Heart Disease Visualization

Mingwei Wu

Background

The database contains 76 attributes, but all published experiments refer to using a subset of 14 of them, at particular, the Cleveland database is the only one that has been used by ML researchers to this data.

Data Description

age: patients' age

sex: 1 is male, 2 is female

cp: chest pain type 4 level

trestbps: resting blood presure

chol: serum cholestoral in mg/dl

fbs: fasting blood sugar >120 mg/dl is value 1

restecg: resting electrocardiographic reults (values 0,1,2)

thalach: maximum heart rate achieved

exang: exercise incuced angina

oldpeak: ST depression induced by exercise relative to rest

slope: the slope of the peak exercise ST segment

ca: number of major vessels (0-3) colored by flourosopy

thal: 3 is normal; 6 fixed defect; 7 reversable defect

target: 1 has heart disease, 0 not

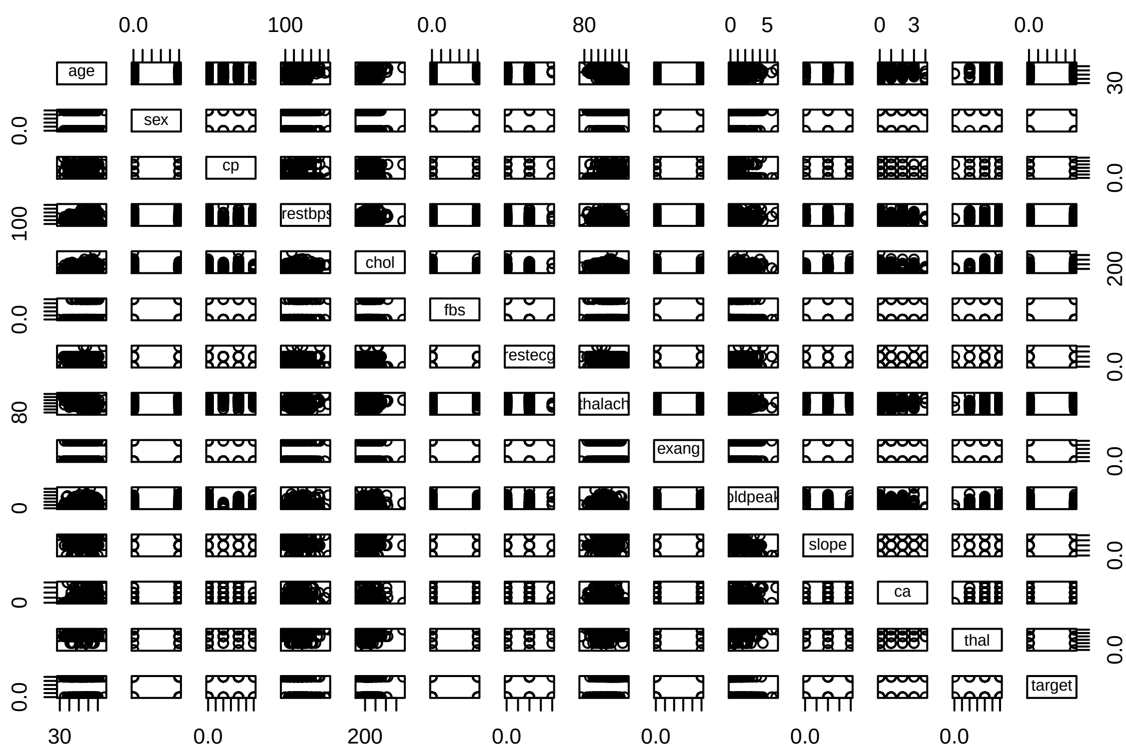
Import data

```
data<-read.csv("heart.csv",header = TRUE)
head(data)
```

```
##   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
```

First step, we compare the relationship on every column. Double check the dependent value.

```
pairs(data) #pairs data to see the relationship in numeric values
```



Edit the column with categories. Change the int value to character values.

```r
library(ggplot2)
library(tidyverse)
```

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.0 --
```

```
## v tibble  3.0.4     v dplyr   1.0.2
## v tidyr   1.1.2     v stringr 1.4.0
## v readr   1.4.0     v forcats 0.5.0
## v purrr   0.3.4
```

```
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```r
library(dplyr)
heart<-data%>%
  mutate(gender= ifelse(sex==1,"Male","Female"),
         chest_pain_level= ifelse(cp==0,"normal",
                                  ifelse(cp==1,"mild",
                                         ifelse(cp==2,"moderate","severe"))),
         fblood_sugar=ifelse(fbs==1,">120","<=120"),
         rest_electrocardigoraphic= ifelse(restecg==0,"normal",
                                           ifelse(restecg==1,"abnormalily","definite")),
         exercise=ifelse(exang==1,"yes","no"),
        heart_condition=ifelse(target==1,"yes","no")) # rebuild the column to the data frame
```

Geder Analysis

Calculate the rate of heart disease in th gender. The attached result below, the rate of Female in database has 75% who had heart disease, and the Male rate had approximate 45%

```r
heart%>%
  group_by(gender)%>%
  summarise(gender_rate=mean(target)) # calculate the heart disease  rate of the gender
```

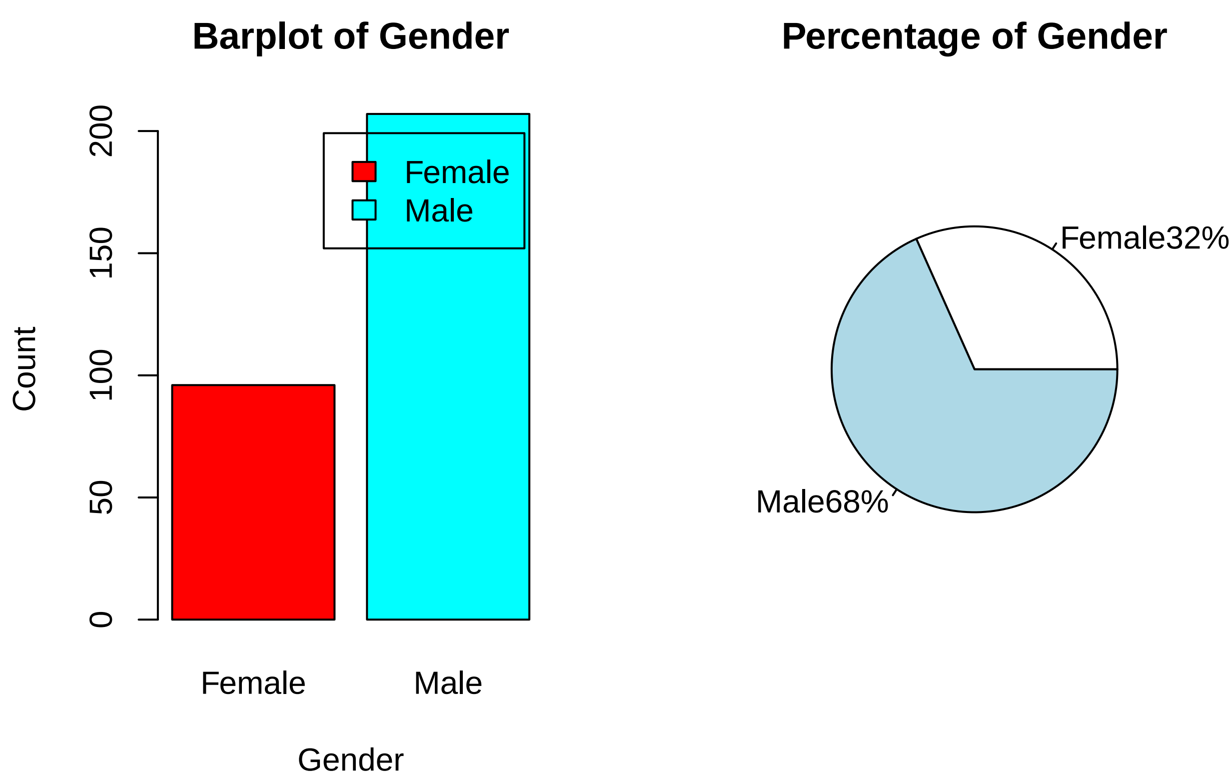
```
## `summarise()` ungrouping output (override with `.groups` argument)
```

