PRACTICA 2: LIMPIEZA Y VALIDACION DE LOS DATOS

Bianca Palacios Pinargotti Araceli Najar García

6 de enero 2023

Contents

1 Resolución de la práctica

1.1 Descripción del dataset

Los datos a utilizar en esta práctica se han obtenido de la página de Kaggle y trata de un conjunto de datos de análisis y predicción de ataques cardíacos. Se puede acceder mediante el siguiente enlace.

Conocer y comprender los factores que aumentan el riesgo de un ataque al corazón es poder. Es necesario centrarse en la prevención de las cardiopatías en las primeras etapas para evalúar sus factores de riesgo y trabajar para mantenerlos bajos. Cuanto antes se los identifique y se empiece a controlarlos, más posibilidades se tendrá de llevar una vida larga y sana.

Importantes estudios indican que existen tres categorías de factores de riesgo:

- **Edad avanzada** dónde las mujeres tienen mayor riesgo a morir cuanto más avanzada de edad están.
- **Sexo masculino** dónde los hombres son los que presentan mayor riesgo de sufrir un ataque al corazón en edades más tempranas.
- **Herencia (incluida la raza)** dónde es más probable que los niños de padres con cardiopatías desarrollen cardiopatías. La mayoría de las personas con antecedentes familiares significativos de cardiopatía tienen uno o más factores de riesgo. Al igual que no puede controlar su edad, sexo y raza, no puede controlar sus antecedentes familiares.

El siguiente conjunto de datos contiene información sobre los pacientes y el estado de su corazón junto a antecedentes de ataques cardíacos. Se limpiará, normalizará de ser necesario y finalmente se analizará los datos para clasificarlos.

```
# lectura dataset
base_heart <- read.csv("heart.csv",header = TRUE)
```

```
# Tipo de dato asignado a cada campo
sapply(base_heart, function(x) class(x))
```

```
##       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"
```

```
# variables del dataset
variables_base <- names(base_heart)
```

El conjunto de datos contiene 14 variables(columnas) y 303 filas o registros.

Las variables que componen el dataset son age, sex, cp, trtbps, chol, fbs, restecg, thalachh, exng, oldpeak, slp, caa, thall, output e indican:

- Variable `age`: Edad del paciente. Valor numérico entero.
- Variable `sex`: Sexo del paciente. Valores 0 y 1. Valor numérico entero (1 = male; 0 = female)
- Variable `cp`: Valor numérico entero que muestra el tipo de dolor en el pecho. Valores que puede tomar:
  - 1: angina típica
  - 2: angina atípica
  - 3: dolor no anginoso
  - 4: asintomático
- Variable `trtbps`: Presión arterial en reposo (en mm Hg). Valor numérico entero.
- Variable `chol`: Colesterol en mg/dl obtenido a través del sensor de IMC. Valor numérico entero.
- Variable `fbs`: Muestra si el valor de la azúcar en la sangre en ayunas es mayor a 120 mg/dl. Si se cumple es 1 (verdadero), de lo contrario es 0 (falso). En el dataset es un valor numérico entero.
- Variable `restecg`: Son los resultados electrocardiográficos en reposo. Valor numérico entero que toma valores:
  - 0: normal.
  - 1: tener anomalía de la onda ST-T (inversiones de la onda T y/o elevación o depresión del ST de $>$ 0,05 mV).
  - 2: mostrar hipertrofia ventricular izquierda probable o definida según el criterio de Estes.
- Variable `thalachh`: Es la frecuencia cardíaca máxima alcanzada. Valor numérico entero.
- Variable `exng`: Es la angina inducida por el ejercicio. Valor numérico entero que indica 1 cuando es 'Si' y 0 cuando es 'No'.
- Variable `oldpeak`: Es el valor antiguo del segmento ST inducida por el ejercicio en relación con el reposo. El segmento ST es la parte isoeléctrica plana del ECG y representa el intervalo entre la despolarización y la repolarización ventricular. En el dataset es un valor numérico.
- Variable `slp`: Es la pendiente del segmento ST cuando hay ejercicio al máximo. Toma valores:
  - 0: Sin pendiente.
  - 1: Plano.
  - 2: Pendiente descendente.
- Variable `caa`: Número de vasos sanguíneos mayores coloreados por fluoroscopia. Valor numérico entero de 0 a 3.
- Variable `thall`: Tasa de thal, es decir, frecuencia cardíaca. Variable numérica que puede ser:
  - 0: Nulo.
  - 1: Defecto fijo.
  - 2: Normal.
  - 3: Defecto reversible.
- Variable `output`: Es el atributo predicho que muestra el diagnóstico de enfermedad cardíaca. Valor numérico entero que puede tomar valores:
  - 0: $<$ 50% de estrechamiento del diámetro, es decir, menos posibilidades de enfermedades del corazón.
  - 1: $>$ 50% de estrechamiento del diámetro, es decir, más posibilidades de enfermedades del corazón.

1.2 Importancia y objetivos de los análisis: ¿Por qué es importante y qué pregunta/problema se pretende responder?

El objetivo del análisis de este conjunto de datos en particular es determinar si existe relación entre el riesgo de sufrir un ataque cardíaco en función de otras variables como la edad del paciente, el valor de colesterol, resultados electrocardiográficos en reposo (`restecg`) y otros, a fin de reducir los casos falsos negativos de pacientes y conocer quienes si son propensos a sufrir una ataque. En base a estos resultados, se puede crear modelos de clasificación entre los pacientes con menor o mayor riesgo, o de regresión para predecir el peligro de un ataque cardíaco. Por tanto, las preguntas que se podrían responder después del análisis del conjunto de datos serían:

- ¿Qué variables tienen mayor influencia en el riesgo de sufrir un ataque cardíaco?
- ¿La edad es un factor que aumenta la probabilidad de sufrir un ataque cardíaco?

Si bien esta base de datos es pequeña, su relevancia radica en su utilidad para practicar y entrenar un modelo de predicción o clasificación. Así, estos análisis se pueden aplicar en el sector de las ciencias médicas como soporte en otros estudios de ataques cardíacos u otras patologías cardíacas.

1.3 Limpieza de los datos

1.3.1 Selección de variables

Luego de conocer las variables que componen el conjunto de datos se procede a seleccionar los atributos de interés para el estudio.

Los campos *age*, *sex*, *cp*, *trtbps*, *chol*, *fbs*, *restecg*, *thalachh*, *exng*, *oldpeak*, *slp*, *caa* y *thall* aportan información de cada paciente, por tanto, son importantes para estudiar los riesgos de sufrir un ataque cardíaco.

La variable *output* muestra la clasificación de los pacientes entre los que tienen menor o mayor riesgo. Será de utilidad para comparar los resultados de los modelos que se creen y determinar si la clasificación es correcta.

Por tanto, se usarán todos los atributos del dataset.

1.3.2 Tipos de variables

La lectura del fichero con la función `read.csv()` ha realizado su asignación a cada variable, donde se tienen enteros en campos que van a ser transformados a categóricos.

```
# tipos de variables iniciales
res <- sapply(base_heart,class)
kable(data.frame(variables=names(res),clase=as.vector(res)))
```

variables	clase
age	integer
sex	integer
cp	integer
trtbps	integer
chol	integer
fbs	integer
restecg	integer
thalachh	integer
exng	integer
oldpeak	numeric
slp	integer
caa	integer
thall	integer
output	integer

Se transformarán las variables *sex*, *cp*, *fbs*, *restecg*, *exng*, *slp*, *caa*, *thall* y *output* de tipo numérico entero a categóricas.

```
base_heart$sex <- as.factor(base_heart$sex)
base_heart$cp <- as.factor(base_heart$cp)
base_heart$fbs <- as.factor(base_heart$fbs)
base_heart$restecg <- as.factor(base_heart$restecg)
base_heart$exng <- as.factor(base_heart$exng)
base_heart$slp <- as.factor(base_heart$slp)
base_heart$caa <- as.factor(base_heart$caa)
base_heart$thall <- as.factor(base_heart$thall)
base_heart$output <- as.factor(base_heart$output)
```

```
res <- sapply(base_heart,class)
kable(data.frame(variables=names(res),clase=as.vector(res)))
```

variables	clase
age	integer
sex	factor
cp	factor
trtbps	integer
chol	integer
fbs	factor
restecg	factor
thalachh	integer
exng	factor
oldpeak	numeric
slp	factor
caa	factor
thall	factor
output	factor

