Machine Learning Course Project

1. Data Description and Project Objective

The data used for this project was from the Weight Lifting Exercises Dataset. This human activity recognition research has traditionally focused on discriminating between different activities. Six young health participants were asked to perform one set of 10 repetitions of the Unilateral Dumbbell Biceps Curl in five different fashions: exactly according to the specification (Class A), throwing the elbows to the front (Class B), lifting the dumbbell only halfway (Class C), lowering the dumbbell only halfway (Class D) and throwing the hips to the front (Class E).

The training data for this project are available here: https://d396qusza40orc.cloudfront.net/predmachlearn/pml-training.csv (https://d396qusza40orc.cloudfront.net/predmachlearn/pml-training.csv)

The test data are available here: https://d396qusza40orc.cloudfront.net/predmachlearn/pml-testing.csv (https://d396qusza40orc.cloudfront.net/predmachlearn/pml-testing.csv)

The goal of this project is to predict the manner in which they did the exercise. This is the “classe” variable in the data set.

2. Load and Clean Data

The first 7 columns in the data set are not useful for model fitting, so they are excluded. Also, lots of columns are statistical summary variables (i.e. min, max, std dev, avg, var, kurtosis, skewness) and contain many missing values, thus they are removed as well. See the code below for the complete list of columns that are excluded from the analysis.

```
training <- read.csv("./training.csv", header = TRUE)
testing <- read.csv("./testing.csv", header = TRUE)

training.new <- training[, -which(grepl('user_name|X|timestamp|window|kurtosis_|skewness_|max_|m
in_|
                                        amplitude_|avg_|var_|stddev_|amplitude', names(trainin
g)))]

testing.new <- testing[, -which(grepl('user_name|X|timestamp|window|kurtosis_|skewness_|max_|min
_|
                                      amplitude_|avg_|var_|stddev_|amplitude', names(testing)))]
```

3. Model Fitting and Selection

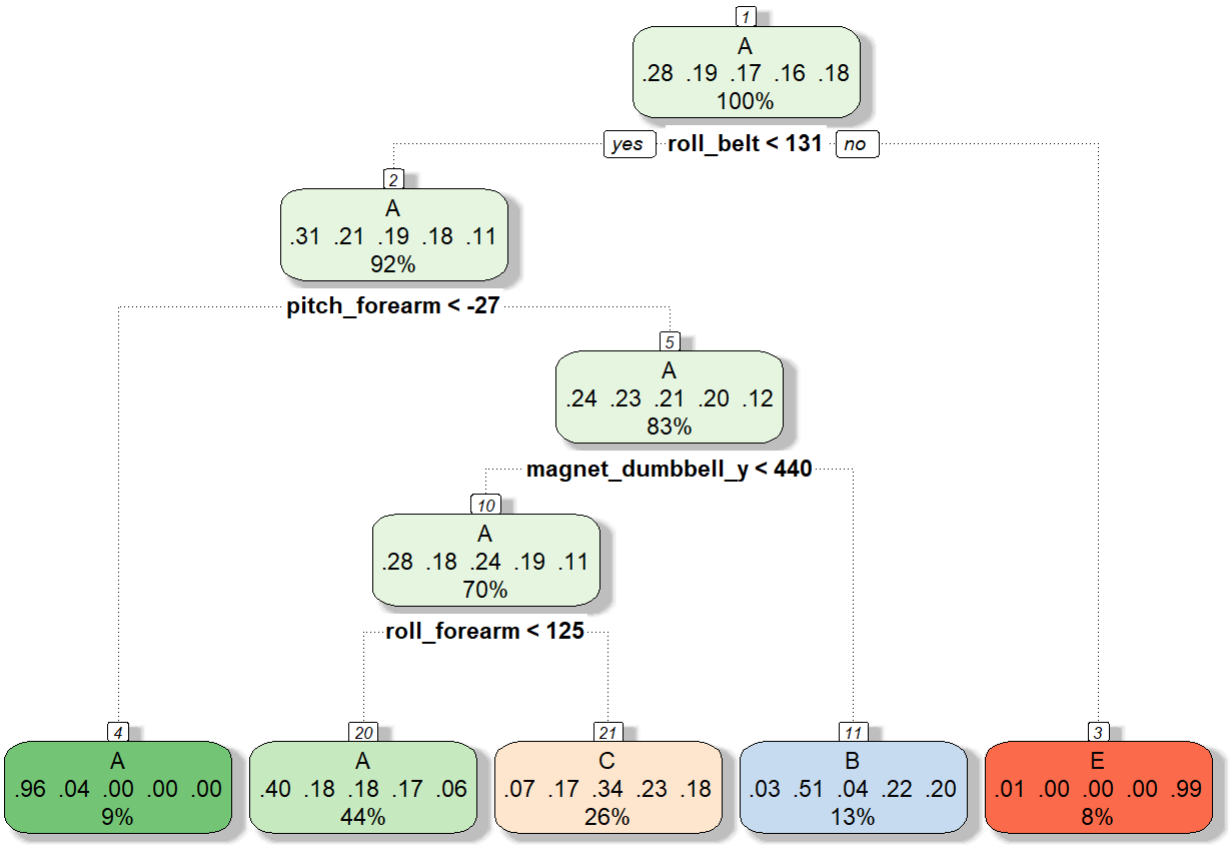
The 60% of testing data are randomly selected for model fitting and the remaining testing data are used for cross validation.

```
library(caret)
library(gbm)
library(AppliedPredictiveModeling)
library(randomForest)
library(rattle)

set.seed(1001)
intrain <- createDataPartition(training.new$classe, p = 0.6, list = FALSE)
train.sub <- training.new[intrain,]
test.sub<- training.new[-intrain,]
```

