A1

Antonio Roldán Andrade

2025-10-27

Disclaimer

Este es el documento de resolución de la actividad 1 para la asignatura de Análisis Estadístico.

1. Estructura de los datos

1.1 Diccionario de los datos

En este apartado simplemente realizamos la carga de datos y una pequeña ordenacion.

```
# Cargamos en primer lugar el WorldSustainabilty
csv_location <- "WorldSustainabilityDataset.csv"
worldSustainabiltyDataSet <- read.csv(csv_location, sep = ",")

# Cargamos a continuación el DataDictonary
# Para cargar el dataset usamos la libreria readxl ==> install.packages("readxl")
# Realizamos la carga en la parte superior del código, limpieza

xlsx_location <- "Data_Dictionary.xlsx"
data_dictionaryDataSet <- read_excel(xlsx_location)


# Tabla 1 ==> ODS | Codigo Variable/Indicador | Descripcion

table1 <- data_dictionaryDataSet %>%
  select(`Associated SDG GOAL`, Code, Description)

# ODS | Descripcion
table2 <- data_dictionaryDataSet %>%
  select(`Associated SDG GOAL`, Description)

# Ordenamos y renombro por comodidad.
table1 <- table1 %>% arrange(`Associated SDG GOAL`)
table2 <- table2 %>% arrange(`Associated SDG GOAL`)

table1 <- table1 %>% rename( SDG = `Associated SDG GOAL`)
table2 <- table2 %>% rename( SDG = `Associated SDG GOAL`)
```

```
head(table1,4)
```

```
## # A tibble: 4 × 3
##   SDG                   Code      Description
##   <chr>                 <chr>     <chr>
## 1 Classification types Regime    Regime classified considering the competitiven…
## 2 Classification types Income    World Bank assigns the world’s economies to fo…
## 3 Classification types Region    World Region as classified by UN.
## 4 Classification types Continent Continent classification according to UN Conve…
```

```
head(table2,4)
```

```
## # A tibble: 4 × 2
##   SDG                   Description
##   <chr>                 <chr>
## 1 Classification types Regime classified considering the competitiveness of acc…
## 2 Classification types World Bank assigns the world’s economies to four income …
## 3 Classification types World Region as classified by UN.
## 4 Classification types Continent classification according to UN Convention
```

1.2 Fichero de datos

Renombramos de forma manual Country y Country Code ya que no estan en el diccionario

```
worldSustainabiltyDataSet <- worldSustainabiltyDataSet %>%
  rename(Country = `Country.Name`)

worldSustainabiltyDataSet<- worldSustainabiltyDataSet %>%
  rename(CountryCode = `Country.Code`)

# Recorremos los nombres y sustitumos basandonos en los códigos
# de Dictionary

currColNames <- names(worldSustainabiltyDataSet)

updateCurrNames <- function(newColName, currColNames){
# Primero obtenemos los códigos en base al campo field
regimeField_Code <- data_dictionaryDataSet %>%
  filter(str_detect(Field,newColName)) %>%
  select(Code,Field)

# Ahora sustituimos por la nueva
currColNames[grep(regimeField_Code$Code, currColNames)]<- newColName
  return (currColNames)
}

# Usamos la funcion
currColNames <-  updateCurrNames('Regime', currColNames)

currColNames <- updateCurrNames('Income',currColNames)

currColNames <- updateCurrNames('Region', currColNames)

names(worldSustainabiltyDataSet) <- currColNames
```

```
head(names(worldSustainabiltyDataSet),3)
```

```
## [1] "Country"     "CountryCode" "Year"
```

2 Tipos de datos y Posibles Inconsistencias

2.1 Variables Cuantitativas

Comenzaremos comprobando los valores centinelas y los valores nulos o erroneos posibles.

```
cleanNames <- names(worldSustainabiltyDataSet)
# Comenzamos solo por los null
worldSustainabiltyDataSet <- worldSustainabiltyDataSet %>% replace(.=="NULL", NA)
```

```
# Sustituimos por todas las variables

worldSustainabiltyDataSet <- worldSustainabiltyDataSet %>% replace(.=="N/A", NA)

worldSustainabiltyDataSet <- worldSustainabiltyDataSet %>% replace(.=="-", NA)

worldSustainabiltyDataSet <- worldSustainabiltyDataSet %>% replace(.=="NA", NA)

worldSustainabiltyDataSet <- worldSustainabiltyDataSet %>% replace(.=="", NA)

worldSustainabiltyDataSet <- worldSustainabiltyDataSet %>% replace(.=="9999", NA)

worldSustainabiltyDataSet <- worldSustainabiltyDataSet %>% replace(.=="Unknown", NA)

names(worldSustainabiltyDataSet) <- cleanNames
```

