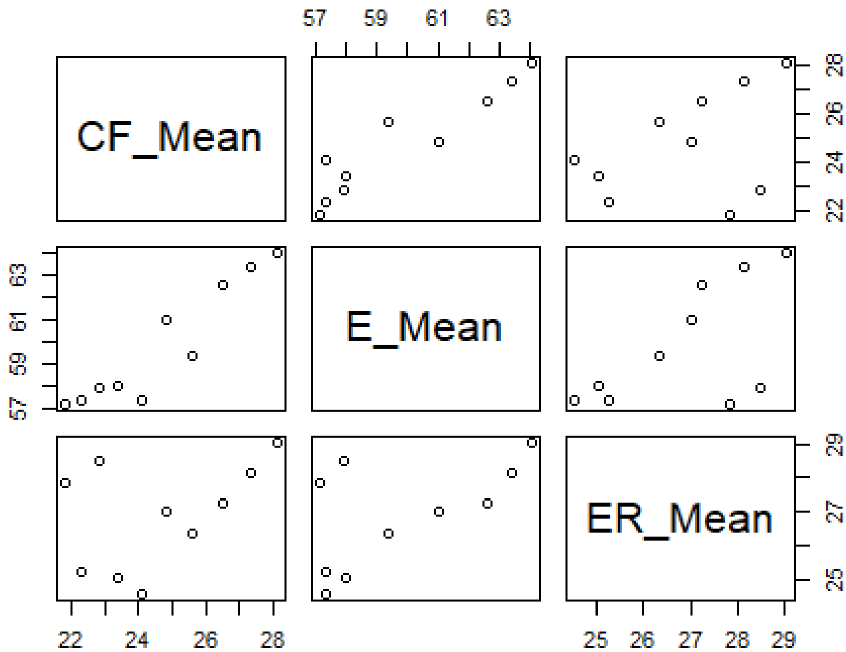


Correlation Analysis

Africa

We prepare our data for the analysis and run the `Shapiro Wilk` test for normality

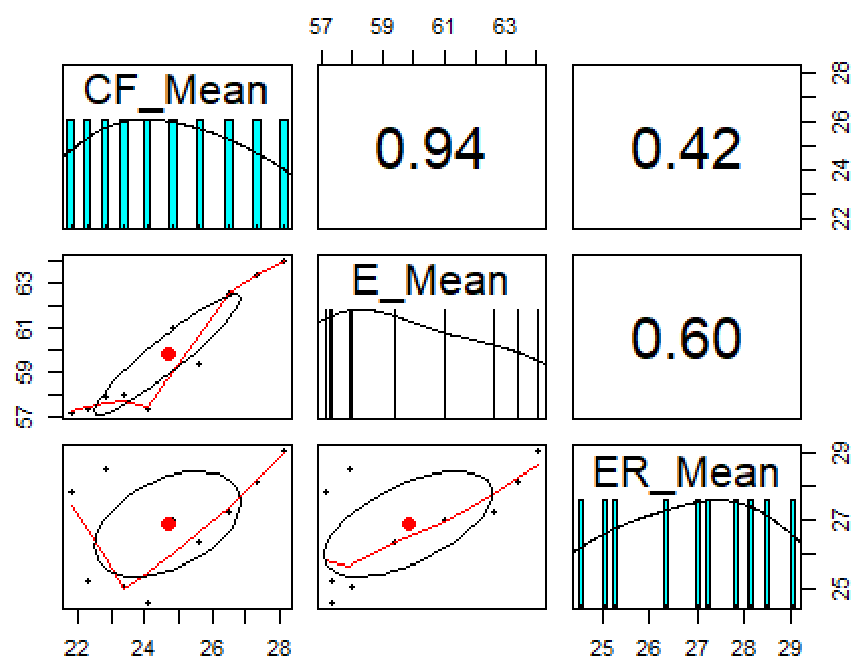
```
SDG_Africa_Mean<- t(rbind(SDG_Africa_CF_Mean, SDG_Africa_E_Mean,
SDG_Africa_ER_Mean))
SDG_Africa_Mean<- data.frame(SDG_Africa_Mean)
SDG_Africa_Mean<- data.frame(SDG_Africa_Mean$SDG_Africa_CF_Mean,
SDG_Africa_Mean$SDG_Africa_E_Mean, SDG_Africa_Mean$SDG_Africa_ER_Mean)

colnames(SDG_Africa_Mean)[1] = "CF_Mean"
colnames(SDG_Africa_Mean)[2] = "E_Mean"
colnames(SDG_Africa_Mean)[3] = "ER_Mean"

plot(SDG_Africa_Mean)
```




```
shapiro.test(SDG_Africa_Mean$CF_Mean)

##
##  Shapiro-Wilk normality test
##
## data:  SDG_Africa_Mean$CF_Mean
## W = 0.95441, p-value = 0.7207

shapiro.test(SDG_Africa_Mean$E_Mean)

##
##  Shapiro-Wilk normality test
##
## data:  SDG_Africa_Mean$E_Mean
## W = 0.85066, p-value = 0.05915

shapiro.test(SDG_Africa_Mean$ER_Mean)

##
##  Shapiro-Wilk normality test
##
## data:  SDG_Africa_Mean$ER_Mean
## W = 0.94568, p-value = 0.6178
```

After confirming that the `p-value` of the data is `>0.05` which means it is not different from a normal distribution, we run our correlation test for the 3 `indicators`

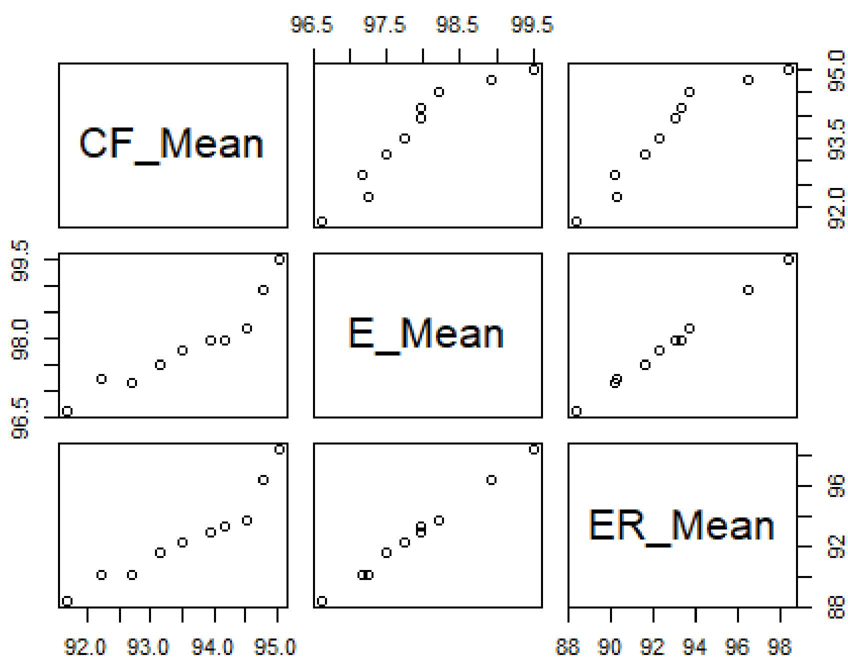
```
corAFR <- rcorr(as.matrix(SDG_Africa_Mean))
corAFR

##          CF_Mean E_Mean ER_Mean
## CF_Mean     1.00   0.94    0.42
## E_Mean      0.94   1.00    0.60
## ER_Mean     0.42   0.60    1.00
##
## n= 10
##
##
## P
##          CF_Mean E_Mean ER_Mean
## CF_Mean          0.0000 0.2211
## E_Mean   0.0000         0.0684
## ER_Mean  0.2211  0.0684

cor(SDG_Africa_Mean)

##            CF_Mean    E_Mean   ER_Mean
## CF_Mean 1.0000000 0.9376448 0.4247646
## E_Mean  0.9376448 1.0000000 0.5970348
## ER_Mean 0.4247646 0.5970348 1.0000000

pairs.panels(SDG_Africa_Mean)
```



South America

We prepare our data for the analysis and run the `Shapiro Wilk` test for normality

```
SDG_SAmerica_Mean<- t(rbind(SDG_SAmerica_CF_Mean, SDG_SAmerica_E_Mean,
SDG_SAmerica_ER_Mean))
SDG_SAmerica_Mean <- data.frame(SDG_SAmerica_Mean)
SDG_SAmerica_Mean<- data.frame(SDG_SAmerica_Mean$SDG_SAmerica_CF_Mean,
SDG_SAmerica_Mean$SDG_SAmerica_E_Mean,
SDG_SAmerica_Mean$SDG_SAmerica_ER_Mean)

colnames(SDG_SAmerica_Mean)[1] = "CF_Mean"
colnames(SDG_SAmerica_Mean)[2] = "E_Mean"
colnames(SDG_SAmerica_Mean)[3] = "ER_Mean"

plot(SDG_SAmerica_Mean)
```




```
shapiro.test(SDG_SAmerica_Mean$CF_Mean)

##
##  Shapiro-Wilk normality test
##
## data:  SDG_SAmerica_Mean$CF_Mean
## W = 0.95812, p-value = 0.7642

shapiro.test(SDG_SAmerica_Mean$E_Mean)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  SDG_SAmerica_Mean$E_Mean
## W = 0.96284, p-value = 0.8177
```

```
shapiro.test(SDG_SAmerica_Mean$ER_Mean)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  SDG_SAmerica_Mean$ER_Mean
## W = 0.96202, p-value = 0.8086
```

After confirming that the `p-value` of the data is `>0.05` which means it is not different from a normal distribution, we run our correlation test for the 3 `indicators`

