Final Exam

Maria George

December 14, 2015

```
# Loading all the required libraries
library("dplyr")
library("ggplot2")
library("car") # Contains the scatterplotMatrix function
#install.packages("boot")
library("boot") # Perform crossvalidation
#install.packages("tree")
library("tree")
library("randomForest")
library(pROC) # Useful for computing and plotting classifer metrics
library("ISLR")
# install.packages("gbm")
library(gbm) # To perform boosting
```

The Wisconsin Breast Cancer dataset is available as a comma-delimited text file on the UCI Machine Learning Repository http://archive.ics.uci.edu/ml. Goal of this project is to predict whether observations (i.e. tumors) are malignant or benign.

Loading the data

```
breastCancer.data <-
  read.csv("http://archive.ics.uci.edu/ml/machine-learning-databases/breast-cancer-wisconsin/breast-can
```

Breast Cancer data was obtained from from the University of Wisconsin Hospitals, Madison from Dr. William H. Wolberg.

It records 699 observations of 11 variables. The data contains different varaiables that measures different characteristics of the tissue sample and records it as value between 1 to 10. The variables include:

- Sample code number: Id number of the patient
- Clump Thickness: (1 - 10)
- Uniformity of Cell Size: (1 - 10)
- Uniformity of Cell Shape: (1 - 10)
- Marginal Adhesion: (1 - 10)

- Single Epithelial Cell Size: (1 - 10)
- Bare Nuclei: (1 - 10)

- Bland Chromatin: (1 - 10)
- Normal Nucleoli: (1 - 10)
- Mitoses: (1 - 10)
- Class: 2 for benign, 4 for malignant

Tidying the data, ensuring that each variable is properly named and cast as the correct data type

```r
# Checking the structure of the data
str(breastCancer.data)
```

```
## 'data.frame':    699 obs. of  11 variables:
##  $ V1 : int  1000025 1002945 1015425 1016277 1017023 1017122 1018099 1018561 1033078 1033078 ...
##  $ V2 : int  5 5 3 6 4 8 1 2 2 4 ...
##  $ V3 : int  1 4 1 8 1 10 1 1 1 2 ...
##  $ V4 : int  1 4 1 8 1 10 1 2 1 1 ...
##  $ V5 : int  1 5 1 1 3 8 1 1 1 1 ...
##  $ V6 : int  2 7 2 3 2 7 2 2 2 2 ...
##  $ V7 : Factor w/ 11 levels "?","1","10","2",..: 2 3 4 6 2 3 3 2 2 2 ...
##  $ V8 : int  3 3 3 3 3 9 3 3 1 2 ...
##  $ V9 : int  1 2 1 7 1 7 1 1 1 1 ...
##  $ V10: int  1 1 1 1 1 1 1 1 5 1 ...
##  $ V11: int  2 2 2 2 2 4 2 2 2 2 ...
```

```r
# Renaming the column names
colnames(breastCancer.data) <- c("sample_code_number", "clump_thickness",
                                 "uniformity_of_cell_size",
                                 "uniformity_of_cell_shape",
                                 "marginal_adhesion",
                                 "single_epithelial_cell_size",
                                 "bare_nuclei", "bland_chromatin",
                                 "normal_nucleoli", "mitoses", "class")

# Checking for Missing Data
breastCancer.data[breastCancer.data == "?"] <- NA

# Displaying the number of rows containing missing values
sum(is.na(breastCancer.data))
```

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

```r
# Removing the rows containing missing values
breastCancer.data <- na.omit(breastCancer.data)

# Data type of column 7(V7 renamed to bare_nuclei) was recorded as factor variable
# Recasting it into integer datatype
breastCancer.data$bare_nuclei <- as.integer(breastCancer.data$bare_nuclei)

# Data type of class was recorded as integer variable
# Recasting it into factor datatype
breastCancer.data$class <- as.factor(breastCancer.data$class)

# Checking for duplicate rows
length(unique(breastCancer.data$sample_code_number))
```

```
## [1] 630
```

```r
# Removing duplicate rows
breastCancer.data <- breastCancer.data[!duplicated(breastCancer.data[,1]), ]
```

Steps followed to tidy the breastCancer.data dataset: 1. All the columns were properly renamed

2. Removed Missing values: There were 16 observations that had missing values denoted by "?". There were 16 instances in Groups 1 to 6 that contain a single missing (i.e., unavailable) attribute value. The rows containing the missing values were removed.