2.2 Variables cuantitativas

2.2.1 Países y códigos de países

```
# definimos una función para hacer los cambios de formato
colDataFormatter<- function(colToFix){
  colToFix <- toupper(colToFix)
  colToFix <- str_replace_all(colToFix, " ", "_")
  return (colToFix)
}
```

```
countryCol <- worldSustainabiltyDataSet$Country
countryCodeCol <- worldSustainabiltyDataSet$CountryCode

# Agregamos los cambios al dataset:
countryCol <- colDataFormatter(countryCol)
countryCodeCol <- colDataFormatter(countryCodeCol)
worldSustainabiltyDataSet$Country <- countryCol
worldSustainabiltyDataSet$CountryCode <- countryCodeCol
```

Cambios aplicados:

```
head(unique(countryCol),4)
```

```
## [1] "ARUBA"                "ANGOLA"               "ALBANIA"
## [4] "UNITED_ARAB_EMIRATES"
```

```
head(unique(countryCodeCol),4)
```

```
## [1] "ABW" "AGO" "ALB" "ARE"
```

2.2.2 Continente

```
worldSustainabiltyDataSet$Continent <- colDataFormatter(worldSustainabiltyDataSet$Continent)
```

```
head(unique(worldSustainabiltyDataSet$Continent))
```

```
## [1] "NORTH_AMERICA" "AFRICA"        "EUROPE"        "ASIA"
## [5] "SOUTH_AMERICA" "OCEANIA"
```

2.2.3 Régimen

```
worldSustainabiltyDataSet$Regime <- colDataFormatter(worldSustainabiltyDataSet$Regime)
```

```
head(unique(worldSustainabiltyDataSet$Regime))
```

```
## [1] NA                    "CLOSED_AUTOCRACY"    "ELECTORAL_AUTOCRACY"
## [4] "ELECTORAL_DEMOCRACY" "LIBERAL_DEMOCRACY"
```

2.2.4 Región

```
worldSustainabiltyDataSet$Region <- colDataFormatter(worldSustainabiltyDataSet$Region)
```

```
head(unique(worldSustainabiltyDataSet$Region))
```

```
## [1] "LATIN_AMERICA_AND_CARIBBEAN"      "SUB-SAHARAN_AFRICA"
## [3] "EUROPE_AND_NORTHERN_AMERICA"      "NORTHERN_AFRICA_AND_WESTERN_ASIA"
## [5] "OCEANIA"                          "CENTRAL_AND_SOUTHERN_ASIA"
```

2.2.5 Región

```
worldSustainabiltyDataSet$Income <- colDataFormatter(worldSustainabiltyDataSet$Income)
```

```
head(unique(worldSustainabiltyDataSet$Income))
```

```
## [1] "HIGH_INCOME"         "LOW_INCOME"          "LOWER-MIDDLE_INCOME"
## [4] "UPPER-MIDDLE_INCOME" NA
```

3 Valores Extremos

3.1 Desigualdad (GINI)

```
  # Renombramos las columnas
  # Corresponde a la columna SI.POV.GINI
  worldSustainabiltyDataSet <- worldSustainabiltyDataSet %>% rename( GINI = `Gini.index..Worl
d.Bank.estimate....SI.POV.GINI`)

  # Corresponde al codigo GH.EM.IC.LUF
  worldSustainabiltyDataSet <- worldSustainabiltyDataSet %>% rename( GHE = `Annual.productio
n.based.emissions.of.carbon.dioxide..CO2..measured.in.million.tonnes...GH.EM.IC.LUF`)
```

```
# Creamos una tabla con las 3 columnas a trabajar
# Country GINI YEAR

tblCountryGiniYear <- worldSustainabiltyDataSet %>%
  select(Country, GINI,Year)
# Vistazo rápido de las columnas
summary(tblCountryGiniYear$Country)
```

```
##    Length     Class      Mode
##      3287 character character
```

```
summary(tblCountryGiniYear$GINI)
```