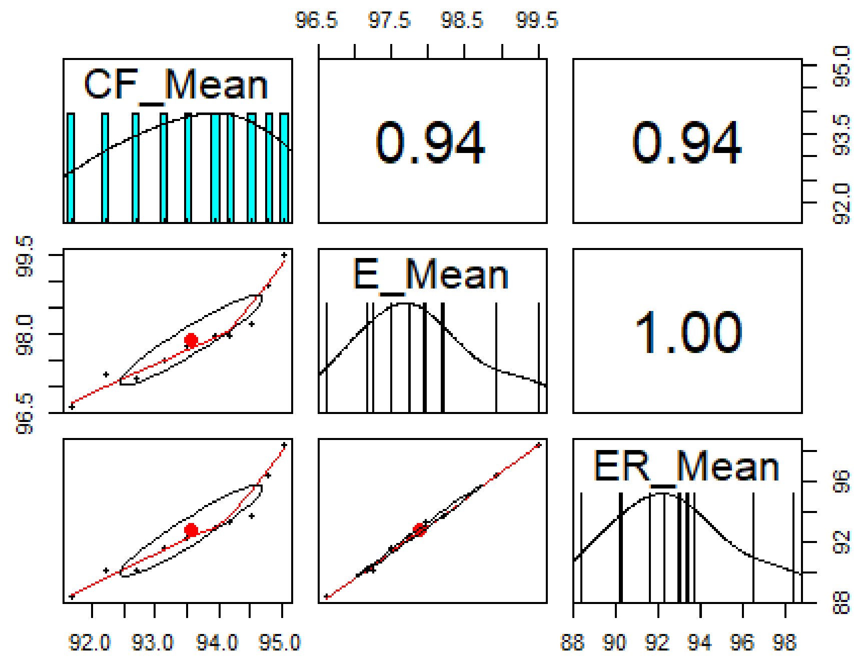
```
corS_A <- rcorr(as.matrix(SDG_SAmerica_Mean))
corS_A
```

```
##         CF_Mean E_Mean ER_Mean
## CF_Mean    1.00   0.94    0.94
## E_Mean     0.94   1.00    1.00
## ER_Mean    0.94   1.00    1.00
##
## n= 10
##
##
## P
##         CF_Mean E_Mean ER_Mean
## CF_Mean          0      0
## E_Mean   0              0
## ER_Mean  0       0
```

```
cor(SDG_SAmerica_Mean)
```

```
##           CF_Mean    E_Mean   ER_Mean
## CF_Mean 1.0000000 0.9364195 0.9428364
## E_Mean  0.9364195 1.0000000 0.9983795
## ER_Mean 0.9428364 0.9983795 1.0000000
```

```
pairs.panels(SDG_SAmerica_Mean)
```



Regression Analysis

Africa

```
multiple_regression_AFR <- lm(CF_Mean ~ E_Mean + ER_Mean, data =
SDG_Africa_Mean)
summary(multiple_regression_AFR)

##
## Call:
## lm(formula = CF_Mean ~ E_Mean + ER_Mean, data = SDG_Africa_Mean)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -0.8517 -0.4245 -0.1349  0.4093  1.1472
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -18.2383     5.6573  -3.224 0.014575 *
## E_Mean        0.8498     0.1146   7.418 0.000147 ***
## ER_Mean      -0.2944     0.2011  -1.465 0.186471
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.7523 on 7 degrees of freedom
## Multiple R-squared:  0.9075, Adjusted R-squared:  0.8811
## F-statistic: 34.34 on 2 and 7 DF,  p-value: 0.0002406
```

South America

```
multiple_regression_SAR <- lm(CF_Mean ~ E_Mean + ER_Mean, data =
SDG_SAmerica_Mean)
summary(multiple_regression_SAR)
```

```
##
## Call:
## lm(formula = CF_Mean ~ E_Mean + ER_Mean, data = SDG_SAmerica_Mean)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.49771 -0.26263 -0.00512  0.19454  0.70848
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 201.8311   200.0821   1.009    0.347
## E_Mean       -1.9808     2.8060  -0.706    0.503
## ER_Mean       0.9231     0.8058   1.146    0.290
##
## Residual standard error: 0.4097 on 7 degrees of freedom
## Multiple R-squared:  0.8963, Adjusted R-squared:  0.8667
## F-statistic: 30.26 on 2 and 7 DF,  p-value: 0.0003588
```

Time Series

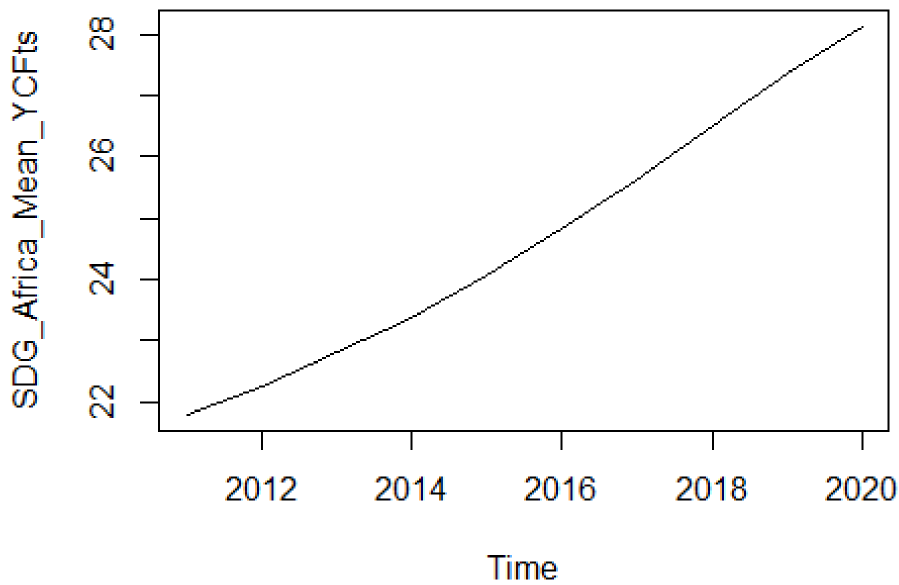
For the time series analysis, we will be doing `Time Series` and `forecast` for each `indicator` for each `Continent`.

Africa

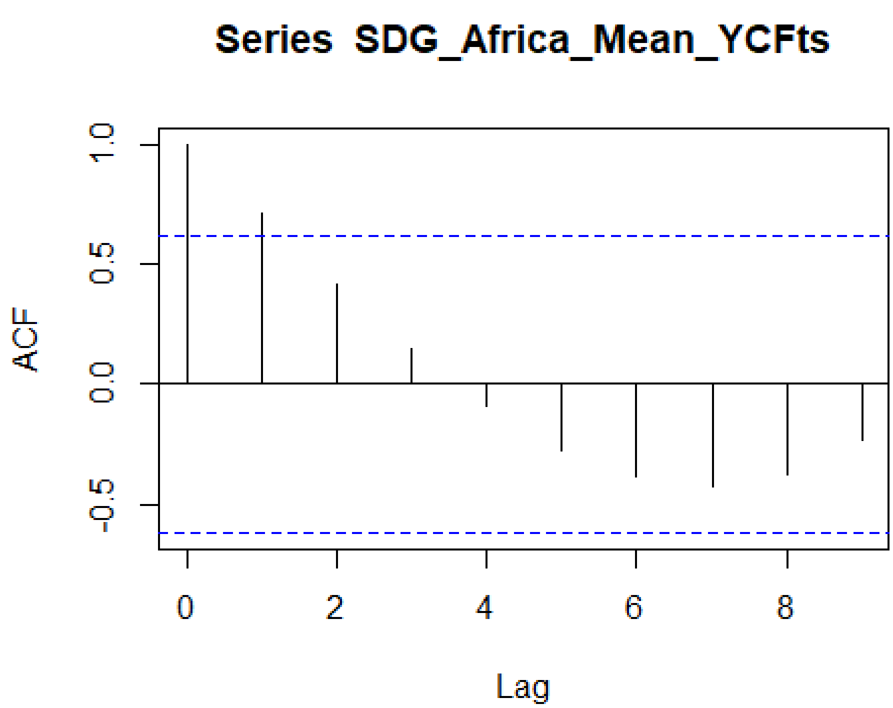
Access to clean fuels and technologies for cooking (CF)

We get the data ready by converting it to a `time Series (ts)` data and plot our data before analysis.

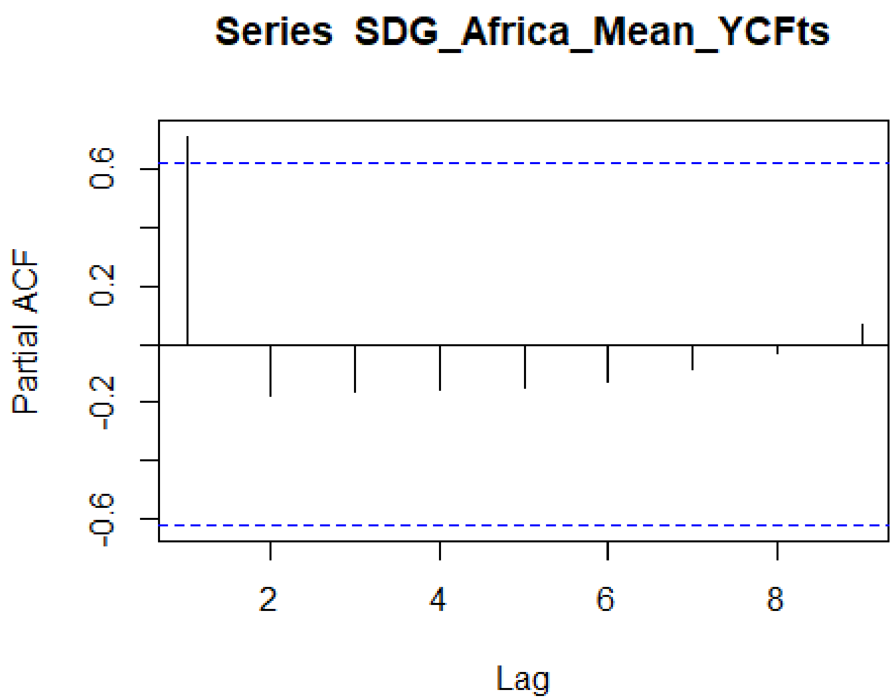
```
SDG_Africa_Mean_YCF <- SDG_Africa_Mean[,-2:-3]
SDG_Africa_Mean_YCFts <- ts(SDG_Africa_Mean_YCF, start = 2011, end = 2020,
frequency = 1)
plot(SDG_Africa_Mean_YCFts)
```




```
acf(SDG_Africa_Mean_YCFts)
```



