HeartAttack&Prediction Analysis

Malay Raj

2023-09-26

## HEART ATTACK & PREDICTION ANALYSIS:

This dataset contains information about people’s health, including details such as age, sex, chest pain, etc.

## Loading Required Packages.

library(tidyverse)

## Warning: package 'tidyverse' was built under R version 4.2.3

## Warning: package 'ggplot2' was built under R version 4.2.3

## ── Attaching core tidyverse packages ──────────────────────── tidyverse 2.0.0 ──  
## ✔ dplyr 1.1.0 ✔ readr 2.1.4  
## ✔ forcats 1.0.0 ✔ stringr 1.5.0  
## ✔ ggplot2 3.4.2 ✔ tibble 3.1.8  
## ✔ lubridate 1.9.2 ✔ tidyr 1.3.0  
## ✔ purrr 1.0.1   
## ── Conflicts ────────────────────────────────────────── tidyverse\_conflicts() ──  
## ✖ dplyr::filter() masks stats::filter()  
## ✖ dplyr::lag() masks stats::lag()  
## ℹ Use the ]8;;http://conflicted.r-lib.org/conflicted package]8;; to force all conflicts to become errors

library(janitor)

## Warning: package 'janitor' was built under R version 4.2.3

##   
## Attaching package: 'janitor'  
##   
## The following objects are masked from 'package:stats':  
##   
## chisq.test, fisher.test

library(dplyr)  
library(ggplot2)  
library(skimr)

## Warning: package 'skimr' was built under R version 4.2.3

library(plyr)

## ------------------------------------------------------------------------------  
## You have loaded plyr after dplyr - this is likely to cause problems.  
## If you need functions from both plyr and dplyr, please load plyr first, then dplyr:  
## library(plyr); library(dplyr)  
## ------------------------------------------------------------------------------  
##   
## Attaching package: 'plyr'  
##   
## The following objects are masked from 'package:dplyr':  
##   
## arrange, count, desc, failwith, id, mutate, rename, summarise,  
## summarize  
##   
## The following object is masked from 'package:purrr':  
##   
## compact

library(lubridate)  
library(plotly)

## Warning: package 'plotly' was built under R version 4.2.3

##   
## Attaching package: 'plotly'  
##   
## The following objects are masked from 'package:plyr':  
##   
## arrange, mutate, rename, summarise  
##   
## The following object is masked from 'package:ggplot2':  
##   
## last\_plot  
##   
## The following object is masked from 'package:stats':  
##   
## filter  
##   
## The following object is masked from 'package:graphics':  
##   
## layout

library(gridExtra)

## Warning: package 'gridExtra' was built under R version 4.2.3

##   
## Attaching package: 'gridExtra'  
##   
## The following object is masked from 'package:dplyr':  
##   
## combine

## Displaying my current working directory.

getwd()

## [1] "C:/Heart Attack & Prediction Analysis"

## Collecting the data.

heart = read.csv("C:/Heart Attack & Prediction Analysis/heart.csv")

## Displaying the first six rows of dataset.

head(heart)

## 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  
## 6 57 1 0 140 192 0 1 148 0 0.4 1 0 1 1

## Checking the dimension of the dataset.

dim(heart)

## [1] 303 14

## Displaying the column names of our dataframe.

colnames(heart)

## [1] "age" "sex" "cp" "trtbps" "chol" "fbs"   
## [7] "restecg" "thalachh" "exng" "oldpeak" "slp" "caa"   
## [13] "thall" "output"

## Inspecting the dataframe and look for inconguencies.

str(heart)

## '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 ...

## Chaning few names of few columns in our dataset.

heart = plyr::rename(heart, replace=c("cp" = "chest\_pain",  
 "fbs" = "fasting\_blood\_sugar",  
 "exng" = "exercise\_induced\_angina",  
 "slp" = "slope",  
 "thall" = "thalassemia",  
 "output" = "target"))

## Checking the column names of our new columns.

colnames(heart)

## [1] "age" "sex"   
## [3] "chest\_pain" "trtbps"   
## [5] "chol" "fasting\_blood\_sugar"   
## [7] "restecg" "thalachh"   
## [9] "exercise\_induced\_angina" "oldpeak"   
## [11] "slope" "caa"   
## [13] "thalassemia" "target"

## Checking for null values in our dataset.

null\_sum <- colSums(is.na(heart))  
null\_sum

## age sex chest\_pain   
## 0 0 0   
## trtbps chol fasting\_blood\_sugar   
## 0 0 0   
## restecg thalachh exercise\_induced\_angina   
## 0 0 0   
## oldpeak slope caa   
## 0 0 0   
## thalassemia target   
## 0 0

## Checking for any duplicate values in our dataset.

anyDuplicated(heart)

## [1] 165

## Removing those duplicate values.

heart = unique(heart)  
anyDuplicated(heart)

## [1] 0

## Checking for the current dimensions of our dataset.

dim(heart)

## [1] 302 14

## Seeing how many observations fall under each of these columns.

table(heart$age)

##   
## 29 34 35 37 38 39 40 41 42 43 44 45 46 47 48 49 50 51 52 53 54 55 56 57 58 59   
## 1 2 4 2 2 4 3 10 8 8 11 8 7 5 7 5 7 12 13 8 16 8 11 17 19 14   
## 60 61 62 63 64 65 66 67 68 69 70 71 74 76 77   
## 11 8 11 9 10 8 7 9 4 3 4 3 1 1 1

table(heart$sex)

##   
## 0 1   
## 96 206

table(heart$chest\_pain)

##   
## 0 1 2 3   
## 143 50 86 23

table(heart$trtbps)

##   
## 94 100 101 102 104 105 106 108 110 112 114 115 117 118 120 122 123 124 125 126   
## 2 4 1 2 1 3 1 6 19 9 1 3 1 7 37 4 1 6 11 3   
## 128 129 130 132 134 135 136 138 140 142 144 145 146 148 150 152 154 155 156 160   
## 12 1 36 8 5 6 3 12 32 3 2 5 2 2 17 5 1 1 1 11   
## 164 165 170 172 174 178 180 192 200   
## 1 1 4 1 1 2 3 1 1

table(heart$chol)

##   
## 126 131 141 149 157 160 164 166 167 168 169 172 174 175 176 177 178 180 182 183   
## 1 1 1 2 1 1 1 1 1 1 1 1 1 2 1 4 1 1 1 1   
## 184 185 186 187 188 192 193 195 196 197 198 199 200 201 203 204 205 206 207 208   
## 1 1 1 1 2 2 2 1 2 6 2 3 1 3 3 6 2 2 2 2   
## 209 210 211 212 213 214 215 216 217 218 219 220 221 222 223 224 225 226 227 228   
## 2 1 4 5 2 2 1 2 1 2 3 3 2 2 3 1 2 4 2 2   
## 229 230 231 232 233 234 235 236 237 239 240 241 242 243 244 245 246 247 248 249   
## 3 3 3 2 4 6 2 3 1 4 4 1 1 4 3 3 3 2 2 3   
## 250 252 253 254 255 256 257 258 259 260 261 262 263 264 265 266 267 268 269 270   
## 3 1 2 5 2 3 1 3 1 2 2 1 3 2 2 2 2 2 5 2   
## 271 273 274 275 276 277 278 281 282 283 284 286 288 289 290 293 294 295 298 299   
## 2 2 3 2 1 2 1 1 4 3 1 2 3 2 1 1 2 2 2 2   
## 300 302 303 304 305 306 307 308 309 311 313 315 318 319 321 322 325 326 327 330   
## 1 2 3 2 1 1 1 2 3 1 1 2 2 1 1 1 2 1 1 2   
## 335 340 341 342 353 354 360 394 407 409 417 564   
## 2 1 1 1 1 1 1 1 1 1 1 1

table(heart$fasting\_blood\_sugar)

##   
## 0 1   
## 257 45

table(heart$restecg)

##   
## 0 1 2   
## 147 151 4

table(heart$thalachh)

##   
## 71 88 90 95 96 97 99 103 105 106 108 109 111 112 113 114 115 116 117 118   
## 1 1 1 1 2 1 1 2 3 1 2 2 3 2 1 3 3 2 1 1   
## 120 121 122 123 124 125 126 127 128 129 130 131 132 133 134 136 137 138 139 140   
## 3 1 4 2 1 7 4 1 1 1 4 4 7 2 1 2 1 3 2 6   
## 141 142 143 144 145 146 147 148 149 150 151 152 153 154 155 156 157 158 159 160   
## 3 6 7 7 4 4 5 3 2 7 4 8 3 5 4 6 5 6 4 9   
## 161 162 163 164 165 166 167 168 169 170 171 172 173 174 175 177 178 179 180 181   
## 5 11 9 2 5 3 1 5 6 5 4 7 7 5 3 1 5 5 2 2   
## 182 184 185 186 187 188 190 192 194 195 202   
## 5 1 1 2 1 1 1 1 1 1 1

table(heart$exercise\_induced\_angina)

##   
## 0 1   
## 203 99

table(heart$oldpeak)

##   
## 0 0.1 0.2 0.3 0.4 0.5 0.6 0.7 0.8 0.9 1 1.1 1.2 1.3 1.4 1.5 1.6 1.8 1.9 2   
## 98 7 12 3 9 5 14 1 13 3 14 2 17 1 13 5 11 10 5 9   
## 2.1 2.2 2.3 2.4 2.5 2.6 2.8 2.9 3 3.1 3.2 3.4 3.5 3.6 3.8 4 4.2 4.4 5.6 6.2   
## 1 4 2 3 2 6 6 1 5 1 2 3 1 4 1 3 2 1 1 1

table(heart$slope)

##   
## 0 1 2   
## 21 140 141

table(heart$caa)

##   
## 0 1 2 3 4   
## 175 65 38 20 4

table(heart$thalassemia)

##   
## 0 1 2 3   
## 2 18 165 117

table(heart$target)

##   
## 0 1   
## 138 164

## CONDUCTNG DESCRIPTIVE ANALYSIS:

Descriptive analysis on few columns of our dataset.

summary(heart$age)

## Min. 1st Qu. Median Mean 3rd Qu. Max.   
## 29.00 48.00 55.50 54.42 61.00 77.00

summary(heart$trtbps)

## Min. 1st Qu. Median Mean 3rd Qu. Max.   
## 94.0 120.0 130.0 131.6 140.0 200.0

summary(heart$chol)

## Min. 1st Qu. Median Mean 3rd Qu. Max.   
## 126.0 211.0 240.5 246.5 274.8 564.0

summary(heart$thalachh)