```
## # A tibble: 2 x 2
##   gender gender_rate
##   <chr>        <dbl>
## 1 Female       0.75
## 2 Male         0.449
```

Question: why female has higher proportion in heart disease.

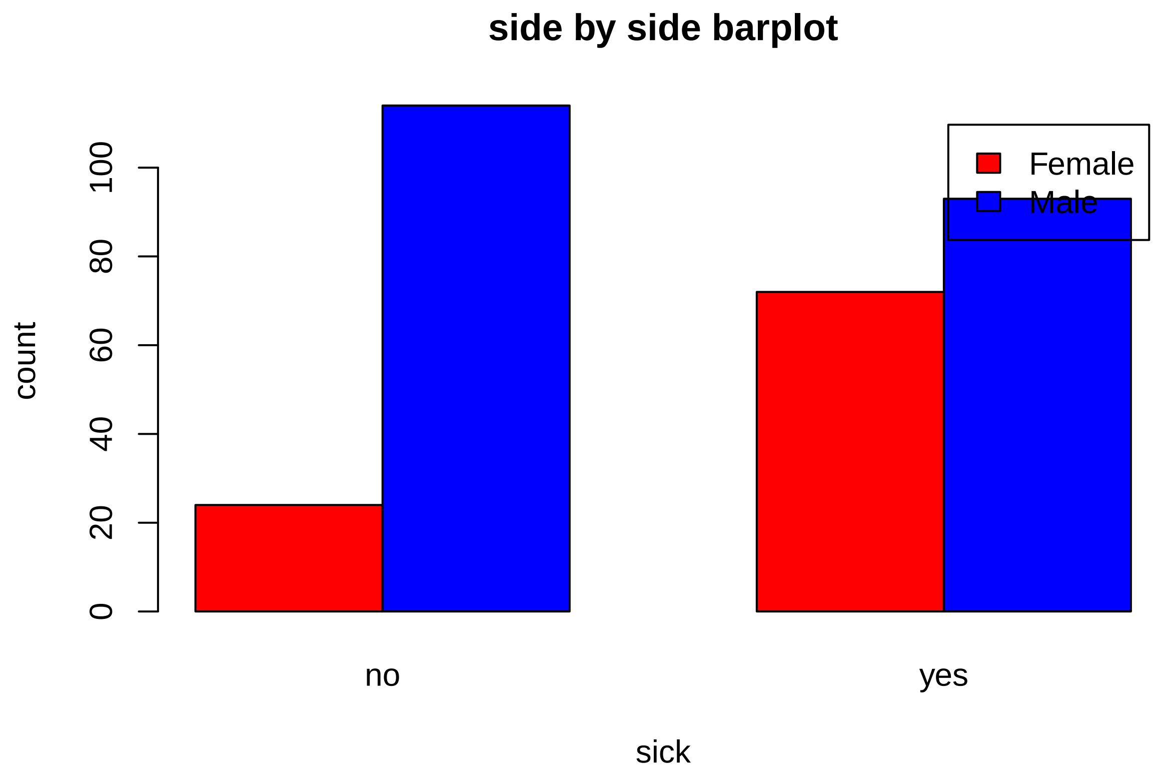
Count the quantity of gender. In the dataframe, the male quantity is domain, almost double quantity than female.

```r
a<-table(heart$gender)
par(mfrow=c(1,2))
barplot(a,
        col=rainbow(2),
        legend=rownames(a),
        xlab="Gender",
        ylab="Count",
        main="Barplot of Gender")
pct<-round(a/sum(a)*100)
lbs<-paste(c("Female","Male"), "",pct,"%",sep="")
pie(a,labels=lbs, main="Percentage of Gender") #data visualization to compare quantity of gender
```



Barplot displays the disease quantity of gender that are almost same. on the contract, the Male count higher than female.

```r
b<-table(heart$gender,heart$heart_condition)
barplot(b,
        col = c("red","blue"),
        legend=rownames(b),
        beside=TRUE,
        xlab="sick",
        ylab="count",
        main="side by side barplot") # data visualization to compare count with disease.
```



Exercise Analysis

The rate of exercise with disease is 23%, and the rate of no_exercise with disease approximate 70%

