Capstone Project - Coronary Heart Disease Study

Sanju Hyacinth C

29/12/2019

```
# PACKAGES REQUIRED:
# install.packages("Hmisc")
```

```
## Data loading

setwd("D:/R Progms/CAPSTONE")
getwd()
```

```
## [1] "D:/R Progms/CAPSTONE"
```

```
heart.d = read.csv("Coronary_heart_risk_study.csv")
#View(heart.d)

head(heart.d)
```

```
##   male age education currentSmoker cigsPerDay BPMeds prevalentStroke
## 1    1  39         4             0          0      0               0
## 2    0  46         2             0          0      0               0
## 3    1  48         1             1         20      0               0
## 4    0  61         3             1         30      0               0
## 5    0  46         3             1         23      0               0
## 6    0  43         2             0          0      0               0
##   prevalentHyp diabetes totChol sysBP diaBP   BMI heartRate glucose
## 1            0        0     195 106.0    70 26.97        80      77
## 2            0        0     250 121.0    81 28.73        95      76
## 3            0        0     245 127.5    80 25.34        75      70
## 4            1        0     225 150.0    95 28.58        65     103
## 5            0        0     285 130.0    84 23.10        85      85
## 6            1        0     228 180.0   110 30.30        77      99
##   TenYearCHD
## 1          0
## 2          0
## 3          0
## 4          1
## 5          0
## 6          0
```

```
## Data Structure and Summary

str(heart.d)
```

```
## 'data.frame':    4240 obs. of  16 variables:
##  $ male           : int  1 0 1 0 0 0 0 0 1 1 ...
##  $ age            : int  39 46 48 61 46 43 63 45 52 43 ...
##  $ education      : int  4 2 1 3 3 2 1 2 1 1 ...
##  $ currentSmoker  : int  0 0 1 1 1 0 0 1 0 1 ...
##  $ cigsPerDay     : int  0 0 20 30 23 0 0 20 0 30 ...
##  $ BPMeds         : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ prevalentStroke: int  0 0 0 0 0 0 0 0 0 0 ...
##  $ prevalentHyp   : int  0 0 0 1 0 1 0 0 1 1 ...
##  $ diabetes       : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ totChol        : int  195 250 245 225 285 228 205 313 260 225 ...
##  $ sysBP          : num  106 121 128 150 130 ...
##  $ diaBP          : num  70 81 80 95 84 110 71 71 89 107 ...
##  $ BMI            : num  27 28.7 25.3 28.6 23.1 ...
##  $ heartRate      : int  80 95 75 65 85 77 60 79 76 93 ...
##  $ glucose        : int  77 76 70 103 85 99 85 78 79 88 ...
##  $ TenYearCHD     : int  0 0 0 1 0 0 1 0 0 0 ...
```

```
# Our target variable is binary, hence to be converted to factor
# Our continuous numeric terms to be converted from chr to num
# And the nominal terms from num to factor

## Structure conversion:

heart.d$male = as.factor(heart.d$male)
heart.d$education = as.factor(heart.d$education)
heart.d$currentSmoker = as.factor(heart.d$currentSmoker)
heart.d$BPMeds = as.factor(heart.d$BPMeds)
heart.d$prevalentStroke = as.factor(heart.d$prevalentStroke)
heart.d$prevalentHyp = as.factor(heart.d$prevalentHyp)
heart.d$diabetes = as.factor(heart.d$diabetes)
heart.d$TenYearCHD = as.factor(heart.d$TenYearCHD)

## Summary:

summary(heart.d)
```

```
##  male          age        education   currentSmoker   cigsPerDay
##  0:2420   Min.   :32.00   1   :1720   0:2145        Min.   : 0.000
##  1:1820   1st Qu.:42.00   2   :1253   1:2095        1st Qu.: 0.000
##           Median :49.00   3   : 689                 Median : 0.000
##           Mean   :49.58   4   : 473                 Mean   : 9.006
##           3rd Qu.:56.00   NA's: 105                 3rd Qu.:20.000
##           Max.   :70.00                             Max.   :70.000
##                                                     NA's   :29
##   BPMeds     prevalentStroke prevalentHyp diabetes    totChol
##  0   :4063   0:4215          0:2923       0:4131   Min.   :107.0
##  1   : 124   1:  25          1:1317       1: 109   1st Qu.:206.0
##  NA's:  53                                         Median :234.0
##                                                    Mean   :236.7
##                                                    3rd Qu.:263.0
##                                                    Max.   :696.0
##                                                    NA's   :50
##      sysBP           diaBP             BMI          heartRate
##  Min.   : 83.5   Min.   : 48.0   Min.   :15.54   Min.   : 44.00
##  1st Qu.:117.0   1st Qu.: 75.0   1st Qu.:23.07   1st Qu.: 68.00
##  Median :128.0   Median : 82.0   Median :25.40   Median : 75.00
##  Mean   :132.4   Mean   : 82.9   Mean   :25.80   Mean   : 75.88
##  3rd Qu.:144.0   3rd Qu.: 90.0   3rd Qu.:28.04   3rd Qu.: 83.00
##  Max.   :295.0   Max.   :142.5   Max.   :56.80   Max.   :143.00
##                                  NA's   :19      NA's   :1
##     glucose        TenYearCHD
##  Min.   : 40.00   0:3596
##  1st Qu.: 71.00   1: 644
##  Median : 78.00
##  Mean   : 81.96
##  3rd Qu.: 87.00
##  Max.   :394.00
##  NA's   :388
```

```r
# From observing the target variable, we find that close to 85% of the patients have been cleared of ha
# More than 50% of the population is women, non current smokers and not under blood pressure medication

## NA counts:

anyNA(heart.d)
```

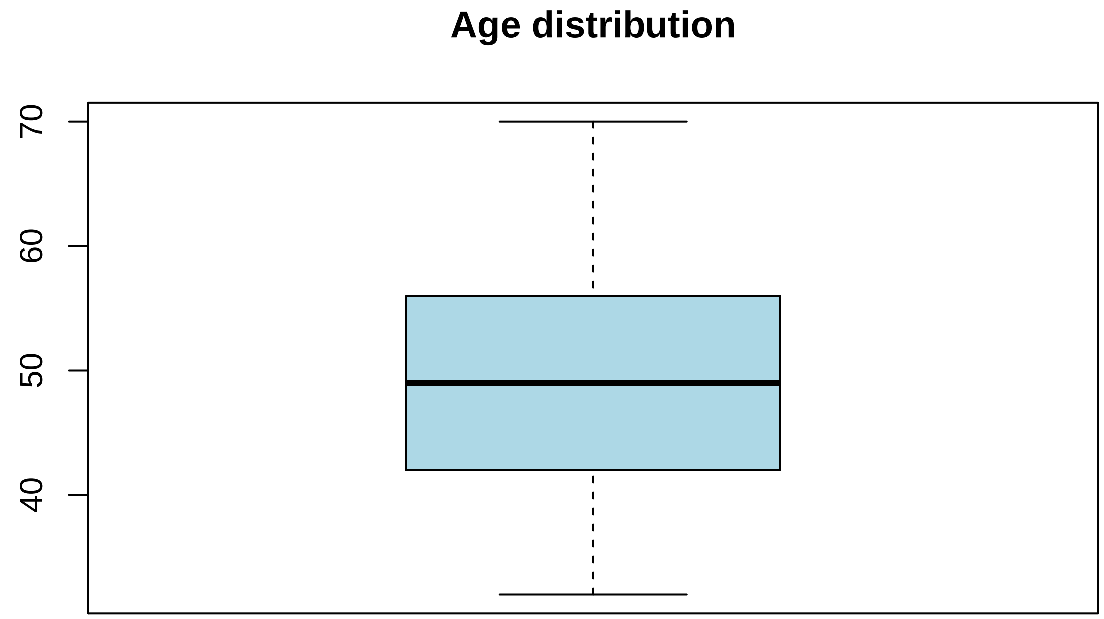
```
## [1] TRUE
```

```r
# A total of 7 fields have NA values with heartrate having the least and glucose having the most
# We will have to impute data for glucose values as it is very important
# We can remove the NA values in heartrate and education as it will not be a significant data loss
#
```

```r
# UNIVARIATE ANALYSIS:

# Age:
summary(heart.d$age)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   32.00   42.00   49.00   49.58   56.00   70.00
```

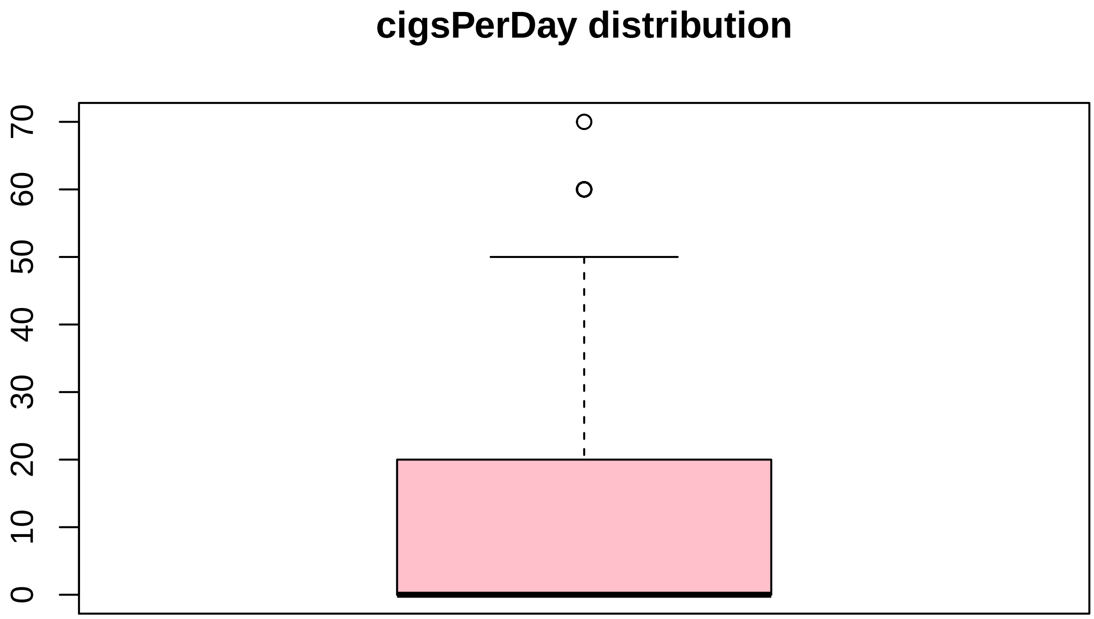
```r
boxplot(heart.d$age, data = heart.d, col = "lightblue", main = "Age distribution")
```



```
# No outliers. Well within the ranges

# Cigerattes per day
summary(heart.d$cigsPerDay)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's 
##   0.000   0.000   0.000   9.006  20.000  70.000      29
```

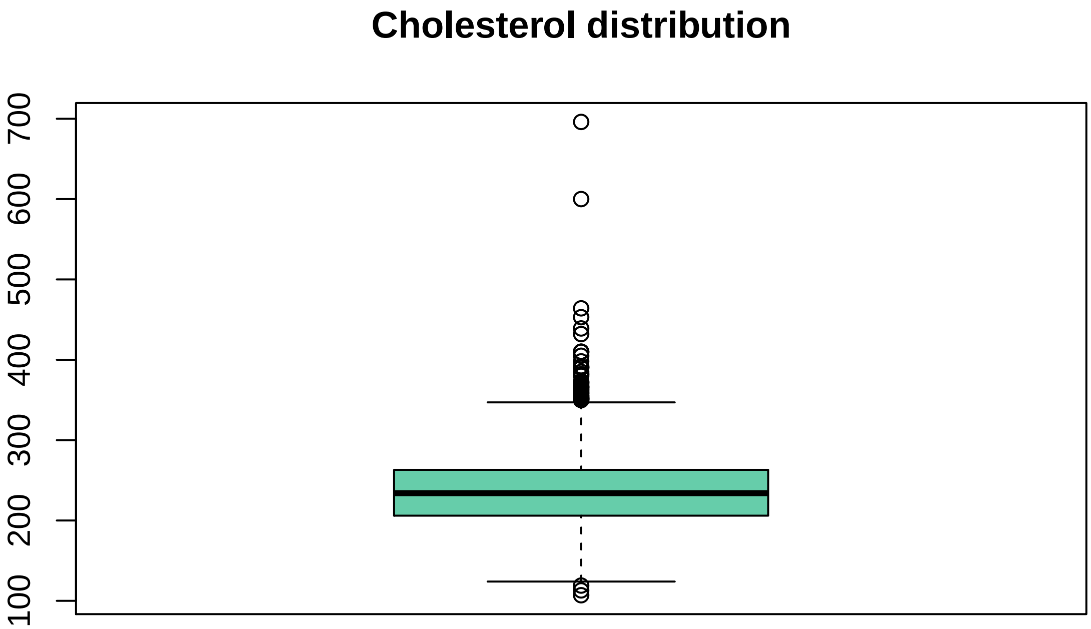
```
boxplot(heart.d$cigsPerDay, data = heart.d, col = "pink", main = "cigsPerDay distribution")
```



```
# 2 outliers present

# Total Cholesterol:
summary(heart.d$totChol)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's 
##   107.0   206.0   234.0   236.7   263.0   696.0      50
```

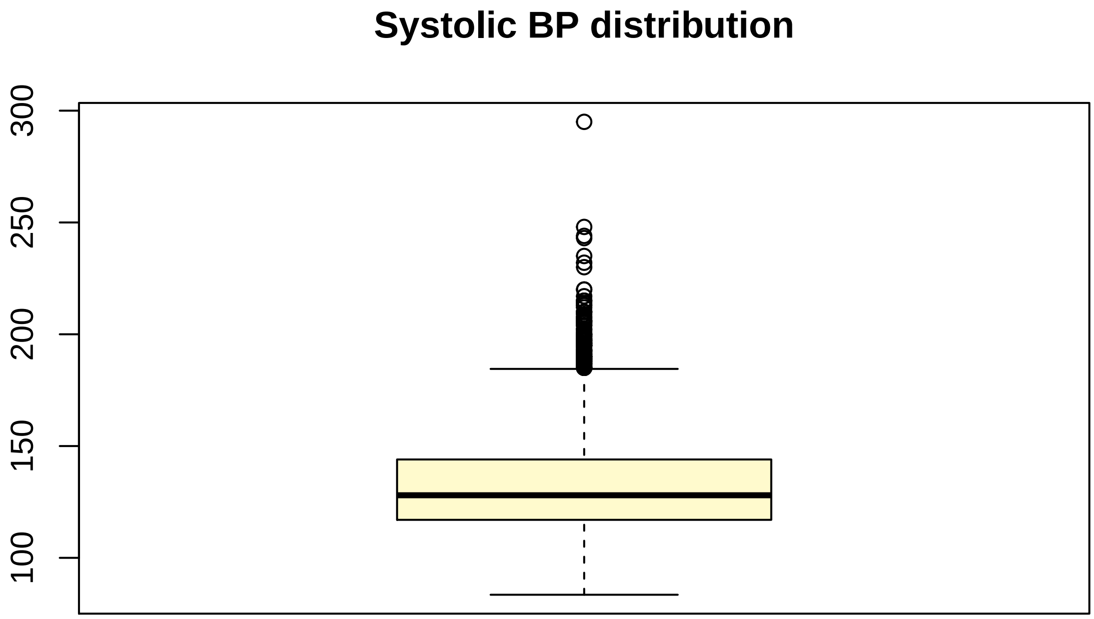
```
boxplot(heart.d$totChol, data = heart.d, col = "aquamarine3", main = "Cholesterol distribution")
```



```
# Systolic BP
summary(heart.d$sysBP)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##    83.5   117.0   128.0   132.4   144.0   295.0
```

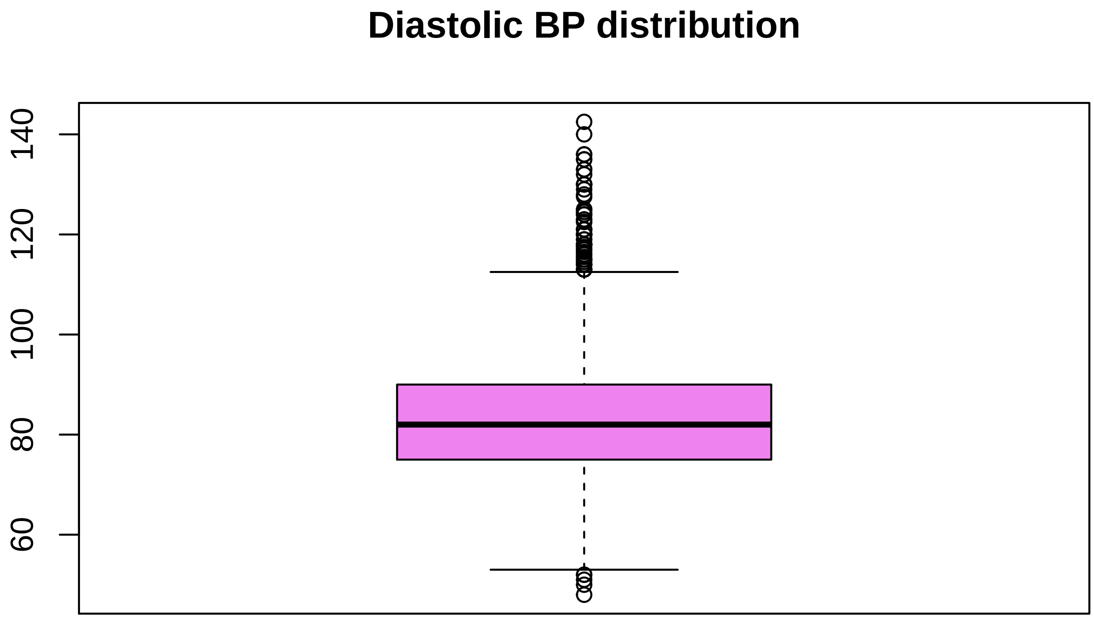
```
boxplot(heart.d$sysBP, data = heart.d, col = "lemonchiffon", main = "Systolic BP distribution")
```



```r
# Diastolic BP
summary(heart.d$diaBP)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max. 
##    48.0    75.0    82.0    82.9    90.0   142.5
```