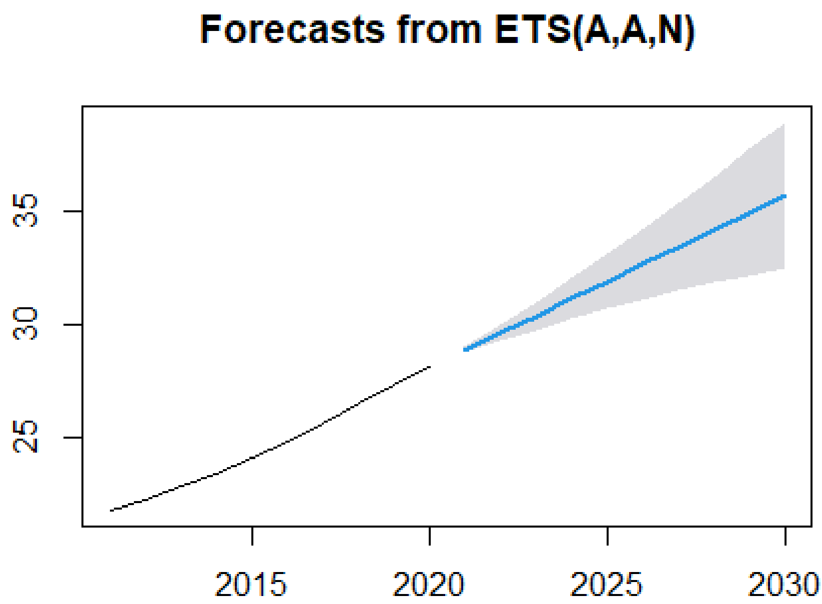

```
pacf(SDG_Africa_Mean_YCFts)
```




```
adf.test(SDG_Africa_Mean_YCFts)

##
##  Augmented Dickey-Fuller Test
##
## data:  SDG_Africa_Mean_YCFts
## Dickey-Fuller = -2.4458, Lag order = 2, p-value = 0.4026
## alternative hypothesis: stationary
```

For the forecast

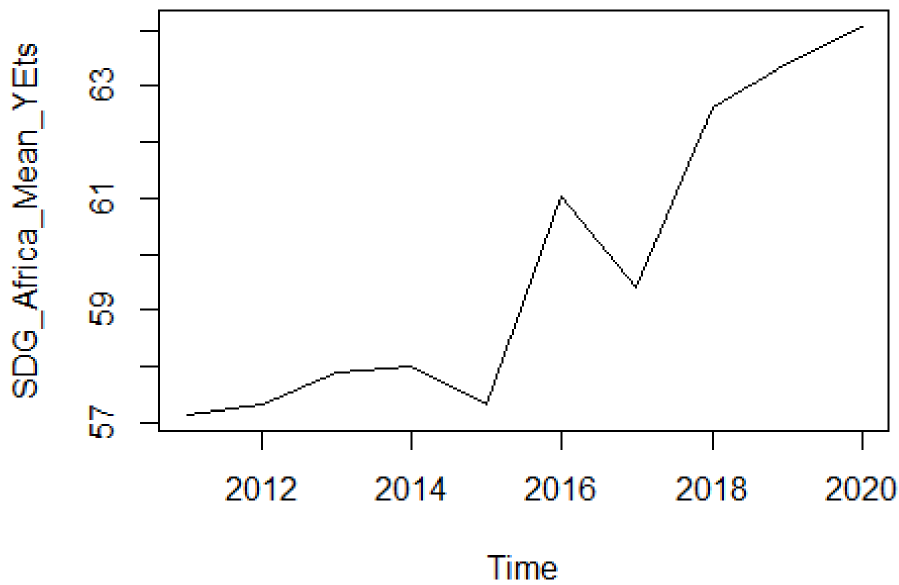
```
myforecast_AYCFts <- forecast(SDG_Africa_Mean_YCFts, level=c(95), h=10*1)
plot(myforecast_AYCFts)
```



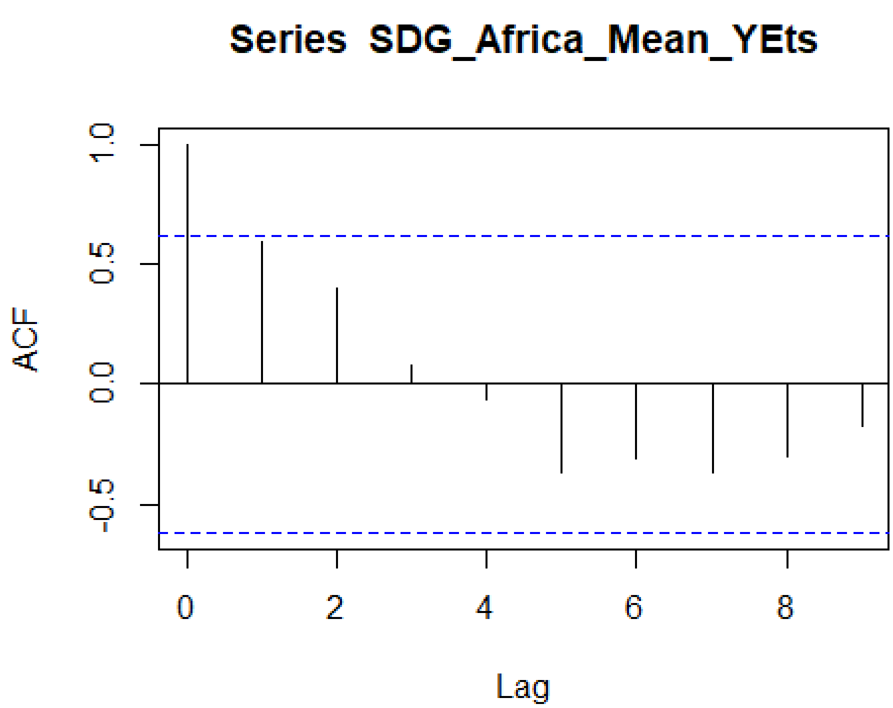
Access to electricity (E)

We get the data ready by converting it to a `time Series (ts)` data and plot our data before analysis.

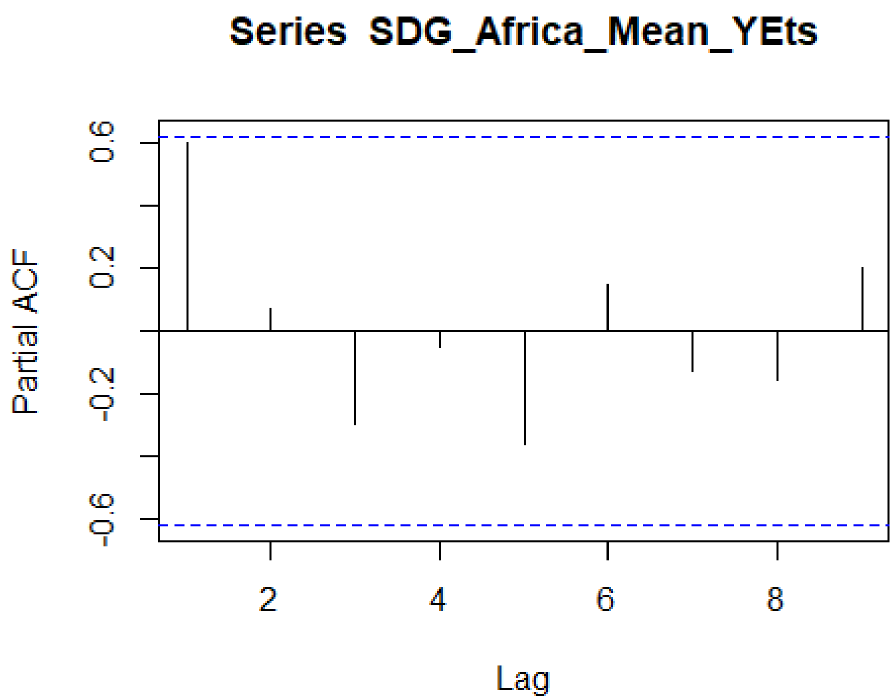
```
SDG_Africa_Mean_YE <- SDG_Africa_Mean[, -c(1,3)]
SDG_Africa_Mean_YEts <- ts(SDG_Africa_Mean_YE, start = 2011, end = 2020,
frequency = 1)
plot(SDG_Africa_Mean_YEts)
```




```
acf(SDG_Africa_Mean_YEts)
```



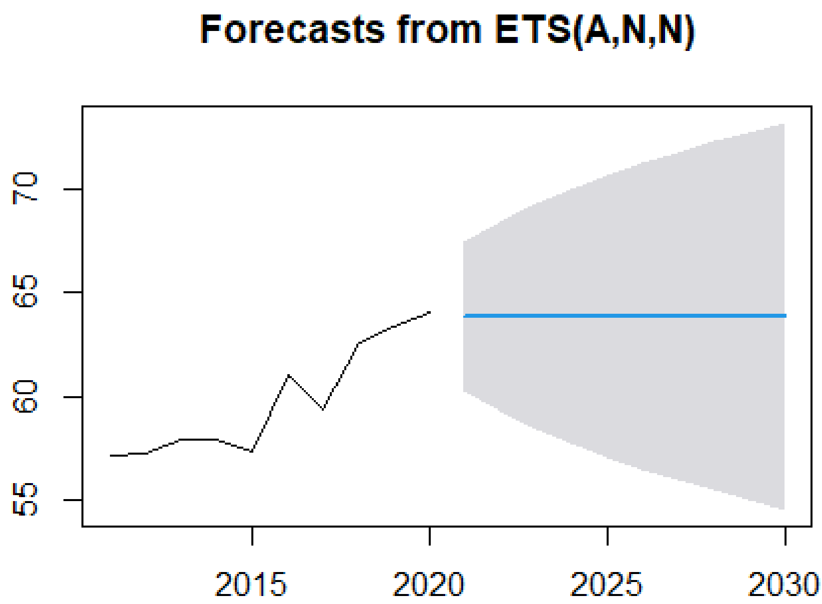

```
pacf(SDG_Africa_Mean_YEts)
```




```
adf.test(SDG_Africa_Mean_YEts)

##
##  Augmented Dickey-Fuller Test
##
## data:  SDG_Africa_Mean_YEts
## Dickey-Fuller = -0.70924, Lag order = 2, p-value = 0.9576
## alternative hypothesis: stationary
```

