¿Cómo realizar la limpieza y análisis de datos?

Autores: Eduardo Mora González y Diego Sánchez De La Fuente

Enero 2023

Contents

CARGA DEL FICHERO DE DATOS **1**

Preprocesado y gestión de características **3**

Valores nulos del conjunto de los datos 3

Normalización del conjunto de los datos 3

Construcción de conjunto de datos final 16

Correlaciones 17

Análisis de componentes principales (PCA) 19

Análisis de los datos **28**

body { text-align: justify}

Instalamos y cargamos las librerías necesarias.

```
if (!require('readr')) install.packages('readr'); library('readr')
if (!require('ggplot2')) install.packages('ggplot2'); library('ggplot2')
if (!require('DataExplorer')) install.packages('DataExplorer'); library('DataExplorer')
if (!require('corrplot')) install.packages("corrplot"); library(corrplot)
if (!require('factoextra')) install.packages("factoextra"); library(factoextra)
```

CARGA DEL FICHERO DE DATOS

```
datos <- read_csv("./fichero_original_datos.csv")
```

Ahora vamos a ver las estructura del juego de datos

```
str(datos)
```

```
## spec_tbl_df [918 x 12] (S3: spec_tbl_df/tbl_df/tbl/data.frame)
##  $ Age           : num [1:918] 40 49 37 48 54 39 45 54 37 48 ...
##  $ Sex           : chr [1:918] "M" "F" "M" "F" ...
##  $ ChestPainType : chr [1:918] "ATA" "NAP" "ATA" "ASY" ...
```

```
##  $ RestingBP     : num [1:918] 140 160 130 138 150 120 130 110 140 120 ...
##  $ Cholesterol   : num [1:918] 289 180 283 214 195 339 237 208 207 284 ...
##  $ FastingBS     : num [1:918] 0 0 0 0 0 0 0 0 0 0 ...
##  $ RestingECG    : chr [1:918] "Normal" "Normal" "ST" "Normal" ...
##  $ MaxHR         : num [1:918] 172 156 98 108 122 170 170 142 130 120 ...
##  $ ExerciseAngina: chr [1:918] "N" "N" "N" "Y" ...
##  $ Oldpeak       : num [1:918] 0 1 0 1.5 0 0 0 0 1.5 0 ...
##  $ ST_Slope      : chr [1:918] "Up" "Flat" "Up" "Flat" ...
##  $ HeartDisease  : num [1:918] 0 1 0 1 0 0 0 0 1 0 ...
##  - attr(*, "spec")=
##   .. cols(
##   ..   Age = col_double(),
##   ..   Sex = col_character(),
##   ..   ChestPainType = col_character(),
##   ..   RestingBP = col_double(),
##   ..   Cholesterol = col_double(),
##   ..   FastingBS = col_double(),
##   ..   RestingECG = col_character(),
##   ..   MaxHR = col_double(),
##   ..   ExerciseAngina = col_character(),
##   ..   Oldpeak = col_double(),
##   ..   ST_Slope = col_character(),
##   ..   HeartDisease = col_double()
##   .. )
##  - attr(*, "problems")=<externalptr>
```

Vamos ahora a sacar estadísticas básicas

```
summary(datos)
```

```
##       Age            Sex            ChestPainType        RestingBP
##  Min.   :28.00   Length:918         Length:918         Min.   :  0.0
##  1st Qu.:47.00   Class :character   Class :character   1st Qu.:120.0
##  Median :54.00   Mode  :character   Mode  :character   Median :130.0
##  Mean   :53.51                                         Mean   :132.4
##  3rd Qu.:60.00                                         3rd Qu.:140.0
##  Max.   :77.00                                         Max.   :200.0
##   Cholesterol      FastingBS       RestingECG           MaxHR
##  Min.   :  0.0   Min.   :0.0000   Length:918         Min.   : 60.0
##  1st Qu.:173.2   1st Qu.:0.0000   Class :character   1st Qu.:120.0
##  Median :223.0   Median :0.0000   Mode  :character   Median :138.0
##  Mean   :198.8   Mean   :0.2331                      Mean   :136.8
##  3rd Qu.:267.0   3rd Qu.:0.0000                      3rd Qu.:156.0
##  Max.   :603.0   Max.   :1.0000                      Max.   :202.0
##  ExerciseAngina        Oldpeak          ST_Slope          HeartDisease
##  Length:918         Min.   :-2.6000   Length:918         Min.   :0.0000
##  Class :character   1st Qu.: 0.0000   Class :character   1st Qu.:0.0000
##  Mode  :character   Median : 0.6000   Mode  :character   Median :1.0000
##                     Mean   : 0.8874                      Mean   :0.5534
##                     3rd Qu.: 1.5000                      3rd Qu.:1.0000
##                     Max.   : 6.2000                      Max.   :1.0000
```

Preprocesado y gestión de características

Valores nulos del conjunto de los datos

De tipo numérico

```
colSums(is.na(datos))
```

```
##             Age             Sex  ChestPainType      RestingBP    Cholesterol
##               0               0               0               0               0
##      FastingBS     RestingECG           MaxHR ExerciseAngina        Oldpeak
##               0               0               0               0               0
##       ST_Slope    HeartDisease
##               0               0
```

De tipo cadena

```
colSums(datos=="")
```

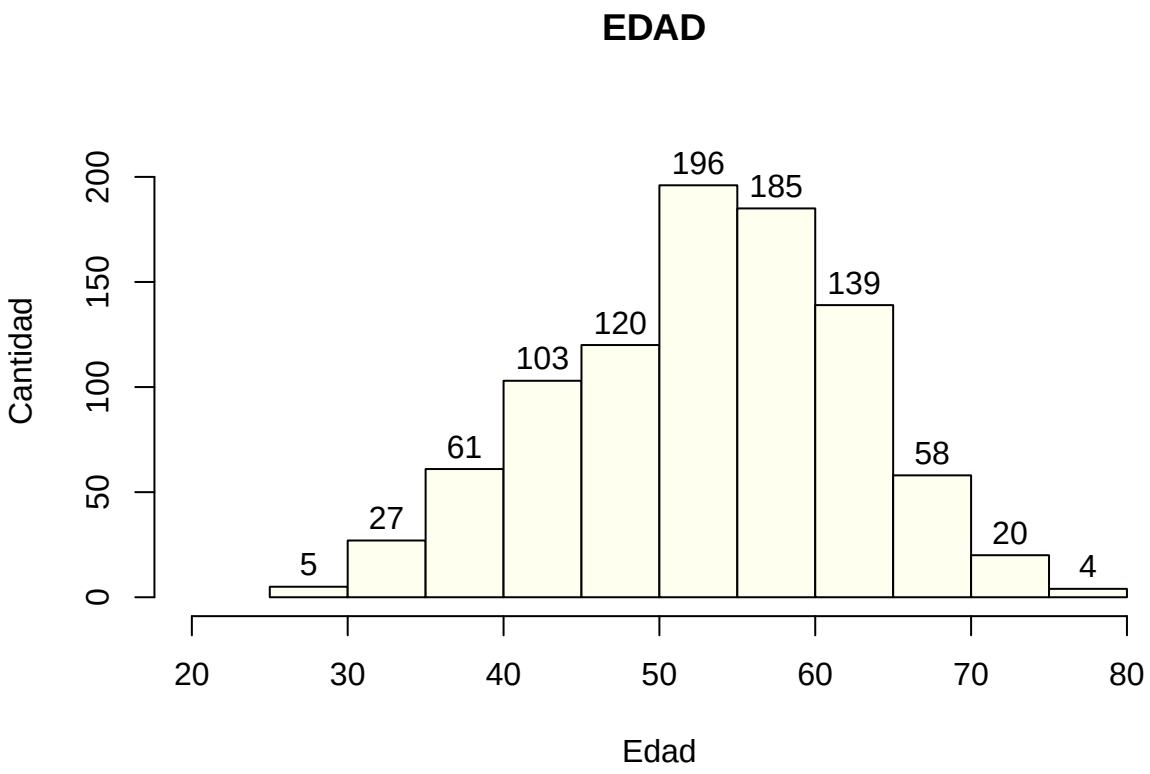
```
##             Age             Sex  ChestPainType      RestingBP    Cholesterol
##               0               0               0               0               0
##      FastingBS     RestingECG           MaxHR ExerciseAngina        Oldpeak
##               0               0               0               0               0
##       ST_Slope    HeartDisease
##               0               0
```

Como se puede comprobar, tenemos la "suerte" de no tener ningún valor nulo o vacío en los dos juegos de datos.

Normalización del conjunto de los datos

EDAD

```
#Histograma de la característica edad del primer conjunto de datos
h1 <- hist(datos$Age, xlab="Edad", col="ivory",
           ylab="Cantidad", main="EDAD ", ylim = c(0, 225), xlim = c(20,80))
text(h1$mids,h1$counts,labels=h1$counts, adj=c(0.5, -0.5))
```



Como se puede observar, la franja de entre los 50 y 60 años son donde más datos existen, mientras que los extremos donde menos datos.

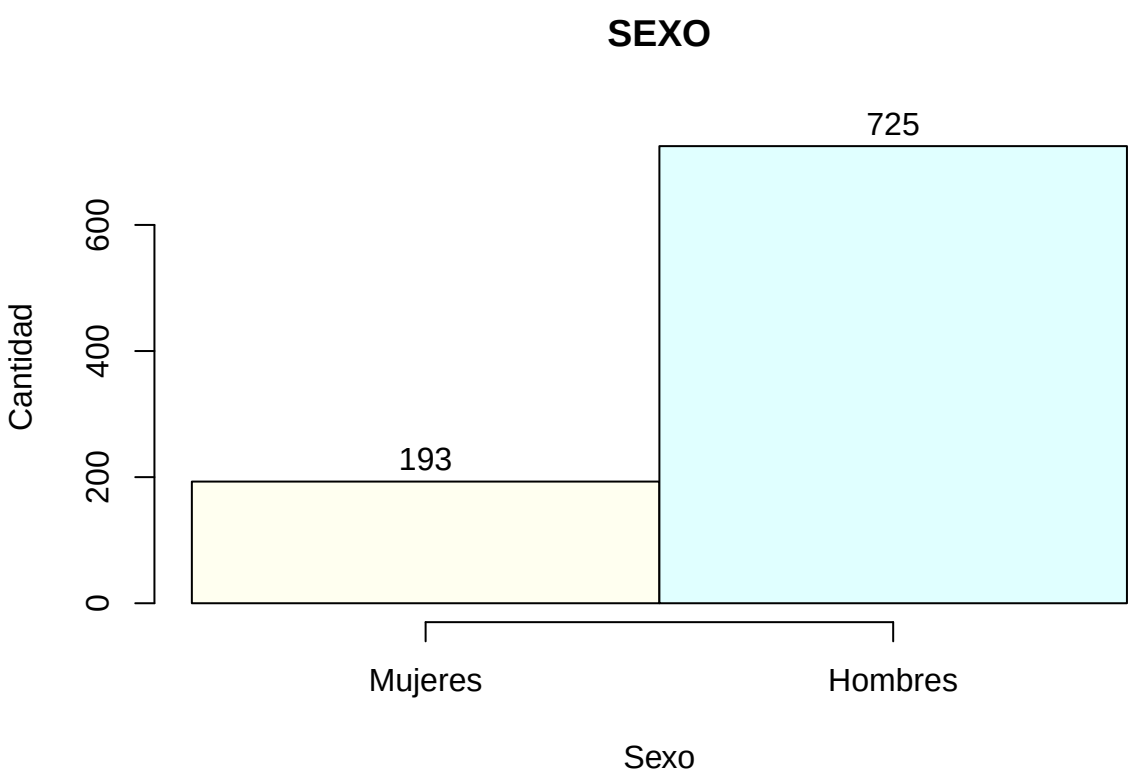
SEXO

Normalizamos para tenerlo de tipo numérico todas la variables

```
#Cambiamos las letras por los números
datos$Sex [datos$Sex == "M"] <- 1
datos$Sex [datos$Sex == "F"] <- 0

#Pasamos de carácter a numérico
datos$Sex <- as.numeric(datos$Sex)
```

Una vez normalizada la característica , analizamos el conjunto de los datos contemplados en esta.

```
h1 <- hist(datos$Sex, xlab="Sexo", col=c("ivory", "lightcyan"),
           ylab="Cantidad", main="SEXO", breaks = 2, ylim = c(0, 750), axes = FALSE)
text(h1$mids,h1$counts,labels=h1$counts, adj=c(0.5, -0.5))
axis(1, at =c(0.25, 0.75), cex.axis=1, labels = c("Mujeres","Hombres" ))
axis(2)
```



TIPO DE DOLOR TORÁCICO (ChestPainType)

