Project 2 Data Mining Portfolio

Kylea Johnson

2023-03-13

## Import and Setup

#load the mlbench package which has the BreastCancer data set  
require(mlbench)

## Loading required package: mlbench

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

# if you don't have any required package, use the install.packages() command  
# load the data set  
data(BreastCancer)  
ls(BreastCancer)

## [1] "Bare.nuclei" "Bl.cromatin" "Cell.shape" "Cell.size"   
## [5] "Cl.thickness" "Class" "Epith.c.size" "Id"   
## [9] "Marg.adhesion" "Mitoses" "Normal.nucleoli"

# some algorithms don't like missing values, so remove rows with missing values  
BreastCancer <- na.omit(BreastCancer)   
# remove the unique identifier, which is useless and would confuse the machine learning algorithms  
BreastCancer$Id <- NULL   
head(BreastCancer)

## Cl.thickness Cell.size Cell.shape Marg.adhesion Epith.c.size Bare.nuclei  
## 1 5 1 1 1 2 1  
## 2 5 4 4 5 7 10  
## 3 3 1 1 1 2 2  
## 4 6 8 8 1 3 4  
## 5 4 1 1 3 2 1  
## 6 8 10 10 8 7 10  
## Bl.cromatin Normal.nucleoli Mitoses Class  
## 1 3 1 1 benign  
## 2 3 2 1 benign  
## 3 3 1 1 benign  
## 4 3 7 1 benign  
## 5 3 1 1 benign  
## 6 9 7 1 malignant

str(BreastCancer)

## 'data.frame': 683 obs. of 10 variables:  
## $ 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 ...  
## - attr(\*, "na.action")= 'omit' Named int [1:16] 24 41 140 146 159 165 236 250 276 293 ...  
## ..- attr(\*, "names")= chr [1:16] "24" "41" "140" "146" ...

df2 <- data.frame(sapply(BreastCancer[1:9], function(x) as.numeric(as.character(x))))  
z <- scale(df2[,1:9],center=TRUE,scale=TRUE)  
head(z)

## Cl.thickness Cell.size Cell.shape Marg.adhesion Epith.c.size Bare.nuclei  
## [1,] 0.1977598 -0.7016978 -0.7412304 -0.63889730 -0.5552016 -0.6983413  
## [2,] 0.1977598 0.2770488 0.2625905 0.75747664 1.6939247 1.7715689  
## [3,] -0.5112687 -0.7016978 -0.7412304 -0.63889730 -0.5552016 -0.4239068  
## [4,] 0.5522740 1.5820442 1.6010185 -0.63889730 -0.1053763 0.1249621  
## [5,] -0.1567545 -0.7016978 -0.7412304 0.05928967 -0.5552016 -0.6983413  
## [6,] 1.2613024 2.2345419 2.2702324 1.80475710 1.6939247 1.7715689  
## Bl.cromatin Normal.nucleoli Mitoses  
## [1,] -0.181694 -0.6124785 -0.3481446  
## [2,] -0.181694 -0.2848960 -0.3481446  
## [3,] -0.181694 -0.6124785 -0.3481446  
## [4,] -0.181694 1.3530163 -0.3481446  
## [5,] -0.181694 -0.6124785 -0.3481446  
## [6,] 2.267589 1.3530163 -0.3481446

library(e1071)

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

library(nnet)  
library(neuralnet)

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

library(caret)

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

## Loading required package: ggplot2

## Warning: package 'ggplot2' was built under R version 4.2.2

## Loading required package: lattice

library(gains)  
library(forecast)

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

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

library(rpart)

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

library(rpart.plot)

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

library(dplyr)

##   
## Attaching package: 'dplyr'

## The following object is masked from 'package:neuralnet':  
##   
## compute

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

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

library(ggplot2)  
library(readxl)  
library(plyr)

## ------------------------------------------------------------------------------

## You have loaded plyr after dplyr - this is likely to cause problems.  
## If you need functions from both plyr and dplyr, please load plyr first, then dplyr:  
## library(plyr); library(dplyr)

