Team 1:

Kaiyu Wang,Chinar,Urvashi,Chun,,

I.Setup

```
#install.packages('readr', dependencies = TRUE, repos='http://cran.rstudio.com/')
library(readr)
library(data.table)
library(ggplot2)
library(dplyr)
```

```
##
## Attaching package: 'dplyr'
```

```
## The following objects are masked from 'package:data.table':
##
##     between, first, last
```

```
## The following objects are masked from 'package:stats':
##
##     filter, lag
```

```
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
library(reshape2)
```

```
##
## Attaching package: 'reshape2'
```

```
## The following objects are masked from 'package:data.table':
##
##     dcast, melt
```

```
library(glmnet)
```

```
## Loading required package: Matrix
```

```
## Loaded glmnet 4.1-2
```

```
library(ROCR)
library(pROC)
```

```
## Type 'citation("pROC")' for a citation.
```

```
##
## Attaching package: 'pROC'
```

```
## The following objects are masked from 'package:stats':
##
##     cov, smooth, var
```

```
library(PRROC)
library(lattice)
library(caret)
library(e1071)
library(randomForest) #for random forest
```

```
## randomForest 4.6-14
```

```
## Type rfNews() to see new features/changes/bug fixes.
```

```
##
## Attaching package: 'randomForest'
```

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

```
## The following object is masked from 'package:ggplot2':
##
##     margin
```

```
CHD <-fread("framingham.csv")
```

II.Clean Data

1. Summary

```
summary(CHD)
```

```
##       male             age          education     currentSmoker
##  Min.   :0.0000   Min.   :32.00   Min.   :1.000   Min.   :0.0000
##  1st Qu.:0.0000   1st Qu.:42.00   1st Qu.:1.000   1st Qu.:0.0000
##  Median :0.0000   Median :49.00   Median :2.000   Median :0.0000
##  Mean   :0.4292   Mean   :49.58   Mean   :1.979   Mean   :0.4941
##  3rd Qu.:1.0000   3rd Qu.:56.00   3rd Qu.:3.000   3rd Qu.:1.0000
##  Max.   :1.0000   Max.   :70.00   Max.   :4.000   Max.   :1.0000
##                                   NA's   :105
##    cigsPerDay         BPMeds        prevalentStroke     prevalentHyp
##  Min.   : 0.000   Min.   :0.00000   Min.   :0.000000   Min.   :0.0000
##  1st Qu.: 0.000   1st Qu.:0.00000   1st Qu.:0.000000   1st Qu.:0.0000
##  Median : 0.000   Median :0.00000   Median :0.000000   Median :0.0000
##  Mean   : 9.003   Mean   :0.02963   Mean   :0.005899   Mean   :0.3105
##  3rd Qu.:20.000   3rd Qu.:0.00000   3rd Qu.:0.000000   3rd Qu.:1.0000
##  Max.   :70.000   Max.   :1.00000   Max.   :1.000000   Max.   :1.0000
##  NA's   :29       NA's   :53
##     diabetes          totChol          sysBP           diaBP
##  Min.   :0.00000   Min.   :107.0   Min.   : 83.5   Min.   : 48.00
##  1st Qu.:0.00000   1st Qu.:206.0   1st Qu.:117.0   1st Qu.: 75.00
##  Median :0.00000   Median :234.0   Median :128.0   Median : 82.00
##  Mean   :0.02572   Mean   :236.7   Mean   :132.4   Mean   : 82.89
##  3rd Qu.:0.00000   3rd Qu.:263.0   3rd Qu.:144.0   3rd Qu.: 89.88
##  Max.   :1.00000   Max.   :696.0   Max.   :295.0   Max.   :142.50
##                    NA's   :50
##       BMI          heartRate         glucose         TenYearCHD
##  Min.   :15.54   Min.   : 44.00   Min.   : 40.00   Min.   :0.000
##  1st Qu.:23.07   1st Qu.: 68.00   1st Qu.: 71.00   1st Qu.:0.000
##  Median :25.40   Median : 75.00   Median : 78.00   Median :0.000
##  Mean   :25.80   Mean   : 75.88   Mean   : 81.97   Mean   :0.152
##  3rd Qu.:28.04   3rd Qu.: 83.00   3rd Qu.: 87.00   3rd Qu.:0.000
##  Max.   :56.80   Max.   :143.00   Max.   :394.00   Max.   :1.000
##  NA's   :19      NA's   :1        NA's   :388
```

2. Replace NA

```
education_median<-median(CHD$education,na.rm=TRUE)
CHD[is.na(education),education:=education_median]

cigsPerDay_median<-median(CHD$cigsPerDay,na.rm=TRUE)
CHD[is.na(cigsPerDay),cigsPerDay:=cigsPerDay_median]

BPMeds_median<-median(CHD$BPMeds,na.rm=TRUE)
CHD[is.na(BPMeds),BPMeds:=BPMeds_median]

totChol_median<-median(CHD$totChol,na.rm=TRUE)
CHD[is.na(totChol),totChol:=totChol_median]

glucose_median<-median(CHD$glucose,na.rm=TRUE)
CHD[is.na(glucose),glucose:=glucose_median]

heartRate_median<-median(CHD$heartRate,na.rm=TRUE)
CHD[is.na(heartRate),heartRate:=heartRate_median]

BMI_median<-median(CHD$BMI,na.rm=TRUE)
CHD[is.na(BMI),BMI:=BMI_median]
```

```
colnames(CHD)[1] <- 'is_male'
```

III.EDA

1. Distribution of Ten Year Risk of CHD

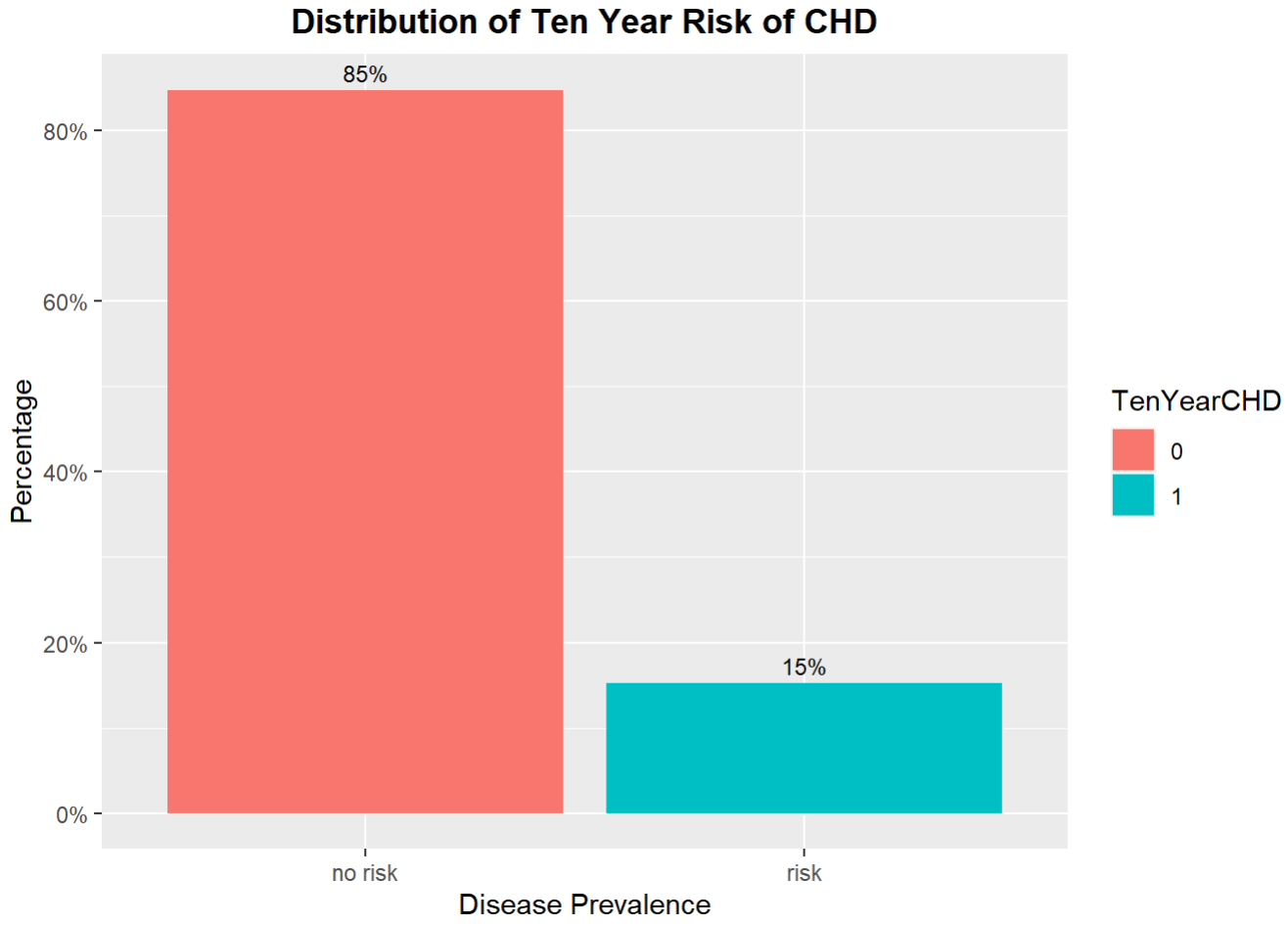
```
count1 <- length(which(CHD$TenYearCHD == 1))
count1
```

```
## [1] 644
```

```
count2 <- length(which(CHD$TenYearCHD == 0))
count2
```

```
## [1] 3594
```

```
common_theme <- theme(plot.title = element_text(hjust = 0.5, face = "bold"))

ggplot(data = CHD, aes(x = factor(TenYearCHD),
                       y = prop.table(stat(count)),
                       fill = factor(TenYearCHD),
                       label = scales::percent(prop.table(stat(count))))) +
    geom_bar(position = "dodge") +
    geom_text(stat = 'count',
              position = position_dodge(.9),
              vjust = -0.5,
              size = 3) +
    scale_x_discrete(labels = c("no risk", "risk"))+
    scale_y_continuous(labels = scales::percent)+
    labs(x = 'Disease Prevalence', y = 'Percentage',fill='TenYearCHD') +
    ggtitle("Distribution of Ten Year Risk of CHD") +
    common_theme
```



