

PRÁCTICA 2: ¿Cómo realizar la limpieza y análisis de datos?

Lukaz Martin Doehne y Pablo Vadillo Berganza

04/01/2023

Índice

1.Descripción del dataset. ¿Por qué es importante y qué pregunta/problema pretende responder?	2
2.Integración y selección	3
3. Limpieza de los datos	4
3.1. ¿Los datos contienen ceros o elementos vacíos?	4
3.2. Identifica y gestiona los valores extremos	4
4. Análisis de los datos	7
4.1. Selección de los grupos de datos que se quieren analizar/comparar	7
4.2. Comprobación de la normalidad y homogeneidad de la varianza	7
4.3. Aplicación de pruebas estadísticas para comparar los grupos de datos	7
5. Representación de los resultados a partir de tablas y gráficas	7
6. Resolución del problema	7

1.Descripción del dataset. ¿Por qué es importante y qué pregunta/problema pretende responder?

En primer lugar, cargaremos los datos con los que trabajar.

Nuestro dataset está compuesto por 303 registros y 14 variables. Contiene información clínica de pacientes con el fin de poder predecir la probabilidad de fallo cardíaco.

Para ello, la variable a predecir, “output”, contiene información acerca del estrechamiento de los vasos sanguíneos obtenida a través de una angiografía. Si el estrechamiento es inferior al 50% toma el valor 0 y por tanto el paciente no se considera en riesgo de padecer una enfermedad del corazón. Por el contrario, si el estrechamiento es superior al 50%, la variable tomará el valor 1 y el paciente tendrá una mayor probabilidad de ataque al corazón.

A partir de aquí, será interesante realizar un análisis del resto de campos para ver cuáles son determinantes en la predicción del fallo cardíaco.

Pero antes, vamos a realizar una breve exploración del dataset.

```
str(df)
```

```
## 'data.frame':    303 obs. of  14 variables:
## $ age      : int  63 37 41 56 57 57 56 44 52 57 ...
## $ sex      : int  1 1 0 1 0 1 0 1 1 1 ...
## $ cp       : int  3 2 1 1 0 0 1 1 2 2 ...
## $ 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      : int  1 0 0 0 0 0 0 0 1 0 ...
## $ restecg  : int  0 1 0 1 1 1 0 1 1 1 ...
## $ thalachh : int  150 187 172 178 163 148 153 173 162 174 ...
## $ exng     : int  0 0 0 0 1 0 0 0 0 0 ...
## $ oldpeak  : num  2.3 3.5 1.4 0.8 0.6 0.4 1.3 0 0.5 1.6 ...
## $ slp      : int  0 0 2 2 2 1 1 2 2 2 ...
## $ caa      : int  0 0 0 0 0 0 0 0 0 0 ...
## $ thall    : int  1 2 2 2 2 1 2 3 3 2 ...
## $ output   : int  1 1 1 1 1 1 1 1 1 1 ...
```