## ------------------------------------------------------------------------------

##   
## Attaching package: 'plyr'

## The following objects are masked from 'package:dplyr':  
##   
## arrange, count, desc, failwith, id, mutate, rename, summarise,  
## summarize

library(tidyverse)

## ── Attaching packages  
## ───────────────────────────────────────  
## tidyverse 1.3.2 ──

## ✔ tibble 3.1.8 ✔ purrr 0.3.4  
## ✔ tidyr 1.2.1 ✔ stringr 1.4.1  
## ✔ readr 2.1.2 ✔ forcats 0.5.2  
## ── Conflicts ────────────────────────────────────────── tidyverse\_conflicts() ──  
## ✖ plyr::arrange() masks dplyr::arrange()  
## ✖ purrr::compact() masks plyr::compact()  
## ✖ dplyr::compute() masks neuralnet::compute()  
## ✖ plyr::count() masks dplyr::count()  
## ✖ plyr::failwith() masks dplyr::failwith()  
## ✖ dplyr::filter() masks stats::filter()  
## ✖ plyr::id() masks dplyr::id()  
## ✖ dplyr::lag() masks stats::lag()  
## ✖ purrr::lift() masks caret::lift()  
## ✖ plyr::mutate() masks dplyr::mutate()  
## ✖ plyr::rename() masks dplyr::rename()  
## ✖ plyr::summarise() masks dplyr::summarise()  
## ✖ plyr::summarize() masks dplyr::summarize()

library(car)

## Loading required package: carData  
##   
## Attaching package: 'car'  
##   
## The following object is masked from 'package:purrr':  
##   
## some  
##   
## The following object is masked from 'package:dplyr':  
##   
## recode

#SVM

# SUPPORT VECTOR MACHINE  
  
mysvm <- svm(Class ~ ., BreastCancer)  
mysvm.pred <- predict(mysvm, BreastCancer)  
table(mysvm.pred,BreastCancer$Class)

##   
## mysvm.pred benign malignant  
## benign 431 8  
## malignant 13 231

#NAIVE BAYES

#Naive Bayes  
mynb <- naiveBayes(Class ~ ., BreastCancer)  
## predict probabilities  
mynb.pred <- predict(mynb, BreastCancer)  
## predict class membership  
mynb.pred <- predict(mynb, BreastCancer, type= "class")  
confusionMatrix(mynb.pred, as.factor(BreastCancer$Class))

## Confusion Matrix and Statistics  
##   
## Reference  
## Prediction benign malignant  
## benign 431 3  
## malignant 13 236  
##   
## Accuracy : 0.9766   
## 95% CI : (0.9622, 0.9866)  
## No Information Rate : 0.6501   
## P-Value [Acc > NIR] : < 2e-16   
##   
## Kappa : 0.949   
##   
## Mcnemar's Test P-Value : 0.02445   
##   
## Sensitivity : 0.9707   
## Specificity : 0.9874   
## Pos Pred Value : 0.9931   
## Neg Pred Value : 0.9478   
## Prevalence : 0.6501   
## Detection Rate : 0.6310   
## Detection Prevalence : 0.6354   
## Balanced Accuracy : 0.9791   
##   
## 'Positive' Class : benign   
##

#NEURAL NET

#Neural Net  
str(BreastCancer)

## 'data.frame': 683 obs. of 10 variables:  
## $ 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 ...  
## - attr(\*, "na.action")= 'omit' Named int [1:16] 24 41 140 146 159 165 236 250 276 293 ...  
## ..- attr(\*, "names")= chr [1:16] "24" "41" "140" "146" ...

for (i in c(1:9)){  
BreastCancer[,i] <-(as.numeric(BreastCancer[,i])-min(as.numeric(BreastCancer[,i]))) /  
 (max(as.numeric(BreastCancer[,i]))-min(as.numeric(BreastCancer[,i])))  
}  
mynnet <- neuralnet(Class ~ ., BreastCancer, hidden=c(5,4))  
head(BreastCancer)

