Project 2- Health-BreastCancer Classification & Prediction

Jayanti Jain

2023-03-13

library(mlbench) #Package with dataset- BreastCancer

## Warning: package 'mlbench' was built under R version 4.2.2

library(tidyverse) #Package for string function

## ── Attaching packages ─────────────────────────────────────── tidyverse 1.3.2 ──  
## ✔ ggplot2 3.4.1 ✔ purrr 1.0.1  
## ✔ tibble 3.1.8 ✔ dplyr 1.1.0  
## ✔ tidyr 1.3.0 ✔ stringr 1.5.0  
## ✔ readr 2.1.3 ✔ forcats 0.5.2

## Warning: package 'tidyr' was built under R version 4.2.2

## Warning: package 'readr' was built under R version 4.2.2

## Warning: package 'purrr' was built under R version 4.2.2

## Warning: package 'dplyr' was built under R version 4.2.2

## Warning: package 'stringr' was built under R version 4.2.2

## ── Conflicts ────────────────────────────────────────── tidyverse\_conflicts() ──  
## ✖ dplyr::filter() masks stats::filter()  
## ✖ dplyr::lag() masks stats::lag()

library(caTools) #Package for splitting the dataset into training and test data

## Warning: package 'caTools' was built under R version 4.2.2

library(caret) #Package for functions for training and plotting models

## Warning: package 'caret' was built under R version 4.2.2

## Loading required package: lattice  
##   
## Attaching package: 'caret'  
##   
## The following object is masked from 'package:purrr':  
##   
## lift

library(mice) #Package for function to remove NA value in dataset

## Warning: package 'mice' was built under R version 4.2.2

##   
## Attaching package: 'mice'  
##   
## The following object is masked from 'package:stats':  
##   
## filter  
##   
## The following objects are masked from 'package:base':  
##   
## cbind, rbind

library(e1071) #Package for function to implement naiveBayes classification algorithm

## Warning: package 'e1071' was built under R version 4.2.2

library(rpart) #Package for function to implement tree algorithm

## Warning: package 'rpart' was built under R version 4.2.2

library(randomForest) #Package for function to implement Random Forest Algorithm

## Warning: package 'randomForest' was built under R version 4.2.2

## randomForest 4.7-1.1  
## 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

library(rpart.plot) #Package for plotting

## Warning: package 'rpart.plot' was built under R version 4.2.2

library(nnet) #Package to implement NN classifiers

## Warning: package 'nnet' was built under R version 4.2.2

#loading & exploring  
#package with breastcancer dataset  
require(mlbench)   
#loading the dataset  
data("BreastCancer")   
#Structure of the dataset  
str(BreastCancer)

## 'data.frame': 699 obs. of 11 variables:  
## $ Id : chr "1000025" "1002945" "1015425" "1016277" ...  
## $ Cl.thickness : Ord.factor w/ 10 levels "1"<"2"<"3"<"4"<..: 5 5 3 6 4 8 1 2 2 4 ...  
## $ Cell.size : Ord.factor w/ 10 levels "1"<"2"<"3"<"4"<..: 1 4 1 8 1 10 1 1 1 2 ...  
## $ Cell.shape : Ord.factor w/ 10 levels "1"<"2"<"3"<"4"<..: 1 4 1 8 1 10 1 2 1 1 ...  
## $ Marg.adhesion : Ord.factor w/ 10 levels "1"<"2"<"3"<"4"<..: 1 5 1 1 3 8 1 1 1 1 ...  
## $ Epith.c.size : Ord.factor w/ 10 levels "1"<"2"<"3"<"4"<..: 2 7 2 3 2 7 2 2 2 2 ...  
## $ Bare.nuclei : Factor w/ 10 levels "1","2","3","4",..: 1 10 2 4 1 10 10 1 1 1 ...  
## $ Bl.cromatin : Factor w/ 10 levels "1","2","3","4",..: 3 3 3 3 3 9 3 3 1 2 ...  
## $ Normal.nucleoli: Factor w/ 10 levels "1","2","3","4",..: 1 2 1 7 1 7 1 1 1 1 ...  
## $ Mitoses : Factor w/ 9 levels "1","2","3","4",..: 1 1 1 1 1 1 1 1 5 1 ...  
## $ Class : Factor w/ 2 levels "benign","malignant": 1 1 1 1 1 2 1 1 1 1 ...

