# Tipologia i cicle de vida de les dades: Pràctica 2

Autor: María Martínez Gil

Gener 2024

## Contents



---

## Descripció de dataset

---

El conjunt suggerit per a l'elaboració de la pràctica és "Heart Attack Analysis & Prediction Dataset" disponible en l'enllaç https://www.kaggle.com/datasets/rashikrahmanpritom/heart-attack-analysis-prediction-dataset?resource=download .

Aquest dataset conté informació de caràcter mèdic que es volen fer servir per tal de predir quina pacient tenen un alt risc de patir un atac al cor.

Aquesta capacitat és interessant per tal de poder fer un seguiment més exhaustiu d'aquell pacients que presenten un alt risc d'atac i poder, en última instància, evitar la seua mort.

La pregunta a respondre seria la següent: "Va a patir aquest pacient un atac al cor?"

Les columnes que podem trobar són les següents:

- age. Edat de la persona
- sex. Sexe de la persona, inclou els valors 0 i 1.

- cp. Tipus de dolor de pit.
- trtbps. Tensió sanguínea en repòs.
- chol. Colesterol en mg/dl mesurat amb sensor via BMI.
- fbs. Glucèmia en dejú.
- restecg. Resultats electrocardiograma en repòs.
- thalachh. Ritme cardíac màxim assolit.
- exng. Angina induïda per exercici. 0 és no, 1 és sí.
- oldpeak. Pic anterior.
- slp. Slope.
- caa. Nombre de vasos majors.
- thall. Rati d'interès.
- output. Eixida de risc d'atac.

```
ruta<-file.choose()

dades <- read.csv(ruta, sep=",", dec = ".")

str(dades)
```

```
## '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 ...
```

Podem veure que el nostre conjunt està format per 303 registres i 14 variables.

---

## Integració i selecció

---

Inicialment, podem veure que tenim dos conjunts de dades, en dos excels diferents.

Com que no observem que hi haja cap variable per poder-los juntar tenim dues opcions:

- No utilitzar el datasat amb les lectures d'oxigen en sang.
- Juntar els dos datasets per la posició que ocupen en cada document. És a dir, la línia 1 amb la 1, la 2 amb la 2, etc.

En el nostre cas, i sense confirmació de com s'ha de fer la unió, anem a considerar la opció més prudent no ajuntar les dades per no donar lloc a interpretacions errònies.

A més a més, anem a destacar que en inici no tenim cap instrucció de seleccionar un cap grup concret d'individus i, per tant, utilitzarem la totalitat del conjunt.

---

# Neteja de les dades

---

## Zeros i elements buits

Aabans de començar a mb les comprovacions, anem a analitzar el tipus de dades que tenim en el conjunt.

```
summary(dades)
```

```
##       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
```

```
str(dades)
```

```
## '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 ...
```

```
dades_num <- apply(dades, 2, as.numeric)
```

Com que veien que hi ha variables de tipus "int" anem a passar-les a tipus numèric. Finalment, comprovarem que el tipus és el que volíem.

```
dades_num <- apply(dades, 2, as.numeric)
str(dades_num)
```

```
##  num [1:303, 1:14] 63 37 41 56 57 57 56 44 52 57 ...
##  - attr(*, "dimnames")=List of 2
##   ..$ : NULL
##   ..$ : chr [1:14] "age" "sex" "cp" "trtbps" ...
```

Anem a comprovar si existeixen valors buits.

```
colSums(is.na(dades_num))
```

```
##      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
```

Podem veure que no apareix cap valor buit.

## Valors extrems

Per als valors extrem anem a mostrar els valors atípics utilitzant una comanda a partir d'un boxplot.

```
boxplot.stats(dades$age)$out
```

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

```
boxplot.stats(dades$sex)$out
```

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

```
boxplot.stats(dades$cp)$out
```

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

```
boxplot.stats(dades$trtbps)$out
```

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

```
boxplot.stats(dades$chol)$out
```

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

```
boxplot.stats(dades$fbs)$out
```

```
##  [1] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
## [39] 1 1 1 1 1 1 1
```

```
boxplot.stats(dades$restecg)$out
```

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

```
boxplot.stats(dades$thalachh)$out
```

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

```
boxplot.stats(dades$exng)$out
```

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

```
boxplot.stats(dades$oldpeak)$out
```

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

```
boxplot.stats(dades$slp)$out
```

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

```
boxplot.stats(dades$caa)$out
```

```
##  [1] 3 4 3 3 4 4 4 3 3 3 3 3 3 3 3 3 3 3 3 4 3 3 3 3
```

```
boxplot.stats(dades$thall)$out
```

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

```
boxplot.stats(dades$output)$out
```

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

Podem veure que encara que ens eixen diversos valors atípics aparentment no són anòmals i els podríem deixar en el conjunt de dades.