Se revisa el conjunto de datos resultante.

```
str(base_heart)
```

```
## 'data.frame':    303 obs. of  14 variables:
##  $ age     : int  63 37 41 56 57 57 56 44 52 57 ...
##  $ sex     : Factor w/ 2 levels "0","1": 2 2 1 2 1 2 1 2 2 2 ...
##  $ cp      : Factor w/ 4 levels "0","1","2","3": 4 3 2 2 1 1 2 2 3 3 ...
##  $ trtbps  : int  145 130 130 120 120 140 140 120 172 150 ...
##  $ chol    : int  233 250 204 236 354 192 294 263 199 168 ...
##  $ fbs     : Factor w/ 2 levels "0","1": 2 1 1 1 1 1 1 1 2 1 ...
##  $ restecg : Factor w/ 3 levels "0","1","2": 1 2 1 2 2 2 1 2 2 2 ...
##  $ thalachh: int  150 187 172 178 163 148 153 173 162 174 ...
##  $ exng    : Factor w/ 2 levels "0","1": 1 1 1 1 2 1 1 1 1 1 ...
##  $ oldpeak : num  2.3 3.5 1.4 0.8 0.6 0.4 1.3 0 0.5 1.6 ...
##  $ slp     : Factor w/ 3 levels "0","1","2": 1 1 3 3 3 2 2 3 3 3 ...
##  $ caa     : Factor w/ 5 levels "0","1","2","3",..: 1 1 1 1 1 1 1 1 1 1 ...
##  $ thall   : Factor w/ 4 levels "0","1","2","3": 2 3 3 3 3 2 3 4 4 3 ...
##  $ output  : Factor w/ 2 levels "0","1": 2 2 2 2 2 2 2 2 2 2 ...
```

1.3.3 Eliminación de valores nulos, outliers

Se debe revisar la cantidad de valores nulos que existen por cada atributo de la base de datos.

```
# Números de valores desconocidos por campo
sapply(base_heart, function(x) sum(is.na(x)))
```

```
##      age      sex       cp   trtbps     chol      fbs  restecg thalachh
##        0        0        0        0        0        0        0        0
##     exng  oldpeak      slp      caa    thall   output
##        0        0        0        0        0        0
```

Se observa que no existen registros que presentan valores desconocidos o vacíos, por tanto, no será necesario aplicar un método de imputación de valores.

Además, se revisará si existen registros duplicados.

```
sum(duplicated(base_heart))
```

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

Dado que existe un registro duplicado, se procede a eliminarlo.

```
index <- which(duplicated(base_heart)==TRUE)
base_heart <- base_heart[-index,]

# revision de duplicados en la base
sum(duplicated(base_heart))
```

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

```
# dimension nueva base de datos
dim(base_heart)
```

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

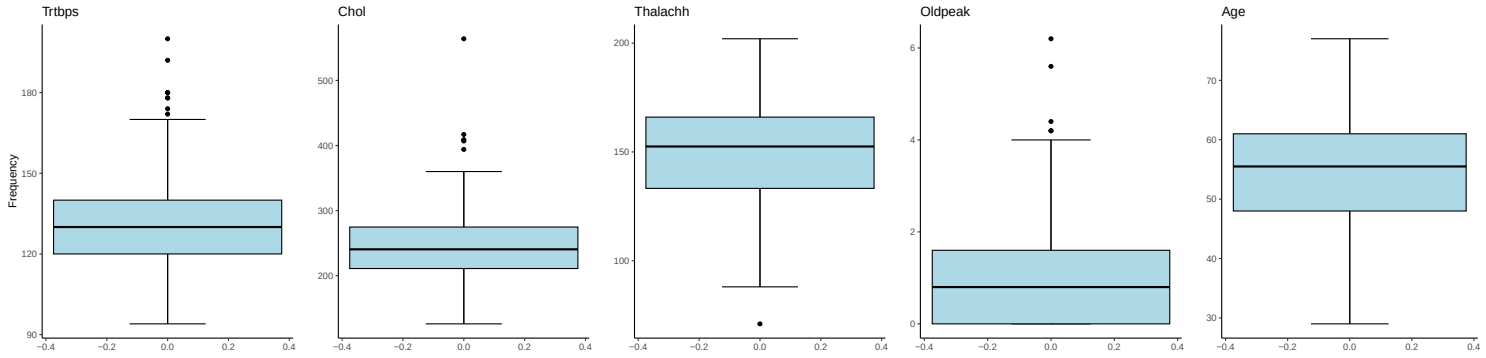
1.3.3.1 Valores extremos(outliers) Los valores extremos o outliers son aquellos que parecen no ser congruentes sin los comparamos con el resto de los datos.

Para identificarlos, se puede representar un diagrama de caja por cada variable numérica de interés, en este caso: *trtbps*, *chol*, *thalachh*, *oldpeak* y *age*.

En este gráfico se puede observar qué valores distan mucho del rango intercuartílico (la caja). Con la función boxplots.stats() se muestran los valores atípicos a analizar.

```
variables_base <- names(base_heart)

# gráfico de caja y bigotes
c1 <- ggplot(base_heart, aes(y = trtbps)) +
  stat_boxplot(geom = "errorbar", width = 0.25) +
  geom_boxplot(color = "black", fill = "lightblue") +
  labs(title = "Trtbps", x='', y='Frequency') + theme_classic()

c2 <- ggplot(base_heart, aes(y = chol)) +
  stat_boxplot(geom = "errorbar", width = 0.25) +
  geom_boxplot(color = "black", fill = "lightblue") +
  labs(title = "Chol", x='', y='') + theme_classic()

c3 <- ggplot(base_heart, aes(y = thalachh)) +
  stat_boxplot(geom = "errorbar", width = 0.25) +
  geom_boxplot(color = "black", fill = "lightblue") +
  labs(title = "Thalachh", x='', y='') + theme_classic()

c4 <- ggplot(base_heart, aes(y = oldpeak)) +
  stat_boxplot(geom = "errorbar", width = 0.25) +
  geom_boxplot(color = "black", fill = "lightblue") +
```

```
  labs(title = "Oldpeak", x='', y='') + theme_classic()

c5 <- ggplot(base_heart, aes(y = age)) +
  stat_boxplot(geom = "errorbar", width = 0.25) +
  geom_boxplot(color = "black", fill = "lightblue") +
  labs(title = "Age", x='', y='') + theme_classic()

plot_grid(c1,c2,c3,c4,c5, nrow=1, ncol=5, labels='')
```




```
# valores atípicos de trtbps
boxplot.stats(base_heart$trtbps)$out
```

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

```
# valores atípicos de chol
boxplot.stats(base_heart$chol)$out
```

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

```
# valores atípicos de thalachh
boxplot.stats(base_heart$thalachh)$out
```

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

```
# valores atípicos de oldpeak
boxplot.stats(base_heart$oldpeak)$out
```

```
## [1] 4.2 6.2 5.6 4.2 4.4
```

Al revisar los valores extremos para estas variables, se observa que son valores que se pueden dar. Es decir, para la variable de colesterol *chol*, estos casos con valores altos son los que se van a estudiar para determinar a los pacientes con más probabilidad de sufrir un ataque cardíaco. Por tanto, se procede a dejar los valores extremos para el estudio.

1.3.4 Conversión de los datos

En la etapa de conversión, los datos son transformados con el objetivo de que el análisis posterior sea más eficiente y/o los resultados obtenidos sean más fácilmente interpretables.

