Exploratory_analysis

Biswajit Chowdhury

28/07/2022

- Here, I used the Heart disease data set from UCI machine learning repository. The dataset contains 14 variables and 303 observations.

Load library

```
library(tidyverse)
```

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.2 --
## v ggplot2 3.3.6     v purrr   0.3.4
## v tibble  3.1.8     v dplyr   1.0.9
## v tidyr   1.2.0     v stringr 1.4.0
## v readr   2.1.2     v forcats 0.5.1
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

Reading data file

```
heart_data<-read.csv("heart.csv")

# see the data set
head(heart_data) # first 6 rows of the data set
```

```
##   age sex cp trestbps chol fbs restecg thalach exang oldpeak slope ca thal
## 1  63   1  3      145  233   1       0     150     0     2.3     0  0    1
## 2  37   1  2      130  250   0       1     187     0     3.5     0  0    2
## 3  41   0  1      130  204   0       0     172     0     1.4     2  0    2
## 4  56   1  1      120  236   0       1     178     0     0.8     2  0    2
## 5  57   0  0      120  354   0       1     163     1     0.6     2  0    2
## 6  57   1  0      140  192   0       1     148     0     0.4     1  0    1
##   target
## 1      1
## 2      1
## 3      1
## 4      1
## 5      1
## 6      1
```

```r
tail(heart_data) # last 6 rows of the data set
```

```
##     age sex cp trestbps chol fbs restecg thalach exang oldpeak slope ca thal
## 298  59   1  0      164  176   1       0      90     0     1.0     1  2    1
## 299  57   0  0      140  241   0       1     123     1     0.2     1  0    3
## 300  45   1  3      110  264   0       1     132     0     1.2     1  0    3
## 301  68   1  0      144  193   1       1     141     0     3.4     1  2    3
## 302  57   1  0      130  131   0       1     115     1     1.2     1  1    3
## 303  57   0  1      130  236   0       0     174     0     0.0     1  1    2
##     target
## 298      0
## 299      0
## 300      0
## 301      0
## 302      0
## 303      0
```

```r
dim(heart_data) # no. of rows and columns
```

```
## [1] 303  14
```

```r
colnames(heart_data) # variable names
```

```
##  [1] "age"      "sex"      "cp"       "trestbps" "chol"     "fbs"
##  [7] "restecg"  "thalach"  "exang"    "oldpeak"  "slope"    "ca"
## [13] "thal"     "target"
```

```r
# change the column names
names(heart_data) <- c("age", "sex", "chest_pain", "resting_bp","cholestrol",
                       "fasting_sugar", "resting_ECG", "max_heart_rate",
                       "exercise_agina", "oldpeak", "slope", "number_major_vessels",
                       "thal", "disease_status")
```

preprocessing data for the exploratory analysis

```r
# Variable sex is coded 0 and 1. We want to include 0=F and 1=M

heart_data$sex <- factor(heart_data$sex,
      levels = c(0,1),
      labels = c("F", "M"))

# Change variable chest_pain into four categories
heart_data$chest_pain <- factor(heart_data$chest_pain,
      levels = c(1,2,3,0),
      labels = c("Typical_angina", "Atypical_angina", "Nonanginal_pain", "Assymptomatic"))

# Change variable fasting_sugar to high and low
```

```
heart_data$fasting_sugar <- factor(heart_data$fasting_sugar,
      levels = c(0,1),
      labels = c("Low", "High"))

# Change resting_ECG into normal, mild and severe

heart_data$resting_ECG <- factor(heart_data$resting_ECG,
      levels = c(0,1,2),
      labels = c("Normal", "Mild", "Severe"))

# Change excercise_angina into "yes" and "no"

heart_data$exercise_agina <- factor(heart_data$exercise_agina,
      levels = c(0,1),
      labels = c("No", "Yes"))

# Similary we also change the labels for number of major vessels and thal


heart_data$number_major_vessels <- factor(heart_data$number_major_vessels,
      levels = c(0,1,2,3,4),
      labels = c("None", "One", "Two", "Three", "Four"))

heart_data$thal <- factor(heart_data$thal,
      levels = c(1, 2,3),
      labels = c("Normal", "Fixed_defect", "Reversible_defect"))

# Finally change the angiographic disease status to normal and disease

heart_data$disease_status <- factor(heart_data$disease_status,
      levels = c(0,1),
      labels = c("Normal", "Disease"))

write.csv(heart_data, "UCI_heart_data.csv")
head(heart_data,3)
```

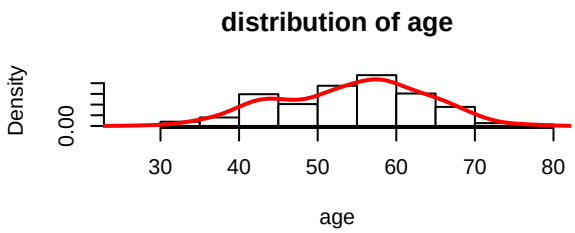
```
##   age sex      chest_pain resting_bp cholestrol fasting_sugar resting_ECG
## 1  63   M Nonanginal_pain        145        233          High      Normal
## 2  37   M Atypical_angina        130        250           Low        Mild
## 3  41   F  Typical_angina        130        204           Low      Normal
##   max_heart_rate exercise_agina oldpeak slope number_major_vessels         thal
## 1            150             No     2.3     0                 None       Normal
## 2            187             No     3.5     0                 None Fixed_defect
## 3            172             No     1.4     2                 None Fixed_defect
##   disease_status
## 1        Disease
## 2        Disease
## 3        Disease
```

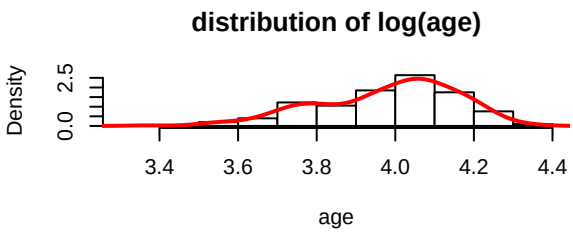
Explore the quantitative variables

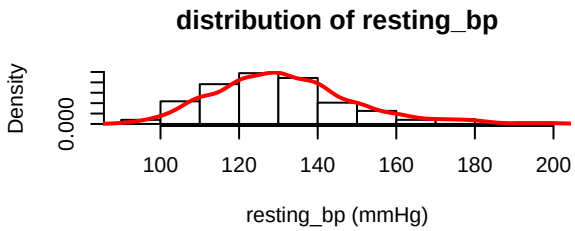
Histogram and density plot for the quantitative variables

