¿Cómo realizar la limpieza y análisis de datos?

Autores: Eduardo Mora González y Diego Sánchez De La Fuente

Enero 2023

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)  
if (!require('dplyr')) install.packages("dplyr"); library(dplyr)  
if (!require('DescTools')) install.packages("DescTools"); library(DescTools)  
if (!require('regclass')) install.packages("regclass"); library(regclass)  
if(!require('randomForest')) install.packages('randomForest',repos='http://cran.us.r-project.org'); library(randomForest)  
if(!require('iml')) install.packages('iml', repos='http://cran.us.r-project.org'); library(iml)  
if(!require("tidyverse"))install.packages("DeskTools");library("tidyverse")  
if(!require("rpart")) install.packages("rpart");library("rpart")  
if(!require("rpart.plot")) install.packages("rpart.plot"); library("rpart.plot")  
if(!require("caret"))install.packages("caret");library("caret")  
if(!require('patchwork'))install.packages('patchwork',repos='http://cran.us.r-project.org');library(patchwork)

# Dataset

## Motivación

En Europa, el paro cardiaco es una de las primeras causas de mortalidad y en España fallecen en torno a 100 personas al día por este suceso [<https://fundaciondelcorazon.com/prensa/notas-de-prensa/2900-solo-el-30-de-espanoles-sabe-realizar-la-reanimacion-cardio-pulmonar-rcp-.html>], esto representa aproximadamente el 31% de las muertes a nivel mundial.

## Descripción del dataset

El conjunto de datos ha sido extraido de Kaggle: <https://www.kaggle.com/datasets/rashikrahmanpritom/heart-attack-analysis-prediction-dataset>, está compuesto de 12 variables y 918 registros. Que correlacionan una serie de caracteristicas recogidas de varios pacientes con la posibilidad de sufrir un ataque al corazón.

Explicación de cada variable:

* **Age**: Edad del paciente
* **Sex**: Sexo del paciente
* **ChestPainType**: Tipo de dolor torácico: Angina Típica Angina Atípica Dolor no debido a una angina Asintomático
* **RestingBP**: Presión arterial en reposo (en mm Hg)
* **Cholesterol**: Colesterol en sangre (mg/dL)
* **FastingBS**: Tiene Glucemia en ayunas > 120 mg/dl -> (1: True, 0: False)
* **RestingECG**: Resultados electrocardiográficos en reposo Value 0: normal Value 1: Tiene anormalidad de la onda ST-T (inversiones de la onda T y/o elevación o depresión del ST > 0.05 mV) Value 2 Muestra hipertrofia ventricular izquierda probable o definitiva según los criterios de Estes
* **MaxHR**: Frecuencia cardíaca máxima alcanzada
* **ExerciseAngina**: Angina inducida por el ejercicio (1 = sí, 0 = no)
* **Oldpeak**: Descenso del segmento ST inducido por el ejercicio en relación con el reposo (‘Segmento ST’ se relaciona con las posiciones en el gráfico de Electro cardiograma).
* **ST\_Slope**: La pendiente del segmento ST de ejercicio máximo: 0: pendiente descendente 1: plano 2: pendiente ascendente
* **HeartDisease**: Variable Objetivo: 0= menos posibilidades de infarto 1= más posibilidades de infarto.

## Carga del fichero de datos

datos <- read\_csv("./fichero\_original\_datos.csv")  
attach(datos)

## Tipos de datos

# Cargamos en un vector los tipos de variable del datase  
vector\_tipos <- sapply(datos, function(x) class(x))  
print(vector\_tipos)

## Age Sex ChestPainType RestingBP Cholesterol   
## "numeric" "character" "character" "numeric" "numeric"   
## FastingBS RestingECG MaxHR ExerciseAngina Oldpeak   
## "numeric" "character" "numeric" "character" "numeric"   
## ST\_Slope HeartDisease   
## "character" "numeric"

Ahora vamos a ver las estructura del juego de datos

str(datos)

## spc\_tbl\_ [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

Observamos los primeros 5 registros:

head(datos, 5L)

## # A tibble: 5 x 12  
## Age Sex ChestPainType RestingBP Cholesterol FastingBS RestingECG MaxHR  
## <dbl> <chr> <chr> <dbl> <dbl> <dbl> <chr> <dbl>  
## 1 40 M ATA 140 289 0 Normal 172  
## 2 49 F NAP 160 180 0 Normal 156  
## 3 37 M ATA 130 283 0 ST 98  
## 4 48 F ASY 138 214 0 Normal 108  
## 5 54 M NAP 150 195 0 Normal 122  
## # ... with 4 more variables: ExerciseAngina <chr>, Oldpeak <dbl>,  
## # ST\_Slope <chr>, HeartDisease <dbl>

## Objetivo buscado

Se puede decir que el objetivo buscado es predecir la posibilidad de que una persona tenga un alto riesgo de ser diagnosticado como un paciente cardíaco a través de las diversas características. Para llegar a al objetivo se tiene pensado realizar diversos métodos de análisis para así relacionar las diversas características para obtener unos parámetros finales y así concluir la posibilidad de que una persona tenga o no una enfermedad cardiaca.

# Preprocesado, gestión de características y exploración de los datos.

## 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=="")

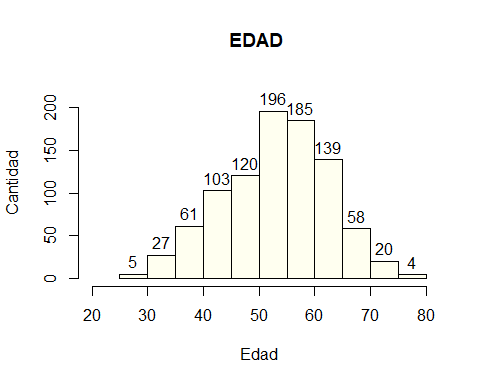
## Age Sex ChestPainType RestingBP Cholesterol   
## 0 0 0 0 0   
## FastingBS RestingECG MaxHR ExerciseAngina Oldpeak   
## 0 0 0 0 0   
## ST\_Slope HeartDisease   
## 0 0

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 (Age)

#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.

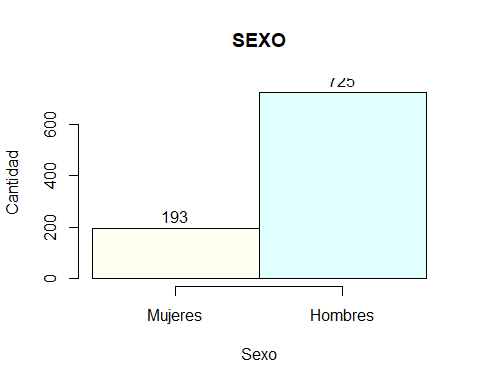
### SEXO (Sex)

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)



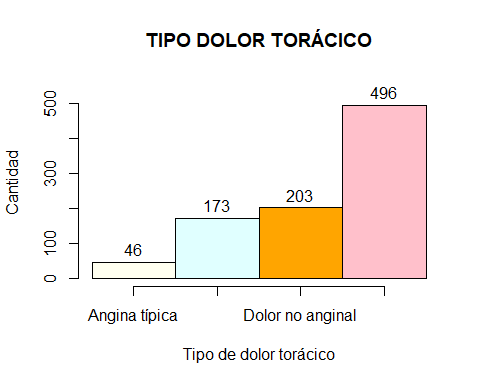
### TIPO DE DOLOR TORÁCICO (ChestPainType)

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

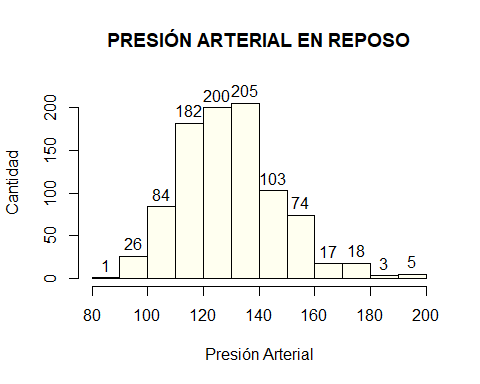
Como solo aparece una vez, se le asignará un valor por defecto. El valor por defecto será el más común.

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

#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)

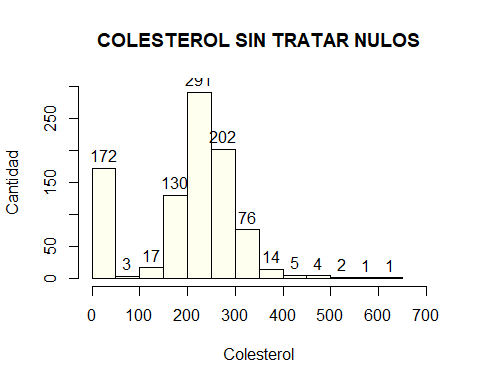
La siguiente característica es de tipo numérico. Al igual que en la presión arterial en reposo, que tenemos valores 0 que debemos analizar. Lo primero que se va a hacer es obtener el numero de casos que el colesterol es 0, y se consideraran las diversas formas de tratar estos datos.

#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))



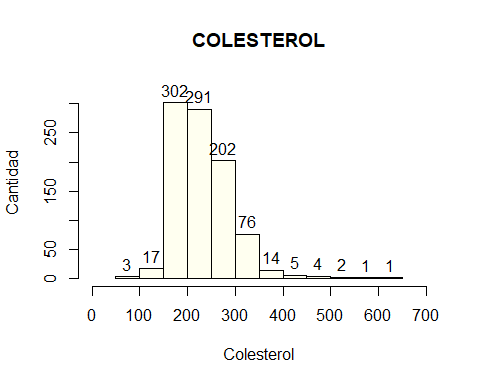
Tras analizar la gráfica y para no perder estos datos, se le asignaran un valor por defecto, que será la media de los datos. Esta decisión se ha tomado ya que poner el más común, nos crearía un conjunto de datos muy distintos entre unas medidas y otras, mientras que poner la media sería un valor que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

#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.

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))

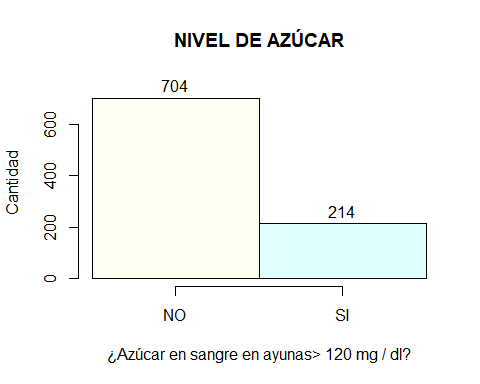


### NIVEL DE AZÚCAR EN SANGRE EN AYUNAS (FastingBS)

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:

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.

