DataCleaningHeartAttack

Maria Blanco González-Mohíno y Eloisa Baquero Candau

2024-01-09

Contents

Descripción del dataset

El conjunto de datos de "DataHeartAttack" extraido de Kaggle es un subconjunto de datos extraido de la base de datos de Cleveland. Esta base de datos cuenta con 76 atributos, pero todos los experimentos publicados hacen referencia al uso de un subconjunto de 14 de ellos, con los que se trabajará a lo largo de este proyecto.

A traves de estes conjunto de datos se puedes estudiar los factores que influyen en la posibilidad de sufrir infarto de corazón con la idea de poder anticiparse y poder envitarlo en alguna medida.

Fuentes de datos

Fuente de este dataset: https://www.kaggle.com/datasets/rashikrahmanpritom/heart-attack-analysis-prediction-dataset/data Fuente de Kaggle: https://archive.ics.uci.edu/dataset/45/heart+disease

El conjunto de datos heart.csv proviene de la base de datos disponible en la plataforma Kaggle.

¿Por qué es importante?

Las enfermedades cardiovasculares (ECV) son la principal causa de muerte a nivel mundial y se estima que cobran 17,9 millones de vidas cada año, lo que representa el 31% de todas las muertes en todo el mundo. Cuatro de cada cinco muertes por ECV se deben a ataques cardíacos y accidentes cerebrovasculares, y un tercio de estas muertes ocurren prematuramente en personas menores de 70 años. La insuficiencia cardíaca es un evento común causado por enfermedades cardiovasculares y este conjunto de datos contiene 11 características que pueden usarse para predecir una posible enfermedad cardíaca.

Las personas con enfermedades cardiovasculares o que tienen un alto riesgo cardiovascular (debido a la presencia de uno o más factores de riesgo como hipertensión, diabetes, hiperlipidemia o enfermedad ya establecida) necesitan una detección y un manejo tempranos en los que un modelo de aprendizaje automático puede ser de gran ayuda.

¿Qué pregunta responde?

Este conjunto de datos pretende elaborar un modelo predictivo y responder a la pregunta: ¿qué factores son clave para la aparición de ataques cardíacos?

Variables del dataset

- **Age**: Edad del paciente.
- **Sex**: Sexo del paciente.
- **cp**: Dolor torácico tipo (1 = angina típica; 2 = angina atípica; 3 = dolor no anginoso; 4 = asintomático).
- **ca**: Número de grandes vasos sanguíneos (0-3).
- **trtbps**: Presión arterial en reposo (en mm Hg).
- **chol**: colesterol en mg/dl obtenido a través del sensor de IMC.
- **fbs**: (glucemia en ayunas > 120 mg/dl) (1 = verdadero; 0 = falso).
- **rest_ecg**: resultados electrocardiográficos en reposo (0 = normal; 1 = presenta anomalía de la onda ST-T (inversiones de la onda T y/o elevación o depresión del ST de > 0,05 mV); 2 = muestra hipertrofia ventricular izquierda probable o definida según los criterios de Estes).
- **thalach**: frecuencia cardiaca máxima alcanzada
- **exng**: Angina inducida por el ejercicio (1 = si; 0 = no).
- **oldpeak**: Depresión del ST inducida por el ejercicio en relación con el reposo.
- **slp**: Slope / Bajada.
- **caa**: Número de vasos mayores (0-3) coloreados por flouroscopia.
- **thall**: No hay información al respecto.
- **output**: 0 = menor probabilidad de infarto; 1 = mayor probabilidad de infarto

Exploración de los datos

Carga del archivo

```
heart <- read_csv("../data/raw/heart.csv", show_col_types = FALSE)
head(heart, 5)
```

```
## # A tibble: 5 x 14
##     age   sex    cp trtbps  chol   fbs restecg thalachh  exng oldpeak   slp
##   <dbl> <dbl> <dbl>  <dbl> <dbl> <dbl>   <dbl>    <dbl> <dbl>   <dbl> <dbl>
## 1    63     1     3    145   233     1       0      150     0     2.3     0
## 2    37     1     2    130   250     0       1      187     0     3.5     0
## 3    41     0     1    130   204     0       0      172     0     1.4     2
## 4    56     1     1    120   236     0       1      178     0     0.8     2
## 5    57     0     0    120   354     0       1      163     1     0.6     2
## # i 3 more variables: caa <dbl>, thall <dbl>, output <dbl>
```

Estructura de los datos

```
str(heart)
```

```
## spc_tbl_ [303 x 14] (S3: spec_tbl_df/tbl_df/tbl/data.frame)
##  $ age     : num [1:303] 63 37 41 56 57 57 56 44 52 57 ...
##  $ sex     : num [1:303] 1 1 0 1 0 1 0 1 1 1 ...
##  $ cp      : num [1:303] 3 2 1 1 0 0 1 1 2 2 ...
##  $ trtbps  : num [1:303] 145 130 130 120 120 140 140 120 172 150 ...
##  $ chol    : num [1:303] 233 250 204 236 354 192 294 263 199 168 ...
##  $ fbs     : num [1:303] 1 0 0 0 0 0 0 0 1 0 ...
##  $ restecg : num [1:303] 0 1 0 1 1 1 0 1 1 1 ...
##  $ thalachh: num [1:303] 150 187 172 178 163 148 153 173 162 174 ...
##  $ exng    : num [1:303] 0 0 0 0 1 0 0 0 0 0 ...
##  $ oldpeak : num [1:303] 2.3 3.5 1.4 0.8 0.6 0.4 1.3 0 0.5 1.6 ...
##  $ slp     : num [1:303] 0 0 2 2 2 1 1 2 2 2 ...
##  $ caa     : num [1:303] 0 0 0 0 0 0 0 0 0 0 ...
##  $ thall   : num [1:303] 1 2 2 2 2 1 2 3 3 2 ...
##  $ output  : num [1:303] 1 1 1 1 1 1 1 1 1 1 ...
##  - attr(*, "spec")=
##   .. cols(
##   ..   age = col_double(),
##   ..   sex = col_double(),
##   ..   cp = col_double(),
##   ..   trtbps = col_double(),
##   ..   chol = col_double(),
##   ..   fbs = col_double(),
##   ..   restecg = col_double(),
##   ..   thalachh = col_double(),
##   ..   exng = col_double(),
##   ..   oldpeak = col_double(),
##   ..   slp = col_double(),
##   ..   caa = col_double(),
```

```
##   ..   thall = col_double(),
##   ..   output = col_double()
##   .. )
##  - attr(*, "problems")=<externalptr>
```

```
summary(heart)
```

```
##       age             sex               cp            trtbps
##  Min.   :29.00   Min.   :0.0000   Min.   :0.000   Min.   : 94.0
##  1st Qu.:47.50   1st Qu.:0.0000   1st Qu.:0.000   1st Qu.:120.0
##  Median :55.00   Median :1.0000   Median :1.000   Median :130.0
##  Mean   :54.37   Mean   :0.6832   Mean   :0.967   Mean   :131.6
##  3rd Qu.:61.00   3rd Qu.:1.0000   3rd Qu.:2.000   3rd Qu.:140.0
##  Max.   :77.00   Max.   :1.0000   Max.   :3.000   Max.   :200.0
##       chol            fbs            restecg          thalachh
##  Min.   :126.0   Min.   :0.0000   Min.   :0.0000   Min.   : 71.0
##  1st Qu.:211.0   1st Qu.:0.0000   1st Qu.:0.0000   1st Qu.:133.5
##  Median :240.0   Median :0.0000   Median :1.0000   Median :153.0
##  Mean   :246.3   Mean   :0.1485   Mean   :0.5281   Mean   :149.6
##  3rd Qu.:274.5   3rd Qu.:0.0000   3rd Qu.:1.0000   3rd Qu.:166.0
##  Max.   :564.0   Max.   :1.0000   Max.   :2.0000   Max.   :202.0
##       exng          oldpeak          slp              caa
##  Min.   :0.0000   Min.   :0.00   Min.   :0.000   Min.   :0.0000
##  1st Qu.:0.0000   1st Qu.:0.00   1st Qu.:1.000   1st Qu.:0.0000
##  Median :0.0000   Median :0.80   Median :1.000   Median :0.0000
##  Mean   :0.3267   Mean   :1.04   Mean   :1.399   Mean   :0.7294
##  3rd Qu.:1.0000   3rd Qu.:1.60   3rd Qu.:2.000   3rd Qu.:1.0000
##  Max.   :1.0000   Max.   :6.20   Max.   :2.000   Max.   :4.0000
##      thall           output
##  Min.   :0.000   Min.   :0.0000
##  1st Qu.:2.000   1st Qu.:0.0000
##  Median :2.000   Median :1.0000
##  Mean   :2.314   Mean   :0.5446
##  3rd Qu.:3.000   3rd Qu.:1.0000
##  Max.   :3.000   Max.   :1.0000
```

```
skim(heart)
```

Table 1: Data summary

Name	heart
Number of rows	303
Number of columns	14
Column type frequency:	
numeric	14
Group variables	None

Variable type: numeric

skim_variable	n_missing	complete_rate	mean	sd	p0	p25	p50	p75	p100	hist
age	0	1	54.37	9.08	29	47.5	55.0	61.0	77.0	
sex	0	1	0.68	0.47	0	0.0	1.0	1.0	1.0	
cp	0	1	0.97	1.03	0	0.0	1.0	2.0	3.0	
trtbps	0	1	131.62	17.54	94	120.0	130.0	140.0	200.0	
chol	0	1	246.26	51.83	126	211.0	240.0	274.5	564.0	
fbs	0	1	0.15	0.36	0	0.0	0.0	0.0	1.0	
restecg	0	1	0.53	0.53	0	0.0	1.0	1.0	2.0	
thalachh	0	1	149.65	22.91	71	133.5	153.0	166.0	202.0	
exng	0	1	0.33	0.47	0	0.0	0.0	1.0	1.0	
oldpeak	0	1	1.04	1.16	0	0.0	0.8	1.6	6.2	
slp	0	1	1.40	0.62	0	1.0	1.0	2.0	2.0	
caa	0	1	0.73	1.02	0	0.0	0.0	1.0	4.0	
thall	0	1	2.31	0.61	0	2.0	2.0	3.0	3.0	
output	0	1	0.54	0.50	0	0.0	1.0	1.0	1.0	

Limpieza de datos

Búsqueda de valores nulos o no válidos

Enfocandonos en la naturaleza del dataset, hay algunos casos especiales que también se consideraran valores nulos. La variable "caa" puede tener valores entre [0-3], el resto serán considerados nulos o no válidos.

```
cat("Elementos no válidos en la columna caa:", sum(heart$caa < 0 | heart$caa >3))
```

```
## Elementos no válidos en la columna caa: 5
```

Se observan 5 valores nulos en esta columna. Debido a que no es un número elevado se considera la eliminación de esas filas.

```
heart <- heart[!(heart$caa < 0 | heart$caa > 3), ]
dim(heart)
```

```
## [1] 298  14
```

La variable thall, debería comprender valores entre 1 y 3.

```
numero_apariciones <- table(heart$thall)
heart <- heart[!(heart$thall < 1 | heart$thall > 3), ]
dim(heart)
```

```
## [1] 296  14
```

Outliers

Un diagrama de caja (o diagrama de caja y bigotes) muestra la distribución de datos cuantitativos de una manera que facilita las comparaciones entre variables. El cuadro muestra los cuartiles del conjunto de datos mientras que los bigotes se extienden para mostrar el resto de la distribución. (también conocido como diagrama de caja y bigotes) es una forma estandarizada de mostrar la distribución de datos basada en el resumen de cinco números:

- Mínimo
- Primer cuartil
- Mediana
- Tercer cuartil
- Máximo

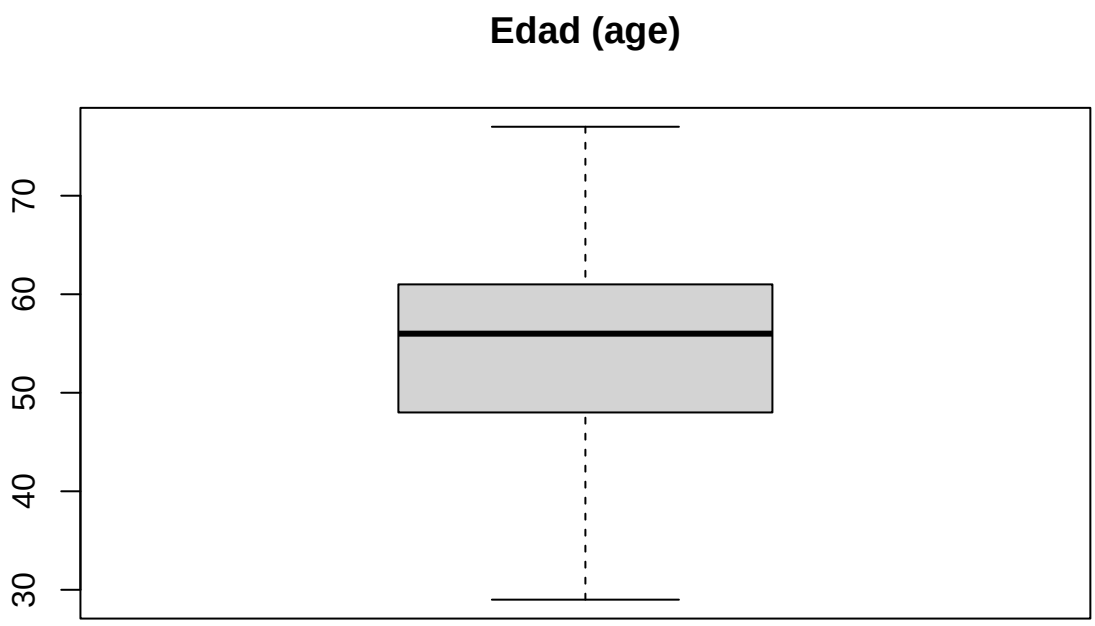
En el diagrama de caja más simple, el rectángulo central abarca desde el primer cuartil hasta el tercer cuartil (el rango intercuartil o IQR). Un segmento dentro del rectángulo muestra la mediana y los "bigotes" encima y debajo del cuadro muestran las ubicaciones del mínimo y el máximo.