Se normalizará las variables numéricas del dataset que son: *trtbps*, *chol*, *thalachh*, *oldpeak*. Se aplica la función *scale()* que aplica el tipo de normalización z-score.

```
base_heart[,c(4,5,8,10)] <- scale(base_heart[,c(4,5,8,10)])
```

```
head(base_heart)
```

```
##   age sex cp      trtbps        chol fbs restecg    thalachh exng    oldpeak
## 1  63   1  3  0.76279965 -0.26085198   1       0  0.01879464    0  1.0822258
## 2  37   1  2 -0.09124939  0.06762829   0       1  1.63426634    0  2.1154150
## 3  41   0  1 -0.09124939 -0.82120068   0       0  0.97934538    0  0.3073339
## 4  56   1  1 -0.66061542 -0.20288487   0       1  1.24131376    0 -0.2092607
## 5  57   0  0 -0.66061542  2.07715466   0       1  0.58639280    1 -0.3814589
## 6  57   1  0  0.47811664 -1.05306911   0       1 -0.06852815    0 -0.5536571
##   slp caa thall output
## 1   0   0     1      1
## 2   0   0     2      1
## 3   2   0     2      1
## 4   2   0     2      1
## 5   2   0     2      1
## 6   1   0     1      1
```

1.3.5 Exportación de los datos preprocesados

Realizado los procedimientos de validación y limpieza en el conjunto de datos inicial, se procede a guardarlos en un nuevo fichero denominado `heart_data_clean.csv`:

```
write.csv(base_heart, "heart_data_clean.csv")
```

1.4 Análisis de los datos

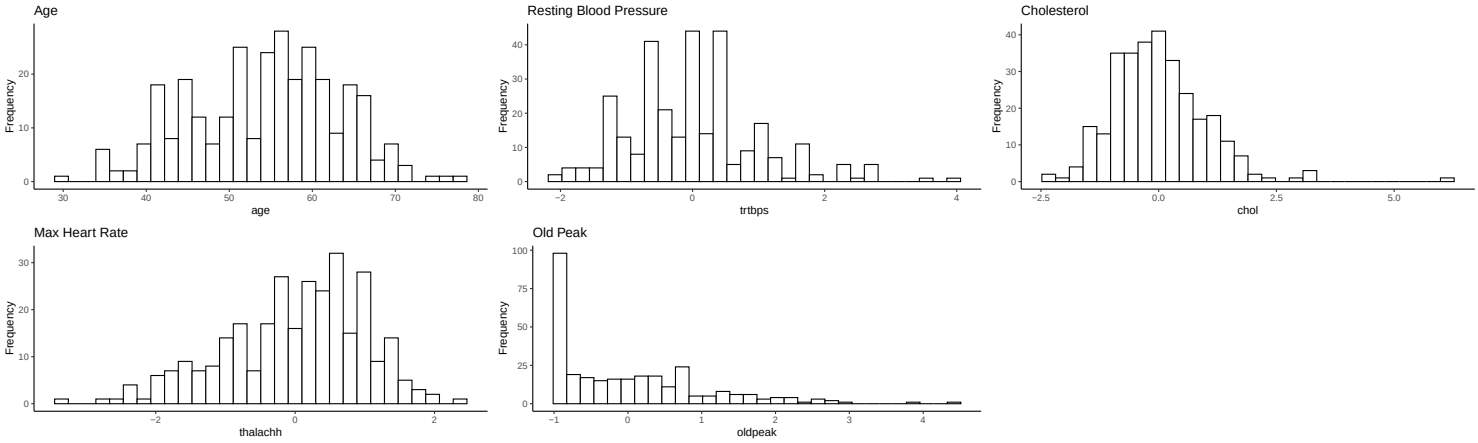
La exploración de los datos busca explicar las principales características de los mismos, con el objetivo de tratar de responder a las preguntas que suscita el trabajo elaborado y los propios datos utilizados.

```
num_heart <- base_heart[, c(1,4,5,8,10)]
categ_heart <- base_heart[, -c(1,4,5,8,10)]
```

Como parte inicial del análisis de los datos de este trabajo, vamos a evaluar de manera gráfica la distribución de cada una de las variables. Contamos con un total de 5 variables numéricas y 9 variables categóricas.

Por un lado, para analizar la distribución de las **variables numéricas**, se realiza una representación gráfica en forma de **histograma**:

```
# Histogramas variables numericas
#############

h1 <-ggplot(num_heart, aes(x = age)) + geom_histogram(color = "black", fill = "white") +
  labs(title = "Age", x = 'age', y = 'Frequency') + theme_classic()

h2 <-ggplot(num_heart, aes(x = trtbps)) + geom_histogram(color = "black", fill = "white") +
  labs(title = "Resting Blood Pressure", x = 'trtbps', y = 'Frequency') + theme_classic()

h3 <-ggplot(num_heart, aes(x = chol)) + geom_histogram(color = "black", fill = "white") +
  labs(title = "Cholesterol", x = 'chol', y = 'Frequency') + theme_classic()

h4 <-ggplot(num_heart, aes(x = thalachh)) + geom_histogram(color = "black", fill = "white") +
  labs(title = "Max Heart Rate", x = 'thalachh', y = 'Frequency') + theme_classic()

h5 <-ggplot(num_heart, aes(x = oldpeak)) + geom_histogram(color = "black", fill = "white") +
  labs(title = "Old Peak", x = 'oldpeak', y = 'Frequency') + theme_classic()

plot_grid(h1, h2, h3, h4, h5, nrow = 2, ncol = 3, labels = '')
```

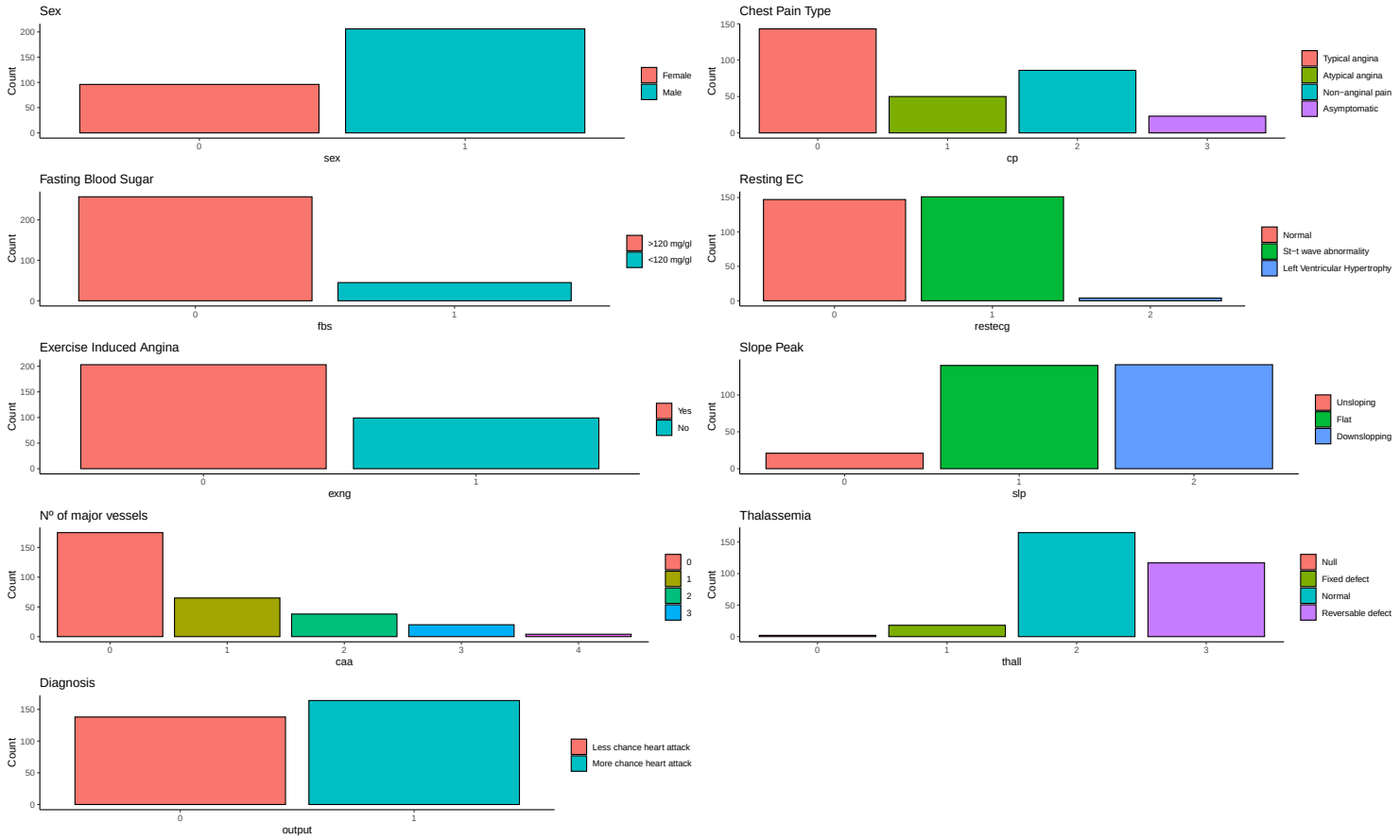


A simple vista parece que ninguna de las variables sigue una distribución normal (lo que evaluaremos con mayor precisión en apartados posteriores).