For the forecast

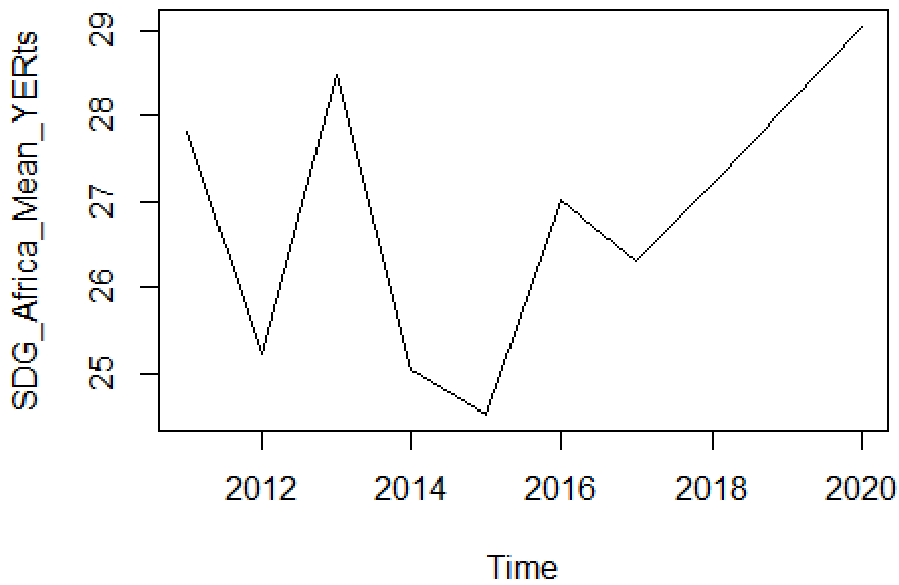
```
myforecast_AYEts <- forecast(SDG_Africa_Mean_YEts, level=c(95), h=10*1)
plot(myforecast_AYEts)
```



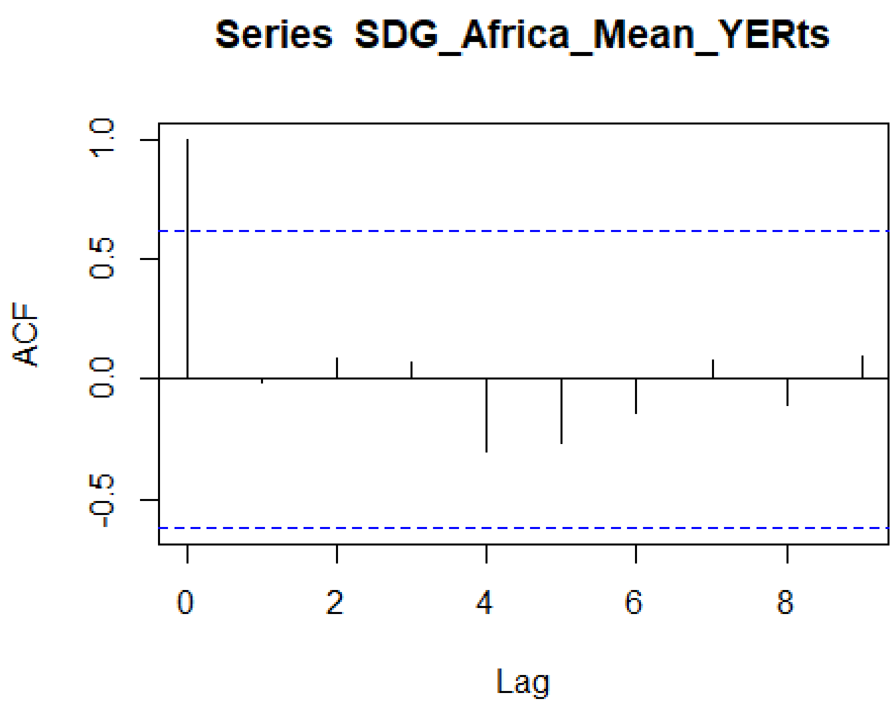
Access to electricity, rural (ER)

We get the data ready by converting it to a `time Series (ts)` data and plot our data before analysis.

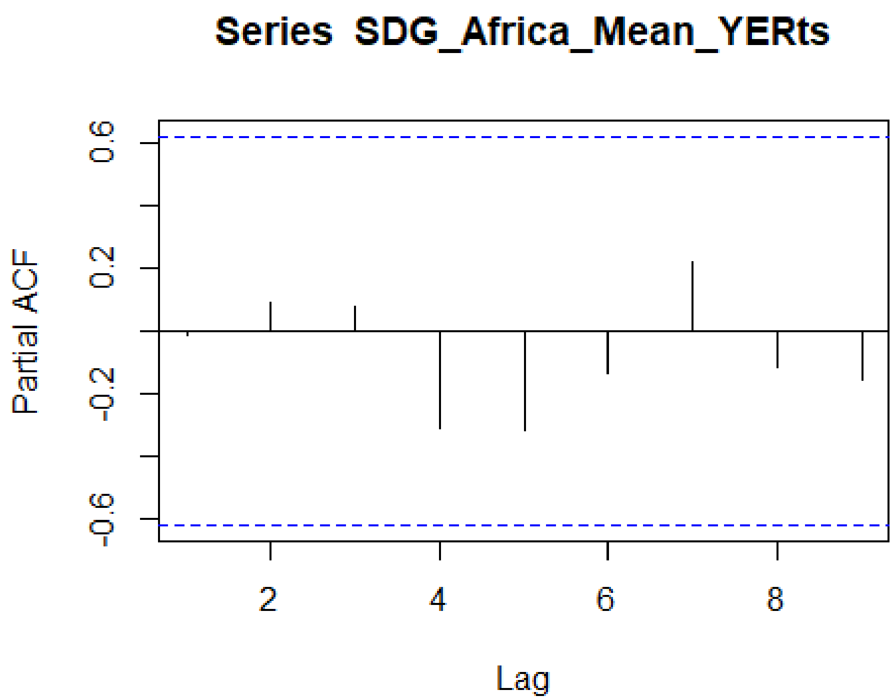
```
SDG_Africa_Mean_YER <- SDG_Africa_Mean[, -c(1,2)]
SDG_Africa_Mean_YERts <- ts(SDG_Africa_Mean_YER, start = 2011, end = 2020,
frequency = 1)
plot(SDG_Africa_Mean_YERts)
```




```
acf(SDG_Africa_Mean_YERts)
```



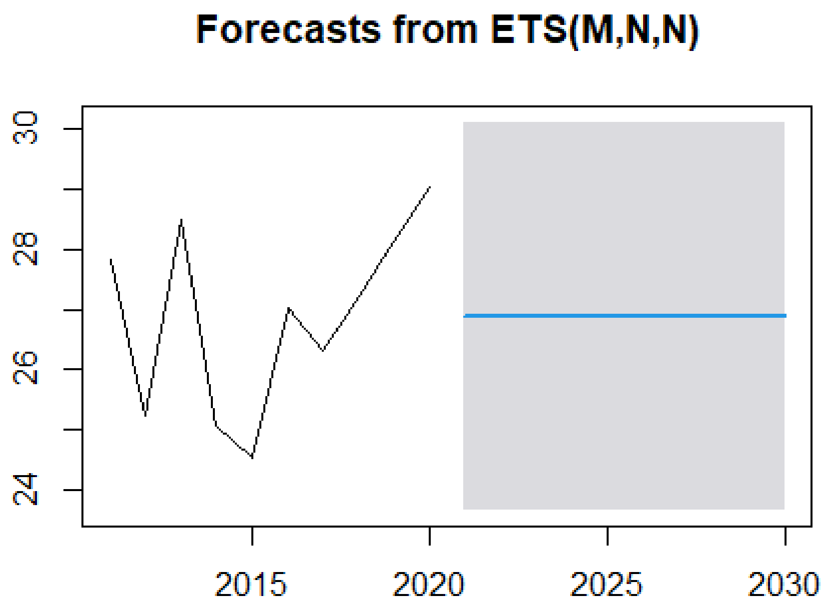

```
pacf(SDG_Africa_Mean_YERts)
```




```
adf.test(SDG_Africa_Mean_YERts)

##
##  Augmented Dickey-Fuller Test
##
## data:  SDG_Africa_Mean_YERts
## Dickey-Fuller = -3.0033, Lag order = 2, p-value = 0.1902
## alternative hypothesis: stationary
```

For the forecast

```
myforecast_AYERts <- forecast(SDG_Africa_Mean_YERts, level=c(95), h=10*1)
plot(myforecast_AYERts)
```