## Min. 1st Qu. Median Mean 3rd Qu. Max.   
## 71.0 133.2 152.5 149.6 166.0 202.0

summary(heart$oldpeak)

## Min. 1st Qu. Median Mean 3rd Qu. Max.   
## 0.000 0.000 0.800 1.043 1.600 6.200

Comparing different columns with each other from each of our dataset.

aggregate(heart$trtbps ~ heart$age, data = heart, FUN = mean)

## heart$age heart$trtbps  
## 1 29 130.0000  
## 2 34 118.0000  
## 3 35 126.5000  
## 4 37 125.0000  
## 5 38 129.0000  
## 6 39 122.5000  
## 7 40 134.0000  
## 8 41 119.0000  
## 9 42 127.0000  
## 10 43 126.3750  
## 11 44 120.7273  
## 12 45 122.3750  
## 13 46 128.0000  
## 14 47 119.6000  
## 15 48 124.2857  
## 16 49 126.4000  
## 17 50 130.4286  
## 18 51 124.5000  
## 19 52 130.6923  
## 20 53 132.6250  
## 21 54 129.1250  
## 22 55 142.1250  
## 23 56 134.6364  
## 24 57 136.0588  
## 25 58 127.9474  
## 26 59 149.2143  
## 27 60 131.7273  
## 28 61 138.1250  
## 29 62 134.5455  
## 30 63 133.5556  
## 31 64 138.8000  
## 32 65 138.5000  
## 33 66 146.5714  
## 34 67 127.7778  
## 35 68 140.5000  
## 36 69 146.6667  
## 37 70 147.7500  
## 38 71 127.3333  
## 39 74 120.0000  
## 40 76 140.0000  
## 41 77 125.0000

aggregate(heart$trtbps ~ heart$age, data = heart, FUN = median)

## heart$age heart$trtbps  
## 1 29 130.0  
## 2 34 118.0  
## 3 35 124.0  
## 4 37 125.0  
## 5 38 129.0  
## 6 39 128.0  
## 7 40 140.0  
## 8 41 116.0  
## 9 42 125.0  
## 10 43 126.0  
## 11 44 120.0  
## 12 45 121.5  
## 13 46 138.0  
## 14 47 112.0  
## 15 48 124.0  
## 16 49 130.0  
## 17 50 129.0  
## 18 51 127.5  
## 19 52 128.0  
## 20 53 130.0  
## 21 54 123.0  
## 22 55 133.5  
## 23 56 130.0  
## 24 57 132.0  
## 25 58 128.0  
## 26 59 145.0  
## 27 60 130.0  
## 28 61 139.0  
## 29 62 130.0  
## 30 63 135.0  
## 31 64 135.0  
## 32 65 139.0  
## 33 66 150.0  
## 34 67 120.0  
## 35 68 132.0  
## 36 69 140.0  
## 37 70 150.5  
## 38 71 112.0  
## 39 74 120.0  
## 40 76 140.0  
## 41 77 125.0

aggregate(heart$trtbps ~ heart$age, data = heart, FUN = max)

## heart$age heart$trtbps  
## 1 29 130  
## 2 34 118  
## 3 35 138  
## 4 37 130  
## 5 38 138  
## 6 39 140  
## 7 40 152  
## 8 41 135  
## 9 42 148  
## 10 43 150  
## 11 44 140  
## 12 45 142  
## 13 46 150  
## 14 47 138  
## 15 48 130  
## 16 49 134  
## 17 50 150  
## 18 51 140  
## 19 52 172  
## 20 53 142  
## 21 54 192  
## 22 55 180  
## 23 56 200  
## 24 57 165  
## 25 58 170  
## 26 59 178  
## 27 60 150  
## 28 61 150  
## 29 62 160  
## 30 63 150  
## 31 64 180  
## 32 65 160  
## 33 66 178  
## 34 67 160  
## 35 68 180  
## 36 69 160  
## 37 70 160  
## 38 71 160  
## 39 74 120  
## 40 76 140  
## 41 77 125

aggregate(heart$trtbps ~ heart$age, data = heart, FUN = min)

## heart$age heart$trtbps  
## 1 29 130  
## 2 34 118  
## 3 35 120  
## 4 37 120  
## 5 38 120  
## 6 39 94  
## 7 40 110  
## 8 41 105  
## 9 42 102  
## 10 43 110  
## 11 44 108  
## 12 45 104  
## 13 46 101  
## 14 47 108  
## 15 48 110  
## 16 49 118  
## 17 50 110  
## 18 51 94  
## 19 52 108  
## 20 53 123  
## 21 54 108  
## 22 55 128  
## 23 56 120  
## 24 57 110  
## 25 58 100  
## 26 59 110  
## 27 60 102  
## 28 61 120  
## 29 62 120  
## 30 63 108  
## 31 64 110  
## 32 65 110  
## 33 66 112  
## 34 67 100  
## 35 68 118  
## 36 69 140  
## 37 70 130  
## 38 71 110  
## 39 74 120  
## 40 76 140  
## 41 77 125

aggregate(heart$chol ~ heart$age, data = heart, FUN = mean)

## heart$age heart$chol  
## 1 29 204.0000  
## 2 34 196.0000  
## 3 35 213.7500  
## 4 37 232.5000  
## 5 38 203.0000  
## 6 39 239.7500  
## 7 40 196.3333  
## 8 41 220.7000  
## 9 42 246.7500  
## 10 43 256.7500  
## 11 44 221.3636  
## 12 45 247.3750  
## 13 46 230.2857  
## 14 47 246.4000  
## 15 48 250.8571  
## 16 49 228.6000  
## 17 50 227.0000  
## 18 51 258.6667  
## 19 52 228.2308  
## 20 53 233.5000  
## 21 54 253.3125  
## 22 55 280.6250  
## 23 56 259.3636  
## 24 57 238.5294  
## 25 58 254.5263  
## 26 59 239.8571  
## 27 60 245.5455  
## 28 61 243.7500  
## 29 62 256.6364  
## 30 63 258.2222  
## 31 64 274.4000  
## 32 65 279.0000  
## 33 66 245.7143  
## 34 67 286.7778  
## 35 68 238.7500  
## 36 69 242.3333  
## 37 70 252.5000  
## 38 71 238.6667  
## 39 74 269.0000  
## 40 76 197.0000  
## 41 77 304.0000

aggregate(heart$chol ~ heart$age, data = heart, FUN = median)

## heart$age heart$chol  
## 1 29 204.0  
## 2 34 196.0  
## 3 35 195.0  
## 4 37 232.5  
## 5 38 203.0  
## 6 39 219.5  
## 7 40 199.0  
## 8 41 209.0  
## 9 42 242.0  
## 10 43 247.0  
## 11 44 226.0  
## 12 45 248.0  
## 13 46 231.0  
## 14 47 253.0  
## 15 48 255.0  
## 16 49 266.0  
## 17 50 233.0  
## 18 51 258.5  
## 19 52 212.0  
## 20 53 230.0  
## 21 54 262.0  
## 22 55 275.5  
## 23 56 249.0  
## 24 57 236.0  
## 25 58 240.0  
## 26 59 236.5  
## 27 60 253.0  
## 28 61 238.5  
## 29 62 263.0  
## 30 63 252.0  
## 31 64 283.0  
## 32 65 261.5  
## 33 66 228.0  
## 34 67 254.0  
## 35 68 242.5  
## 36 69 239.0  
## 37 70 257.0  
## 38 71 265.0  
## 39 74 269.0  
## 40 76 197.0  
## 41 77 304.0

aggregate(heart$chol ~ heart$age, data = heart, FUN = max)

## heart$age heart$chol  
## 1 29 204  
## 2 34 210  
## 3 35 282  
## 4 37 250  
## 5 38 231  
## 6 39 321  
## 7 40 223  
## 8 41 306  
## 9 42 315  
## 10 43 341  
## 11 44 290  
## 12 45 309  
## 13 46 311  
## 14 47 275  
## 15 48 275  
## 16 49 271  
## 17 50 254  
## 18 51 308  
## 19 52 325  
## 20 53 282  
## 21 54 309  
## 22 55 353  
## 23 56 409  
## 24 57 354  
## 25 58 340  
## 26 59 326  
## 27 60 318  
## 28 61 330  
## 29 62 394  
## 30 63 407  
## 31 64 335  
## 32 65 417  
## 33 66 302  
## 34 67 564  
## 35 68 277  
## 36 69 254  
## 37 70 322  
## 38 71 302  
## 39 74 269  
## 40 76 197  
## 41 77 304

aggregate(heart$chol ~ heart$age, data = heart, FUN = min)

## heart$age heart$chol  
## 1 29 204  
## 2 34 182  
## 3 35 183  
## 4 37 215  
## 5 38 175  
## 6 39 199  
## 7 40 167  
## 8 41 157  
## 9 42 180  
## 10 43 177  
## 11 44 141  
## 12 45 160  
## 13 46 177  
## 14 47 204  
## 15 48 222  
## 16 49 149  
## 17 50 196  
## 18 51 175  
## 19 52 186  
## 20 53 197  
## 21 54 188  
## 22 55 205  
## 23 56 184  
## 24 57 126  
## 25 58 197  
## 26 59 176  
## 27 60 178  
## 28 61 166  
## 29 62 164  
## 30 63 187  
## 31 64 211  
## 32 65 177  
## 33 66 212  
## 34 67 212  
## 35 68 193  
## 36 69 234  
## 37 70 174  
## 38 71 149  
## 39 74 269  
## 40 76 197  
## 41 77 304

aggregate(heart$thalachh ~ heart$age, data = heart, FUN = mean)

## heart$age heart$thalachh  
## 1 29 202.0000  
## 2 34 183.0000  
## 3 35 160.5000  
## 4 37 178.5000  
## 5 38 177.5000  
## 6 39 163.2500  
## 7 40 157.6667  
## 8 41 164.7000  
## 9 42 160.2500  
## 10 43 154.8750  
## 11 44 168.8182  
## 12 45 155.8750  
## 13 46 150.1429  
## 14 47 149.6000  
## 15 48 166.2857  
## 16 49 152.2000  
## 17 50 151.2857  
## 18 51 148.5000  
## 19 52 167.2308  
## 20 53 138.0000  
## 21 54 147.7500  
## 22 55 139.6250  
## 23 56 145.6364  
## 24 57 143.8235  
## 25 58 146.8421  
## 26 59 147.5714  
## 27 60 148.0000  
## 28 61 145.1250  
## 29 62 133.6364  
## 30 63 153.6667  
## 31 64 133.0000  
## 32 65 146.1250  
## 33 66 138.8571  
## 34 67 135.5556  
## 35 68 139.2500  
## 36 69 142.6667  
## 37 70 122.2500  
## 38 71 139.0000  
## 39 74 121.0000  
## 40 76 116.0000  
## 41 77 162.0000

aggregate(heart$thalachh ~ heart$age, data = heart, FUN = median)