---

# Anàlisi de les dades

---

## Selecció de grups

```
dades_pca <- prcomp(dades[,c(1:13)], center = TRUE, scale = TRUE)
summary(dades_pca)
```

```
## Importance of components:
##                           PC1    PC2     PC3     PC4     PC5     PC6     PC7
## Standard deviation     1.6622 1.2396 1.10582 1.08681 1.01092 0.98489 0.92885
## Proportion of Variance 0.2125 0.1182 0.09406 0.09086 0.07861 0.07462 0.06637
## Cumulative Proportion  0.2125 0.3307 0.42481 0.51567 0.59428 0.66890 0.73527
##                           PC8    PC9    PC10    PC11    PC12   PC13
## Standard deviation     0.88088 0.8479 0.78840 0.72808 0.65049 0.6098
## Proportion of Variance 0.05969 0.0553 0.04781 0.04078 0.03255 0.0286
## Cumulative Proportion  0.79495 0.8503 0.89807 0.93885 0.97140 1.0000
```

Podem veure que segons l'anàlisi de components principals, per a explicar, al menys, el 95% de la variància necessitem totes les variables i, per tant, no en descartarem cap.

```r
dades.scaled <- scale(dades[,c(1:13)])
```

A més a més, com que el conjunt de dades no és particularment gran, no anem a reduir tampoc la quantitat de registres que tenim.

## Comprovació de normalitat

Anem a comprovar la normalitat de les nostres variables.

```r
shapiro_results <- apply(dades_num, 2, shapiro.test)

print(shapiro_results)
```

```
## $age
##
##  Shapiro-Wilk normality test
##
## data:  newX[, i]
## W = 0.98637, p-value = 0.005798
##
##
## $sex
##
##  Shapiro-Wilk normality test
##
## data:  newX[, i]
## W = 0.58573, p-value < 2.2e-16
##
##
## $cp
##
##  Shapiro-Wilk normality test
##
## data:  newX[, i]
## W = 0.79016, p-value < 2.2e-16
##
##
## $trtbps
##
##  Shapiro-Wilk normality test
##
## data:  newX[, i]
## W = 0.96592, p-value = 1.458e-06
##
##
## $chol
##
##  Shapiro-Wilk normality test
##
## data:  newX[, i]
## W = 0.94688, p-value = 5.365e-09
##
##
## $fbs
##
```

```
##  Shapiro-Wilk normality test
##
## data:  newX[, i]
## W = 0.42399, p-value < 2.2e-16
##
##
## $restecg
##
##  Shapiro-Wilk normality test
##
## data:  newX[, i]
## W = 0.67932, p-value < 2.2e-16
##
##
## $thalachh
##
##  Shapiro-Wilk normality test
##
## data:  newX[, i]
## W = 0.97632, p-value = 6.621e-05
##
##
## $exng
##
##  Shapiro-Wilk normality test
##
## data:  newX[, i]
## W = 0.59126, p-value < 2.2e-16
##
##
## $oldpeak
##
##  Shapiro-Wilk normality test
##
## data:  newX[, i]
## W = 0.84418, p-value < 2.2e-16
##
##
## $slp
##
##  Shapiro-Wilk normality test
##
## data:  newX[, i]
## W = 0.74465, p-value < 2.2e-16
##
##
## $caa
##
##  Shapiro-Wilk normality test
##
## data:  newX[, i]
## W = 0.72812, p-value < 2.2e-16
##
##
```

```
## $thall
##
##  Shapiro-Wilk normality test
##
## data:  newX[, i]
## W = 0.75058, p-value < 2.2e-16
##
##
## $output
##
##  Shapiro-Wilk normality test
##
## data:  newX[, i]
## W = 0.63359, p-value < 2.2e-16
```

Hem comprovat que, segons el test de Shapiro-Wilk cap de les vaiables compleix la condició de normalitat. Per tant, haurem d'utilitzar versions no paramètriques per a les comprovacions d'homocesticitat. No obstant això, no anem a fer un anàlisi d'homocesticitat ja que no ebservem un grup clar de variables en les quals tindria sentit comprara les variàncies.