- **Edad (age)**

```
summary(heart$age)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   29.00   48.00   56.00   54.52   61.00   77.00
```

```
boxplot(heart$age, main="Edad (age)")
```



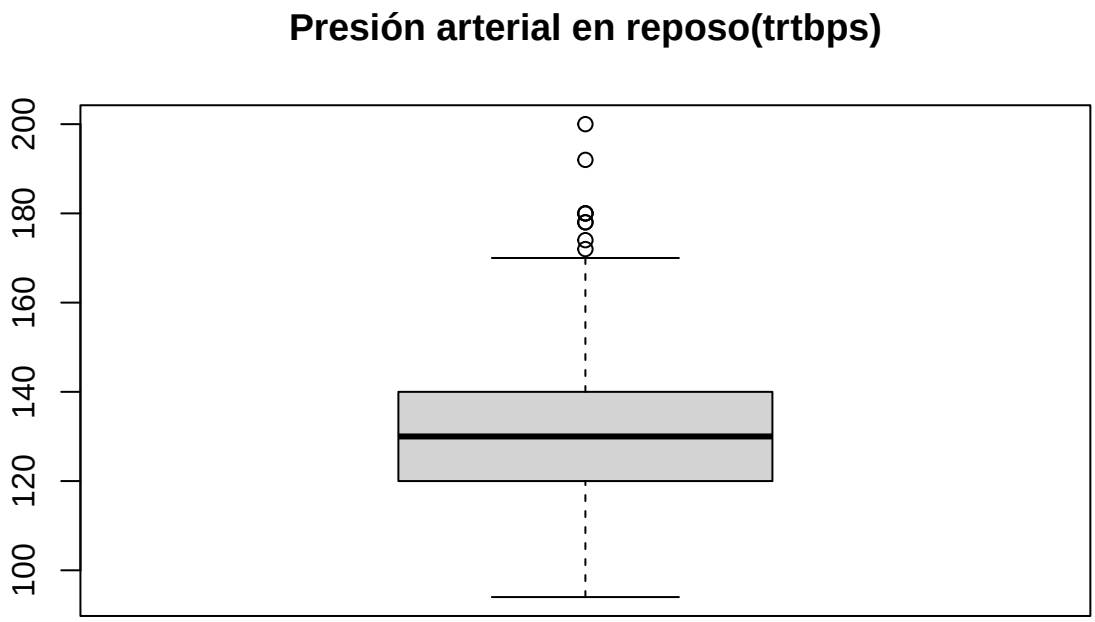
En este gráfico se muestra que la variable edad no contiene valores atípicos ya que todos los valores se encuentrar en el rango de 29 - 77 como valores minimo y máximo. La edad media es de 55 años.

- **Presión arterial en reposo (trtbps)**

```r
summary(heart$trtbps)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##    94.0   120.0   130.0   131.6   140.0   200.0
```

```r
boxplot(heart$trtbps, main="Presión arterial en reposo(trtbps)")
```



En el caso de la variable trtbps encontramos valores atípico con presiones arteriales en reposo por encima de 170 llegando a encontrar un valor máximo de 200 mm Hg. No se encuentran valores extremos en la cola inferior. Los valores atípicos son:

```r
x <- boxplot.stats(heart$trtbps)$out
idx <- which(heart$trtbps %in% x)
sort(heart$trtbps[idx])
```

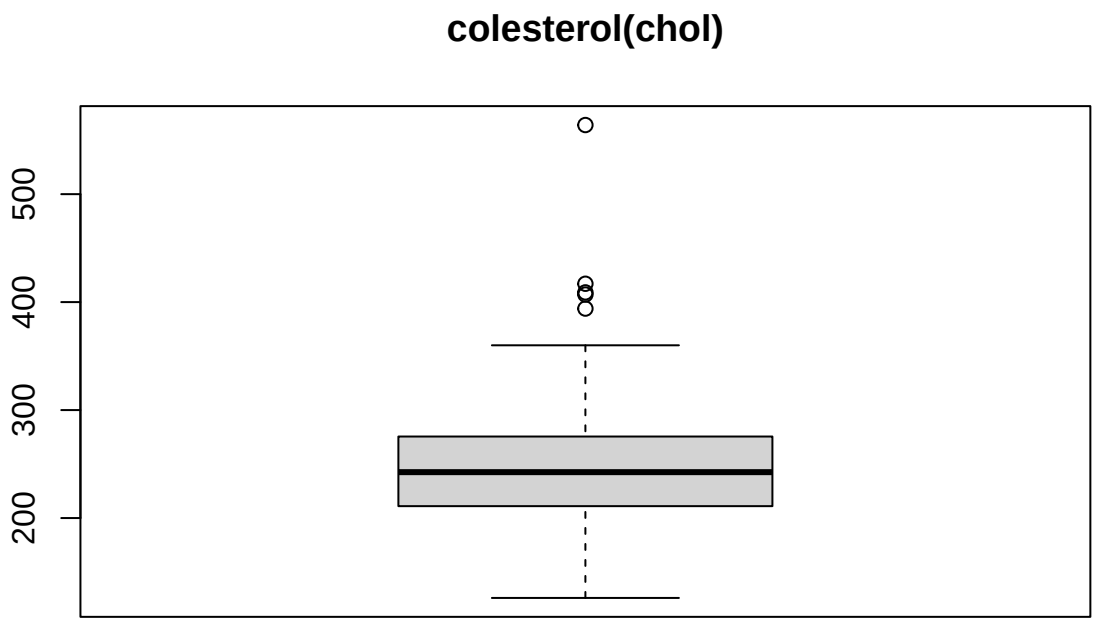
```
## [1] 172 174 178 178 180 180 180 192 200
```

- **colesterol (chol)**

```r
summary(heart$chol)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   126.0   211.0   242.5   247.2   275.2   564.0
```

```r
a <- boxplot(heart$chol, main="colesterol(chol)")
```




```r
x <- boxplot.stats(heart$chol)$out
idx <- which(heart$chol %in% x)
sort(heart$chol[idx])
```

```
## [1] 394 407 409 417 564
```

La variables colesterol (chol) presenta 4 valores extremos superiores a 360 destacando un valor máximo muy por encima de 360 que es 564. Este valor es mucho mayor que los demás valores atípicos encontrados en esta variable y se debería eliminar.

```r
heart$chol[idx]
```

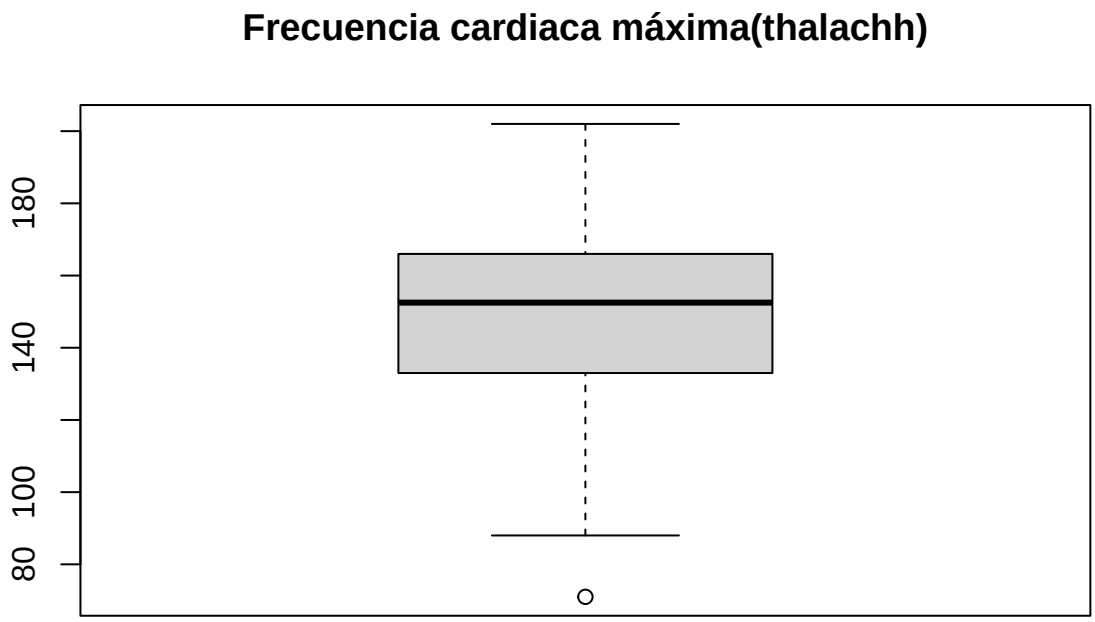
```
## [1] 417 564 394 407 409
```

- **Frecuencia cardiaca máxima (thalachh)**

```r
summary(heart$thalachh)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##    71.0   133.0   152.5   149.6   166.0   202.0
```

```
a <- boxplot(heart$thalachh, main="Frecuencia cardiaca máxima(thalachh)")
```




```
x <- boxplot.stats(heart$thalachh)$out
idx <- which(heart$thalachh %in% x)
sort(heart$thalachh[idx])
```

```
## [1] 71
```

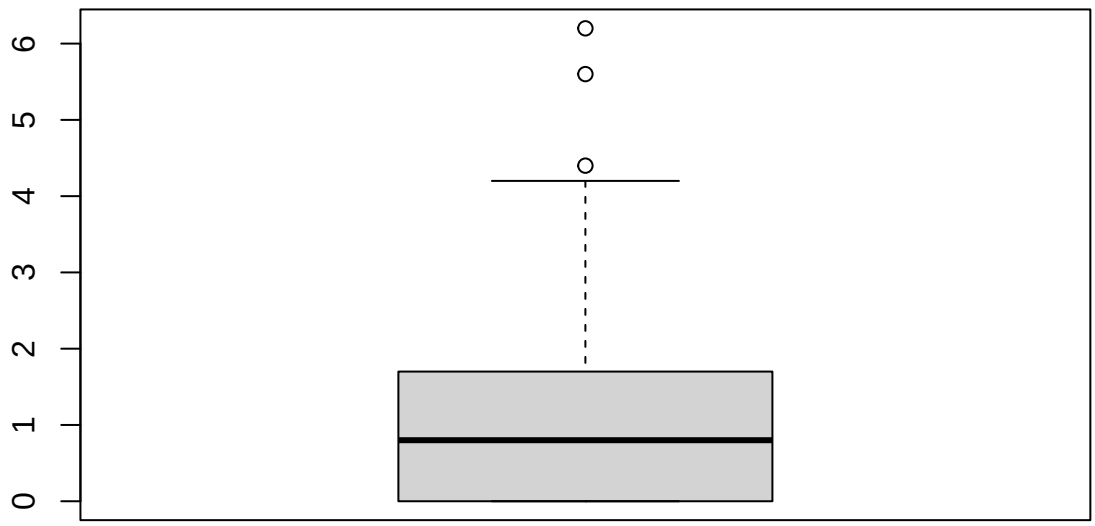
Para la frecuencia cardiaca máxima solo encontramos un valor extremo en la parte inferior con un valor de 71. Al tratarse de un único valor se debería eliminar.

- **Depresión del ST(oldpeak)**

```
summary(heart$oldpeak)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   0.000   0.000   0.800   1.059   1.650   6.200
```

```
a <- boxplot(heart$oldpeak, main="Depresión del ST inducida por ejercicio(oldpeak)")
```




```
x <- boxplot.stats(heart$oldpeak)$out
idx <- which(heart$oldpeak %in% x)
sort(heart$oldpeak[idx])
```

```
## [1] 4.4 5.6 6.2
```

Esta variable presenta 5 valores extremos ya que están por encima del valor 4 siendo el mayor valor de 6,2.

Visualización de variables de interes

- **Conversión de datos (Será útil en la visualización de datos).**

```
data<- heart %>%
   mutate(sex = if_else(sex ==1,"MALE","FEMALE"),
          fbs = if_else(fbs ==1,">120","<=120"),
          exng = if_else(exng ==1,"YES","NO"),
          cp = if_else(cp==1,"ATYPICAL ANGINA",
                       if_else(cp==2,"NON-ANGINAL PAIN","ASYMPTOTIC")),
          restecg = if_else(restecg==0,"Normal",
                            if_else(restecg==1,"ABNORMALITY","PROBABLE OF DEFINITE")),
          slp = as.factor(slp),
          caa = as.factor(caa),
          thall = as.factor(thall),
          output = if_else(output==1,"YES","NO")
```

```
        ) %>%
mutate_if(is.character,as.factor)%>%
dplyr::select(output,sex,fbs,exng,cp,restecg,slp,caa,thall, everything())
```

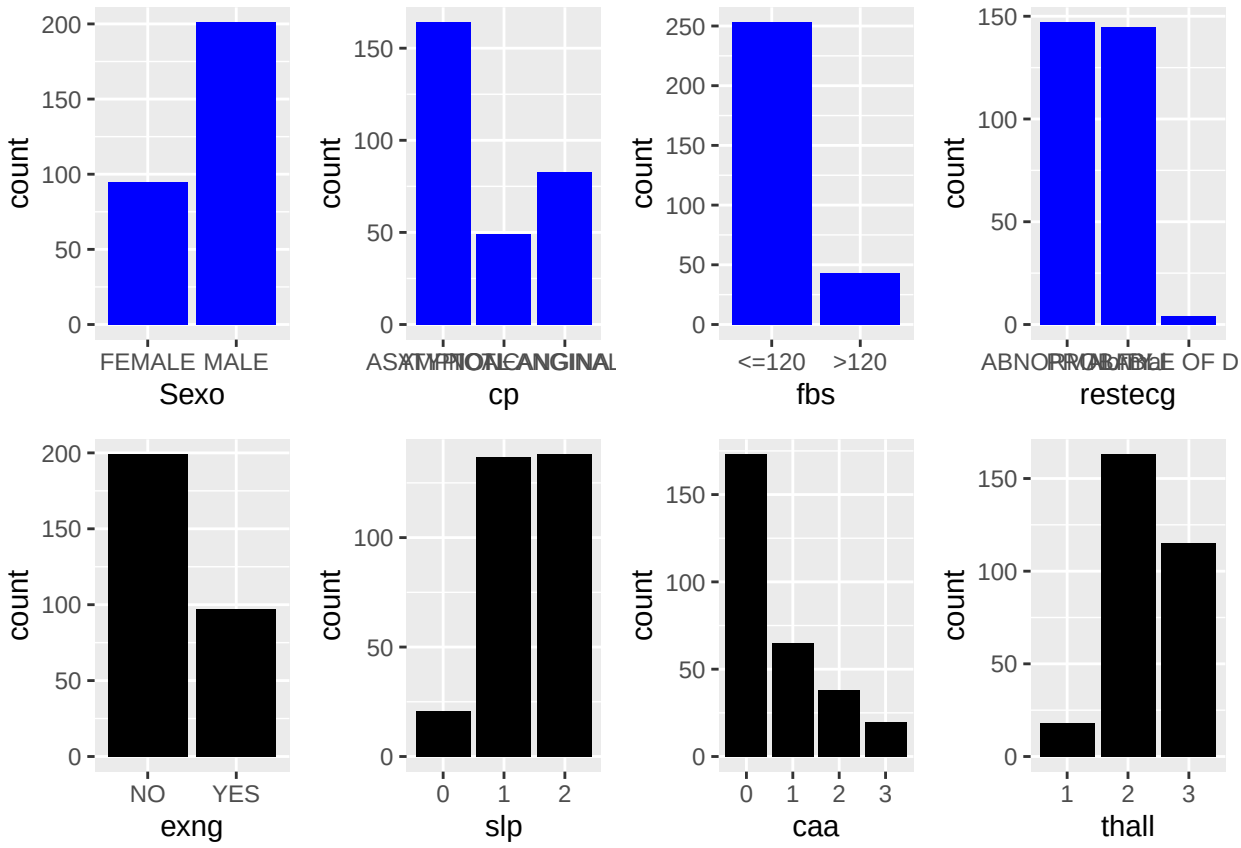
```
# Convertimos el target a factor
heart <- heart %>%
    mutate(output = as.factor(output))
```