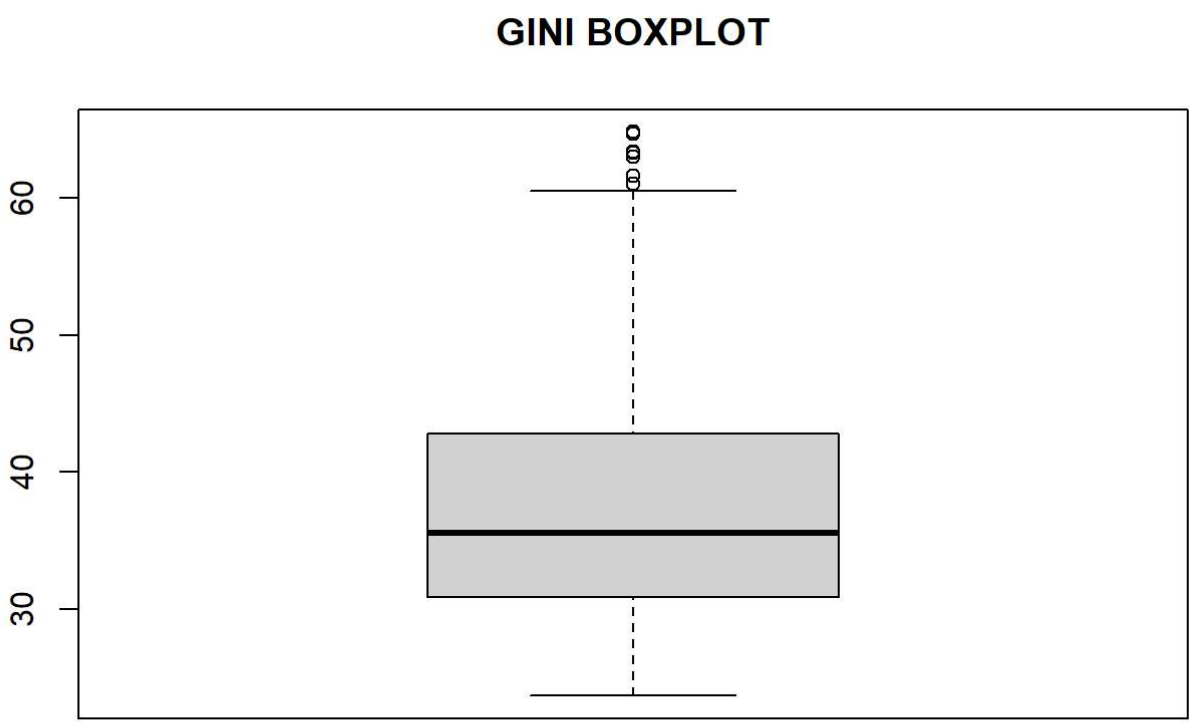
```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
##   23.70   30.90   35.60   37.51   42.80   64.80    1984
```

```
summary(tblCountryGiniYear$Year)
```

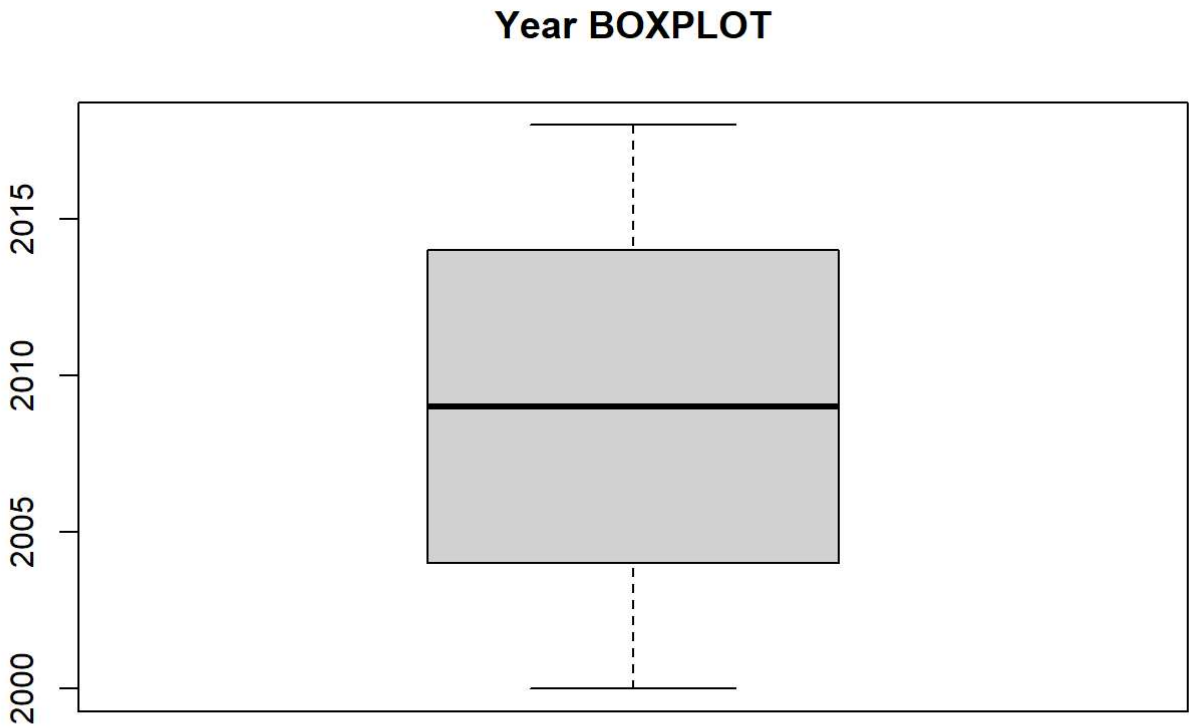
```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##    2000    2004    2009    2009    2014    2018
```

