Practica 2

Marc González Planes i Maria Sunyer Rigau

LLiberies necessaries

```
if (!require('VIM')) install.packages('VIM');library('VIM')
```

```
## Loading required package: VIM
```

```
## Loading required package: colorspace
```

```
## Loading required package: grid
```

```
## VIM is ready to use.
```

```
## Suggestions and bug-reports can be submitted at: https://github.com/statistikat/VIM/issues
```

```
##
## Attaching package: 'VIM'
```

```
## The following object is masked from 'package:datasets':
##
##     sleep
```

```
if (!require('dplyr')) install.packages('dplyr');library('dplyr')
```

```
## Loading required package: dplyr
```

```
##
## Attaching package: 'dplyr'
```

```
## The following objects are masked from 'package:stats':
##
##     filter, lag
```

```
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
if (!require('missForest')) install.packages('missForest');library('missForest')
```

```
## Loading required package: missForest
```

```
##
## Attaching package: 'missForest'
```

```
## The following object is masked from 'package:VIM':
##
##     nrmse
```

```
if (!require('car')) install.packages('car');library('car')
```

```
## Loading required package: car
```

```
## Loading required package: carData
```

```
##
## Attaching package: 'car'
```

```
## The following object is masked from 'package:dplyr':
##
##     recode
```

```
if (!require('psych')) install.packages('psych');library('psych')
```

```
## Loading required package: psych
```

```
##
## Attaching package: 'psych'
```

```
## The following object is masked from 'package:car':
##
##     logit
```

```
if (!require('ggplot2')) install.packages('ggplot2');library('ggplot2')
```

```
## Loading required package: ggplot2
```

```
##
## Attaching package: 'ggplot2'
```

```
## The following objects are masked from 'package:psych':
##
##     %+%, alpha
```

```
if (!require('caTools')) install.packages('caTools');library('caTools')
```

```
## Loading required package: caTools
```

```
if (!require('e1071')) install.packages('e1071');library('e1071')
```

```
## Loading required package: e1071
```

```
if (!require('caret')) install.packages('caret');library('caret')
```

```
## Loading required package: caret
```

```
## Loading required package: lattice
```

Descripció del Dataset

El dataset utilitzat intenta respondre la pregunta de quins factors influeixen en la salut cardiovascular. Aquest consisteix en les dades mèdiques d'un conjunt de pacients. El dataset utilitzat consisteix en diferents parametres mèdics els quals s'utilitcen per determinar riscos cardiovascolars. El nostre analisis intenta donar resposta a dues preguntes: - Quins són els parametres més i menys rellevants? - Podem predir el risc d'un pacient amb les dades actuals?

Age : Age of the patient

Sex : Sex of the patient

exng: exercise induced angina (1 = yes; 0 = no)

caa: number of major vessels (0-3)

cp : Chest Pain type

Value 1: typical angina

Value 2: atypical angina

Value 3: non-anginal pain

Value 4: asymptomatic

trtbps : resting blood pressure (in mm Hg)

chol : cholestoral in mg/dl fetched via BMI sensor

fbs : (fasting blood sugar > 120 mg/dl) (1 = true; 0 = false)

rest_ecg : resting electrocardiographic results

- Value 0: normal
- Value 1: having ST-T wave abnormality (T wave inversions and/or ST elevation or depression of > 0.05 mV)
- Value 2: showing probable or definite left ventricular hypertrophy by Estes' criteria

thalach : maximum heart rate achieved

target : 0 = less chance of heart attack 1 = more chance of heart attack

Integració i selecció de dades

```
data_heart <- read.csv("./heart.csv")

num_fil <- dim(data_heart)[1]
num_col <- dim(data_heart)[2]

print("El data set te una mida:")
```

```
## [1] "El data set te una mida:"
```

```
sprintf("Files: %d", num_fil)
```

```
## [1] "Files: 303"
```

```
sprintf("Columnes: %d", num_col)
```

```
## [1] "Columnes: 14"
```

Neteja de dades

Les dades contenen zeros o elements buits:

```
# Calculem si tenim dades nulles
colSums(is.na(data_heart))
```

```
##      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
```

```
print("No hi ha valors nulls")
```

```
## [1] "No hi ha valors nulls"
```

En cas de rebre noves dades, aquestes podrien contenir valors nulls o elements buits. Per tant, procedim a "inventar" valors nulls per tal de fer un tractament de dades que en un futur ens permeti acceptar i tractar dades no completes.

KNN per suplir valors

Utilitzarem el metode KNN per suplir aquest valors i no "descartar" les dades.

```
# Enllaç de referencia: https://rpubs.com/harshaash/KNN_imputation
# Enllaç dos: https://www.analyticsvidhya.com/blog/2016/03/tutorial-powerful-packages-imputing-missing-values/

# Introduim valors aleatoris: https://search.r-project.org/CRAN/refmans/missForest/html/prodNA.html
data_nulls <- data_heart

# 1% de valors nulls
data_nulls <- prodNA(data_nulls, noNA = 0.01)

print("Data amb valors nulls")
```

```
## [1] "Data amb valors nulls"
```

```
colSums(is.na(data_nulls))
```

```
##      age      sex       cp   trtbps     chol      fbs  restecg thalachh
##        5        6        4        1        1        1        4        6
##     exng  oldpeak      slp      caa    thall   output
##        2        2        1        2        4        3
```

```
# Realitzem els calculs de KNN
data_knn_filled <- kNN(data_nulls, variable = names(data_nulls), k = 10)
```

```
## Warning in `[<-.data.table`(`*tmp*`, indexNA2s[, variable[j]], variable[j], :
## 45.500000 (type 'double') at RHS position 2 truncated (precision lost) when
## assigning to type 'integer' (column 1 named 'age')
```

```
## Warning in `[<-.data.table`(`*tmp*`, indexNA2s[, variable[j]], variable[j], :
## 0.500000 (type 'double') at RHS position 1 truncated (precision lost) when
## assigning to type 'integer' (column 3 named 'cp')
```

```
## Warning in `[<-.data.table`(`*tmp*`, indexNA2s[, variable[j]], variable[j], :
## 270.500000 (type 'double') at RHS position 1 truncated (precision lost) when
## assigning to type 'integer' (column 5 named 'chol')
```

```
## Warning in `[<-.data.table`(`*tmp*`, indexNA2s[, variable[j]], variable[j], :
## 157.500000 (type 'double') at RHS position 1 truncated (precision lost) when
## assigning to type 'integer' (column 8 named 'thalachh')
```

```
colSums(is.na(data_knn_filled))
```

```
##          age          sex           cp       trtbps         chol          fbs
##            0            0            0            0            0            0
##      restecg     thalachh         exng      oldpeak          slp          caa
##            0            0            0            0            0            0
##        thall       output      age_imp      sex_imp       cp_imp   trtbps_imp
##            0            0            0            0            0            0
##     chol_imp      fbs_imp  restecg_imp thalachh_imp     exng_imp  oldpeak_imp
##            0            0            0            0            0            0
##      slp_imp      caa_imp    thall_imp   output_imp
##            0            0            0            0
```

```
# Observem que data_knn_filled augmenta les variables a 28.
# Ens quedem amb les dades utils, les que coincideixen amb el dataframe inicial
data_knn_filtred <- subset(data_knn_filled, select = names(data_nulls))
summary(data_knn_filtred)
```

```
##       age             sex               cp             trtbps     
##  Min.   :29.00   Min.   :0.0000   Min.   :0.0000   Min.   : 94.0  
##  1st Qu.:47.00   1st Qu.:0.0000   1st Qu.:0.0000   1st Qu.:120.0  
##  Median :55.00   Median :1.0000   Median :1.0000   Median :130.0  
##  Mean   :54.31   Mean   :0.6898   Mean   :0.9538   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  
##       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.:135.0  
##  Median :240.0   Median :0.0000   Median :1.0000   Median :153.0  
##  Mean   :246.2   Mean   :0.1485   Mean   :0.5215   Mean   :149.8  
##  3rd Qu.:274.0   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.   :0.000   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   :0.3267   Mean   :1.032   Mean   :1.396   Mean   :0.7327  
##  3rd Qu.:1.0000   3rd Qu.:1.600   3rd Qu.:2.000   3rd Qu.:1.0000  
##  Max.   :1.0000   Max.   :6.200   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
```

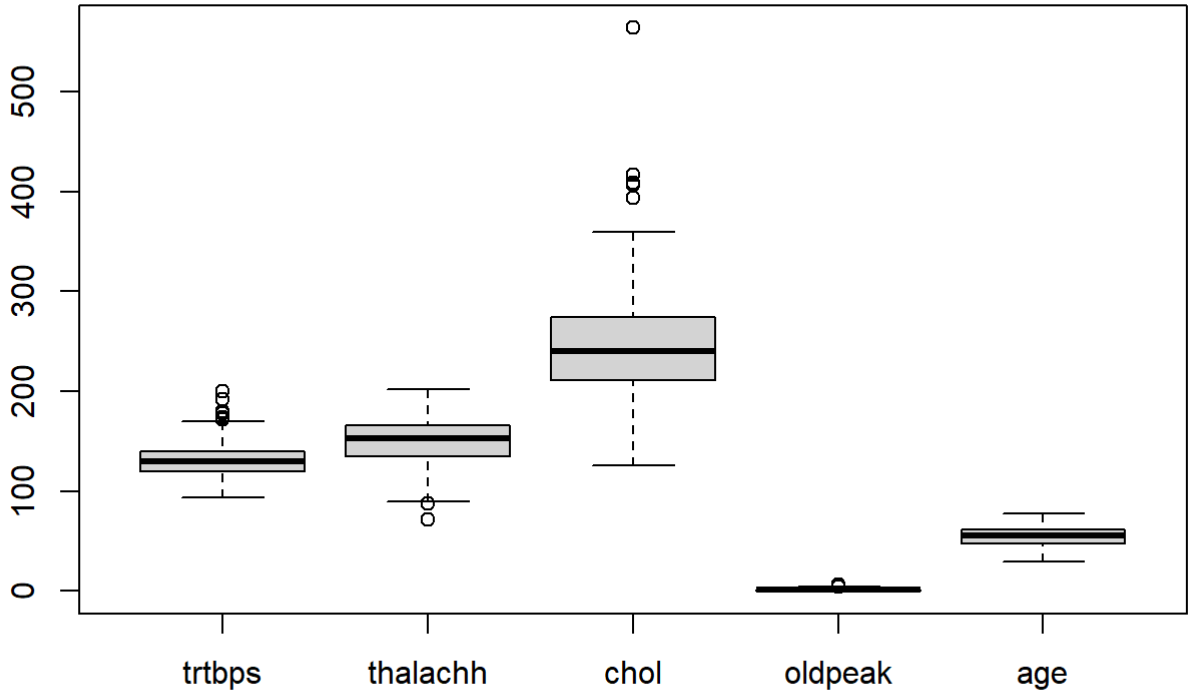
Identifica i gestiona valors extrems

Per trobar els valors extrems, hem considerat que quan un valor es troba allunyat 3 desviacions estàndard respecte a la mitjana del conjunt és un outlier. Per trobar-los hem representant les dades en boxplots. Hem realitzat aquesta comprovació a les variables numeriques seguent:

```
data_numerical <- as.data.frame(data_knn_filtred %>%
                  select("trtbps","thalachh","chol","oldpeak", "age"))
```

BoxPlot:

```
boxplot <- boxplot(data_numerical)
```

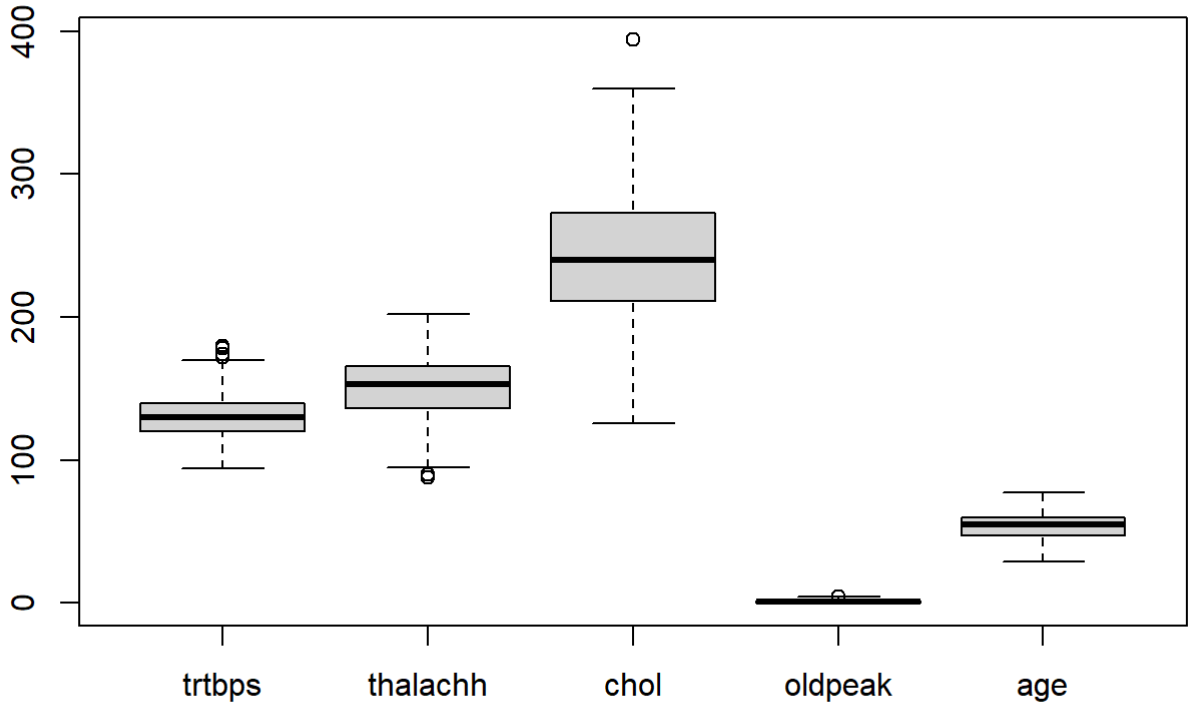


Com es pot veure, les columnes oldpeak, chol, thalachh i trtbps tenen possibles outliers. Per eliminar-los, he creat la següents funcions per seleccionar aquells valor 3 desviacions standars allunyats:

```
outliers = function(x) {

  standardD <- sd(x)
  mitjana <- mean(x)

 #Upper Range
 upper_limit = mitjana + 3*standardD
 #Lower Range
 lower_limit = mitjana - 3*standardD

 x > upper_limit | x < lower_limit
}
# remove the outliers
elimina_outliers <- function(df_outliers, cols = names(df_outliers)) {
  for (col in cols) {
    df_outliers<- df_outliers[!outliers(df_outliers[[col]]),]
  }
  df_outliers
}
```