Análisis univariante

- **Variables categóricas**

Vamos a estudiar la distribución de las variables categóricas mediantes diagramas de barras:

```
bp1 <- ggplot(data,aes(x=factor(sex)))+geom_bar(fill = "blue")+xlab("Sexo")
bp2 <- ggplot(data,aes(x=factor(cp)))+geom_bar(fill = "blue")+xlab("cp")
bp3 <- ggplot(data,aes(x=factor(fbs)))+geom_bar(fill = "blue")+xlab("fbs")
bp4 <- ggplot(data,aes(x=factor(restecg)))+geom_bar(fill = "blue")+xlab("restecg")
bp5 <- ggplot(data,aes(x=factor(exng)))+geom_bar(fill = "black")+xlab("exng")
bp6 <- ggplot(data,aes(x=factor(slp)))+geom_bar(fill = "black")+xlab("slp")
bp7 <- ggplot(data,aes(x=factor(caa)))+geom_bar(fill = "black")+xlab("caa")
bp8 <- ggplot(data,aes(x=factor(thall)))+geom_bar(fill = "black")+xlab("thall")
grid.arrange(bp1,bp2,bp3,bp4,bp5,bp6,bp7,bp8, nrow=2)
```



A continuación calculamos la frecuencia de cada variable categórica.

```r
prop.table(table(data$sex))
```

```
##
##    FEMALE      MALE
## 0.3209459 0.6790541
```

```r
prop.table(table(data$fbs))
```

```
##
##     <=120      >120
## 0.8547297 0.1452703
```

```r
prop.table(table(data$exng))
```

```
##
##        NO       YES
## 0.6722973 0.3277027
```

```r
prop.table(table(data$sex))
```

```
##
##    FEMALE      MALE
## 0.3209459 0.6790541
```

```r
prop.table(table(data$cp))
```

```
##
##       ASYMPTOTIC  ATYPICAL ANGINA NON-ANGINAL PAIN
##        0.5540541        0.1655405        0.2804054
```

```r
prop.table(table(data$restecg))
```

```
##
##          ABNORMALITY               Normal PROBABLE OF DEFINITE
##           0.49662162          0.48986486           0.01351351
```

```r
prop.table(table(data$slp))
```

```
##
##          0          1          2
## 0.07094595 0.46283784 0.46621622
```

```r
prop.table(table(data$cca))
```

```
## Warning: Unknown or uninitialised column: `cca`.
```

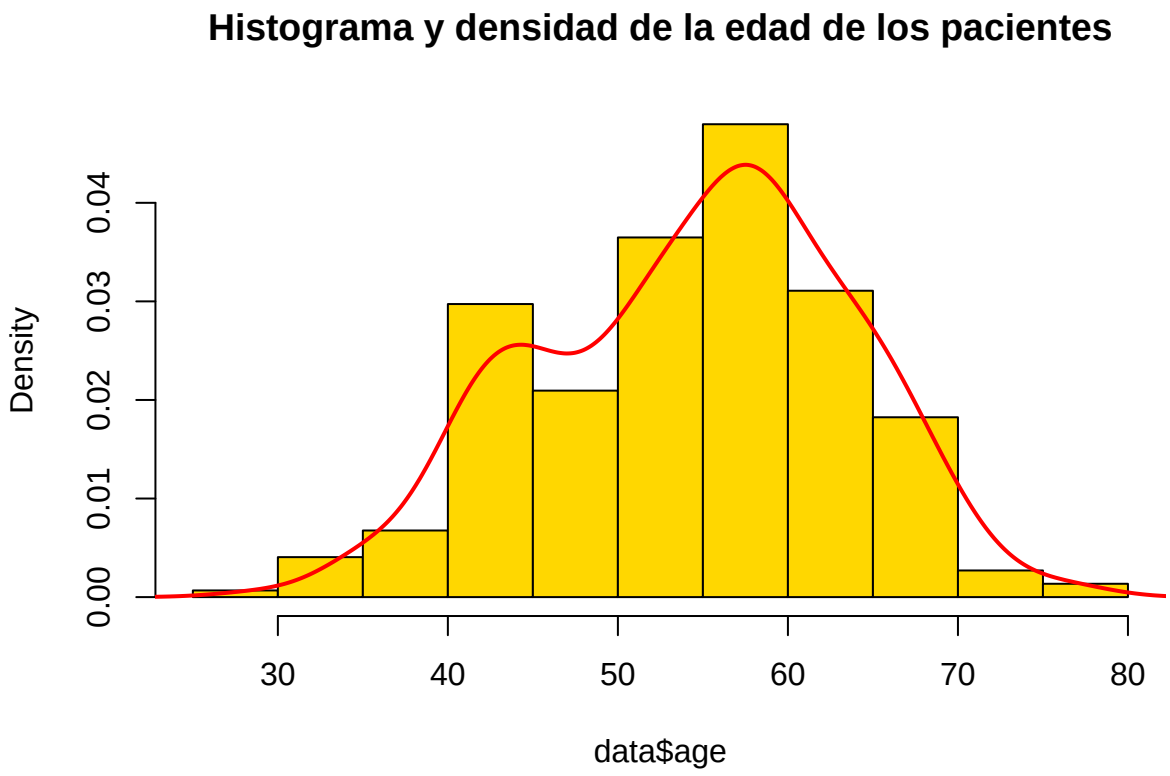
```
## numeric(0)
```

```r
prop.table(table(data$thall))
```

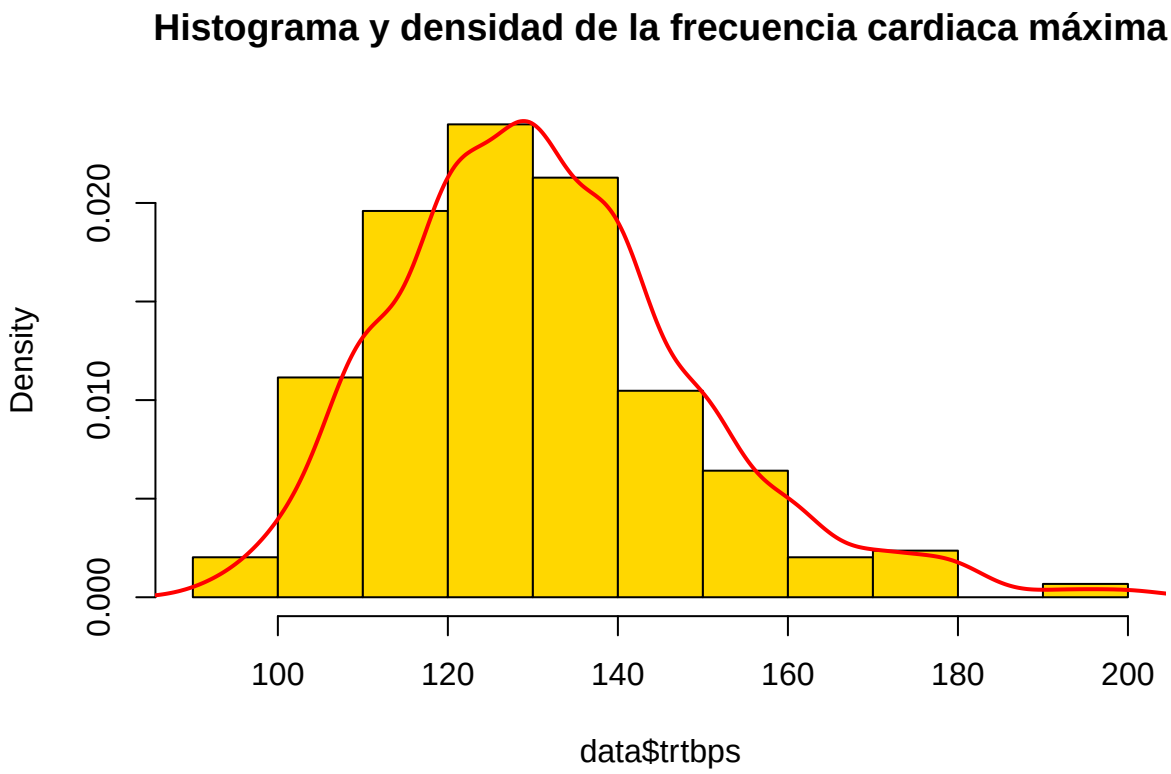
```
##
##          1          2          3
## 0.06081081 0.55067568 0.38851351
```

- **Variables numéricas**

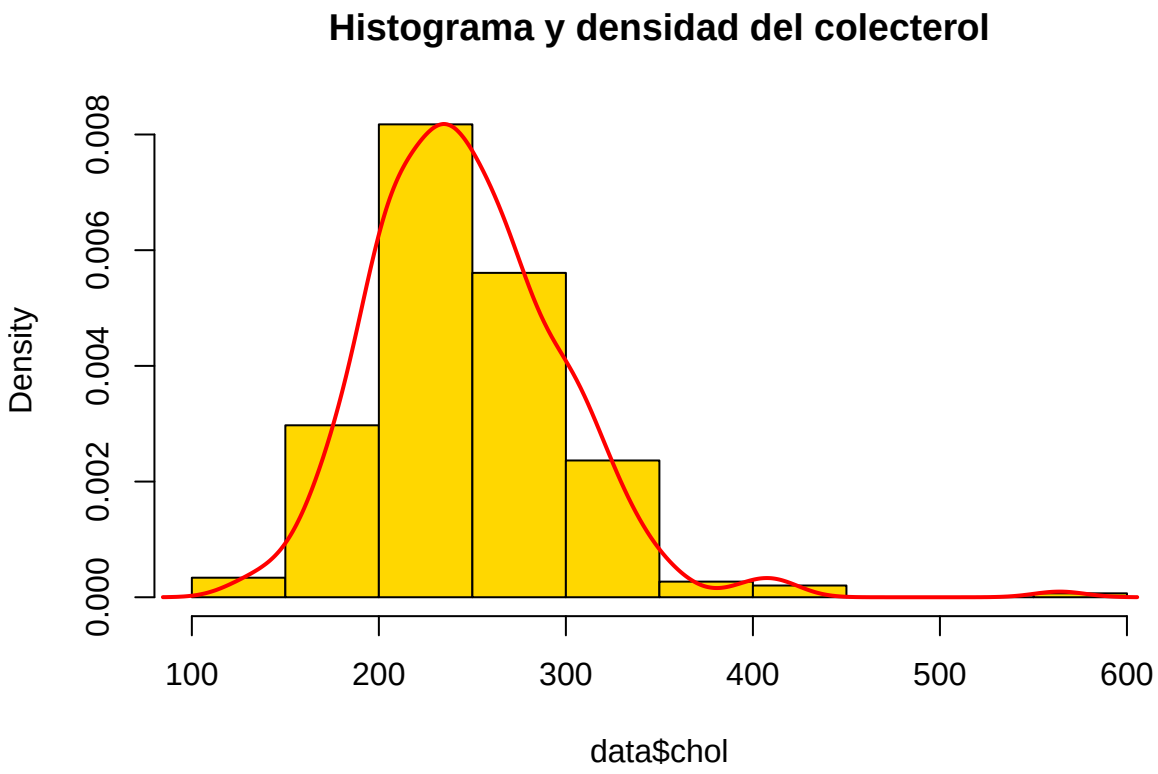
```r
den_age <- density(data$age)
hist1 <- hist(data$age, main = "Histograma y densidad de la edad de los pacientes", col = "gold", freq =
lines(den_age, col = "red", lwd=2)
```




```r
den_trtbps <- density(data$trtbps)
hist2 <- hist(data$trtbps, main = "Histograma y densidad de la frecuencia cardiaca máxima", col = "gold
lines(den_trtbps, col = "red", lwd=2)
```

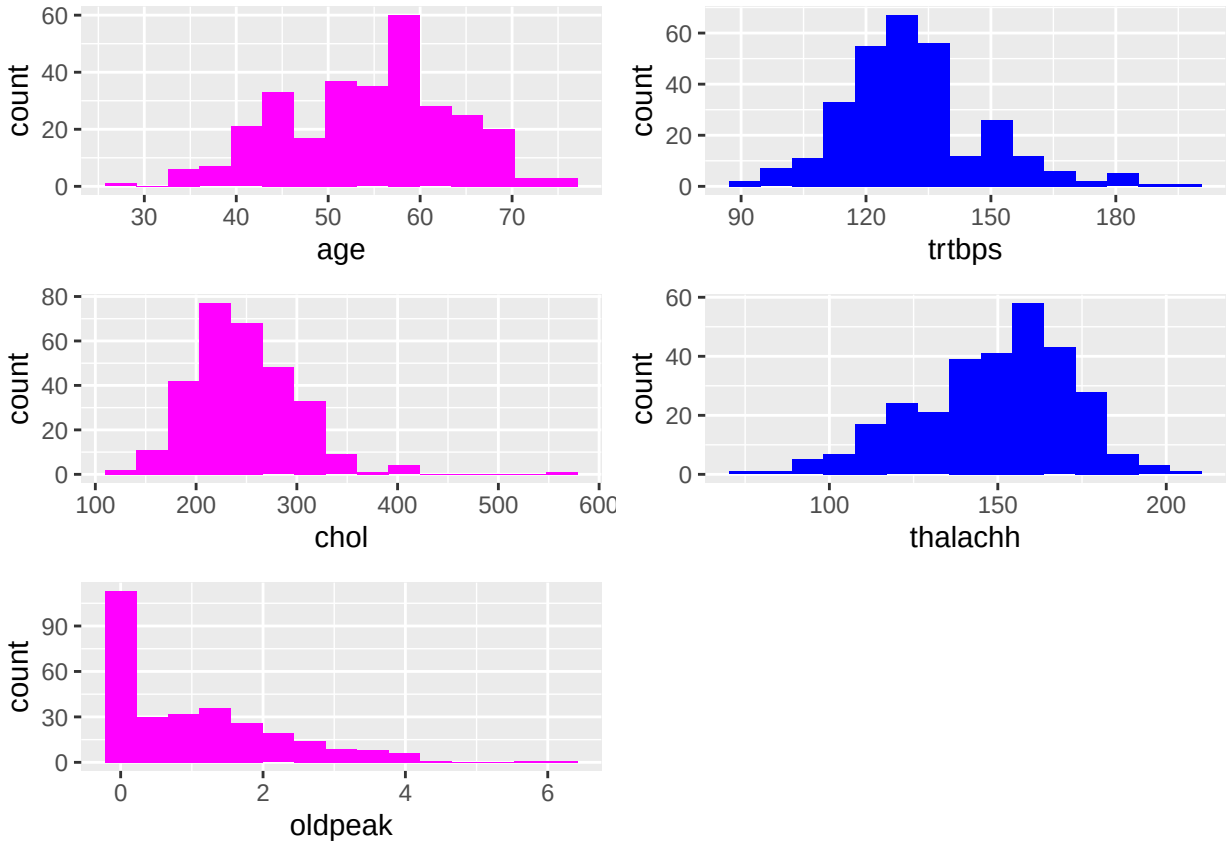



```
den_chol <- density(data$chol)
hist3 <- hist(data$chol, main = "Histograma y densidad del colecterol", col = "gold", freq = FALSE)
lines(den_chol, col = "red", lwd=2)
```




```
hist1 <- ggplot(data,aes(x=age))+geom_histogram(bins=15,fill='magenta')
hist2 <- ggplot(data,aes(x=trtbps))+geom_histogram(bins=15,fill='blue')
hist3 <- ggplot(data,aes(x=chol))+geom_histogram(bins=15,fill='magenta')
hist4 <- ggplot(data,aes(x=thalachh))+geom_histogram(bins=15,fill='blue')
hist5 <- ggplot(data,aes(x=oldpeak))+geom_histogram(bins=15,fill='magenta')

grid.arrange(hist1,hist2,hist3,hist4,hist5,nrow=3)
```



