Práctica 2

Marco Rizzetto

2023-06-16

1.Descripción del dataset

El conjunto de datos objeto de estudio contiene datos médicos relativos a las condiciones de salud de más de 300 pacientes. El objetivo del dataset es el de proporcionar un valor predictivo para determinar si el paciente tiene un alto riesgo de padecer enfermedades cardíacas o no.

Antes de generar un nuevo modelo de regresión logística basado en los datos disponibles se quiere responder a algunas preguntas básicas relativas a diferencias entre grupos demográficos, o buscar correlaciones entre algunos parámetros médicos sencillos y el resultado de la predicción contenida en el dataset. Aplicar algoritmos automáticos para determinar si un paciente puede padecer enfermedades cardíacas es muy útil en campo médico y sobre todo de medicina preventiva, pudiendo ser muy útil detectar factores de riesgo en determinadas poblaciones para empezar a hacer screening en el momento más adecuado.

Se carga el dataset utilizando la función *read.csv()* y se puede comprobar que se ha cargado correctamente utilizando las funciones *dim()* y *head()*. En este caso se puede ver que se ha importado bien.

Se puede proceder a comprobar que las variables se hayan importado en el formato correcto utilizando la función *sapply()* y luego comprobando los valores de cada variable.

- **age**: la edad del paciente en años, es una variable de tipo integer, los valores son los esperados, entre 29 y 77 años
- **sex**: el sexo del paciente, debería ser una variable dicotómica y se ha importado como integer, por tanto se cambian al nuevo formato utilizando la función *factor()*. 1 es Masculino y 0 es Femenino, por comodidad se decide cambiar los valores de números a letras M y F.
- **cp**: tipo de dolor en el pecho, es una variable de tipo factor que, con valores entre 0 y 3. Se ha importado como integer, se podría dejar así pero lo correcto es convertirla a factor con la función *as.factor()*. 0 representa un paciente sin síntomas, 1 un paciente con síntomas típicos de angina de pecho, 2 un paciente con angina atípico y 3 con dolor no relacionado con angina de pecho.
- **trtbps**: valor de la tensión arterial en reposo, medida en mmHg: Es una variable de tipo integer, se puede ver que los valores son correctos encontrándose entre 94 y 200 mmHg, no hay valores extremos.
- **chol**: valor de colesterol en mg/dl, es una variable de tipo integer, se puede ver que los valores son correctos encontrándose entre 126 y 564. Más adelante se comprobará si hay outliers entre los valores extremos.
- **fbs**: es una variable dicotómica que toma valor 1 si el valor de la glucosa supera los 120 mg/dl y 0 en caso contrario, se ha importado como integer pero se puede convertir a boolean con la función *as.logical()*. No se detectan anomalías con los valores.
- **restecg**: resultados del electrocardiograma. Debería ser una variable categórica con tres valores, 0, 1 y 2 pero se ha importado como variable de tipo integer. Se convierte a factor con la función *as.factor()*.
- **thalachh**: frecuencia cardíaca máxima alcanzada, es una variable numérica. Se ha importado correctamente con valores entre 71 y 202. No se detectan valores extremos.
- **exng**: variable dicotómica toma el valor de 1 si el ejercicio produce angina, 0 en caso contrario. Se ha importado como int se convierte a una variable booleana utilizando la función *as.logical()*.
- **oldpeak**: indica el valor de la depresión del segmento ST del ECG causada por el ejercicio físico. Se ha importado correctamente como número, no se detectan valores extremos.
- **slp**: indica la inclinación del segmento ST del ECG durante el ejercicio, puede tomar tres valores, 2, 1, 0. Se ha importado correctamente como integer.

- **caa**: indica el número de vasos sanguíneos afectados, puede tomar valores de 0 a 4. Se ha importado correctamente como integer.
- **thall**: representa el resultado de la prueba de "thalium stress test" puede tomar los valores de 1 a 3. Es una variable categórica que se ha importado como variable numérica. Se convierte a factor con la función *as.factor()*
- **output**: representa la predicción sobre enfermedades cardiacas, puede asumir valores 0 o 1. Se ha importado como integer pero debería ser una variable categórica, se convierte con la función *as.factor()*. También se podría convertir a variable de tipo booleano.

```
#Se importa el datased
dset <- read.csv(file="heart.csv")
dim(dset)
```

```
## [1] 303  14
```

```
head(dset)
```

```
##   age sex cp trtbps chol fbs restecg thalachh exng oldpeak slp caa thall output
## 1  63   1  3    145  233   1       0      150    0     2.3   0   0     1      1
## 2  37   1  2    130  250   0       1      187    0     3.5   0   0     2      1
## 3  41   0  1    130  204   0       0      172    0     1.4   2   0     2      1
## 4  56   1  1    120  236   0       1      178    0     0.8   2   0     2      1
## 5  57   0  0    120  354   0       1      163    1     0.6   2   0     2      1
## 6  57   1  0    140  192   0       1      148    0     0.4   1   0     1      1
```

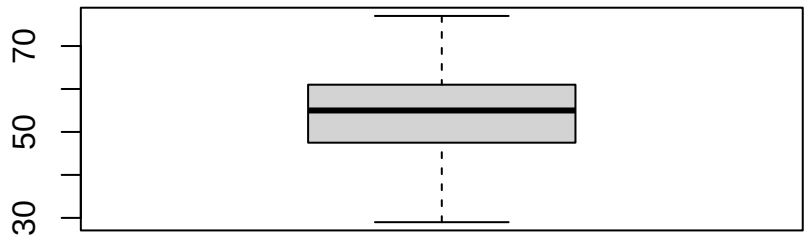
```
#Se comprueba tipo de variables importadas
sapply(dset, class)
```

```
##       age       sex        cp    trtbps      chol       fbs   restecg  thalachh
## "integer" "integer" "integer" "integer" "integer" "integer" "integer" "integer"
##      exng   oldpeak       slp       caa     thall    output
## "integer" "numeric" "integer" "integer" "integer" "integer"
```

```
#Copia del dataset original
dset.clean <- dset
#Comprobación Age
summary(dset.clean$age)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   29.00   47.50   55.00   54.37   61.00   77.00
```

```
boxplot(dset.clean$age)
```



```r
#Comprobación Sex
unique(dset.clean$sex)
```

```
## [1] 1 0
```

```r
dset.clean$sex <- factor(dset.clean$sex,
                         levels=c(0,1),
                         labels=c("F", "M"))
#Comprobación cp
unique(dset.clean$cp)
```

```
## [1] 3 2 1 0
```

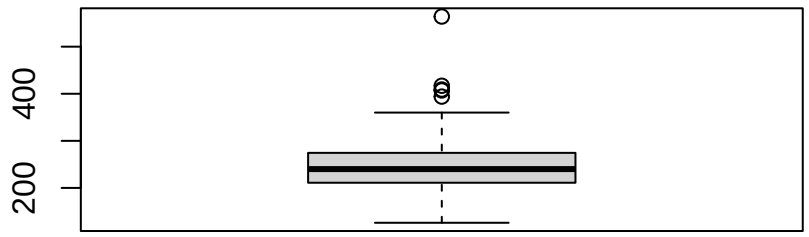
```r
dset.clean$cp <- as.factor(dset.clean$cp)
#Comprobación trtbps
summary(dset.clean$trtbps)
```

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

```r
#Comprobación chol
summary(dset.clean$chol)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   126.0   211.0   240.0   246.3   274.5   564.0
```

```r
boxplot(dset.clean$chol)
```




```r
#Comprobación fbs
unique(dset.clean$fbs)
```

```
## [1] 1 0
```

```r
dset.clean$fbs <- as.logical(dset.clean$fbs)
#Comprobación restecg
unique(dset.clean$restecg)
```

```
## [1] 0 1 2
```

```r
dset.clean$restecg <- as.factor(dset.clean$restecg)
#Comprobación thalachh
summary(dset.clean$thalachh)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##    71.0   133.5   153.0   149.6   166.0   202.0
```

```
#Comprobación exng
unique(dset.clean$exng)
```

```
## [1] 0 1
```

```
dset.clean$exng <- as.logical(dset.clean$exng)
#Comprobación oldpeak
summary(dset.clean$oldpeak)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##    0.00    0.00    0.80    1.04    1.60    6.20
```

```
#Comprobación slope
unique(dset.clean$slp)
```

```
## [1] 0 2 1
```

```
dset.clean$slp <- as.logical(dset.clean$slp)
#Comprobación caa
unique(dset.clean$caa)
```

```
## [1] 0 2 1 3 4
```

```
summary(dset.clean$caa)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##  0.0000  0.0000  0.0000  0.7294  1.0000  4.0000
```

```
#Comprobación thal
unique(dset.clean$thall)
```

```
## [1] 1 2 3 0
```

```
dset.clean$thall <- as.factor(dset.clean$thall)
#Comprobación output
unique(dset.clean$output)
```

```
## [1] 1 0
```

```
dset.clean$output <- as.factor(dset.clean$output)
#Comprobación tipos variables final
sapply(dset.clean, class)
```

```
##       age       sex        cp    trtbps      chol       fbs   restecg  thalachh
## "integer"  "factor"  "factor" "integer" "integer" "logical"  "factor" "integer"
##      exng   oldpeak       slp       caa     thall    output
## "logical" "numeric" "logical" "integer"  "factor"  "factor"
```

2.Limpieza de los datos

Se detecta un valor extremo solamente en la variable *chol*. El valor máximo de 564 mg/dl se encuentra bien por encima del "bigote" superior del diagrama de caja y debería inducirnos a pensar que se trata de un error. Sin embargo el valor no es tan elevado como para no ser un valor plausible así que se considerará como un valor elevado pero real.