Apliquem aquesta funció i mostrem els nous boxplots, a on es pot veure la disminució de outliers:

```
data_heart<-elimina_outliers(data_knn_filtred,c("trtbps","oldpeak" ,"thalachh", "chol"))
data_numerical <- as.data.frame(data_heart %>%
                  select("trtbps","thalachh","chol","oldpeak", "age"))
boxplot <- boxplot(data_numerical)
```



#

Anàlisi de les dades

Selecció de dades

Obtenim les dades estadístiques bàsiques del dataset

```
summary(data_heart)
```

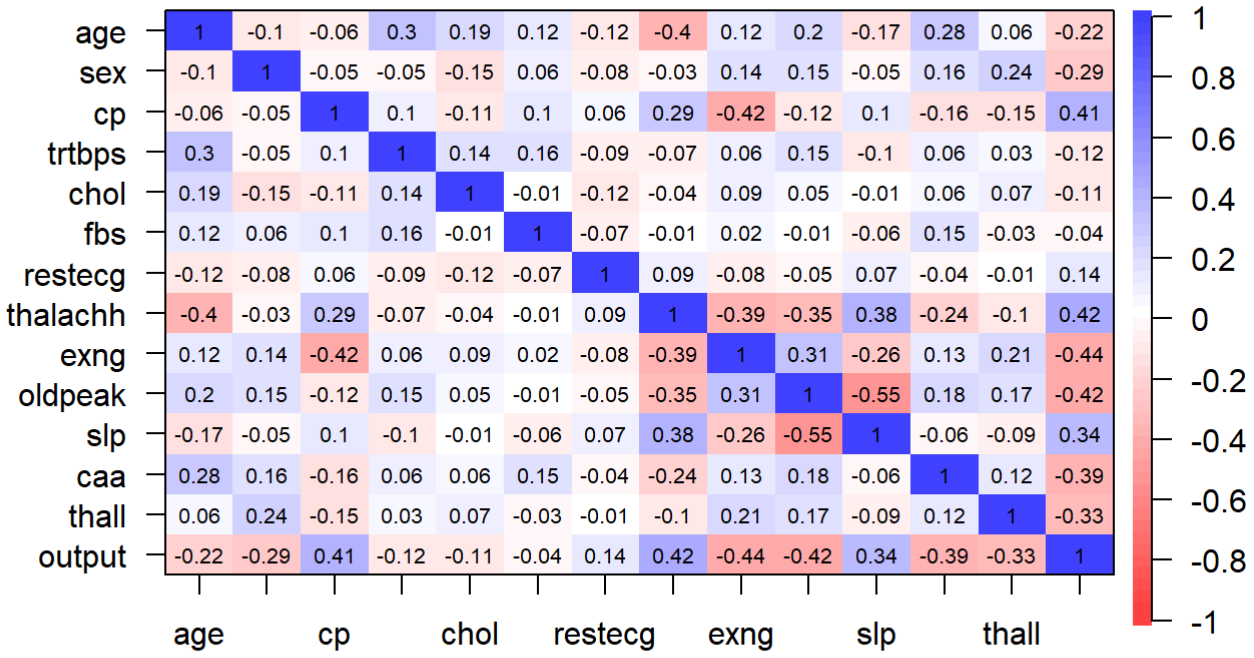
```
##       age             sex               cp            trtbps     
##  Min.   :29.00   Min.   :0.0000   Min.   :0.000   Min.   : 94.0  
##  1st Qu.:47.00   1st Qu.:0.0000   1st Qu.:0.000   1st Qu.:120.0  
##  Median :55.00   Median :1.0000   Median :1.000   Median :130.0  
##  Mean   :54.12   Mean   :0.6973   Mean   :0.966   Mean   :131.1  
##  3rd Qu.:60.00   3rd Qu.:1.0000   3rd Qu.:2.000   3rd Qu.:140.0  
##  Max.   :77.00   Max.   :1.0000   Max.   :3.000   Max.   :180.0  
##       chol            fbs             restecg          thalachh    
##  Min.   :126.0   Min.   :0.0000   Min.   :0.0000   Min.   : 88    
##  1st Qu.:211.0   1st Qu.:0.0000   1st Qu.:0.0000   1st Qu.:136    
##  Median :240.0   Median :0.0000   Median :1.0000   Median :153    
##  Mean   :243.6   Mean   :0.1463   Mean   :0.5306   Mean   :150    
##  3rd Qu.:272.5   3rd Qu.:0.0000   3rd Qu.:1.0000   3rd Qu.:166    
##  Max.   :394.0   Max.   :1.0000   Max.   :2.0000   Max.   :202    
##       exng           oldpeak            slp              caa        
##  Min.   :0.0000   Min.   :0.0000   Min.   :0.000   Min.   :0.0000  
##  1st Qu.:0.0000   1st Qu.:0.0000   1st Qu.:1.000   1st Qu.:0.0000  
##  Median :0.0000   Median :0.6000   Median :1.000   Median :0.0000  
##  Mean   :0.3265   Mean   :0.9786   Mean   :1.412   Mean   :0.7143  
##  3rd Qu.:1.0000   3rd Qu.:1.6000   3rd Qu.:2.000   3rd Qu.:1.0000  
##  Max.   :1.0000   Max.   :4.4000   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   :2.299   Mean   :0.5544  
##  3rd Qu.:3.000   3rd Qu.:1.0000  
##  Max.   :3.000   Max.   :1.0000
```

```
# Mostrem els primers 5 valors
head(data_heart, n = 5)
```

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

Aprofitarem que totes les variables estan introduides com a Integers per tal d'obtenir una matriu de correlació i veure quins atributs tenen major correlació amb el resultat i per tant, semblen més rellevants a analitzar.

```
# Enllaç: https://sparkbyexamples.com/r-programming/r-select-function-from-dplyr/

# Analisis de correlacio, quines dades tenen mes relacio entre elles i el output?

corr_df <- cor(data_heart)

corPlot(corr_df, main = "Matriu de correlacio")
```

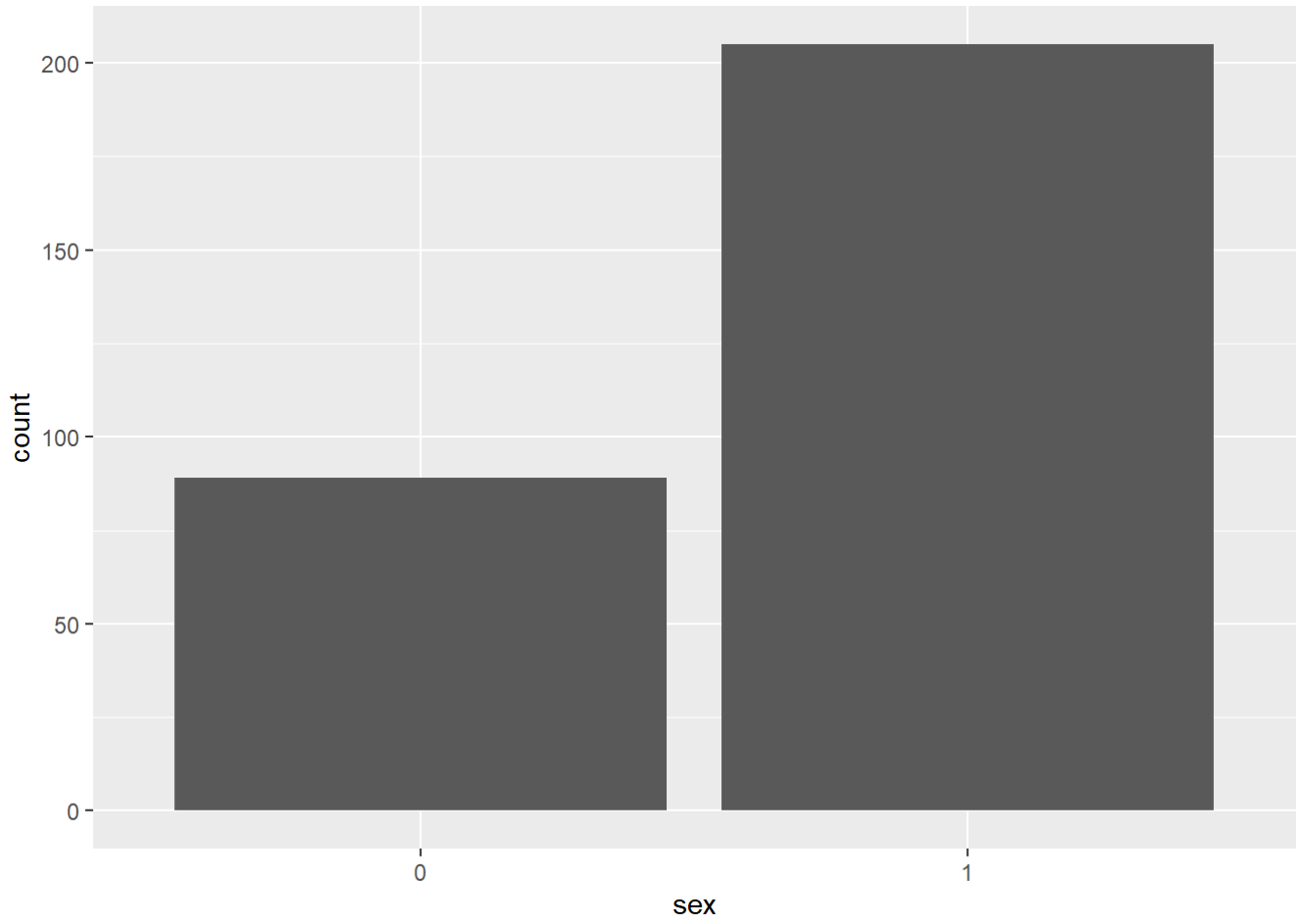


Observem en la matriu de correlació que: - La variable “sex” no té força correlació amb cap dels atributs, sembla que no es un factor rellevant. Tot i així, sorpren que a nivell “general” la percepció de la població és que hi ha més homes amb aquest problema. - La variable “edad” sembla no influir gairé en les variables ‘categoriques’. - La variable “output” esta correlativament influenciada per “cp” i “thalachh”.

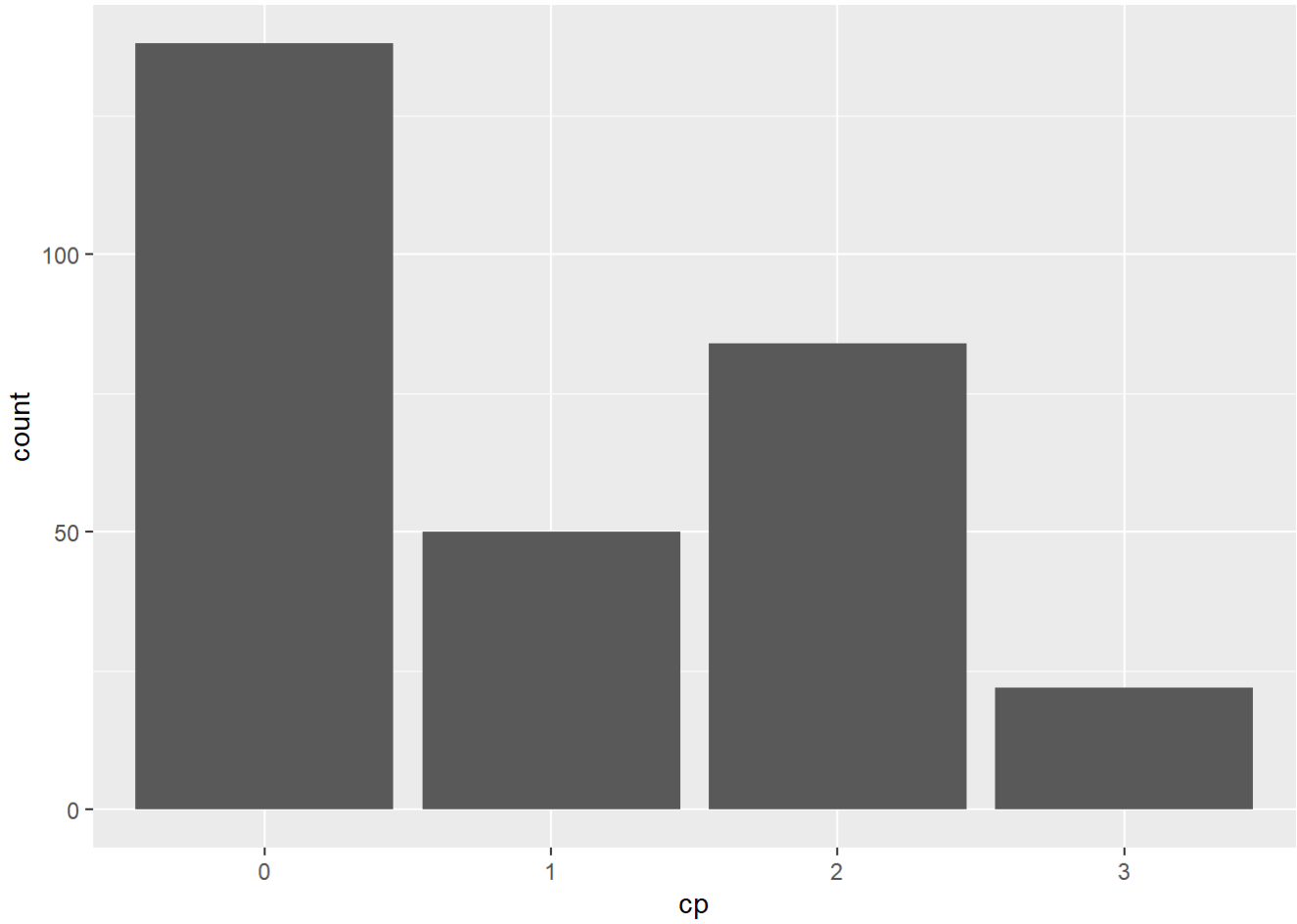
Per tant, semblem tenir algunes hipotesis que podem comprobar, ¿Hi ha més problemes cardiovasculars entre homes o entre dones?

Seguim amb l’anàlisis preliminar per visualitzar gràficament les variables categoriques i veure si hi ha relació o no amb l’output.

