Cardio Disease prediction

```r
##dplyr is a package which provides a set of tools for efficiently manipulating datasets
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
```

```r
##plyr is a package that makes it simple to split data apart, do stuff to it, and mash it back together
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
```

```r
##ggplot2 is a system for declaratively creating graphics, based on The Grammar of Graphics.
library(ggplot2)
library(tidyverse)
```

```
## -- Attaching packages ---------------------------------------------------------------------------

## v tibble  3.0.3     v purrr   0.3.4
## v tidyr   1.1.1     v stringr 1.4.0
## v readr   1.3.1     v forcats 0.5.0
```

```
## -- Conflicts -------------------------------------------------------------------------------------
## x plyr::arrange()   masks dplyr::arrange()
## x purrr::compact()  masks plyr::compact()
## x plyr::count()     masks dplyr::count()
## x plyr::failwith()  masks dplyr::failwith()
## x dplyr::filter()   masks stats::filter()
## x plyr::id()        masks dplyr::id()
## x dplyr::lag()      masks stats::lag()
## x plyr::mutate()    masks dplyr::mutate()
## x plyr::rename()    masks dplyr::rename()
## x plyr::summarise() masks dplyr::summarise()
## x plyr::summarize() masks dplyr::summarize()
```

```
##The caret package (short for Classification And REgression Training) contains functions to streamline
library(caret)
```

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

```
##
## Attaching package: 'caret'
```

```
## The following object is masked from 'package:purrr':
##
##     lift
```

```
##The corrplot package is a graphical display of a correlation matrix, confidence interval.
library(corrplot)
```

```
## corrplot 0.84 loaded
```

```
library(psych)
```

```
## Warning: package 'psych' was built under R version 4.0.3
```

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

```
## The following objects are masked from 'package:ggplot2':
##
##     %+%, alpha
```

```
library(multcomp)
```

```
## Loading required package: mvtnorm
```

```
## Loading required package: survival
```

```
##
## Attaching package: 'survival'
```

```
## The following object is masked from 'package:caret':
##
##     cluster

## Loading required package: TH.data

## Loading required package: MASS

##
## Attaching package: 'MASS'

## The following object is masked from 'package:dplyr':
##
##     select

##
## Attaching package: 'TH.data'

## The following object is masked from 'package:MASS':
##
##     geyser
```

```r
library(caret)
library(e1071)
library(rpart)
library(rpart.plot)
```

```
## Warning: package 'rpart.plot' was built under R version 4.0.3
```

##In this project we are using Cardio Vascular disease dataset. ##The dataset is imported to the cardio.

```r
cardio <- read.csv("D:/sem-7/FDA/cardio_train.csv", sep = ";")
head(cardio)
```

```
##   id   age gender height weight ap_hi ap_lo cholesterol gluc smoke alco active
## 1  0 18393      2    168     62   110    80           1    1     0    0      1
## 2  1 20228      1    156     85   140    90           3    1     0    0      1
## 3  2 18857      1    165     64   130    70           3    1     0    0      0
## 4  3 17623      2    169     82   150   100           1    1     0    0      1
## 5  4 17474      1    156     56   100    60           1    1     0    0      0
## 6  8 21914      1    151     67   120    80           2    2     0    0      0
##   cardio
## 1      0
## 2      1
## 3      1
## 4      1
## 5      0
## 6      0
```

```r
##Here we are checking for null values in the dataset by using is.na()
colSums(is.na(cardio))
```

```
##          id         age      gender      height      weight       ap_hi
##           0           0           0           0           0           0
##       ap_lo cholesterol        gluc       smoke        alco      active
##           0           0           0           0           0           0
##      cardio
##           0
```

```r
##str() gives sructure of the dataset.
str(cardio)
```

```
## 'data.frame':    70000 obs. of  13 variables:
##  $ id         : int  0 1 2 3 4 8 9 12 13 14 ...
##  $ age        : int  18393 20228 18857 17623 17474 21914 22113 22584 17668 19834 ...
##  $ gender     : int  2 1 1 2 1 1 1 2 1 1 ...
##  $ height     : int  168 156 165 169 156 151 157 178 158 164 ...
##  $ weight     : num  62 85 64 82 56 67 93 95 71 68 ...
##  $ ap_hi      : int  110 140 130 150 100 120 130 130 110 110 ...
##  $ ap_lo      : int  80 90 70 100 60 80 80 90 70 60 ...
##  $ cholesterol: int  1 3 3 1 1 2 3 3 1 1 ...
##  $ gluc       : int  1 1 1 1 1 2 1 3 1 1 ...
##  $ smoke      : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ alco       : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ active     : int  1 1 0 1 0 0 1 1 1 0 ...
##  $ cardio     : int  0 1 1 1 0 0 0 1 0 0 ...
```

###As from the above we can conclude that there is no null values in the dataset

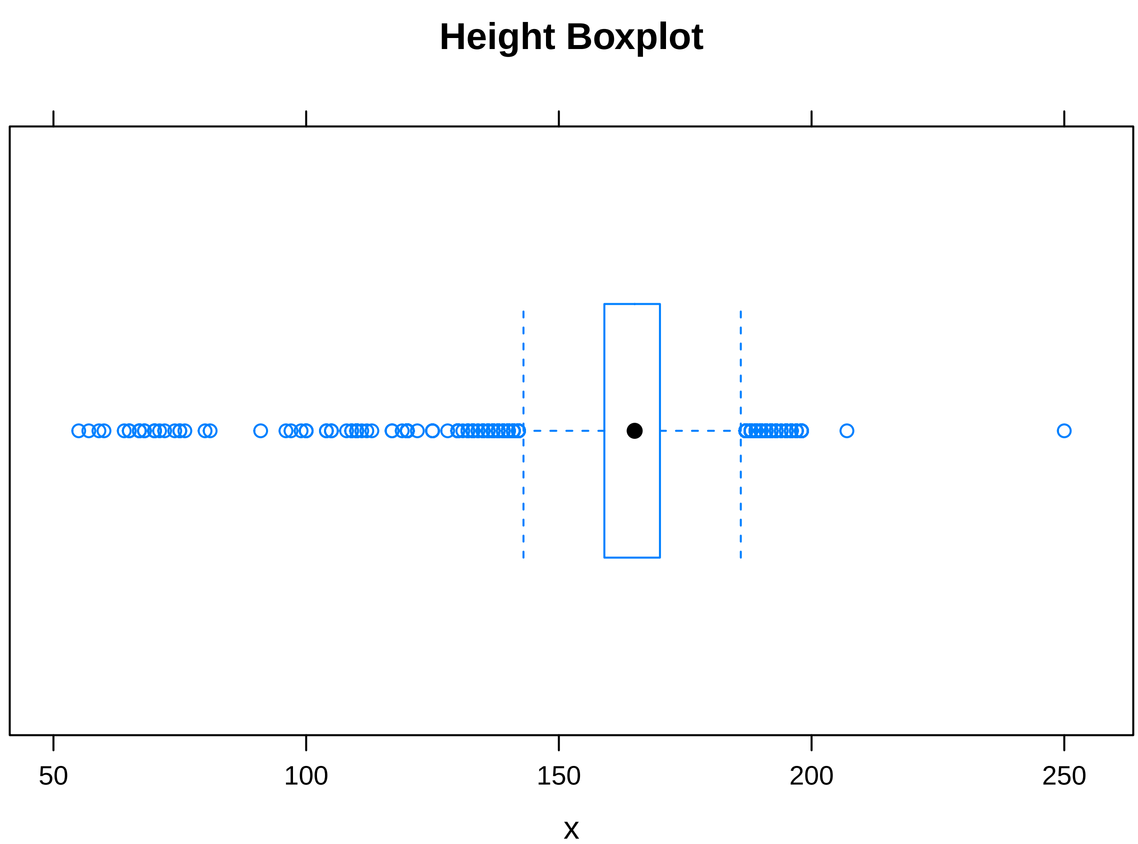
```r
##Removing the first attribute because it won't be used to predict the cardio disease and doesn't impac
cardio1 <- cardio[, 2:13]
head(cardio1)
```

```
##     age gender height weight ap_hi ap_lo cholesterol gluc smoke alco active
## 1 18393      2    168     62   110    80           1    1     0    0      1
## 2 20228      1    156     85   140    90           3    1     0    0      1
## 3 18857      1    165     64   130    70           3    1     0    0      0
## 4 17623      2    169     82   150   100           1    1     0    0      1
## 5 17474      1    156     56   100    60           1    1     0    0      0
## 6 21914      1    151     67   120    80           2    2     0    0      0
##   cardio
## 1      0
## 2      1
## 3      1
## 4      1
## 5      0
## 6      0
```