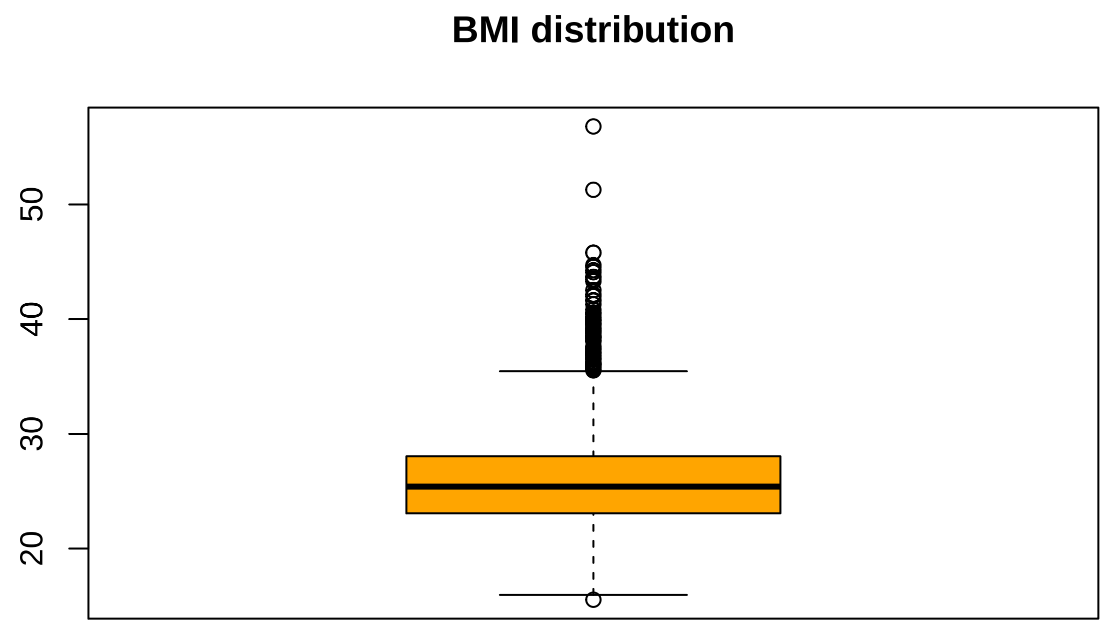
```r
boxplot(heart.d$diaBP, data = heart.d, col = "violet", main = "Diastolic BP distribution")
```



```r
# Body Mass Index:
summary(heart.d$BMI)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's 
##   15.54   23.07   25.40   25.80   28.04   56.80      19
```

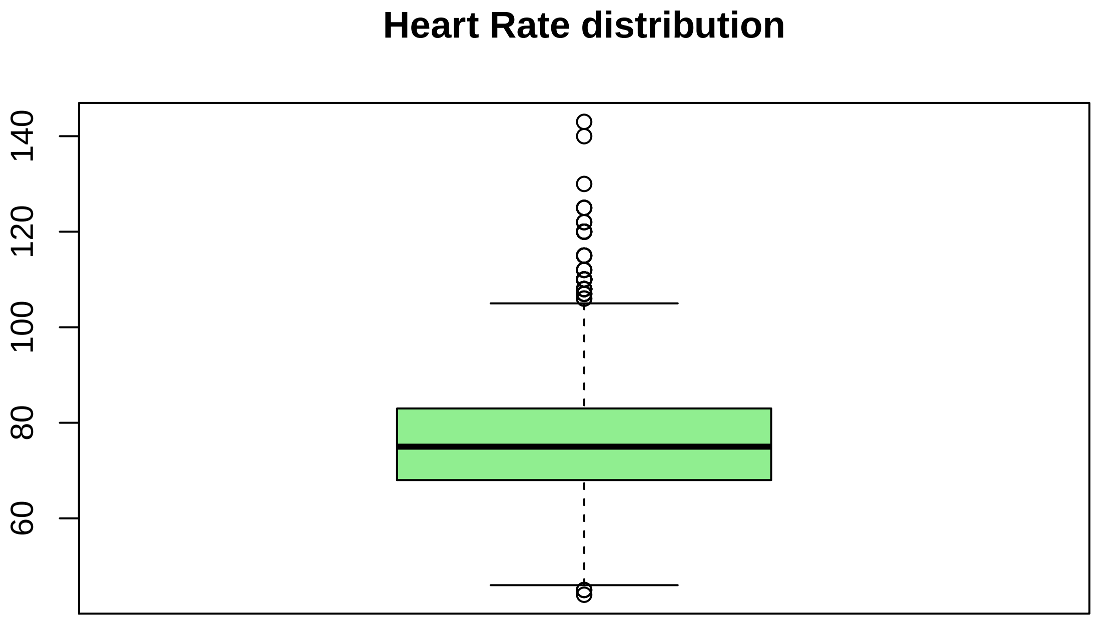
```r
boxplot(heart.d$BMI, data = heart.d, col = "orange", main = "BMI distribution")
```



```
# Heart Rate:
summary(heart.d$heartRate)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
##   44.00   68.00   75.00   75.88   83.00  143.00       1
```

```
boxplot(heart.d$heartRate, data = heart.d, col = "lightgreen", main = "Heart Rate distribution")
```



```
# Glucose
summary(heart.d$glucose)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's 
##   40.00   71.00   78.00   81.96   87.00  394.00     388
```

```
boxplot(heart.d$glucose, data = heart.d, col = "tomato", main = "Glucose distribution")

table(heart.d$male, heart.d$TenYearCHD)
```

```
##    
##        0    1
##   0 2119  301
##   1 1477  343
```

```
table(heart.d$male, heart.d$currentSmoker)
```

```
##    
##        0    1
##   0 1431  989
##   1  714 1106
```

```
# Women: 2420, Men: 1820
# Most women (1431 - 59%) are non-smokers while the others (989) are smokers.
# Most of the men (1106 - 60%) are smokers while the others (714) are non-smokers.

table(heart.d$male, heart.d$BPMeds)
```

```
## 
```

```
##         0    1
##   0 2293   89
##   1 1770   35
```

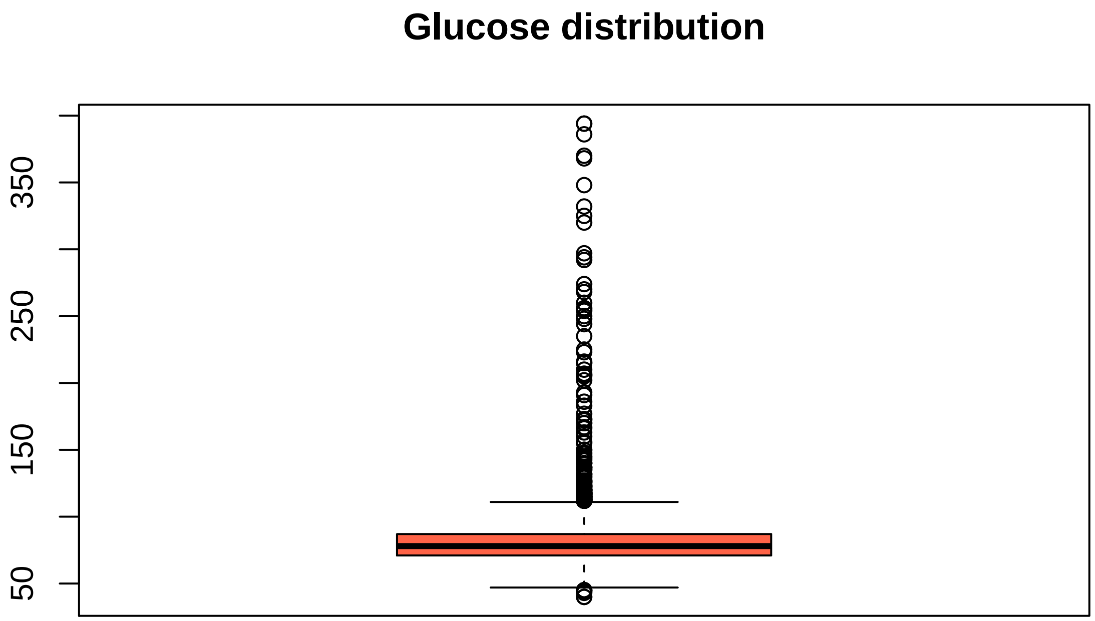
```
# Close to 95% of the women and 98% of the men do not take BP medication

table(heart.d$male, heart.d$prevalentStroke)
```

```
##
##         0    1
##   0 2405   15
##   1 1810   10
```

```
# Almost the entire patient population is free of having had prevalent Stroke

table(heart.d$male, heart.d$prevalentHyp)
```

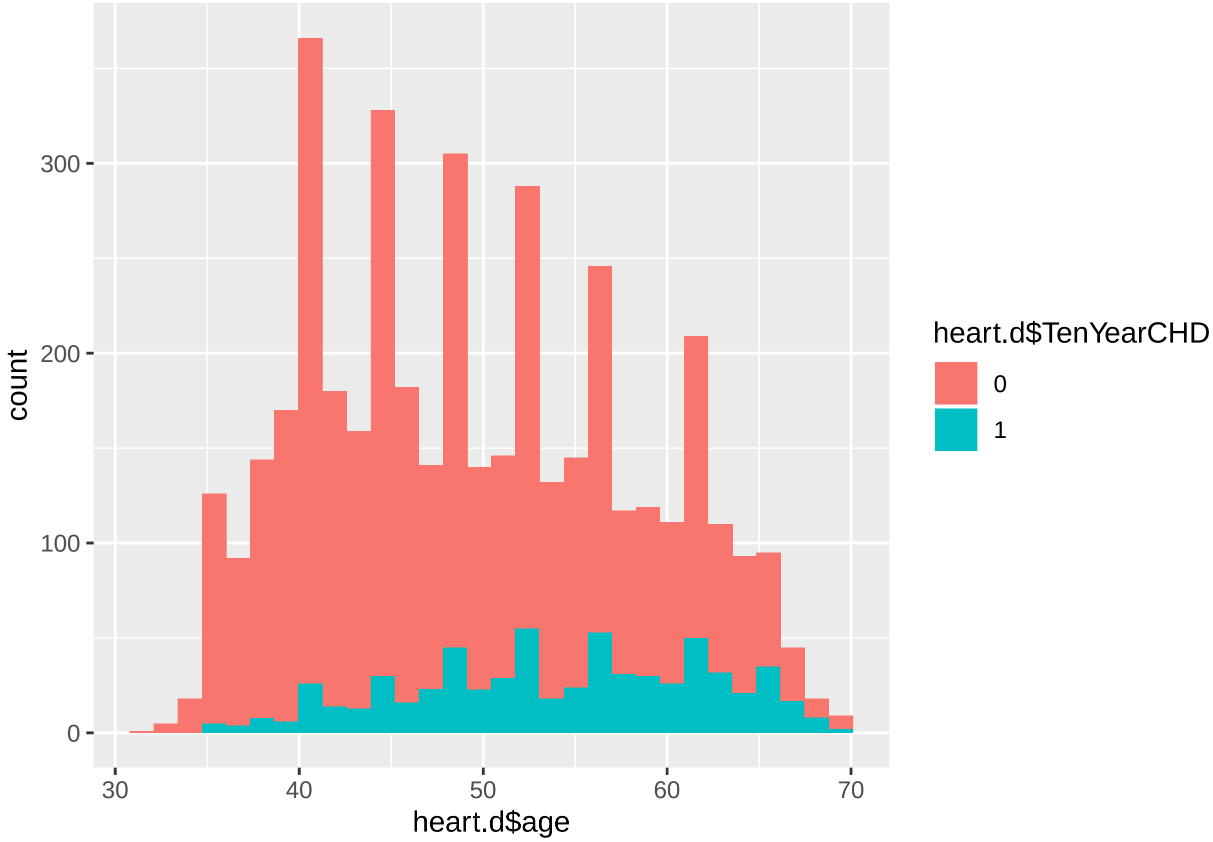
```
##
##         0    1
##   0 1674  746
##   1 1249  571
```

```
# Almost 69% of the female popultion and male population have not suffered prevalent Hypertension

## Histograms on ggplot2:
library(ggplot2)
```



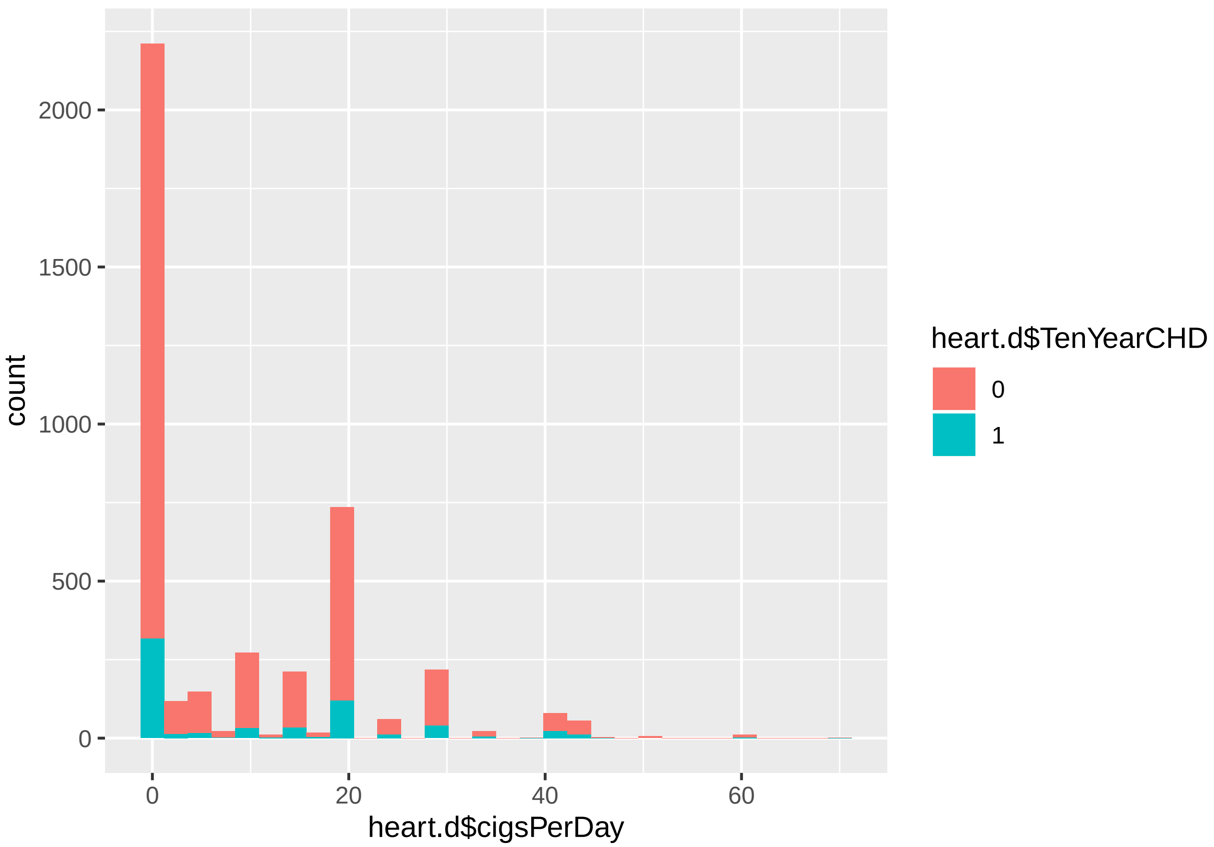
```
agep1 = ggplot(data = heart.d) + geom_histogram(mapping = aes(x = heart.d$age, fill = heart.d$TenYearCH

cigsp2 = ggplot(data = heart.d) + geom_histogram(mapping = aes(x = heart.d$cigsPerDay, fill = heart.d$T

sysp3 = ggplot(data = heart.d) + geom_histogram(mapping = aes(x = heart.d$sysBP, fill = heart.d$TenYear

diap4 = ggplot(data = heart.d) + geom_histogram(mapping = aes(x = heart.d$diaBP, fill = heart.d$TenYear

BMIp5 = ggplot(data = heart.d) + geom_histogram(mapping = aes(x = heart.d$BMI, fill = heart.d$TenYearCH

heartRp6 = ggplot(data = heart.d) + geom_histogram(mapping = aes(x = heart.d$heartRate, fill = heart.d$

glucp7 = ggplot(data = heart.d) + geom_histogram(mapping = aes(x = heart.d$glucose, fill = heart.d$TenY

totchop8 = ggplot(data = heart.d) + geom_histogram(mapping = aes(x = heart.d$totChol, fill = heart.d$Te

# View charts
agep1
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



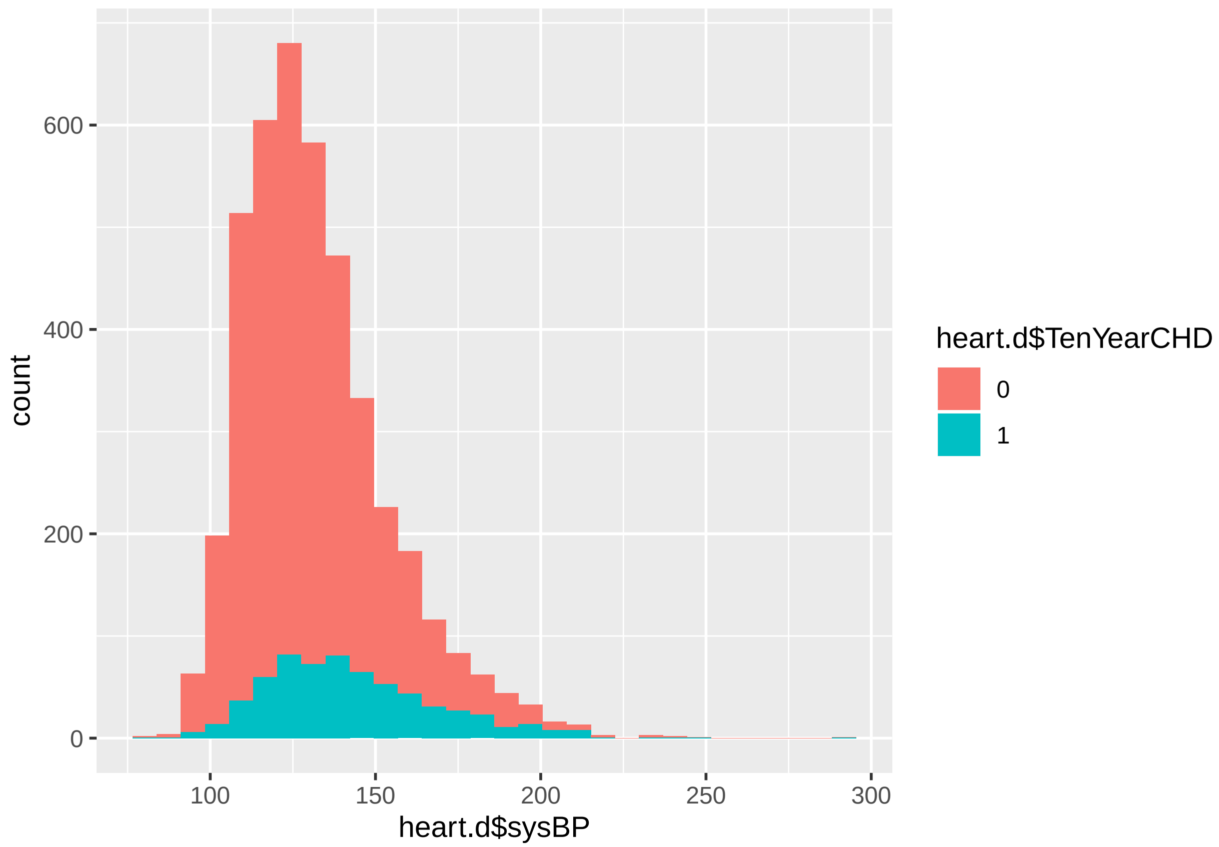
```
cigsp2
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.