## Cl.thickness Cell.size Cell.shape Marg.adhesion Epith.c.size Bare.nuclei  
## 1 0.4444444 0.0000000 0.0000000 0.0000000 0.1111111 0.0000000  
## 2 0.4444444 0.3333333 0.3333333 0.4444444 0.6666667 1.0000000  
## 3 0.2222222 0.0000000 0.0000000 0.0000000 0.1111111 0.1111111  
## 4 0.5555556 0.7777778 0.7777778 0.0000000 0.2222222 0.3333333  
## 5 0.3333333 0.0000000 0.0000000 0.2222222 0.1111111 0.0000000  
## 6 0.7777778 1.0000000 1.0000000 0.7777778 0.6666667 1.0000000  
## Bl.cromatin Normal.nucleoli Mitoses Class  
## 1 0.2222222 0.0000000 0 benign  
## 2 0.2222222 0.1111111 0 benign  
## 3 0.2222222 0.0000000 0 benign  
## 4 0.2222222 0.6666667 0 benign  
## 5 0.2222222 0.0000000 0 benign  
## 6 0.8888889 0.6666667 0 malignant

str(mynnet)

## List of 14  
## $ call : language neuralnet(formula = Class ~ ., data = BreastCancer, hidden = c(5, 4))  
## $ response : logi [1:683, 1:2] TRUE TRUE TRUE TRUE TRUE FALSE ...  
## ..- attr(\*, "dimnames")=List of 2  
## .. ..$ : chr [1:683] "1" "2" "3" "4" ...  
## .. ..$ : chr [1:2] "benign" "malignant"  
## $ covariate : num [1:683, 1:9] 0.444 0.444 0.222 0.556 0.333 ...  
## ..- attr(\*, "dimnames")=List of 2  
## .. ..$ : chr [1:683] "1" "2" "3" "4" ...  
## .. ..$ : chr [1:9] "Cl.thickness" "Cell.size" "Cell.shape" "Marg.adhesion" ...  
## $ model.list :List of 2  
## ..$ response : chr [1:2] "benign" "malignant"  
## ..$ variables: chr [1:9] "Cl.thickness" "Cell.size" "Cell.shape" "Marg.adhesion" ...  
## $ err.fct :function (x, y)   
## ..- attr(\*, "type")= chr "sse"  
## $ act.fct :function (x)   
## ..- attr(\*, "type")= chr "logistic"  
## $ linear.output : logi TRUE  
## $ data :'data.frame': 683 obs. of 10 variables:  
## ..$ Cl.thickness : num [1:683] 0.444 0.444 0.222 0.556 0.333 ...  
## ..$ Cell.size : num [1:683] 0 0.333 0 0.778 0 ...  
## ..