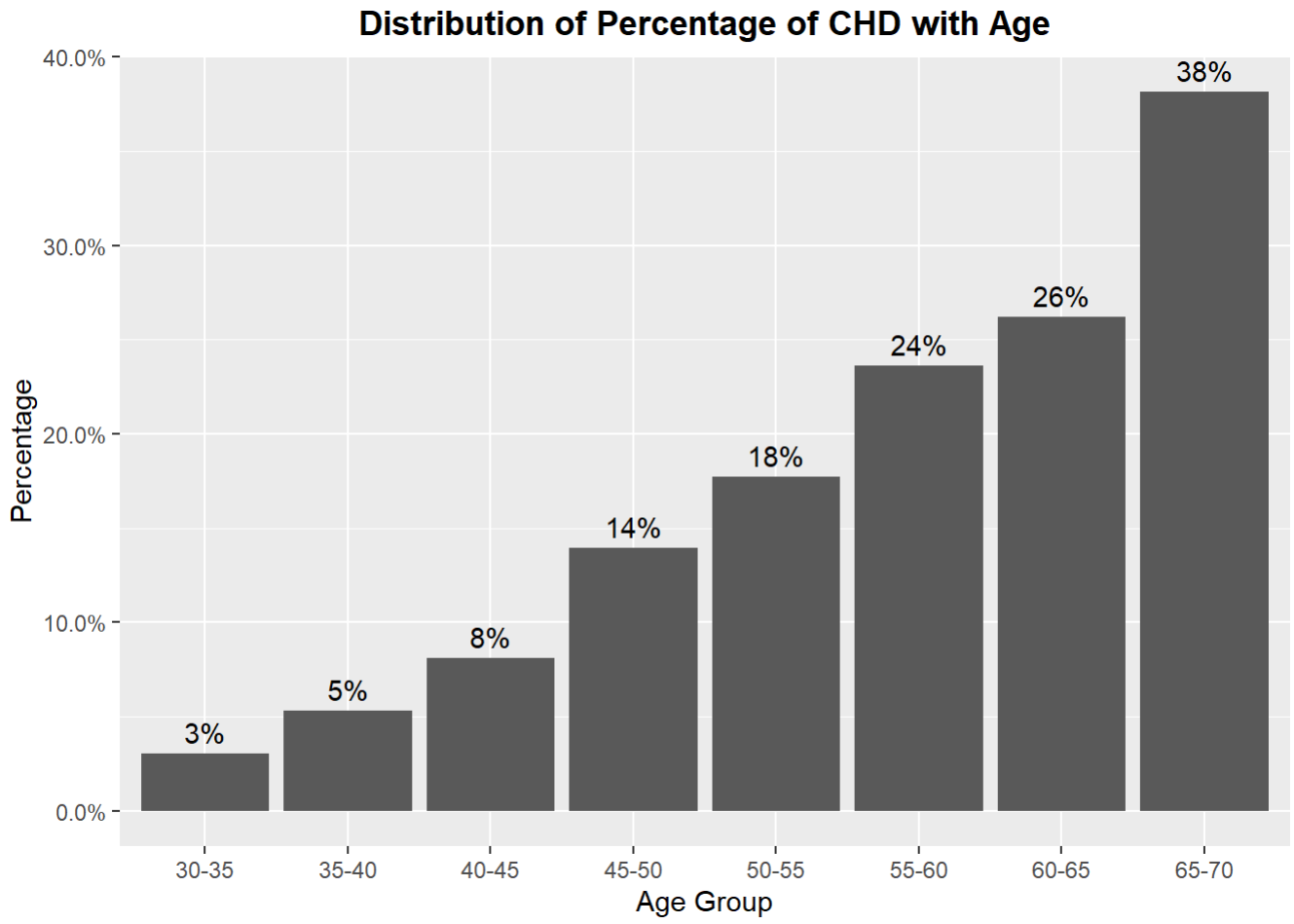
2. Distribution of Percentage of CHD with Age

```
CHD$agec <-
  cut(CHD$age, breaks = c(30,35,40,45,50,55,60,65,70),
      labels = c("30-35","35-40","40-45","45-50","50-55","55-60","60-65","65-70"))

d <- CHD %>% group_by(agec) %>% summarise(perc = mean(TenYearCHD=='1'))
d$perc_r <- round(d$perc,2)*100
d$perc_r <- interaction(d$perc_r, "%", sep = "")
d
```

agec <fct>	perc <dbl>	perc_r <fct>
30-35	0.03030303	3%
35-40	0.05294118	5%
40-45	0.08085612	8%
45-50	0.13932292	14%
50-55	0.17721519	18%
55-60	0.23608769	24%
60-65	0.26226013	26%
65-70	0.38181818	38%

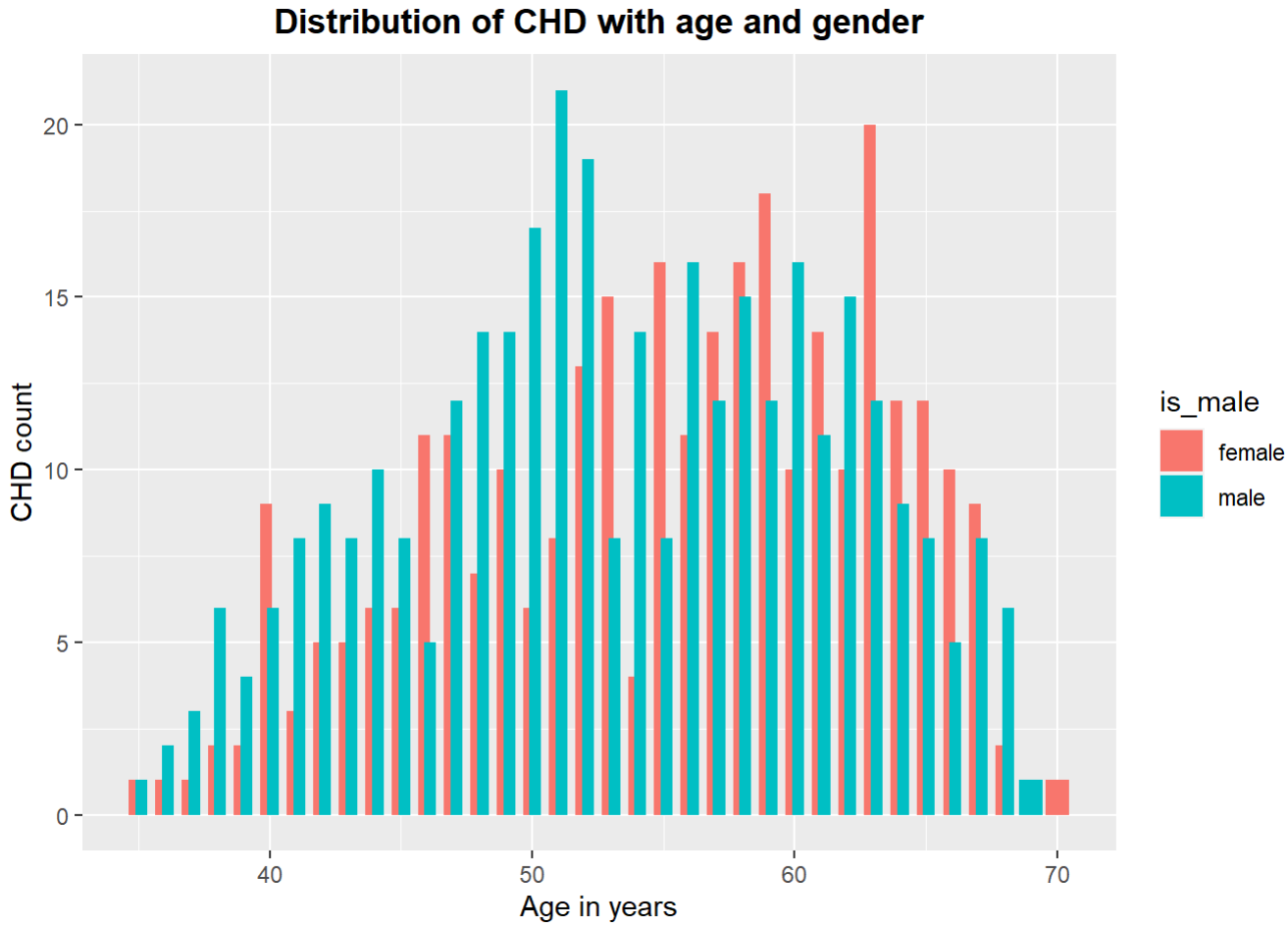
8 rows

```
ggplot(d,aes(x=agec,y=perc)) +
geom_col()+
scale_y_continuous(labels=scales::percent)+
geom_text(aes(label = perc_r), vjust = -0.5)+
labs(x='Age Group',y='Percentage')+
ggtitle("Distribution of Percentage of CHD with Age")+
common_theme
```



3. Histogram of CHD with age and gender

```
#cbPalette <- c("#999999", "#E69F00", "#56B4E9", "#009E73", #"#F0E442", "#0072B2", "#D55E00", "#
CC79A7")
CHD_1 <- CHD[ CHD$TenYearCHD=='1',]
CHD_1$is_male[CHD_1$is_male == 0] <- "female"
CHD_1$is_male[CHD_1$is_male == 1] <- "male"
ggplot(data=CHD_1,aes(age,fill=is_male))+
  geom_bar(position = position_dodge(width = 0.5))+
#  scale_fill_brewer(palette=cbPalette)+
  labs(x = "Age in years",y = "CHD count")+
  ggtitle("Distribution of CHD with age and gender")+
  common_theme
```



4. Probability of disease in smokers

```
d2 <- CHD %>% group_by(currentSmoker) %>% summarise(perc = mean(TenYearCHD=='1'))
d2
```

currentSmoker <int>	perc <dbl>
0	0.1450560
1	0.1590258

2 rows

5. Line Chart of Percentage of CHD with Age and Gender

```
d3 <- CHD %>% group_by(agec,factor(is_male)) %>% summarise(perc = mean(TenYearCHD=='1'))
```

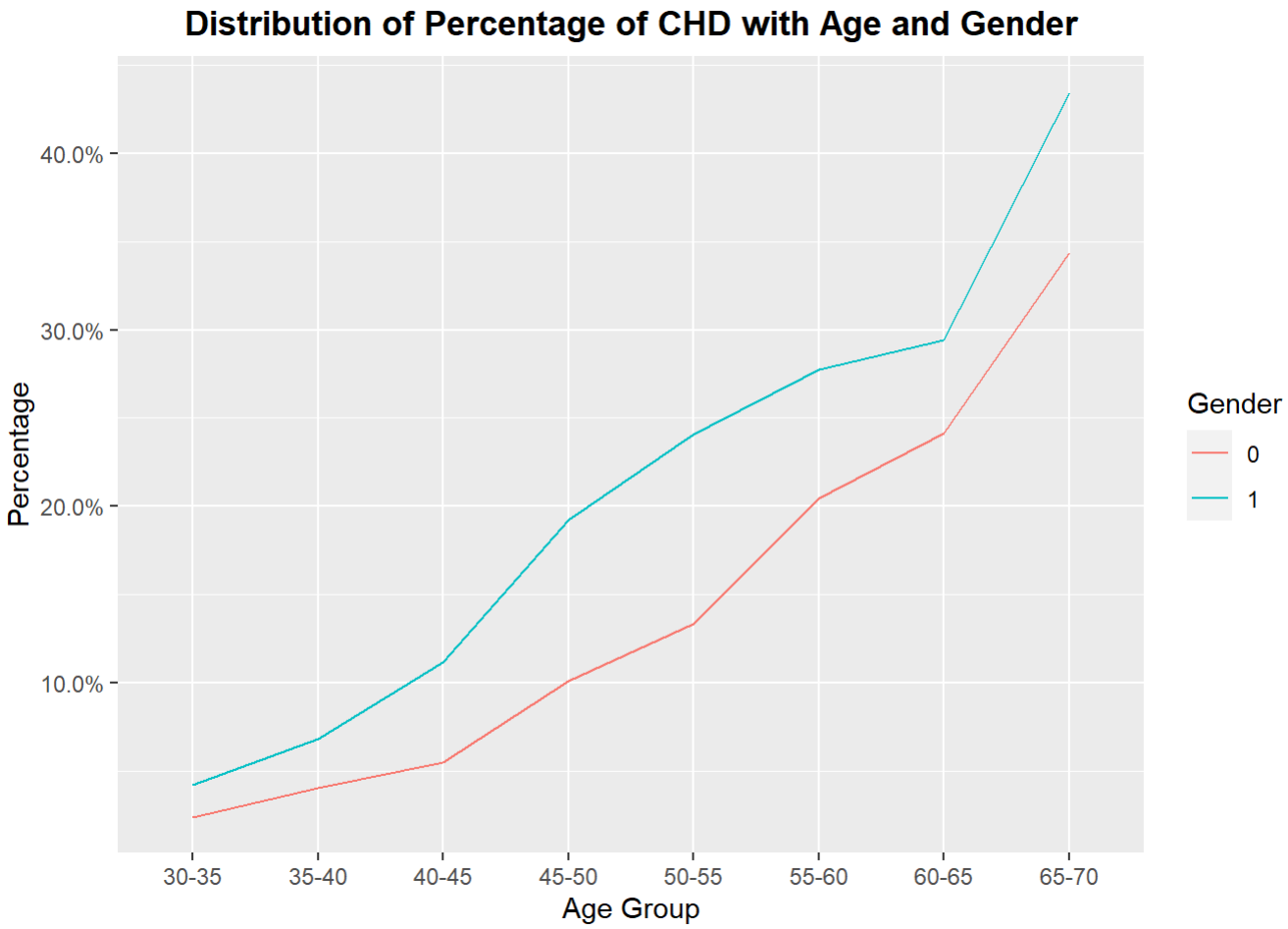
```
## `summarise()` has grouped output by 'agec'. You can override using the `.groups` argument.
```

```
d3
```

agec <fct>	factor(is_male) <fct>	perc <dbl>
30-35	0	0.02380952
30-35	1	0.04166667
35-40	0	0.04032258
35-40	1	0.06818182
40-45	0	0.05482456
40-45	1	0.11168831
45-50	0	0.10089686
45-50	1	0.19254658
50-55	0	0.13333333
50-55	1	0.24054983