Entre las variables numéricas se detectan varios ceros en la variable *oldpeak*. Esta variable describe la depresión del segmento ST en el ECG del paciente durante el ejercicio, el valor cero es un valor normal que indica que no hay depresión en el segmento ST del ECG durante el ejercicio y por tanto se aceptan todos los ceros.

Se detecta una entrada duplicada, se elimina dicha entrada ya que al tener exactamente los mismos datos se trata probablemente de un error en la introducción de los datos.

```
#Se buscan los valores duplicados
#duplicated(dset.clean)
dset.clean[duplicated(dset.clean),]
```

```
##     age sex cp trtbps chol   fbs restecg thalachh  exng oldpeak  slp caa thall
```

```
## 165  38   M  2    138  175 FALSE       1      173 FALSE       0 TRUE   4     2
##     output
## 165      1
```

```
dim(dset.clean)
```

```
## [1] 303  14
```

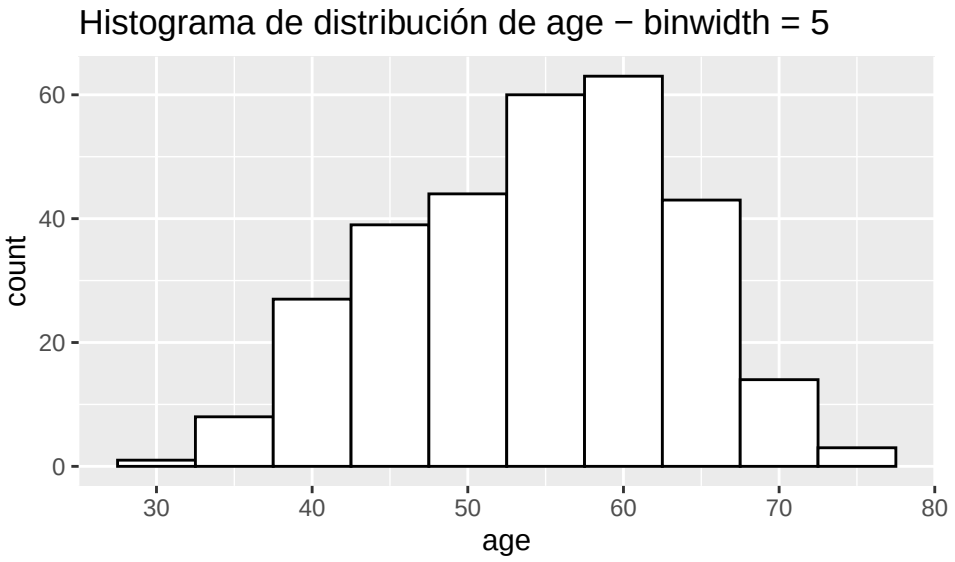
```
dset.clean <- unique(dset.clean)
dim(dset.clean)
```

```
## [1] 302  14
```

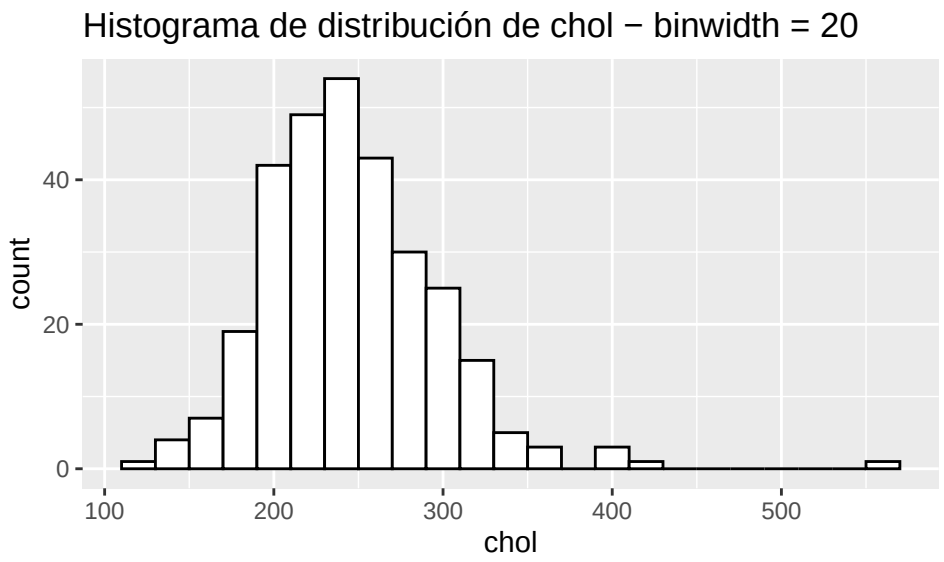
```
#Se buscan los NAs
which(is.na(dset.clean))
```

```
## integer(0)
```

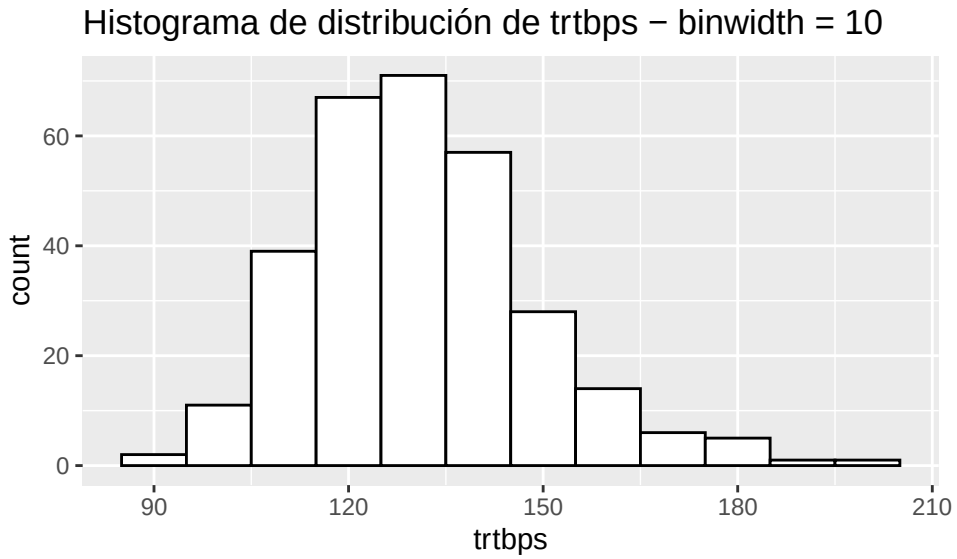
```
#Se visualiza la distribución de algunas variables cuantitativas: age, chol, trtbps
ggplot(dset.clean, aes(x=age)) +
  geom_histogram(binwidth = 5, fill = "white", colour = "black") +
  ggtitle("Histograma de distribución de age - binwidth = 5")
```



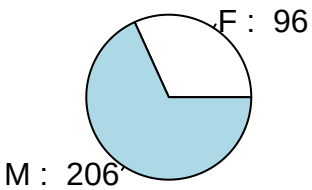

```
#Se detecta una distribución unimodal con asimetría  la izquierda
ggplot(dset.clean, aes(x=chol)) +
  geom_histogram(binwidth = 20, fill = "white", colour = "black") +
  ggtitle("Histograma de distribución de chol - binwidth = 20")
```




```r
#Se detecta una distribución que se aproxima bastante a auna normal,  unimodal,
#con una cola a la derecha bastante larga, con varias clases vacías y outliers a la derecha
ggplot(dset.clean, aes(x=trtbps)) +
  geom_histogram(binwidth = 10, fill = "white", colour = "black") +
  ggtitle("Histograma de distribución de trtbps - binwidth = 10")
```