```
heart%>%
  group_by(exercise)%>%
  summarise(exercise_rate=mean(target))
```

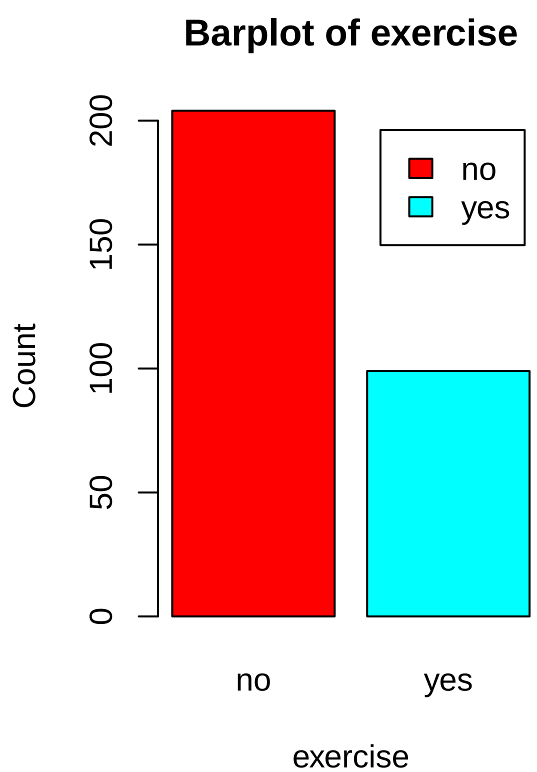
```
## `summarise()` ungrouping output (override with `.groups` argument)
```

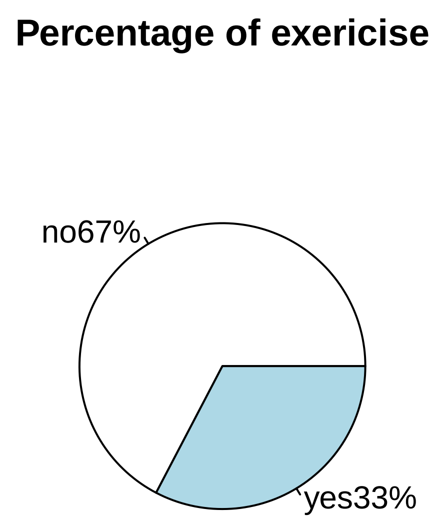
```
## # A tibble: 2 x 2
##   exercise exercise_rate
##   <chr>            <dbl>
## 1 no               0.696
## 2 yes              0.232
```

Only 33% in the dataframe who do exercise everyday.

```
a1<-table(heart$exercise)
par(mfrow=c(1,2))
```

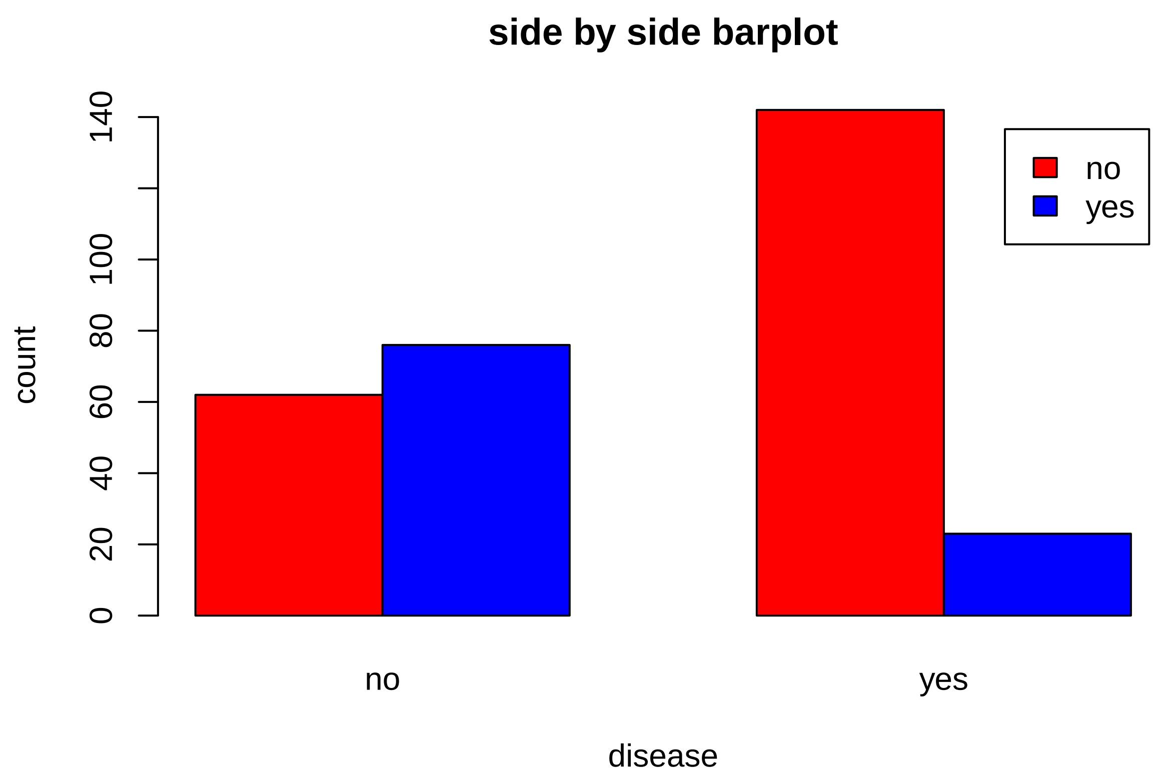
```r
barplot(a1,
        col=rainbow(2),
        legend=rownames(a1),
        xlab="exercise",
        ylab="Count",
        main="Barplot of exercise")
pct<-round(a1/sum(a1)*100)
lbs<-paste(c("no","yes"), "",pct,"%",sep="")
pie(a1,labels=lbs, main="Percentage of exericise") #data visualization to compare quantity of gender
```





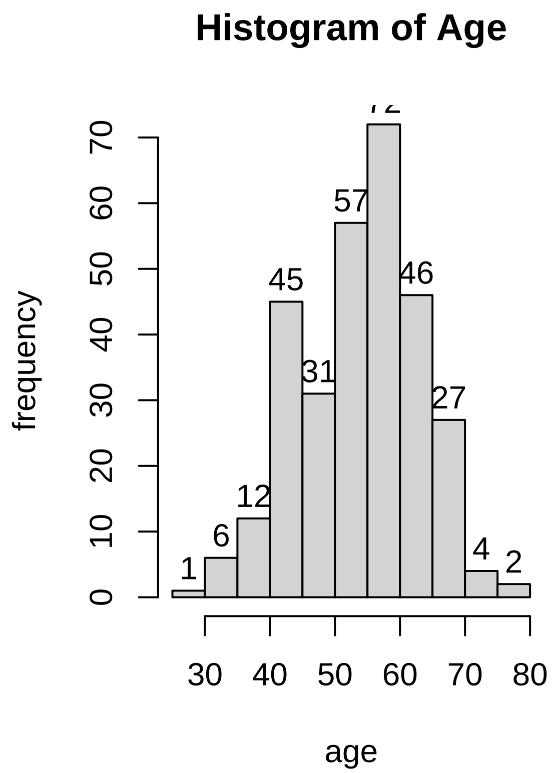
The proportion of people with no disease who do not exercise are almost same, but the proportion of disease, no-exercise is extreme higher than do-exercise.

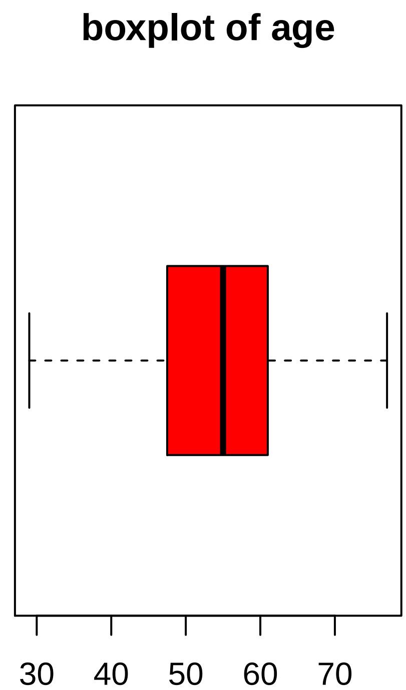
```r
b1<-table(heart$exercise,heart$heart_condition)
barplot(b1,
        col = c("red","blue"),
        legend=rownames(b1),
        beside=TRUE,
        xlab="disease",
        ylab="count",
        main="side by side barplot") # data visualization to compare count with disease.
```



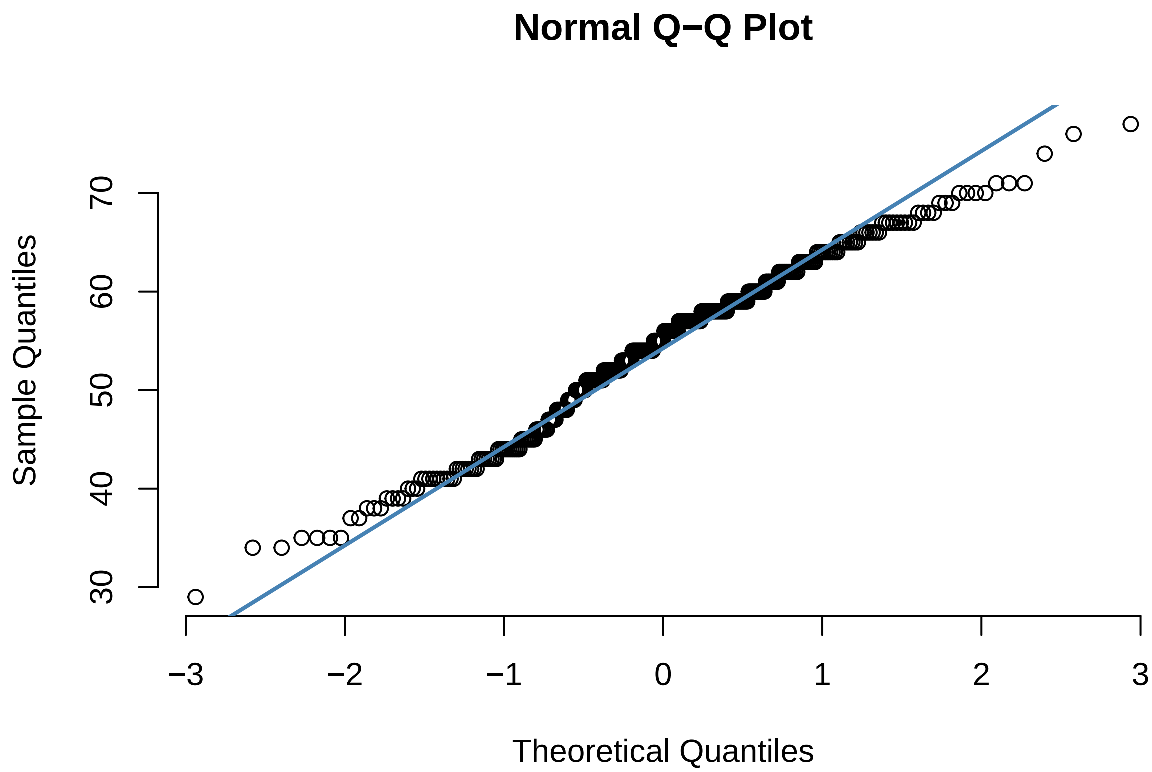
Age Analysis ### the age histogram shows the normal distribution, and estimate the high proportion of disease of age range 50-60.