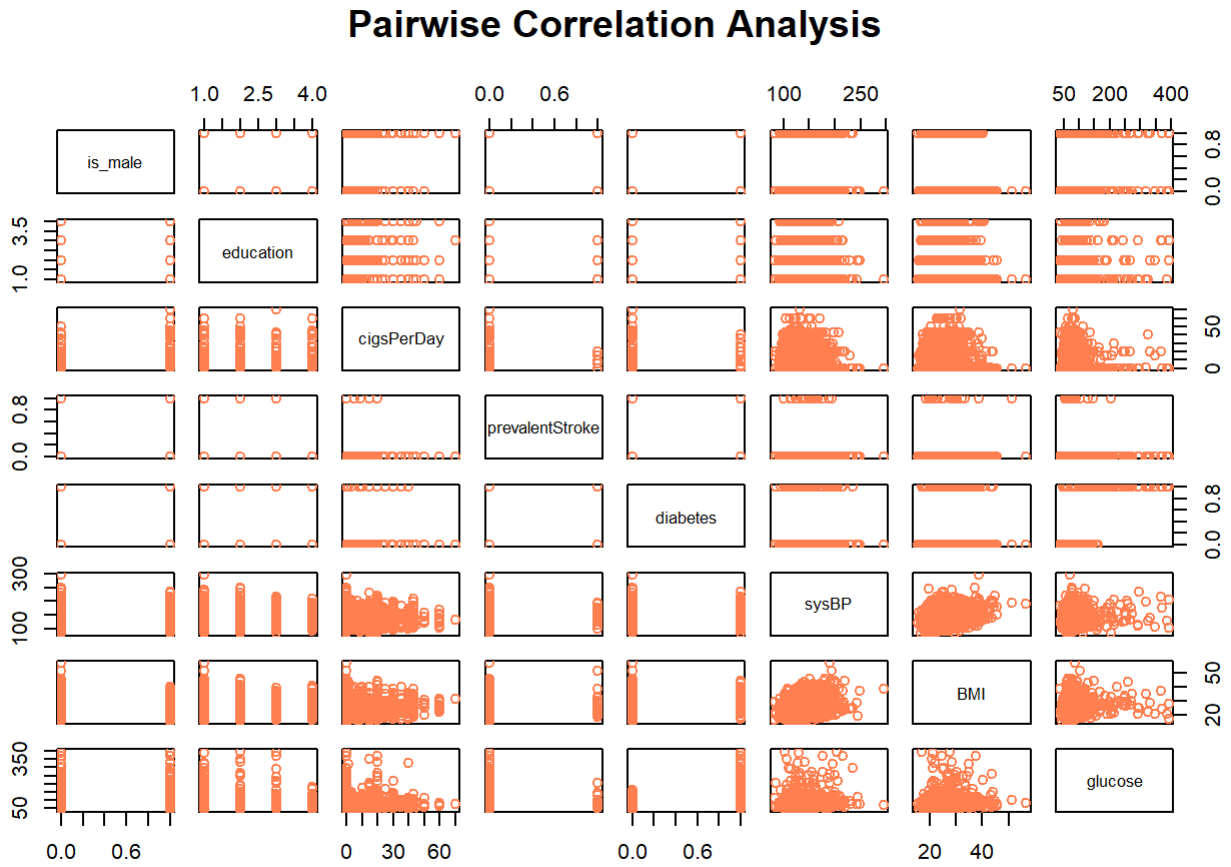
1-10 of 16 rows Previous **1** 2 Next

```
ggplot() +
geom_line(data=d3,aes(agec, perc,group =`factor(is_male)`,color =`factor(is_male)` ))+
scale_y_continuous(labels=scales::percent)+
labs(x='Age Group',y='Percentage',color='Gender' )+
ggtitle("Distribution of Percentage of CHD with Age and Gender")+
common_theme
```



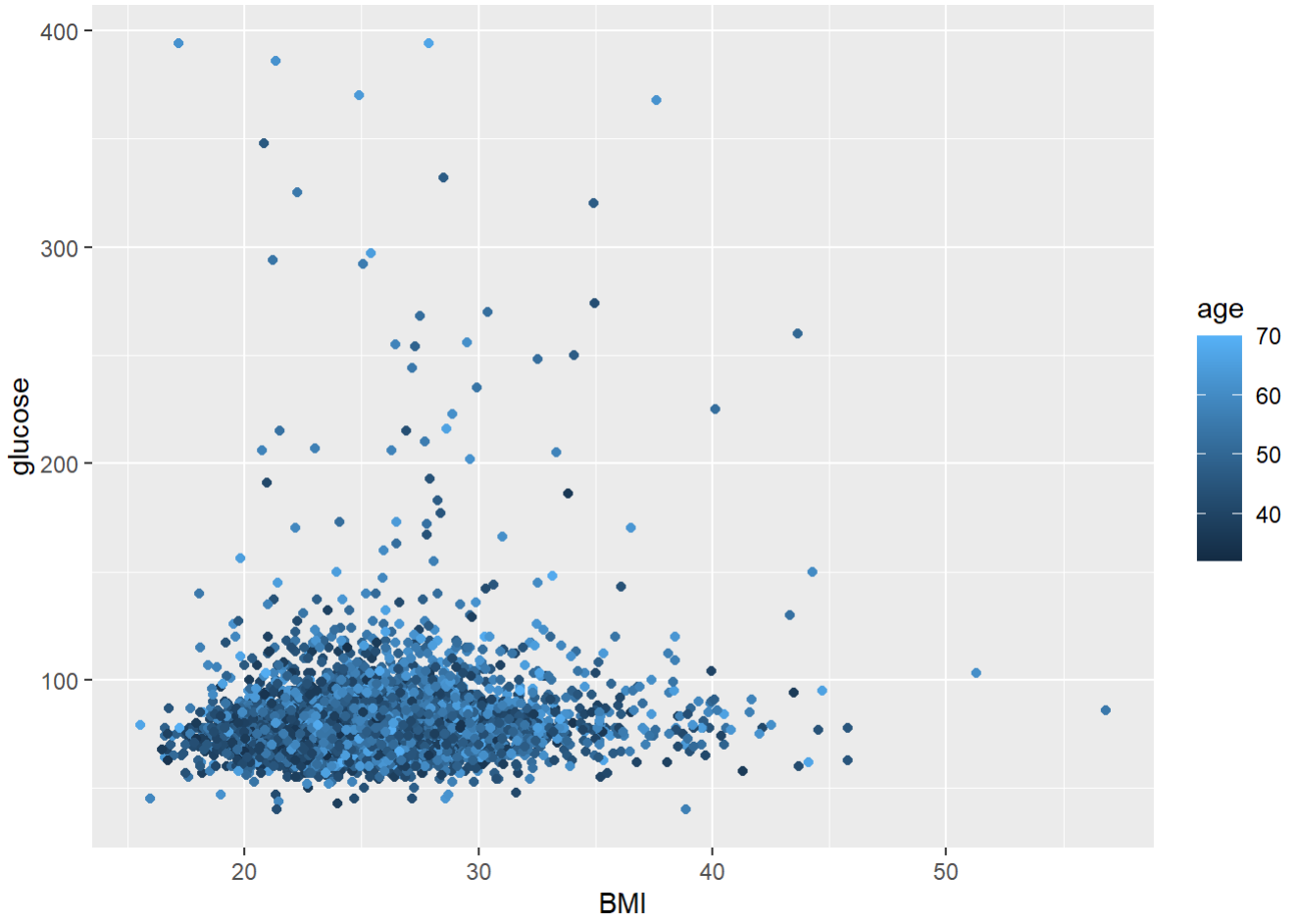
6. Pairwise Correlation Analysis

```
a <- CHD[,c(1,3,5,7,9,11,13,15)]
pairs(a, col = "coral",main = "Pairwise Correlation Analysis")
```



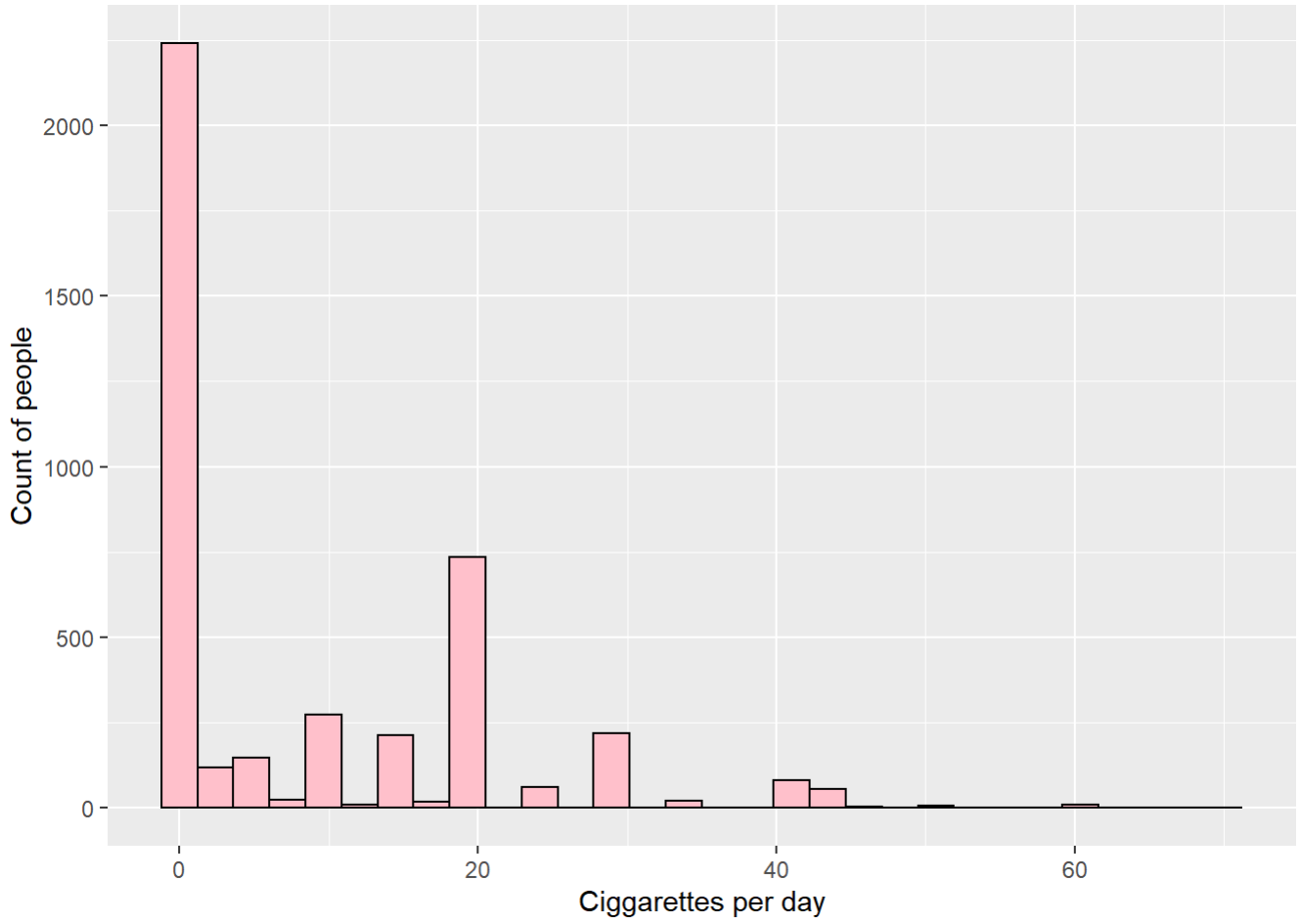
7. ???

```
ggplot(data = CHD, aes(BMI,glucose,color = age)) + geom_point(fill = "blue")
```




```
ggplot(data = CHD, aes(x = cigsPerDay, color = education)) + geom_histogram(color="black", fill=
"pink")+labs(x='Ciggarettes per day', y = 'Count of people')
```