Por otro lado, podemos observar la distribución o frecuencia de registros en los diferentes grupos de las **variables categóricas** a través de **gráficos de barras**:

```
# Gráficos de barras variables categóricas
b1 <- ggplot(categ_heart, aes(x = sex, fill = sex)) + geom_bar(color = "black") +
  labs(title = "Sex", x = 'sex', y = 'Count') + theme_classic() +
  theme(legend.title = element_blank()) +
  scale_fill_discrete(breaks=c("0", "1"), labels=c("Female", "Male"))

b2 <- ggplot(categ_heart, aes(x = cp, fill = cp)) + geom_bar(color = "black") +
  labs(title = "Chest Pain Type", x = 'cp', y = 'Count') + theme_classic() +
  theme(legend.title = element_blank()) +
  scale_fill_discrete(breaks=c("0", "1", "2", "3"), labels=c("Typical angina", "Atypical angina", "Non-anginal p

b3 <- ggplot(categ_heart, aes(x = fbs, fill = fbs)) + geom_bar(color = "black") +
  labs(title = "Fasting Blood Sugar", x = 'fbs', y = 'Count') + theme_classic() +
  theme(legend.title = element_blank()) +
  scale_fill_discrete(breaks=c("0", "1"),labels=c(">120 mg/gl", "<120 mg/gl"))

b4 <- ggplot(categ_heart, aes(x = restecg, fill = restecg)) + geom_bar(color = "black") +
  labs(title = "Resting EC", x = 'restecg', y = 'Count') + theme_classic() +
  theme(legend.title = element_blank()) +
  scale_fill_discrete(breaks=c("0", "1", "2"), labels=c("Normal", "St-t wave abnormality", "Left Ventricular Hyp

b5 <- ggplot(categ_heart, aes(x = exng, fill = exng)) + geom_bar(color = "black") +
  labs(title = "Exercise Induced Angina", x = 'exng', y = 'Count') + theme_classic() +
  theme(legend.title = element_blank()) +
  scale_fill_discrete(breaks=c("0", "1"), labels=c("Yes", "No"))

b6 <- ggplot(categ_heart, aes(x = slp, fill = slp)) + geom_bar(color = "black") +
  labs(title = "Slope Peak", x = 'slp', y = 'Count') + theme_classic() +
  theme(legend.title = element_blank()) +
  scale_fill_discrete(breaks=c("0", "1", "2"), labels=c("Unsloping", "Flat", "Downslopping"))

b7 <- ggplot(categ_heart, aes(x = caa, fill = caa)) + geom_bar(color = "black") +
  labs(title = "Nº of major vessels", x = 'caa', y = 'Count') + theme_classic() +
  theme(legend.title = element_blank()) +
  scale_fill_discrete(breaks=c("0", "1", "2", "3"), labels=c("0", "1", "2", "3"))

b8 <- ggplot(categ_heart, aes(x = thall, fill = thall)) + geom_bar(color = "black") +
  labs(title = "Thalassemia", x = 'thall', y = 'Count') + theme_classic() +
  theme(legend.title = element_blank()) +
  scale_fill_discrete(breaks=c("0", "1", "2", "3"), labels=c("Null", "Fixed defect", "Normal", "Reversable defec
```

```
b9 <- ggplot(categ_heart, aes(x = output, fill = output)) + geom_bar(color = "black") +
  labs(title = "Diagnosis", x = 'output', y = 'Count') + theme_classic() +
  theme(legend.title = element_blank()) +
  scale_fill_discrete(breaks=c("0", "1"),
                      labels=c("Less chance heart attack", "More chance heart attack"))

plot_grid(b1, b2, b3, b4, b5, b6, b7, b8, b9, nrow = 5, ncol = 2, labels = '')
```



Con estos resultados es posible extraer algunas conclusiones: por ejemplo, el número de mujeres es superior al de hombres, el tipo de dolor del pecho más frecuente es la angina típica, la mayoría de pacientes presentan un valor de azúcar en sangre en ayunas mayor a 120 mg/fl, la mayoría de pacientes cuentan con un total de 0 vasos sanguíneos mayores coloreados con fluoroscopia, etc.

1.4.1 Selección de los grupos de datos que se quieren analizar/comparar.

Tras este primer análisis visual de los datos, podemos añadir una capa extra de información observando las distribuciones de estas variables en función de los grupos de datos que se van a escoger para analizar y comparar. En este caso, el objetivo principal del trabajo es evaluar qué factores (datos clínicos de los pacientes, es decir, las variables del estudio) pueden ser relevantes y de interés a la hora de determinar la mayor o menos posibilidad de que un paciente sufra un ataque cardíaco (variable objetivo o target `output`).

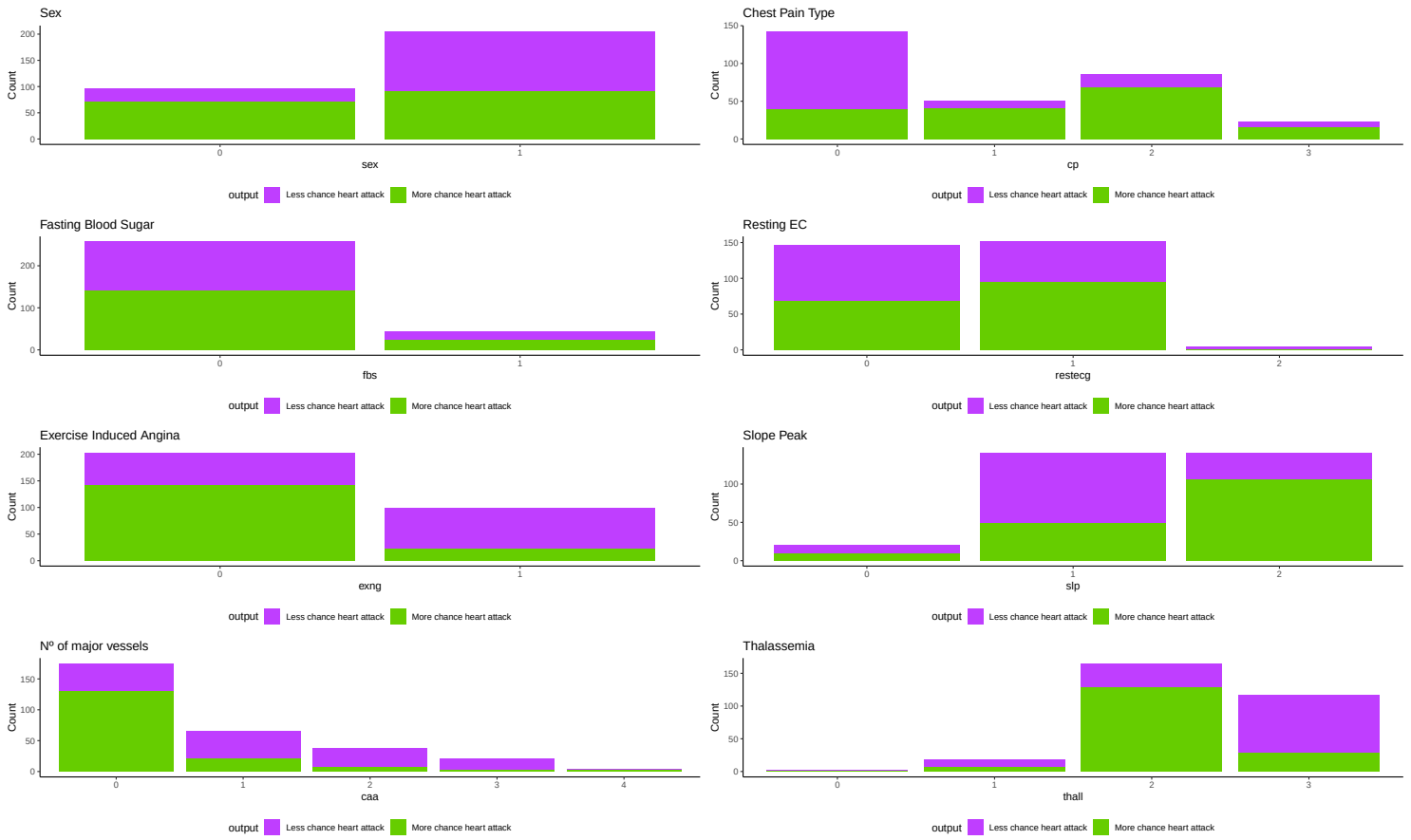
Por ello, los grupos de datos a comparar van a ser dos: pacientes con menor riesgo de ataque al corazón y pacientes con mayor riesgo de ataque al corazón. En base a esta división de los datos, a continuación vamos a añadir una capa extra de información a los gráficos realizados con anterioridad, en este caso representando la distribución de las variables del estudio en función de los 2 grupos de datos.

```
plt <- ggpairs(num_heart,columns=1:5,ggplot2::aes(alpha=0.75,color=base_heart$output),legend=2,upper = list(cont
               lower = list(continuous = wrap("points",alpha = 0.75,size=2.8)))+ theme(text=element_text(size=22
plt
```