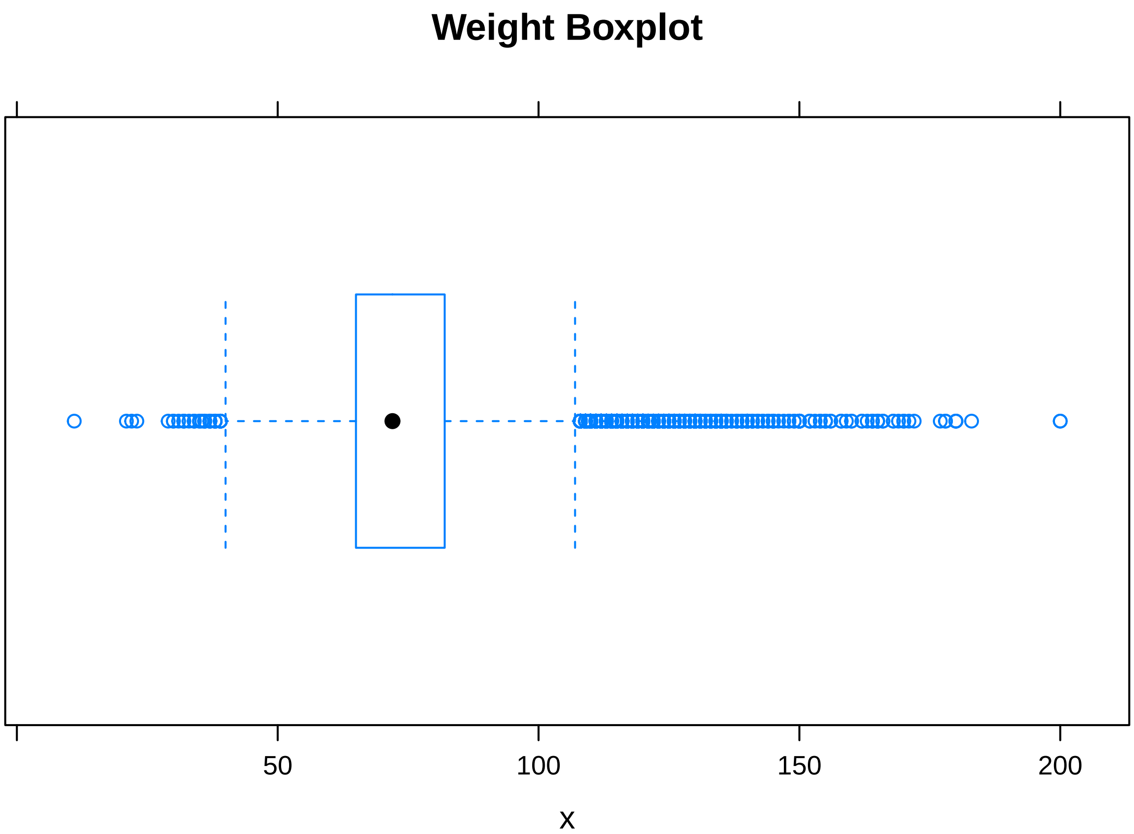
```r
##Manuplating the dataset.
cardio1$age <- as.numeric(cardio1$age)
```

```
cardio1$gender <- as.numeric(cardio1$gender)
cardio1$height<- as.numeric(cardio1$height)
cardio1$weight <- as.numeric(cardio1$weight)
cardio1$ap_hi <- as.numeric(cardio1$ap_hi)
cardio1$ap_lo <- as.numeric(cardio1$ap_lo)
cardio1$cholesterol<- as.numeric(cardio1$cholesterol)
cardio1$gluc <- as.numeric(cardio1$gluc)
cardio1$smoke <- as.numeric(cardio1$smoke)
cardio1$alco <- as.numeric(cardio1$alco)
cardio1$active <- as.numeric(cardio1$active)



cardio1$cardio[cardio1$cardio == 1] <- "Yes"
cardio1$cardio[cardio1$cardio == 0] <- "No"

cardio1$cardio <- as.factor(cardio1$cardio)


##Remove the rows with systolic blood pressure lower than diastolic blood pressure i.e. ap_hi < ap_lo
ap_cleaned <- cardio1 %>% filter(cardio1$ap_hi > cardio1$ap_lo)
```

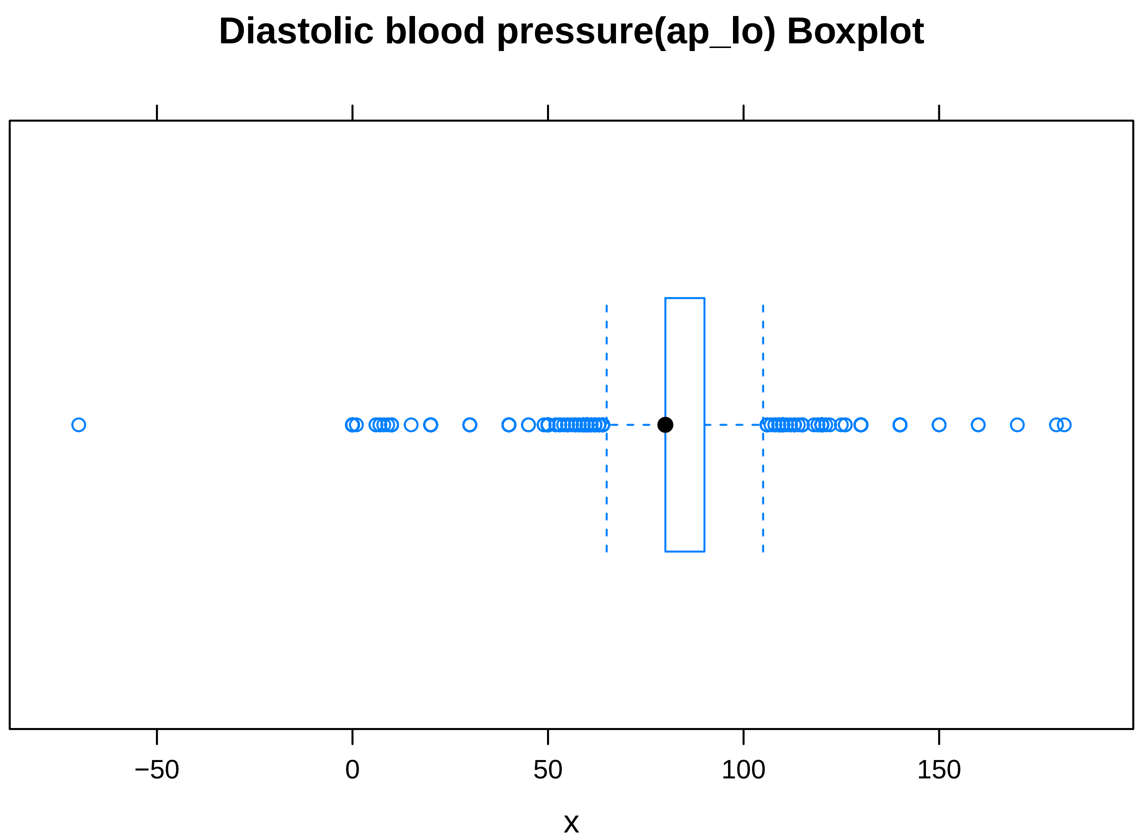
```
##Using boxplot graph we can find the outtlier points and we clean the dataset by removing outliers.
bwplot(~ap_cleaned$height,xlab="x",main="Height Boxplot")
```



```
height_cleaned <- ap_cleaned %>% filter(ap_cleaned$height >= 140 & ap_cleaned$height <= 200)

bwplot(~height_cleaned$weight,xlab="x",main="Weight Boxplot")
```




```
weight_cleaned <- height_cleaned %>% filter(height_cleaned$weight >= 30)

bwplot(~weight_cleaned$ap_lo,xlab="x",main="Diastolic blood pressure(ap_lo) Boxplot")
```




```r
ap_cleaned2 <- weight_cleaned %>% filter(weight_cleaned$ap_lo >= 30 & weight_cleaned$ap_lo <= 140)


cleaned_cardio <- ap_cleaned2 %>% filter(ap_cleaned2$ap_hi >= 70 & ap_cleaned2$ap_hi < 240)

head(cleaned_cardio)
```

