Tipología y ciclo de vida de los datos - Práctica 2

Alfredo Rubio Navarro

2/6/2020

1. Descripción del dataset.

¿Por qué es importante y qué pregunta/problema pretende responder?

El dataset elegido para la práctica estudia la presencia o ausencia de enfermedad cardiaca basándose en los datos de 1025 pacientes de cuatro centros hospitalarios distintos: Cleveland, Hungary, Switzerland, and Long Beach V.

La URL correspondiente al dataset es https://www.kaggle.com/johnsmith88/heart-disease-dataset/data (https://www.kaggle.com/johnsmith88/heart-disease-dataset/data).

La pregunta que intenta responder es: ¿Cuáles son los factores que debemos tener en cuenta a la hora de clasificar un nuevo paciente con síntomas de enfermedad coronaria?

Está claro que cuanto antes se diagnostique correctamente y se trate al paciente, mayores posibilidades tiene de recuperarse o de no sufrir secuelas.
El conjunto de datos incluye sujetos sanos y pacientes con enfermedades cardíacas, de 29 a 77 años.

El dataset contiene un total de 14 características clínicas para cada caso.

Los atributos del dataset son:

1. **age**: Edad del paciente en años.
2. **sex**: Sexo (1 = hombre; 0 = mujer).
3. **cp**: Tipo de dolor en el pecho:
   (0 = asintomático; 1 = angina atípica; 2 = dolor no anginal; 3 = angina típica)
4. **trestbps**: Presión sanguínea en reposo al ser admitido en el hospital en mm Hg.
5. **chol**: Nivel de colesterol en sangre en mg/dl.
6. **fbs**: Nivel de azúcar en sangre en ayunas > 120 mg/dl (1 = si; 0 = no).
7. **restecg**: Resultados electrocardiográficos en reposo.
   (0 = hipertrofia ventricular izquierda; 1 = normal; 2 = anormalidad onda ST-T)
8. **thalach**: Frecuencia cardíaca máxima alcanzada.
9. **exang**: Angina inducida por el ejercicio (1 = si; 0 = no).
10. **oldpeak**: Depresión onda ST inducida por el ejercicio relativo al descanso.
11. **slope**: Pendiente del segmento ST del ejercicio pico.
    (0 = bajada; 1 = plano; 2 = pendiente ascendente)
12. **ca**: Número de vasos principales (0-4) coloreados por flouroscopia. (indicador específico de insuficiencia isquémica).
13. **thal**: Exploración cardíaca de talio.
    (1 = defecto permanente; 2 = normal;3 = defecto reversible).
14. **target**: Diagnóstico de enfermedad cardíaca.
    (0 = enfermo; 1 = sano)

2. Integración y selección de los datos de interés a analizar.

Cargamos el dataset en memoria.

```
datos <- read.table("datasets_216167_477177_heart.csv", header = TRUE, sep = ",")
```

Les cambiamos los nombres a las variables para mejorar la claridad.

```
colnames(datos) <- c('edad','sexo','dolor','tension','colesterol','azucar',
                     'ecografia','frecmax','ejercicio','depST','pendiente',
                     'vasos.coloreados','exploracion.talio','diagnostico')
```

Obtenemos el resumen de los datos obtenidos.

```
summary(datos)
```

```
##       edad            sexo            dolor           tension
##  Min.   :29.00   Min.   :0.0000   Min.   :0.0000   Min.   : 94.0
##  1st Qu.:48.00   1st Qu.:0.0000   1st Qu.:0.0000   1st Qu.:120.0
##  Median :56.00   Median :1.0000   Median :1.0000   Median :130.0
##  Mean   :54.43   Mean   :0.6956   Mean   :0.9424   Mean   :131.6
##  3rd Qu.:61.00   3rd Qu.:1.0000   3rd Qu.:2.0000   3rd Qu.:140.0
##  Max.   :77.00   Max.   :1.0000   Max.   :3.0000   Max.   :200.0
##    colesterol      azucar         ecografia         frecmax
##  Min.   :126   Min.   :0.0000   Min.   :0.0000   Min.   : 71.0
##  1st Qu.:211   1st Qu.:0.0000   1st Qu.:0.0000   1st Qu.:132.0
##  Median :240   Median :0.0000   Median :1.0000   Median :152.0
##  Mean   :246   Mean   :0.1493   Mean   :0.5298   Mean   :149.1
##  3rd Qu.:275   3rd Qu.:0.0000   3rd Qu.:1.0000   3rd Qu.:166.0
##  Max.   :564   Max.   :1.0000   Max.   :2.0000   Max.   :202.0
##    ejercicio          depST         pendiente     vasos.coloreados
##  Min.   :0.0000   Min.   :0.000   Min.   :0.000   Min.   :0.0000
##  1st Qu.:0.0000   1st Qu.:0.000   1st Qu.:1.000   1st Qu.:0.0000
##  Median :0.0000   Median :0.800   Median :1.000   Median :0.0000
##  Mean   :0.3366   Mean   :1.072   Mean   :1.385   Mean   :0.7541
##  3rd Qu.:1.0000   3rd Qu.:1.800   3rd Qu.:2.000   3rd Qu.:1.0000
##  Max.   :1.0000   Max.   :6.200   Max.   :2.000   Max.   :4.0000
##  exploracion.talio  diagnostico
##  Min.   :0.000     Min.   :0.0000
##  1st Qu.:2.000     1st Qu.:0.0000
##  Median :2.000     Median :1.0000
##  Mean   :2.324     Mean   :0.5132
##  3rd Qu.:3.000     3rd Qu.:1.0000
##  Max.   :3.000     Max.   :1.0000
```

Y su estructura.

```
str(datos)
```

```
## 'data.frame':    1025 obs. of  14 variables:
##  $ edad             : int  52 53 70 61 62 58 58 55 46 54 ...
##  $ sexo             : int  1 1 1 1 0 0 1 1 1 1 ...
##  $ dolor            : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ tension          : int  125 140 145 148 138 100 114 160 120 122 ...
##  $ colesterol       : int  212 203 174 203 294 248 318 289 249 286 ...
##  $ azucar           : int  0 1 0 0 1 0 0 0 0 0 ...
##  $ ecografia        : int  1 0 1 1 1 0 2 0 0 0 ...
##  $ frecmax          : int  168 155 125 161 106 122 140 145 144 116 ...
##  $ ejercicio        : int  0 1 1 0 0 0 0 1 0 1 ...
##  $ depST            : num  1 3.1 2.6 0 1.9 1 4.4 0.8 0.8 3.2 ...
##  $ pendiente        : int  2 0 0 2 1 1 0 1 2 1 ...
##  $ vasos.coloreados : int  2 0 0 1 3 0 3 1 0 2 ...
##  $ exploracion.talio: int  3 3 3 3 2 2 1 3 3 2 ...
##  $ diagnostico      : int  0 0 0 0 0 1 0 0 0 0 ...
```

Podemos ver una muestra de los datos.

```
head(datos)
```

```
##   edad sexo dolor tension colesterol azucar ecografia frecmax ejercicio depST
## 1   52    1     0     125        212      0         1     168         0   1.0
## 2   53    1     0     140        203      1         0     155         1   3.1
## 3   70    1     0     145        174      0         1     125         1   2.6
## 4   61    1     0     148        203      0         1     161         0   0.0
## 5   62    0     0     138        294      1         1     106         0   1.9
## 6   58    0     0     100        248      0         0     122         0   1.0
##   pendiente vasos.coloreados exploracion.talio diagnostico
## 1         2                2                 3           0
## 2         0                0                 3           0
## 3         0                0                 3           0
## 4         2                1                 3           0
## 5         1                3                 2           0
## 6         1                0                 2           1
```

Nuestro interés se centra en identificar y cuantificar las variables que tienen un mayor impacto sobre nuestra variable objetivo, el diagnóstico del paciente.

3. Limpieza de los datos.

3.1. Datos ausentes.

¿Los datos contienen ceros o elementos vacíos? ¿Cómo gestionarías cada uno de estos casos?

Estudiamos caso por caso todas las variables.

Primero la variable edad. Ya hemos visto en la salida del comando *summary()* que el rango de edad está comprendido entre 29 y 77 años, lo cual es plausible. Veamos si tiene falta algún valor:

```
table(is.na(datos$edad))
```

```
##
## FALSE
##  1025
```

Las 1025 son válidas.
En el caso de que la edad estuviera fuera del rango razonable podríamos suponer que se trata de un error y descartar la observación o bien sustituirlo por un valor de tendencia central como la media, o la mediana (si hay presencia de valores extremos).

Veamos como se distribuye la variable sexo.

```
table(datos$sexo)
```

```
##
##   0   1
## 312 713
```

Si alguno de los valores no fuera 0 ó 1, ya sea NA o valores distintos, podríamos utilizar una tercera categoría para la variable que indicara que el sexo del paciente no es conocido y realizar el análisis con tres niveles de la variable.

Se trata realmente de una variable categórica, así que la convertimos a tipo factor.

```
datos$sexo[datos$sexo == 0] <- "mujer"
datos$sexo[datos$sexo == 1] <- "hombre"
datos$sexo <- as.factor(datos$sexo)
table(datos$sexo)
```

```
##
## hombre  mujer
##    713    312
```

Variable dolor.

```
table(is.na(datos$dolor))
```

```
##
## FALSE
##  1025
```

Veamos que no falta ningún valor. Veamos ahora como se distribuye la variable.

```
table(datos$dolor)
```

```
##
##   0   1   2   3
## 497 167 284  77
```

Al igual quen con la variable *sexo*, se trata también de una variable categórica con cuatro valores posibles, así que la convertimos a tipo factor según los datos de los tipos de dolor.

```
datos$dolor[datos$dolor == 0] <- "asintomatico"
datos$dolor[datos$dolor == 1] <- "atipico"
datos$dolor[datos$dolor == 2] <- "otro"
datos$dolor[datos$dolor == 3] <- "tipico"
datos$dolor <- as.factor(datos$dolor)
```

Los datos quedan ahora de la siguiente forma.

```
table(datos$dolor)
```

```
##
## asintomatico      atipico         otro       tipico
##          497          167          284           77
```

Estudiamos valores faltantes en la variable tensión.

```
table(is.na(datos$tension))
```

```
##
## FALSE
##  1025
```

En cuanto a sus valores, según la salida del comando *summary* el rango está entre 94 y 200, dentro de los valores posibles.

Veamos el colesterol.

```
table(is.na(datos$colesterol))
```

```
##
## FALSE
##  1025
```

No falta ningún valor. El rango de valores de la variable es de 126 a 564. Valores plausibles a pesar de que en algunos casos son muy elevados. Un valor de 0 por ejemplo, tendríamos que descartarlo o sustituirlo (por la media, por ejemplo) ya que en este caso no tiene sentido que un paciente tenga colesterol 0.

En cuanto a la variable *azucar*, tampoco le faltan datos.

```
table(is.na(datos$azucar))
```

```
##
## FALSE
##  1025
```

La distribución de sus valores es la siguiente.

```
table(datos$azucar)
```

```
##
##   0   1
## 872 153
```

Vemos que solo tiene dos posibles valores, la convertimos de entera a categórica.

```
datos$azucar[datos$azucar == 1] <- "si"
datos$azucar[datos$azucar == 0] <- "no"
datos$azucar <- as.factor(datos$azucar)
table(datos$azucar)
```

```
##
##  no  si
## 872 153
```

En el caso de que la variable tuviera valores distintos al 0 y al 1, podríamos crear una nueva categoría para la variable que fuera “valor desconocido”, e incluir en este caso el resto de valores.

Para la variable *ecografía* ocurre lo mismo.

```
table(is.na(datos$ecografia))
```

```
##
## FALSE
##  1025
```

```
table(datos$ecografia)
```

```
##
##   0   1   2
## 497 513  15
```

Solo tiene 3 posibles valores, la reconvertimos a tipo factor.

```
datos$ecografia[datos$ecografia == 0] <- "hipertrofia"
datos$ecografia[datos$ecografia == 1] <- "normal"
datos$ecografia[datos$ecografia == 2] <- "anormal"
datos$ecografia <- as.factor(datos$ecografia)
table(datos$ecografia)
```

```
##
##     anormal hipertrofia      normal
##          15         497         513
```

Frecuencia cardiaca máxima.

```
table(is.na(datos$frecmax))
```

```
##
## FALSE
##  1025
```

No falta ningún valor y todos se encuentran dentro de valores posibles (71-202). Un valor de debajo de 20 pulsaciones o por encima de 300 (por poner unos límites) podríamos considerarlo como un dato erróneo y habría que descartarlo o sustituirlo.

Variable *ejercicio*.

```
table(datos$ejercicio)
```

```
##
##   0   1
## 680 345
```

La tratamos de forma similar.

```
datos$ejercicio[datos$ejercicio == 1] <- "si"
datos$ejercicio[datos$ejercicio == 0] <- "no"
datos$ejercicio <- as.factor(datos$ejercicio)
table(datos$ejercicio)
```

```
##
##  no  si
## 680 345
```

Variable *depST*.

```
table(is.na(datos$depST))
```

```
##
## FALSE
##  1025
```

Esta variable mide la depresión de la onda ST del electrocardiograma en una prueba de esfuerzo. El rango de la variable va de 0 a 6.2.

Variale *pendiente*.

```
table(datos$pendiente)
```

```
##
##   0   1   2
##  74 482 469
```

La reconvertimos a factor.

```
datos$pendiente[datos$pendiente == 0] <- "bajada"
datos$pendiente[datos$pendiente == 1] <- "plana"
datos$pendiente[datos$pendiente == 2] <- "subida"
datos$pendiente <- as.factor(datos$pendiente)
table(datos$pendiente)
```

```
##
## bajada  plana subida
##     74    482    469
```

Variable *vasos.coloreados*.

```
table(datos$vasos.coloreados)
```

```
##
##   0   1   2   3   4
## 578 226 134  69  18
```

No le faltan valores. Solo tiene 5 opciones, la podemos tratar como una variable categórica, así que la convertimos a factor. En este caso podríamos mantener la variable como numérica pero no tiene mucho sentido porque no vamos a realizar operaciones numéricas con ella.

```
datos$vasos.coloreados <- as.factor(datos$vasos.coloreados)
table(datos$vasos.coloreados)
```

```
##
##   0   1   2   3   4
## 578 226 134  69  18
```

Variable *exploracion.talio*.

```
table(datos$exploracion.talio)
```

```
##
##   0   1   2   3
##   7  64 544 410
```

Vemos que hay 7 pacientes de los que no tenemos información, así que los codificamos como pacientes “sin información” sobre este dato. El resto de caso, los asociamos según la meta información que tenemos sobre el dataset.

```
datos$exploracion.talio[datos$exploracion.talio == 0] <- "desconocido"
datos$exploracion.talio[datos$exploracion.talio == 1] <- "daño_permanente"
datos$exploracion.talio[datos$exploracion.talio == 2] <- "sin_daño"
datos$exploracion.talio[datos$exploracion.talio == 3] <- "daño_reversible"
datos$exploracion.talio <- as.factor(datos$exploracion.talio)
table(datos$exploracion.talio)
```

```
##
## daño_permanente daño_reversible     desconocido        sin_daño
##              64             410               7             544
```

Por último tenemos la variable que nos muestra el resultado final del paciente: *diagnostico*

```
table(datos$diagnostico)
```

```
##
##   0   1
## 499 526
```

Variable dicotómica que convertimos a tipo factor.

```
datos$diagnostico[datos$diagnostico == 0] <- "enfermo"
datos$diagnostico[datos$diagnostico == 1] <- "sano"
datos$diagnostico <- as.factor(datos$diagnostico)
table(datos$diagnostico)
```

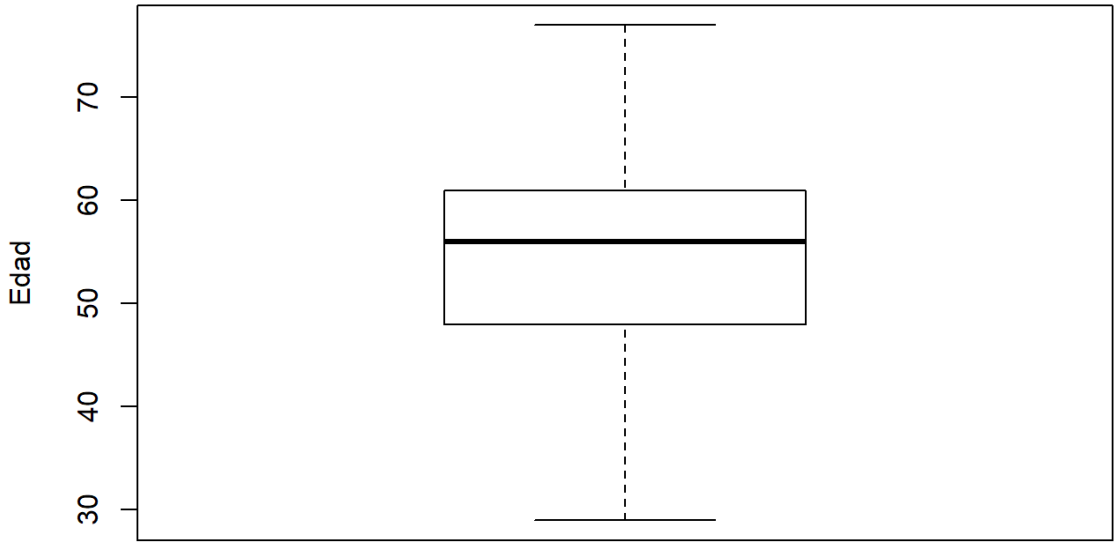
```
##
## enfermo    sano
##     499     526
```

3.2. Identificación y tratamiento de valores extremos.

Estudiamos la distribución de las variables cualitativas.

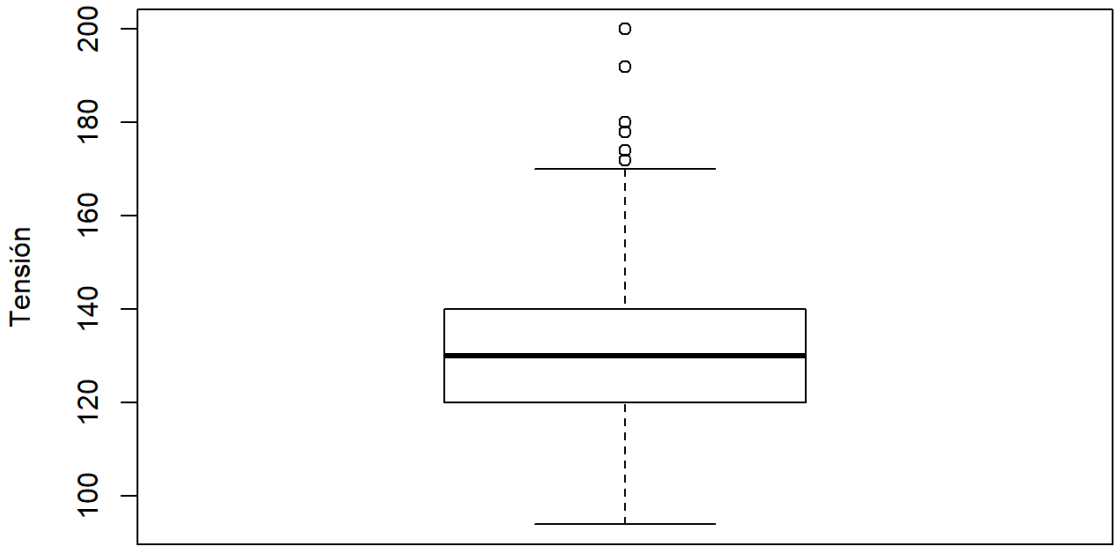
En el caso de la edad vemos que no hay valores extremos.

```
caja.edad <- boxplot(datos$edad, ylab = "Edad"  )
```



Para la variable *tensión*.

```
caja.tension <- boxplot(datos$tension, ylab = "Tensión")
```



La lista de los valores extermos es la siguiente.