En cuanto a las variables numéricas, por ejemplo es posible observar que los pacientes del estudio que presentan menos probabilidades de ataque cardíaco presentan edades más elevadas. Además, presentan mayor posibilidades de padecer esta patología aquellos que han alcanzado una frecuencia cardíaca máxima de mayor magnitud.

```
p6 <- ggplot(categ_heart, aes(x = sex, fill = output)) + geom_bar() +
  labs(title = "Sex", y = 'Count') + theme_classic() + theme(legend.position = "bottom") +
  scale_fill_manual(labels = c("Less chance heart attack", "More chance heart attack"),
                    values = c("darkorchid1", "chartreuse3"))

p7 <- ggplot(categ_heart, aes(x = cp, fill = output)) + geom_bar() +
  labs(title = "Chest Pain Type", y = 'Count') + theme_classic() + theme(legend.position = "bottom") + scale_fil

p8 <- ggplot(categ_heart, aes(x = fbs, fill = output)) + geom_bar() +
  labs(title = "Fasting Blood Sugar", y = 'Count') + theme_classic() +
  theme(legend.position = "bottom") +
  scale_fill_manual(labels = c("Less chance heart attack", "More chance heart attack"),
                    values = c("darkorchid1", "chartreuse3"))

p9 <- ggplot(categ_heart, aes(x = restecg, fill = output)) + geom_bar() +
  labs(title = "Resting EC", y = 'Count') + theme_classic() +
  theme(legend.position = "bottom") +
  scale_fill_manual(labels = c("Less chance heart attack", "More chance heart attack"),
                    values = c("darkorchid1", "chartreuse3"))

p10 <- ggplot(categ_heart, aes(x = exng, fill = output)) + geom_bar() +
  labs(title = "Exercise Induced Angina", y = 'Count') + theme_classic() +
  theme(legend.position = "bottom") +
  scale_fill_manual(labels = c("Less chance heart attack", "More chance heart attack"),
                    values = c("darkorchid1", "chartreuse3"))

p11 <- ggplot(categ_heart, aes(x = slp, fill = output)) + geom_bar() +
  labs(title = "Slope Peak", y = 'Count') + theme_classic() +
  theme(legend.position = "bottom") +
  scale_fill_manual(labels = c("Less chance heart attack", "More chance heart attack"),
                    values = c("darkorchid1", "chartreuse3"))

p12 <- ggplot(categ_heart, aes(x = caa, fill = output)) + geom_bar() +
  labs(title = "Nº of major vessels", y = 'Count') + theme_classic() +
  theme(legend.position = "bottom") +
  scale_fill_manual(labels = c("Less chance heart attack", "More chance heart attack"),
                    values = c("darkorchid1", "chartreuse3"))

p13 <- ggplot(categ_heart, aes(x = thall, fill = output)) + geom_bar() +
```

```
  labs(title = "Thalassemia", y = 'Count') + theme_classic() +
  theme(legend.position = "bottom") +
  scale_fill_manual(labels = c("Less chance heart attack", "More chance heart attack"),
                    values = c("darkorchid1", "chartreuse3"))
options(repr.plot.width = 10, repr.plot.height =20)
plot_grid(p6, p7, p8, p9, p10, p11, p12, p13, nrow = 4, ncol = 2, labels = '')
```



Respecto a las variables categóricas, cabe mencionar, por ejemplo, que la proporción de hombres con mayor probabilidad de ataque cardiaco supera a la proporción de hombres con menor probabilidad de ataque cardíaco (hecho que sucede de manera inversa en el caso de las mujeres). El siguiente paso del análisis de datos va a consistir en un **análisis estadístico descriptivo** de las variables del estudio. Este análisis incluye los principales estadísticos descriptivos, una serie de valores que describen las principales características intrínsecas de un conjunto de datos. Estos estadísticos se calculan a partir de dicha muestra de datos y, principalmente, se pueden dividir en dos tipos:

- Medidas de **tendencia central**, como la media, la mediana, la moda o el rango medio.
- Medidas de **dispersión**, como el rango, los cuartiles, la varianza o la desviación estándar.

Los resultados del análisis estadístico descriptivo básico se muestran a continuación:

```
summary(base_heart)
```

```
##       age        sex     cp          trtbps             chol         fbs
##  Min.   :29.00   0: 96   0:143   Min.   :-2.14097   Min.   :-2.3283   0:257
##  1st Qu.:48.00   1:206   1: 50   1st Qu.:-0.66061   1st Qu.:-0.6859   1: 45
##  Median :55.50           2: 86   Median :-0.09125   Median :-0.1159
##  Mean   :54.42           3: 23   Mean   : 0.00000   Mean   : 0.0000
##  3rd Qu.:61.00                   3rd Qu.: 0.47812   3rd Qu.: 0.5459
##  Max.   :77.00                   Max.   : 3.89431   Max.   : 6.1349
##  restecg    thalachh       exng       oldpeak        slp     caa     thall
```

```
##  0:147   Min.   :-3.4305   0:203   Min.   :-0.8981   0: 21   0:175   0:  2
##  1:151   1st Qu.:-0.7125   1: 99   1st Qu.:-0.8981   1:140   1: 65   1: 18
##  2:  4   Median : 0.1279           Median :-0.2093   2:141   2: 38   2:165
##          Mean   : 0.0000           Mean   : 0.0000           3: 20   3:117
##          3rd Qu.: 0.7174           3rd Qu.: 0.4795           4:  4
##          Max.   : 2.2892           Max.   : 4.4401
##  output
##  0:138
##  1:164
##
##
##
##
```

1.4.2 Comprobación de la normalidad y homogeneidad de la varianza.

Como se ha mencionado anteriormente, uno de los objetivos del trabajo es comparar grupos de datos. Por tanto, también se va a realizar un análisis estadístico inferencial, concretamente se aplicarán pruebas estadísticas de contraste de hipótesis, entre otras pruebas.

Antes de llevar a cabo los contrastes de hipótesis es necesario evaluar si los datos cumplen los principios de **normalidad** y **homocedasticidad** o igualdad de varianzas. En función de que los cumplan o no, para los contrastes de hipótesis emplearemos pruebas paramétricas o no paramétricas, respectivamente.