3. While observing the structure of the dataset, the column bare_nuclei was found to be recorded as factor variable. the column was recasted to integer datatype.Also, the column class was of integer datype, this was converted to a factor variable as it has only 2 states.

4. The first column, sample_code_number represents unique number of the patient. On inspection, it was found that the column was not unique and there was multiple entries for the same patient. Since the class of the patient (beningn, malignant) were same in the duplicate rows for all the paients, the duplicate rows were removed.

The tidied dataset contains 630 observations of 11 variables.

Splitting the data into a training and validation set such that a random 70% of the observations are in the training set.

```r
# Counting the number of rows in the dataset
rowcount <- nrow(breastCancer.data)
# Setting a random seed so that results can be reproduced
set.seed(1)

# Defining the indices of the training dataset
# 70% of the observations are in training set
train <- sample(rowcount, as.integer(0.7*rowcount))
# Calculating the length of training dataset
length(train)
```

```
## [1] 441
```

```r
# Defining the training datasset based on the indices obatined
training.Data <- breastCancer.data[train, ]
# Displaying the number of observations in training dataset
nrow(training.Data)
```

```
## [1] 441
```

```r
# Defining the testing dataset
test.Data <- breastCancer.data[-train, ]
# Displaying the number of observations in training dataset
nrow(test.Data)
```

```
## [1] 189
```

The sample() function is used to randomly calculate the indices for training data. 70% of the observations are used as training data. The original data records 630 observations of 11 variables. Training data contains 441 observations. Testing data contains 189 observations.

Fitting a regression model to predict whether tissue samples are malignant or benign. Classifying cases in the validation set.

```r
# Fitting a logictic regression model to predict whether tissue samples are
# malignant or benign
glm.breastCancer.fit <- glm(class ~ clump_thickness + uniformity_of_cell_size   +
                              uniformity_of_cell_shape + marginal_adhesion +
                              single_epithelial_cell_size + bare_nuclei +
                              bland_chromatin + normal_nucleoli +   mitoses,
                            training.Data, family = binomial)

# Displaying the summary statistics of the fitted model
summary(glm.breastCancer.fit)
```

```
##
## Call:
## glm(formula = class ~ clump_thickness + uniformity_of_cell_size +
##     uniformity_of_cell_shape + marginal_adhesion + single_epithelial_cell_size +
##     bare_nuclei + bland_chromatin + normal_nucleoli + mitoses,
##     family = binomial, data = training.Data)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -3.9827  -0.1218  -0.0442   0.0160   2.1094
##
## Coefficients:
##                               Estimate Std. Error z value Pr(>|z|)
## (Intercept)                  -12.17409    1.83386  -6.639 3.17e-11 ***
## clump_thickness                0.71274    0.17729   4.020 5.81e-05 ***
## uniformity_of_cell_size        0.04123    0.30890   0.133  0.89382
## uniformity_of_cell_shape       0.31045    0.30902   1.005  0.31508
## marginal_adhesion              0.60124    0.21987   2.735  0.00625 **
## single_epithelial_cell_size    0.39974    0.19995   1.999  0.04559 *
## bare_nuclei                    0.45769    0.16758   2.731  0.00631 **
## bland_chromatin                0.41113    0.21542   1.909  0.05632 .
## normal_nucleoli                0.02176    0.13869   0.157  0.87535
## mitoses                        0.60976    0.40271   1.514  0.12999
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 577.727  on 440  degrees of freedom
## Residual deviance:  71.065  on 431  degrees of freedom
## AIC: 91.065
##
## Number of Fisher Scoring iterations: 8
```