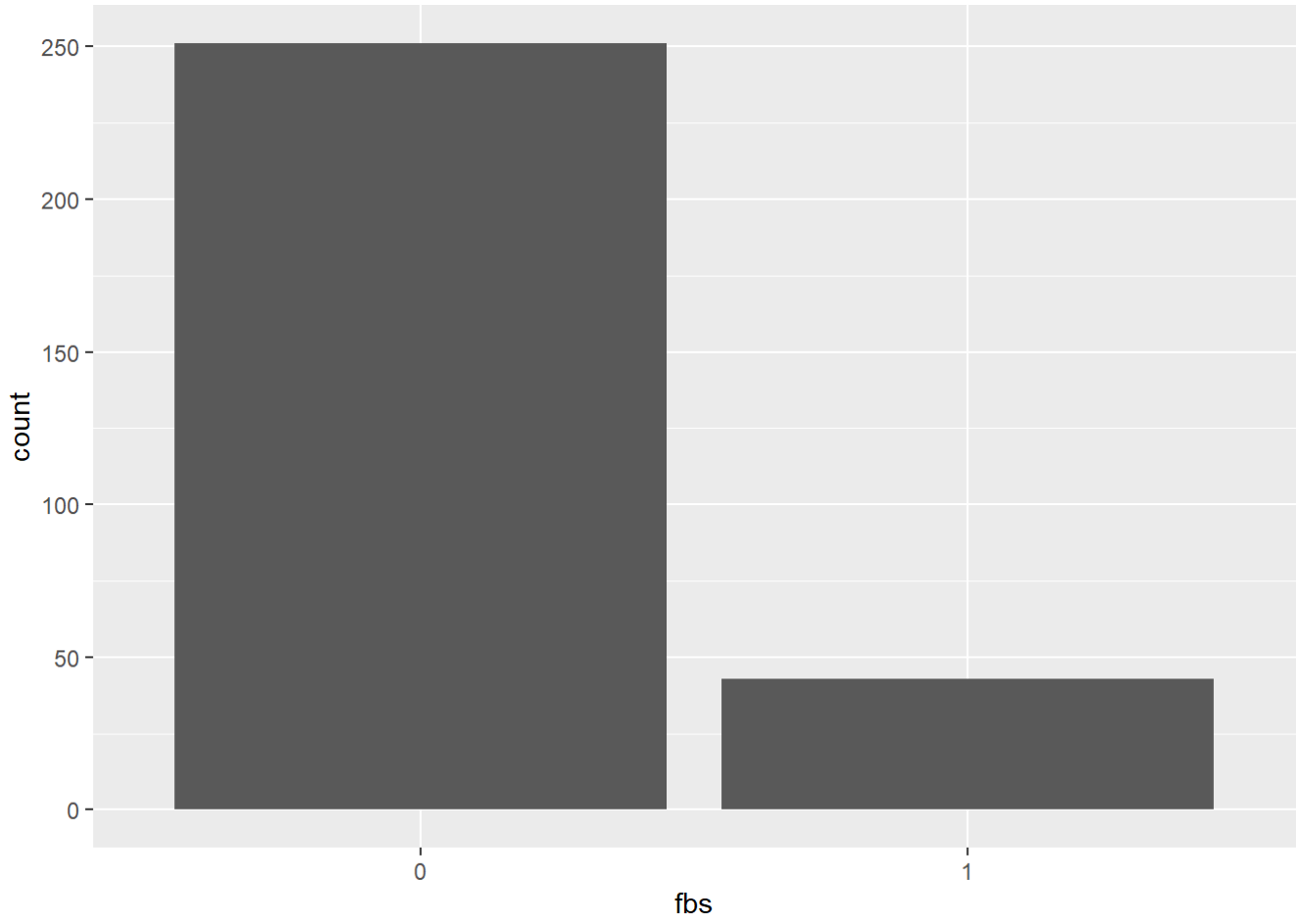
```
# Reestablim les dades ara sense els outliers.
data_numerical <- as.data.frame(data_heart %>%
                  select("trtbps","thalachh","chol","oldpeak", "age"))

# Obtenim les variables categoriques i les transformem al tipus necesari
# Link : https://gist.github.com/ramhiser/93fe37be439c480dc26c4bed8aab03dd

data_categorical <- as.data.frame(data_heart %>%
                  select("sex","cp","fbs","restecg", "exng", "slp", "caa", "thall", "outpu
t"))


data_categorical <- data_categorical %>% mutate_if(is.numeric, as.factor)

# Representem graficament les diferents variables categoriques
ggplot(data = data_categorical, aes(x = sex))+geom_bar()
```



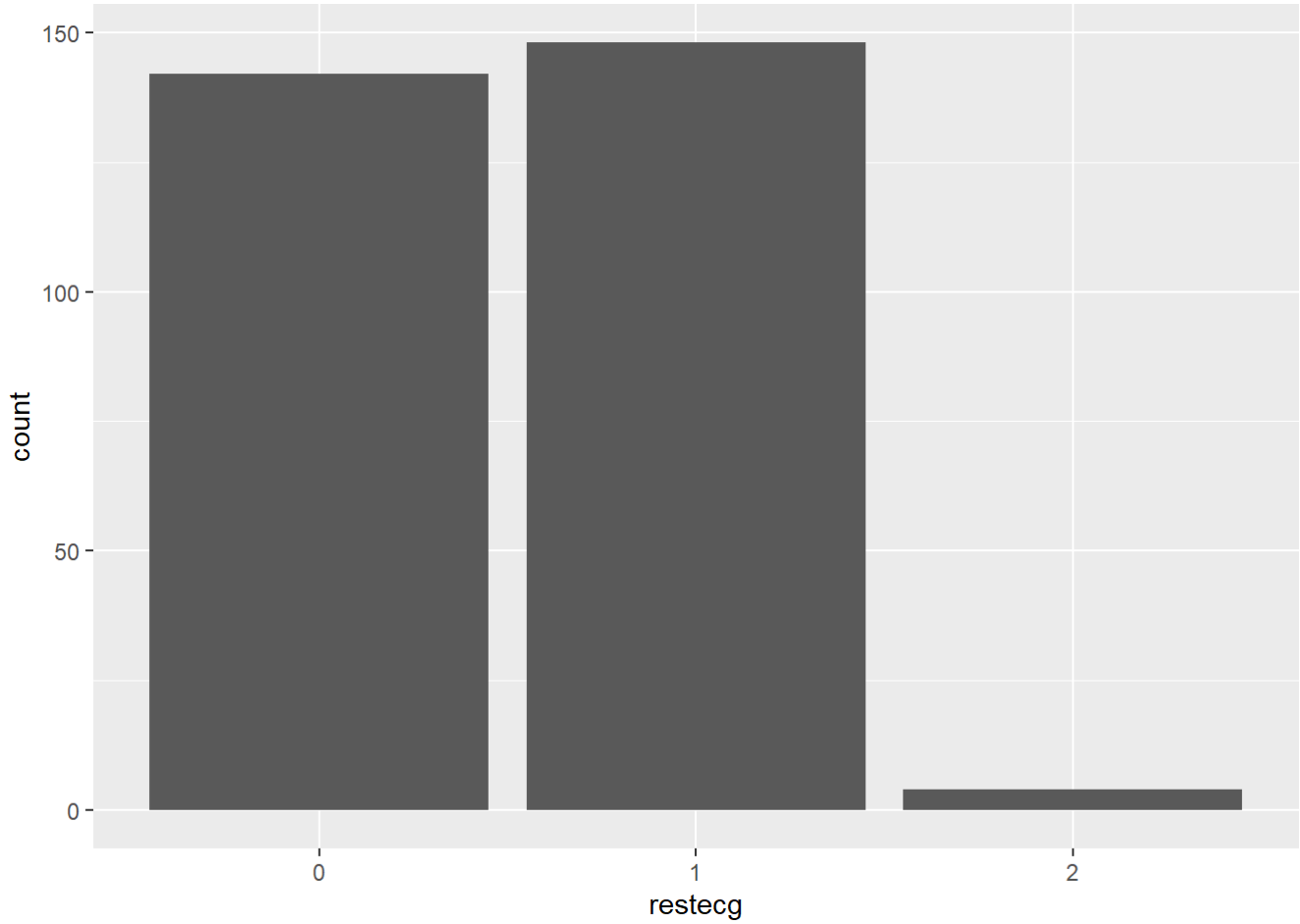

```
ggplot(data = data_categorical, aes(x = cp))+geom_bar()
```



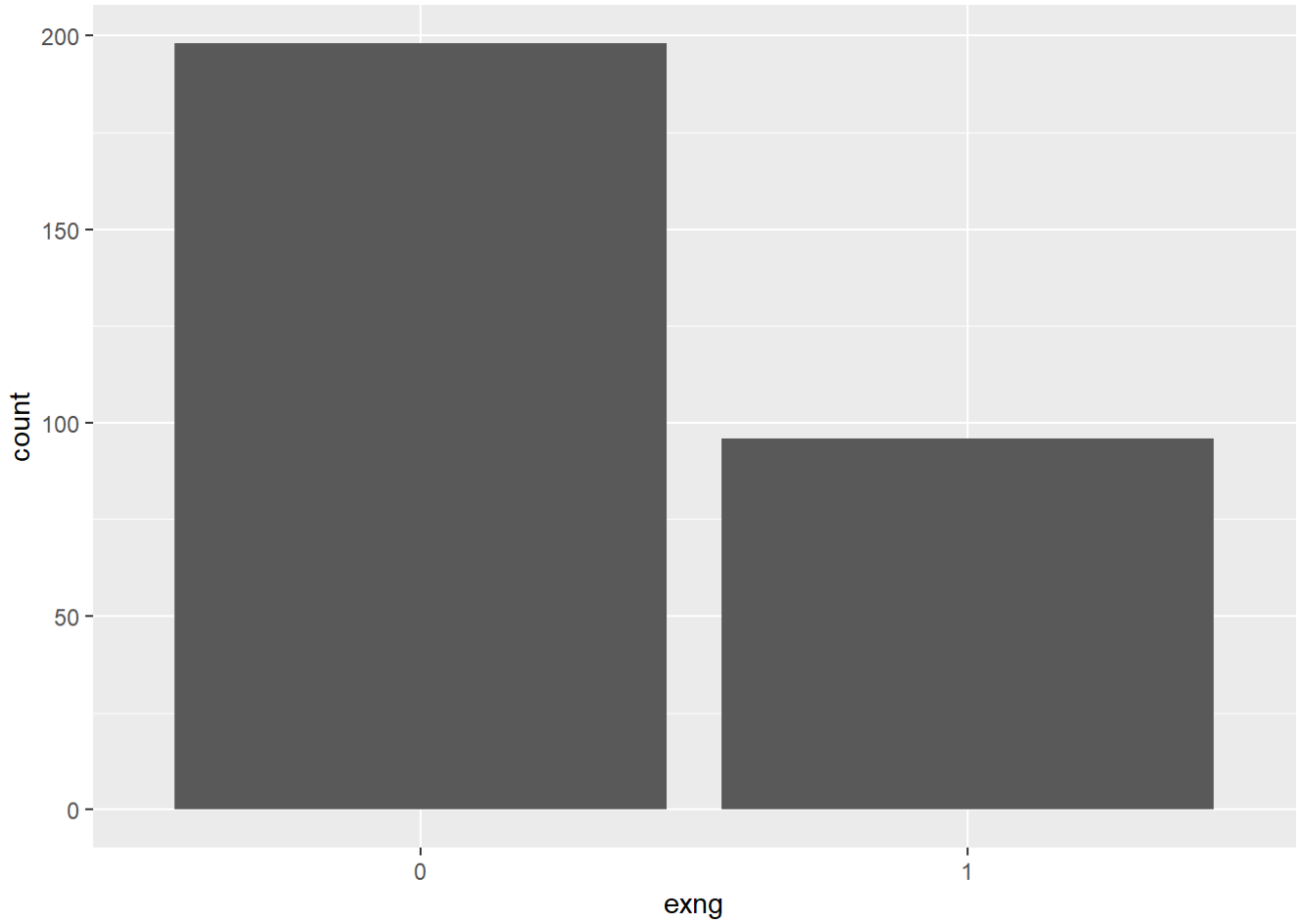
```
ggplot(data = data_categorical, aes(x = fbs))+geom_bar()
```



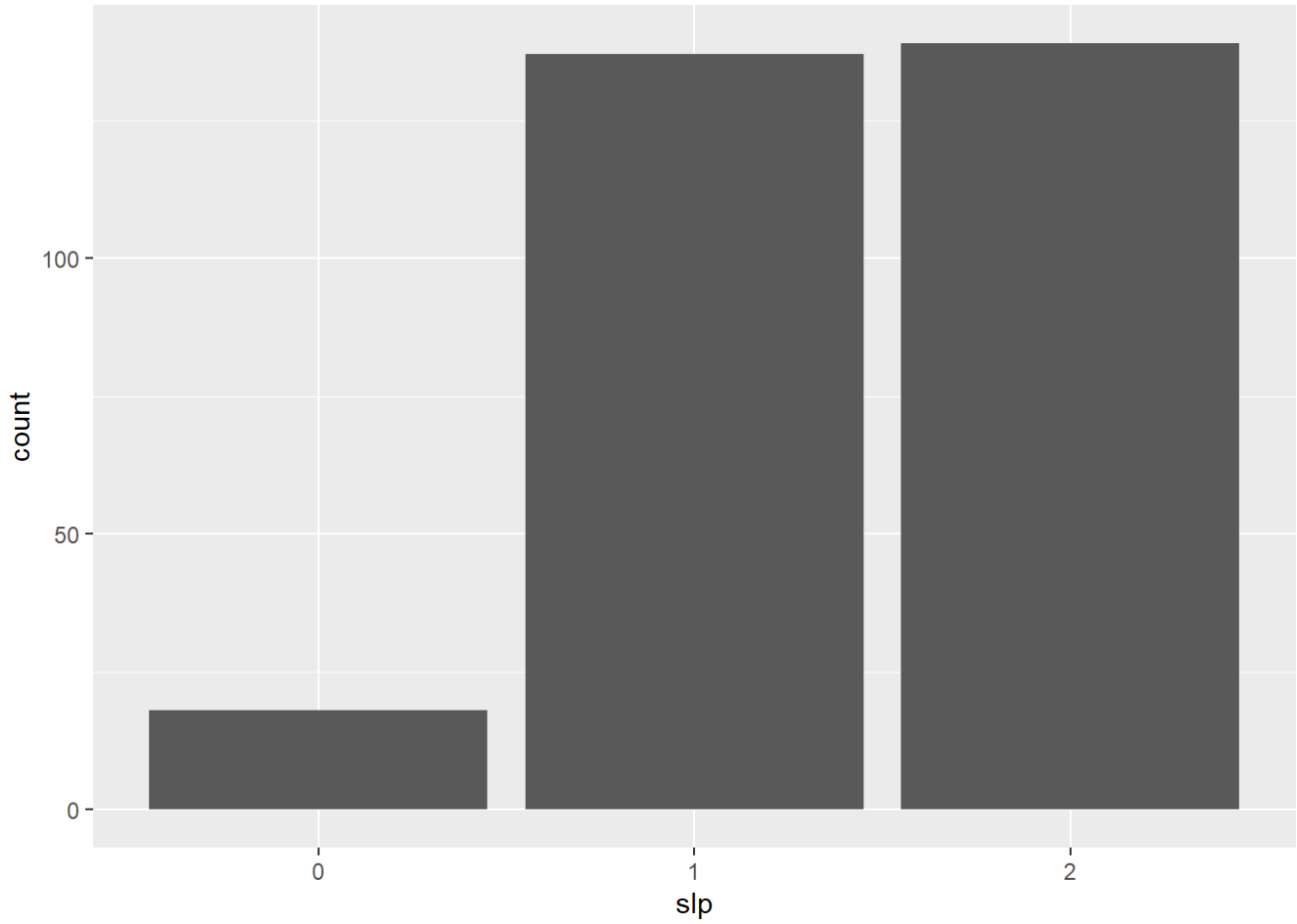

```
ggplot(data = data_categorical, aes(x = restecg))+geom_bar()
```