## Aplicació de proves estadístiques

```
response_variable <- dades_num[, "output"]

predictor_variables <- dades_num[, -which(names(dades_num) == "response_column_name")]

model <- glm(response_variable ~ ., family = binomial, data = as.data.frame(dades_num))

summary(model)
```

```
##
## Call:
## glm(formula = response_variable ~ ., family = binomial, data = as.data.frame(dades_num))
##
## Deviance Residuals:
##        Min          1Q      Median          3Q         Max
## -2.409e-06  -2.409e-06   2.409e-06   2.409e-06   2.409e-06
##
## Coefficients:
##               Estimate Std. Error z value Pr(>|z|)
## (Intercept) -2.657e+01  2.986e+05   0.000    1.000
## age         -2.358e-11  2.711e+03   0.000    1.000
## sex          7.763e-10  4.880e+04   0.000    1.000
## cp          -1.831e-10  2.347e+04   0.000    1.000
## trtbps       8.633e-12  1.270e+03   0.000    1.000
## chol        -1.794e-12  4.245e+02   0.000    1.000
## fbs         -8.066e-12  6.000e+04   0.000    1.000
## restecg      2.747e-10  4.025e+04   0.000    1.000
## thalachh     7.330e-12  1.150e+03   0.000    1.000
## exng        -9.580e-11  5.235e+04   0.000    1.000
## oldpeak     -2.717e-11  2.331e+04   0.000    1.000
## slp         -1.109e-10  4.288e+04   0.000    1.000
## caa          1.271e-10  2.277e+04   0.000    1.000
## thall       -3.810e-11  3.653e+04   0.000    1.000
```

```
## output       5.313e+01  5.914e+04   0.001    0.999
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 4.1764e+02  on 302  degrees of freedom
## Residual deviance: 1.7579e-09  on 288  degrees of freedom
## AIC: 30
##
## Number of Fisher Scoring iterations: 25
```

Notem que en quest cas la mesura que utilitzem per avalacuar la qualitat del model és el AIC.

```
selected_columns <- dades_num[, -ncol(dades_num)]

correlation_matrix <- cor(selected_columns, method = "spearman")

print(correlation_matrix)
```

```
##                 age         sex          cp      trtbps        chol
## age       1.00000000 -0.09913088 -0.08749412  0.28561681  0.19578599
## sex      -0.09913088  1.00000000 -0.06204094 -0.05294119 -0.15134205
## cp       -0.08749412 -0.06204094  1.00000000  0.03541319 -0.09172085
## trtbps    0.28561681 -0.05294119  0.03541319  1.00000000  0.12656163
## chol      0.19578599 -0.15134205 -0.09172085  0.12656163  1.00000000
## fbs       0.11397832  0.04503179  0.08977463  0.15198393  0.01846298
## restecg  -0.13276876 -0.04838909  0.06563997 -0.12584121 -0.16193312
## thalachh -0.39805244 -0.03986798  0.32401302 -0.04040735 -0.04676639
## exng      0.08967860  0.14166381 -0.41825595  0.05291815  0.09151399
## oldpeak   0.26829122  0.10071533 -0.16144910  0.15426674  0.04525960
## slp      -0.18404841 -0.02501041  0.15947787 -0.08656953 -0.01255073
## caa       0.34095479  0.11936769 -0.21600615  0.09013959  0.11198119
## thall     0.08725391  0.25082085 -0.20784032  0.05967277  0.08362788
##                  fbs     restecg    thalachh        exng     oldpeak
## age       0.113978316 -0.13276876 -0.39805244  0.08967860  0.26829122
## sex       0.045031789 -0.04838909 -0.03986798  0.14166381  0.10071533
## cp        0.089774633  0.06563997  0.32401302 -0.41825595 -0.16144910
## trtbps    0.151983926 -0.12584121 -0.04040735  0.05291815  0.15426674
## chol      0.018462985 -0.16193312 -0.04676639  0.09151399  0.04525960
## fbs       1.000000000 -0.08150785 -0.01427341  0.02566515  0.02836271
## restecg  -0.081507846  1.00000000  0.08786325 -0.07739900 -0.07737235
## thalachh -0.014273407  0.08786325  1.00000000 -0.40085981 -0.43324053
## exng      0.025665147 -0.07739900 -0.40085981  1.00000000  0.29717297
## oldpeak   0.028362712 -0.07737235 -0.43324053  0.29717297  1.00000000
## slp      -0.045785534  0.11366148  0.43696753 -0.27447469 -0.59484671
## caa       0.134512530 -0.09786191 -0.25734715  0.16202496  0.22489523
## thall    -0.006737388 -0.01098232 -0.16058130  0.24711322  0.25502616
##                  slp         caa        thall
## age      -0.18404841  0.34095479  0.087253908
## sex      -0.02501041  0.11936769  0.250820845
## cp        0.15947787 -0.21600615 -0.207840318
## trtbps   -0.08656953  0.09013959  0.059672772
## chol     -0.01255073  0.11198119  0.083627883
## fbs      -0.04578553  0.13451253 -0.006737388
## restecg   0.11366148 -0.09786191 -0.010982317
## thalachh  0.43696753 -0.25734715 -0.160581298
```

```
## exng     -0.27447469  0.16202496  0.247113221
## oldpeak  -0.59484671  0.22489523  0.255026159
## slp       1.00000000 -0.09990056 -0.154885651
## caa      -0.09990056  1.00000000  0.189102994
## thall    -0.15488565  0.18910299  1.000000000
```