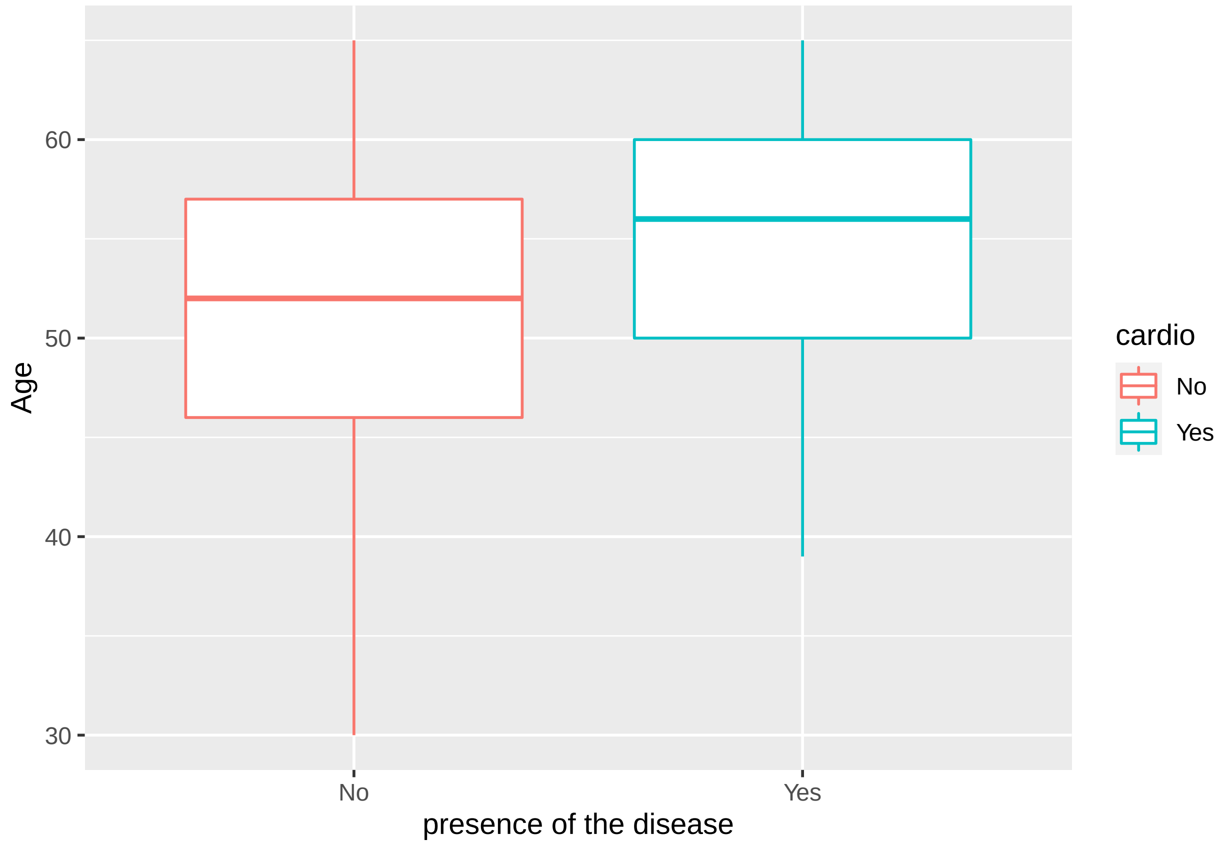
```
##     age gender height weight ap_hi ap_lo cholesterol gluc smoke alco active
## 1 18393      2    168     62   110    80           1    1     0    0      1
## 2 20228      1    156     85   140    90           3    1     0    0      1
## 3 18857      1    165     64   130    70           3    1     0    0      0
## 4 17623      2    169     82   150   100           1    1     0    0      1
## 5 17474      1    156     56   100    60           1    1     0    0      0
## 6 21914      1    151     67   120    80           2    2     0    0      0
##   cardio
## 1     No
## 2    Yes
## 3    Yes
## 4    Yes
## 5     No
## 6     No
```

```r
summary(cleaned_cardio)
```

```
##       age            gender          height          weight     
##  Min.   :10798   Min.   :1.000   Min.   :140.0   Min.   : 30.00  
##  1st Qu.:17658   1st Qu.:1.000   1st Qu.:159.0   1st Qu.: 65.00  
##  Median :19701   Median :1.000   Median :165.0   Median : 72.00  
##  Mean   :19465   Mean   :1.349   Mean   :164.5   Mean   : 74.12  
##  3rd Qu.:21324   3rd Qu.:2.000   3rd Qu.:170.0   3rd Qu.: 82.00  
##  Max.   :23713   Max.   :2.000   Max.   :198.0   Max.   :200.00  
##      ap_hi           ap_lo         cholesterol         gluc      
##  Min.   : 70.0   Min.   : 30.00   Min.   :1.000   Min.   :1.000  
##  1st Qu.:120.0   1st Qu.: 80.00   1st Qu.:1.000   1st Qu.:1.000  
##  Median :120.0   Median : 80.00   Median :1.000   Median :1.000  
##  Mean   :126.7   Mean   : 81.29   Mean   :1.365   Mean   :1.226  
##  3rd Qu.:140.0   3rd Qu.: 90.00   3rd Qu.:2.000   3rd Qu.:1.000  
##  Max.   :230.0   Max.   :140.00   Max.   :3.000   Max.   :3.000  
##      smoke              alco             active        cardio     
##  Min.   :0.00000   Min.   :0.00000   Min.   :0.0000   No :34616  
##  1st Qu.:0.00000   1st Qu.:0.00000   1st Qu.:1.0000   Yes:33892  
##  Median :0.00000   Median :0.00000   Median :1.0000              
##  Mean   :0.08802   Mean   :0.05337   Mean   :0.8035              
##  3rd Qu.:0.00000   3rd Qu.:0.00000   3rd Qu.:1.0000              
##  Max.   :1.00000   Max.   :1.00000   Max.   :1.0000              
```

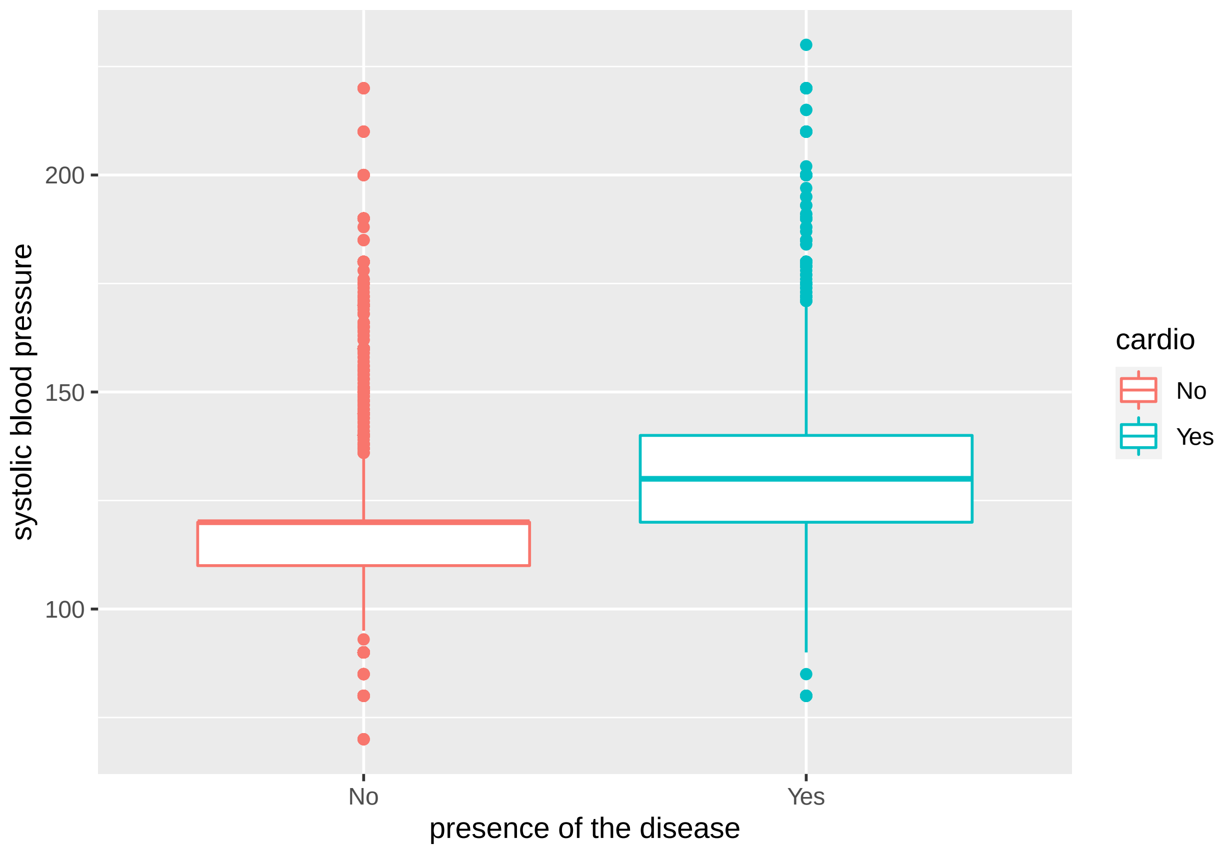
```r
##Converting age from days to years and will become easy to study.

cleaned_cardio$age <- round(cleaned_cardio$age/365)
```

```r
# Age vs Presence of the Disease

ggplot(data = cleaned_cardio,aes(x=cardio,y=age,col=cardio))+
  geom_boxplot()+
  xlab("presence of the disease")+
  ylab("Age")
```