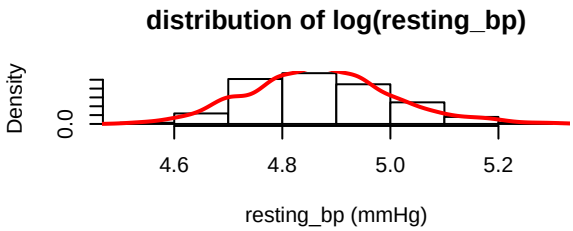
```
par(mfrow=c(3,2))
hist(heart_data$age,
     col="white",
     border="black",
     prob = TRUE,
     xlab = "age",
     main = "distribution of age")
  lines(density(heart_data$age),
      lwd = 2,
      col = "red")

hist(log(heart_data$age),
     col="white",
     border="black",
     prob = TRUE,
     xlab = "age",
     main = "distribution of log(age)")
  lines(density(log(heart_data$age)),
      lwd = 2,
      col = "red")

hist(heart_data$resting_bp,
     col="white",
     border="black",
     prob = TRUE,
     xlab = "resting_bp (mmHg)",
     main = "distribution of resting_bp")
  lines(density(heart_data$resting_bp),
      lwd = 2,
      col = "red")

hist(log(heart_data$resting_bp),
     col="white",
     border="black",
     prob = TRUE,
     xlab = "resting_bp (mmHg)",
     main = "distribution of log(resting_bp)")
  lines(density(log(heart_data$resting_bp)),
      lwd = 2,
      col = "red")

hist(heart_data$cholestrol,
     col="white",
     border="black",
     prob = TRUE,
     xlab = "cholestrol levels (mg/dl)",
     main = "distribution of cholestrol levels")
  lines(density(heart_data$cholestrol),
```

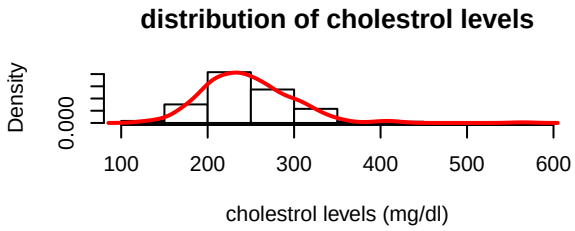
```
     lwd = 2,
     col = "red")

hist(log(heart_data$cholestrol),
    col="white",
    border="black",
    prob = TRUE,
    xlab = "cholestrol levels (mg/dl)",
    main = "distribution of log(cholestrol levels)")
 lines(density(log(heart_data$cholestrol)),
     lwd = 2,
     col = "red")
```

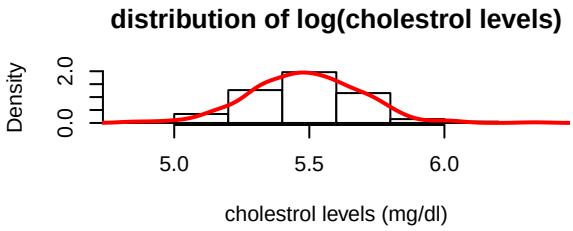






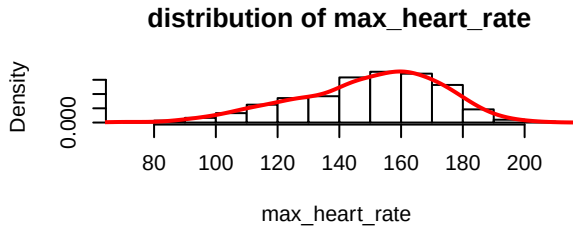


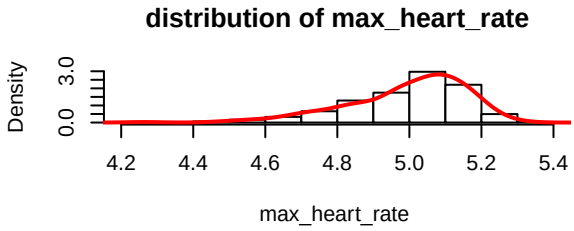





```
hist(heart_data$max_heart_rate,
    col="white",
    border="black",
    prob = TRUE,
    xlab = "max_heart_rate ",
    main = "distribution of max_heart_rate")
lines(density(heart_data$max_heart_rate),
     lwd = 2,
     col = "red")

hist(log(heart_data$max_heart_rate),
    col="white",
```

```
    border="black",
    prob = TRUE,
    xlab = "max_heart_rate ",
    main = "distribution of max_heart_rate")
lines(density(log(heart_data$max_heart_rate)),
     lwd = 2,
     col = "red")
```





Shapiro-Wilk test for checking the normal distribution

- null hypothesis: the data are normally distributed
- alternative hypothesis: the data are not normally distributed

```
# normality test for cholestrol
shapiro.test(heart_data$cholestrol)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  heart_data$cholestrol
## W = 0.94688, p-value = 5.365e-09
```