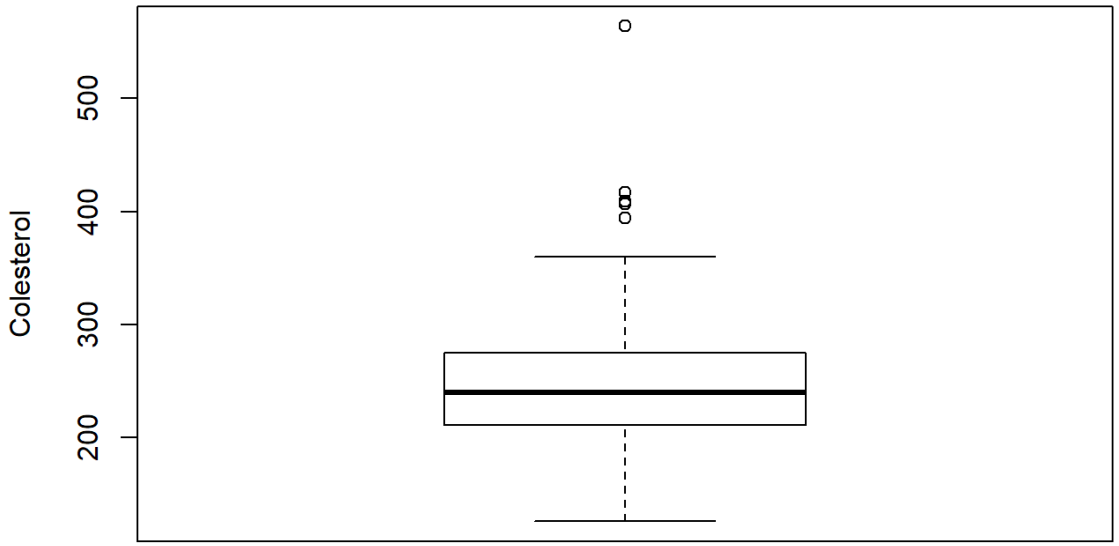
```
table(caja.tension$out)
```

```
##
## 172 174 178 180 192 200
##   3   3   7  10   3   4
```

Podemos ver que hay unos cuantos valores extremos, en concreto 30. Ya que son valores plausibles, entendemos que corresponden a la realidad, no se trata de errores de medida.

Para la variable *colesterol*.

```
caja.colesterol <- boxplot(datos$colesterol, ylab = "Colesterol")
```



La lista de los valores extremos es la siguiente.

```
table(caja.colesterol$out)
```

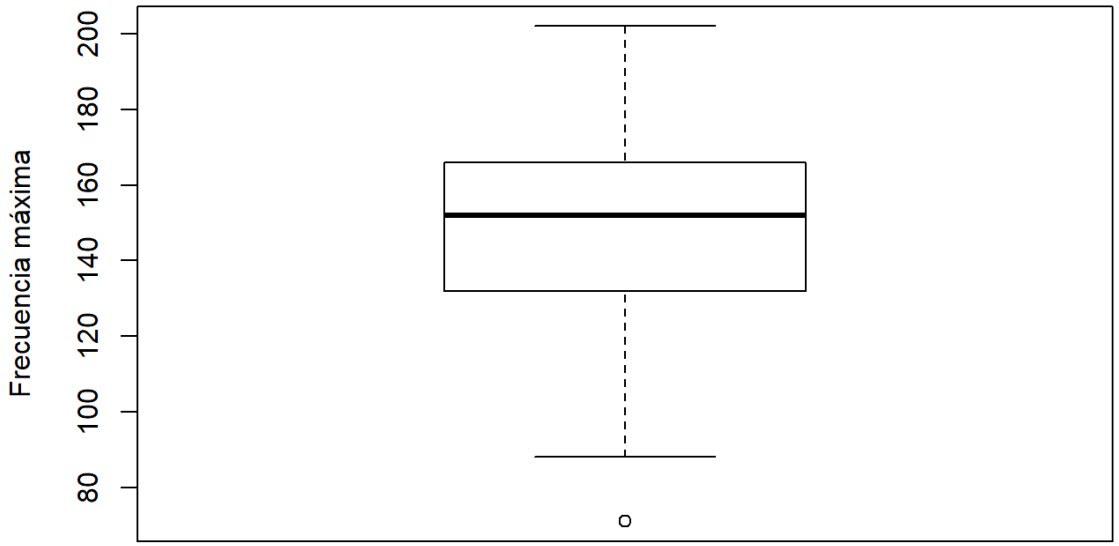
```
##
## 394 407 409 417 564
##   3   4   3   3   3
```

Ocurre algo similar al caso de la tensión, son valores que pueden ser válidos.

Estos valores tan altos pueden ser debidos a la hipercolesterolemia familiar, un trastorno grave ocasionado por mutaciones en las lipoproteínas que transportan el colesterol. En estos casos el nivel de colesterol se sitúa entre los 300 y 500 miligramos por decilitro (mg/dl).

Variable *frecmax*.

```
caja.frecmax <- boxplot(datos$frecmax, ylab = "Frecuencia máxima")
```



Los valores extermos son.

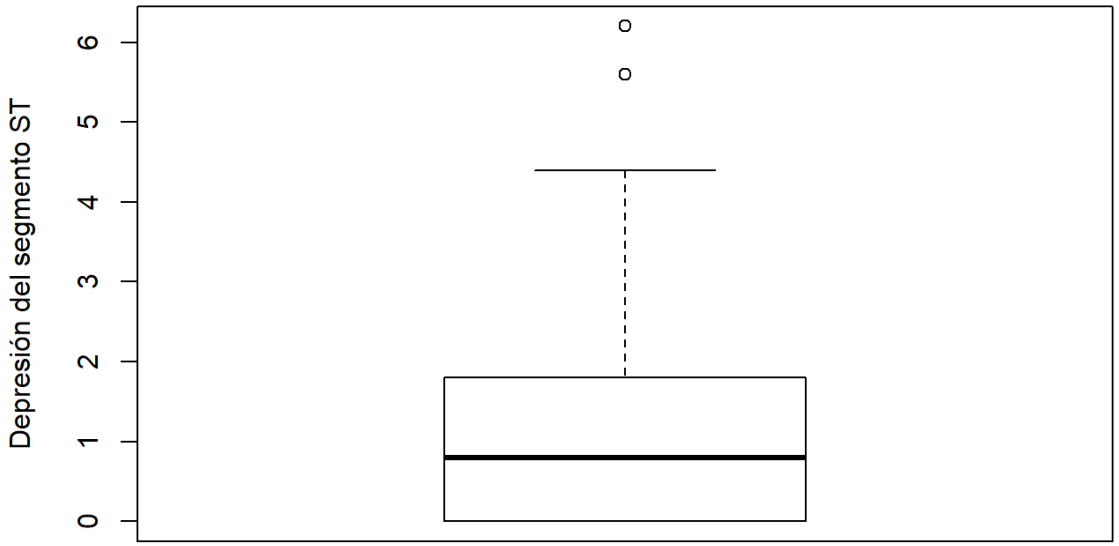
```
table(caja.frecmax$out)
```

```
##
## 71
##  4
```

Solo hay cuatro casos pero las pulsaciones pueden considerarse normales, no errores de medida.

Para la variable *depST*. Una de las variables más importantes a evaluar es el comportamiento del segmento ST dentro de un electrocardiograma. La depresión descendente como horizontal del segmento ST son potentes predictores de enfermedad coronaria.

```
caja.depST <- boxplot(datos$depST, ylab = "Depresión del segmento ST")
```



La lista de los valores extremos es la siguiente.

```
table(caja.depST$out)
```

```
##
## 5.6 6.2
##   4   3
```

4. Análisis de los datos.

4.1. Agrupamiento.

Selección de los grupos de datos que se quieren analizar/comparar (planificación de los análisis a aplicar).

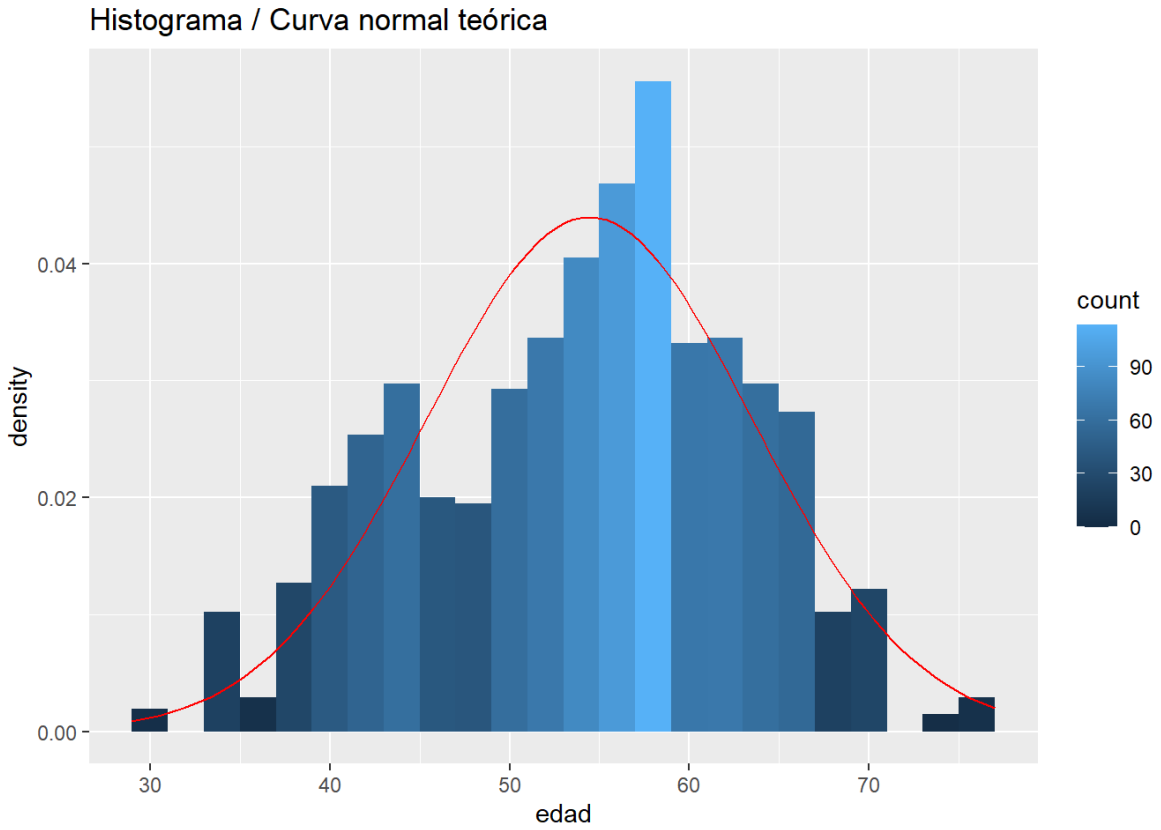
Nos interesa comparar la incidencia que tiene cada una de las variables del dataset en el diagnóstico final del paciente para conocer de qué manera afectan al resultado, tanto por separado y como en conjunto.

4.2. Comprobación de la normalidad y homogeneidad de la varianza.

Estudiamos la normalidad de la variable *edad*.

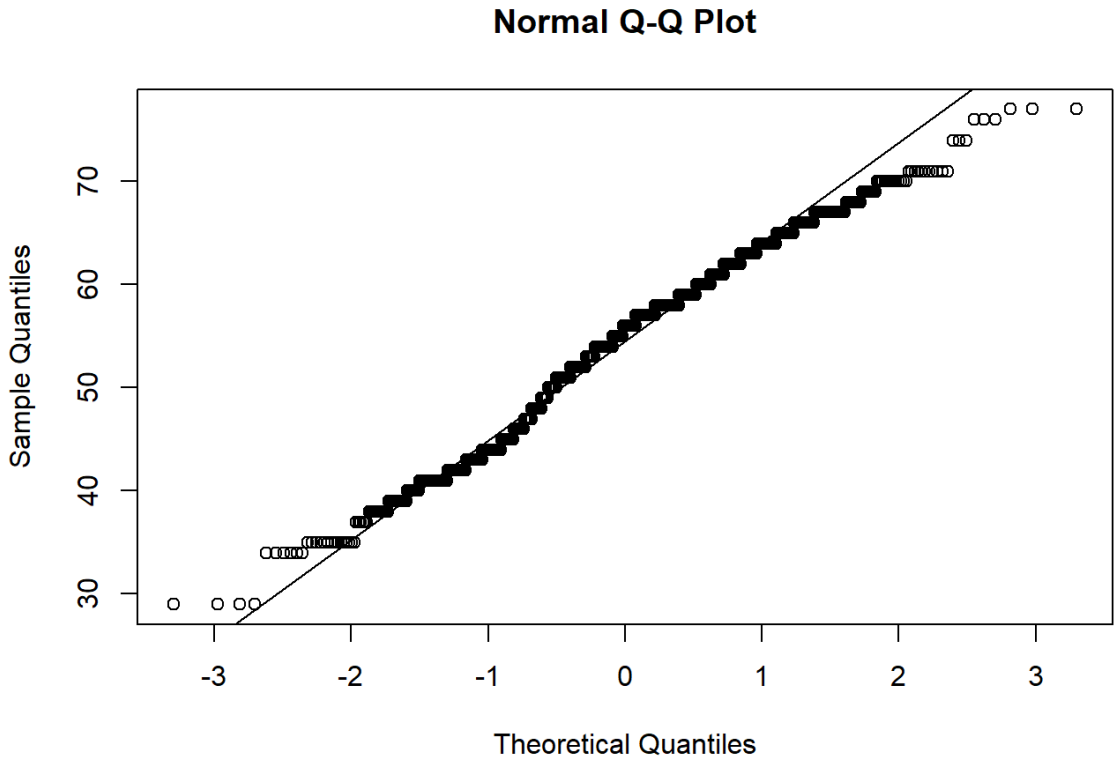
Visualizamos el histograma de la variable junto a la curva de la distribución normal a la que equivaldría.

```
ggplot(data = datos, aes(x = edad)) + geom_histogram(aes(y = ..density.., fill = ..count..), bins = 25) +
  stat_function(fun = dnorm, colour = "red",args = list(mean = mean(datos$edad), sd = sd(datos$edad))) +
  ggtitle("Histograma / Curva normal teórica")
```



Revisamos también la gráfica Q-Q para ver las diferencias frente a la normal.

```
qqnorm(datos$edad)
qqline(datos$edad)
```



Se aproxima pero tiene ciertas discrepancias, sobretodo en los extermos.

Podemos usar el test de Shapiro-Wilk si tenemos memos de 5000 muestras, cuya hipótesis nula es que se ajusta a una distribución normal.

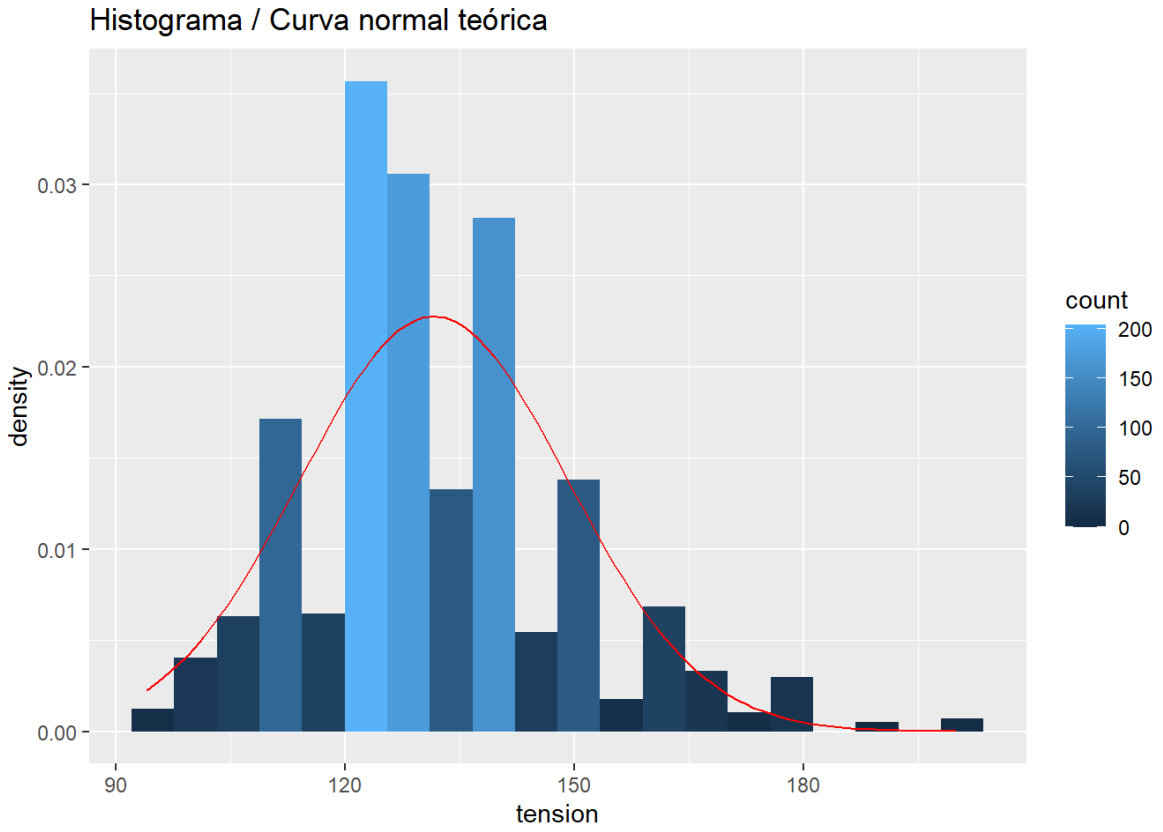
```
shapiro.test(x = datos$edad)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  datos$edad
## W = 0.98436, p-value = 5.039e-09
```

Tenemos un p < 0.05 por lo que tenemos una distribución que no es normal (rechazamos hipótesis nula).

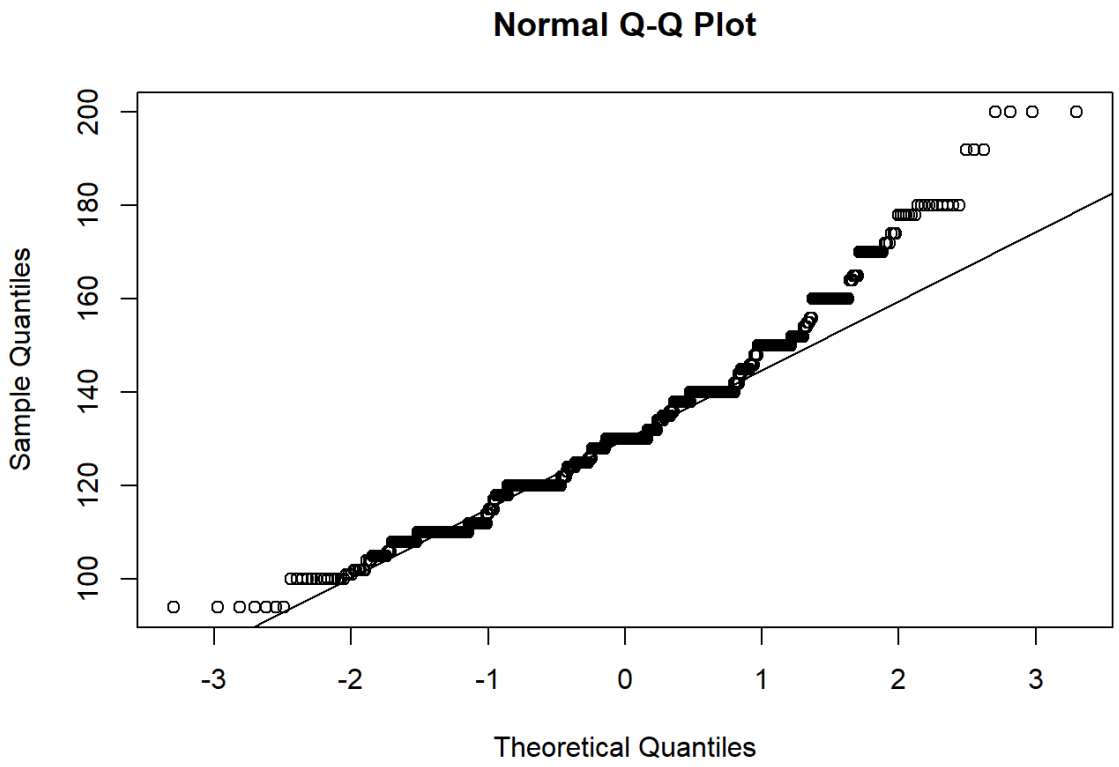
Variable *tensión*.

```
ggplot(data = datos, aes(x = tension)) + geom_histogram(aes(y = ..density.., fill = ..count..), bins = 20)
 +
  stat_function(fun = dnorm, colour = "red",args = list(mean = mean(datos$tension), sd = sd(datos$tensio
n))) +
  ggtitle("Histograma / Curva normal teórica")
```



Revisamos también la gráfica Q-Q.

```
qqnorm(datos$tension)
qqline(datos$tension)
```



A simple vista vemos que no se ajusta mucho a la normal. Probamos con el test de normalidad.