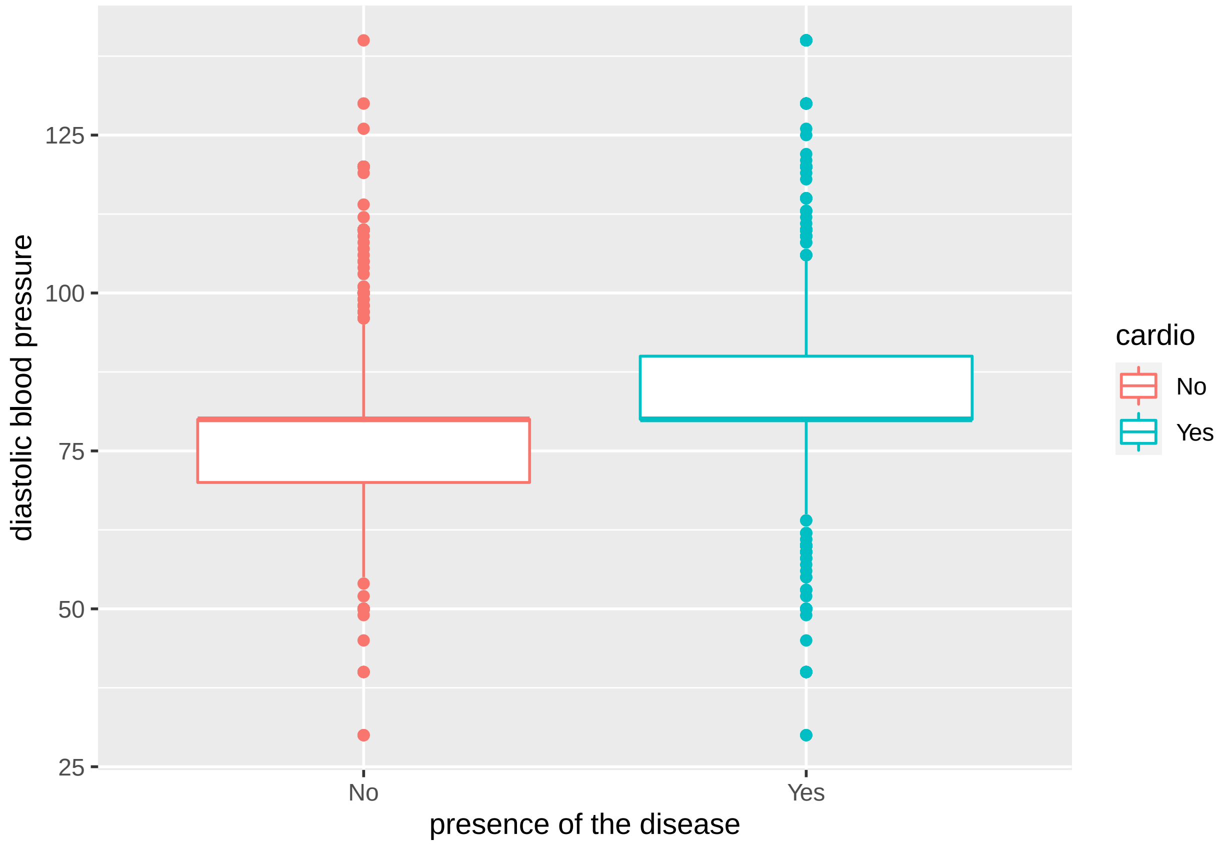
##Elder people tend to have Cardio disease more than younger people.

```
#Systolic Blood pressure vs Presence of the Disease

ggplot(data = cleaned_cardio,aes(x=cardio,y=ap_hi,col=cardio))+
  geom_boxplot()+
  xlab("presence of the disease")+
  ylab("systolic blood pressure")
```



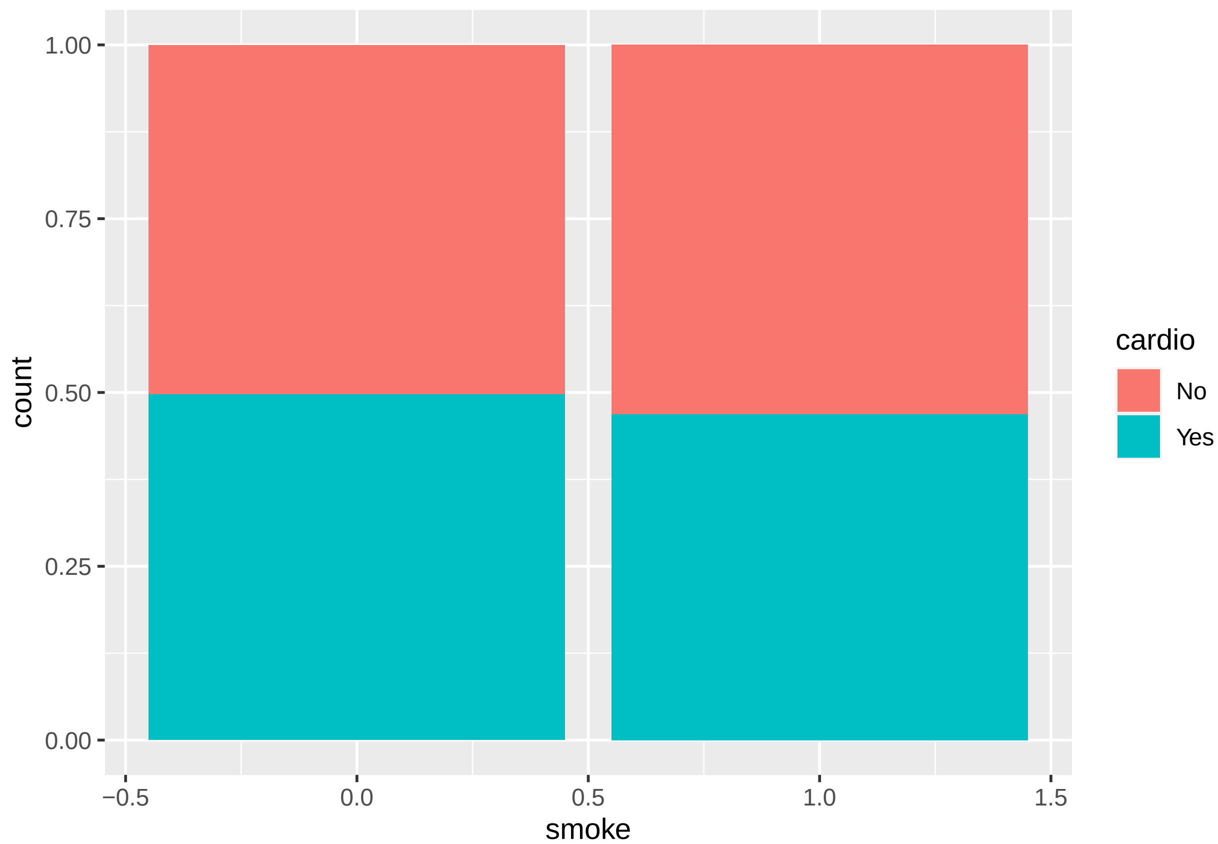
##Median Systolic blood pressure is higher for the people with Cardio Disease than for the people without Cardio Disease.

```
#Diastolic Blood pressure vs Presence of the Disease
ggplot(data = cleaned_cardio,aes(x=cardio,y=ap_lo,col=cardio))+
  geom_boxplot()+
  xlab("presence of the disease")+
  ylab("diastolic blood pressure")
```

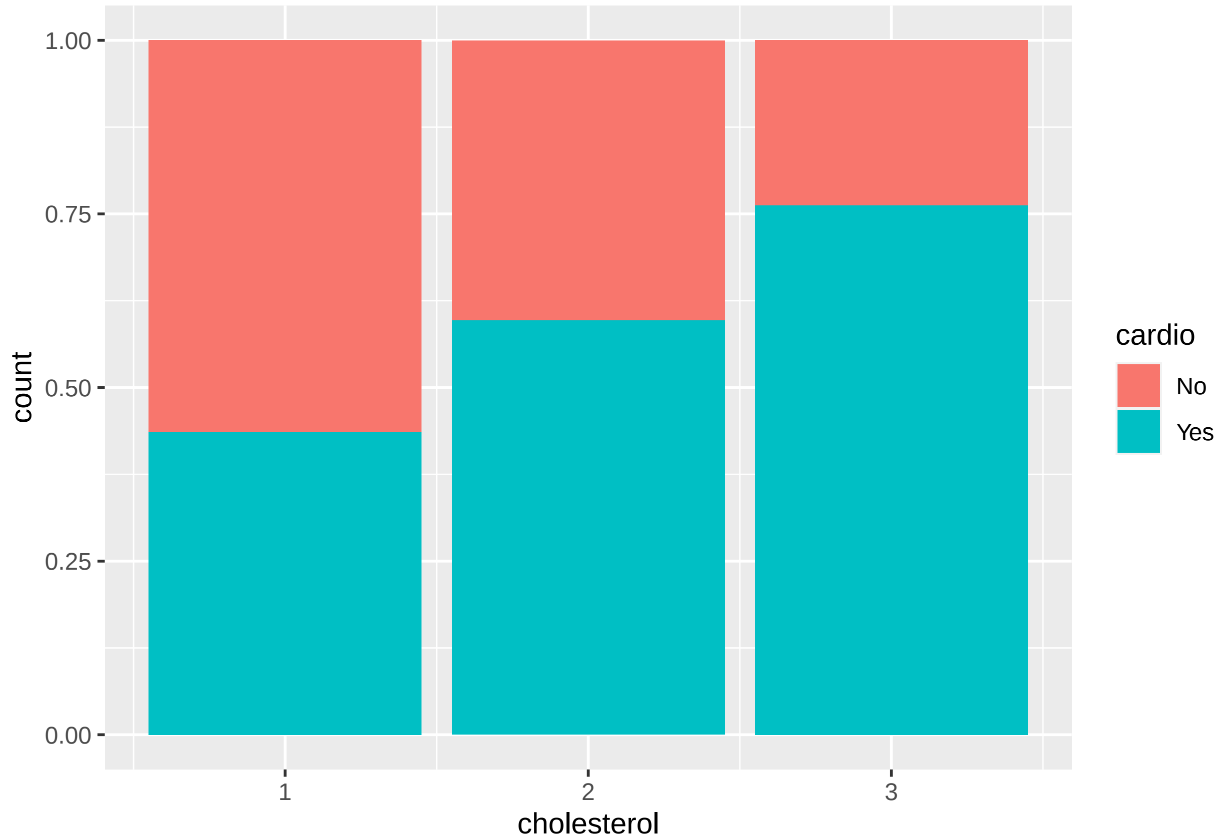


##Diastolic blood pressure is also high for the people with CVD.

```
#Smoke vs Presence of the Disease
ggplot(data = cleaned_cardio) +
  geom_bar(aes(x =smoke , fill = cardio), position = "fill")
```