Todas las variables son numéricas.

```
summary(df)
```

```
##      age      sex      cp      trtbps
## Min.   :29.00  Min.   :0.0000  Min.   :0.000  Min.    : 94.0
## 1st Qu.:47.50  1st Qu.:0.0000  1st Qu.:0.000  1st Qu.:120.0
## Median :55.00  Median :1.0000  Median :1.000  Median :130.0
## Mean   :54.37  Mean   :0.6832  Mean   :0.967  Mean   :131.6
## 3rd Qu.:61.00  3rd Qu.:1.0000  3rd Qu.:2.000  3rd Qu.:140.0
## Max.   :77.00  Max.   :1.0000  Max.   :3.000  Max.   :200.0
##      chol      fbs      restecg      thalachh
## Min.   :126.0  Min.   :0.0000  Min.   :0.0000  Min.    : 71.0
## 1st Qu.:211.0  1st Qu.:0.0000  1st Qu.:0.0000  1st Qu.:133.5
## Median :240.0  Median :0.0000  Median :1.0000  Median :153.0
## Mean   :246.3  Mean   :0.1485  Mean   :0.5281  Mean   :149.6
## 3rd Qu.:274.5  3rd Qu.:0.0000  3rd Qu.:1.0000  3rd Qu.:166.0
```

```

## Max.      :564.0    Max.      :1.0000    Max.      :2.0000    Max.      :202.0
##      exng      oldpeak      slp      caa
## Min.      :0.0000    Min.      :0.00    Min.      :0.000    Min.      :0.0000
## 1st Qu.:0.0000    1st Qu.:0.00    1st Qu.:1.000    1st Qu.:0.0000
## Median :0.0000    Median :0.80    Median :1.000    Median :0.0000
## Mean    :0.3267    Mean    :1.04    Mean    :1.399    Mean    :0.7294
## 3rd Qu.:1.0000    3rd Qu.:1.60    3rd Qu.:2.000    3rd Qu.:1.0000
## Max.    :1.0000    Max.    :6.20    Max.    :2.000    Max.    :4.0000
##      thall      output
## Min.      :0.000    Min.      :0.0000
## 1st Qu.:2.000    1st Qu.:0.0000
## Median :2.000    Median :1.0000
## Mean    :2.314    Mean    :0.5446
## 3rd Qu.:3.000    3rd Qu.:1.0000
## Max.    :3.000    Max.    :1.0000

```

Para terminar con este apartado, vamos a explicar brevemente todos y cada uno de los campos del dataset:

- Age: Edad de los pacientes en años. Toma valores entre 29 y 77. La media es 54,37.
- Sex: Sexo de los pacientes. (1 = hombre; 0 = mujer).
- Cp: Dolor de pecho. (1 = angina típica; 2 = angina atípica; 3 = dolor no anginal; 4 = asintomático).
- Trtbps: Presión arterial en reposo. Se trata del valor tomado en el ingreso al hospital, en mm Hg. Toma valores entre 94 y 200, siendo la media de 131,6.
- Chol: Colesterol sérico. Medido en mg/dl. Toma valores entre 126 y 564, con una media de 246,3.
- Fbs: Nivel de azúcar en sangre en ayunas. (1 = > 120 mg/dl; 0 = <= 120 mg/dl).
- Restecg: Resultados del electrocardiograma en reposo. (0 = normal; 1 = onda ST-T anómala; 2 = hipertrofia ventricular izquierda).
- Thalachh: Máximo pulso cardíaco obtenido. Toma valores entre 71 y 202. La media es 149,6.
- Exng: Angina inducida del ejercicio. (1 = sí; 0 = no).
- Oldpeak: Depresión del segmento ST inducida por ejercicio relativo al descanso. Toma valores entre 0 y 6,2, con una media de 1,04.
- Slp: Pendiente del segmento ST del pico del ejercicio. (1 = ascendente; 2 = plano; 3 = descendente).
- Caa: Número de los principales vasos sanguíneos coloreados por la fluoroscopia. Toma valores entre 0 y 4, con una media de 0,73.
- Thall: Talasemia. Menor nivel de hemoglobina. (1 = defecto fijo; 2 = normal; 3 = defecto reversible). El defecto fijo hace referencia a un defecto que ocurre tanto en reposo como durante el esfuerzo. El defecto reversible, por el contrario, hace referencia a un defecto que ocurre durante el esfuerzo que no existía durante el reposo.
- Output: La variable a predecir. Diagnóstico de fallo cardíaco (0 = Estrechamiento de vasos sanguíneos < 50%; 1 = Estrechamiento de vasos sanguíneos > 50%).

2.Integración y selección

3. Limpieza de los datos

3.1. ¿Los datos contienen ceros o elementos vacíos?

Ante elementos vacíos se pueden seguir principalmente dos estrategias: la **imputación** y la **eliminación**.

Para la primera, se trata de rellenar los valores nulos basandose en los valores no vacíos. Esto se realizar típicamente mediante *knn*, interpolación o cogiendo la media de una columna.

La segunda estrategia consiste en eliminar las filas con valores nulos. Esta se suele usar cuando se tiene un dataset grande y los valores nulos no suponen un gran porcentaje.