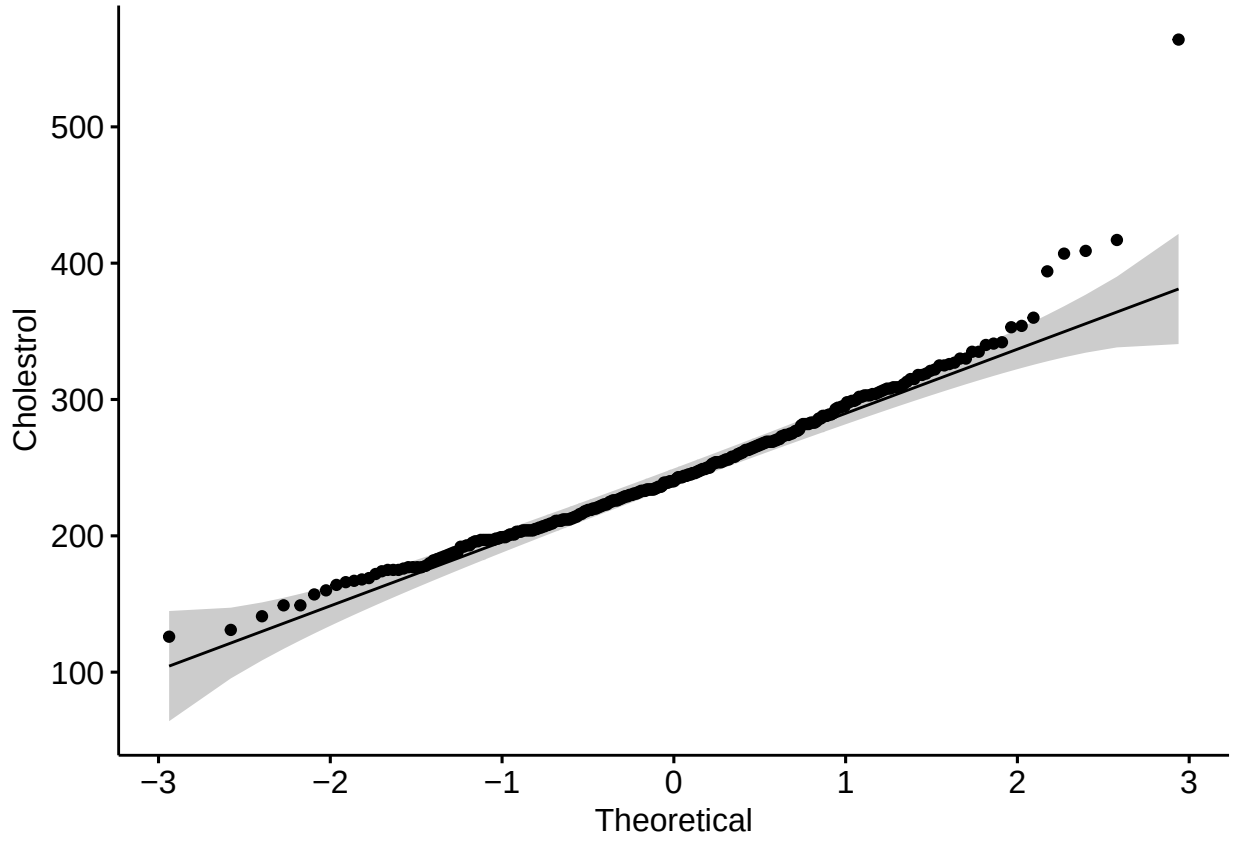
```
# normality test for heart ratel
shapiro.test(heart_data$max_heart_rate)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  heart_data$max_heart_rate
## W = 0.97632, p-value = 6.621e-05
```

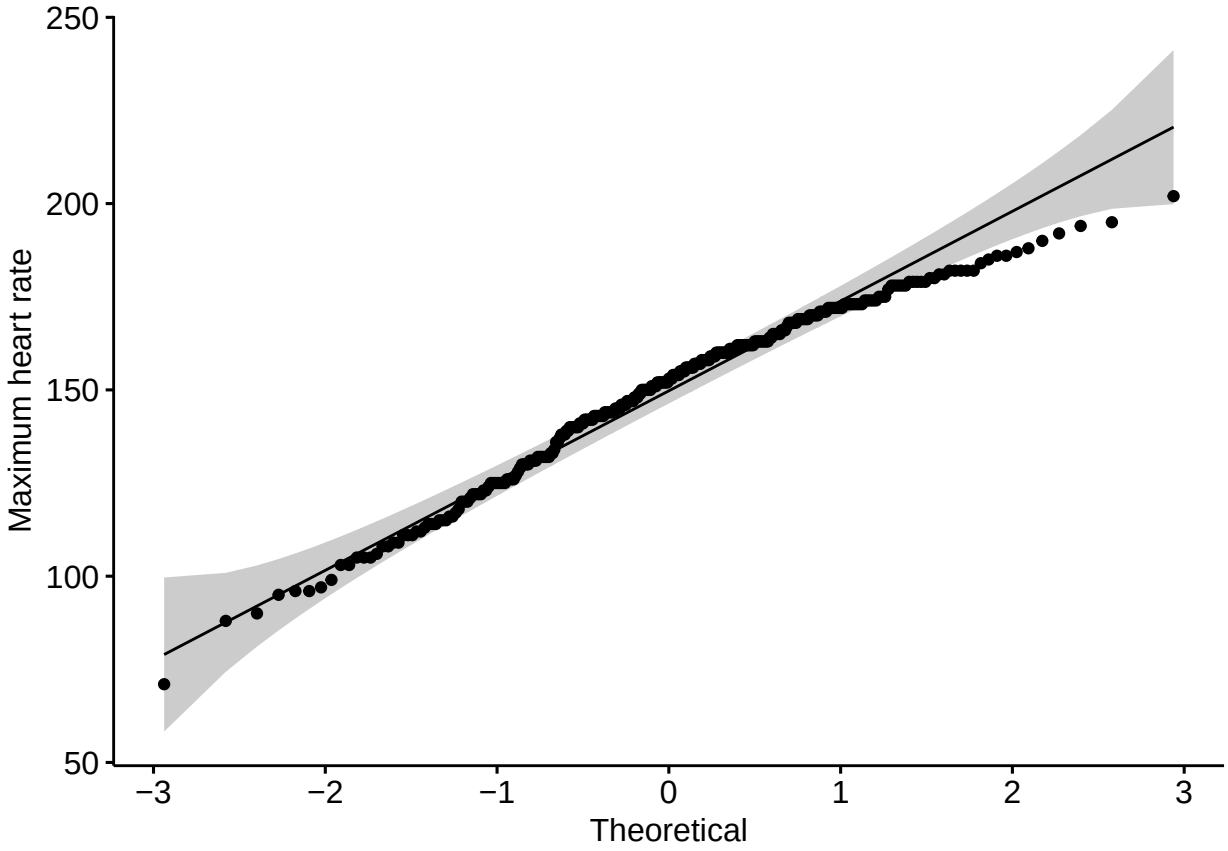
- from the above p-value, we can say the data are not normally distributed.