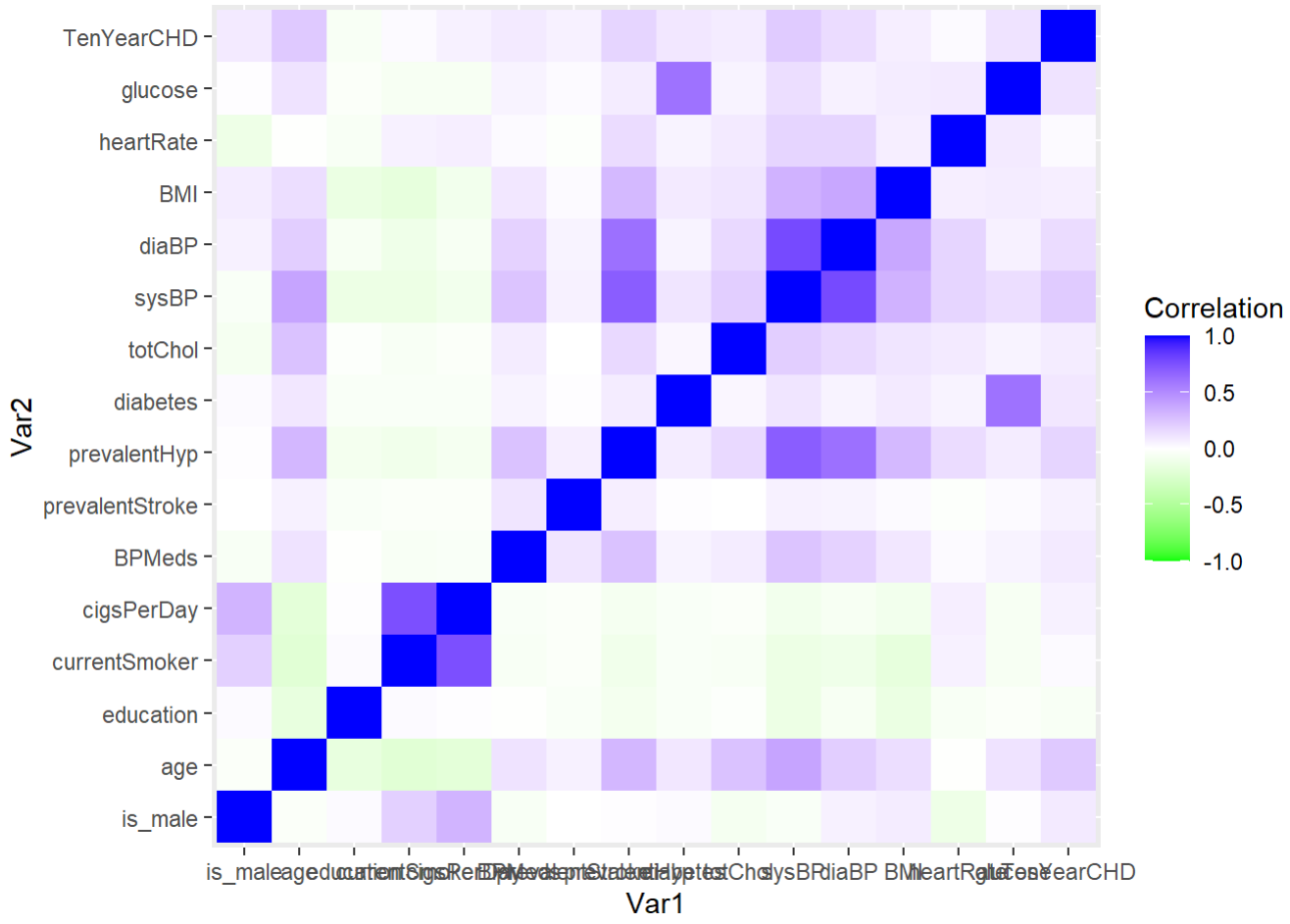
```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```




```
#+  + geom_vline(aes(intercept=mean(cigsPerDay)), color="blue", linetype="dashed", size=12)
```

8. Correlation Heatmap

```
CHD<-subset(CHD,select=-c(17))
cormat <- round(cor(CHD),2)
melted_cormat <- melt(cormat)
ggplot(data = melted_cormat, aes(x=Var1, y=Var2, fill=value)) +
common_theme+
geom_tile()+
scale_fill_gradient2(low = "green", high = "blue",
   midpoint = 0, limit = c(-1,1),
   name="Correlation")
```



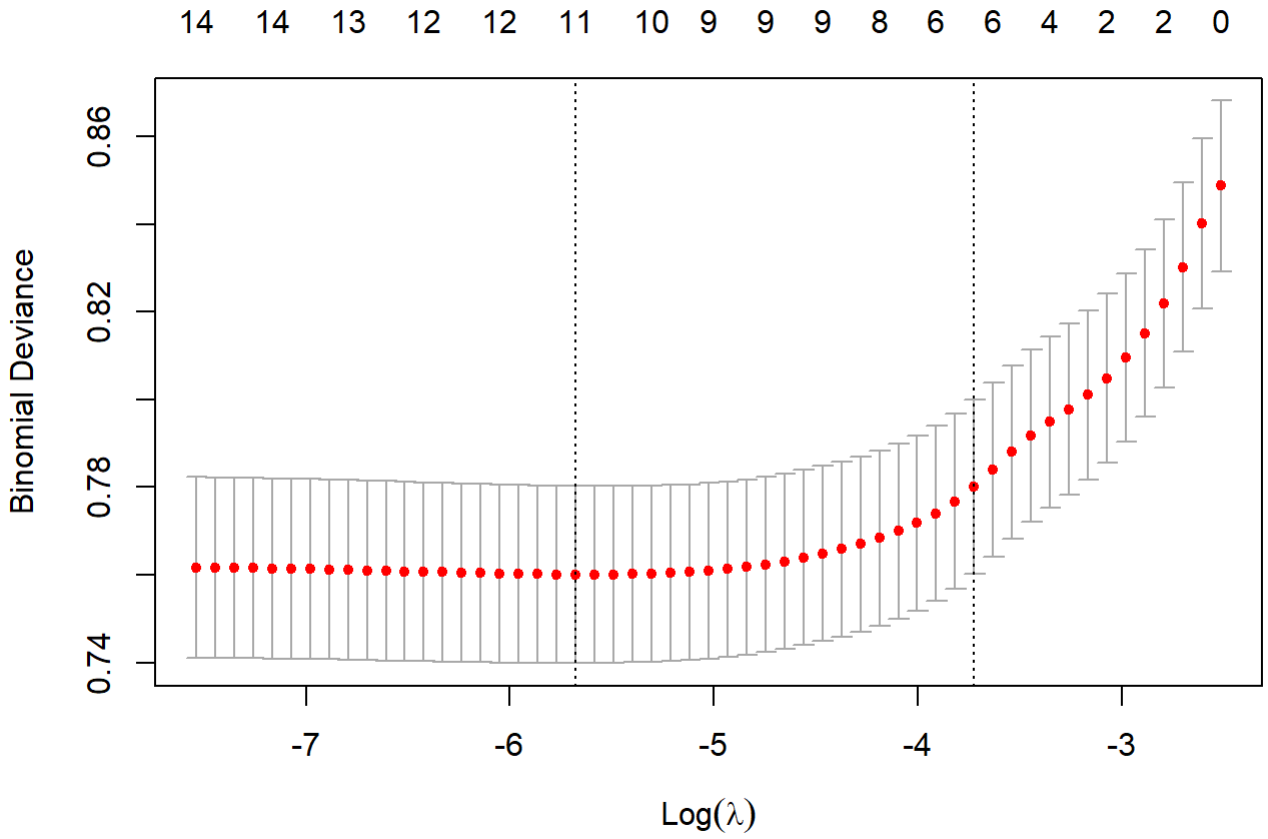
IV.Mechine Learning

1. Split Dataset

```
set.seed(1)
#train-test split ratio 0.8
id <- createDataPartition(CHD$TenYearCHD, p = 0.8, list = FALSE)

train<-CHD[id, ]
test<-CHD[-id, ]
```

2. Lasso Classification

```
# Create formula
formula <- as.formula(TenYearCHD ~ .)

# Training set modeling
train.matrix <- model.matrix(formula, train)[, -1]
train_y <- train$TenYearCHD
fit <- cv.glmnet(train.matrix, train_y, family = "binomial", alpha = 1, nfolds = 10)
#plot
plot(fit)
```




```
# Create testing matrices
test.matrix <- model.matrix(formula, test) [, -1]
```

```
coef(fit,s=fit$lambda.min)
```

```
## 16 x 1 sparse Matrix of class "dgCMatrix"
##                              s1
## (Intercept)     -7.8490631713
## is_male          0.4203721407
## age              0.0590477198
## education       -0.0166774422
## currentSmoker    .
## cigsPerDay       0.0184795736
## BPMeds           .
## prevalentStroke  0.8951979864
## prevalentHyp     0.2946043976
## diabetes         0.2048997981
## totChol          0.0018545443
## sysBP            0.0120846121
## diaBP            .
## BMI              0.0002624377
## heartRate        .
## glucose          0.0061765364
```

```
# Predicting test data

test.predictions <- predict(fit, test.matrix, s = fit$lambda.min, type = "response")

##F1 score, select cutoff which makes the F1 score largest
Fmeasure <- c()
cutoffs <- seq(0.05, 0.85, 0.01)
for(cutoff in cutoffs) {

predicted.CHD <- ifelse(test.predictions > cutoff, 1, 0)
cmat <- confusionMatrix(as.factor(predicted.CHD), as.factor(test$TenYearCHD), positive = "1")