Comenzaremos con la realización de una boxplot para visualizar los datos de forma sencilla:

```
boxplot(tblCountryGiniYear$GINI, main="GINI BOXPLOT")
```




```
boxplot(tblCountryGiniYear$Year, main="Year BOXPLOT")
```



```
# Comprobamos los stats para ver claramente los outliers
boxplot.stats(tblCountryGiniYear$GINI)$out
```

```
## [1] 61.6 64.7 63.3 61.0 64.8 63.0 63.4 63.0
```

```
# Vamos a ver cuantos registr
length(tblCountryGiniYear$GINI)
```

```
## [1] 3287
```

```
length(which(is.na(tblCountryGiniYear$GINI)))
```

```
## [1] 1984
```

```
length(boxplot.stats(tblCountryGiniYear$GINI)$out)
```

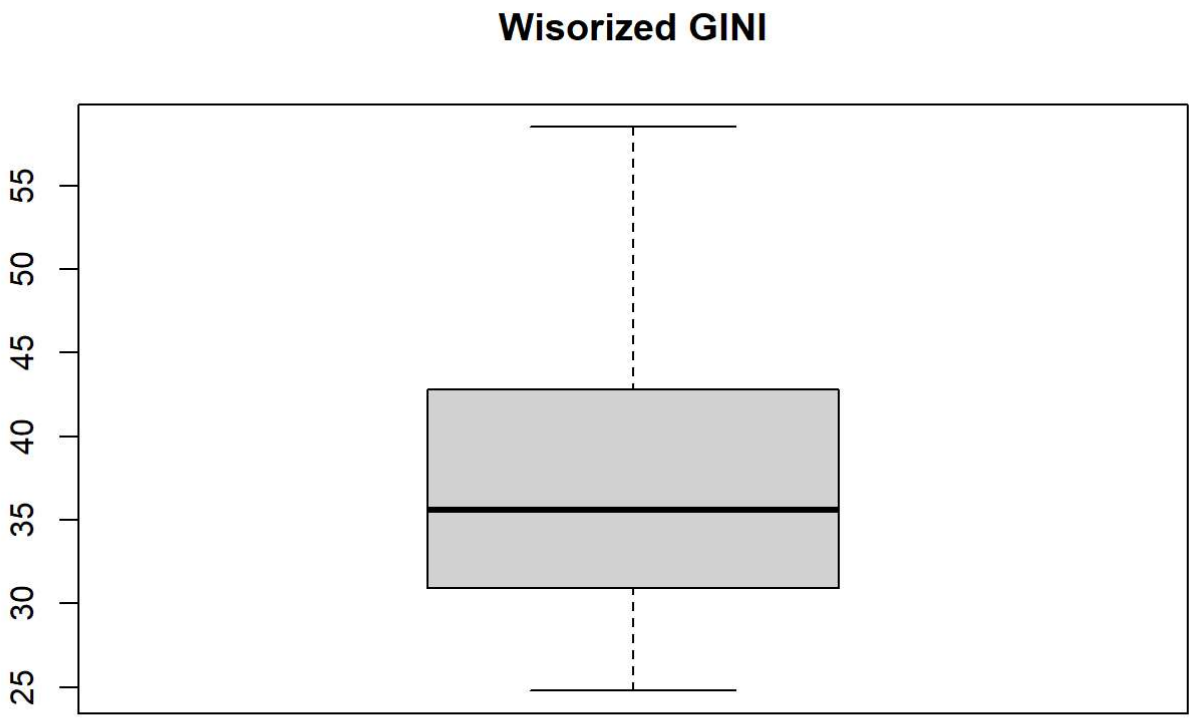
```
## [1] 8
```

Como podemos ver en el “boxplot” de GINI tenemos una gran cantidad de valores atípicos, existiendo una asímetría de datos positiva, puesto que tenemos bastantes valores outliers en GINI, esto conlleva cierta dispersión ya que tenemos valores alejados de la media y mediana. Si nos centramos en los valores del “boxplot” de Year podemos observar como los mismos tienen cierta dispersión.