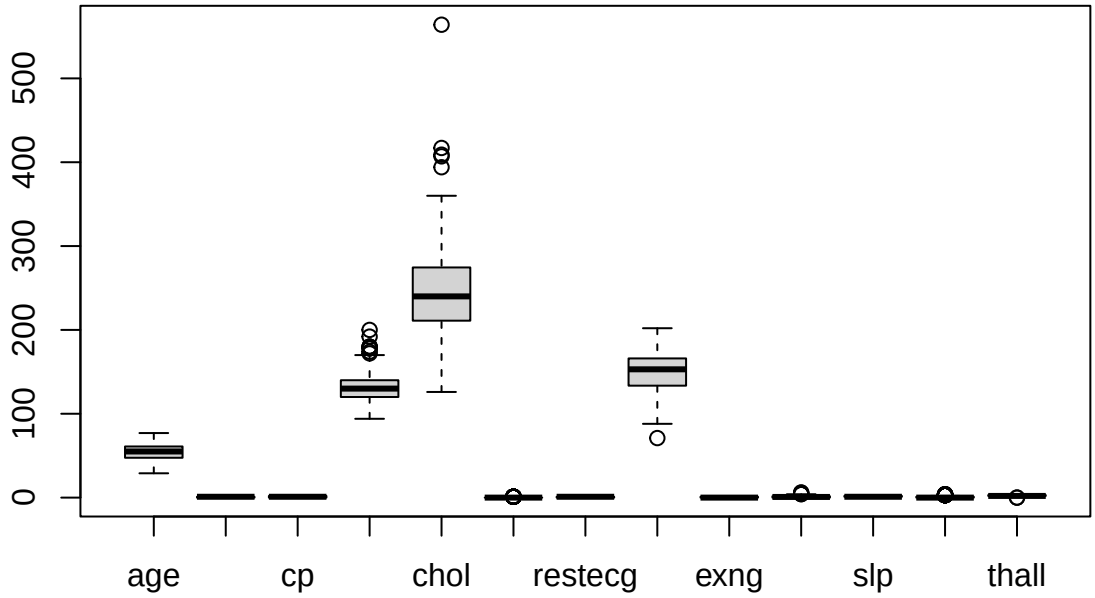
Podem veure que no existeixen correlacions entre les variables del nostre conjunt ja que en cap d'elles es super el 0,60 en valor absolut.