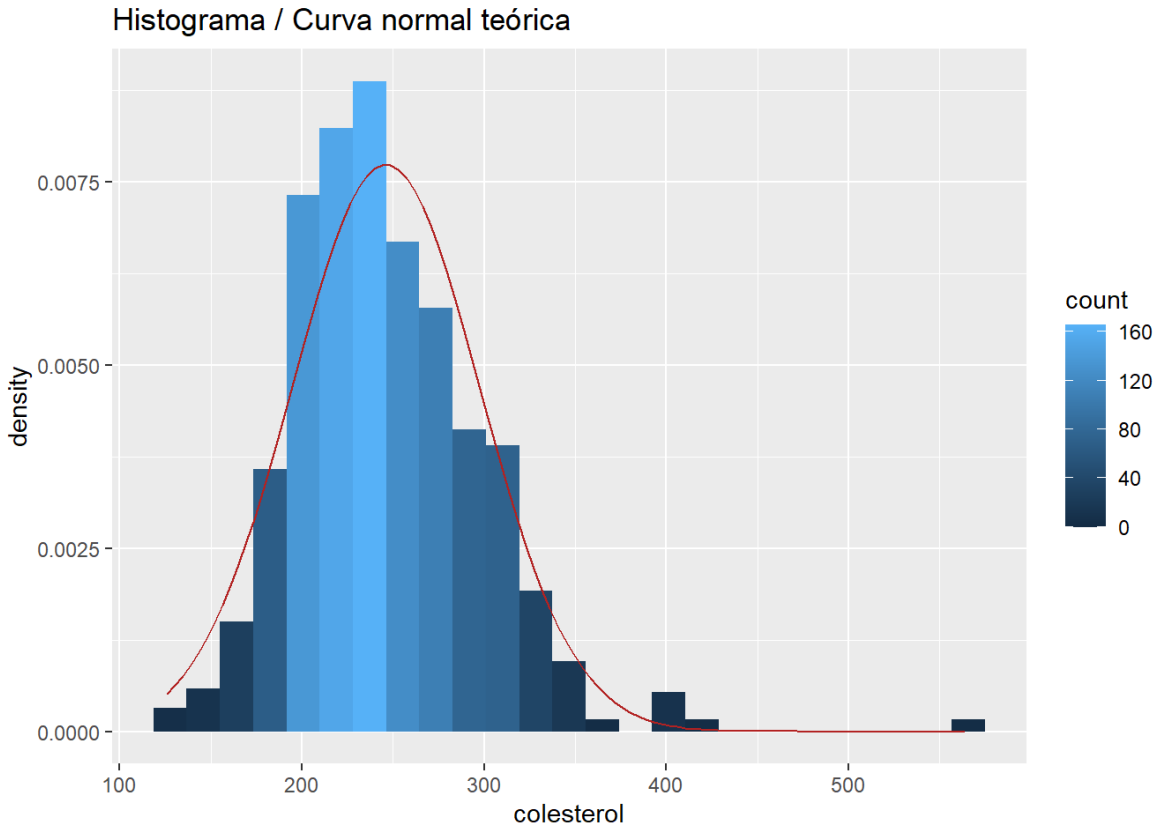
```
shapiro.test(x = datos$tension)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  datos$tension
## W = 0.96331, p-value = 2.194e-15
```

El p-valor es menor que 0.05, rechazamos hipótesis nula, luego no se ajusta a una distribución normal.

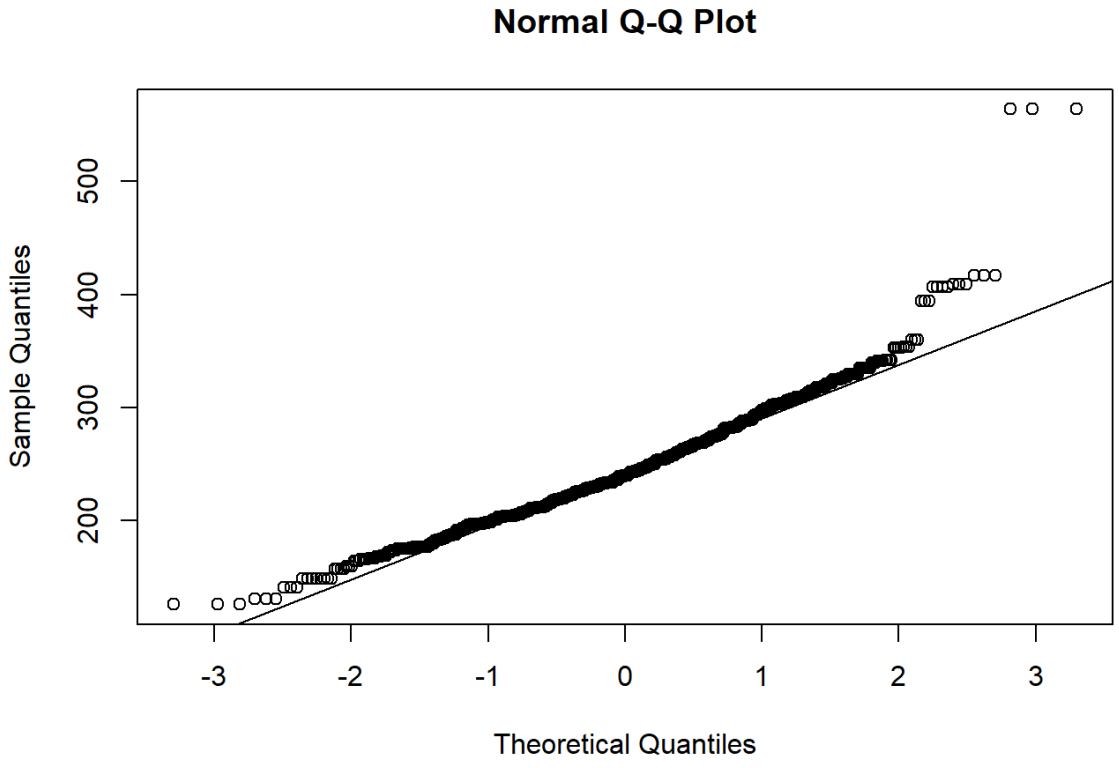
Variable *colesterol*.

```
ggplot(data = datos, aes(x = colesterol)) + geom_histogram(aes(y = ..density.., fill = ..count..), bins = 25) +
  stat_function(fun = dnorm, colour = "firebrick",args = list(mean = mean(datos$colesterol), sd = sd(datos$colesterol))) +
  ggtitle("Histograma / Curva normal teórica")
```



Se aproxima un poco mejor a la normal.

```
qqnorm(datos$colesterol)
qqline(datos$colesterol)
```



Según el test de normalidad.

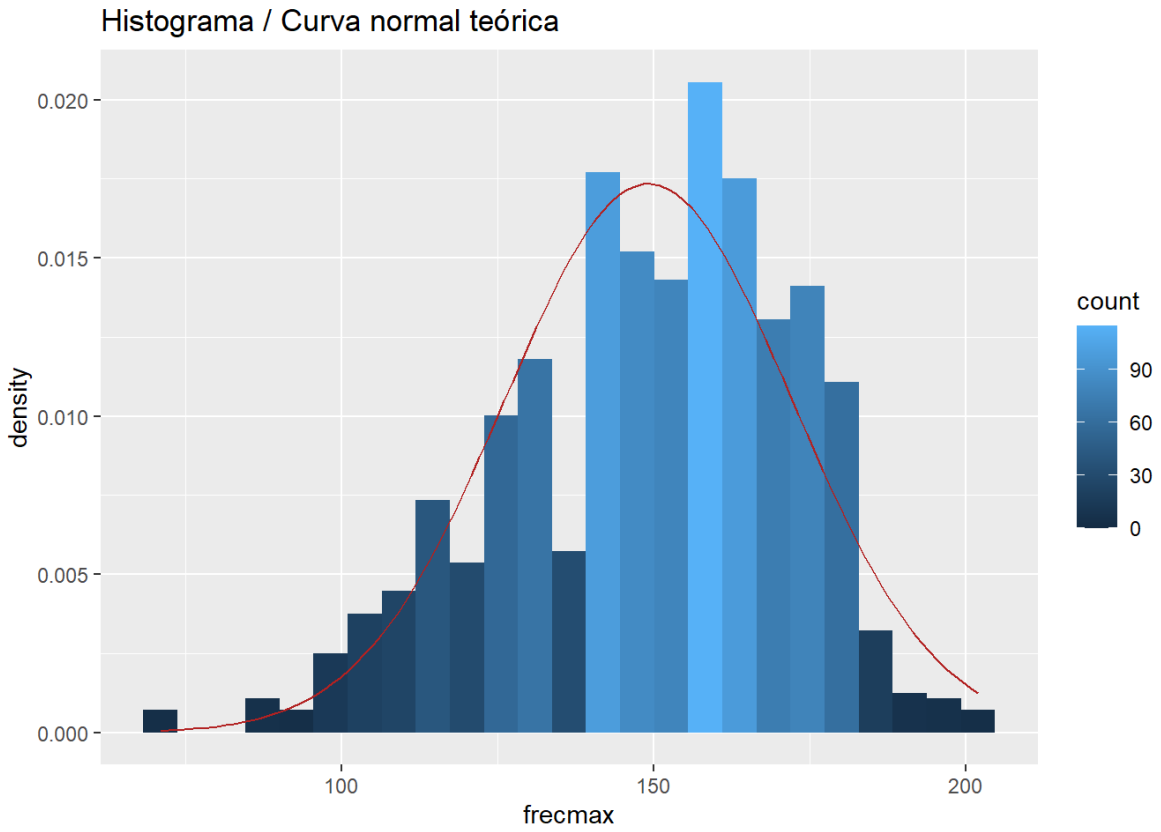
```
shapiro.test(x = datos$colesterol)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  datos$colesterol
## W = 0.95022, p-value < 2.2e-16
```

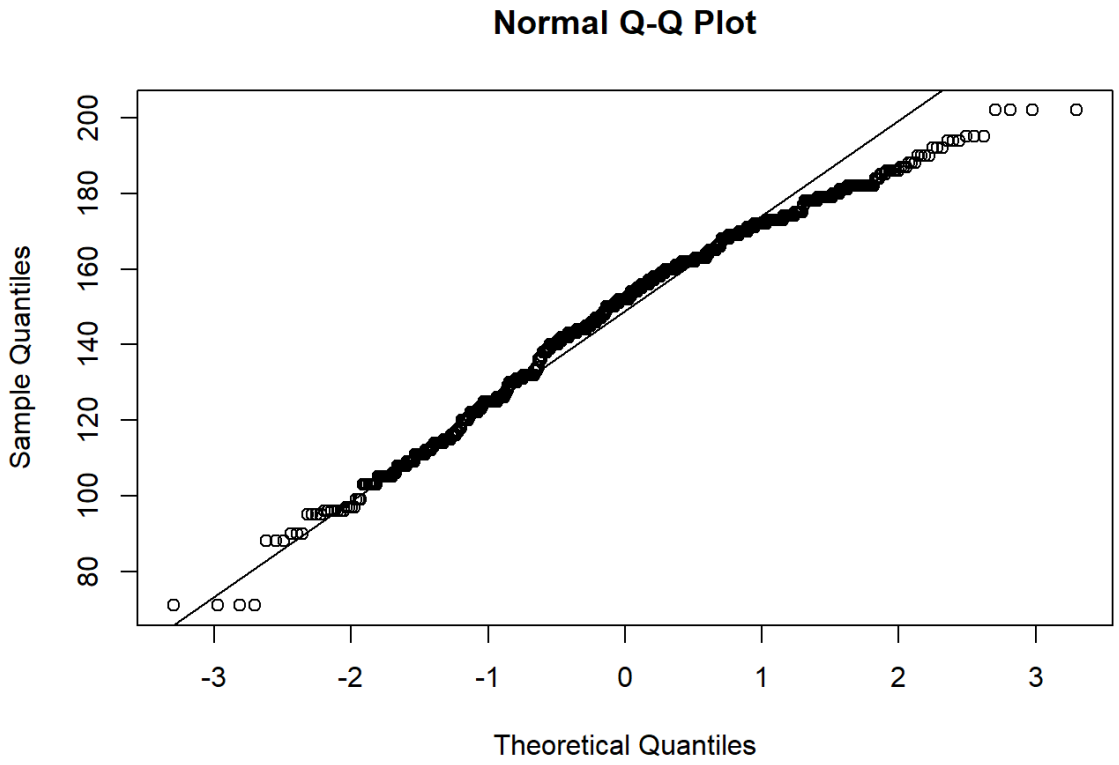
Seguimos afirmando que según el test de Shapiro-Wilk, nos indica que el colesterol tampoco se ajusta a una distribución normal.

Variable *frecmax*.

```
ggplot(data = datos, aes(x = frecmax)) +  geom_histogram(aes(y = ..density.., fill = ..count..), bins = 25)
+
  stat_function(fun = dnorm, colour = "firebrick",args = list(mean = mean(datos$frecmax), sd = sd(datos$fre
cmax))) +
  ggtitle("Histograma / Curva normal teórica")
```




```
qqnorm(datos$frecmax)
qqline(datos$frecmax)
```



Realizamos el test de normalidad.

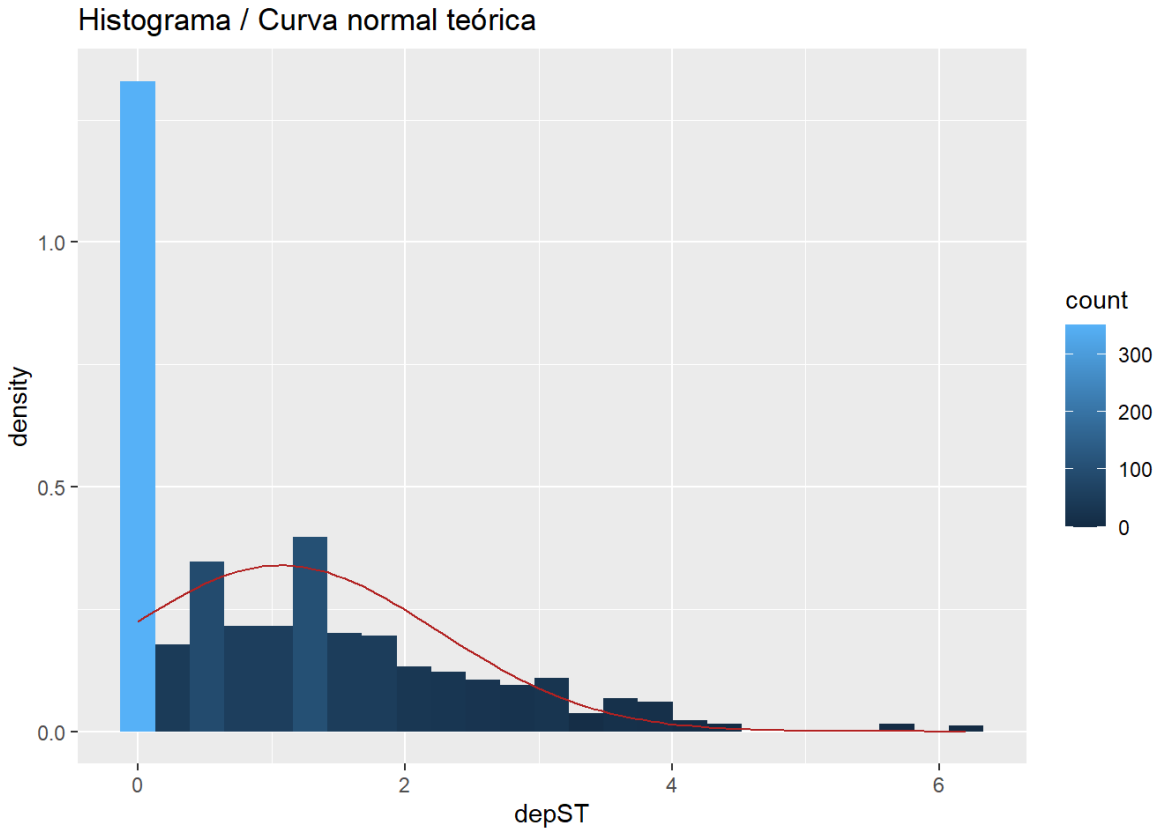
```
shapiro.test(x = datos$frecmax)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  datos$frecmax
## W = 0.9774, p-value = 1.55e-11
```

Obtenemos el mismo resultado, p-valor < 0.05, no es distribución normal (rechazamos hipótesis nula).

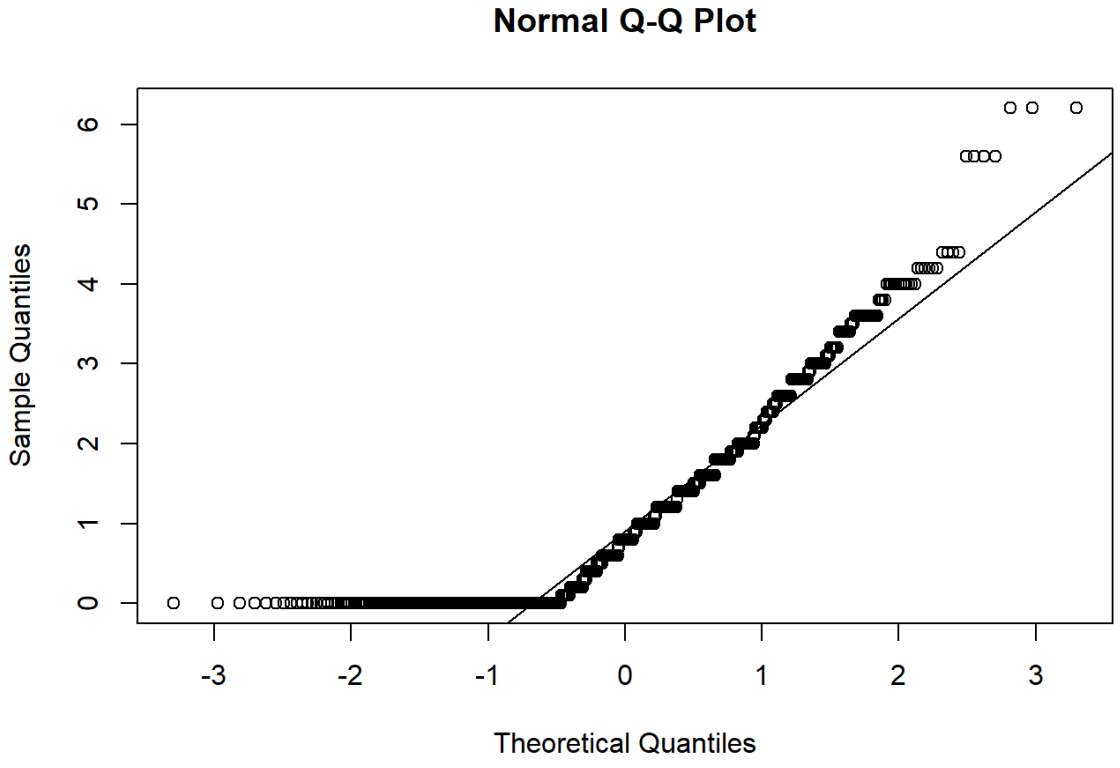
Variable *depST*.

```
ggplot(data = datos, aes(x = depST)) +  geom_histogram(aes(y = ..density.., fill = ..count..), bins = 25) +
  stat_function(fun = dnorm, colour = "firebrick",args = list(mean = mean(datos$depST), sd = sd(datos$depS
T))) +
  ggtitle("Histograma / Curva normal teórica")
```



En este caso se aleja bastante de la normal porque los casos con 0 sobresalen mucho sobre el resto. Podemos constatarlo en el diagrama Q-Q.

```
qqnorm(datos$depST)
qqline(datos$depST)
```



Y como podíamos sospechar, el test de Shapiro-Wilk también confirma que no es una distribución normal.

```
shapiro.test(x = datos$depST)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  datos$depST
## W = 0.85025, p-value < 2.2e-16
```

Estudiamos ahora la homogeneidad de la varianza (homocedasticidad) de la variable edad entre los grupos de estudio, los enfermos y los sanos.