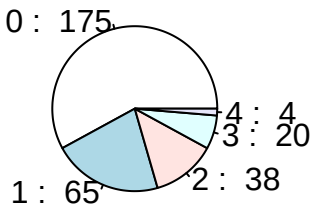

```r
#Se detecta una distribución que se aproxima bastante a auna normal,  unimodal,
#asimétrica con cola a la derecha

#Se visualiza la distribución de algunas variables cualitativas
#Se puede ver que hay una mayoría de sujetos de sexo masculino
pie(table(dset.clean$sex), labels=paste(c("F", "M"), ": ", table(dset.clean$sex)))
```




```
table(dset.clean$sex)
```

```
##
##   F   M
##  96 206
```

```
#Se puede ver que la mayoría de los pacientes no presenta vasos sanguinéos afectados
pie(table(dset.clean$caa), labels=paste(c("0", "1", "2", "3", "4"), ": ",
                                        table(dset.clean$caa)))
```




```
table(dset.clean$caa)
```

```
##
##   0   1   2   3   4
## 175  65  38  20   4
```

3.Integración y selección

Se crean algunas subselecciones de los datos de grupos de pacientes que permiten hacer análisis específicas: Pacientes de edad superior o igual a 50 años (*dset.clean.over50*) y pacientes más jóvenes(*dset.clean.under50*). Pacientes de sexo masculino (*dset.clean.M*) y pacientes de sexo femenino (*dset.clean.F*).

Asimismo se crea una nueva variable cualitativa (discretización) que identifica a los pacientes con más de 50 años, es la variable booleana U50, TRUE si el paciente tiene 50 o menos años, esta variable es útil para poder trabajar con el mismo

dataset. Se crea también una variable cualitativa (discretización) con 3 niveles para el nivel de colesterol, N (normal) si está por debajo de 200, H (high) entre 200 y 240 y R (risk) por encima de 240.

```
dset.clean.over50 <- subset(dset.clean, age >= 50)
dset.clean.under50 <- subset(dset.clean, age < 50)
summary(dset.clean.over50$age)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   50.00   54.00   58.00   59.07   63.00   77.00
```

```
summary(dset.clean.under50$age)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   29.00   41.00   43.00   42.93   46.00   49.00
```

```
dset.clean$U50 <- with(dset.clean, ifelse(age < 50, TRUE, FALSE))
dset.clean$U50 <- as.factor(dset.clean$U50)

dset.clean$levchol <- as.factor(with(dset.clean, ifelse(chol <= 200, "N",
                                          ifelse(chol <= 240, "H", "R"))))

dset.clean.M <- subset(dset.clean, sex == "M")
dset.clean.F <- subset(dset.clean, sex == "F")
summary(dset.clean.M$sex)
```

```
##   F   M
##   0 206
```

```
summary(dset.clean.F$sex)
```

```
##  F  M
## 96  0
```

4.Análisis de los datos

4.a Contraste de hipótesis sobre la media

El primer análisis que se hace es sobre los niveles de colesterol, las preguntas de investigación son:

1. Los pacientes de sexo masculino presentan valores de colesterol significativamente diferentes que las pacientes de sexo femenino?
2. Los pacientes (de ambos sexos) con menos de 50 años presentan niveles de colesterol significativamente diferentes a los pacientes con más de 50 años?

Los 4 grupos tienen más de 30 elementos cada uno, por tanto se puede considerar que por el teorema del límite central el contraste de hipótesis sobre la media de una muestra se aproxima a una distribución normal aunque la población original no siga una distribución normal. La varianza de la variable "chol" de las poblaciones es desconocida pero se puede comprobar si hay homogeneidad de varianzas entre las parejas a comparar utilizando la prueba de Levene *leveneTest()*.

Se comprueba la homogeneidad de la varianza entre grupos de pacientes de más y menos de 50 años y entre pacientes de sexo Masculino y Femenino utilizando la prueba de Levene. Considerando un nivel de significancia $\alpha = 0.05$, podemos ver que:

- el p-value en el caso de la prueba entre grupos de edades es de 0.2423, un valor superior a α, por tanto tenemos que aceptar la hipótesis nula y considerar que las varianzas son iguales.
- el p-value en el caso de la prueba entre sexos es de 0.0007684, un valor inferior a α, por tanto tenemos que rechazar la hipótesis nula y considerar que las varianzas no son iguales.

```
summary(dset.clean.under50$chol)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   141.0   203.5   231.0   232.9   261.5   341.0
```

```
length((dset.clean.under50$chol))
```

```
## [1] 87
```

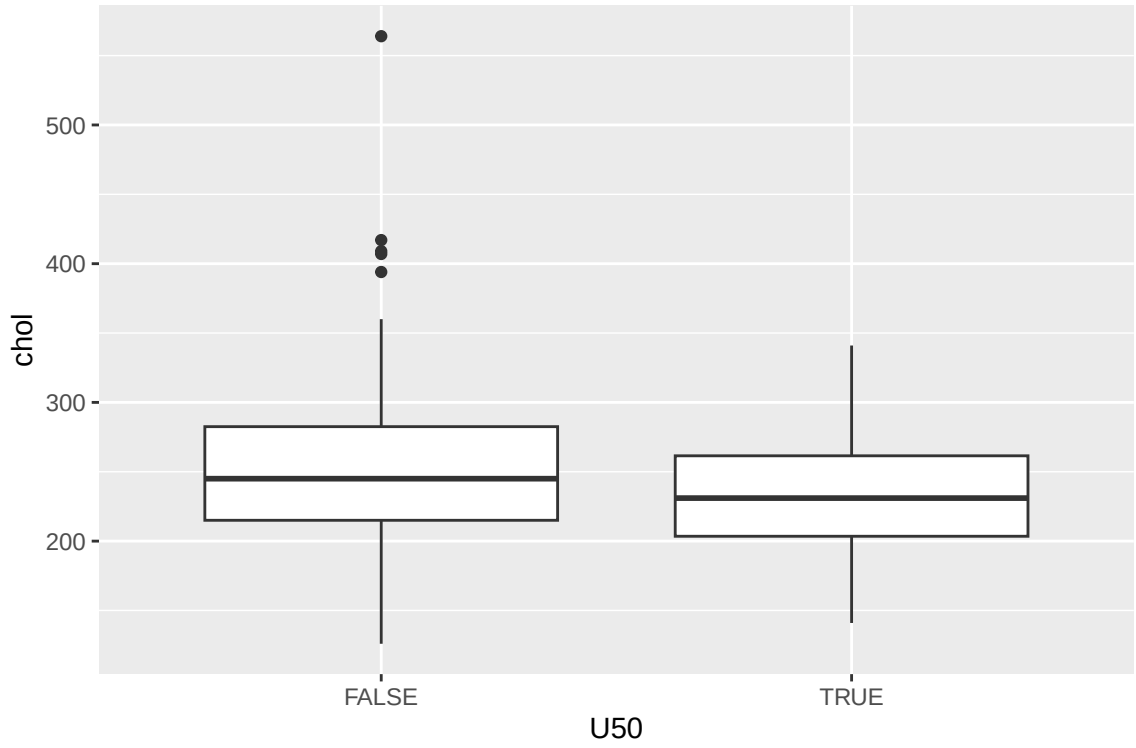
```
length((dset.clean.over50$chol))
```

```
## [1] 215
```

```
leveneTest(chol ~ U50, data=dset.clean)
```

```
## Levene's Test for Homogeneity of Variance (center = median)
##       Df F value Pr(>F)
## group  1  1.3725 0.2423
##       300
```