From the logistic regression model fitted to predict whether tissue samples are malignant or benign, we can see that the predictor variables clump_thickness, is statistically significant at the 0.001 level. marginal_adhesion and bare_nuclei are significant at the 0.01 level and single_epithelial_cell_size is significant at the 0.05 level.

```r
# Predicting the class of tissue samples from the test data using the fitted
# model
glm.probs <- predict(glm.breastCancer.fit, test.Data, type = "response")

# Creating a vector of 2 with length same as the number of rows of test data
glm.pred <- rep(2, nrow(test.Data))

# Transformimg to 4(malignant) for observations for which predicted probability
# exceeds 0.5
glm.pred[glm.probs >.5] <- 4

# Creating a confusion matrix to determine how many observations were
# correctly or incorrectly classified
confmatrix.cancer <-table(glm.pred,test.Data$class)

# Displaing the confusion matrix
confmatrix.cancer
```

```
##
## glm.pred   2   4
##        2 115   2
##        4   4  68
```

```r
# Calculating the accuracy
sum(diag(confmatrix.cancer))/sum(confmatrix.cancer)
```

```
## [1] 0.968254
```

```r
# Calculating test error rate
mean(glm.pred != test.Data$class)
```

```
## [1] 0.03174603
```

From the confusion matrix, it can be observed that the model has a prediction accuracy of 0.968 or 96.8%. The test error rate is 0.0317, with only 2 observations incorrectly classified as benign and 4 observations as malignant.

Fitting a random forest model to predict whether tissue samples are malignant or benign. Classifying cases in the validation set.

```r
# Fitting a random forest model to predict whether tissue samples are
# malignant or benign
set.seed(1)
rf.breastCancer <- randomForest(class ~ clump_thickness + uniformity_of_cell_size  +
                                uniformity_of_cell_shape + marginal_adhesion +
                                single_epithelial_cell_size + bare_nuclei +
                                bland_chromatin + normal_nucleoli +   mitoses,
                              data=training.Data,
                              importance =TRUE)

# Predicting the class of tissue samples from the test data using the fitted
# model
rf.pred <- predict(rf.breastCancer, test.Data)
rf.probs <- predict(rf.breastCancer, test.Data, type = "prob")

# Creating a confusion matrix to determine how many observations were
# correctly or incorrectly classified
confmatrix.rf <- table(rf.pred, test.Data$class)

# Displaying the confusion matrix
confmatrix.rf
```

```
##
## rf.pred   2   4
##       2 114   0
##       4   5  70
```

```r
# Displaying the prediction accuracy
sum(diag(confmatrix.rf))/sum(confmatrix.rf)
```

```
## [1] 0.973545
```

```r
# Calculating test error rate
mean(rf.pred != test.Data$class)
```

```
## [1] 0.02645503
```

