Final Project- Heart Disease Detection

Puneeth Reddy Golla

2024-05-02

#Initialising Dataset:

library(readr)  
library(dplyr)

## Warning: package 'dplyr' was built under R version 4.3.3

##   
## Attaching package: 'dplyr'

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

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

library(caret)

## Warning: package 'caret' was built under R version 4.3.3

## Loading required package: ggplot2

## Warning: package 'ggplot2' was built under R version 4.3.3

## Loading required package: lattice

library(lubridate)

## Warning: package 'lubridate' was built under R version 4.3.3

##   
## Attaching package: 'lubridate'

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

library(readr)  
 heart <- read.csv("C:/Users/punee/Downloads/heart.csv")

#Renaming columns for better understanding

colnames(heart) <- c("age", "gender", "chest\_pain\_type", "resting\_blood\_pressure",   
 "serum\_cholesterol", "fasting\_blood\_sugar", "resting\_ecg\_results",   
 "max\_heart\_rate", "exercise\_induced\_angina", "ST\_depression",   
 "slope\_peak\_ST\_segment", "num\_major\_vessels", "thalium\_stress\_test",   
 "heart\_disease\_status")  
  
#Understanding Dataset  
str(heart)

## 'data.frame': 303 obs. of 14 variables:  
## $ age : int 63 37 41 56 57 57 56 44 52 57 ...  
## $ gender : int 1 1 0 1 0 1 0 1 1 1 ...  
## $ chest\_pain\_type : int 3 2 1 1 0 0 1 1 2 2 ...  
## $ resting\_blood\_pressure : int 145 130 130 120 120 140 140 120 172 150 ...  
## $ serum\_cholesterol : int 233 250 204 236 354 192 294 263 199 168 ...  
## $ fasting\_blood\_sugar : int 1 0 0 0 0 0 0 0 1 0 ...  
## $ resting\_ecg\_results : int 0 1 0 1 1 1 0 1 1 1 ...  
## $ max\_heart\_rate : int 150 187 172 178 163 148 153 173 162 174 ...  
## $ exercise\_induced\_angina: int 0 0 0 0 1 0 0 0 0 0 ...  
## $ ST\_depression : num 2.3 3.5 1.4 0.8 0.6 0.4 1.3 0 0.5 1.6 ...  
## $ slope\_peak\_ST\_segment : int 0 0 2 2 2 1 1 2 2 2 ...  
## $ num\_major\_vessels : int 0 0 0 0 0 0 0 0 0 0 ...  
## $ thalium\_stress\_test : int 1 2 2 2 2 1 2 3 3 2 ...  
## $ heart\_disease\_status : int 1 1 1 1 1 1 1 1 1 1 ...

#first few rows of dataset  
head(heart)

## age gender chest\_pain\_type resting\_blood\_pressure serum\_cholesterol  
## 1 63 1 3 145 233  
## 2 37 1 2 130 250  
## 3 41 0 1 130 204  
## 4 56 1 1 120 236  
## 5 57 0 0 120 354  
## 6 57 1 0 140 192  
## fasting\_blood\_sugar resting\_ecg\_results max\_heart\_rate  
## 1 1 0 150  
## 2 0 1 187  
## 3 0 0 172  
## 4 0 1 178  
## 5 0 1 163  
## 6 0 1 148  
## exercise\_induced\_angina ST\_depression slope\_peak\_ST\_segment num\_major\_vessels  
## 1 0 2.3 0 0  
## 2 0 3.5 0 0  
## 3 0 1.4 2 0  
## 4 0 0.8 2 0  
## 5 1 0.6 2 0  
## 6 0 0.4 1 0  
## thalium\_stress\_test heart\_disease\_status  
## 1 1 1  
## 2 2 1  
## 3 2 1  
## 4 2 1  
## 5 2 1  
## 6 1 1