En cuanto a los valores outliers estos requieren una transformación puesto que aunque son pocos pueden distorsionar el estudio de valores esenciales como son la media/mediana, por lo que aplicamos winsorización:

```
tblCountryGiniYear$GINI <- psych::winsor(
  x = tblCountryGiniYear$GINI,
  trim = 0.01
)

boxplot(tblCountryGiniYear$GINI, main="Wisorized GINI")
```

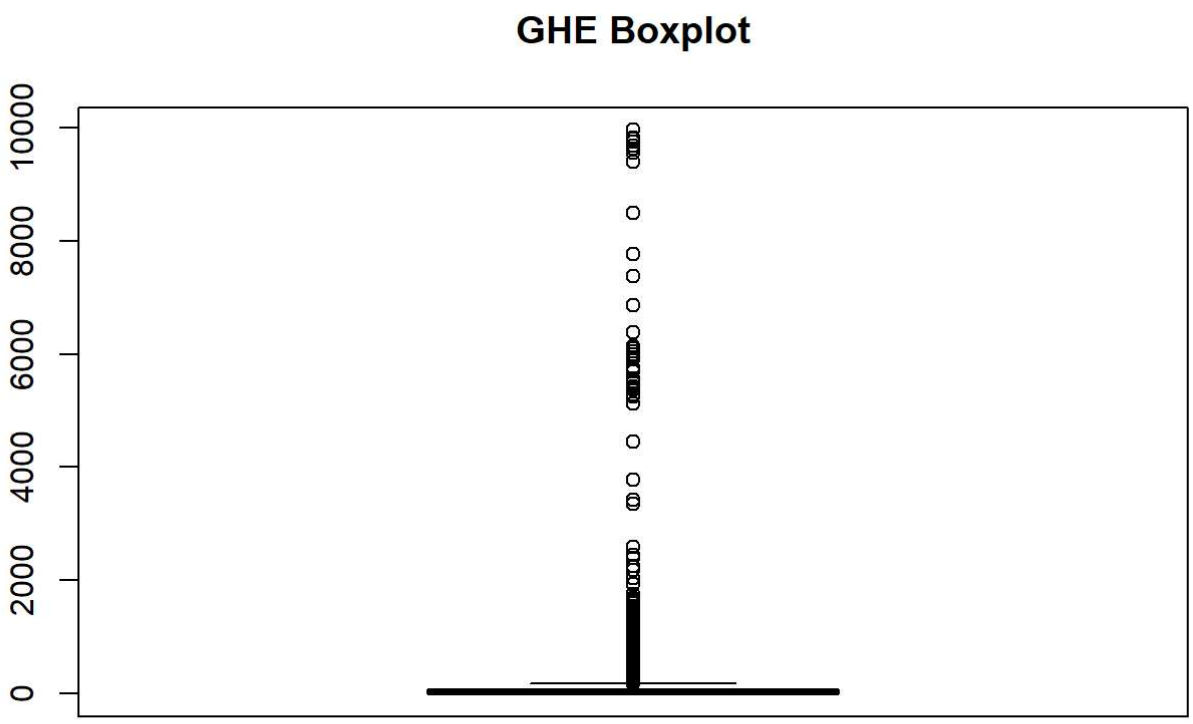


3.2 Green House Emissions

1ª Parte

Estudiaremos los valores outliers de forma similar al apartado 3.1:

```
boxplot(worldSustainabiltyDataSet$GHE, main="GHE Boxplot")
```




```
summary(worldSustainabiltyDataSet$GHE)
```

```
##     Min.  1st Qu.   Median     Mean  3rd Qu.     Max.     NA's
##    0.048    2.426   12.621  174.647   72.012 9956.569        2
```

```
# Comparamos cuantos outliers hay sobre la población.
length(worldSustainabiltyDataSet$GHE)
```

```
## [1] 3287
```

```
length(boxplot.stats(worldSustainabiltyDataSet$GHE)$out)
```

```
## [1] 517
```

Como podemos apreciar existe una gran cantidad de valores outliers dentro de los datos, pero en este caso no tiene sentido su transformación puesto que corresponde a aproximadamente el 16% del total de los mismos. En caso de realizar una transformación únicamente conseguiriamos alterar el resultado del estudio, al eliminar tantos resultados.

2ª Parte

Seleccionamos en una lista compuesta por los países más contaminantes en el año 2018, ordenandola de mayor a menor cantidad de emisiones.

```
# Pillamos los valores de 2018
tblCountryGHE <- worldSustainabiltyDataSet %>%
  filter(Year == 2018) %>%
  select(Country, GHE)

# filtramos ahora por los valores outliers

tblCountryGHE <- tblCountryGHE %>%
  filter(GHE >= min(boxplot.stats(tblCountryGHE$GHE)$out) ) %>%
  arrange(desc(GHE))

head(tblCountryGHE)
```

```
##              Country      GHE
## 1              CHINA 9956.569
## 2      UNITED_STATES 5424.882
## 3              INDIA 2591.324
## 4 RUSSIAN_FEDERATION 1691.360
## 5              JAPAN 1135.688
## 6 IRAN,_ISLAMIC_REP.  755.402
```