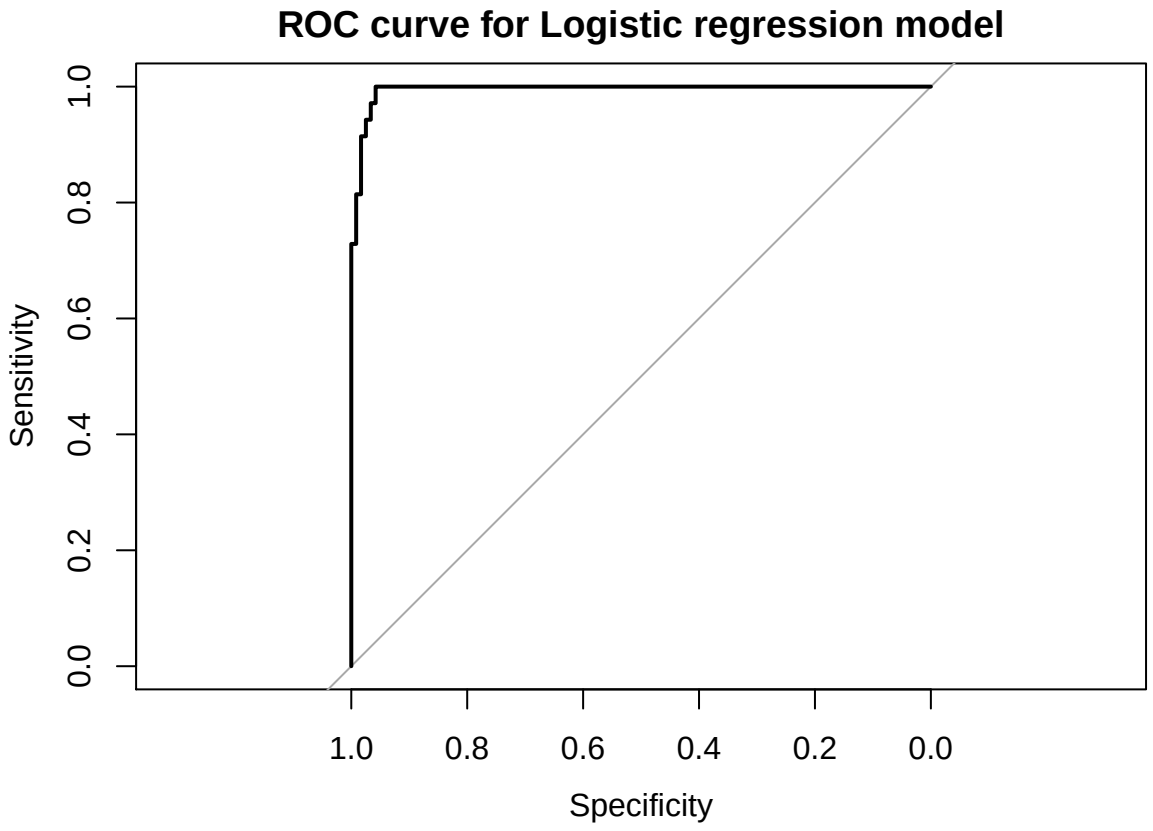
From the confusion matrix, it can be observed that the model has a prediction accuracy of 0.974 or 97.4%. The test error rate is 0.0265, with only 5 obervations incorrectly classified as malignant.

Compare the two models

```r
# Building a roc curve for the logistic regression model
roc.glm <- roc(test.Data$class, glm.probs)
# Displaying the roc object
roc.glm
```

```
##
## Call:
## roc.default(response = test.Data$class, predictor = glm.probs)
##
## Data: glm.probs in 119 controls (test.Data$class 2) < 70 cases (test.Data$class 4).
## Area under the curve: 0.9947
```

```r
# Plotting the roc curve
plot(roc.glm, main = "ROC curve for Logistic regression model")
```