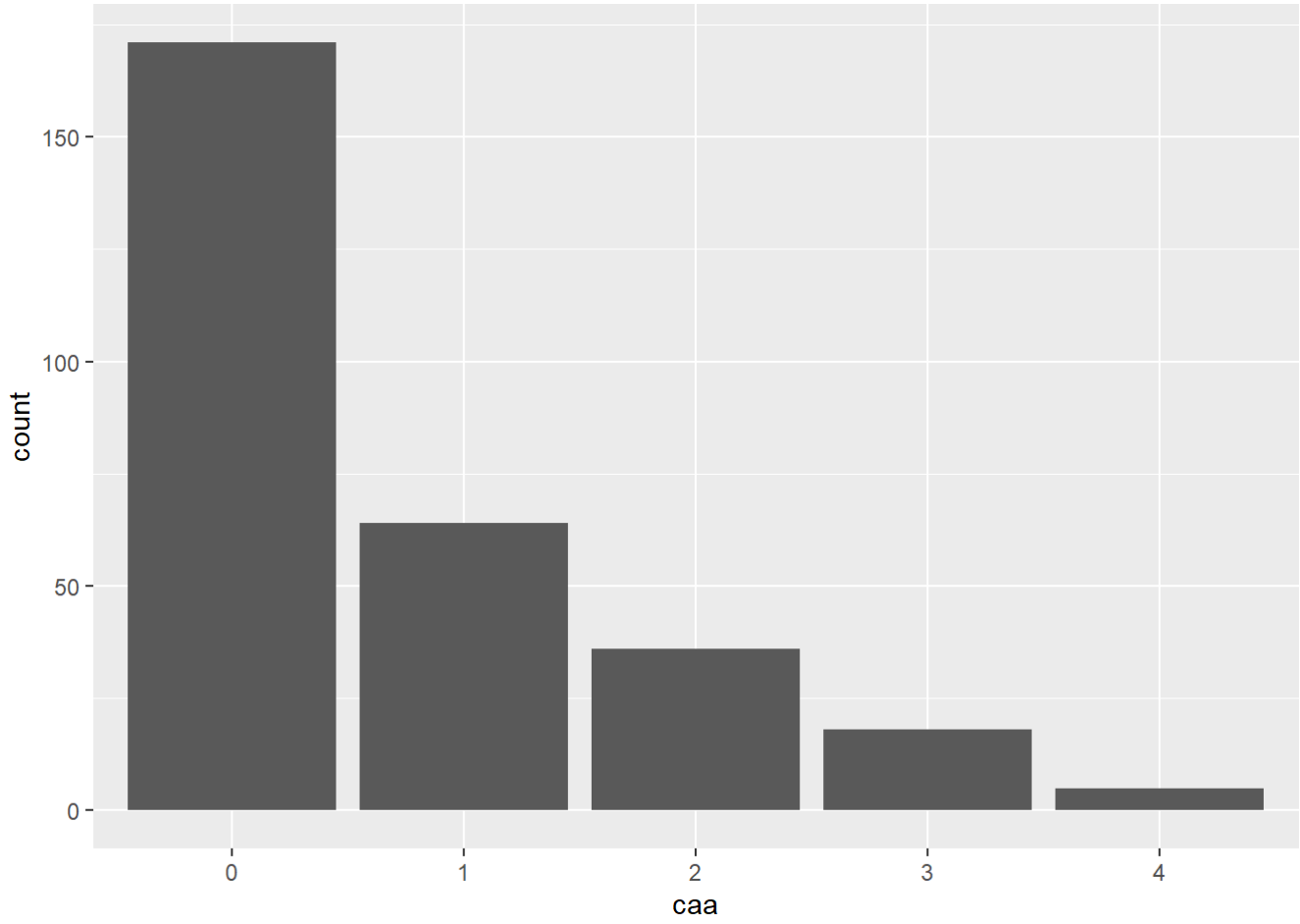

```
ggplot(data = data_categorical, aes(x = exng))+geom_bar()
```



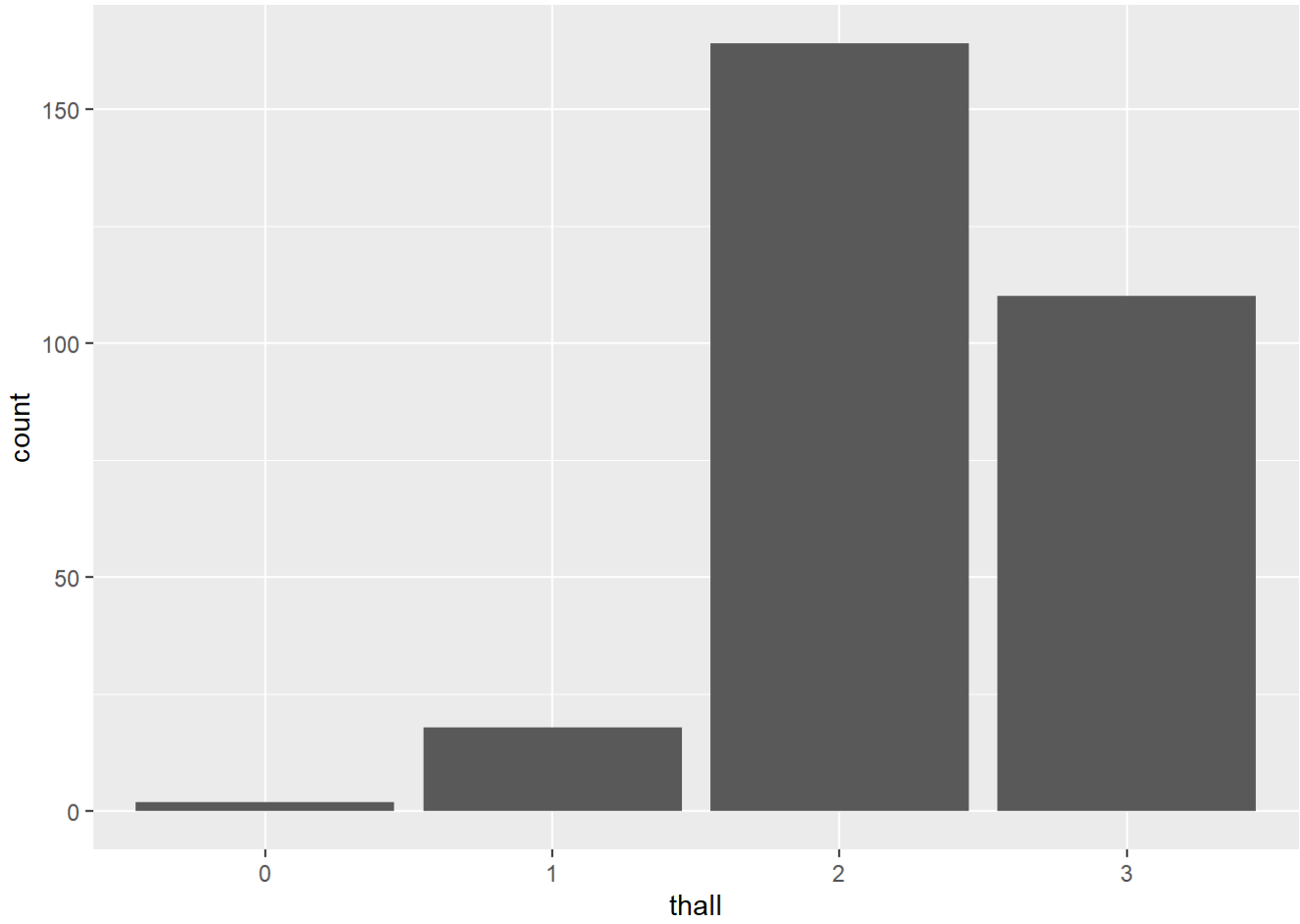
```
ggplot(data = data_categorical, aes(x = slp))+geom_bar()
```



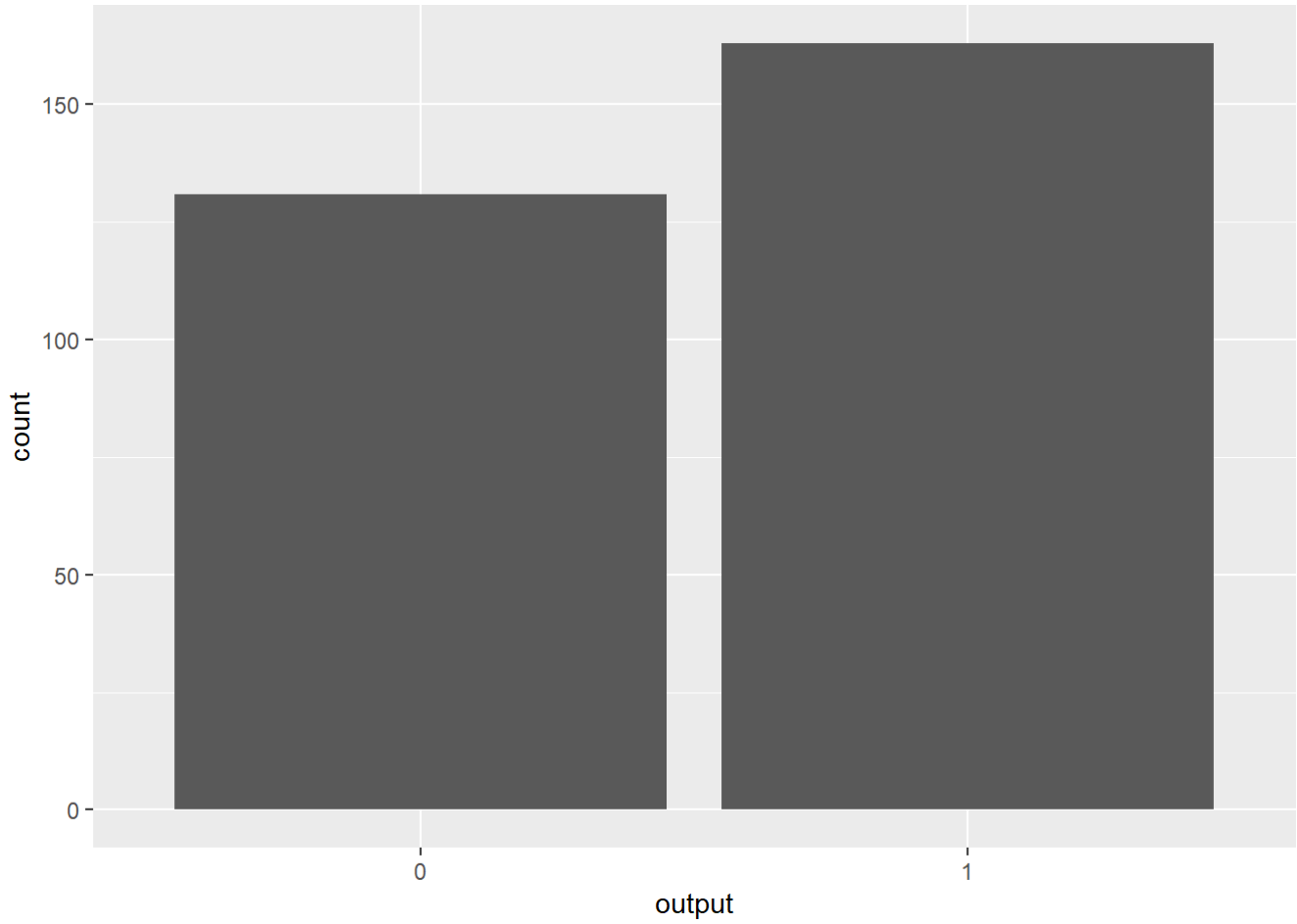

```
ggplot(data = data_categorical, aes(x = caa))+geom_bar()
```



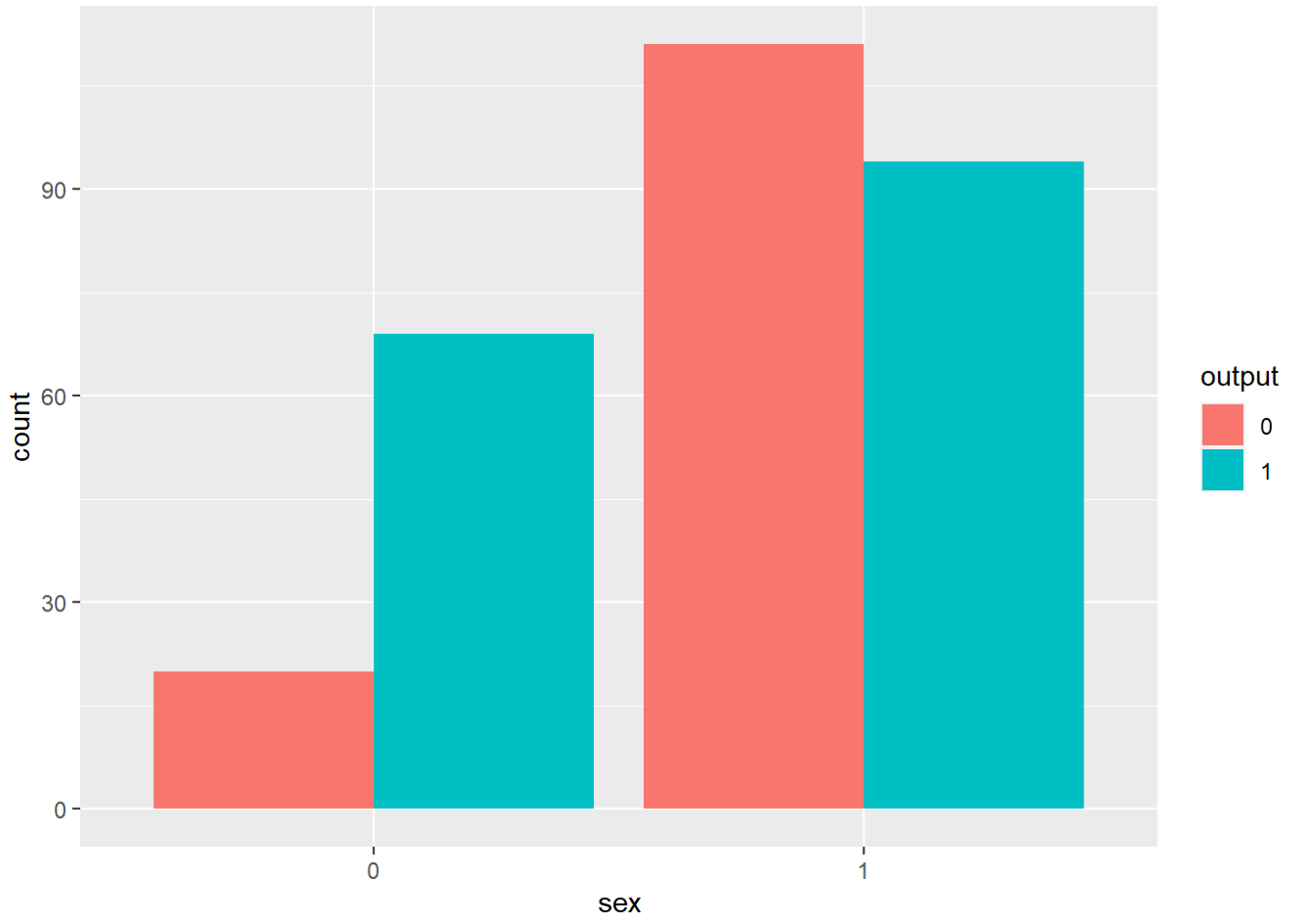

```
ggplot(data = data_categorical, aes(x = thall))+geom_bar()
```



