Statistička analiza podataka - Projekt Analiza čimbenika rizika za srčane bolesti

Saprofiti

2025-01-15

library(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

library(ggplot2)  
library(magrittr)  
library(tidyr)

##   
## Attaching package: 'tidyr'

## The following object is masked from 'package:magrittr':  
##   
## extract

library(readr)

## R Markdown

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:

Note that the echo = FALSE parameter was added to the code chunk to prevent printing of the R code that generated the plot.

# Učitavanje podataka

data <- read\_csv("heart.csv")

## Rows: 918 Columns: 12  
## ── Column specification ────────────────────────────────────────────────────────  
## Delimiter: ","  
## chr (5): Sex, ChestPainType, RestingECG, ExerciseAngina, ST\_Slope  
## dbl (7): Age, RestingBP, Cholesterol, FastingBS, MaxHR, Oldpeak, HeartDisease  
##   
## ℹ Use `spec()` to retrieve the full column specification for this data.  
## ℹ Specify the column types or set `show\_col\_types = FALSE` to quiet this message.

### DESKRIPTIVNA STATISTIKA i VIZUALIZACIJA

# Prikaz prvih 6 redova

head(data)

## # A tibble: 6 × 12  
## Age Sex ChestPainType RestingBP Cholesterol FastingBS RestingECG MaxHR  
## <dbl> <chr> <chr> <dbl> <dbl> <dbl> <chr> <dbl>  
## 1 40 M ATA 140 289 0 Normal 172  
## 2 49 F NAP 160 180 0 Normal 156  
## 3 37 M ATA 130 283 0 ST 98  
## 4 48 F ASY 138 214 0 Normal 108  
## 5 54 M NAP 150 195 0 Normal 122  
## 6 39 M NAP 120 339 0 Normal 170  
## # ℹ 4 more variables: ExerciseAngina <chr>, Oldpeak <dbl>, ST\_Slope <chr>,  
## # HeartDisease <dbl>

# Opis značajki

AGE -> Age of the patient [years] Sex -> Sex of the patient [M: Male, F: Female] ChestPainType -> chest pain type [TA: Typical Angina, ATA: Atypical Angina, NAP: Non-Anginal Pain, ASY: Asymptomatic] RestingBP -> Resting blood pressure [mm Hg] Cholesterol -> Serum cholesterol [mm/dl] FastingBS -> Fasting blood sugar [1: if Fasting blood sugar > 120 mg/dl, 0: otherwise] RestingECG -> Resting electrocardiogram results [Normal: Normal, ST: having ST-T wave abnormality (T wave inversions and/or ST elevation or depression of > 0.05 mV), LVH: showing probable or definite left ventricular hypertrophy by Estes’ criteria] MaxHR -> Maximum heart rate achieved [Numeric value between 60 and 202] ExerciseAngina -> Exercise-induced angina [Y: Yes, N: No] Oldpeak -> Exercise-induced ST depression [ST depression induced by exercise relative to rest] ST\_Slope -> Slope of the peak exercise ST segment [Up: upsloping, Flat: flat, Down: downsloping] HeartDisease -> Output class [1: heart disease, 0: Normal]

# Opis podataka

dim(data) # broj redaka, broj stupaca (broj primjera, broj varijabli)

## [1] 918 12

cat("Number of rows: ", nrow(data), "\n")

## Number of rows: 918

cat("Number of columns: ", ncol(data), "\n")

## Number of columns: 12

cat("Column names:", paste(colnames(data), collapse = ", "), "\n")

## Column names: Age, Sex, ChestPainType, RestingBP, Cholesterol, FastingBS, RestingECG, MaxHR, ExerciseAngina, Oldpeak, ST\_Slope, HeartDisease

# Čišćenje podataka

# Monotone i konstantne vrijednosti

cat("Number of rows is ", nrow(data), "\n")

## Number of rows is 918

for (column in colnames(data)){  
 cat("Column: ", column, "has ", length(unique(data[[column]])), " unique values\n")  
}

## Column: Age has 50 unique values  
## Column: Sex has 2 unique values  
## Column: ChestPainType has 4 unique values  
## Column: RestingBP has 67 unique values  
## Column: Cholesterol has 222 unique values  
## Column: FastingBS has 2 unique values  
## Column: RestingECG has 3 unique values  
## Column: MaxHR has 119 unique values  
## Column: ExerciseAngina has 2 unique values  
## Column: Oldpeak has 53 unique values  
## Column: ST\_Slope has 3 unique values  
## Column: HeartDisease has 2 unique values

Pošto je broj jedinstvenih vrijednosti za svaku značajku manji od broja redaka, možemo zaključiti da nema monotone vrijednosti (napomena: provjerili smo i ručno kako se ne bi dogodilo da zbog nedostajućih vrijednosti izgleda kao da nema monotonih vrijednosti). Također, vidimo da nema ni konstantnih vrijednosti jer je broj jedinstvenih vrijednosti za svaku značajku veći od 1.

# Nedostajuće vrijednosti

data %>% is.na() %>% colSums()

## Age Sex ChestPainType RestingBP Cholesterol   
## 0 0 0 0 0   
## FastingBS RestingECG MaxHR ExerciseAngina Oldpeak   
## 0 0 0 0 0   
## ST\_Slope HeartDisease   
## 0 0

Nema nedostajućih vrijednosti, pa preskačemo korak zamjene nedostajućih vrijednosti

# Transformiranje kategoričkih značajki u numeričke

Mogli smo koristiti i one-hot encoding za neke značajke, no smatramo da ipak između svih značajki postoji neko uređenje te smo se zato odlučili za label encoding (npr. za značajku ChestPainType može biti TA -> ATA -> NAP -> ASY gdje određeni simptomi postaju manje karakteristični za anginu; ST\_Slope također ima uređenje up -> flat -> down)

cat(unique(data$Sex), "\n")

## M F

oldCategoricalData <- data  
data$Sex <- as.numeric(factor(data$Sex, levels = c("F", "M"), labels = c(0, 1))) - 1  
data$ExerciseAngina <- as.numeric(factor(data$ExerciseAngina, levels = c("N", "Y"), labels = c(0, 1))) - 1  
data$ChestPainType <- as.numeric(factor(data$ChestPainType, levels = c("TA", "ATA", "NAP", "ASY"), labels = c(0, 1, 2, 3)))  
data$RestingECG <- as.numeric(factor(data$RestingECG, levels = c("Normal", "ST", "LVH"), labels = c(0, 1, 2)))  
data$ST\_Slope <- as.numeric(factor(data$ST\_Slope, levels = c("Up", "Flat", "Down"), labels = c(0, 1, 2)))

# Nastavak deskriptivne statistike

Sada kad smo očistili podatke, možemo nastaviti s deskriptivnom statistikom. Prvo ćemo izvući osnovne statistike za podatke kao što su minimum, maksimum, kvantili i srednja vrijednost za numeričke podatke i učestalost pojavljivanja po kategorijama za kategoričke.

data[-c(2,3,6,7,9,11,12)] %>% summary()

## Age RestingBP Cholesterol MaxHR   
## Min. :28.00 Min. : 0.0 Min. : 0.0 Min. : 60.0   
## 1st Qu.:47.00 1st Qu.:120.0 1st Qu.:173.2 1st Qu.:120.0   
## Median :54.00 Median :130.0 Median :223.0 Median :138.0   
## Mean :53.51 Mean :132.4 Mean :198.8 Mean :136.8   
## 3rd Qu.:60.00 3rd Qu.:140.0 3rd Qu.:267.0 3rd Qu.:156.0   
## Max. :77.00 Max. :200.0 Max. :603.0 Max. :202.0   
## Oldpeak   
## Min. :-2.6000   
## 1st Qu.: 0.0000   
## Median : 0.6000   
## Mean : 0.8874   
## 3rd Qu.: 1.5000   
## Max. : 6.2000