Ya que no vamos a suponer normalidad de los grupos, usaremos el test de Levene con la mediana.

Hipótesis nula: La varianza es igual entre los grupos (enfermos y sanos).
Hipótesis alternativa: La varianza es distinta.

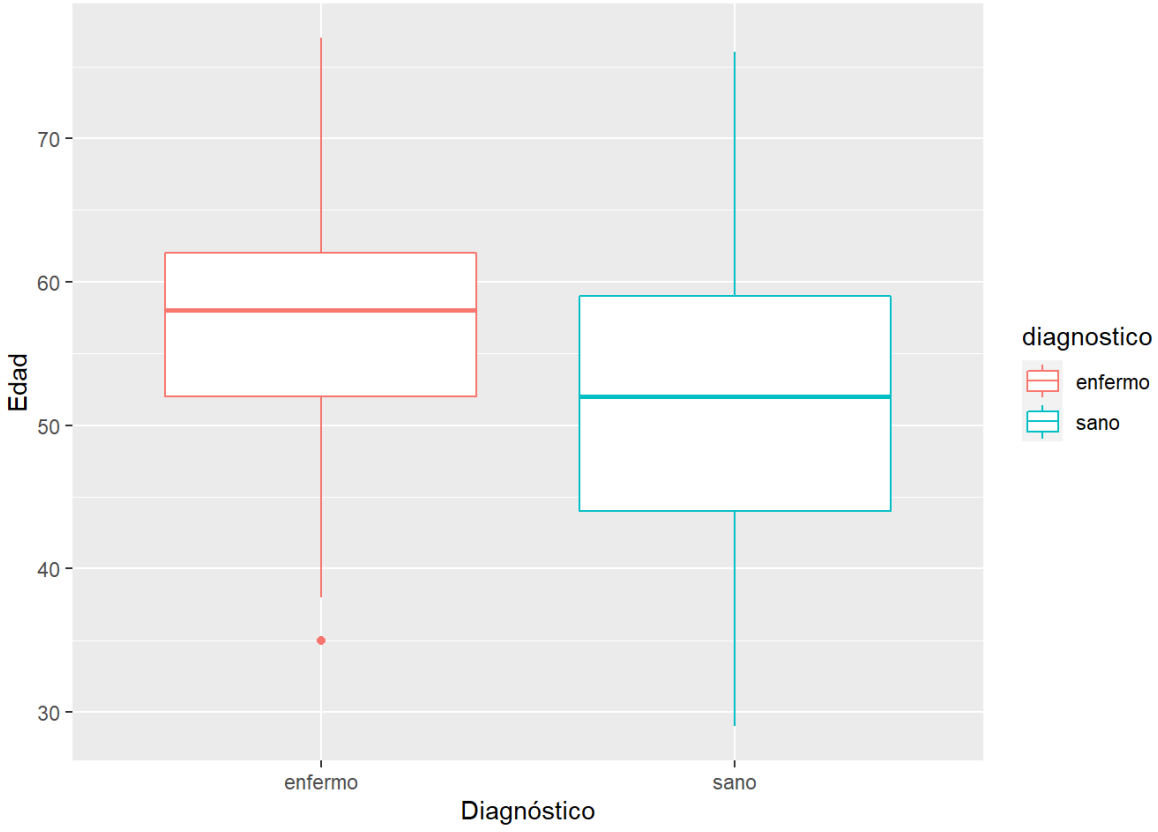
```
leveneTest(y = datos$edad,  group = datos$diagnostico, center = "median")
```

```
## Levene's Test for Homogeneity of Variance (center = "median")
##         Df F value    Pr(>F)
## group    1  30.838 3.575e-08 ***
##       1023
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Obtenemos un p-valor < 0.05, luego rechazamos la hipótesis nula, y consideraremos que la varianza es distinta.

Podemos verlo gráficamente.

```
ggplot(datos,aes(x=diagnostico)) + geom_boxplot(aes(y=edad, col = diagnostico)) + xlab("Diagnóstico")+ ylab
("Edad")
```



Estudiamos ahora la homogeneidad de la varianza de la variable *tensión* entre los grupos de estudio, los enfermos y los sanos.

Como la variable no se distribuye de forma normal, volvemos a usaremos el test de Levene con la mediana.

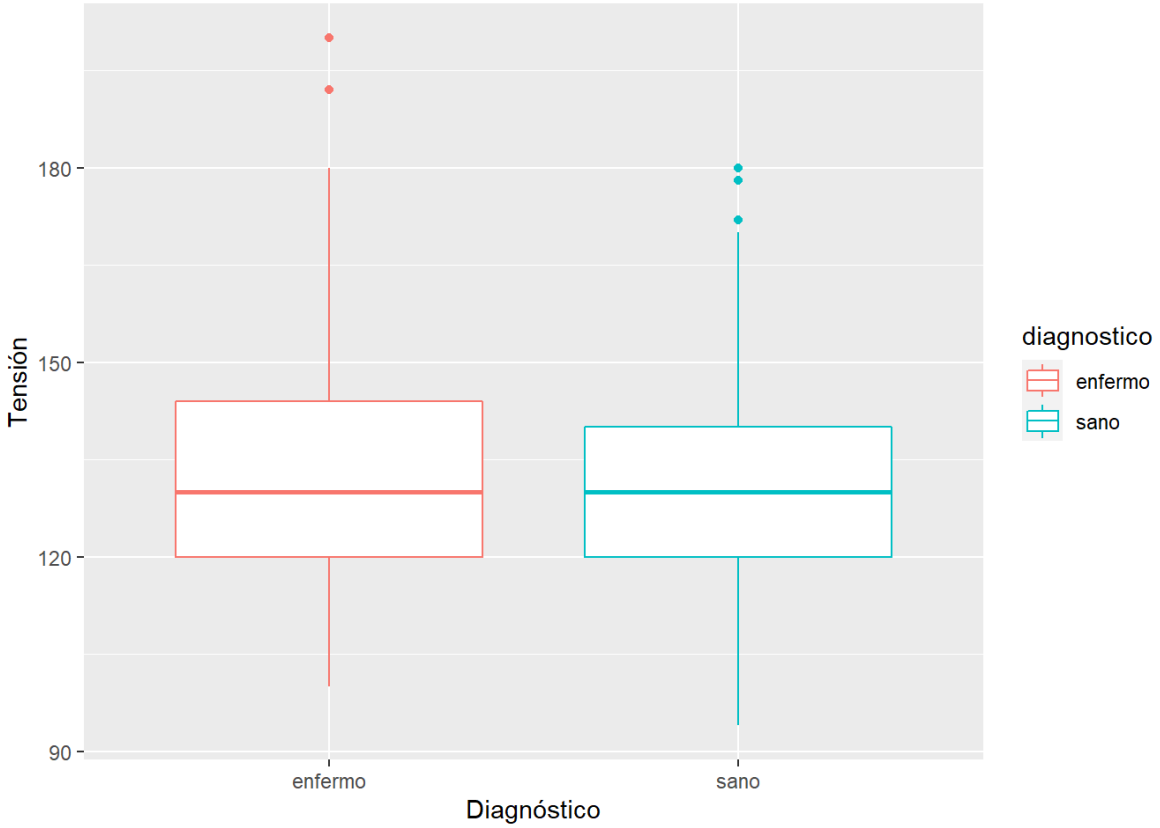
```
leveneTest(y = datos$tension,  group = datos$diagnostico, center = "median")
```

```
## Levene's Test for Homogeneity of Variance (center = "median")
##         Df F value  Pr(>F)
## group    1  5.5492 0.01868 *
##       1023
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Obtenemos un p-valor < 0.05, luego rechazamos la hipótesis nula, y consideraremos que la varianza es distinta. En este caso, para un test con nivel de significancia de 0.01, no podríamos rechazar la hipótesis nula.

Podemos verlo gráficamente.

```
ggplot(datos,aes(x=diagnostico)) + geom_boxplot(aes(y=tension, col = diagnostico)) + xlab("Diagnóstico")+ y
lab("Tensión")
```



Observamos que las distribuciones en este caso son parecidas.

Estudiamos la homogeneidad de la varianza de la variable *colesterol* entre los grupos de estudio, los enfermos y los sanos.

Como la variable no se distribuye de forma normal, volvemos a usaremos el test de Levene con la mediana.

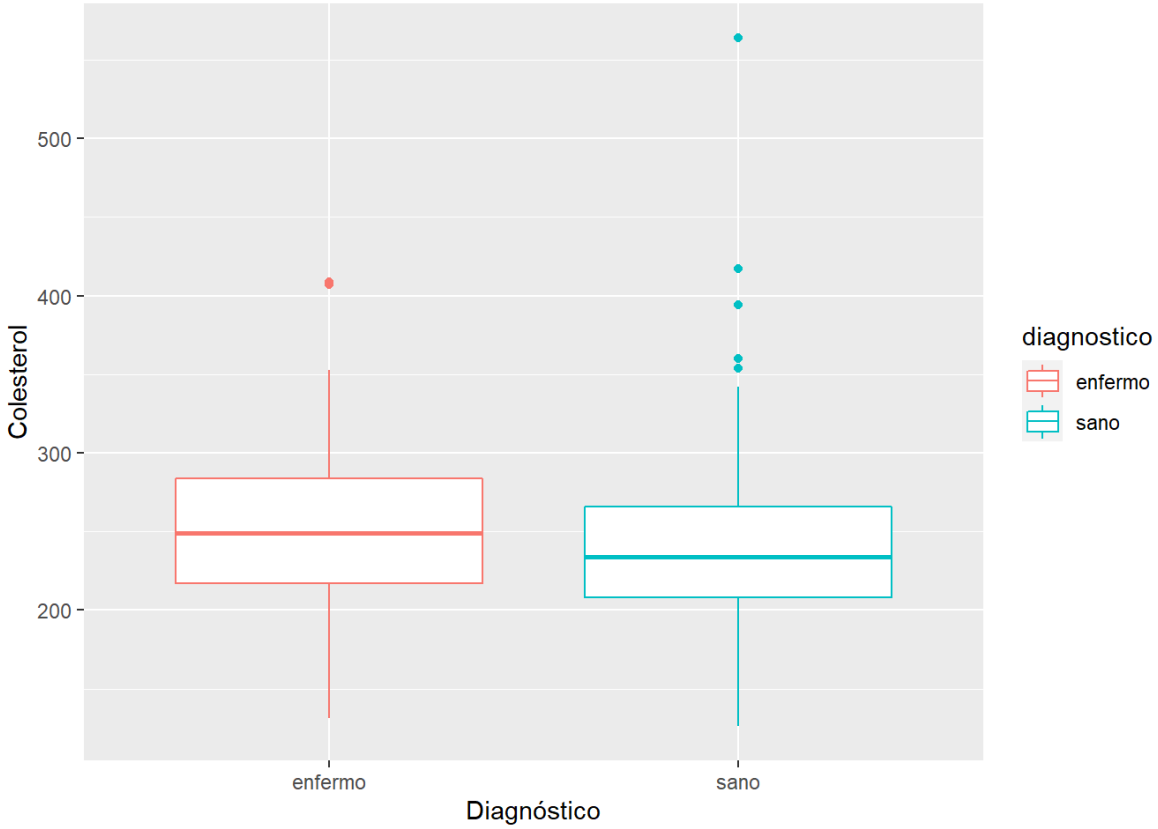
```
leveneTest(y = datos$colesterol,  group = datos$diagnostico, center = "median")
```

```
## Levene's Test for Homogeneity of Variance (center = "median")
##         Df F value Pr(>F)
## group    1   0.893 0.3449
##       1023
```

Obtenemos un p-valor > 0.05, luego no podemos rechazar la hipótesis nula, y consideraremos que la varianza no es distinta para los dos grupos.

Visto gráficamente.

```
ggplot(datos,aes(x=diagnostico)) + geom_boxplot(aes(y=colesterol, col = diagnostico)) + xlab("Diagnóstico")
+ ylab("Colesterol")
```



Para el caso de la variable *frecmax*, seguimos el mismo procedimiento para estudiar la homocedasticidad.

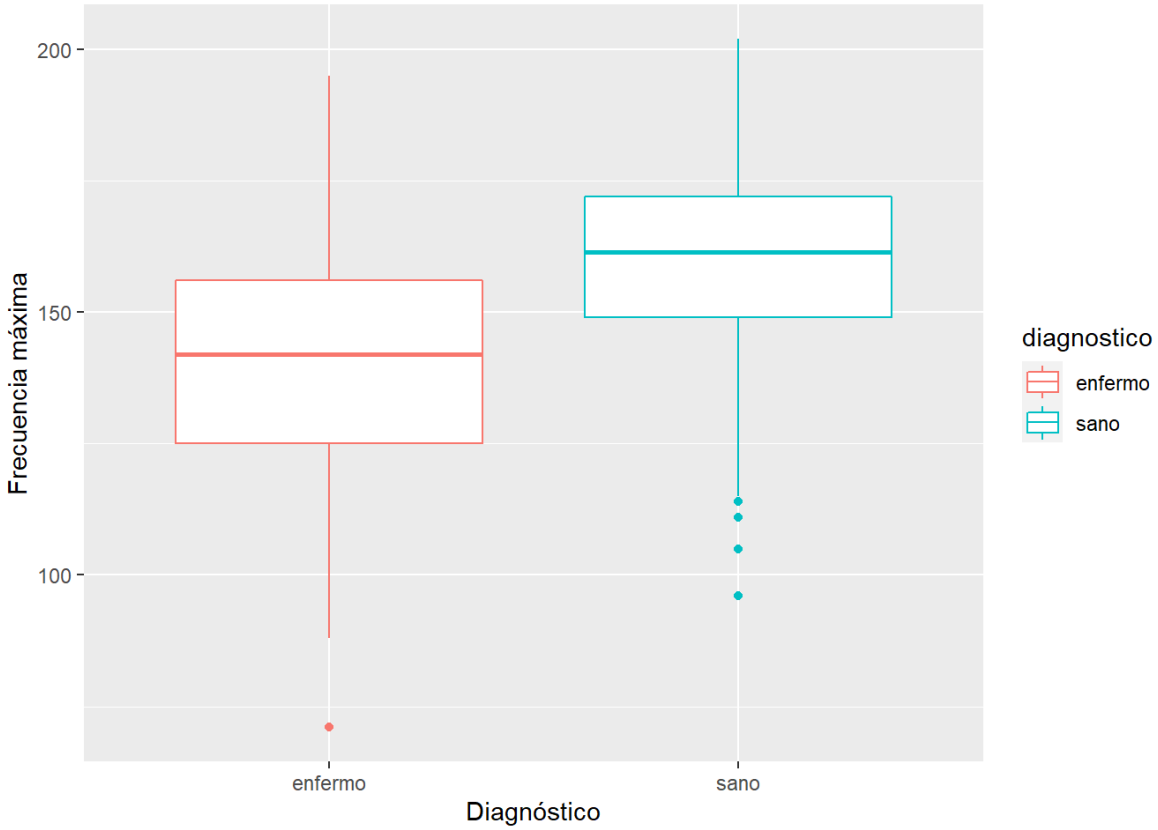
```
leveneTest(y = datos$frecmax,  group = datos$diagnostico, center = "median")
```

```
## Levene's Test for Homogeneity of Variance (center = "median")
##         Df F value    Pr(>F)
## group    1  18.195 2.179e-05 ***
##       1023
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Obtenemos un p-valor < 0.05, rechazamos por tanto la hipótesis nula, y consideraremos que la varianza es distinta.

Gráficamente.

```
ggplot(datos,aes(x=diagnostico)) + geom_boxplot(aes(y=frecmax, col = diagnostico)) + xlab("Diagnóstico")+ y
lab("Frecuencia máxima")
```



Por último, para la variable cuantitativa *depSt*, estudiamos la hogeneidad de la varianza con el test de Levene.

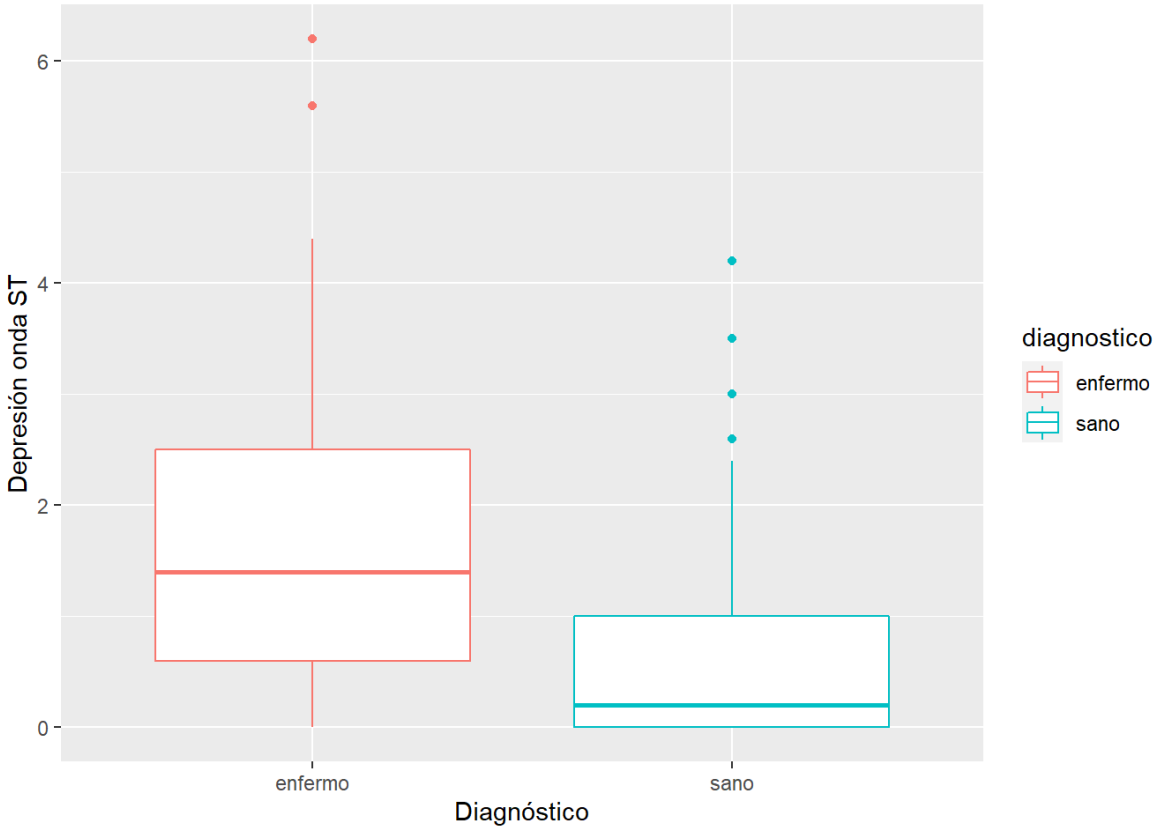
```
leveneTest(y = datos$depST,  group = datos$diagnostico, center = "median")
```

```
## Levene's Test for Homogeneity of Variance (center = "median")
##         Df F value    Pr(>F)
## group    1  115.27 < 2.2e-16 ***
##       1023
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Obtenemos un p-valor < 0.05, rechazamos por tanto la hipótesis nula, y consideraremos que la varianza es distinta.