Visual inspection of the data normality using Q-Q plots (quantile-quantile plots). Q-Q plot draws the correlation between a given sample and the normal distribution.

```
library(ggpubr)
ggqqplot(heart_data$cholestrol, ylab="Cholestrol")
```




```
ggqqplot(heart_data$max_heart_rate, ylab="Maximum heart rate")
```



Pearson correlation test

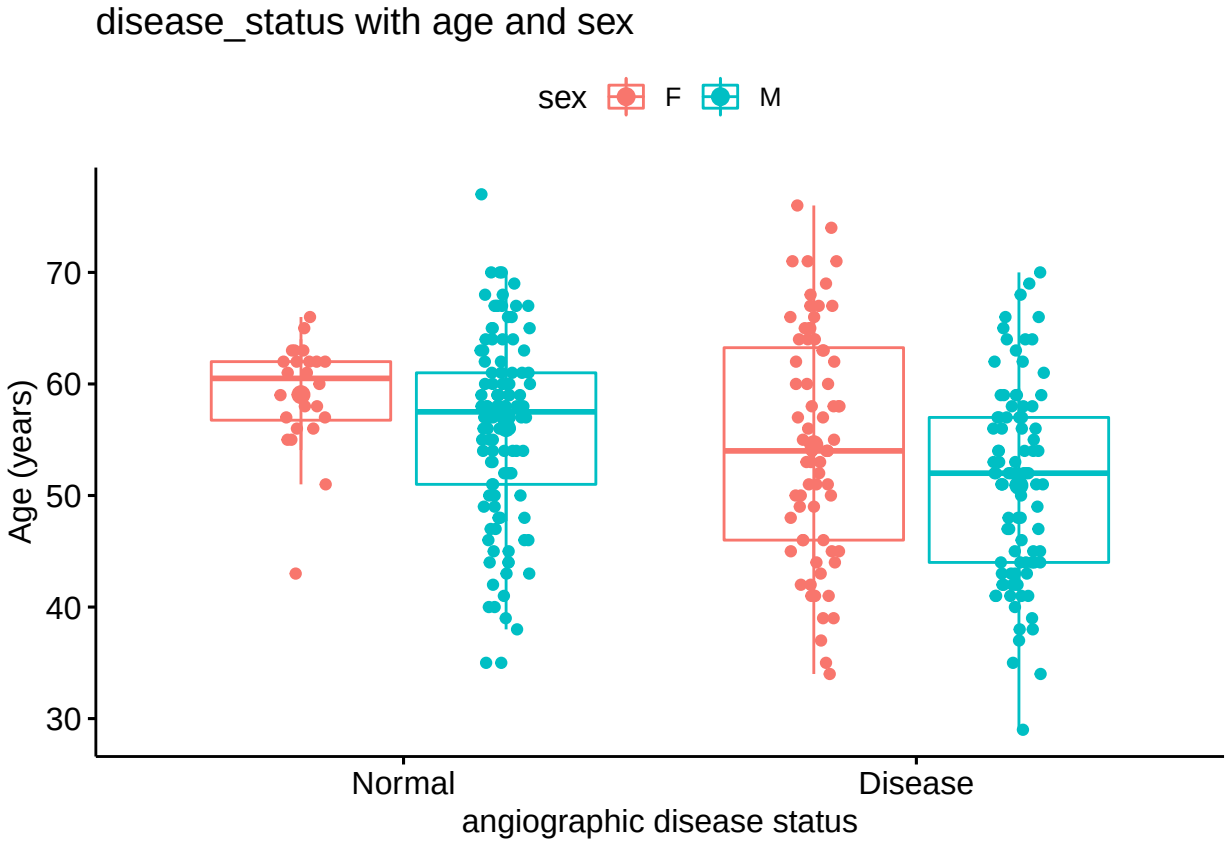
- Correlation between cholestrol and heart_rate

```
# we can use three different methods: pearson, kendal, spearman
correlation<-cor.test(heart_data$cholestrol, heart_data$max_heart_rate, method="pearson")
correlation
```

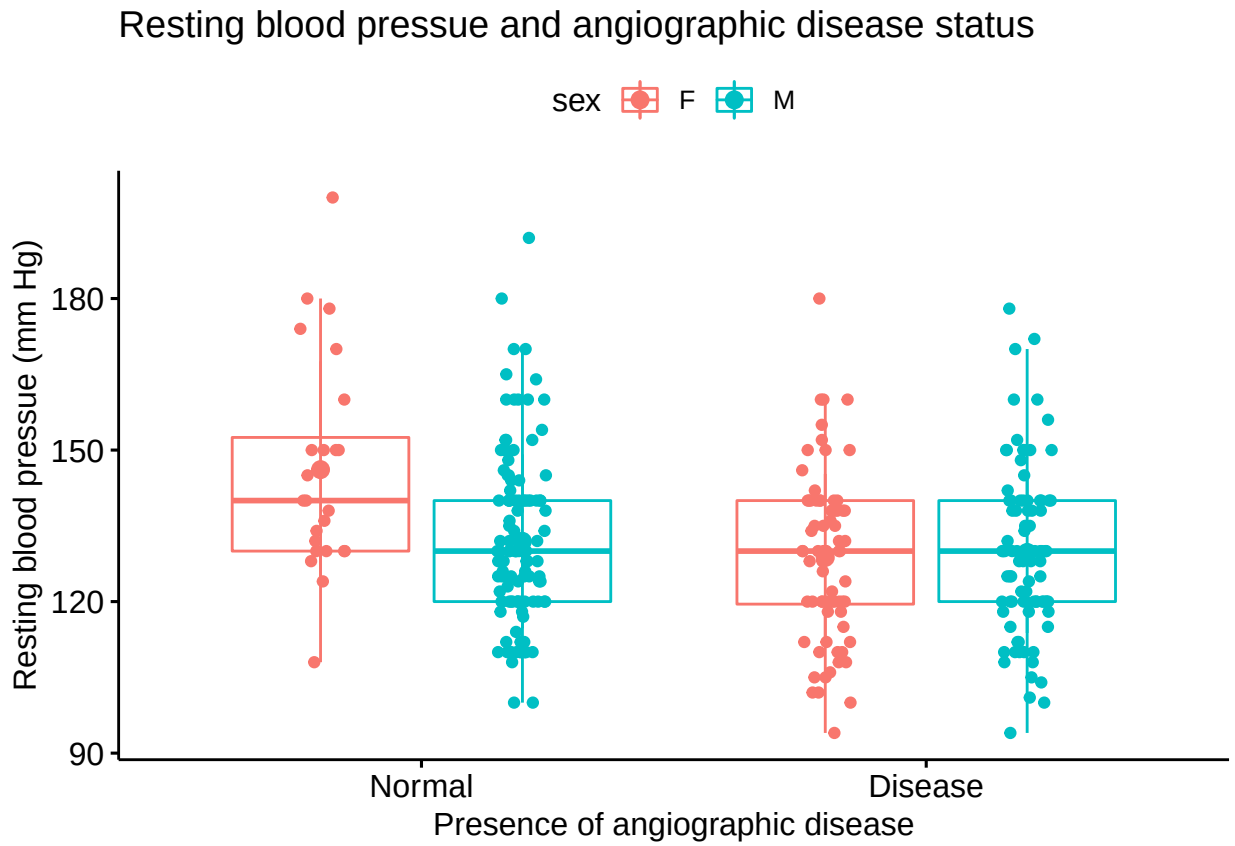
```
##
##  Pearson's product-moment correlation
##
## data:  heart_data$cholestrol and heart_data$max_heart_rate
## t = -0.17246, df = 301, p-value = 0.8632
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  -0.1224807  0.1028534
## sample estimates:
##          cor
## -0.009939839
```

p-value is greater than the significance level alpha =0.05. Therefore, we can conclude that above two variables are not significantly correlated with a correlation coefficiengt of -0.009 and the p-value of 0.8632.