```
par(mfrow=c(1,2))
hist(heart$age,labels=TRUE,main="Histogram of Age",xlab = "age",ylab = "frequency")
boxplot(heart$age,horizontal = TRUE,col="red",main="boxplot of age")
```






```r
qqnorm(heart$age,frame=FALSE)
qqline(heart$age,col="steelblue",lwd=2)
```



Fasting Blood Sugar Analysis

The fasting blood sugar either less than 120 or greater than 120, it is not significat effect to the disease.

```
heart%>%
  group_by(fblood_sugar)%>%
  summarise(fblood_sugar_rate=mean(target))
```

```
## `summarise()` ungrouping output (override with `.groups` argument)
```

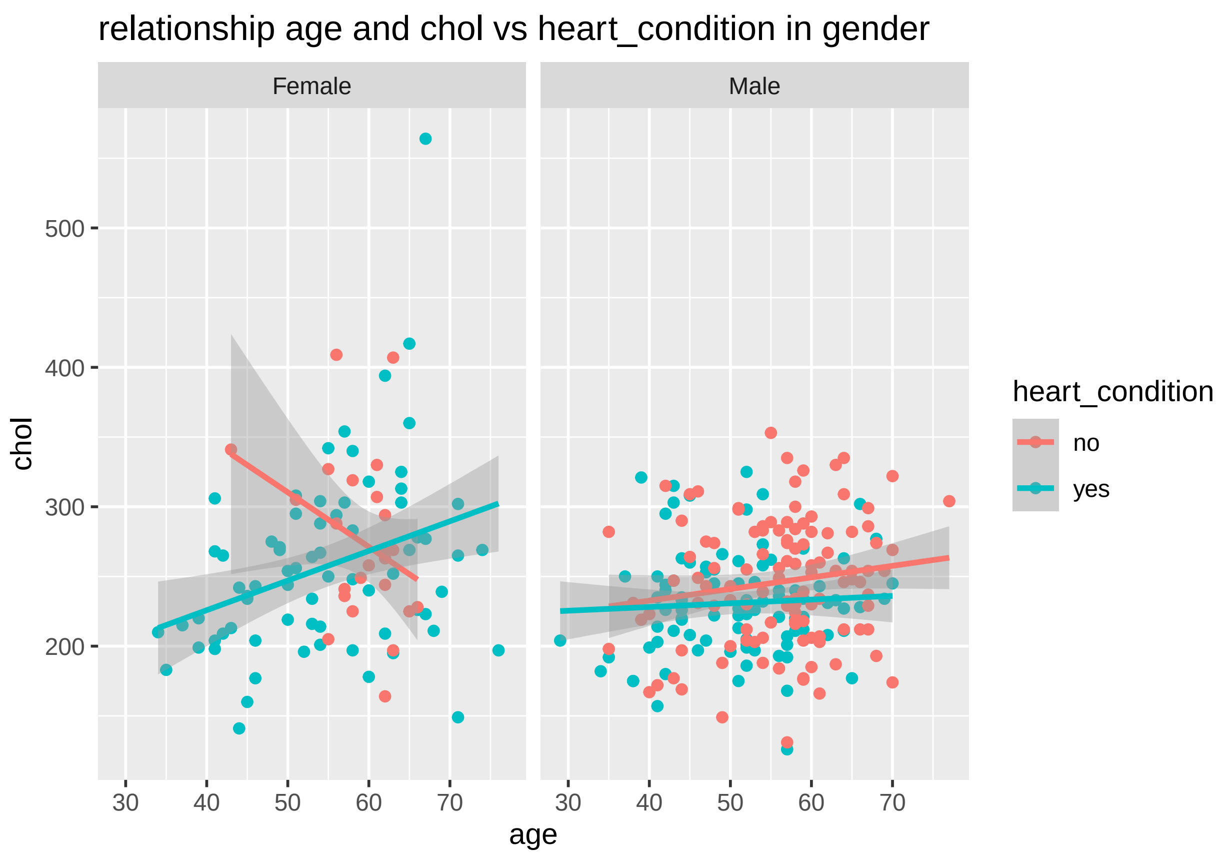
```
## # A tibble: 2 x 2
##   fblood_sugar fblood_sugar_rate
##   <chr>                    <dbl>
## 1 <=120                    0.550
## 2 >120                     0.511
```