Interpretación del gráfico:

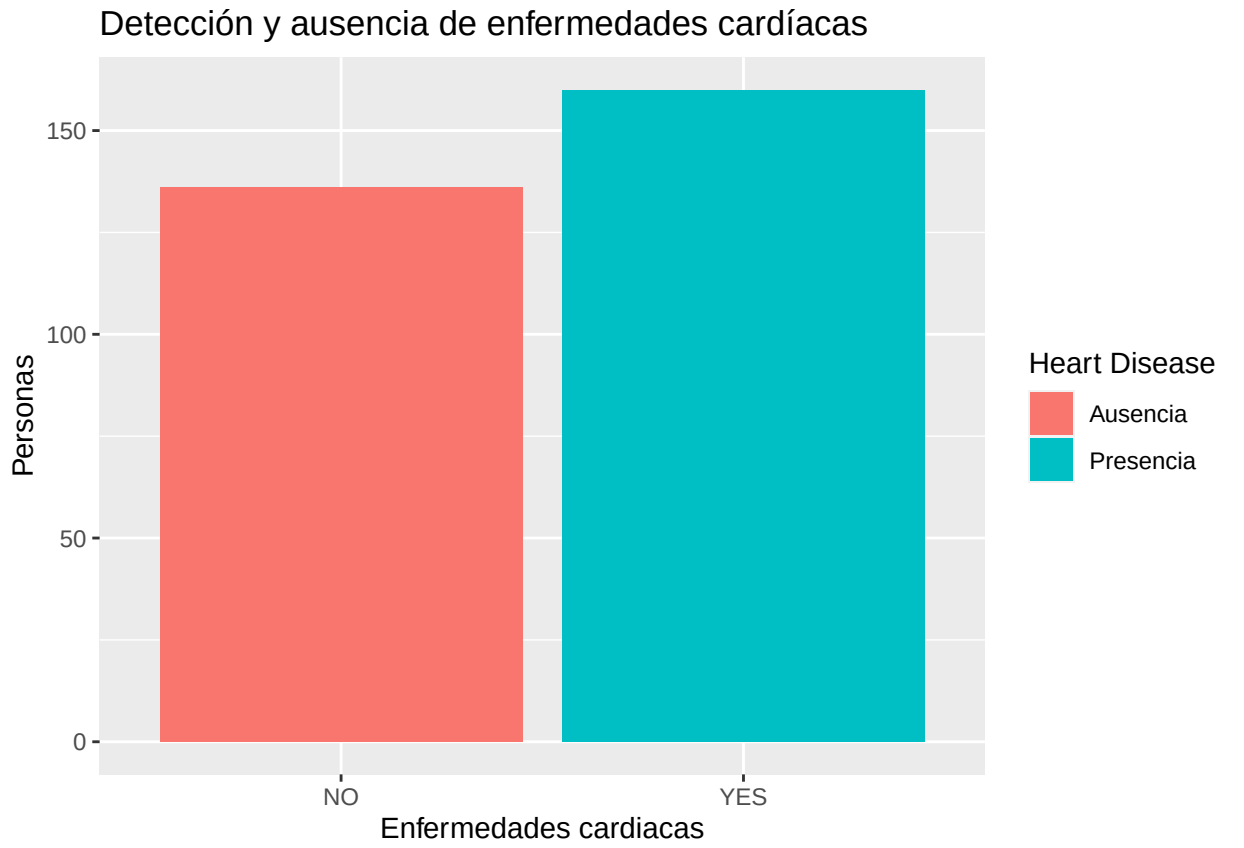
- *age:* La edad sigue una distribución casi normal en cuando al número de ataques sufridos. Parece que hay una franja de edad determinante en cuando al número de estas dolencias. En este caso, valores por debajo de 40 podrían considerarse outliers, un caso en torno a los 29 años parece ser un valor atípico.

Vemos como en el grupo de edad comprendido entre 56 y 60 años, los registros asumen un mayor número de ataques sufridos, estos pueden deberse a la vejez y necesidad de chequeos de salud regulares.

- *trtbps:* Esta variable parece que no sigue una distribución normal concentrandose los valores más frecuentes entre 110 y 140. también vemos un pico en el valor 150.
- *chol:* El colecterol se distribuye siguiendo una distribución normal pero podemos ver como existen algunos valores extremos en torno al valor 600
- *thalachh:* Esta variable se distribuye siguiendo una distribución normal desplazada a la derecha.
- *oldpeak:* Esta variable se aprecia perfectamente que presenta un pico en el valor 0.
- **Variable output**

A continuación, vamos a analizar la variable "ouput".Vamos a obtener el gráfico de barras para comparar cuántas personas tienen o no enfermedades cardíacas

```r
ggplot(data , aes(x=output,fill=output))+
   geom_bar()+
   xlab("Enfermedades cardiacas")+
   ylab("Personas")+
   ggtitle("Detección y ausencia de enfermedades cardíacas")+
   scale_fill_discrete(name="Heart Disease", labels=c("Ausencia","Presencia"))
```



En este conjunto de datos existe mayor cantidad de personas que han sufrido enfermedades cardiacas que persona que no las han sufrido.

Ahora vamos a encontrar la proporción de personas para cuantificar cuántas de ellas tienen enfermedades cardíacas y cuántas no.

```r
prop.table(table(data$output))
```

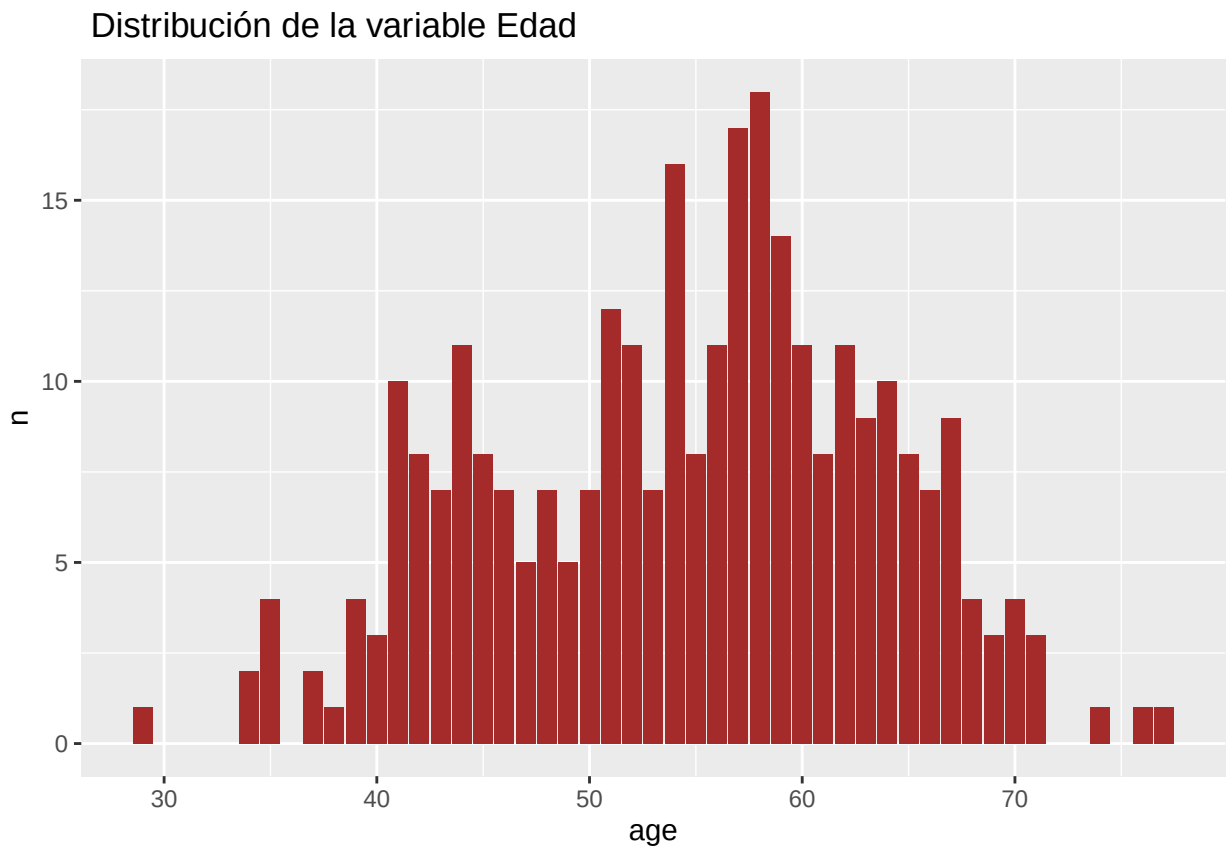
```
##
##        NO       YES
## 0.4594595 0.5405405
```

Los datos anteriores cuantifican que el 54,45% de la población está diagnosticada con alguna enfermedad cardíaca mientras que el 45,54% de la población no padece ningún tipo de enfermedad cardíaca. Además, esta variable presenta datos balanceados.

- **Variable age**

Vamos a comprobar como se distribuye los grupos de edad en el conjunto de datos.

```
data %>%
  group_by(age)%>%
  count()%>%
  ggplot()+
  geom_col(aes(age,n), fill= 'brown')+
  ggtitle(" Distribución de la variable Edad")
```

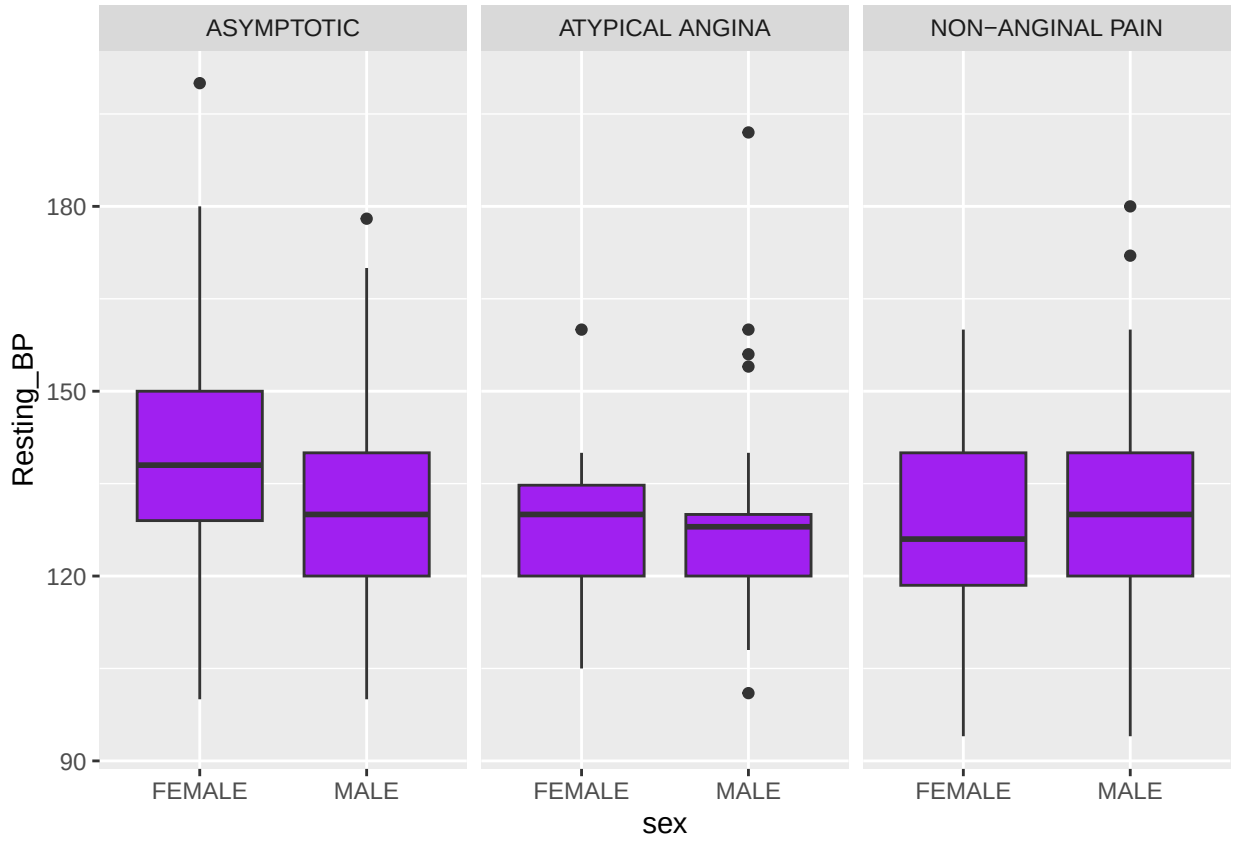


Análisis multivariante

- **Evaluación de la relación entre el diferentes tipos de dolor en el pecho y la presión arterial**

Vamos a Comparar de la "presión arterial" y los "diferentes tipos de dolor en el pecho" entre hombres y mujeres:

```
data%>%
 ggplot(aes(x=sex, y=trtbps))+
 geom_boxplot(fill="purple")+
 xlab('sex')+
 ylab("Resting_BP")+
 facet_grid(~cp )
```



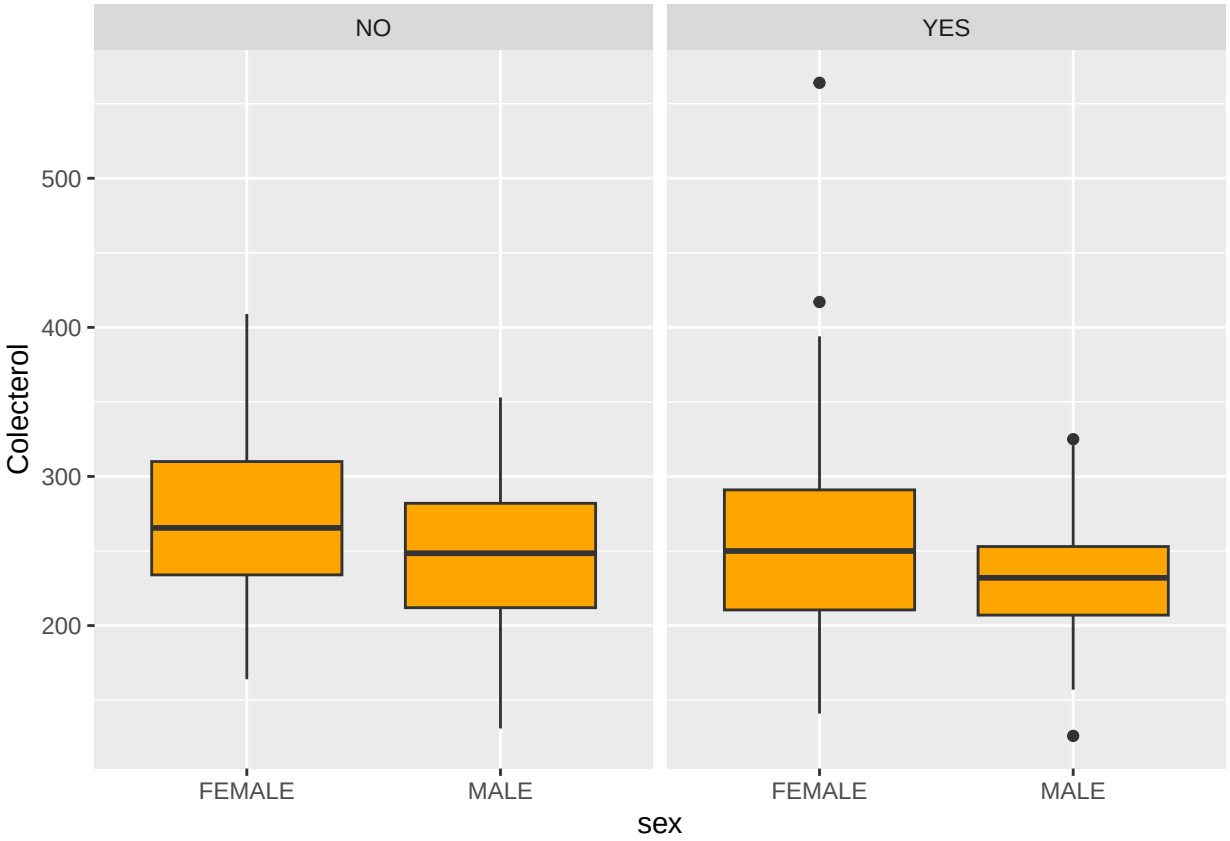
El diagrama de caja anterior muestra tres tipos de dolor y la presión arterial correspondiente a hombres y mujeres. El rango de presión arterial normal es 120, y por encima de 140 se considera una presión arterial más alta. Si cruza 180 puede causar daños graves.