## heart$age heart$thalachh  
## 1 29 202.0  
## 2 34 183.0  
## 3 35 165.0  
## 4 37 178.5  
## 5 38 177.5  
## 6 39 165.5  
## 7 40 178.0  
## 8 41 168.0  
## 9 42 167.5  
## 10 43 161.5  
## 11 44 173.0  
## 12 45 150.0  
## 13 46 152.0  
## 14 47 152.0  
## 15 48 168.0  
## 16 49 162.0  
## 17 50 159.0  
## 18 51 146.0  
## 19 52 168.0  
## 20 53 147.5  
## 21 54 157.0  
## 22 55 138.5  
## 23 56 150.0  
## 24 57 148.0  
## 25 58 152.0  
## 26 59 151.0  
## 27 60 155.0  
## 28 61 142.5  
## 29 62 145.0  
## 30 63 150.0  
## 31 64 132.5  
## 32 65 149.5  
## 33 66 138.0  
## 34 67 142.0  
## 35 68 145.5  
## 36 69 146.0  
## 37 70 118.5  
## 38 71 130.0  
## 39 74 121.0  
## 40 76 116.0  
## 41 77 162.0

aggregate(heart$thalachh ~ heart$age, data = heart, FUN = max)

## heart$age heart$thalachh  
## 1 29 202  
## 2 34 192  
## 3 35 182  
## 4 37 187  
## 5 38 182  
## 6 39 182  
## 7 40 181  
## 8 41 182  
## 9 42 194  
## 10 43 181  
## 11 44 188  
## 12 45 185  
## 13 46 172  
## 14 47 179  
## 15 48 186  
## 16 49 171  
## 17 50 163  
## 18 51 186  
## 19 52 190  
## 20 53 173  
## 21 54 195  
## 22 55 166  
## 23 56 178  
## 24 57 174  
## 25 58 173  
## 26 59 182  
## 27 60 171  
## 28 61 169  
## 29 62 163  
## 30 63 179  
## 31 64 158  
## 32 65 174  
## 33 66 165  
## 34 67 172  
## 35 68 151  
## 36 69 151  
## 37 70 143  
## 38 71 162  
## 39 74 121  
## 40 76 116  
## 41 77 162

aggregate(heart$thalachh ~ heart$age, data = heart, FUN = min)

## heart$age heart$thalachh  
## 1 29 202  
## 2 34 174  
## 3 35 130  
## 4 37 170  
## 5 38 173  
## 6 39 140  
## 7 40 114  
## 8 41 132  
## 9 42 122  
## 10 43 120  
## 11 44 144  
## 12 45 132  
## 13 46 120  
## 14 47 118  
## 15 48 139  
## 16 49 126  
## 17 50 126  
## 18 51 122  
## 19 52 147  
## 20 53 95  
## 21 54 108  
## 22 55 111  
## 23 56 103  
## 24 57 88  
## 25 58 105  
## 26 59 90  
## 27 60 96  
## 28 61 125  
## 29 62 97  
## 30 63 132  
## 31 64 96  
## 32 65 114  
## 33 66 114  
## 34 67 71  
## 35 68 115  
## 36 69 131  
## 37 70 109  
## 38 71 125  
## 39 74 121  
## 40 76 116  
## 41 77 162

aggregate(heart$oldpeak ~ heart$age, data = heart, FUN = mean)

## heart$age heart$oldpeak  
## 1 29 0.0000000  
## 2 34 0.3500000  
## 3 35 0.7500000  
## 4 37 1.7500000  
## 5 38 1.9000000  
## 6 39 0.3000000  
## 7 40 1.1333333  
## 8 41 0.3400000  
## 9 42 0.5000000  
## 10 43 1.3000000  
## 11 44 0.3727273  
## 12 45 0.6250000  
## 13 46 1.0857143  
## 14 47 0.2200000  
## 15 48 0.2714286  
## 16 49 0.6800000  
## 17 50 0.9714286  
## 18 51 1.2666667  
## 19 52 0.3769231  
## 20 53 0.8375000  
## 21 54 0.9312500  
## 22 55 1.9500000  
## 23 56 1.4000000  
## 24 57 0.7176471  
## 25 58 1.3894737  
## 26 59 1.0785714  
## 27 60 1.6818182  
## 28 61 1.7125000  
## 29 62 1.8636364  
## 30 63 1.7000000  
## 31 64 1.0800000  
## 32 65 1.0750000  
## 33 66 0.9142857  
## 34 67 0.9888889  
## 35 68 1.8750000  
## 36 69 1.3000000  
## 37 70 1.9750000  
## 38 71 0.6666667  
## 39 74 0.2000000  
## 40 76 1.1000000  
## 41 77 0.0000000

aggregate(heart$oldpeak ~ heart$age, data = heart, FUN = median)

## heart$age heart$oldpeak  
## 1 29 0.00  
## 2 34 0.35  
## 3 35 0.70  
## 4 37 1.75  
## 5 38 1.90  
## 6 39 0.00  
## 7 40 1.40  
## 8 41 0.00  
## 9 42 0.30  
## 10 43 1.35  
## 11 44 0.00  
## 12 45 0.10  
## 13 46 0.80  
## 14 47 0.00  
## 15 48 0.20  
## 16 49 0.60  
## 17 50 0.90  
## 18 51 1.20  
## 19 52 0.10  
## 20 53 0.20  
## 21 54 0.45  
## 22 55 1.30  
## 23 56 1.30  
## 24 57 0.40  
## 25 58 1.00  
## 26 59 0.65  
## 27 60 1.40  
## 28 61 1.45  
## 29 62 1.40  
## 30 63 1.80  
## 31 64 1.20  
## 32 65 0.80  
## 33 66 0.40  
## 34 67 0.90  
## 35 68 1.55  
## 36 69 1.80  
## 37 70 2.50  
## 38 71 0.40  
## 39 74 0.20  
## 40 76 1.10  
## 41 77 0.00

aggregate(heart$oldpeak ~ heart$age, data = heart, FUN = max)

## heart$age heart$oldpeak  
## 1 29 0.0  
## 2 34 0.7  
## 3 35 1.6  
## 4 37 3.5  
## 5 38 3.8  
## 6 39 1.2  
## 7 40 2.0  
## 8 41 2.0  
## 9 42 1.8  
## 10 43 3.0  
## 11 44 2.8  
## 12 45 3.0  
## 13 46 3.6  
## 14 47 1.0  
## 15 48 1.0  
## 16 49 2.0  
## 17 50 2.6  
## 18 51 4.2  
## 19 52 1.2  
## 20 53 3.1  
## 21 54 3.2  
## 22 55 5.6  
## 23 56 4.0  
## 24 57 3.0  
## 25 58 4.4  
## 26 59 4.2  
## 27 60 3.0  
## 28 61 3.6  
## 29 62 6.2  
## 30 63 4.0  
## 31 64 2.2  
## 32 65 2.8  
## 33 66 2.6  
## 34 67 2.6  
## 35 68 3.4  
## 36 69 2.0  
## 37 70 2.9  
## 38 71 1.6  
## 39 74 0.2  
## 40 76 1.1  
## 41 77 0.0

aggregate(heart$oldpeak ~ heart$age, data = heart, FUN = min)

## heart$age heart$oldpeak  
## 1 29 0.0  
## 2 34 0.0  
## 3 35 0.0  
## 4 37 0.0  
## 5 38 0.0  
## 6 39 0.0  
## 7 40 0.0  
## 8 41 0.0  
## 9 42 0.0  
## 10 43 0.0  
## 11 44 0.0  
## 12 45 0.0  
## 13 46 0.0  
## 14 47 0.0  
## 15 48 0.0  
## 16 49 0.0  
## 17 50 0.0  
## 18 51 0.0  
## 19 52 0.0  
## 20 53 0.0  
## 21 54 0.0  
## 22 55 0.0  
## 23 56 0.0  
## 24 57 0.0  
## 25 58 0.0  
## 26 59 0.0  
## 27 60 0.0  
## 28 61 0.0  
## 29 62 0.0  
## 30 63 0.0  
## 31 64 0.0  
## 32 65 0.4  
## 33 66 0.0  
## 34 67 0.0  
## 35 68 1.0  
## 36 69 0.1  
## 37 70 0.0  
## 38 71 0.0  
## 39 74 0.2  
## 40 76 1.1  
## 41 77 0.0

aggregate(heart$age ~ heart$sex, data = heart, FUN = mean)

## heart$sex heart$age  
## 1 0 55.67708  
## 2 1 53.83495

aggregate(heart$age ~ heart$sex, data = heart, FUN = median)

## heart$sex heart$age  
## 1 0 57.0  
## 2 1 54.5

aggregate(heart$age ~ heart$sex, data = heart, FUN = max)

## heart$sex heart$age  
## 1 0 76  
## 2 1 77

aggregate(heart$age ~ heart$sex, data = heart, FUN = min)

## heart$sex heart$age  
## 1 0 34  
## 2 1 29

aggregate(heart$trtbps ~ heart$sex, data = heart, FUN = mean)

## heart$sex heart$trtbps  
## 1 0 133.0833  
## 2 1 130.9126

aggregate(heart$trtbps ~ heart$sex, data = heart, FUN = median)

## heart$sex heart$trtbps  
## 1 0 131  
## 2 1 130

aggregate(heart$trtbps ~ heart$sex, data = heart, FUN = max)

## heart$sex heart$trtbps  
## 1 0 200  
## 2 1 192

aggregate(heart$trtbps ~ heart$sex, data = heart, FUN = min)

## heart$sex heart$trtbps  
## 1 0 94  
## 2 1 94

aggregate(heart$thalachh ~ heart$sex, data = heart, FUN = mean)

## heart$sex heart$thalachh  
## 1 0 151.1250  
## 2 1 148.8447

aggregate(heart$thalachh ~ heart$sex, data = heart, FUN = median)

## heart$sex heart$thalachh  
## 1 0 157.0  
## 2 1 150.5

aggregate(heart$thalachh ~ heart$sex, data = heart, FUN = max)

