

## HarshAgrawal\_HW6.R

Asus

2021-11-15

```
# Course      : CS 513
# First Name  : Harsh
# Last Name   : Agrawal
# CWID        : 10475285

rm(list=ls())

library(C50)

setwd("C:/Users/Asus/Desktop/Github/MSCS/CS-513/HW6")

db = read.csv('breast-cancer-wisconsin.csv', header=TRUE, sep=",")
summary(db)

##      Sample      F1      F2      F3
## Min.   : 61634   Min.   : 1.000   Min.   : 1.000   Min.   : 1.000
## 1st Qu.: 870688   1st Qu.: 2.000   1st Qu.: 1.000   1st Qu.: 1.000
## Median : 1171710   Median : 4.000   Median : 1.000   Median : 1.000
## Mean   : 1071704   Mean    : 4.418   Mean    : 3.134   Mean    : 3.207
## 3rd Qu.: 1238298   3rd Qu.: 6.000   3rd Qu.: 5.000   3rd Qu.: 5.000
## Max.    :13454352   Max.    :10.000   Max.    :10.000   Max.    :10.000
##      F4      F5      F6      F7
## Min.   : 1.000   Min.   : 1.000   Length:699   Min.   : 1.000
## 1st Qu.: 1.000   1st Qu.: 2.000   Class :character   1st Qu.: 2.000
## Median : 1.000   Median : 2.000   Mode  :character   Median : 3.000
## Mean   : 2.807   Mean    : 3.216                   Mean    : 3.438
## 3rd Qu.: 4.000   3rd Qu.: 4.000                   3rd Qu.: 5.000
## Max.    :10.000   Max.    :10.000                   Max.    :10.000
##      F8      F9      Class
## Min.   : 1.000   Min.   : 1.000   Min.    :2.00
## 1st Qu.: 1.000   1st Qu.: 1.000   1st Qu.:2.00
## Median : 1.000   Median : 1.000   Median :2.00
## Mean   : 2.867   Mean    : 1.589   Mean    :2.69
## 3rd Qu.: 4.000   3rd Qu.: 1.000   3rd Qu.:4.00
## Max.    :10.000   Max.    :10.000   Max.    :4.00

db<-na.omit(db)

db$Class <- factor(db$Class, levels = c(2, 4), labels = c("benign",
"malignant"))
```

```

View(db)

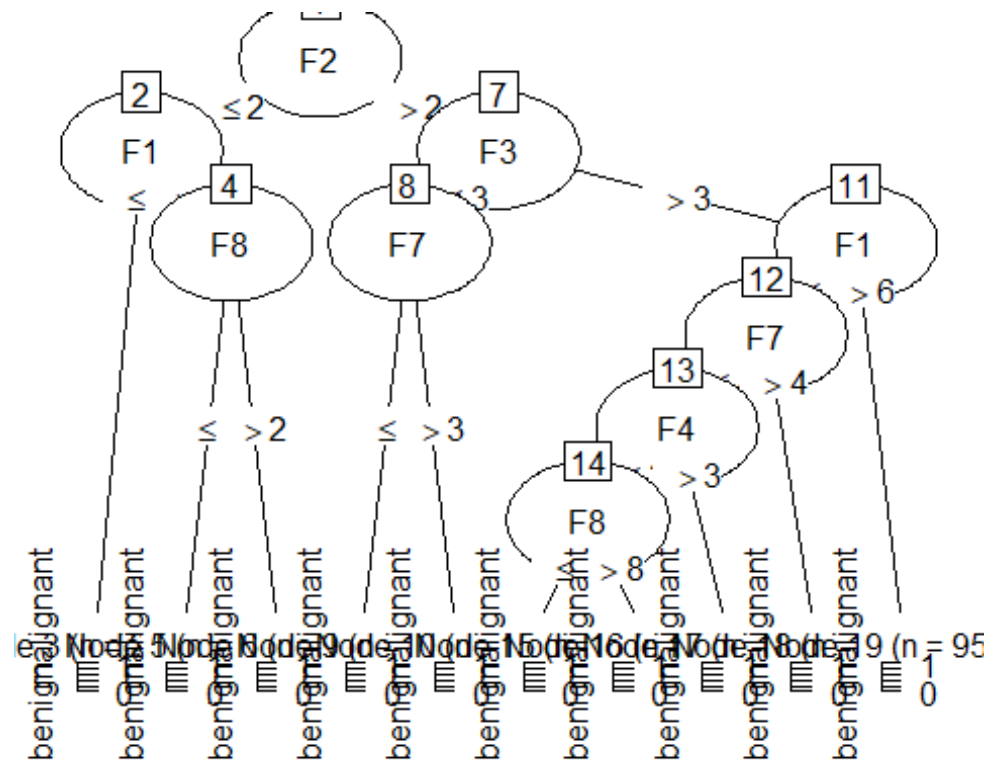
train_index <- sample(nrow(db),as.integer(.70*nrow(db)))
train_data<-db[train_index,]
test_data<-db[-train_index,]

c50<-C5.0(Class~.,train_data[, -1])
summary(c50)

##
## Call:
## C5.0.formula(formula = Class ~ ., data = train_data[, -1])
##
##
## C5.0 [Release 2.07 GPL Edition]      Mon Nov 15 00:03:07 2021
## -----
##
## Class specified by attribute `outcome'
##
## Read 489 cases (10 attributes) from undefined.data
##
## Decision tree:
##
## F2 <= 2:
## :...F1 <= 5: benign (278/2)
## :   F1 > 5:
## :     :...F8 <= 2: benign (9/1)
## :       F8 > 2: malignant (6)
## F2 > 2:
## :...F3 <= 3:
## :   :...F7 <= 3: benign (15/2)
## :     F7 > 3: malignant (16/3)
## :   F3 > 3:
## :     :...F1 > 6: malignant (95)
## :       F1 <= 6:
## :         :...F7 > 4: malignant (48/1)
## :           F7 <= 4:
## :             :...F4 > 3: malignant (10/2)
## :               F4 <= 3:
## :                 :...F8 <= 8: benign (9/1)
## :                   F8 > 8: malignant (3)
##
##
## Evaluation on training data (489 cases):
##
##      Decision Tree
##      -----
##      Size      Errors
##
##      10    12( 2.5%)  <<