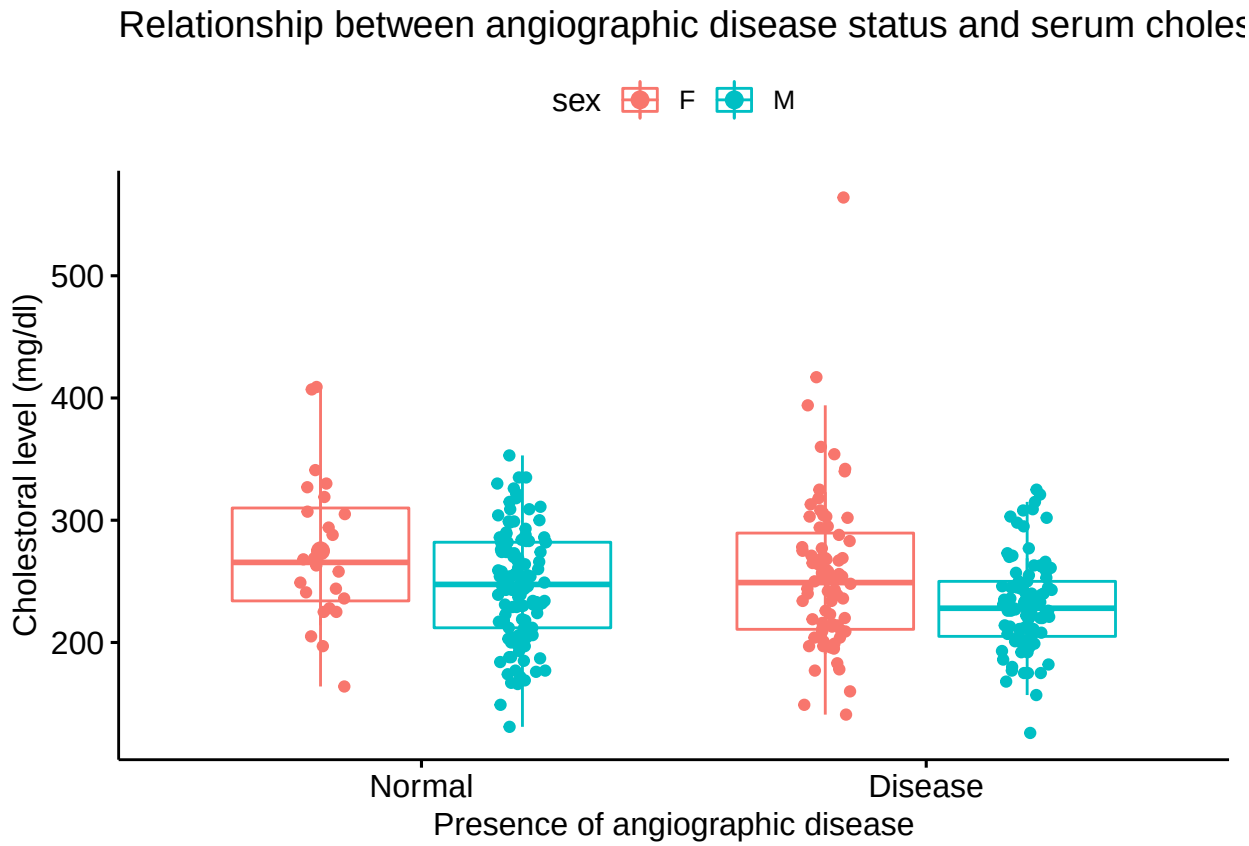
```
library(ggpubr)

# Presence of angiographic disease status among males and females due to age

ggboxplot(heart_data, x="disease_status", y="age",
          xlab="angiographic disease status", ylab= "Age (years)",
          title= "disease_status with age and sex",
          color = "sex", # color by gender
          add = c("jitter", "mean_sd")) # show the distribution of the observed values
```