Gráficamente.

```
ggplot(datos,aes(x=diagnostico)) + geom_boxplot(aes(y=depST, col = diagnostico)) + xlab("Diagnóstico")+ yla
b("Depresión onda ST")
```



En este caso vemos la diferencia bastante pronunciada entre los grupos como constata el valor tan alto de F del test.

4.3. Aplicación de pruebas estadísticas para comparar los grupos de datos.

En función de los datos y el objetivo del estudio, aplicar pruebas de contraste de hipótesis, correlaciones, regresiones, etc. Aplicar al menos tres métodos de análisis diferentes.

Vamos a realizar test sobre algunas de las variables para conocer en qué medida pueden afectar al diagnóstico. Todos los test los haremos con un nivel de confianza del 95% salvo que se indique lo contrario.

¿Es la edad factor de riesgo de sufrir enfermedad coronaria?

Planteamos un contraste de hipótesis:

Hipótesis nula : promedio edad enfermos = promedio edad sanos Hipótesis alternativa: promedio edad enfermos > promedio edad sanos

Realizaremos dos pruebas distintas, una suponiendo que las poblaciones son normales con varianzas poblacionales iguales, y otro test no paramétrico que no suponga normalidad de las muestras.

1. Población normal con varianzas poblacionales desconocidas pero iguales.

El estadístico de contraste en este caso corresponde a una observación de una distribución t de Student con n1 + n2 - 2 grados de libertad.

$$t = \frac{\overline{x_1} - \overline{x_2}}{s\sqrt{\frac{1}{n_1} + \frac{1}{n_2}}}$$

Dividimos los datos en los dos grupos que vamos a estudiar, las edades de los pacientes enfermos y las edades de los pacientes sanos.

```
enfermos <- datos$edad[datos$diagnostico == "enfermo"]
sanos <- datos$edad[datos$diagnostico == "sano"]
```

Realizamos el test

```
x1 <- mean(enfermos) # media enfermos
x2 <- mean(sanos) # media sanos
n1 <- length(enfermos) # número de muestras enfermos
n2 <- length(sanos) # número de muestras sanos
df = n1+n2-2 # grados de libertad
s1 <- sd(enfermos) # desviación estándar muestral enfermos
s2 <- sd(sanos) # desviación estándar muestral sanos

s <- sqrt(((n1-1)*s1^2 + (n2-1)*s2^2) / df)
t <- (x1-x2)/( s*sqrt((1/n1)+(1/n2))) # estadístico de contraste
p.valor <- pt(t, df,lower.tail = FALSE) # probabilidad cola derecha de la distribución t
p.valor
```

```
## [1] 5.33861e-14
```

Según el p-valor obtenido en comparación con el nivel de significación escogido (p-valor < 0.05), rechazaremos la hipótesis nula en favor de la hipótesis alternativa, es decir, que la edad promedio de los pacientes enfermos es mayor que la edad promedio de los paciente sanos.

2. Poblaciones no normales. Si no podemos asumir normalidad, utilizamos el test U de Mann-Whitney (Wilcoxon) que se puede aplicar cuando los datos son independientes.

La función wilcox.test realiza una prueba de suma de rango de Wilcoxox comparando las medianas de las distribuciones.

Hipótesis nula: Mediana de edad de pacientes enfermos = Mediana de edad de pacientes sanos.
Hipótesis alternativa: Mediana de edad de pacientes enfermos > Mediana de edad de pacientes sanos.

Relizamos el test con la función *wilcox.test*.

```
wilcox.test(x = enfermos, y = sanos, alternative = "greater", conf.level =  0.95, conf.int = TRUE)
```

```
##
##  Wilcoxon rank sum test with continuity correction
##
## data:  enfermos and sanos
## W = 167644, p-value = 7.338e-15
## alternative hypothesis: true location shift is greater than 0
## 95 percent confidence interval:
##  3.999999      Inf
## sample estimates:
## difference in location
##               4.999965
```

El resultado del test es significativo, nos proporciona un p-valor < 0.05, por lo tanto rechazamos la hipótesis nula, y podemos considerar que la mediana de los pacientes enfermos es mayor que la mediana de los pacientes sanos.

Este resultado corrobora el obtenido en el primer test suponiendo normalidad de las muestras.

¿Es el sexo del paciente relevante para obtener un diagnóstico?
Queremos saber si el sexo es independiente del diagnóstico obtenido por el paciente.

Establecemos la siguiente hipótesis: Hipótesis nula: El sexo del paciente y su diagnóstico son independientes.
Hipótesis alternativa: El sexo del paciente y su diagnóstico no son independientes.

Las dos variables son cualititativas, aplicaremos un contraste chi-cuadrado para determinar si las dos variables son independientes.
Que dos variables sean independientes significa que no tienen relación, y que por lo tanto una no depende de la otra, ni viceversa.
Suponemos que las muestras son independientes y comprobamos que todos los valores esperados son mayores que 5 para poder aplicar el test.

El estadístico chi-cuadrado tomará un valor igual a 0 si existe concordancia perfecta entre las frecuencias observadas y las esperadas y tomará un valor grande si existe una gran discrepancia entre estas frecuencias, y consecuentemente se deberá rechazar la hipótesis nula.

Obtenemos primero la matriz de contingencia.

```
mat.contingencia <- table(datos$diagnostico, datos$sexo)
mat.contingencia
```

```
##
##           hombre mujer
##   enfermo    413    86
##   sano       300   226
```

Y con ella realizamos el test.

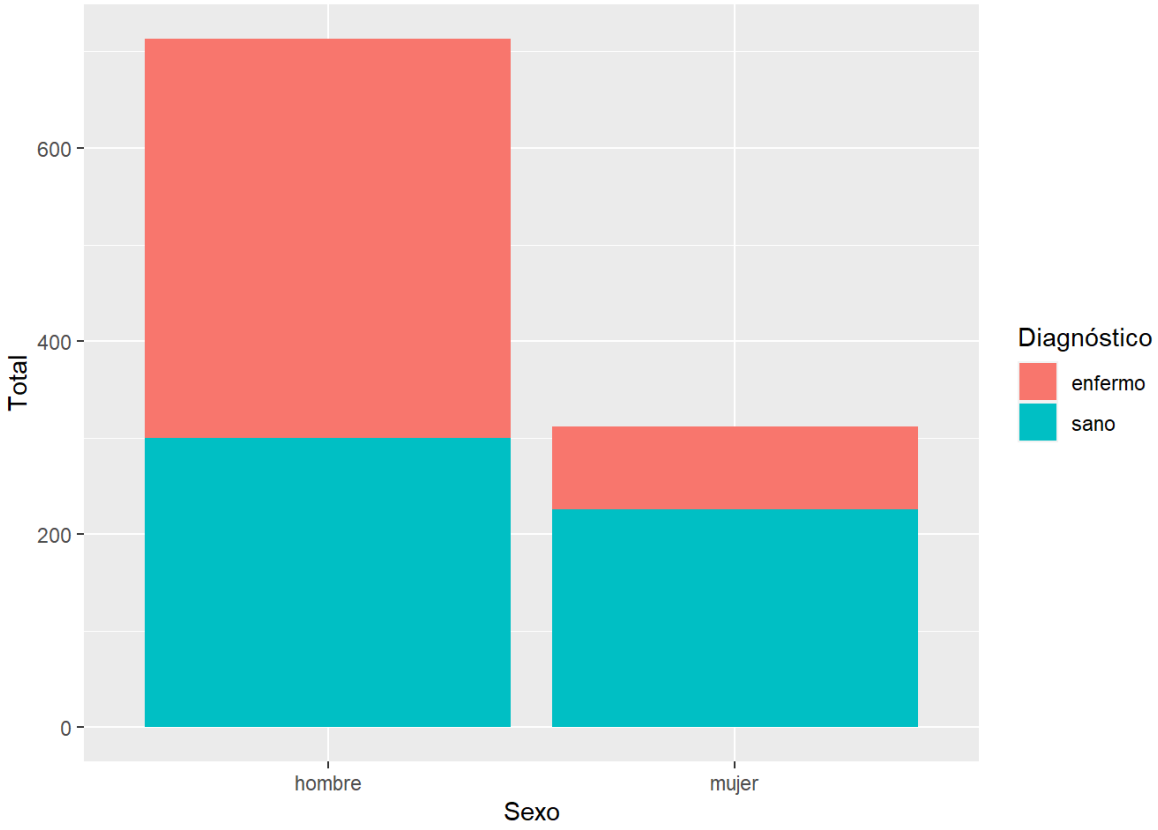
```
chisq.test(mat.contingencia)
```

```
##
##  Pearson's Chi-squared test with Yates' continuity correction
##
## data:  mat.contingencia
## X-squared = 78.863, df = 1, p-value < 2.2e-16
```

La prueba es significativa (p-valor < 0.05), el valor p está por debajo del nivel de significación, así que rechazamos la hipótesis nula y aceptamos la hipótesis alternativa, por lo tanto las dos variables no son independientes.

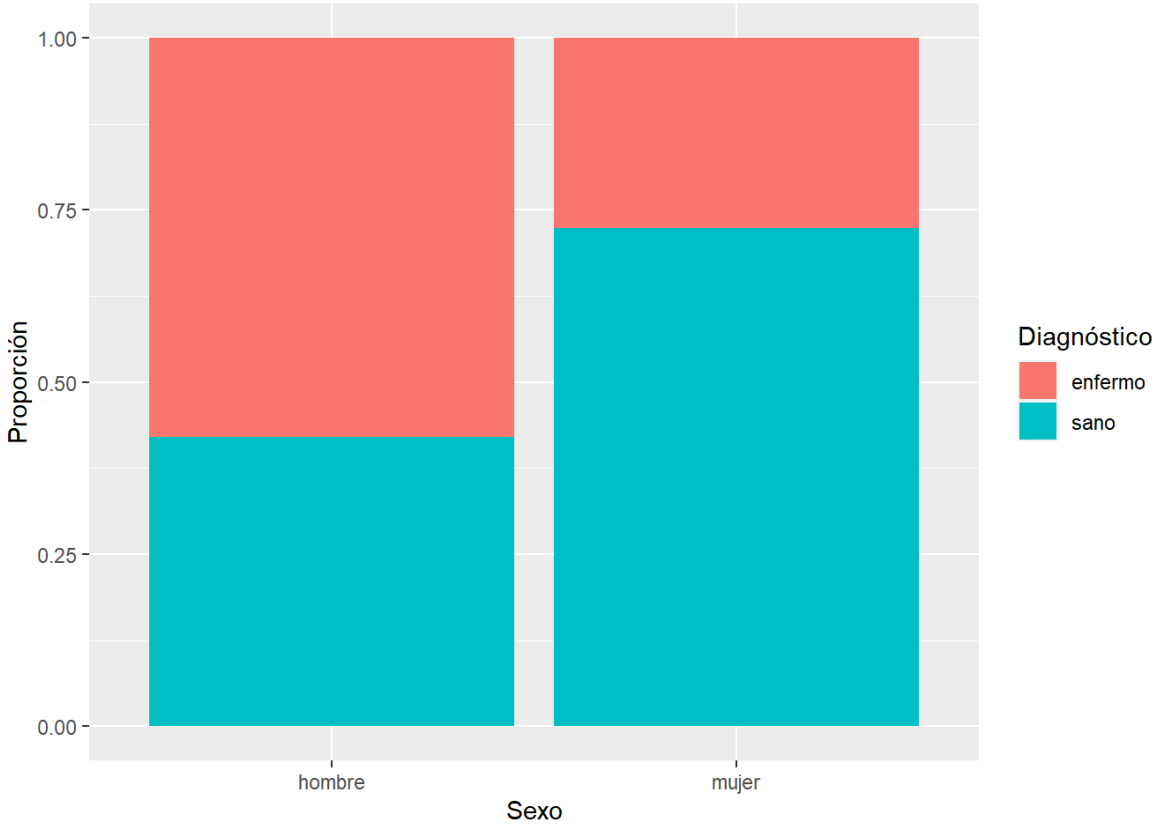
Podemos visualizar las dos variables.

```
ggplot(datos,aes(x=sexo,fill=diagnostico)) +geom_bar() + xlab("Sexo") + ylab("Total") + scale_fill_discrete
(name="Diagnóstico")
```



Y como proporción.

```
ggplot(datos,aes(x=sexo,fill=diagnostico)) +geom_bar(position="fill") + xlab("Sexo") + ylab("Proporción") +
scale_fill_discrete(name="Diagnóstico")
```



O visto en forma de mosaico.

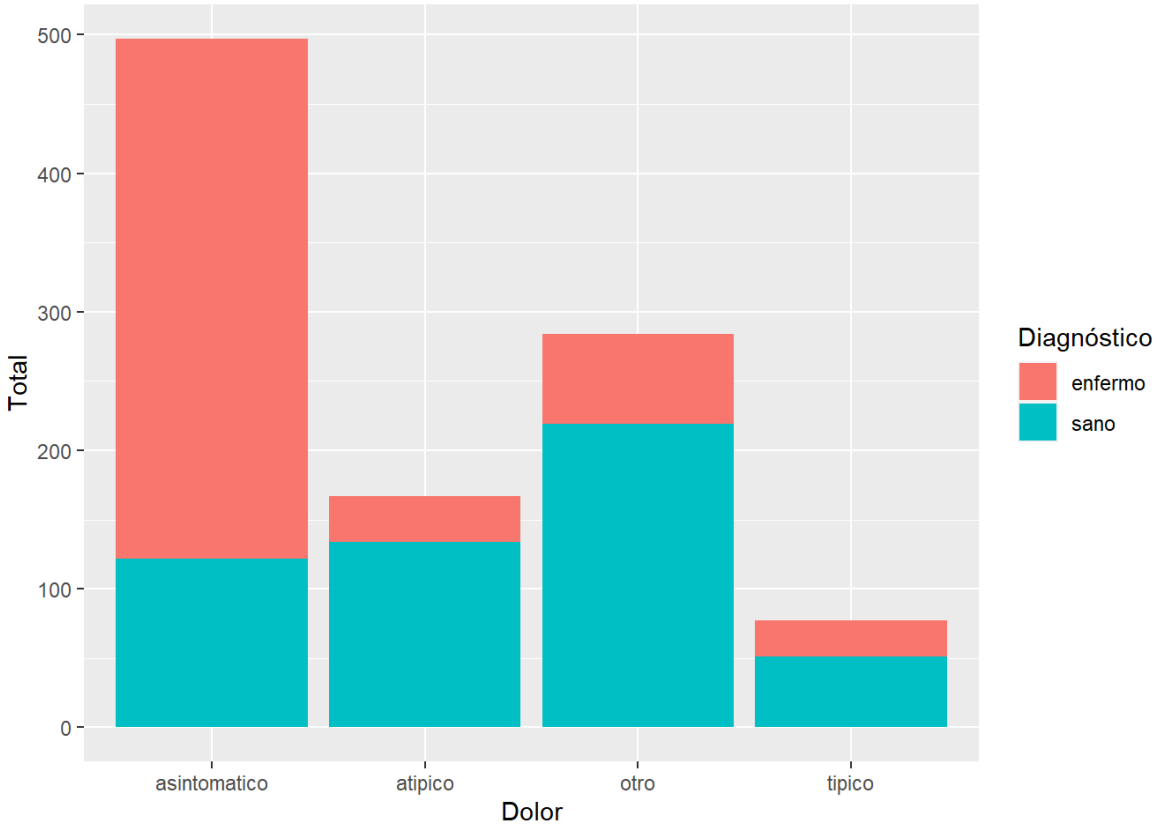
```
mosaicplot(mat.contingencia, color=TRUE, main="Plot de mosaico")
```

Para el resto de variables podemos proceder de la misma forma.

Visualizaremos la relación entre el resto de variables categóricas y el diagnóstico para tener una idea aproximada de la incidencia que pueden tener sobre el resultado.
Las variables cuantitativas ya las hemos visualizado gráficamente al estudiar su homocedasticidad.

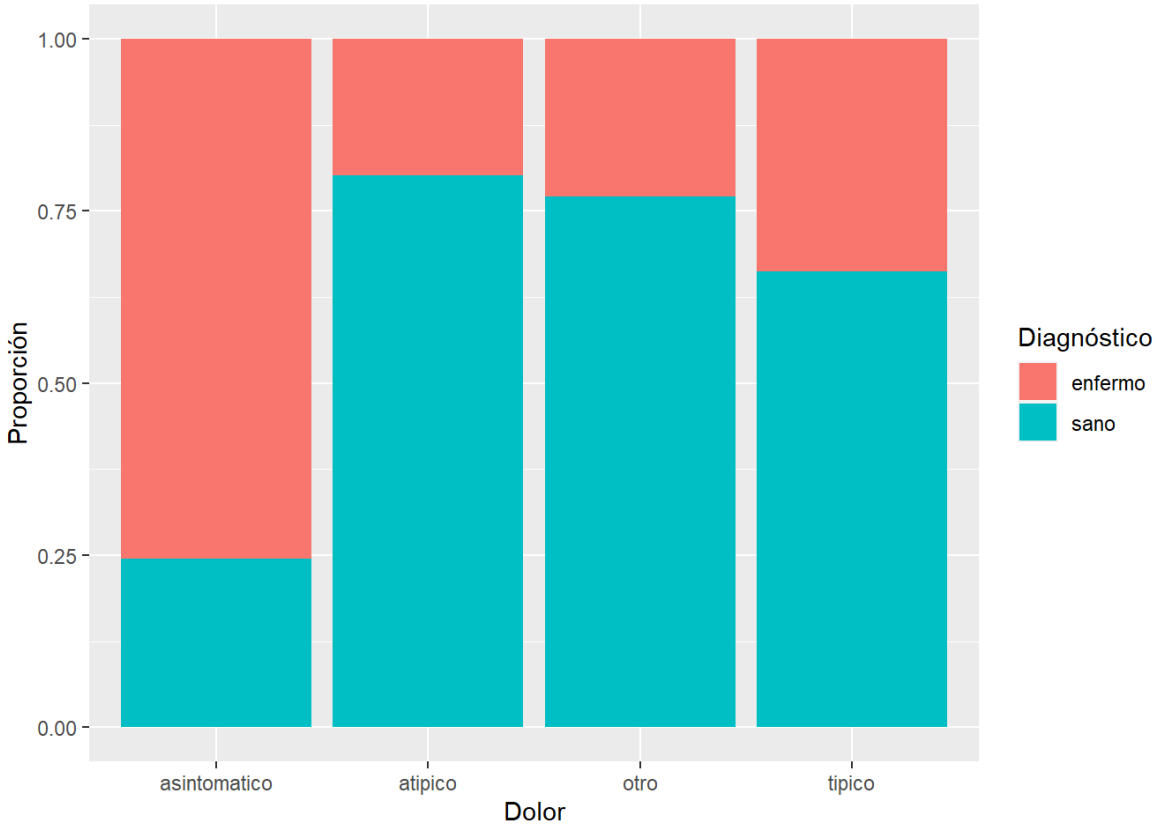
Variable *dolor*.

```
ggplot(datos,aes(x=dolor,fill=diagnostico)) +geom_bar() + xlab("Dolor") + ylab("Total") + scale_fill_discrete(name="Diagnóstico")
```



