# Prediction Assignment Writeup

BHavesh Jain

March 18, 2018

## Objective

The purpose of this project was to quantify how well the participants performed a barbell lifting exercise and to classify the measurement read from an accelerometer into 5 different classes (Class A:Class E).

Please reference the links below for the data sources:

http://groupware.les.inf.puc-rio.br/har

https://d396qusza40orc.cloudfront.net/predmachlearn/pml-training.csv

https://d396qusza40orc.cloudfront.net/predmachlearn/pml-testing.csv

*Install/load the required packages needed for the creation of the model*
```
library(caret)
library(rpart)
library(randomForest)
```

*Load the training and testing datasets*
```
train<-read.csv("C:/Users/aao1009/Desktop/pml-
training.csv",na.strings=c("NA","#DIV/0!",""))
test<-read.csv("C:/Users/aao1009/Desktop/pml-
testing.csv",na.strings=c("NA","#DIV/0!",""))
```

*Remove null columns and the first 7 columns that will not be used*
```
test_clean <- names(test[,colSums(is.na(test)) == 0]) [8:59]
clean_train<-train[,c(test_clean,"classe")]
clean_test<-test[,c(test_clean,"problem_id")]
```

*Check the dimensions of the clean test and train sets*
```
dim(clean_test)
dim(clean_train)
```

*Split the data into the training and testing datasets*
```
set.seed(100)
inTrain<-createDataPartition(clean_train$classe, p=0.7, list=FALSE)
training<-clean_train[inTrain,]
testing<-clean_train[-inTrain,]
dim(training)
dim(testing)
```

### Predicting the outcome using 3 different models

*LDA Model*

```
lda_model<-train(classe~ ., data=training, method="lda")
set.seed(200)
predict<-predict(lda_model,testing)
confusionMatrix(predict,testing$classe)
```

The LDA model gave a 70% accuracy on the testing set, with the expected out of sample error around 30%.

*Decision Tree Model*

```
decision_tree_model<-rpart(classe~ ., data=training,method="class")
set.seed(300)
predict<-predict(decision_tree_model,testing,type="class")
confusionMatrix(predict,testing$classe)
```

The Decision Tree Model gave a 74% accuracy on the testing set, with the expected out of sample error around 26%.

*Random Forest Model*

```
random_forest_mod<-randomForest(classe~ ., data=training, ntree=500)
set.seed(300)
predict<-predict(random_forest_mod, testing, type ="class")
confusionMatrix(predict,testing$classe)
```

The Random Forest Model gave a 99.6% accuracy on the testing set, with the expected out of sample error around 0.4%.

### Conclusion

The greatest accuracy was achieved using the Random Forest Model, which gave an accuracy of 99.6%. Hence, this model was further used to make predictions on the exercise performance for 20 participants.

```
predict<-predict(random_forest_mod, clean_test, type ="class")
predict
```