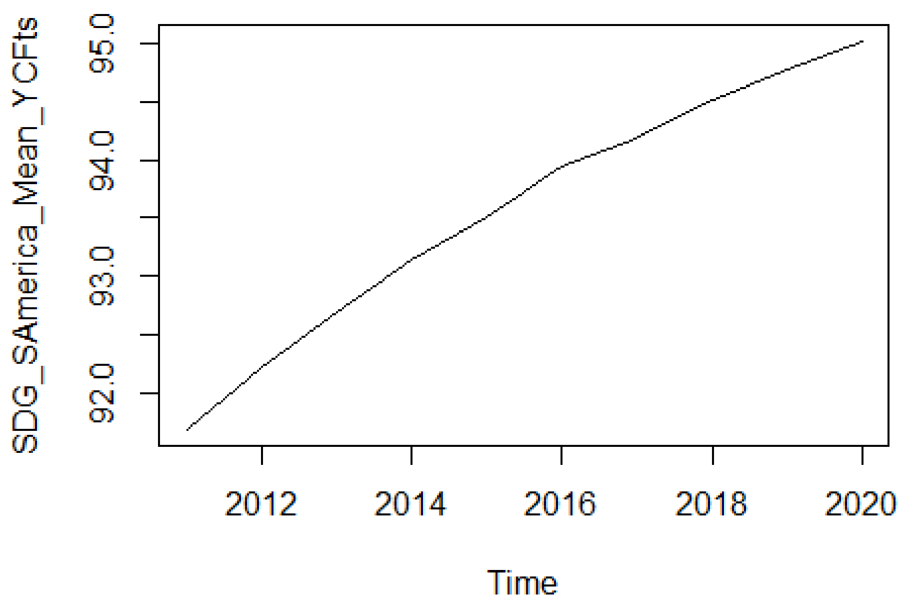


South America

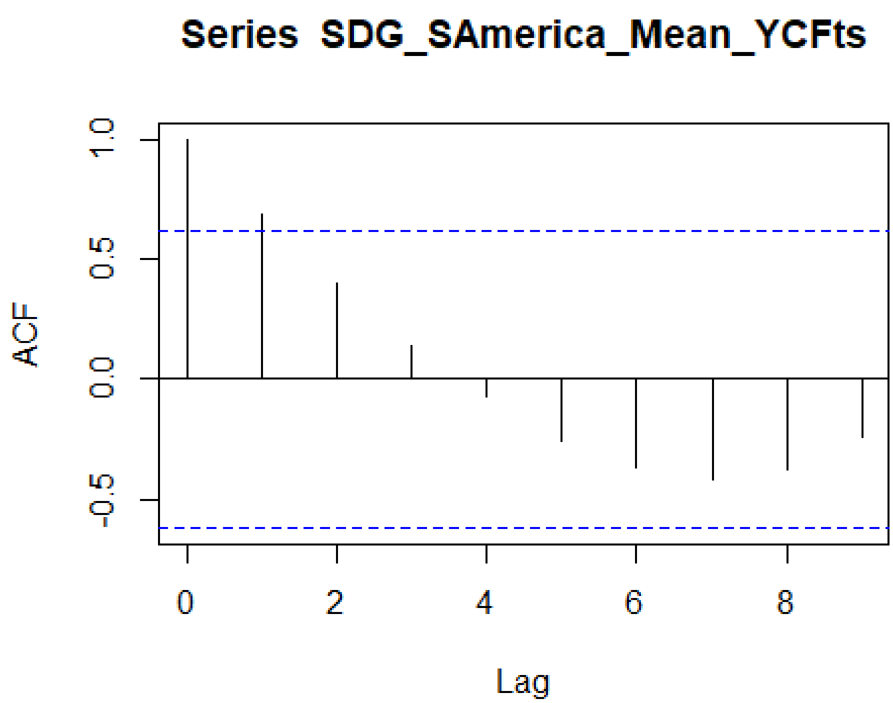
Access to clean fuels and technologies for cooking (CF)

We get the data ready by converting it to a `time Series (ts)` data and plot our data before analysis.

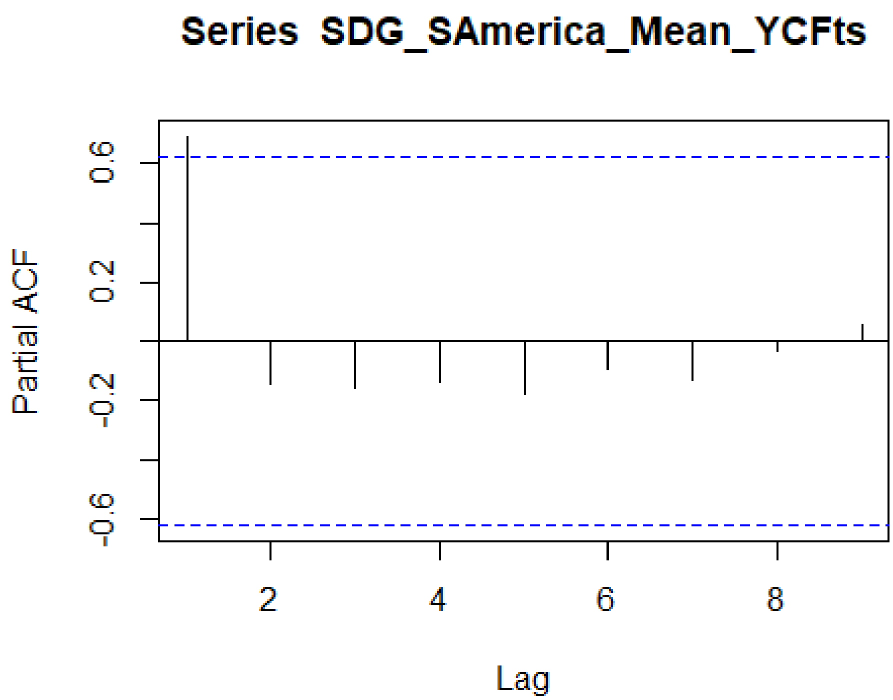
```
SDG_SAmerica_Mean_YCF <- SDG_SAmerica_Mean[,-2:-3]
SDG_SAmerica_Mean_YCFts <- ts(SDG_SAmerica_Mean_YCF, start = 2011, end =
2020, frequency = 1)
plot(SDG_SAmerica_Mean_YCFts)
```




```
acf(SDG_SAmerica_Mean_YCFts)
```



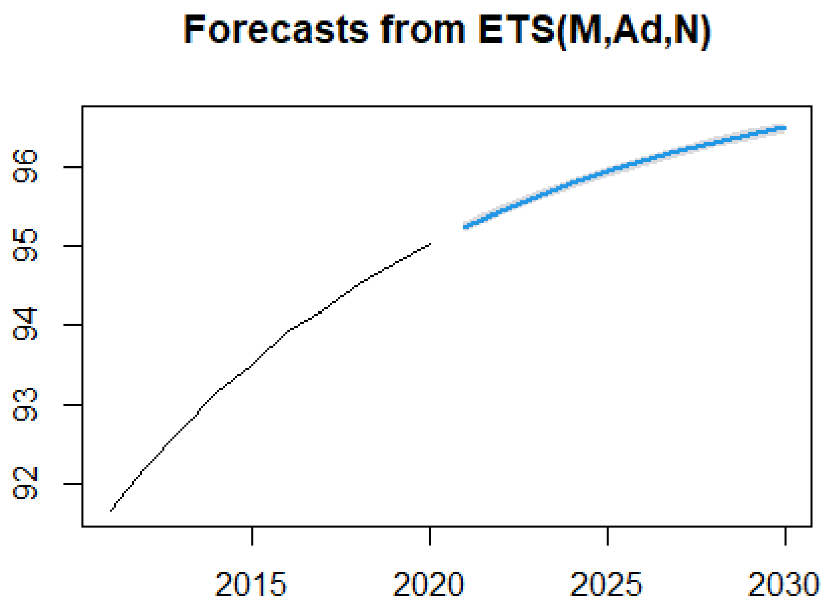

```
pacf(SDG_SAmerica_Mean_YCFts)
```




```
adf.test(SDG_SAmerica_Mean_YCFts)

##
##  Augmented Dickey-Fuller Test
##
## data:  SDG_SAmerica_Mean_YCFts
## Dickey-Fuller = -1.2987, Lag order = 2, p-value = 0.8395
## alternative hypothesis: stationary
```

For the forecast

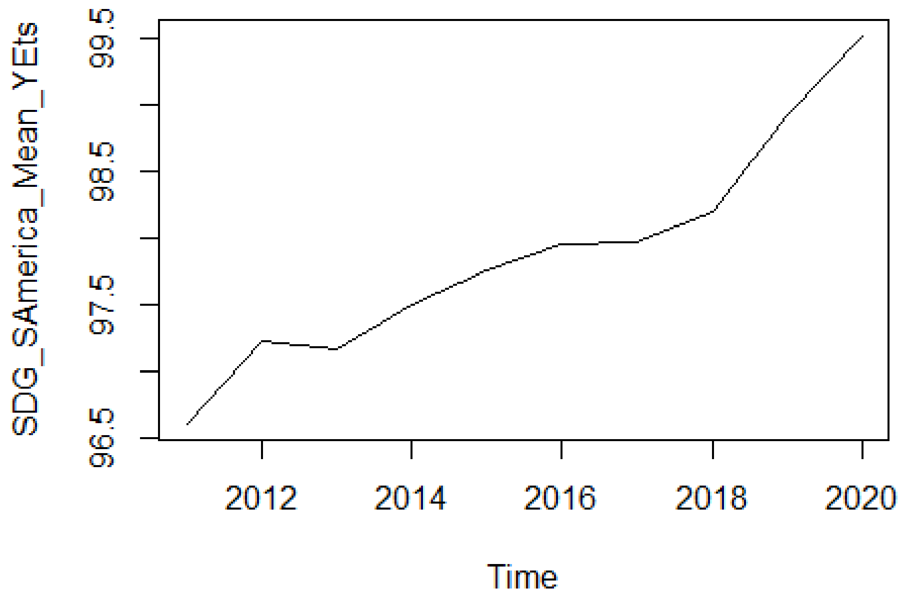
```
myforecast_YCFts <- forecast(SDG_SAmerica_Mean_YCFts, level=c(95), h=10*1)
plot(myforecast_YCFts)
```