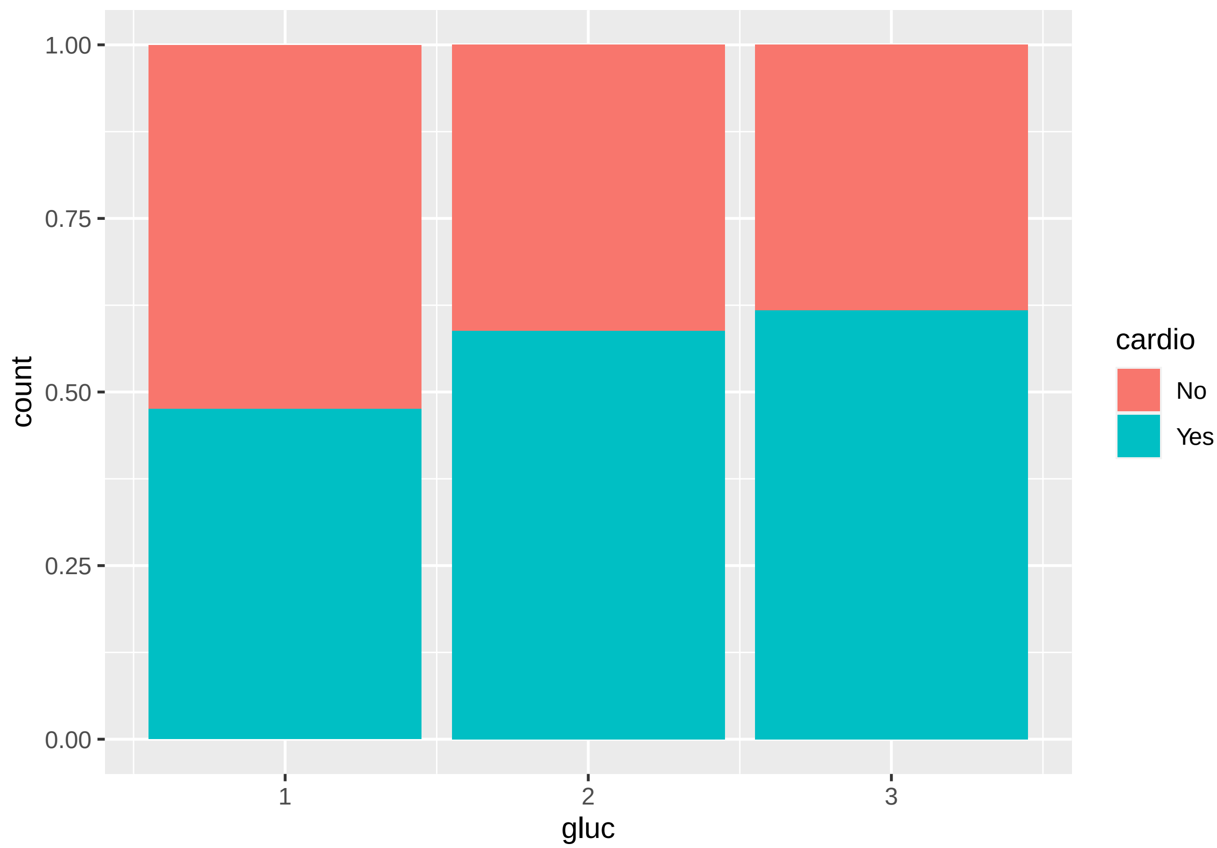



```r
#cholesterol vs Presence of the Disease
ggplot(data = cleaned_cardio) +
  geom_bar(aes(x =cholesterol , fill = cardio), position = "fill")
```



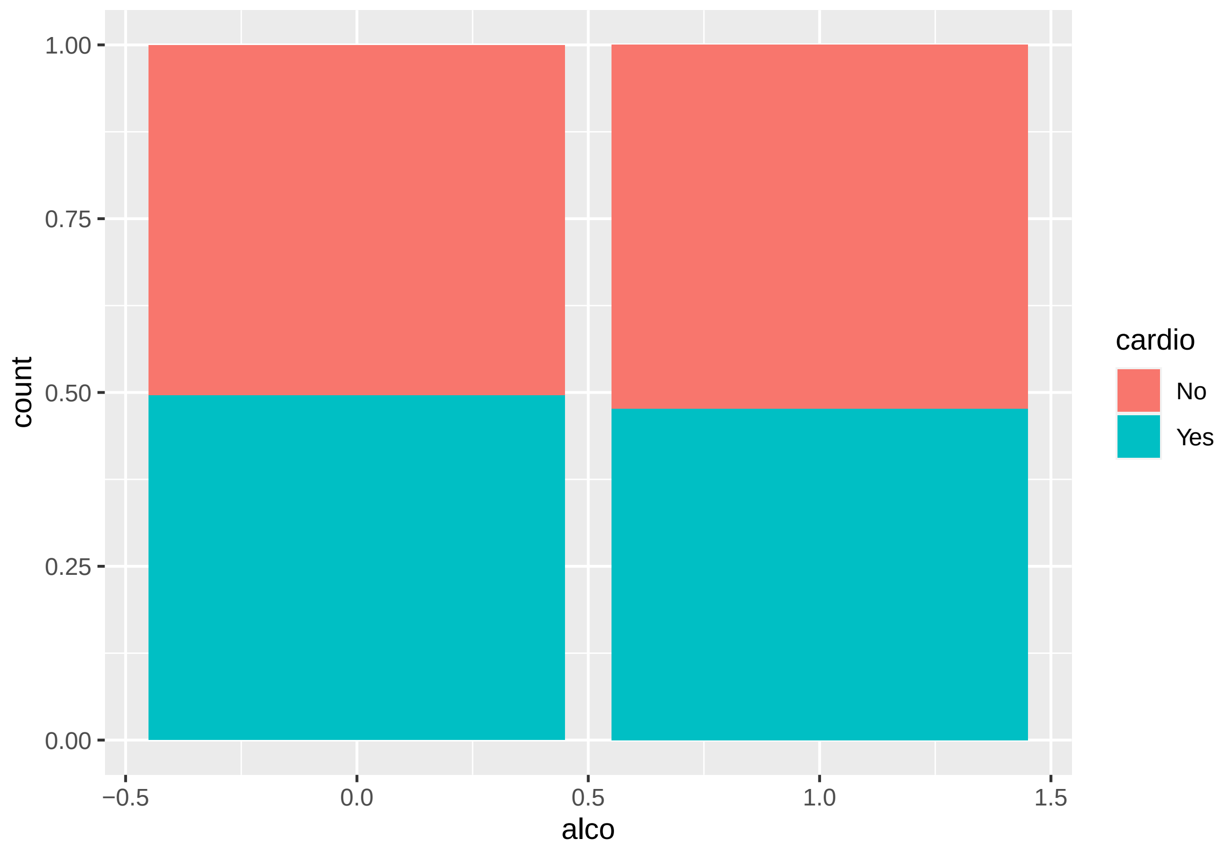
##cholesterol are high then cardio disease chances are also high

```
#glucose vs Presence of the Disease
ggplot(data = cleaned_cardio) +
  geom_bar(aes(x =gluc , fill = cardio), position = "fill")
```



##glucose are high then cardio disease chances are also high

```
ggplot(data = cleaned_cardio) +
  geom_bar(aes(x =alco , fill = cardio), position = "fill")
```



##For the feasibility and accuracy of analysis, we select 10000 records out of the cleaned dataset using simple random sampling and split the data into the training set, validation set and test set according to the 70:15:15 partition. The training set is to fit the model; the validation set is to fine-tune the model hyperparameters and combat overfitting; the test set is to evaluate the model performances based on some indicators, such as accuracy and precision.

```
data <- sample_n(cleaned_cardio, 10000)
idx <- sample(seq(1, 2), size = nrow(data), replace = TRUE, prob = c(.75, .25))
train <- data[idx == 1,]
test <- data[idx == 2,]
```