```
heart%>%
  ggplot(aes(heart_condition,fill=fblood_sugar))+geom_bar()+ggtitle("blood sugar > 120 vs heart conditi
```

Cholestoral Analysis ### Plot the dependent value of chol and age to disease. For the Female, age and chol with disease display increased trend line, but non-disease is decreased. However, for the male, either disease or not, both trends increased line.

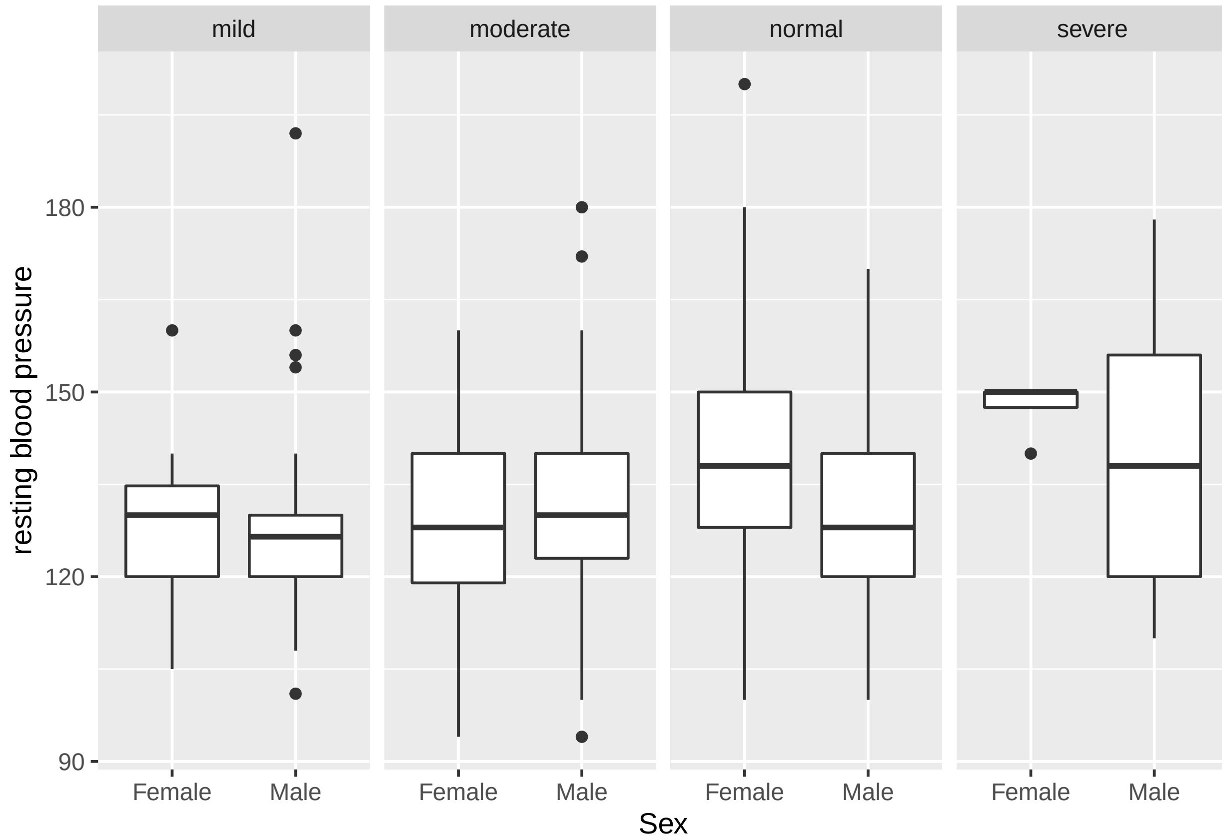
```
heart%>%
  ggplot(aes(x=age,y=chol,color=heart_condition))+geom_point()+geom_smooth(method="lm")+ggtitle("relati
```