---

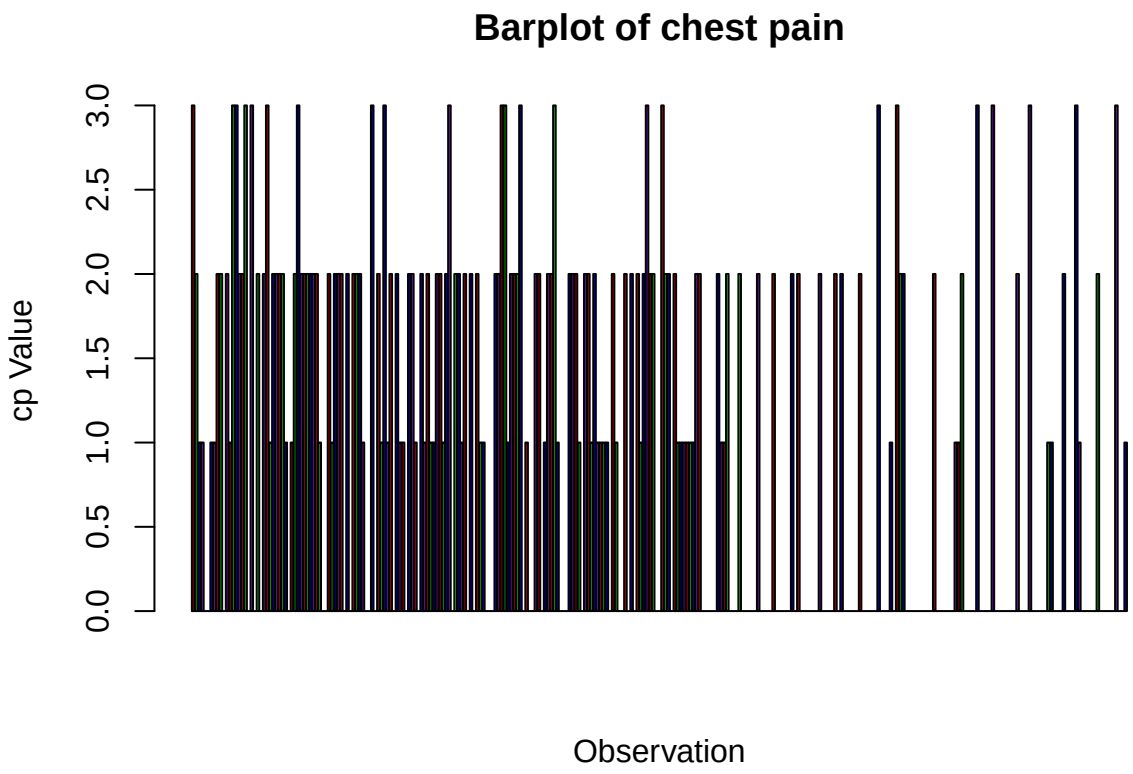
## Representació dels resultats

---

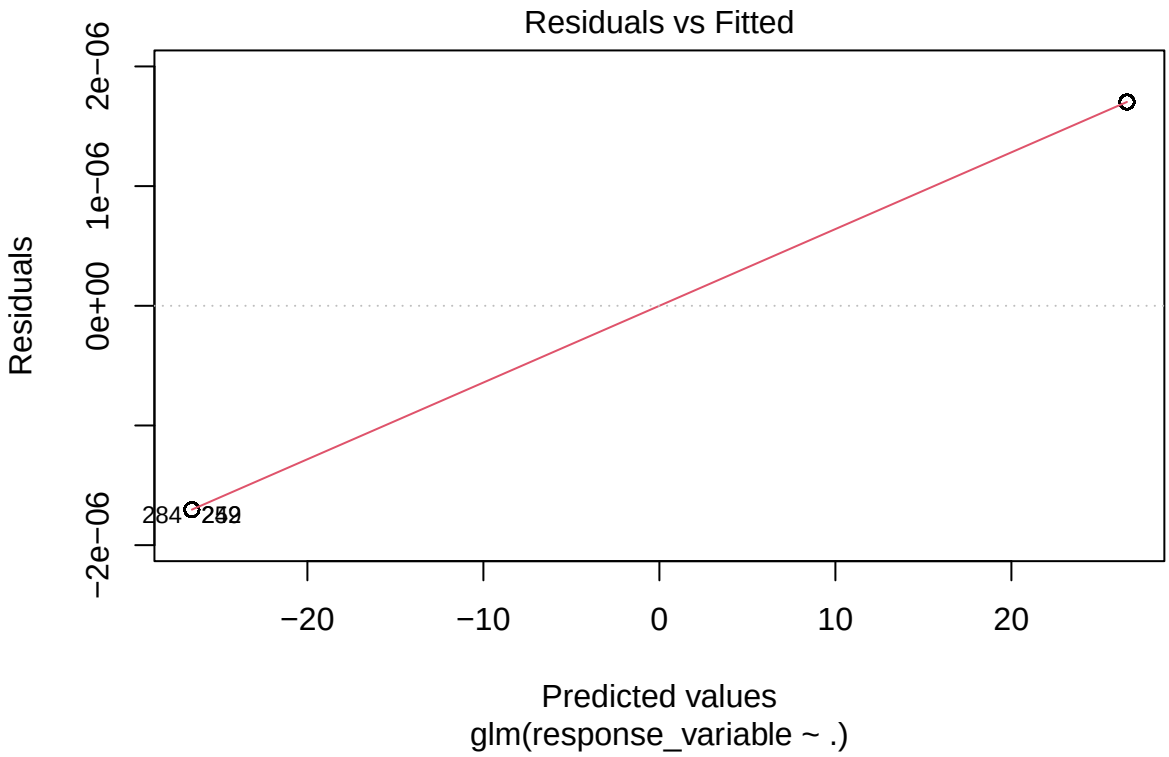
```
selected_columns <- dades_num[, -ncol(dades_num)]
boxplot(selected_columns)
```