## heart$sex heart$thalachh  
## 1 0 192  
## 2 1 202

aggregate(heart$thalachh ~ heart$sex, data = heart, FUN = min)

## heart$sex heart$thalachh  
## 1 0 96  
## 2 1 71

aggregate(heart$oldpeak ~ heart$sex, data = heart, FUN = mean)

## heart$sex heart$oldpeak  
## 1 0 0.8760417  
## 2 1 1.1208738

aggregate(heart$oldpeak ~ heart$sex, data = heart, FUN = median)

## heart$sex heart$oldpeak  
## 1 0 0.6  
## 2 1 0.8

aggregate(heart$oldpeak ~ heart$sex, data = heart, FUN = max)

## heart$sex heart$oldpeak  
## 1 0 6.2  
## 2 1 5.6

aggregate(heart$oldpeak ~ heart$sex, data = heart, FUN = min)

## heart$sex heart$oldpeak  
## 1 0 0  
## 2 1 0

## Comparing different columns of our dataframe with Age and Sex of people.

aggregate(heart$chest\_pain ~ heart$sex + heart$age, FUN = mean)

## heart$sex heart$age heart$chest\_pain  
## 1 1 29 1.0000000  
## 2 0 34 1.0000000  
## 3 1 34 3.0000000  
## 4 0 35 0.0000000  
## 5 1 35 0.3333333  
## 6 0 37 2.0000000  
## 7 1 37 2.0000000  
## 8 1 38 2.5000000  
## 9 0 39 2.0000000  
## 10 1 39 1.0000000  
## 11 1 40 1.0000000  
## 12 0 41 1.2500000  
## 13 1 41 1.1666667  
## 14 0 42 1.0000000  
## 15 1 42 1.3333333  
## 16 0 43 1.0000000  
## 17 1 43 0.3333333  
## 18 0 44 2.0000000  
## 19 1 44 1.0000000  
## 20 0 45 0.6666667  
## 21 1 45 0.8000000  
## 22 0 46 1.0000000  
## 23 1 46 0.7500000  
## 24 1 47 1.2000000  
## 25 0 48 2.0000000  
## 26 1 48 0.6666667  
## 27 0 49 0.5000000  
## 28 1 49 1.6666667  
## 29 0 50 1.0000000  
## 30 1 50 1.0000000  
## 31 0 51 1.5000000  
## 32 1 51 1.3750000  
## 33 0 52 2.0000000  
## 34 1 52 1.0833333  
## 35 0 53 0.6666667  
## 36 1 53 0.8000000  
## 37 0 54 1.8000000  
## 38 1 54 0.7272727  
## 39 0 55 0.5000000  
## 40 1 55 0.2500000  
## 41 0 56 0.3333333  
## 42 1 56 1.0000000  
## 43 0 57 0.2500000  
## 44 1 57 0.6153846  
## 45 0 58 1.0000000  
## 46 1 58 0.7692308  
## 47 0 59 0.0000000  
## 48 1 59 1.3076923  
## 49 0 60 1.7500000  
## 50 1 60 0.2857143  
## 51 0 61 0.0000000  
## 52 1 61 0.8333333  
## 53 0 62 0.2857143  
## 54 1 62 1.0000000  
## 55 0 63 0.6000000  
## 56 1 63 0.7500000  
## 57 0 64 0.6666667  
## 58 1 64 1.4285714  
## 59 0 65 1.5000000  
## 60 1 65 0.7500000  
## 61 0 66 1.6666667  
## 62 1 66 0.2500000  
## 63 0 67 1.3333333  
## 64 1 67 0.3333333  
## 65 0 68 2.0000000  
## 66 1 68 1.3333333  
## 67 0 69 3.0000000  
## 68 1 69 2.5000000  
## 69 1 70 0.7500000  
## 70 0 71 1.0000000  
## 71 0 74 1.0000000  
## 72 0 76 2.0000000  
## 73 1 77 0.0000000

aggregate(heart$trtbps ~ heart$sex + heart$age, FUN = mean)

## heart$sex heart$age heart$trtbps  
## 1 1 29 130.0000  
## 2 0 34 118.0000  
## 3 1 34 118.0000  
## 4 0 35 138.0000  
## 5 1 35 122.6667  
## 6 0 37 120.0000  
## 7 1 37 130.0000  
## 8 1 38 129.0000  
## 9 0 39 116.0000  
## 10 1 39 129.0000  
## 11 1 40 134.0000  
## 12 0 41 118.2500  
## 13 1 41 119.5000  
## 14 0 42 111.0000  
## 15 1 42 132.3333  
## 16 0 43 127.0000  
## 17 1 43 126.1667  
## 18 0 44 113.0000  
## 19 1 44 122.4444  
## 20 0 45 126.6667  
## 21 1 45 119.8000  
## 22 0 46 128.3333  
## 23 1 46 127.7500  
## 24 1 47 119.6000  
## 25 0 48 130.0000  
## 26 1 48 123.3333  
## 27 0 49 132.0000  
## 28 1 49 122.6667  
## 29 0 50 116.6667  
## 30 1 50 140.7500  
## 31 0 51 130.0000  
## 32 1 51 121.7500  
## 33 0 52 136.0000  
## 34 1 52 130.2500  
## 35 0 53 132.0000  
## 36 1 53 133.0000  
## 37 0 54 129.0000  
## 38 1 54 129.1818  
## 39 0 55 143.7500  
## 40 1 55 140.5000  
## 41 0 56 158.0000  
## 42 1 56 125.8750  
## 43 0 57 129.5000  
## 44 1 57 138.0769  
## 45 0 58 134.3333  
## 46 1 58 125.0000  
## 47 0 59 174.0000  
## 48 1 59 147.3077  
## 49 0 60 130.5000  
## 50 1 60 132.4286  
## 51 0 61 137.5000  
## 52 1 61 138.3333  
## 53 0 62 140.2857  
## 54 1 62 124.5000  
## 55 0 63 131.4000  
## 56 1 63 136.2500  
## 57 0 64 150.0000  
## 58 1 64 134.0000  
## 59 0 65 151.2500  
## 60 1 65 125.7500  
## 61 0 66 158.0000  
## 62 1 66 138.0000  
## 63 0 67 124.3333  
## 64 1 67 129.5000  
## 65 0 68 120.0000  
## 66 1 68 147.3333  
## 67 0 69 140.0000  
## 68 1 69 150.0000  
## 69 1 70 147.7500  
## 70 0 71 127.3333  
## 71 0 74 120.0000  
## 72 0 76 140.0000  
## 73 1 77 125.0000

aggregate(heart$chol ~ heart$sex + heart$age, FUN = mean)

## heart$sex heart$age heart$chol  
## 1 1 29 204.0000  
## 2 0 34 210.0000  
## 3 1 34 182.0000  
## 4 0 35 183.0000  
## 5 1 35 224.0000  
## 6 0 37 215.0000  
## 7 1 37 250.0000  
## 8 1 38 203.0000  
## 9 0 39 209.5000  
## 10 1 39 270.0000  
## 11 1 40 196.3333  
## 12 0 41 244.0000  
## 13 1 41 205.1667  
## 14 0 42 237.0000  
## 15 1 42 250.0000  
## 16 0 43 277.0000  
## 17 1 43 250.0000  
## 18 0 44 191.5000  
## 19 1 44 228.0000  
## 20 0 45 210.0000  
## 21 1 45 269.8000  
## 22 0 46 208.0000  
## 23 1 46 247.0000  
## 24 1 47 246.4000  
## 25 0 48 275.0000  
## 26 1 48 246.8333  
## 27 0 49 270.0000  
## 28 1 49 201.0000  
## 29 0 50 239.0000  
## 30 1 50 218.0000  
## 31 0 51 291.0000  
## 32 1 51 242.5000  
## 33 0 52 196.0000  
## 34 1 52 230.9167  
## 35 0 53 238.0000  
## 36 1 53 230.8000  
## 37 0 54 254.8000  
## 38 1 54 252.6364  
## 39 0 55 281.0000  
## 40 1 55 280.2500  
## 41 0 56 330.3333  
## 42 1 56 232.7500  
## 43 0 57 283.5000  
## 44 1 57 224.6923  
## 45 0 58 268.6667  
## 46 1 58 248.0000  
## 47 0 59 249.0000  
## 48 1 59 239.1538  
## 49 0 60 248.5000  
## 50 1 60 243.8571  
## 51 0 61 318.5000  
## 52 1 61 218.8333  
## 53 0 62 262.2857  
## 54 1 62 246.7500  
## 55 0 63 264.0000  
## 56 1 63 251.0000  
## 57 0 64 313.6667  
## 58 1 64 257.5714  
## 59 0 65 317.7500  
## 60 1 65 240.2500  
## 61 0 66 244.0000  
## 62 1 66 247.0000  
## 63 0 67 354.6667  
## 64 1 67 252.8333  
## 65 0 68 211.0000  
## 66 1 68 248.0000  
## 67 0 69 239.0000  
## 68 1 69 244.0000  
## 69 1 70 252.5000  
## 70 0 71 238.6667  
## 71 0 74 269.0000  
## 72 0 76 197.0000  
## 73 1 77 304.0000

aggregate(heart$fasting\_blood\_sugar ~ heart$sex + heart$age, FUN = mean)