- Para el dolor *asintótico*, las mujeres tienen una presión arterial promedio más alta que los hombres. En las mujeres podemos encotrar algunos valores por encima de 180, que está muy por encima de lo normal, mientras que los hombres también presentan una presión arterial más alta, pero el máximo está por debajo de 180.
- Para el dolor de tipo *atípico*, tanto hombres como mujeres tienen una presión arterial promedio inferior a 150 pero sigue siendo algo superior en las mujeres. En los hpmbres podemos encontrar algunas presiones más altas, por encima de 150 y algunas incluso por encima de 180, lo cual puede ser muy negativo, lo que significa que este tipo de dolor en el pecho puede ser peligroso. En las mujeres solo encontramos un bajor más extreno por encima de 150.
- Para el dolor *no anginoso*, tanto hombres como mujeres tienen una presión arterial promedio inferior a 150, ninguna de las mujeres tiene una presión arterial más alta, algunos hombres tienen cerca de 180. Por otro lado, algunos hombres han visto una presión arterial baja, por debajo de 100 con este tipo de dolor.

Evaluación de la relación entre el colesterol y las enfermedades cardíacas

```
data%>%
 ggplot(aes(x=sex, y=chol))+
 geom_boxplot(fill="orange")+
 xlab('sex')+
```

```
ylab("Colecterol")+
facet_grid(~output)
```



Como se muestra en el diagrama de caja anterior, se muestra la relación entre el rango de colesterol y la presencia de enfermedades cardíacas.

En el diagrama:

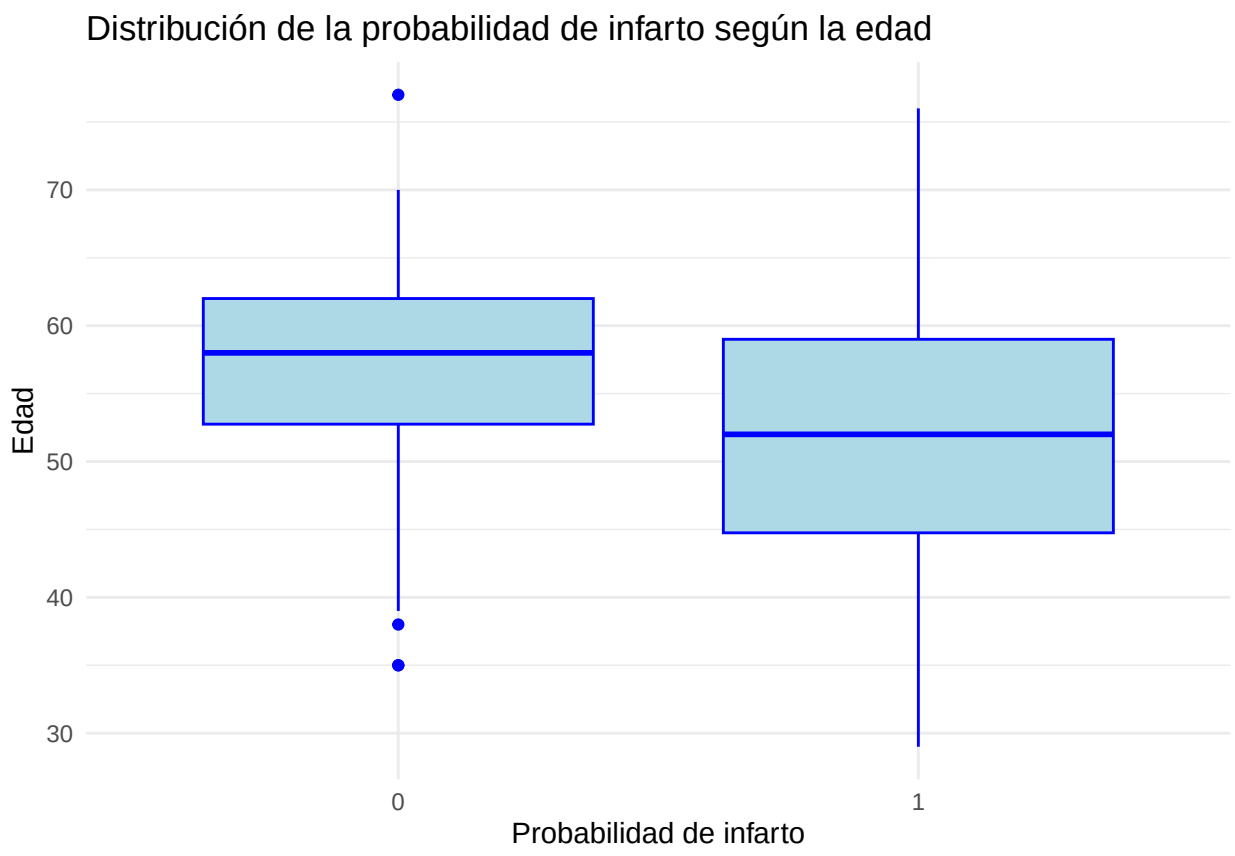
- Menos de 200 en colecteroles es bueno, más de 240 es un colecterol alto, por lo que puede ser peligroso.
- Sí significa presencia de enfermedad cardíaca y
- No significa ausencia de enfermedad cardíaca.

Por tanto podemos deducir:

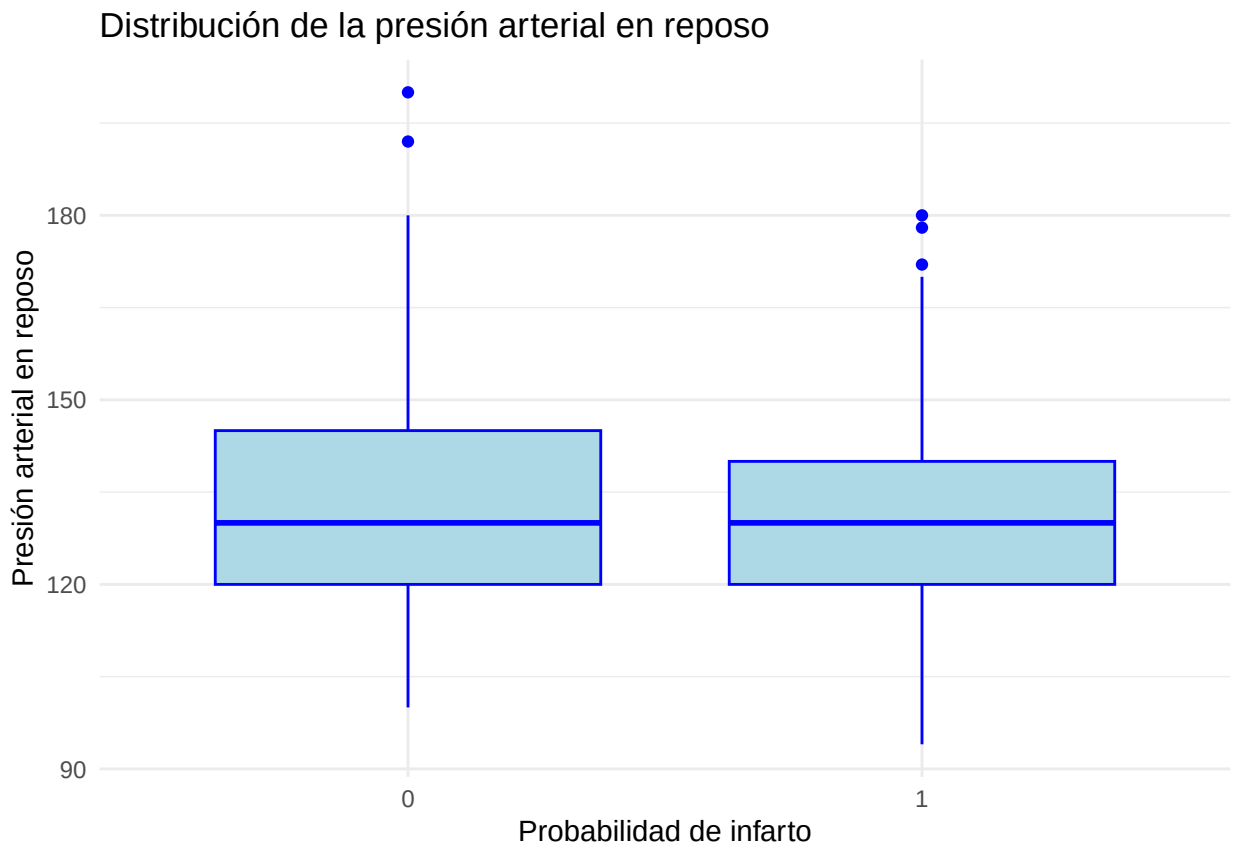
- *Sin enfermedad cardíaca:* Aquellos que no han padecido una enfermedad cardíaca tienen un nivel de colesterol promedio por encima de 250, los hombres tienen menos colesterol, puede deberse a más movimiento físico o menos consumo de alimentos grasos. Además, ninguno de ellos tiene niveles de colesterol elevados.
- *Con presencia de enfermedad cardíaca:* Aquellos que tienen enfermedad cardíaca, el promedio de hombres y mujeres tienen niveles de colesterol en el rango inferior a 250, pero algunos hombres y mujeres tienen valores de colesterol muy altos. En los hombres, algunos valores supera los 300, lo cual no es una buena señal. En las mujeres algunos valores superan los 400 e incluso a algunas se les detectó un nivel de colesterol por encima de 500, encontrandose estos valores en el grupo de mujeres con efermedades cardiacas.

Entonces, esto indica que para deshacerse de las enfermedades cardíacas siempre es mejor controlar el nivel de colesterol por debajo de 250.

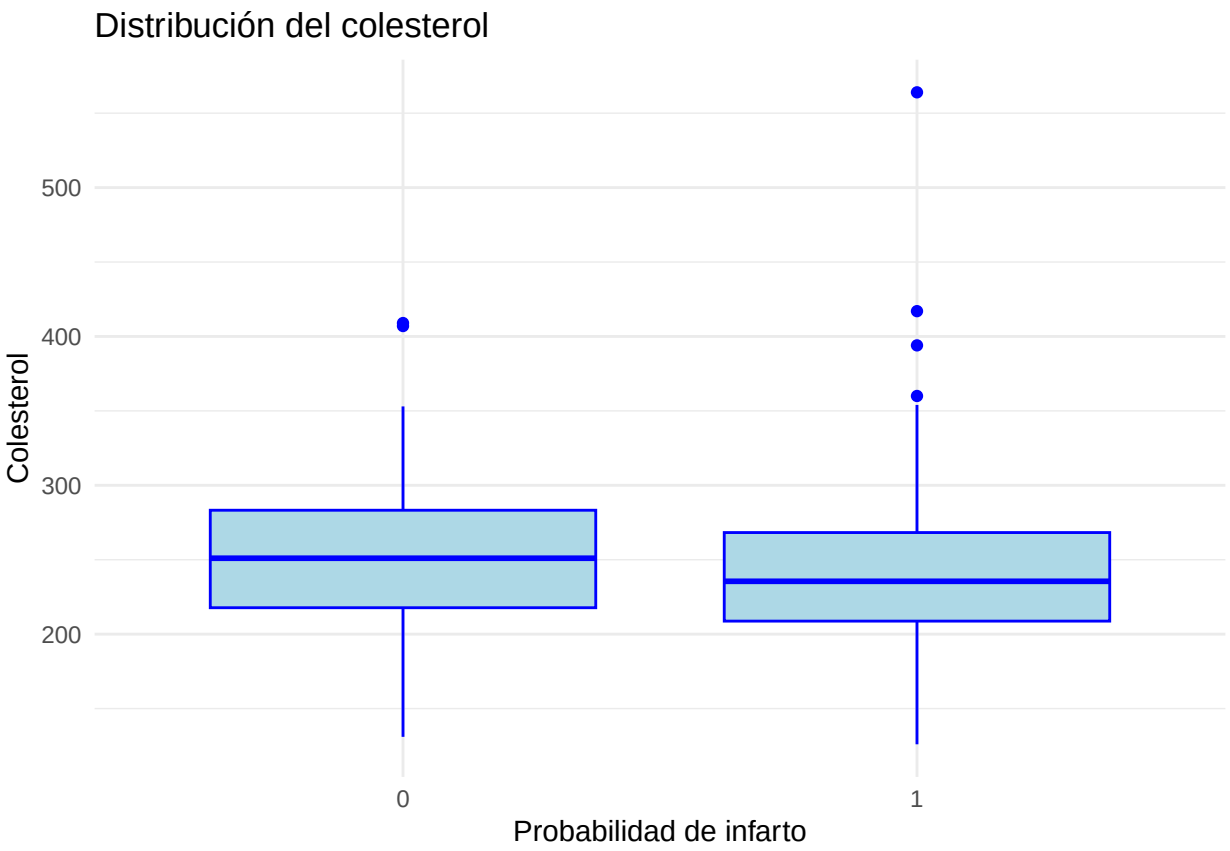
```
# Crear el gráfico de caja
ggplot(heart, aes(x = output, y = age)) +
  geom_boxplot(fill = "lightblue", color = "blue") +
  labs(title = "Distribución de la probabilidad de infarto según la edad",
       x = "Probabilidad de infarto",
       y = "Edad") +
  theme_minimal()
```




```
# Crear el gráfico de caja
ggplot(heart, aes(x = output, y = trtbps)) +
  geom_boxplot(fill = "lightblue", color = "blue") +
  labs(title = "Distribución de la presión arterial en reposo",
       x = "Probabilidad de infarto",
       y = "Presión arterial en reposo") +
  theme_minimal()
```




```r
# Crear el gráfico de caja
ggplot(heart, aes(x = output, y = chol)) +
  geom_boxplot(fill = "lightblue", color = "blue") +
  labs(title = "Distribución del colesterol",
       x = "Probabilidad de infarto",
       y = "Colesterol") +
  theme_minimal()
```