## Data Preprocessing

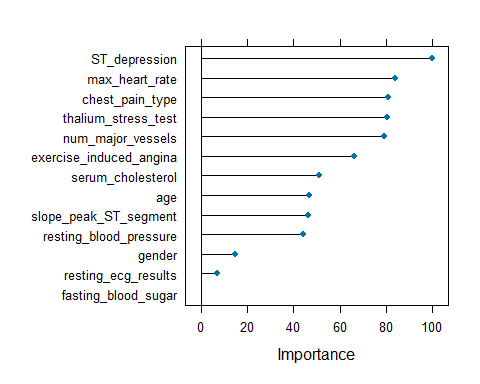
## age gender chest\_pain\_type   
## 0 0 0   
## resting\_blood\_pressure serum\_cholesterol fasting\_blood\_sugar   
## 0 0 0   
## resting\_ecg\_results max\_heart\_rate exercise\_induced\_angina   
## 0 0 0   
## ST\_depression slope\_peak\_ST\_segment num\_major\_vessels   
## 0 0 0   
## thalium\_stress\_test heart\_disease\_status   
## 0 0

#Bagging

cvcontrol <- trainControl(method="repeatedcv",   
 number = 5,  
 repeats = 1,  
 allowParallel=TRUE)  
  
  
set.seed(222)  
  
bag <- train(heart\_disease\_status ~ .,   
 data=train,  
 method="treebag",  
 trControl=cvcontrol,  
 importance=TRUE)  
  
  
bag

## Bagged CART   
##   
## 240 samples  
## 13 predictor  
## 2 classes: '0', '1'   
##   
## No pre-processing  
## Resampling: Cross-Validated (5 fold, repeated 1 times)   
## Summary of sample sizes: 192, 192, 191, 192, 193   
## Resampling results:  
##   
## Accuracy Kappa   
## 0.7961373 0.5866441

plot(varImp(bag))



p <- predict(bag, test, type = 'raw')  
confusionMatrix(p, test$heart\_disease\_status)

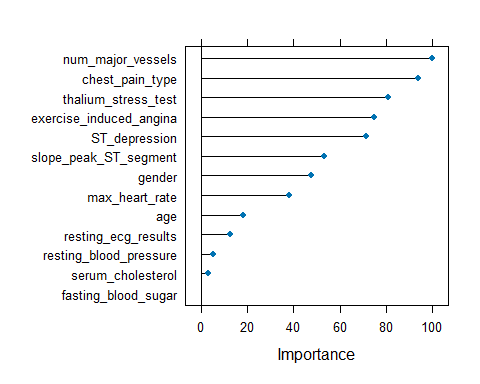
## Confusion Matrix and Statistics  
##   
## Reference  
## Prediction 0 1  
## 0 18 5  
## 1 7 33  
##   
## Accuracy : 0.8095   
## 95% CI : (0.6909, 0.8975)  
## No Information Rate : 0.6032   
## P-Value [Acc > NIR] : 0.000388   
##   
## Kappa : 0.5966   
##   
## Mcnemar's Test P-Value : 0.772830   
##   
## Sensitivity : 0.7200   
## Specificity : 0.8684   
## Pos Pred Value : 0.7826   
## Neg Pred Value : 0.8250   
## Prevalence : 0.3968   
## Detection Rate : 0.2857   
## Detection Prevalence : 0.3651   
## Balanced Accuracy : 0.7942   
##   
## 'Positive' Class : 0   
##

#Random Forest Prediction

set.seed(222)  
forest <- train(heart\_disease\_status ~ . ,   
 data=train,  
 method="rf",  
 trControl=cvcontrol,  
 importance=TRUE)  
  
forest

## Random Forest   
##   
## 240 samples  
## 13 predictor  
## 2 classes: '0', '1'   
##   
## No pre-processing  
## Resampling: Cross-Validated (5 fold, repeated 1 times)   
## Summary of sample sizes: 192, 192, 191, 192, 193   
## Resampling results across tuning parameters:  
##   
## mtry Accuracy Kappa   
## 2 0.8254813 0.6472124  
## 7 0.8047330 0.6048100  
## 13 0.8045593 0.6043473  
##   
## Accuracy was used to select the optimal model using the largest value.  
## The final value used for the model was mtry = 2.