Para evaluar si las variables numéricas de este proyecto siguen una distribución **normal**, empleamos el test de **Shapiro-Wilk**, uno de los métodos más potentes y robustos para contrastar la normalidad. En este test la hipótesis nula plantea que la población está distribuida normalmente y se fija como valor de significancia un $alpha = 0.05$. Por tanto, si el p-valor obtenido es inferior al nivel de significancia, se rechaza la hipótesis nula y se llega a la conclusión de que los datos no siguen una distribución normal. En cambio, si el p-valor es mayor al nivel de significancia, no es posible rechazar la hipótesis nula y, en definitiva, se asume que los datos cumplen el principio de normalidad.

```
apply(num_heart, 2, shapiro.test)
```

```
## $age
##
##  Shapiro-Wilk normality test
##
## data:  newX[, i]
## W = 0.98664, p-value = 0.006745
##
##
## $trtbps
##
##  Shapiro-Wilk normality test
##
## data:  newX[, i]
## W = 0.96573, p-value = 1.419e-06
##
##
## $chol
##
##  Shapiro-Wilk normality test
##
## data:  newX[, i]
## W = 0.94658, p-value = 5.196e-09
##
##
## $thalachh
##
##  Shapiro-Wilk normality test
##
```

```
## data:  newX[, i]
## W = 0.97679, p-value = 8.268e-05
##
##
## $oldpeak
##
##  Shapiro-Wilk normality test
##
## data:  newX[, i]
## W = 0.84522, p-value < 2.2e-16
```

Como podemos ver, el test de Shapiro-Wilk nos indica que para las cinco variables numéricas el p-valor obtenido es inferior al nivel de significancia y que, por tanto, **ninguna de estas variables sigue una distribución normal**.

El siguiente paso consiste en comprobar la **homocedasticidad** de los datos, es decir, si entre los dos grupos a comparar existe igualdad de varianzas. Para ello empleamos el test de **Fligner-Killeen**, ya que se trata de una prueba no paramétrica y debemos utilizar un test de este tipo ya que hemos comprobado que nuestros datos no cumplen la condición de normalidad. En esta prueba, la hipótesis nula asume igualdad de varianzas entre los grupos de datos. Por tanto, un p-valor inferior al nivel de significancia supondrá un rechazo de la hipótesis nula y, en definitiva, indicará heterocedasticidad.

```
fligner.test(age ~ output, base_heart)
```

```
##
##  Fligner-Killeen test of homogeneity of variances
##
## data:  age by output
## Fligner-Killeen:med chi-squared = 7.0118, df = 1, p-value = 0.008097
```

```
fligner.test(trtbps ~ output, base_heart)
```

```
##
##  Fligner-Killeen test of homogeneity of variances
##
## data:  trtbps by output
## Fligner-Killeen:med chi-squared = 1.2235, df = 1, p-value = 0.2687
```

```
fligner.test(chol ~ output, base_heart)
```

```
##
##  Fligner-Killeen test of homogeneity of variances
##
## data:  chol by output
## Fligner-Killeen:med chi-squared = 0.70632, df = 1, p-value = 0.4007
```

```
fligner.test(thalachh ~ output, base_heart)
```

```
##
##  Fligner-Killeen test of homogeneity of variances
##
## data:  thalachh by output
## Fligner-Killeen:med chi-squared = 5.5359, df = 1, p-value = 0.01863
```

```
fligner.test(oldpeak ~ output, base_heart)
```

```
##
##  Fligner-Killeen test of homogeneity of variances
##
## data:  oldpeak by output
## Fligner-Killeen:med chi-squared = 37.685, df = 1, p-value = 8.315e-10
```

El resultado del test de Fligner-Killeen muestra que solo se cumple el principio de homocedasticidad en las variables `trtbps` y `chol`, mientras que el resto de variables **no presentan igualdad de varianzas**.

En conclusión, debido a que ninguna de las variables cumple el principio de normalidad y ya que la mayoría de ellas no cumple el principio de homocedasticidad, las pruebas que se emplearán en el próximo apartado para los contrastes de hipótesis serán de tipo no paramétrico.

1.5 Pruebas estadísticas

Aplicación de pruebas estadísticas para comparar los grupos de datos.

1.5.1 Contraste de hipótesis

En primer lugar, vamos a realizar pruebas estadísticas de **contraste de hipótesis** para comparar grupos de datos. Al no cumplirse los principios de normalidad e igualdad de varianzas, emplearemos pruebas no paramétricas. Además, al tratarse de muestras independientes o no pareadas, y al compararse 2 grupos de datos (menor probabilidad de ataque cardíaco versus mayor probabilidad), los test adecuados serían los siguientes:

- Test Chi-cuadrado para las variables cualitativas.
- Test de Mann-Whitney para las variables cuantitativas.

El resultado de emplear el test **Chi-cuadrado** para evaluar si existen diferencias en las variables categóricas entre los 2 grupos a comparar es el siguiente:

```
sex_output <- table(base_heart$sex, base_heart$output)
chisq.test(sex_output)
```

```
##
##  Pearson's Chi-squared test with Yates' continuity correction
##
## data:  sex_output
## X-squared = 23.084, df = 1, p-value = 1.551e-06
```

```
cp_output <- table(base_heart$cp, base_heart$output)
chisq.test(cp_output)
```

```
##
##  Pearson's Chi-squared test
##
## data:  cp_output
## X-squared = 80.979, df = 3, p-value < 2.2e-16
```

```
fbs_output <- table(base_heart$fbs, base_heart$output)
chisq.test(fbs_output)
```

```
##
##  Pearson's Chi-squared test with Yates' continuity correction
##
## data:  fbs_output
## X-squared = 0.092408, df = 1, p-value = 0.7611
```

```
restecg_output <- table(base_heart$restecg, base_heart$output)
chisq.test(restecg_output)
```

```
##
##  Pearson's Chi-squared test
##
## data:  restecg_output
## X-squared = 9.7297, df = 2, p-value = 0.007713
```

```r
exng_output <- table(base_heart$exng, base_heart$output)
chisq.test(exng_output)
```

```
##
##  Pearson's Chi-squared test with Yates' continuity correction
##
## data:  exng_output
## X-squared = 55.456, df = 1, p-value = 9.556e-14
```

```r
slp_output <- table(base_heart$slp, base_heart$output)
chisq.test(slp_output)
```

```
##
##  Pearson's Chi-squared test
##
## data:  slp_output
## X-squared = 46.889, df = 2, p-value = 6.578e-11
```

```r
caa_output <- table(base_heart$caa, base_heart$output)
chisq.test(caa_output)
```

```
##
##  Pearson's Chi-squared test
##
## data:  caa_output
## X-squared = 73.69, df = 4, p-value = 3.771e-15
```

```r
thall_output <- table(base_heart$thall, base_heart$output)
chisq.test(thall_output)
```

```
##
##  Pearson's Chi-squared test
##
## data:  thall_output
## X-squared = 84.61, df = 3, p-value < 2.2e-16
```

En este test, la hipótesis nula asume que no existen diferencias significativas en una variable categórica entre los grupos a comparar definidos por la variable categórica `output`. En el caso de las variables `sex`, `cp`, `restecg`, `exng`, `slp`, `caa` y `thall`, el p-valor es inferior al nivel de significancia, por lo que se rechaza la hipótesis nula y se concluye en que sí existen diferencias significativas. Para la variable `fbs`, el p-valor calculado es superior al nivel de significancia y por tanto no podemos rechazar la hipótesis nula ni concluir que existan diferencias significativas entre los dos grupos comparados.

El resultado de emplear el test de **Mann-Whitney** para analizar si hay diferencias en cada una de las variables numéricas entre los 2 grupos a comparar es el siguiente:

```r
lapply(base_heart[,c(1,4,5,8,10)], function(x) wilcox.test(x ~ base_heart$output, paired = F))
```

```
## $age
##
##  Wilcoxon rank sum test with continuity correction
##
## data:  x by base_heart$output
## W = 14394, p-value = 4.626e-05
## alternative hypothesis: true location shift is not equal to 0
##
##
## $trtbps
```

```
##
##  Wilcoxon rank sum test with continuity correction
##
## data:  x by base_heart$output
## W = 12931, p-value = 0.03223
## alternative hypothesis: true location shift is not equal to 0
##
##
## $chol
##
##  Wilcoxon rank sum test with continuity correction
##
## data:  x by base_heart$output
## W = 12850, p-value = 0.04243
## alternative hypothesis: true location shift is not equal to 0
##
##
## $thalachh
##
##  Wilcoxon rank sum test with continuity correction
##
## data:  x by base_heart$output
## W = 5725, p-value = 1.398e-13
## alternative hypothesis: true location shift is not equal to 0
##
##
## $oldpeak
##
##  Wilcoxon rank sum test with continuity correction
##
## data:  x by base_heart$output
## W = 16723, p-value = 3.347e-13
## alternative hypothesis: true location shift is not equal to 0
```

En este test, la hipótesis nula asume que las distribuciones de los dos grupos de datos que se comparan son las mismas. Por ello, como para todas las variables numéricas el p-valor calculado es inferior al nivel de significancia, es posible rechazar la hipótesis nula y concluir que existen diferencias significativas entre los pacientes con mayor riesgo de ataque cardíaco aquellos con un menor riesgo.