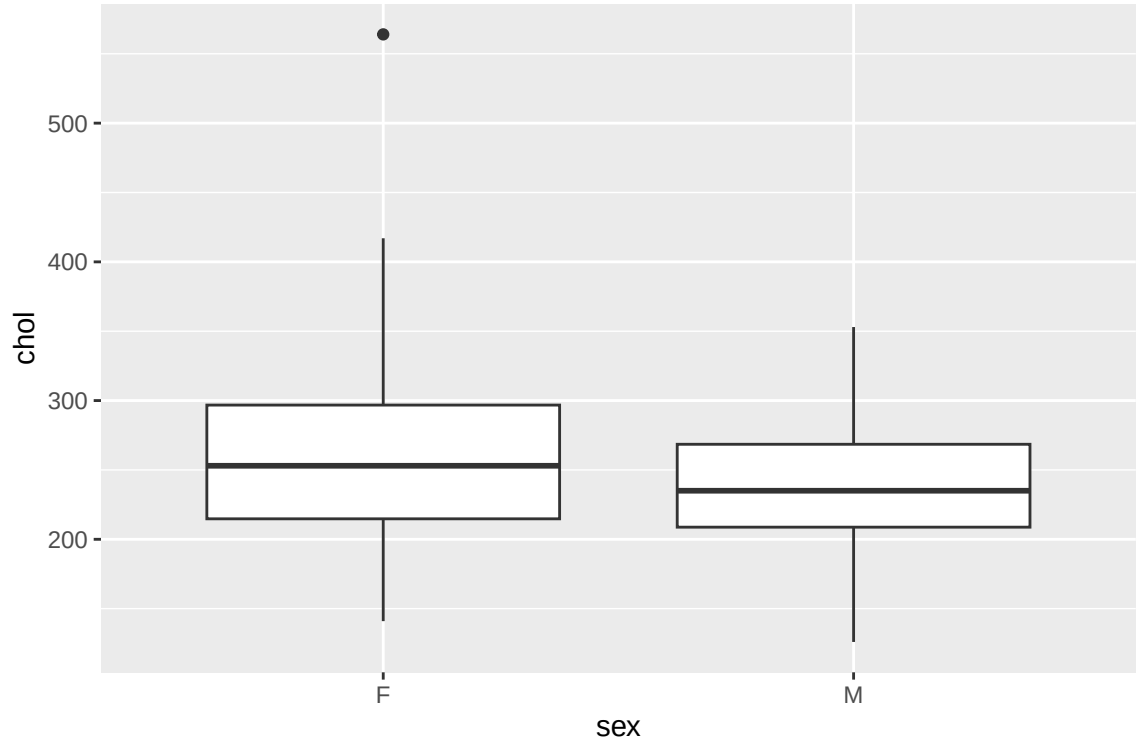
```
ggplot(dset.clean, aes(x = U50, y=chol))+
  geom_boxplot()
```




```
leveneTest(chol ~ sex, data=dset.clean)
```

```
## Levene's Test for Homogeneity of Variance (center = median)
##       Df F value    Pr(>F)
## group  1  11.552 0.0007684 ***
##       300
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
ggplot(dset.clean, aes(x = sex, y=chol))+
  geom_boxplot()
```



Una vez comprobada la normalidad (con el teorema del límite central) y la homoscedasticidad de las variables podemos proceder con los contrastes de hipótesis para responder a las preguntas de investigación.

En el caso de varianzas iguales se utilizará un T-Test (prueba t de Student) mientras que para variables con varianzas no iguales se utilizará el T-Test de Welch.

Considerando un valor de significancia $\alpha = 0.05$ y observando los resultados de las pruebas podemos observar que:

- hay una diferencia significativa entre los niveles de colesterol en la población de sexo masculino y de sexo femenino. El p-value es de 0.003403, por tanto hay que rechazar la hipótesis nula de igualdad de niveles de colesterol.
- hay una diferencia significativa entre los niveles de colesterol en la población de referencia de edad inferior o superior a 50 años. El p-value es 0.00351, por tanto hay que rechzar la hipótesis nula de igualdad de niveles de colesterol.

```
t.test(dset.clean$chol[dset.clean$sex=="M"], dset.clean$chol[dset.clean$sex=="F"],
       alternative = "two.sided", var.equal=FALSE)
```

```
##
##  Welch Two Sample t-test
##
## data:  dset.clean$chol[dset.clean$sex == "M"] and dset.clean$chol[dset.clean$sex == "F"]
## t = -2.9818, df = 134.33, p-value = 0.003403
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -36.093319  -7.306964
## sample estimates:
## mean of x mean of y
##  239.6019  261.3021
```

```
t.test(dset.clean$chol[dset.clean$U50==TRUE], dset.clean$chol[dset.clean$U50==FALSE],
       alternative = "two.sided", var.equal=TRUE)
```

```
##
##  Two Sample t-test
##
## data:  dset.clean$chol[dset.clean$U50 == TRUE] and dset.clean$chol[dset.clean$U50 == FALSE]
## t = -2.9424, df = 300, p-value = 0.00351
## alternative hypothesis: true difference in means is not equal to 0
```

```
## 95 percent confidence interval:
##  -31.887557  -6.328642
## sample estimates:
## mean of x mean of y
##  232.8966  252.0047
```

4.b Correlación entre variables cuantitativas

En este caso se quiere determinar si existe algún tipo de correlación entre variables cuantitativas:

- Edad y nivel de colesterol. ¿El nivel de colesterol aumenta con el aumentar de la edad?
- Edad y tensión arterial. ¿La tensión arterial aumenta con el aumentar de la edad?
- Nivel de colesterol y tensión arterial. ¿La tensión arterial aumenta con el aumentar del nivel de colesterol?

Se lleva a cabo el test de normalidad de Shapiro-Wilk para las 3 variables objeto de estudio, el resultado del test obliga a rechazar la hipótesis de normalidad para las tres variables. Será necesario, por tanto, llevar a cabo una prueba de correlación no paramétrica, por esta razón se prefiere la prueba de Spearman a la de Pearson, ya que la primera no conlleva ninguna suposición sobre la distribución de los datos.

Asimismo se visualiza con un gráfico de tipo scatterplot la distribución de las variables, parece evidente que no hay ninguna correlación entre ellas. El resultado de las pruebas de correlación de Spearman confirma este primer resultado, los valores son siempre bajos e indican una correlación muy débil:

- Edad y nivel de colesterol: 0.2897
- Edad y tensión arterial: 0.1889
- Nivel de colesterol y tensión arterial: 0.1302

```
shapiro.test(dset.clean$age)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  dset.clean$age
## W = 0.98664, p-value = 0.006745
```

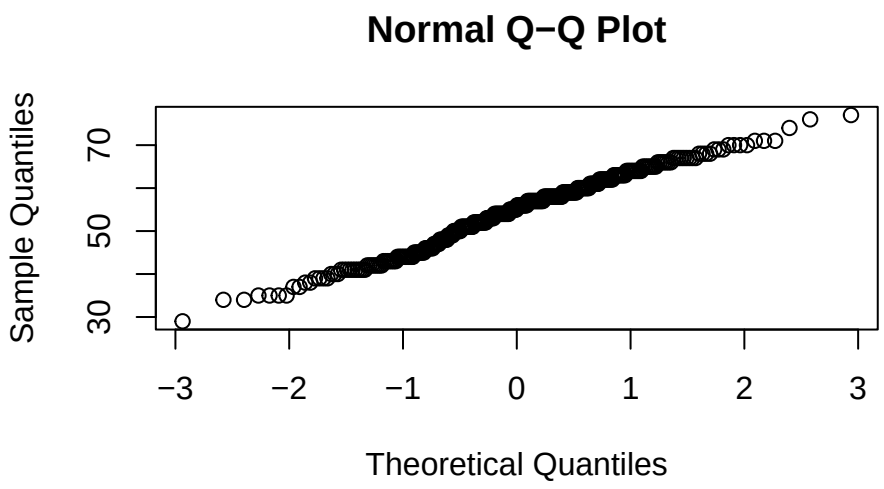
```
shapiro.test(dset.clean$chol)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  dset.clean$chol
## W = 0.94658, p-value = 5.196e-09
```

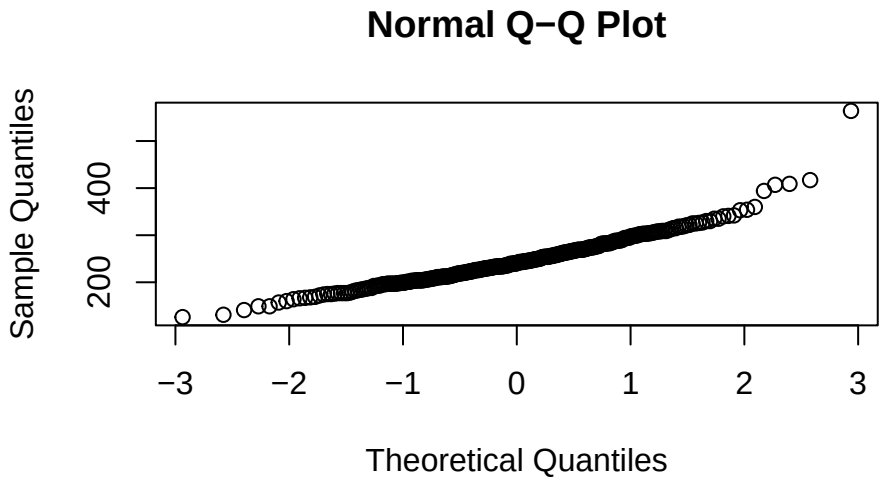
```
shapiro.test(dset.clean$trtbps)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  dset.clean$trtbps
## W = 0.96573, p-value = 1.419e-06
```

