A1.Ingresos Sanitarios por Paises

Alicia Perdices Guerra

3 de mayo, 2021

Contents

1.PROCESAMIENTO DE LOS DATOS.

- En primer lugar leemos el fichero:

```
ingreso<-read.csv("C:/temp/IngresosSanitario_Financiacion.csv",sep= ",")
```

- Realicemos una breve inspección de los datos

```
str(ingreso)
```

```
## 'data.frame':    220 obs. of  6 variables:
##  $ TIME              : int  2009 2009 2009 2009 2009 2009 2009 2009 2009 2009 ...
##  $ GEO               : Factor w/ 22 levels "Belgium","Croatia",..: 1 4 5 8 6 11 19 2 3 12 ...
##  $ UNIT              : Factor w/ 1 level "Million euro": 1 1 1 1 1 1 1 1 1 1 ...
##  $ ICHA11_FS         : Factor w/ 1 level "All revenues of financing schemes": 1 1 1 1 1 1 1 1 1 1 ..
##  $ Value             : Factor w/ 159 levels ":","1 042.18",..: 101 1 1 76 147 1 153 1 1 1 ...
##  $ Flag.and.Footnotes: Factor w/ 2 levels "","b": 1 1 1 1 1 1 1 1 1 1 ...
```

```
colnames(ingreso) #Nombre de las variables
```

```
## [1] "TIME"               "GEO"                "UNIT"
## [4] "ICHA11_FS"          "Value"              "Flag.and.Footnotes"
```

```
nrow(ingreso) #Número de registros
```

```
## [1] 220
```

```
ncol(ingreso) #Número de variables
```

```
## [1] 6
```

*Observamos las siguientes variables:

- **TIME**: variable cuantitativa. Indica el año en el que se ha realizado la medida, en este caso el valor de la variable "Value". Se ha cargado bien como número entero.
- **GEO**: variable cualitativa. Indica el país o región en el que se ha realizado la medida.Se ha cargado bien como factor.
- **UNIT**: variable cualitativa. Indica la medida de la variable valor.Se ha cargado bien como factor.
- **ICHA11_FS**: variable cualitativa. Indica que la variable "Value" corresponde a todo tipo de ingresos por paises.
- **Value**: Variable cuantitativa. Indica el valor en Millones de Euros de estos ingresos.Se ha cargado mal como factor. Haremos la transformación a valor numérico.
- **Fal.and.footnotes**. Notas sobre etiquetas. Eliminamos esta columna.

*Años de las mediciones:

```
unique(ingreso$TIME)
```

```
##  [1] 2009 2010 2011 2012 2013 2014 2015 2016 2017 2018
```

*Paises:

```r
unique(ingreso$GEO)
```

```
##  [1] Belgium
##  [2] Czechia
##  [3] Denmark
##  [4] Germany (until 1990 former territory of the FRG)
##  [5] Estonia
##  [6] Ireland
##  [7] Spain
##  [8] Croatia
##  [9] Cyprus
## [10] Latvia
## [11] Lithuania
## [12] Luxembourg
## [13] Hungary
## [14] Malta
## [15] Poland
## [16] Slovenia
## [17] Finland
## [18] Sweden
## [19] Iceland
## [20] Norway
## [21] Switzerland
## [22] United Kingdom
## 22 Levels: Belgium Croatia Cyprus Czechia Denmark Estonia ... United Kingdom
```

*Unidad de las mediciones:

```r
unique(ingreso$UNIT)
```

```
## [1] Million euro
## Levels: Million euro
```

*Variable que indica que la variable value corresonde a todo tipo de ingresos por paises.

```r
unique(ingreso$ICHA11_FS)
```

```
## [1] All revenues of financing schemes
## Levels: All revenues of financing schemes
```

- Eliminamos la columna Fal.and.footnotes.

```r
ingreso<-ingreso[,-6]
```

- Tendríamos que convertir la columna Value a numérico porque se ha cargado como factor y es erróneo. El resto de variables tienen el tipo correcto.

```r
ingreso$Value<-as.character(ingreso$Value )
ingreso$Value<-(gsub(',','.',ingreso$Value) )
ingreso$Value<-(gsub(' ','',ingreso$Value) )
ingreso$Value<-as.numeric(ingreso$Value)
```

```
## Warning: NAs introducidos por coerción
```