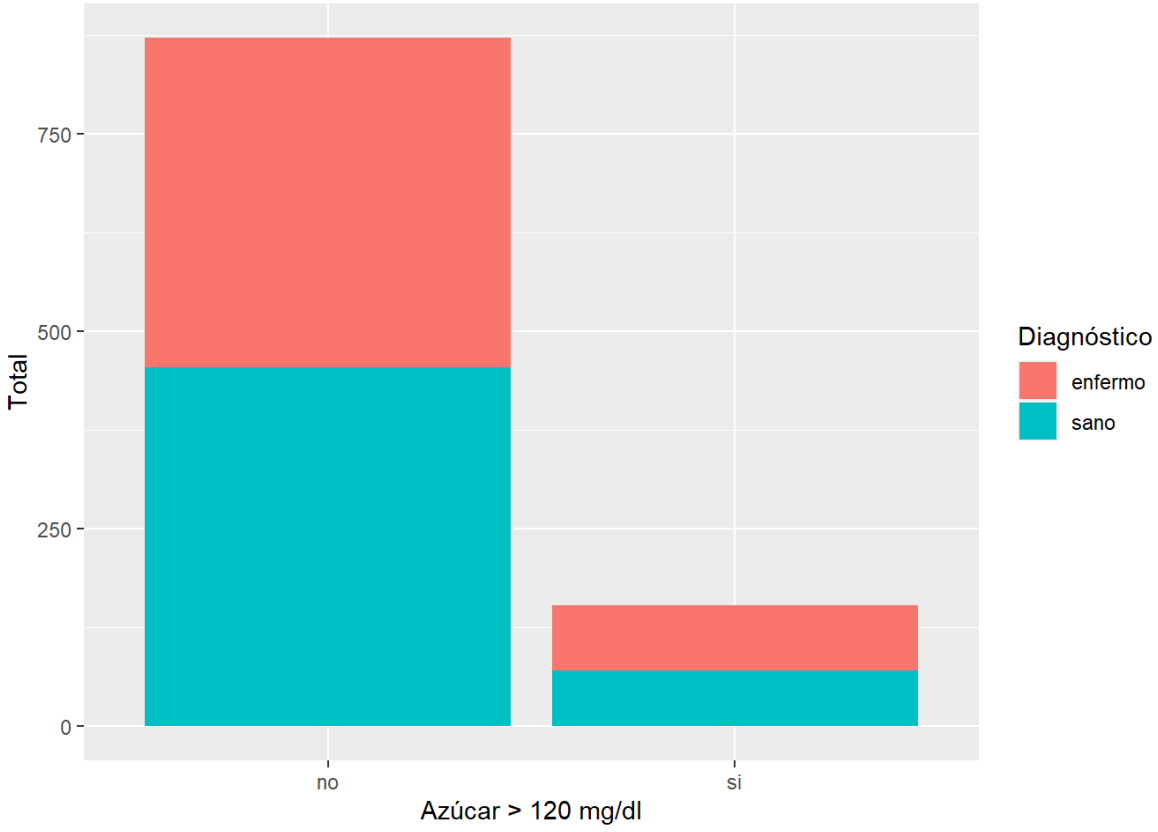
Visto en proporción.

```
ggplot(datos,aes(x=dolor,fill=diagnostico)) +geom_bar(position="fill") + xlab("Dolor") + ylab("Proporción") + scale_fill_discrete(name="Diagnóstico")
```



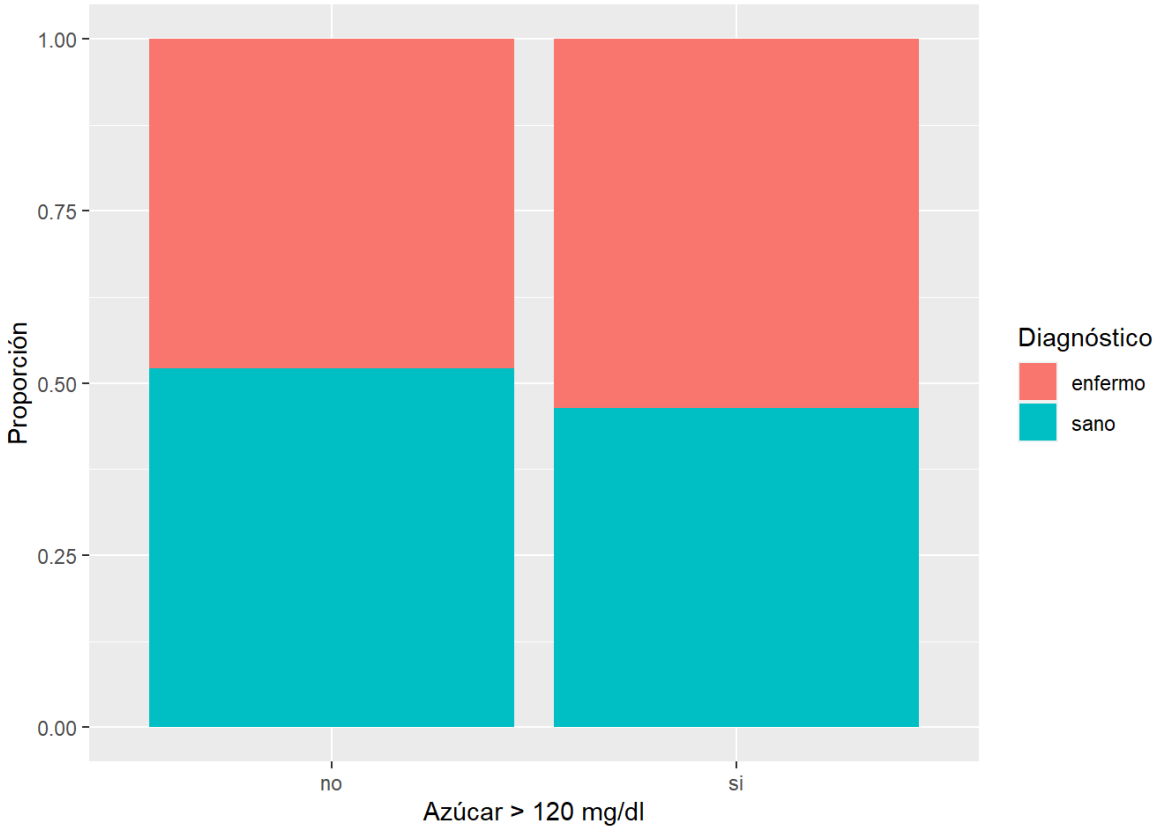
Variable *azucar* en sangre.

```
ggplot(datos,aes(x=azucar,fill=diagnostico)) +geom_bar() + xlab("Azúcar > 120 mg/dl") + ylab("Total") + sca
le_fill_discrete(name="Diagnóstico")
```



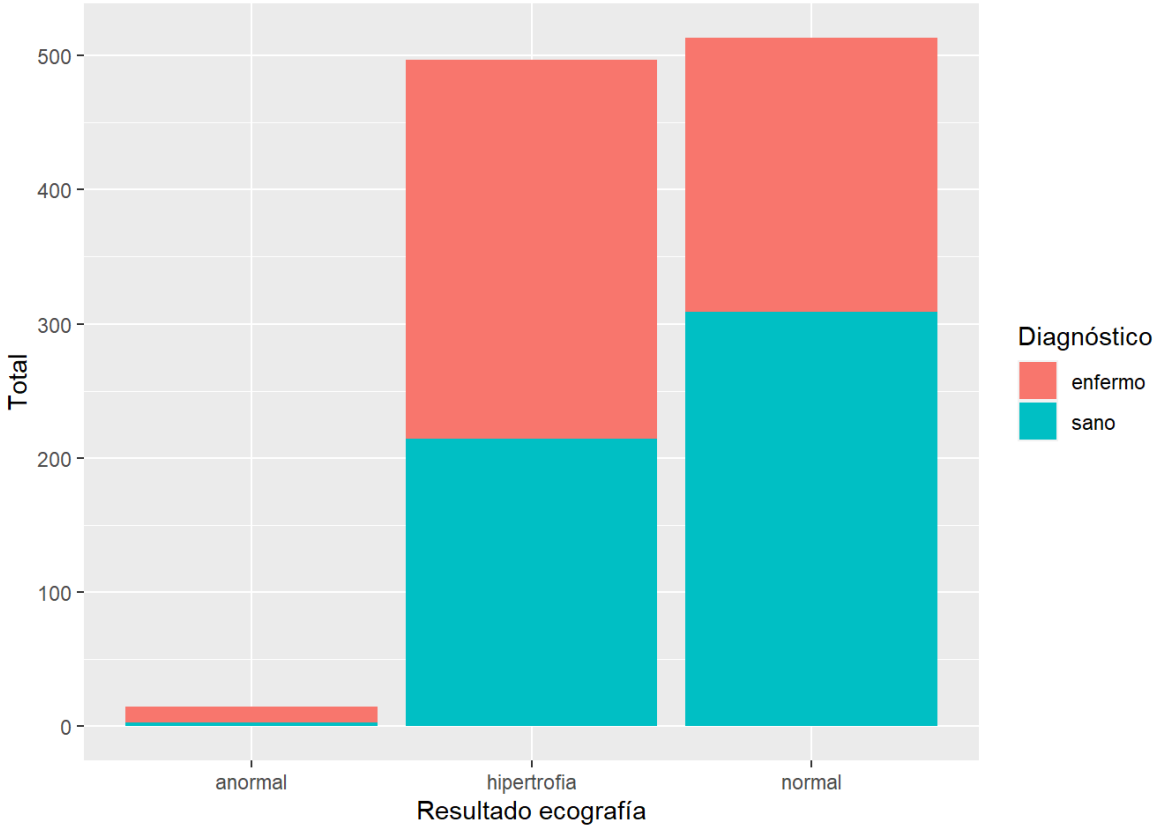
En proporción.

```
ggplot(datos,aes(x=azucar,fill=diagnostico)) +geom_bar(position="fill") + xlab("Azúcar > 120 mg/dl") + ylab
("Proporción") + scale_fill_discrete(name="Diagnóstico")
```



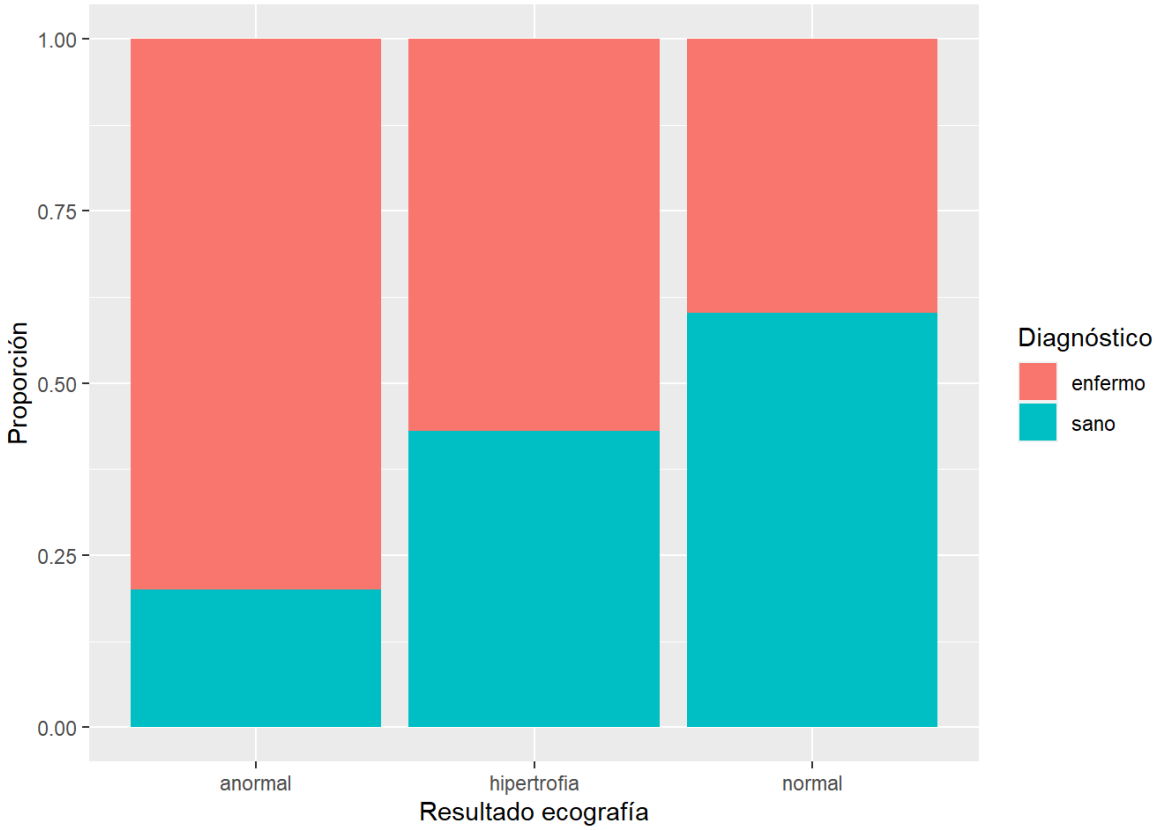
Variable *ecografía*.

```
ggplot(datos,aes(x=ecografia,fill=diagnostico)) +geom_bar() + xlab("Resultado ecografía") + ylab("Total") +
scale_fill_discrete(name="Diagnóstico")
```



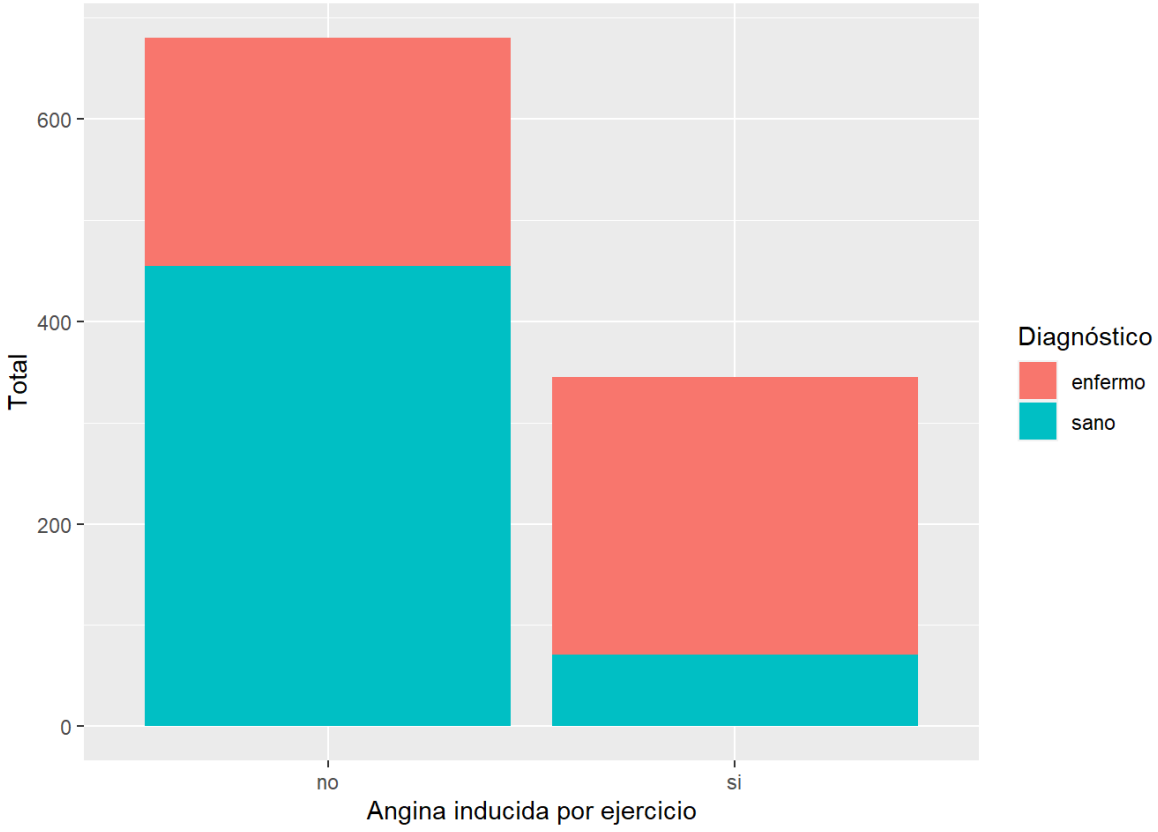
En proporción.

```
ggplot(datos,aes(x=ecografia,fill=diagnostico)) +geom_bar(position="fill") + xlab("Resultado ecografía") +
 ylab("Proporción") + scale_fill_discrete(name="Diagnóstico")
```



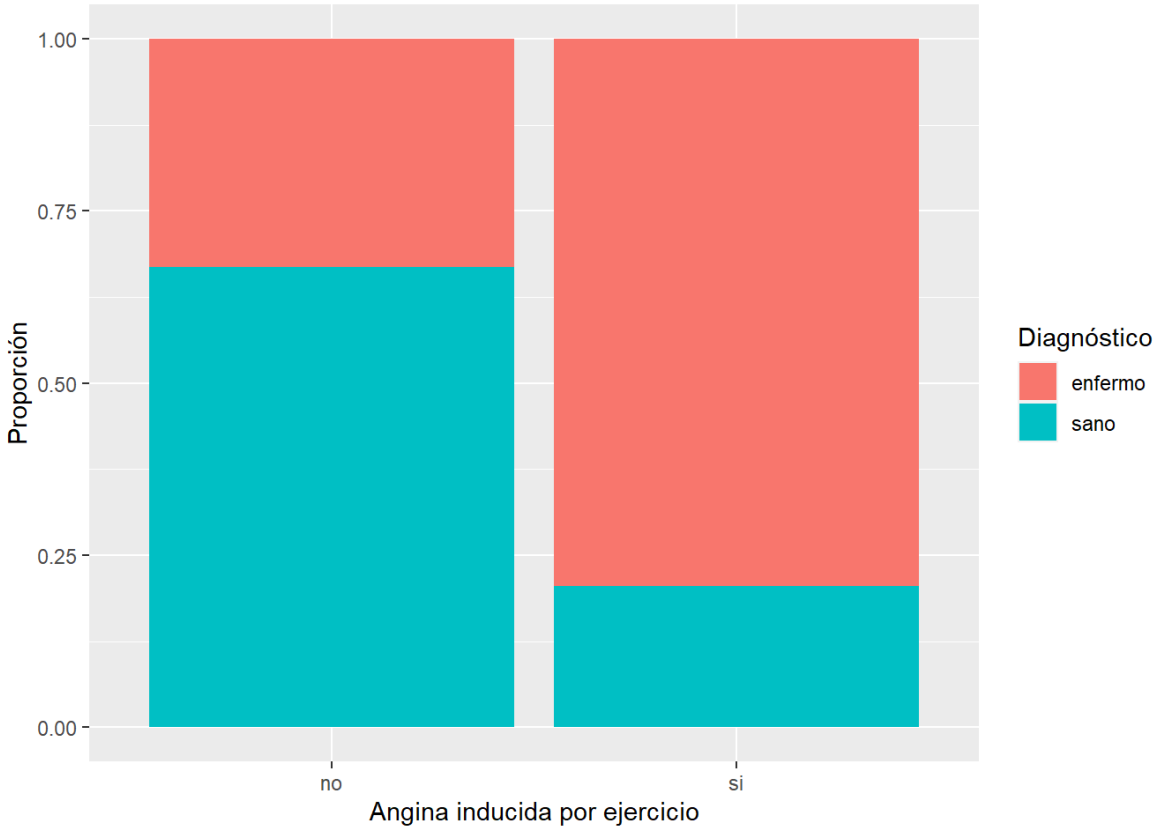
Variable *ejercicio* (angina inducida por el ejercicio físico).

```
ggplot(datos,aes(x=ejercicio,fill=diagnostico)) +geom_bar() + xlab("Angina inducida por ejercicio") + ylab("Total") + scale_fill_discrete(name="Diagnóstico")
```