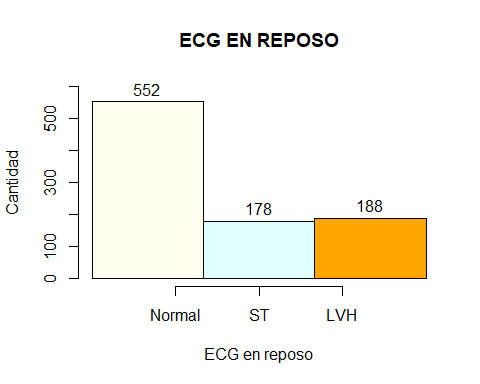
### ECG EN REPOSO (RestingECG)

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:

#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.

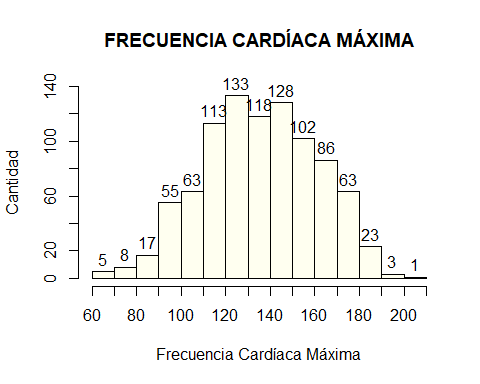
h1 <- hist(datos$RestingECG, xlab="ECG en reposo",  
 col= c("ivory", "lightcyan", "ORANGE"),  
 ylab="Cantidad", main="ECG EN REPOSO",  
 ylim = c(0, 600), axes = FALSE,  
 breaks=seq(min(datos$RestingECG)-0.5,  
 max(datos$RestingECG)+0.5, by=1) )  
text(h1$mids,h1$counts,labels=h1$counts, adj=c(0.5,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

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).

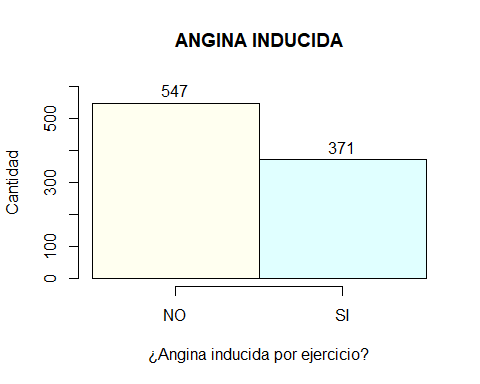
### ANGINA INDUCIDA POR EJERCICIO (ExerciseAngina)

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)

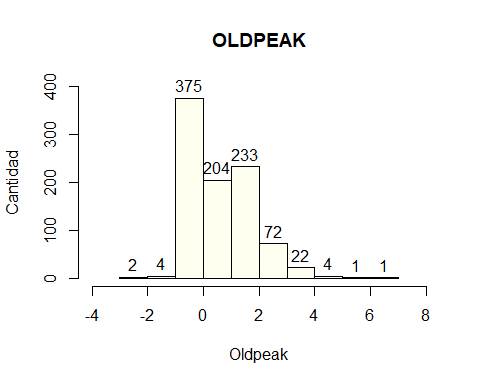


Como se puede apreciar, hay mas casos en que NO se ha producido una angina inducida por el ejercicio de que Si se haya producido.

### OLDPEAK

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

h1 <- hist(datos$Oldpeak, xlab="Oldpeak", col="ivory", ylab="Cantidad", main="OLDPEAK", ylim = c(0,400), xlim = c(-4, 8))  
text(h1$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

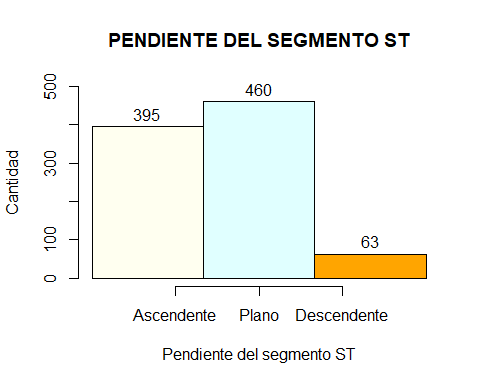
### PENDIENTE DEL SEGMENTO ST (ST\_Slope)

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)

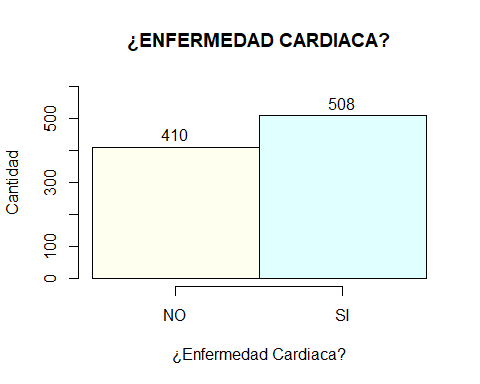


El caso más común es que la pendiente sea plana, teniendo menos casos en los casos descendentes.

### ¿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.

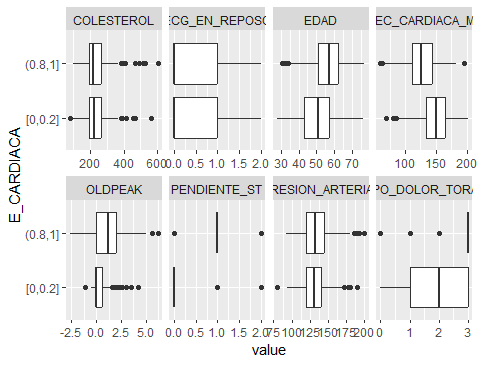
## Construcción de conjunto de datos final

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]<- "PRESION\_ARTERIAL"  
colnames(datos\_final)[5]<- "COLESTEROL"  
colnames(datos\_final)[6]<- "NIVEL\_DE\_AZUCAR"  
colnames(datos\_final)[7]<- "ECG\_EN\_REPOSO"  
colnames(datos\_final)[8]<- "FREC\_CARDIACA\_MAX"  
colnames(datos\_final)[9]<- "ANGINA\_x\_EJERCICIO"  
colnames(datos\_final)[10]<- "OLDPEAK"  
colnames(datos\_final)[11]<- "PENDIENTE\_ST"  
colnames(datos\_final)[12]<- "E\_CARDIACA"

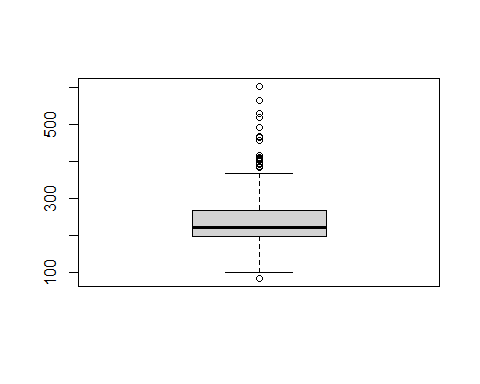
Por ultimo se va a mirar a través de los diagramas de cajas el rango de las características enfrentado a si un paciente tiene una enfermedad cardiaca o no.

#Diagrama de caja de todas las características enfrentadas a si un paciente tiene enfermedad cardiaca  
plot\_boxplot(datos\_final, by = "E\_CARDIACA")



## Eliminamos outliers

datos\_bp.colesterol <- boxplot(datos\_final$COLESTEROL)



datos\_bp.colesterol.out <- datos\_bp.colesterol$out  
print("Eliminamos Outliers de la variable COLESTEROL con valores: ")

## [1] "Eliminamos Outliers de la variable COLESTEROL con valores: "

datos\_bp.colesterol.out

## [1] 468 518 412 529 85 392 466 393 388 603 404 491 394 458 384 385 564 407 417  
## [20] 409 394

datos\_final <- datos\_final %>% filter(!(COLESTEROL %in% datos\_bp.colesterol.out))  
dev.off()

## null device   
## 1

datos\_frec.cardiaca.max <- boxplot(datos\_final$FREC\_CARDIACA\_MAX)  
datos\_frec.cardiaca.max.out <- datos\_frec.cardiaca.max$out  
print("Eliminamos Outliers de la variable FREC CARDIACA MAX con valores: ")

## [1] "Eliminamos Outliers de la variable FREC CARDIACA MAX con valores: "

datos\_frec.cardiaca.max.out

## [1] 63 60

datos\_final <- datos\_final %>% filter(!(FREC\_CARDIACA\_MAX %in% datos\_frec.cardiaca.max.out))  
dev.off()

## null device   
## 1

datos\_oldpeak <- boxplot(datos\_final$OLDPEAK)  
datos\_oldpeak.out <- datos\_oldpeak$out  
print("Eliminamos Outliers de la variable OLDPEAK con valores: ")

## [1] "Eliminamos Outliers de la variable OLDPEAK con valores: "

datos\_oldpeak.out

## [1] 4.0 5.0 -2.6 4.0 4.0 4.0 4.0 4.2 4.0 5.6 3.8 4.2 6.2 4.4 4.0

datos\_final <- datos\_final %>% filter(!(OLDPEAK %in% datos\_oldpeak.out))  
dev.off()

## null device   
## 1

datos\_bp.presion\_arterial <- boxplot(datos\_final$PRESION\_ARTERIAL)  
datos\_bp.presion\_arterial.out <- datos\_bp.presion\_arterial$out  
print("Eliminamos Outliers de la variable PRESIÓN ARTERIAL ST con valores: ")

## [1] "Eliminamos Outliers de la variable PRESIÓN ARTERIAL ST con valores: "

datos\_bp.presion\_arterial.out

## [1] 190 180 180 200 180 180 180 80 200 185 200 180 180 178 172 180 190 174 180  
## [20] 192 178 180 180 172

datos\_final <- datos\_final %>% filter(!(PRESION\_ARTERIAL %in% datos\_bp.presion\_arterial.out))  
dev.off()