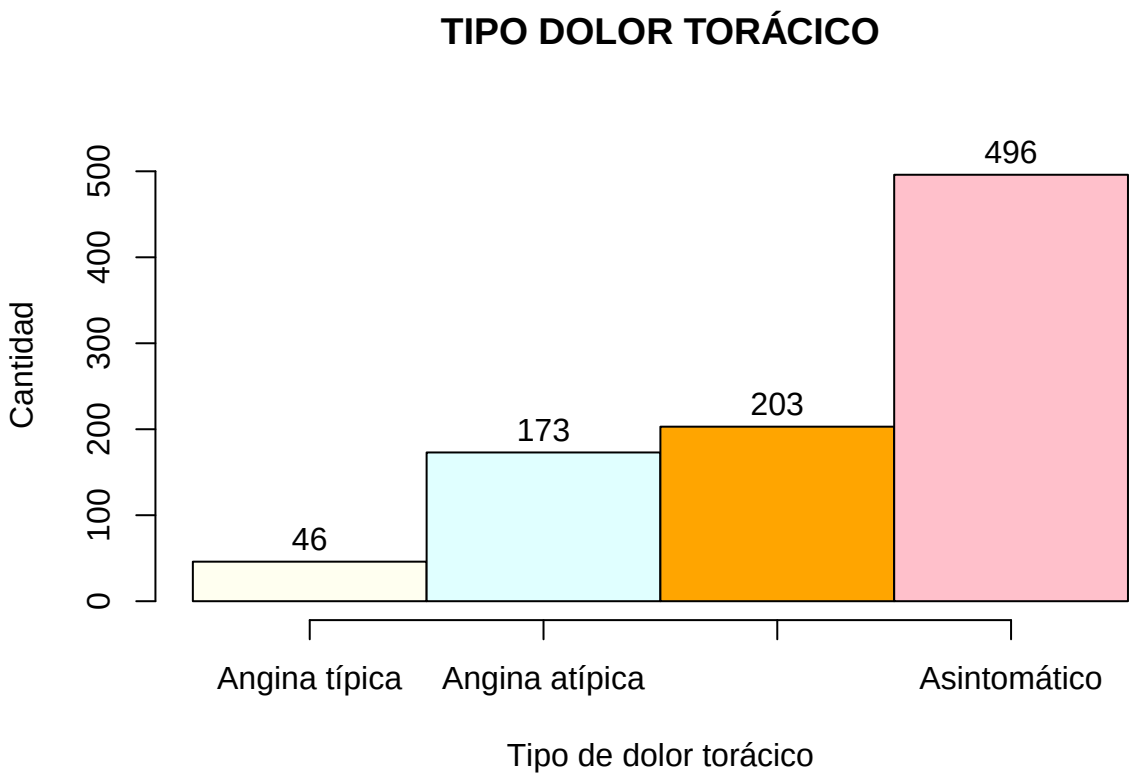
Nos damos cuenta de que el conjunto de datos viene identificado por 4 variables categóricas (TA: angina típica, ATA: angina atípica, NAP: dolor no anginal, ASY: asintomático). Normalizamos para tenerlo de tipo numérico todas la variables:

```
#Cambiamos las letras por los números
datos$ChestPainType [datos$ChestPainType == "TA"]  <- 0
datos$ChestPainType [datos$ChestPainType == "ATA"] <- 1
datos$ChestPainType [datos$ChestPainType == "NAP"] <- 2
datos$ChestPainType [datos$ChestPainType == "ASY"] <- 3

#Pasamos de carácter a numérico
datos$ChestPainType <- as.numeric(datos$ChestPainType)
```

Una vez normalizada la característica , analizamos el conjunto de los datos contemplados en esta.

```
h1 <- hist(datos$ChestPainType, xlab="Tipo de dolor torácico",
           col= c("ivory", "lightcyan", "ORANGE", "PINK"),
           ylab="Cantidad", main="TIPO DOLOR TORÁCICO",
           ylim = c(0, 550),axes = FALSE,
           breaks=seq(min(datos$ChestPainType)-0.5,
                      max(datos$ChestPainType)+0.5, by=1) )
text(h1$mids,h1$counts,labels=h1$counts, adj=c(0.5, -0.5))
axis(1, at =c(0,1,2,3), cex.axis=1,
```

```
    labels = c("Angina típica", "Angina atípica","Dolor no anginal", "Asintomático" ))
axis(2)
```



como se puede comprobar, tenemos mas casos de de asintomaticos que del resto.

PRESIÓN ARTERIAL EN REPOSO (RestingBP)

Como se muestran en las estadísticas esta característica es de tipo numérico y en el conjunto de datos va desde 0 hasta 200. Como se puede apreciar, tener una presión arterial de 0 es estar considerado muerto, por lo que considero que el valor 0 es un valor nulo.

Lo primero que se va a hacer es obtener el número de casos que la presión arterial es 0, y se consideraran las diversas formas de tratar estos datos:

```
#Veces que aparece el valor cero en la presion arterial
length(datos$RestingBP[datos$RestingBP == 0])
```

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

Como solo aparece una vez, se le asignará un valor por defecto. El valor por defecto será el más común.

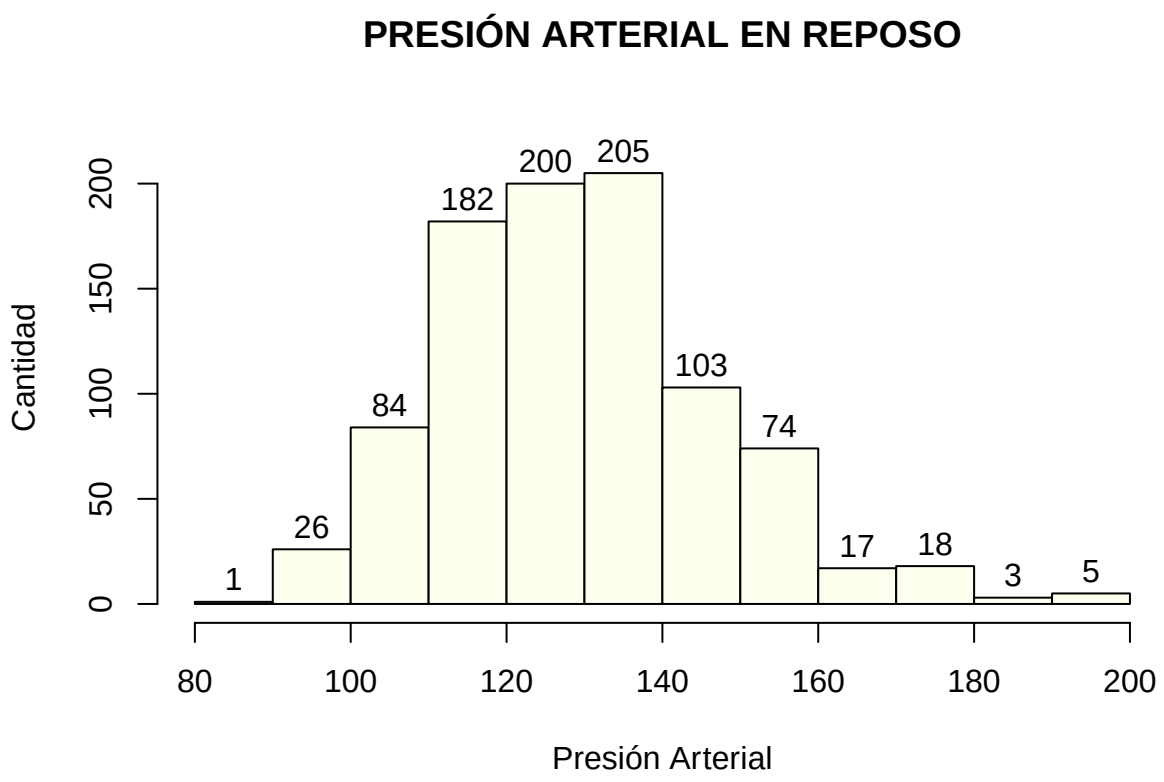
```
#Función para calcular el valor más común
common_value <- function(x) {
uniqx <- unique(na.omit(x))
```

```r
uniqx[which.max(tabulate(match(x, uniqx)))]
}

#Calculamos el valor más comun
BP_comun <- common_value(datos$RestingBP)

#Asignamos el valor
datos$RestingBP[datos$RestingBP == 0] <- BP_comun

#vemos las estaditicas del dato
summary(datos$RestingBP)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##    80.0   120.0   130.0   132.5   140.0   200.0
```

Ahora ya tenemos los valores entre 80 y 200 que son un rango normal para estos valores.

```r
#Histograma de la característica Presión Arterial del primer conjunto de datos
h1 <- hist(datos$RestingBP, xlab="Presión Arterial", col="ivory",
           ylab="Cantidad", main="PRESIÓN ARTERIAL EN REPOSO",
           ylim = c(0, 225), xlim = c(80,200))
text(h1$mids,h1$counts,labels=h1$counts, adj=c(0.5, -0.5))
```



COLESTEROL (Cholesterol)

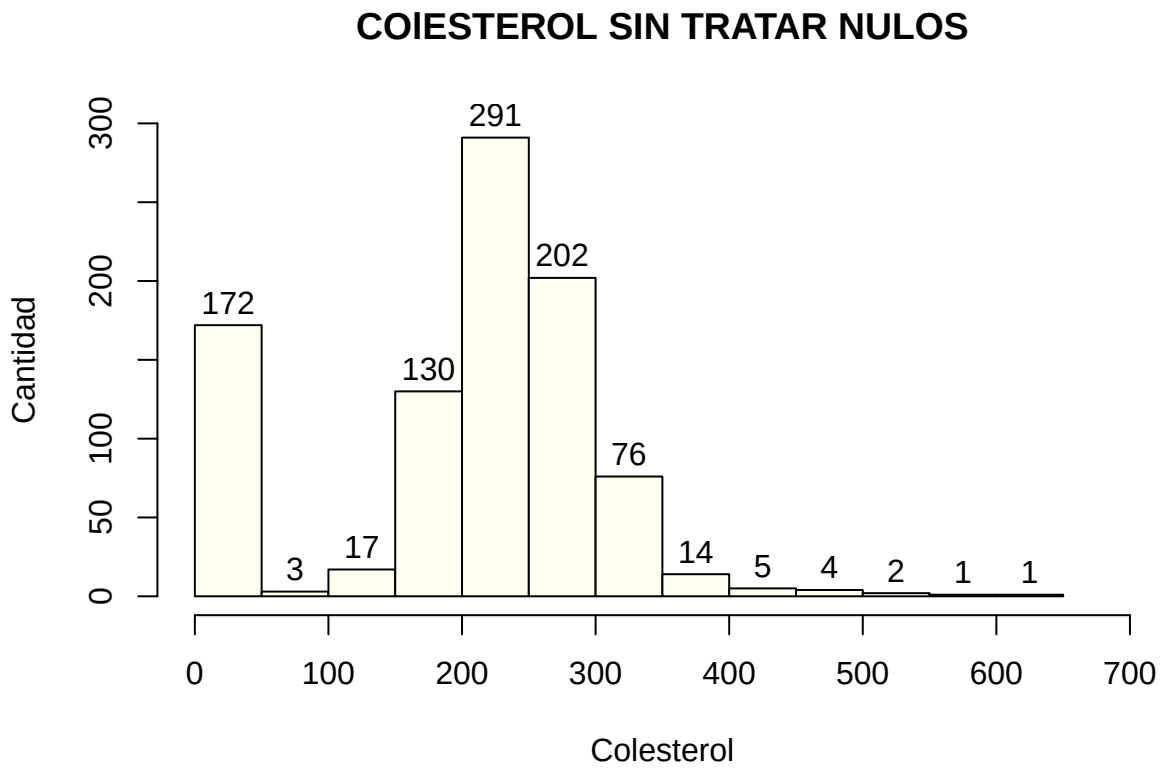
La siguiente característica es de tipo numérico. Al igual que en la presión arterial en reposo, que tenemos valores 0 que debemos analizar. Lo primero que se va a hacer es obtener el numero de casos que el colesterol es 0, y se consideraran las diversas formas de tratar estos datos.

```
#Veces que aparece el valor cero en la presion arterial
length(datos$RestingBP[datos$Cholesterol == 0])
```

```
## [1] 172
```

Esta vez tenemos 172 casos en lo que ocurre esto (equivale a un 18% de los casos totales). Antes de ver que valor se le asignan, se va a graficar los datos para ver de manera grafica que opción tomar: el valor medio o el más común.

```
h1 <- hist(datos$Cholesterol, xlab="Colesterol", col="ivory",
           ylab="Cantidad", main="COlESTEROL SIN TRATAR NULOS", ylim = c(0,300),
           xlim = c(0, 700))
text(h1$mids,h1$counts,labels=h1$counts, adj=c(0.5, -0.5))
```