# učestalost pojavljivanja vrijednosti za kategoričke značajke  
for (column in c(2, 3, 6, 7, 9, 11, 12)) {  
 column\_name <- colnames(oldCategoricalData)[column]  
 column\_values <- oldCategoricalData[[column]]  
 cat("Column:", column\_name, "has values:\n")  
 print(table(column\_values))  
 cat("\n")  
}

## Column: Sex has values:  
## column\_values  
## F M   
## 193 725   
##   
## Column: ChestPainType has values:  
## column\_values  
## ASY ATA NAP TA   
## 496 173 203 46   
##   
## Column: FastingBS has values:  
## column\_values  
## 0 1   
## 704 214   
##   
## Column: RestingECG has values:  
## column\_values  
## LVH Normal ST   
## 188 552 178   
##   
## Column: ExerciseAngina has values:  
## column\_values  
## N Y   
## 547 371   
##   
## Column: ST\_Slope has values:  
## column\_values  
## Down Flat Up   
## 63 460 395   
##   
## Column: HeartDisease has values:  
## column\_values  
## 0 1   
## 410 508

# Račun mjera centralne tendencije i rasipanja numeričkih podataka u raznim kombinacijama

## Analiza podataka ovisno o spolu

library(tidyverse)

## ── Attaching core tidyverse packages ──────────────────────── tidyverse 2.0.0 ──  
## ✔ forcats 1.0.0 ✔ stringr 1.5.1  
## ✔ lubridate 1.9.4 ✔ tibble 3.2.1  
## ✔ purrr 1.0.2   
## ── Conflicts ────────────────────────────────────────── tidyverse\_conflicts() ──  
## ✖ tidyr::extract() masks magrittr::extract()  
## ✖ dplyr::filter() masks stats::filter()  
## ✖ dplyr::lag() masks stats::lag()  
## ✖ purrr::set\_names() masks magrittr::set\_names()  
## ℹ Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors

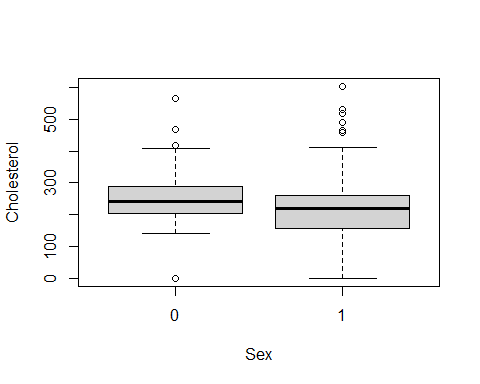
data %>% group\_by(Sex) %>% summarise(  
 NumOfPatients = n(),  
 HeartDiseaseRatio = sum(HeartDisease)/n(),  
 FastingBSRatio = sum(FastingBS)/n(),  
 ExerciseAnginaRatio = sum(ExerciseAngina)/n(),  
 Mean.Age = mean(Age),  
 Med.Age = median(Age),  
 Mean.RestingBP = mean(RestingBP),  
 Med.RestingBP = median(RestingBP),  
 StdDev.RestingBP = sd(RestingBP),  
 Mean.Cholesterol = mean(Cholesterol),  
 StdDev.Cholesterol = sd(Cholesterol),  
 Med.Cholesterol = median(Cholesterol),  
 Mean.MaxHR = mean(MaxHR),  
 StdDev.MaxHR = sd(MaxHR),  
 Med.MaxHR = median(MaxHR),  
 Mean.OldPeak = mean(Oldpeak),  
 StdDev.OldPeak = sd(Oldpeak),  
 Med.OldPeak = median(Oldpeak)  
)

## # A tibble: 2 × 19  
## Sex NumOfPatients HeartDiseaseRatio FastingBSRatio ExerciseAnginaRatio  
## <dbl> <int> <dbl> <dbl> <dbl>  
## 1 0 193 0.259 0.135 0.223  
## 2 1 725 0.632 0.259 0.452  
## # ℹ 14 more variables: Mean.Age <dbl>, Med.Age <dbl>, Mean.RestingBP <dbl>,  
## # Med.RestingBP <dbl>, StdDev.RestingBP <dbl>, Mean.Cholesterol <dbl>,  
## # StdDev.Cholesterol <dbl>, Med.Cholesterol <dbl>, Mean.MaxHR <dbl>,  
## # StdDev.MaxHR <dbl>, Med.MaxHR <dbl>, Mean.OldPeak <dbl>,  
## # StdDev.OldPeak <dbl>, Med.OldPeak <dbl>

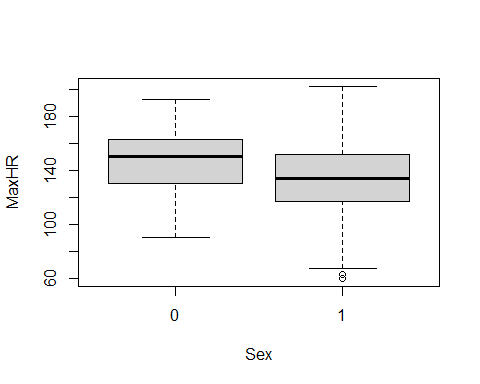
Na prvu uočavamo razlike između srednjih vrijednosti (a i medijana) najvećeg broja otkucaja srca (MaxHR) i količine kolesterola (Cholesterol), razliku u udjelu onih s visokim šećerom i vježbom induciranom anginom u muškaraca i žena, te, možda najvažnije, razliku u udjelu srčanih bolesnika među muškarcima i ženama. To bi moglo motivirati pitanja o utjecaju spola na vrijednosti tih parametara.

Možemo prvo analizirati boxplotovima.

boxplot(Cholesterol ~ Sex, data = data)



boxplot(MaxHR ~ Sex, data = data)



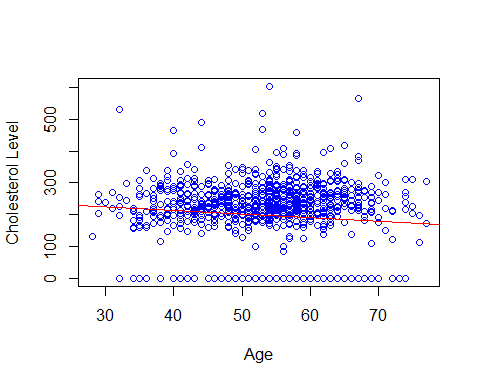
## Analiza podataka s obzirom na dob

library(tidyverse)  
  
helperData <- data  
dataWithAgeCategory <- helperData %>% mutate(ageCategory = cut(data$Age,  
 breaks = c(0, 30, 50, 70, 90, 120), # Define the break points  
 labels = c("Category-1", "Category-2", "Category-3", "Category-4", "Category-5"),   
 right = TRUE))  
  
  
dataWithAgeCategory %>% group\_by(ageCategory) %>% summarise(  
 NumOfPatients = n(),  
 HeartDiseaseRatio = sum(HeartDisease)/n(),  
 FastingBSRatio = sum(FastingBS)/n(),  
 ExerciseAnginaRatio = sum(ExerciseAngina)/n(),  
 Mean.Age = mean(Age),  
 Med.Age = median(Age),  
 Mean.RestingBP = mean(RestingBP),  
 Med.RestingBP = median(RestingBP),  
 StdDev.RestingBP = sd(RestingBP),  
 Mean.Cholesterol = mean(Cholesterol),  
 StdDev.Cholesterol = sd(Cholesterol),  
 Med.Cholesterol = median(Cholesterol),  
 Mean.MaxHR = mean(MaxHR),  
 StdDev.MaxHR = sd(MaxHR),  
 Med.MaxHR = median(MaxHR),  
 Mean.OldPeak = mean(Oldpeak),  
 StdDev.OldPeak = sd(Oldpeak),  
 Med.OldPeak = median(Oldpeak)  
)