plot(varImp(forest))



p <- predict(forest, test, type = 'raw')  
confusionMatrix(p, test$heart\_disease\_status)

## Confusion Matrix and Statistics  
##   
## Reference  
## Prediction 0 1  
## 0 18 5  
## 1 7 33  
##   
## Accuracy : 0.8095   
## 95% CI : (0.6909, 0.8975)  
## No Information Rate : 0.6032   
## P-Value [Acc > NIR] : 0.000388   
##   
## Kappa : 0.5966   
##   
## Mcnemar's Test P-Value : 0.772830   
##   
## Sensitivity : 0.7200   
## Specificity : 0.8684   
## Pos Pred Value : 0.7826   
## Neg Pred Value : 0.8250   
## Prevalence : 0.3968   
## Detection Rate : 0.2857   
## Detection Prevalence : 0.3651   
## Balanced Accuracy : 0.7942   
##   
## 'Positive' Class : 0   
##

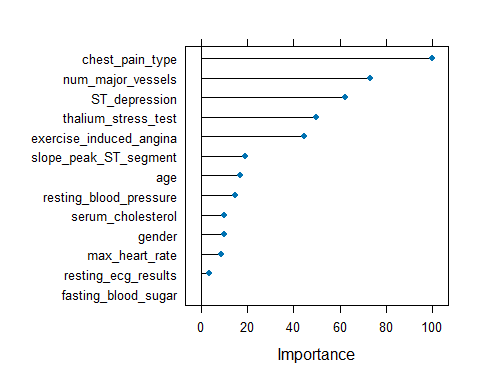
#Extreme Gradient Boosting

set.seed(222)  
xgb\_model <- train(heart\_disease\_status ~ . ,   
 data = train,  
 method = "xgbTree",  
 trControl = cvcontrol,  
 importance = TRUE)