## Warning: Removed 29 rows containing non-finite values (stat_bin).
```



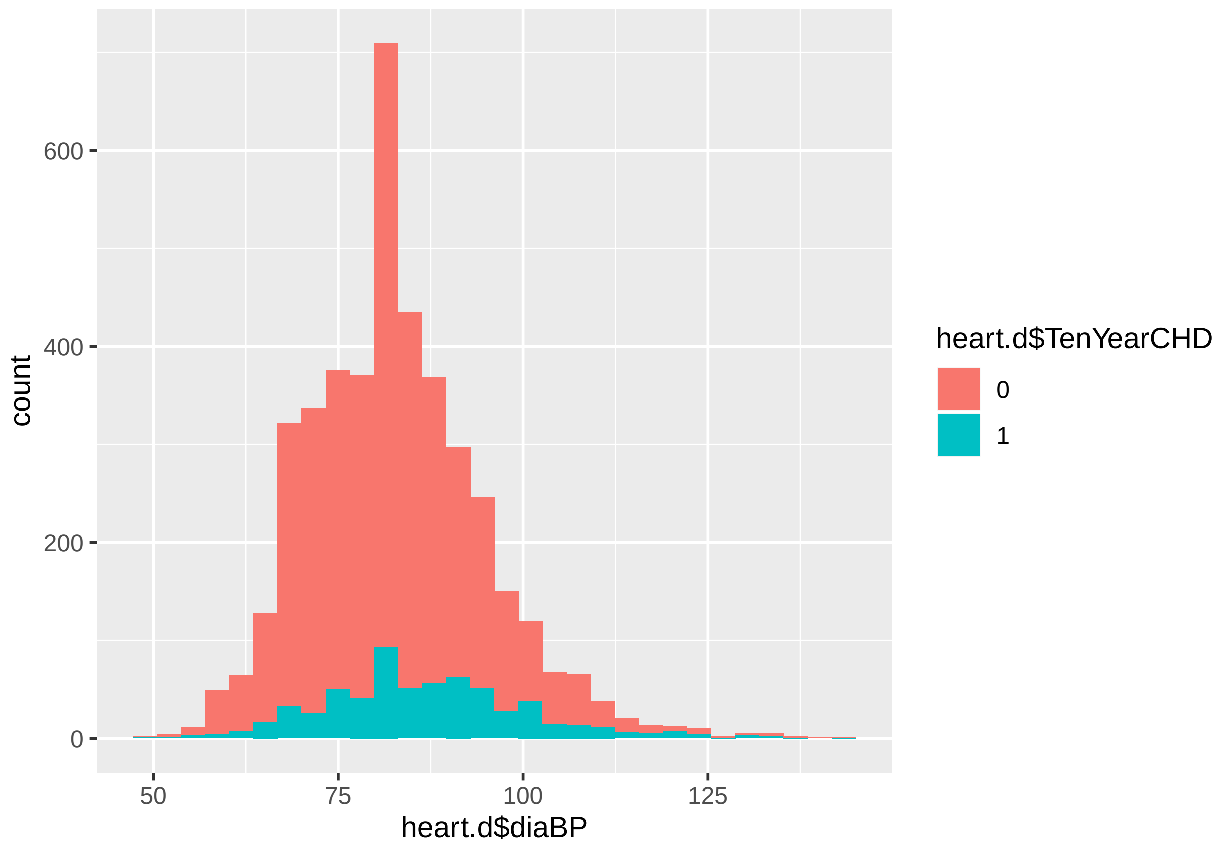
```
sysp3
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



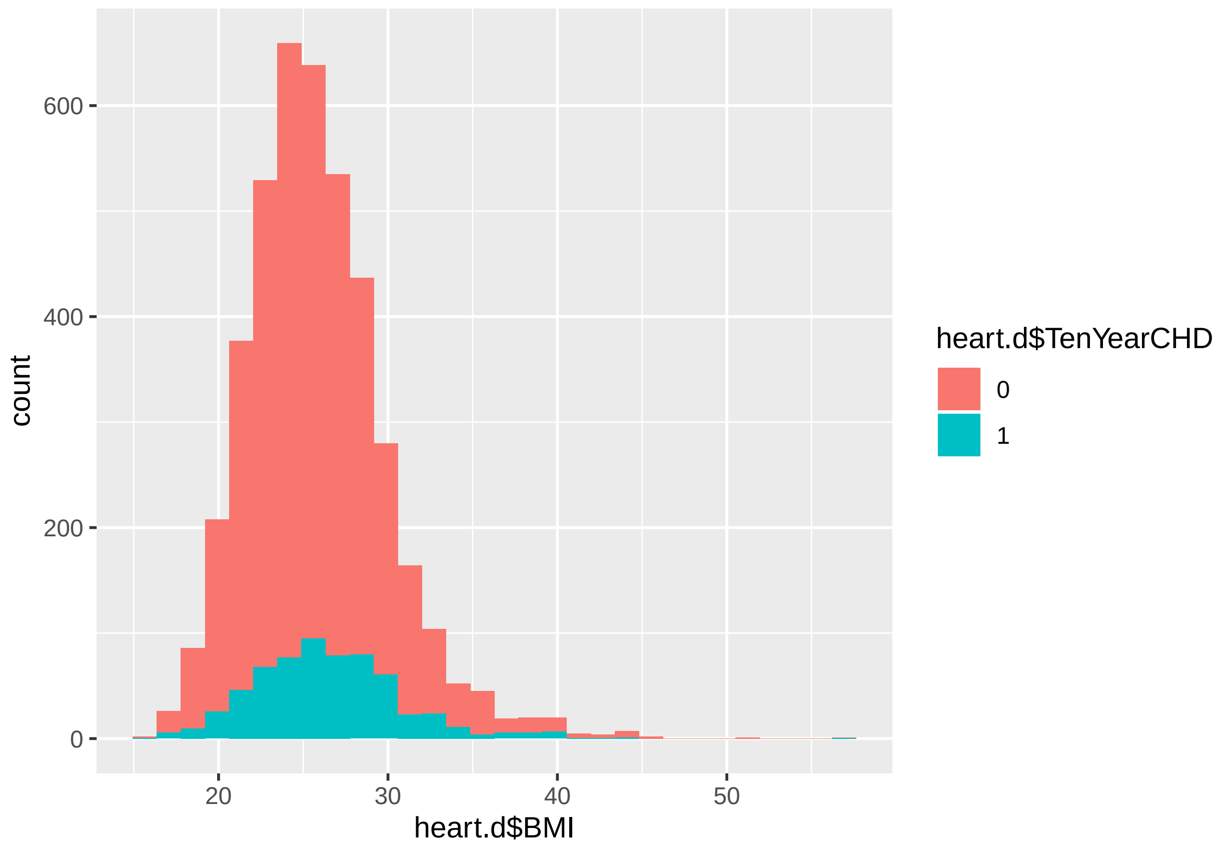
```
diap4
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



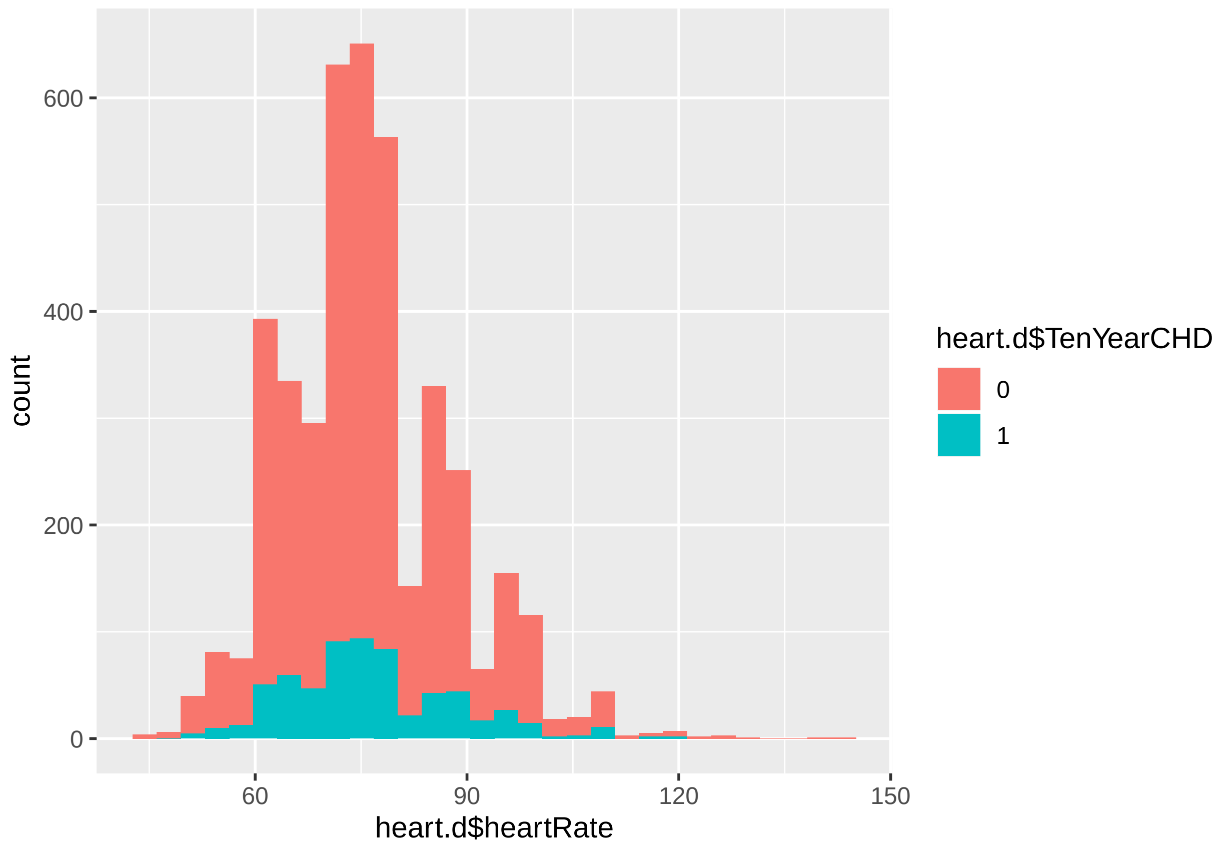
```
BMIp5
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.

## Warning: Removed 19 rows containing non-finite values (stat_bin).
```



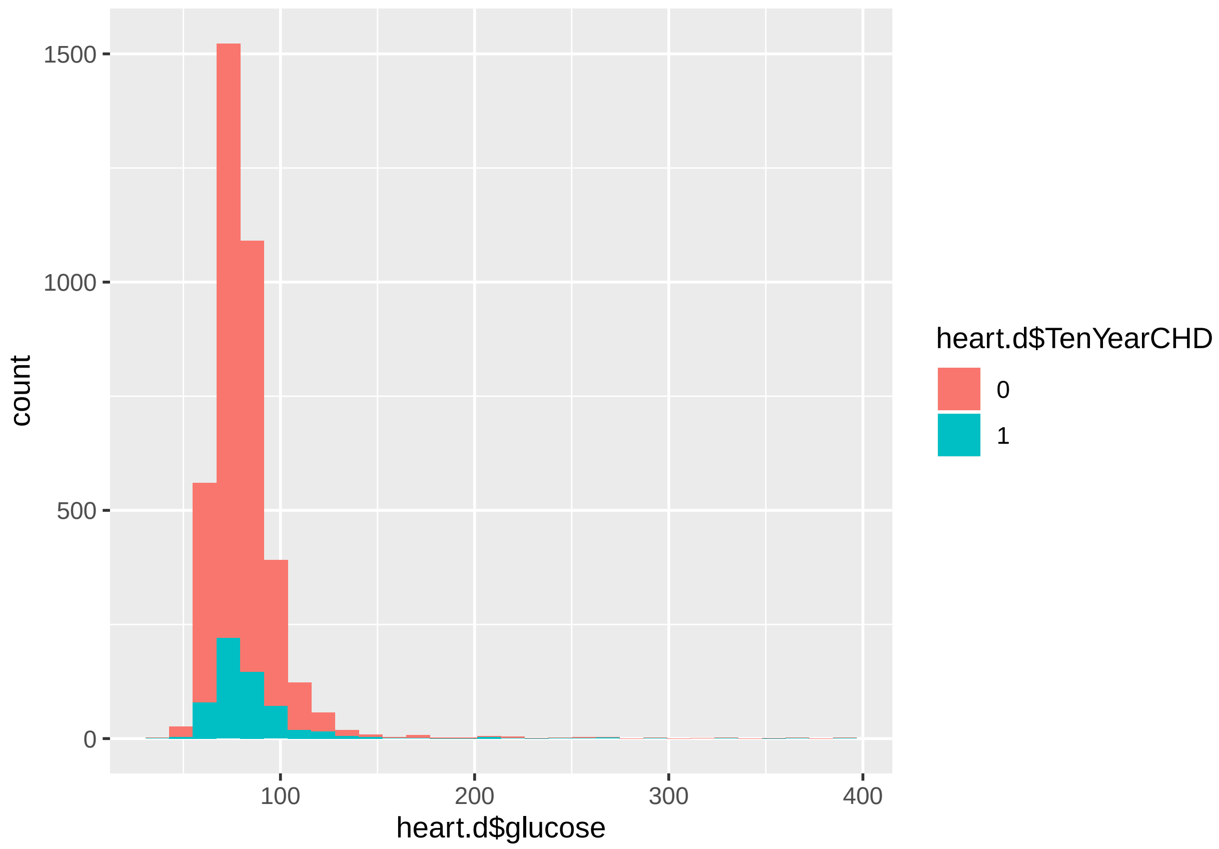
```
heartRp6
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.

## Warning: Removed 1 rows containing non-finite values (stat_bin).
```



```
glucp7
```

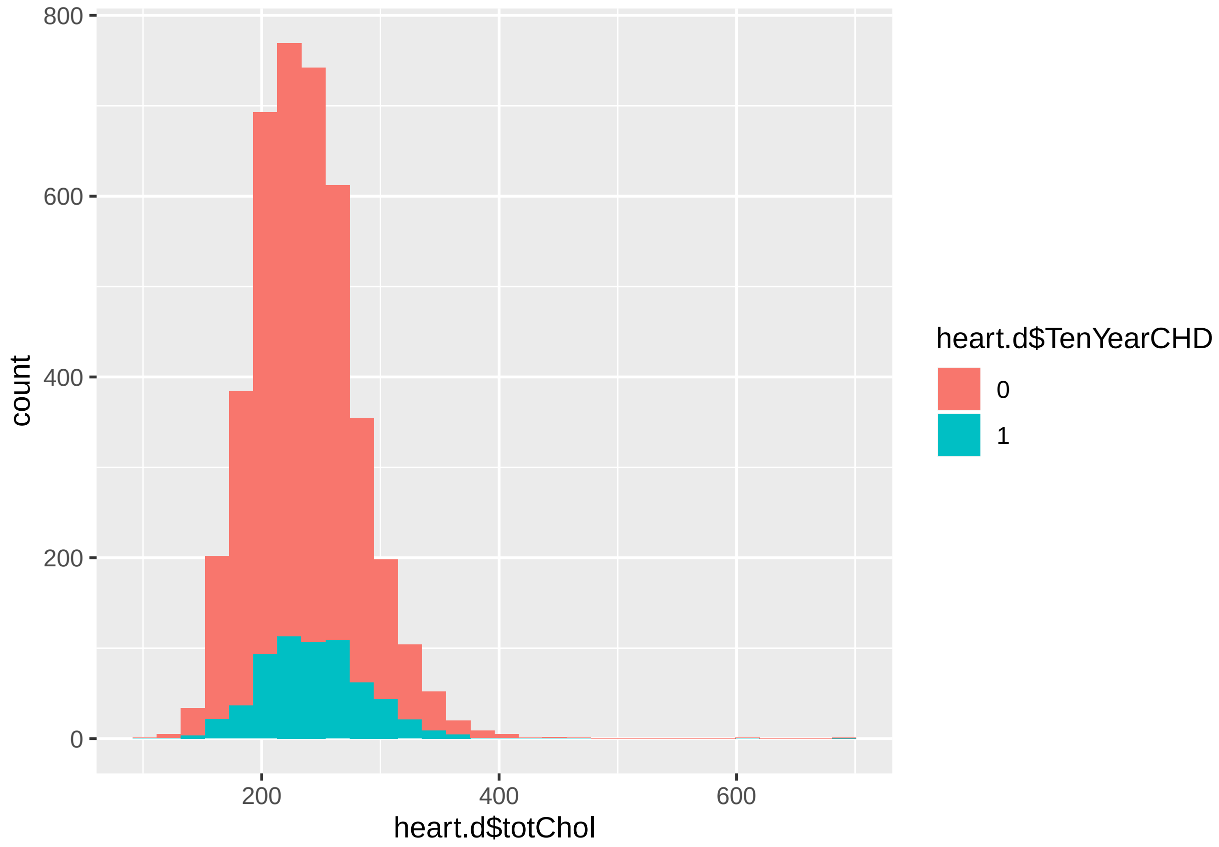
```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.