## null device   
## 1

datos\_bp.tipo\_dolor\_torax <- boxplot(datos\_final$TIPO\_DOLOR\_TORAX)  
datos\_bp.tipo\_dolor\_torax.out <- datos\_bp.tipo\_dolor\_torax$out  
print("Eliminamos Outliers de la variable TIPO DOLOR TORAX con valores: ")

## [1] "Eliminamos Outliers de la variable TIPO DOLOR TORAX con valores: "

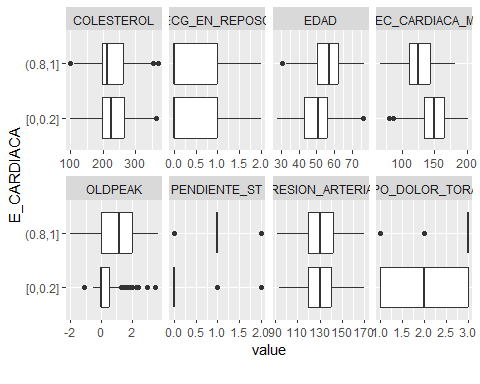
datos\_bp.tipo\_dolor\_torax.out

## [1] 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0  
## [39] 0 0 0 0 0 0

datos\_final <- datos\_final %>% filter(!(TIPO\_DOLOR\_TORAX %in% datos\_bp.tipo\_dolor\_torax.out))  
dev.off()

## null device   
## 1

#Diagrama de caja de todas las características enfrentadas a si un paciente tiene enfermedad cardiaca  
plot\_boxplot(datos\_final, by = "E\_CARDIACA")

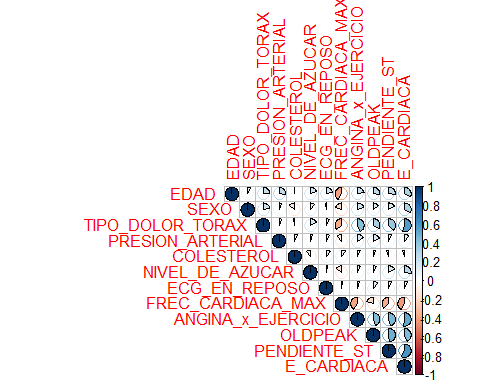


## Correlaciones

#Calculamos las correlaciones  
cor\_datos <- cor(datos\_final)  
cor\_datos

## EDAD SEXO TIPO\_DOLOR\_TORAX PRESION\_ARTERIAL  
## EDAD 1.0000000000 0.07987476 0.23222966 0.26263094  
## SEXO 0.0798747606 1.00000000 0.21446723 0.06480334  
## TIPO\_DOLOR\_TORAX 0.2322296603 0.21446723 1.00000000 0.05761116  
## PRESION\_ARTERIAL 0.2626309425 0.06480334 0.05761116 1.00000000  
## COLESTEROL -0.0002226103 -0.13574432 -0.04056882 0.06464654  
## NIVEL\_DE\_AZUCAR 0.1888535829 0.11982017 0.17113944 0.04940893  
## ECG\_EN\_REPOSO 0.2045327265 -0.01651334 0.09834667 0.07584846  
## FREC\_CARDIACA\_MAX -0.4065414509 -0.19479093 -0.36601006 -0.11900267  
## ANGINA\_x\_EJERCICIO 0.2232404235 0.20949172 0.42401538 0.17055230  
## OLDPEAK 0.2647072227 0.13871819 0.32323764 0.18156990  
## PENDIENTE\_ST 0.2649516503 0.16347763 0.38922582 0.06577099  
## E\_CARDIACA 0.3112477185 0.31757032 0.55357954 0.11663793  
## COLESTEROL NIVEL\_DE\_AZUCAR ECG\_EN\_REPOSO  
## EDAD -0.0002226103 0.18885358 0.20453273  
## SEXO -0.1357443154 0.11982017 -0.01651334  
## TIPO\_DOLOR\_TORAX -0.0405688154 0.17113944 0.09834667  
## PRESION\_ARTERIAL 0.0646465372 0.04940893 0.07584846  
## COLESTEROL 1.0000000000 -0.11313552 0.10184968  
## NIVEL\_DE\_AZUCAR -0.1131355201 1.00000000 0.03428603  
## ECG\_EN\_REPOSO 0.1018496822 0.03428603 1.00000000  
## FREC\_CARDIACA\_MAX 0.1075182058 -0.14779921 0.01659835  
## ANGINA\_x\_EJERCICIO 0.0643936183 0.07501356 0.05301848  
## OLDPEAK 0.0471006517 0.08503826 0.11254750  
## PENDIENTE\_ST -0.0493791928 0.17525733 0.06863116  
## E\_CARDIACA -0.0477984682 0.28341365 0.08182717  
## FREC\_CARDIACA\_MAX ANGINA\_x\_EJERCICIO OLDPEAK  
## EDAD -0.40654145 0.22324042 0.26470722  
## SEXO -0.19479093 0.20949172 0.13871819  
## TIPO\_DOLOR\_TORAX -0.36601006 0.42401538 0.32323764  
## PRESION\_ARTERIAL -0.11900267 0.17055230 0.18156990  
## COLESTEROL 0.10751821 0.06439362 0.04710065  
## NIVEL\_DE\_AZUCAR -0.14779921 0.07501356 0.08503826  
## ECG\_EN\_REPOSO 0.01659835 0.05301848 0.11254750  
## FREC\_CARDIACA\_MAX 1.00000000 -0.39834669 -0.18975488  
## ANGINA\_x\_EJERCICIO -0.39834669 1.00000000 0.43365158  
## OLDPEAK -0.18975488 0.43365158 1.00000000  
## PENDIENTE\_ST -0.36930737 0.45642629 0.48875764  
## E\_CARDIACA -0.42346641 0.51376813 0.43751373  
## PENDIENTE\_ST E\_CARDIACA  
## EDAD 0.26495165 0.31124772  
## SEXO 0.16347763 0.31757032  
## TIPO\_DOLOR\_TORAX 0.38922582 0.55357954  
## PRESION\_ARTERIAL 0.06577099 0.11663793  
## COLESTEROL -0.04937919 -0.04779847  
## NIVEL\_DE\_AZUCAR 0.17525733 0.28341365  
## ECG\_EN\_REPOSO 0.06863116 0.08182717  
## FREC\_CARDIACA\_MAX -0.36930737 -0.42346641  
## ANGINA\_x\_EJERCICIO 0.45642629 0.51376813  
## OLDPEAK 0.48875764 0.43751373  
## PENDIENTE\_ST 1.00000000 0.58120599  
## E\_CARDIACA 0.58120599 1.00000000

#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 PRESION\_ARTERIAL   
## 88.1377792 0.1621106 0.6374466 245.0442666   
## COLESTEROL NIVEL\_DE\_AZUCAR ECG\_EN\_REPOSO FREC\_CARDIACA\_MAX   
## 2160.3249895 0.1754691 0.6335349 637.4350996   
## ANGINA\_x\_EJERCICIO OLDPEAK PENDIENTE\_ST E\_CARDIACA   
## 0.2422160 0.9478357 0.3549759 0.2478786

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

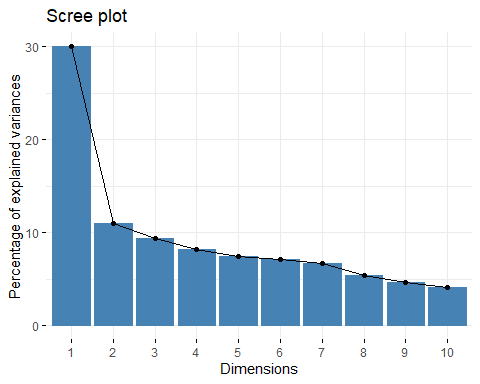
Lo siguiente es centrar y escalar las características, para que así las variables pierdan esa variabilidad. Una vez calculada la matriz se la asigno al pca

#Calculo de la descomposición de los componentes  
pca <- prcomp(datos\_final, scale = TRUE, center = TRUE)  
pca

## Standard deviations (1, .., p=12):  
## [1] 1.8974493 1.1475086 1.0577031 0.9894707 0.9414635 0.9202567 0.8976197  
## [8] 0.8068512 0.7422484 0.6983699 0.6456073 0.5828417  
##   
## Rotation (n x k) = (12 x 12):  
## PC1 PC2 PC3 PC4 PC5  
## EDAD 0.28014843 -0.25245541 0.48732200 -0.07304964 0.256104929  
## SEXO 0.20039780 0.33847469 0.03514677 -0.16476622 -0.763305970  
## TIPO\_DOLOR\_TORAX 0.35837588 0.09136430 -0.13473172 0.16514131 0.005882123  
## PRESION\_ARTERIAL 0.13829428 -0.37958423 0.36362321 -0.59017736 -0.255760964  
## COLESTEROL -0.03058803 -0.57356528 -0.32677314 0.01280796 -0.005505378  
## NIVEL\_DE\_AZUCAR 0.17188118 0.22329356 0.46905923 0.37182111 0.024273900  
## ECG\_EN\_REPOSO 0.08311302 -0.46159376 0.23741130 0.58968414 -0.321590348  
## FREC\_CARDIACA\_MAX -0.33268947 -0.13119755 -0.16819561 0.23310113 -0.385916552  
## ANGINA\_x\_EJERCICIO 0.37320411 -0.07541656 -0.29323460 -0.15482224 0.002822528  
## OLDPEAK 0.33368710 -0.20542115 -0.24820847 0.01026826 -0.089164578  
## PENDIENTE\_ST 0.38478714 0.03185974 -0.20887652 0.12195037 0.151652559  
## E\_CARDIACA 0.43248195 0.10606714 -0.09073580 0.11184415 -0.048966672  
## PC6 PC7 PC8 PC9 PC10  
## EDAD -0.19012829 0.07602767 -0.39586282 0.38465564 0.152868605  
## SEXO -0.11787912 0.27816619 -0.34470023 0.04446174 -0.013857771  
## TIPO\_DOLOR\_TORAX -0.10759466 0.20321763 0.60091060 0.54901398 0.003766156  
## PRESION\_ARTERIAL 0.22105853 -0.19709636 0.37321438 -0.07017404 -0.234844352  
## COLESTEROL 0.33735336 0.62122599 -0.19252011 0.05017177 -0.123957417  
## NIVEL\_DE\_AZUCAR 0.68485907 0.11737352 0.02332883 -0.16064936 0.172660255  
## ECG\_EN\_REPOSO -0.40796143 -0.05485229 0.10908375 -0.27682556 -0.025367300  
## FREC\_CARDIACA\_MAX 0.31452613 -0.32365282 0.06117644 0.29035242 0.035570020  
## ANGINA\_x\_EJERCICIO -0.01691429 0.05956734 0.17454230 -0.48906281 0.549950590  
## OLDPEAK 0.16611307 -0.49474050 -0.28538178 0.26579163 0.355664752  
## PENDIENTE\_ST 0.07037292 -0.27927242 -0.23036132 -0.20690027 -0.608826995  
## E\_CARDIACA 0.09209677 0.05414122 0.07660088 0.04271721 -0.273082385  
## PC11 PC12  
## EDAD -0.41570328 -0.08124084  
## SEXO 0.02697977 -0.14584048  
## TIPO\_DOLOR\_TORAX 0.08643164 -0.30804525  
## PRESION\_ARTERIAL 0.05681729 -0.02320323  
## COLESTEROL 0.08777809 -0.02704705  
## NIVEL\_DE\_AZUCAR 0.10225997 -0.11095277  
## ECG\_EN\_REPOSO 0.11546071 0.02374709  
## FREC\_CARDIACA\_MAX -0.58343555 -0.08461831  
## ANGINA\_x\_EJERCICIO -0.38969530 -0.13144093  
## OLDPEAK 0.46313399 0.10654434  
## PENDIENTE\_ST -0.12372310 -0.45235695  
## E\_CARDIACA -0.24942136 0.78909996

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.