Podemos comprobar que no se encuentran elementos nulos en el csv con el siguiente código:

```
sapply(df, 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
```

Dado que no encontramos valores nulos no aplicamos ninguna de las estrategias. Pero vamos a discretizar la columna *age* para mejor procesamiento.

Discretización

Para la variable edad, en lugar de tener el valor exacto crearemos grupos por decenas de edad.

E.g:

· 30-39 -> 3

· 40-49 -> 4

...

Vemos el resultado

```
##
##  2   3   4   5   6   7
##  1  15  72 125  80  10
```

3.2. Identifica y gestiona los valores extremos

Los valores extremos (*outliers*) los identificaremos como aquellos que se encuentran fuera del rango [Lo,Ho] donde:

· $Lo = Q1 - (1.5 * IQR)$

· $Ho = Q3 + (1.5 * IQR)$

y $IQR = Q3 - Q1$. También hay una fórmula que trata outliers los que sobrepasan la media +/- la desviación estándar, pero optaremos por la fórmula descrita, ya que también es la que usa *R* por defecto en la función *boxplot*.

A continuación, mostramos la cantidad de outliers por columna con la siguiente función:

```

outliers<-function(x){
  outliers<-boxplot.stats(x)$out
  return (length(outliers))
}

```

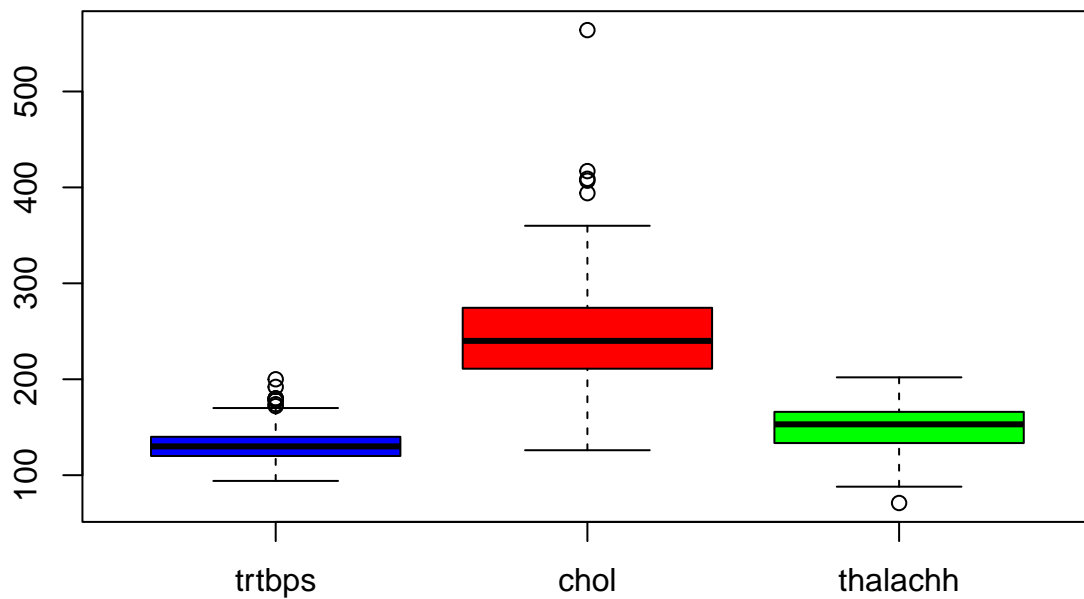
```

##      age      sex      cp  trtbps      chol      fbs  restecg thalachh
##       0       0       0      9       5      45       0       1
##    exng  oldpeak    slp     caa     thall  output
##       0       5      0     25      2      0

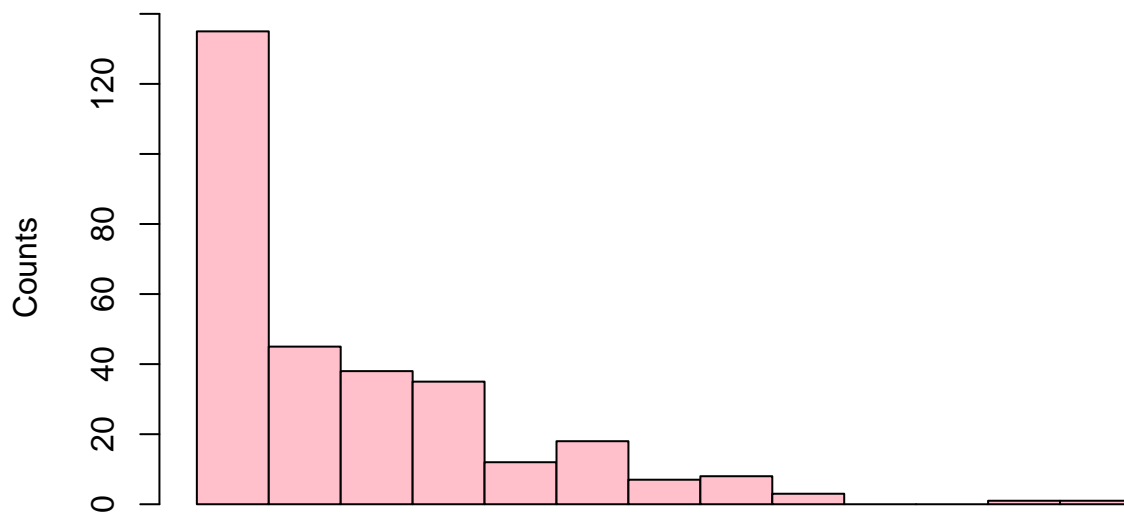
```