#Finding the levels of target class  
levels(BreastCancer$Class)

## [1] "benign" "malignant"

#Summary of Dataset  
summary(BreastCancer)

## Id Cl.thickness Cell.size Cell.shape Marg.adhesion  
## Length:699 1 :145 1 :384 1 :353 1 :407   
## Class :character 5 :130 10 : 67 2 : 59 2 : 58   
## Mode :character 3 :108 3 : 52 10 : 58 3 : 58   
## 4 : 80 2 : 45 3 : 56 10 : 55   
## 10 : 69 4 : 40 4 : 44 4 : 33   
## 2 : 50 5 : 30 5 : 34 8 : 25   
## (Other):117 (Other): 81 (Other): 95 (Other): 63   
## Epith.c.size Bare.nuclei Bl.cromatin Normal.nucleoli Mitoses   
## 2 :386 1 :402 2 :166 1 :443 1 :579   
## 3 : 72 10 :132 3 :165 10 : 61 2 : 35   
## 4 : 48 2 : 30 1 :152 3 : 44 3 : 33   
## 1 : 47 5 : 30 7 : 73 2 : 36 10 : 14   
## 6 : 41 3 : 28 4 : 40 8 : 24 4 : 12   
## 5 : 39 (Other): 61 5 : 34 6 : 22 7 : 9   
## (Other): 66 NA's : 16 (Other): 69 (Other): 69 (Other): 17   
## Class   
## benign :458   
## malignant:241   
##   
##   
##   
##   
##

# class of each variables  
sapply(BreastCancer, function(x) class(x)[1])

## Id Cl.thickness Cell.size Cell.shape Marg.adhesion   
## "character" "ordered" "ordered" "ordered" "ordered"   
## Epith.c.size Bare.nuclei Bl.cromatin Normal.nucleoli Mitoses   
## "ordered" "factor" "factor" "factor" "factor"   
## Class   
## "factor"

#Cleaning the data  
#Removing NA values and ID(1st column) from dataset using library mice  
dataset\_impute <- mice(BreastCancer[,2:10], print = FALSE)   
#Adding Target class to the imputed dataset without NA  
BreastCancer <- cbind(BreastCancer[,11, drop = FALSE], mice::complete(dataset\_impute, 1))  
#Summary of the cleaned Dataset  
summary(BreastCancer)

## Class Cl.thickness Cell.size Cell.shape Marg.adhesion  
## benign :458 1 :145 1 :384 1 :353 1 :407   
## malignant:241 5 :130 10 : 67 2 : 59 2 : 58   
## 3 :108 3 : 52 10 : 58 3 : 58   
## 4 : 80 2 : 45 3 : 56 10 : 55   
## 10 : 69 4 : 40 4 : 44 4 : 33   
## 2 : 50 5 : 30 5 : 34 8 : 25   
## (Other):117 (Other): 81 (Other): 95 (Other): 63   
## Epith.c.size Bare.nuclei Bl.cromatin Normal.nucleoli Mitoses   
## 2 :386 1 :412 2 :166 1 :443 1 :579   
## 3 : 72 10 :133 3 :165 10 : 61 2 : 35   
## 4 : 48 2 : 31 1 :152 3 : 44 3 : 33   
## 1 : 47 5 : 30 7 : 73 2 : 36 10 : 14   
## 6 : 41 3 : 29 4 : 40 8 : 24 4 : 12   
## 5 : 39 8 : 21 5 : 34 6 : 22 7 : 9   
## (Other): 66 (Other): 43 (Other): 69 (Other): 69 (Other): 17