## Warning: Removed 388 rows containing non-finite values (stat_bin).
```



```
totchop8
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```

```
## Warning: Removed 50 rows containing non-finite values (stat_bin).
```

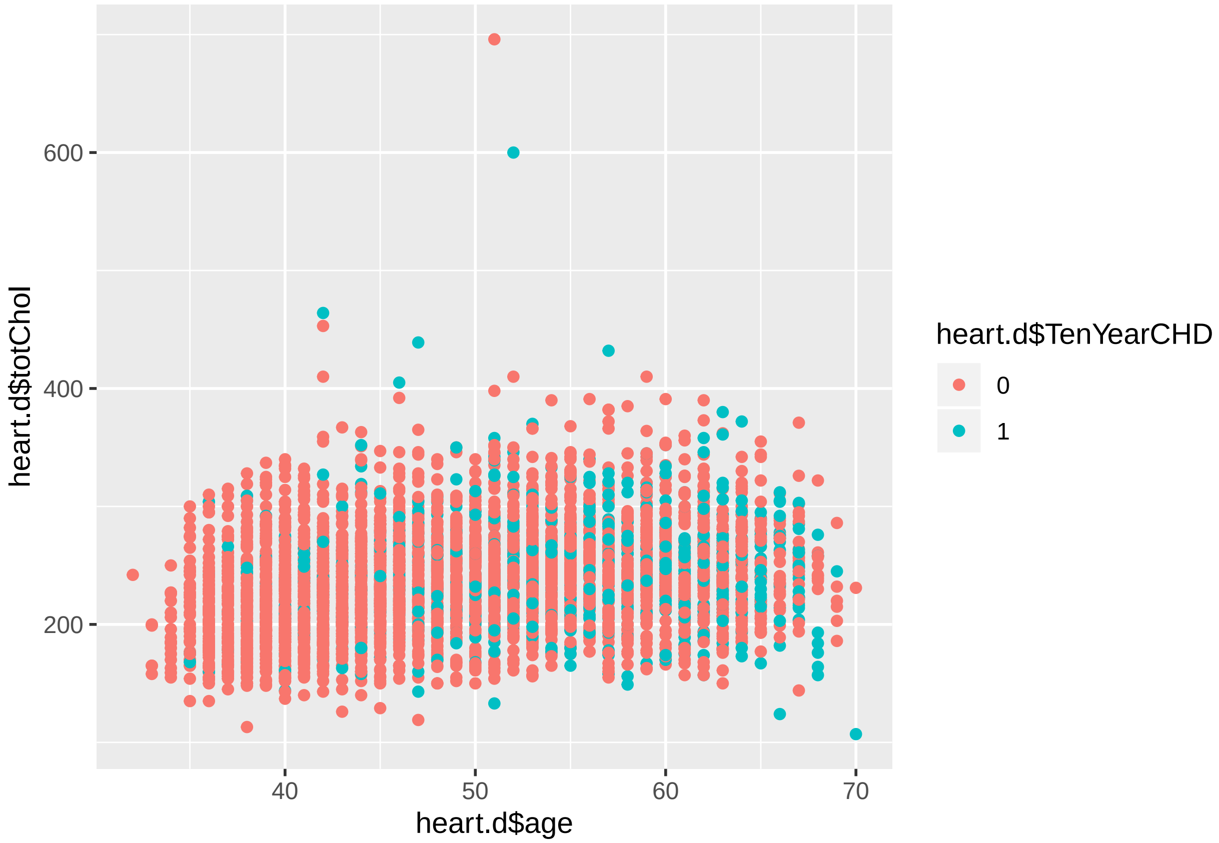


BIVARIATE ANALYSIS:

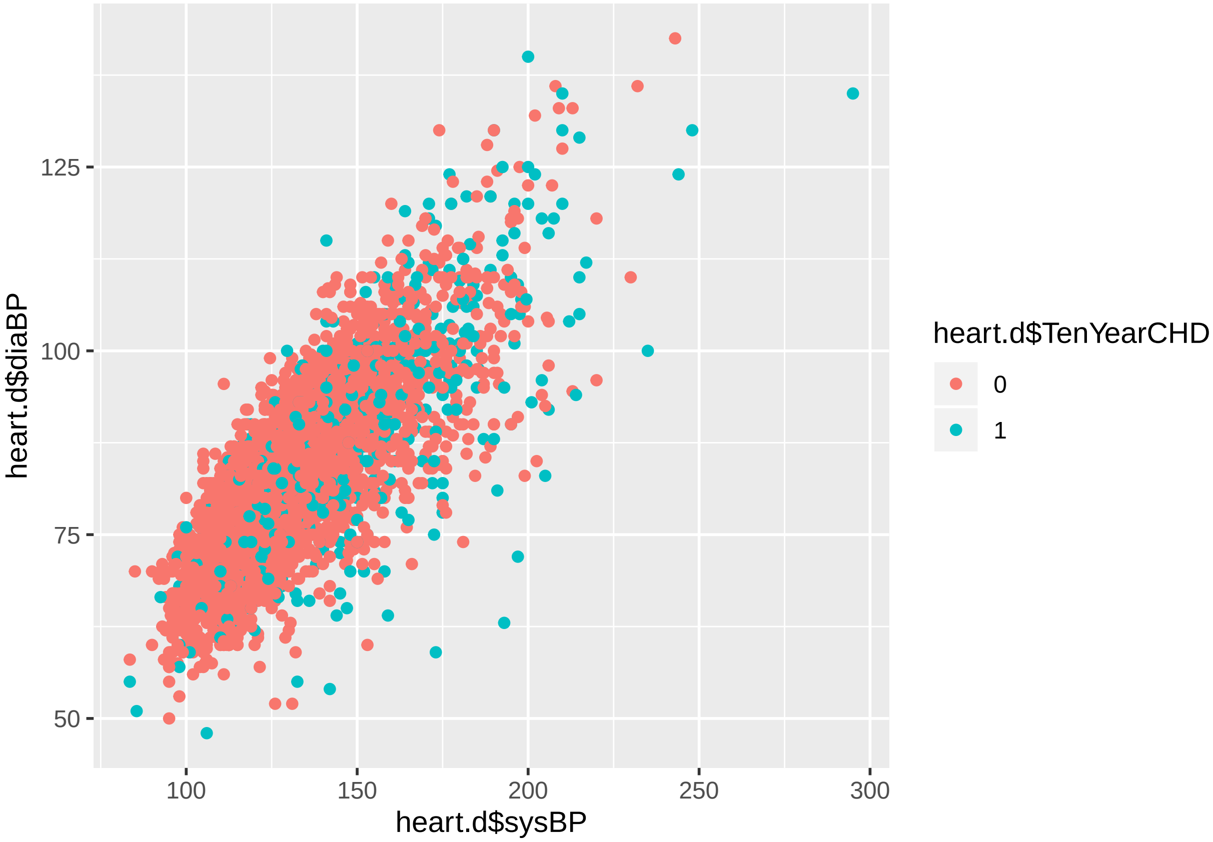
```
library(ggplot2)

# age and cholesterol
agechol1 = ggplot(data = heart.d, aes(x = heart.d$age,
y = heart.d$totChol ,color = heart.d$TenYearCHD)) + geom_point()

# SysBP and DiaBP
sysdia2 = ggplot(data = heart.d, aes(x = heart.d$sysBP,
y = heart.d$diaBP ,color = heart.d$TenYearCHD)) + geom_point()

# Cigs count and age
Cigsage3 = ggplot(data = heart.d, aes(x = heart.d$cigsPerDay,
y = heart.d$age ,color = heart.d$TenYearCHD)) + geom_point()

# Prevalent Hyp and Heart rate:
HypHr4 = ggplot(data = heart.d)+ geom_boxplot(aes(x = heart.d$prevalentHyp,
                                                       y = heart.d$heartRate,
                                                       fill = heart.d$TenYearCHD))

# Prevalent Hyp and Age
HypHr4.2 = ggplot(data = heart.d)+ geom_boxplot(aes(x = heart.d$prevalentHyp,
                                                       y = heart.d$age,
                                                       fill = heart.d$TenYearCHD))
```

```
# BP Meds and BMI
BPMBMI5 = ggplot(data = heart.d)+ geom_boxplot(aes(x = heart.d$BPMeds,
                                                   y = heart.d$BMI,
                                                   fill = heart.d$TenYearCHD))


agechol1
```

```
## Warning: Removed 50 rows containing missing values (geom_point).
```

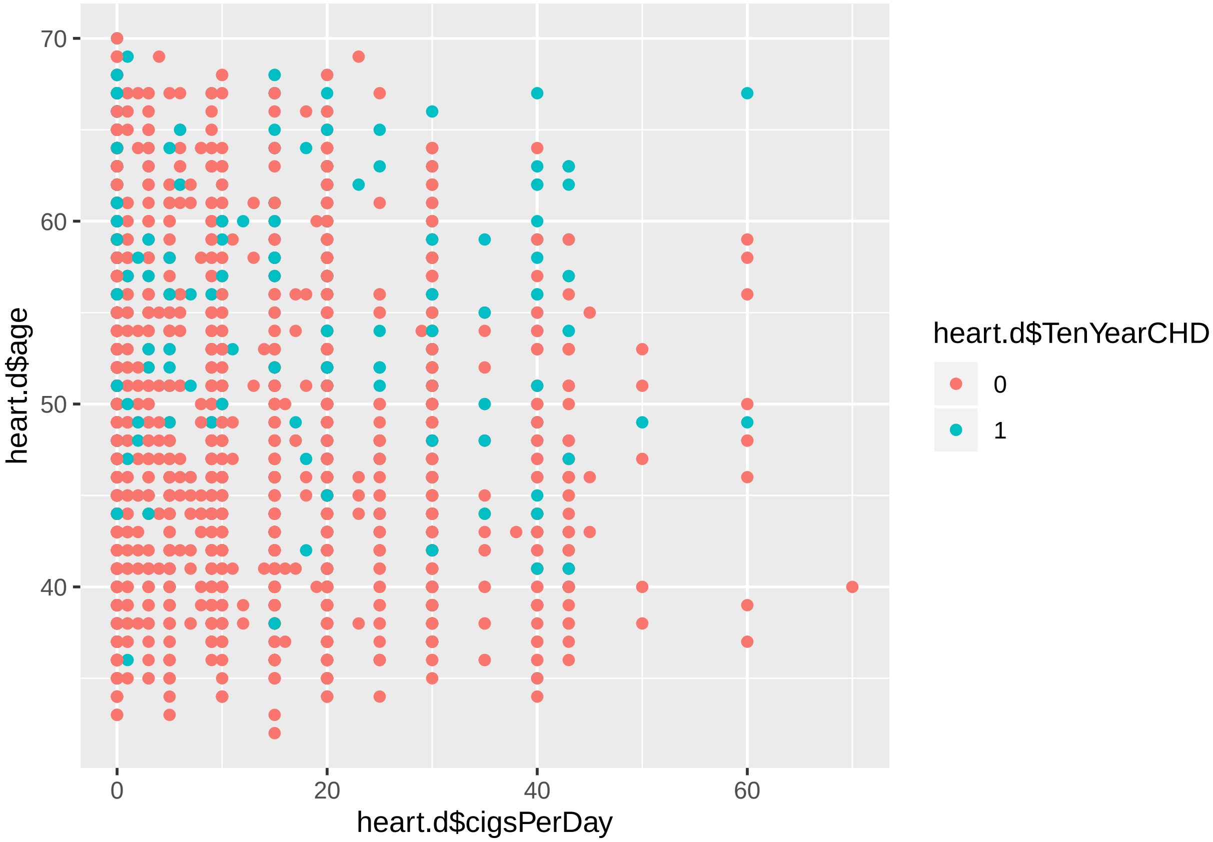


```
sysdia2
```



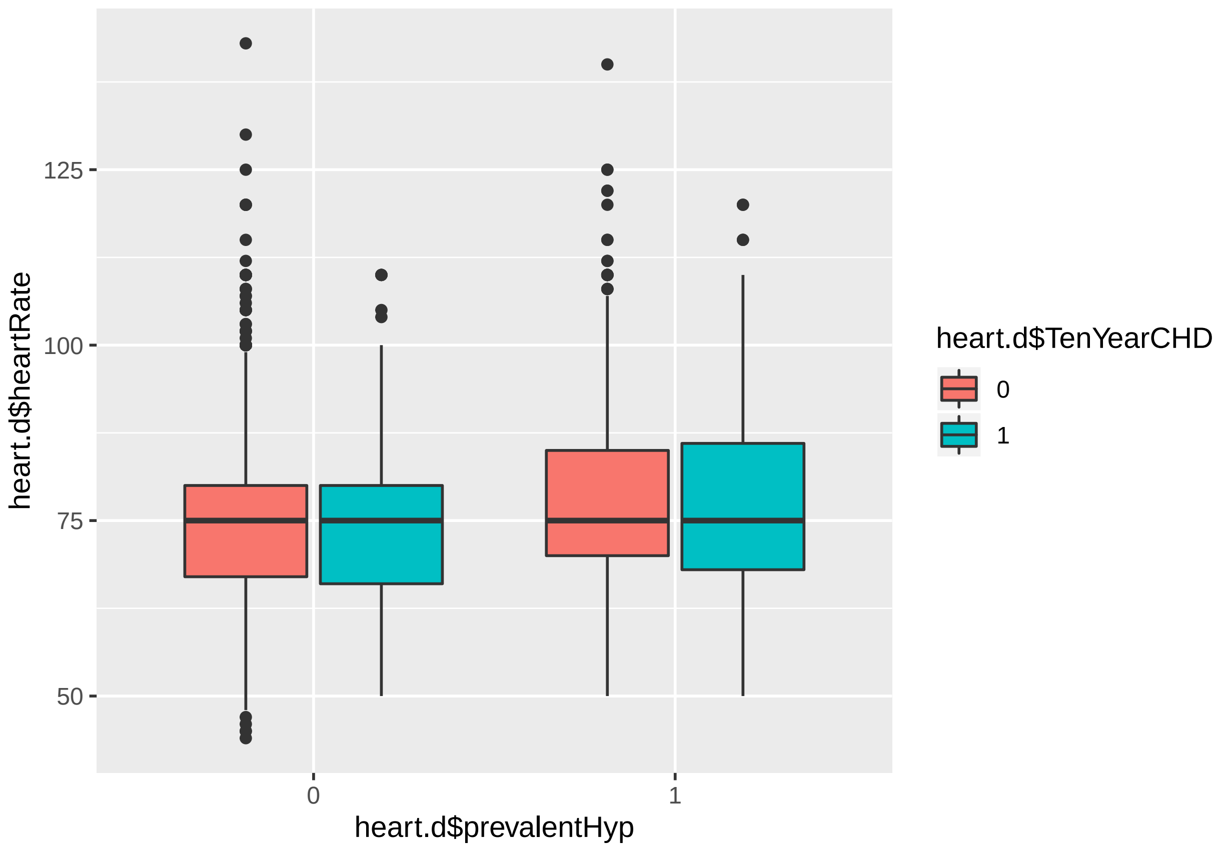
```
Cigsage3
```

```
## Warning: Removed 29 rows containing missing values (geom_point).
```

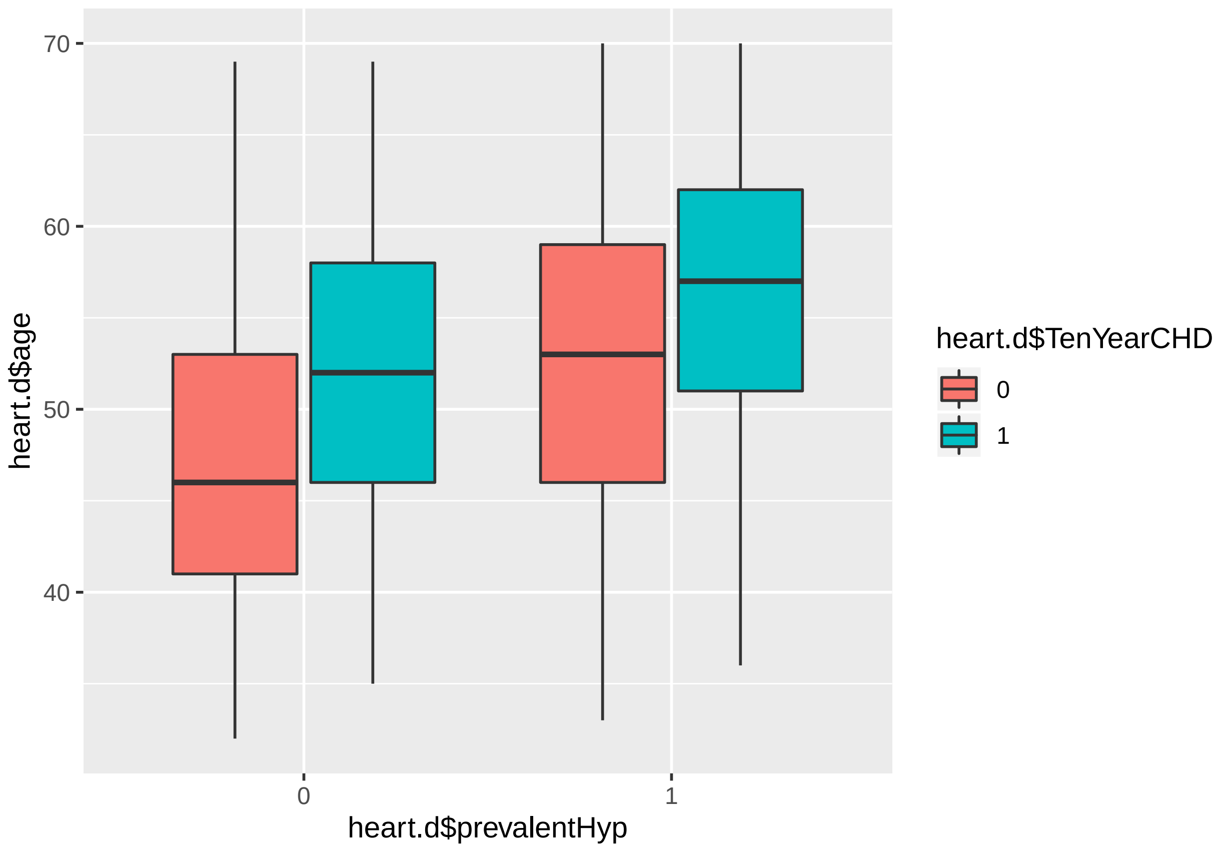


```
HypHr4
```

```
## Warning: Removed 1 rows containing non-finite values (stat_boxplot).
```

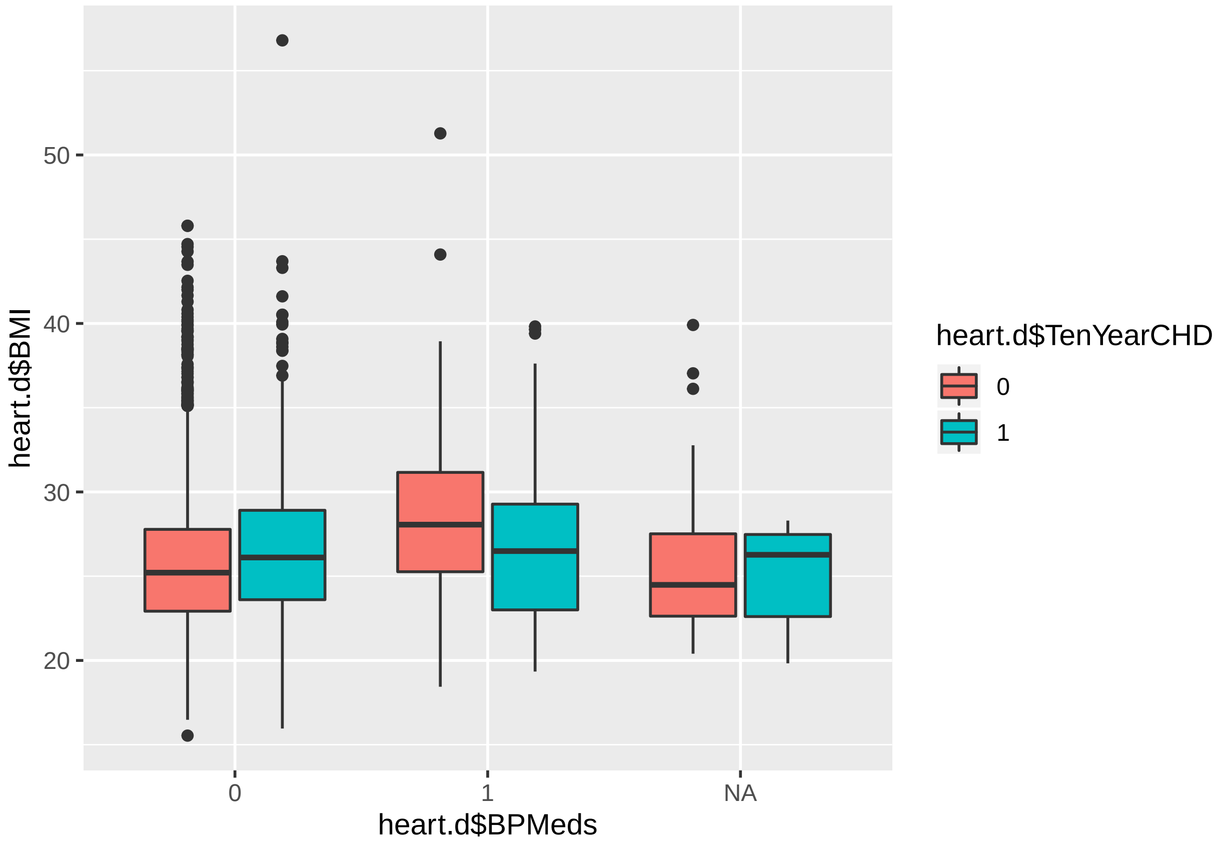

heart.d$heartRate
125
100
75
50
0
1
heart.d$prevalentHyp
heart.d$TenYearCHD
0
1

```
HypHr4.2
```



```
BPMBMI5
```

```
## Warning: Removed 19 rows containing non-finite values (stat_boxplot).
```



```
# Missing value treatment

# MICE:
library(mice)
```