$ Cell.shape : num [1:683] 0 0.333 0 0.778 0 ...  
## ..$ Marg.adhesion : num [1:683] 0 0.444 0 0 0.222 ...  
## ..$ Epith.c.size : num [1:683] 0.111 0.667 0.111 0.222 0.111 ...  
## ..$ Bare.nuclei : num [1:683] 0 1 0.111 0.333 0 ...  
## ..$ Bl.cromatin : num [1:683] 0.222 0.222 0.222 0.222 0.222 ...  
## ..$ Normal.nucleoli: num [1:683] 0 0.111 0 0.667 0 ...  
## ..$ Mitoses : num [1:683] 0 0 0 0 0 0 0 0 0.5 0 ...  
## ..$ Class : Factor w/ 2 levels "benign","malignant": 1 1 1 1 1 2 1 1 1 1 ...  
## ..- attr(\*, "na.action")= 'omit' Named int [1:16] 24 41 140 146 159 165 236 250 276 293 ...  
## .. ..- attr(\*, "names")= chr [1:16] "24" "41" "140" "146" ...  
## $ exclude : NULL  
## $ net.result :List of 1  
## ..$ : num [1:683, 1:2] 1 0.987 1 0.991 1 ...  
## .. ..- attr(\*, "dimnames")=List of 2  
## .. .. ..$ : chr [1:683] "1" "2" "3" "4" ...  
## .. .. ..$ : NULL  
## $ weights :List of 1  
## ..$ :List of 3  
## .. ..$ : num [1:10, 1:5] 12.059 -13.883 -0.784 -14.987 -6.903 ...  
## .. ..$ : num [1:6, 1:4] 0.605 89.111 5.4 -0.633 33.978 ...  
## .. ..$ : num [1:5, 1:2] 0.3364 1.0339 -0.988 -0.7607 0.0311 ...  
## $ generalized.weights:List of 1  
## ..$ : num [1:683, 1:18] 5.89 -12.9 2.56 -12.56 1.12 ...  
## .. ..- attr(\*, "dimnames")=List of 2  
## .. .. ..$ : chr [1:683] "1" "2" "3" "4" ...  
## .. .. ..$ : NULL  
## $ startweights :List of 1  
## ..$ :List of 3  
## .. ..$ : num [1:10, 1:5] -0.3565 0.0241 0.0769 -1.3328 -0.0992 ...  
## .. ..$ : num [1:6, 1:4] 0.921 1.636 -0.408 -0.279 0.324 ...  
## .. ..$ : num [1:5, 1:2] 0.522 1.332 -0.416 0.33 0.174 ...  
## $ result.matrix : num [1:87, 1] 3.93 9.73e-03 3.32e+03 1.21e+01 -1.39e+01 ...  
## ..- attr(\*, "dimnames")=List of 2  
## .. ..$ : chr [1:87] "error" "reached.threshold" "steps" "Intercept.to.1layhid1" ...  
## .. ..$ : NULL  
## - attr(\*, "class")= chr "nn"