Fmeasure <- c(Fmeasure,  cmat$byClass[7] )
}


cutoffs[which.max(Fmeasure)]
```

```
## [1] 0.15
```

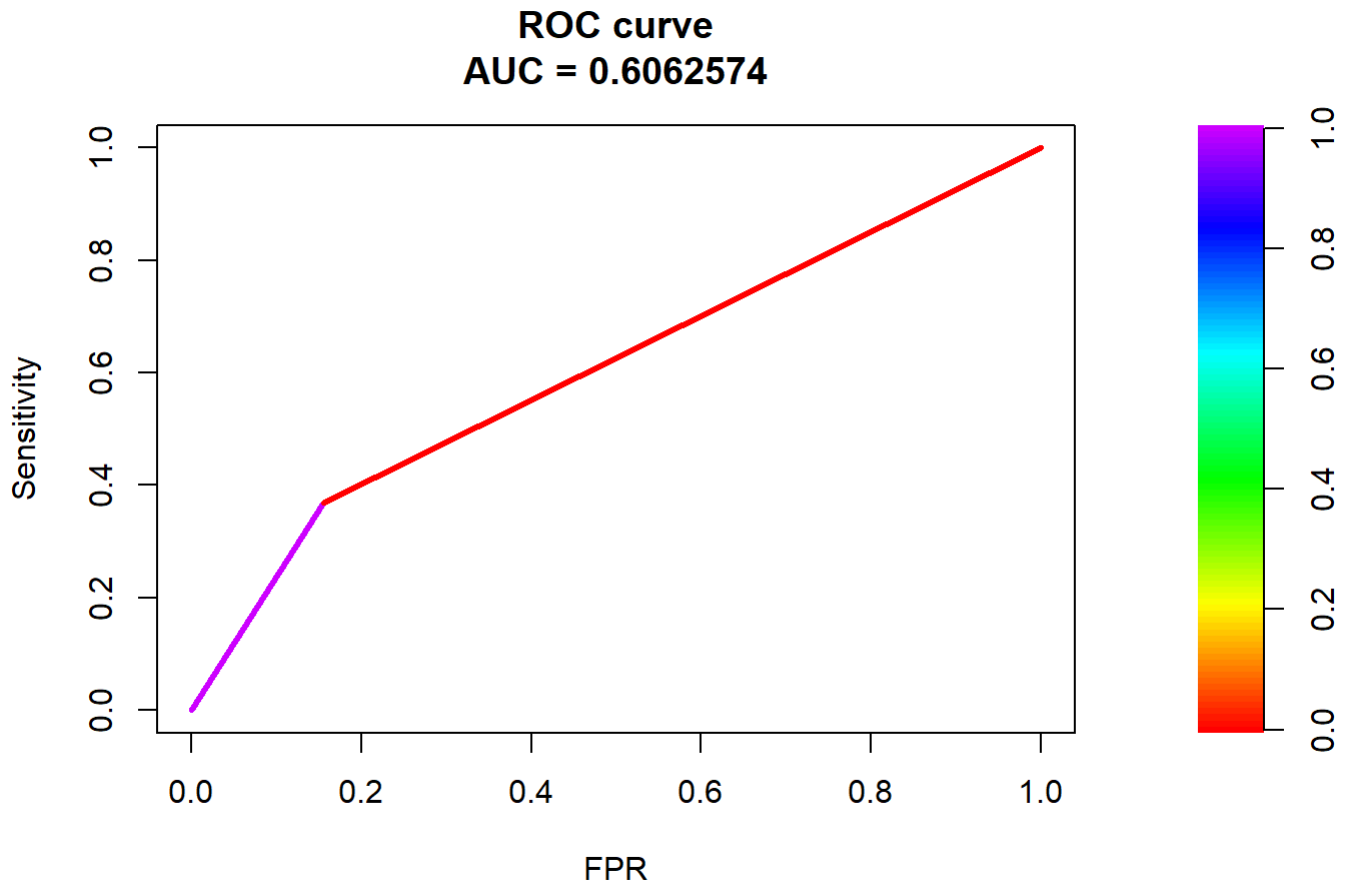
```
#0.15

predicted.CHD <- ifelse(test.predictions > cutoffs[which.max(Fmeasure)], 1, 0)
cmat <- confusionMatrix(as.factor(predicted.CHD), as.factor(test$TenYearCHD), positive = "1")
cmat
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction   0   1
##          0 492  43
##          1 223  89
##
##                Accuracy : 0.686
##                  95% CI : (0.6535, 0.7171)
##     No Information Rate : 0.8442
##     P-Value [Acc > NIR] : 1
##
##                   Kappa : 0.2329
##
##  Mcnemar's Test P-Value : <2e-16
##
##             Sensitivity : 0.6742
##             Specificity : 0.6881
##          Pos Pred Value : 0.2853
##          Neg Pred Value : 0.9196
##              Prevalence : 0.1558
##          Detection Rate : 0.1051
##    Detection Prevalence : 0.3684
##       Balanced Accuracy : 0.6812
##
##        'Positive' Class : 1
##
```

```
#F1 score
cmat$byClass[7]
```

```
##        F1
## 0.4009009
```

```
c<-roc.curve( as.numeric(predicted.CHD),as.numeric(test$TenYearCHD), curve = TRUE)
plot(c)
```



3. Logistic Classification

```
#use variables selected by lasso
coefs <- coef(fit,s=fit$lambda.min)
variables <- which(coefs !=0)

selectvariables <- names(coefs[variables,])[-1]
selectvariables
```

```
##  [1] "is_male"         "age"             "education"       "cigsPerDay"
##  [5] "prevalentStroke" "prevalentHyp"    "diabetes"        "totChol"
##  [9] "sysBP"           "BMI"             "glucose"
```

```
train2<-train.matrix[,selectvariables]
test2<-test.matrix[,selectvariables]

newtrain <- data.frame(train2, TenYearCHD = train$TenYearCHD)
newtest <- data.frame(test2, TenYearCHD = test$TenYearCHD)

fit2 <- glm(TenYearCHD ~ ., data = newtrain, family = binomial(link = "logit"))
summary(fit2)
```

```
##
## Call:
## glm(formula = TenYearCHD ~ ., family = binomial(link = "logit"),
##     data = newtrain)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.4614  -0.5980  -0.4288  -0.2840   2.8083
##
## Coefficients:
##                  Estimate Std. Error z value Pr(>|z|)
## (Intercept)     -8.398638   0.656126 -12.800  < 2e-16 ***
## is_male          0.476991   0.110994   4.297 1.73e-05 ***
## age              0.062486   0.006816   9.168  < 2e-16 ***
## education       -0.037488   0.052288  -0.717  0.47341
## cigsPerDay       0.021195   0.004427   4.788 1.69e-06 ***
## prevalentStroke  1.085868   0.504250   2.153  0.03128 *
## prevalentHyp     0.318169   0.140293   2.268  0.02334 *
## diabetes         0.245586   0.328234   0.748  0.45434
## totChol          0.002446   0.001149   2.128  0.03330 *
## sysBP            0.012255   0.003048   4.021 5.79e-05 ***
## BMI              0.004756   0.012760   0.373  0.70935
## glucose          0.006751   0.002419   2.791  0.00525 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 2878.4  on 3390  degrees of freedom
## Residual deviance: 2552.3  on 3379  degrees of freedom
## AIC: 2576.3
##
## Number of Fisher Scoring iterations: 5
```

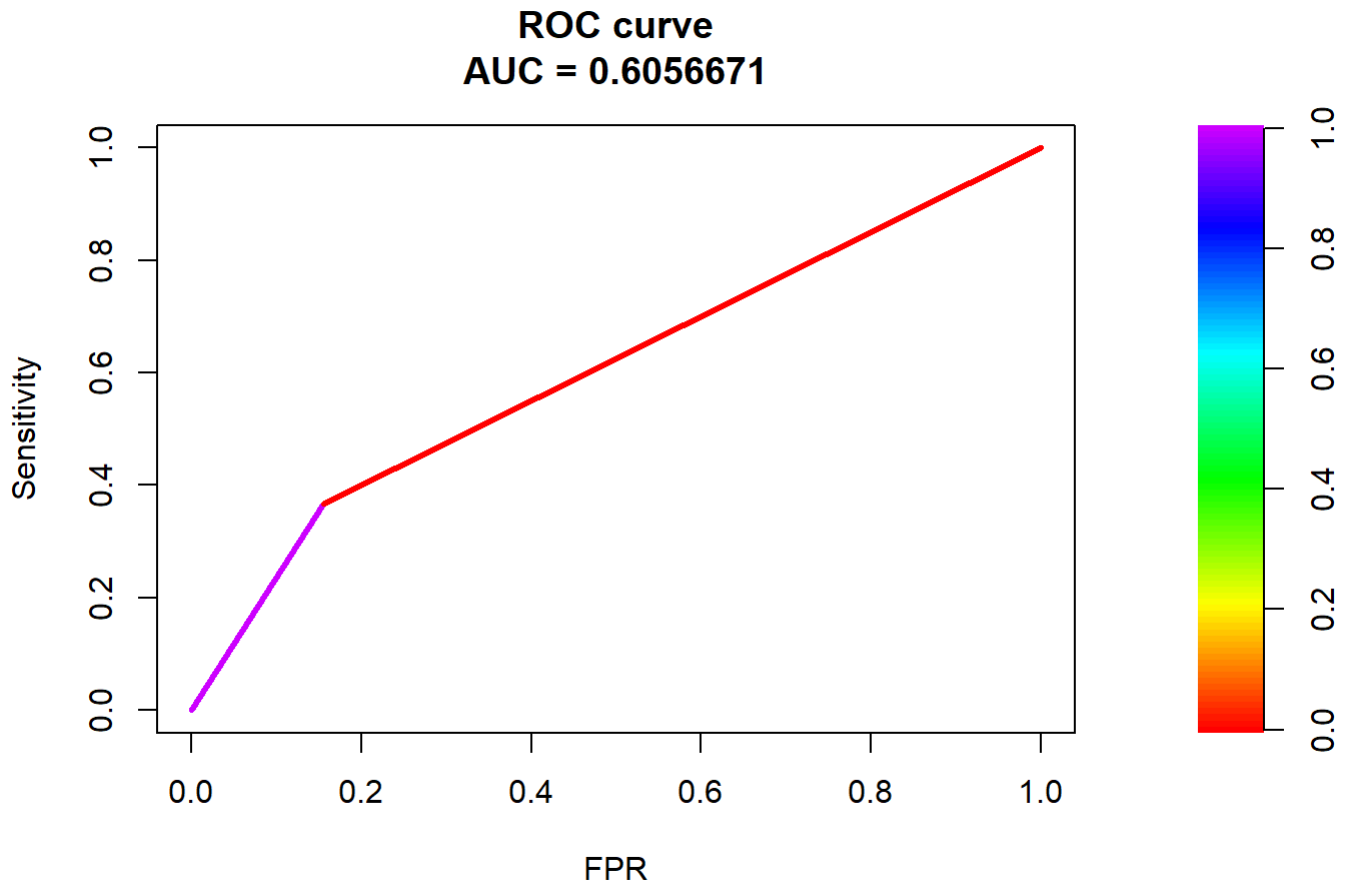
```
# Predicting test data

test.predictions <- predict(fit2, newtest,  type = "response")


predicted.CHD <- ifelse(test.predictions > cutoffs[which.max(Fmeasure)], 1, 0)
cmat <- confusionMatrix(as.factor(predicted.CHD), as.factor(test$TenYearCHD), positive = "1")
cmat
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction   0   1
##          0 493  43
##          1 222  89
##
##                Accuracy : 0.6871
##                  95% CI : (0.6547, 0.7182)
##     No Information Rate : 0.8442
##     P-Value [Acc > NIR] : 1
##
##                   Kappa : 0.2342
##
##  Mcnemar's Test P-Value : <2e-16
##
##             Sensitivity : 0.6742
##             Specificity : 0.6895
##          Pos Pred Value : 0.2862
##          Neg Pred Value : 0.9198
##              Prevalence : 0.1558
##          Detection Rate : 0.1051
##    Detection Prevalence : 0.3672
##       Balanced Accuracy : 0.6819
##
##        'Positive' Class : 1
##
```

```
#F1 score
cmat$byClass[7]
```

```
##        F1
## 0.4018059
```

```
c<-roc.curve( as.numeric(predicted.CHD),as.numeric(test$TenYearCHD), curve = TRUE)
plot(c)
```




```
#use full data
fit3 <- glm(TenYearCHD ~ ., data = train, family = binomial(link = "logit"))
summary(fit3)
```