xgb\_model

## eXtreme Gradient Boosting   
##   
## 240 samples  
## 13 predictor  
## 2 classes: '0', '1'   
##   
## No pre-processing  
## Resampling: Cross-Validated (5 fold, repeated 1 times)   
## Summary of sample sizes: 192, 192, 191, 192, 193   
## Resampling results across tuning parameters:  
##   
## eta max\_depth colsample\_bytree subsample nrounds Accuracy Kappa   
## 0.3 1 0.6 0.50 50 0.8501447 0.6983073  
## 0.3 1 0.6 0.50 100 0.8421515 0.6812791  
## 0.3 1 0.6 0.50 150 0.8213146 0.6403158  
## 0.3 1 0.6 0.75 50 0.8504885 0.6971833  
## 0.3 1 0.6 0.75 100 0.8381586 0.6723385  
## 0.3 1 0.6 0.75 150 0.8255663 0.6473444  
## 0.3 1 0.6 1.00 50 0.8463218 0.6889466  
## 0.3 1 0.6 1.00 100 0.8380735 0.6719316  
## 0.3 1 0.6 1.00 150 0.8381586 0.6723385  
## 0.3 1 0.8 0.50 50 0.8295629 0.6557774  
## 0.3 1 0.8 0.50 100 0.8213146 0.6388288  
## 0.3 1 0.8 0.50 150 0.8005663 0.5965399  
## 0.3 1 0.8 0.75 50 0.8548288 0.7064612  
## 0.3 1 0.8 0.75 100 0.8339955 0.6647745  
## 0.3 1 0.8 0.75 150 0.8213146 0.6387701  
## 0.3 1 0.8 1.00 50 0.8545701 0.7057164  
## 0.3 1 0.8 1.00 100 0.8379849 0.6721808  
## 0.3 1 0.8 1.00 150 0.8380699 0.6726271  
## 0.3 2 0.6 0.50 50 0.8008214 0.5975106  
## 0.3 2 0.6 0.50 100 0.7964810 0.5885289  
## 0.3 2 0.6 0.50 150 0.7963074 0.5873684  
## 0.3 2 0.6 0.75 50 0.8214032 0.6382412  
## 0.3 2 0.6 0.75 100 0.7922293 0.5795799  
## 0.3 2 0.6 0.75 150 0.7838924 0.5633719  
## 0.3 2 0.6 1.00 50 0.7961373 0.5880188  
## 0.3 2 0.6 1.00 100 0.7792897 0.5532040  
## 0.3 2 0.6 1.00 150 0.7793783 0.5539932  
## 0.3 2 0.8 0.50 50 0.8168856 0.6297248  
## 0.3 2 0.8 0.50 100 0.8084672 0.6129695  
## 0.3 2 0.8 0.50 150 0.7794670 0.5552477  
## 0.3 2 0.8 0.75 50 0.8170593 0.6303621  
## 0.3 2 0.8 0.75 100 0.7963074 0.5880058  
## 0.3 2 0.8 0.75 150 0.7796334 0.5554786  
## 0.3 2 0.8 1.00 50 0.7919706 0.5791513  
## 0.3 2 0.8 1.00 100 0.7795520 0.5558563  
## 0.3 2 0.8 1.00 150 0.7835450 0.5632966  
## 0.3 3 0.6 0.50 50 0.7837187 0.5635294  
## 0.3 3 0.6 0.50 100 0.7838924 0.5638986  
## 0.3 3 0.6 0.50 150 0.7838924 0.5641204  
## 0.3 3 0.6 0.75 50 0.8128040 0.6209463  
## 0.3 3 0.6 0.75 100 0.7878003 0.5711771  
## 0.3 3 0.6 0.75 150 0.7835450 0.5627771  
## 0.3 3 0.6 1.00 50 0.7961337 0.5866698  
## 0.3 3 0.6 1.00 100 0.7920557 0.5797511  
## 0.3 3 0.6 1.00 150 0.7920557 0.5799329  
## 0.3 3 0.8 0.50 50 0.8089883 0.6138800  
## 0.3 3 0.8 0.50 100 0.8128076 0.6221391  
## 0.3 3 0.8 0.50 150 0.8004776 0.5968127  
## 0.3 3 0.8 0.75 50 0.7835450 0.5612261  
## 0.3 3 0.8 0.75 100 0.7917933 0.5788874  
## 0.3 3 0.8 0.75 150 0.7876266 0.5701647  
## 0.3 3 0.8 1.00 50 0.8046479 0.6045276  
## 0.3 3 0.8 1.00 100 0.7838960 0.5625736  
## 0.3 3 0.8 1.00 150 0.7754740 0.5454528  
## 0.4 1 0.6 0.50 50 0.8421515 0.6803256  
## 0.4 1 0.6 0.50 100 0.8256549 0.6474388  
## 0.4 1 0.6 0.50 150 0.8299066 0.6560431  
## 0.4 1 0.6 0.75 50 0.8504035 0.6974712  
## 0.4 1 0.6 0.75 100 0.8378112 0.6711597  
## 0.4 1 0.6 0.75 150 0.8044706 0.6032778  
## 0.4 1 0.6 1.00 50 0.8464069 0.6885694  
## 0.4 1 0.6 1.00 100 0.8381586 0.6729287  
## 0.4 1 0.6 1.00 