```
## `geom_smooth()` using formula 'y ~ x'
```



Resting electrocardiographic Analysis

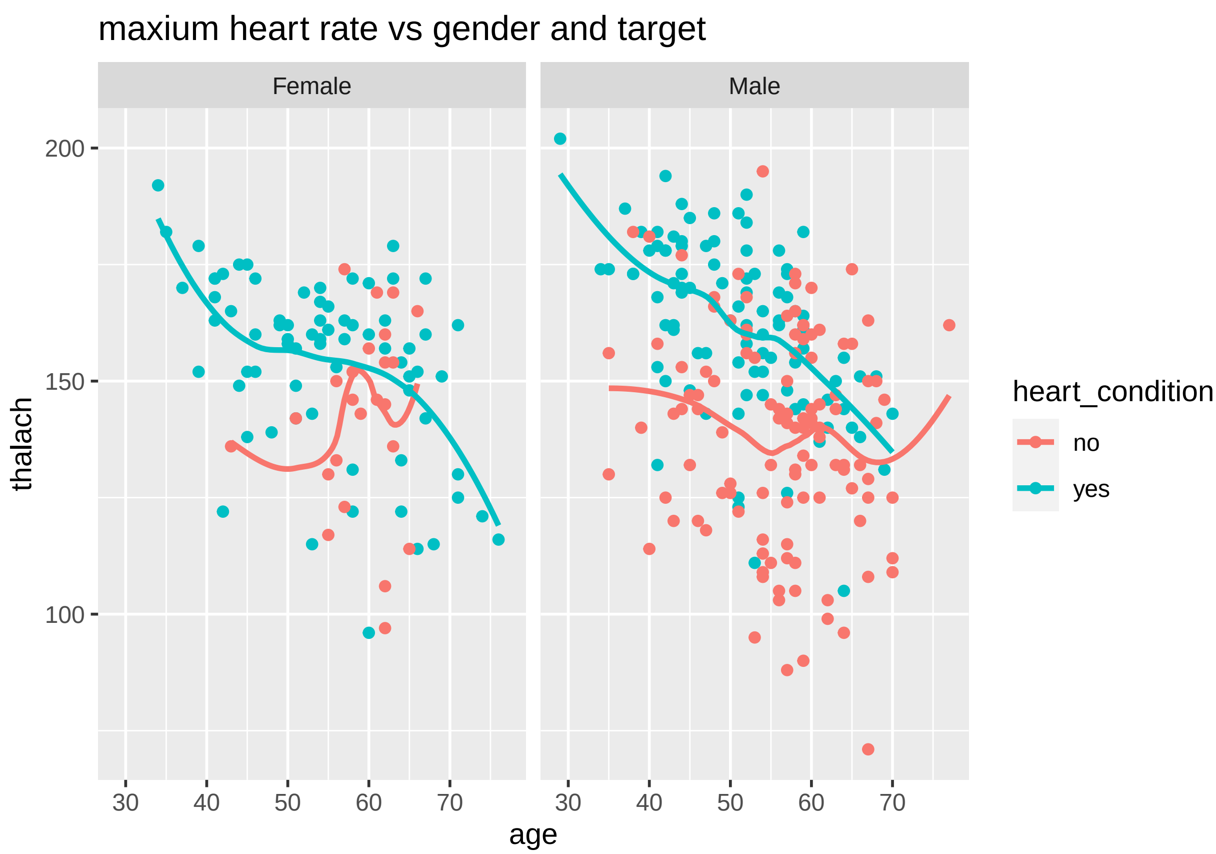
```
heart%>%
  ggplot(aes(gender,trestbps))+geom_boxplot()+xlab("Sex")+ylab("resting blood pressure")+facet_grid(~ch
```



Maximum Heart Rate Achieved Analysis ### No matter with gender, if people have disease, the maximum heart rate achieved is higher than non-disease, also decresed with age. the non-disease heart rate achieved is relatively stable.

```
heart%>%
  ggplot(aes(age,thalach,color=heart_condition))+geom_point()+geom_smooth(se=FALSE)+facet_grid(~gender)
```

```
## `geom_smooth()` using method = 'loess' and formula 'y ~ x'
```



Principle Component Analysis

```
pca<-prcomp(heart[,4:10],scale=TRUE)
pca
```

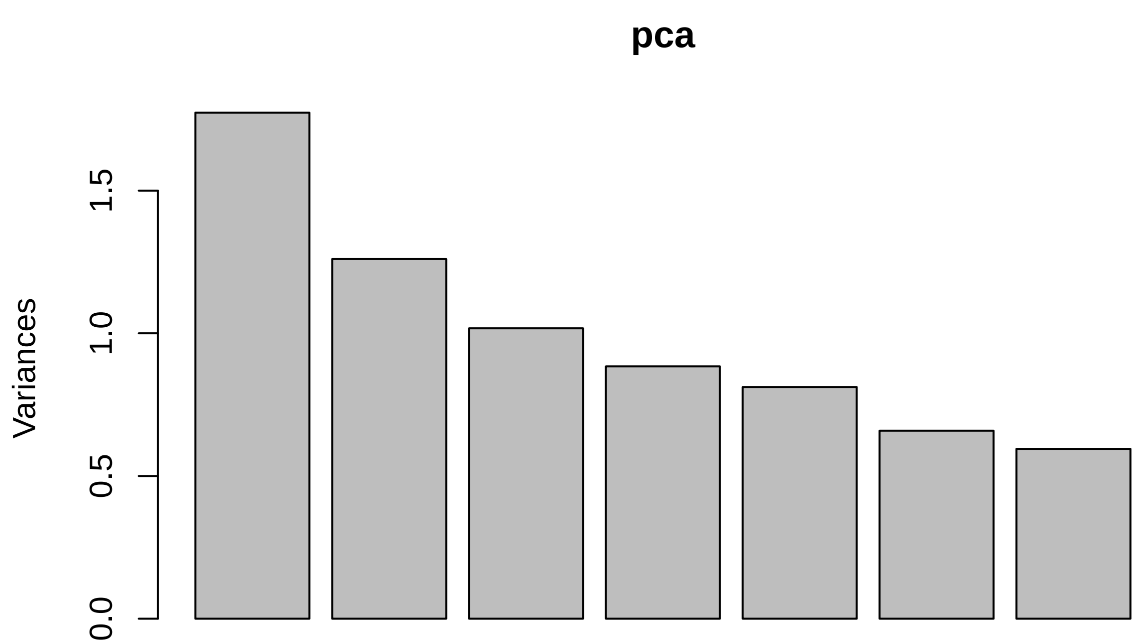
```
## Standard deviations (1, .., p=7):
## [1] 1.3315929 1.1225306 1.0087031 0.9402348 0.9008774 0.8115652 0.7713909
##
## Rotation (n x k) = (7 x 7):
##                 PC1        PC2         PC3        PC4         PC5         PC6
## trestbps -0.2962509  0.4836945 -0.28351741  0.5166899 -0.33164295 -0.41673496
## chol     -0.1787205  0.4122420  0.61954768  0.3047407  0.54268445  0.15616166
## fbs      -0.1208119  0.4598572 -0.63005504 -0.3290019  0.45347280  0.23869262
## restecg   0.2092606 -0.4497390 -0.36233415  0.6286817  0.47640056 -0.02035216
## thalach   0.5229093  0.3295268  0.03266711  0.1562670 -0.09496291 -0.13444831
## exang    -0.5180609 -0.2336930  0.04786729 -0.1924133  0.27701992 -0.61645776
## oldpeak  -0.5292331 -0.1384268 -0.06453488  0.2742982 -0.27776514  0.58860905
##                   PC7
## trestbps  0.217415421
## chol      0.050163486
## fbs      -0.078182227
## restecg   0.006815801
## thalach  -0.751928923
```

