

## Assignment 2

#Importing Libraries

```
library(randomForest)

## randomForest 4.6-14

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

library(caret)

## Loading required package: lattice

## Loading required package: ggplot2

##
## Attaching package: 'ggplot2'

## The following object is masked from 'package:randomForest':
##
##     margin

library(ggplot2)
library(mlbench)
```

#Importing Data

```
data(BreastCancer)
head(BreastCancer)

##      Id Cl.thickness Cell.size Cell.shape Marg.adhesion Epith.c.size
## 1 1000025           5         1         1             1             2
## 2 1002945           5         4         4             5             7
## 3 1015425           3         1         1             1             2
## 4 1016277           6         8         8             1             3
## 5 1017023           4         1         1             3             2
## 6 1017122           8        10        10             8             7
##   Bare.nuclei Bl.cromatin Normal.nucleoli Mitoses      Class
## 1           1           3             1         1    benign
## 2          10           3             2         1    benign
## 3           2           3             1         1    benign
## 4           4           3             7         1    benign
## 5           1           3             1         1    benign
## 6          10           9             7         1 malignant

levels(BreastCancer$Class)

## [1] "benign"      "malignant"
```

#Finding missing values and removing them

```
sum(is.na(BreastCancer))
```

```
## [1] 16
```

```
BreastCancer <- na.omit(BreastCancer)[,c(2:11)]
```

#Splitting the data into 70% training and 30% testing.

```
set.seed(2)
```

```
intrain <- createDataPartition(y = BreastCancer$Class, p = 0.7, list = FALSE)
```

```
training <- BreastCancer[intrain,]
```

```
testing <- BreastCancer[-intrain,]
```

#RF model without grid search.

```
set.seed(2)
```

```
rf.model <- train(Class~., data = training, method = 'rf')
```

```
rf.model
```

```
## Random Forest
```

```
##
```

```
## 479 samples
```

```
## 9 predictor
```

```
## 2 classes: 'benign', 'malignant'
```

```
##
```

```
## No pre-processing
```

```
## Resampling: Bootstrapped (25 reps)
```

```
## Summary of sample sizes: 479, 479, 479, 479, 479, 479, ...
```

```
## Resampling results across tuning parameters:
```

```
##
```

```
## mtry Accuracy Kappa
```

```
## 2 0.9511689 0.8934024
```

```
## 41 0.9432530 0.8754655
```

```
## 80 0.9374226 0.8629657
```

```
##
```

```
## Accuracy was used to select the optimal model using the largest value.
```

```
## The final value used for the model was mtry = 2.
```

#Confusion Matrix:

```
results <- predict(rf.model, testing, type = "raw")
```

```
table(results, testing$Class)
```

```
##
```

```
## results      benign malignant
```

```
## benign       132          3
```

```
## malignant     1         68
```

#Examining the following values for the mtry variable: 2, 6, 8

```
set.seed(2)
```

```
Grid_Serach <- expand.grid(.mtry = c(2,6,8))
```

```
rf.model2 <- train(Class~., data = training, method='rf', tuneGrid =
Grid_Serach)
rf.model2

## Random Forest
##
## 479 samples
## 9 predictor
## 2 classes: 'benign', 'malignant'
##
## No pre-processing
## Resampling: Bootstrapped (25 reps)
## Summary of sample sizes: 479, 479, 479, 479, 479, 479, ...
## Resampling results across tuning parameters:
##
## mtry Accuracy Kappa
## 2 0.9511689 0.8934024
## 6 0.9479830 0.8864116
## 8 0.9448512 0.8794912
##
## Accuracy was used to select the optimal model using the largest value.
## The final value used for the model was mtry = 2.
```

#Probablistic Value:

```
probs <- predict(rf.model2, testing, type="prob")
head(probs)

## benign malignant
## 5 0.938 0.062
## 6 0.042 0.958
## 11 0.952 0.048
## 13 0.706 0.294
## 15 0.090 0.910
## 19 0.028 0.972
```

#Confusion Matrix:

```
pred_class <- predict(rf.model2, testing, type="raw")
head(pred_class)

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

table(pred_class, testing$Class)

##
## pred_class benign malignant
## benign 132 3
## malignant 1 68
```