## # A tibble: 4 × 19  
## ageCategory NumOfPatients HeartDiseaseRatio FastingBSRatio ExerciseAnginaRatio  
## <fct> <int> <dbl> <dbl> <dbl>  
## 1 Category-1 5 0 0 0   
## 2 Category-2 311 0.402 0.116 0.280  
## 3 Category-3 578 0.635 0.294 0.471  
## 4 Category-4 24 0.667 0.333 0.5   
## # ℹ 14 more variables: Mean.Age <dbl>, Med.Age <dbl>, Mean.RestingBP <dbl>,  
## # Med.RestingBP <dbl>, StdDev.RestingBP <dbl>, Mean.Cholesterol <dbl>,  
## # StdDev.Cholesterol <dbl>, Med.Cholesterol <dbl>, Mean.MaxHR <dbl>,  
## # StdDev.MaxHR <dbl>, Med.MaxHR <dbl>, Mean.OldPeak <dbl>,  
## # StdDev.OldPeak <dbl>, Med.OldPeak <dbl>

Ovdje uočavamo kako udio srčanih bolesnika, bolesnika s anginom i bolesnika s visokim šećerom raste svakom dobnom kategorijom, kao i vrijednosti OldPeak, dok MaxHR i Cholesterol padaju. To bi moglo potaknuti pitanja o ovisnosti dobi i vrijednosti tih parametara.

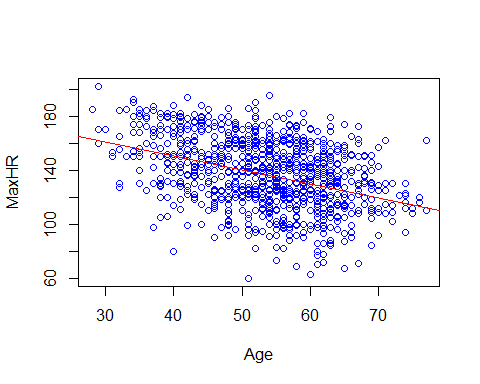
{plot(data$Age, data$Cholesterol,  
 col="blue",  
 xlab='Age',  
 ylab='Cholesterol Level')  
 abline(lm(data$Cholesterol ~ data$Age), col="red")}

 Iz ovog scatterplota vidimo gotovo nepostojeću korelaciju između dobi i razine kolesterola.

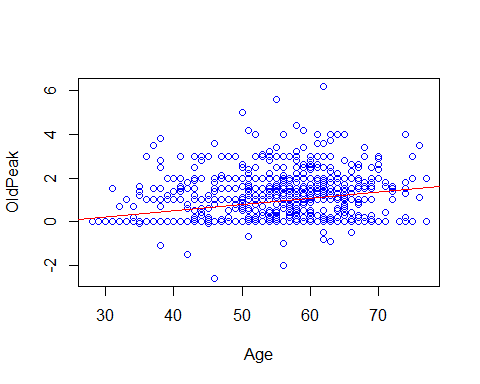
lowCholesterolCnt = nrow(data[data$Cholesterol < 120,])  
lowCholesterolCnt

## [1] 178

x <- data$Age  
y <- data$MaxHR  
  
model = lm(y ~ x)  
{plot(x, y,  
 col="blue",  
 xlab='Age',  
 ylab='MaxHR')  
abline(model, col = "red")}

 Ovaj scatterplot ukazuje na slabu negativnu korelaciju između vrijednosti parametara MaxHR i Age.

{plot(data$Age, data$Oldpeak,  
 col="blue",  
 xlab='Age',  
 ylab='OldPeak')  
  
abline(lm(data$Oldpeak ~ data$Age), col = "red")}

 Grafovi ukazuju na to da nam se najviše isplati postaviti pitanje o ovisnosti MaxHR o dobi (iako je ta korelacija također dosta blaga), jer ostale ovisnosti iz grafova izgledaju još više nekorelirano.

## Analiza ovisno o prisutnosti srčane bolesti

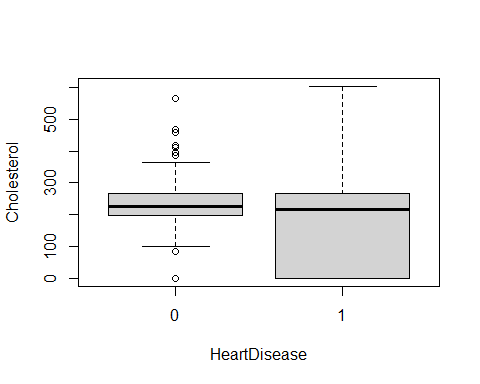
library(tidyverse)  
  
data %>% group\_by(HeartDisease) %>% summarise(  
 NumOfPatients = n(),  
 FastingBSRatio = sum(FastingBS)/n(),  
 ExerciseAnginaRatio = sum(ExerciseAngina)/n(),  
 Mean.Age = mean(Age),  
 Med.Age = median(Age),  
 Mean.RestingBP = mean(RestingBP),  
 Med.RestingBP = median(RestingBP),  
 StdDev.RestingBP = sd(RestingBP),  
 Mean.Cholesterol = mean(Cholesterol),  
 StdDev.Cholesterol = sd(Cholesterol),  
 Med.Cholesterol = median(Cholesterol),  
 Mean.MaxHR = mean(MaxHR),  
 StdDev.MaxHR = sd(MaxHR),  
 Med.MaxHR = median(MaxHR),  
 Mean.OldPeak = mean(Oldpeak),  
 StdDev.OldPeak = sd(Oldpeak),  
 Med.OldPeak = median(Oldpeak)  
)

## # A tibble: 2 × 18  
## HeartDisease NumOfPatients FastingBSRatio ExerciseAnginaRatio Mean.Age Med.Age  
## <dbl> <int> <dbl> <dbl> <dbl> <dbl>  
## 1 0 410 0.107 0.134 50.6 51  
## 2 1 508 0.335 0.622 55.9 57  
## # ℹ 12 more variables: Mean.RestingBP <dbl>, Med.RestingBP <dbl>,  
## # StdDev.RestingBP <dbl>, Mean.Cholesterol <dbl>, StdDev.Cholesterol <dbl>,  
## # Med.Cholesterol <dbl>, Mean.MaxHR <dbl>, StdDev.MaxHR <dbl>,  
## # Med.MaxHR <dbl>, Mean.OldPeak <dbl>, StdDev.OldPeak <dbl>,  
## # Med.OldPeak <dbl>

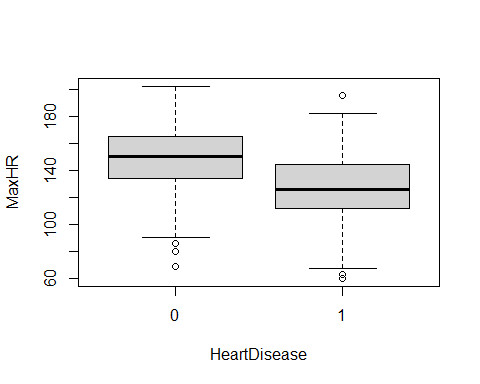
Ovdje konkretno vidimo izračunate mjere srednjih vrijednosti parametara za srčane bolesnike i one koji to nisu. Vidimo da se vrijednosti parametara Cholesterol, MaxHR i OldPeak razlikuju po grupama, kao i udio onih s povišenim šećerom i onih s vježbom induciranom anginom. To bi moglo motivirati pitanja može li se na temelju vrijednosti takvih parametara predvidjeti postojanje srčane bolesti.

Najbolje je to prvo analizirati boxplot dijagramima, paralelno za srčane bolesnike i zdrave pacijente.