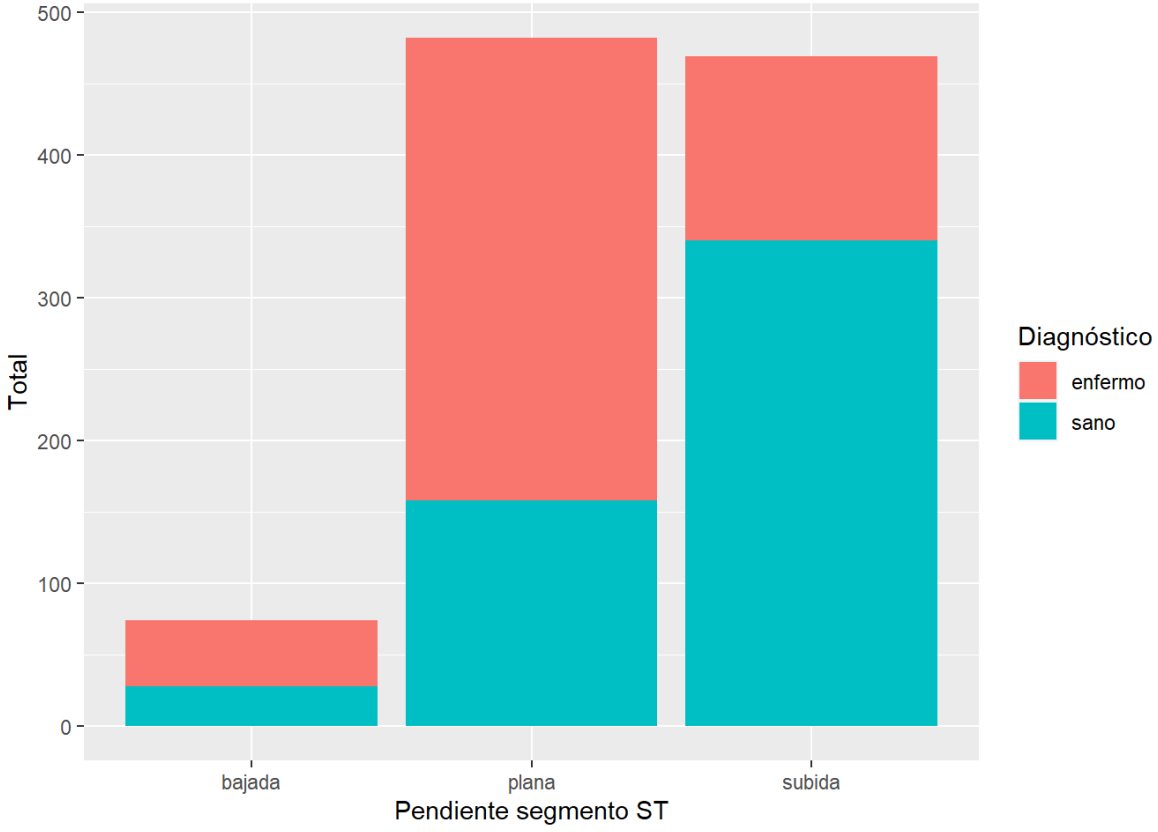
En proporción.

```
ggplot(datos,aes(x=ejercicio,fill=diagnostico)) +geom_bar(position="fill") + xlab("Angina inducida por ejercicio") + ylab("Proporción") + scale_fill_discrete(name="Diagnóstico")
```



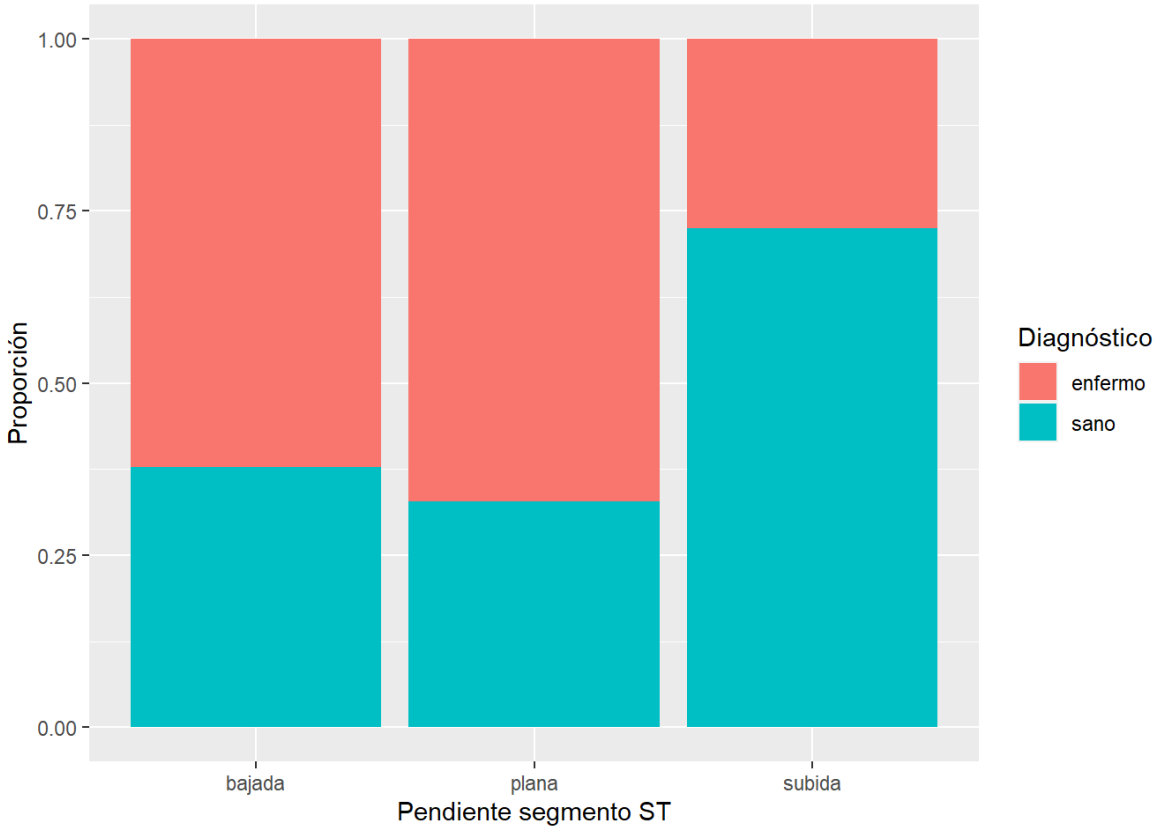
Variable *pendiente* (pendiente del segmento ST del electrocardiograma en el pico de ejercicio).

```
ggplot(datos,aes(x=pendiente,fill=diagnostico)) +geom_bar() + xlab("Pendiente segmento ST") + ylab("Total")
+ scale_fill_discrete(name="Diagnóstico")
```



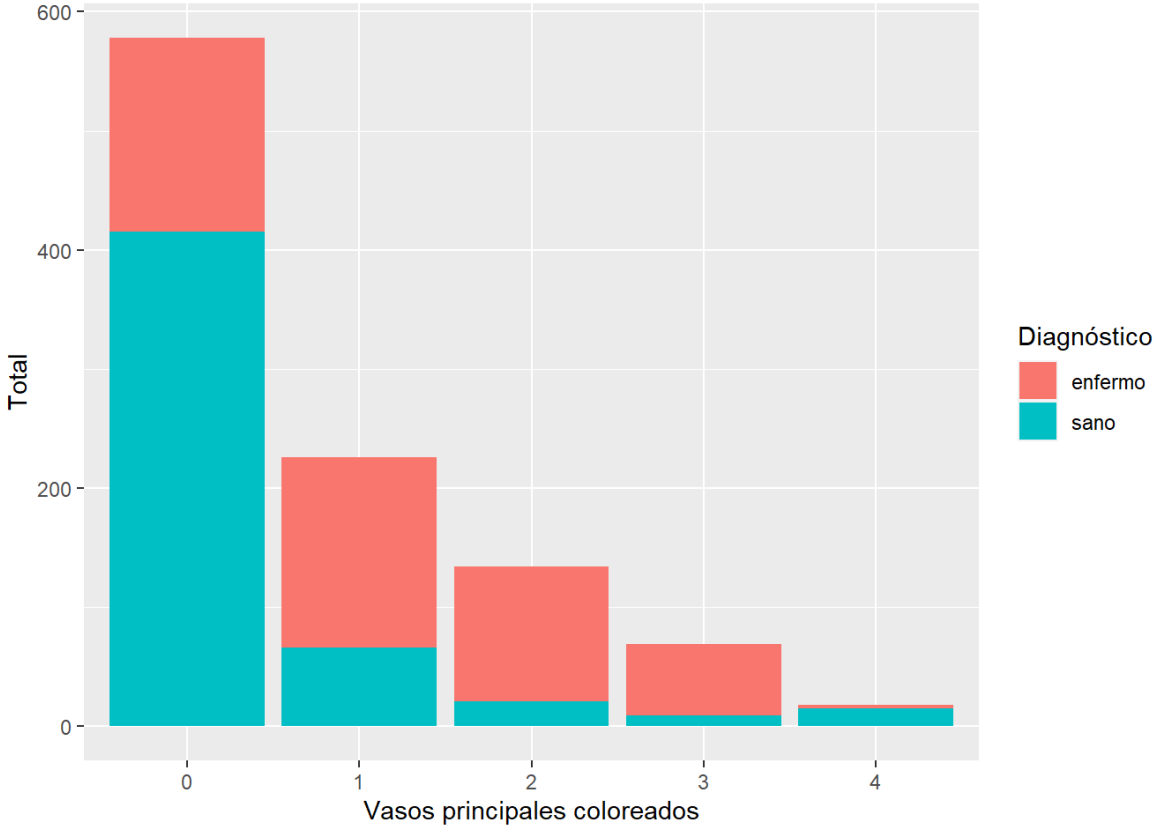
En proporción.

```
ggplot(datos,aes(x=pendiente,fill=diagnostico)) +geom_bar(position="fill") + xlab("Pendiente segmento ST")
 + ylab("Proporción") + scale_fill_discrete(name="Diagnóstico")
```



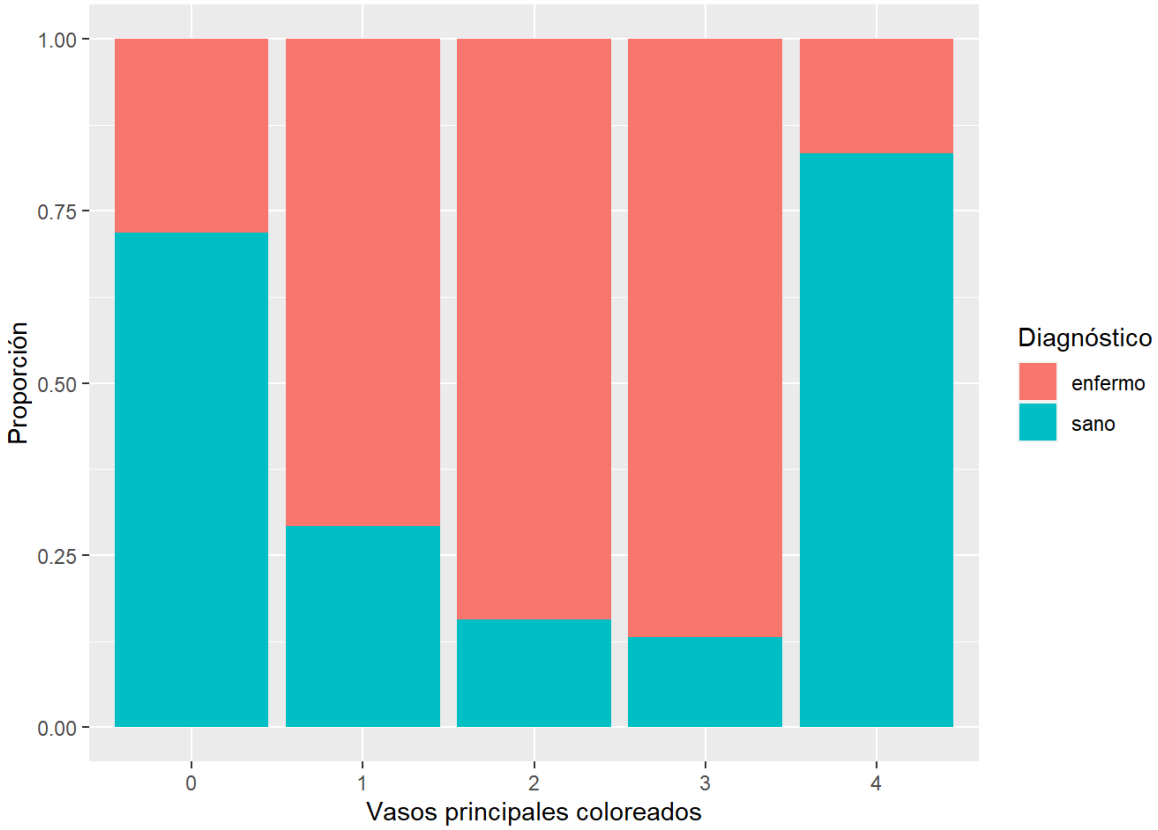
Variable *vasos.coloreados* (número de vasos sanguíneos principales coloreados en la fluoroscopia).

```
ggplot(datos,aes(x=vasos.coloreados,fill=diagnostico)) +geom_bar() + xlab("Vasos principales coloreados") +
ylab("Total") + scale_fill_discrete(name="Diagnóstico")
```



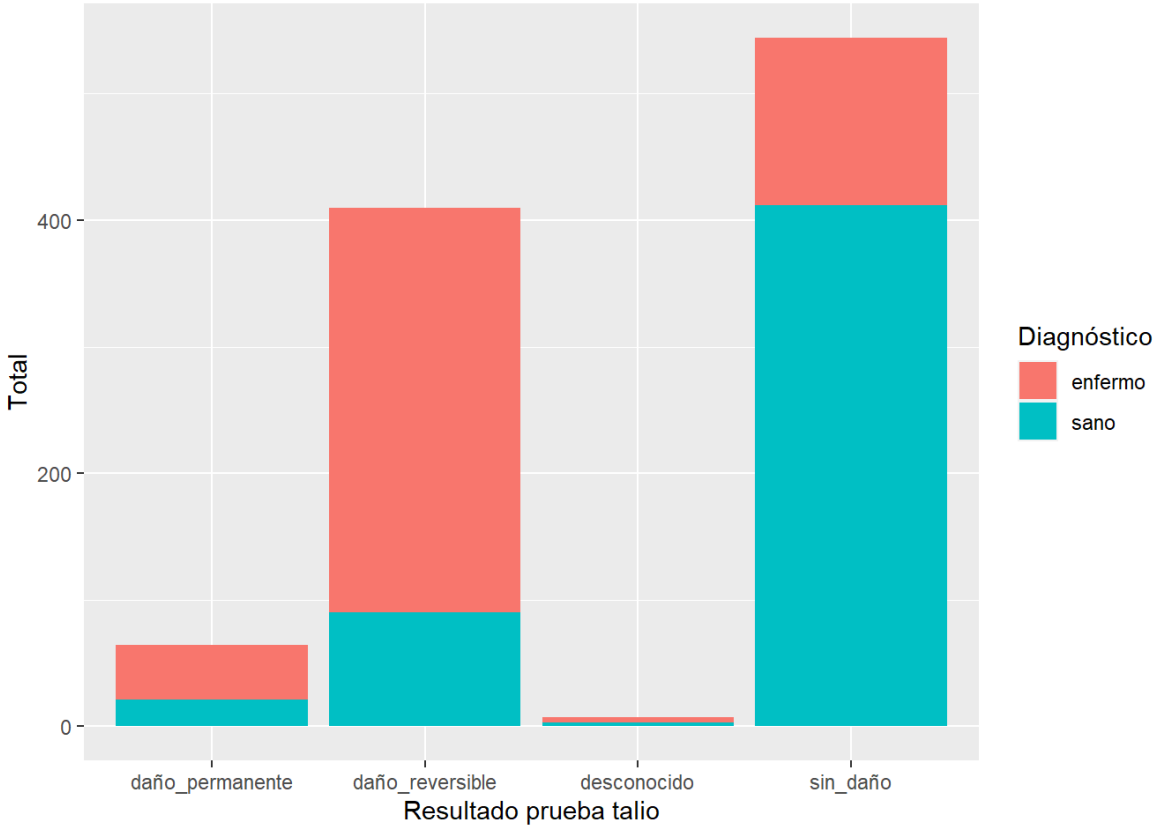
En proporción.

```
ggplot(datos,aes(x=vasos.coloreados,fill=diagnostico)) +geom_bar(position="fill") + xlab("Vasos principales
coloreados") + ylab("Proporción") + scale_fill_discrete(name="Diagnóstico")
```



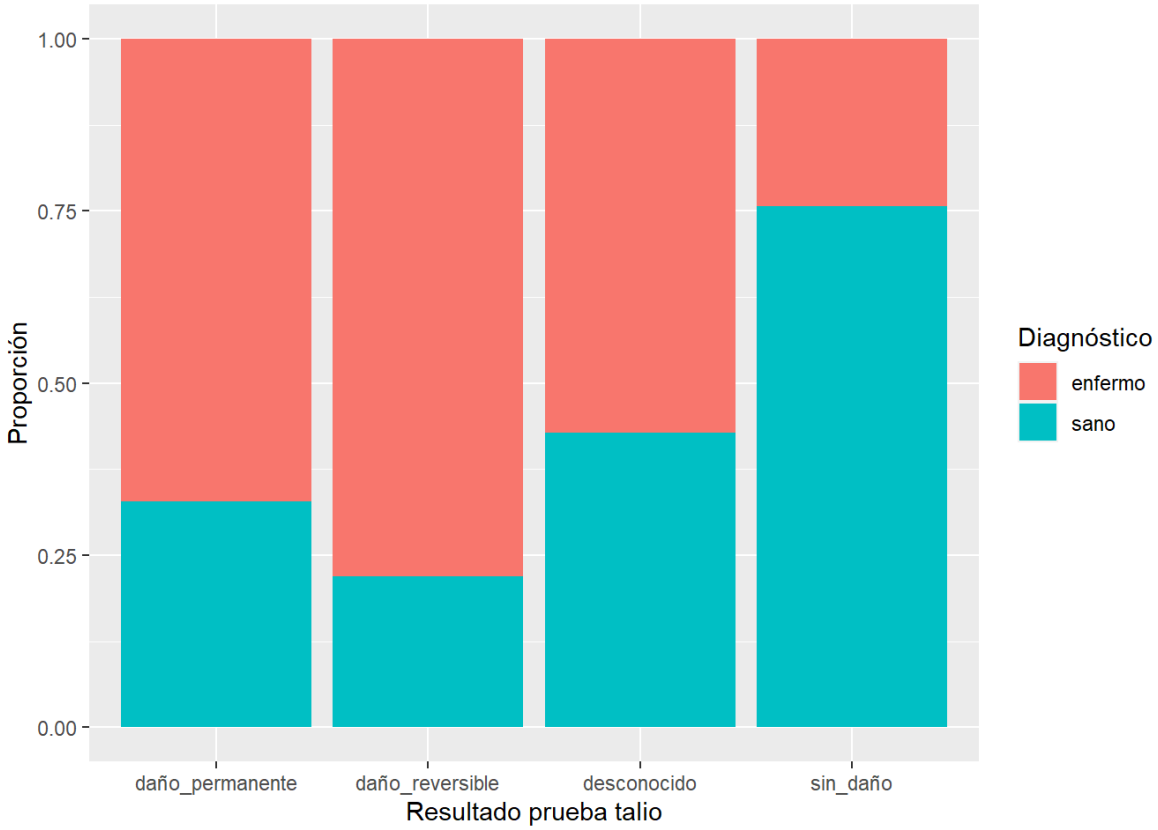
Variable *exploracion.talio* (resultado de la prueba de esfuerzo con talio).

```
ggplot(datos,aes(x=exploracion.talio,fill=diagnostico)) +geom_bar() + xlab("Resultado prueba talio") + ylab
("Total") + scale_fill_discrete(name="Diagnóstico")
```



En proporción.

```
ggplot(datos,aes(x=exploracion.talio,fill=diagnostico)) +geom_bar(position="fill") + xlab("Resultado prueba
talio") + ylab("Proporción") + scale_fill_discrete(name="Diagnóstico")
```



Vamos a generar con los datos un modelo de **regresión logística** para poder predecir el valor de la variable diagnóstico. La regresión logística nos proporciona una estimación de probabilidad para la predicción.