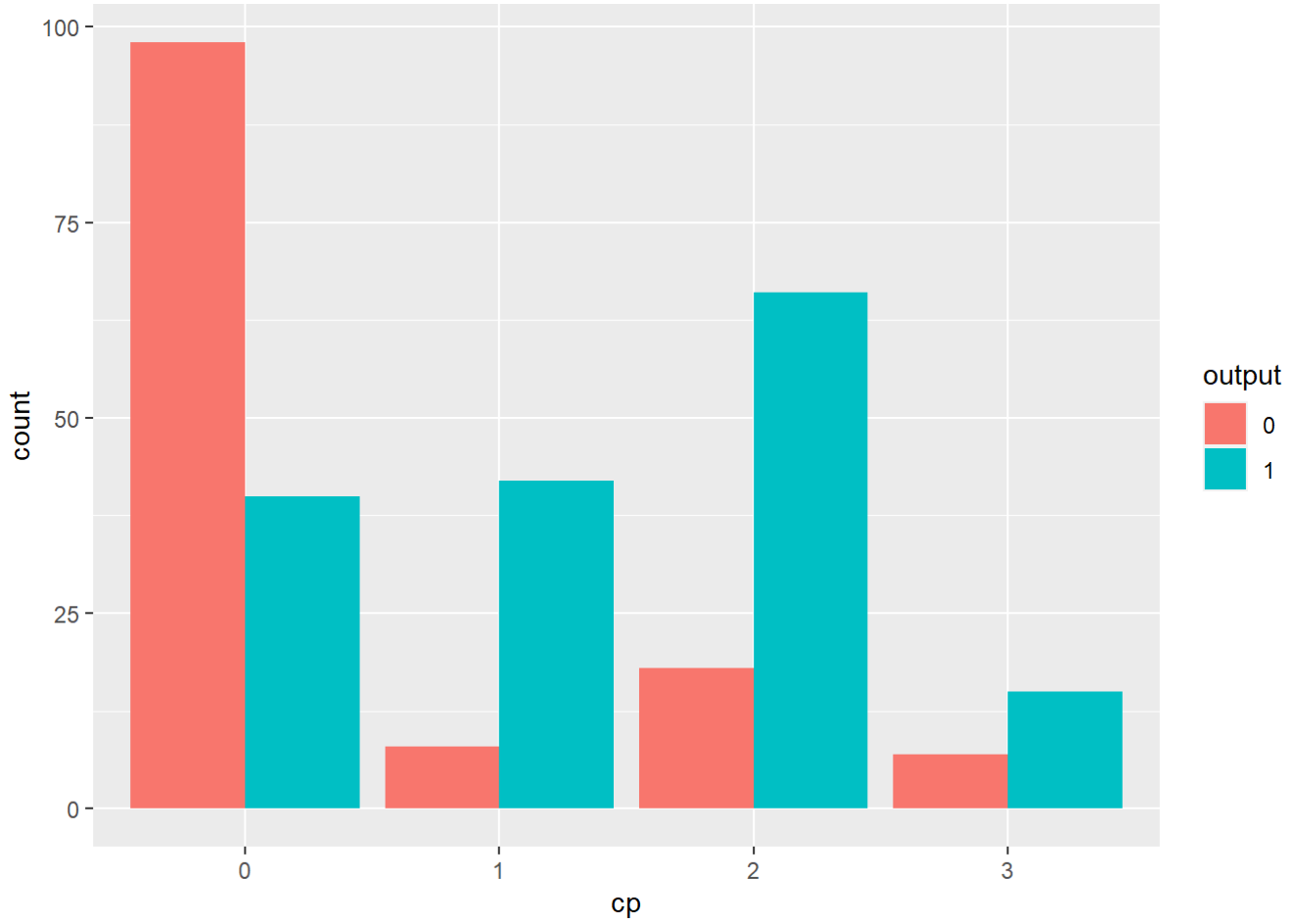
```
ggplot(data = data_categorical, aes(x = output))+geom_bar()
```



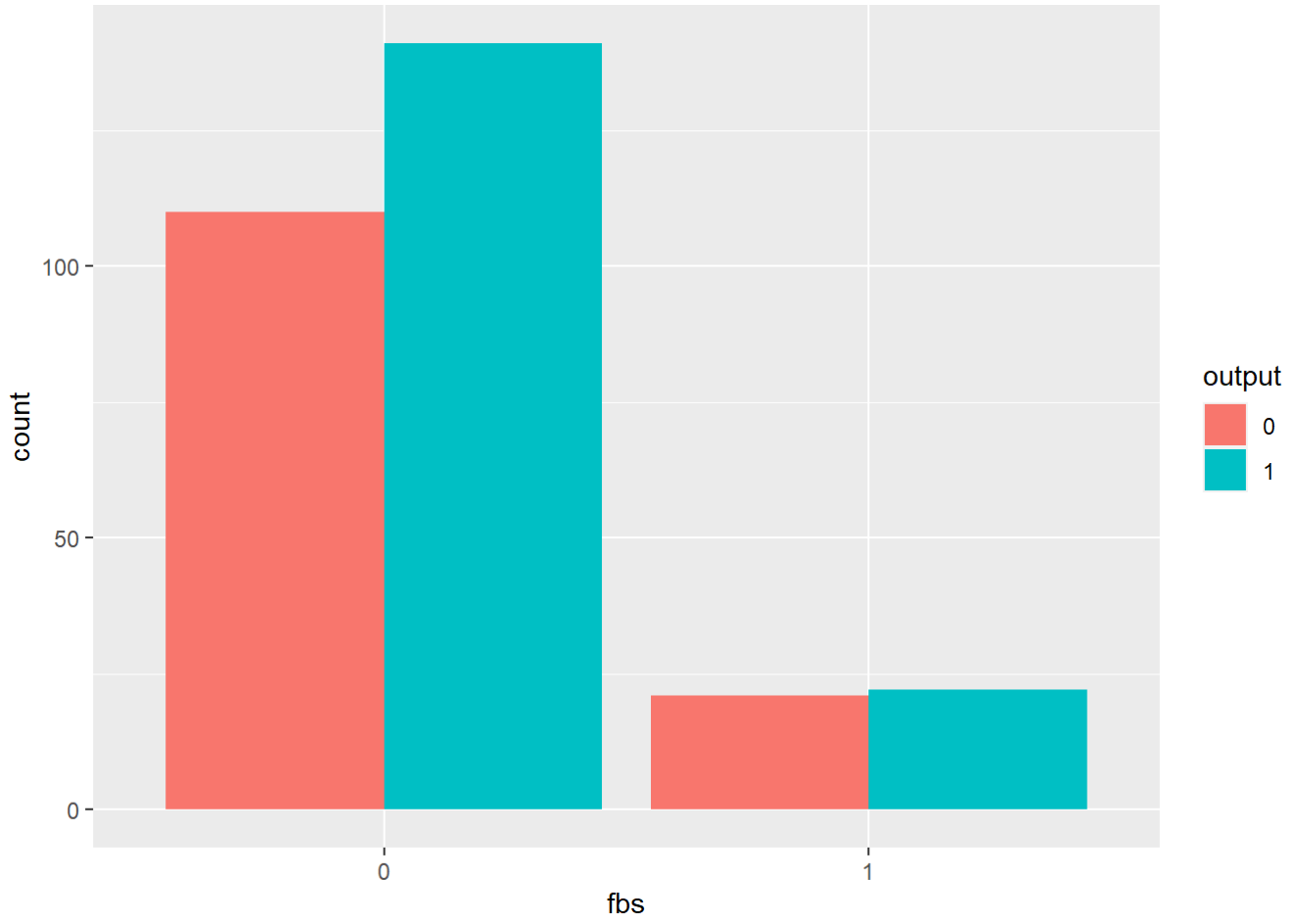

```
# Representem les variables dividides per genere per veure si hi ha diferencies en les variables que sortissin a la corrMatrix (?):
# Error: https://stackoverflow.com/questions/24895575/ggplot2-bar-plot-with-two-categorical-variables
ggplot(data = data_categorical, aes(x = sex, after_stat(count) ))+geom_bar(aes(fill = output), position = "dodge")
```



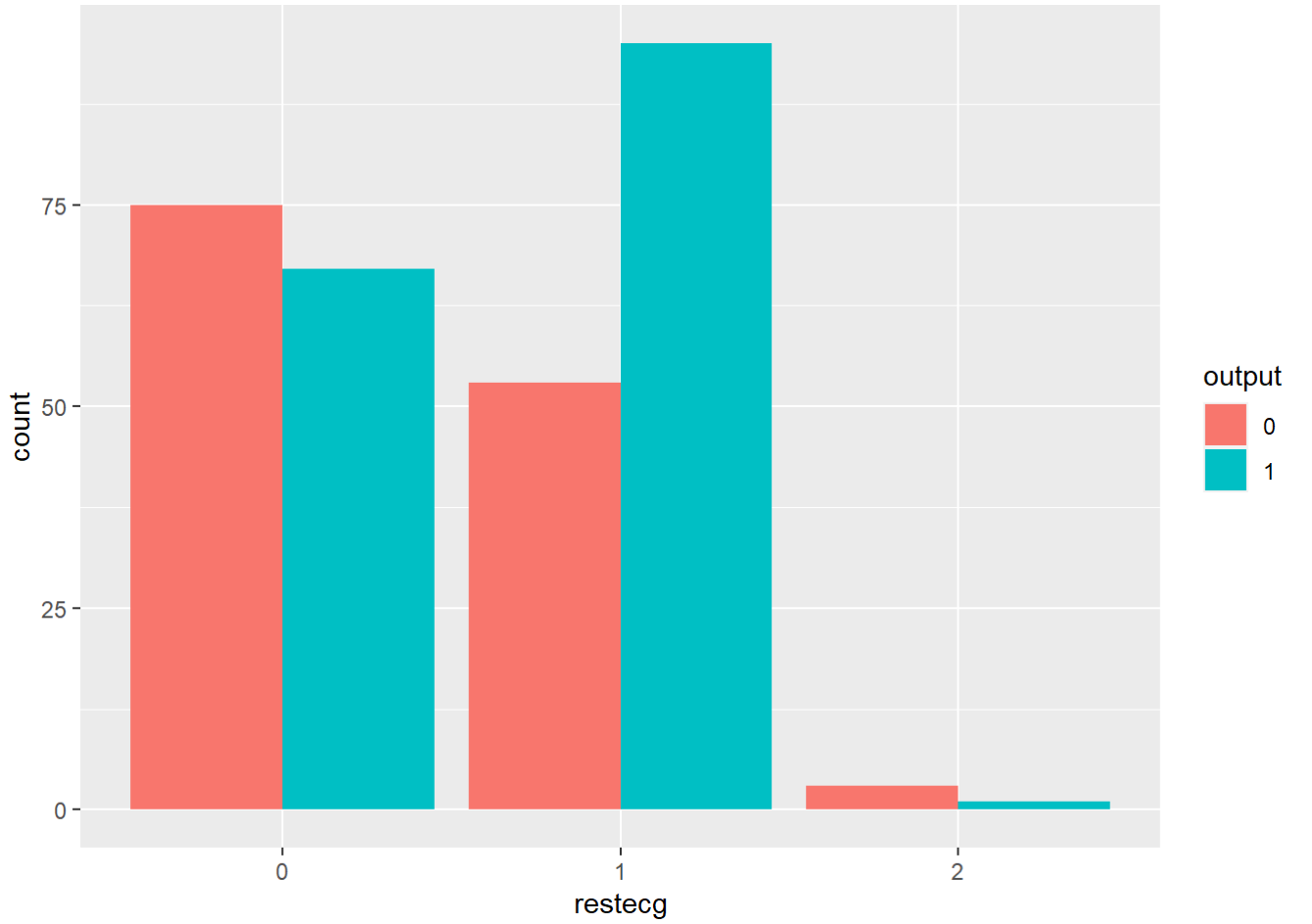

```
ggplot(data = data_categorical, aes(x = cp, after_stat(count) ))+geom_bar(aes(fill = output),
position = "dodge")
```