```

```
##
##
##      (a)  (b)    <-classified as
##      ----  ----
##      305    6    (a): class benign
##      6    172   (b): class malignant
##
##
## Attribute usage:
##
## 100.00% F2
## 93.66% F1
## 40.08% F3
## 20.65% F7
## 5.52% F8
## 4.50% F4
##
##
## Time: 0.0 secs
plot(c50)
```



```
pred<-predict(c50,test_data[,-1],type="class")
conf_matrix<-table(test_data[,11],pred)
conf_matrix
```

```

##           pred
##           benign malignant
##  benign      142         5
##  malignant    7         56

inc<-sum(test_data[,11]!=pred)
accuracy<- 1 - inc/length(test_data[,11])
accuracy

## [1] 0.9428571

rm(list=ls())

# Random Forest
library('randomForest')

## randomForest 4.6-14

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

setwd("C:/Users/Asus/Desktop/Github/MSCS/CS-513/HW6")

db = read.csv('breast-cancer-wisconsin.csv',header=TRUE, sep=",")
summary(db)

##      Sample      F1      F2      F3
##  Min.   : 61634   Min.   : 1.000   Min.   : 1.000   Min.   : 1.000
##  1st Qu.: 870688   1st Qu.: 2.000   1st Qu.: 1.000   1st Qu.: 1.000
##  Median : 1171710   Median : 4.000   Median : 1.000   Median : 1.000
##  Mean    : 1071704   Mean    : 4.418   Mean    : 3.134   Mean    : 3.207
##  3rd Qu.: 1238298   3rd Qu.: 6.000   3rd Qu.: 5.000   3rd Qu.: 5.000
##  Max.    :13454352   Max.    :10.000   Max.    :10.000   Max.    :10.000
##      F4      F5      F6      F7
##  Min.   : 1.000   Min.   : 1.000   Length:699   Min.   : 1.000
##  1st Qu.: 1.000   1st Qu.: 2.000   Class :character   1st Qu.: 2.000
##  Median : 1.000   Median : 2.000   Mode  :character   Median : 3.000
##  Mean    : 2.807   Mean    : 3.216                   Mean    : 3.438
##  3rd Qu.: 4.000   3rd Qu.: 4.000                   3rd Qu.: 5.000
##  Max.    :10.000   Max.    :10.000                   Max.    :10.000
##      F8      F9      Class
##  Min.   : 1.000   Min.   : 1.000   Min.   :2.00
##  1st Qu.: 1.000   1st Qu.: 1.000   1st Qu.:2.00
##  Median : 1.000   Median : 1.000   Median :2.00
##  Mean    : 2.867   Mean    : 1.589   Mean    :2.69
##  3rd Qu.: 4.000   3rd Qu.: 1.000   3rd Qu.:4.00
##  Max.    :10.000   Max.    :10.000   Max.    :4.00

db<-na.omit(db)

db$Class <- factor(db$Class, levels = c(2, 4), labels = c("benign",
"malignant"))