```
##
## Call:
## glm(formula = TenYearCHD ~ ., family = binomial(link = "logit"),
##     data = train)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.4575  -0.5961  -0.4290  -0.2858   2.8016
##
## Coefficients:
##                  Estimate Std. Error z value Pr(>|z|)
## (Intercept)     -8.236599   0.750388 -10.976  < 2e-16 ***
## is_male          0.479179   0.112859   4.246 2.18e-05 ***
## age              0.061703   0.007005   8.808  < 2e-16 ***
## education       -0.036202   0.052465  -0.690 0.490188
## currentSmoker    0.057498   0.162072   0.355 0.722762
## cigsPerDay       0.019604   0.006550   2.993 0.002762 **
## BPMeds          -0.028488   0.253049  -0.113 0.910364
## prevalentStroke  1.085554   0.508730   2.134 0.032855 *
## prevalentHyp     0.332749   0.142534   2.335 0.019568 *
## diabetes         0.239344   0.328934   0.728 0.466838
## totChol          0.002476   0.001154   2.146 0.031859 *
## sysBP            0.013626   0.003965   3.437 0.000589 ***
## diaBP           -0.003255   0.006664  -0.489 0.625179
## BMI              0.006589   0.013076   0.504 0.614317
## heartRate       -0.001469   0.004353  -0.337 0.735800
## glucose          0.006772   0.002422   2.796 0.005180 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 2878.4  on 3390  degrees of freedom
## Residual deviance: 2551.8  on 3375  degrees of freedom
## AIC: 2583.8
##
## Number of Fisher Scoring iterations: 5
```

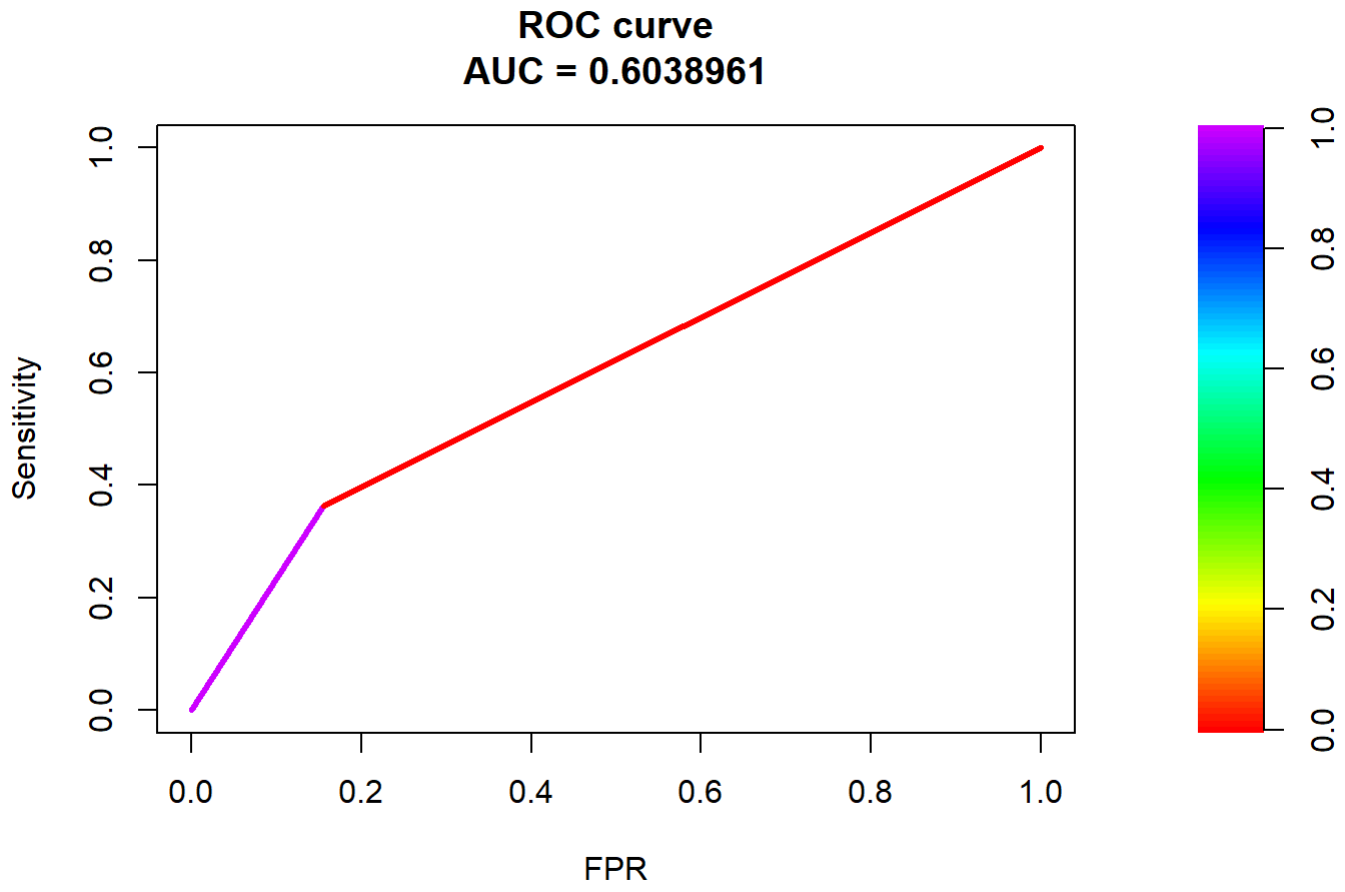
```
# Predicting test data

test.predictions <- predict(fit3, test,  type = "response")


predicted.CHD <- ifelse(test.predictions > cutoffs[which.max(Fmeasure)], 1, 0)
cmat <- confusionMatrix(as.factor(predicted.CHD), as.factor(test$TenYearCHD), positive = "1")
cmat
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction   0   1
##          0 495  44
##          1 220  88
##
##                Accuracy : 0.6883
##                  95% CI : (0.6559, 0.7194)
##     No Information Rate : 0.8442
##     P-Value [Acc > NIR] : 1
##
##                   Kappa : 0.2326
##
##  Mcnemar's Test P-Value : <2e-16
##
##             Sensitivity : 0.6667
##             Specificity : 0.6923
##          Pos Pred Value : 0.2857
##          Neg Pred Value : 0.9184
##              Prevalence : 0.1558
##          Detection Rate : 0.1039
##    Detection Prevalence : 0.3636
##       Balanced Accuracy : 0.6795
##
##        'Positive' Class : 1
##
```

```
#F1 score
cmat$byClass[7]
```

```
##  F1
## 0.4
```

```
c<-roc.curve( as.numeric(predicted.CHD),as.numeric(test$TenYearCHD), curve = TRUE)
plot(c)
```



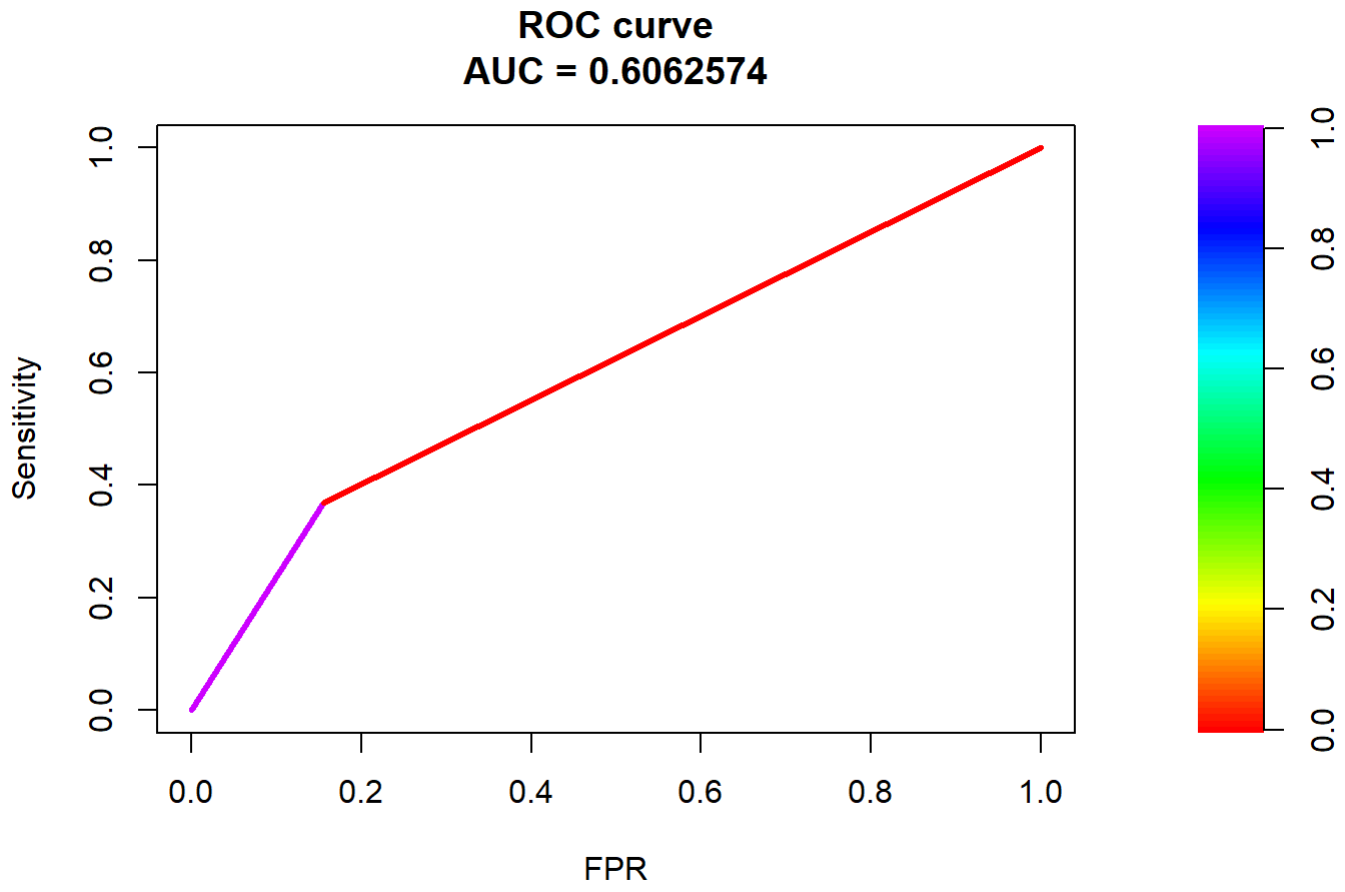

```
#use  backward selection with AIC criterion

fit4 <- step(fit3,trace = F)
summary(fit4)
```

```
##
## Call:
## glm(formula = TenYearCHD ~ is_male + age + cigsPerDay + prevalentStroke +
##     prevalentHyp + totChol + sysBP + glucose, family = binomial(link = "logit"),
##     data = train)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.4779  -0.5995  -0.4289  -0.2856   2.8100
##
## Coefficients:
##                  Estimate Std. Error z value Pr(>|z|)
## (Intercept)     -8.526981   0.543181 -15.698  < 2e-16 ***
## is_male          0.483458   0.110627   4.370 1.24e-05 ***
## age              0.063097   0.006740   9.362  < 2e-16 ***
## cigsPerDay       0.021070   0.004409   4.779 1.76e-06 ***
## prevalentStroke  1.095744   0.502536   2.180   0.0292 *
## prevalentHyp     0.322538   0.139643   2.310   0.0209 *
## totChol          0.002427   0.001149   2.112   0.0347 *
## sysBP            0.012658   0.003004   4.214 2.51e-05 ***
## glucose          0.007998   0.001856   4.309 1.64e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 2878.4  on 3390  degrees of freedom
## Residual deviance: 2553.6  on 3382  degrees of freedom
## AIC: 2571.6
##
## Number of Fisher Scoring iterations: 5
```