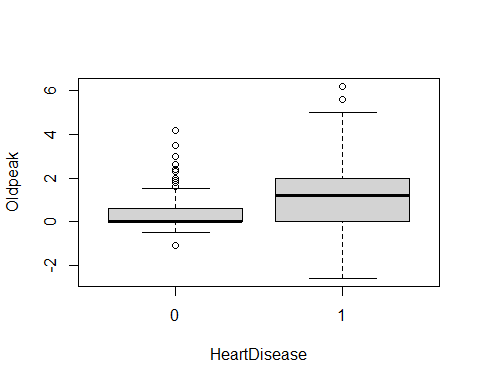
boxplot(Cholesterol ~ HeartDisease, data = data)

 Na ovom boxplotu vidimo kako se količine kolesterola više rasipaju u bolesnika te je njihova distribucija desno zakrivljena, ali u pacijenata koji nisu srčani bolesnici ima mnogo stršećih vrijednosti pa je dobro promatrati medijane umjesto srednjih vrijednosti za ovaj primjer.

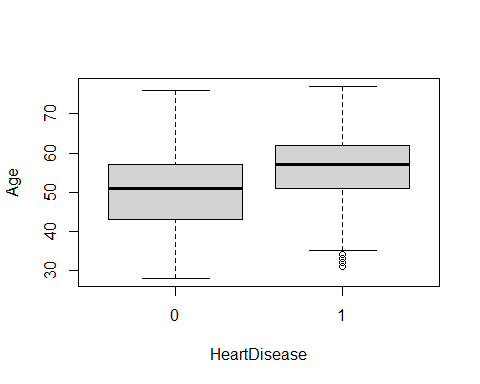
boxplot(MaxHR ~ HeartDisease, data = data)

 Ovaj bi boxplot mogao sugerirati na razliku u vrijednosti MaxHR parametra ovisno o tome je li pacijent srčani bolesnik ili ne - vidimo da za srčane bolesnike vrijedi da su vrijednosti MaxHR-a niže te da medijan bude manji te čak izvan vrijednosti prvog kvartila u skupini srčanih nebolesnika.

boxplot(Oldpeak ~ HeartDisease, data = data)

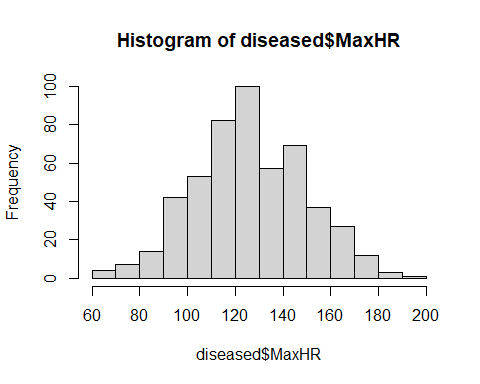
 Slično prošlom primjeri, ovaj boxplot mogao sugerirati na razliku u vrijednosti Oldpeak parametra ovisno o tome je li pacijent srčani bolesnik ili ne - vidimo da za srčane bolesnike vrijedi da su vrijednosti Oldpeaka-a više te da medijan bude veći te čak izvan vrijednosti trećeg kvartila u skupini srčanih nebolesnika. Slično kao za kolesterol, za srčane bolesnike se vrijednost Oldpeaka manje rasipa (bude konzistentnija), ali ponovno imamo mnogo stršećih (atipičnih vrijednosti), zato je dobro promatrati medijane umjesto srednjih vrijednosti za ovaj primjer.

boxplot(Age ~ HeartDisease, data = data)

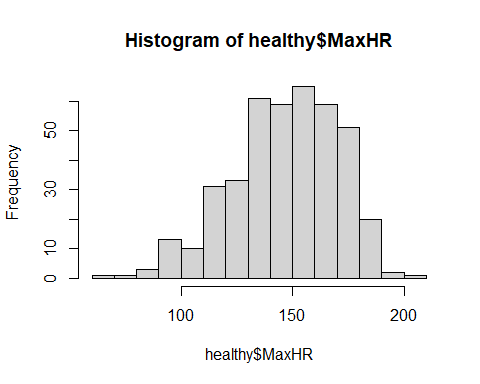
 Ponovno, ovaj boxplot mogao bi sugerirati razliku u dobi pacijenata srčanih bolesnika i nebolesnika - vidimo da za srčane bolesnike vrijedi da su veće dobi te da medijan bude veći te čak izvan vrijednosti trećeg kvartila u skupini srčanih nebolesnika.

diseased <- data[data$HeartDisease == 1, ]  
healthy <- data[data$HeartDisease == 0, ]

hist(diseased$MaxHR)



hist(healthy$MaxHR)



## Analiza ovisno o razini šećera

library(tidyverse)  
  
data %>% group\_by(FastingBS) %>% summarise(  
 NumOfPatients = n(),  
 HeartDiseaseRatio = sum(HeartDisease)/n(),  
 Mean.Age = mean(Age),  
 Med.Age = median(Age),  
 Mean.RestingBP = mean(RestingBP),  
 Med.RestingBP = median(RestingBP),  
 StdDev.RestingBP = sd(RestingBP),  
 Mean.Cholesterol = mean(Cholesterol),  
 StdDev.Cholesterol = sd(Cholesterol),  
 Med.Cholesterol = median(Cholesterol),  
 Mean.MaxHR = mean(MaxHR),  
 StdDev.MaxHR = sd(MaxHR),  
 Med.MaxHR = median(MaxHR),  
 Mean.OldPeak = mean(Oldpeak),  
 StdDev.OldPeak = sd(Oldpeak),  
 Med.OldPeak = median(Oldpeak)  
)

## # A tibble: 2 × 17  
## FastingBS NumOfPatients HeartDiseaseRatio Mean.Age Med.Age Mean.RestingBP  
## <dbl> <int> <dbl> <dbl> <dbl> <dbl>  
## 1 0 704 0.480 52.5 53 132.  
## 2 1 214 0.794 56.9 57 135.  
## # ℹ 11 more variables: Med.RestingBP <dbl>, StdDev.RestingBP <dbl>,  
## # Mean.Cholesterol <dbl>, StdDev.Cholesterol <dbl>, Med.Cholesterol <dbl>,  
## # Mean.MaxHR <dbl>, StdDev.MaxHR <dbl>, Med.MaxHR <dbl>, Mean.OldPeak <dbl>,  
## # StdDev.OldPeak <dbl>, Med.OldPeak <dbl>

# Normalizacija numeričkih značajki

unNormalizedData <- data  
data[-c(2,3,6,7,9,11,12)] <- scale(data[-c(2,3,6,7,9,11,12)])

### ISTRAŽIVAČKA PITANJA

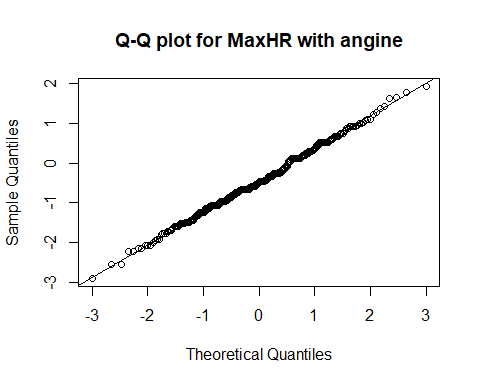
VAŽNA NAPOMENA: Alfa vrijednost određuje se prije provođenja statističkog testa; mi ćemo u svim testovima uzeti da je alfa = 0.05 odnosno 5%.

## 1. pitanje

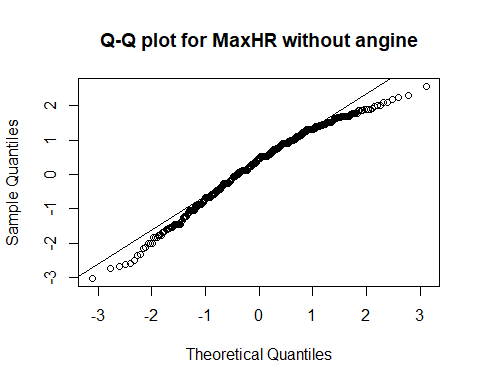
# Postoji li razlika u maksimalnom broju otkucaja srca između pacijenata s anginom i onih bez angine?