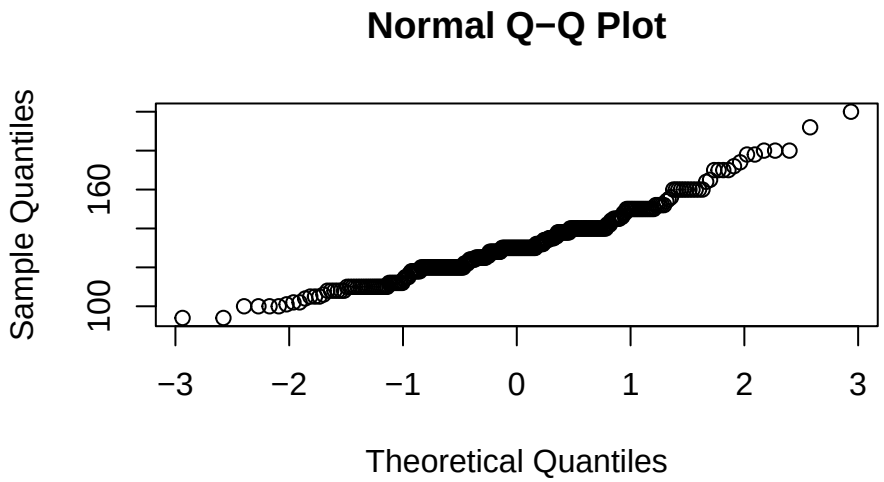
```
qqnorm(dset.clean$age, pch = 1)
```



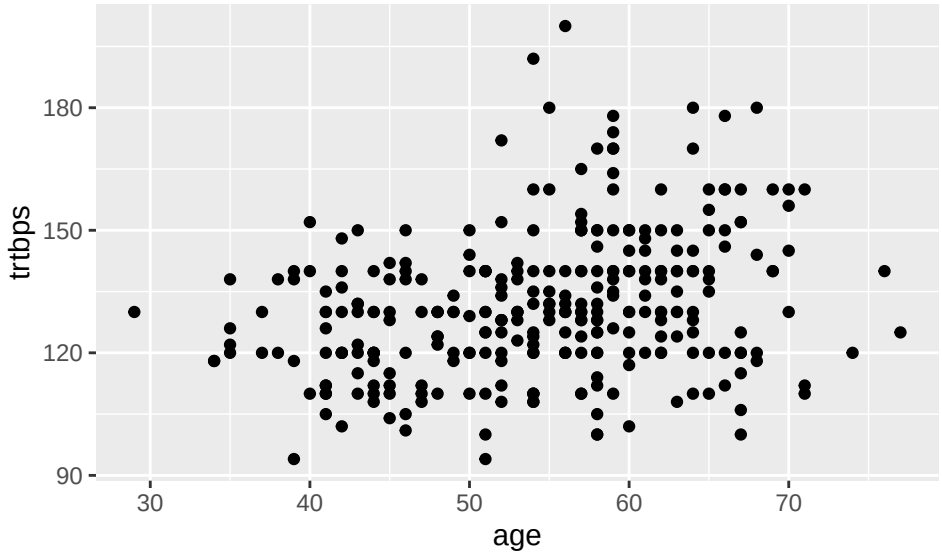

```
qqnorm(dset.clean$chol, pch = 1)
```




```
qqnorm(dset.clean$trtbps, pch = 1)
```



```
ggplot(dset.clean, aes(x=age, y=trtbps))+
  geom_point()
```




```
cor(dset.clean$age, dset.clean$trtbps, method="spearman")
```

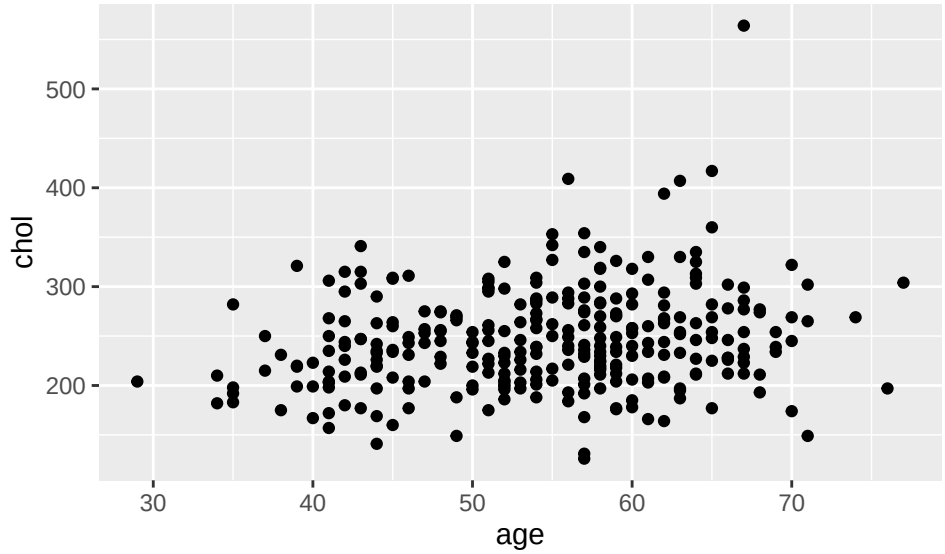
```
## [1] 0.289705
```

```
cor.test(dset.clean$age, dset.clean$trtbps, method="spearman")
```

```
## Warning in cor.test.default(dset.clean$age, dset.clean$trtbps, method =
## "spearman"): Cannot compute exact p-value with ties
```

```
##
##  Spearman's rank correlation rho
##
## data:  dset.clean$age and dset.clean$trtbps
## S = 3260645, p-value = 2.992e-07
## alternative hypothesis: true rho is not equal to 0
## sample estimates:
##      rho
## 0.289705
```

```
ggplot(dset.clean, aes(x=age, y=chol))+
  geom_point()
```




```
cor(dset.clean$age, dset.clean$chol, method="spearman")
```

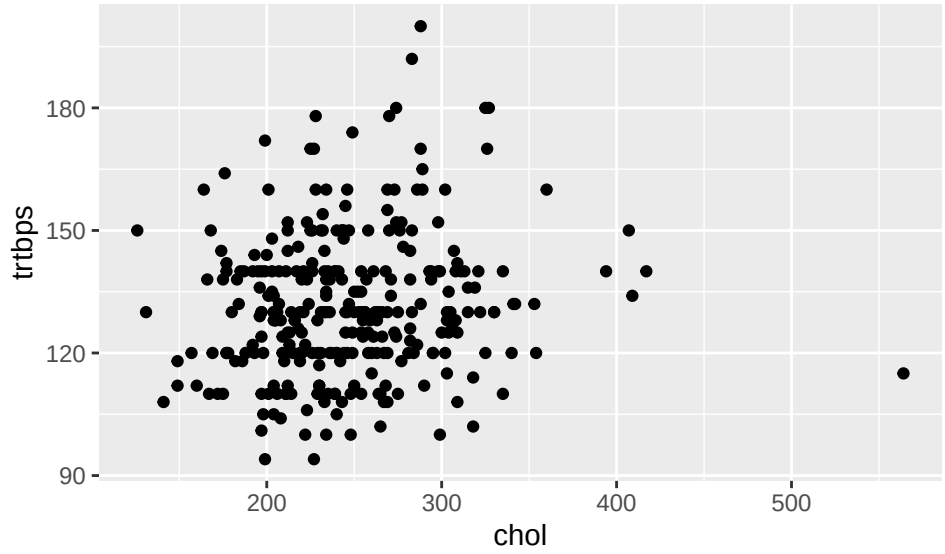
```
## [1] 0.1889029
```

```
cor.test(dset.clean$age, dset.clean$chol, method="spearman")
```

```
## Warning in cor.test.default(dset.clean$age, dset.clean$chol, method =
## "spearman"): Cannot compute exact p-value with ties
```

```
##
##  Spearman's rank correlation rho
##
## data:  dset.clean$age and dset.clean$chol
## S = 3723383, p-value = 0.0009706
## alternative hypothesis: true rho is not equal to 0
## sample estimates:
##       rho
## 0.1889029
```

```
ggplot(dset.clean, aes(x=chol, y=trtbps))+
  geom_point()
```




```
cor(dset.clean$chol, dset.clean$trtbps, method="spearman")
```

```
## [1] 0.1302102
```

```
cor.test(dset.clean$chol, dset.clean$trtbps, method="spearman")
```

```
## Warning in cor.test.default(dset.clean$chol, dset.clean$trtbps, method =
## "spearman"): Cannot compute exact p-value with ties
```

```
##
##  Spearman's rank correlation rho
##
## data:  dset.clean$chol and dset.clean$trtbps
## S = 3992814, p-value = 0.02363
## alternative hypothesis: true rho is not equal to 0
## sample estimates:
##       rho
## 0.1302102
```

4.c Correlación entre variables categoricas

En este caso se quiere determinar si existe alguna relación entre el sexo del paciente y tipo de dolor en el pecho que presenta. Se recuerda que el tipo de dolor puede asumir 4 valores, entre 0,1,2,3. Estos valores no se consideran como números, si no más bien como niveles, es decir como una variable cualitativa.