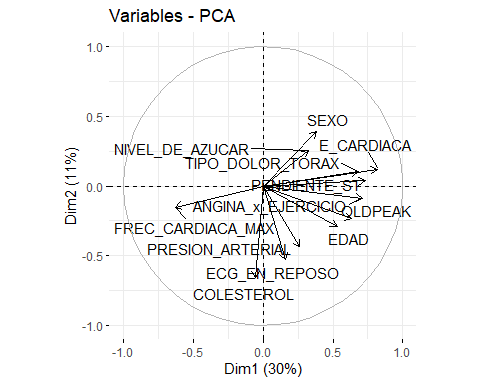
Para confirmar la interpretación, no estaría de más obtener las estadísticas de todas las componentes

#Estadísticas de las componentes  
summary(pca)

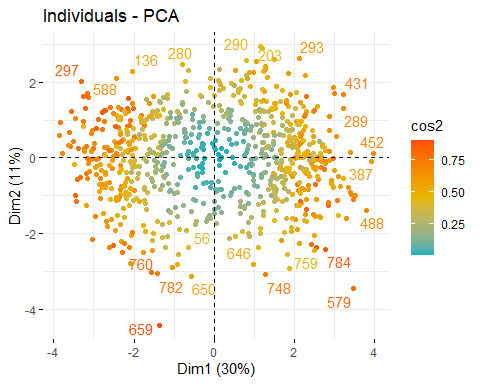
## Importance of components:  
## PC1 PC2 PC3 PC4 PC5 PC6 PC7  
## Standard deviation 1.897 1.1475 1.05770 0.98947 0.94146 0.92026 0.89762  
## Proportion of Variance 0.300 0.1097 0.09323 0.08159 0.07386 0.07057 0.06714  
## Cumulative Proportion 0.300 0.4098 0.50299 0.58457 0.65844 0.72901 0.79615  
## PC8 PC9 PC10 PC11 PC12  
## Standard deviation 0.80685 0.74225 0.69837 0.64561 0.58284  
## Proportion of Variance 0.05425 0.04591 0.04064 0.03473 0.02831  
## Cumulative Proportion 0.85040 0.89631 0.93696 0.97169 1.00000

Viendo las estadísticas vemos que con las dos primeras componentes solamente podríamos explicar un 39,75% 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.

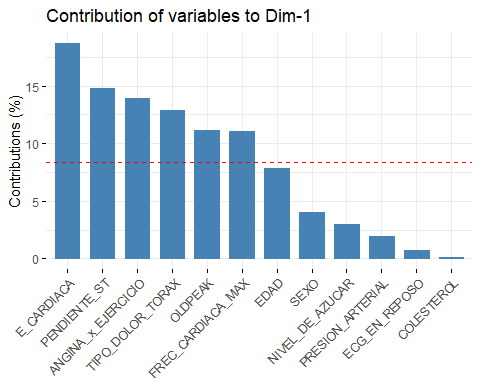
#Representación de variables sobre componentes principales  
fviz\_pca\_var(pca, repel = TRUE, scale = 0)



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

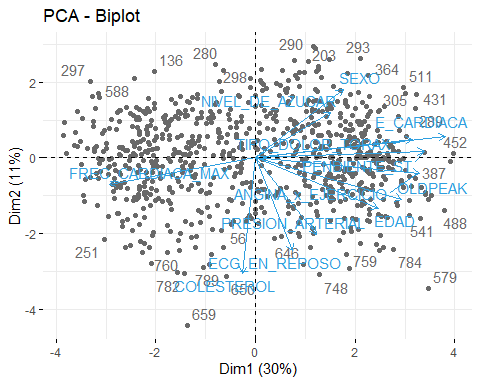


#Representa la contribución de filas/columnas de los resultados de un pca  
fviz\_contrib(pca,choice = "var")



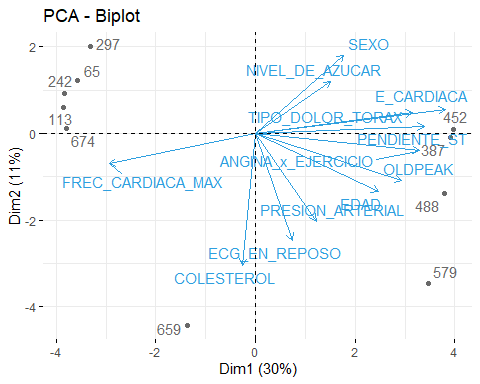
Una vez que hemos representada las variables y los individuos, se va a fusionar estas dos gráficas

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

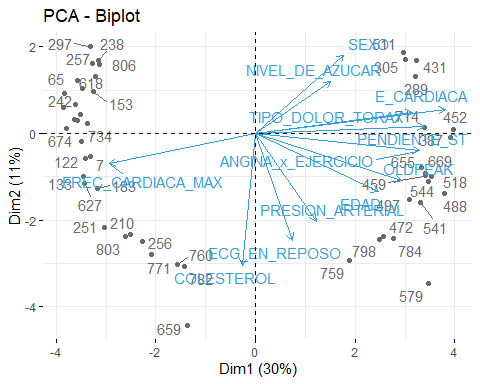


Aunque la opción de repelerse esta activada al ser bastantes casos no se puede ver una manera correcta, así que se a mostrar solamente los 10, 50 y 100 casos más influyentes

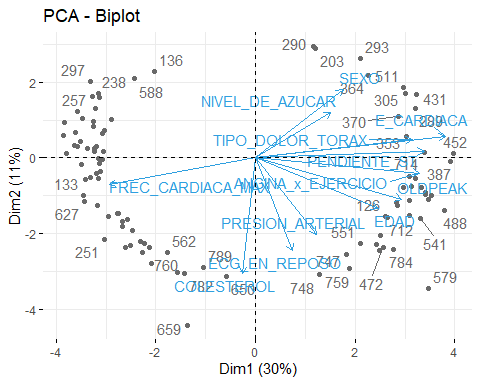
#Representación de variables y los 10 individuos más influyentes en la misma gráfica  
fviz\_pca\_biplot(pca, repel = TRUE, col.var = "#2E9FDF",  
 col.ind = "#696969", select.ind = list(contrib = 10))



#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.

## Comprobación de la normalidad y la homogeneidad de la varianza variables numericas

Para la comprobación de que los valores que toman nuestra variables cuantativa provienen de una distirbución normal vamos a utilizar la prueba de normalidad de Anderson-Darling.

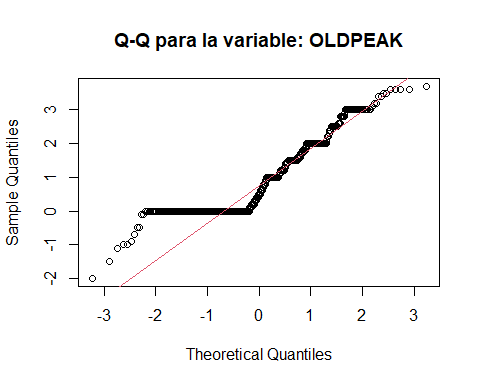
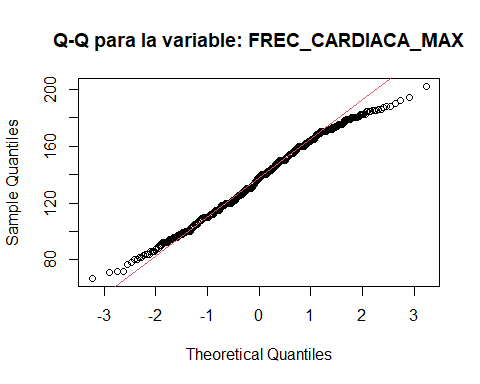
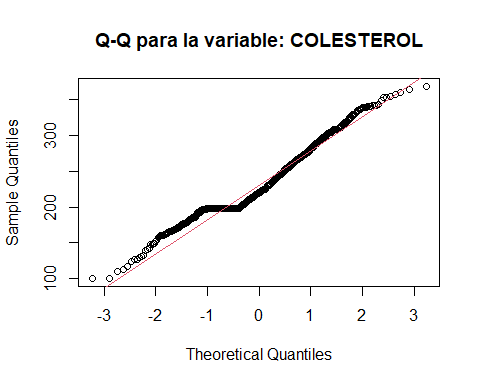
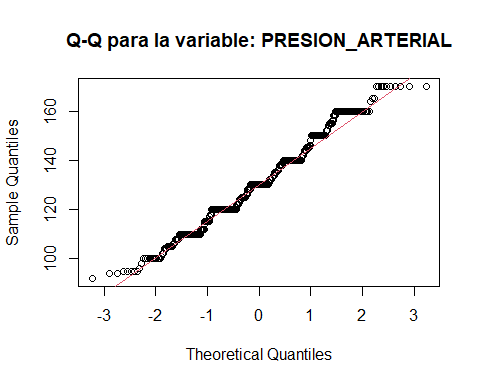
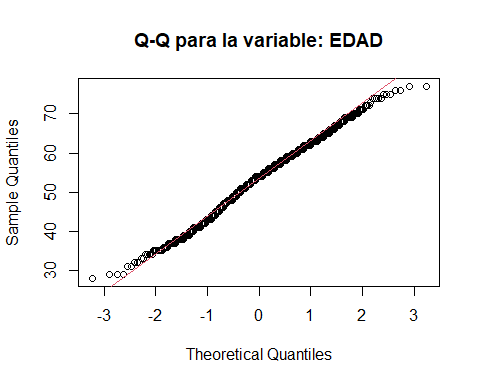
Podemos comprobar que para cada prueba se obtiene un p-valor superior al nivel de significancia estadistica prefijado en alpha = 0,05. Si esto se cumple, entonces se considera que variable en cuestion sigue la distribución normal.