##Since the dataset varies at each time of sampling, for simplicity, I will continue the analysis with my sampling set of train, validation and test data.

```
cols = c("gender", "cholesterol", "gluc", "smoke", "alco", "active", "cardio")
train[cols] = lapply(train[cols], factor)
test[cols] = lapply(test[cols], factor)
```

##Summary Statistics

```
summary(train)
```

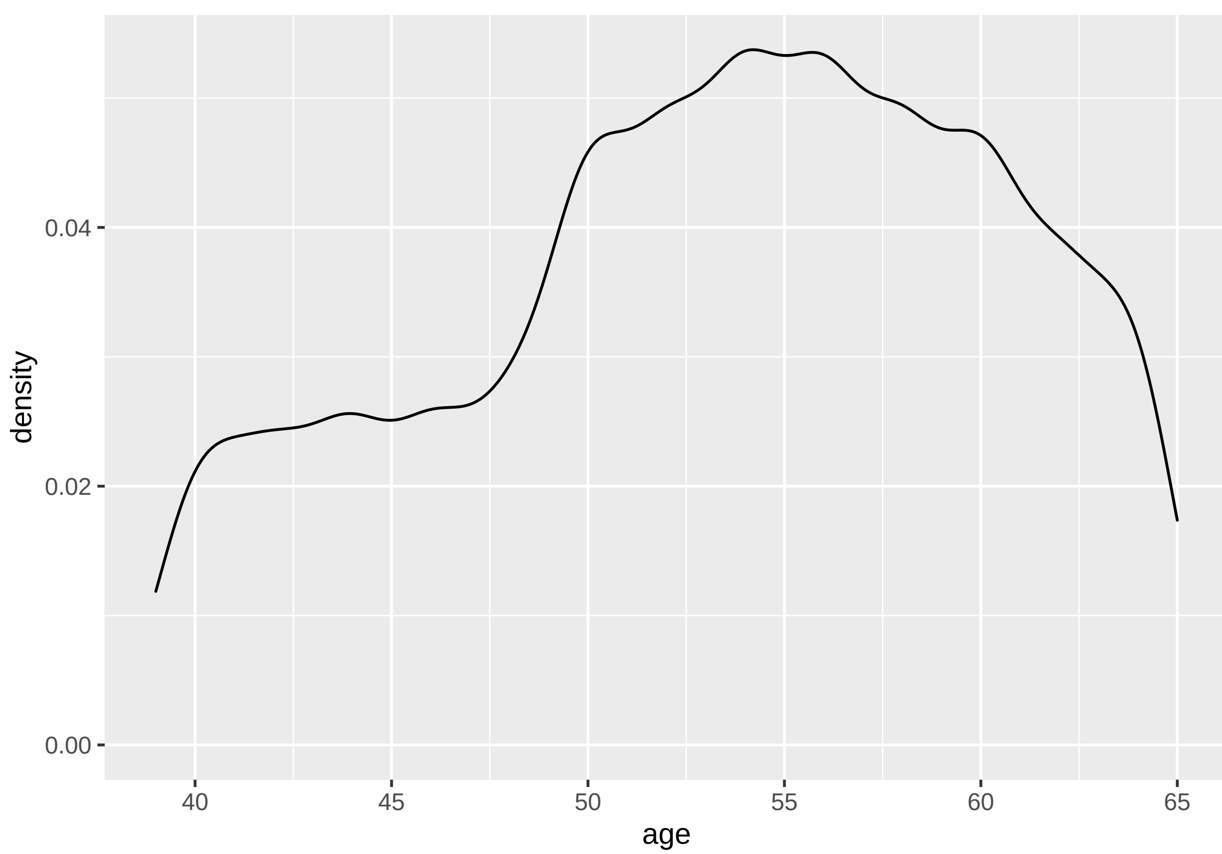
```
##       age          gender       height          weight           ap_hi
##  Min.   :39.00   1:4843   Min.   :140.0   Min.   : 34.00   Min.   : 70.0
##  1st Qu.:49.00   2:2649   1st Qu.:159.0   1st Qu.: 65.00   1st Qu.:120.0
```

```
##  Median :54.00           Median :165.0   Median : 72.00   Median :120.0  
##  Mean   :53.42           Mean   :164.5   Mean   : 73.93   Mean   :126.9  
##  3rd Qu.:59.00           3rd Qu.:170.0   3rd Qu.: 81.00   3rd Qu.:140.0  
##  Max.   :65.00           Max.   :198.0   Max.   :168.00   Max.   :220.0  
##      ap_lo        cholesterol gluc     smoke    alco     active   cardio  
##  Min.   : 40.00   1:5611      1:6361   0:6776   0:7063   0:1484   No :3741  
##  1st Qu.: 80.00   2:1033      2: 585   1: 716   1: 429   1:6008   Yes:3751  
##  Median : 80.00   3: 848      3: 546                                       
##  Mean   : 81.27                                                            
##  3rd Qu.: 90.00                                                            
##  Max.   :140.00                                                            
```

##4959 observations are females (gender = 1) and 2526 are males (gender = 2). There is a significant difference in the number of observations between gender. Thus, the data may be slightly biased due to the unequal distribution of gender.

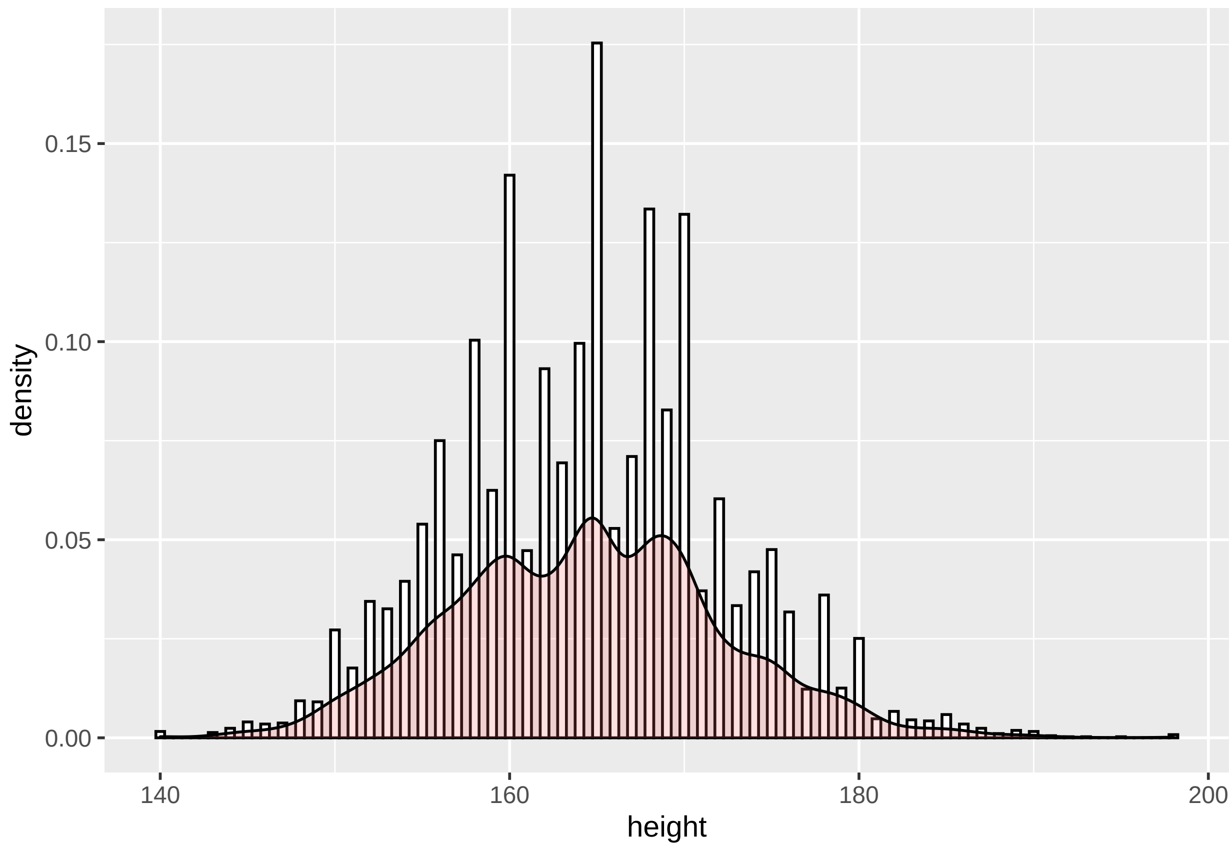
##3821 observations are not having cardiovascular diseases and the other 3664 observations suffer from cardiovascular diseases. The approximate ratio is 1:1 which suggests that the dependent variable "cardio" is distributed evenly, and accuracy is thus a reliable method to evaluate how a model performs.

```r
##Histogram and density plots are used to identify the distribution of continuous variables.
ggplot(train, aes(x=age)) + geom_density()
```