4 Correlaciones

```
# Codigos implicados:
# SP.DYN.LE00.IN ==> LE [48]
# SN_ITK_DEFC ==> DEFC [36]
# SP_ACS_BSRVH2O ==> BSRVW [40]
# SI_POV_DAY1 ==> POVL [37]
# SE.PRM.UNER.ZS ==> COSCH [13]
# NY.GDP.MKTP.CD ==> GDP [19]

# Renombramos y extraemos para mayor facilidad

worldSustainabiltyDataSet <- worldSustainabiltyDataSet %>% rename( LE = `Life.expectancy.at.b
irth..total..years....SP.DYN.LE00.IN`)

worldSustainabiltyDataSet <- worldSustainabiltyDataSet %>% rename( DEFC = `Prevalence.of.unde
rnourishment.......SN_ITK_DEFC`)

worldSustainabiltyDataSet <- worldSustainabiltyDataSet %>% rename( BSRVW = `Proportion.of.pop
ulation.using.basic.drinking.water.services.......SP_ACS_BSRVH2O`)

worldSustainabiltyDataSet <- worldSustainabiltyDataSet %>% rename( POVL = `Proportion.of.popu
lation.below.international.poverty.line.......SI_POV_DAY1`)

worldSustainabiltyDataSet <- worldSustainabiltyDataSet %>% rename( COSCH = `Children.out.of.s
chool....of.primary.school.age....SE.PRM.UNER.ZS`)

worldSustainabiltyDataSet <- worldSustainabiltyDataSet %>% rename( GDP = `GDP..current.U
S.....NY.GDP.MKTP.CD`)

colsToCorr <- c("LE", "DEFC", "BSRVW", "POVL", "COSCH", "GDP" )

dataToCorr <- worldSustainabiltyDataSet %>% select(all_of(colsToCorr))

head(dataToCorr)
```

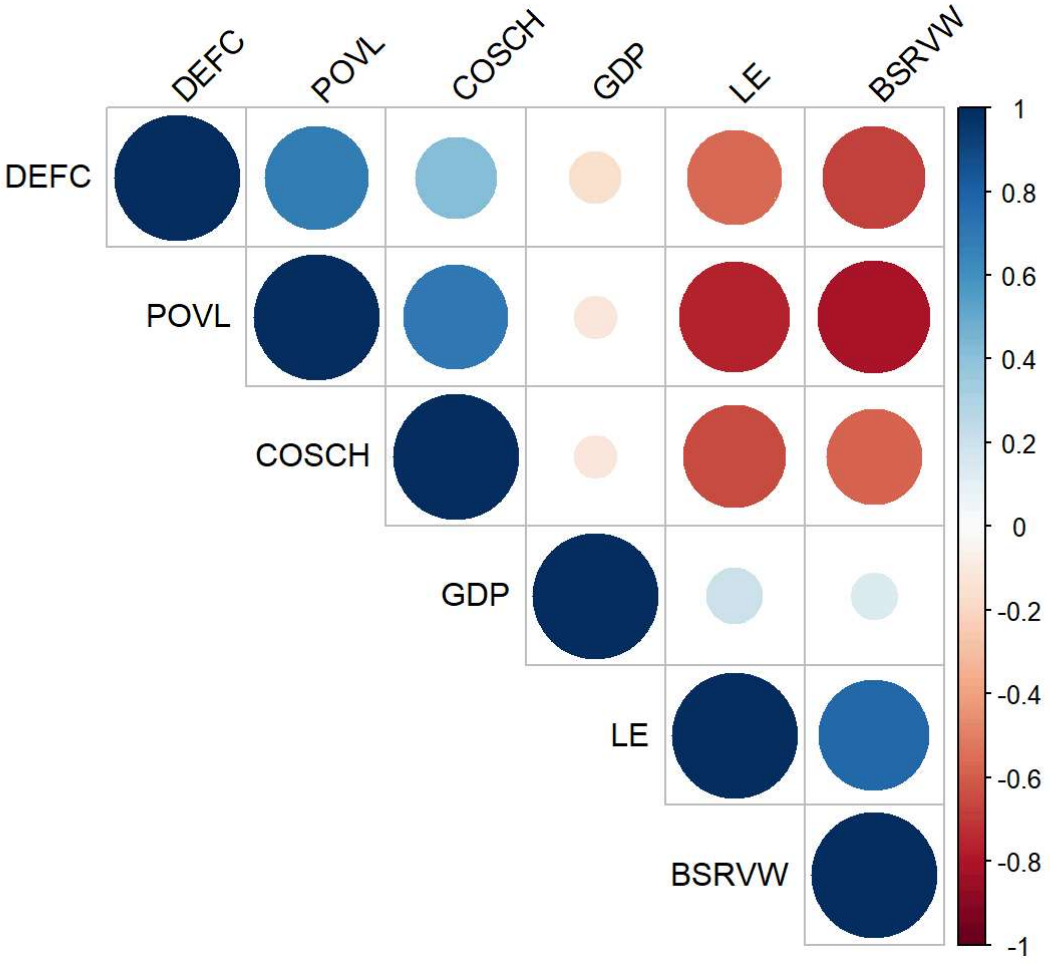
```
##       LE DEFC BSRVW POVL   COSCH        GDP
## 1     NA   NA    NA   NA 1.60268 1873452514
## 2 73.853   NA    NA   NA 0.32258 1920111732
## 3 73.937   NA    NA   NA 1.81634 1941340782
## 4 74.038   NA    NA   NA 3.32156 2021229050
## 5 74.156   NA    NA   NA 2.17652 2228491620
## 6 74.287   NA    NA   NA 1.64077 2330726257
```

```
corrMatrix <- cor(dataToCorr,
                  use = "pairwise.complete.obs") #Ya que tiene elementos NA
corrMatrix <- round(corrMatrix,2) # Para tratar con 2 decimales

corrMatrix
```

```
##           LE  DEFC BSRVW  POVL COSCH   GDP
## LE     1.00 -0.57  0.77 -0.77 -0.66  0.20
## DEFC  -0.57  1.00 -0.67  0.68  0.42 -0.17
## BSRVW  0.77 -0.67  1.00 -0.80 -0.58  0.14
## POVL  -0.77  0.68 -0.80  1.00  0.69 -0.12
## COSCH -0.66  0.42 -0.58  0.69  1.00 -0.12
## GDP    0.20 -0.17  0.14 -0.12 -0.12  1.00
```

```
# Usamos la libreria corrplot para mostrar de forma simplificada la matriz
corrplot(corrMatrix,
         type = "upper",
         order = "hclust",
         tl.col = "black",
         tl.srt = 45
         )
```