- Comprobamos que valores tenemos en la columna **Value**:

```r
table(ingreso$Value, useNA = "ifany")
```

```
##
```

```
##    795.04    889.47    898.48    925.55     932.1    939.05    945.12    970.49
##         1         1         1         1         1         1         1         1
##    991.84   1042.18   1045.15    1108.6    1109.7   1137.77    1211.8   1227.09
##         1         1         1         1         1         1         1         1
##   1234.64   1249.79   1265.08    1274.3   1274.97   1277.15   1289.82    1318.9
##         1         1         1         1         1         1         1         1
##   1350.33   1410.14   1430.98   1522.48   1572.66   1609.73   1734.68   1804.22
##         1         1         1         1         1         1         1         1
##   1810.89   1862.21   2265.58   2423.88   2463.12   2570.38   2581.36   2638.25
##         1         1         1         1         1         1         1         1
##    2708.9   2732.83   2751.04   2850.33   2907.78   2972.85   2987.17   3027.78
##         1         1         1         1         1         1         1         1
##   3174.33   3183.72   3199.66    3309.2   3327.75   3428.78   3520.39   3524.46
##         1         1         1         1         1         1         1         1
##   3797.15   6832.62   7396.44   7428.99   7431.57   7488.05    7642.3   7730.72
##         1         1         1         1         1         1         1         1
##   8123.68   8531.31    8963.5  15871.89  16650.25  17200.09  18261.42  18505.51
##         1         1         1         1         1         1         1         1
##  18850.22  19231.95     19271  20034.38   20143.2  20236.91  20388.59  20398.75
##         1         1         1         1         1         1         1         1
##  20653.82  21116.97  21259.26  22451.65  25126.67   25166.2  25167.02  25681.21
##         1         1         1         1         1         1         1         1
##  26072.23  26313.05  27032.54  27280.04  27603.75  27756.39  27921.96  28720.24
##         1         1         1         1         1         1         1         1
##  29597.66  30449.93   30663.8  31202.33  31501.68  35220.23  35318.92  35879.39
##         1         1         1         1         1         1         1         1
##  36447.73  36971.09  37162.79  39071.17  40574.75  41494.19  42073.83  43024.65
##         1         1         1         1         1         1         1         1
##  43449.59  44235.18  45327.09  46166.63  46406.61  47417.47  48043.85     48178
##         1         1         1         1         1         1         1         1
##  49180.41  50545.47  51296.32  51775.18  52119.65   55183.3  56143.31  58808.84
##         1         1         1         1         1         1         1         1
##  69655.06  70902.02  71046.79  71640.74   92518.8  93824.25  94417.66  97384.01
##         1         1         1         1         1         1         1         1
##  97532.09  97815.78  98350.22  99715.25 103899.87  108109.7 209392.49 229998.79
##         1         1         1         1         1         1         1         1
## 232178.14 240259.87 242300.03 261567.48    274841    284568    290266    297784
##         1         1         1         1         1         1         1         1
##    309020    322481    338267    352045    369091    383636      <NA>
##         1         1         1         1         1         1        62
```

- Observamos que tenemos **62 valores perdidos**.Guardamos en la variable **idx** los índices de los registros con valores **NA** de la variable **Value**.