Análisis estadistico

Comprobación de la normalidad

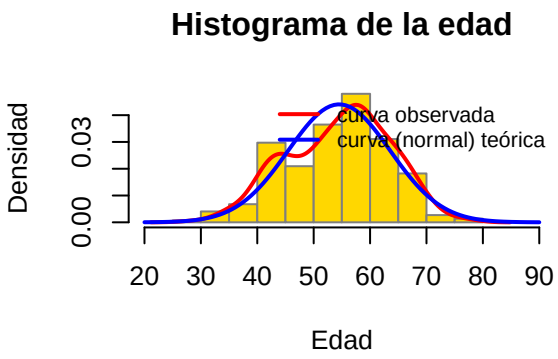
- **Comprobación visual**

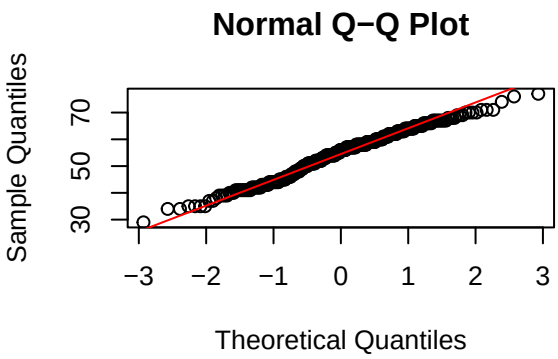
Se pueden emplear visualizaciones de los datos para inspeccionar la normalidad de estos. Para esto podemos usar histogramas, curvas de densidad (comparada con la teórica) así como los gráficos Q-Q.

```
par(mfrow=c(2,2))

hist(data$age, main = "Histograma de la edad", col = "gold",freq = FALSE, xlim = c(20, 90), border = "g
lines(density(data$age),col = "red", lwd = 2)
curve(dnorm(x, mean(data$age), sd(data$age)), lwd = 2, col="blue", add = TRUE)
legend("topright", c("curva observada", "curva (normal) teórica"), lty = 1, lwd = 2, col = c("red", "blu

qqnorm(data$age,  colnames(data$age))
qqline(data$age, col= "red")
```

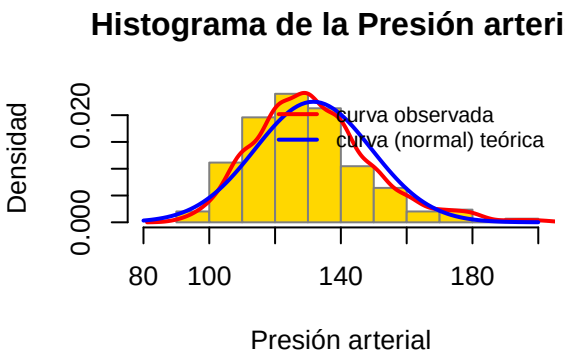


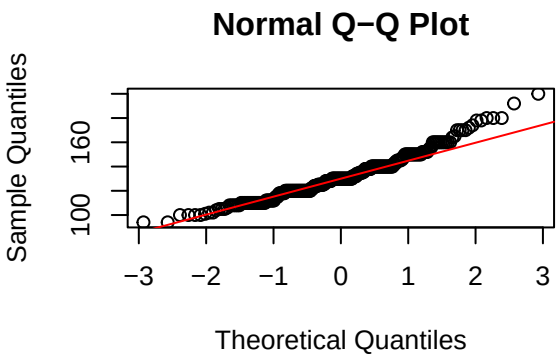



```r
par(mfrow=c(2,2))

hist(data$trtbps, main = "Histograma de la Presión arterial", col = "gold",freq = FALSE, xlim = c(80, 2
lines(density(data$trtbps), lwd = 2, col = "red")
curve(dnorm(x, mean(data$trtbps), sd(data$trtbps)), lwd = 2, col="blue", add = TRUE)
legend("topright", c("curva observada", "curva (normal) teórica"), lty = 1, lwd = 2, col = c("red", "blu

qqnorm(data$trtbps,  colnames(data$trtbps))
qqline(data$trtbps, col= "red")
```

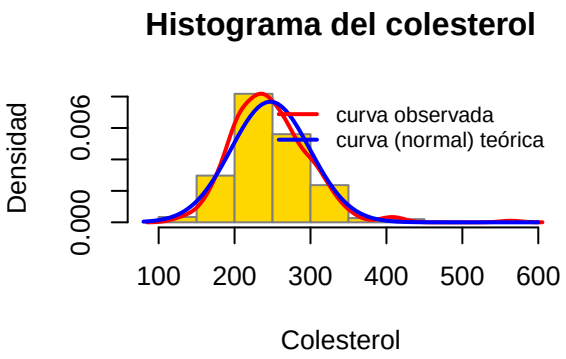


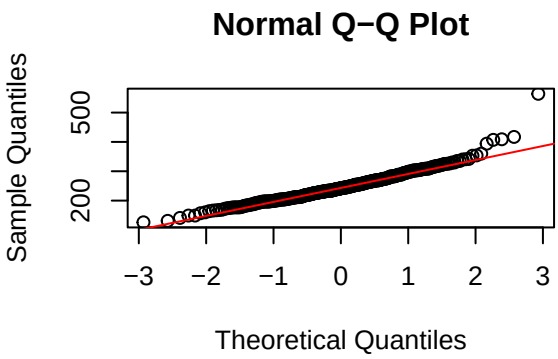



```r
par(mfrow=c(2,2))

hist(data$chol, main = "Histograma del colesterol", col = "gold",freq = FALSE, xlim = c(80, 600), borde
lines(density(data$chol), lwd = 2, col = "red")
curve(dnorm(x, mean(data$chol), sd(data$chol)), lwd = 2, col="blue", add = TRUE)
legend("topright", c("curva observada", "curva (normal) teórica"), lty = 1, lwd = 2, col = c("red", "blu

qqnorm(data$chol,  colnames(data$chol))
qqline(data$chol, col= "red")
```

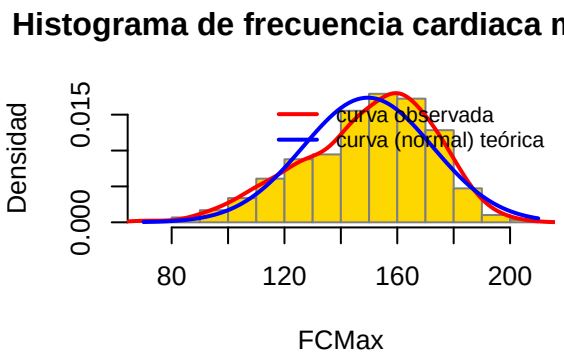


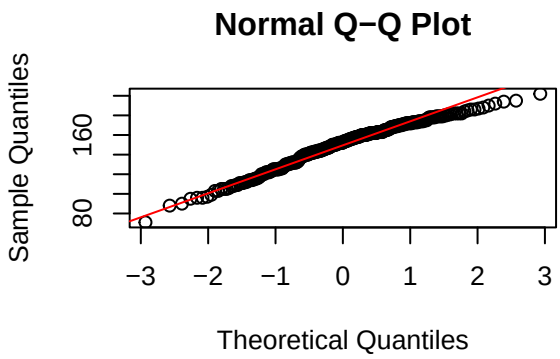



```r
par(mfrow=c(2,2))

hist(data$thalachh, main = "Histograma de frecuencia cardiaca máxima", col = "gold",freq = FALSE, borde
lines(density(data$thalachh),col = "red", lwd = 2)
curve(dnorm(x, mean(data$thalachh), sd(data$thalachh)), lwd = 2, col="blue", add = TRUE)
legend("topright", c("curva observada", "curva (normal) teórica"), lty = 1, lwd = 2, col = c("red", "blu

qqnorm(data$thalachh,  colnames(data$thalachh))
qqline(data$thalachh, col= "red")
```

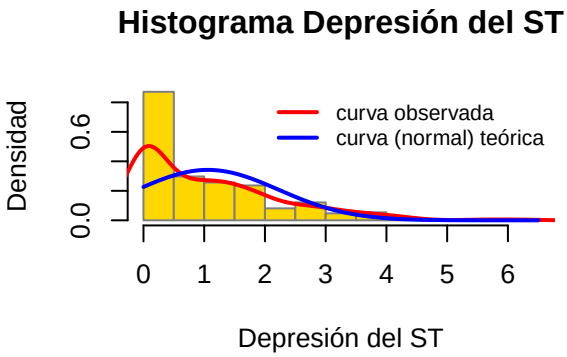


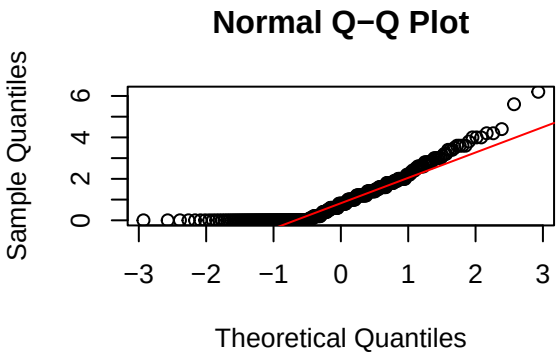



```r
par(mfrow=c(2,2))

hist(data$oldpeak, main = "Histograma Depresión del ST", col = "gold",freq = FALSE, border = "gray50", 
lines(density(data$oldpeak),col = "red", lwd = 2)
curve(dnorm(x, mean(data$oldpeak), sd(data$oldpeak)), lwd = 2, col="blue", add = TRUE)
legend("topright", c("curva observada", "curva (normal) teórica"), lty = 1, lwd = 2, col = c("red", "blu

qqnorm(data$oldpeak,  colnames(data$oldpeak))
qqline(data$oldpeak, col= "red")
```





Interpretación de los gráficos:

- *age:* En el gráfico podemos observar como los puntos de la muestra se representan prácticamente sobre la línea que indica la distribución normal teórica, separándose un poco por los extremos, sobre todo por el extremo superior. Esto significa que podemos asumir normalidad.
- *trtbps:* En el gráfico podemos observar como los puntos de la muestra se representan prácticamente sobre la línea que indica la distribución normal teórica, separándose un poco por los extremos. Esto significa que podemos asumir normalidad.
- *chol:* En el gráfico podemos observar como los puntos de la muestra se representan prácticamente sobre la línea que indica la distribución normal teórica, En el extremo superior hay algunos valores que se separan un poco se la linea que puede deberse a valores extremo que quizas haya que eliminar.Parece que en este caso pdemos asumir normalidad.
- *thalachh:* En el gráfico podemos observar como los puntos de la muestra se representan prácticamente sobre la línea que indica la distribución normal teórica, separándose un poco por los extremos. Esto significa que podemos asumir normalidad.
- *oldpeak:* En el gráfico podemos observar como los puntos de la muestra se representan sobre la linea en la parte central pero los extreno se sapara bastante de la linea presentando asimetría positiva. Esto puede significar que no podemos asumir normalidad.
- **Contraste de normalidad**

Cuando esta normalidad no queda clara mediante métodos visuales se pueden usar pruebas estadísticas que nos den una respuesta menos subjetiva.

Existen varias pruebas para hacer esto, posiblemente las más conocidas son la prueba de Shapiro-Wilk, que se utiliza para muestras pequeña (n < 50) y la prueba de Kolmogorov-Smirnov o normalidad de Lilliefors para muestras grandes (n>50).

La decisión para describir la normalidad de nuestro conjunto de datos estará en función del resultado de un contraste de hipótesis.

Contraste de normalidad de Lilliefors (función lillie.test de la libreria nortest).

```
# Test de Kolmogorov-Smirnov
resultado.normalidad <- lillie.test(data$age) #contraste
resultado.normalidad
```

```
##
##  Lilliefors (Kolmogorov-Smirnov) normality test
##
## data:  data$age
## D = 0.0773, p-value = 0.0002063
```

```
resultado.normalidad$p.value
```

```
## [1] 0.000206259
```

```
# Test de Shapiro-Wilk
resultados.shapiro <- shapiro.test(data$age)
resultados.shapiro
```

```
##
##  Shapiro-Wilk normality test
##
## data:  data$age
## W = 0.98616, p-value = 0.006045
```

```
resultados.shapiro$p.value
```

```
## [1] 0.006044836
```

La idea es simple, si el valor de probabilidad (p-value) que obtenemos por la prueba es menor a al nivel de significancia (0.05) diremos que nuestros datos no siguen una distribución normal. Si el valor de probabilidad es mayor a 0.05, diremos que nuestros datos sí siguen una distribución normal.

Adicionalmente, se va a realizar el método shapiro-wilk, el cuál se utiliza para muestras de menos de 50. Resulta interesante hacerlo debido a que es lo estudiado y se puede ver el contraste entre un método que se ajusta a los datos y uno que no.

```
# Test de Kolmogorov-Smirnov
resultado.normalidad <- lillie.test(data$trtbps) #contraste
resultado.normalidad
```

```
##
##  Lilliefors (Kolmogorov-Smirnov) normality test
##
## data:  data$trtbps
## D = 0.10363, p-value = 3.853e-08
```

```
resultado.normalidad$p.value
```

```
## [1] 3.853401e-08
```

```
# Test de Shapiro-Wilk
resultados.shapiro <- shapiro.test(data$trtbps)
resultados.shapiro
```

```
##
##  Shapiro-Wilk normality test
##
## data:  data$trtbps
## W = 0.96595, p-value = 1.913e-06
```

```
resultados.shapiro$p.value
```

```
## [1] 1.912894e-06
```

```
# Test de Kolmogorov-Smirnov
resultado.normalidad <- lillie.test(data$chol) #contraste
resultado.normalidad
```

```
##
##  Lilliefors (Kolmogorov-Smirnov) normality test
##
## data:  data$chol
## D = 0.052786, p-value = 0.04522
```

```
resultado.normalidad$p.value
```

```
## [1] 0.04521829
```

```
# Test de Shapiro-Wilk
resultados.shapiro <- shapiro.test(data$chol)
resultados.shapiro
```

```
##
##  Shapiro-Wilk normality test
##
## data:  data$chol
## W = 0.94763, p-value = 8.986e-09
```

```
resultados.shapiro$p.value
```

```
## [1] 8.985582e-09
```

```r
# Test de Kolmogorov-Smirnov
resultado.normalidad <- lillie.test(data$thalachh) #contraste
resultado.normalidad
```

```
##
##  Lilliefors (Kolmogorov-Smirnov) normality test
##
## data:  data$thalachh
## D = 0.071816, p-value = 0.0008564
```

```r
resultado.normalidad$p.value
```

```
## [1] 0.0008564483
```

```r
# Test de Shapiro-Wilk
resultados.shapiro <- shapiro.test(data$thalachh)
resultados.shapiro
```

```
##
##  Shapiro-Wilk normality test
##
## data:  data$thalachh
## W = 0.97699, p-value = 0.0001073
```

```r
resultados.shapiro$p.value
```

```
## [1] 0.0001072688
```

```r
# Test de Kolmogorov-Smirnov
resultado.normalidad <- lillie.test(data$oldpeak) #contraste
resultado.normalidad
```

```
##
##  Lilliefors (Kolmogorov-Smirnov) normality test
##
## data:  data$oldpeak
## D = 0.18195, p-value < 2.2e-16
```

```r
resultado.normalidad$p.value
```

```
## [1] 1.166261e-26
```

```r
# Test de Shapiro-Wilk
resultados.shapiro <- shapiro.test(data$oldpeak)
resultados.shapiro
```

```
##
##  Shapiro-Wilk normality test
##
## data:  data$oldpeak
## W = 0.8492, p-value = 2.501e-16
```

```
resultados.shapiro$p.value
```

```
## [1] 2.500522e-16
```