mynnet.pred <- neuralnet::compute(mynnet, BreastCancer)  
  
#Get the actual classes out  
predicted.class <- apply(mynnet.pred$net.result,1,which.max)-1  
  
mynnet.pred <- predicted.class

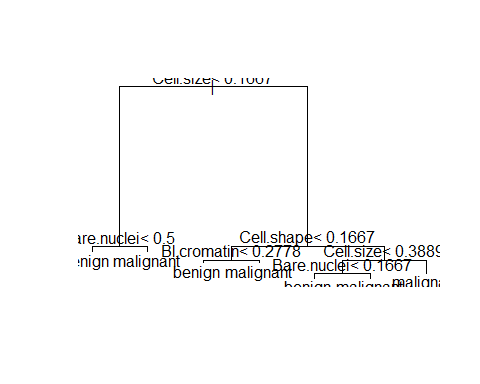
#DECISION TREE

library(MASS)

##   
## Attaching package: 'MASS'

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

#Decision trees  
library(rpart)  
mytree <- rpart(Class ~ ., BreastCancer)  
plot(mytree); text(mytree)



summary(mytree)

## Call:  
## rpart(formula = Class ~ ., data = BreastCancer)  
## n= 683   
##   
## CP nsplit rel error xerror xstd  
## 1 0.79079498 0 1.00000000 1.0000000 0.05215335  
## 2 0.05439331 1 0.20920502 0.2468619 0.03071921  
## 3 0.02510460 2 0.15481172 0.1506276 0.02443403  
## 4 0.01255230 3 0.12970711 0.1506276 0.02443403  
## 5 0.01000000 6 0.09205021 0.1506276 0.02443403  
##   
## Variable importance  
## Cell.size Cell.shape Bare.nuclei Epith.c.size Bl.cromatin   
## 21 18 16 15 15   
## Normal.nucleoli Cl.thickness   
## 14 1   
##   
## Node number 1: 683 observations, complexity param=0.790795  
## predicted class=benign expected loss=0.3499268 P(node) =1  
## class counts: 444 239  
## probabilities: 0.650 0.350   
## left son=2 (418 obs) right son=3 (265 obs)  
## Primary splits:  
## Cell.size < 0.1666667 to the left, improve=222.3221, (0 missing)  
## Cell.shape < 0.2777778 to the left, improve=216.4111, (0 missing)  
## Bare.nuclei < 0.1666667 to the left, improve=203.7284, (0 missing)  
## Bl.cromatin < 0.2777778 to the left, improve=196.3903, (0 missing)  
## Epith.c.size < 0.1666667 to the left, improve=193.1310, (0 missing)  
## Surrogate splits:  
## Cell.shape < 0.2777778 to the left, agree=0.917, adj=0.785, (0 split)  
## Epith.c.size < 0.1666667 to the left, agree=0.900, adj=0.743, (0 split)  
## Bare.nuclei < 0.1666667 to the left, agree=0.880, adj=0.691, (0 split)  
## Normal.nucleoli < 0.1666667 to the left, agree=0.877, adj=0.683, (0 split)  
## Bl.cromatin < 0.2777778 to the left, agree=0.876, adj=0.679, (0 split)  
##   
## Node number 2: 418 observations, complexity param=0.0251046  
## predicted class=benign expected loss=0.02870813 P(node) =0.6120059  
## class counts: 406 12  
## probabilities: 0.971 0.029   
## left son=4 (410 obs) right son=5 (8 obs)  
## Primary splits:  
## Bare.nuclei < 0.5 to the left, improve=11.68296, (0 missing)  
## Cl.thickness < 0.6111111 to the left, improve=10.32214, (0 missing)  
## Normal.nucleoli < 0.2777778 to the left, improve=10.32214, (0 missing)  
## Bl.cromatin < 0.3888889 to the left, improve= 8.53307, (0 missing)  
## Epith.c.size < 0.2777778 to the left, improve= 4.63208, (0 missing)  
## Surrogate splits:  
## Cl.thickness < 0.8333333 to the left, agree=0.988, adj=0.375, (0 split)  
## Normal.nucleoli < 0.2777778 to the left, agree=0.983, adj=0.125, (0 split)  
##   
## Node number 3: 265 observations, complexity param=0.05439331  
## predicted class=malignant expected loss=0.1433962 P(node) =0.3879941  
## class counts: 38 227  
## probabilities: 0.143 0.857   
## left son=6 (23 obs) right son=7 (242 obs)  
## Primary splits:  
## Cell.shape < 0.1666667 to the left, improve=20.58158, (0 missing)  
## Cell.size < 0.2777778 to the left, improve=18.27650, (0 missing)  
## Bare.nuclei < 0.05555556 to the left, improve=16.81493, (0 missing)  
## Bl.cromatin < 0.1666667 to the left, improve=13.91034, (0 missing)  
## Marg.adhesion < 0.1666667 to the left, improve=11.17148, (0 missing)  
## Surrogate splits:  
## Bl.cromatin < 0.05555556 to the left, agree=0.932, adj=0.217, (0 split)  
##   
## Node number 4: 410 observations  
## predicted class=benign expected loss=0.01219512 P(node) =0.6002928  