Kako bismo odgovorili na ovo pitanje trebamo koristiti T-test za dva uzorka. Znamo da T-test ima neke osnovne pretpostavke koje moramo provjeriti, a moramo provjeriti i jesu li varijance podataka jednake ili različite. Prvo ćemo pogledati jesu li podatci uzorkovani iz normalne razdiobe, a nakon toga provesti F-test kako bismo utvrdili jesu li varijance jednake ili različite.

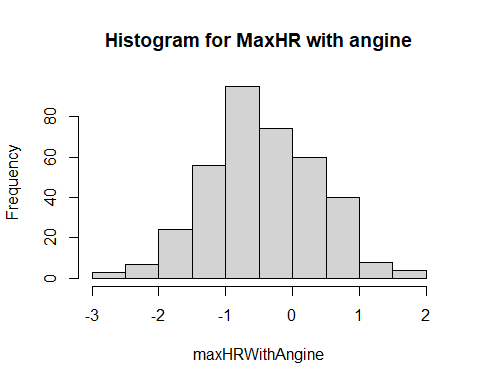
withAngine <- data[data$ExerciseAngina == 1,]  
withoutAngine <- data[data$ExerciseAngina == 0,]  
maxHRWithAngine <- withAngine$MaxHR  
maxHRWithoutAngine <- withoutAngine$MaxHR  
  
# crtamo Q-Q plot kako bismo vidjeli odgovaraju li podatci normalnoj raspodjeli  
qqnorm(maxHRWithAngine, main = "Q-Q plot for MaxHR with angine")  
qqline(maxHRWithAngine)



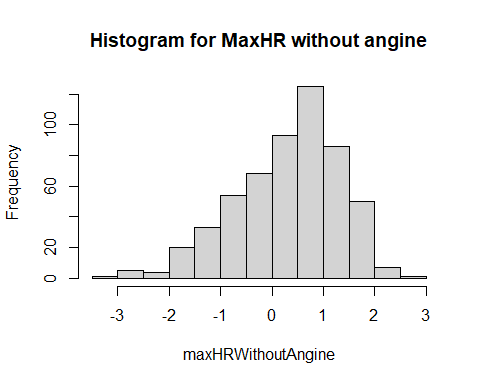
qqnorm(maxHRWithoutAngine , main = "Q-Q plot for MaxHR without angine")  
qqline(maxHRWithoutAngine)



# isto radimo s histogramima  
hist(maxHRWithAngine, main = "Histogram for MaxHR with angine")



hist(maxHRWithoutAngine, main = "Histogram for MaxHR without angine")



Iz grafova možemo zaključiti da se za osobe s anginom značajka MaxHR ravna po normalnoj razdiobi, dok se za osobe bez angine sredina ravna po normalnoj raspodjeli, no repovi su različiti od normalne raspodjele. Dodatno, histogram za osobe bez angine pokazuje da su podatci malo pomaknuti od sredine 0. Možemo provesti Lillieforsov test kako bi provjerili odgovaraju li podatci normalnoj razdiobi. Ipak, T razdioba robusna je na normalnost podataka, pa možemo nastaviti s testiranjem razlike u srednjim vrijednostima. Pretpostavku nezavisnosti podataka temeljimo na činjenici da su podatci preuzeti iz različitih skupina odnosno od različitih ljudi.

library(nortest)  
lillie.test(maxHRWithoutAngine)

##   
## Lilliefors (Kolmogorov-Smirnov) normality test  
##   
## data: maxHRWithoutAngine  
## D = 0.075696, p-value = 6.45e-08

lillie.test(maxHRWithAngine)

##   
## Lilliefors (Kolmogorov-Smirnov) normality test  
##   
## data: maxHRWithAngine  
## D = 0.054536, p-value = 0.009992

Nastavljamo s F-testom kako bismo utvrdili jesu li varijance obaju skupova podataka jednake ili različite.

var(maxHRWithoutAngine)

## [1] 1.011808

var(maxHRWithAngine)

## [1] 0.645209

var.test(maxHRWithAngine, maxHRWithoutAngine)

##   
## F test to compare two variances  
##   
## data: maxHRWithAngine and maxHRWithoutAngine  
## F = 0.63768, num df = 370, denom df = 546, p-value = 3.641e-06  
## alternative hypothesis: true ratio of variances is not equal to 1  
## 95 percent confidence interval:  
## 0.5298767 0.7700287  
## sample estimates:  
## ratio of variances   
## 0.6376795

Vidimo kako je p-vrijednost značajno manja od alfa = 5% pa zaključujemo da su varijance različite; radi toga koristimo T-test za dva uzorka s različitim varijancama.

t.test(maxHRWithAngine, maxHRWithoutAngine, var.equal = FALSE)

##   
## Welch Two Sample t-test  
##   
## data: maxHRWithAngine and maxHRWithoutAngine  
## t = -12.594, df = 891.9, p-value < 2.2e-16  
## alternative hypothesis: true difference in means is not equal to 0  
## 95 percent confidence interval:  
## -0.8720165 -0.6368663  
## sample estimates:  
## mean of x mean of y   
## -0.4495419 0.3048995

Iz rezultata vidimo kako je p-vrijednost značajno manja od alfa = 5% pa zaključujemo da postoji razlika u maksimalnom broju otkucaja srca između pacijenata s anginom i onih bez angine

## 2. pitanje

# Postoji li značajna razlika u starosti pacijenata s obzirom na različite vrste boli u prsima?

## 3. pitanje

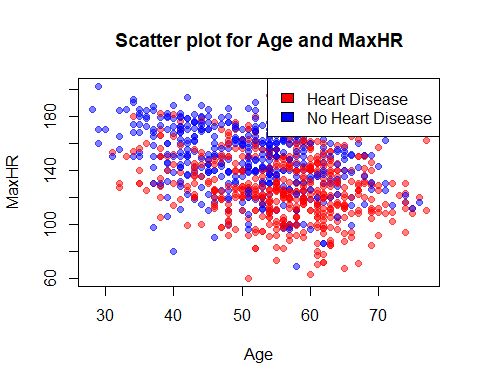
# Možemo li predvidjeti prisutnost srčane bolesti na temelju maksimalnog broja otkucaja srca i starosti?

Za pronalazak odgovora na treće pitanje koristimo logističku regresiju. Linearna regresija nije dobar izbor jer je problem klasifikacijski. Linearna regresija loše klasificira jer funkcija pogreške kažnjava i točno klasificirane primjere koji su daleko od granice odluke te zbog toga stršeće vrijednosti mogu značajno utjecati na pomak granice odluke (čak i kada su točno klasificirane).

Za logističku regresiju trebamo provjeriti nekoliko pretpostavki: 1. Ne smije biti multikolinearnosti u podatcima 2. Podatci moraju biti nezavisni (ulazi su nezavisne varijable dok je izlaz y zavisna varijabla) 3. Nema nedostajućih vrijednosti i klase imaju balansirani broj primjera (u našem slučaju oba uvjeta su ispunjena, provjera je napravljena u fazi čišćenja podataka)

Prije provedbe logističke regresije poželjno je vizualizirati podatke kako bismo dobili bolji uvid u njih. U našem slučaju možemo koristiti graf raspršenja jer imamo samo dvije varijable.

colors <- ifelse(oldCategoricalData$HeartDisease == 1, rgb(1,0,0,0.5), rgb(0,0,1,0.5))   
plot(oldCategoricalData$Age, oldCategoricalData$MaxHR, col = colors, main = "Scatter plot for Age and MaxHR", xlab = "Age", ylab = "MaxHR", pch = 19)  
legend("topright", legend = c("Heart Disease", "No Heart Disease"), fill = c("red", "blue"))

 Iz grafa je vidljivo kako su srčane bolesti rjeđe kod mlađih ljudi koji imaju veći maksimalan broj otkucaja srca, dok stariji ljudi s manjim maksimalnim brojem otkucaja srca češće imaju srčane bolesti.