```

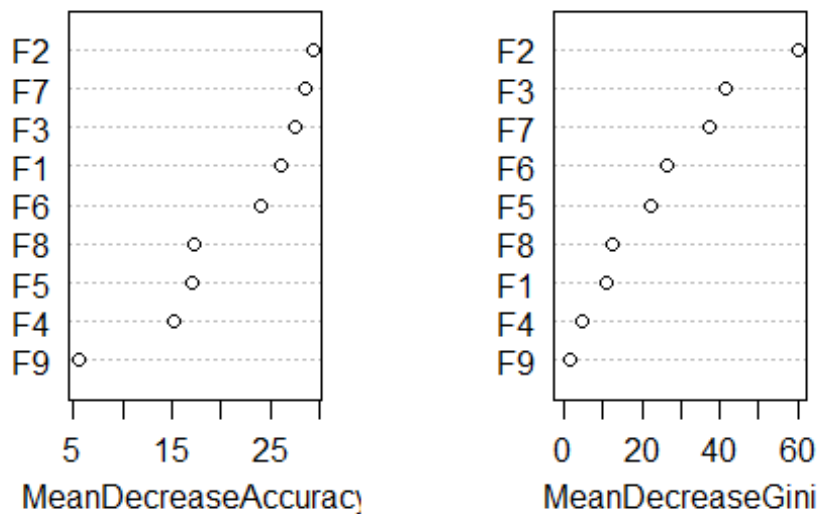
```
db = db[-1]

View(db)

train_index <- sample(nrow(db),as.integer(.70*nrow(db)))
train_data<-db[train_index,]
test_data<-db[-train_index,]

rf<-randomForest(Class~.,data = train_data, importance = TRUE, ntree=1000)
varImpPlot(rf)
```

rf



```
importance(rf)
```

	benign	malignant	MeanDecreaseAccuracy	MeanDecreaseGini
## F1	16.985768	27.3382953	26.016587	11.060658
## F2	20.113069	23.3821169	29.339351	60.020764
## F3	11.000507	24.9943595	27.490966	41.476595
## F4	9.501326	11.9761281	15.137191	4.473377
## F5	12.974604	10.6928908	17.076794	22.289537
## F6	11.424254	23.7728339	24.016485	26.568185
## F7	17.784815	25.8751499	28.541621	37.485781
## F8	17.073066	4.1804225	17.265850	12.286474
## F9	5.991248	0.1703767	5.594091	1.361032

*#Hence F2 is the most important feature followed by F3 and then F7 in the order shown in the graph. F9 is the least important feature.*

```
pred <- predict( rf ,test_data , type="class" )
inc <- sum(test_data[,10] != pred)
accuracy <- 1 - inc/length(test_data[,10])
accuracy

## [1] 0.952381

rm(list=ls())
```