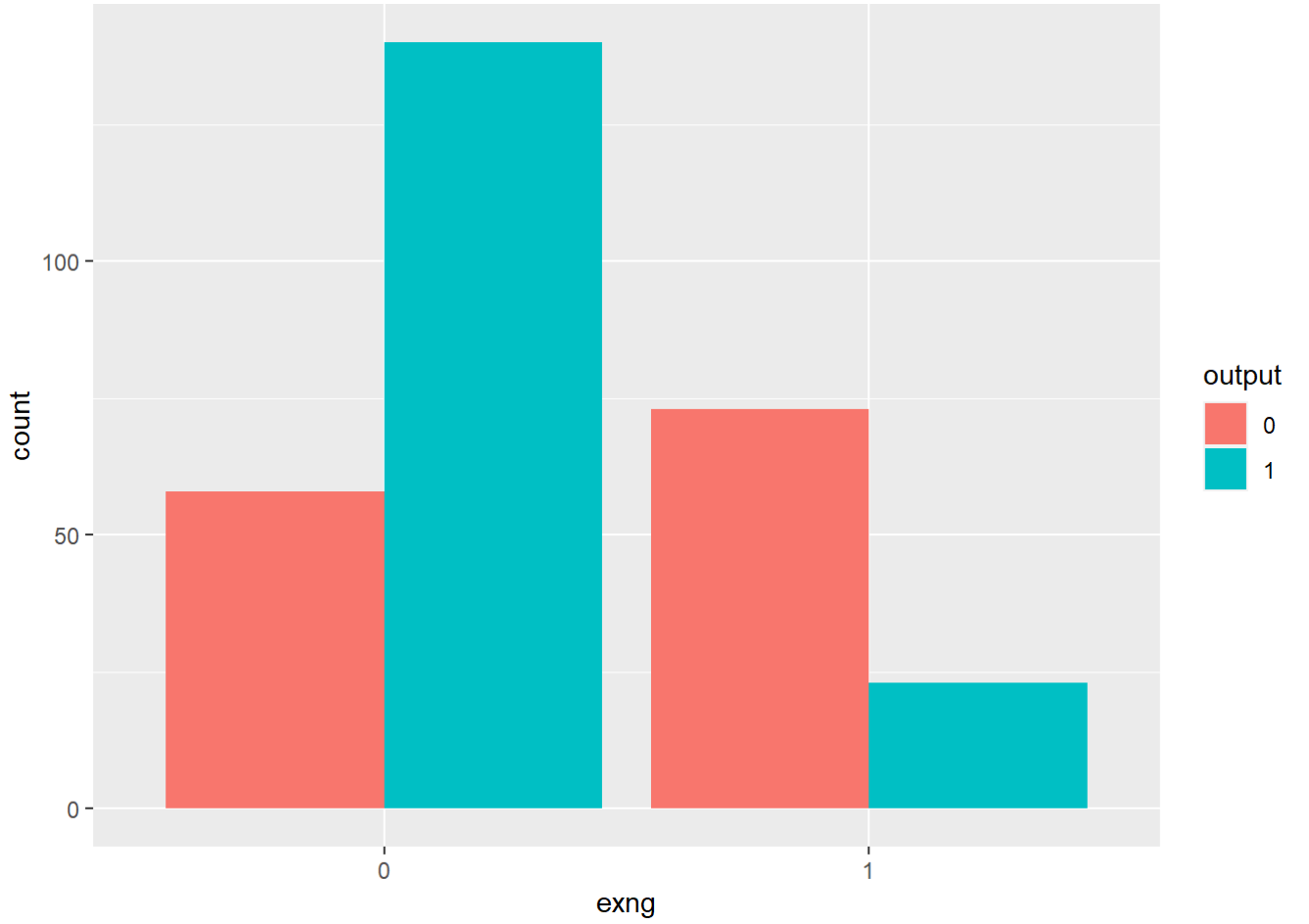

```
ggplot(data = data_categorical, aes(x = fbs, after_stat(count) ))+geom_bar(aes(fill = outpu
t), position = "dodge")
```



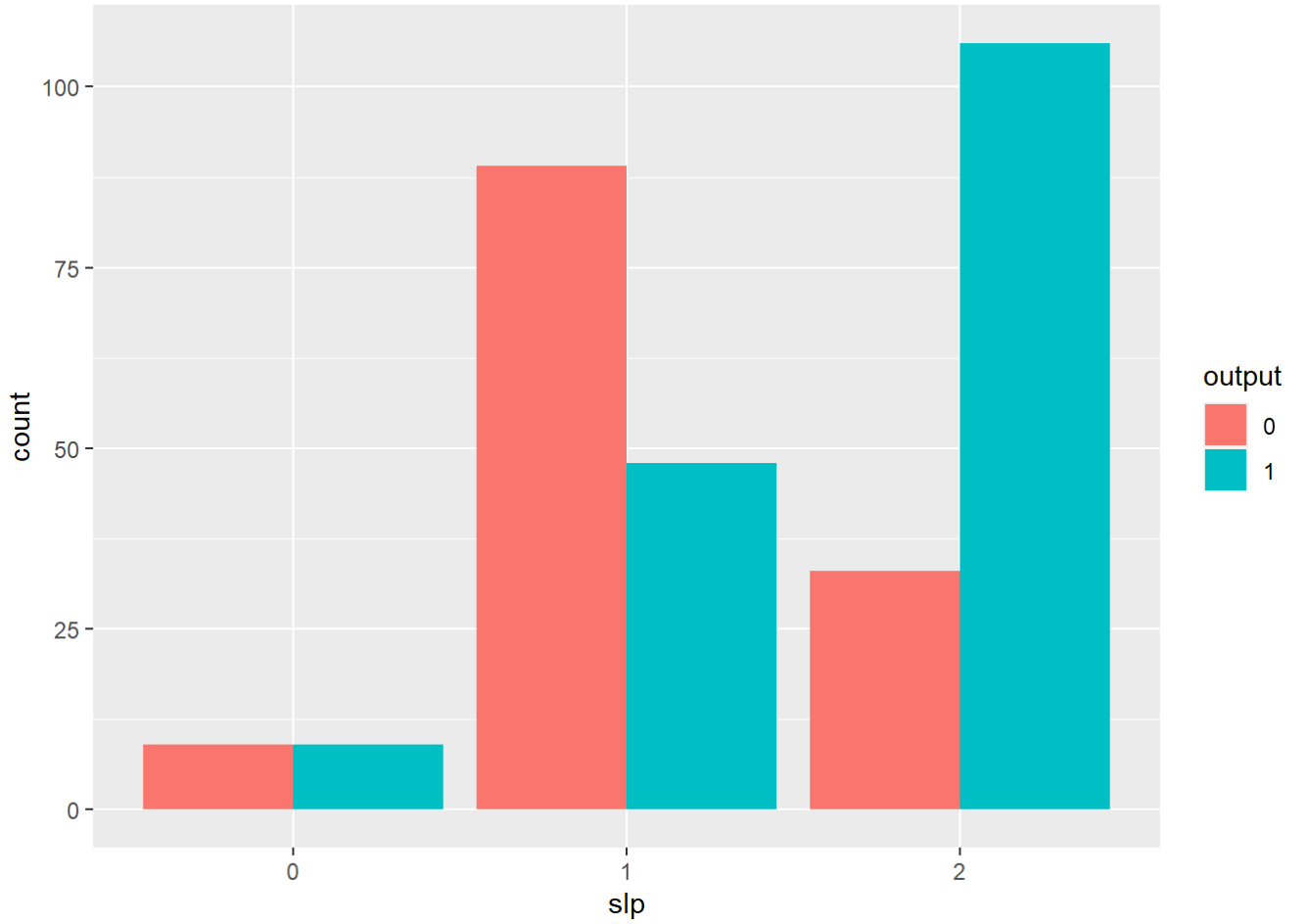
```
ggplot(data = data_categorical, aes(x = restecg, after_stat(count) ))+geom_bar(aes(fill = out
put), position = "dodge")
```



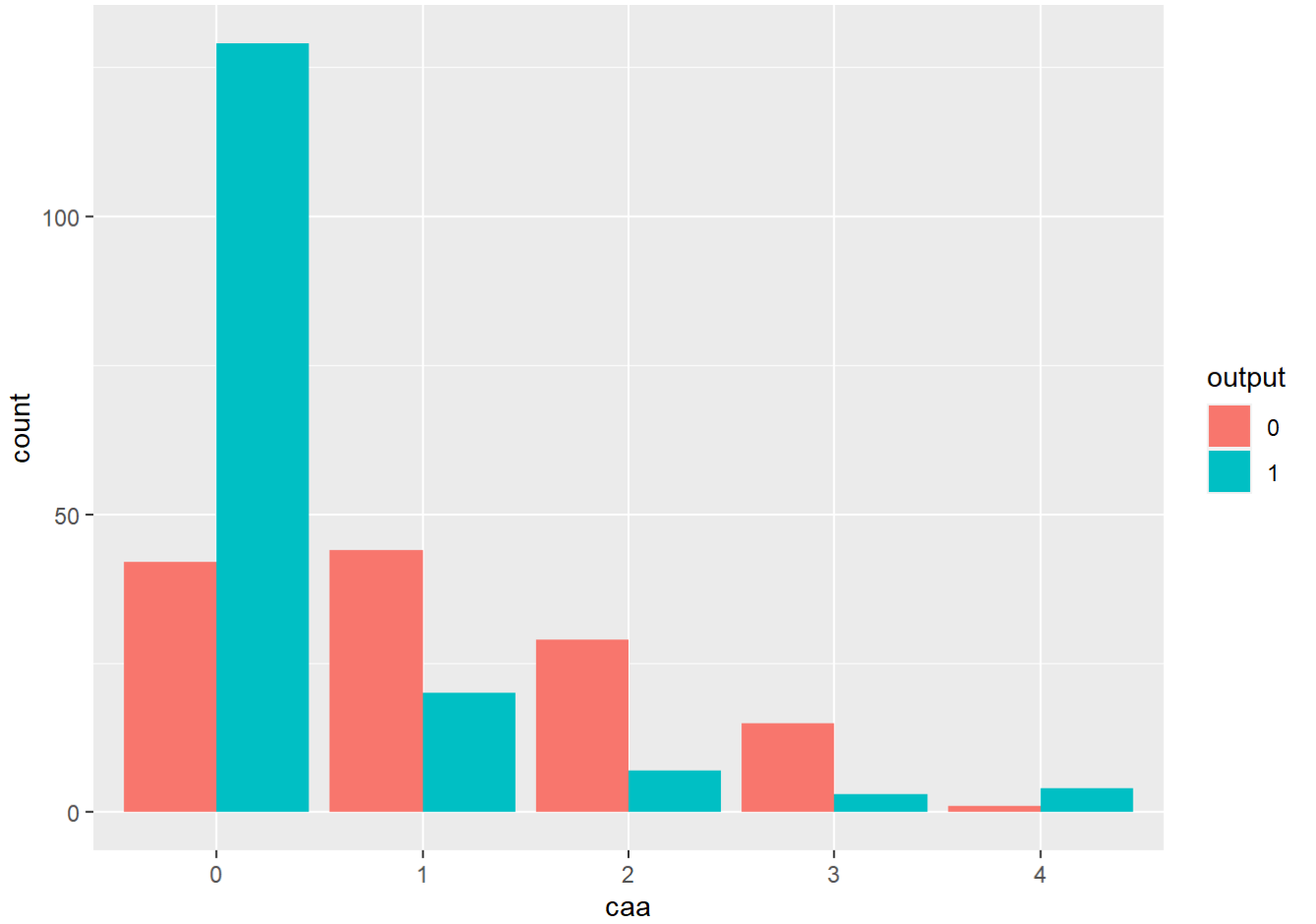

```
ggplot(data = data_categorical, aes(x = exng, after_stat(count) ))+geom_bar(aes(fill = outpu
t), position = "dodge")
```



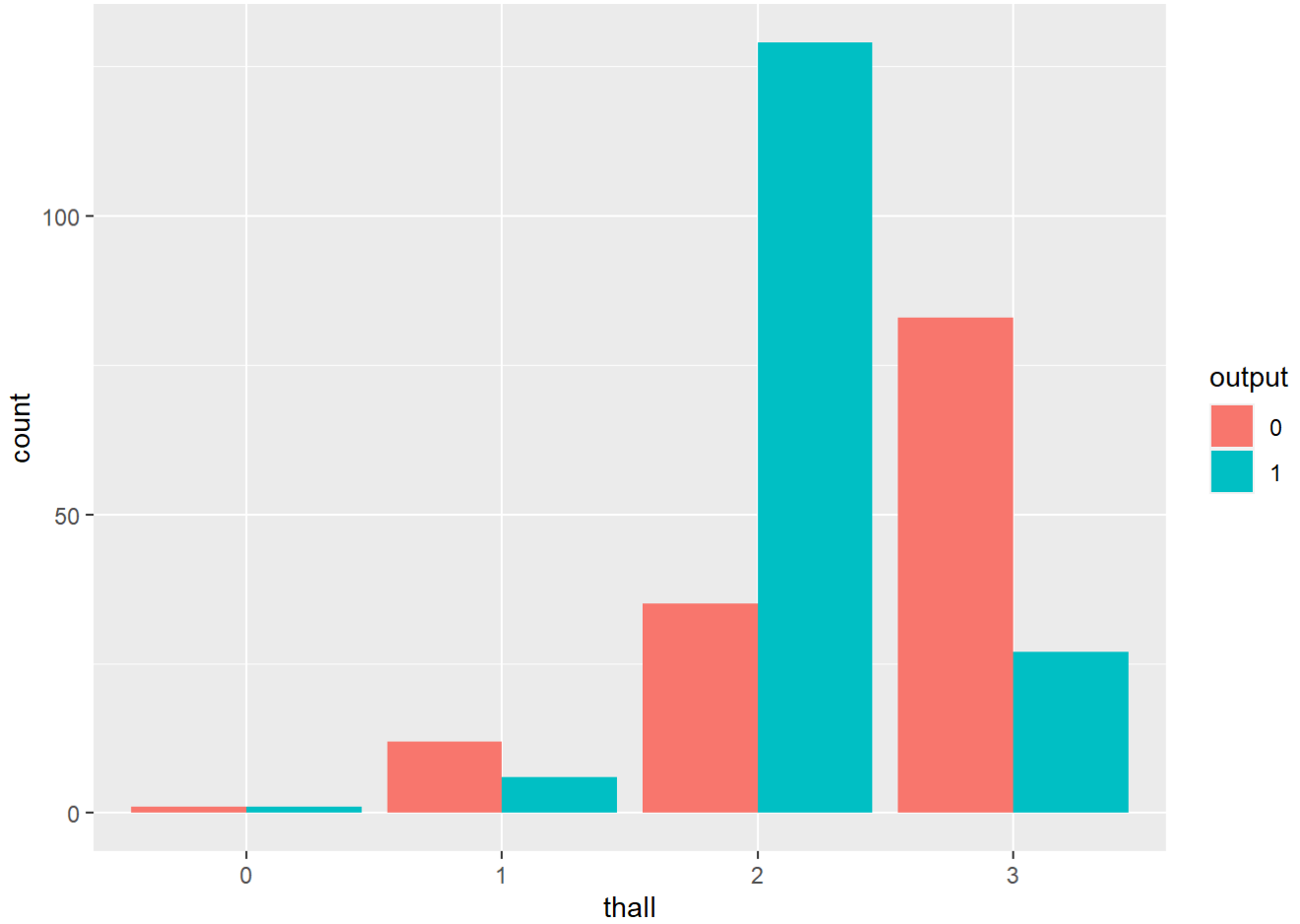

```
ggplot(data = data_categorical, aes(x = slp, after_stat(count) ))+geom_bar(aes(fill = outpu
t), position = "dodge")
```



```
ggplot(data = data_categorical, aes(x = caa, after_stat(count) ))+geom_bar(aes(fill = outpu
t), position = "dodge")
```




```
ggplot(data = data_categorical, aes(x = thall, after_stat(count) ))+geom_bar(aes(fill = outpu
t), position = "dodge")
```



Veiem forçes valors que poden deures a les poques mostres del dataset (14 variables per 300 mostres, no són gaires), tot i així sembla que hi ha més del doble de pacients femenins que masculins, per tant les dades son 100~200).