if (!require('nortest')) install.packages('nortest'); library('nortest')  
alpha = 0.05  
col.names = colnames(datos\_final)  
ind = 1  
  
# Comprobanos unicamente lS variables que inicialmente eran de tipo numericas  
  
for (i in colnames(datos\_final)) {  
 if (ind == 1) cat("Variables que no siguen una distribución normal:\n")  
 if(vector\_tipos[ind] == "numeric")  
 {  
 p\_val = ad.test(unlist(datos\_final[i]))$p.value  
 if (p\_val < alpha) {  
 cat(i)  
 # Format output  
 if (ind < ncol(datos) - 1) cat(", ")  
 if (ind %% 3 == 0) cat("\n")  
 }  
 }  
 ind = ind + 1  
 }

## Variables que no siguen una distribución normal:  
## EDAD, PRESION\_ARTERIAL, COLESTEROL, NIVEL\_DE\_AZUCAR,   
## FREC\_CARDIACA\_MAX, OLDPEAK, E\_CARDIACA

Podemos realizar un Q-Q plot para comprobar si las variables obtenidas en el anterior punto no siguen una distribución normal.

variables <- c("EDAD", "PRESION\_ARTERIAL", "COLESTEROL",  
 "FREC\_CARDIACA\_MAX", "OLDPEAK")  
  
  
for(i in(variables))  
{  
 qqnorm(unlist(datos\_final[i]), main = paste0("Q-Q para la variable: ", i));qqline(unlist(datos\_final[i]), col = 2)  
}



Vemos que tanto la distribución de los valores de las caracteristicas de EDAD y de la Frecuencia Cardica máxima se acercan mucho a la normalidad, por otro lado la distribución de los valores de la caracteristica Presión Arterial, Colesterol y Old distan de la normal.

# Exportación de los datos

Una vez que hemos acometido sobre el conjunto de datos inicial los procedimientos de integración, validación y limpieza anteriores, procedemos a guardar estos en un nuevo fichero denominado heart\_dissease\_data\_clean.csv:

# Exportación de los datos limpios en .csv  
write.csv(datos\_final, "./heart\_dissease\_data\_clean.csv")

# Análisis de los datos

## ¿Cuáles son los factores o parámetros que más influyen a la hora de tener una enfermedad cardiaca? (Correlaciones y Regresión logística)

### Test de Spearman

corr\_matrix <- matrix(nc=2, nr=0)  
colnames(corr\_matrix) <- c("estimate", "p-value")  
  
# Calculamos el coficiente de correlacion para cada variable cuantitativa  
# con respecto al campo E. CARDICA  
  
for(i in 1:(ncol(datos\_final) -1 ))  
{  
 if(vector\_tipos[i] == "numeric")  
 {  
 spearman\_test = cor.test(unlist(datos\_final[,i]),  
 unlist(datos\_final[,length(datos\_final)]),  
 method = "spearman")  
 corr\_coef <- spearman\_test$estimate  
 p\_val <- spearman\_test$p.value  
   
 # Aañade a la matriz  
 pair <- matrix(ncol = 2, nrow = 1)  
 pair[1][1] = corr\_coef  
 pair[2][1] = p\_val  
 corr\_matrix <- rbind(corr\_matrix, pair)  
 rownames(corr\_matrix)[nrow(corr\_matrix)] <- colnames(datos\_final)[i]  
 }  
   
}  
  
print(corr\_matrix)

## estimate p-value  
## EDAD 0.32133836 5.839139e-21  
## PRESION\_ARTERIAL 0.11079517 1.566612e-03  
## COLESTEROL -0.06760546 5.414249e-02  
## NIVEL\_DE\_AZUCAR 0.28341365 1.825875e-16  
## FREC\_CARDIACA\_MAX -0.42471804 6.745047e-37  
## OLDPEAK 0.44538146 8.078414e-41

Podemos identificar cual es la variable más correlacionada con la variable Enfermedad Cardiaca, viendo cuales de los valores de la columan estimate se acercan más al valor +1 o -1, en este caso los más cercanos y por lo tanto los que más correlacionados están con la varibale objetivo son: OLDPEAK y FREC CARDÍACA MAX.

Por otro lado en la columna p-value, tenemos el indicador del peso estadistico de diga variable, en este caso las variables que tienen peso estadistico más alto son: COLESTEROL y PRESION ARTERIAL.

### Regresión logística

#### Generación de los conjuntos de entrenamiento y de test

set.seed(123)  
ind <- sample(seq\_len(nrow(datos\_final)), size = round(.8 \* dim(datos\_final)[1]))  
training <- datos\_final[ind, ]  
testing <- datos\_final[-ind, ]  
prop.table(table(datos\_final$E\_CARDIACA))

##   
## 0 1   
## 0.4507389 0.5492611

prop.table(table(training$E\_CARDIACA))

##   
## 0 1   
## 0.4569231 0.5430769

prop.table(table(testing$E\_CARDIACA))

##   
## 0 1   
## 0.4259259 0.5740741

#### Estimación del modelo con el conjunto de entrenamiento e interpretación

#Estimación del modelo  
mod1<- glm(E\_CARDIACA~.,data=training[,-1], family=binomial)  
summary(mod1)

##   
## Call:  
## glm(formula = E\_CARDIACA ~ ., family = binomial, data = training[,   
## -1])  
##   
## Deviance Residuals:   
## Min 1Q Median 3Q Max   
## -3.5113 -0.4332 0.1592 0.4636 2.4653   
##   
## Coefficients:  
## Estimate Std. Error z value Pr(>|z|)   
## (Intercept) -3.7391233 1.6052230 -2.329 0.019841 \*   
## SEXO 1.3672239 0.3170565 4.312 1.62e-05 \*\*\*  
## TIPO\_DOLOR\_TORAX 1.0365159 0.1755062 5.906 3.51e-09 \*\*\*  
## PRESION\_ARTERIAL 0.0010092 0.0081799 0.123 0.901809   
## COLESTEROL 0.0001034 0.0027892 0.037 0.970441   
## NIVEL\_DE\_AZUCAR 1.4315653 0.3213654 4.455 8.40e-06 \*\*\*  
## ECG\_EN\_REPOSO 0.1695613 0.1510045 1.123 0.261484   
## FREC\_CARDIACA\_MAX -0.0132493 0.0053744 -2.465 0.013691 \*   
## ANGINA\_x\_EJERCICIO 0.9887369 0.2889107 3.422 0.000621 \*\*\*  
## OLDPEAK 0.3031768 0.1508919 2.009 0.044513 \*   
## PENDIENTE\_ST 1.8443446 0.2498882 7.381 1.57e-13 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## (Dispersion parameter for binomial family taken to be 1)  
##   
## Null deviance: 896.26 on 649 degrees of freedom  
## Residual deviance: 442.58 on 639 degrees of freedom  
## AIC: 464.58  
##   
## Number of Fisher Scoring iterations: 5

Existe colinealidad en todas la variables menos en: FREC\_CARDIACA\_MAX, ANGINA\_x\_EJERCICIO, OLDPEAK y PENDIENTE\_ST

training2 = training %>%  
select(-SEXO,-TIPO\_DOLOR\_TORAX, -PRESION\_ARTERIAL, -COLESTEROL, -NIVEL\_DE\_AZUCAR, -ECG\_EN\_REPOSO)  
ModlgF<- glm(E\_CARDIACA~.,data=training2, family=binomial)  
summary(ModlgF)

##   
## Call:  
## glm(formula = E\_CARDIACA ~ ., family = binomial, data = training2)  
##   
## Deviance Residuals:   
## Min 1Q Median 3Q Max   
## -3.2307 -0.5666 0.2488 0.5422 2.2897   
##   
## Coefficients:  
## Estimate Std. Error z value Pr(>|z|)   
## (Intercept) -0.788651 1.161330 -0.679 0.497079   
## EDAD 0.029939 0.013153 2.276 0.022831 \*   
## FREC\_CARDIACA\_MAX -0.017371 0.005019 -3.461 0.000538 \*\*\*  
## ANGINA\_x\_EJERCICIO 1.242050 0.254904 4.873 1.1e-06 \*\*\*  
## OLDPEAK 0.351152 0.140126 2.506 0.012212 \*   
## PENDIENTE\_ST 1.878777 0.219396 8.563 < 2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## (Dispersion parameter for binomial family taken to be 1)  
##   
## Null deviance: 896.26 on 649 degrees of freedom  
## Residual deviance: 533.76 on 644 degrees of freedom  
## AIC: 545.76  
##   
## Number of Fisher Scoring iterations: 5

Se observa que todas las variables explicativas son significativas con un nivel de significación del 5%

#### Predicciones con con casos del dataframe

pred\_1 <-predict (ModlgF, datos\_final[1,], type = "response")  
cat("La probabilidad de que el primer usuario tenga una enfermedad cardiaca es del: ", pred\_1\*100)

## La probabilidad de que el primer usuario tenga una enfermedad cardiaca es del: 7.051109

pred\_14 <-predict (ModlgF, datos\_final[14,], type = "response")  
cat("La probabilidad de que el usuario 14 tenga una enfermedad cardiaca es del: ", pred\_14\*100)

## La probabilidad de que el usuario 14 tenga una enfermedad cardiaca es del: 84.79263

Nos damos cuenta de que la probabilidad obtenida es muy acertada a si esos usuarios han tenido o no enfermedad cardiaca.