1.5.2 Correlación

El segundo método de análisis que se va a emplear consiste en estudiar la **correlación** entre las variables numéricas.

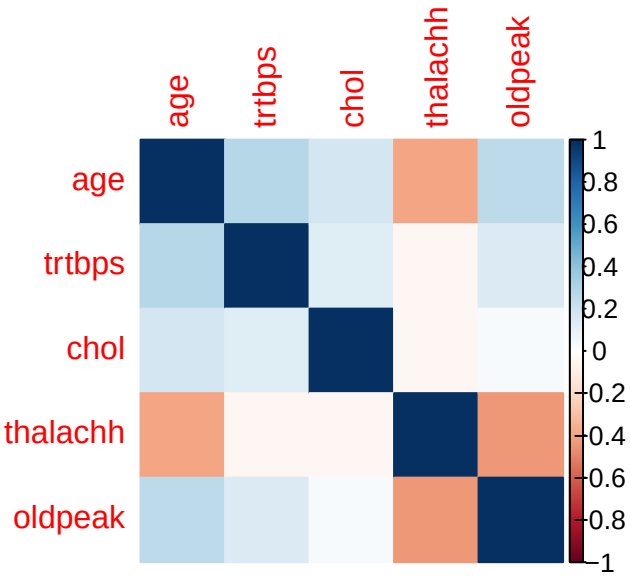
El coeficiente de correlación mide la asociación entre dos variables. Esta medida puede tomar valores entre -1 y 1, donde los extremos indican una correlación perfecta y un valor de 0 indica la ausencia de correlación. SI el valor presenta signo negativo, las variables se asocian de manera inversamente proporcional, mientras que un signo positivo indica una relación directamente proporcional (es decir, ambas variables aumentan o disminuyen de manera simultánea).

Para este análisis de correlación se empleará un test no paramétrico, en concreto la correlación de **Spearman**. Estos son los resultados:

```r
cm <- cor(num_heart, method = "spearman")
cm
```

```
##                age      trtbps        chol    thalachh     oldpeak
## age       1.0000000  0.28970501  0.18890292 -0.39345342  0.26362540
## trtbps    0.2897050  1.00000000  0.13021023 -0.04269948  0.15680732
## chol      0.1889029  0.13021023  1.00000000 -0.04036747  0.03956479
## thalachh -0.3934534 -0.04269948 -0.04036747  1.00000000 -0.43049461
## oldpeak   0.2636254  0.15680732  0.03956479 -0.43049461  1.00000000
```

```
corrplot(cm, method="color") # Representación gráfica
```



Como podemos observar, no se obtiene ninguna correlación perfecta entre variables, y por lo general los valores no se aproximan demasiado a los valores extremos, por lo que no encontramos correlaciones fuertes. Las variables que presentan una mayor correlación son `thalachh` y `oldpeak`. Sin embargo, este resultado no indica si la correlación es significativamente diferentes de cero, algo que analizamos a continuación:

```
# Correlacion Spearman + significancia
cor1 <- cor.test(num_heart$age,num_heart$trtbps, method="spearman")
cor1$p.value
```

```
## [1] 2.991875e-07
```

```
cor2 <- cor.test(num_heart$age,num_heart$chol, method="spearman")
cor2$p.value
```

```
## [1] 0.0009705773
```

```
cor3 <- cor.test(num_heart$age,num_heart$thalachh, method="spearman")
cor3$p.value
```

```
## [1] 1.270836e-12
```

```
cor4 <- cor.test(num_heart$age,num_heart$oldpeak, method="spearman")
cor4$p.value
```

```
## [1] 3.40496e-06
```

```
cor5 <- cor.test(num_heart$trtbps,num_heart$chol, method="spearman")
cor5$p.value
```

```
## [1] 0.02363128
```

```
cor6 <- cor.test(num_heart$trtbps,num_heart$thalachh, method="spearman")
cor6$p.value
```

```
## [1] 0.4597261
```

```r
cor7 <- cor.test(num_heart$trtbps,num_heart$oldpeak, method="spearman")
cor7$p.value
```

```
## [1] 0.006321296
```

```r
cor8 <- cor.test(num_heart$chol,num_heart$thalachh, method="spearman")
cor8$p.value
```

```
## [1] 0.4846222
```

```r
cor9 <- cor.test(num_heart$chol,num_heart$oldpeak, method="spearman")
cor9$p.value
```

```
## [1] 0.4933567
```

```r
cor10 <- cor.test(num_heart$thalachh,num_heart$oldpeak, method="spearman")
cor10$p.value
```

```
## [1] 4.683291e-15
```

Observamos que la correlación de mayor magnitud, la existente entre las variables `thalachh` y `oldpeak`, es estadísticamente significativa.

1.5.3 Métodos supervisados de clasificación.

En tercer lugar, vamos a elaborar una serie de modelos supervisados de clasificación y, finalmente, compararemos el rendimiento de estos modelos para evaluar cuál es el más adecuado para realizar predicciones sobre la probablidad de que un paciente posea mayor o menor riesgo de ataque al corazón en función de los valores que tomen el resto de variables.

El aprendizaje supervisado estima una función o modelo a partir de una serie de datos de entrenamiento, con el objetivo de predecir posteriormente el resultado de nuevos datos desconocidos. Los conjuntos de datos de entrenamiento están formados por pares de objetos que representan los datos de entrada y los resultados deseados. En nuestro caso, los resultados son etiquetas de clase, por lo que los modelos a elaborar serán de clasificación.

1.5.3.1 Partición de los datos Antes de la elaboración de los modelos de clasificación, es necesario dividir el conjunto de datos en un subconjunto de **entrenamiento** y otro de **test.** Con el primero, se entrenará un modelo de clasificación de forma que se definan una serie de reglas de clasificación. Con los datos del segundo, se estimará la exactitud (accuracy) del modelo, de manera que, si esta es aceptable, las reglas de clasificación definidas podrán ser utilizadas en nuevos datos de entrada con las mismas características, con el objetivo de predecir su resultado. En concreto, se utiliza el método de exclusión (holdout) para la división de datos en los dos subconjuntos. Así, los datos se dividen de manera aleatoria, asignando dos tercios de los datos al conjunto de entrenamiento y el tercio restante al conjunto de test.

Además, se emplea validación cruzada de tipo 4-fold para dividir los datos originales aleatoriamente en 4 subconjuntos (folds) mutuamente exclusivos y de tamaños similares. Esto indica que el entrenamiento y el testeo de los datos se realizará 4 veces, a partir de todas las combinaciones posibles de 3 subconjuntos para entrenamiento y dejando el subconjunto restante para testear el modelo.

```r
# Division Test-train
set.seed(1234)
h <- holdout(base_heart$output, ratio = 2/3, mode = "stratified")
train_heart <- base_heart[h$tr,]
test_heart <- base_heart[h$ts,]

# Validacion cruzada k-fold
train_control <- trainControl(method = "cv", number = 4)
```