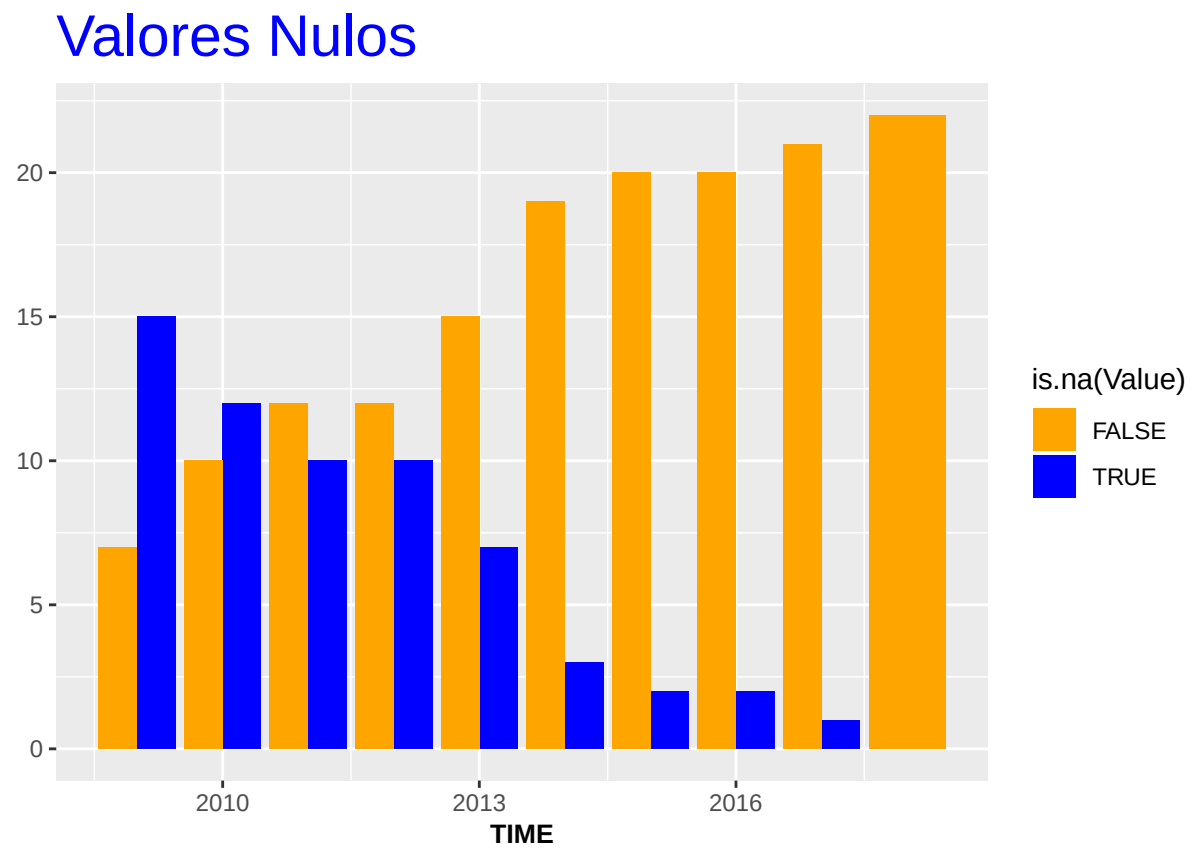
```
idx<-which(is.na(ingreso$Value))
length(idx)
```

```
## [1] 62
```

- Grafiquemos la información que contiene la variable **Value**

```
library(ggplot2)
library(scales)
g = ggplot(ingreso, aes(TIME, fill=is.na(Value)) ) +
labs(title = "Valores Nulos")+ylab("") +
theme(plot.title = element_text(size = rel(2), colour = "blue"))
```

```
g+geom_bar(position="dodge") + scale_fill_manual(values = alpha(c("orange", "blue"), 1)) +
theme(axis.title.x = element_text(face="bold", size=10))
```



- En caso de detectar algún valor anómalo (en nuestro caso los NAS) en las variables tendríamos que realizar una imputación de esos valores o bien sustituyéndolos por la media o usando el algoritmo KNN (k-Nearest Neighbour) con los 3 vecinos más cercanos usando la distancia que consideremos, en este caso usaremos Gower(Mediana), por ser una medida más robusa frente a extremos.

```
library(VIM)
```