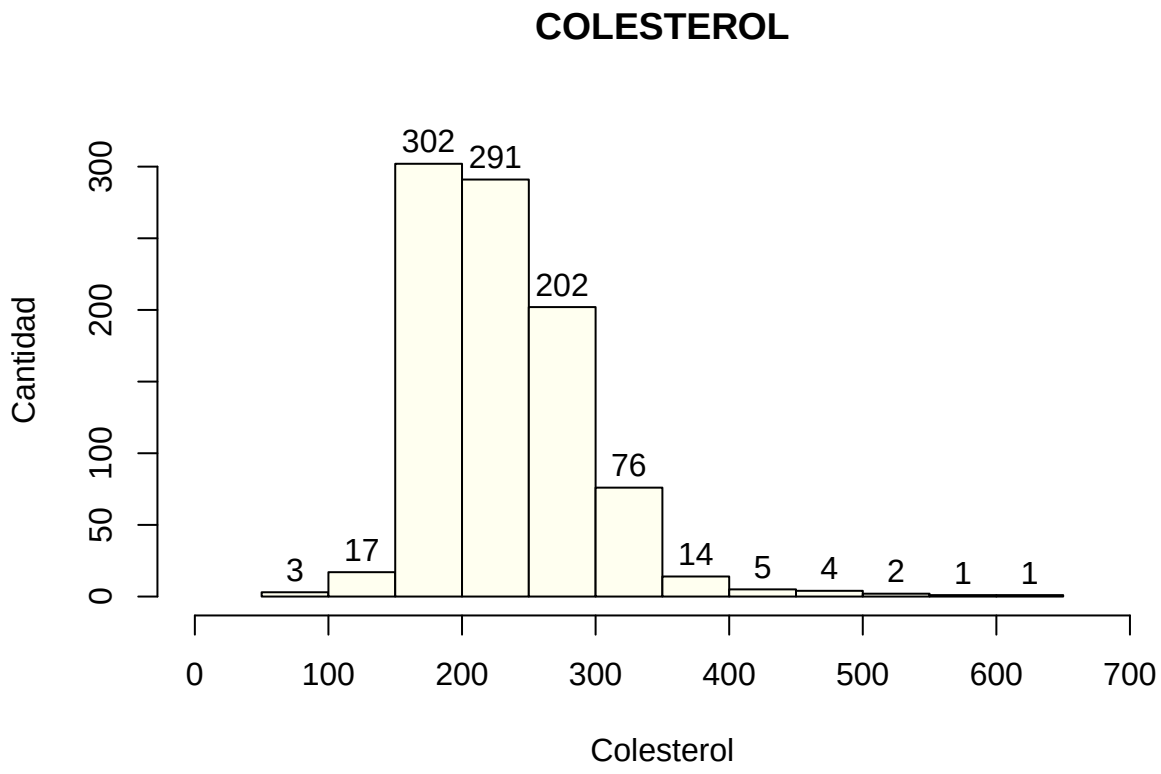
Tras analizar la gráfica y para no perder estos datos, se le asignaran un valor por defecto, que será la media de los datos. Esta decisión se ha tomado ya que poner el más común, nos crearía un conjunto de datos muy distintos entre unas medidas y otras, mientras que poner la media sería un valor que tenga en cuenta el grueso de todos los datos.

```r
#Calculamos el valor más comun
colesterol_media <- mean(datos$Cholesterol)

#Asignamos el valor truncado para evitar decimales
datos$Cholesterol[datos$Cholesterol == 0] <- trunc(colesterol_media)

#vemos las estaditicas del dato
summary(datos$RestingBP)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##    80.0   120.0   130.0   132.5   140.0   200.0
```

Ahora ya tenemos los valores entre 80 y 200 que son un rango normal para estos valores.

```r
h1 <- hist(datos$Cholesterol, xlab="Colesterol", col="ivory",
           ylab="Cantidad", main="COLESTEROL", ylim = c(0,330), xlim = c(0, 700))
text(h1$mids,h1$counts,labels=h1$counts, adj=c(0.5, -0.5))
```

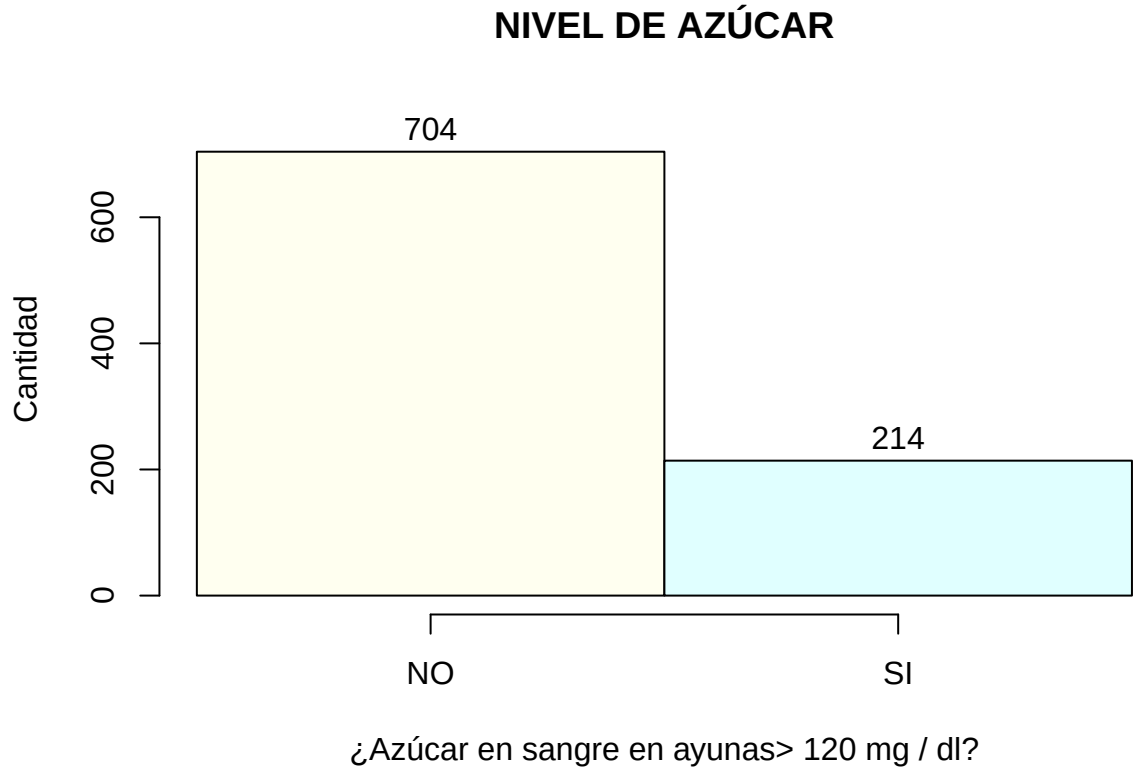


NIVEL DE AZÚCAR EN SANGRE EN AYUNAS (FastingBS)

Como se puede comprobar el conjunto de los datos puedes ser 1 o 0, es decir verdadero o falso si se cumple la siguiente condición: si nivel de azúcar en sangre en ayunas> 120 mg / dl.

En esta característica no tenemos valores nulos, así que vamos a ver la distribución de las dos opciones:

```r
h1 <- hist(datos$FastingBS, xlab="¿Azúcar en sangre en ayunas> 120 mg / dl?",
           col=c("ivory", "lightcyan"), ylab="Cantidad",
           main="NIVEL DE AZÚCAR", breaks = 2, ylim = c(0, 750), axes = FALSE)
text(h1$mids,h1$counts,labels=h1$counts, adj=c(0.5, -0.5))
axis(1, at =c(0.25, 0.75), cex.axis=1, labels = c("NO","SI" ))
axis(2)
```



Como se puede comprobar que hay mas casos que NO se cumple esa condición de que SÍ.

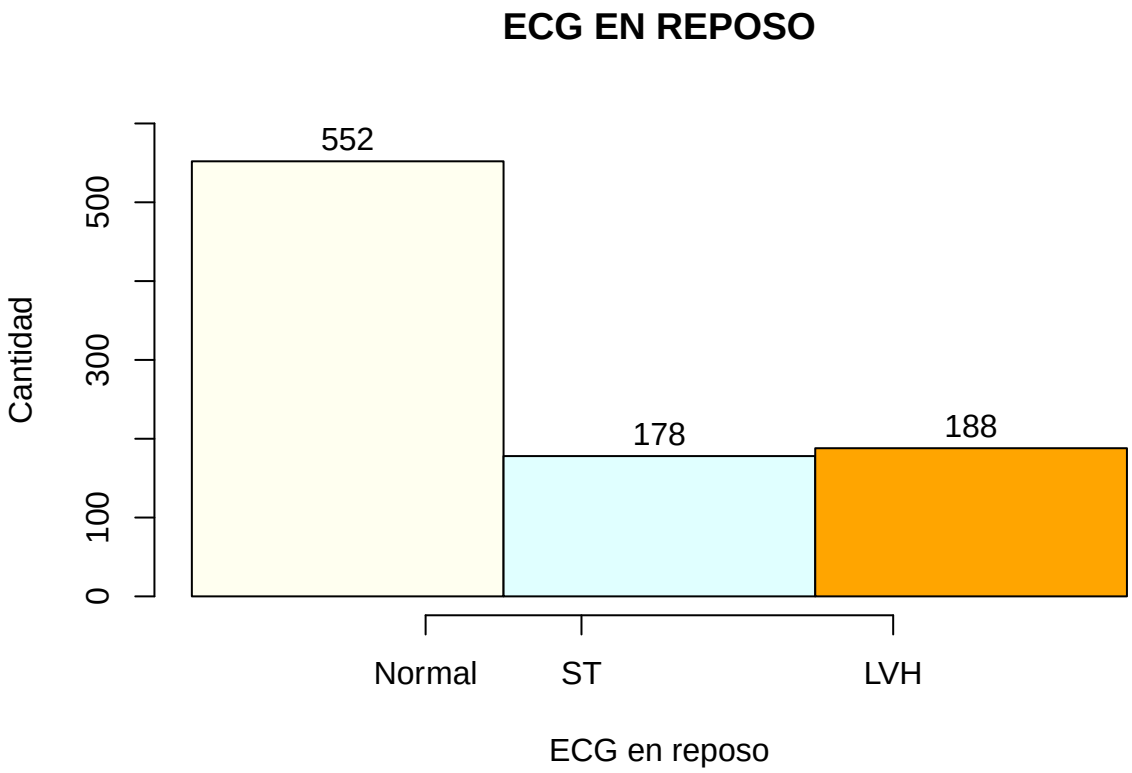
ECG EN REPOSO (RestingECG)

Nos damos cuenta de que el conjunto de datos viene identificado por 3 variables categóricas: + Normal: Normal, + ST: con anomalía de la onda ST-T + LVH: que muestra una hipertrofia ventricular izquierda probable o definitiva según los criterios de Estes. Normalizamos para tenerlo de tipo numérico todas la variables:

```r
#Cambiamos las letras por los números
datos$RestingECG [datos$RestingECG == "Normal"]  <- 0
datos$RestingECG [datos$RestingECG == "ST"] <- 1
datos$RestingECG [datos$RestingECG == "LVH"] <- 2

#Pasamos de carácter a numérico
datos$RestingECG <- as.numeric(datos$RestingECG)
```

Una vez normalizada la característica , analizamos el conjunto de los datos contemplados en esta.