En una primera aproximación utilizaremos todas las variables disponibles para generar el modelo y luego eliminaremos aquellas que no sean significativas para el resultado.

```
modelo.diagnostico.todas <- glm(datos$diagnostico ~ datos$edad + datos$sexo + datos$dolor + datos$tension +
datos$colesterol + datos$azucar + datos$ecografia + datos$frecmax + datos$ejercicio + datos$depST + datos$p
endiente + datos$vasos.coloreados + datos$exploracion.talio , data = datos, family = "binomial")
summary(modelo.diagnostico.todas)
```

```
##
## Call:
## glm(formula = datos$diagnostico ~ datos$edad + datos$sexo + datos$dolor +
##     datos$tension + datos$colesterol + datos$azucar + datos$ecografia +
##     datos$frecmax + datos$ejercicio + datos$depST + datos$pendiente +
##     datos$vasos.coloreados + datos$exploracion.talio, family = "binomial",
##     data = datos)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.8582  -0.2917   0.0718   0.4167   3.1908
##
## Coefficients:
##                                       Estimate Std. Error z value Pr(>|z|)
## (Intercept)                          -0.077851   2.171147  -0.036 0.971396
## datos$edad                            0.026846   0.013950   1.924 0.054297
## datos$sexomujer                       1.992347   0.314204   6.341 2.28e-10
## datos$doloratipico                    0.886380   0.308803   2.870 0.004100
## datos$dolorotro                       2.006394   0.286281   7.008 2.41e-12
## datos$dolortipico                     2.409722   0.391965   6.148 7.86e-10
## datos$tension                        -0.024979   0.006537  -3.821 0.000133
## datos$colesterol                     -0.005462   0.002307  -2.367 0.017914
## datos$azucarsi                        0.380096   0.319620   1.189 0.234356
## datos$ecografiahipertrofia            0.800417   1.536998   0.521 0.602530
## datos$ecografianormal                 1.197685   1.538426   0.779 0.436267
## datos$frecmax                         0.021692   0.006525   3.324 0.000886
## datos$ejerciciosi                    -0.750331   0.248746  -3.016 0.002557
## datos$depST                          -0.403411   0.132156  -3.053 0.002269
## datos$pendienteplana                 -0.595618   0.472076  -1.262 0.207057
## datos$pendientesubida                 0.799689   0.504500   1.585 0.112941
## datos$vasos.coloreados1              -2.334076   0.286781  -8.139 3.99e-16
## datos$vasos.coloreados2              -3.597039   0.444870  -8.086 6.19e-16
## datos$vasos.coloreados3              -2.288131   0.532138  -4.300 1.71e-05
## datos$vasos.coloreados4               1.565677   0.930256   1.683 0.092363
## datos$exploracion.taliodaño_reversible -1.805570   0.435590  -4.145 3.40e-05
## datos$exploracion.taliodesconocido    -2.796813   1.466219  -1.908 0.056456
## datos$exploracion.taliosin_daño       -0.392167   0.441819  -0.888 0.374745
##
## (Intercept)
## datos$edad                           .
## datos$sexomujer                      ***
## datos$doloratipico                   **
## datos$dolorotro                      ***
## datos$dolortipico                    ***
## datos$tension                        ***
## datos$colesterol                     *
## datos$azucarsi
## datos$ecografiahipertrofia
## datos$ecografianormal
## datos$frecmax                        ***
## datos$ejerciciosi                    **
## datos$depST                          **
## datos$pendienteplana
## datos$pendientesubida
## datos$vasos.coloreados1              ***
## datos$vasos.coloreados2              ***
## datos$vasos.coloreados3              ***
## datos$vasos.coloreados4              .
## datos$exploracion.taliodaño_reversible ***
## datos$exploracion.taliodesconocido    .
## datos$exploracion.taliosin_daño
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
```

```
##
##     Null deviance: 1420.24  on 1024  degrees of freedom
## Residual deviance:  606.82  on 1002  degrees of freedom
## AIC: 652.82
##
## Number of Fisher Scoring iterations: 6
```

La función *glm* nos ha generado las variables auxiliares para cada uno de los valores de las variables categóricas. El nivel base de referencia que usa R por defecto es el primer nivel de la variable de tipo factor e interpreta el resto de niveles en base a este nivel.

Estudiamos la bondad del ajuste del modelo que hemos obtenido con el test de Hosman-Lemeshow. En la librería (ResourceSelection) hay una función que ajusta el test de Hosmer- Lemeshow.

En el test de Hosman-Lemeshow la hipótesis nula es que los valores observados corresponden con los valores esperados. Realizamos el test al modelo.

```
hl <- hoslem.test(modelo.diagnostico.todas$y, fitted(modelo.diagnostico.todas))
hl
```

```
##
##  Hosmer and Lemeshow goodness of fit (GOF) test
##
## data:  modelo.diagnostico.todas$y, fitted(modelo.diagnostico.todas)
## X-squared = 83.157, df = 8, p-value = 1.132e-14
```

Obtenemos un valor p < 0.05, que es significativo, con lo que podemos rechazar la hipótesis nula de que los valores esperados se corresponden a los valores predichos por el modelo de regresión logística.

Podemos visualizar las diferencias entre los valores esperados y los obtenidos por el modelo.

```
expected <- round(hl$expected, 0)
observed <- round(hl$observed, 0)
cbind(expected, observed)
```

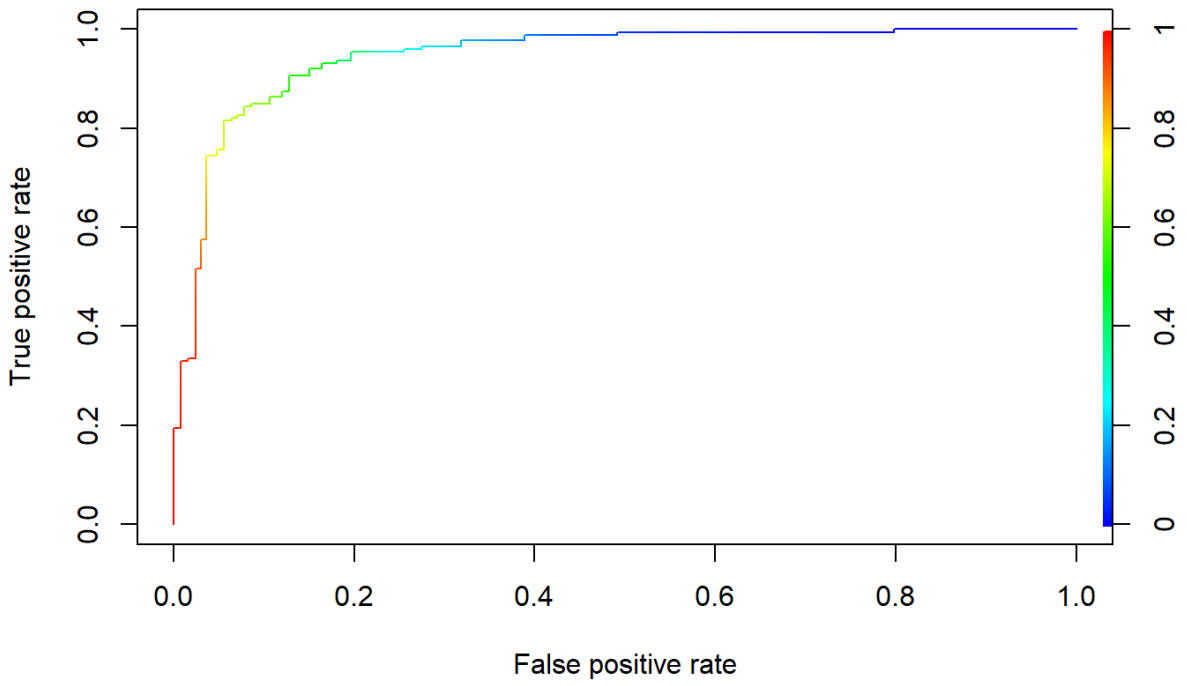
```
##                    yhat0 yhat1  y0  y1
## [0.000447,0.00615]   104     0 101   3
## (0.00615,0.0245]     102     1 103   0
## (0.0245,0.113]        98     6 101   3
## (0.113,0.291]         80    19  81  18
## (0.291,0.587]         55    49  56  48
## (0.587,0.744]         32    69  39  62
## (0.744,0.883]         18    84   3  99
## (0.883,0.96]           7    98   3 102
## (0.96,0.983]           3    98  12  89
## (0.983,0.999]          1   101   0 102
```

```
difer<-abs(expected - observed)
colSums(difer)
```

```
## yhat0 yhat1
##    45    45
```

Si dibujamos la curva ROC.

```
predicciones <- predict(modelo.diagnostico.todas, type = "response")
pred <- prediction(predicciones, datos$diagnostico)
perf <- performance(pred,"tpr","fpr")
plot(perf,colorize=TRUE)
```



El área bajo la curva es.

```
as.numeric(performance(pred,"auc")@y.values)
```

```
## [1] 0.9464823
```

La precisión del modelo es bastante alta, un 0.94, significa que hay un 94% de posibilidades de que el modelo pueda distinguir entre un paciente sano y otro enfermo.

Reducción del modelo:

Si revisamos los coeficientes del modelo vemos que solo algunos de ellos tienen valores p significativos, podemos suponer que solo éstos realizan una contribución significativa a la predicción del resultado.

El resto los eliminamos del modelo, quedando de la siguiente forma.

```
modelo.diagnostico <- glm(datos$diagnostico ~ datos$sexo + datos$dolor + datos$tension + datos$colesterol +
datos$frecmax + datos$ejercicio + datos$depST  + datos$vasos.coloreados + datos$exploracion.talio , data =
datos, family = "binomial")
summary(modelo.diagnostico)
```

```
##
## Call:
## glm(formula = datos$diagnostico ~ datos$sexo + datos$dolor +
##     datos$tension + datos$colesterol + datos$frecmax + datos$ejercicio +
##     datos$depST + datos$vasos.coloreados + datos$exploracion.talio,
##     family = "binomial", data = datos)
##
## Deviance Residuals:
##      Min        1Q    Median        3Q       Max
## -2.64187  -0.35085   0.09337   0.46325   3.03373
##
## Coefficients:
##                                      Estimate Std. Error z value Pr(>|z|)
## (Intercept)                          1.586900   1.200317   1.322 0.186146
## datos$sexomujer                      1.741367   0.291945   5.965 2.45e-09
## datos$doloratipico                   1.047683   0.301023   3.480 0.000501
## datos$dolorotro                      1.940095   0.270072   7.184 6.79e-13
## datos$dolortipico                    2.087828   0.361097   5.782 7.39e-09
## datos$tension                       -0.020645   0.005863  -3.521 0.000430
## datos$colesterol                    -0.006044   0.002116  -2.856 0.004294
## datos$frecmax                        0.025213   0.005738   4.394 1.11e-05
## datos$ejerciciosi                   -0.792451   0.242190  -3.272 0.001068
## datos$depST                         -0.627303   0.117716  -5.329 9.88e-08
## datos$vasos.coloreados1             -2.010009   0.260309  -7.722 1.15e-14
## datos$vasos.coloreados2             -2.775652   0.373075  -7.440 1.01e-13
## datos$vasos.coloreados3             -1.960216   0.498543  -3.932 8.43e-05
## datos$vasos.coloreados4              0.841112   0.853752   0.985 0.324528
## datos$exploracion.taliodaño_reversible -1.665954   0.423497  -3.934 8.36e-05
## datos$exploracion.taliodesconocido   -2.385804   1.253438  -1.903 0.056987
## datos$exploracion.taliosin_daño      -0.202342   0.428766  -0.472 0.636986
##
## (Intercept)
## datos$sexomujer                     ***
## datos$doloratipico                  ***
## datos$dolorotro                     ***
## datos$dolortipico                   ***
## datos$tension                       ***
## datos$colesterol                    **
## datos$frecmax                       ***
## datos$ejerciciosi                   **
## datos$depST                         ***
## datos$vasos.coloreados1             ***
## datos$vasos.coloreados2             ***
## datos$vasos.coloreados3             ***
## datos$vasos.coloreados4
## datos$exploracion.taliodaño_reversible ***
## datos$exploracion.taliodesconocido   .
## datos$exploracion.taliosin_daño
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 1420.24  on 1024  degrees of freedom
## Residual deviance:  642.69  on 1008  degrees of freedom
## AIC: 676.69
##
## Number of Fisher Scoring iterations: 6
```

Calulamos los OR del modelo.

```
exp(modelo.diagnostico$coefficients)
```

```
##                            (Intercept)                          datos$sexomujer
##                              4.88856982                              5.70513628
##                      datos$doloratipico                          datos$dolorotro
##                              2.85103662                              6.95941109
##                       datos$dolortipico                           datos$tension
##                              8.06737147                              0.97956667
##                        datos$colesterol                           datos$frecmax
##                              0.99397446                              1.02553364
##                       datos$ejerciciosi                             datos$depST
##                              0.45273382                              0.53403009
##                 datos$vasos.coloreados1                 datos$vasos.coloreados2
##                              0.13398748                              0.06230884
##                 datos$vasos.coloreados3                 datos$vasos.coloreados4
##                              0.14082800                              2.31894403
## datos$exploracion.taliodaño_reversible     datos$exploracion.taliodesconocido
##                              0.18901024                              0.09201495
##         datos$exploracion.taliosin_daño
##                              0.81681538
```

Si el valor es mayor que 1, entonces indica que a medida que aumenta el predictor, las probabilidades de los resultados aumentan. A la inversa, un valor menor que 1 indica que a medida que aumenta el predictor, las probabilidades de los resultados disminuyen.

En base a los coeficientes obtenidos, podemos decir que el ser mujer influye de manera positiva en el modelo, aumentando la probabilidad de que el pacientes esté sano, respecto a que esté enfermo.

En este caso, podemos decir que las probabilidades de que un paciente esté sano si es mujer son 5.7 veces superiores a las de un paciente varón.

De esta forma podemos decir que las variables que más contribuyen de forma positiva a que el paciente esté sano son:
El ser mujer (5.7).
El dolor de tipo atípico (8.06).
El dolor de otro tipo (6.96).

Y las que contribuyen a que el paciente esté enfermo son:
1 vaso principal coloreado (0.14).
2 vasos principales coloreados (0.06).
3 vasos principales coloreados (0.14).
Exploración talio con daño reversible (0.18).

Evaluamos al igual que antes la bondad del modelo reducido.

```
hl <- hoslem.test(modelo.diagnostico$y, fitted(modelo.diagnostico))
hl
```

```
##
##  Hosmer and Lemeshow goodness of fit (GOF) test
##
## data:  modelo.diagnostico$y, fitted(modelo.diagnostico)
## X-squared = 5.2479, df = 8, p-value = 0.7308
```

En este caso, obtenemos un valor p > 0.05, que no es significativo, con lo que no podemos rechazar la hipótesis nula y por lo tanto podemos afirmar que los valores esperados se corresponden a los valores predichos por el modelo de regresión logística.

Comparamos los valores esperados con los observados.

```
expected <- round(hl$expected, 0)
observed <- round(hl$observed, 0)
cbind(expected, observed)
```

```
##                 yhat0 yhat1  y0  y1
## [0.00084,0.0087]   104     0 104   0
## (0.0087,0.0355]    101     2 100   3
## (0.0355,0.105]      94     7  92   9
## (0.105,0.304]       82    22  85  19
## (0.304,0.603]       56    48  58  46
## (0.603,0.742]       32    69  32  69
## (0.742,0.886]       17    84  13  88
## (0.886,0.943]        9    95  11  93
## (0.943,0.977]        4    97   4  97
## (0.977,0.998]        1   101   0 102
```

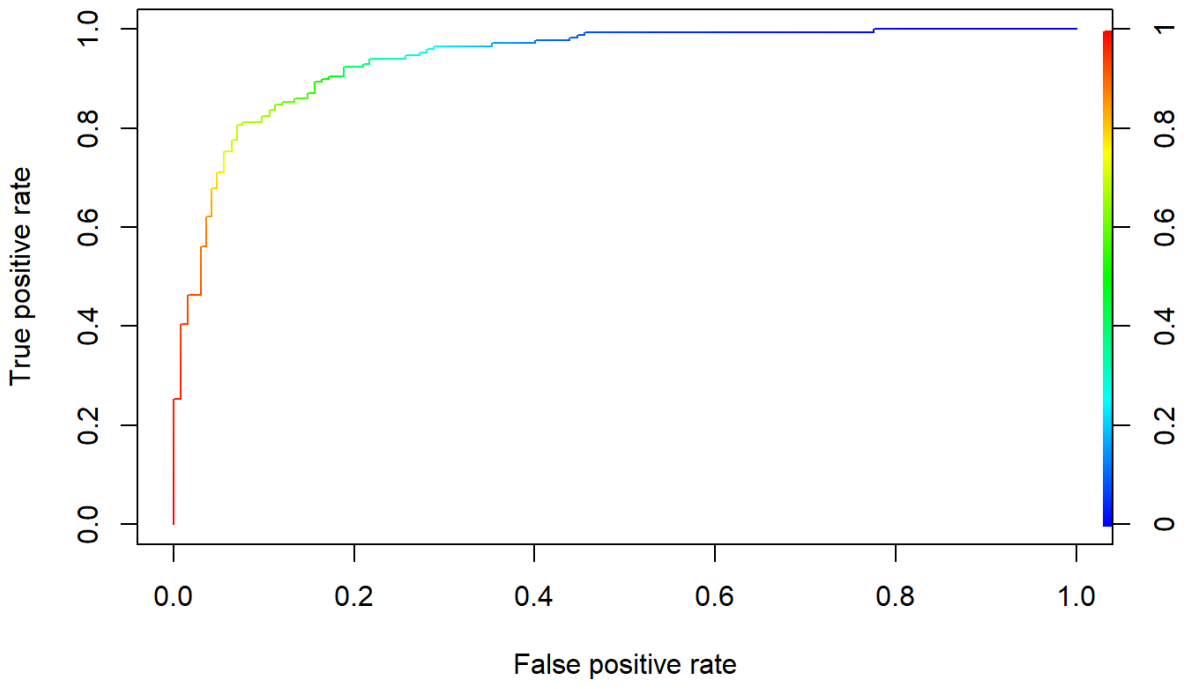
```
difer<-abs(expected - observed)
colSums(difer)
```

```
## yhat0 yhat1
##    15    15
```

Comparando con el modelo anterior, se clasifican erróneamente 15 casos en lugar de 45.

Curva ROC

```
predicciones <- predict(modelo.diagnostico, type = "response")
pred <- prediction(predicciones, datos$diagnostico)
perf <- performance(pred,"tpr","fpr")
plot(perf,colorize=TRUE)
```



El área bajo la curva es

```
as.numeric(performance(pred,"auc")@y.values)
```

```
## [1] 0.9395521
```

Al eliminar las variables del modelo hemos perdido algo de precisión, pero prácticamente es la misma que con todas las variables.

5. Representación de los resultados a partir de tablas y gráficas.

A lo largo del desarrollo de la práctica se han ido mostrando las tablas y representaciones gráficas que ilustran cada caso.

6. Resolución del problema.

A partir de los resultados obtenidos, ¿cuáles son las conclusiones? ¿Los resultados permiten responder al problema?

La conclusión del estudio es que, como era de esperar, no todas las variables influyen de forma significativa en el diagnóstico del paciente como enfermo coronario o sano.

Hemos podido determinar que las variables que más influencia tienen sobre el resultado final son:

El sexo del paciente, las mujeres tienen mejor pronóstico que los hombres.
El tipo de dolor que presenta el paciente, si no es el típico dolor de angina de pecho o es de otro tipo, las opciones de padecer enfermedad coronaria disminuyen considerablemente.

Si el número de vasos principales coloreados en la fluoroscopia está entre 1 y 3, el paciente tiene muchas posibilidades de estar enfermo.
Si la exploración con talio muestra daño reversible también empeora considerablemente el pronóstico.

Nuestro objetivo era identificar los parámetros que nos permitan distinguir entre un paciente sano y enfermo, por lo que sí podríamos decir que hemos obtenido una respuesta al problema con el modelo de regresión.