Vamos a utilizar 5 métodos para crear distintos modelos de clasificación de nuestros datos, en concreto: Naive-Bayes, Random Forest, Support Vector Machine (svm), regresión logística y k Nearest Neighbors (kNN). A continuación se muestra la creación de cada uno de los modelos de clasificación y los resultados obtenidos en estos.

```
# 1) Naive Bayes
set.seed(1234)
# Entrenamiento modelo
nb_mod <-train(output ~ . , data = train_heart, method = "nb", trControl = train_control)
# Predicciones
nb_pred <- predict(nb_mod, newdata = test_heart)
# Resultados y accuracy
nb_results <- confusionMatrix(nb_pred, test_heart$output, positive = "1")
nb_results$table
```

```
##            Reference
## Prediction  0  1
##          0 28  3
##          1 18 52
```

```
nb_results$overall
```

```
##       Accuracy          Kappa  AccuracyLower  AccuracyUpper   AccuracyNull
##   7.920792e-01   5.693401e-01   6.999032e-01   8.664453e-01   5.445545e-01
## AccuracyPValue  McnemarPValue
##   1.897109e-07   2.250227e-03
```

```
nb_acc <- nb_results$overall['Accuracy']

# 2) RF
set.seed(1234)
rf_mod <-train(output ~ . , data = train_heart, method = "rf", trControl = train_control)
rf_pred <- predict(rf_mod, newdata = test_heart)
rf_results <-confusionMatrix(rf_pred, test_heart$output, positive = "1")
rf_results$table
```

```
##            Reference
## Prediction  0  1
##          0 35  6
##          1 11 49
```

```
rf_results$overall
```

```
##       Accuracy          Kappa  AccuracyLower  AccuracyUpper   AccuracyNull
##   8.316832e-01   6.576271e-01   7.442421e-01   8.987816e-01   5.445545e-01
## AccuracyPValue  McnemarPValue
##   1.095575e-09   3.319755e-01
```

```
rf_acc <- rf_results$overall['Accuracy']


# 3) SVM
set.seed(1234)
svm_mod <-train(output ~ . , data = train_heart, method = "svmRadial", trControl = train_control)
svm_pred <- predict(svm_mod, newdata = test_heart)
svm_results <-confusionMatrix(svm_pred, test_heart$output, positive = "1")
svm_results$table
```

```
##          Reference
## Prediction  0  1
##          0 34  7
##          1 12 48
```

```
svm_results$overall
```

```
##       Accuracy          Kappa  AccuracyLower  AccuracyUpper   AccuracyNull
##   8.118812e-01   6.173480e-01   7.219221e-01   8.827724e-01   5.445545e-01
## AccuracyPValue  McnemarPValue
##   1.633444e-08   3.587954e-01
```

```
svm_acc <- svm_results$overall['Accuracy']

# 4) glm logistic regression
set.seed(1234)
lg_mod <-train(output ~ . , data = train_heart, method = "glm", trControl = train_control)
lg_pred <- predict(lg_mod, newdata = test_heart)
lg_results <-confusionMatrix(lg_pred, test_heart$output, positive = "1")
lg_results$table
```

```
##          Reference
## Prediction  0  1
##          0 32  2
##          1 14 53
```

```
lg_results$overall
```

```
##       Accuracy          Kappa  AccuracyLower  AccuracyUpper   AccuracyNull
##   8.415842e-01   6.736672e-01   7.555271e-01   9.066534e-01   5.445545e-01
## AccuracyPValue  McnemarPValue
##   2.565733e-10   5.959526e-03
```

```
lg_acc <- lg_results$overall['Accuracy']

# 5) knn
set.seed(1234)
knn_mod <-train(output ~ . , data = train_heart, method = "knn", trControl = train_control)
knn_pred <- predict(knn_mod, newdata = test_heart)
knn_results <-confusionMatrix(knn_pred, test_heart$output, positive = "1")
knn_results$table
```

```
##          Reference
## Prediction  0  1
##          0 32 14
##          1 14 41
```

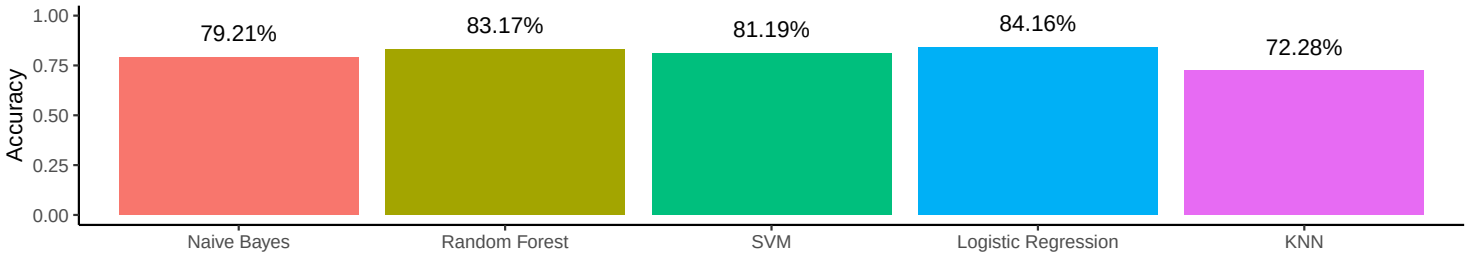
```
knn_results$overall
```

```
##       Accuracy          Kappa  AccuracyLower  AccuracyUpper   AccuracyNull
##   0.7227722772   0.4411067194   0.6248176919   0.8072312813   0.5445544554
## AccuracyPValue  McnemarPValue
##   0.0001804183   1.0000000000
```

```
knn_acc <- knn_results$overall['Accuracy']
```

Finalmente, comparamos la exactitud de los diferentes modelos:

```r
model_names <- c("Naive Bayes", "Random Forest", "SVM", "Logistic Regression", 'KNN')

acc <- c(nb_acc, rf_acc, svm_acc, lg_acc, knn_acc)
df_acc <- data.frame(model_names, acc)
df_acc$model_names <- factor(df_acc$model_names, levels = df_acc$model_names)

ggplot( mapping = aes(x=df_acc$model_names)) +
  geom_bar(aes(y = ..acc.., fill = df_acc$model_names),width = 0.9,show.legend = FALSE)+
  geom_text(aes( y = ..acc.., label = scales::percent(..acc..)),
            size=4, stat = "count", vjust = -1)+ ylim(0, 1)+labs(y = "Accuracy", x="")+
  theme(text = element_text(size = 15)) + theme_classic()
```



Podemos observar que la exactitud mayor se obtiene al utilizar el método de **regresión logística**, seguido del método random forest.

1.6 Resolución del problema

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

Se ha concluido que:

- Al comparar los grupos de pacientes con menor riesgo de ataque frente a los de mayor riesgo de ataque se ha obtenido que los pacientes del estudio que presentan menos probabilidades de ataque cardíaco son los de edad más elevada. Así, la edad no es un factor determinante ante la posibilidad de sufrir un ataque cardíaco.
- De los 302 registros del estudio, existe mayor proporción de hombres y menor proporción de mujeres con mayor probabilidad de ataque cardíaco. Si se profundiza el estudio, se puede clasificar por edad las mujeres y hombres con mayor riesgo de ataque.
- En la prueba de contraste de hipótesis entre los grupos de pacientes con mayor riesgo de ataque frente a los de menor riesgo, y al comparar las variables categóricas y su nivel de significancia para estos grupos, se concluye que las variables `sex`, `cp`, `restecg`, `exng`, `slp`, `caa` y `thall` influyen ante el riesgo de padecer un ataque cardíaco. Y presentan mayor posibilidades de padecer esta patología aquellos que han alcanzado una frecuencia cardíaca máxima de mayor magnitud.
- Al comparar las variables numéricas `age`, `resting blood pressure`, `cholesterol`, `Max Heart rate` y `old peak` entre los 2 grupos de estudio, se obtiene que existen diferencias significativas entre los pacientes con mayor riesgo de ataque cardíaco aquellos con un menor riesgo. Es decir, estas variables también influyen en el riesgo de padecer esta afección.
- En el análisis de correlación entre las variables numéricas, se obtuvo que existe una fuerte correlación entre las variables `thalachh`(frecuencia cardíaca máxima alcanzada) y `oldpeak` (punto máximo anterior inducido por el ejercicio).
- Al aplicar diferentes métodos supervisados de clasificación (Naive-Bayes, Random Forest, SVM, regresión logística y kNN) y comparar sus rendimientos para predecir si un paciente presenta mayor o menor riesgo de ataque de corazón, el modelo de regresión logística obtuvo la mayor exactitud, de un 84.16% frente al 83.17% del modelo random forest y 81.19% del modelo SVM.
- En definitiva, el análisis de los datos que forman este dataset sirve de utilidad para comprender parte de esta problemática, que sigue siendo un asunto de interés en la actualidad y que todavía se necesita estudiar en mayor profundidad. Con los resultados de este estudio podemos observar que existen algunos factores sociodemográficos y de salud que pueden propiciar a que un individuo presente una mayor probabilidad de sufrir un ataque cardíaco.

1.7 Tabla de contribuciones

Contribuciones	**Firma**
Investigación previa	AN, BP
Redacción de las respuestas	AN, BP
Desarrollo del código	AN, BP
Participación en el video	AN, BP