Comprovació de la normalitat i homogeneïtat de

la variància:

Comprovació de la normalitat

```
columns = c("name","normalitat")
df = data.frame(matrix(nrow = 0, ncol = length(columns)))
colnames(df) = columns

data_numerical <- as.data.frame(data_heart %>%
                  select("trtbps","thalachh","chol","oldpeak", "age"))


for (col in names(data_numerical)) {
  normalitat <- TRUE

  resultat <- shapiro.test(data_numerical[[col]])
  if(resultat$p.value<0.05){
    #No Normalitat
    normalitat<-FALSE
  }
  df[nrow(df)+1,1] = col
  df[nrow(df),2] = normalitat

}
print(df)
```

```
##       name normalitat
## 1   trtbps      FALSE
## 2 thalachh      FALSE
## 3     chol       TRUE
## 4  oldpeak      FALSE
## 5      age      FALSE
```

Comprovació Homoscedasticitat

Comprovació homoscedasticitat per la variable chol (unica variable amb normalitat en la variancia) amb els seguents grups: sex, cp, fbs, restecg, exng, slp, thall, output

```
columns = c("name1","name2","homoscedasticitat")
df = data.frame(matrix(nrow = 0, ncol = length(columns)))
colnames(df) = columns

chol <- data_heart[["chol"]]



for (col in names(data_categorical)) {
  homogeneitat <- TRUE
  test <- leveneTest(chol~data_categorical[[col]])

  if(test$`Pr(>F)`[1]<0.05){
    homogeneitat <- FALSE
  }
  df[nrow(df)+1,1:2] <- c("chol", col)
  df[nrow(df),3] <- homogeneitat

}
print(df)
```

```
##   name1   name2 homoscedasticitat
## 1  chol     sex              TRUE
## 2  chol      cp              TRUE
## 3  chol     fbs              TRUE
## 4  chol restecg              TRUE
## 5  chol    exng              TRUE
## 6  chol     slp              TRUE
## 7  chol     caa              TRUE
## 8  chol   thall              TRUE
## 9  chol  output              TRUE
```

```
data_numerical_nn <- as.data.frame(data_heart %>%
                  select("trtbps","thalachh","oldpeak", "age"))

for(numerical in names(data_numerical_nn)){
  for (col in names(data_categorical)) {
    homogeneitat <- TRUE
    test <- fligner.test(data_numerical_nn[[numerical]]~data_categorical[[col]])

    if(test$p.value<0.05){
      homogeneitat <- FALSE
    }
    df[nrow(df)+1,1:2] <- c(numerical,col)
    df[nrow(df),3] <- homogeneitat

  }
}
print(df)
```

```
##       name1   name2 homoscedasticitat
## 1      chol     sex              TRUE
## 2      chol      cp              TRUE
## 3      chol     fbs              TRUE
## 4      chol restecg              TRUE
## 5      chol    exng              TRUE
## 6      chol     slp              TRUE
## 7      chol     caa              TRUE
## 8      chol   thall              TRUE
## 9      chol  output              TRUE
## 10   trtbps     sex              TRUE
## 11   trtbps      cp              TRUE
## 12   trtbps     fbs              TRUE
## 13   trtbps restecg              TRUE
## 14   trtbps    exng              TRUE
## 15   trtbps     slp              TRUE
## 16   trtbps     caa              TRUE
## 17   trtbps   thall              TRUE
## 18   trtbps  output              TRUE
## 19 thalachh     sex              TRUE
## 20 thalachh      cp              TRUE
## 21 thalachh     fbs              TRUE
## 22 thalachh restecg              TRUE
## 23 thalachh    exng              TRUE
## 24 thalachh     slp             FALSE
## 25 thalachh     caa              TRUE
## 26 thalachh   thall              TRUE
## 27 thalachh  output              TRUE
## 28  oldpeak     sex             FALSE
## 29  oldpeak      cp             FALSE
## 30  oldpeak     fbs              TRUE
## 31  oldpeak restecg             FALSE
## 32  oldpeak    exng             FALSE
## 33  oldpeak     slp             FALSE
## 34  oldpeak     caa             FALSE
## 35  oldpeak   thall              TRUE
## 36  oldpeak  output             FALSE
## 37      age     sex              TRUE
## 38      age      cp              TRUE
## 39      age     fbs             FALSE
## 40      age restecg              TRUE
## 41      age    exng              TRUE
## 42      age     slp              TRUE
## 43      age     caa             FALSE
## 44      age   thall             FALSE
## 45      age  output             FALSE
```

Proves estadístiques:

Contrast d’hipotesis

El contrast d’hipòtesis el farem per determinar el si el gènere influeix.

Es planteja el següent contrast d’hipòtesis:

- Hipòtesi nul·la: La probabilitat de patir malaltia és independent al gènere.
- Hhipòtesi alternativa: La probabilitat de patir malalties és major en homes.

```
home_mostra <- data_heart[data_heart$sex == 0,]$output
dona_mostra <- data_heart[data_heart$sex == 1,]$output

t.test(dona_mostra, home_mostra, alternative = 'less')
```

```
##
##  Welch Two Sample t-test
##
## data:  dona_mostra and home_mostra
## t = -5.6021, df = 197.29, p-value = 3.525e-08
## alternative hypothesis: true difference in means is less than 0
## 95 percent confidence interval:
##        -Inf -0.2233045
## sample estimates:
## mean of x mean of y
## 0.4585366 0.7752809
```

Com es pot veure, es pot confirmar la hipòtesis plantejada ja que el gènere influeix. Aquest resultat es podia intuir en els grafic entre sex i output.

Models

Preperació de les dades

```
# Recuperem les dades numeriques i les normalitzem
data_numerical_norm <- scale(data_numerical)
data_categorical
```

```
##     sex cp fbs restecg exng slp caa thall output
## 1     1  3   1       0    0   0   0     1      1
## 2     1  2   0       1    0   0   0     2      1
## 3     0  1   0       0    0   2   0     2      1
## 4     1  1   0       1    0   2   0     2      1
## 5     0  0   0       1    1   2   0     2      1
## 6     1  0   0       1    0   1   0     1      1
## 7     0  1   0       0    0   1   0     2      1
## 8     1  1   0       1    0   2   0     3      1
## 9     1  2   1       1    0   2   0     3      1
## 10    1  2   0       1    0   2   0     2      1
## 11    1  0   0       1    0   2   0     2      1
## 12    0  2   0       1    0   2   0     2      1
## 13    1  1   0       1    0   2   0     2      1
## 14    1  0   0       0    1   1   0     2      1
## 15    0  3   1       0    0   2   0     2      1
## 16    0  2   0       1    0   1   0     2      1
## 17    0  2   0       1    0   2   0     2      1
## 18    0  3   0       1    0   0   0     2      1
## 19    1  0   0       1    0   2   0     2      1
## 20    0  3   0       1    0   2   2     2      1
## 21    1  0   0       1    0   1   0     3      1
## 22    1  2   0       1    1   2   0     2      1
## 23    1  0   0       1    0   2   0     2      1
## 24    1  2   1       1    1   1   0     2      1
## 25    1  3   0       1    1   2   0     3      1
## 26    0  1   0       1    0   2   2     2      1
## 27    1  2   1       1    0   2   0     2      1
## 28    1  2   0       1    0   2   0     2      1
## 30    1  2   1       0    0   0   0     2      1
## 31    0  1   0       1    0   2   1     2      1
## 32    1  0   0       1    0   2   0     3      1
## 33    1  1   0       0    0   2   0     2      1
## 34    1  2   0       0    0   0   1     2      1
## 35    1  3   0       0    1   2   1     2      1
## 36    0  2   0       0    1   0   0     2      1
## 37    0  2   1       1    0   2   0     2      1
## 38    1  2   0       0    0   2   0     3      1
## 39    0  2   0       1    0   2   0     2      1
## 40    0  2   0       0    0   2   0     2      1
## 41    0  2   0       0    0   2   1     2      1
## 42    1  1   0       0    0   1   0     2      1
## 43    1  0   0       0    1   1   0     2      1
## 44    0  0   0       0    0   1   0     2      1
## 45    1  2   0       0    0   2   0     2      1
## 46    1  1   0       1    0   2   0     2      1
## 47    1  2   0       0    0   2   0     2      1
## 48    1  2   0       0    0   2   0     2      1
## 49    0  2   0       0    0   2   0     0      1
## 50    0  1   0       0    0   2   0     2      1
## 51    0  2   0       0    0   2   0     2      1
## 52    1  0   0       0    0   1   0     2      1
## 53    1  2   0       1    0   1   3     3      1
## 54    0  2   0       1    0   1   0     2      1
## 55    0  2   0       0    0   2   0     2      1
```

```
## 56    1  1   0       1    0   2   1     2      1
## 57    1  0   0       0    0   2   0     2      1
## 58    1  0   0       0    0   2   0     2      1
## 59    1  3   0       0    0   2   0     2      1
## 60    0  0   0       0    0   2   1     2      1
## 61    0  2   1       0    0   2   1     2      1
## 62    1  1   0       1    0   2   0     3      1
## 63    1  3   0       0    0   1   0     1      1
## 64    1  1   0       1    0   1   0     1      1
## 65    1  2   1       0    0   2   0     2      1
## 66    0  0   0       1    0   2   0     2      1
## 67    1  2   0       1    1   1   0     2      1
## 68    0  1   0       0    0   1   0     2      1
## 69    1  1   0       1    0   2   0     2      1
## 70    0  0   0       1    0   2   0     2      1
## 71    1  2   0       0    0   1   0     3      1
## 72    1  2   0       1    1   2   1     3      1
## 73    1  1   0       0    0   2   0     2      1
## 74    1  0   0       0    1   2   0     2      1
## 75    0  2   0       1    0   1   0     2      1
## 76    0  1   0       0    0   1   0     2      1
## 77    1  2   1       0    0   1   0     2      1
## 78    1  1   0       1    1   2   0     2      1
## 79    1  1   1       1    0   2   0     2      1
## 80    1  2   0       0    1   1   0     3      1
## 81    1  2   0       1    0   2   0     2      1
## 82    1  1   0       0    0   2   0     2      1
## 83    0  2   0       1    0   2   1     2      1
## 84    1  3   1       1    0   1   0     3      1
## 85    0  0   0       0    0   1   0     2      1
## 87    1  2   0       1    0   2   1     3      1
## 88    1  1   1       1    0   2   0     3      1
## 89    0  2   0       1    0   1   0     2      1
## 90    0  0   0       0    0   1   0     2      1
## 91    1  2   1       1    0   2   2     2      1
## 92    1  0   0       1    1   2   0     3      1
## 93    1  2   0       1    0   2   4     2      1
## 94    0  1   1       0    1   2   1     2      1
## 95    0  1   0       1    0   1   0     2      1
## 96    1  0   0       0    1   2   0     3      1
## 97    0  0   0       0    0   1   0     2      1
## 98    1  0   1       1    0   2   3     3      1
## 99    1  2   0       1    0   2   1     2      1
## 100   1  2   1       0    0   2   3     2      1
## 101   1  3   0       0    0   2   2     2      1
## 102   1  3   0       0    0   0   0     3      1
## 103   0  1   0       1    0   2   2     2      1
## 104   1  2   1       1    0   0   0     3      1
## 105   1  2   0       1    0   2   0     2      1
## 106   0  2   0       0    0   1   0     2      1
## 107   1  3   1       0    0   1   1     2      1
## 108   0  0   0       0    1   1   0     2      1
## 109   0  1   0       1    0   2   0     2      1
## 110   0  0   0       0    0   2   0     2      1
## 111   0  0   0       1    1   2   0     2      1
## 112   1  2   1       1    0   2   1     3      1
```

```
## 113   0  2   0       0    0   2   0     3      1
## 114   1  0   0       1    0   2   0     3      1
## 115   1  1   0       1    0   2   0     2      1
## 116   0  2   0       1    0   2   0     2      1
## 117   1  2   0       0    0   1   0     2      1
## 118   1  3   0       0    0   1   0     3      1
## 119   0  1   0       1    0   2   0     2      1
## 120   0  0   0       0    1   1   0     2      1
## 121   0  0   0       1    0   1   2     2      1
## 122   1  0   0       0    0   2   0     2      1
## 123   0  0   0       0    1   2   0     2      1
## 124   0  2   0       0    0   2   0     2      1
## 125   1  2   0       1    0   2   0     2      1
## 126   0  1   0       1    0   2   0     2      1
## 127   1  0   0       1    0   2   0     2      1
## 128   0  2   0       1    0   2   1     2      1
## 129   0  2   0       0    0   1   0     2      1
## 130   0  1   0       0    1   2   1     2      1
## 131   0  2   0       1    0   2   1     2      1
## 132   0  1   0       1    0   1   0     2      1
## 133   1  1   0       1    0   2   0     2      1
## 134   1  1   0       1    0   2   0     2      1
## 135   0  1   0       1    0   2   0     2      1
## 136   0  0   0       1    0   2   0     2      1
## 137   0  2   1       1    0   2   0     2      1
## 138   1  1   1       0    0   2   0     2      1
## 139   1  0   0       1    1   1   0     1      1
## 140   1  0   0       1    1   1   1     3      1
## 141   0  2   0       0    0   2   0     2      1
## 142   1  0   0       1    0   1   0     2      1
## 143   0  2   0       1    0   1   0     2      1
## 144   0  0   0       1    0   2   2     2      1
## 145   0  2   0       2    0   1   0     2      1
## 146   1  1   0       0    0   2   0     2      1
## 147   0  2   0       1    0   1   1     2      1
## 148   0  3   0       1    0   2   0     2      1
## 149   1  2   0       1    0   2   0     2      1
## 150   1  2   0       1    0   2   0     2      1
## 151   1  0   0       0    0   2   0     1      1
## 152   0  0   0       1    0   1   0     2      1
## 153   1  3   0       0    0   1   0     3      1
## 154   0  2   0       0    0   1   1     2      1
## 155   0  2   0       1    0   1   0     2      1
## 156   0  0   0       1    0   1   0     2      1
## 157   1  2   0       1    0   2   0     2      1
## 158   1  1   0       1    0   2   0     2      1
## 159   1  1   0       1    0   1   4     3      1
## 160   1  1   0       0    0   2   0     3      1
## 161   1  1   0       1    0   0   0     2      1
## 162   0  1   0       1    0   2   0     2      1
## 163   1  1   0       1    0   2   0     2      1
## 164   1  2   0       1    0   2   4     2      1
## 165   1  2   0       1    0   2   4     2      1
## 166   1  0   0       0    1   1   3     2      0
## 167   1  0   0       0    1   1   2     3      0
## 168   0  0   0       0    0   0   2     2      0
```

```
## 169   1  0   0       0    0   1   1     3      0
## 170   1  0   1       0    1   0   0     3      0
## 171   1  2   1       0    1   1   1     1      0
## 172   1  1   0       1    0   0   0     3      0
## 173   1  1   0       0    0   1   0     2      0
## 174   1  2   0       0    0   2   2     3      0
## 175   1  0   0       0    1   1   2     3      0
## 176   1  0   0       0    1   1   0     3      0
## 177   1  0   1       1    1   2   2     3      0
## 178   1  2   0       1    0   2   0     2      0
## 179   1  0   0       0    1   1   0     3      0
## 180   1  0   0       0    1   1   1     1      0
## 181   1  0   0       1    1   1   1     3      0
## 182   0  0   0       0    0   1   3     3      0
## 183   0  0   0       0    0   2   0     2      0
## 184   1  2   0       0    0   1   1     3      0
## 185   1  0   0       0    0   1   0     3      0
## 186   1  0   0       0    0   2   1     2      0
## 187   1  0   0       0    1   2   1     3      0
## 188   1  0   0       0    1   1   1     3      0
## 189   1  2   0       1    0   1   1     3      0
## 190   1  0   0       0    0   2   0     3      0
## 191   0  0   0       1    1   1   0     3      0
## 192   1  0   0       0    1   1   3     3      0
## 193   1  0   0       1    0   1   1     3      0
## 194   1  0   0       0    1   1   2     3      0
## 195   1  2   0       0    0   1   0     2      0
## 196   1  0   0       0    1   0   0     3      0
## 197   1  2   0       1    0   1   0     2      0
## 198   1  0   1       1    0   1   2     3      0
## 199   1  0   0       1    1   1   2     3      0
## 200   1  0   0       0    0   2   2     1      0
## 201   1  0   0       0    0   2   1     2      0
## 202   1  0   0       0    1   1   1     3      0
## 203   1  0   0       0    1   2   0     3      0
## 204   1  2   1       0    1   1   0     3      0
## 206   1  0   0       1    1   2   1     3      0
## 207   1  0   0       0    1   1   1     3      0
## 208   0  0   0       0    0   1   2     3      0
## 209   1  2   0       1    0   1   3     3      0
## 210   1  0   0       1    1   2   1     3      0
## 211   1  2   0       0    0   1   1     3      0
## 212   1  0   0       1    1   1   1     3      0
## 213   1  0   0       1    0   1   0     3      0
## 214   0  0   0       0    1   1   0     3      0
## 215   1  0   1       0    1   1   1     2      0
## 216   0  0   1       0    1   1   0     3      0
## 217   0  2   0       1    0   1   1     3      0
## 218   1  0   1       0    1   2   3     3      0
## 219   1  0   0       0    0   1   1     3      0
## 220   1  0   1       0    1   2   2     3      0
## 223   1  3   1       0    0   1   1     2      0
## 225   1  0   0       1    1   1   1     3      0
## 226   1  0   0       1    1   0   0     3      0
## 227   1  1   0       0    0   1   1     3      0
## 228   1  0   0       1    1   1   0     3      0
```

```
## 229   1  3   0       0    0   1   0     3      0
## 230   1  2   0       1    1   1   0     3      0
## 231   1  2   0       1    0   2   0     2      0
## 232   1  0   1       0    0   1   3     3      0
## 233   1  0   0       0    1   1   1     3      0
## 234   1  0   0       0    1   0   1     2      0
## 235   1  0   0       0    0   1   3     2      0
## 236   1  0   0       1    1   1   0     3      0
## 237   1  0   0       0    0   2   2     3      0
## 238   1  0   0       0    0   1   2     3      0
## 239   1  0   0       0    1   2   3     2      0
## 240   1  0   0       0    1   2   0     3      0
## 241   1  2   0       1    1   1   1     3      0
## 242   0  0   0       1    1   1   0     2      0
## 243   1  0   0       0    0   1   2     1      0
## 244   1  0   0       1    1   1   1     3      0
## 245   1  0   0       0    1   1   1     1      0
## 246   1  0   0       0    0   1   0     3      0
## 248   1  1   0       1    1   1   3     1      0
## 250   1  2   0       0    0   1   3     3      0
## 251   1  0   0       1    1   1   3     3      0
## 252   1  0   1       0    1   1   4     3      0
## 253   0  0   1       1    0   1   3     2      0
## 254   1  0   0       0    1   1   2     2      0
## 255   1  3   0       0    0   2   0     2      0
## 256   1  0   0       0    1   1   3     3      0
## 257   1  0   0       0    1   1   2     3      0
## 258   1  0   0       0    1   1   0     3      0
## 259   0  0   0       1    1   1   0     2      0
## 260   1  3   0       1    1   1   0     3      0
## 261   0  0   1       1    1   1   2     3      0
## 262   1  0   0       1    0   2   1     2      0
## 263   1  0   0       1    1   1   2     3      0
## 264   0  0   0       1    1   1   2     2      0
## 265   1  0   0       0    1   1   1     2      0
## 266   1  0   0       0    1   2   1     2      0
## 267   0  0   0       2    1   1   0     2      0
## 268   1  2   0       0    0   2   3     2      0
## 269   1  0   0       0    1   1   2     2      0
## 270   1  0   1       0    1   0   0     3      0
## 271   1  0   0       0    0   2   0     3      0
## 272   1  3   0       1    0   1   2     2      0
## 274   1  0   0       1    0   2   1     3      0
## 275   1  0   0       0    1   1   1     2      0
## 276   1  0   0       1    0   2   2     3      0
## 277   1  0   0       1    0   1   1     3      0
## 278   1  1   0       1    0   2   0     3      0
## 279   0  1   1       0    0   2   2     2      0
## 280   1  0   0       0    1   1   1     2      0
## 281   1  0   0       1    1   1   0     1      0
## 282   1  0   1       1    1   1   1     0      0
## 283   1  2   1       1    0   1   1     1      0
## 284   1  0   0       1    0   2   0     3      0
## 285   1  0   0       0    1   2   1     3      0
## 286   1  0   0       1    1   1   2     3      0
## 287   1  3   0       1    0   2   2     2      0
```

```
## 288   1  1   0       0    0   2   1     2      0
## 289   1  0   0       1    1   1   1     3      0
## 290   0  0   0       2    1   1   1     3      0
## 291   1  0   0       1    0   2   1     3      0
## 292   1  0   0       2    0   0   3     1      0
## 293   0  0   1       0    1   1   2     1      0
## 294   1  2   0       0    0   1   0     3      0
## 295   1  0   0       1    1   0   0     1      0
## 296   1  0   0       0    1   2   2     3      0
## 297   0  0   0       1    1   1   0     2      0
## 298   1  0   1       0    0   1   2     1      0
## 299   0  0   0       1    1   1   0     3      0
## 300   1  3   0       1    0   1   0     3      0
## 301   1  0   1       1    0   1   2     3      0
## 302   1  0   0       1    1   1   1     3      0
## 303   0  1   0       0    0   1   1     2      0
```

```
#combinem les dades i creem el dataframe definitiu amb el que farem els models
data_model = cbind(data_numerical_norm, data_categorical)

# Dividirem el dataset en train - test
set.seed(123)
split = sample.split(data_model$output, SplitRatio = 0.80)
train_set = subset(data_model, split == TRUE)
test_set = subset(data_model, split == FALSE)
```