# Create 70% training and 30% validation data  
  
set.seed(120)   
# Splitting data into training and test dataset  
split=sample.split(BreastCancer, SplitRatio = 0.7)  
# Training dataset  
training\_set=subset(BreastCancer,split==TRUE)  
# Test dataset  
test\_set=subset(BreastCancer,split==FALSE)   
# Dimenstions of training dataset  
dim(training\_set)

## [1] 490 10

# Dimesnions of test dataset  
dim(test\_set)

## [1] 209 10

# Removing target class  
topredict\_set<-test\_set[2:10]   
dim(topredict\_set)

## [1] 209 9

#Naive Bayes Classification  
#Implementing NaiveBayes   
model\_naive<- naiveBayes(Class ~ ., data = training\_set)   
#Predicting target class for the Validation set  
preds\_naive <- predict(model\_naive, newdata = topredict\_set)   
(conf\_matrix\_naive <- table(preds\_naive, test\_set$Class))

##   
## preds\_naive benign malignant  
## benign 127 1  
## malignant 4 77

#Confusion matrix for finding Accuracy of the model  
confusionMatrix(conf\_matrix\_naive)

## Confusion Matrix and Statistics  
##   
##   
## preds\_naive benign malignant  
## benign 127 1  
## malignant 4 77  
##   
## Accuracy : 0.9761   
## 95% CI : (0.9451, 0.9922)  
## No Information Rate : 0.6268   
## P-Value [Acc > NIR] : <2e-16   
##   
## Kappa : 0.9493   
##   
## Mcnemar's Test P-Value : 0.3711   
##   
## Sensitivity : 0.9695   
## Specificity : 0.9872   
## Pos Pred Value : 0.9922   
## Neg Pred Value : 0.9506   
## Prevalence : 0.6268   
## Detection Rate : 0.6077   
## Detection Prevalence : 0.6124   
## Balanced Accuracy : 0.9783   
##   
## 'Positive' Class : benign   
##

#Randomforest classifier  
# Implementing RandomForest  
model\_rf <- randomForest(Class ~ ., data = training\_set, importance=TRUE, ntree = 5)   
#Predicting target class for the Validation set  
preds\_rf <- predict(model\_rf, topredict\_set)   
(conf\_matrix\_forest <- table(preds\_rf, test\_set$Class))

##   
## preds\_rf benign malignant  
## benign 123 4  
## malignant 8 74

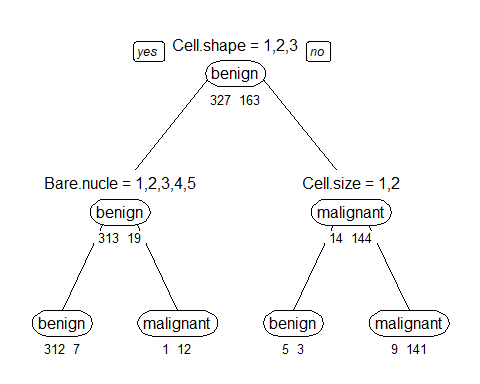
#Confusion matrix for finding Accuracy of the model  
confusionMatrix(conf\_matrix\_forest)

## Confusion Matrix and Statistics  
##   
##   
## preds\_rf benign malignant  
## benign 123 4  
## malignant 8 74  
##   
## Accuracy : 0.9426   
## 95% CI : (0.9019, 0.97)  
## No Information Rate : 0.6268   
## P-Value [Acc > NIR] : <2e-16   
##   
## Kappa : 0.8785   
##   
## Mcnemar's Test P-Value : 0.3865   
##   
## Sensitivity : 0.9389   
## Specificity : 0.9487   
## Pos Pred Value : 0.9685   
## Neg Pred Value : 0.9024   
## Prevalence : 0.6268   
## Detection Rate : 0.5885   
## Detection Prevalence : 0.6077   
## Balanced Accuracy : 0.9438   
##   
## 'Positive' Class : benign   
##