# Provjera pretpostavki logističke regresije

Funkcijom cor tražimo korelaciju između naših dviju varijabli.

library(ggcorrplot)  
  
predictor <- data.frame(data$Age, data$MaxHR)  
cor(predictor)

## data.Age data.MaxHR  
## data.Age 1.0000000 -0.3820447  
## data.MaxHR -0.3820447 1.0000000

#ggcorrplot(cor(predictor), hc.order = TRUE, type = "lower", lab = TRUE)

Dobili smo korelaciju od -0.38 što je relativno slaba korelacija, pa možemo nastaviti s provođenjem logističke regresije.

logisticModel <- glm(HeartDisease ~ MaxHR + Age, data = data, family = binomial)  
summary(logisticModel)

##   
## Call:  
## glm(formula = HeartDisease ~ MaxHR + Age, family = binomial,   
## data = data)  
##   
## Coefficients:  
## Estimate Std. Error z value Pr(>|z|)   
## (Intercept) 0.25847 0.07373 3.505 0.000456 \*\*\*  
## MaxHR -0.82458 0.08708 -9.469 < 2e-16 \*\*\*  
## Age 0.35662 0.08012 4.451 8.55e-06 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## (Dispersion parameter for binomial family taken to be 1)  
##   
## Null deviance: 1262.1 on 917 degrees of freedom  
## Residual deviance: 1082.6 on 915 degrees of freedom  
## AIC: 1088.6  
##   
## Number of Fisher Scoring iterations: 3

Nakon što smo proveli logističku regresiju možemo analizirati dobivene rezultate.

Vidimo kako je koeficijent MaxHR jednak -0.83 što sugerira da se s povećanjem maksimalnog broja otkucaja srca smanjuje vjerojatnost imanja srčane bolesti. Koeficijent uz Age je 0.3566 što sugerira da starije osobe s većom vjerojatnošću imaju srčanu bolest. Dobiveni rezultati slažu se s vizualizacijom podataka koju smo napravili ranije.

P-vrijednosti su male: 2 \* 10^-16 za MaxHR i 8.55 \* 10^-6 za Age, iz čega možemo zaključiti kako oba koeficijenta imaju statistički značajan utjecaj.

# Predikcija

library(pROC)

## Type 'citation("pROC")' for a citation.

##   
## Attaching package: 'pROC'

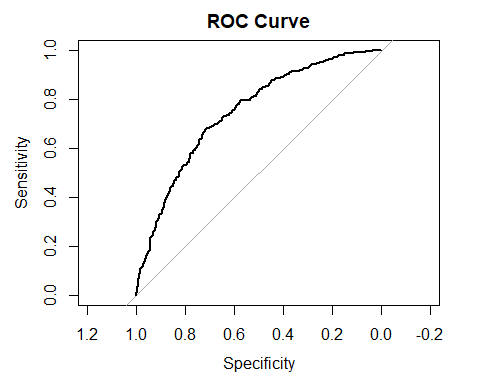
## The following objects are masked from 'package:stats':  
##   
## cov, smooth, var

# Predikcije  
predictions <- predict(logisticModel, type = "response")  
# ROC krivulja  
roc\_curve <- roc(data$HeartDisease, predictions)

## Setting levels: control = 0, case = 1

## Setting direction: controls < cases

plot(roc\_curve, main = "ROC Curve")



# AUC vrijednost  
auc(roc\_curve)

## Area under the curve: 0.7467

Tzv. “area under the curve” predstavlja površinu ispod ROC krivulje. Znamo kako je ROC krivulja jednostavno pravac pod kutem od 45 stupnjeva ako je predikcija potpuno slučajna. Što je veća površina ispod grafa, bolja je sposobnost predikcije. Idealno je AUC jednak 1.

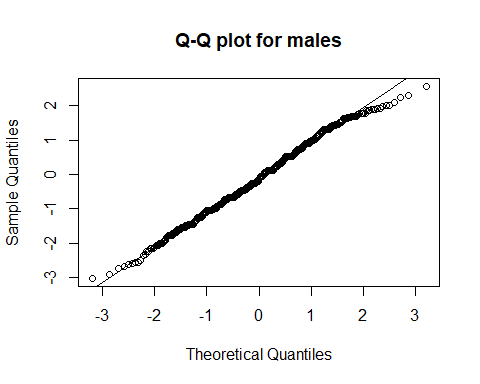
Za naš slučaj AUC je 0.7467, iz čega zaključujemo kako naš model ima dobru sposobnost predikcije. Poboljšanje rezultata eventualno bismo mogli postići tako da bismo u predikciju dodali još relevatnih značajki.

## 4. pitanje

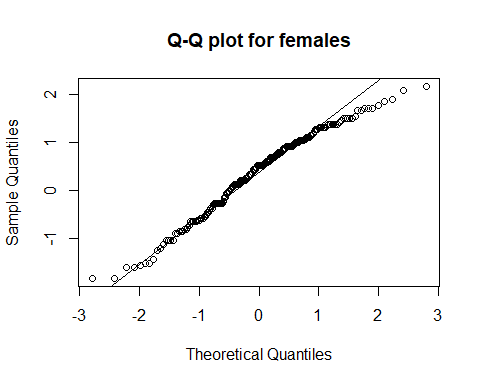
# Postoji li razlika u razini kolesterola između muškaraca i žena?

Kao i u prvom istraživačkom pitanju koristimo T-test za dva uzorka. Opet ćemo prvo provjeriti normalnost podataka i nakon toga varijanci pomoću F-testa. Opet pretpostavljamo nezavisnost podataka jer su podatci uzeti iz različitih skupina.

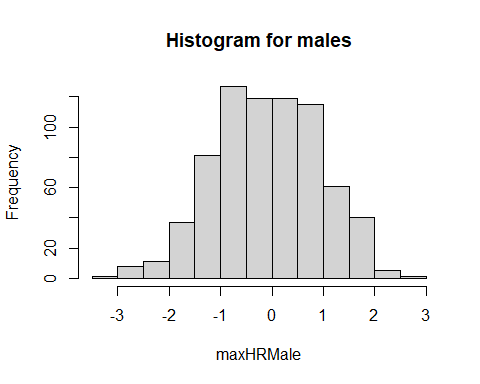
# crtamo Q-Q plot da vidimo odgovaraju li podatci normalnoj raspodjeli  
male <- data[data$Sex == 1,]  
female <- data[data$Sex == 0,]  
  
maxHRMale <- male$MaxHR  
maxHRFemale <- female$MaxHR  
  
qqnorm(maxHRMale, main = "Q-Q plot for males")  
qqline(maxHRMale)



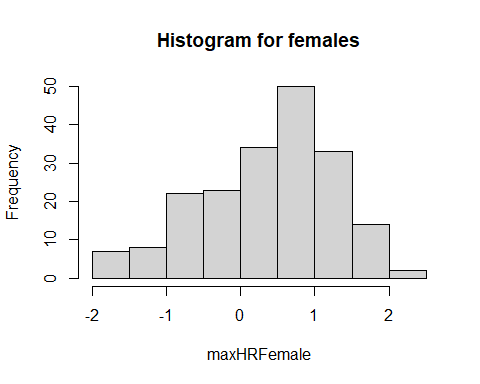
qqnorm(maxHRFemale , main = "Q-Q plot for females")  
qqline(maxHRFemale)



# histogrami  
  
hist(maxHRMale, main = "Histogram for males")



hist(maxHRFemale, main = "Histogram for females")

 Vidimo kako se podatci ne ravnaju savršeno po normalnoj razdiobi na repovima ali sličnost je dovoljno velika da bi mogli nastaviti s provođenjem testa (mala odstupanja od normalne razdiobe neće utjecati na rezultate zato što je T-test robusan na normalnost podataka).