Naive Bayes

Utilitzem el model Naive Bayes per predir la classificació dels pacients. Per tant podriem donar resposta a la pregunta formalitzada a l'inici de la pràctica "Podem predir el risc d'un pacient de patir problemes cardiacs?".

```
# https://www.rdocumentation.org/packages/e1071/versions/1.7-12/topics/naiveBayes
set.seed(123)
model_nB = naiveBayes(output~., data=train_set)
y_pred = predict(model_nB, newdata = test_set)

y_pred
```

```
##  [1] 1 1 1 1 1 1 0 1 1 1 0 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 0 1 1 0 0 0 1 0 0
## [39] 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 1 0 1
## Levels: 0 1
```

```
cm_results = confusionMatrix(table(test_set$output, y_pred))
print(cm_results)
```

```
## Confusion Matrix and Statistics
##
##    y_pred
##      0  1
##   0 22  4
##   1  4 29
##
##                Accuracy : 0.8644
##                  95% CI : (0.7502, 0.9396)
##     No Information Rate : 0.5593
##     P-Value [Acc > NIR] : 5.256e-07
##
##                   Kappa : 0.7249
##
##  Mcnemar's Test P-Value : 1
##
##             Sensitivity : 0.8462
##             Specificity : 0.8788
##          Pos Pred Value : 0.8462
##          Neg Pred Value : 0.8788
##              Prevalence : 0.4407
##          Detection Rate : 0.3729
##    Detection Prevalence : 0.4407
##       Balanced Accuracy : 0.8625
##
##        'Positive' Class : 0
##
```

Com podem veure, amb aquest model hem aconseguit una accuracy de 0.7195.

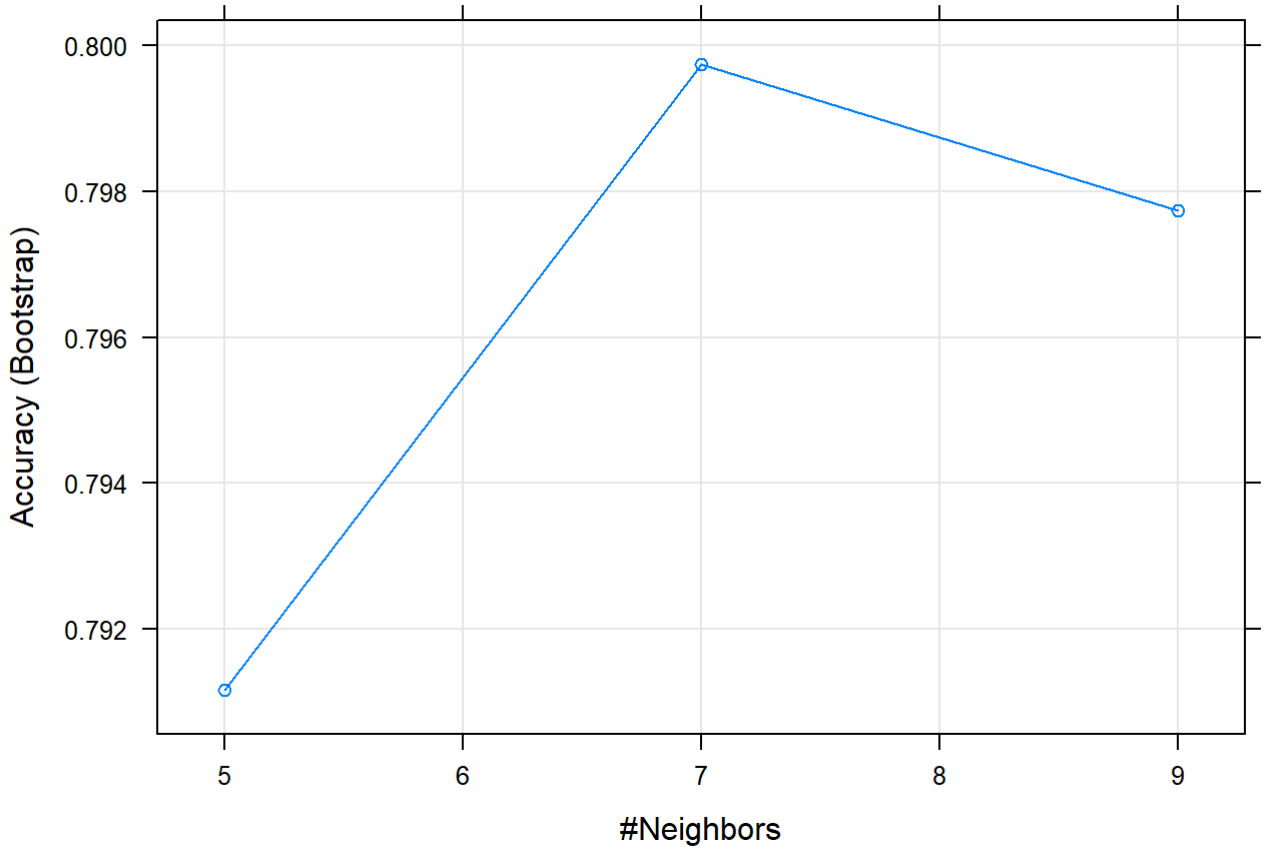
KNN

Utilitzem el model KNN per predir la classificació dels pacients. Per tant podriem donar resposta a la pregunta formalitzada a l'inici de la pràctica "Podem predir el risc d'un pacient de patir problemes cardiacs?". Utilitzem aquest algoritme perque és un algoritme classificador i ens podria servir per agrupar diferents tipus de pacients i fer diferents seguiments medics.

```
# https://rpubs.com/njvijay/16444
set.seed(123)

knn_fit <- train(output ~., data = train_set, method = "knn")
knn_fit
```

```
## k-Nearest Neighbors
##
## 235 samples
##  13 predictor
##   2 classes: '0', '1'
##
## No pre-processing
## Resampling: Bootstrapped (25 reps)
## Summary of sample sizes: 235, 235, 235, 235, 235, 235, ...
## Resampling results across tuning parameters:
##
##   k  Accuracy   Kappa
##   5  0.7911554  0.5767966
##   7  0.7997473  0.5949919
##   9  0.7977410  0.5904384
##
## Accuracy was used to select the optimal model using the largest value.
## The final value used for the model was k = 7.
```

```
plot(knn_fit)
```



Extracció Dataset

```
res <- c(data_heart, "./heartClean.csv", row.names=TRUE)
```

Conclusió:

Inicialment voliem donar resposta als parametres més importants i si podiem preveure nous pacients amb èxit.

Creiem que amb els resultats de la matriu de correlació i els anàlisis fets a les variables tant númeriques com categoriques, sabem que els atributs cp i thalachh tenen més impacte.

En quant a preveure nous pacients, concluim creient que és un camí possible sempre que hi hagi col·laboració d'experts en el domini, i que les mostres de dades siguin molt més representatives, ja que hi havia força dones en comparació i les franjes d'edad estaven poc representades (impedint comprovar que pasa en diferents franjes d'edat).

Considerem interesant aplicar diferents algoritmes com el KNN per pacients amb una determinada edat i veure si l'accuracy augmenta o no..

Hi ha per tant moltes preguntes a fer amb aquestes dades, esperem que amb el temps es doni resposta i la ciencia de dades sigui clau en el desenvolupament de la salut de les persones.

Contribucions

Contribucions	Firma
Investigació prèvia	Marc González i Maria Sunyer
Redacció de les respostes	Marc González i Maria Sunyer
Desenvolupament del codi	Marc González i Maria Sunyer
Participació al vídeo	Marc González i Maria Sunyer