## class counts: 405 5  
## probabilities: 0.988 0.012   
##   
## Node number 5: 8 observations  
## predicted class=malignant expected loss=0.125 P(node) =0.01171303  
## class counts: 1 7  
## probabilities: 0.125 0.875   
##   
## Node number 6: 23 observations, complexity param=0.0125523  
## predicted class=benign expected loss=0.2173913 P(node) =0.03367496  
## class counts: 18 5  
## probabilities: 0.783 0.217   
## left son=12 (16 obs) right son=13 (7 obs)  
## Primary splits:  
## Bl.cromatin < 0.2777778 to the left, improve=4.968944, (0 missing)  
## Cl.thickness < 0.3888889 to the left, improve=3.381643, (0 missing)  
## Bare.nuclei < 0.05555556 to the left, improve=2.826087, (0 missing)  
## Mitoses < 0.0625 to the left, improve=2.522516, (0 missing)  
## Epith.c.size < 0.1666667 to the left, improve=1.992754, (0 missing)  
## Surrogate splits:  
## Cl.thickness < 0.5 to the left, agree=0.870, adj=0.571, (0 split)  
## Marg.adhesion < 0.6666667 to the left, agree=0.826, adj=0.429, (0 split)  
## Normal.nucleoli < 0.1666667 to the left, agree=0.826, adj=0.429, (0 split)  
## Mitoses < 0.0625 to the left, agree=0.826, adj=0.429, (0 split)  
## Epith.c.size < 0.3333333 to the left, agree=0.783, adj=0.286, (0 split)  
##   
## Node number 7: 242 observations, complexity param=0.0125523  
## predicted class=malignant expected loss=0.08264463 P(node) =0.3543192  
## class counts: 20 222  
## probabilities: 0.083 0.917   
## left son=14 (68 obs) right son=15 (174 obs)  
## Primary splits:  
## Cell.size < 0.3888889 to the left, improve=5.297663, (0 missing)  
## Bare.nuclei < 0.1666667 to the left, improve=4.093695, (0 missing)  
## Cell.shape < 0.3888889 to the left, improve=2.958548, (0 missing)  
## Bl.cromatin < 0.2777778 to the left, improve=2.805426, (0 missing)  
## Marg.adhesion < 0.5 to the left, improve=2.754821, (0 missing)  
## Surrogate splits:  
## Cell.shape < 0.3888889 to the left, agree=0.789, adj=0.250, (0 split)  
## Epith.c.size < 0.1666667 to the left, agree=0.777, adj=0.206, (0 split)  
## Marg.adhesion < 0.05555556 to the left, agree=0.744, adj=0.088, (0 split)  
## Bl.cromatin < 0.1666667 to the left, agree=0.736, adj=0.059, (0 split)  
##   
## Node number 12: 16 observations  
## predicted class=benign expected loss=0 P(node) =0.02342606  
## class counts: 16 0  
## probabilities: 1.000 0.000   
##   
## Node number 13: 7 observations  
## predicted class=malignant expected loss=0.2857143 P(node) =0.0102489  
## class counts: 2 5  
## probabilities: 0.286 0.714   
##   
## Node number 14: 68 observations, complexity param=0.0125523  
## predicted class=malignant expected loss=0.25 P(node) =0.09956076  
## class counts: 17 51  
## probabilities: 0.250 0.750   
## left son=28 (14 obs) right son=29 (54 obs)  
## Primary splits:  
## Bare.nuclei < 0.1666667 to the left, improve=7.600529, (0 missing)  
## Cl.thickness < 0.6111111 to the left, improve=3.558824, (0 missing)  
## Marg.adhesion < 0.5 to the left, improve=2.615385, (0 missing)  
## Normal.nucleoli < 0.1666667 to the left, improve=1.937690, (0 missing)  
## Bl.cromatin < 0.2777778 to the left, improve=1.525641, (0 missing)  
##   
## Node number 15: 174 observations  
## predicted class=malignant expected loss=0.01724138 P(node) =0.2547584  
## class counts: 3 171  
## probabilities: 0.017 0.983   
##   
## Node number 28: 14 observations  
## predicted class=benign expected loss=0.2857143 P(node) =0.0204978  
## class counts: 10 4  
## probabilities: 0.714 0.286   
##   
## Node number 29: 54 observations  
## predicted class=malignant expected loss=0.1296296 P(node) =0.07906296  
## class counts: 7 47  
## probabilities: 0.130 0.870