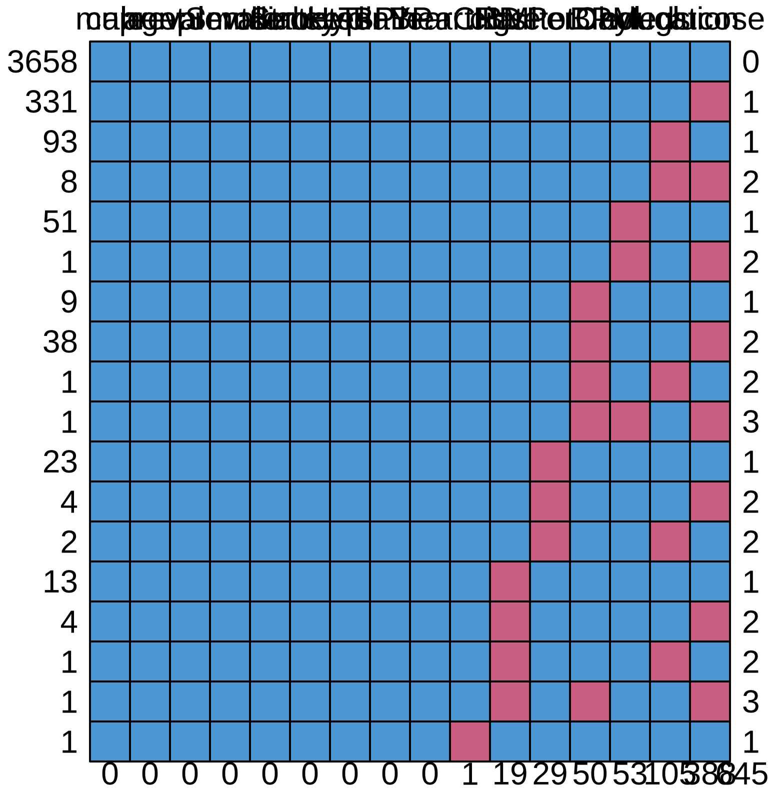
```
## Warning: package 'mice' was built under R version 3.6.1
```

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

```
##
## Attaching package: 'mice'
```

```
## The following objects are masked from 'package:base':
##
##     cbind, rbind
```

```
md.pattern(heart.d)
```



```
##      male age currentSmoker prevalentStroke prevalentHyp diabetes sysBP
## 3658    1   1             1               1            1        1     1
## 331     1   1             1               1            1        1     1
## 93      1   1             1               1            1        1     1
## 8       1   1             1               1            1        1     1
## 51      1   1             1               1            1        1     1
## 1       1   1             1               1            1        1     1
## 9       1   1             1               1            1        1     1
## 38      1   1             1               1            1        1     1
## 1       1   1             1               1            1        1     1
## 1       1   1             1               1            1        1     1
## 23      1   1             1               1            1        1     1
## 4       1   1             1               1            1        1     1
## 2       1   1             1               1            1        1     1
## 13      1   1             1               1            1        1     1
## 4       1   1             1               1            1        1     1
## 1       1   1             1               1            1        1     1
## 1       1   1             1               1            1        1     1
## 1       1   1             1               1            1        1     1
##          0   0             0               0            0        0     0
##      diaBP TenYearCHD heartRate BMI cigsPerDay totChol BPMeds education
## 3658     1          1         1   1          1       1      1         1
## 331      1          1         1   1          1       1      1         1
## 93       1          1         1   1          1       1      1         0
## 8        1          1         1   1          1       1      1         0
## 51       1          1         1   1          1       1      0         1
## 1        1          1         1   1          1       1      0          1
## 9        1          1         1   1          1       0      1          1
## 38       1          1         1   1          1       0      1          1
## 1        1          1         1   1          1       0      1          0
## 1        1          1         1   1          1       0      0          1
## 23       1          1         1   1          0       1      1          1
## 4        1          1         1   1          0       1      1          1
## 2        1          1         1   1          0       1      1          0
## 13       1          1         1   0          1       1      1          1
## 4        1          1         1   0          1       1      1          1
## 1        1          1         1   0          1       1      1          0
## 1        1          1         1   0          1       0      1          1
## 1        1          1         0   1          1       1      1          1
##          0          0         1  19         29      50     53        105
##      glucose
## 3658       1   0
## 331        0   1
## 93         1   1
## 8          0   2
## 51         1   1
## 1          0   2
## 9          1   1
## 38         0   2
## 1          1   2
## 1          0   3
## 23         1   1
## 4          0   2
## 2          1   2
## 13         1   1
## 4          0   2
## 1          1   2
## 1          0   3
## 1          1   1
##           388 645
```

```
# 1. REMOVING NA VALUES:
dataforNA = heart.d
dataforNA = na.omit(dataforNA)
1-(3658/4240)
```

```
## [1] 0.1372642
```

```
# 86.27% data is only clean
# 13.73% data is missing

anyNA(dataforNA)
```

```
## [1] FALSE
```

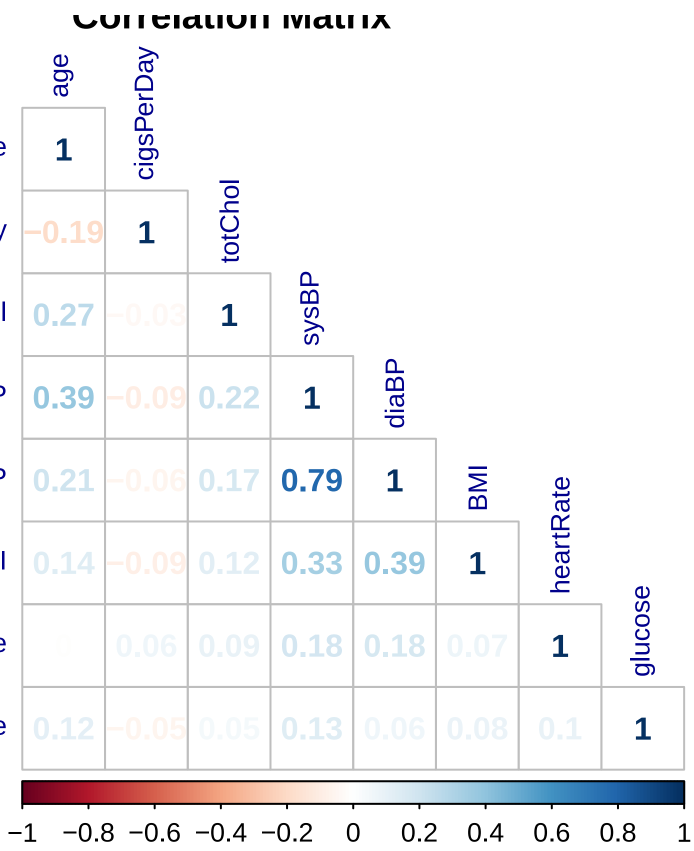
CORRELATION:

```
library(corrplot)
```

```
## Warning: package 'corrplot' was built under R version 3.6.2
```

```
## corrplot 0.84 loaded
```

```
heartcorr = corrplot(cor(dataforNA[, c(2,5, 10:15)]), method = "number", type = "lower", title = "Correl
```



```
#Pairwise correlation

library(ppcor)
```

```
## Warning: package 'ppcor' was built under R version 3.6.1

## Loading required package: MASS
```

```
pcor(dataforNA[, c(2,5, 10:15)], method = "pearson")
```

```
## $estimate
##                    age   cigsPerDay     totChol       sysBP        diaBP
## age         1.00000000 -0.154835298 0.210205060  0.34076598 -0.166223048
## cigsPerDay -0.15483530  1.000000000 0.022591777 -0.01798002  0.008122947
## totChol     0.21020506  0.022591777 1.000000000  0.04379225  0.024858780
## sysBP       0.34076598 -0.017980017 0.043792252  1.00000000  0.749437970
## diaBP      -0.16622305  0.008122947 0.024858780  0.74943797  1.000000000
## BMI         0.02543646 -0.059999728 0.044952785  0.01798060  0.220981347
## heartRate  -0.07965411  0.072165033 0.069386258  0.08378496  0.042539111
## glucose     0.05768621 -0.032898880 0.001038711  0.09550652 -0.071830119
##                    BMI    heartRate      glucose
## age         0.025436465 -0.079654107  0.057686208
## cigsPerDay -0.059999728  0.072165033 -0.032898880
## totChol     0.044952785  0.069386258  0.001038711
## sysBP       0.017980596  0.083784957  0.095506519
## diaBP       0.220981347  0.042539111 -0.071830119
## BMI         1.000000000  0.002510351  0.052899322
## heartRate   0.002510351  1.000000000  0.084876926
## glucose     0.052899322  0.084876926  1.000000000
## 
## $p.value
##                      age   cigsPerDay      totChol         sysBP
## age         0.000000e+00 4.934421e-21 9.481537e-38 5.673968e-100
## cigsPerDay  4.934421e-21 0.000000e+00 1.722631e-01  2.773550e-01
## totChol     9.481537e-38 1.722631e-01 0.000000e+00  8.125559e-03
## sysBP      5.673968e-100 2.773550e-01 8.125559e-03  0.000000e+00
## diaBP       4.880701e-24 6.236215e-01 1.331028e-01  0.000000e+00
## BMI         1.243195e-01 2.857163e-04 6.587258e-03  2.773396e-01
## heartRate   1.438097e-06 1.269713e-05 2.707628e-05  3.968599e-07
## glucose     4.870281e-04 4.681220e-02 9.499658e-01  7.343429e-09
##                   diaBP          BMI    heartRate      glucose
## age        4.880701e-24 1.243195e-01 1.438097e-06 4.870281e-04
## cigsPerDay 6.236215e-01 2.857163e-04 1.269713e-05 4.681220e-02
## totChol    1.331028e-01 6.587258e-03 2.707628e-05 9.499658e-01
## sysBP      0.000000e+00 2.773396e-01 3.968599e-07 7.343429e-09
## diaBP      0.000000e+00 1.238622e-41 1.014059e-02 1.393080e-05
## BMI        1.238622e-41 0.000000e+00 8.794604e-01 1.384107e-03
## heartRate  1.014059e-02 8.794604e-01 0.000000e+00 2.794874e-07
## glucose    1.393080e-05 1.384107e-03 2.794874e-07 0.000000e+00
## 
## $statistic
##                   age cigsPerDay     totChol     sysBP       diaBP
## age          0.000000 -9.4685986 12.98981429 21.898097 -10.1840828
## cigsPerDay  -9.468599  0.0000000  1.36523582 -1.086442   0.4907659
## totChol     12.989814  1.3652358  0.00000000  2.648260   1.5023131
## sysBP       21.898097 -1.0864425  2.64825958  0.000000  68.3872629
## diaBP      -10.184083  0.4907659  1.50231314 68.387263   0.0000000
## BMI          1.537247 -3.6314398  2.71858099  1.086477  13.6890593
## heartRate   -4.827661  4.3712642  4.20211438  5.079749   2.5723386
## glucose      3.490939 -1.9886699  0.06275398  5.796545  -4.3508720
##                   BMI  heartRate     glucose
## age         1.5372472 -4.8276607  3.49093878
## cigsPerDay -3.6314398  4.3712642 -1.98866991
## totChol     2.7185810  4.2021144  0.06275398
## sysBP       1.0864775  5.0797486  5.79654545
## diaBP      13.6890593  2.5723386 -4.35087196
## BMI         0.0000000  0.1516639  3.20040573
## heartRate   0.1516639  0.0000000  5.14643016
## glucose     3.2004057  5.1464302  0.00000000
## 
## $n
## [1] 3658
## 
## $gp
## [1] 6
## 
## $method
## [1] "pearson"
```

```
# The major correlation (78%) is between Systolic and Diastolic BP
# A minor correlation between SysBP and Age, and DiaBP and BMI.
```

```
# OUTLIERS TREATMENT:

treatOut <- function(x) {
  quant <- quantile(x, probs=c(.25, .75), na.rm = T)
  cap <- quantile(x, probs=c(.05, .95), na.rm = T)
  D <- 1.5 * IQR(x, na.rm = T)
  x[ x < (quant[1] - D )] <- cap[1]
  x[ x > (quant[2] + D) ] <- cap[2]
  return(x)
}

# treating the outliers with the function above. It helps to replace the lower 25th percentile with the

dataforNA$cigsPerDay= treatOut(dataforNA$cigsPerDay)
summary(dataforNA$cigsPerDay)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   0.000   0.000   0.000   8.941  20.000  50.000
```

```
dataforNA$totChol= treatOut(dataforNA$totChol)
summary(dataforNA$totChol)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##     124     206     234     236     263     347
```

```
dataforNA$sysBP= treatOut(dataforNA$sysBP)
summary(dataforNA$sysBP)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##    83.5   117.0   128.0   131.6   143.9   184.0
```

```
dataforNA$diaBP= treatOut(dataforNA$diaBP)
summary(dataforNA$diaBP)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   53.00   75.00   82.00   82.64   90.00  112.50
```

```
dataforNA$BMI= treatOut(dataforNA$BMI)
summary(dataforNA$BMI)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   15.96   23.08   25.38   25.64   28.04   35.42
```

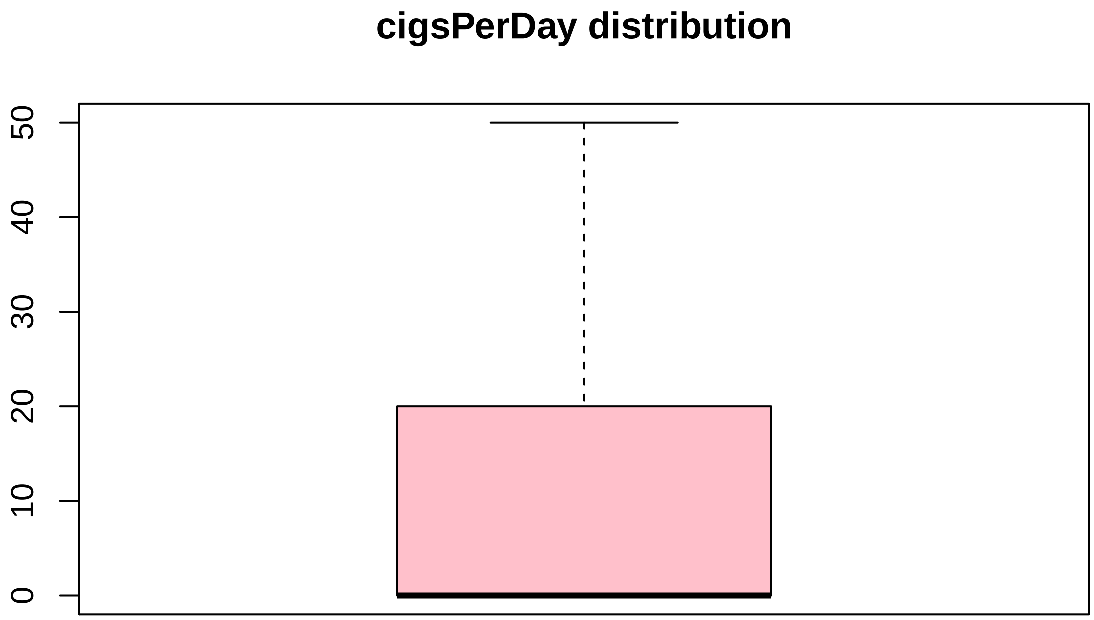
```
dataforNA$heartRate= treatOut(dataforNA$heartRate)
summary(dataforNA$heartRate)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   47.00   68.00   75.00   75.42   82.00  103.00
```

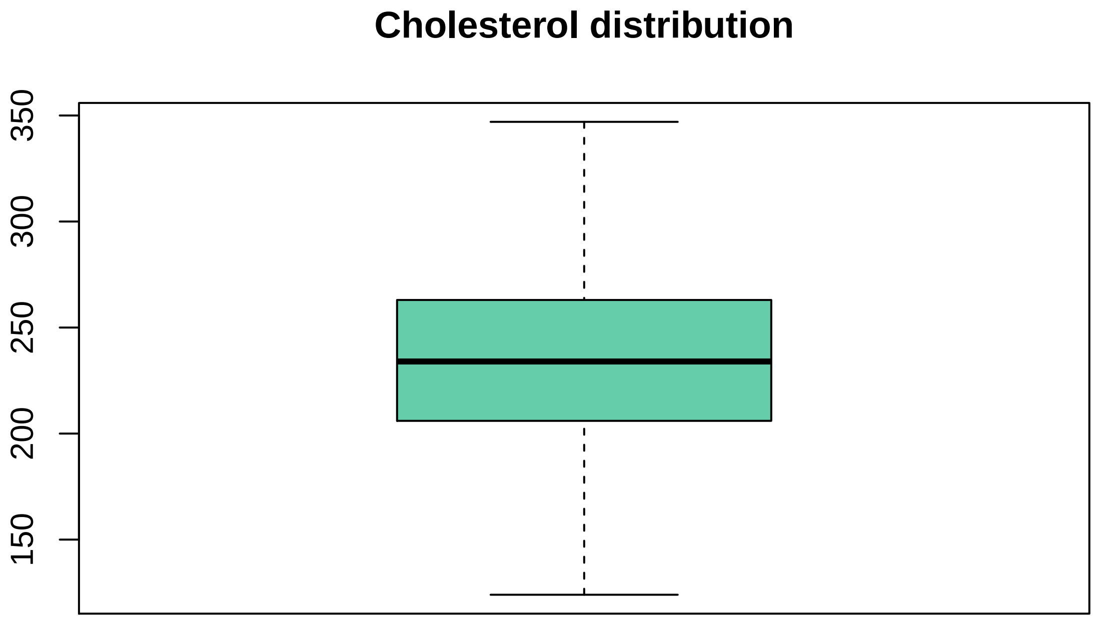
```
dataforNA$glucose= treatOut(dataforNA$glucose)
summary(dataforNA$glucose)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   47.00   71.00   78.00   79.74   87.00  111.00
```