## heart$sex heart$age heart$fasting\_blood\_sugar  
## 1 1 29 0.00000000  
## 2 0 34 0.00000000  
## 3 1 34 0.00000000  
## 4 0 35 0.00000000  
## 5 1 35 0.00000000  
## 6 0 37 0.00000000  
## 7 1 37 0.00000000  
## 8 1 38 0.00000000  
## 9 0 39 0.00000000  
## 10 1 39 0.00000000  
## 11 1 40 0.00000000  
## 12 0 41 0.00000000  
## 13 1 41 0.00000000  
## 14 0 42 0.00000000  
## 15 1 42 0.16666667  
## 16 0 43 0.50000000  
## 17 1 43 0.16666667  
## 18 0 44 0.00000000  
## 19 1 44 0.00000000  
## 20 0 45 0.00000000  
## 21 1 45 0.00000000  
## 22 0 46 0.00000000  
## 23 1 46 0.25000000  
## 24 1 47 0.00000000  
## 25 0 48 0.00000000  
## 26 1 48 0.33333333  
## 27 0 49 0.00000000  
## 28 1 49 0.00000000  
## 29 0 50 0.00000000  
## 30 1 50 0.00000000  
## 31 0 51 0.00000000  
## 32 1 51 0.12500000  
## 33 0 52 0.00000000  
## 34 1 52 0.41666667  
## 35 0 53 0.00000000  
## 36 1 53 0.60000000  
## 37 0 54 0.40000000  
## 38 1 54 0.00000000  
## 39 0 55 0.00000000  
## 40 1 55 0.00000000  
## 41 0 56 0.33333333  
## 42 1 56 0.37500000  
## 43 0 57 0.00000000  
## 44 1 57 0.15384615  
## 45 0 58 0.50000000  
## 46 1 58 0.07692308  
## 47 0 59 0.00000000  
## 48 1 59 0.23076923  
## 49 0 60 0.25000000  
## 50 1 60 0.14285714  
## 51 0 61 0.00000000  
## 52 1 61 0.16666667  
## 53 0 62 0.14285714  
## 54 1 62 0.25000000  
## 55 0 63 0.00000000  
## 56 1 63 0.50000000  
## 57 0 64 0.00000000  
## 58 1 64 0.00000000  
## 59 0 65 0.25000000  
## 60 1 65 0.25000000  
## 61 0 66 0.33333333  
## 62 1 66 0.00000000  
## 63 0 67 0.00000000  
## 64 1 67 0.16666667  
## 65 0 68 0.00000000  
## 66 1 68 0.66666667  
## 67 0 69 0.00000000  
## 68 1 69 0.50000000  
## 69 1 70 0.00000000  
## 70 0 71 0.33333333  
## 71 0 74 0.00000000  
## 72 0 76 0.00000000  
## 73 1 77 0.00000000

aggregate(heart$restecg ~ heart$sex + heart$age, FUN = mean)

## heart$sex heart$age heart$restecg  
## 1 1 29 0.0000000  
## 2 0 34 1.0000000  
## 3 1 34 0.0000000  
## 4 0 35 1.0000000  
## 5 1 35 0.6666667  
## 6 0 37 1.0000000  
## 7 1 37 1.0000000  
## 8 1 38 1.0000000  
## 9 0 39 1.0000000  
## 10 1 39 0.5000000  
## 11 1 40 0.6666667  
## 12 0 41 0.5000000  
## 13 1 41 0.6666667  
## 14 0 42 0.5000000  
## 15 1 42 0.8333333  
## 16 0 43 0.5000000  
## 17 1 43 0.6666667  
## 18 0 44 1.0000000  
## 19 1 44 0.5555556  
## 20 0 45 0.3333333  
## 21 1 45 0.2000000  
## 22 0 46 0.3333333  
## 23 1 46 0.7500000  
## 24 1 47 0.6000000  
## 25 0 48 1.0000000  
## 26 1 48 0.3333333  
## 27 0 49 1.0000000  
## 28 1 49 0.6666667  
## 29 0 50 0.6666667  
## 30 1 50 0.5000000  
## 31 0 51 0.2500000  
## 32 1 51 0.6250000  
## 33 0 52 0.0000000  
## 34 1 52 0.9166667  
## 35 0 53 0.0000000  
## 36 1 53 0.2000000  
## 37 0 54 0.6000000  
## 38 1 54 0.3636364  
## 39 0 55 1.2500000  
## 40 1 55 0.7500000  
## 41 0 56 0.0000000  
## 42 1 56 0.2500000  
## 43 0 57 0.5000000  
## 44 1 57 0.6923077  
## 45 0 58 0.3333333  
## 46 1 58 0.3846154  
## 47 0 59 1.0000000  
## 48 1 59 0.4615385  
## 49 0 60 0.7500000  
## 50 1 60 0.2857143  
## 51 0 61 0.0000000  
## 52 1 61 0.6666667  
## 53 0 62 0.5714286  
## 54 1 62 0.5000000  
## 55 0 63 0.6000000  
## 56 1 63 0.0000000  
## 57 0 64 1.0000000  
## 58 1 64 0.4285714  
## 59 0 65 0.2500000  
## 60 1 65 0.2500000  
## 61 0 66 0.6666667  
## 62 1 66 0.2500000  
## 63 0 67 0.6666667  
## 64 1 67 0.3333333  
## 65 0 68 0.0000000  
## 66 1 68 0.6666667  
## 67 0 69 1.0000000  
## 68 1 69 0.0000000  
## 69 1 70 0.5000000  
## 70 0 71 0.6666667  
## 71 0 74 0.0000000  
## 72 0 76 2.0000000  
## 73 1 77 0.0000000

aggregate(heart$thalachh ~ heart$sex + heart$age, FUN = mean)

## heart$sex heart$age heart$thalachh  
## 1 1 29 202.0000  
## 2 0 34 192.0000  
## 3 1 34 174.0000  
## 4 0 35 182.0000  
## 5 1 35 153.3333  
## 6 0 37 170.0000  
## 7 1 37 187.0000  
## 8 1 38 177.5000  
## 9 0 39 165.5000  
## 10 1 39 161.0000  
## 11 1 40 157.6667  
## 12 0 41 168.7500  
## 13 1 41 162.0000  
## 14 0 42 147.5000  
## 15 1 42 164.5000  
## 16 0 43 150.5000  
## 17 1 43 156.3333  
## 18 0 44 162.0000  
## 19 1 44 170.3333  
## 20 0 45 155.0000  
## 21 1 45 156.4000  
## 22 0 46 161.3333  
## 23 1 46 141.7500  
## 24 1 47 149.6000  
## 25 0 48 139.0000  
## 26 1 48 170.8333  
## 27 0 49 162.5000  
## 28 1 49 145.3333  
## 29 0 50 159.6667  
## 30 1 50 145.0000  
## 31 0 51 147.5000  
## 32 1 51 149.0000  
## 33 0 52 169.0000  
## 34 1 52 167.0833  
## 35 0 53 139.3333  
## 36 1 53 137.2000  
## 37 0 54 163.4000  
## 38 1 54 140.6364  
## 39 0 55 143.5000  
## 40 1 55 135.7500  
## 41 0 56 145.3333  
## 42 1 56 145.7500  
## 43 0 57 154.7500  
## 44 1 57 140.4615  
## 45 0 58 147.5000  
## 46 1 58 146.5385  
## 47 0 59 143.0000  
## 48 1 59 147.9231  
## 49 0 60 146.0000  
## 50 1 60 149.1429  
## 51 0 61 157.5000  
## 52 1 61 141.0000  
## 53 0 62 140.2857  
## 54 1 62 122.0000  
## 55 0 63 162.0000  
## 56 1 63 143.2500  
## 57 0 64 136.3333  
## 58 1 64 131.5714  
## 59 0 65 142.5000  
## 60 1 65 149.7500  
## 61 0 66 143.6667  
## 62 1 66 135.2500  
## 63 0 67 158.0000  
## 64 1 67 124.3333  
## 65 0 68 115.0000  
## 66 1 68 147.3333  
## 67 0 69 151.0000  
## 68 1 69 138.5000  
## 69 1 70 122.2500  
## 70 0 71 139.0000  
## 71 0 74 121.0000  
## 72 0 76 116.0000  
## 73 1 77 162.0000

aggregate(heart$exercise\_induced\_angina ~ heart$sex + heart$age, FUN = mean)

## heart$sex heart$age heart$exercise\_induced\_angina  
## 1 1 29 0.0000000  
## 2 0 34 0.0000000  
## 3 1 34 0.0000000  
## 4 0 35 0.0000000  
## 5 1 35 0.6666667  
## 6 0 37 0.0000000  
## 7 1 37 0.0000000  
## 8 1 38 0.5000000  
## 9 0 39 0.0000000  
## 10 1 39 0.0000000  
## 11 1 40 0.6666667  
## 12 0 41 0.2500000  
## 13 1 41 0.0000000  
## 14 0 42 0.0000000  
## 15 1 42 0.1666667  
## 16 0 43 0.5000000  
## 17 1 43 0.3333333  
## 18 0 44 0.0000000  
## 19 1 44 0.2222222  
## 20 0 45 0.3333333  
## 21 1 45 0.4000000  
## 22 0 46 0.6666667  
## 23 1 46 0.2500000  
## 24 1 47 0.2000000  
## 25 0 48 0.0000000  
## 26 1 48 0.1666667  
## 27 0 49 0.0000000  
## 28 1 49 0.0000000  
## 29 0 50 0.0000000  
## 30 1 50 0.2500000  
## 31 0 51 0.2500000  
## 32 1 51 0.7500000  
## 33 0 52 0.0000000  
## 34 1 52 0.1666667  
## 35 0 53 0.0000000  
## 36 1 53 0.6000000  
## 37 0 54 0.2000000  
## 38 1 54 0.3636364  
## 39 0 55 0.5000000  
## 40 1 55 0.7500000  
## 41 0 56 0.6666667  
## 42 1 56 0.5000000  
## 43 0 57 0.5000000  
## 44 1 57 0.4615385  
## 45 0 58 0.1666667  
## 46 1 58 0.3076923  
## 47 0 59 1.0000000  
## 48 1 59 0.3076923  
## 49 0 60 0.0000000  
## 50 1 60 0.7142857  
## 51 0 61 0.5000000  
## 52 1 61 0.6666667  
## 53 0 62 0.1428571  
## 54 1 62 0.2500000  
## 55 0 63 0.4000000  
## 56 1 63 0.5000000  
## 57 0 64 0.3333333  
## 58 1 64 0.5714286  
## 59 0 65 0.0000000  
## 60 1 65 0.0000000  
## 61 0 66 0.3333333  
## 62 1 66 0.5000000  
## 63 0 67 0.0000000  
## 64 1 67 0.5000000  
## 65 0 68 0.0000000  
## 66 1 68 0.3333333  
## 67 0 69 0.0000000  
## 68 1 69 0.0000000  
## 69 1 70 0.5000000  
## 70 0 71 0.0000000  
## 71 0 74 1.0000000  
## 72 0 76 0.0000000  
## 73 1 77 1.0000000

aggregate(heart$oldpeak ~ heart$sex + heart$age, FUN = mean)