Since “classe”" is a factor varaible, the methods of decision tree and random forest are considered for model fitting. The model with more accuracy based on the cross validation will be chosen as a final model to predict for the test data.

```
# Model fitting
modfit.rpart <- train(classe ~., data = train.sub, method = "rpart")
fancyRpartPlot(modfit.rpart$finalModel)
```



Rattle 2019-Jul-21 10:17:33 Mansen

```
pred.rpart <- predict(modfit.rpart, test.sub)
confusionMatrix(pred.rpart, test.sub$classe)
```

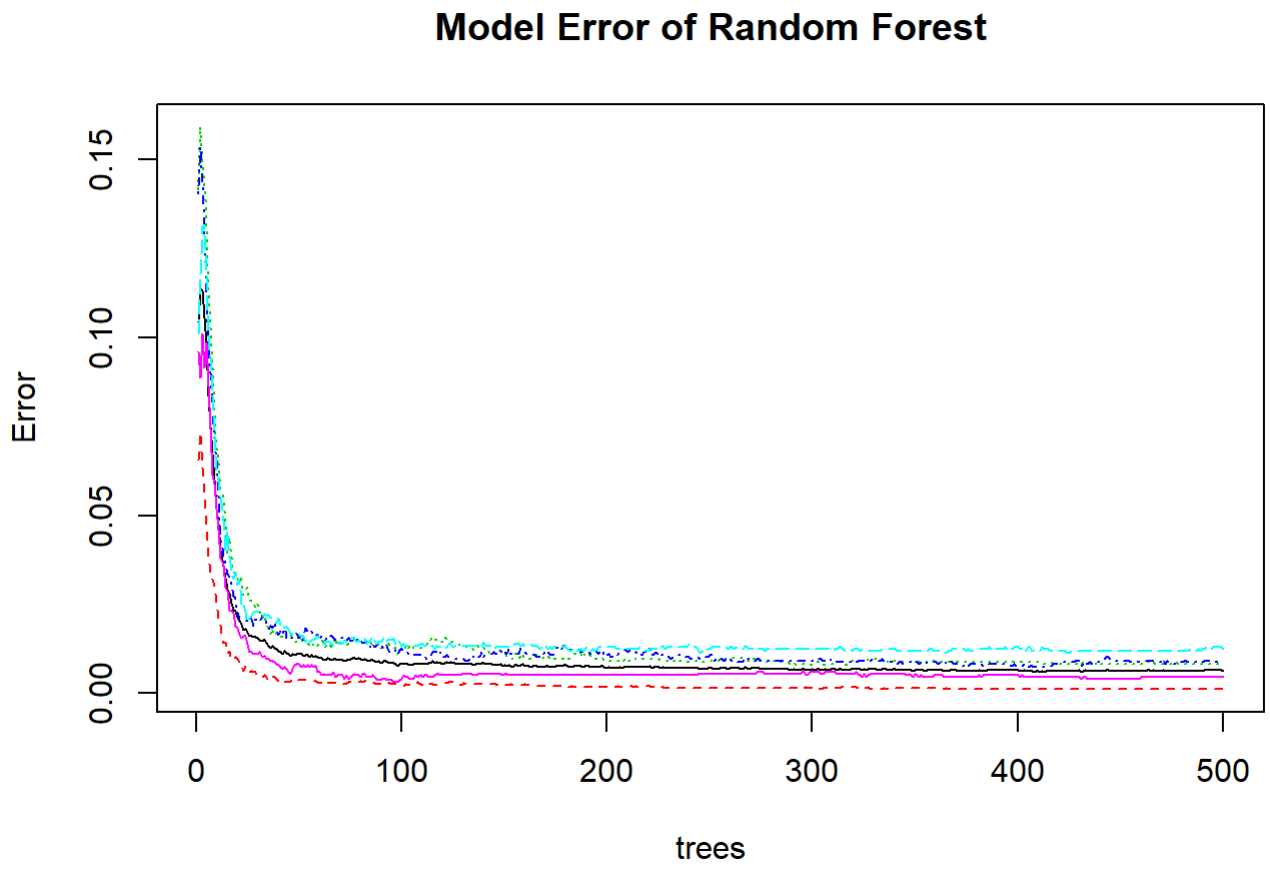
```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction    A    B    C    D    E
##          A 2031  666  658  559  192
##          B   33  506   44  241  190
##          C  161  346  666  486  404
##          D    0    0    0    0    0
##          E    7    0    0    0  656
##
## Overall Statistics
##
##                Accuracy : 0.4918
##                  95% CI : (0.4807, 0.503)
##     No Information Rate : 0.2845
##     P-Value [Acc > NIR] : < 2.2e-16
##
##                   Kappa : 0.3355
##
##  Mcnemar's Test P-Value : NA
##
## Statistics by Class:
##
##                      Class: A Class: B Class: C Class: D Class: E
## Sensitivity            0.9099  0.33333  0.48684   0.0000  0.45492
## Specificity            0.6304  0.91972  0.78435   1.0000  0.99891
## Pos Pred Value         0.4946  0.49901  0.32283      NaN  0.98944
## Neg Pred Value         0.9463  0.85187  0.87861   0.8361  0.89057
## Prevalence             0.2845  0.19347  0.17436   0.1639  0.18379
## Detection Rate         0.2589  0.06449  0.08488   0.0000  0.08361
## Detection Prevalence   0.5233  0.12924  0.26294   0.0000  0.08450
## Balanced Accuracy      0.7702  0.62653  0.63559   0.5000  0.72692
```