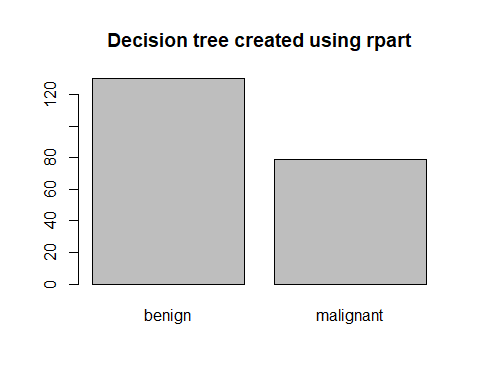
#Decision tree  
#Implementing Decision Tree  
model\_dtree<- rpart(Class ~ ., data=training\_set)   
#Predicting target class for the Validation set  
preds\_dtree <- predict(model\_dtree,newdata=topredict\_set, type = "class")  
(conf\_matrix\_dtree <- table(preds\_dtree, test\_set$Class))

##   
## preds\_dtree benign malignant  
## benign 124 6  
## malignant 7 72

# plot tree  
prp(model\_dtree, type = 1, extra = 1, under = TRUE, split.font = 1, varlen = -10)



plot(preds\_dtree, main="Decision tree created using rpart")



#plot(model\_dtree, main="Decision tree created using rpart")  
  
#Confusion matrix for finding Accuracy of the model  
confusionMatrix(conf\_matrix\_dtree)

## Confusion Matrix and Statistics  
##   
##   
## preds\_dtree benign malignant  
## benign 124 6  
## malignant 7 72  
##   
## Accuracy : 0.9378   
## 95% CI : (0.896, 0.9665)  
## No Information Rate : 0.6268   
## P-Value [Acc > NIR] : <2e-16   
##   
## Kappa : 0.8674   
##   
## Mcnemar's Test P-Value : 1   
##   
## Sensitivity : 0.9466   
## Specificity : 0.9231   
## Pos Pred Value : 0.9538   
## Neg Pred Value : 0.9114   
## Prevalence : 0.6268   
## Detection Rate : 0.5933   
## Detection Prevalence : 0.6220   
## Balanced Accuracy : 0.9348   
##   
## 'Positive' Class : benign   
##

#Neuralnet classifier  
#Implementing Nnet Classifier  
nn\_ntree<- nnet(Class ~ ., data=training\_set, size=1)

## # weights: 83  
## initial value 425.566544   
## iter 10 value 18.067294  
## iter 20 value 8.724712  
## iter 30 value 7.241015  
## iter 40 value 6.823543  
## iter 50 value 6.796592  
## iter 60 value 6.792872  
## iter 70 value 6.792070  
## iter 80 value 6.791784  
## iter 90 value 6.791630  
## iter 100 value 6.791588  
## final value 6.791588   
## stopped after 100 iterations

#Predicting target class for the Validation set  
preds\_ntree <- predict(nn\_ntree,newdata=topredict\_set, type = "class")  
(conf\_matrix\_ntree <- table(preds\_ntree, test\_set$Class))

##   
## preds\_ntree benign malignant  
## benign 125 17  
## malignant 6 61

#Confusion matrix for finding Accuracy of the model  
confusionMatrix(conf\_matrix\_ntree)

## Confusion Matrix and Statistics  
##   
##   
## preds\_ntree benign malignant  
## benign 125 17  
## malignant 6 61  
##   
## Accuracy : 0.89   
## 95% CI : (0.8395, 0.9289)  
## No Information Rate : 0.6268   
## P-Value [Acc > NIR] : < 2e-16   
##   
## Kappa : 0.7579   
##   
## Mcnemar's Test P-Value : 0.03706   
##   
## Sensitivity : 0.9542   
## Specificity : 0.7821   
## Pos Pred Value : 0.8803   
## Neg Pred Value : 0.9104   
## Prevalence : 0.6268   
## Detection Rate : 0.5981   
## Detection Prevalence : 0.6794   
## Balanced Accuracy : 0.8681   
##   
## 'Positive' Class : benign   
##