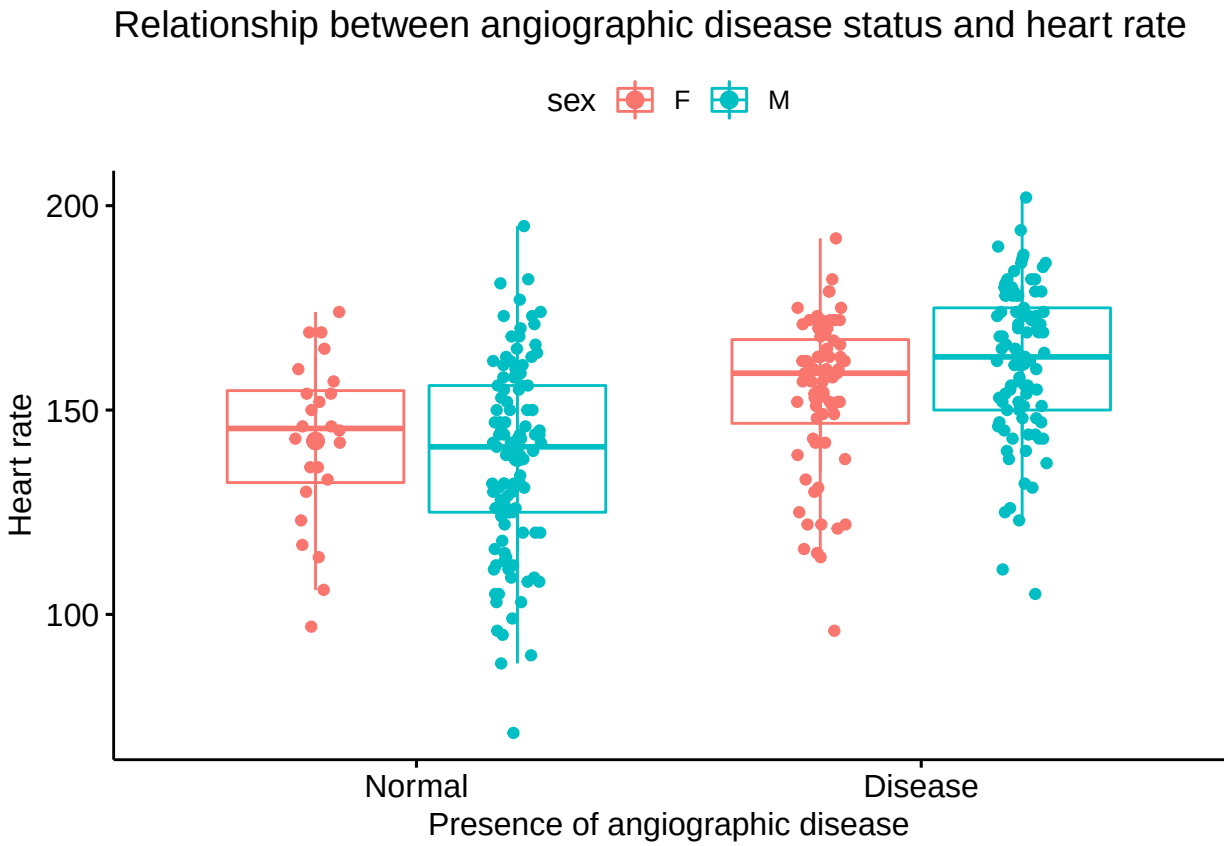

```
# Presence of angiographic disease status among males and females due to resting blood pressue

ggboxplot(heart_data, x = "disease_status", y = "resting_bp",
          xlab = "Presence of angiographic disease", ylab = "Resting blood pressue (mm Hg)",
          title = "Resting blood pressue and angiographic disease status",
          color = "sex", add = c("jitter", "mean_sd"))
```




```
#par(mfrow=c(1,2))
# Changes of chlestreol level between angiographic disease status and normal people

ggboxplot(heart_data, x = "disease_status", y = "cholestrol",
          xlab = "Presence of angiographic disease", ylab = "Cholestoral level (mg/dl)",
          title = "Relationship between angiographic disease status and serum cholestoral",
          color = "sex", add = c("jitter", "mean_sd"))
```




```
# Chnages of heart rate between normal and angiographic disease

ggboxplot(heart_data, x = "disease_status", y = "max_heart_rate",
          xlab = "Presence of angiographic disease", ylab = "Heart rate",
          title = "Relationship between angiographic disease status and heart rate",
          color = "sex", add = c("jitter", "mean_sd"))
```



Explore the categorical variables

- Calculate the frequency of categorical variables

```
# Counts for the sex categories

table(heart_data$sex)
```

```
##
##   F   M
##  96 207
```

```
# cross classification for sex with angiographic disease status
table(heart_data$disease_status, heart_data$sex)
```

```
##
##              F   M
##   Normal    24 114
##   Disease   72  93
```

- Assess subjects by disease type, gender, fasting sugar and chest pain

```
# Multidimensional tables based on three or more categorical variables

table1 <- table(heart_data$disease_status, heart_data$sex,
                heart_data$fasting_sugar)

ftable(table1) # print the results with more attractively
```

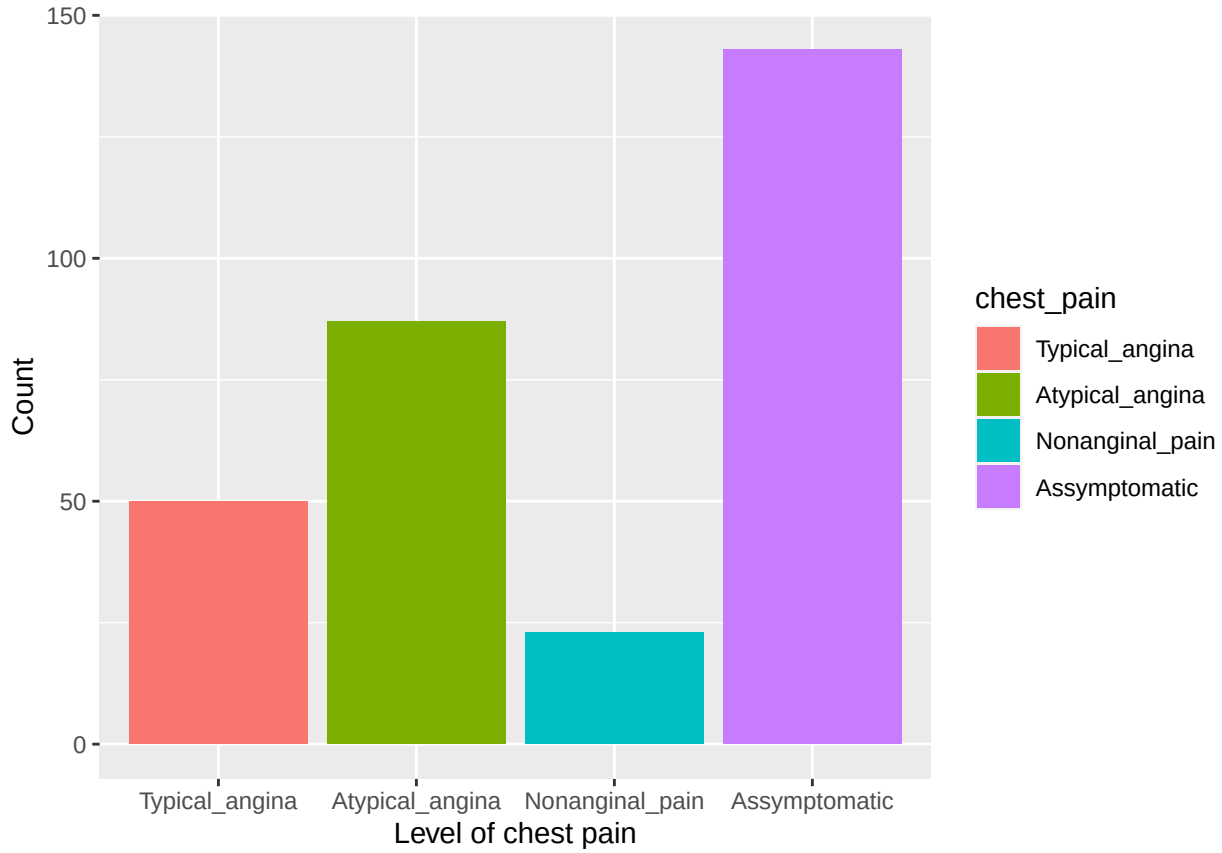
```
##            Low High
##
## Normal  F   18    6
##         M   98   16
## Disease F   66    6
##         M   76   17
```

```
table2 <- table(heart_data$disease_status, heart_data$sex,
                heart_data$chest_pain)

ftable(table2)
```

```
##            Typical_angina Atypical_angina Nonanginal_pain Assymptomatic
##
## Normal  F               2               1               0            21
##         M               7              17               7            83
## Disease F              16              34               4            18
##         M              25              35              12            21
```