Encontramos varios outliers en los datos. Las columnas *fbs*, *caa* y *thall* son columnas **númericas discretas** por lo que no los consideramos outliers, sino solo un dataset desequilibrado.

Visualizamos para entender mejor las demás variables.



La variable *oldpeak* la visualizamos en un histograma al estar en una escala diferente a las demás.



Oldpeak Histogram

Viendo las visualizaciones suponemos que no ha habido errores a la hora de capturar los datos. Si que es cierto que en la columna *chol* encontramos un valor muy alejado del valor medio, pero lo mantendremos dentro del analisis al igual que el resto de los outliers. Si creamos un modelo de predicci3n podemos asegurar de incluir los outliers tanto en el train como en el test set.

4. Análisis de los datos

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

Hay 3 grupos en especial que nos interesa estudiar en función del fallo cardíaco:

- La variable **sex**. En función de hombres y mujeres.
- La variable **age**. En función de la edad de los pacientes.
- La variable **trtbps**. En función de la presión arterial de los pacientes. Información extraída de: mayoclinic.org

```
# Agrupación por género
df.male <- df[df$sex == 1,]
df.female <- df[df$sex == 0,]

# Agrupación por edad
df.young_adult <- df[df$age<4,]
df.adult <- df[df$age>=4&df$age<6,]
df.old_adult <- df[df$age>=6,]

# Agrupación por presión arterial
df.normal_pressure <- df[df$trtbps<120,]
df.high_pressure <- df[df$trtbps>=120&df$trtbps<130,]
df.hipertension1 <- df[df$trtbps>=130&df$trtbps<140,]
df.hipertension2 <- df[df$trtbps>=140&df$trtbps<180,]
df.crisis_hipertension <- df[df$trtbps>=180,]
```

4.2. Comprobación de la normalidad y homogeneidad de la varianza

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

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

6. Resolución del problema

This is an R Markdown document. Markdown is a simple formatting syntax for authoring HTML, PDF, and MS Word documents. For more details on using R Markdown see <http://rmarkdown.rstudio.com>.

When you click the **Knit** button a document will be generated that includes both content as well as the output of any embedded R code chunks within the document. You can embed an R code chunk like this:

```
##      speed      dist
## Min.   : 4.0    Min.   :  2.00
## 1st Qu.:12.0    1st Qu.: 26.00
## Median :15.0    Median : 36.00
## Mean   :15.4    Mean   : 42.98
## 3rd Qu.:19.0    3rd Qu.: 56.00
## Max.   :25.0    Max.   :120.00
```