Interpretación del gráfico:

- *age:* El valor de p-value obtenido en las pruebas es de 0.0002231645 para Kolmogorov-Smirnov y de 0.005798359 para Shapiro-Wilk. Ambos resultado presentan un p-values por debajo del nivel de significancia por lo que se rechaza la hipótesis nula y se concluye que los datos no cuentan con una distribución normal.
- *trtbps:* El valor de p-value obtenido en las pruebas es de 4.683143e-08 para Kolmogorov-Smirnov y de 1.458097e-06 para Shapiro-Wilk. Ambos resultado presentan un p-values por debajo del nivel de significancia por lo que se rechaza la hipótesis nula y se concluye que los datos no cuentan con una distribución normal.
- *chol:* El valor de p-value obtenido en las pruebas es de 0.02542595 para Kolmogorov-Smirnov y de 5.364848e-09 para Shapiro-Wilk. Ambos resultado presentan un p-values por debajo del nivel de significancia por lo que se rechaza la hipótesis nula y se concluye que los datos no cuentan con una distribución normal.
- *thalachh:* El valor de p-value obtenido en las pruebas es de 0.0007944562 para Kolmogorov-Smirnov y de 6.620819e-05 para Shapiro-Wilk. Ambos resultado presentan un p-values por debajo del nivel de significancia por lo que se rechaza la hipótesis nula y se concluye que los datos no cuentan con una distribución normal.
- *oldpeak:* El valor de p-value obtenido en las pruebas es de 2.264788e-28 para Kolmogorov-Smirnov y de 8.183378e-17 para Shapiro-Wilk. Ambos resultado presentan un p-values por debajo del nivel de significancia por lo que se rechaza la hipótesis nula y se concluye que los datos no cuentan con una distribución normal.

Aunque en los test de normalidad ninguna de las variables sigue una distribución normal, asumimos normalidad por el **teorema del límite central**, dado que el tamaño de las muestras es suficientemente grade (mayor de 30) conteniendo un total de 296 observaciones. Por tanto asumimos que siguen distribuciones normales.

Comprobación de la homogeneidad de varianzas

Aunque asumimos la normalidad por el teorema del límite central, para calcular la homogeneidad de las varianzas se va a utilizar el test de Filigner-Killeen, es un test no paramétrico que compara las varianzas centrandose en la mediana. Se suele utilizar cuando no se cumple la condición de normalidad.

```
heartData_numeric<- heart %>% select_if(is.numeric)
fligner.test(x = heartData_numeric)
```

```
##
##  Fligner-Killeen test of homogeneity of variances
##
## data:  heartData_numeric
## Fligner-Killeen:med chi-squared = 2691.1, df = 12, p-value < 2.2e-16
```

Como se puede observar no se cumple la condición de homogeneidad de varianzas donde p>0.05, por lo que no se aplicarán pruebas paramétricas como podias ser t-Student.

Pruebas estadísticas

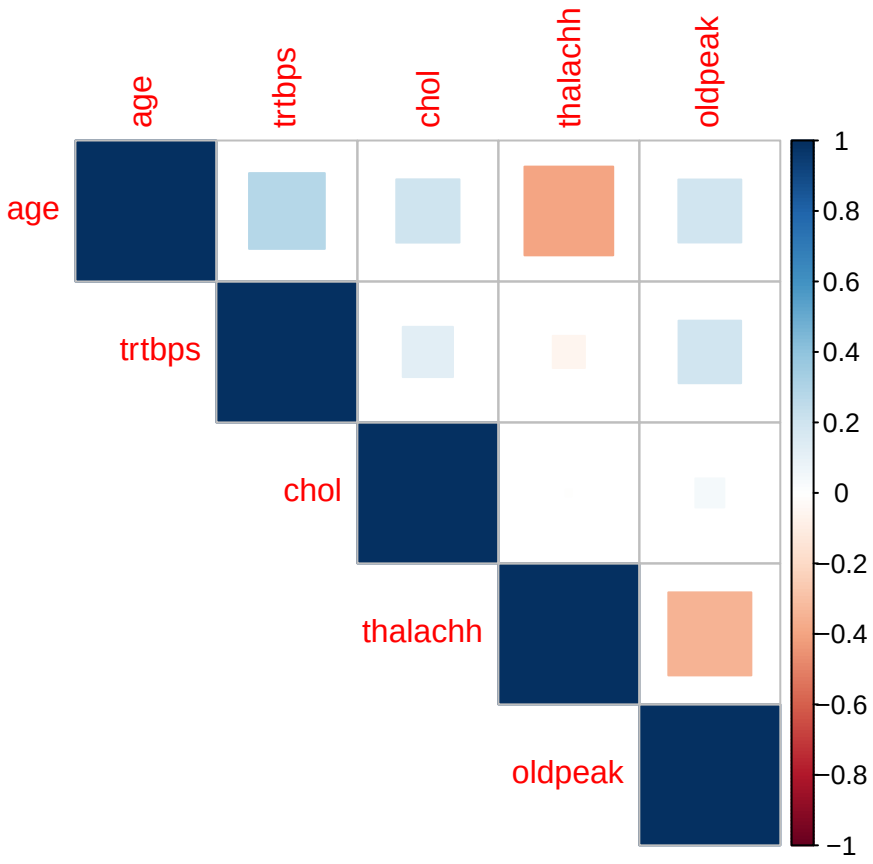
Correlaciones y estudio estadístico

Desarrollar una correlación entre diferentes columnas como edad, sexo, cp, trtbps, chol, fbs, restecg, thalachh, exng, oldpeak, slp, caa, thalloutput para encontrar cuál está fuertemente relacionada con qué variable y cuál no tiene ningún vínculo con ninguna variable.

```
cor_heart <- cor(data[,10:14])
cor_heart
```

```
##                 age      trtbps         chol     thalachh     oldpeak
## age       1.0000000  0.28870116  0.200919604 -0.395988704  0.19937301
## trtbps    0.2887012  1.00000000  0.126704949 -0.051817031  0.19679275
## chol      0.2009196  0.12670495  1.000000000 -0.001947092  0.04214406
## thalachh -0.3959887 -0.05181703 -0.001947092  1.000000000 -0.34674198
## oldpeak   0.1993730  0.19679275  0.042144058 -0.346741982  1.00000000
```

```
corrplot(cor_heart, method="square", type="upper",)
```



El gráfico de correlación anterior indica qué tan fuertemente se relaciona una variable con otras variables.

- La magnitud 1 indica una fuerte correlación y los valores negativos próximos a -1 indican que existe una correlación inversamente proporcional, es decir, cuando los valores de una variable aumentan, los valores de la otra disminuyen.

- Del mismo modo, cuanto más oscuro sea el color, más fuerte será la relación, mientras que con el atenuador el color será más amplio, la relación será más débil.

Sabiendo esto se deduce del gráfico anteror lo siguiente:

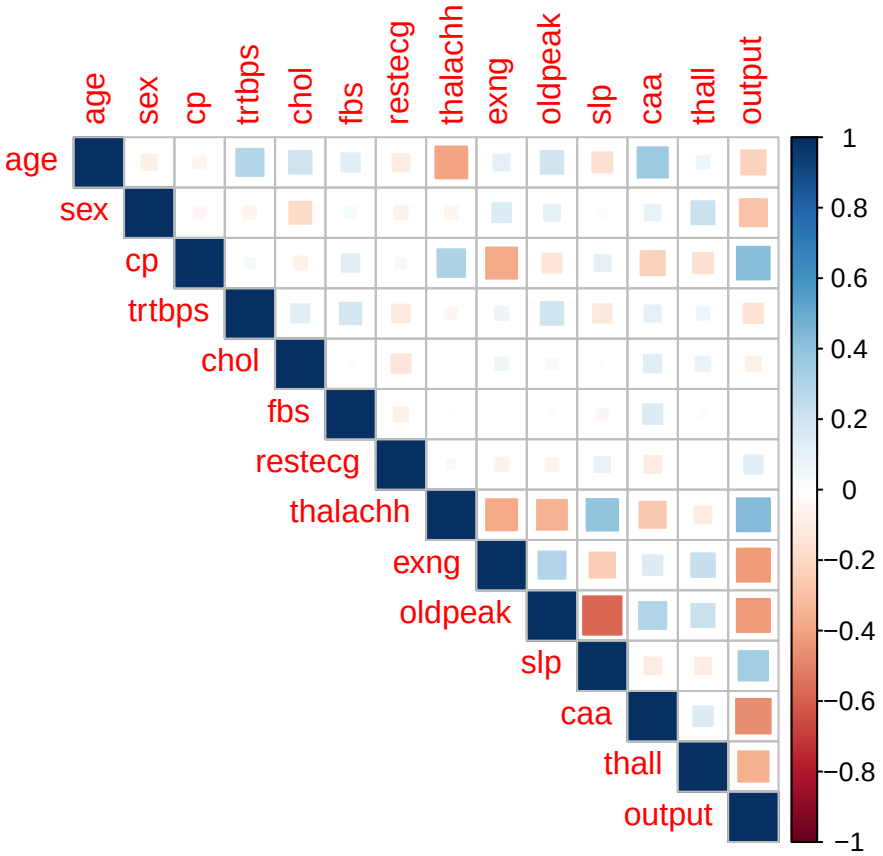
1.- La edad está fuertemente relacionada con la presión arterial, el colesterol, mientras que no tiene ninguna relación con la frecuencia cardíaca máxima alcanzada (thalach).

2.- De manera similar, la PA en reposo también se relaciona con el nivel de colesterol (chol) y la depresión del ST inducida por el ejercicio en relación con el reposo (oldpeak).

3.- La frecuencia cardíaca máxima (thalach) no tiene ningún vínculo con la depresión del ST inducida por el ejercicio en relación con el reposo (oldpeak)

A continuación, se desarrolla una correlación con enfoque en la variable output, la cual nos indicará como estará relacionada cada una de las variables en comparación.

```
heart$output <- as.numeric(as.character(heart$output))
cor_heart_output <- cor(heart)
cor_heart_output <- round(cor_heart_output, 2)
corrplot(cor_heart_output, method = "square", type = "upper")
```



Deducciones centrandonos en la variable output:

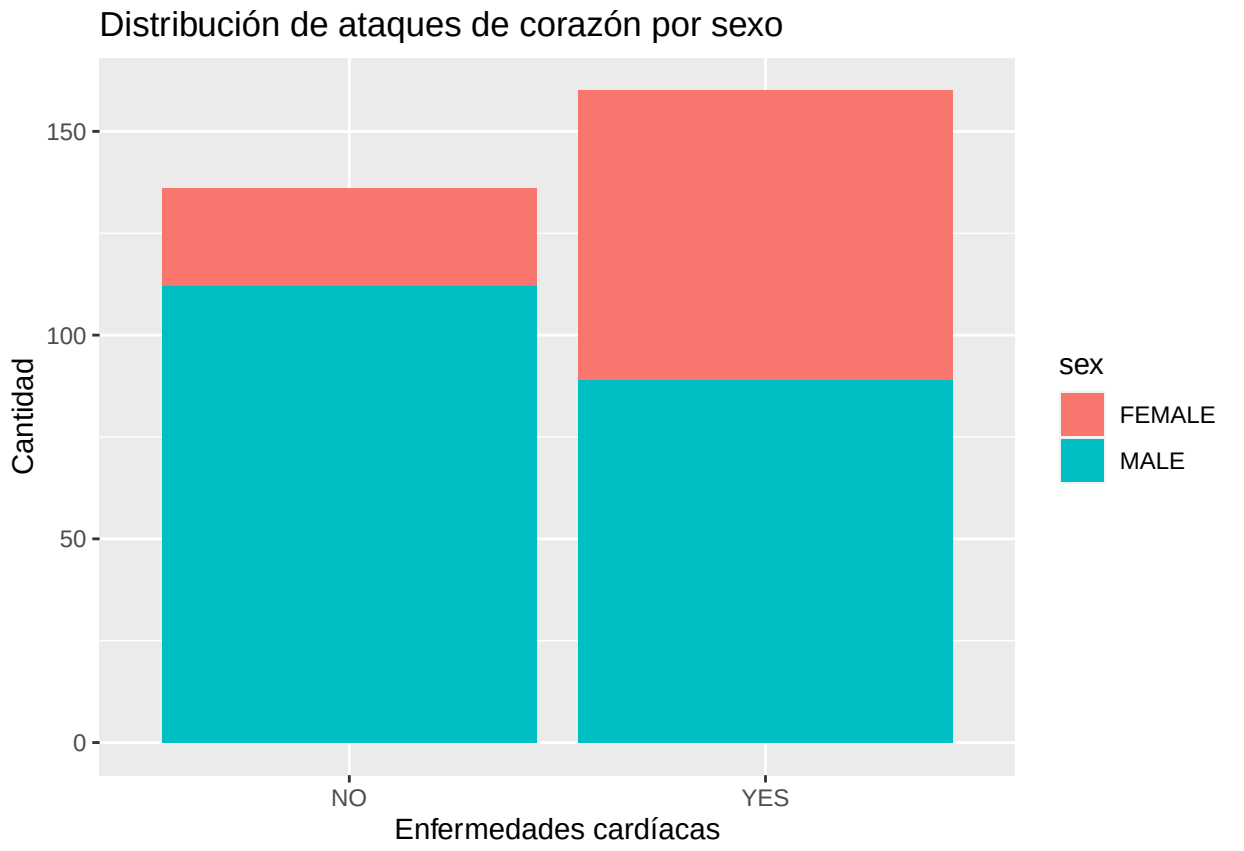
- La variable con mayor correlación, aunque negativa es “caa”, el número de vasos sanguineos, con una correlación de -0.46.
- La variables “cp” y “exng” y “thalach”, también presentan una correlación moderada entorno al 0.42.

- Por otro lado, las correlaciones por debajo del 0.30, aproximadamente, son catalogadas como correlaciones débiles, puesto, que si queremos realizar una reducción de dimensionalidades las variables "trtbps", "chol", "fbs" y "restecg" serían las más estudiadas.
- Las variables "age" y "sex", que a priori parecen ser factores determinantes, parece que no estan tan relacionadas como se podría imaginar y se tendría que estudiar a fondo el impacto que tendría en la presencia de ataques, pues presentan una correlación de -0.22 y -0.29, respectivamente.

Vista la correlación, a continuación, se va a representar la distribución de ataques en función del sexo del paciente.