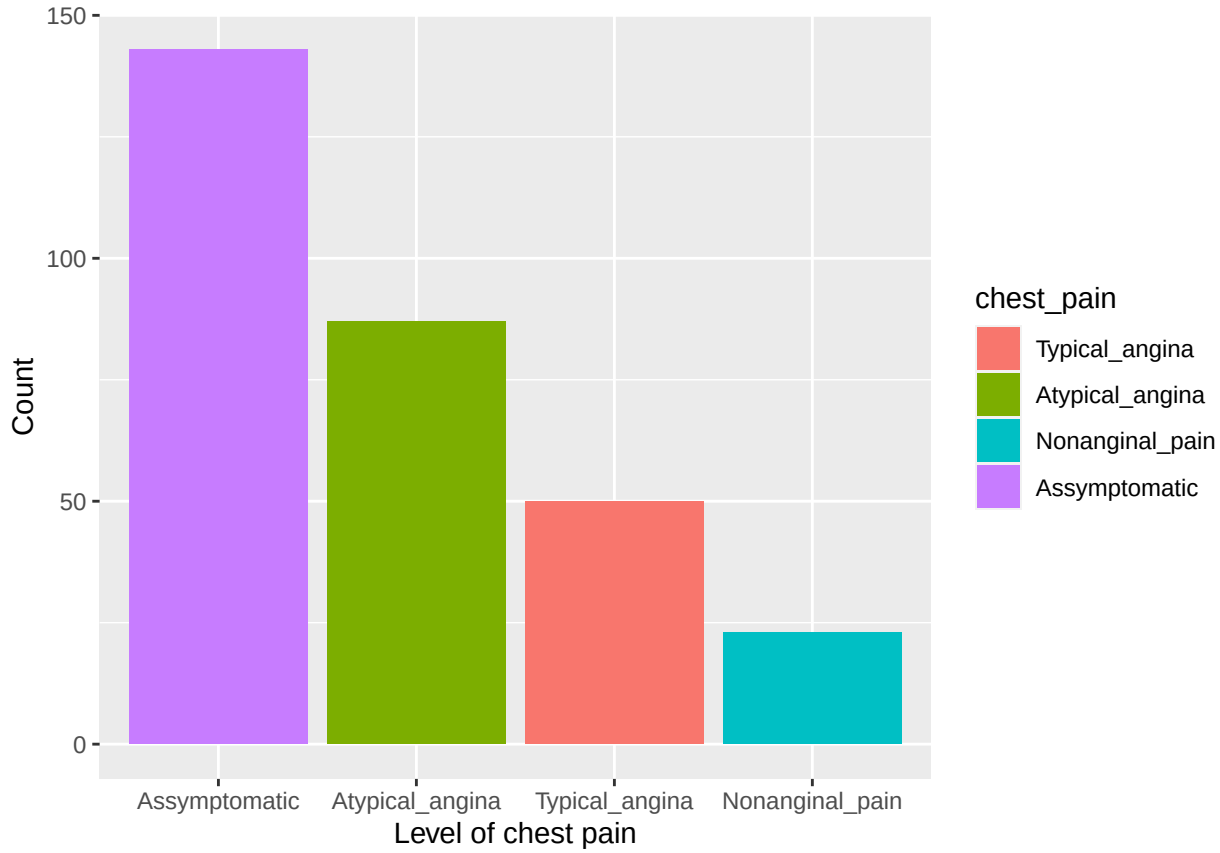
Visualization of the categorical variables

```
# Here we assess the number of subjects diagnosed with chest pain
ggplot(heart_data, aes(x = chest_pain, fill=chest_pain)) +
        geom_bar() +  xlab('Level of chest pain') + ylab("Count")
```



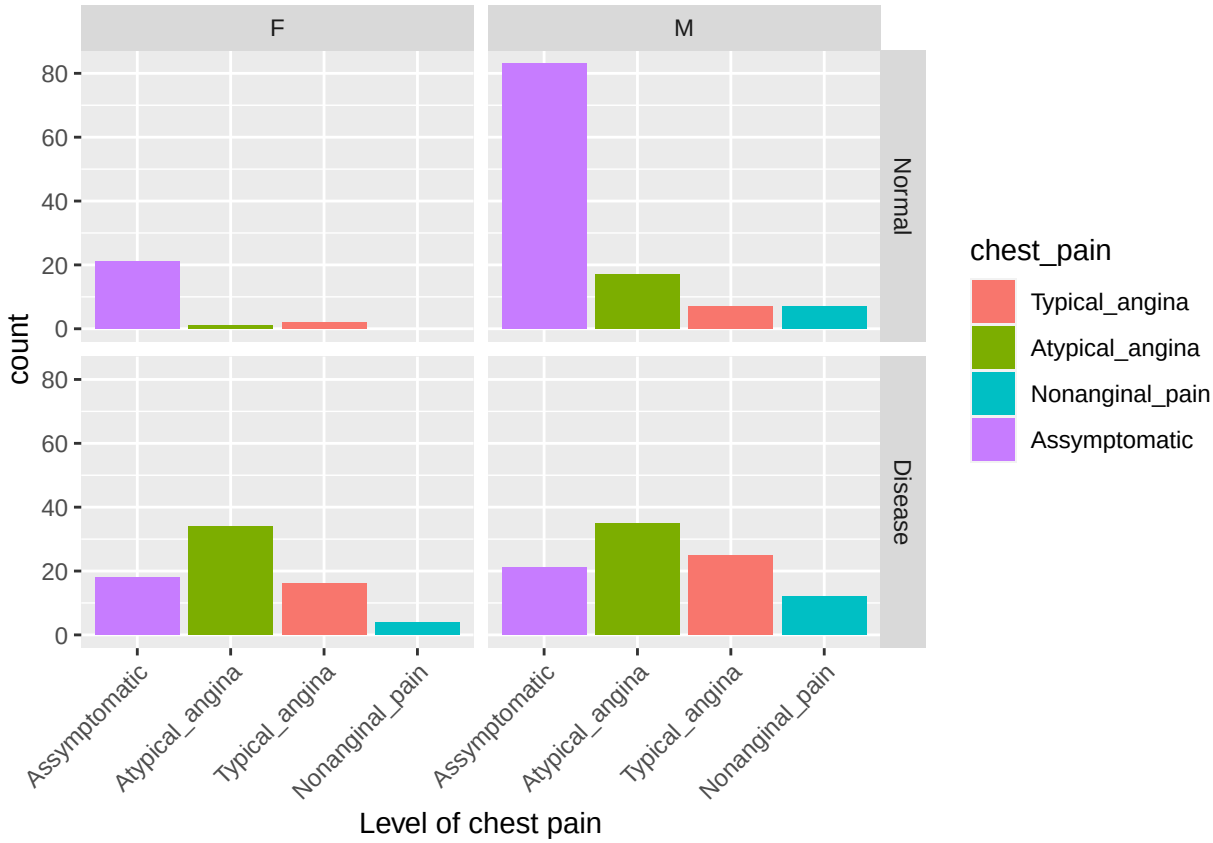
- To easy to digest of the above image, we can make it in desending order. For this we will make a functions that sort the variables crosponding their total counts

```
# re-order levels
reorder_size <- function(x) {
        factor(x, levels = names(sort(table(x), decreasing = TRUE)))
}

ggplot(heart_data, aes(x = reorder_size(chest_pain), fill=chest_pain)) +
        geom_bar() +
  xlab('Level of chest pain') + ylab("Count")
```