```
# Predicting test data

test.predictions <- predict(fit4, test,  type = "response")


predicted.CHD <- ifelse(test.predictions > cutoffs[which.max(Fmeasure)], 1, 0)
cmat <- confusionMatrix(as.factor(predicted.CHD), as.factor(test$TenYearCHD), positive = "1")
cmat
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction   0   1
##          0 493  42
##          1 222  90
##
##                Accuracy : 0.6883
##                  95% CI : (0.6559, 0.7194)
##     No Information Rate : 0.8442
##     P-Value [Acc > NIR] : 1
##
##                   Kappa : 0.2387
##
##  Mcnemar's Test P-Value : <2e-16
##
##             Sensitivity : 0.6818
##             Specificity : 0.6895
##          Pos Pred Value : 0.2885
##          Neg Pred Value : 0.9215
##              Prevalence : 0.1558
##          Detection Rate : 0.1063
##    Detection Prevalence : 0.3684
##       Balanced Accuracy : 0.6857
##
##        'Positive' Class : 1
##
```

```
c<-roc.curve( as.numeric(predicted.CHD),as.numeric(test$TenYearCHD), curve = TRUE)
plot(c)
```



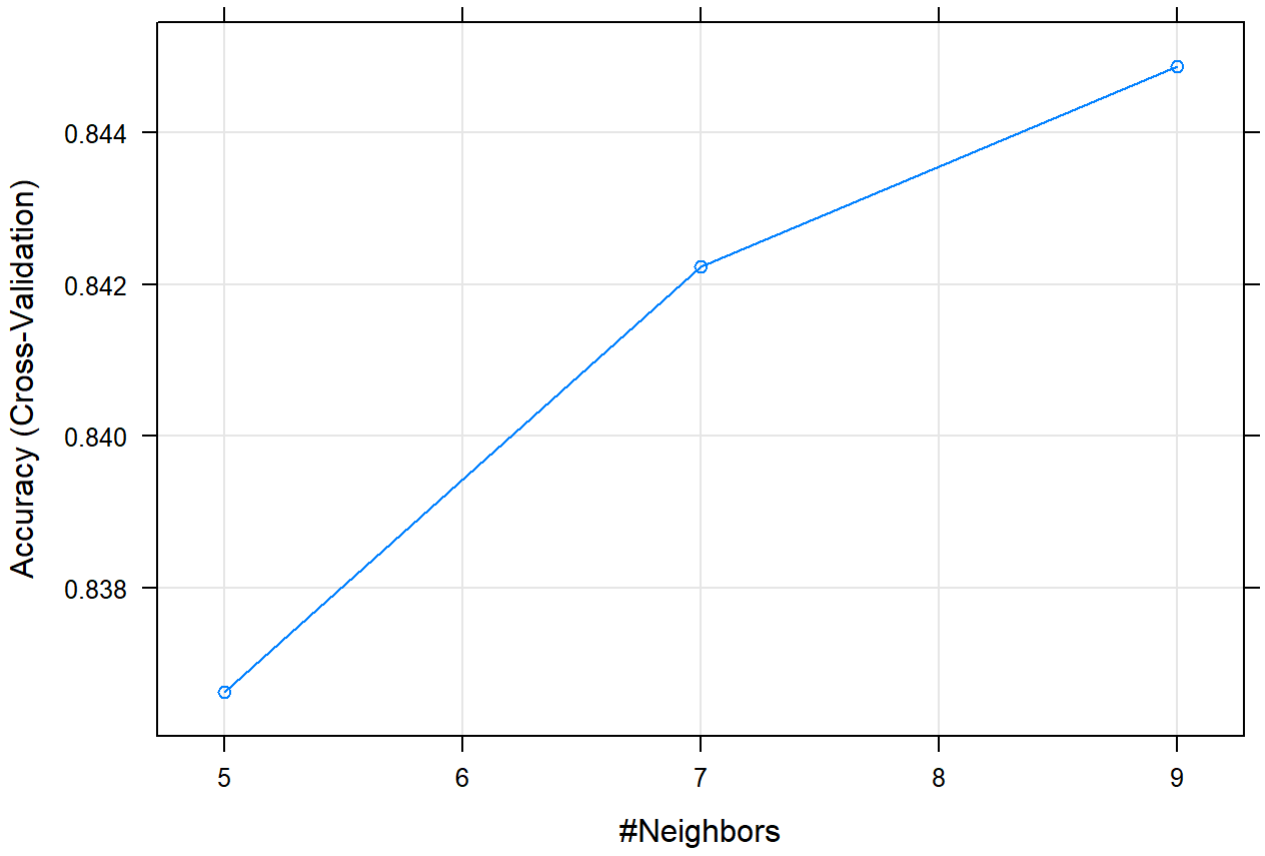

```
#all logistic models are similar
```

4. KNN - k-nearest neighbors

```
set.seed(1)
#set  10-folds cross validation
ctrl <- trainControl(method = "cv",
                     number = 10)
#KNN for k-nearest neighbors

#check parameters tuning results
m <- train(factor(TenYearCHD) ~ ., data = train,
           method = "knn",
           trControl = ctrl)

m
```

```
## k-Nearest Neighbors
##
## 3391 samples
##   15 predictor
##    2 classes: '0', '1'
##
## No pre-processing
## Resampling: Cross-Validated (10 fold)
## Summary of sample sizes: 3053, 3052, 3052, 3051, 3052, 3052, ...
## Resampling results across tuning parameters:
##
##   k  Accuracy   Kappa
##   5  0.8366204  0.10312146
##   7  0.8422329  0.09901922
##   9  0.8448817  0.06651901
##
## Accuracy was used to select the optimal model using the largest value.
## The final value used for the model was k = 9.
```

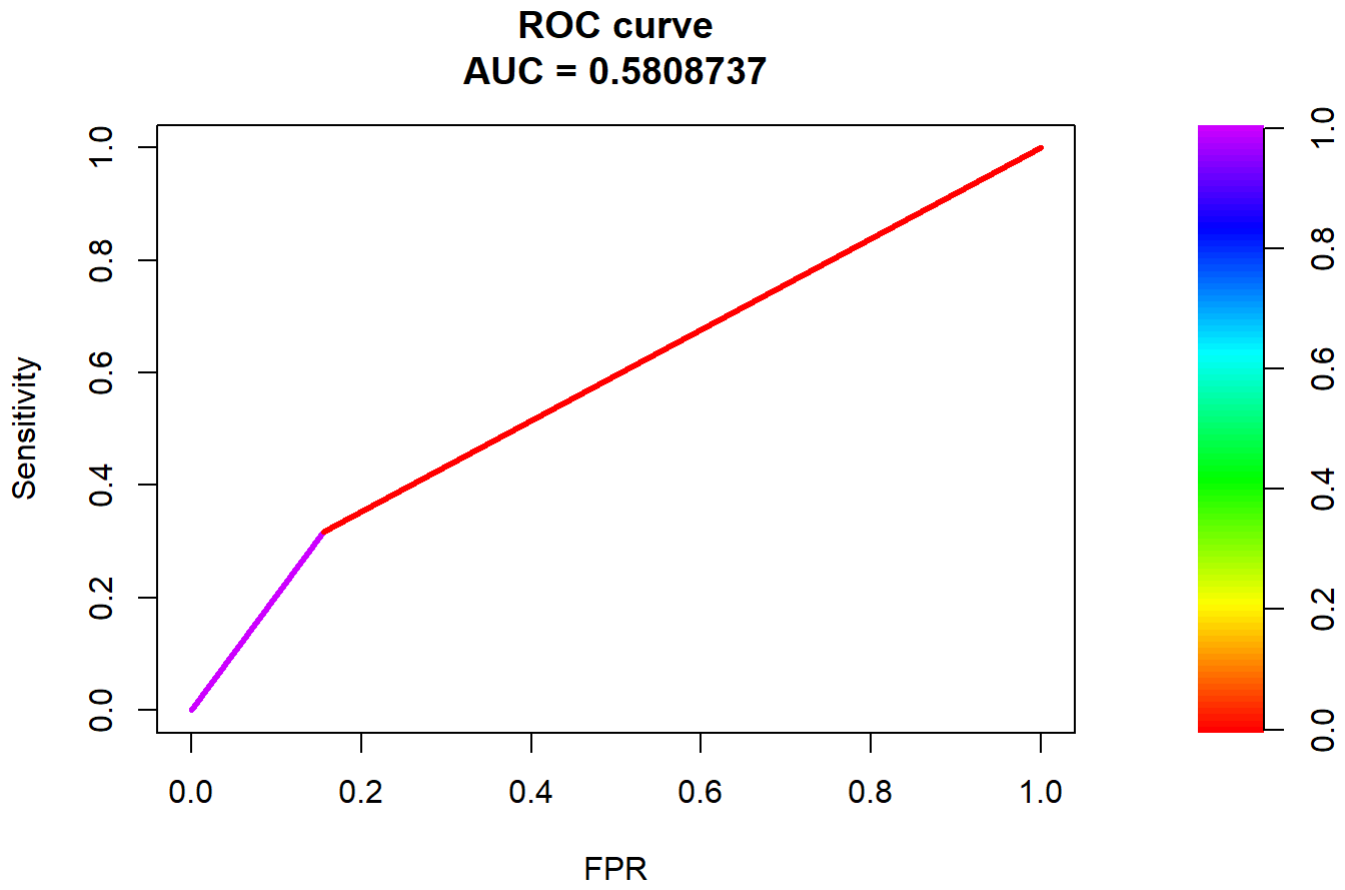
```
plot(m)
```



```
test.predictions <- predict(m, test,  type = "prob")[,2]


predicted.CHD <- ifelse(test.predictions > cutoffs[which.max(Fmeasure)], 1, 0)
cmat <- confusionMatrix(as.factor(predicted.CHD), as.factor(test$TenYearCHD), positive = "1")
cmat
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction   0   1
##          0 513  65
##          1 202  67
##
##                Accuracy : 0.6848
##                  95% CI : (0.6523, 0.716)
##     No Information Rate : 0.8442
##     P-Value [Acc > NIR] : 1
##
##                   Kappa : 0.1581
##
##  Mcnemar's Test P-Value : <2e-16
##
##             Sensitivity : 0.5076
##             Specificity : 0.7175
##          Pos Pred Value : 0.2491
##          Neg Pred Value : 0.8875
##              Prevalence : 0.1558
##          Detection Rate : 0.0791
##    Detection Prevalence : 0.3176
##       Balanced Accuracy : 0.6125
##
##        'Positive' Class : 1
##
```