## ¿Influye el sexo en tener una enfermedad cardiaca? (Contraste de hipótesis)

Vamos a realizar una prueba estadistica para establecer un contraste de hipótesis sobre dos muestras (una con presión arterial alta y otra con presión arterial normal) y ver cual de ellas tiene mayor probabilidades de sufrir enfermedad cardiaca.

### Hipótesis nula y la alternativa

H0 : Enf.Cardiaca\_Mujer = Enf.Cardiaca\_Hombre

H1 : Enf.Cardiaca\_Mujer != Enf.Cardiaca\_Hombre

### Preparación datos

hombres <- datos\_final[datos\_final$SEXO==1,]  
mujeres <- datos\_final[datos\_final$SEXO==0,]  
  
var.test( as.numeric(hombres$E\_CARDIACA), as.numeric(mujeres$E\_CARDIACA) )

##   
## F test to compare two variances  
##   
## data: as.numeric(hombres$E\_CARDIACA) and as.numeric(mujeres$E\_CARDIACA)  
## F = 1.2869, num df = 646, denom df = 164, p-value = 0.04968  
## alternative hypothesis: true ratio of variances is not equal to 1  
## 95 percent confidence interval:  
## 1.000355 1.626469  
## sample estimates:  
## ratio of variances   
## 1.286949

El resultado del test muestra diferencias significativas entre varianzas (p=0.04968). Por tanto, aplicaremos un test de dos muestras independientes sobre la media con varianzas desconocidas diferentes.

### Cálculo

t.test( as.numeric(datos\_final$SEXO), as.numeric(datos\_final$E\_CARDIACA), alternative="greater", conf.level=0.95)

##   
## Welch Two Sample t-test  
##   
## data: as.numeric(datos\_final$SEXO) and as.numeric(datos\_final$E\_CARDIACA)  
## t = 11.016, df = 1554, p-value < 2.2e-16  
## alternative hypothesis: true difference in means is greater than 0  
## 95 percent confidence interval:  
## 0.2105546 Inf  
## sample estimates:  
## mean of x mean of y   
## 0.7967980 0.5492611

### Interpretación del test

Existen diferencias significativas en tener enfermedad cardiaca entre los hombres y la mujeres (p=2.2e-16) por lo que se encuentra fuera de la zona de aceptación de la hipótesis nula, es decir, se acepta la hipótesis alternativa.

## Modelo Arbol de Decision

Podemos elaborar un arbol de decisión, para ver que variables tienen más influencia en la enfermedad cardiaca y establecer las reglas que definen dicha variable.

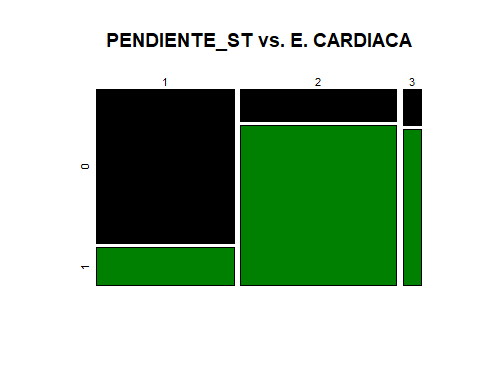
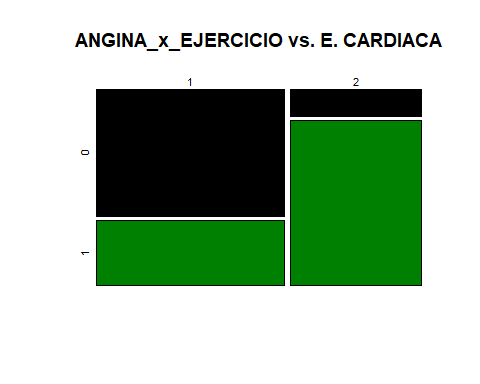
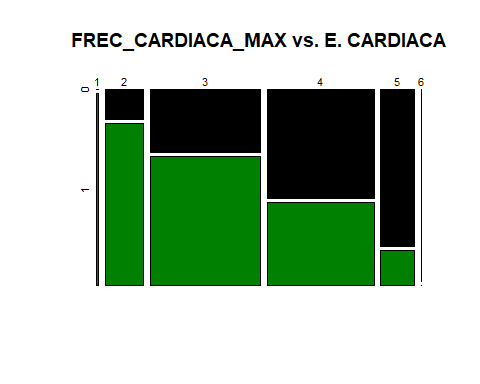
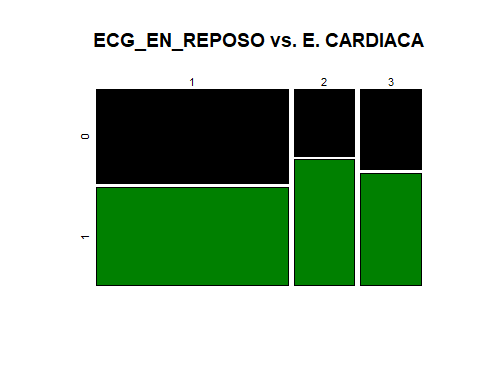
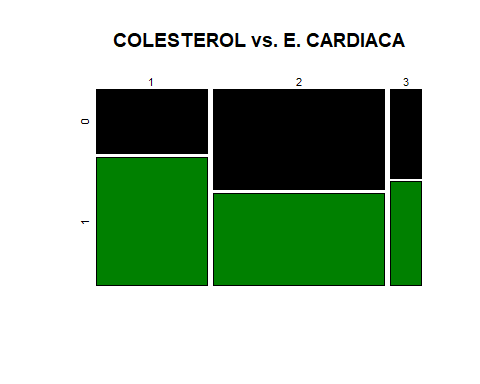
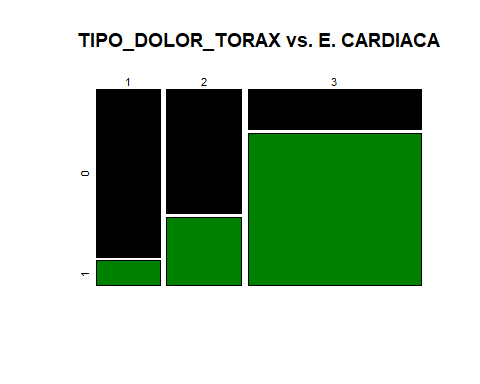
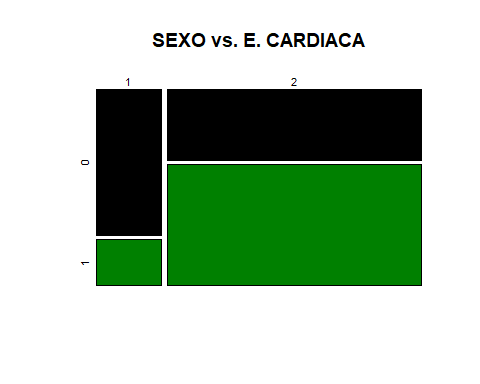
### Test estadísticos de significancia

Antes de proceder a la clasificación de los parámetros de pacientes con más probabilidades de sufrir enfermedad cardiaca, deberemos de hacer una selección previa de las caracteristicas a utilizar en nuestro modelo.

Para ello nos vamos a ayudar de una matriz de correlación con el objetivo de confirmar las conclusiones en cuanto a correlación de variables obtenidas en el apartado anterior.

Para aplicar los modelos de arbol de decision debemos de discretizar las variables COLESTEROL y FREC CARDIACA MAXIMA.

# Backup dataset inicial:  
datos\_final\_orig <- datos\_final  
  
datos\_final <- datos\_final %>% mutate(COLESTEROL = case\_when(  
 COLESTEROL < 100 ~ 0,  
 (COLESTEROL >= 100 & COLESTEROL < 200) ~ 1,  
 (COLESTEROL >= 200 & COLESTEROL < 300) ~ 2,  
 (COLESTEROL >= 300 & COLESTEROL < 400) ~ 3,  
 (COLESTEROL >= 400 & COLESTEROL < 500) ~ 4,  
 COLESTEROL >= 600 ~ 5,  
 ))  
  
  
datos\_final <- datos\_final %>% mutate(FREC\_CARDIACA\_MAX = case\_when(  
 FREC\_CARDIACA\_MAX < 50 ~ 0,  
 (FREC\_CARDIACA\_MAX >= 50 & FREC\_CARDIACA\_MAX < 80) ~ 1,  
 (FREC\_CARDIACA\_MAX >= 80 & FREC\_CARDIACA\_MAX < 110) ~ 2,  
 (FREC\_CARDIACA\_MAX >= 110 & FREC\_CARDIACA\_MAX < 140) ~ 3,  
 (FREC\_CARDIACA\_MAX >= 140 & FREC\_CARDIACA\_MAX < 170) ~ 4,  
 (FREC\_CARDIACA\_MAX >= 170 & FREC\_CARDIACA\_MAX < 200) ~ 5,  
 FREC\_CARDIACA\_MAX >= 200 ~ 6,  
 ))  
  
  
# Convertimos todas las variables a tipo factor  
datos\_final[] <- lapply(datos\_final, factor)  
  
# Analizamos las correlaciones de todos las caracteristicas de tipo categoricas con "E. CARDIACA"  
# Lo añadimos a una tabla  
  
datos\_corr.Phi <- list()  
datos\_corr.CramerV <- list()  
datos\_corr.nombre <- list()  
  
vector\_tipos[8] <- "character"  
vector\_tipos[5] <- "character"  
  
ind <- 1  
for (i in colnames(datos\_final))  
{  
   
 if(i != "E\_CARDIACA")  
 {  
 tabla\_cruzada <- table(as.numeric(unlist(datos\_final[i])), datos\_final$E\_CARDIACA)  
 datos\_corr.CramerV <- append(datos\_corr.CramerV, CramerV(tabla\_cruzada))  
 datos\_corr.Phi <- append(datos\_corr.Phi, Phi(tabla\_cruzada))  
 datos\_corr.nombre <- append(datos\_corr.nombre, i)  
 }  
   
 if (vector\_tipos[ind] != 'numeric')  
 {  
 # Solo pintamos las variables categoricas ya que con las de tipo numerico no se aprecian los valores  
 plot(tabla\_cruzada, col = c("black","#008000"), main = paste0(i, " vs. E. CARDIACA"))  
 }  
   
 ind <- ind + 1  
   
}



n\_list <- list(nombre=as.character(datos\_corr.nombre),  
 CamerV=as.numeric(datos\_corr.CramerV),  
 Phi=as.numeric(datos\_corr.Phi))  
  
df\_CramerV <- (as.data.frame(do.call(cbind, n\_list)))  
print(df\_CramerV[order(df\_CramerV$Phi, decreasing = TRUE),])