## heart$sex heart$age heart$oldpeak  
## 1 1 29 0.0000000  
## 2 0 34 0.7000000  
## 3 1 34 0.0000000  
## 4 0 35 1.4000000  
## 5 1 35 0.5333333  
## 6 0 37 0.0000000  
## 7 1 37 3.5000000  
## 8 1 38 1.9000000  
## 9 0 39 0.0000000  
## 10 1 39 0.6000000  
## 11 1 40 1.1333333  
## 12 0 41 0.3500000  
## 13 1 41 0.3333333  
## 14 0 42 0.3000000  
## 15 1 42 0.5666667  
## 16 0 43 1.6000000  
## 17 1 43 1.2000000  
## 18 0 44 0.4500000  
## 19 1 44 0.3555556  
## 20 0 45 0.2666667  
## 21 1 45 0.8400000  
## 22 0 46 0.4666667  
## 23 1 46 1.5500000  
## 24 1 47 0.2200000  
## 25 0 48 0.2000000  
## 26 1 48 0.2833333  
## 27 0 49 0.0000000  
## 28 1 49 1.1333333  
## 29 0 50 0.9000000  
## 30 1 50 1.0250000  
## 31 0 51 0.9500000  
## 32 1 51 1.4250000  
## 33 0 52 0.1000000  
## 34 1 52 0.4000000  
## 35 0 53 0.1333333  
## 36 1 53 1.2600000  
## 37 0 54 0.3200000  
## 38 1 54 1.2090909  
## 39 0 55 2.0000000  
## 40 1 55 1.9000000  
## 41 0 56 2.4000000  
## 42 1 56 1.0250000  
## 43 0 57 0.2000000  
## 44 1 57 0.8769231  
## 45 0 58 0.9000000  
## 46 1 58 1.6153846  
## 47 0 59 0.0000000  
## 48 1 59 1.1615385  
## 49 0 60 0.8750000  
## 50 1 60 2.1428571  
## 51 0 61 0.5000000  
## 52 1 61 2.1166667  
## 53 0 62 2.2142857  
## 54 1 62 1.2500000  
## 55 0 63 1.1600000  
## 56 1 63 2.3750000  
## 57 0 64 0.7333333  
## 58 1 64 1.2285714  
## 59 0 65 0.8500000  
## 60 1 65 1.3000000  
## 61 0 66 1.2000000  
## 62 1 66 0.7000000  
## 63 0 67 0.6333333  
## 64 1 67 1.1666667  
## 65 0 68 1.5000000  
## 66 1 68 2.0000000  
## 67 0 69 1.8000000  
## 68 1 69 1.0500000  
## 69 1 70 1.9750000  
## 70 0 71 0.6666667  
## 71 0 74 0.2000000  
## 72 0 76 1.1000000  
## 73 1 77 0.0000000

aggregate(heart$slope ~ heart$sex + heart$age, FUN = mean)

## heart$sex heart$age heart$slope  
## 1 1 29 2.0000000  
## 2 0 34 2.0000000  
## 3 1 34 2.0000000  
## 4 0 35 2.0000000  
## 5 1 35 1.6666667  
## 6 0 37 2.0000000  
## 7 1 37 0.0000000  
## 8 1 38 1.5000000  
## 9 0 39 1.5000000  
## 10 1 39 1.5000000  
## 11 1 40 1.6666667  
## 12 0 41 2.0000000  
## 13 1 41 1.6666667  
## 14 0 42 1.0000000  
## 15 1 42 1.5000000  
## 16 0 43 1.0000000  
## 17 1 43 1.5000000  
## 18 0 44 1.0000000  
## 19 1 44 1.7777778  
## 20 0 45 1.0000000  
## 21 1 45 1.4000000  
## 22 0 46 1.0000000  
## 23 1 46 1.5000000  
## 24 1 47 1.8000000  
## 25 0 48 2.0000000  
## 26 1 48 1.3333333  
## 27 0 49 1.5000000  
## 28 1 49 1.6666667  
## 29 0 50 1.6666667  
## 30 1 50 1.2500000  
## 31 0 51 1.7500000  
## 32 1 51 1.6250000  
## 33 0 52 1.0000000  
## 34 1 52 1.7500000  
## 35 0 53 1.6666667  
## 36 1 53 1.0000000  
## 37 0 54 1.8000000  
## 38 1 54 1.2727273  
## 39 0 55 1.2500000  
## 40 1 55 1.0000000  
## 41 0 56 0.6666667  
## 42 1 56 1.0000000  
## 43 0 57 1.5000000  
## 44 1 57 1.3846154  
## 45 0 58 1.5000000  
## 46 1 58 1.3076923  
## 47 0 59 1.0000000  
## 48 1 59 1.3076923  
## 49 0 60 1.7500000  
## 50 1 60 1.2857143  
## 51 0 61 1.5000000  
## 52 1 61 1.3333333  
## 53 0 62 0.8571429  
## 54 1 62 1.2500000  
## 55 0 63 1.4000000  
## 56 1 63 1.2500000  
## 57 0 64 1.6666667  
## 58 1 64 1.0000000  
## 59 0 65 1.7500000  
## 60 1 65 1.5000000  
## 61 0 66 0.6666667  
## 62 1 66 1.5000000  
## 63 0 67 1.6666667  
## 64 1 67 1.0000000  
## 65 0 68 1.0000000  
## 66 1 68 1.3333333  
## 67 0 69 2.0000000  
## 68 1 69 1.0000000  
## 69 1 70 1.0000000  
## 70 0 71 1.6666667  
## 71 0 74 2.0000000  
## 72 0 76 1.0000000  
## 73 1 77 2.0000000

aggregate(heart$caa ~ heart$sex + heart$age, FUN = mean)

## heart$sex heart$age heart$caa  
## 1 1 29 0.0000000  
## 2 0 34 0.0000000  
## 3 1 34 0.0000000  
## 4 0 35 0.0000000  
## 5 1 35 0.0000000  
## 6 0 37 0.0000000  
## 7 1 37 0.0000000  
## 8 1 38 2.0000000  
## 9 0 39 0.0000000  
## 10 1 39 0.0000000  
## 11 1 40 0.0000000  
## 12 0 41 0.2500000  
## 13 1 41 0.0000000  
## 14 0 42 0.0000000  
## 15 1 42 0.3333333  
## 16 0 43 0.0000000  
## 17 1 43 0.8333333  
## 18 0 44 0.5000000  
## 19 1 44 0.2222222  
## 20 0 45 0.0000000  
## 21 1 45 0.6000000  
## 22 0 46 0.0000000  
## 23 1 46 0.5000000  
## 24 1 47 0.2000000  
## 25 0 48 0.0000000  
## 26 1 48 0.6666667  
## 27 0 49 0.0000000  
## 28 1 49 2.0000000  
## 29 0 50 0.0000000  
## 30 1 50 0.2500000  
## 31 0 51 0.2500000  
## 32 1 51 0.6250000  
## 33 0 52 0.0000000  
## 34 1 52 1.0000000  
## 35 0 53 0.0000000  
## 36 1 53 1.0000000  
## 37 0 54 0.4000000  
## 38 1 54 0.7272727  
## 39 0 55 0.2500000  
## 40 1 55 0.5000000  
## 41 0 56 1.3333333  
## 42 1 56 0.3750000  
## 43 0 57 0.5000000  
## 44 1 57 0.7692308  
## 45 0 58 0.6666667  
## 46 1 58 1.4615385  
## 47 0 59 0.0000000  
## 48 1 59 0.5384615  
## 49 0 60 0.7500000  
## 50 1 60 1.4285714  
## 51 0 61 0.0000000  
## 52 1 61 1.0000000  
## 53 0 62 1.2857143  
## 54 1 62 1.5000000  
## 55 0 63 1.4000000  
## 56 1 63 1.5000000  
## 57 0 64 0.6666667  
## 58 1 64 0.5714286  
## 59 0 65 1.0000000  
## 60 1 65 1.0000000  
## 61 0 66 1.0000000  
## 62 1 66 1.0000000  
## 63 0 67 1.0000000  
## 64 1 67 1.5000000  
## 65 0 68 0.0000000  
## 66 1 68 1.0000000  
## 67 0 69 2.0000000  
## 68 1 69 2.0000000  
## 69 1 70 1.0000000  
## 70 0 71 1.0000000  
## 71 0 74 1.0000000  
## 72 0 76 0.0000000  
## 73 1 77 3.0000000

aggregate(heart$thalassemia ~ heart$sex + heart$age, FUN = mean)

## heart$sex heart$age heart$thalassemia  
## 1 1 29 2.000000  
## 2 0 34 2.000000  
## 3 1 34 2.000000  
## 4 0 35 2.000000  
## 5 1 35 2.666667  
## 6 0 37 2.000000  
## 7 1 37 2.000000  
## 8 1 38 2.500000  
## 9 0 39 2.000000  
## 10 1 39 2.500000  
## 11 1 40 3.000000  
## 12 0 41 2.000000  
## 13 1 41 2.000000  
## 14 0 42 2.000000  
## 15 1 42 2.000000  
## 16 0 43 2.500000  
## 17 1 43 2.500000  
## 18 0 44 2.000000  
## 19 1 44 2.000000  
## 20 0 45 2.000000  
## 21 1 45 2.400000  
## 22 0 46 2.000000  
## 23 1 46 2.750000  
## 24 1 47 2.000000  
## 25 0 48 2.000000  
## 26 1 48 2.500000  
## 27 0 49 2.000000  
## 28 1 49 2.333333  
## 29 0 50 2.000000  
## 30 1 50 2.750000  
## 31 0 51 2.250000  
## 32 1 51 2.375000  
## 33 0 52 2.000000  
## 34 1 52 2.166667  
## 35 0 53 1.333333  
## 36 1 53 2.600000  
## 37 0 54 2.000000  
## 38 1 54 2.636364  
## 39 0 55 2.250000  
## 40 1 55 2.750000  
## 41 0 56 2.666667  
## 42 1 56 2.125000  
## 43 0 57 2.250000  
## 44 1 57 2.384615  
## 45 0 58 1.833333  
## 46 1 58 2.692308  
## 47 0 59 2.000000  
## 48 1 59 2.307692  
## 49 0 60 2.250000  
## 50 1 60 2.857143  
## 51 0 61 2.500000  
## 52 1 61 2.500000  
## 53 0 62 2.285714  
## 54 1 62 2.750000  
## 55 0 63 2.200000  
## 56 1 63 2.500000  
## 57 0 64 2.333333  
## 58 1 64 2.285714  
## 59 0 65 2.250000  
## 60 1 65 2.250000  
## 61 0 66 2.333333  
## 62 1 66 1.500000  
## 63 0 67 2.333333  
## 64 1 67 2.500000  
## 65 0 68 2.000000  
## 66 1 68 3.000000  
## 67 0 69 2.000000  
## 68 1 69 2.500000  
## 69 1 70 2.500000  
## 70 0 71 2.000000  
## 71 0 74 2.000000  
## 72 0 76 2.000000  
## 73 1 77 2.000000

aggregate(heart$target ~ heart$sex + heart$age, FUN = mean)