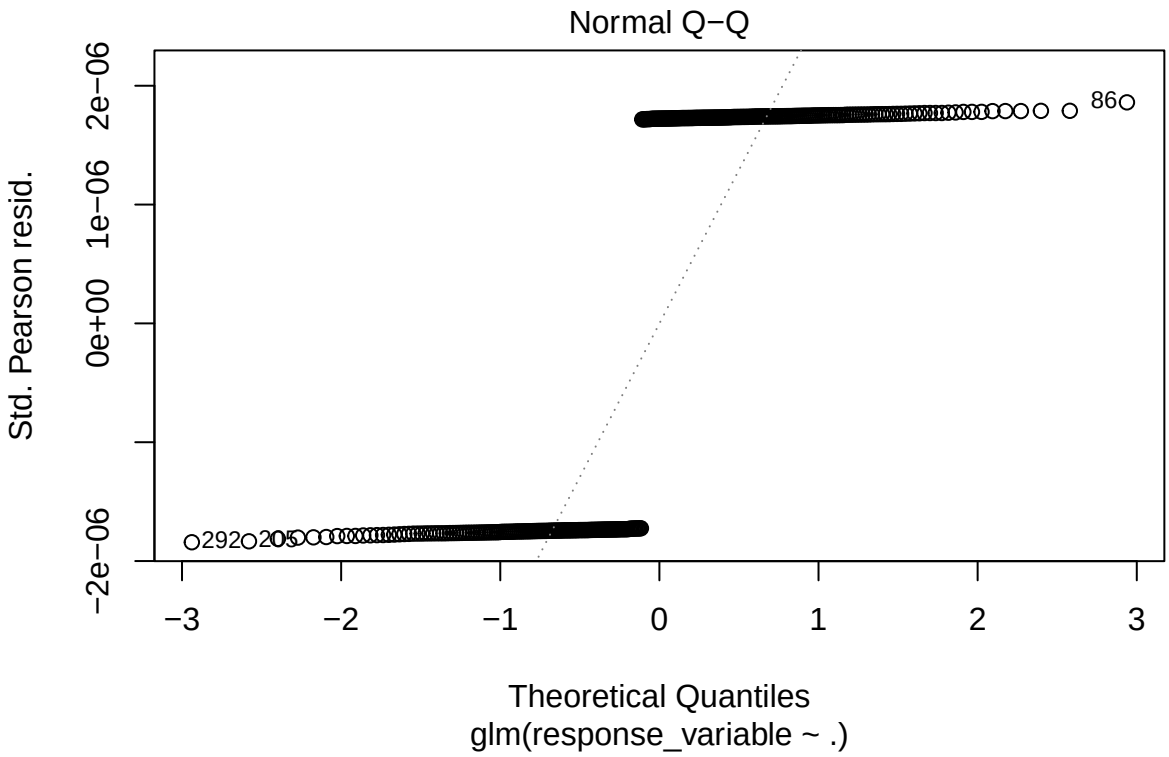


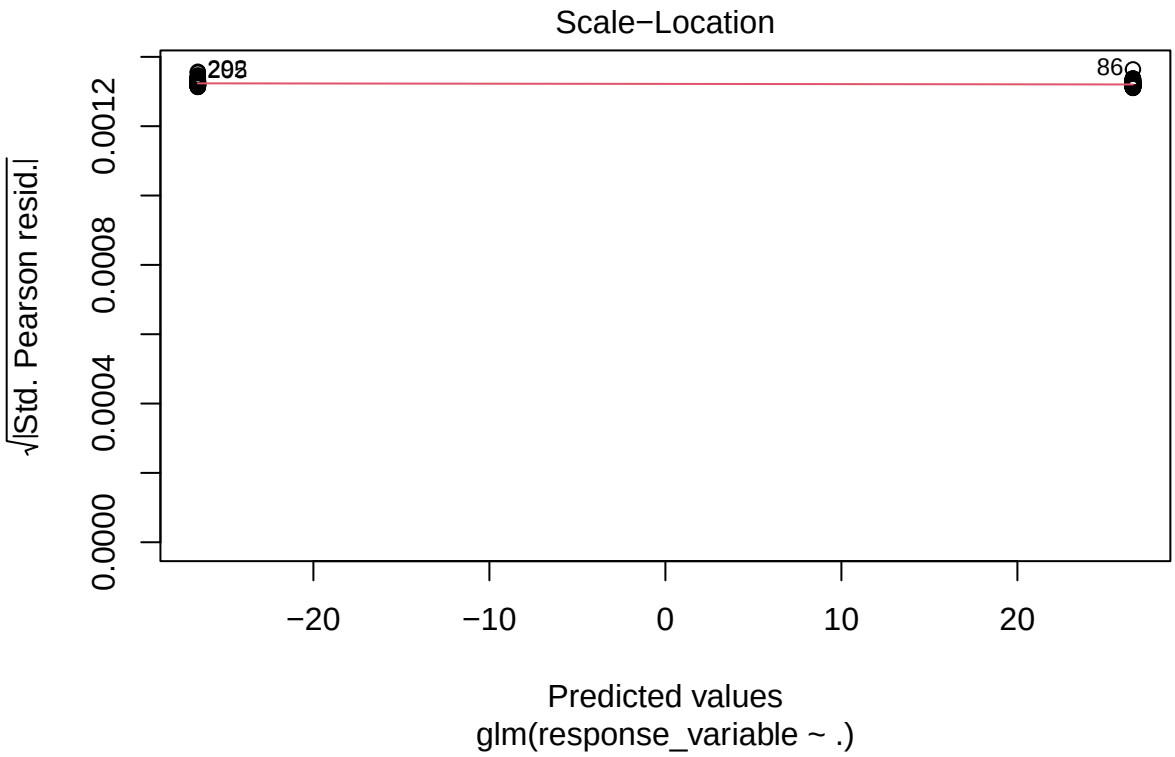

```
cp_column <- dades_num[, "cp"]
cp_colors <- c("red", "green", "blue", "purple")
barplot(cp_column, main = "Barplot of chest pain", xlab = "Observation", ylab = "cp Value", col = cp_col
```