# create model using conditional inference trees  
  
require(party)

## Loading required package: party

## Loading required package: grid

## Loading required package: mvtnorm

## Loading required package: modeltools

## Loading required package: stats4

## Loading required package: strucchange

## Warning: package 'strucchange' was built under R version 4.2.2

## Loading required package: zoo

##   
## Attaching package: 'zoo'

## The following objects are masked from 'package:base':  
##   
## as.Date, as.Date.numeric

## Loading required package: sandwich

## Warning: package 'sandwich' was built under R version 4.2.2

##   
## Attaching package: 'strucchange'

## The following object is masked from 'package:stringr':  
##   
## boundary

##   
## Attaching package: 'party'

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

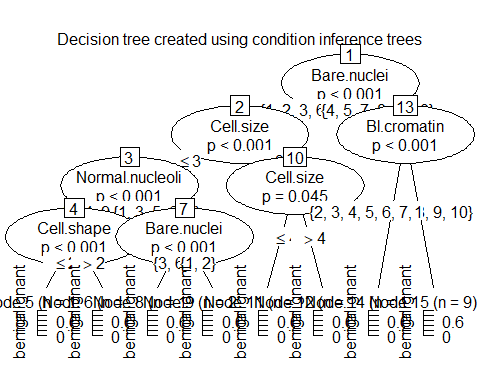
model\_ct <- ctree(Class ~ ., data=training\_set)  
x.ct.pred <- predict(model\_ct, newdata=topredict\_set)  
x.ct.prob <- 1- unlist(treeresponse(model\_ct, topredict\_set), use.names=F)[seq(1,nrow(topredict\_set)\*2,2)]  
(conf\_matrix\_ct <- table(x.ct.pred, test\_set$Class))

##   
## x.ct.pred benign malignant  
## benign 124 3  
## malignant 7 75

#Confusion matrix for finding Accuracy of the model  
confusionMatrix(conf\_matrix\_ct)

## Confusion Matrix and Statistics  
##   
##   
## x.ct.pred benign malignant  
## benign 124 3  
## malignant 7 75  
##   
## Accuracy : 0.9522   
## 95% CI : (0.9138, 0.9768)  
## No Information Rate : 0.6268   
## P-Value [Acc > NIR] : <2e-16   
##   
## Kappa : 0.8988   
##   
## Mcnemar's Test P-Value : 0.3428   
##   
## Sensitivity : 0.9466   
## Specificity : 0.9615   
## Pos Pred Value : 0.9764   
## Neg Pred Value : 0.9146   
## Prevalence : 0.6268   
## Detection Rate : 0.5933   
## Detection Prevalence : 0.6077   
## Balanced Accuracy : 0.9541   
##   
## 'Positive' Class : benign   
##

# To view the decision tree, uncomment this line.  
plot(model\_ct, main="Decision tree created using condition inference trees")



## create model using svm (support vector machine)  
  
require(e1071)  
  
# svm requires tuning  
x.svm.tune <- tune(svm, Class~., data = training\_set,  
 ranges = list(gamma = 2^(-8:1), cost = 2^(0:4)),  
 tunecontrol = tune.control(sampling = "fix"))  
# display the tuning results (in text format)  
x.svm.tune

##   
## Parameter tuning of 'svm':  
##   
## - sampling method: fixed training/validation set   
##   
## - best parameters:  
## gamma cost  
## 0.25 1  
##   
## - best performance: 0.01829268

# If the tuning results are on the margin of the parameters (e.g., gamma = 2^-8),   
# then widen the parameters.  
# I manually copied the cost and gamma from console messages above to parameters below.  
x.svm <- svm(Class~., data = training\_set, cost=4, gamma=0.0625, probability = TRUE)  
x.svm.prob <- predict(x.svm, type="prob", newdata=topredict\_set, probability = TRUE)  
(conf\_matrix\_svm <- table(x.svm.prob, test\_set$Class))