Para determinar si existe algún tipo de relación entre las dos variables se utiliza el coeficiente V de Cramér. Para ello primero es necesario crear una tabla de contingencia utilizando la función *table()* con las dos variables *sex* y *cp*. Luego se puede utilizar la función *cramerV()* para calcular el coeficiente V de Cramér, este coeficiente oscila entre 0 (independencia) y 1. Se puede ver que el resultado es un número bajo, 0,1533 por tanto se puede asumir que no hay correlación entre el sexo y el tipo dolor en el pecho del paciente.

Se comprueba si existe también algún tipo de correlación entre la variable *sex* y la variable *output*, es lógico que exista algún tipo de correlación ya que la variable *output* es el resultado de una predicción basada en los datos disponibles. El resultado del coeficiente de Cramér es 0.2836. Indíca que hay una correlación débil entre el sexo y el resultado de la predicción sobre el riesgo cardíaco.

Se quiere comprobar también si existe correlación entre niveles altos o muy altos de colesterol y el valor de la variable output. El coeficiente V de Cramér para esta correlación es muy bajo así que no se puede decir que exista correlación entre las dos variables.

```
table.sex.cp <- table(dset.clean$sex, dset.clean$cp)
table.sex.cp
```

```
##
##       0   1   2   3
##   F  39  18  35   4
##   M 104  32  51  19
```

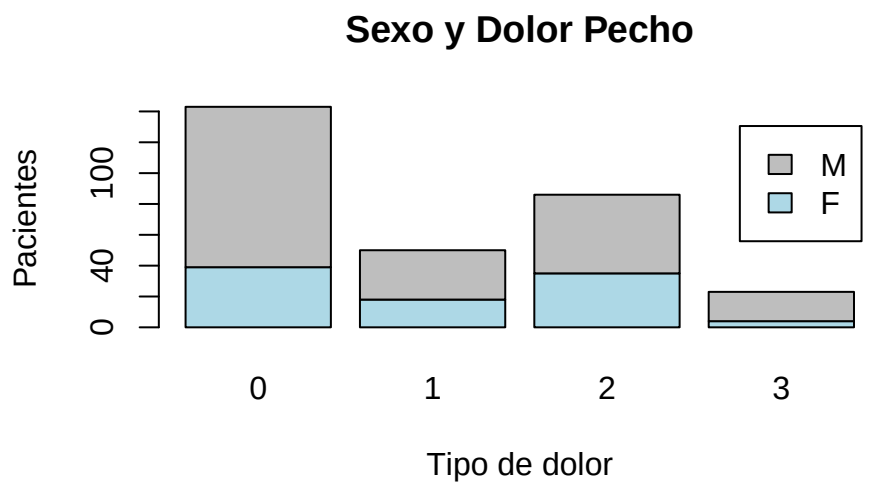
```
prop.table.sex.cp <- prop.table(table.sex.cp)
prop.table.sex.cp
```

```
##
##              0          1          2          3
##   F 0.12913907 0.05960265 0.11589404 0.01324503
##   M 0.34437086 0.10596026 0.16887417 0.06291391
```

```
cramerV(table.sex.cp)
```

```
## Cramer V
##   0.1533
```

```
barplot(table.sex.cp, main = "Sexo y Dolor Pecho", xlab= "Tipo de dolor",
        ylab="Pacientes", col = c("lightblue", "gray"), legend.text = TRUE )
```



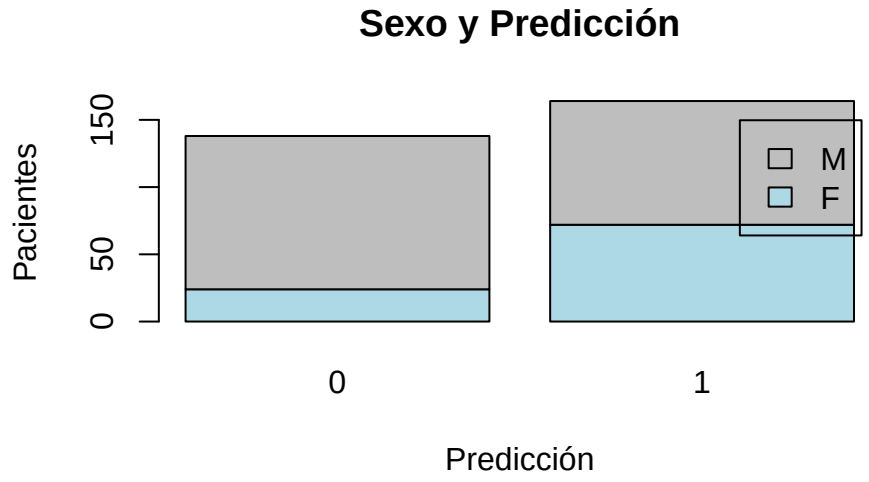

```
table.sex.output <- table(dset.clean$sex, dset.clean$output)
table.sex.output
```

```
##
##       0   1
##   F  24  72
##   M 114  92
```

```
cramerV(table.sex.output)
```

```
## Cramer V
##   0.2836
```

```
barplot(table.sex.output, main = "Sexo y Predicción", xlab= "Predicción",
        ylab="Pacientes", col = c("lightblue", "gray"), legend.text = TRUE )
```




```
table.levchol.output <- table(dset.clean$levchol, dset.clean$output)
table.levchol.output
```

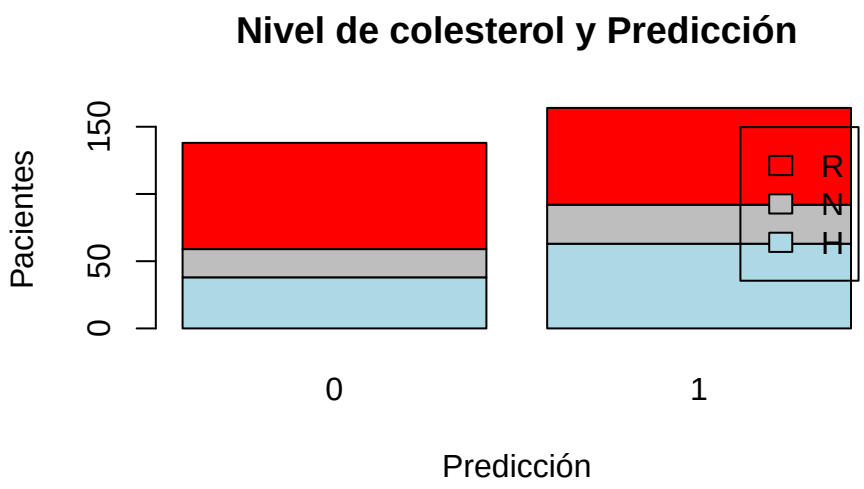
```
##
##      0  1
##   H 38 63
##   N 21 29
```

```
##   R 79 72
```

```
cramerV(table.levchol.output)
```

```
## Cramer V
##   0.1361
```

```
barplot(table.levchol.output, main = "Nivel de colesterol y Predicción",
        xlab= "Predicción", ylab="Pacientes", col = c("lightblue", "gray", "red"),
        legend.text = TRUE )
```