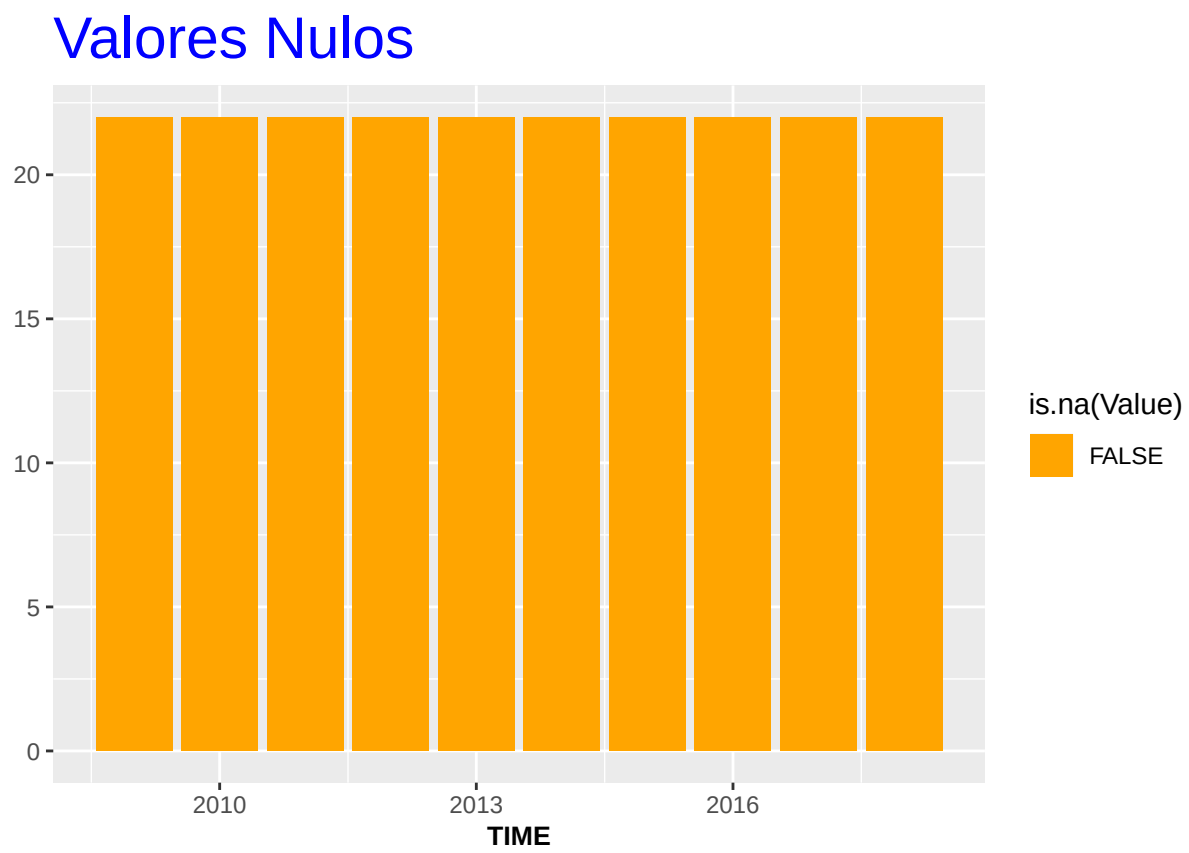
```
## Loading required package: colorspace
## Loading required package: grid
## VIM is ready to use.
## Suggestions and bug-reports can be submitted at: https://github.com/statistikat/VIM/issues
##
## Attaching package: 'VIM'
## The following object is masked from 'package:datasets':
##
##     sleep
```

```
output<-kNN(ingreso, variable=c("Value"),k=3)
ingreso<-output
```