The accuracy with decision tree based on cross validation is 49.2% only, which is very low. Thus, it would not provide good prediction for the test data. Next, random forest is used for model fitting.

```
modfit.rf <- randomForest(classe ~., data = train.sub)
pred.rf <- predict(modfit.rf, test.sub)
confusionMatrix(pred.rf, test.sub$classe)
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction    A    B    C    D    E
##          A 2231    9    0    0    0
##          B    1 1505    3    0    0
##          C    0    4 1364   20    3
##          D    0    0    1 1265    6
##          E    0    0    0    1 1433
##
## Overall Statistics
##
##                Accuracy : 0.9939
##                  95% CI : (0.9919, 0.9955)
##     No Information Rate : 0.2845
##     P-Value [Acc > NIR] : < 2.2e-16
##
##                   Kappa : 0.9923
##
##  Mcnemar's Test P-Value : NA
##
## Statistics by Class:
##
##                      Class: A Class: B Class: C Class: D Class: E
## Sensitivity            0.9996   0.9914   0.9971   0.9837   0.9938
## Specificity            0.9984   0.9994   0.9958   0.9989   0.9998
## Pos Pred Value         0.9960   0.9973   0.9806   0.9945   0.9993
## Neg Pred Value         0.9998   0.9979   0.9994   0.9968   0.9986
## Prevalence             0.2845   0.1935   0.1744   0.1639   0.1838
## Detection Rate         0.2843   0.1918   0.1738   0.1612   0.1826
## Detection Prevalence   0.2855   0.1923   0.1773   0.1621   0.1828
## Balanced Accuracy      0.9990   0.9954   0.9965   0.9913   0.9968
```

```
plot(modfit.rf, main = "Model Error of Random Forest")
```



The random forest gives the accuracy of 99.4%. With this, the expected out of sample error would be small. Thus, the random forest is used as final model to predict for the test data.

4. Predict for the Final Test Data

```
pred.test <- predict(modfit.rf, testing.new)
print(pred.test)
```

```
##  1  2  3  4  5  6  7  8  9 10 11 12 13 14 15 16 17 18 19 20
##  B  A  B  A  A  E  D  B  A  A  B  C  B  A  E  E  A  B  B  B
## Levels: A B C D E
```

The result above displays the predictions for the test data based on the model of random forest.