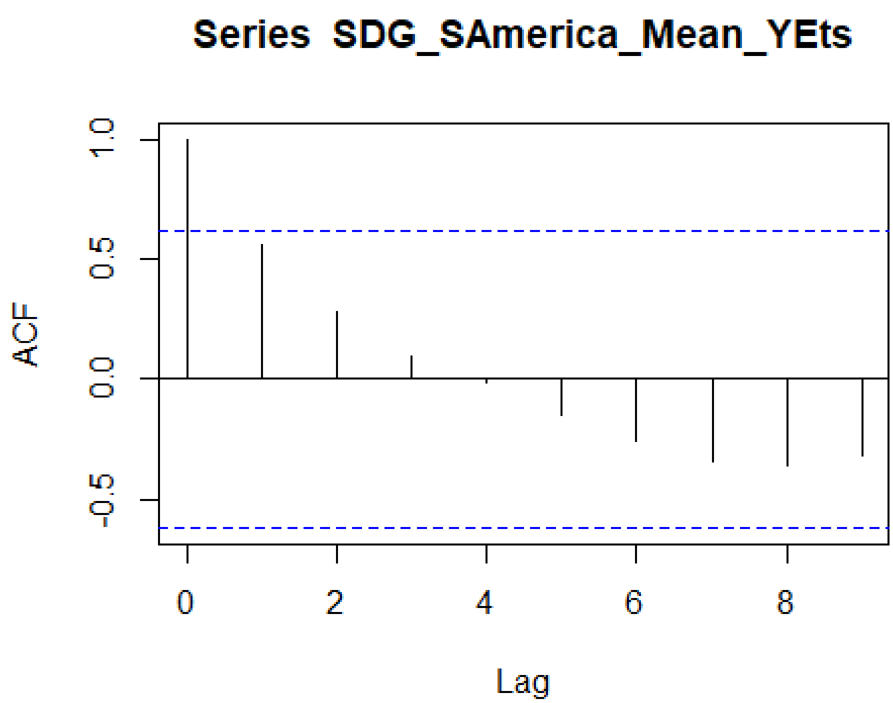
Access to electricity (E)

We get the data ready by converting it to a `time Series (ts)` data and plot our data before analysis.

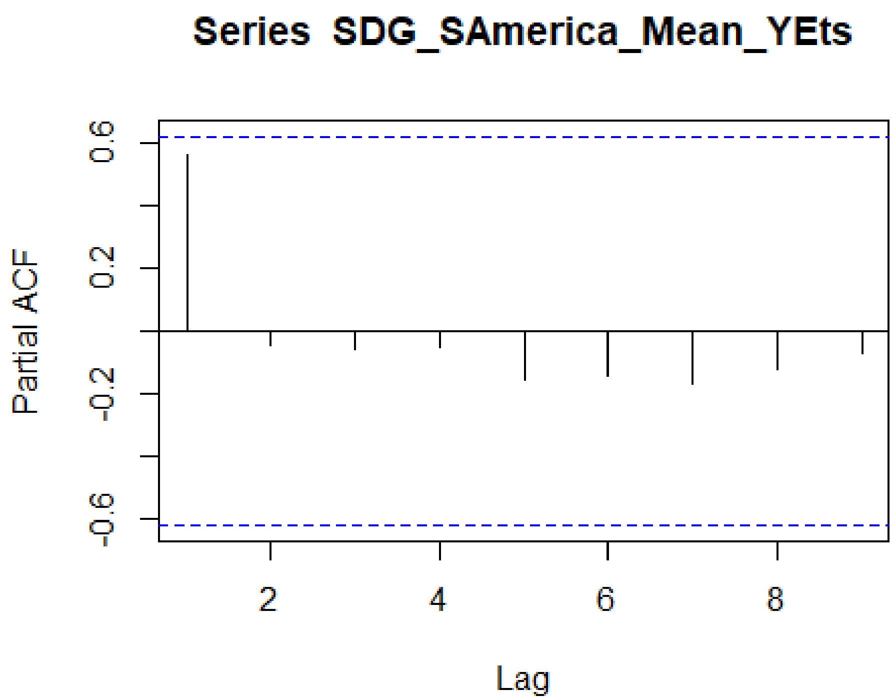
```
SDG_SAmerica_Mean_YE <- SDG_SAmerica_Mean[, -c(1,3)]
SDG_SAmerica_Mean_YEts <- ts(SDG_SAmerica_Mean_YE, start = 2011, end = 2020,
frequency = 1)
plot(SDG_SAmerica_Mean_YEts)
```




```
acf(SDG_SAmerica_Mean_YEts)
```




```
pacf(SDG_SAmerica_Mean_YEts)
```



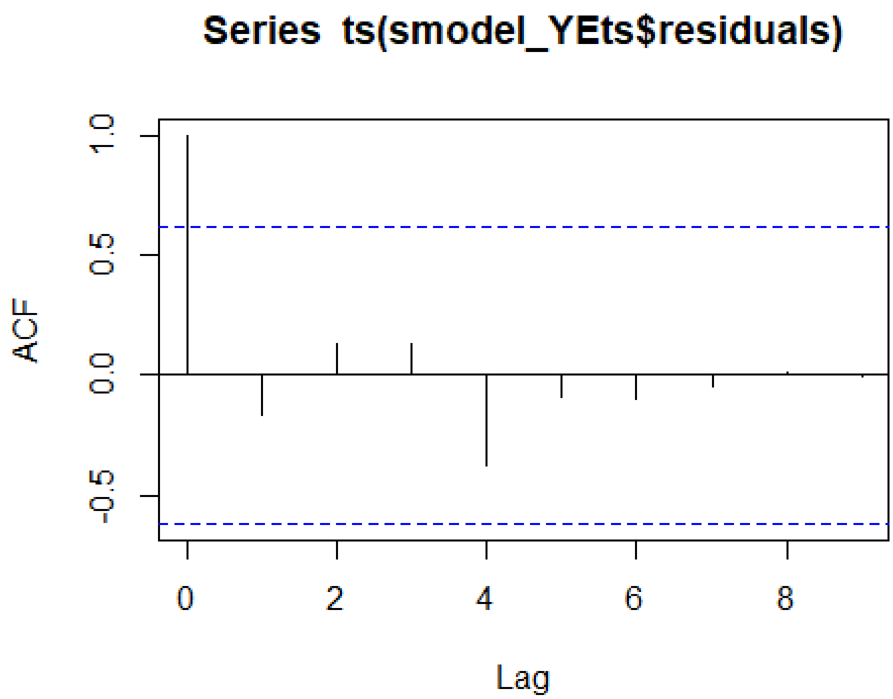

```
adf.test(SDG_SAmerica_Mean_YEts)

##
##  Augmented Dickey-Fuller Test
##
## data:  SDG_SAmerica_Mean_YEts
## Dickey-Fuller = -3.7623, Lag order = 2, p-value = 0.03841
## alternative hypothesis: stationary
```

For the forecast

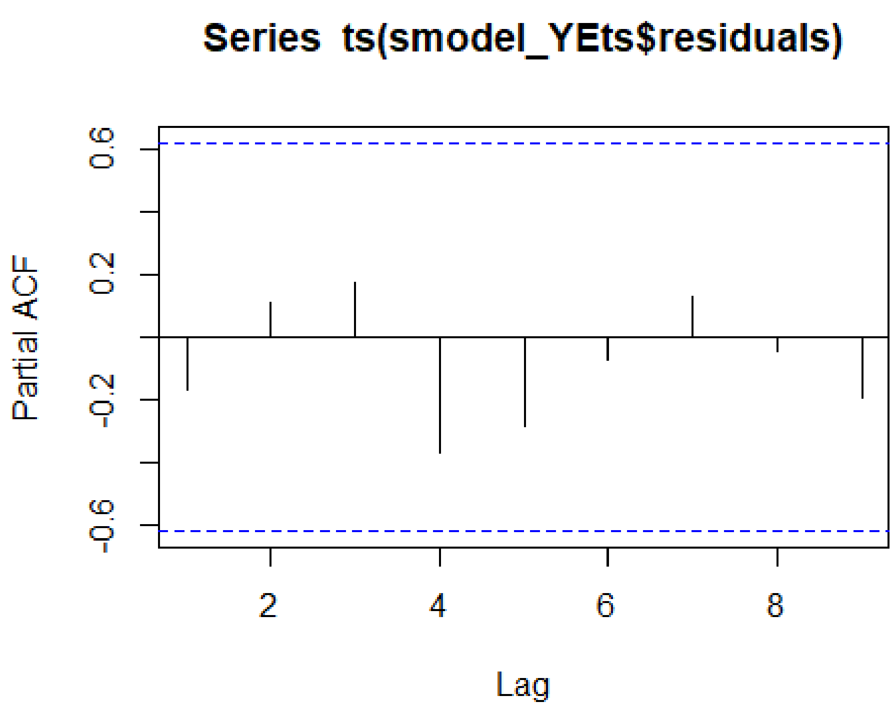
Because the `p-value` is `<0.05`, we need to use the `Arima Model` for our forecast.

```
smodel_YEts <- auto.arima(SDG_Africa_Mean_YEts , ic="aic", trace = TRUE)

##
##  ARIMA(2,1,2) with drift         : Inf
##  ARIMA(0,1,0) with drift         : 38.16022
##  ARIMA(1,1,0) with drift         : 34.9384
##  ARIMA(0,1,1) with drift         : Inf
##  ARIMA(0,1,0)                    : 38.00321
##  ARIMA(2,1,0) with drift         : 36.69087
##  ARIMA(1,1,1) with drift         : 36.79122
##  ARIMA(2,1,1) with drift         : Inf
##  ARIMA(1,1,0)                    : 38.76949
##
##  Best model: ARIMA(1,1,0) with drift
```