## nombre CamerV Phi  
## 11 PENDIENTE\_ST 0.633914979247456 0.633914979247456  
## 3 TIPO\_DOLOR\_TORAX 0.560309860170942 0.560309860170942  
## 10 OLDPEAK 0.520727759286187 0.520727759286187  
## 9 ANGINA\_x\_EJERCICIO 0.513768132124416 0.513768132124416  
## 1 EDAD 0.404380273871888 0.404380273871888  
## 8 FREC\_CARDIACA\_MAX 0.398095648097636 0.398095648097636  
## 4 PRESION\_ARTERIAL 0.331282489356957 0.331282489356957  
## 2 SEXO 0.317570321818606 0.317570321818606  
## 6 NIVEL\_DE\_AZUCAR 0.283413646865563 0.283413646865563  
## 5 COLESTEROL 0.17862988864165 0.17862988864165  
## 7 ECG\_EN\_REPOSO 0.115604279173622 0.115604279173622

Obtenemos las siguientes conclusiones del dataset:

* En cuanto al Sexo, las Mujeres tienen menos probabilidad de sufrir una enfermedad cardiaca.
* En cuanto al tipo de dolor de Torax, los pacientes que sufren de dolor tipo Asintomáticos son los que pese a lo que se pdoría pensar tienen más probabilidades de sufrir enfermedad cardiaca.
* COLESTEROL: Vemos que cuando el colesterol está en valores comprendidos entre 100-200 y 300-400 hay más posibilidades de enfermedad cardiaca que en valores entre 200-300, probablemente porque existan medicación para pacientes con dichas enfermedades que se enfocan en reducir el colesterol
* FREC CARDIACA MAXIMA hay correlación negativa con la enfermedad cardiaca es decir contra menor frecuencia más posibilidades de sufrir enfermedad cardiaca.
* En cuanto al ECG en Reposo, tenemos que hay más probabilidades de Enfermedad Cardiaca cuando esta variable toma el valor 1 (ST -> Tiene anormalidad de la onda ST-T (inversiones de la onda T y/o elevación o depresión del ST > 0.05 mV))
* Angina por Ejercicio -> Cuando esta toma el valor 1 (es decir hay angina inducida por ejercicio) hay más probabilidades de Enfermedad Cardiaca.
* En cuanto a la variable Pendiente ST, si esta indica valor 1 y 2, hay una alta probabilidad de sufrir enfermedad cardíaca.

### Prueba de la C de Crámer

Valores de la V de Cramér (<https://en.wikipedia.org/wiki/Cramér%27s_V>) y Phi (<https://en.wikipedia.org/wiki/Phi_coefficient>) entre 0.1 y 0.3 nos indican que la asociación estadística es baja, y entre 0.3 y 0.5 se puede considerar una asociación media. Finalmente, si los valores fueran superiores a 0.5 la asociación estadística entre las variables sería alta.

Podemos observar dentro del dataframe: df\_cramerV que las variables PENDIENTES ST, COLESTEROL, TIPO DOLOR TORAX, FREC CARDIACA MAX y OLDPEAK tienen correlación alta con E. CARDIACA.

Usaremos dichas variables para la obtención del Arbol de decisión, se podria utilizar un número mayor de variables, pero podría hacerse mucho complejo (con muchas reglas de decisión)

# caracteristicas con significancia estadistica:  
nombres\_columnas <- df\_CramerV$nombre[df\_CramerV$Phi > 0.5]  
nombres\_columnas

## [1] "TIPO\_DOLOR\_TORAX" "ANGINA\_x\_EJERCICIO" "OLDPEAK"   
## [4] "PENDIENTE\_ST"

### Aplicación del modelo Decision Tree (Arbol de decisión)

Aplicamos el modelo Decision Tree sobre las 3 caracteristicas que hemos obtenido en el test de significacia estadistica de Cramer V.

# Reducimos el datset  
for(i in colnames(datos\_final))  
{  
 if(!i %in% nombres\_columnas)  
 {  
 if (!i == "E\_CARDIACA")  
 {  
 datos\_final$i <- NULL  
 }  
 }  
}

Separamos conjunto de test y de entrenamiento con una proporción 33% Test 66% Training.

set.seed(666)  
y <- datos\_final$E\_CARDIACA # Variable objetivo  
X <- datos\_final[nombres\_columnas] # Variables predictoras  
  
split\_prop <- 3  
indexes = sample(1:nrow(X), size=floor(((split\_prop-1)/split\_prop)\*nrow(X)))  
train\_X <- X[indexes,]  
train\_y <- y[indexes]  
test\_X <- X[-indexes,]  
test\_y <- y[-indexes]

Comprobamos que las variables train\_y y test\_y estén balanceadas reflejo de la variable y.

print("y:")

## [1] "y:"

summary(y)

## 0 1   
## 366 446

print("train\_y:")

## [1] "train\_y:"

summary(train\_y)

## 0 1   
## 249 292

print("test\_y:")

## [1] "test\_y:"

summary(test\_y)

## 0 1   
## 117 154

Hacemos lo mismo con las variales train\_crX y test\_crX y X

print("X:")

## [1] "X:"

summary(X)

## TIPO\_DOLOR\_TORAX ANGINA\_x\_EJERCICIO OLDPEAK PENDIENTE\_ST  
## 1:166 0:479 0 :336 0:359   
## 2:195 1:333 1 : 78 1:405   
## 3:451 2 : 67 2: 48   
## 1.5 : 49   
## 3 : 27   
## 1.2 : 23   
## (Other):232

print("train\_X:")

## [1] "train\_X:"

summary(train\_X)

## TIPO\_DOLOR\_TORAX ANGINA\_x\_EJERCICIO OLDPEAK PENDIENTE\_ST  
## 1:114 0:327 0 :227 0:246   
## 2:129 1:214 1 : 49 1:260   
## 3:298 2 : 44 2: 35   
## 1.5 : 35   
## 3 : 18   
## 1.2 : 15   
## (Other):153

print("test\_X:")

## [1] "test\_X:"

summary(test\_X)

## TIPO\_DOLOR\_TORAX ANGINA\_x\_EJERCICIO OLDPEAK PENDIENTE\_ST  
## 1: 52 0:152 0 :109 0:113   
## 2: 66 1:119 1 : 29 1:145   
## 3:153 2 : 23 2: 13   
## 1.5 : 14   
## 3 : 9   
## 1.2 : 8   
## (Other): 79

Se crea el árbol de decisión usando los datos de entrenamiento (no hay que olvidar que la variable outcome es de tipo factor):

tree <- C50::C5.0(train\_X, train\_y, rules=TRUE )  
summary(tree)

##   
## Call:  
## C5.0.default(x = train\_X, y = train\_y, rules = TRUE)  
##   
##   
## C5.0 [Release 2.07 GPL Edition] Sat Jan 07 09:44:25 2023  
## -------------------------------  
##   
## Class specified by attribute `outcome'  
##   
## Read 541 cases (5 attributes) from undefined.data  
##   
## Rules:  
##   
## Rule 1: (160/7, lift 2.1)  
## TIPO\_DOLOR\_TORAX in {1, 2}  
## PENDIENTE\_ST = 0  
## -> class 0 [0.951]  
##   
## Rule 2: (193/20, lift 1.9)  
## OLDPEAK in {-1.1, -0.5, -0.1, 0, 0.2, 0.3, 0.4, 0.6, 0.7, 1.1, 1.2, 1.9,  
## 2.3, 3}  
## PENDIENTE\_ST = 0  
## -> class 0 [0.892]  
##   
## Rule 3: (140/15, lift 1.6)  
## TIPO\_DOLOR\_TORAX = 3  
## OLDPEAK in {-1, -0.7, 0.1, 0.5, 0.8, 0.9, 1, 1.4, 1.5, 1.6, 1.8, 2, 2.8}  
## -> class 1 [0.887]  
##   
## Rule 4: (295/50, lift 1.5)  
## PENDIENTE\_ST in {1, 2}  
## -> class 1 [0.828]  
##   
## Default class: 1  
##   
##   
## Evaluation on training data (541 cases):  
##   
## Rules   
## ----------------  
## No Errors  
##   
## 4 77(14.2%) <<  
##   
##   
## (a) (b) <-classified as  
## ---- ----  
## 193 56 (a): class 0  
## 21 271 (b): class 1  
##   
##   
## Attribute usage:  
##   
## 94.09% PENDIENTE\_ST  
## 61.55% OLDPEAK  
## 55.45% TIPO\_DOLOR\_TORAX  
##   
##   
## Time: 0.0 secs

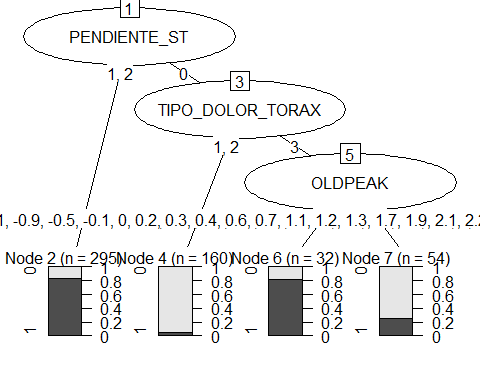
El modelo decision tree explica con dos reglas la probabilidad de sufrir una enfermedad cardiaca en función de las variables: TIPO DOLOR TORAX, COLESTEROL FREC CARDÍACA MÁX, OLDPEAK, PENDIENTE ST:

* Regla: 1 -> TIPO DE DOLOR DE TORAX con valores entre {1, 2} y PENDIENTE\_ST <- 0 No tienen probabilidad de sufrir enfermadad cardiaca.
* Regla: 2 -> OLDPEAK entre {-1.1 y 3} con PENDIENTE\_ST <- 0 No tienen probabilidad de sufrir Enfremedad Cardiaca.
* Regla: 3 -> TIPO\_DOLOR\_TORAX <- 3 y OLDPEAK entre {-1, 2.8} Tienen probabilidad de padecer enfermedad cardiaca.
* Regla 4 -> entre {1,2} -> Tienen probabilidad de tener enfermedad cardiaca.

El modelo solo usa la variable predictora PENDIENTE ST, y tiene una tasa de error de 14.2 % es decir es capaz de explicar el 82.6 % de los casos.