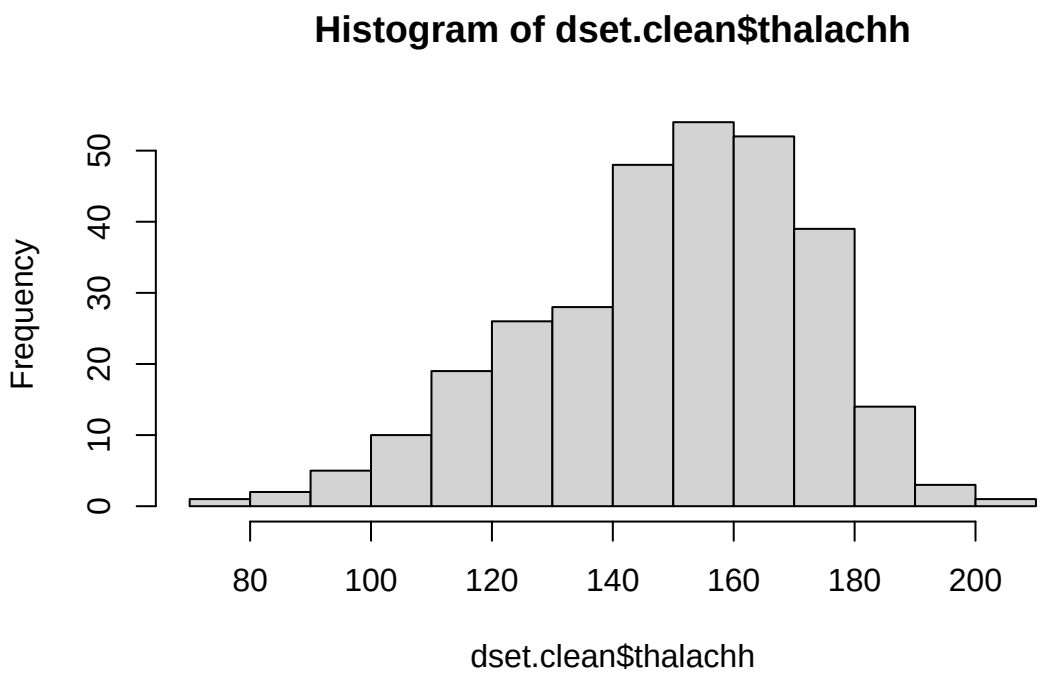
4.d Comparación entre más de dos grupos

En este apartado se quiere comparar la media de la variable *thalachh* (frecuencia máxima alcanzada) entre los grupos de personas con colesterol normal (N), alto (H) o muy alto (R). Se puede comprobar que la frecuencia máxima no sigue una distribución normal ya que el test de Shapiro-Wilk da un valor p-value inferior a 0,05 que es el nivel de significancia establecido. Sin embargo con el test de Flighner-Killeen se puede establecer que hay homogeneidad de varianzas. En este caso se utiliza el test de Kruskal-Wallis para establecer si hay diferencia significativa entre las medias de la variable *thalachh* y las 3 categorías de pacientes por nivel de colesterol en la sangre. Considerando el resultado del test, p-value = 0.1496, no es posible rechazar la hipótesis nula y por tanto no se puede afirmar que hay diferencias significativas en la frecuencia máxima alcanzada dependiendo de los niveles de colesterol.

```
#Test de normalidad Shapiro-Wilk
shapiro.test(dset.clean$thalachh)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  dset.clean$thalachh
## W = 0.97679, p-value = 8.268e-05
```

```
hist(dset.clean$thalachh)
```




```
#Test de homogeneidad de varianzas de Fligner-Killeen
fligner.test(thalachh ~ levchol, data=dset.clean)
```

```
##
##  Fligner-Killeen test of homogeneity of variances
##
## data:  thalachh by levchol
## Fligner-Killeen:med chi-squared = 3.9714, df = 2, p-value = 0.1373
```

```
#Test de contraste de hipótesis entre más de dos grupos de Kruskal-Wallis
kruskal.test(thalachh ~ levchol, data=dset.clean)
```

```
##
##  Kruskal-Wallis rank sum test
##
## data:  thalachh by levchol
## Kruskal-Wallis chi-squared = 3.7989, df = 2, p-value = 0.1496
```

4.e Regresión logística

En este apartado se quiere estimar un modelo de regresión logística para predecir el valor de la variable *output* y comprobar cuales variables son realmente significativas para la predicción del resultado. Los pasos a seguir serán los siguientes: - **1**: Generación de los conjuntos de entrenamiento y de test. Se utiliza el 80% de los datos para el entrenamiento y el 20% para el testing. - **2**: Estimación del modelo con el conjunto de entrenamiento, interpretación de las variables significativas. Es curioso ver como la edad no parece ser una variable significativa para la predicción, así como la tensión arterial en reposo, el valor de la glucosa (fbs), los resultados del ECG (restecg), la inclinación del segmento ST del ECT (slp), el resultado del thalium stress test (thall). - **3**: Estimación del modelo con el conjunto de entrenamiento y las variables significativas. No hay colinealidad entre las variables seleccionadas. - **4**: Cálculo de las OR - **5**: Predicción con el conjunto de test y cálculo de la matriz de confusión. En la matriz de confusión se puede ver que la sensibilidad (proporción de casos positivos correctamente clasificados) del modelo es de 0.7083 mientras que la especificidad (proporción de casos negativos correctamente clasificados) es de 0.84. - **6**: Estimación de la curva ROC y del área debajo de la curva. El AUC=0.907, un valor alto que indica que el modelo discrimina muy bien. - **7**: Evaluación de la bondad del ajuste con el test Chi-cuadrado. A la vista de los resultados el ajuste es bueno con un p-value de 0.

```
#Generación de los dos conjuntos de entrenamiento (80%) y de testing (%20)
set.seed(1)
```

```r
sample <- sample(c(TRUE, FALSE), nrow(dset.clean), replace = TRUE, prob=c(0.8,0.2))
dset.train <- dset.clean[sample,]
dset.test <- dset.clean[!sample,]

#Se genera el modelo de regresión logística utilizando el conjunto de entrenamiento
dset.model <- glm(formula = output ~ age + sex + cp + trtbps + chol + fbs +
                    restecg + thalachh + exng + oldpeak + slp + caa + thall,
                  family = binomial (link=logit), data = dset.train)

summary(dset.model)
```

```
##
## Call:
## glm(formula = output ~ age + sex + cp + trtbps + chol + fbs +
##     restecg + thalachh + exng + oldpeak + slp + caa + thall,
##     family = binomial(link = logit), data = dset.train)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.4622  -0.3631   0.1601   0.4513   2.7121
##
## Coefficients:
##               Estimate Std. Error z value Pr(>|z|)
## (Intercept)  0.6690900  3.3546276   0.199  0.84191
## age          0.0002624  0.0270780   0.010  0.99227
## sexM        -1.1256781  0.5584642  -2.016  0.04383 *
## cp1          1.0913935  0.6291837   1.735  0.08281 .
## cp2          2.0294688  0.5526179   3.672  0.00024 ***
## cp3          1.4598052  0.6754945   2.161  0.03069 *
## trtbps      -0.0175863  0.0121467  -1.448  0.14766
## chol        -0.0077070  0.0044567  -1.729  0.08376 .
## fbsTRUE      0.1805915  0.6591913   0.274  0.78412
## restecg1     0.2214232  0.4245681   0.522  0.60200
## restecg2    -0.3467492  2.6234842  -0.132  0.89485
## thalachh     0.0278078  0.0116567   2.386  0.01705 *
## exngTRUE    -1.1408631  0.4804933  -2.374  0.01758 *
## oldpeak     -0.5192505  0.2572749  -2.018  0.04356 *
## slpTRUE     -0.4041633  1.0506471  -0.385  0.70047
## caa         -0.8897908  0.2250697  -3.953  7.7e-05 ***
## thall1       0.8479966  2.1044895   0.403  0.68699
## thall2       1.7667098  1.9572617   0.903  0.36672
## thall3       0.4630670  1.9768595   0.234  0.81480
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 348.26  on 252  degrees of freedom
## Residual deviance: 164.44  on 234  degrees of freedom
## AIC: 202.44
##
## Number of Fisher Scoring iterations: 6
```

```r
#Se puede ver que algunas de las variables no son significativas, se pueden por tanto excluir del modelo
dset.model1 <- glm(formula = output ~ sex + cp + chol + thalachh + exng +
                     oldpeak + caa, family = binomial (link=logit), data = dset.train)

summary(dset.model1)
```

```
##
## Call:
## glm(formula = output ~ sex + cp + chol + thalachh + exng + oldpeak +
##     caa, family = binomial(link = logit), data = dset.train)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.5566  -0.4088   0.1773   0.5505   2.6840
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept) -0.290688   1.749898  -0.166  0.86806
## sexM        -1.633909   0.482050  -3.390  0.00070 ***
## cp1          1.212814   0.589206   2.058  0.03955 *
## cp2          2.117707   0.504863   4.195 2.73e-05 ***
## cp3          1.388044   0.635831   2.183  0.02903 *
## chol        -0.010578   0.003992  -2.650  0.00805 **
## thalachh     0.032705   0.010373   3.153  0.00162 **
## exngTRUE    -1.296619   0.451815  -2.870  0.00411 **
## oldpeak     -0.590494   0.215417  -2.741  0.00612 **
## caa         -0.887094   0.203372  -4.362 1.29e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 348.26  on 252  degrees of freedom
## Residual deviance: 176.84  on 243  degrees of freedom
## AIC: 196.84
##
## Number of Fisher Scoring iterations: 6
```

```
#Se calcula la colinealidad del modelo, no se aprecia colinealidad entre las variables
vif(dset.model1)
```

```
##              GVIF Df GVIF^(1/(2*Df))
## sex      1.271281  1        1.127511
## cp       1.412940  3        1.059304
## chol     1.226755  1        1.107590
## thalachh 1.172055  1        1.082615
## exng     1.159798  1        1.076939
## oldpeak  1.205971  1        1.098167
## caa      1.134915  1        1.065324
```

```
#Cálculo de las Odds-Ratio
dset.model1.OR <- exp(coefficients(dset.model1))
dset.model1.OR
```

```
## (Intercept)        sexM         cp1         cp2         cp3        chol
##   0.7477487   0.1951652   3.3629344   8.3120590   4.0070066   0.9894779
##    thalachh    exngTRUE     oldpeak         caa
##   1.0332460   0.2734548   0.5540538   0.4118510
```

```
#Se calcula la predicción de la probabilidad ajustada de output con los datos del conjunto de testing
dset.model1.predicted <- predict(dset.model1, dset.test, type="response")
#Se define un valor límite para discriminar entre pacientes de riesgo y no
threshold <- 0.5
#Se asigna dicho valor a una variable dicotómica
dset.model1.predicted.class <- as.factor(ifelse(dset.model1.predicted>threshold,1,0))
dset.model1.predicted.class
```

```
##   4   6   7  18  21  29  35  41  52  61  70  72  76  77  80  94  99 104 109 111
```

```
##   1   1   1   1   1   1   0   1   0   1   1   1   1   1   1   1   1   1   1   1
## 121 135 139 150 162 166 173 174 177 181 184 186 190 195 199 212 215 216 219 220
##   0   1   0   1   1   0   1   0   0   0   1   0   1   1   0   0   0   0   0   0
## 226 231 244 251 253 261 284 294 301
##   0   1   0   0   0   0   1   1   0
## Levels: 0 1
```

```
#Se genera un dataframe con la variable dicotómica ajustada y la observada
performance.data <- data.frame(observed=dset.test$output,
                               predicted=dset.model1.predicted.class)

#Se calcula la matriz de confusión
performance.data.cm <- confusionMatrix(data=performance.data$predicted, reference=performance.data$observed)
performance.data.cm
```