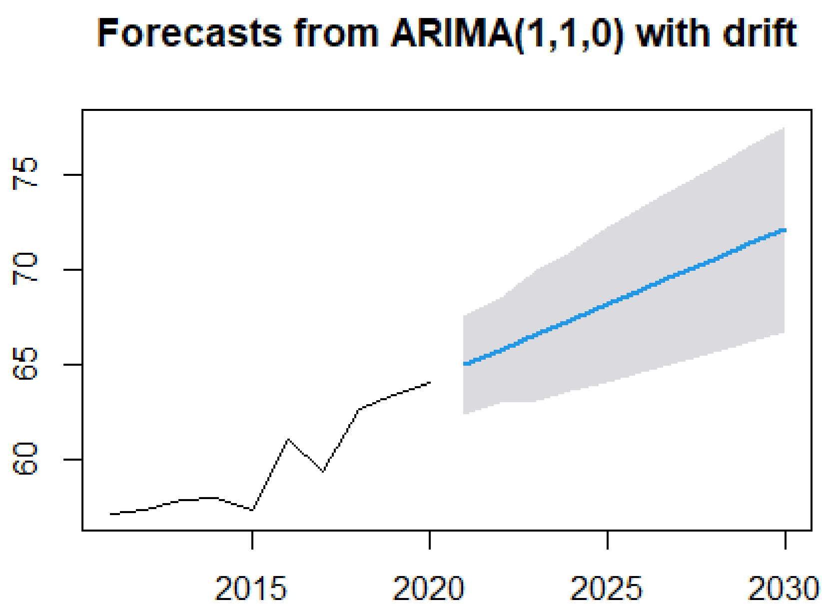
```
acf(ts(smodel_YEts$residuals))
```




```
pacf(ts(smodel_YEts$residuals))
```



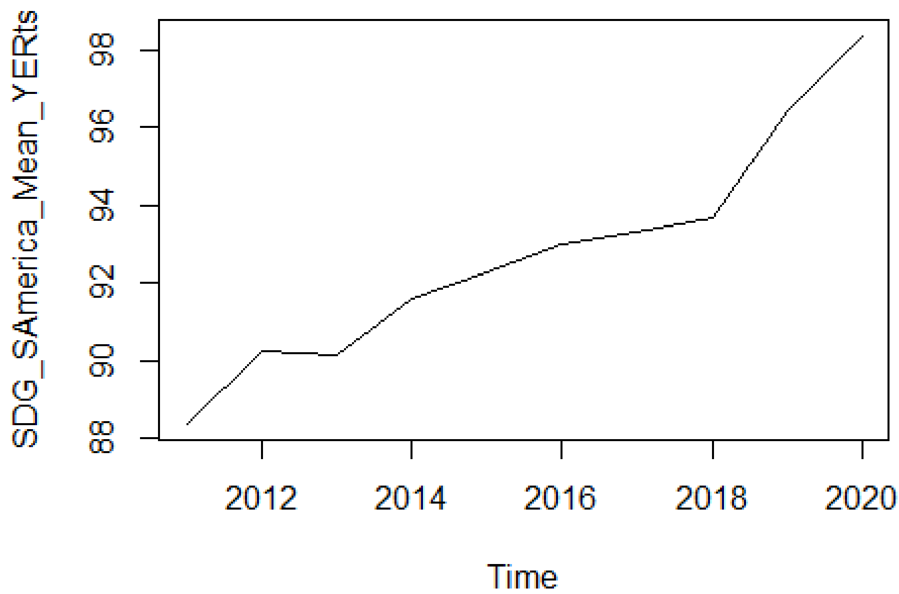
```
myforecast_YEts <- forecast(smodel_YEts, level=c(95), h=10*1)
plot(myforecast_YEts)
```



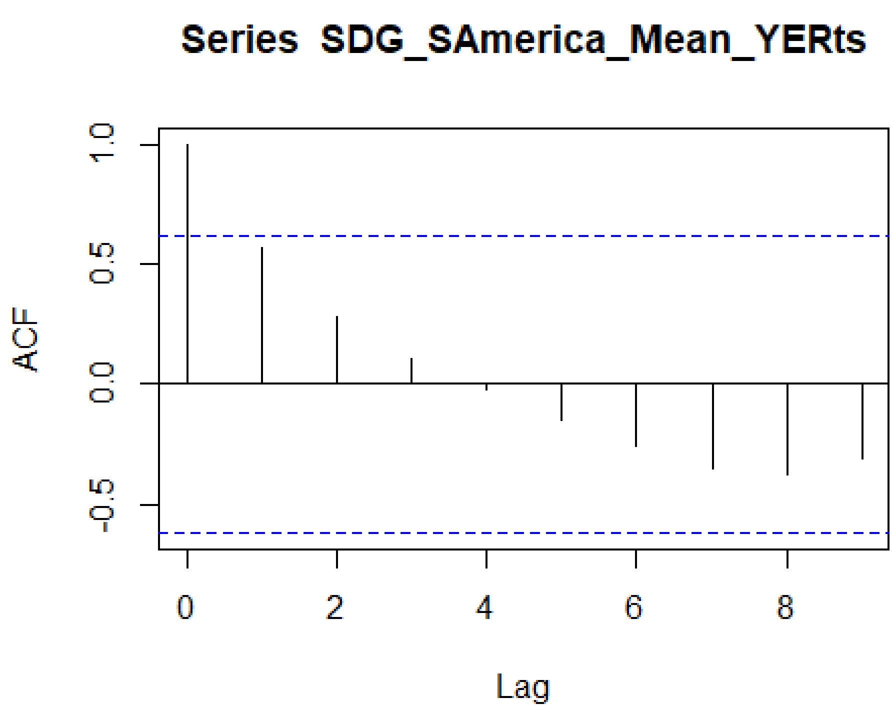
Access to electricity, rural (ER)

We get the data ready by converting it to a `time Series (ts)` data and plot our data before analysis.

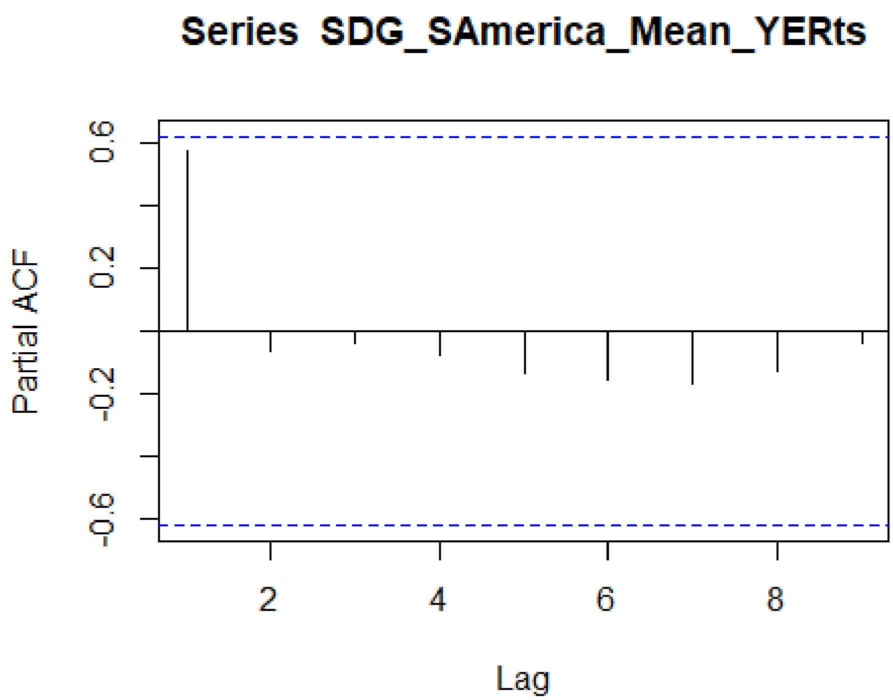
```
SDG_SAmerica_Mean_YER <- SDG_SAmerica_Mean[, -c(1,2)]
SDG_SAmerica_Mean_YERts <- ts(SDG_SAmerica_Mean_YER, start = 2011, end =
2020, frequency = 1)
plot(SDG_SAmerica_Mean_YERts)
```




```
acf(SDG_SAmerica_Mean_YERts)
```




```
pacf(SDG_SAmerica_Mean_YERts)
```

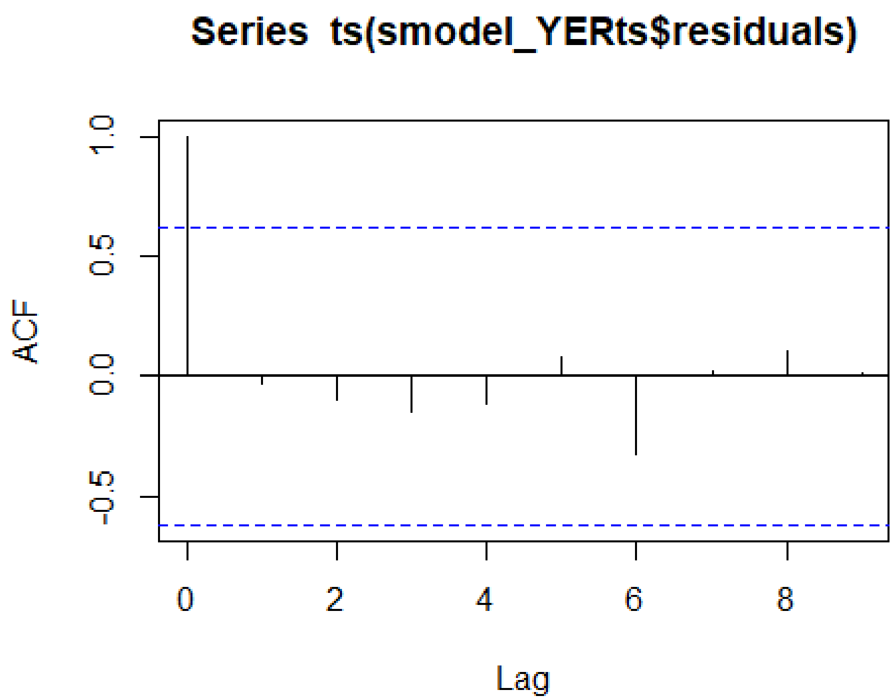



```
adf.test(SDG_SAmerica_Mean_YERts)

##
##  Augmented Dickey-Fuller Test
##
## data:  SDG_SAmerica_Mean_YERts
## Dickey-Fuller = -3.7522, Lag order = 2, p-value = 0.03913
## alternative hypothesis: stationary
```