Dependencia de variables Sex y Output

```
# Crear el gráfico de barras apiladas
ggplot(data, aes(x = output, fill = sex)) +
  geom_bar(position = "stack") +
  labs(title = "Distribución de ataques de corazón por sexo",
       x = "Enfermedades cardíacas",
       y = "Cantidad")
```



Haciendo foco en los ataques existentes se puede ver un balance en el número de ellos en hombres y mujeres, lo que a priori, parece quitar relevancia en cuanto a influencia se refiere en estos ataque. Si nos fijamos en la frecuencia de variables, dentro de las muestras, tenemos un 32.22% de mujeres y un 67.78% de hombres, por lo que podemos discernir que, aunque la cantidad de ataques se encuentra balanceada, el número de ellos es mujeres resulta mayor, puesto que el porcentaje de mujeres en la muestra es menor. Para comprobar esta suposición se va a realizar un contraste de hipotesis:

- Hipotesis nula: el factor Sex y el factor Output son independientes.
- Hipotesis alternativa: los dos factores son dependientes.

Para comprobar la dependencia entre dos variables categóricas, se aplica el test chi-cuadrado. Primero, se genera una tabla de contingencia entre ambos factores.

```
contingence_sex <- table(data$output, data$sex)
contingence_sex
```

```
##
##       FEMALE MALE
##   NO      24  112
##   YES     71   89
```

Una vez creada la tabla se aplica el test chi-cuadrado

```
chisq.test(contingence_sex)
```

```
##
##  Pearson's Chi-squared test with Yates' continuity correction
##
## data:  contingence_sex
## X-squared = 22.886, df = 1, p-value = 1.719e-06
```

Este resultado indica que hay evidencia significativa para rechazar la hipótesis nula de independencia entre las variables output y sex. Por lo tanto se asume la dependencia de factores, el test asume que el sexo es determinante en la existencia de ataque al corazón.

Dependencia de variables Age y Output

A continuación, se comprueba si la edad de los pacientes influye en los ataques de corazón, para ello se realizará una regresión logística simple con el fin de poder discernir si existe dependencia de variable:

```
modelo_logistico.age <- glm(output ~ age, data = data, family = "binomial")

summary(modelo_logistico.age)
```

```
##
## Call:
## glm(formula = output ~ age, family = "binomial", data = data)
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)  3.03304    0.76846   3.947 7.92e-05 ***
## age         -0.05245    0.01381  -3.797 0.000147 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 408.40  on 295  degrees of freedom
## Residual deviance: 392.98  on 294  degrees of freedom
## AIC: 396.98
##
## Number of Fisher Scoring iterations: 4
```

Podemos ver que el valor p asociado con el coeficiente de age es 0.000152. Este valor p es menor que el nivel de significancia 0.05, lo que indica que el coeficiente de age es estadísticamente significativo.

Regresión logística

En este punto vamos a realizar una serie de regresiones logísticas, empezando desde un modelo básico, creando modelos posteriores para observar la mejora. Para poder estimar de forma más objetiva la precisión del modelo, separaremos el conjunto de datos en dos partes: el conjunto de entrenamiento (training) y el conjunto de prueba (testing). Ajustaremos el modelo de regresión logística con el conjunto de entrenamiento, y evaluaremos la precisión con el conjunto de prueba.

Debido a las conclusiones llegadas en los apartados anteriores las variables a utilizar en el desarrollo de estas regresiones son: sex, age, cp, thalachh, exng, oldpeak, slp, caa, thall.

```
prop_train <- 0.8

# Crear un vector de índices aleatorios para el conjunto de entrenamiento
indices_train <- sample(1:nrow(heart),
size = round(prop_train * nrow(heart)))

# Crear conjuntos de entrenamiento y prueba
train <- heart[indices_train, ]
test <- heart[-indices_train, ]
dim(train)
```

```
## [1] 237  14
```

```
dim(test)
```

```
## [1] 59 14
```

Modelo básico

Primero, se creará un modelo de regresión logístico básico con la variable más correlacionada, caa, para tener una versión inicial sobre la que trabajar.

```
modelo_logistico.basic <- glm(output ~ caa, data = train,
                      family = "binomial"(link=logit))
summary(modelo_logistico.basic)
```

```
##
## Call:
## glm(formula = output ~ caa, family = binomial(link = logit),
##     data = train)
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)   0.9202     0.1797    5.12 3.05e-07 ***
## caa          -1.3321     0.2069   -6.44 1.19e-10 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
```

```
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 327.84  on 236  degrees of freedom
## Residual deviance: 265.43  on 235  degrees of freedom
## AIC: 269.43
##
## Number of Fisher Scoring iterations: 4
```

```
positive <- sum(test$output==1)
negative <- sum(test$output==0)

# Función para el cálculo de métricas de rendimiento
calcular_metricas <- function(modelo, threshold = 0.5) {
  prediccion <- predict(modelo, test, type="response")
  pred_class <- ifelse(prediccion > threshold, 1, 0)

  matriz_confusion <- table(test$output, pred_class)
  cat("Matriz de confusion\n")
  print(matriz_confusion)

  TP <- matriz_confusion[2, 2]  # Verdaderos positivos
  TN <- matriz_confusion[1, 1]  # Verdaderos negativos
  FP <- matriz_confusion[2, 1]  # Falsos positivos
  FN <- matriz_confusion[1, 2]  # Falsos negativos

  accuracy <- (TP + TN) / (TP + TN + FP + FN)
  cat("Accuracy del modelo:", accuracy, "\n")

  attr(modelo, "threshold") <- threshold
  attr(modelo, "matriz_confusion") <- matriz_confusion
  attr(modelo, "TP") <- TP
  attr(modelo, "TN") <- TN
  attr(modelo, "FP") <- FP
  attr(modelo, "FN") <- FN
  attr(modelo, "accuracy") <- accuracy

  attr(modelo, "sensitivity") <- TP/positive
  attr(modelo, "specificity") <- TN/negative

  cat("Sensitividad:", attr(modelo, "sensitivity"), "\n")
  cat("Especificidad:", attr(modelo, "specificity"))
  return(modelo)
}

modelo_logistico.basic <- calcular_metricas(modelo_logistico.basic)
```

```
## Matriz de confusion
##    pred_class
##      0  1
##   0 16  8
##   1 10 25
## Accuracy del modelo: 0.6949153
## Sensitividad: 0.7142857
## Especificidad: 0.6666667
```

Modelo con correlaciones más altas

Una vez hemos construido un modelo básico vamos a tratar de mejorarlo, para ellos vamos a introducir las variables más correlacionadas. En este caso, existe una correlación moderada de las variables, caa, cp, thalachh, exng + thall, oldpeak u slp.

```
modelo_logistico.more_correlated <- glm(output ~ caa + cp + thalachh + exng + thall + oldpeak + slp, da
                      family = "binomial"(link=logit))
summary(modelo_logistico.more_correlated)
```

```
##
## Call:
## glm(formula = output ~ caa + cp + thalachh + exng + thall + oldpeak +
##     slp, family = binomial(link = logit), data = train)
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)  0.036817   1.600745   0.023 0.981650
## caa         -1.285272   0.270493  -4.752 2.02e-06 ***
## cp           0.568928   0.202187   2.814 0.004895 **
## thalachh     0.019191   0.009471   2.026 0.042721 *
## exng        -0.980585   0.454856  -2.156 0.031098 *
## thall       -1.182764   0.328087  -3.605 0.000312 ***
## oldpeak     -0.563741   0.232598  -2.424 0.015364 *
## slp          0.712391   0.366845   1.942 0.052145 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 327.84  on 236  degrees of freedom
## Residual deviance: 175.29  on 229  degrees of freedom
## AIC: 191.29
##
## Number of Fisher Scoring iterations: 6
```

Métricas de este modelo:

```
modelo_logistico.more_correlated <- calcular_metricas(modelo_logistico.more_correlated)
```

```
## Matriz de confusion
##    pred_class
##      0  1
##   0 20  4
##   1  3 32
## Accuracy del modelo: 0.8813559
## Sensitividad: 0.9142857
## Especificidad: 0.8333333
```

Añadiendo las variables con correlación moderada se ve una mejora en el modelo en general. Aumenta la sensitividad o recall, esto quiere decir que el modelo predice mejor los positivos reales. Aumenta tambien la especificidad, la cuál representa la capacidad del modelo para identificar correctamente las instancias negativas en un conjunto de datos.

Modelo añadiendo age y sex

Vistos estos resultados, vamos a incluir en el modelo aquellas variables sobre las que realizamos procedimientos estadísticos para discernir su dependencia, **age** y **sex**. En apartados anteriores concluimos su relevancia y correlación con la variable output, por lo que añadir estas variables debería suponer una mejora.

```
modelo_logistico.complex <- glm(output ~ caa + cp + thalachh + exng + thall + oldpeak + slp + age + sex
                      family = "binomial"(link=logit))
summary(modelo_logistico.complex)
```

```
##
## Call:
## glm(formula = output ~ caa + cp + thalachh + exng + thall + oldpeak +
##     slp + age + sex, family = binomial(link = logit), data = train)
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)  1.38758    2.64375   0.525  0.59969
## caa         -1.21548    0.29649  -4.100 4.14e-05 ***
## cp           0.63994    0.21007   3.046  0.00232 **
## thalachh     0.02025    0.01081   1.872  0.06116 .
## exng        -0.97427    0.47061  -2.070  0.03843 *
## thall       -0.99561    0.33324  -2.988  0.00281 **
## oldpeak     -0.52271    0.23481  -2.226  0.02601 *
## slp          0.69723    0.37538   1.857  0.06326 .
## age         -0.01941    0.02662  -0.729  0.46593
## sex         -1.42725    0.47874  -2.981  0.00287 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 327.84  on 236  degrees of freedom
## Residual deviance: 165.43  on 227  degrees of freedom
## AIC: 185.43
##
## Number of Fisher Scoring iterations: 6
```

```
modelo_logistico.complex <- calcular_metricas(modelo_logistico.complex)
```

```
## Matriz de confusion
##    pred_class
##      0  1
##   0 20  4
##   1  3 32
## Accuracy del modelo: 0.8813559
## Sensitividad: 0.9142857
## Especificidad: 0.8333333
```

Se puede observar como con estas variables se realiza una mejora de la precisión del modelo, aumentando la sensitividad o recall y la especificidad, obtenemos un modelo con bastante precisión en cuanto a la clasificación de pacientes con mayor riesgo de ocurrencia de ataque en función de variables clave.

Modelo añadiendo resto de variables

Visto que este dataset no tiene un número elevado de muestras y la complejidad del modelo no es elevada, por lo que no se tiene porque acudir a técnicas de reducción de características, se van a introducir en un último modelo el resto de variables con correlaciones más debil

```
modelo_logistico.complet <- glm(output ~ caa + cp + thalachh + exng + thall + oldpeak + slp + age + sex
                      family = "binomial"(link=logit))
summary(modelo_logistico.complet)
```

```
##
## Call:
## glm(formula = output ~ caa + cp + thalachh + exng + thall + oldpeak +
##     slp + age + sex + trtbps + chol + fbs + restecg, family = binomial(link = logit),
##     data = train)
##
## Coefficients:
##               Estimate Std. Error z value Pr(>|z|)
## (Intercept)  3.8088511  3.0156322   1.263  0.20658
## caa         -1.4054810  0.3300638  -4.258 2.06e-05 ***
## cp           0.6653952  0.2223897   2.992  0.00277 **
## thalachh     0.0268234  0.0117971   2.274  0.02298 *
## exng        -0.9182000  0.4894255  -1.876  0.06064 .
## thall       -0.9019028  0.3532544  -2.553  0.01068 *
## oldpeak     -0.4848678  0.2525672  -1.920  0.05489 .
## slp          0.7864993  0.4030811   1.951  0.05103 .
## age          0.0009601  0.0278667   0.034  0.97251
## sex         -1.7576102  0.5364885  -3.276  0.00105 **
## trtbps      -0.0296130  0.0121852  -2.430  0.01509 *
## chol        -0.0045444  0.0043219  -1.051  0.29303
## fbs          0.8710130  0.6633125   1.313  0.18914
## restecg      0.5040004  0.4123106   1.222  0.22156
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 327.84  on 236  degrees of freedom
## Residual deviance: 155.14  on 223  degrees of freedom
## AIC: 183.14
##
## Number of Fisher Scoring iterations: 6
```

```
modelo_logistico.complet <- calcular_metricas(modelo_logistico.complet)
```

```
## Matriz de confusion
##    pred_class
##      0  1
##   0 20  4
##   1  4 31
## Accuracy del modelo: 0.8644068
## Sensitividad: 0.8857143
## Especificidad: 0.8333333
```

Podemos ver como se han mejorado las métricas de rendimiento con las variables con correlación más

débiles. Parece ser que es un buen modelo. También podemos ver como todas las variables han tenido relativa relevancia sobre la variable **output**.

Conclusiones

Recordando la importancia de este dataset y una de las muchas preguntas que puede responder, vamos a unificar todas las conclusiones que nos permiten responder a esta pregunta en este apartado.

Antes de poder realizar algún análisis o prueba estadística, se procedió con la limpieza de los datos, en la que se eliminaron algunos valores no válidos en las columnas **caa** y **thall**, también se descubrieron algunos valores outliers en las columnas **trtbps**, **chol**, **thalachh** y **oldpeak**, pudiendo estos ser relevantes en cuanto a tener o no ataques cardíacos.

Para poder comprender mejor la naturaleza y distribución de los datos se visualizaron algunas variables de interés, con la descripción detallada de cada una de las variables que componen el conjunto de datos.

Se realizó también un análisis de la normalidad, tanto visual como estadístico, donde se pudo comprobar la no normalidad de las variables, aunque, se puede asumir la normalidad por el teorema del límite central. También se realizó un análisis estadístico de la homogeneidad de varianzas, donde se indicó la heterogeneidad de varianza.

En cuanto a las pruebas estadísticas, se realizaron pruebas no paramétricas. Primero se realizaró la correlación de Pearson, para comprobar relaciones entre variables y como de significativas estas son. En esta correlación se dedujo que **caa**, **cp**, **exng** y **thalach**, son las variables más correlacionadas, mientras que, sorpresivamente, **age** y **sex** no estaban tan correlacionadas como se esperaba. Por esta razón se realizaron dos test, contraste de hipótesis y regresión logística, para comprobar la independencia de estas variables. Como resultado se obtuvo que ambas variables son significativas en la existencia de ataques al corazón.

Por último, para comprobar esta significancia se realizaron múltiples modelos logísticos para comprobar cómo estas variables afectan a la variable independiente output. En estos modelos, se pudieron ver las variables clave, como se puntualizó, fueron aquellas con mayor correlación, **caa**, **cp**, **exng** y **thalach**. Las variables **age** y **sex** pudieron comprobar su relevancia en otro modelo que obtuvo un incremento en sus métricas. Las variables más débiles también fueron buenas para aumentar la precisión, por lo que se pudo comprobar que también eran significativas. Dadas las dos dimensiones del dataset, no es necesaria la reducción de dimensionalidad, por lo que se puede ver la influencia de cada una de las variables en la variable independiente **output**.

Datos finales analizados:

```
write.csv(heart, "../data/processed/heart.csv", row.names = FALSE)
```