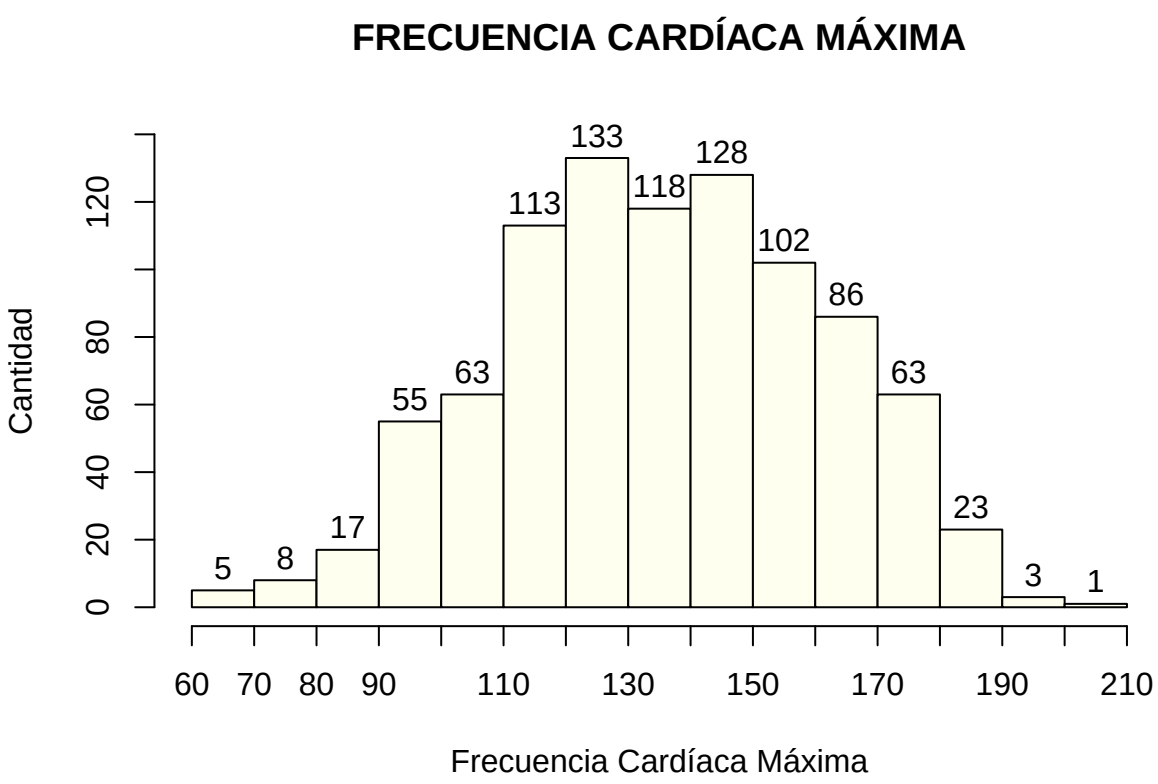
```r
h1 <- hist(datos$RestingECG, xlab="ECG en reposo",
           col= c("ivory", "lightcyan", "ORANGE"),
           ylab="Cantidad", main="ECG EN REPOSO",
           ylim = c(0, 600), axes = FALSE,
           breaks=seq(min(datos$RestingECG)-0.5,
                      max(datos$RestingECG)+0.5, by=1) )
text(h1$mids,h1$counts,labels=h1$counts, adj=c(0.5, -0.5))
axis(1, at =c(0.25, 0.75, 1.75 ), cex.axis=1, labels = c("Normal","ST", "LVH"))
axis(2)
```



FRECUENCIA CARDÍACA MÁXIMA (MaxHR)

Dicha característica es de carácter numérica y en el conjunto de datos contempla valores desde el 60 al 202

```r
h1 <- hist(datos$MaxHR, xlab="Frecuencia Cardíaca Máxima",
           col="ivory", ylab="Cantidad", main="FRECUENCIA CARDÍACA MÁXIMA",
           ylim = c(0,140), axes = FALSE)
text(h1$mids,h1$counts,labels=h1$counts, adj=c(0.5, -0.5))
axis(1, at =c(60, 70, 80,90,100,110,120,130,140,150,160,170,180,190,200,210), cex.axis=1)
axis(2)
```



Se puede comprobar que los extremos en el conjunto de datos tienen menos valores, y que el grueso de las muestras se encuentran entre los valores centrales (desde 100 a 180).

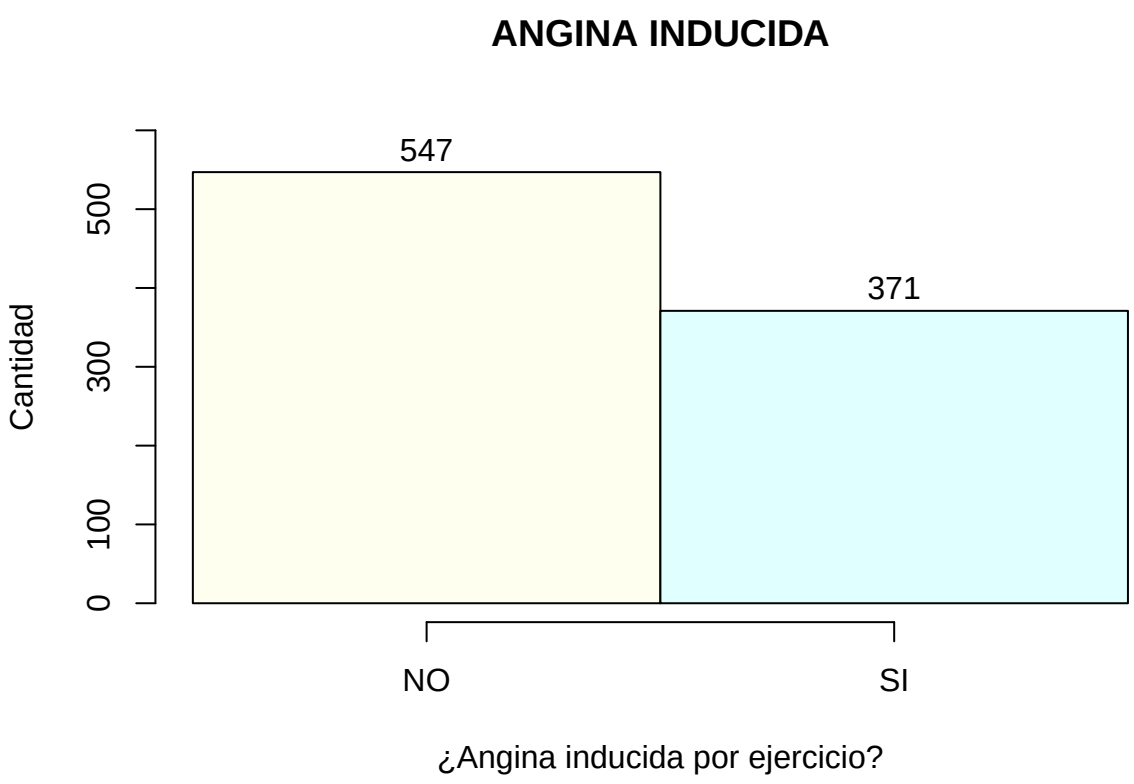
ANGINA INDUCIDA POR EJERCICIO (ExerciseAngina)

En el conjunto de datos tiene los valores Y: Sí, N: No. Al igual que se ha hecho con otras características, se normalizará el conjunto.

```
#Cambiamos las letras por los números
datos$ExerciseAngina [datos$ExerciseAngina == "N"]  <- 0
datos$ExerciseAngina [datos$ExerciseAngina == "Y"]  <- 1

#Pasamos de carácter a numérico
datos$ExerciseAngina <- as.numeric(datos$ExerciseAngina)
```

Una vez normalizada la característica , analizamos el conjunto de los datos contemplados en esta.

```
h1 <- hist(datos$ExerciseAngina, xlab="¿Angina inducida por ejercicio?",
           col=c("ivory", "lightcyan"), ylab="Cantidad", main="ANGINA INDUCIDA",
           breaks = 2, ylim = c(0, 600), axes = FALSE)
text(h1$mids,h1$counts,labels=h1$counts, adj=c(0.5, -0.5))
axis(1, at =c(0.25, 0.75), cex.axis=1, labels = c("NO","SI" ))
axis(2)
```

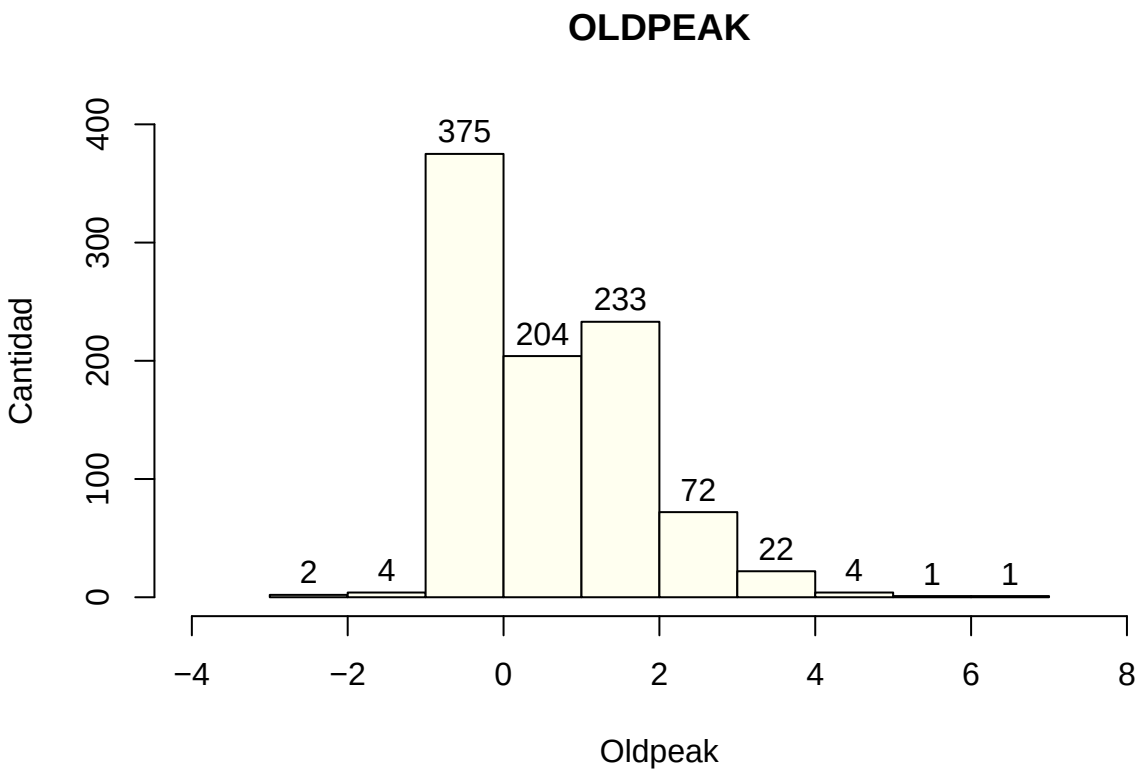


Como se puede apreciar, hay mas casos en que NO se ha producido una angina inducida por el ejercicio de que Si se haya producido.

OLDPEAK

Esta característica de tipo numérica puede abarcar valores negativos hasta hasta un máximo de un valor igual a 6,2.

```
h1 <- hist(datos$Oldpeak, xlab="Oldpeak", col="ivory", ylab="Cantidad", main="OLDPEAK", ylim = c(0,400)
text(h1$mids,h1$counts,labels=h1$counts, adj=c(0.5, -0.5))
```



Se puede comprobar que el grueso de las muestras se encuentra entre los valores centrales teniendo una distribución normal

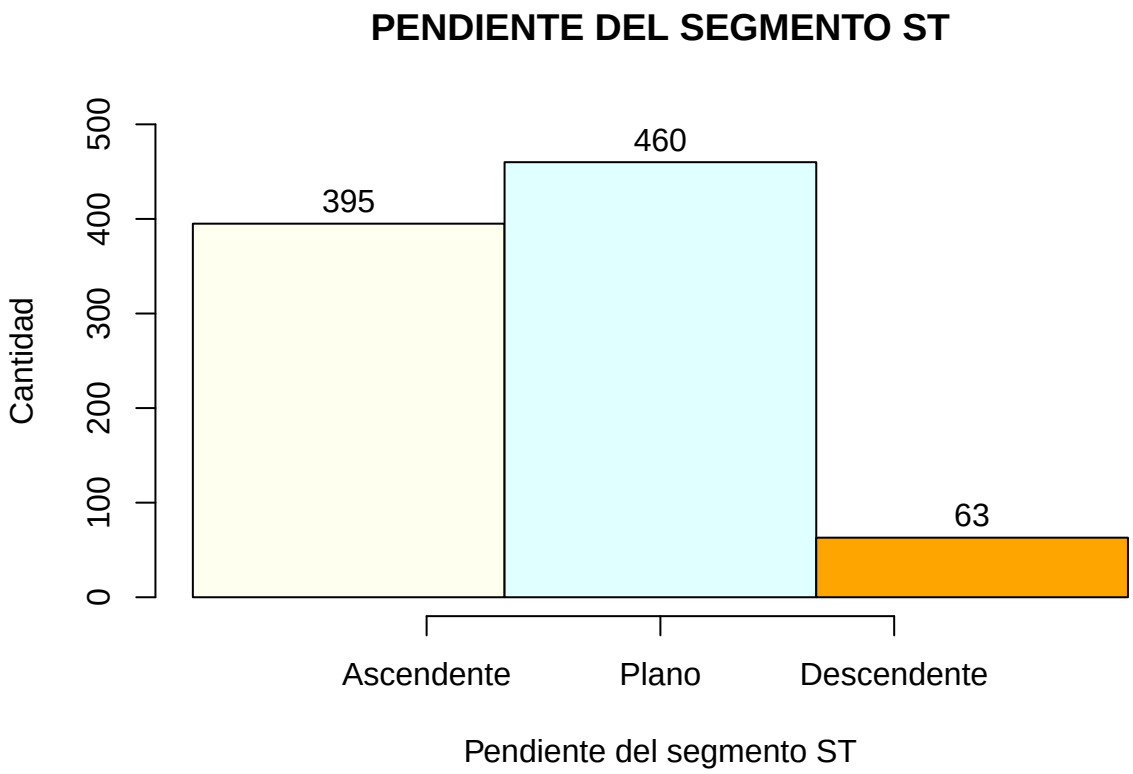
PENDIENTE DEL SEGMENTO ST (ST_Slope)