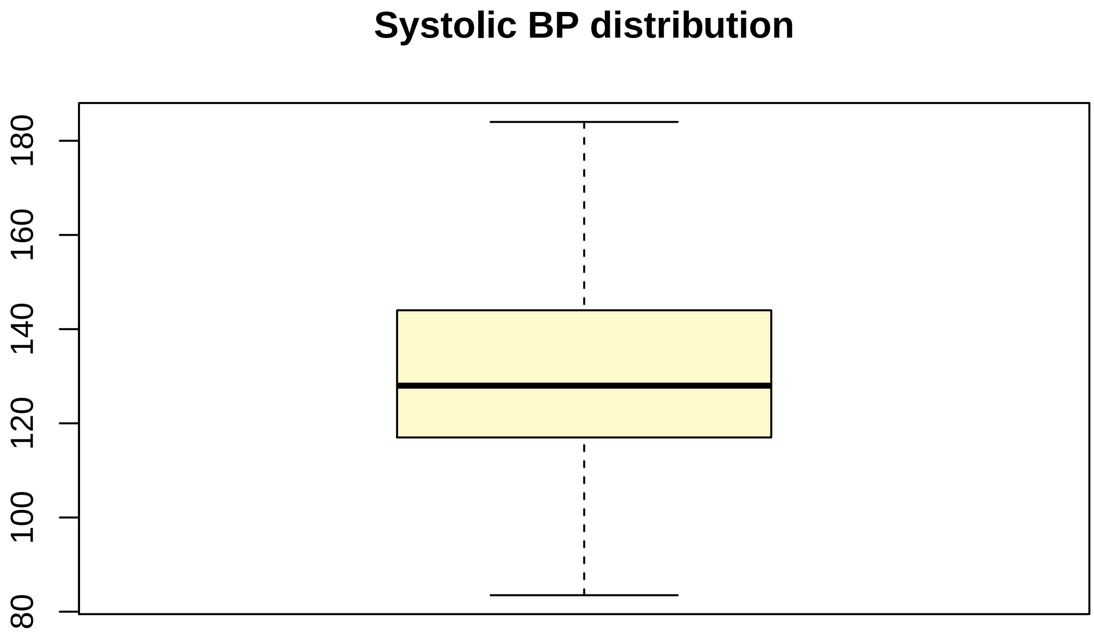
```r
# BOXPLOTS:


# Cigerattes per day
boxplot(dataforNA$cigsPerDay, data = dataforNA, col = "pink", main = "cigsPerDay distribution")
```



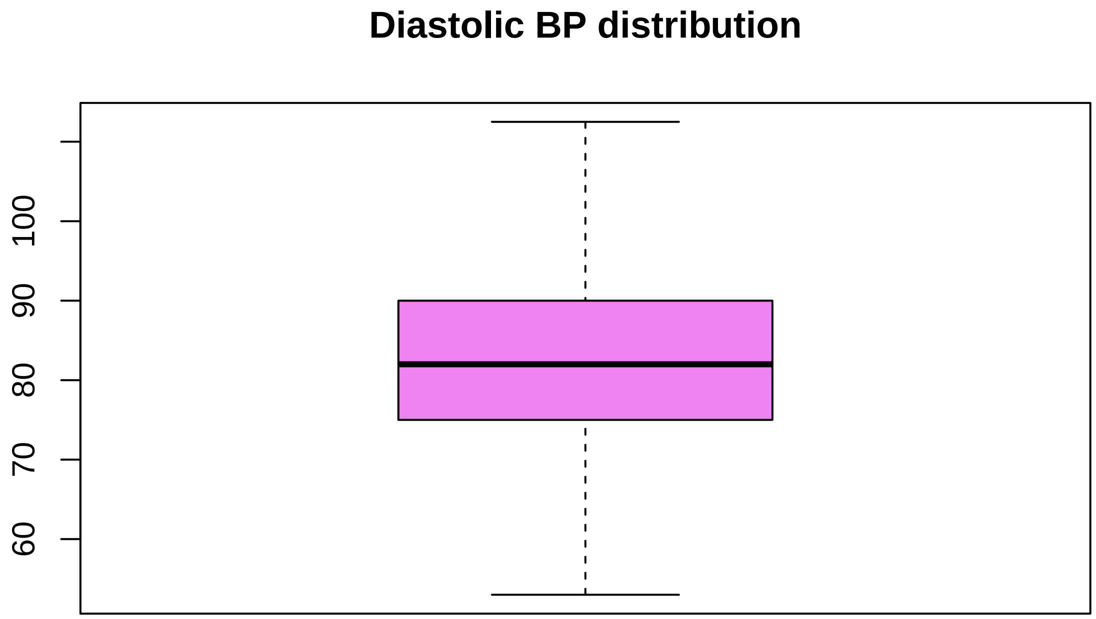
```r
# 2 outliers present

# Total Cholesterol:
boxplot(dataforNA$totChol, data = dataforNA, col = "aquamarine3", main = "Cholesterol distribution")
```



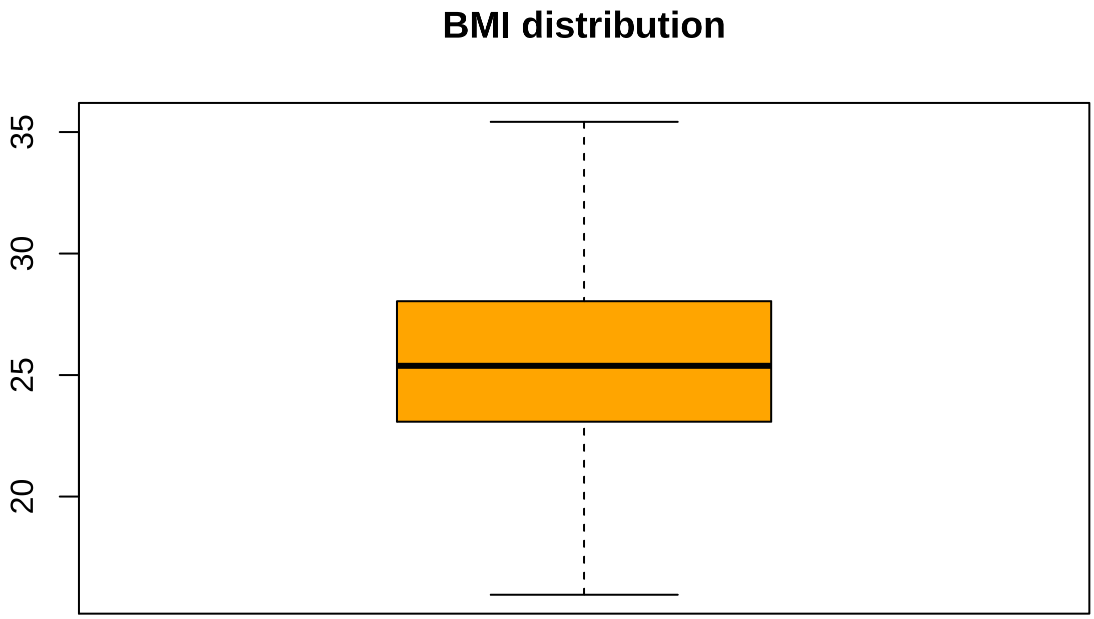
```
# Systolic BP
boxplot(dataforNA$sysBP, data = dataforNA, col = "lemonchiffon", main = "Systolic BP distribution")
```



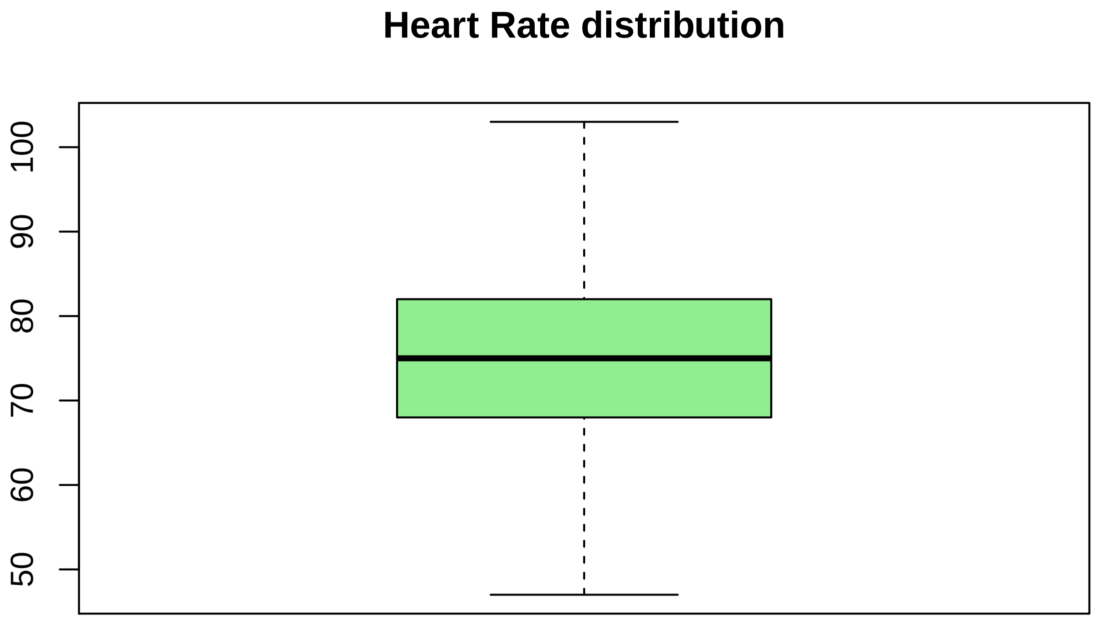
```
# Diastolic BP
boxplot(dataforNA$diaBP, data = dataforNA, col = "violet", main = "Diastolic BP distribution")
```



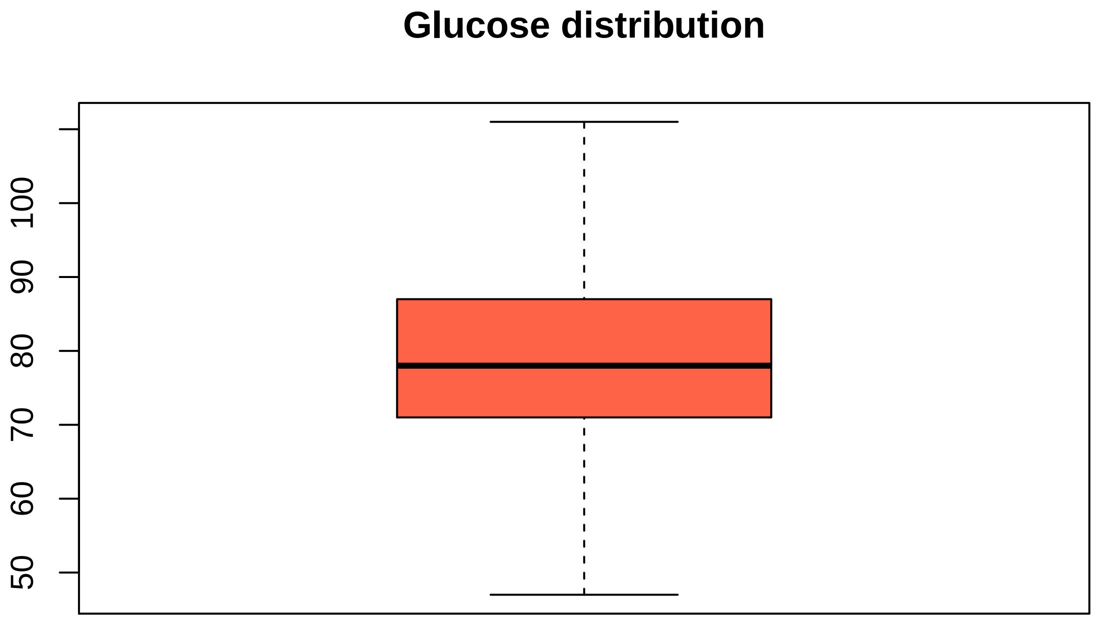
```r
# Body Mass Index:
boxplot(dataforNA$BMI, data = dataforNA, col = "orange", main = "BMI distribution")
```



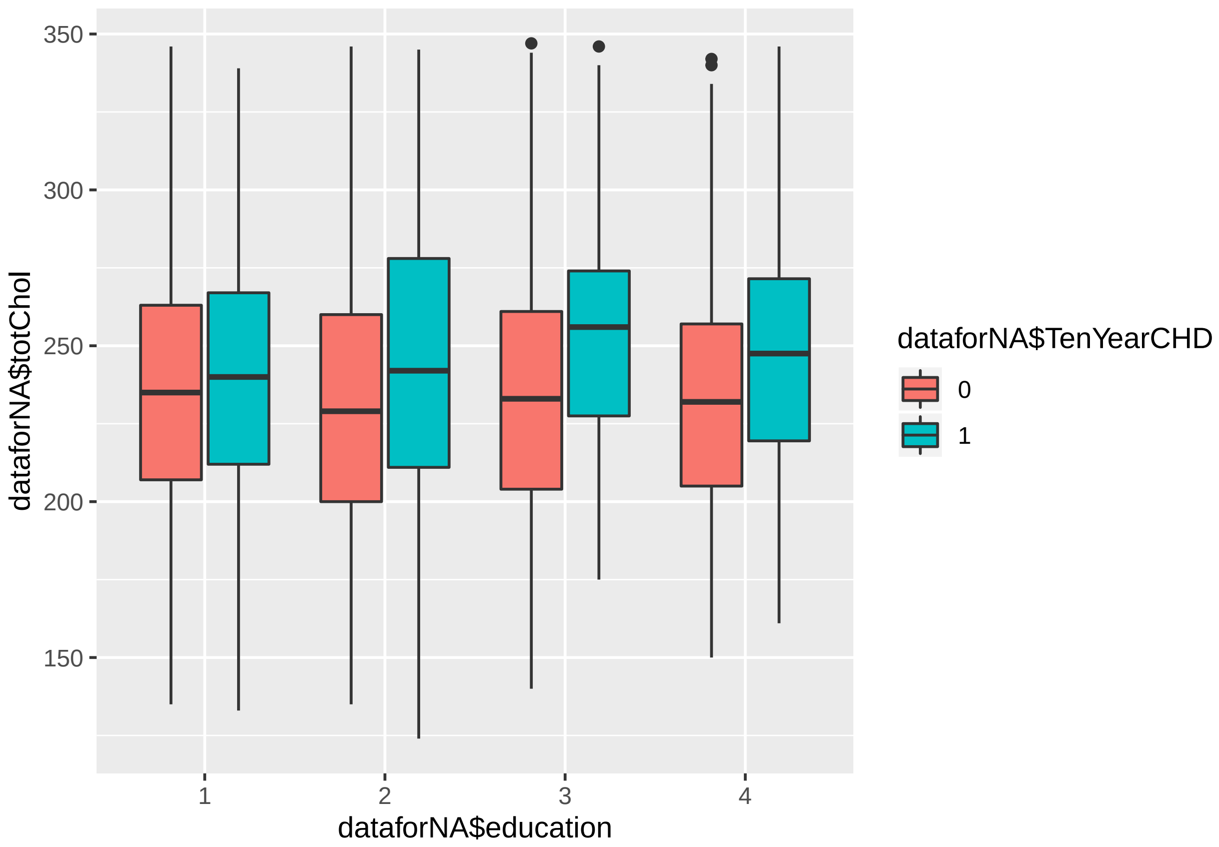
```r
# Heart Rate:
boxplot(dataforNA$heartRate, data = dataforNA, col = "lightgreen", main = "Heart Rate distribution")
```



```r
# Glucose
boxplot(dataforNA$glucose, data = dataforNA, col = "tomato", main = "Glucose distribution")
```



```
# Education and glucose:
EdGluc6 = ggplot(data = dataforNA)+ geom_boxplot(aes(x = dataforNA$education,
                                                     y = dataforNA$totChol,
                                                     fill = dataforNA$TenYearCHD))
EdGluc6
```



```
# We should make a change in the approach not to include or manipulate the NA values.
# Pure data is available (about 88%). So let us go with the available data.
# We should build a model with the manipulated values as well as the removed values.

library(car)
```

```
## Warning: package 'car' was built under R version 3.6.1
```

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

```
## Registered S3 methods overwritten by 'car':
##   method                          from
##   influence.merMod                lme4
##   cooks.distance.influence.merMod lme4
##   dfbeta.influence.merMod         lme4
##   dfbetas.influence.merMod        lme4
```

```
library(caTools)
```

```
## Warning: package 'caTools' was built under R version 3.6.1
```

```
set.seed(248)
samplNA = sample.split(dataforNA, SplitRatio = 0.75)
wNAtrain = subset(dataforNA,samplNA == TRUE)
wNAtest = subset(dataforNA, samplNA == FALSE)

prop.table(table(wNAtest$TenYearCHD))
```

```
##
```

```
##         0         1 
## 0.8535519 0.1464481
```

```r
prop.table(table(wNAtrain$TenYearCHD))
```

```
## 
##         0         1 
## 0.8457893 0.1542107
```

```r
mod1 = glm(TenYearCHD~. , data = wNAtrain, family = "binomial")
summary(mod1)
```

```
## 
## Call:
## glm(formula = TenYearCHD ~ ., family = "binomial", data = wNAtrain)
## 
## Deviance Residuals: 
##     Min       1Q   Median       3Q      Max  
## -1.6626  -0.5950  -0.4200  -0.2775   2.9039  
## 
## Coefficients:
##                   Estimate Std. Error z value Pr(>|z|)    
## (Intercept)      -8.233475   0.919343  -8.956  < 2e-16 ***
## male1             0.485478   0.126960   3.824 0.000131 ***
## age               0.067377   0.007750   8.694  < 2e-16 ***
## education2       -0.159283   0.143141  -1.113 0.265807    
## education3       -0.100605   0.170122  -0.591 0.554275    
## education4       -0.092057   0.193485  -0.476 0.634228    
## currentSmoker1   -0.063816   0.185302  -0.344 0.730556    
## cigsPerDay        0.025313   0.007547   3.354 0.000796 ***
## BPMeds1          -0.033053   0.280175  -0.118 0.906090    
## prevalentStroke1  1.923565   0.676970   2.841 0.004491 ** 
## prevalentHyp1     0.178703   0.167020   1.070 0.284640    
## diabetes1         0.715954   0.289083   2.477 0.013263 *  
## totChol           0.001653   0.001428   1.158 0.246814    
## sysBP             0.022811   0.005150   4.429 9.45e-06 ***
## diaBP            -0.014704   0.007994  -1.840 0.065840 .  
## BMI               0.004932   0.016870   0.292 0.770001    
## heartRate        -0.001299   0.005245  -0.248 0.804407    
## glucose           0.003280   0.004630   0.708 0.478642    
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 
## (Dispersion parameter for binomial family taken to be 1)
## 
##     Null deviance: 2358.7  on 2742  degrees of freedom
## Residual deviance: 2067.5  on 2725  degrees of freedom
## AIC: 2103.5
## 
## Number of Fisher Scoring iterations: 5
```

```r
# Prediction on test data:
# Let us predict on the test with mod1 logistic model

test.predict1 = predict(mod1, newdata = wNAtest, type = "response")
table(wNAtest$TenYearCHD, test.predict1>0.5)
```

```
##
##     FALSE TRUE
##   0   769   12
##   1   127    7
```

```
# overall accuracy - 84.81%
# specificity - 98.46%
# 5.2% sensitivity
# 2103.5 -> AIC

#    FALSE TRUE
#  0   769   12
#  1   127    7

mod2 = glm(TenYearCHD~male+age+sysBP+cigsPerDay+prevalentStroke+diabetes, data = wNAtrain, family = "bi
summary(mod2)
```

```
##
## Call:
## glm(formula = TenYearCHD ~ male + age + sysBP + cigsPerDay +
##     prevalentStroke + diabetes, family = "binomial", data = wNAtrain)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.7014  -0.5972  -0.4227  -0.2811   2.8625
##
## Coefficients:
##                   Estimate Std. Error z value Pr(>|z|)
## (Intercept)      -8.692898   0.500445 -17.370  < 2e-16 ***
## male1             0.460567   0.121493   3.791  0.00015 ***
## age               0.073153   0.007373   9.922  < 2e-16 ***
## sysBP             0.019959   0.002837   7.035 1.99e-12 ***
## cigsPerDay        0.023060   0.004928   4.679 2.88e-06 ***
## prevalentStroke1  1.895331   0.668940   2.833  0.00461 **
## diabetes1         0.820029   0.268703   3.052  0.00227 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 2358.7  on 2742  degrees of freedom
## Residual deviance: 2074.9  on 2736  degrees of freedom
## AIC: 2088.9
##
## Number of Fisher Scoring iterations: 5
```

```
test.predict2 = predict(mod2, newdata = wNAtest, type = "response")
table(wNAtest$TenYearCHD, test.predict2>0.5)
```

```
##
##     FALSE TRUE
##   0   769   12
##   1   127    7
```

```
# Not a change in the model is seen. The values are just he same.
# 2088.9 -> AIC
# doing smote

library(DMwR)
```

```
## Warning: package 'DMwR' was built under R version 3.6.1

## Loading required package: grid

## Registered S3 method overwritten by 'xts':
##   method     from
##   as.zoo.xts zoo

## Registered S3 method overwritten by 'quantmod':
##   method            from
##   as.zoo.data.frame zoo
```

```
sm.train = subset(dataforNA, samplNA == TRUE)
sm.test = subset(dataforNA, samplNA == FALSE)

# on the NA set

prop.table(table(sm.train$TenYearCHD))
```

```
##
##         0         1
## 0.8457893 0.1542107
```

```
prop.table(table(sm.test$TenYearCHD))
```

```
##
##         0         1
## 0.8535519 0.1464481
```