Nakon što smo utvrdili nezavisnost i normalnost podataka, provodimo F-test kako bismo utvrdili jesu li varijance jednake ili različite.

var(maxHRMale)

## [1] 1.020437

var(maxHRFemale)

## [1] 0.7572015

var.test(maxHRMale, maxHRFemale)

##   
## F test to compare two variances  
##   
## data: maxHRMale and maxHRFemale  
## F = 1.3476, num df = 724, denom df = 192, p-value = 0.01245  
## alternative hypothesis: true ratio of variances is not equal to 1  
## 95 percent confidence interval:  
## 1.067533 1.676004  
## sample estimates:  
## ratio of variances   
## 1.347643

Opet vidimo kako je p-vrijednost manja od alfa = 5% pa zaključujemo da su varijance različite. Sada znamo da trebamo koristiti T-test za dva uzorka s različitim varijancama.

t.test(maxHRMale, maxHRFemale, var.equal = FALSE)

##   
## Welch Two Sample t-test  
##   
## data: maxHRMale and maxHRFemale  
## t = -6.3555, df = 342.77, p-value = 6.621e-10  
## alternative hypothesis: true difference in means is not equal to 0  
## 95 percent confidence interval:  
## -0.6076397 -0.3204219  
## sample estimates:  
## mean of x mean of y   
## -0.09755767 0.36647312

Iz rezultata vidimo kako je p-vrijednost značajno manja od alfa = 5% pa zaključujemo da postoji razlika u maksimalnom broju otkucaja srca između muškaraca i žena.

## 5. pitanje

# Postoji li korelacija ili homogenost između spola i kategorijskih čimbenika?

data\_copy = data.frame(data);  
tracemem(data)==tracemem(data\_copy);

## [1] FALSE

untracemem(data\_copy);

tbl\_angina = table(data\_copy$Sex)  
print(tbl\_angina)

##   
## 0 1   
## 193 725

tbl\_angina = table(data\_copy$Sex,   
 data\_copy$ExerciseAngina)  
tbl\_angina

##   
## 0 1  
## 0 150 43  
## 1 397 328

added\_margins\_tbl\_angina = addmargins(tbl\_angina)  
print(added\_margins\_tbl\_angina)

##   
## 0 1 Sum  
## 0 150 43 193  
## 1 397 328 725  
## Sum 547 371 918

for (col\_names in colnames(added\_margins\_tbl\_angina)){  
 for (row\_names in rownames(added\_margins\_tbl\_angina)){  
 if (!(row\_names == 'Sum' | col\_names == 'Sum') ){  
 cat('Očekivane frekvencije za razred ',col\_names,'-',row\_names,': ',(added\_margins\_tbl\_angina[row\_names,'Sum'] \* added\_margins\_tbl\_angina['Sum',col\_names]) / added\_margins\_tbl\_angina['Sum','Sum'],'\n')  
 }  
 }  
}

## Očekivane frekvencije za razred 0 - 0 : 115.0011   
## Očekivane frekvencije za razred 0 - 1 : 431.9989   
## Očekivane frekvencije za razred 1 - 0 : 77.99891   
## Očekivane frekvencije za razred 1 - 1 : 293.0011

chisq.test(tbl\_angina, correct=F)

##   
## Pearson's Chi-squared test  
##   
## data: tbl\_angina  
## X-squared = 33.372, df = 1, p-value = 7.612e-09

Vidimo da je p vrijednost jako mala što znači da odbacujemo hipotezu H0 u korist hipoteze H1 koja glasi da postoji zavisnost između podataka. Tj. da pojava angine izazvane vježbanjem ovisi o spolu.

data\_copy = data.frame(data);

## tracemem[0x000001f798a6adf8 -> 0x000001f7a35bef88]: as.data.frame.spec\_tbl\_df as.data.frame data.frame eval eval withVisible withCallingHandlers eval eval with\_handlers doWithOneRestart withOneRestart withRestartList doWithOneRestart withOneRestart withRestartList doWithOneRestart withOneRestart withRestartList withRestarts <Anonymous> evaluate in\_dir in\_input\_dir eng\_r block\_exec call\_block process\_group withCallingHandlers <Anonymous> process\_file <Anonymous> <Anonymous>   
## tracemem[0x000001f7a35bef88 -> 0x000001f7a35beed8]: as.data.frame.tbl\_df NextMethod as.data.frame.spec\_tbl\_df as.data.frame data.frame eval eval withVisible withCallingHandlers eval eval with\_handlers doWithOneRestart withOneRestart withRestartList doWithOneRestart withOneRestart withRestartList doWithOneRestart withOneRestart withRestartList withRestarts <Anonymous> evaluate in\_dir in\_input\_dir eng\_r block\_exec call\_block process\_group withCallingHandlers <Anonymous> process\_file <Anonymous> <Anonymous>   
## tracemem[0x000001f7a35beed8 -> 0x000001f7a35bee28]: as.list.data.frame as.list vapply map\_mold map\_lgl which as.data.frame.tbl\_df NextMethod as.data.frame.spec\_tbl\_df as.data.frame data.frame eval eval withVisible withCallingHandlers eval eval with\_handlers doWithOneRestart withOneRestart withRestartList doWithOneRestart withOneRestart withRestartList doWithOneRestart withOneRestart withRestartList withRestarts <Anonymous> evaluate in\_dir in\_input\_dir eng\_r block\_exec call\_block process\_group withCallingHandlers <Anonymous> process\_file <Anonymous> <Anonymous>   
## tracemem[0x000001f7a35beed8 -> 0x000001f7a35beab8]: tbl\_subassign [<-.tbl\_df [<- as.data.frame.tbl\_df NextMethod as.data.frame.spec\_tbl\_df as.data.frame data.frame eval eval withVisible withCallingHandlers eval eval with\_handlers doWithOneRestart withOneRestart withRestartList doWithOneRestart withOneRestart withRestartList doWithOneRestart withOneRestart withRestartList withRestarts <Anonymous> evaluate in\_dir in\_input\_dir eng\_r block\_exec call\_block process\_group withCallingHandlers <Anonymous> process\_file <Anonymous> <Anonymous>   
## tracemem[0x000001f7a35beab8 -> 0x000001f7a35be698]: tbl\_subassign\_col tbl\_subassign [<-.tbl\_df [<- as.data.frame.tbl\_df NextMethod as.data.frame.spec\_tbl\_df as.data.frame data.frame eval eval withVisible withCallingHandlers eval eval with\_handlers doWithOneRestart withOneRestart withRestartList doWithOneRestart withOneRestart withRestartList doWithOneRestart withOneRestart withRestartList withRestarts <Anonymous> evaluate in\_dir in\_input\_dir eng\_r block\_exec call\_block process\_group withCallingHandlers <Anonymous> process\_file <Anonymous> <Anonymous>   
## tracemem[0x000001f7a35be698 -> 0x000001f7a35be5e8]: vectbl\_restore tbl\_subassign [<-.tbl\_df [<- as.data.frame.tbl\_df NextMethod as.data.frame.spec\_tbl\_df as.data.frame data.frame eval eval withVisible withCallingHandlers eval eval with\_handlers doWithOneRestart withOneRestart withRestartList doWithOneRestart withOneRestart withRestartList doWithOneRestart withOneRestart withRestartList withRestarts <Anonymous> evaluate in\_dir in\_input\_dir eng\_r block\_exec call\_block process\_group withCallingHandlers <Anonymous> process\_file <Anonymous> <Anonymous>

tracemem(data)==tracemem(data\_copy);

## [1] FALSE

untracemem(data\_copy);

tbl\_heart\_disease = table(data\_copy$Sex)  
print(tbl\_heart\_disease)

##   
## 0 1   
## 193 725

tbl\_heart\_disease = table(data\_copy$Sex,   
 data\_copy$HeartDisease)  
tbl\_heart\_disease