Como ocurría en otras características anteriores el conjunto tiene los valores para esta caracteristica de la siguiente forma: + Up: uploping + Flat: flat + Down: downsloping Y como se ha realizado antes, se normalizará para solo tener datos numericos.

```
#Cambiamos las letras por los números
datos$ST_Slope [datos$ST_Slope == "Up"]   <- 0
datos$ST_Slope [datos$ST_Slope == "Flat"] <- 1
datos$ST_Slope [datos$ST_Slope == "Down"] <- 2

#Pasamos de carácter a numérico
datos$ST_Slope <- as.numeric(datos$ST_Slope)
```

Una vez normalizada la característica , analizamos el conjunto de los datos contemplados en esta.

```
h1 <- hist(datos$ST_Slope, xlab="Pendiente del segmento ST",
           col= c("ivory", "lightcyan", "ORANGE"), ylab="Cantidad",
           main="PENDIENTE DEL SEGMENTO ST", ylim = c(0, 500),
           axes = FALSE,breaks=seq(min(datos$ST_Slope)-0.5, max(datos$ST_Slope)+0.5, by=1) )
text(h1$mids,h1$counts,labels=h1$counts, adj=c(0.5, -0.5))
axis(1, at =c(0.25,1,1.75), cex.axis=1, labels = c("Ascendente","Plano", "Descendente"))
axis(2)
```

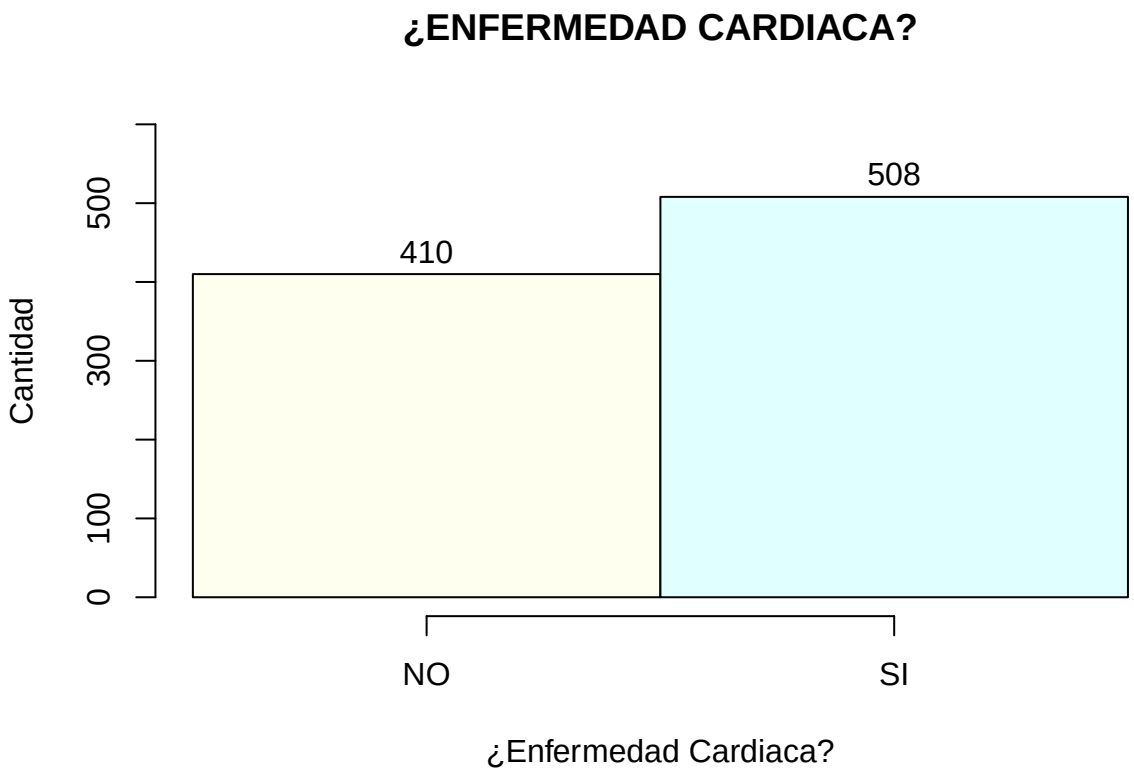


El caso más común es que la pendiente sea plana, teniendo menos casos en los casos descendentes.

¿ENFERMEDAD CARDIACA? (HeartDisease)

En el conjunto de datos tienen normalizada la salida usando el valor 1: enfermedad cardíaca, y el valor 0: Normal.

```
h1 <- hist(datos$HeartDisease, xlab="¿Enfermedad Cardiaca?",
           col=c("ivory", "lightcyan"),
           ylab="Cantidad", main="¿ENFERMEDAD CARDIACA?",
           breaks = 2, ylim = c(0, 600), axes = FALSE)
text(h1$mids,h1$counts,labels=h1$counts, adj=c(0.5, -0.5))
axis(1, at =c(0.25, 0.75), cex.axis=1, labels = c("NO","SI" ))
axis(2)
```



Como se puede observar hay mas casos en que SI hay enfermedad cardiaca que caso en los que NO hay.

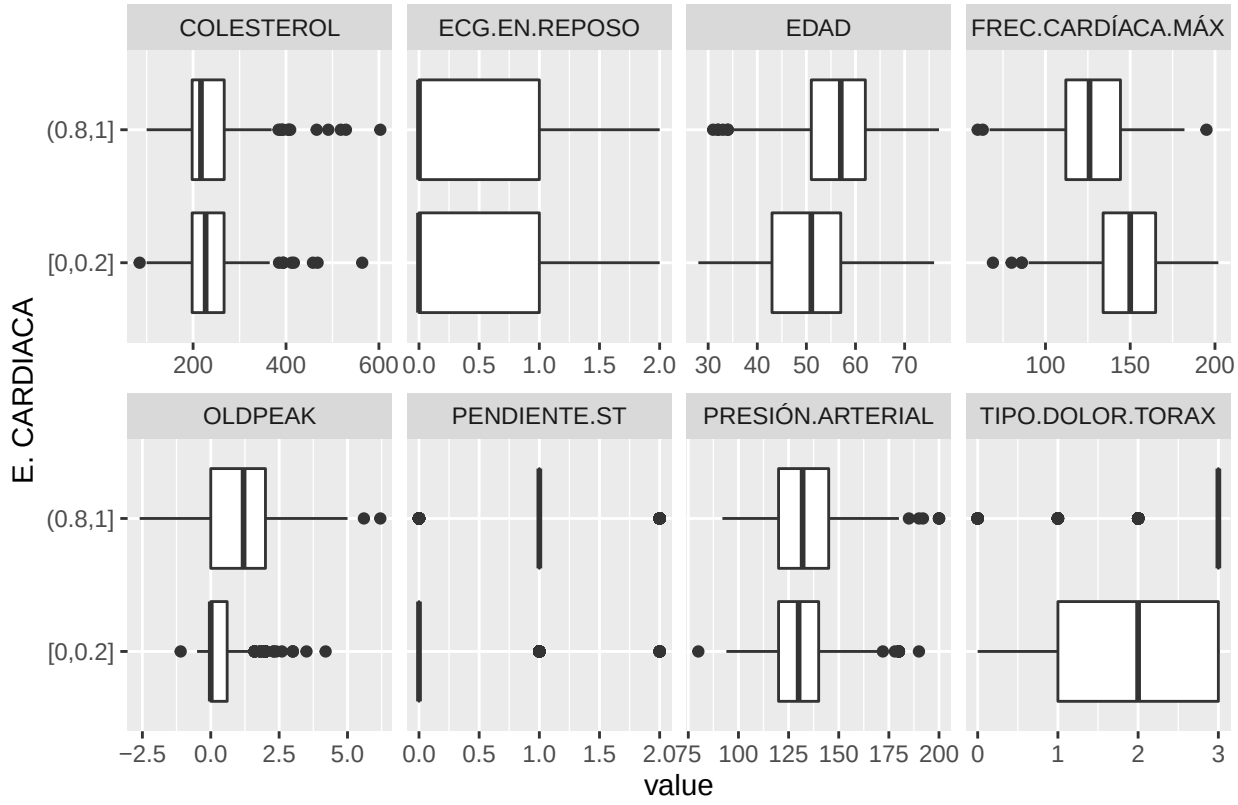
Construcción de conjunto de datos final

Renombramos las columnas para que tenga uno mas significativo y creamos el conjunto final de datos.

```
datos_final <- datos

colnames(datos_final)[1]<-  "EDAD"
colnames(datos_final)[2]<-  "SEXO"
colnames(datos_final)[3]<-  "TIPO DOLOR TORAX"
colnames(datos_final)[4]<-  "PRESIÓN ARTERIAL"
colnames(datos_final)[5]<-  "COLESTEROL"
colnames(datos_final)[6]<-  "NIVEL DE AZÚCAR"
colnames(datos_final)[7]<-  "ECG EN REPOSO"
colnames(datos_final)[8]<-  "FREC CARDÍACA MÁX"
colnames(datos_final)[9]<-  "ANGINA x EJERCICIO"
colnames(datos_final)[10]<- "OLDPEAK"
colnames(datos_final)[11]<- "PENDIENTE ST"
colnames(datos_final)[12]<- "E. CARDIACA"
```

Por ultimo se va a mirar a través de los diagramas de cajas el rango de las características enfrentado a si un paciente tiene una enfermedad cardiaca o no.

```r
#Diagrama de caja de todas las características enfrentadas a si un paciente tiene enfermedad cardiaca
plot_boxplot(datos_final, by = "E. CARDIACA")
```