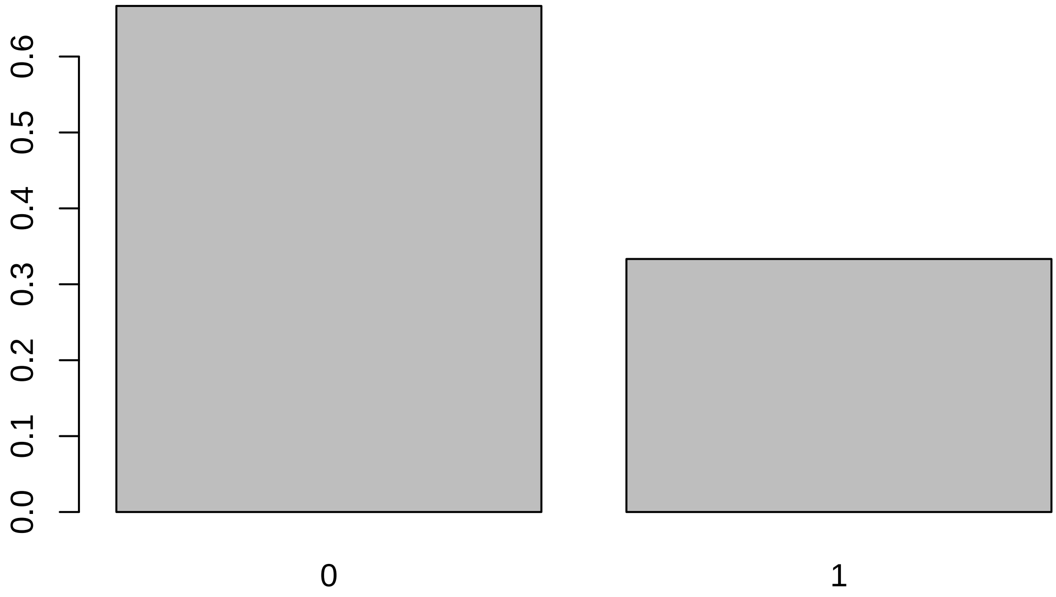
```
balanced.train = SMOTE(TenYearCHD~., sm.train, perc.over = 100, k = 5, perc.under = 400)
table(balanced.train$TenYearCHD)
```

```
##
##    0    1
## 1692  846
```

```
prop.table(table(balanced.train$TenYearCHD))
```

```
##
##         0         1
## 0.6666667 0.3333333
```

```
barplot(prop.table(table(balanced.train$TenYearCHD)))
```



```r
# we have like 66 to 33 percentage now
```

```r
library(car)

mod3 = glm(TenYearCHD~. , data = balanced.train, family = "binomial")
summary(mod3)
```

```
##
## Call:
## glm(formula = TenYearCHD ~ ., family = "binomial", data = balanced.train)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.4681  -0.7826  -0.5168   0.8738   2.7006
##
## Coefficients:
##                     Estimate Std. Error z value Pr(>|z|)
## (Intercept)        -8.868449   0.774217 -11.455  < 2e-16 ***
## male1               0.383538   0.105083   3.650 0.000262 ***
## age                 0.061691   0.006754   9.134  < 2e-16 ***
## education2         -0.209279   0.120403  -1.738 0.082184 .
## education3         -0.156604   0.140596  -1.114 0.265340
## education4         -0.092581   0.159704  -0.580 0.562114
## currentSmoker1      0.341686   0.134928   2.532 0.011330 *
## cigsPerDay          0.009243   0.005916   1.562 0.118187
## BPMeds1             1.271206   0.204256   6.224 4.86e-10 ***
## prevalentStroke1    1.868723   0.439452   4.252 2.11e-05 ***
## prevalentHyp1     0.002853   0.130039   0.022 0.982499
## diabetes1         2.003430   0.256585   7.808 5.81e-15 ***
## totChol           0.002451   0.001241   1.975 0.048243 *
## sysBP             0.022239   0.004413   5.040 4.66e-07 ***
## diaBP            -0.008034   0.007143  -1.125 0.260705
## BMI               0.035924   0.014749   2.436 0.014862 *
## heartRate        -0.002859   0.004522  -0.632 0.527210
## glucose           0.009055   0.003940   2.298 0.021551 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 3230.9  on 2537  degrees of freedom
## Residual deviance: 2608.8  on 2520  degrees of freedom
## AIC: 2644.8
##
## Number of Fisher Scoring iterations: 4
```

```
vif(mod3)
```

```
##                      GVIF Df GVIF^(1/(2*Df))
## male             1.190536  1        1.091117
## age              1.271062  1        1.127414
## education        1.100553  3        1.016097
## currentSmoker    1.964845  1        1.401729
## cigsPerDay       2.086939  1        1.444624
## BPMeds           1.048776  1        1.024097
## prevalentStroke  1.011705  1        1.005836
## prevalentHyp     1.793576  1        1.339245
## diabetes         1.068260  1        1.033567
## totChol          1.075378  1        1.037005
## sysBP            3.218069  1        1.793898
## diaBP            2.682250  1        1.637758
## BMI              1.148236  1        1.071558
## heartRate        1.095848  1        1.046827
## glucose          1.099035  1        1.048349
```

```
# Prediction on test data:
# Let us predict on the test with mod1 logistic model

test.predict3 = predict(mod3, newdata = sm.test, type = "response")
table(sm.test$TenYearCHD, test.predict3>0.3)
```

```
##
##     FALSE TRUE
##   0   552  229
##   1    49   85
```

```
# MOST SIGNIFICANT *** : age, male, BP meds, prevStroke, diabetes, sysBP
# SIGNIFICANT * : current smoker, totchol, BMI, glucose

mod4 = glm(TenYearCHD~age+male+BPMeds+prevalentStroke+sysBP+diabetes+currentSmoker, data = balanced.tra
summary(mod4)
```

```
##
```

```
## Call:
## glm(formula = TenYearCHD ~ age + male + BPMeds + prevalentStroke +
##     sysBP + diabetes + currentSmoker, family = "binomial", data = balanced.train)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.5421  -0.7783  -0.5338   0.8866   2.5245
##
## Coefficients:
##                  Estimate Std. Error z value Pr(>|z|)
## (Intercept)     -7.604870   0.436384 -17.427  < 2e-16 ***
## age              0.065048   0.006363  10.223  < 2e-16 ***
## male1            0.435761   0.098597   4.420 9.89e-06 ***
## BPMeds1          1.275555   0.203113   6.280 3.38e-10 ***
## prevalentStroke1  1.859899   0.437129   4.255 2.09e-05 ***
## sysBP            0.021047   0.002620   8.032 9.61e-16 ***
## diabetes1        2.206345   0.248383   8.883  < 2e-16 ***
## currentSmoker1   0.426354   0.100645   4.236 2.27e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 3230.9  on 2537  degrees of freedom
## Residual deviance: 2631.3  on 2530  degrees of freedom
## AIC: 2647.3
##
## Number of Fisher Scoring iterations: 4
```

```
test.predict4 = predict(mod4, newdata = sm.test, type = "response")
table(sm.test$TenYearCHD, test.predict4>0.3)
```

```
##
##     FALSE TRUE
##   0   563  218
##   1    51   83
```

```
#    FALSE TRUE
# 0    563  218
# 1     51   83

# Spec -> 72.09
# Sens -> 61.94
# Over -> 70.60

test.predict4.1 = predict(mod4, newdata = sm.test, type = "response")
table(sm.test$TenYearCHD, test.predict4.1>0.4)
```

```
##
##     FALSE TRUE
##   0   652  129
##   1    77   57
```

```
## NAIVE BAYES:
library(e1071)
```

```
## Warning: package 'e1071' was built under R version 3.6.1
```

```r
library(caret)
```

```
## Warning: package 'caret' was built under R version 3.6.1
```

```r
# doing on the omitted NA set:
NB1 = naiveBayes(TenYearCHD~., data = balanced.train)
print(NB1)
```

```
##
## Naive Bayes Classifier for Discrete Predictors
##
## Call:
## naiveBayes.default(x = X, y = Y, laplace = laplace)
##
## A-priori probabilities:
## Y
##         0         1
## 0.6666667 0.3333333
##
## Conditional probabilities:
##    male
## Y           0         1
##   0 0.5626478 0.4373522
##   1 0.4657210 0.5342790
##
##    age
## Y       [,1]     [,2]
##   0 48.91726 8.532417
##   1 54.72766 7.214810
##
##    education
## Y           1         2         3         4
##   0 0.3788416 0.3102837 0.1891253 0.1217494
##   1 0.4917258 0.2434988 0.1560284 0.1087470
##
##    currentSmoker
## Y           0         1
##   0 0.5254137 0.4745863
##   1 0.4988180 0.5011820
##
##    cigsPerDay
## Y       [,1]     [,2]
##   0 8.765366 11.67928
##   1 9.382491 10.80129
##
##    BPMeds
## Y           0         1
##   0 0.9751773 0.0248227
##   1 0.8557920 0.1442080
##
##    prevalentStroke
## Y             0           1
##   0 0.995862884 0.004137116
##   1 0.959810875 0.040189125
##
##    prevalentHyp
## Y           0         1
##   0 0.7056738 0.2943262
##   1 0.4598109 0.5401891
##
##    diabetes
## Y            0          1
##   0 0.98699764 0.01300236
##   1 0.86288416 0.13711584
##
##    totChol
## Y        [,1]     [,2]
##   0 234.3960 41.28831
##   1 245.8707 38.37990
##
##    sysBP
## Y        [,1]     [,2]
##   0 130.2503 18.66421
##   1 143.6089 20.11318
##
##    diaBP
## Y        [,1]     [,2]
##   0 82.30230 10.72804
##   1 87.16569 11.45281
##
##    BMI
## Y        [,1]     [,2]
##   0 25.49467 3.595087
##   1 26.54812 3.460878
##
##    heartRate
## Y        [,1]     [,2]
##   0 75.64190 11.55883
##   1 76.51296 10.37642
##
##    glucose
## Y        [,1]     [,2]
##   0 78.97991 12.26938
##   1 83.17445 13.37521
```

```r
NB.pred1 = predict(NB1, sm.test, type = "class" )
table(NB.pred1, sm.test$TenYearCHD, dnn = c("Prediction", "Actual"))
```

```
##           Actual
## Prediction   0   1
##          0 619  70
##          1 162  64
```

```r
# overall accuracy - 74.64%
# specificity - 89.84%
# sensitivity - 28.31%

# this model where we have omitted for has given an approx 40% accuracy of finding risk bearers.
# The prediction of non risk bearers is good with the dataset that has NA removed (86.10%)

# We have an overall accuracy of 80.84% with only 88% of the data being useful

#           Actual
# Prediction   0   1
#          0  619  70
#          1  162  64

# only on the categorical variables:

NB2 = naiveBayes(TenYearCHD~., data = balanced.train[,-c(2,3,5,10:15)])
print(NB2)
```

```
##
## Naive Bayes Classifier for Discrete Predictors
##
## Call:
## naiveBayes.default(x = X, y = Y, laplace = laplace)
##
## A-priori probabilities:
## Y
##         0         1
## 0.6666667 0.3333333
##
## Conditional probabilities:
##    male
## Y           0         1
##   0 0.5626478 0.4373522
##   1 0.4657210 0.5342790
##
##    currentSmoker
## Y           0         1
##   0 0.5254137 0.4745863
##   1 0.4988180 0.5011820
##
##    BPMeds
## Y           0         1
##   0 0.9751773 0.0248227
##   1 0.8557920 0.1442080
##
##    prevalentStroke
## Y             0           1
##   0 0.995862884 0.004137116
##   1 0.959810875 0.040189125
##
##    prevalentHyp
## Y           0         1
##   0 0.7056738 0.2943262
##   1 0.4598109 0.5401891
##
##    diabetes
## Y            0          1
##   0 0.98699764 0.01300236
##   1 0.86288416 0.13711584
```

```
NB.pred2 = predict(NB2, sm.test, type = "class" )
table(NB.pred2, sm.test$TenYearCHD, dnn = c("Prediction", "Actual"))
```

```
##           Actual
## Prediction   0   1
##          0 738 114
##          1  43  20
```

```
#           Actual
# Prediction   0   1
#         0  738 114
#         1  43  20

# overall accuracy - 82.84%
# specificity - 86.62%
# sensitivity - 31.75%
```

```
# random forest:

library(randomForest)
```

```
## Warning: package 'randomForest' was built under R version 3.6.1
## randomForest 4.6-14
## Type rfNews() to see new features/changes/bug fixes.
##
## Attaching package: 'randomForest'
## The following object is masked from 'package:ggplot2':
##
##     margin
```

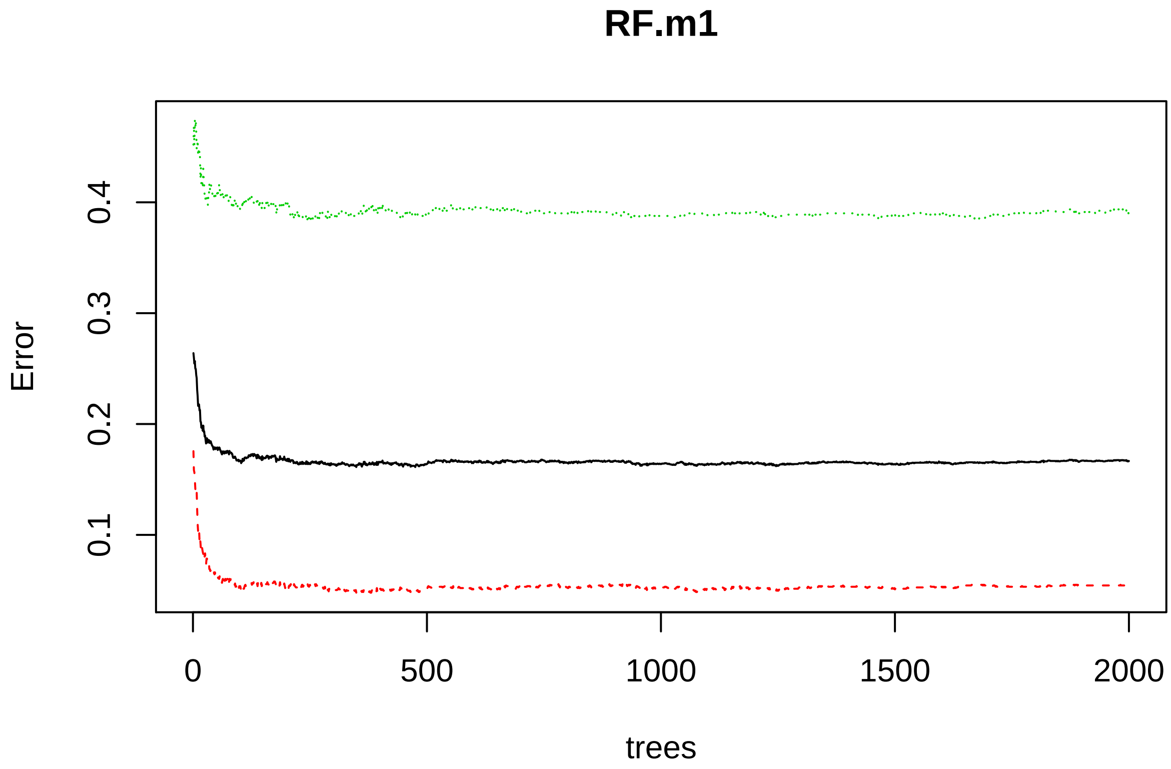
```
RFmtry.val = floor(sqrt(ncol(balanced.train)))
RFmtry.val
```

```
## [1] 4
```

```
# RF1
RF.m1 = randomForest(TenYearCHD~.,data = balanced.train,
                     ntree = 2000, mtry = RFmtry.val, nodesize = 10, importance = TRUE)

print(RF.m1)
```

```
##
## Call:
##  randomForest(formula = TenYearCHD ~ ., data = balanced.train,      ntree = 2000, mtry = RFmtry.val,
##                Type of random forest: classification
##                      Number of trees: 2000
## No. of variables tried at each split: 4
##
##         OOB estimate of  error rate: 16.67%
## Confusion matrix:
##       0    1 class.error
## 0 1600   92  0.05437352
## 1  331 515  0.39125296
```

```r
# We see an increased OOB rate, but the class error is lowered (39%).
# We also are able to see a good rise in the prediction of the heart risk bearers.
# Let us try to boost this model for a better result.

# Confusion matrix:
#      0   1 class.error
# 0 1602  90  0.05319149
# 1  334 512  0.39479905

# overall -> 83.29%
# spec -> 94.68%
# sens -> 60.52%

plot(RF.m1)
```