Por los resultados obtenidos existe cierta correlación, una correlación relativamente elevada puesto que la mayor correlación que existe es una correlación negativa entre las variables POVL (población viviendo bajo el umbral de la pobreza) y LE (Expectativa de Vida) y POVL con BSRVW (Población que usa fuentes básicas de agua), siendo la primera de -0.77 (POVL <-> LE) y -0.8 (POVL <-> BSRW).

Es decir, la esperanza de vida está fuertemente correlacionada con los indicadores sociales y de servicios básicos (Pobreza, Educación y Acceso al Agua) más que con el indicador económico crudo (PIB). Esto sugiere que la inversión en bienestar social y servicios esenciales es más efectiva para mejorar la salud de la población que el crecimiento económico general por sí solo.

5 Imputación

Con respecto a este apartado usaremos las medias de los vecinos más cercanos usando las variables con una mayor correlación, es decir, mediante las conclusiones obtenidas en el apartado anterior tenemos que las variables a utilizar serán: BSRVW, POVL y COSCH puesto que son aquellas con una correlación lo suficientemente alta.

```
# Comprobamos que columnas estan vacías para aplicarle la imputación
tblNAYear <- worldSustainabiltyDataSet %>%
  group_by(Year) %>%
  filter(all(is.na(LE)) == TRUE) %>%
  select(Year) %>%
  ungroup()

unique(tblNAYear$Year)
```

```
## [1] 2000
```

```
# Creamos una tabla con los LE del año 2001 ya que no puede tener NA a la hora de hacer kNN
datoskNNImputados <- worldSustainabiltyDataSet %>% filter(Year == 2001) %>%
  select(LE, BSRVW, POVL, COSCH )

# 2. Aplicar kNN. Imputamos LE (variable)
datoskNNClean <- kNN(
  data = datoskNNImputados,
  variable = c("LE"),
  k = 5,
  dist_var = c("BSRVW", "POVL", "COSCH")
)
```

```
##    BSRVW     POVL    COSCH    BSRVW     POVL    COSCH
##  29.0000   0.0000   0.0000 100.0000  68.4000  69.4751
```

```
#Aqui tenemos los datos de KNN con datos ya imputados
head(datoskNNClean$LE)
```

```
## [1] 73.853 47.059 74.288 74.544 73.755 71.800
```

```
# Preparamos la sustitución agregando una columna
worldSustainabiltyDataSet$LE2000 <- datoskNNClean$LE

# Hacemos un mutate para sustituir los valores del año 2000 por los valores guardados en LE20
00
worldSustainabiltyDataSet <- worldSustainabiltyDataSet %>%
  mutate(
    LE = case_when(
      # Condición: Si el año es 2000, usa el valor imputado
      Year == 2000 ~ LE2000,
      # Si el año no es 2000, mantén el valor original de la columna LE
      TRUE ~ LE
    )
  ) %>%
  select(-LE2000) # Eliminar la columna auxiliar
```