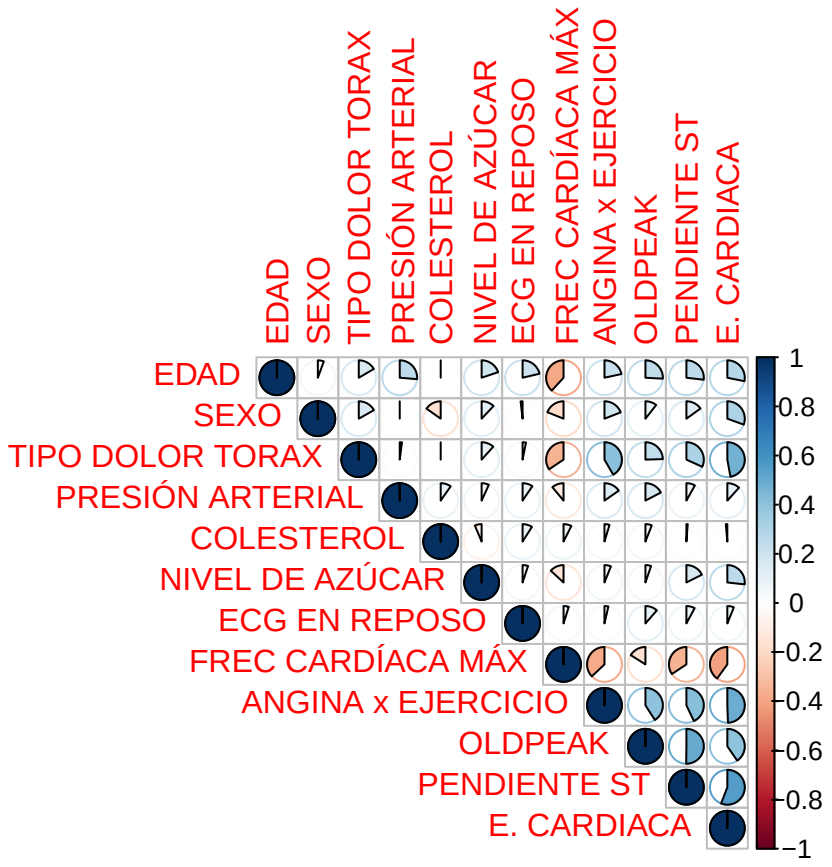
Correlaciones

```r
#Calculamos las correlaciones
cor_datos <- cor(datos_final)
cor_datos
```

```
##                          EDAD         SEXO TIPO DOLOR TORAX PRESIÓN ARTERIAL
## EDAD                1.000000000  0.055750099      0.165895861      0.262891276
## SEXO                0.055750099  1.000000000      0.168254135      0.009030955
## TIPO DOLOR TORAX    0.165895861  0.168254135      1.000000000      0.020842792
## PRESIÓN ARTERIAL    0.262891276  0.009030955      0.020842792      1.000000000
## COLESTEROL          0.005616323 -0.151955874      0.006208824      0.097141454
## NIVEL DE AZÚCAR     0.198039066  0.120075988      0.116702543      0.068212345
## ECG EN REPOSO       0.213151961 -0.018343366      0.031383214      0.095035439
## FREC CARDÍACA MÁX  -0.382044675 -0.189185764     -0.343653677     -0.110176267
## ANGINA x EJERCICIO  0.215792691  0.190664102      0.416624805      0.153592641
## OLDPEAK             0.258611536  0.105733537      0.245026820      0.173737648
## PENDIENTE ST        0.268263994  0.150692544      0.317479540      0.081664371
## E. CARDIACA         0.282038506  0.305444916      0.471354496      0.117224468
##                      COLESTEROL NIVEL DE AZÚCAR ECG EN REPOSO FREC CARDÍACA MÁX
```

```
## EDAD                 0.005616323      0.19803907    0.21315196       -0.38204468
## SEXO                -0.151955874      0.12007599   -0.01834337       -0.18918576
## TIPO DOLOR TORAX     0.006208824      0.11670254    0.03138321       -0.34365368
## PRESIÓN ARTERIAL     0.097141454      0.06821235    0.09503544       -0.11017627
## COLESTEROL           1.000000000     -0.06364625    0.09019491        0.07406413
## NIVEL DE AZÚCAR     -0.063646250      1.00000000    0.05070670       -0.13143849
## ECG EN REPOSO        0.090194905      0.05070670    1.00000000        0.04855228
## FREC CARDÍACA MÁX    0.074064127     -0.13143849    0.04855228        1.00000000
## ANGINA x EJERCICIO   0.046753247      0.06045067    0.03611881       -0.37042487
## OLDPEAK              0.059177330      0.05269786    0.11442795       -0.16069055
## PENDIENTE ST         0.012148055      0.17577434    0.07880669       -0.34341944
## E. CARDIACA         -0.014085607      0.26729119    0.06101109       -0.40042077
##                    ANGINA x EJERCICIO     OLDPEAK PENDIENTE ST E. CARDIACA
## EDAD                         0.21579269  0.25861154   0.26826399  0.28203851
## SEXO                         0.19066410  0.10573354   0.15069254  0.30544492
## TIPO DOLOR TORAX             0.41662480  0.24502682   0.31747954  0.47135450
## PRESIÓN ARTERIAL             0.15359264  0.17373765   0.08166437  0.11722447
## COLESTEROL                   0.04675325  0.05917733   0.01214805 -0.01408561
## NIVEL DE AZÚCAR              0.06045067  0.05269786   0.17577434  0.26729119
## ECG EN REPOSO                0.03611881  0.11442795   0.07880669  0.06101109
## FREC CARDÍACA MÁX           -0.37042487 -0.16069055  -0.34341944 -0.40042077
## ANGINA x EJERCICIO           1.00000000  0.40875250   0.42870594  0.49428199
## OLDPEAK                      0.40875250  1.00000000   0.50192127  0.40395072
## PENDIENTE ST                 0.42870594  0.50192127   1.00000000  0.55877071
## E. CARDIACA                  0.49428199  0.40395072   0.55877071  1.00000000
```

```r
#Representación de las correlaciones
corrplot(cor_datos, method = "pie", type="upper")
```



Análisis de componentes principales (PCA)

Ahora se va a realizar un análisis de componentes sobre el conjunto de datos final. Lo primero que vamos a calcular es la varianza de todas las caracteristicas

```
#Cálculo de la varianza de los componentes.
var <- apply(datos_final, 2, var)
var
```

```
##               EDAD               SEXO    TIPO DOLOR TORAX   PRESIÓN ARTERIAL
##         88.9742542          0.1662200          0.8668185        323.8089631
##         COLESTEROL    NIVEL DE AZÚCAR      ECG EN REPOSO  FREC CARDÍACA MÁX
##       3174.3144763          0.1789676          0.6495843        648.2286144
## ANGINA x EJERCICIO            OLDPEAK       PENDIENTE ST        E. CARDIACA
##          0.2410734          1.1375719          0.3685172          0.2474204
```

Como se puede observar de una manera bastante clara, el colesterol es la característica que mas varia de un individuo a otro.

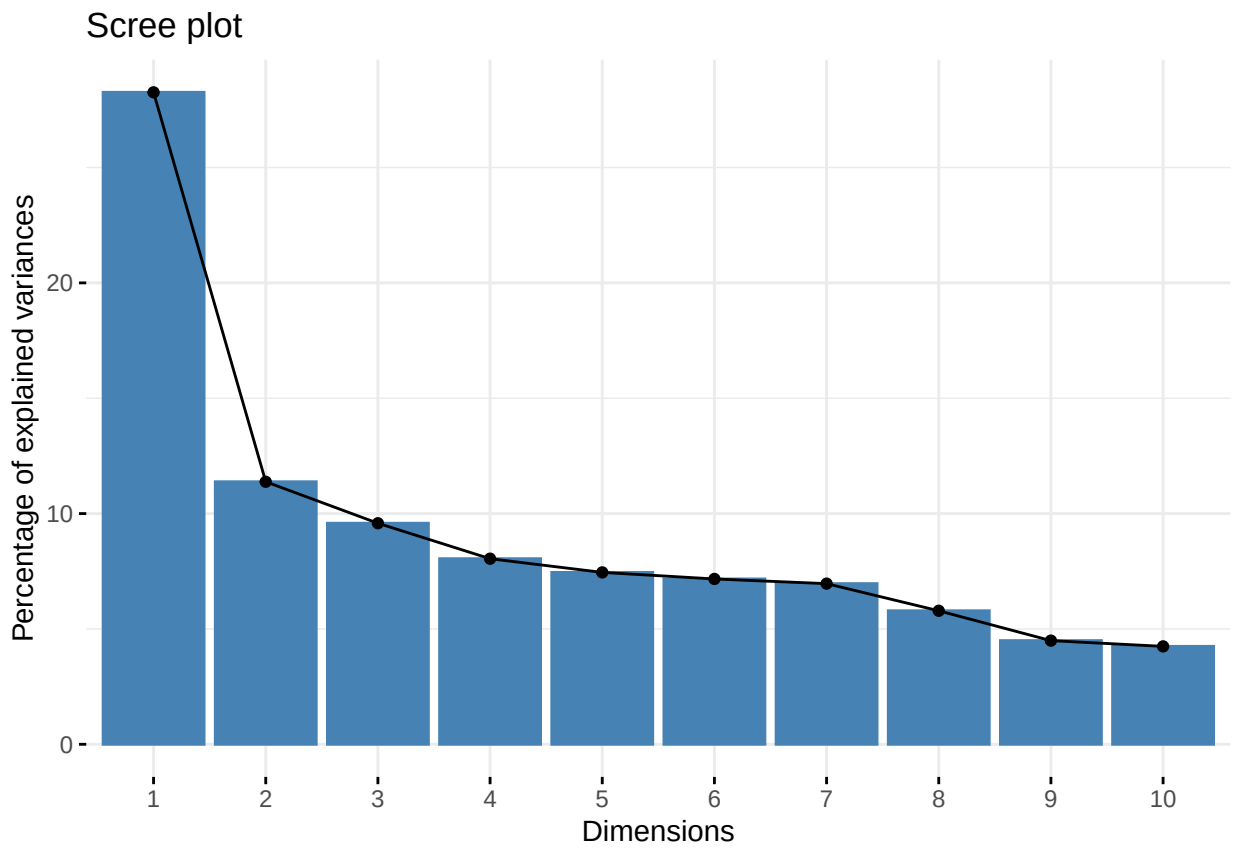
Lo siguiente es centrar y escalar las características, para que así las variables pierdan esa variabilidad. Una vez calculada la matriz se la asigno al pca

```
#Calculo de la descomposición de los componentes
pca <- prcomp(datos_final, scale = TRUE, center = TRUE)
pca
```

```
## Standard deviations (1, .., p=12):
##  [1] 1.8414832 1.1685629 1.0722732 0.9825030 0.9455585 0.9273675 0.9141203
##  [8] 0.8333359 0.7343364 0.7136601 0.6547302 0.6057475
##
## Rotation (n x k) = (12 x 12):
##                              PC1         PC2         PC3         PC4
## EDAD                  0.284048866 -0.30517683  0.41856210 -0.20384320
## SEXO                  0.190969276  0.39449511  0.16865300  0.16193107
## TIPO DOLOR TORAX      0.331417653  0.17210497 -0.19749484  0.01547865
## PRESIÓN ARTERIAL      0.141960876 -0.43892391  0.23377688 -0.47589663
## COLESTEROL           -0.001532397 -0.47547187 -0.39885412 -0.01852007
## NIVEL DE AZÚCAR       0.168340228  0.06772899  0.54782117  0.28581383
## ECG EN REPOSO         0.076620122 -0.46831464  0.20600012  0.59215851
## FREC CARDÍACA MÁX    -0.334312700 -0.15316692 -0.15282868  0.43303167
## ANGINA x EJERCICIO    0.383470802  0.02215612 -0.28290592 -0.10456616
## OLDPEAK               0.335675898 -0.20014135 -0.26643201  0.16590620
## PENDIENTE ST          0.396691578 -0.01301685 -0.15815999  0.17420934
## E. CARDIACA           0.435646241  0.12169670 -0.04104176  0.12845454
##                              PC5          PC6         PC7          PC8
## EDAD                  -0.006489118 -0.312031444 -0.06405294 -0.32124883
## SEXO                  -0.386071630  0.353492167  0.55921470 -0.34599647
## TIPO DOLOR TORAX       0.249043402 -0.239923410  0.28918408  0.53864350
## PRESIÓN ARTERIAL      -0.265972899  0.447735016  0.11325151  0.38838123
## COLESTEROL             0.496855304  0.243491441  0.40219228 -0.37882230
## NIVEL DE AZÚCAR        0.552646219  0.378304662 -0.16436894  0.12362529
## ECG EN REPOSO         -0.212556343 -0.369296127  0.30339786  0.13490917
## FREC CARDÍACA MÁX     -0.139721391  0.367351598 -0.10327852  0.25782660
## ANGINA x EJERCICIO    -0.072320086 -0.002050541  0.09095717  0.19379880
## OLDPEAK               -0.297002638  0.179176718 -0.39250476 -0.09082588
## PENDIENTE ST          -0.006639676  0.046735046 -0.36365433 -0.21221325
## E. CARDIACA            0.087591321  0.107832221  0.02554697  0.04245550
##                              PC9         PC10         PC11         PC12
## EDAD                   0.44832484  0.21455152 -0.38417330  0.100553993
## SEXO                   0.13734894  0.01592761  0.05171501  0.158124372
## TIPO DOLOR TORAX       0.48363798 -0.17549553  0.10336521  0.229258739
## PRESIÓN ARTERIAL      -0.09211930 -0.24241053  0.02963464  0.055340082
## COLESTEROL             0.03986805 -0.01236702  0.03624603  0.004704984
## NIVEL DE AZÚCAR       -0.05688082  0.21827866  0.19692582  0.069161874
## ECG EN REPOSO         -0.25825153 -0.05541829  0.13718428 -0.036439818
## FREC CARDÍACA MÁX      0.32360339  0.12769412 -0.52767019  0.151098767
## ANGINA x EJERCICIO    -0.39257347  0.70652520 -0.18512896  0.151192927
## OLDPEAK                0.36962505  0.17225006  0.49729875 -0.229196969
## PENDIENTE ST          -0.25244498 -0.45034071 -0.15996107  0.561669758
## E. CARDIACA           -0.08393729 -0.25558212 -0.44006661 -0.699596894
```

Se puede ver que la primera componente tiene la mayor desviación estándar de todos los componentes. Para verlo de una manera mas clara, se va a representar de una manera grafica la salida anterior

```
#Representación PCA´s anteriores
fviz_eig(pca)
```



Como se ha visto antes, tanto de una manera numérica como gráfica, el PC1 es el que mejor de todos con una diferencia notable. Si usamos la técnica del codo, deberíamos coger solamente las dos primeras componentes.