```
c<-roc.curve( as.numeric(predicted.CHD),as.numeric(test$TenYearCHD), curve = TRUE)
plot(c)
```

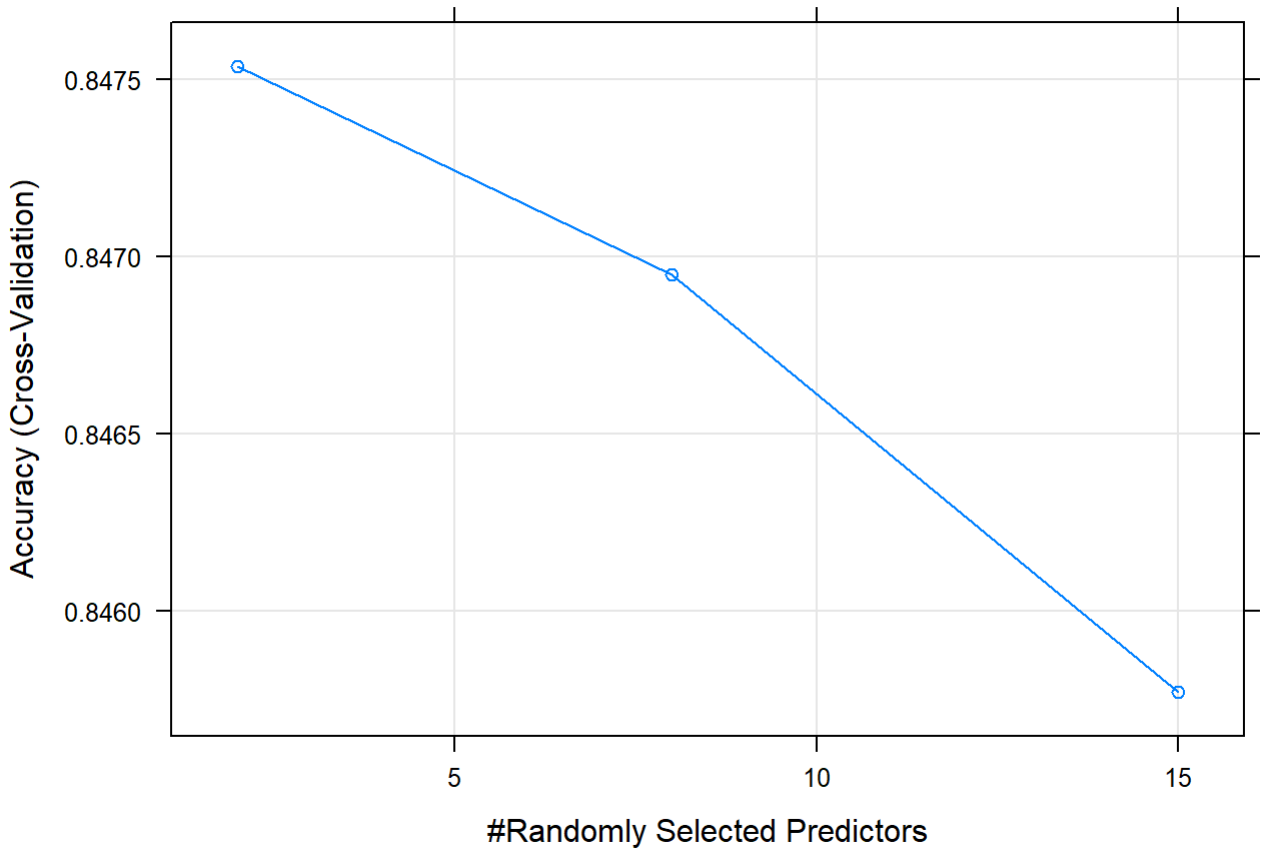


5. random forest

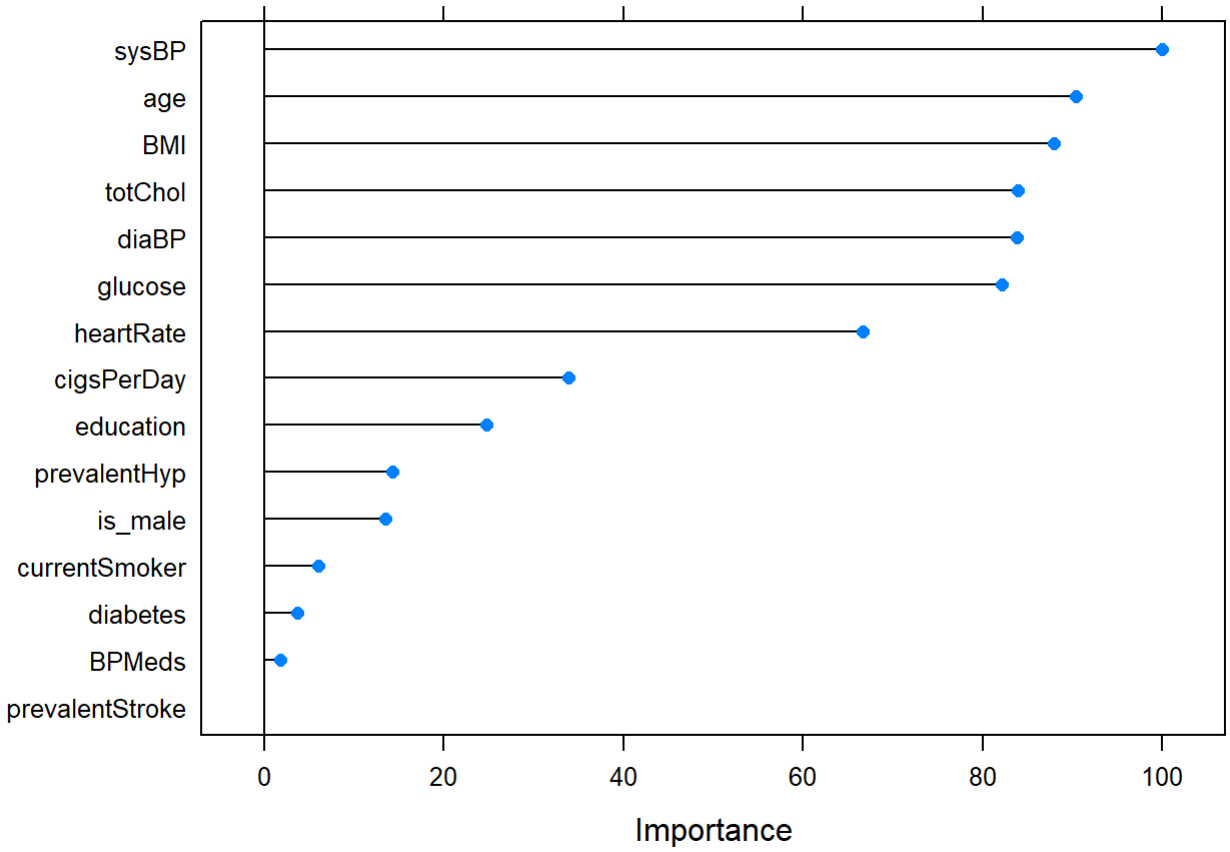
```
set.seed(1)
#set  10-folds cross validation
ctrl <- trainControl(method = "cv",
                     number = 10)
#rf for random forest

#check parameters tuning results
m <- train(factor(TenYearCHD) ~ ., data = train,
           method = "rf",
           trControl = ctrl)

m
```

```
## Random Forest
##
## 3391 samples
##   15 predictor
##    2 classes: '0', '1'
##
## No pre-processing
## Resampling: Cross-Validated (10 fold)
## Summary of sample sizes: 3053, 3052, 3052, 3051, 3052, 3052, ...
## Resampling results across tuning parameters:
##
##   mtry  Accuracy   Kappa
##    2    0.8475375  0.01539953
##    8    0.8469492  0.10051590
##   15    0.8457702  0.10693950
##
## Accuracy was used to select the optimal model using the largest value.
## The final value used for the model was mtry = 2.
```

```
plot(m)
```



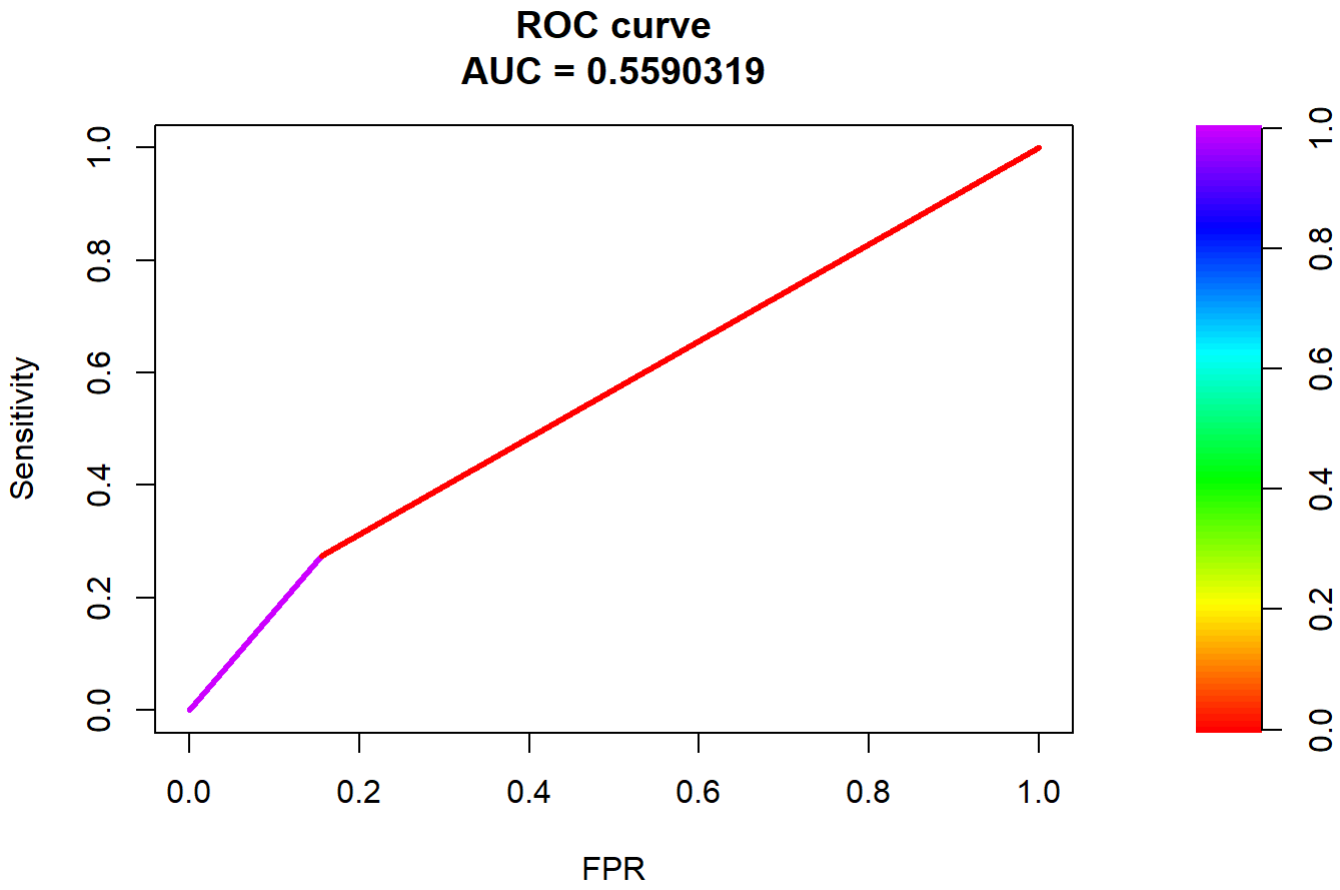

```
#variable important plot
plot(varImp(m))
```




```
test.predictions <- predict(m, test,  type = "prob")[,2]


predicted.CHD <- ifelse(test.predictions > cutoffs[which.max(Fmeasure)], 1, 0)
cmat <- confusionMatrix(as.factor(predicted.CHD), as.factor(test$TenYearCHD), positive = "1")
cmat
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction   0   1
##          0 548  67
##          1 167  65
##
##                Accuracy : 0.7237
##                  95% CI : (0.6923, 0.7536)
##     No Information Rate : 0.8442
##     P-Value [Acc > NIR] : 1
##
##                   Kappa : 0.1978
##
##  Mcnemar's Test P-Value : 9.682e-11
##
##             Sensitivity : 0.49242
##             Specificity : 0.76643
##          Pos Pred Value : 0.28017
##          Neg Pred Value : 0.89106
##              Prevalence : 0.15584
##          Detection Rate : 0.07674
##    Detection Prevalence : 0.27391
##       Balanced Accuracy : 0.62943
##
##        'Positive' Class : 1
##
```

```
c<-roc.curve( as.numeric(predicted.CHD),as.numeric(test$TenYearCHD), curve = TRUE)
plot(c)
```


ROC curve
AUC = 0.5590319
Sensitivity
0.0
0.2
0.4
0.6
0.8
1.0
FPR
0.0
0.2
0.4
0.6
0.8
1.0