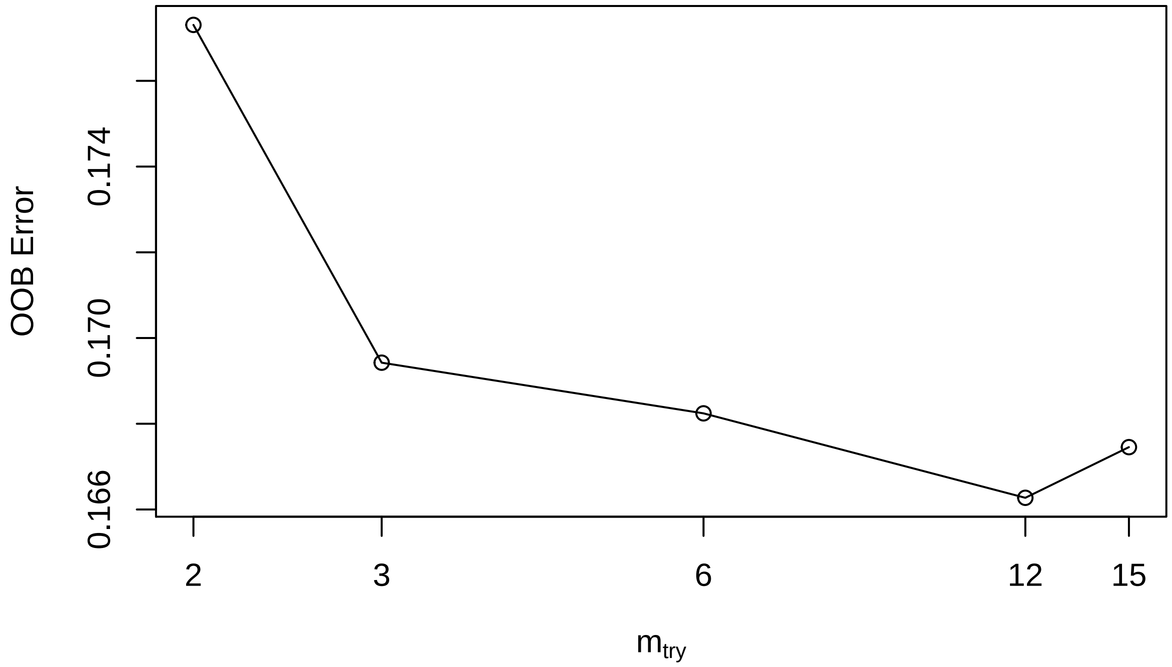


```r
importance(RF.m1)
```

```
##                         0          1 MeanDecreaseAccuracy MeanDecreaseGini
## male             37.64950   3.910741             35.70239        10.637036
## age             108.64038  82.337889            129.92255       131.447952
## education        41.49978   2.192878             37.05837        24.146251
## currentSmoker    67.37669 -64.453255             62.67301        38.236650
## cigsPerDay      102.58427  25.760910            111.51891       100.746873
## BPMeds           69.60891   9.351757             67.19038        25.540830
## prevalentStroke  25.97593  21.503032             31.92736         6.376223
## prevalentHyp     55.74786 -29.955942            53.94800        21.701717
## diabetes         74.50619  40.108679            77.18216        38.653019
## totChol          59.31384  17.337663            58.11372        63.411348
## sysBP            87.24799  31.341389           100.47808       111.279116
## diaBP            70.11349  18.736832            78.65527        74.437231
## BMI              65.55799  22.393193            66.47565        71.647512
## heartRate        59.95301  19.683526            59.50893        58.890256
## glucose          65.75196  26.565808            66.08974        65.411160
```

```
# from the importance values, we find that age seems to be the most important parameter to predict the

# TUNING:

t_RF.m1 = tuneRF(x=balanced.train[,-c(16)],y=balanced.train$TenYearCHD, mtrystart = 15, stepfactor= 1.5
                 ntree= 1700 , improve = 0.0001, nodesize=10, trace=TRUE, plot=TRUE, doBest=TRUE, impor
```

```
## mtry = 3  OOB error = 16.94%
## Searching left ...
## mtry = 2     OOB error = 17.73%
## -0.04651163 1e-04
## Searching right ...
## mtry = 6     OOB error = 16.82%
## 0.006976744 1e-04
## mtry = 12    OOB error = 16.63%
## 0.0117096 1e-04
## mtry = 15    OOB error = 16.75%
## -0.007109005 1e-04
```



```
# tuned model shows the best value at 12, OOB error rate - 16.59%

# REFINED MODEL: 2

RF.m1 = randomForest(TenYearCHD~.,data = balanced.train, ntree = 1700, mtry = 12, nodesize = 10, importa

print(RF.m1)
```

```
##
## Call:
##  randomForest(formula = TenYearCHD ~ ., data = balanced.train,      ntree = 1700, mtry = 12, nodesiz
##                Type of random forest: classification
##                      Number of trees: 1700
## No. of variables tried at each split: 12
##
##         OOB estimate of  error rate: 16.71%
## Confusion matrix:
##      0   1 class.error
## 0 1595  97  0.05732861
## 1  327 519  0.38652482
```

```
# Confusion matrix:
#      0   1 class.error
# 0 1596  96  0.05673759
# 1  324 522  0.38297872

# overall -> 83.45%
# spec -> 94.33%
# sens -> 61.70%

importance(RF.m1)
```

```
##                         0           1 MeanDecreaseAccuracy
## male             28.92889   1.7241395             28.57315
## age             123.53177  75.6453917            148.41907
## education        41.11019   1.8224797             37.58902
## currentSmoker    97.02103 -99.2640007             85.04859
## cigsPerDay      158.76259  10.8188352            160.49151
## BPMeds           61.11769   0.8321436             58.79695
## prevalentStroke  21.75942  21.2515871             30.29471
## prevalentHyp     36.72438 -25.8615770             35.52930
## diabetes         90.62315  31.1861887             89.42834
## totChol          60.62424  12.7192657             59.78540
## sysBP            83.84634  32.8960302             97.42501
## diaBP            70.89521  24.0066777             83.49570
## BMI              70.77156  24.3861981             73.48797
## heartRate        57.86200  14.9250877             56.41450
## glucose          67.91891  23.9712715             69.24409
##                 MeanDecreaseGini
## male                    8.132826
## age                   157.975515
## education              23.487557
## currentSmoker          63.133566
## cigsPerDay            113.485575
## BPMeds                  16.723957
## prevalentStroke          6.078960
## prevalentHyp             9.523759
## diabetes                37.143289
## totChol                 68.485520
## sysBP                  109.804005
## diaBP                   81.542630
## BMI                     82.736909
## heartRate               60.592978
## glucose                 69.418448
```

```
balanced.train$RF.Pred = predict(RF.m1, data = balanced.train, type = "class")
balanced.train$RF.Score = 1-predict(RF.m1, data = balanced.train, type = "prob")[,2]
sm.test$RF.Pred = predict(RF.m1, newdata = sm.test, type = "class")
sm.test$RF.Score = 1-predict(RF.m1, newdata = sm.test, type = "prob")[,2]

t_devRF <- with(balanced.train,table(TenYearCHD,RF.Pred))
t_devRF
```

```
##           RF.Pred
## TenYearCHD    0    1
##          0 1595   97
##          1  327  519
```

```
#           RF.Pred
# TenYearCHD    0    1
#         0   1597   95
#         1    329   517

# RF has made a good prediction - 61.11% CORRECT PREDICTION OF RISK BEARERS
# Spec - 94.38%
# Overall Acc - 83.29%

# MODEL PERFORMANCE MEASURES:
library(InformationValue)
```

```
## Warning: package 'InformationValue' was built under R version 3.6.1
```

```
##
## Attaching package: 'InformationValue'
```

```
## The following objects are masked from 'package:caret':
##
##     confusionMatrix, precision, sensitivity, specificity
```

```
library(ROCR)
```

```
## Warning: package 'ROCR' was built under R version 3.6.1
```

```
## Loading required package: gplots
```

```
## Warning: package 'gplots' was built under R version 3.6.1
```

```
##
## Attaching package: 'gplots'
```

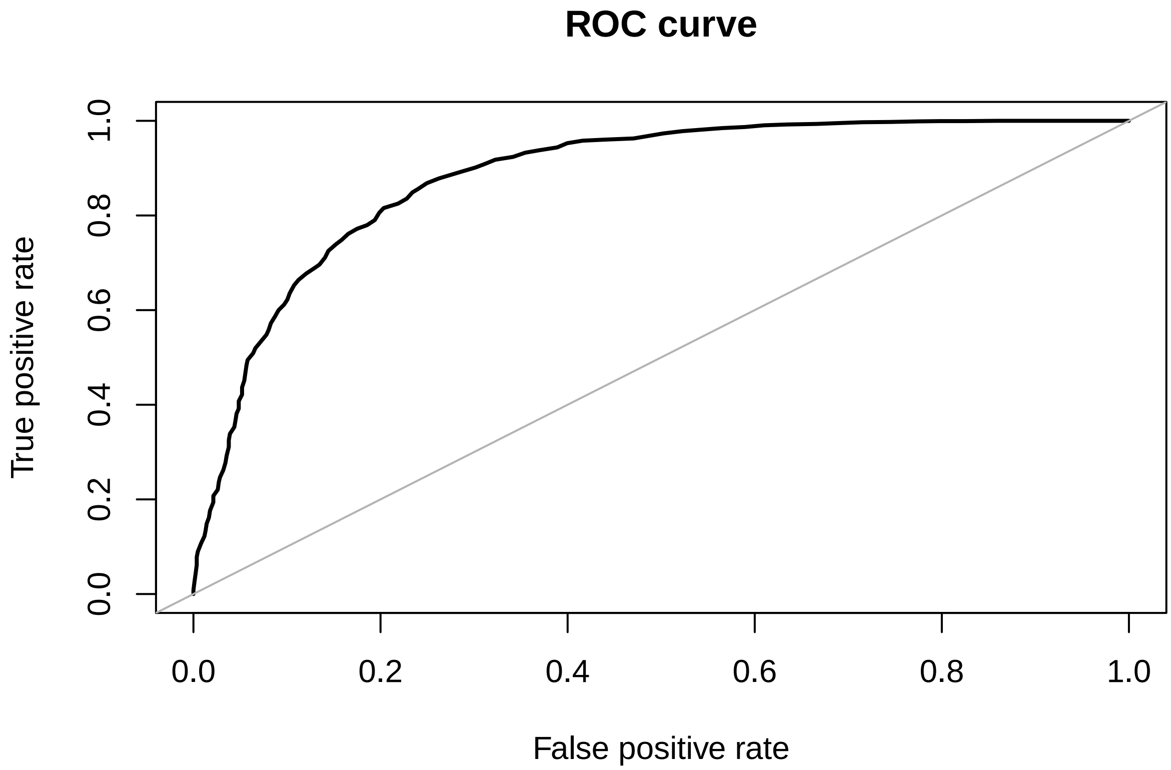
```
## The following object is masked from 'package:stats':
##
##     lowess
```

```
library(ineq)
library(ROSE)
```

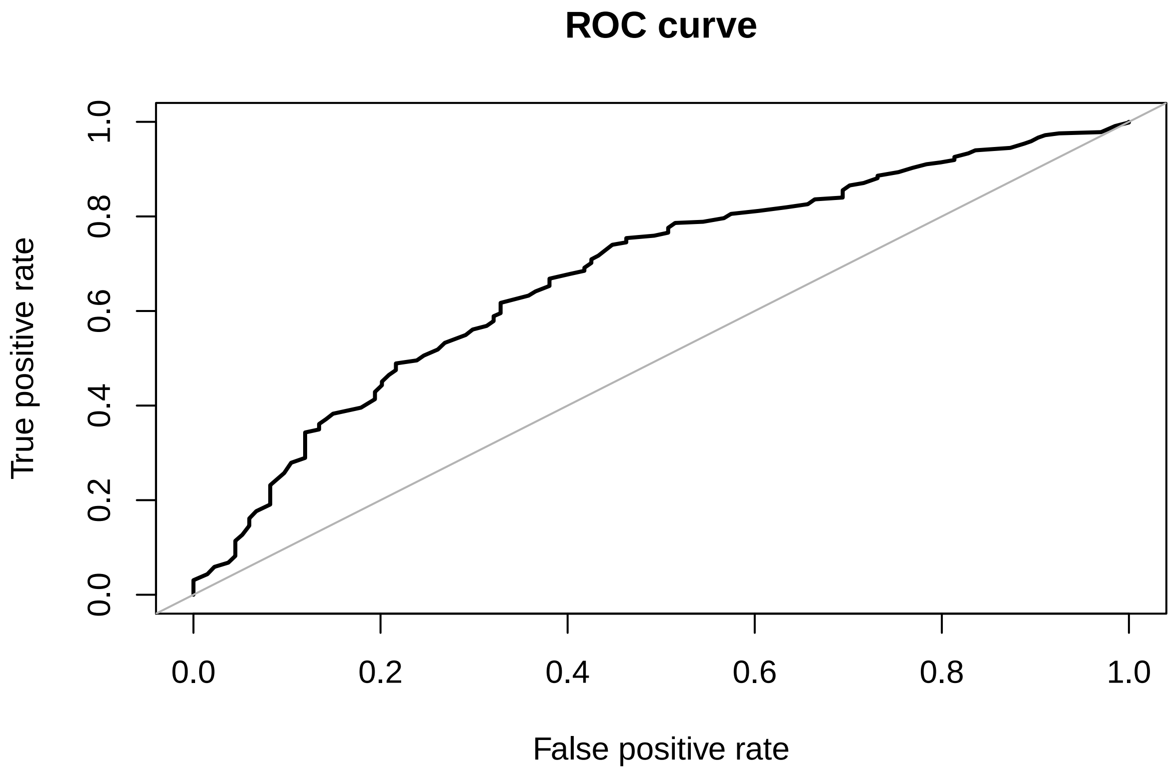
```
## Warning: package 'ROSE' was built under R version 3.6.1
```

```
## Loaded ROSE 0.0-3
```

```
# RF:

#AUC

auc.train = roc.curve(balanced.train$TenYearCHD, balanced.train$RF.Score)
```



```
auc.test = roc.curve(sm.test$TenYearCHD, sm.test$RF.Score)
```



```
# The AUC is a little over 60% as we can see, and is more than the random prediction.

ineq(sm.test$RF.Score,"gini")
```

```
## [1] 0.150951
```

```
# lower gini of 15.01% indicated higher equality or lower inequality in distribution of the risk factor
```

Boosting

XGBoosting is tried here with only our numeric variables of the NA removed dataset.

```
library(xgboost)
```

```
## Warning: package 'xgboost' was built under R version 3.6.1
```

```
# XGBoost works with matrices that contain all numeric variables
# we also need to split the training data and label

gd_features_train<-as.matrix(balanced.train[, c(2,5,10:15)])
gd_label_train<-as.matrix(balanced.train[,16])
gd_features_test<-as.matrix(sm.test[,c(2,5,10:15)])

xgb.fit <- xgboost(
  data = gd_features_train,
  label = gd_label_train,
  eta = 0.001,#this is like shrinkage in the previous algorithm
  max_depth = 3,#Larger the depth, more complex the model; higher chances of overfitting. There is no s
  min_child_weight = 3,#it blocks the potential feature interactions to prevent overfitting
  nrounds = 1000,#controls the maximum number of iterations. For classification, it is similar to the n
  nfold = 5,
  objective = "binary:logistic",  # for regression models
  verbose = 0,                # silent,
  early_stopping_rounds = 10 # stop if no improvement for 10 consecutive trees
)

#gd_features_test<-as.matrix(gd_features_test[,1:ncol(gd_features_test)-1])

sm.test$xgb.pred.class <- predict(xgb.fit, gd_features_test)

table(sm.test$TenYearCHD,sm.test$xgb.pred.class>0.3)
```

```
##
##     TRUE
##   0  781
##   1  134
```

```
#this model was definitely better
#or simply the total correct of the minority class

sum(sm.test$TenYearCHD==1 & sm.test$xgb.pred.class>=0.3)
```

```
## [1] 134
```

```
#     TRUE
#  0  781
#  1  134
```

```
#in this code chunk we will playing around with all the values untill we find the best fit
#let's play with shrinkage, known as eta in xbg

tp_xgb<-vector()
lr <- c(0.001, 0.01, 0.1, 0.3, 0.5, 0.7, 1)
md<-c(1,3,5,7,9,15)
nr<-c(2, 50, 100, 1000, 10000)
for (i in md) {

  xgb.fit1 <- xgboost(
    data = gd_features_train,
    label = gd_label_train,
    eta = 0.2,
    max_depth = 15,
    nrounds = 10,
    nfold = 5,
    objective = "binary:logistic",  # for regression models
    verbose = 1,                # silent,
    early_stopping_rounds = 14 # stop if no improvement for 10 consecutive trees
  )

  sm.test$xgb.pred.class <- predict(xgb.fit1, gd_features_test)

  tp_xgb<-cbind(tp_xgb,sum(sm.test$TenYearCHD==1 & sm.test$xgb.pred.class>=0.3))
  #if your class=1 and our prediction=0.2, we are going to display it with the next line compare the sa

}
```

```
## [1]  train-error:0.098897
## Will train until train_error hasn't improved in 14 rounds.
##
## [2]  train-error:0.085894
## [3]  train-error:0.069740
## [4]  train-error:0.057526
## [5]  train-error:0.047281
## [6]  train-error:0.038219
## [7]  train-error:0.034673
## [8]  train-error:0.030733
## [9]  train-error:0.024823
## [10] train-error:0.020883
## [1]  train-error:0.098897
## Will train until train_error hasn't improved in 14 rounds.
##
## [2]  train-error:0.085894
## [3]  train-error:0.069740
## [4]  train-error:0.057526
## [5]  train-error:0.047281
## [6]  train-error:0.038219
## [7]  train-error:0.034673
## [8]  train-error:0.030733
## [9]  train-error:0.024823
## [10] train-error:0.020883
## [1]  train-error:0.098897
## Will train until train_error hasn't improved in 14 rounds.
##
## [2]  train-error:0.085894
## [3]  train-error:0.069740
## [4]  train-error:0.057526
## [5]  train-error:0.047281
## [6]  train-error:0.038219
## [7]  train-error:0.034673
## [8]  train-error:0.030733
## [9]  train-error:0.024823
## [10] train-error:0.020883
## [1]  train-error:0.098897
## Will train until train_error hasn't improved in 14 rounds.
##
## [2]  train-error:0.085894
## [3]  train-error:0.069740
## [4]  train-error:0.057526
## [5]  train-error:0.047281
## [6]  train-error:0.038219
## [7]  train-error:0.034673
## [8]  train-error:0.030733
## [9]  train-error:0.024823
## [10] train-error:0.020883
## [1]  train-error:0.098897
## Will train until train_error hasn't improved in 14 rounds.
```

```
##
## [2]  train-error:0.085894
## [3]  train-error:0.069740
## [4]  train-error:0.057526
## [5]  train-error:0.047281
## [6]  train-error:0.038219
## [7]  train-error:0.034673
## [8]  train-error:0.030733
## [9]  train-error:0.024823
## [10] train-error:0.020883
## [1]  train-error:0.098897
## Will train until train_error hasn't improved in 14 rounds.
##
## [2]  train-error:0.085894
## [3]  train-error:0.069740
## [4]  train-error:0.057526
## [5]  train-error:0.047281
## [6]  train-error:0.038219
## [7]  train-error:0.034673
## [8]  train-error:0.030733
## [9]  train-error:0.024823
## [10] train-error:0.020883
```

```
tp_xgb
```

```
##      [,1] [,2] [,3] [,4] [,5] [,6]
## [1,]   76   76   76   76   76   76
```

```
table(sm.test$TenYearCHD, sm.test$xgb.pred.class>=0.3)
```

```
##
##     FALSE TRUE
##   0   502  279
##   1    58   76
```

```
# here there is significant imporvement of the model compared to our logistic model
# sensitivity is found to be 56.72%, spec 64.27%, overall 63.17% accurate

# wNAtest = wNAtest[, -17]

#    FALSE TRUE
#  0   502  279
#  1    58   76
```