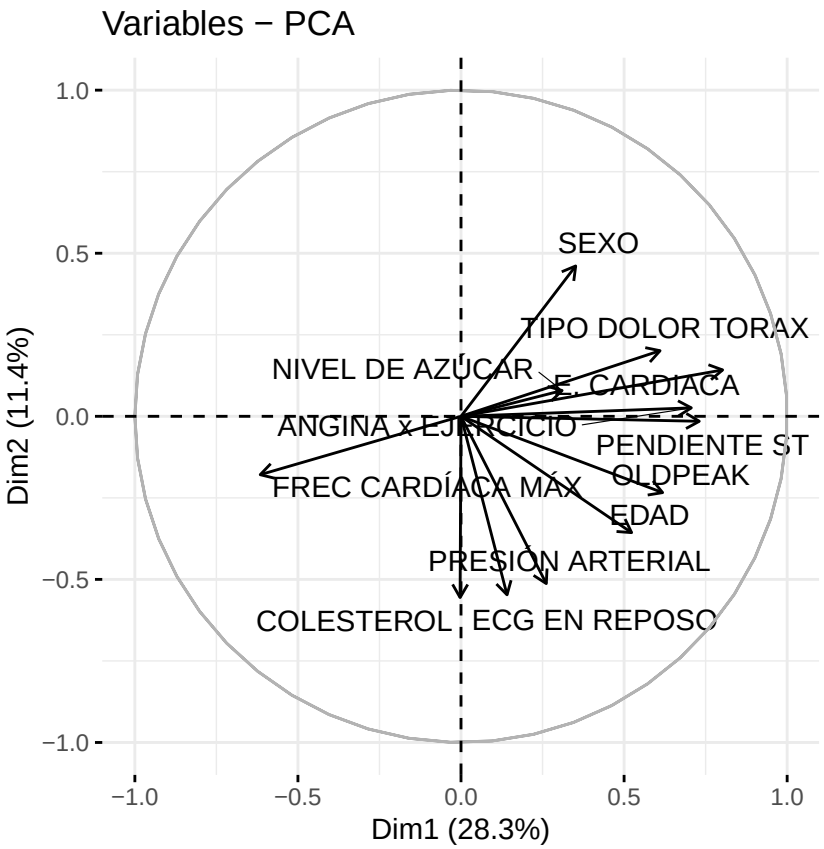
Para confirmar la interpretación, no estaría de más obtener las estadísticas de todas las componentes

```
#Estadísticas de las componentes
summary(pca)
```

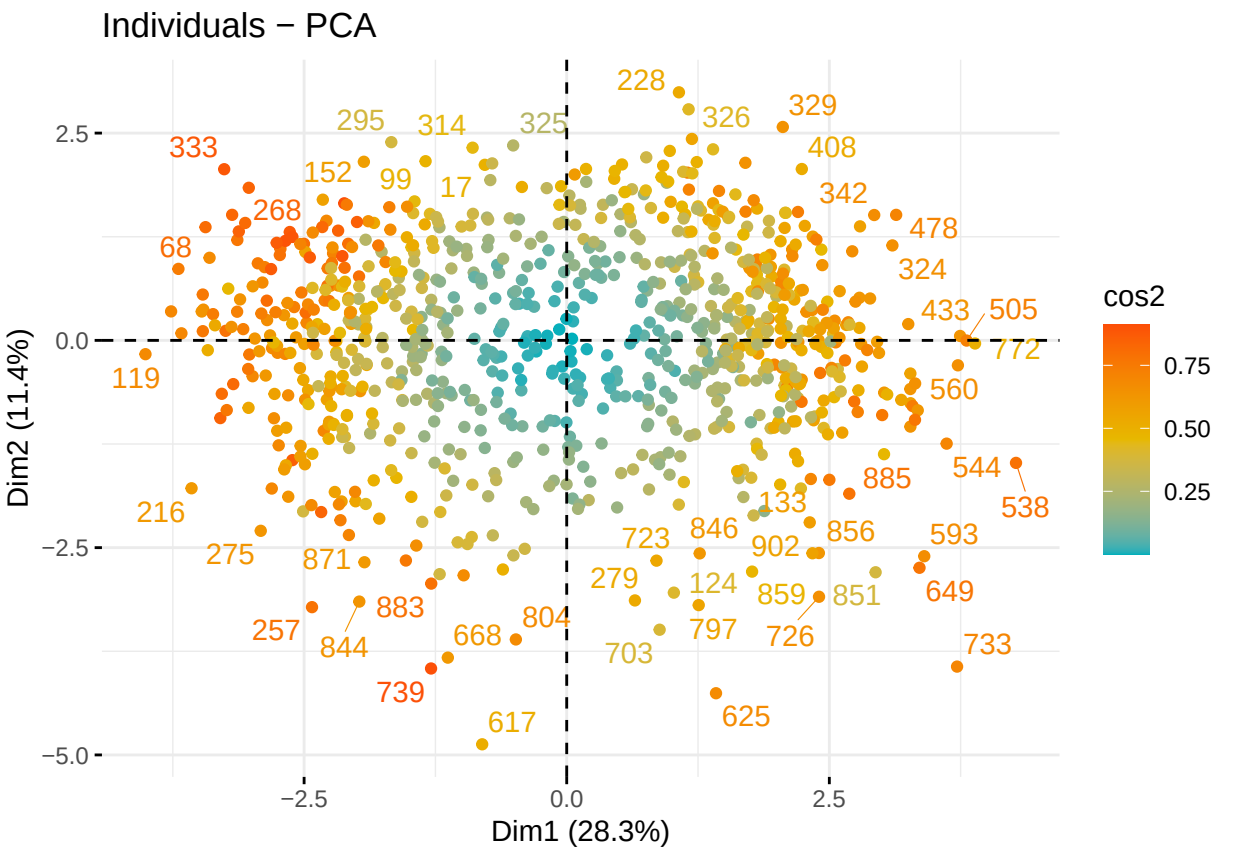
```
## Importance of components:
##                           PC1    PC2     PC3     PC4     PC5     PC6     PC7
## Standard deviation     1.8415 1.1686 1.07227 0.98250 0.94556 0.92737 0.91412
## Proportion of Variance 0.2826 0.1138 0.09581 0.08044 0.07451 0.07167 0.06963
## Cumulative Proportion  0.2826 0.3964 0.49220 0.57264 0.64715 0.71881 0.78845
##                           PC8     PC9    PC10    PC11    PC12
## Standard deviation     0.83334 0.73434 0.71366 0.65473 0.60575
## Proportion of Variance 0.05787 0.04494 0.04244 0.03572 0.03058
## Cumulative Proportion  0.84632 0.89126 0.93370 0.96942 1.00000
```

Viendo las estadísticas vemos que con las dos primeras componentes solamente podríamos explicar un 39,64% de los datos.Como no queremos perder información en el modelo, nos tendríamos que quedar con todas las componentes. Para verlo de una manera visual, se va a representar la PCA de una manera gráfica.

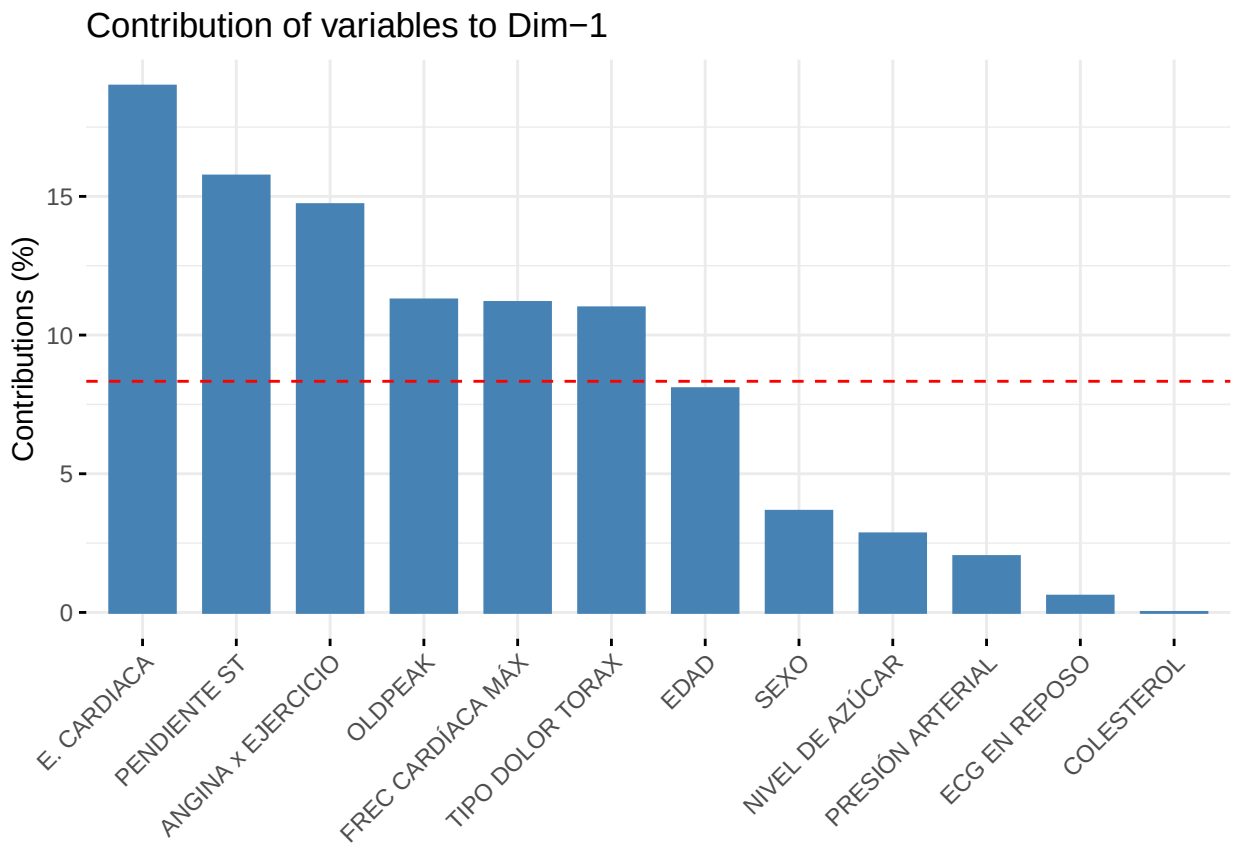
```
#Representación de variables sobre componentes principales
fviz_pca_var(pca, repel = TRUE, scale = 0)
```




```
#Representación de observaciones sobre componentes principales
fviz_pca_ind(pca, col.ind = "cos2", gradient.cols = c("#00AFBB", "#E7B800", "#FC4E07"), repel = TRUE)
```

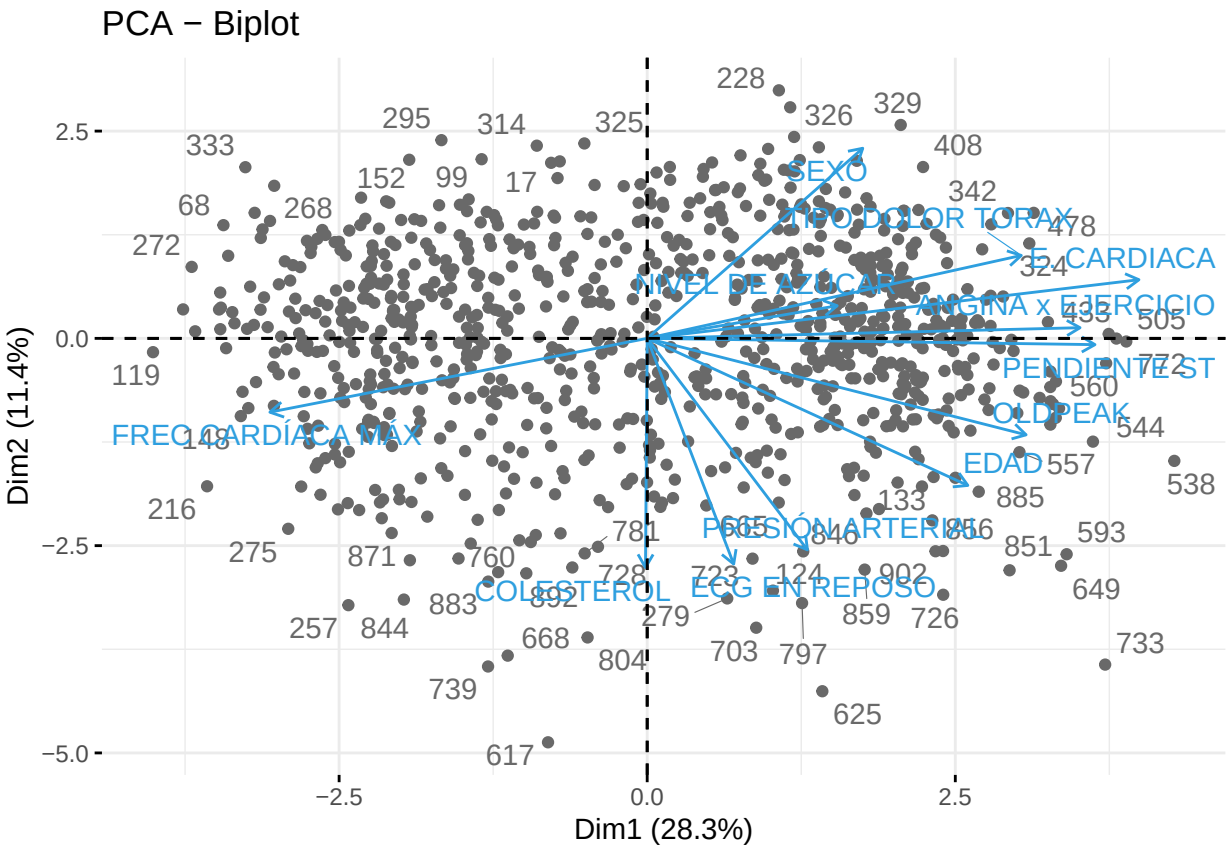



```
#Representa la contribución de filas/columnas de los resultados de un pca
fviz_contrib(pca,choice = "var")
```