```
## exang    -0.425354114
## oldpeak  -0.444670682
```

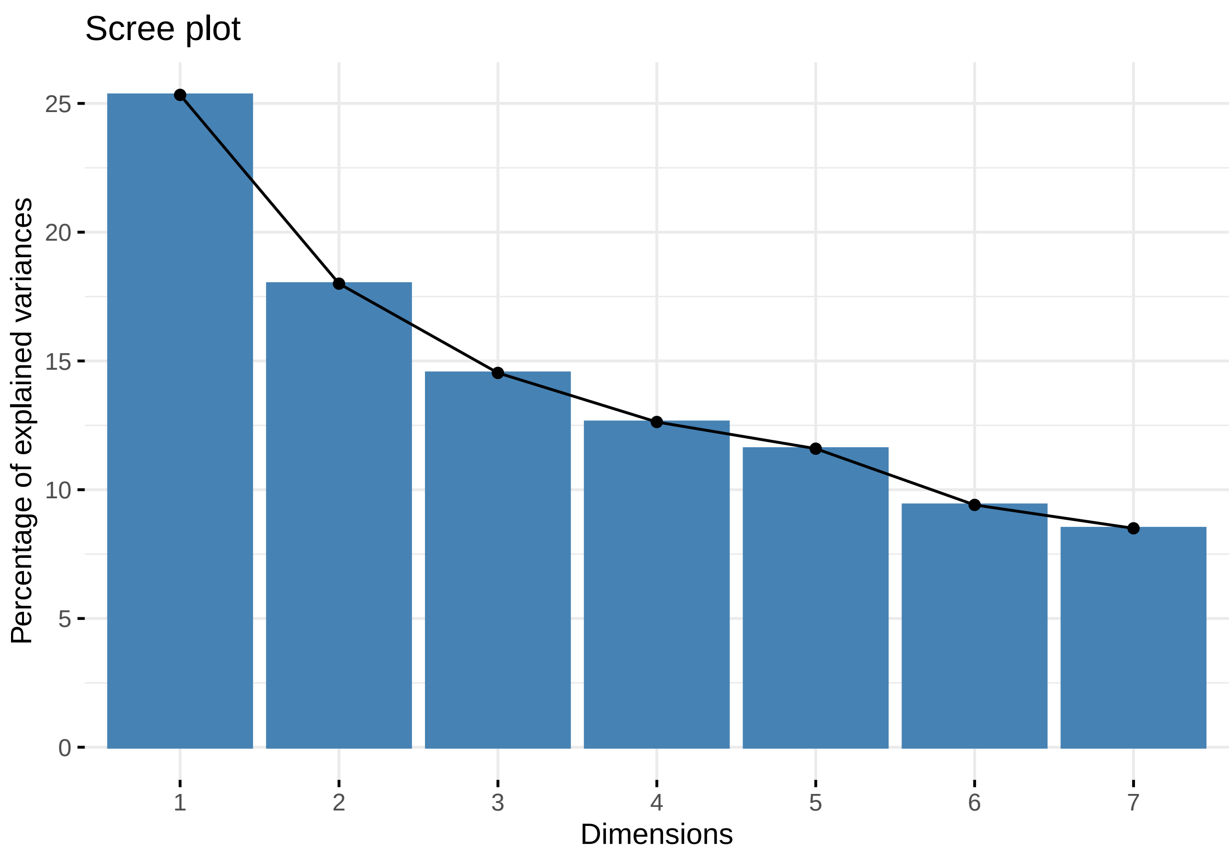
```r
library(factoextra)
```

```
## Welcome! Want to learn more? See two factoextra-related books at https://goo.gl/ve3WBa
```

```r
screeplot(pca)
```




```r
fviz_screeplot(pca) #plot the tendency of principle component analysis
```



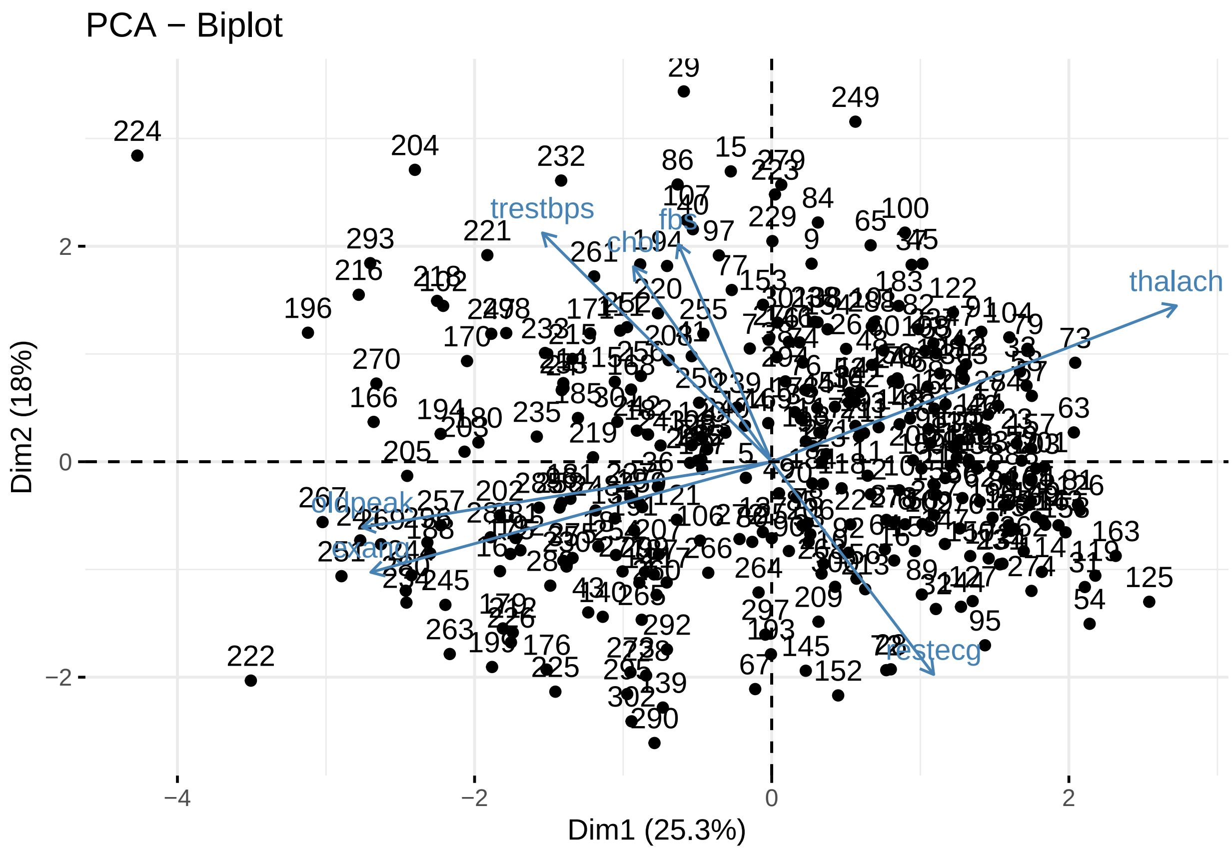

```
pca$sdev^2
```

```
## [1] 1.7731395 1.2600750 1.0174820 0.8840414 0.8115801 0.6586380 0.5950439
```