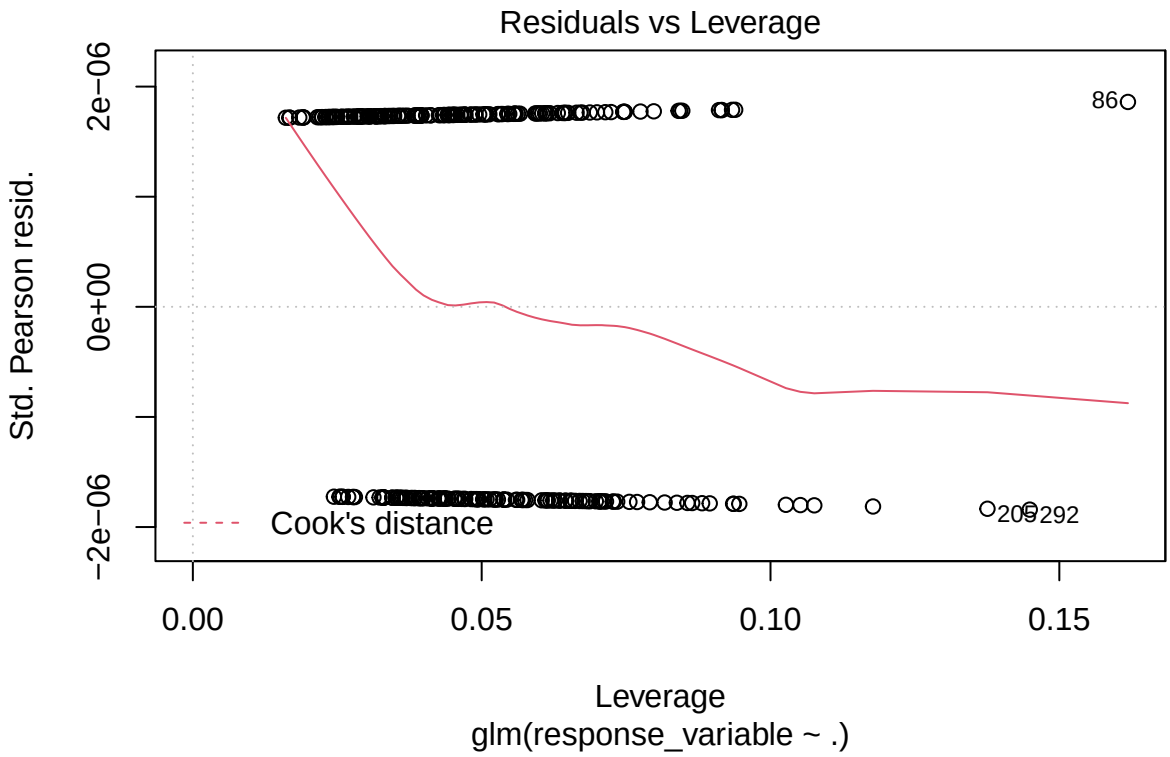

Barplot of chest pain
cp Value
0.0
0.5
1.0
1.5
2.0
2.5
3.0
Observation

```
plot(model)
```


Residuals vs Fitted
Residuals
2e−06
1e−06
0e+00
−2e−06
284
−20
−10
0
10
20
Predicted values
glm(response_variable ~ .)


Normal Q−Q
Std. Pearson resid.
2e−06
1e−06
0e+00
−2e−06
86
292
205
−3
−2
−1
0
1
2
3
Theoretical Quantiles
glm(response_variable ~ .)


Scale−Location
√|Std. Pearson resid.|
0.0000
0.0004
0.0008
0.0012
−20
−10
0
10
20
Predicted values
glm(response_variable ~ .)




Als gràfics de boxplot podem veure els valors atípics que havíem comentat en el punt de neteja de dades.

També hem fet un gràfic on podem veure la distribució de valors dels tipus de dolor de pit.

A més a més hem proporcionat la representació de model de regressió logística que havíem fet en l'apartat anterior.

---

## Resolució del problema

---

En aquest problema hem pogut traure les següents conclusions:

- Les variables utilitzades contribueixen de manera similar a l'explicació de la variable output, sense destacar cap d'elles ni aportar cap informació especialment rellevant cap d'elles.
- Trobar un model que explique el comportament dels atacs al cor amb resultat acceptablement bons no ha sigut possible.
- Tampoc ha hagut correlacions importats entre les variables, per tant, deixant descartada la possibilitat de poder investigar-les.

---

## Codi

---

El codi es pot trobar pujat al repositori de GitHub:
https://github.com/MariaMartinezGil/Practica-2/tree/main

---

## Vídeo

---

El video explicatiu de la pràctica es pot trobar a l'enllaç de Google Drive:
https://drive.google.com/drive/folders/1y06IIinZsgDLK_R3JXjOA8Drx40QAQS6?usp=sharing