##   
## x.svm.prob benign malignant  
## benign 125 4  
## malignant 6 74

#Confusion matrix for finding Accuracy of the model  
confusionMatrix(conf\_matrix\_svm)

## Confusion Matrix and Statistics  
##   
##   
## x.svm.prob benign malignant  
## benign 125 4  
## malignant 6 74  
##   
## Accuracy : 0.9522   
## 95% CI : (0.9138, 0.9768)  
## No Information Rate : 0.6268   
## P-Value [Acc > NIR] : <2e-16   
##   
## Kappa : 0.8983   
##   
## Mcnemar's Test P-Value : 0.7518   
##   
## Sensitivity : 0.9542   
## Specificity : 0.9487   
## Pos Pred Value : 0.9690   
## Neg Pred Value : 0.9250   
## Prevalence : 0.6268   
## Detection Rate : 0.5981   
## Detection Prevalence : 0.6172   
## Balanced Accuracy : 0.9515   
##   
## 'Positive' Class : benign   
##

#combining classifiers  
  
combine.classes<-data.frame(preds\_rf,preds\_dtree,preds\_ntree,x.svm.prob, x.ct.pred, preds\_naive)  
head(combine.classes)

## preds\_rf preds\_dtree preds\_ntree x.svm.prob x.ct.pred preds\_naive  
## 7 malignant malignant benign benign malignant benign  
## 9 benign benign benign benign benign benign  
## 10 benign benign benign benign benign benign  
## 17 benign benign benign benign benign benign  
## 19 malignant malignant malignant malignant malignant malignant  
## 20 benign benign benign benign benign benign

head(preds\_rf)

## 7 9 10 17 19 20   
## malignant benign benign benign malignant benign   
## Levels: benign malignant

#head(myrda.pred)  
combine.classes$preds\_rf<-ifelse(combine.classes$preds\_rf=="benign", 0, 1)  
combine.classes[,2]<-ifelse(combine.classes[,2]=="benign", 0, 1)  
combine.classes[,3]<-ifelse(combine.classes[,3]=="benign", 0, 1)  
combine.classes[,4]<-ifelse(combine.classes[,4]=="benign", 0, 1)  
combine.classes[,5]<-ifelse(combine.classes[,5]=="benign", 0, 1)  
combine.classes[,6]<-ifelse(combine.classes[,6]=="benign", 0, 1)  
str(combine.classes)

## 'data.frame': 209 obs. of 6 variables:  
## $ preds\_rf : num 1 0 0 0 1 0 0 0 0 1 ...  
## $ preds\_dtree: num 1 0 0 0 1 0 0 0 0 1 ...  
## $ preds\_ntree: num 0 0 0 0 1 0 0 0 0 1 ...  
## $ x.svm.prob : num 0 0 0 0 1 0 0 0 0 1 ...  
## $ x.ct.pred : num 1 0 0 0 1 0 0 0 0 1 ...  
## $ preds\_naive: num 0 0 0 0 1 0 0 0 0 1 ...

combine.cl<-combine.classes[, -c(7,8)]  
majority.vote=rowSums(combine.classes[,-c(7,8)])  
head(majority.vote)

## 7 9 10 17 19 20   
## 3 0 0 0 6 0

#combine.classes[,7]<-rowSums(combine.classes[,-c(7,8)])  
# Subset the BreastCancer data frame to only include the rows corresponding to the combine.classes data frame  
breast\_cancer\_subset <- BreastCancer[1:nrow(combine.classes),]  
  
combine.classes[,6]<-ifelse(combine.classes[,6]>=4, "malignant", "benign")  
table(combine.classes[,6], breast\_cancer\_subset$Class)

##   
## benign malignant  
## benign 121 88

#table(combine.classes[,6], BreastCancer$Class)