```
pca$rotation
```

```
##                 PC1        PC2         PC3        PC4         PC5         PC6
## trestbps -0.2962509  0.4836945 -0.28351741  0.5166899 -0.33164295 -0.41673496
## chol     -0.1787205  0.4122420  0.61954768  0.3047407  0.54268445  0.15616166
## fbs      -0.1208119  0.4598572 -0.63005504 -0.3290019  0.45347280  0.23869262
## restecg   0.2092606 -0.4497390 -0.36233415  0.6286817  0.47640056 -0.02035216
## thalach   0.5229093  0.3295268  0.03266711  0.1562670 -0.09496291 -0.13444831
## exang    -0.5180609 -0.2336930  0.04786729 -0.1924133  0.27701992 -0.61645776
## oldpeak  -0.5292331 -0.1384268 -0.06453488  0.2742982 -0.27776514  0.58860905
##                   PC7
## trestbps  0.217415421
## chol      0.050163486
## fbs      -0.078182227
## restecg   0.006815801
## thalach  -0.751928923
## exang    -0.425354114
## oldpeak  -0.444670682
```

```r
fviz_pca(pca) #data visualization for pca
```



Statistics Analysis

Randomly split the data into 70% train_set and 30% test_set for logistic regression.

```r
library(caTools)
set.seed(927)
sample<-sample.split(data$target,SplitRatio=0.70)
train_set<-subset(data,sample==TRUE)
test_set<-subset(data,sample==FALSE)
head(heart,10)
```

```
##    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
## 7   56   0  1      140  294   0       0     153     0     1.3     1  0    2
## 8   44   1  1      120  263   0       1     173     0     0.0     2  0    3
## 9   52   1  2      172  199   1       1     162     0     0.5     2  0    3
```

```
## 10  57   1  2      150  168   0       1     174     0     1.6     2  0    2
##    target gender chest_pain_level fblood_sugar rest_electrocardigoraphic
## 1       1   Male           severe         >120                    normal
## 2       1   Male         moderate        <=120               abnormalily
## 3       1 Female             mild        <=120                    normal
## 4       1   Male             mild        <=120               abnormalily
## 5       1 Female           normal        <=120               abnormalily
## 6       1   Male           normal        <=120               abnormalily
## 7       1 Female             mild        <=120                    normal
## 8       1   Male             mild        <=120               abnormalily
## 9       1   Male         moderate         >120               abnormalily
## 10      1   Male         moderate        <=120               abnormalily
##    exercise heart_condition
## 1        no             yes
## 2        no             yes
## 3        no             yes
## 4        no             yes
## 5       yes             yes
## 6        no             yes
## 7        no             yes
## 8        no             yes
## 9        no             yes
## 10       no             yes
```

```r
logistic<-glm(target~.,train_set,
              family=binomial())
summary(logistic)
```

```
##
## Call:
## glm(formula = target ~ ., family = binomial(), data = train_set)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.4446  -0.3966   0.1437   0.5971   2.5361
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)  1.963599   2.911957   0.674  0.50011
## age          0.003474   0.027706   0.125  0.90023
## sex         -1.744546   0.563229  -3.097  0.00195 **
## cp           0.982057   0.229104   4.287 1.82e-05 ***
## trestbps    -0.014493   0.012554  -1.154  0.24830
## chol        -0.002281   0.004659  -0.490  0.62444
## fbs         -0.124715   0.667916  -0.187  0.85188
## restecg      0.574945   0.420552   1.367  0.17159
## thalach      0.021966   0.012269   1.790  0.07340 .
## exang       -0.869894   0.472047  -1.843  0.06536 .
## oldpeak     -0.612452   0.259405  -2.361  0.01823 *
## slope        0.557911   0.455675   1.224  0.22082
## ca          -0.852029   0.235122  -3.624  0.00029 ***
## thal        -0.918692   0.356196  -2.579  0.00990 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 292.36  on 211  degrees of freedom
## Residual deviance: 150.91  on 198  degrees of freedom
## AIC: 178.91
##
## Number of Fisher Scoring iterations: 6
```

Remove insignificant factors

After removing the insignificant factors, the AIC value is going down.

```
logistic1<-glm(target~sex+cp+thalach+oldpeak+ca+thal,
               train_set,
               family=binomial())
summary(logistic1)
```

```
##
## Call:
## glm(formula = target ~ sex + cp + thalach + oldpeak + ca + thal,
##     family = binomial(), data = train_set)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.3901  -0.4855   0.1999   0.5545   2.4572
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept) -0.06097    1.73144  -0.035 0.971910
## sex         -1.49045    0.47503  -3.138 0.001703 **
## cp           0.99727    0.20969   4.756 1.98e-06 ***
## thalach      0.02621    0.01024   2.559 0.010483 *
## oldpeak     -0.83470    0.21597  -3.865 0.000111 ***
## ca          -0.75831    0.21032  -3.605 0.000312 ***
## thal        -0.97051    0.32970  -2.944 0.003244 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 292.36  on 211  degrees of freedom
## Residual deviance: 161.50  on 205  degrees of freedom
## AIC: 175.5
##
## Number of Fisher Scoring iterations: 5
```

coefficients build up formular

```
logistic1$coefficients
```

```
## (Intercept)         sex          cp     thalach     oldpeak          ca
## -0.06096859 -1.49044986  0.99726987  0.02620721 -0.83469979 -0.75830641
##        thal
## -0.97050613
```

prediction on test_set

```
pred<-predict(logistic1,test_set,type="response")
pred_new<-as.data.frame(pred)
categorise<-function(x){
  return(ifelse(x>0.5,1,0))
}
pred_new<-apply(pred_new,2,categorise)
head(pred_new,10)
```

```
##    pred
## 2     1
## 4     1
## 6     1
## 12    1
## 13    1
## 15    1
## 20    1
## 32    0
## 34    1
## 38    1
```

Model Evaluation

Model has 85.7% accuracy in predecting future data with logistic regression model.

```
library(caret)
```

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

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

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

```
confusionMatrix(as.factor(test_set$target),as.factor(pred_new))
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction  0  1
##          0 32  9
##          1  4 46
##
##                Accuracy : 0.8571
##                  95% CI : (0.7681, 0.9217)
##     No Information Rate : 0.6044
##     P-Value [Acc > NIR] : 1.294e-07
##
##                   Kappa : 0.7083
##
##  Mcnemar's Test P-Value : 0.2673
##
##             Sensitivity : 0.8889
##             Specificity : 0.8364
##          Pos Pred Value : 0.7805
##          Neg Pred Value : 0.9200
##              Prevalence : 0.3956
##          Detection Rate : 0.3516
##    Detection Prevalence : 0.4505
##       Balanced Accuracy : 0.8626
##
##        'Positive' Class : 0
##
```