De manera más gráfica:

model <- C50::C5.0(train\_X, train\_y)  
plot(model)



Como podemos observar de manara visual el modelo basado en arbol de decisión, solo tiene en cuenta la variable “PENDIENTE\_ST”, para decidir entre si un paciente es propenso a sufrir una enfermedad cardiaca o no.

### Evaluación del modelo arbol de decision

Una vez tenemos el modelo, podemos comprobar su calidad prediciendo la clase para los datos de prueba que nos hemos reservado al principio.

predicted\_model <- predict( tree, test\_X, type="class" )  
print(sprintf("La precisión del árbol es: %.4f %%",100\*sum(predicted\_model == test\_y) / length(predicted\_model)))

## [1] "La precisión del árbol es: 83.7638 %"

Cuando hay pocas clases, la calidad de la predicción se puede analizar mediante una matriz de confusión que identifica los tipos de errores cometidos.

mat\_conf <- table(test\_y, Predicted=predicted\_model)  
mat\_conf

## Predicted  
## test\_y 0 1  
## 0 85 32  
## 1 12 142

De la matriz de confusión observamos los siguientes valores:

* Verdaderos Negativos (E. CARDIACA): 85
* Verdaderos Positivos (E. CARDIACA): 142
* Falsos Negativos (E. CARDIACA): 32
* Falsos Positivos (E. CARDIACA): 12

El modelo podria mejorarse sesgando a minimizar los falos negativos, ya que no queremos que se nos escapen del diagnostico pacientes que puedan desarrollar una enfermedad cardiaca.

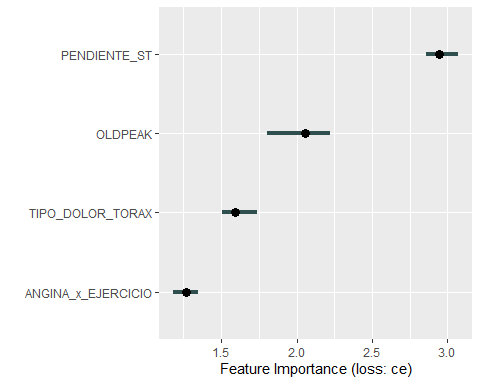
## Random Forest

Nos interesa saber para las predicciones que variable son las que tienen más influencia. Así, probaremos con un enfoque algorítmico de Random Forest y obtendremos métricas de interpretabilidad con la librería IML (<https://cran.r-project.org/web/packages/iml/iml.pdf>). As:

colnames(train\_X)

## [1] "TIPO\_DOLOR\_TORAX" "ANGINA\_x\_EJERCICIO" "OLDPEAK"   
## [4] "PENDIENTE\_ST"

train.data <- as.data.frame(cbind(train\_X ,train\_y))  
colnames(train.data)[5] <- "E\_CARDIACA"  
rf <- randomForest(E\_CARDIACA ~ ., data = train.data, ntree = 50)  
  
X <- train.data[which(names(train.data) != "E\_CARDIACA")]  
predictor <- Predictor$new(rf, data = X, y = train.data$E\_CARDIACA)   
imp <- FeatureImp$new(predictor, loss = "ce")  
plot(imp)

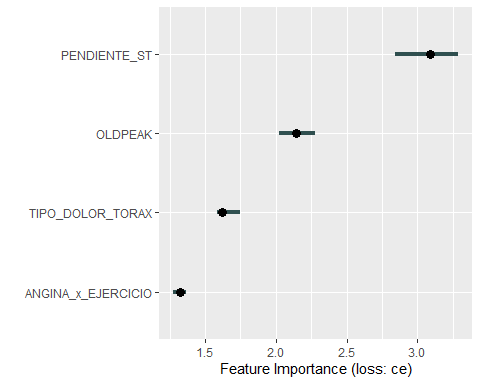


imp$results

## feature importance.05 importance importance.95 permutation.error  
## 1 PENDIENTE\_ST 2.860714 2.946429 3.071429 0.3049908  
## 2 OLDPEAK 1.807143 2.053571 2.225000 0.2125693  
## 3 TIPO\_DOLOR\_TORAX 1.503571 1.589286 1.739286 0.1645102  
## 4 ANGINA\_x\_EJERCICIO 1.182143 1.267857 1.346429 0.1312384

Podemos medir y graficar la importancia de cada variable para las predicciones del random forest con *FeatureImp*. La medida se basa funciones de pérdida de rendimiento que en nuestro caso será con el objetivo de clasificación (“ce”).

X <- train.data[which(names(train.data) != "E\_CARDIACA")]  
predictor <- Predictor$new(rf, data = X, y = train.data$E\_CARDIACA)   
imp <- FeatureImp$new(predictor, loss = "ce")  
plot(imp)



imp$results

## feature importance.05 importance importance.95 permutation.error  
## 1 PENDIENTE\_ST 2.843636 3.090909 3.287273 0.3142329  
## 2 OLDPEAK 2.021818 2.145455 2.276364 0.2181146  
## 3 TIPO\_DOLOR\_TORAX 1.581818 1.618182 1.749091 0.1645102  
## 4 ANGINA\_x\_EJERCICIO 1.276364 1.327273 1.363636 0.1349353

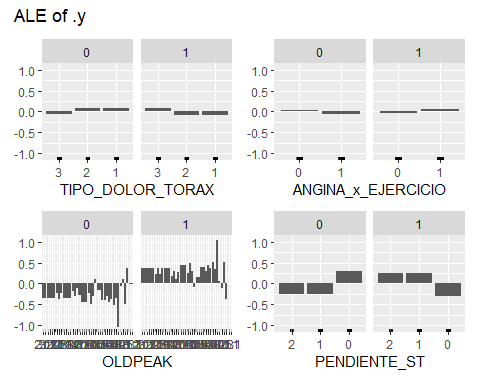
Precisión del modelo Random Forest

# Extraido de la matriz de confusion  
print(paste0("La precsión del modelo randomforest es: ",  
 (as.numeric(rf$confusion[1,][1]) / (as.numeric(rf$confusion[1,][1]) +  
 as.numeric(rf$confusion[1,][2]))) \* 100))

## [1] "La precsión del modelo randomforest es: 75.5020080321285"

Podemos observar el grado de importancia de las variables:

X <- train.data[which(names(train.data) != "E\_CARDIACA")]  
predictor\_cr <- Predictor$new(rf, data = X, y = train.data$E\_CARDIACA)   
  
  
effs <- FeatureEffects$new(predictor\_cr)  
plot(effs)



Parece ser que para el modelo de clasificación Random Forest la variable que toma mayor importancia es OLDPEAK.

Podemos verlo de manera textual:

rf$importance

## MeanDecreaseGini  
## TIPO\_DOLOR\_TORAX 42.92313  
## ANGINA\_x\_EJERCICIO 21.40851  
## OLDPEAK 47.57807  
## PENDIENTE\_ST 79.24776

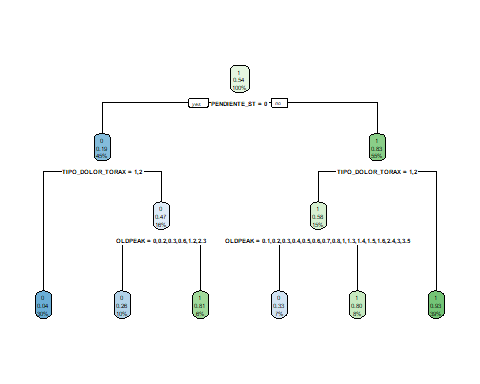
Otros modelos basados en arboles de decision:

tree\_2 <- rpart(formula = train\_y ~ ., data = train\_X)  
tree\_2

## n= 541   
##   
## node), split, n, loss, yval, (yprob)  
## \* denotes terminal node  
##   
## 1) root 541 249 1 (0.46025878 0.53974122)   
## 2) PENDIENTE\_ST=0 246 47 0 (0.80894309 0.19105691)   
## 4) TIPO\_DOLOR\_TORAX=1,2 160 7 0 (0.95625000 0.04375000) \*  
## 5) TIPO\_DOLOR\_TORAX=3 86 40 0 (0.53488372 0.46511628)   
## 10) OLDPEAK=0,0.2,0.3,0.6,1.2,2.3 54 14 0 (0.74074074 0.25925926) \*  
## 11) OLDPEAK=-1,-0.7,0.1,0.5,0.8,0.9,1,1.4,1.5,1.6,1.8,2,2.8 32 6 1 (0.18750000 0.81250000) \*  
## 3) PENDIENTE\_ST=1,2 295 50 1 (0.16949153 0.83050847)   
## 6) TIPO\_DOLOR\_TORAX=1,2 83 35 1 (0.42168675 0.57831325)   
## 12) OLDPEAK=0.1,0.2,0.3,0.4,0.5,0.6,0.7,0.8,1,1.3,1.4,1.5,1.6,2.4,3,3.5 39 13 0 (0.66666667 0.33333333) \*  
## 13) OLDPEAK=0,1.2,1.7,1.8,2,2.2,2.5,2.6,2.9,3.6 44 9 1 (0.20454545 0.79545455) \*  
## 7) TIPO\_DOLOR\_TORAX=3 212 15 1 (0.07075472 0.92924528) \*

Visualizamos el modelo:

rpart.plot(tree\_2)



El modelo inicialmente se fija en la variable PENDIENTE\_ST tanto como si es 0 como si es 1 se fija en TIPO\_DOLOR\_TORX (en cualquier caso se fija en los valores 1, 2 de dicha variable), después en funcion del rango de valores de OLDPEAK.

La probabilidades de sifrir enfermedad cardiaca son menores con PENDIENTE\_ST=0, tanto en el caso en el que TIPO\_DOLOR\_TORAX esté en rango de valores de {1,2} al igual de que TIPO\_DOLOR\_TORAX no este en dicho rango, si se cumple que OLDPEAK esté en el rango {0-2.3}. En el otro caso habrá probabilidad alta de enfermedad cardiaca.

Para el caso de PENDIENTE\_ST distinto a 0, y TIPO\_DOLOR\_TORAX en rango {1,2} si OLDPEAK está comprendido en rango {0-3.5} habrá pocas probabilidades de ENFERMEDAD CARDIACA, para los demás casos habrá altas probabilidades de enfermedad cardiaca.

# Conclusiones