## heart$sex heart$age heart$target  
## 1 1 29 1.0000000  
## 2 0 34 1.0000000  
## 3 1 34 1.0000000  
## 4 0 35 1.0000000  
## 5 1 35 0.3333333  
## 6 0 37 1.0000000  
## 7 1 37 1.0000000  
## 8 1 38 0.5000000  
## 9 0 39 1.0000000  
## 10 1 39 0.5000000  
## 11 1 40 0.3333333  
## 12 0 41 1.0000000  
## 13 1 41 0.8333333  
## 14 0 42 1.0000000  
## 15 1 42 0.8333333  
## 16 0 43 0.5000000  
## 17 1 43 0.6666667  
## 18 0 44 1.0000000  
## 19 1 44 0.6666667  
## 20 0 45 1.0000000  
## 21 1 45 0.6000000  
## 22 0 46 1.0000000  
## 23 1 46 0.2500000  
## 24 1 47 0.6000000  
## 25 0 48 1.0000000  
## 26 1 48 0.5000000  
## 27 0 49 1.0000000  
## 28 1 49 0.3333333  
## 29 0 50 1.0000000  
## 30 1 50 0.2500000  
## 31 0 51 0.7500000  
## 32 1 51 0.7500000  
## 33 0 52 1.0000000  
## 34 1 52 0.6666667  
## 35 0 53 1.0000000  
## 36 1 53 0.6000000  
## 37 0 54 1.0000000  
## 38 1 54 0.4545455  
## 39 0 55 0.5000000  
## 40 1 55 0.2500000  
## 41 0 56 0.3333333  
## 42 1 56 0.5000000  
## 43 0 57 0.5000000  
## 44 1 57 0.3846154  
## 45 0 58 0.6666667  
## 46 1 58 0.2307692  
## 47 0 59 0.0000000  
## 48 1 59 0.3846154  
## 49 0 60 0.7500000  
## 50 1 60 0.0000000  
## 51 0 61 0.0000000  
## 52 1 61 0.1666667  
## 53 0 62 0.2857143  
## 54 1 62 0.5000000  
## 55 0 63 0.4000000  
## 56 1 63 0.2500000  
## 57 0 64 1.0000000  
## 58 1 64 0.4285714  
## 59 0 65 0.7500000  
## 60 1 65 0.2500000  
## 61 0 66 0.6666667  
## 62 1 66 0.5000000  
## 63 0 67 1.0000000  
## 64 1 67 0.0000000  
## 65 0 68 1.0000000  
## 66 1 68 0.3333333  
## 67 0 69 1.0000000  
## 68 1 69 0.5000000  
## 69 1 70 0.2500000  
## 70 0 71 1.0000000  
## 71 0 74 1.0000000  
## 72 0 76 1.0000000  
## 73 1 77 0.0000000

## Changing the values of few columns which will help us in better under standing of the visualization.

heart$sex <- ifelse(heart$sex == 0, "female", "male")

heart$chest\_pain <- ifelse(heart$chest\_pain == 0, "typical angina",  
 ifelse(heart$chest\_pain == 1, "atypical angina",  
 ifelse(heart$chest\_pain == 2, "non-anginal pain",   
 ifelse(heart$chest\_pain == 3, "asymptomatic", NA))))

heart$fasting\_blood\_sugar <- ifelse(heart$fasting\_blood\_sugar == 1, "true", "false")

heart$restecg <- ifelse(heart$restecg == 0, "normal",  
 ifelse(heart$restecg == 1, "ST-T wave abnormality",  
 ifelse(heart$restecg == 2, "showing probable", NA)))

heart$exercise\_induced\_angina <- ifelse(heart$exercise\_induced\_angina == 1, "yes", "no")

heart$slope <- ifelse(heart$slope == 0, "unsloping",  
 ifelse(heart$slope == 1, "flat",  
 ifelse(heart$slope == 2, "downsloping", NA)))

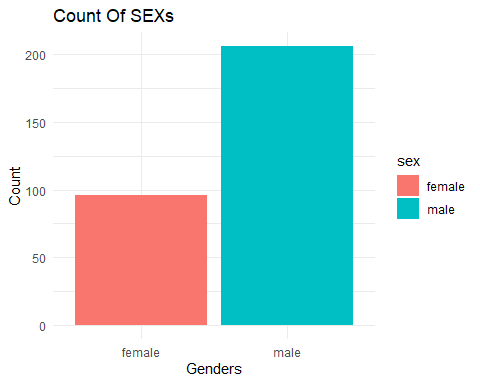
heart$thalassemia <- ifelse(heart$thalassemia == 0, "null",  
 ifelse(heart$thalassemia == 1, "fixed defect",  
 ifelse(heart$thalassemia == 2, "normal",  
 ifelse(heart$thalassemia == 3, "reversable defect", NA))))

heart$target <- ifelse(heart$target == 0, "less chance", "more chance")

## Data Visualization:

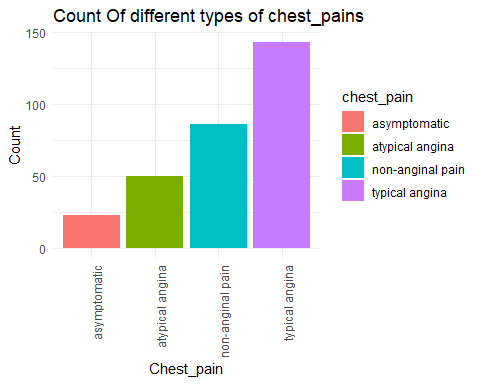
Let’s see the count of total male and female in our dataset.

ggplot(heart, aes(x = sex, fill = sex)) +  
 geom\_bar() +  
 labs(title = "Count Of SEXs", x = "Genders", y = "Count") +  
 theme\_minimal()



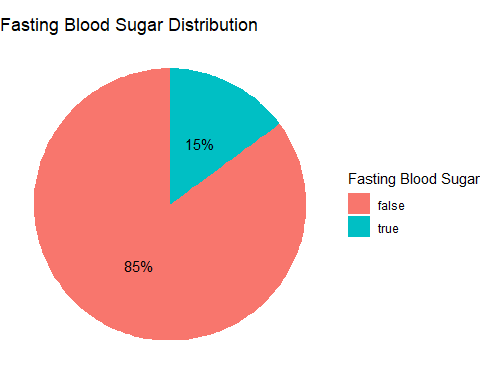
Let’s see the count of different types of chest pain in our dataset.

ggplot(heart, aes(x = chest\_pain, fill = chest\_pain)) +  
 geom\_bar() +  
 labs(title = "Count Of different types of chest\_pains", x = "Chest\_pain", y = "Count") +  
 theme\_minimal() +  
 theme(axis.text.x = element\_text(angle = 90, hjust = 1))



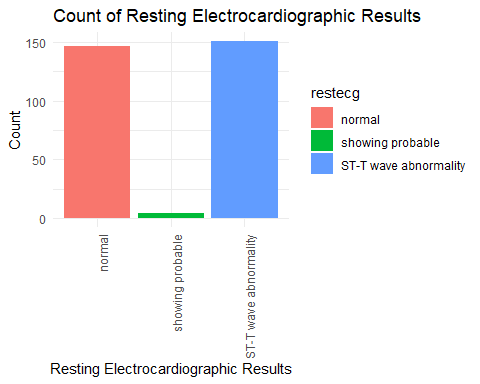
Let’s see the percentage distribution of fasting blood sugars.

data <- as.data.frame(table(heart$fasting\_blood\_sugar))  
pie\_chart <- ggplot(data, aes(x = "", y = Freq, fill = Var1)) +  
 geom\_bar(stat = "identity") +  
 coord\_polar(theta = "y") +  
 geom\_text(aes(label = paste0(round((Freq/sum(Freq))\*100), "%")),  
 position = position\_stack(vjust = 0.5)) +  
 labs(title = "Fasting Blood Sugar Distribution",  
 fill = "Fasting Blood Sugar") +  
 theme\_void()  
print(pie\_chart)



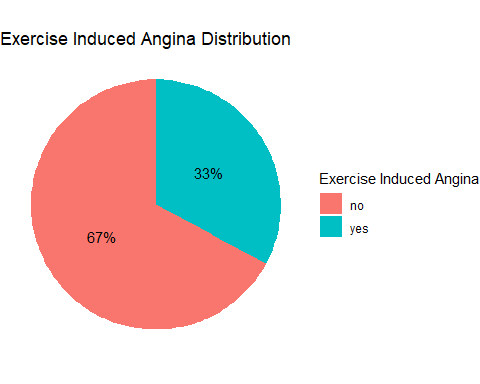
Let’s see the count of resting electrocardiographic results in our dataset.

ggplot(heart, aes(x = restecg, fill = restecg)) +  
 geom\_bar() +  
 labs(title = "Count of Resting Electrocardiographic Results", x = "Resting Electrocardiographic Results", y = "Count") +  
 theme\_minimal() +  
 theme(axis.text.x = element\_text(angle = 90, hjust = 1))



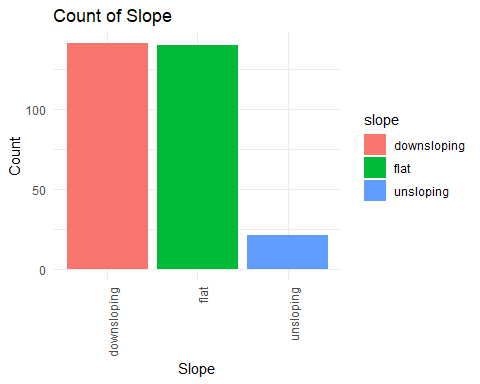
Let’s see the percentage distribution of exercise induced angina in our dataset.

data <- as.data.frame(table(heart$exercise\_induced\_angina))  
pie\_chart <- ggplot(data, aes(x = "", y = Freq, fill = Var1)) +  
 geom\_bar(stat = "identity") +  
 coord\_polar(theta = "y") +  
 geom\_text(aes(label = paste0(round((Freq/sum(Freq))\*100), "%")),  
 position = position\_stack(vjust = 0.5)) +  
 labs(title = "Exercise Induced Angina Distribution",  
 fill = "Exercise Induced Angina") +  
 theme\_void()  
print(pie\_chart)