150 0.8299066 0.6557134  
## 0.4 1 0.8 0.50 50 0.8338146 0.6637713  
## 0.4 1 0.8 0.50 100 0.8086373 0.6128007  
## 0.4 1 0.8 0.50 150 0.8252189 0.6473509  
## 0.4 1 0.8 0.75 50 0.8423288 0.6802379  
## 0.4 1 0.8 0.75 100 0.8255627 0.6466937  
## 0.4 1 0.8 0.75 150 0.7922293 0.5804219  
## 0.4 1 0.8 1.00 50 0.8461481 0.6889788  
## 0.4 1 0.8 1.00 100 0.8339032 0.6644093  
## 0.4 1 0.8 1.00 150 0.8256549 0.6476365  
## 0.4 2 0.6 0.50 50 0.8378148 0.6733462  
## 0.4 2 0.6 0.50 100 0.8127225 0.6217187  
## 0.4 2 0.6 0.50 150 0.7836300 0.5643623  
## 0.4 2 0.6 0.75 50 0.7964810 0.5878972  
## 0.4 2 0.6 0.75 100 0.7880627 0.5713055  
## 0.4 2 0.6 0.75 150 0.7798108 0.5557346  
## 0.4 2 0.6 1.00 50 0.8088110 0.6127824  
## 0.4 2 0.6 1.00 100 0.7920557 0.5794284  
## 0.4 2 0.6 1.00 150 0.7753854 0.5458177  
## 0.4 2 0.8 0.50 50 0.8127225 0.6209868  
## 0.4 2 0.8 0.50 100 0.8002189 0.5957621  
## 0.4 2 0.8 0.50 150 0.8044706 0.6042439  
## 0.4 2 0.8 0.75 50 0.7963960 0.5885125  
## 0.4 2 0.8 0.75 100 0.7755591 0.5473469  
## 0.4 2 0.8 0.75 150 0.7753817 0.5464465  
## 0.4 2 0.8 1.00 50 0.7751303 0.5448157  
## 0.4 2 0.8 1.00 100 0.7835486 0.5633088  
## 0.4 2 0.8 1.00 150 0.7792047 0.5548619  
## 0.4 3 0.6 0.50 50 0.8046443 0.6052680  
## 0.4 3 0.6 0.50 100 0.7837187 0.5647090  
## 0.4 3 0.6 0.50 150 0.7877967 0.5727448  
## 0.4 3 0.6 0.75 50 0.8001303 0.5967655  
## 0.4 3 0.6 0.75 100 0.7917969 0.5808368  
## 0.4 3 0.6 0.75 150 0.7752117 0.5480049  
## 0.4 3 0.6 1.00 50 0.7838037 0.5632534  
## 0.4 3 0.6 1.00 100 0.7754704 0.5463973  
## 0.4 3 0.6 1.00 150 0.7754704 0.5458957  
## 0.4 3 0.8 0.50 50 0.8044742 0.6049335  
## 0.4 3 0.8 0.50 100 0.8004776 0.5963485  
## 0.4 3 0.8 0.50 150 0.7878890 0.5706801  
## 0.4 3 0.8 0.75 50 0.7879704 0.5735529  
## 0.4 3 0.8 0.75 100 0.7754668 0.5474628  
## 0.4 3 0.8 0.75 150 0.7713001 0.5399004  
## 0.4 3 0.8 1.00 50 0.7754740 0.5466632  
## 0.4 3 0.8 1.00 100 0.7837187 0.5632320  
## 0.4 3 0.8 1.00 150 0.7838037 0.5630617  
##   
## Tuning parameter 'gamma' was held constant at a value of 0  
## Tuning  
## parameter 'min\_child\_weight' was held constant at a value of 1  
## Accuracy was used to select the optimal model using the largest value.  
## The final values used for the model were nrounds = 50, max\_depth = 1, eta  
## = 0.3, gamma = 0, colsample\_bytree = 0.8, min\_child\_weight = 1 and subsample  
## = 0.75.

plot(varImp(xgb\_model))



p <- predict(xgb\_model, test, type = 'raw')  
confusionMatrix(p, test$heart\_disease\_status)

## Confusion Matrix and Statistics  
##   
## Reference  
## Prediction 0 1  
## 0 19 5  
## 1 6 33  
##   
## Accuracy : 0.8254   
## 95% CI : (0.709, 0.9095)  
## No Information Rate : 0.6032   
## P-Value [Acc > NIR] : 0.0001297   
##   
## Kappa : 0.6328   
##   
## Mcnemar's Test P-Value : 1.0000000   
##   
## Sensitivity : 0.7600   
## Specificity : 0.8684   
## Pos Pred Value : 0.7917   
## Neg Pred Value : 0.8462   
## Prevalence : 0.3968   
## Detection Rate : 0.3016   
## Detection Prevalence : 0.3810   
## Balanced Accuracy : 0.8142   
##   
## 'Positive' Class : 0   
##