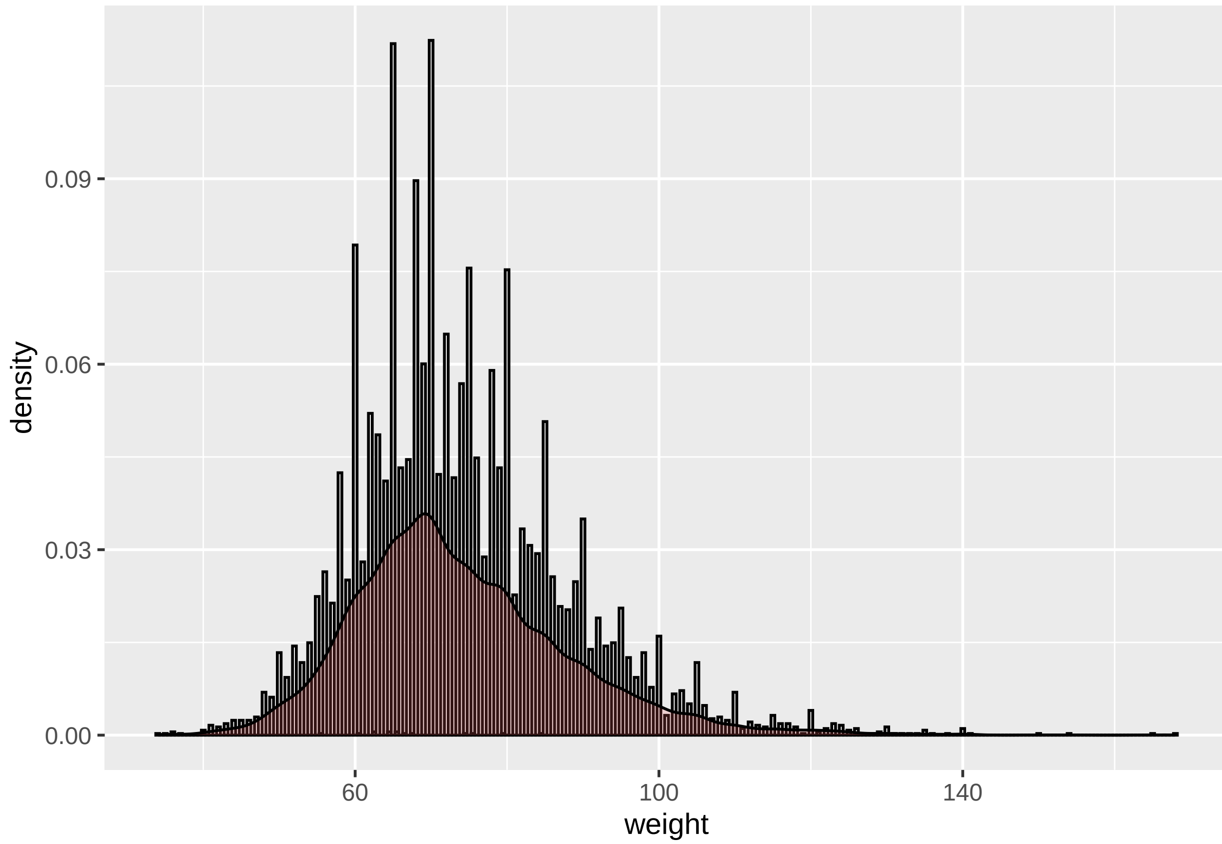



```r
ggplot(train, aes(x=height)) +
  geom_histogram(aes(y=..density..),
                 binwidth=.5,
```

```r
                 colour="black", fill="white") +
  geom_density(alpha=.2, fill="#FF6666")
```




```r
ggplot(train, aes(x=weight)) +
  geom_histogram(aes(y=..density..),
                 binwidth=.5,
                 colour="black", fill="white") +
  geom_density(alpha=.2, fill="#FF6666")
```

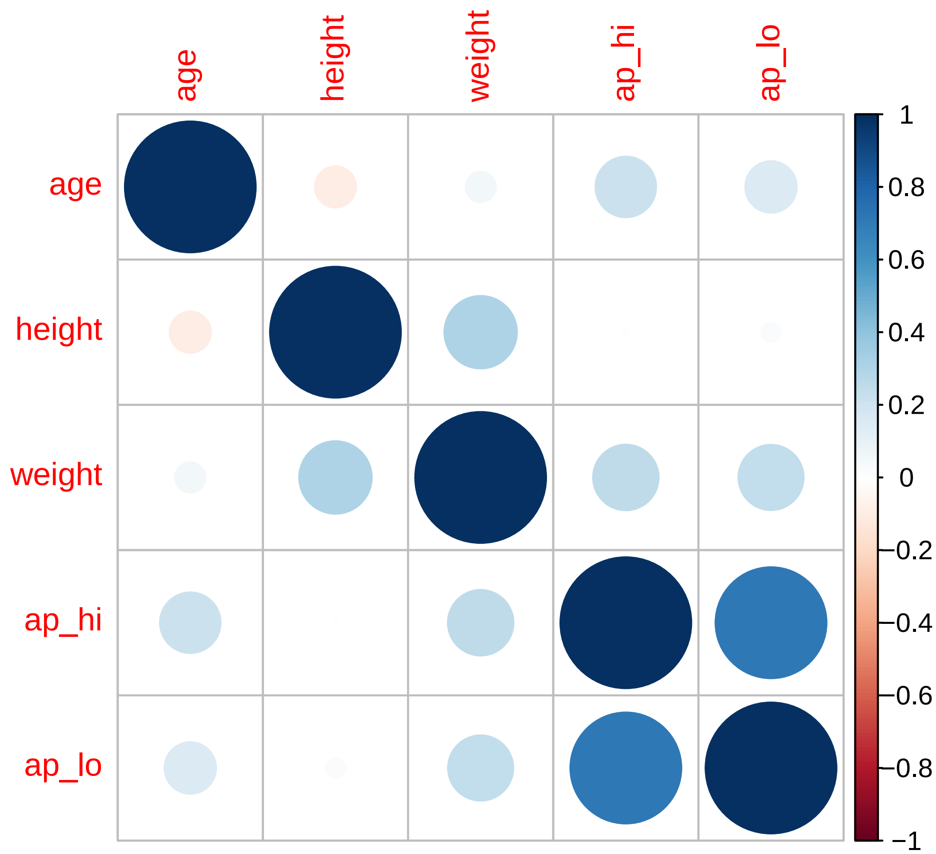


##From the above three plots, height, weight and age are not normally distributed and they are all somewhat skewed.

```
##Scatter plots and correlation plots can visualise and roughly identify the possible correlation betwe

train.corr <- cor(train[, c(1, 3, 4, 5, 6)])
train.corr
```

```
##                age       height     weight        ap_hi      ap_lo
## age     1.00000000 -0.098909120 0.05390128  0.214414526 0.15481044
## height -0.09890912  1.000000000 0.30688688 -0.002997929 0.02129369
## weight  0.05390128  0.306886879 1.00000000  0.250758734 0.24787897
## ap_hi   0.21441453 -0.002997929 0.25075873  1.000000000 0.71870328
## ap_lo   0.15481044  0.021293689 0.24787897  0.718703280 1.00000000
```

```
corrplot(train.corr, method = "circle")
```

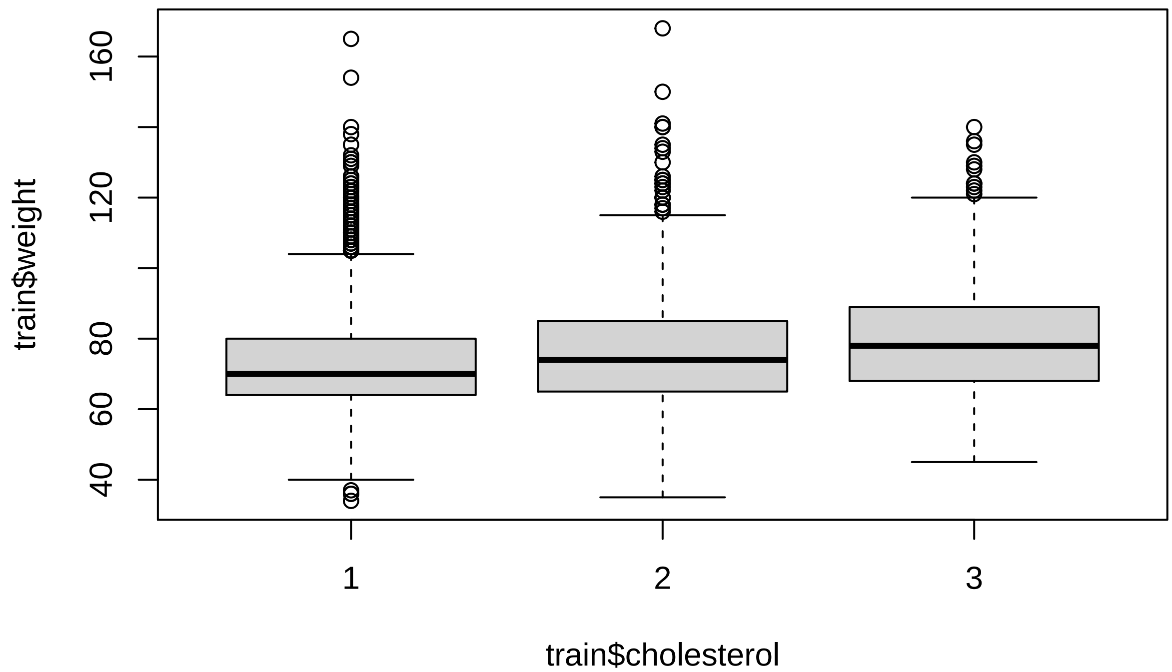


##From the above graphs, there are correlations between two pairs of continuous variables: ap_hi and ap_low, height and weight. The rest have very low correlation coefficient.

```
oneway.test(train$weight~train$cholesterol, var.equal = TRUE)
```

```
##
##  One-way analysis of means
##
## data:  train$weight and train$cholesterol
## F = 91.705, num df = 2, denom df = 7489, p-value < 2.2e-16
```

```
boxplot(train$weight~train$cholesterol)
```