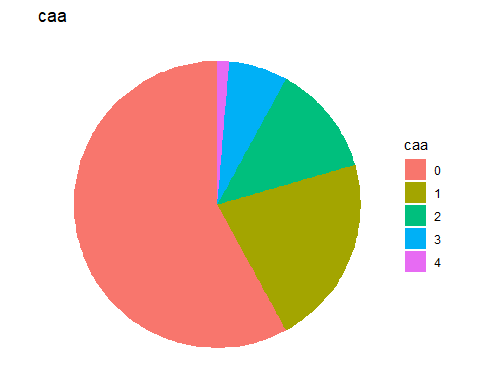
Let’s see the count of slopes in our dataset.

ggplot(heart, aes(x = slope, fill = slope)) +  
 geom\_bar() +  
 labs(title = "Count of Slope", x = "Slope", y = "Count") +  
 theme\_minimal() +  
 theme(axis.text.x = element\_text(angle = 90, hjust = 1))



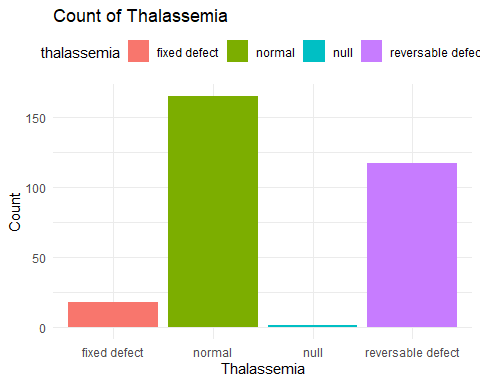
Let’s see the distribution in pie chart of caa.

data <- as.data.frame(table(heart$caa))  
pie\_chart <- ggplot(data, aes(x = "", y = Freq, fill = Var1)) +  
 geom\_bar(stat = "identity") +  
 coord\_polar(theta = "y") +  
 labs(title = "caa",  
 fill = "caa") +  
 theme\_void()  
print(pie\_chart)



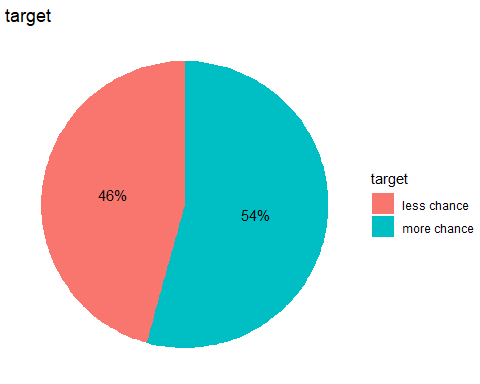
Let’s see the count of different thalassemia in our dataset.

ggplot(heart, aes(x = thalassemia, fill = thalassemia)) +  
 geom\_bar() +  
 labs(title = "Count of Thalassemia", x = "Thalassemia", y = "Count") +  
 theme\_minimal() +  
 theme(legend.position = "top")



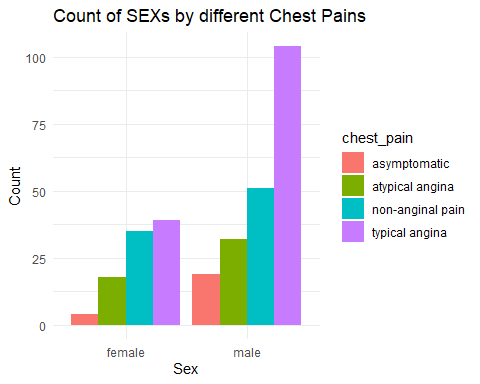
Let’s see the percentage distribution of people having chances of heart attack.

data <- as.data.frame(table(heart$target))  
pie\_chart <- ggplot(data, aes(x = "", y = Freq, fill = Var1)) +  
 geom\_bar(stat = "identity") +  
 coord\_polar(theta = "y") +  
 geom\_text(aes(label = paste0(round((Freq/sum(Freq))\*100), "%")),  
 position = position\_stack(vjust = 0.5)) +  
 labs(title = "target",  
 fill = "target") +  
 theme\_void()  
print(pie\_chart)



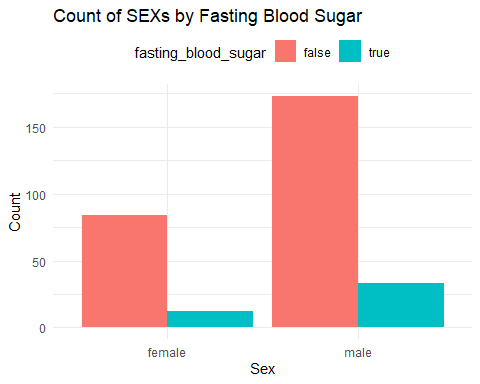
Let’s see the count of chest pains in different genders.

sex\_count <- heart %>%  
 group\_by(sex, chest\_pain) %>%  
 summarise(sex\_count = n(), .groups = "drop")  
ggplot(sex\_count, aes(x = sex, y = sex\_count, fill = chest\_pain)) +  
 geom\_col(position = "dodge") +  
 labs(title = "Count of SEXs by different Chest Pains",  
 x = "Sex",  
 y = "Count") +  
 theme\_minimal()



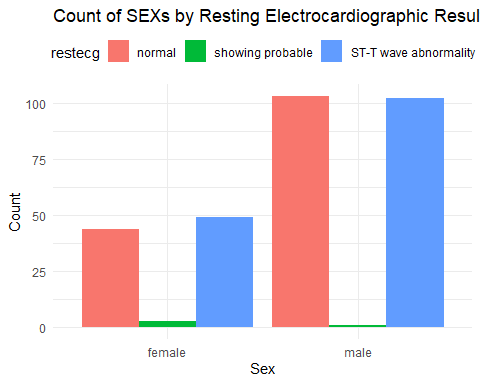
Let’s see the count of fasting blood sugar in different genders.

count\_fbs <- heart %>%  
 group\_by(sex, fasting\_blood\_sugar) %>%  
 summarise(count\_fbs = n(), .groups = "drop")  
ggplot(count\_fbs, aes(x = sex, y = count\_fbs, fill = fasting\_blood\_sugar)) +  
 geom\_col(position = "dodge") +  
 labs(title = "Count of SEXs by Fasting Blood Sugar",  
 x = "Sex",  
 y = "Count") +  
 theme\_minimal() +  
 theme(legend.position = "top")



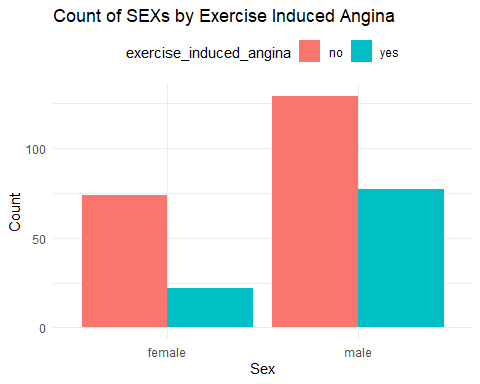
Let’s see the count of resting ecg results in different genders.

count\_restecg <- heart %>%  
 group\_by(sex, restecg) %>%  
 summarise(count\_restecg = n(), .groups = "drop")  
ggplot(count\_restecg, aes(x = sex, y = count\_restecg, fill = restecg)) +  
 geom\_col(position = "dodge") +  
 labs(title = "Count of SEXs by Resting Electrocardiographic Results",  
 x = "Sex",  
 y = "Count") +  
 theme\_minimal() +  
 theme(legend.position = "top")



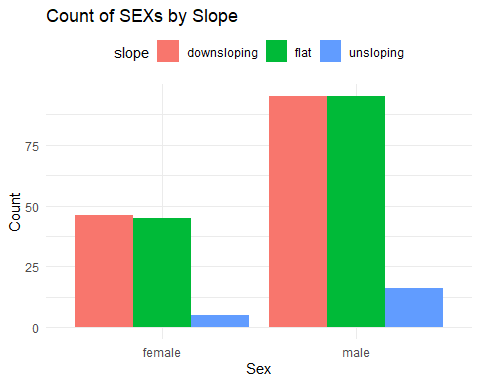
Let’s see the count of exercise induced angina in different genders.

count\_eia <- heart %>%  
 group\_by(sex, exercise\_induced\_angina) %>%  
 summarise(count\_eia = n(), .groups = "drop")  
ggplot(count\_eia, aes(x = sex, y = count\_eia, fill = exercise\_induced\_angina)) +  
 geom\_col(position = "dodge") +  
 labs(title = "Count of SEXs by Exercise Induced Angina",  
 x = "Sex",  
 y = "Count") +  
 theme\_minimal() +  
 theme(legend.position = "top")



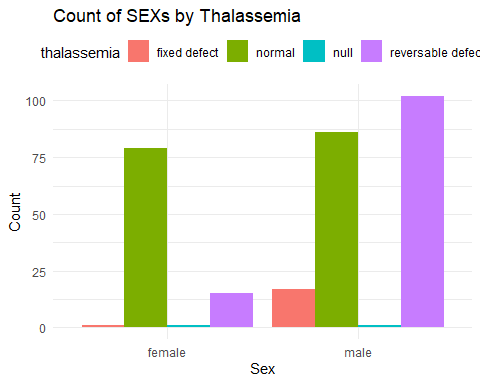
Let’s see the count of slopes in different genders.

count\_slope <- heart %>%  
 group\_by(sex, slope) %>%  
 summarise(count\_slope = n(), .groups = "drop")  
ggplot(count\_slope, aes(x = sex, y = count\_slope, fill = slope)) +  
 geom\_col(position = "dodge") +  
 labs(title = "Count of SEXs by Slope",  
 x = "Sex",  
 y = "Count") +  
 theme\_minimal() +  
 theme(legend.position = "top")



Let’s see the count of different thalassemias in different genders.

count\_thala <- heart %>%  
 group\_by(sex, thalassemia) %>%  
 summarise(count\_thala = n(), .groups = "drop")  
ggplot(count\_thala, aes(x = sex, y = count\_thala, fill = thalassemia)) +  
 geom\_col(position = "dodge") +  
 labs(title = "Count of SEXs by Thalassemia",  
 x = "Sex",  
 y = "Count") +  
 theme\_minimal() +  
 theme(legend.position = "top")



Finally, let’s see the chance of getting heart attack in different genders.

count\_target <- heart %>%  
 group\_by(sex, target) %>%  
 summarise(count\_target = n(), .groups = "drop")  
ggplot(count\_target, aes(x = sex, y = count\_target, fill = target)) +  
 geom\_col(position = "dodge") +  
 labs(title = "Count of SEXs by Target",  
 x = "Sex",  
 y = "Count") +  
 theme\_minimal() +  
 theme(legend.position = "top")