- Comprobamos que no tenemos valores nulos después de la imputación

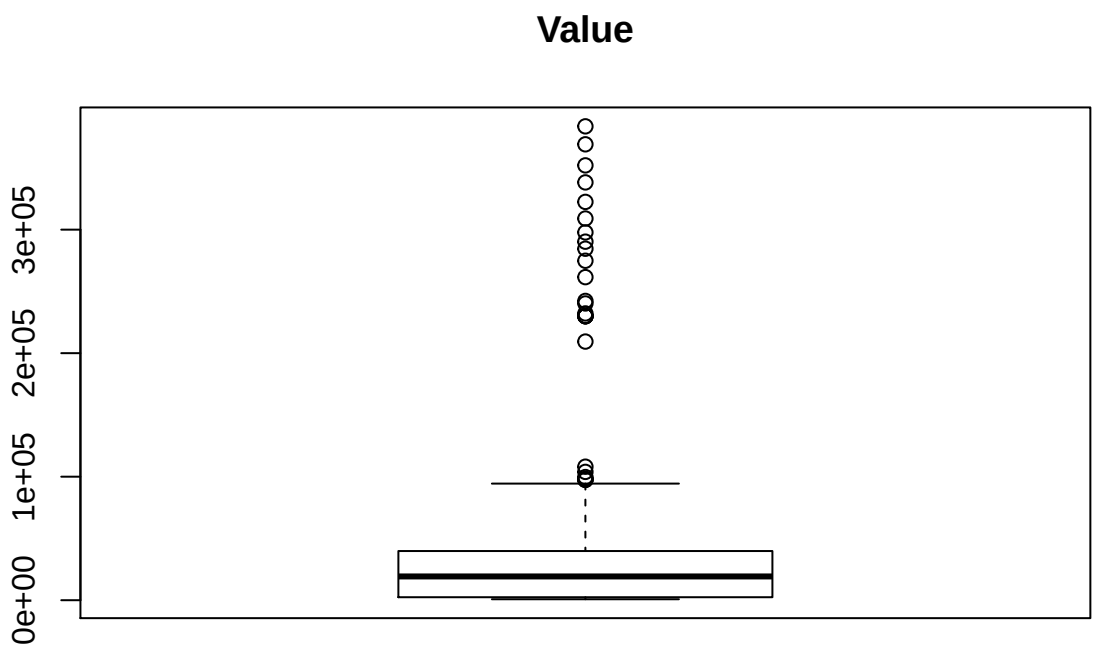
```r
g = ggplot(ingreso, aes(TIME, fill=is.na(Value)) ) +
labs(title = "Valores Nulos")+ylab("") +
theme(plot.title = element_text(size = rel(2), colour = "blue"))

g+geom_bar(position="dodge") + scale_fill_manual(values = alpha(c("orange", "blue"), 1)) +
theme(axis.title.x = element_text(face="bold", size=10))
```



- Con el siguiente gráfico, observaremos que la variable **Value** tiene outliers o valores extremos

```r
boxplot(ingreso$Value, main="Value")
```



- Por otro lado, revisamos para el resto de columnas si tenemos valores NA.(desconocidos o perdidos)

```
table(ingreso$TIME, useNA = "ifany")
```

```
##
## 2009 2010 2011 2012 2013 2014 2015 2016 2017 2018
##   22   22   22   22   22   22   22   22   22   22
```

```
table(ingreso$GEO, useNA = "ifany")
```

```
##
##                                          Belgium
##                                               10
##                                          Croatia
##                                               10
##                                           Cyprus
##                                               10
##                                          Czechia
##                                               10
##                                          Denmark
##                                               10
##                                          Estonia
##                                               10
##                                          Finland
##                                               10
## Germany (until 1990 former territory of the FRG)
##                                               10
##                                          Hungary
```

```
##                                         10
##                                    Iceland
##                                         10
##                                    Ireland
##                                         10
##                                     Latvia
##                                         10
##                                  Lithuania
##                                         10
##                                 Luxembourg
##                                         10
##                                      Malta
##                                         10
##                                     Norway
##                                         10
##                                     Poland
##                                         10
##                                   Slovenia
##                                         10
##                                      Spain
##                                         10
##                                     Sweden
##                                         10
##                                Switzerland
##                                         10
##                             United Kingdom
##                                         10
```

```
table(ingreso$UNIT, useNA = "ifany")
```

```
##
## Million euro
##          220
```

```
table(ingreso$ICHA11_FS, useNA = "ifany")
```

```
##
## All revenues of financing schemes
##                               220
```

Observamos que no existen ahora valores perdidos después de la imputación.La suma de las cantidades de cada variable, suman el total.

- Finalmente, creamos un fichero con toda la información corregida.

```
write.csv(ingreso, file="IngresosSanitarios_Financiacion_clean.csv", row.names = FALSE)
```