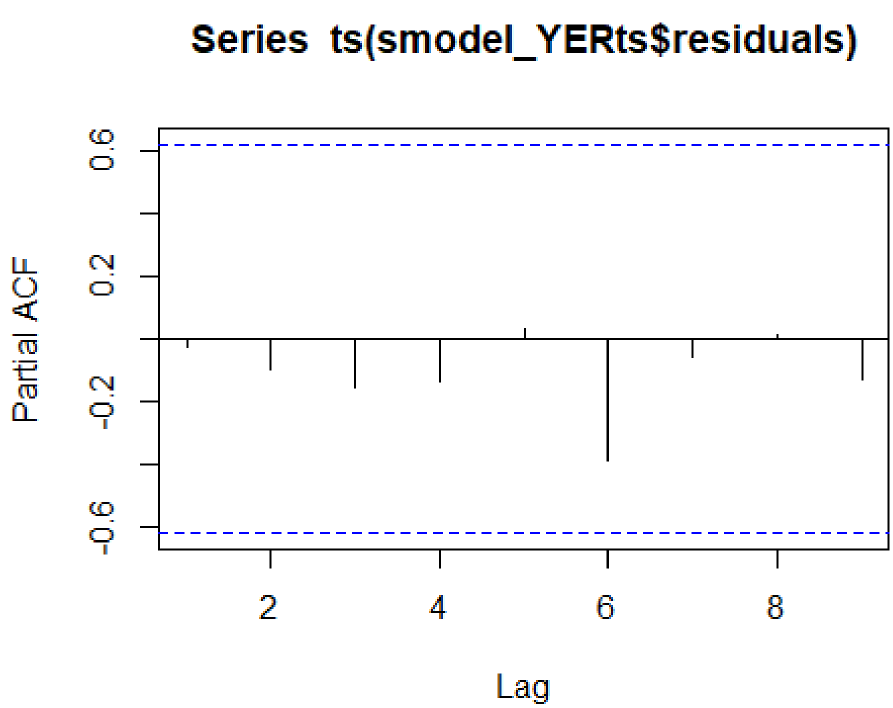
For the forecast

Because the `p-value` is `<0.05`, we need to use the `Arima Model` for our forecast.

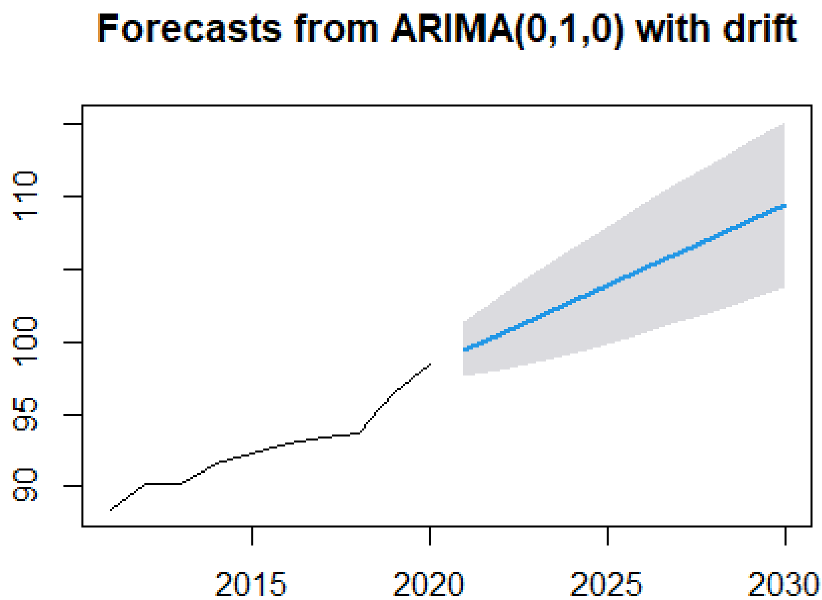
```
smodel_YERts <- auto.arima(SDG_SAmerica_Mean_YERts , ic="aic", trace = TRUE)

##
##  ARIMA(2,1,2) with drift         : Inf
##  ARIMA(0,1,0) with drift         : 27.10457
##  ARIMA(1,1,0) with drift         : 29.08943
##  ARIMA(0,1,1) with drift         : 29.08496
##  ARIMA(0,1,0)                    : 33.73477
##  ARIMA(1,1,1) with drift         : Inf
##
##  Best model: ARIMA(0,1,0) with drift

acf(ts(smodel_YERts$residuals))
```




```
pacf(ts(smodel_YERts$residuals))
```



```
myforecast_YERts <- forecast(smodel_YERts, level=c(95), h=10*1)
plot(myforecast_YERts)
```

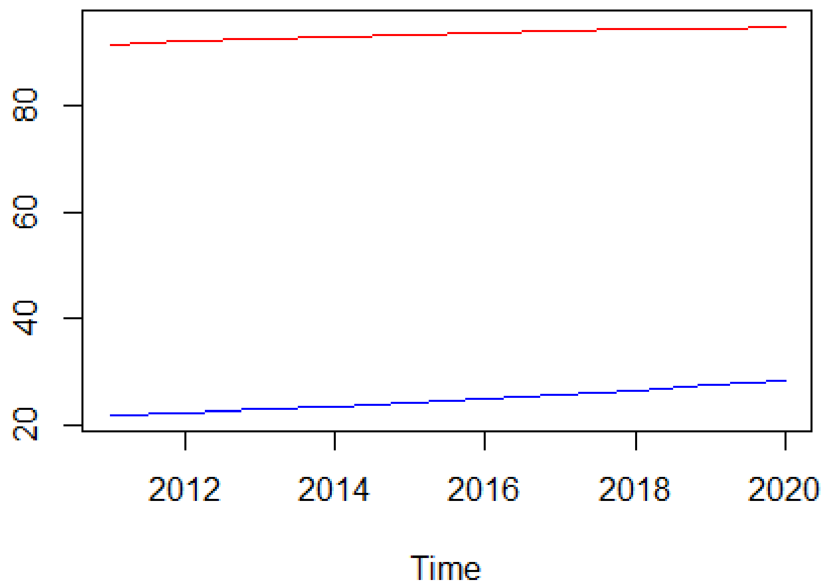


Comparison

I plotted the average of the `indicators` in each continent against each other to see how well each continent performs as opposed to the other.

Access to clean fuels and technologies for cooking (CF)

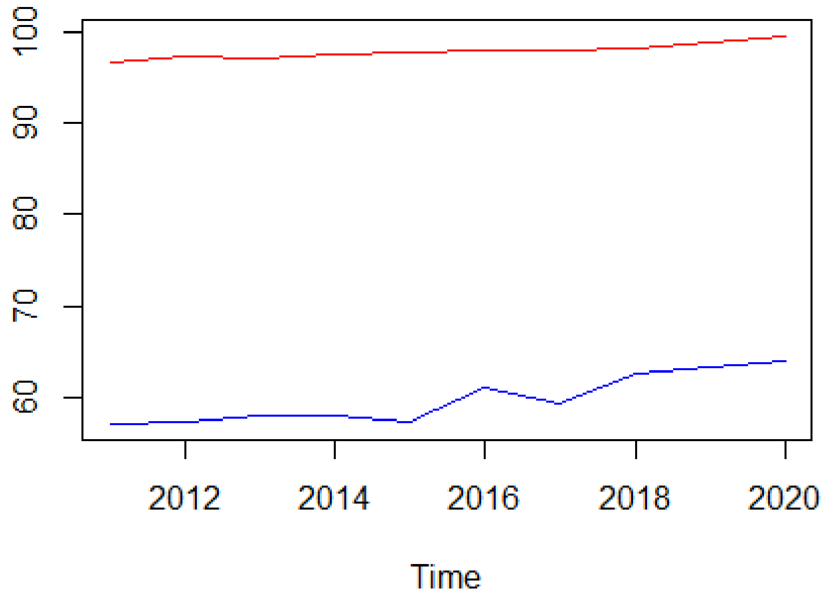
```
ts.plot(SDG_Africa_Mean_YCFts, SDG_SAmerica_Mean_YCFts, col=c("blue", "red"))
```



NOTE: The `blue` plot is for *Africa* while the `red` plot is for *South America*

Access to electricity (E)

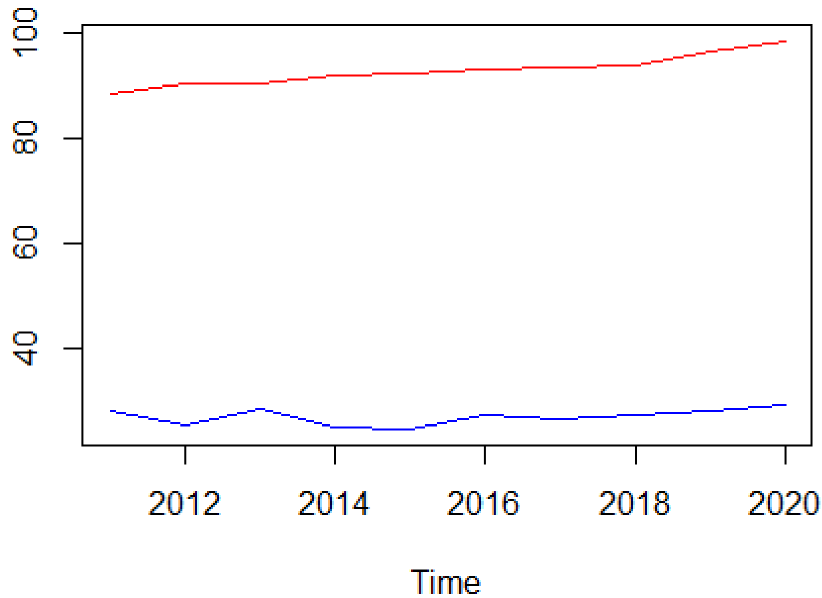
```
ts.plot(SDG_Africa_Mean_YEts, SDG_SAmerica_Mean_YEts, col=c("blue", "red"))
```



NOTE: The `blue` plot is for *Africa* while the `red` plot is for *South America*

Access to electricity, rural (ER)

```
ts.plot(SDG_Africa_Mean_YERts, SDG_SAmerica_Mean_YERts, col=c("blue", "red"))
```



NOTE: The `blue` plot is for *Africa* while the `red` plot is for *South America*