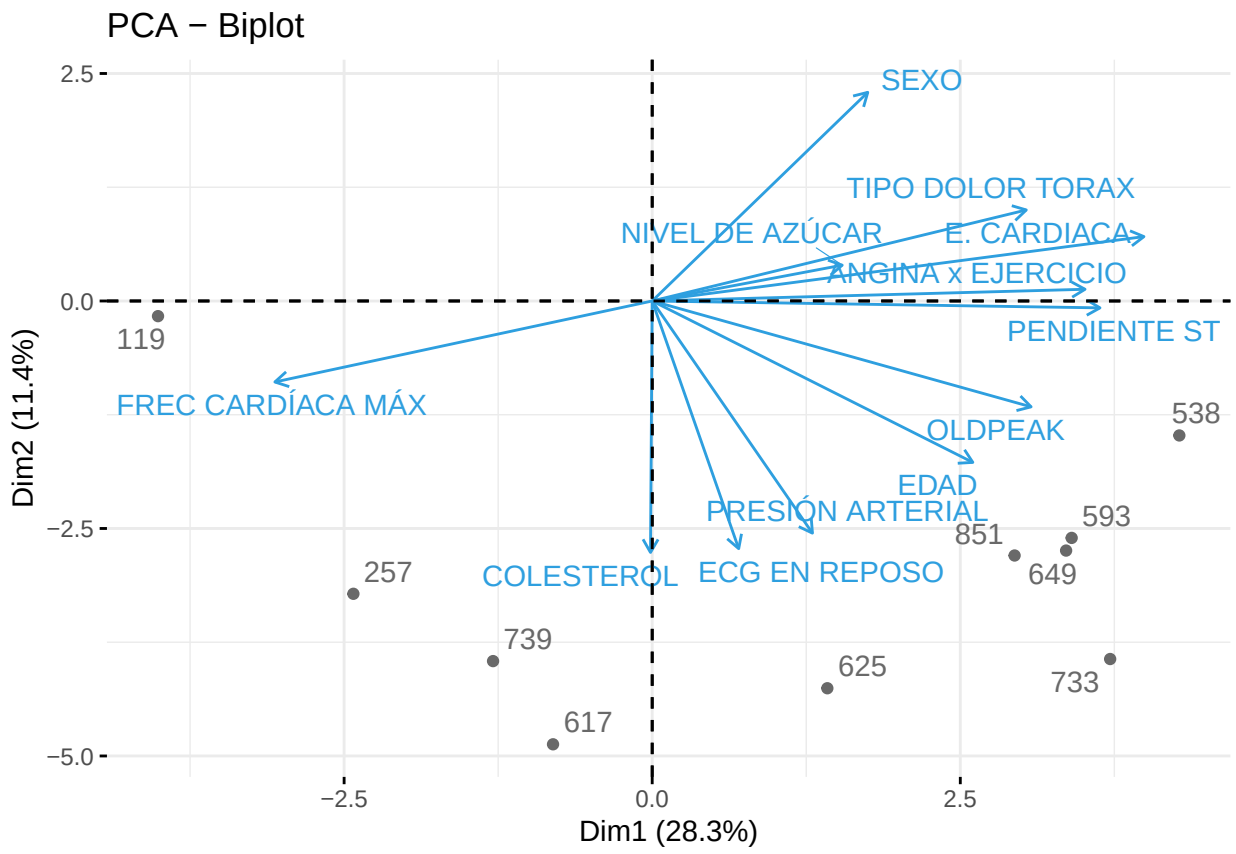
Una vez que hemos representada las variables y los individuos, se va a fusionar estas dos gráficas

```
#Representación de variables y los individuos en la misma gráfica
fviz_pca_biplot(pca, repel = TRUE, col.var = "#2E9FDF", col.ind = "#696969")
```

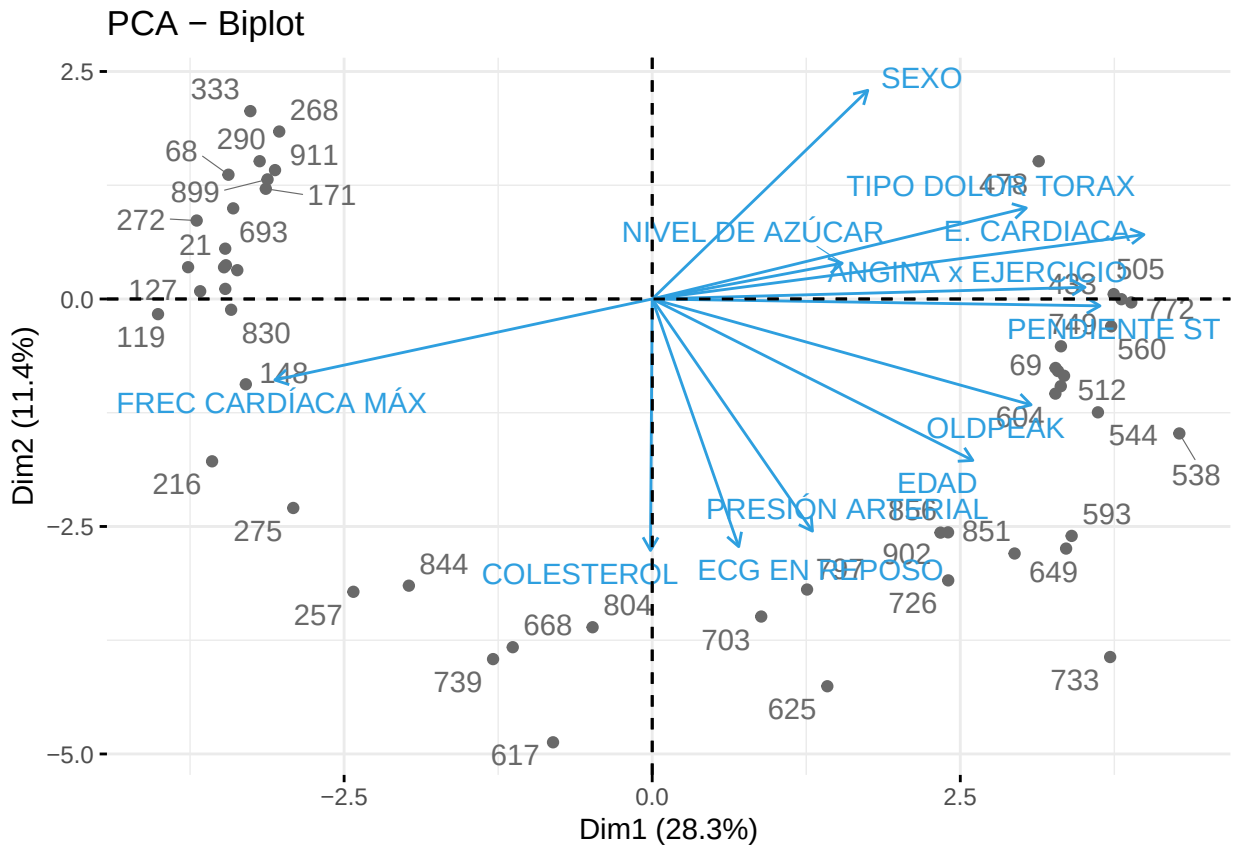


Aunque la opción de repelerse esta activada al ser bastantes casos no se puede ver una manera correcta, así que se a mostrar solamente los 10, 50 y 100 casos más influyentes

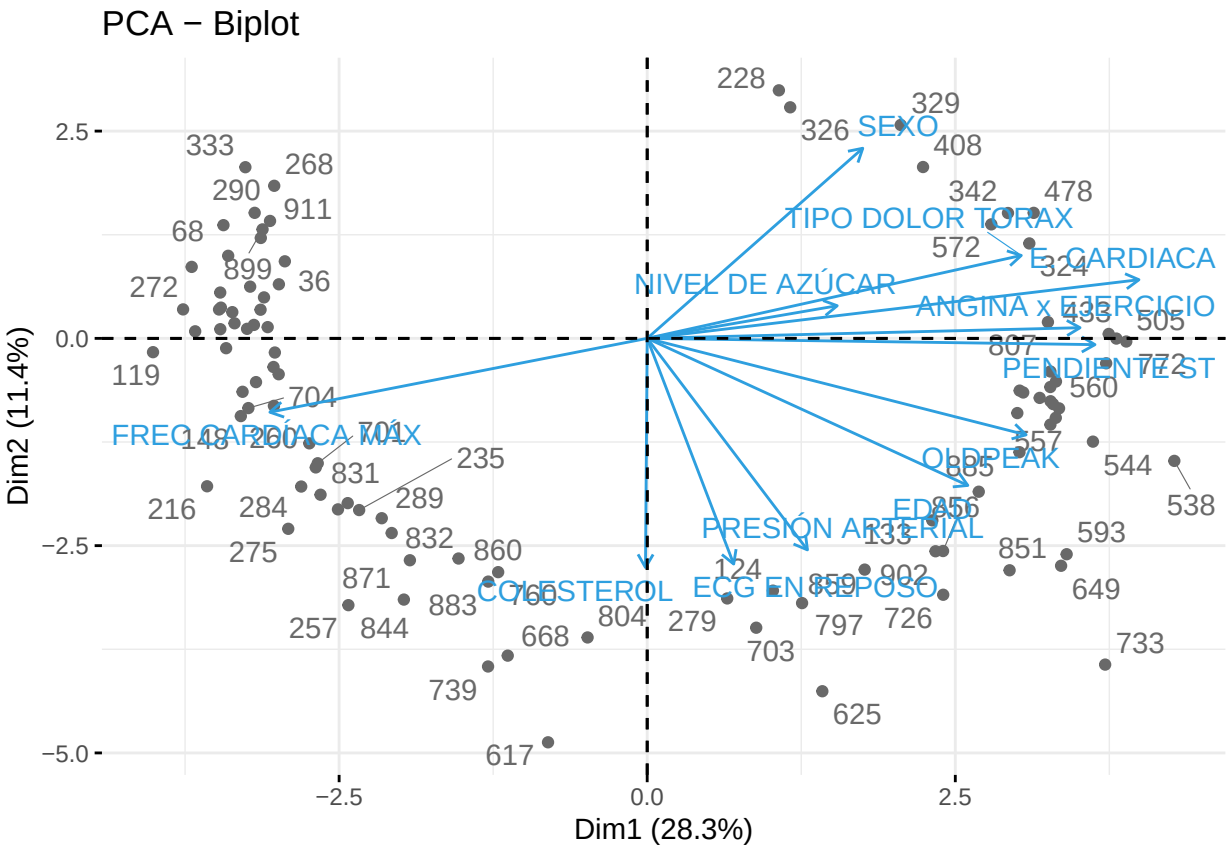
```
#Representación de variables y los 10 individuos más influyentes en la misma gráfica
fviz_pca_biplot(pca, repel = TRUE, col.var = "#2E9FDF",
                col.ind = "#696969", select.ind = list(contrib = 10))
```




```
#Representación de variables y los 50 individuos más influyentes en la misma gráfica
fviz_pca_biplot(pca, repel = TRUE, col.var = "#2E9FDF",
                col.ind = "#696969", select.ind = list(contrib = 50))
```




```
#Representación de variables y los 100 individuos más influyentes en la misma gráfica
fviz_pca_biplot(pca, repel = TRUE, col.var = "#2E9FDF",
                col.ind = "#696969", select.ind = list(contrib = 100))
```



Al mostrar solamente los casos mas influyentes, se puede ver con mas claridad las relaciones entre los individuos y las características. Podemos concluir de este análisis de componentes, que no se puede quitar ninguna característica ya que se perdería información.

Análisis de los datos