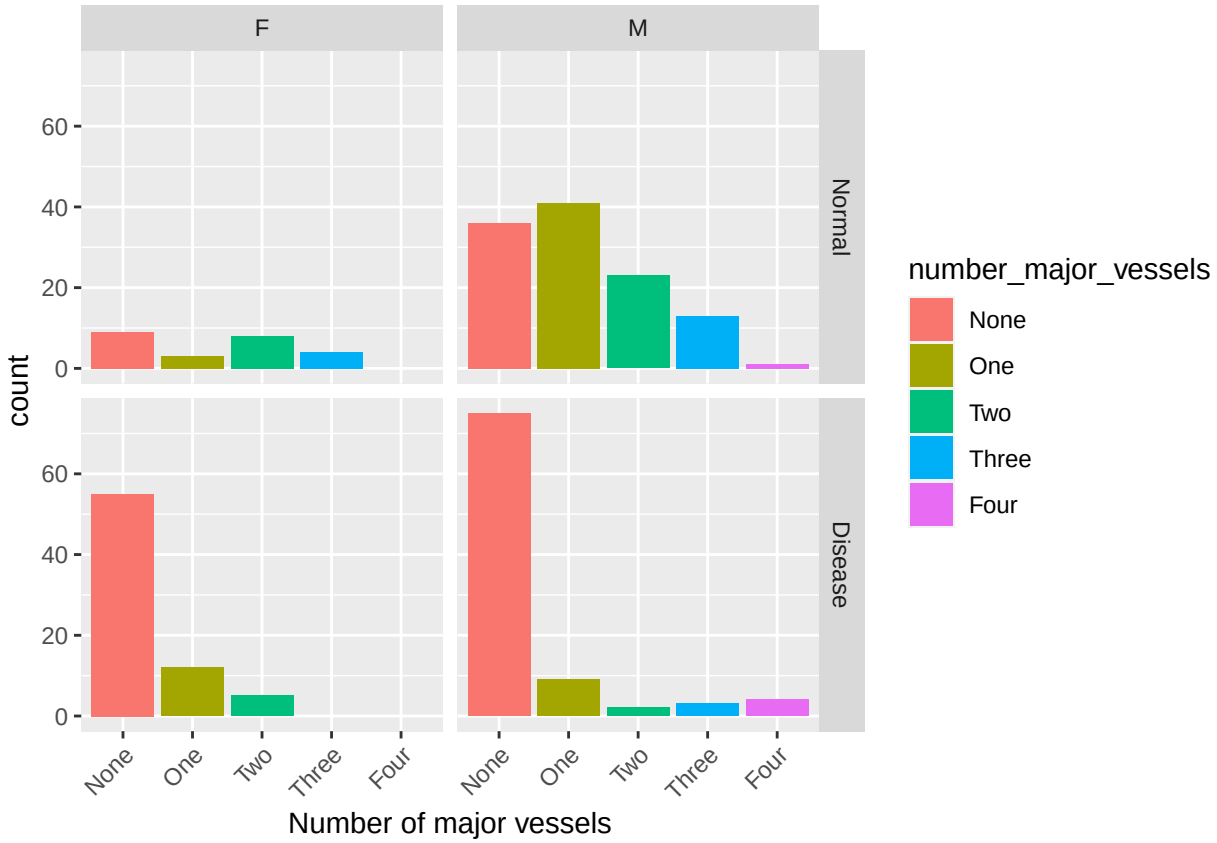
Changes of chest pain with sex and disease status

```
ggplot(heart_data, aes(x = reorder_size(chest_pain), fill=chest_pain)) +
       geom_bar() +
  xlab('Level of chest pain') +
  facet_grid(disease_status~ sex)+
  theme(axis.text.x = element_text(angle = 45, hjust = 1))
```



Number of major vessels with gender and heart disease

```
ggplot(heart_data, aes(x = reorder_size(number_major_vessels), fill=number_major_vessels)) +
       geom_bar() +
  xlab('Number of major vessels') +
  facet_grid(disease_status~ sex) +
       theme(axis.text.x = element_text(angle = 45, hjust = 1))
```



Proportion/Percentage

```
# percentages of gender categories
table1<- table(heart_data$sex)
prop.table(table1)
```

```
##
##         F         M
## 0.3168317 0.6831683
```

```
# percentage of cross classication counts for gender by disease types

table2<- table(heart_data$disease_status, heart_data$sex)
prop.table(table2)
```

```
##
##                    F          M
##   Normal  0.07920792 0.37623762
##   Disease 0.23762376 0.30693069
```

```
round(prop.table(table2), 3)*100
```

```
##
##               F    M
##   Normal   7.9 37.6
##   Disease 23.8 30.7
```

co-relation plot between quantitative variables

```
library(corrplot)
```

```
## corrplot 0.92 loaded
```

```
corrplot(cor(heart_data[, c(1,4,5,8,10)]), type="upper")
```