mytree.pred <- predict(mytree,BreastCancer,type="class")  
table(mytree.pred,BreastCancer$Class)

##   
## mytree.pred benign malignant  
## benign 431 9  
## malignant 13 230

#LOOCV

# Leave-1-Out Cross Validation (LOOCV)  
ans <- numeric(length(BreastCancer[,1]))  
for (i in 1:length(BreastCancer[,1])) {  
 mytree <- rpart(Class ~ ., BreastCancer[-i,])  
 mytree.pred <- predict(mytree,BreastCancer[i,],type="class")  
 ans[i] <- mytree.pred  
}  
ans <- factor(ans,labels=levels(BreastCancer$Class))  
table(ans,BreastCancer$Class)

##   
## ans benign malignant  
## benign 431 24  
## malignant 13 215

#QUADRATIC DISCRIMINANT ANALYSIS

#Quadratic Discriminant Analysis  
library(MASS)  
  
myqda <- qda(Species ~ ., iris)  
myqda.pred <- predict(myqda, iris)  
table(myqda.pred$class,iris$Species)

##   
## setosa versicolor virginica  
## setosa 50 0 0  
## versicolor 0 48 1  
## virginica 0 2 49

myqda <- qda(Class ~ ., BreastCancer)  
myqda.pred <- predict(myqda, BreastCancer)  
head(myqda.pred$class)

## [1] benign malignant benign malignant benign malignant  
## Levels: benign malignant

table(myqda.pred$class,BreastCancer$Class)

##   
## benign malignant  
## benign 422 6  
## malignant 22 233

# RDA

#Regularised Discriminant Analysis  
#install.packages("rda")  
library(rda)

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

#myrda <- rda(Class ~ ., BreastCancer)  
############### add back in if can  
#myrda.pred <- predict(myrda, BreastCancer)  
  
#table(myrda.pred$class,BreastCancer$Class)

# RANDOM FOREST

#Random Forests  
#install.packages("randomForest")  
library(randomForest)

## 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

myrf <- randomForest(Class ~ ., BreastCancer)  
  
  
myrf.pred <- predict(myrf, BreastCancer)  
  
  
head(myrf.pred)

## 1 2 3 4 5 6   
## benign benign benign benign benign malignant   
## Levels: benign malignant

table(myrf.pred, BreastCancer$Class)

##   
## myrf.pred benign malignant  
## benign 444 0  
## malignant 0 239

## COMBINE RESULTS

#Combine all the predictions  
combine.classes<-data.frame(myrf.pred, myqda.pred$class, myqda.pred$class,   
 mytree.pred,mynnet.pred,mysvm.pred, mynb.pred)  
  
##Combine  
head(combine.classes)

## myrf.pred myqda.pred.class myqda.pred.class.1 mytree.pred mynnet.pred  
## 1 benign benign benign malignant 0  
## 2 benign malignant malignant malignant 0  
## 3 benign benign benign malignant 0  
## 4 benign malignant malignant malignant 0  
## 5 benign benign benign malignant 0  
## 6 malignant malignant malignant malignant 1  
## mysvm.pred mynb.pred  
## 1 benign benign  
## 2 malignant malignant  
## 3 benign benign  
## 4 malignant malignant  
## 5 benign benign  
## 6 malignant malignant

head(myrf.pred)

## 1 2 3 4 5 6   
## benign benign benign benign benign malignant   
## Levels: benign malignant

#Convert the naive bayes and random forest results to 1/0  
  
combine.classes$myrf.pred<-ifelse(combine.classes$myrf.pred=="benign", 0, 1)  
combine.classes$mynb.pred<-ifelse(combine.classes$mynb.pred=="benign", 0, 1)  
  
#Convert the other categorical results to numeric  
  
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': 683 obs. of 7 variables:  
## $ myrf.pred : num 0 0 0 0 0 1 0 0 0 0 ...  
## $ myqda.pred.class : num 0 1 0 1 0 1 1 0 1 0 ...  
## $ myqda.pred.class.1: num 0 1 0 1 0 1 1 0 1 0 ...  
## $ mytree.pred : num 1 1 1 1 1 1 1 1 1 1 ...  
## $ mynnet.pred : num 1 1 1 1 1 1 1 1 1 1 ...  
## $ mysvm.pred : num 0 1 0 1 0 1 0 0 0 0 ...  
## $ mynb.pred : num 0 1 0 1 0 1 0 0 0 0 ...

#Finally, use majority vote of >4 to classify

#Conduct Majority Vote  
combine.classes$majority.vote<- rowSums(combine.classes)  
  
combine.classes$class <-ifelse(combine.classes$majority.vote>=4, "malignant", "benign")  
table(BreastCancer$Class, combine.classes$class )

##   
## benign malignant  
## benign 421 23  
## malignant 2 237