##   
## 0 1  
## 0 143 50  
## 1 267 458

added\_margins\_tbl\_heart\_disease = addmargins(tbl\_heart\_disease)  
print(added\_margins\_tbl\_heart\_disease)

##   
## 0 1 Sum  
## 0 143 50 193  
## 1 267 458 725  
## Sum 410 508 918

for (col\_names in colnames(added\_margins\_tbl\_heart\_disease)){  
 for (row\_names in rownames(added\_margins\_tbl\_heart\_disease)){  
 if (!(row\_names == 'Sum' | col\_names == 'Sum') ){  
 cat('Očekivane frekvencije za razred ',col\_names,'-',row\_names,': ',(added\_margins\_tbl\_heart\_disease[row\_names,'Sum'] \* added\_margins\_tbl\_heart\_disease['Sum',col\_names]) / added\_margins\_tbl\_heart\_disease['Sum','Sum'],'\n')  
 }  
 }  
}

## Očekivane frekvencije za razred 0 - 0 : 86.19826   
## Očekivane frekvencije za razred 0 - 1 : 323.8017   
## Očekivane frekvencije za razred 1 - 0 : 106.8017   
## Očekivane frekvencije za razred 1 - 1 : 401.1983

chisq.test(tbl\_heart\_disease, correct=F)

##   
## Pearson's Chi-squared test  
##   
## data: tbl\_heart\_disease  
## X-squared = 85.646, df = 1, p-value < 2.2e-16

Vidimo da je p-vrijednost jako mala što znači da odbacujemo hipotezu H0 u korist H1 koja tvrdi da postoji zavisnost između pojave srčanih bolesti i spola.

data\_copy = data.frame(data);

## tracemem[0x000001f798a6adf8 -> 0x000001f7a4e6d7f8]: as.data.frame.spec\_tbl\_df as.data.frame data.frame eval eval withVisible withCallingHandlers eval eval with\_handlers doWithOneRestart withOneRestart withRestartList doWithOneRestart withOneRestart withRestartList doWithOneRestart withOneRestart withRestartList withRestarts <Anonymous> evaluate in\_dir in\_input\_dir eng\_r block\_exec call\_block process\_group withCallingHandlers <Anonymous> process\_file <Anonymous> <Anonymous>   
## tracemem[0x000001f7a4e6d7f8 -> 0x000001f7a4e6d748]: as.data.frame.tbl\_df NextMethod as.data.frame.spec\_tbl\_df as.data.frame data.frame eval eval withVisible withCallingHandlers eval eval with\_handlers doWithOneRestart withOneRestart withRestartList doWithOneRestart withOneRestart withRestartList doWithOneRestart withOneRestart withRestartList withRestarts <Anonymous> evaluate in\_dir in\_input\_dir eng\_r block\_exec call\_block process\_group withCallingHandlers <Anonymous> process\_file <Anonymous> <Anonymous>   
## tracemem[0x000001f7a4e6d748 -> 0x000001f7a4e6d698]: as.list.data.frame as.list vapply map\_mold map\_lgl which as.data.frame.tbl\_df NextMethod as.data.frame.spec\_tbl\_df as.data.frame data.frame eval eval withVisible withCallingHandlers eval eval with\_handlers doWithOneRestart withOneRestart withRestartList doWithOneRestart withOneRestart withRestartList doWithOneRestart withOneRestart withRestartList withRestarts <Anonymous> evaluate in\_dir in\_input\_dir eng\_r block\_exec call\_block process\_group withCallingHandlers <Anonymous> process\_file <Anonymous> <Anonymous>   
## tracemem[0x000001f7a4e6d748 -> 0x000001f7a4e6d328]: tbl\_subassign [<-.tbl\_df [<- as.data.frame.tbl\_df NextMethod as.data.frame.spec\_tbl\_df as.data.frame data.frame eval eval withVisible withCallingHandlers eval eval with\_handlers doWithOneRestart withOneRestart withRestartList doWithOneRestart withOneRestart withRestartList doWithOneRestart withOneRestart withRestartList withRestarts <Anonymous> evaluate in\_dir in\_input\_dir eng\_r block\_exec call\_block process\_group withCallingHandlers <Anonymous> process\_file <Anonymous> <Anonymous>   
## tracemem[0x000001f7a4e6d328 -> 0x000001f7a5007dd8]: tbl\_subassign\_col tbl\_subassign [<-.tbl\_df [<- as.data.frame.tbl\_df NextMethod as.data.frame.spec\_tbl\_df as.data.frame data.frame eval eval withVisible withCallingHandlers eval eval with\_handlers doWithOneRestart withOneRestart withRestartList doWithOneRestart withOneRestart withRestartList doWithOneRestart withOneRestart withRestartList withRestarts <Anonymous> evaluate in\_dir in\_input\_dir eng\_r block\_exec call\_block process\_group withCallingHandlers <Anonymous> process\_file <Anonymous> <Anonymous>   
## tracemem[0x000001f7a5007dd8 -> 0x000001f7a5007d28]: vectbl\_restore tbl\_subassign [<-.tbl\_df [<- as.data.frame.tbl\_df NextMethod as.data.frame.spec\_tbl\_df as.data.frame data.frame eval eval withVisible withCallingHandlers eval eval with\_handlers doWithOneRestart withOneRestart withRestartList doWithOneRestart withOneRestart withRestartList doWithOneRestart withOneRestart withRestartList withRestarts <Anonymous> evaluate in\_dir in\_input\_dir eng\_r block\_exec call\_block process\_group withCallingHandlers <Anonymous> process\_file <Anonymous> <Anonymous>

tracemem(data)==tracemem(data\_copy);

## [1] FALSE

untracemem(data\_copy);

tbl\_pain\_type = table(data\_copy$Sex)  
print(tbl\_pain\_type)

##   
## 0 1   
## 193 725

tbl\_pain\_type = table(data\_copy$Sex,   
 data\_copy$ChestPainType)  
tbl\_pain\_type

##   
## 1 2 3 4  
## 0 10 60 53 70  
## 1 36 113 150 426

added\_margins\_tbl\_pain\_type = addmargins(tbl\_pain\_type)  
print(added\_margins\_tbl\_pain\_type)

##   
## 1 2 3 4 Sum  
## 0 10 60 53 70 193  
## 1 36 113 150 426 725  
## Sum 46 173 203 496 918

for (col\_names in colnames(added\_margins\_tbl\_pain\_type)){  
 for (row\_names in rownames(added\_margins\_tbl\_pain\_type)){  
 if (!(row\_names == 'Sum' | col\_names == 'Sum') ){  
 cat('Očekivane frekvencije za razred ',col\_names,'-',row\_names,': ',(added\_margins\_tbl\_pain\_type[row\_names,'Sum'] \* added\_margins\_tbl\_pain\_type['Sum',col\_names]) / added\_margins\_tbl\_pain\_type['Sum','Sum'],'\n')  
 }  
 }  
}

## Očekivane frekvencije za razred 1 - 0 : 9.671024   
## Očekivane frekvencije za razred 1 - 1 : 36.32898   
## Očekivane frekvencije za razred 2 - 0 : 36.37146   
## Očekivane frekvencije za razred 2 - 1 : 136.6285   
## Očekivane frekvencije za razred 3 - 0 : 42.67865   
## Očekivane frekvencije za razred 3 - 1 : 160.3214   
## Očekivane frekvencije za razred 4 - 0 : 104.2789   
## Očekivane frekvencije za razred 4 - 1 : 391.7211

chisq.test(tbl\_pain\_type, correct=F)

##   
## Pearson's Chi-squared test  
##   
## data: tbl\_pain\_type  
## X-squared = 36.879, df = 3, p-value = 4.88e-08

Ponovno vidimo da je p vrijednost jako mala što znači da odbijamo hipotezu H0 u korist H1 koja tvrdi da postoji povezanost između tipa boli i spola.