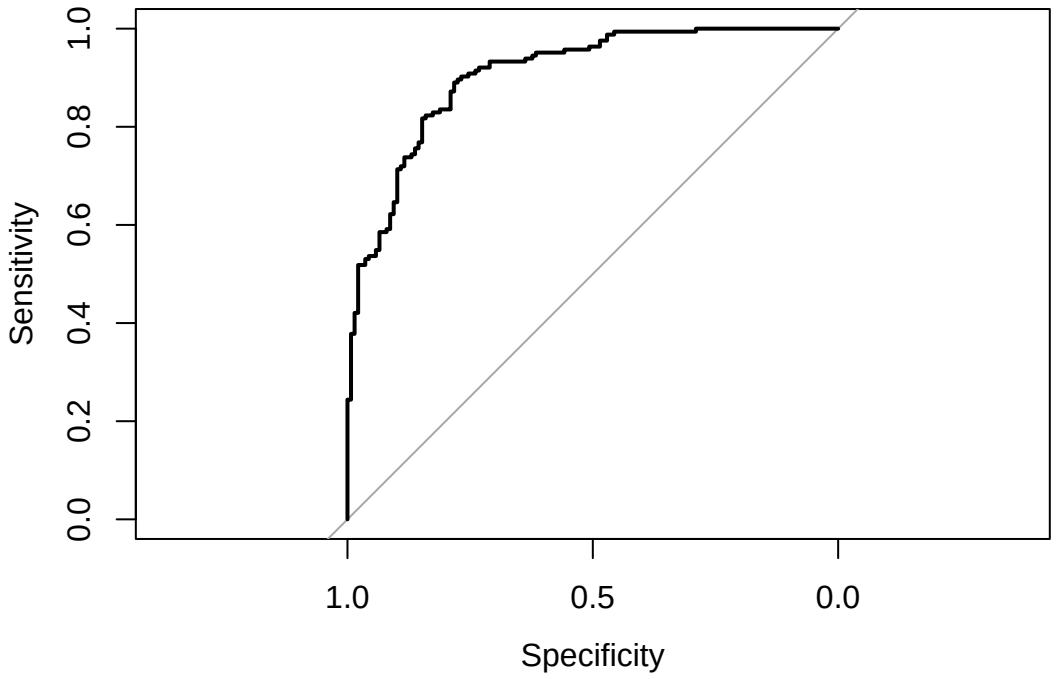
```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction  0  1
##          0 17  4
##          1  7 21
##
##                Accuracy : 0.7755
##                  95% CI : (0.6338, 0.8823)
##     No Information Rate : 0.5102
##     P-Value [Acc > NIR] : 0.0001237
##
##                   Kappa : 0.5497
##
##  Mcnemar's Test P-Value : 0.5464936
##
##             Sensitivity : 0.7083
##             Specificity : 0.8400
##          Pos Pred Value : 0.8095
##          Neg Pred Value : 0.7500
##              Prevalence : 0.4898
##          Detection Rate : 0.3469
##    Detection Prevalence : 0.4286
##       Balanced Accuracy : 0.7742
##
##        'Positive' Class : 0
##
```

```
#Se estima la curva ROC y el área debajo de la curva
dset.model1.predicted.total <- predict(dset.model1, dset.clean, type="response")
curva.roc <- roc(dset.clean$output, dset.model1.predicted.total)
```

```
## Setting levels: control = 0, case = 1
```

```
## Setting direction: controls < cases
```

```
plot(curva.roc)
```




```
auc(curva.roc)
```

```
## Area under the curve: 0.907
```

```
dset.model1
```

```
##
## Call:  glm(formula = output ~ sex + cp + chol + thalachh + exng + oldpeak +
##     caa, family = binomial(link = logit), data = dset.train)
##
## Coefficients:
## (Intercept)         sexM          cp1          cp2          cp3         chol
##    -0.29069     -1.63391      1.21281      2.11771      1.38804     -0.01058
##    thalachh     exngTRUE      oldpeak          caa
##     0.03271     -1.29662     -0.59049     -0.88709
##
## Degrees of Freedom: 252 Total (i.e. Null);  243 Residual
## Null Deviance:       348.3
## Residual Deviance: 176.8     AIC: 196.8
```

```
model1.sumresiduals <- sum(residuals(dset.model1, type="pearson")^2)
model1.null <- dset.model1$df.null
model1.residual <- dset.model1$df.residual

dset.model1.pchisq <- 1-pchisq(model1.sumresiduals, model1.null-model1.residual)
dset.model1.pchisq
```

```
## [1] 0
```

6. Resolución del problema

Los resultados obtenidos a partir de la información disponible son los siguientes:

1. Hay una diferencia significativa entre los niveles de colesterol entre la población de sexo masculino y femenino.
2. Hay una diferencia significativa entre los niveles de colesterol entre la población con más de 50 años y con menos de 50 años.
3. Hay una correlación muy débil entre la edad y el nivel de colesterol del paciente

4. No hay correlación entre la edad y la tensión arterial
5. No hay correlación entre el nivel de colesterol y la tensión arterial
6. No hay relación entre el sexo y el tipo de dolor en el pecho del paciente
7. No hay relación significativa entre el sexo del paciente y la variable output
8. No hay relación entre el nivel de colesterol y la variable output
9. No hay diferencias significativas entre las medias de la frecuencia cardiaca alcanzada y las categorías de nivel de colesterol (normal, alto, muy alto)
10. Es posible calcular un buen modelo predictivo de regresión logística basado en algunas de las variables disponibles.

```
#Genera el csv del nuevo dataset
write.csv(dset.clean, "dset_clean.csv")
```

7. Contribuciones

```
Contribuciones <- c("Investigación previa", "Redacción de las respuestas",
                    "Desarrollo del Código", "Participación en el vídeo")
Firma <- c("MR", "MR", "MR", "MR")

tasks <- data.frame(Contribuciones, Firma)
tasks
```

```
##                Contribuciones Firma
## 1        Investigación previa    MR
## 2 Redacción de las respuestas    MR
## 3       Desarrollo del Código    MR
## 4   Participación en el vídeo    MR
```

```
knitr::kable(tasks, "pipe")
```

Contribuciones	Firma
Investigación previa	MR
Redacción de las respuestas	MR
Desarrollo del Código	MR
Participación en el vídeo	MR