##The null hypothesis is that the mean weight for people having different cholesterol levels is the same. Since the p-value is < 2.2e-16, which is smaller than 0.05. We reject the null hypothesis and conclude that there is a difference in the mean weight among people with various cholesterol levels. Also, from the box plots, we can see that median, upper and lower quartile all increase as cholesterol level rises from 1 (normal) to 3 (well above normal). This implies that independent variables of cholesterol and weight are positively correlated.

```
##Hypothesis 2: Is there a correlation between independent variables height and weight?

corr.test(train$height, train$weight)
```

```
## Call:corr.test(x = train$height, y = train$weight)
## Correlation matrix
## [1] 0.31
## Sample Size
## [1] 7492
## Probability values  adjusted for multiple tests.
## [1] 0
##
##  To see confidence intervals of the correlations, print with the short=FALSE option
```

##The correlation coefficient between height and weight is 0.31. Hence, the two variables are moderately correlated.

```r
##Hypothesis 3: Are systolic blood pressure (ap_high) and diastolic blood pressure (ap_low) correlated?
corr.test(train$ap_hi, train$ap_lo)
```

```
## Call:corr.test(x = train$ap_hi, y = train$ap_lo)
## Correlation matrix
## [1] 0.72
## Sample Size
## [1] 7492
## Probability values  adjusted for multiple tests.
## [1] 0
##
##  To see confidence intervals of the correlations, print with the short=FALSE option
```

##By applying the correlation test, the correlation coefficient between systolic blood pressure (ap_hi) and diastolic blood pressure (ap_lo) is 0.74, indicating a strong correlation between these two variables. Thus, interaction term ap_hi * ap_lo should be included in the model.

```r
##Hypothesis 4: Will gender affect someone's smoking habit?
chisq.test(train$gender, train$smoke, correct=FALSE)
```

```
##
##  Pearson's Chi-squared test
##
## data:  train$gender and train$smoke
## X-squared = 979.92, df = 1, p-value < 2.2e-16
```

##By conducting a Chi-Square test with a contingency table, the above result is obtained. The null hypothesis for the Chi-Square test is that the variables are independent of one another while the alternative hypothesis is that they are correlated in some way. The p-value is less than 2.2e-16, which is smaller than 0.05. There is sufficient evidence at 5% significance level to reject the null hypothesis and conclude that there is a correlation between gender and smoke.

```r
##Introduce a new term BMI to replace variables weight and height
train$BMI <- NA
train$BMI <- (train$weight/ ((train$height/100)^2))

test$BMI <- NA
test$BMI <- (test$weight/ ((test$height/100)^2))
```

##Include interaction terms

##Gender and smoke are correlated. Thus, an interaction term gender * smoke should be included. ##Systolic blood pressure and diastolic blood pressureare are strongly correlated. Hence, an interaction term ap_lo * ap_high should be included. ##Cholesterol and weight are correlated. ##Therefore, an interaction term cholesterol * BMI should be included.

```r
## Logistic Regression Model
lm1 <- glm(cardio~age + gender + height+weight+BMI + ap_hi + ap_lo + cholesterol + gluc + smoke + alco
summary(lm1)
```

```
##
```

```
## Call:
## glm(formula = cardio ~ age + gender + height + weight + BMI +
##     ap_hi + ap_lo + cholesterol + gluc + smoke + alco + active,
##     family = "binomial", data = train)
##
## Deviance Residuals:
##      Min        1Q    Median        3Q       Max
## -3.07934  -0.92574   0.09417   0.92655   2.48630
##
## Coefficients:
##               Estimate Std. Error z value Pr(>|z|)
## (Intercept)  -9.330732   3.166222  -2.947 0.003209 **
## age           0.054107   0.004118  13.140  < 2e-16 ***
## gender2      -0.084254   0.067974  -1.240 0.215157
## height       -0.014186   0.019292  -0.735 0.462153
## weight        0.025304   0.020947   1.208 0.227061
## BMI          -0.036382   0.055848  -0.651 0.514753
## ap_hi         0.052883   0.002701  19.580  < 2e-16 ***
## ap_lo         0.016274   0.004435   3.670 0.000243 ***
## cholesterol2  0.412690   0.084027   4.911 9.04e-07 ***
## cholesterol3  1.092877   0.110069   9.929  < 2e-16 ***
## gluc2         0.014552   0.109784   0.133 0.894545
## gluc3        -0.301435   0.123241  -2.446 0.014449 *
## smoke1       -0.016649   0.102477  -0.162 0.870940
## alco1        -0.284966   0.125425  -2.272 0.023086 *
## active1      -0.233093   0.066090  -3.527 0.000420 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 10386.1  on 7491  degrees of freedom
## Residual deviance:  8363.2  on 7477  degrees of freedom
## AIC: 8393.2
##
## Number of Fisher Scoring iterations: 4
```

```r
prob <- predict(lm1, test, type = "response")
test$pred <- NA
test$pred[prob >= 0.50] <- "Yes"
test$pred[prob < 0.50] <- "No"
table(test$pred, test$cardio)
```

```
##
##        No Yes
##   No  987 412
##   Yes 281 828
```

```r
Cardio_knn <- train(cardio ~ age + gender + BMI + ap_hi + ap_lo + cholesterol + gluc + smoke + alco + ac
                      data = train, method = "knn"

                      )
Cardio_knn
```

```
## k-Nearest Neighbors
##
## 7492 samples
##   10 predictor
##    2 classes: 'No', 'Yes'
##
## No pre-processing
## Resampling: Bootstrapped (25 reps)
## Summary of sample sizes: 7492, 7492, 7492, 7492, 7492, 7492, ...
## Resampling results across tuning parameters:
##
##   k  Accuracy   Kappa
##   5  0.6624892  0.3250238
##   7  0.6750947  0.3502556
##   9  0.6818372  0.3637548
##
## Accuracy was used to select the optimal model using the largest value.
## The final value used for the model was k = 9.
```

```r
pred <- predict(Cardio_knn, test)

confusionMatrix(pred, test$cardio, positive = "Yes")
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction  No Yes
##        No  922 428
##        Yes 346 812
##
##                Accuracy : 0.6914
##                  95% CI : (0.6729, 0.7094)
##     No Information Rate : 0.5056
##     P-Value [Acc > NIR] : < 2.2e-16
##
##                   Kappa : 0.3822
##
##  Mcnemar's Test P-Value : 0.003597
##
##             Sensitivity : 0.6548
##             Specificity : 0.7271
##          Pos Pred Value : 0.7012
##          Neg Pred Value : 0.6830
##              Prevalence : 0.4944
##          Detection Rate : 0.3238
##    Detection Prevalence : 0.4617
##       Balanced Accuracy : 0.6910
##
##        'Positive' Class : Yes
##
```

##The accuracy when handling unseen test data is 69.42%

##Naive Bayes classifies observations based on posterior probability, prior probability and the conditional probability of test data. Also, it assumes that the value of a feature in a given class is independent of the values of other features. The confusion matrix showing the performance of decision tree on test data is below.

```
NB <- naiveBayes(cardio ~ age + gender + BMI + ap_hi + ap_lo + cholesterol + gluc + smoke + alco + acti

train_predict <- predict(NB, test)

confusionMatrix(train_predict, test$cardio,positive = "Yes")
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction   No  Yes
##        No  1025  482
##        Yes  243  758
##
##                Accuracy : 0.7109
##                  95% CI : (0.6927, 0.7286)
##     No Information Rate : 0.5056
##     P-Value [Acc > NIR] : < 2.2e-16
##
##                   Kappa : 0.4205
##
##  Mcnemar's Test P-Value : < 2.2e-16
##
##             Sensitivity : 0.6113
##             Specificity : 0.8084
##          Pos Pred Value : 0.7572
##          Neg Pred Value : 0.6802
##              Prevalence : 0.4944
##          Detection Rate : 0.3022
##    Detection Prevalence : 0.3991
##       Balanced Accuracy : 0.7098
##
##        'Positive' Class : Yes
##
```

##The accuracy when handling unseen test data is 71.05%, which is Higher than the accuracy in the KNN model on the test data i.e. 69.42%.