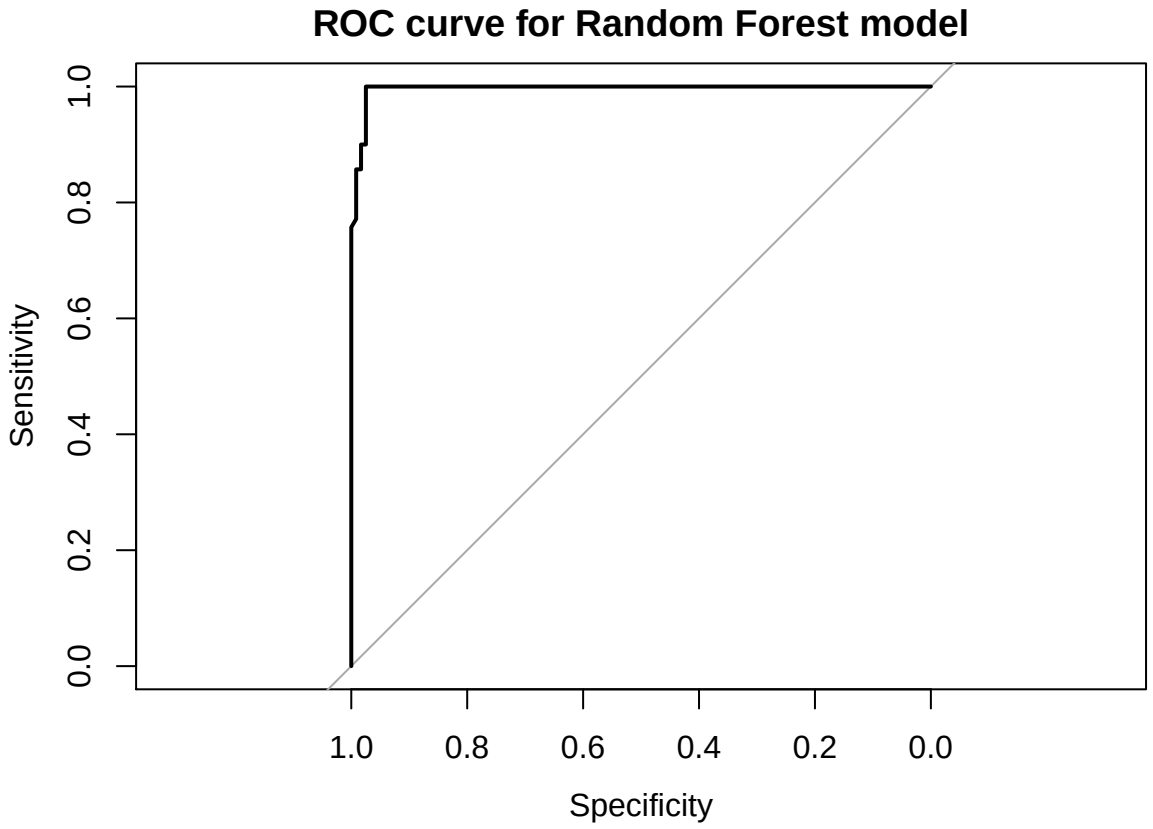

```
##
## Call:
## roc.default(response = test.Data$class, predictor = glm.probs)
##
## Data: glm.probs in 119 controls (test.Data$class 2) < 70 cases (test.Data$class 4).
## Area under the curve: 0.9947
```

```r
# Ordering the predicted values by the random forest model
rf.pred <- as.ordered(rf.pred)
# Building a roc curve for the logistic regression model
```

```
roc.rf <- roc(test.Data$class, rf.probs[,2])
# Displaying the roc object
roc.rf
```

```
##
## Call:
## roc.default(response = test.Data$class, predictor = rf.probs[,     2])
##
## Data: rf.probs[, 2] in 119 controls (test.Data$class 2) < 70 cases (test.Data$class 4).
## Area under the curve: 0.996
```

```
# Plotting the roc curve
plot(roc.rf, main = "ROC curve for Random Forest model")
```




```
##
## Call:
## roc.default(response = test.Data$class, predictor = rf.probs[,     2])
##
## Data: rf.probs[, 2] in 119 controls (test.Data$class 2) < 70 cases (test.Data$class 4).
## Area under the curve: 0.996
```

The overall performance of a classifier, summarized over all possible thresholds, is given by the area under the (ROC). The more the area under the curve, the better is the model perfomance.

From the roc summary statistic of logistic regression model, roc.glm, the Area under the curve: 0.995 and that of random forest model, roc.rf, the Area under the curve: 0.996.

Also, as seen from the confusin matrix of both the models, confmatrix.cancer and confmatrix.rf, the prediction accuracy of the random forest model(97.4%) is better than simple logistic regression model(96.8%).

Thus based on both prediction accuracy and Area under the ROC curve, random forest model is preferred over simple logistic regression model to predict the tumor as benign or malignant.