6 Tabla resumen

```
# Variables relacionadas:
# SI_POV_DAY1 --> POVL
# GINI
# LE

# Se pide calcular las medidas de tendencia central y dispersión (robustas y no robustas)



# Datos: Agrupamos los datos por regiones, mostramos solo los datos más recientes.

# Comenzamos agrupando los distintos datos:

recentData <- worldSustainabiltyDataSet %>%
  group_by(Region) %>%
  filter(Year == max(Year, na.rm = TRUE)) %>%
  select(POVL, GINI, LE, Region)


# Realizamos los cálculos al mismo tiempo y después separamos en tablas
recentData <- recentData %>%
  summarise(
    # Aplicar las cuatro funciones a las variables definidas
    across(
      .cols = all_of(c('LE', 'GINI', 'POVL')),
      .fns = list(
        # TB1: Tendencia Central
        Media = ~ mean(.x, na.rm = TRUE), # No Robusta
        Mediana = ~ median(.x, na.rm = TRUE), # Robusta

        # Dispersión
        Desviacion_Estandar = ~ sd(.x, na.rm = TRUE), # No Robusta
        Desviacion_Absoluta_Mediana = ~ mad(.x, na.rm = TRUE) # Robusta
      ),
      .names = "{.col}_{.fn}" #  columnas dinámicas ('LE_Media', 'GINI_Mediana',... )
    ),
    .groups = 'drop' # No requerimos hacer agrupaciones tras realizar el cálculo.
  )

# Tabla 1 : Medidas de tendencia Central (media, mediana)

centralTendencyTable <- recentData %>%
  select(
    Region,
    ends_with("Media"),
    ends_with("Mediana")
  )

print(centralTendencyTable)
```

```
## # A tibble: 8 × 10
##   Region        LE_Media GINI_Media POVL_Media LE_Mediana LE_Desviacion_Absolu…¹
##   <chr>            <dbl>      <dbl>      <dbl>      <dbl>                   <dbl>
## 1 CENTRAL_AND_…     72.5       32.3      1.38        71.5                    1.98
## 2 EASTERN_AND_…     75.4       37.3      3.1         75.5                    8.73
## 3 EUROPE_AND_N…     79.1       31.0      0.448       80.9                    3.61
## 4 LATIN_AMERIC…     74.9       45.4      3.21        75.0                    2.68
## 5 NORTHERN_AFR…     76.0       34.3      1.18        76.5                    3.04
## 6 OCEANIA           74.3      NaN      NaN           71.8                    4.43
## 7 SUB-SAHARAN_…     62.6       38.6     33.1         63.0                    4.40
## 8 <NA>             NaN        NaN       38.5         NA                     NA
## # ℹ abbreviated name: ¹LE_Desviacion_Absoluta_Mediana
## # ℹ 4 more variables: GINI_Mediana <dbl>,
## #   GINI_Desviacion_Absoluta_Mediana <dbl>, POVL_Mediana <dbl>,
## #   POVL_Desviacion_Absoluta_Mediana <dbl>
```

```
# Tabla 2 : Medias de dispersión (desviación estandar, desviación absoluta con respecto a la mediana)

dispersionTable <- recentData %>%
  select(
    Region,
    ends_with("Desviacion_Estandar"),
    ends_with("Desviacion_Absoluta_Mediana")
  )

print(dispersionTable)
```

```
## # A tibble: 8 × 7
##   Region    LE_Desviacion_Estandar GINI_Desviacion_Esta…¹ POVL_Desviacion_Esta…²
##   <chr>                      <dbl>                  <dbl>                   <dbl>
## 1 CENTRAL_…                   3.31                   6.73                    2.03
## 2 EASTERN_…                   6.16                   3.23                    3.64
## 3 EUROPE_A…                   3.51                   4.61                    0.733
## 4 LATIN_AM…                   3.31                   4.35                    3.91
## 5 NORTHERN…                   3.13                   5.78                    1.95
## 6 OCEANIA                     6.44                  NA                      NA
## 7 SUB-SAHA…                   5.03                   8.64                   22.2
## 8 <NA>                       NA                     NA                      NA
## # ℹ abbreviated names: ¹GINI_Desviacion_Estandar, ²POVL_Desviacion_Estandar
## # ℹ 3 more variables: LE_Desviacion_Absoluta_Mediana <dbl>,
## #   GINI_Desviacion_Absoluta_Mediana <dbl>,
## #   POVL_Desviacion_Absoluta_Mediana <dbl>
```

```
# Como fin de práctica guardamos los cambios realizados en el csv
write.csv(worldSustainabiltyDataSet, csv_location)
```