Practical Machine Learning Course Project

KK

September 24, 2015

Getting and Cleaning Data

```
WD       <- "C:/Users/KK/Documents/Outside Learning/Specialization-Data Science/08_Practical Machine Lea
setwd(WD)
data     <- read.csv("pml-training.csv", na.strings = c("NA","#DIV/0!",""))
submission    <- read.csv("pml-testing.csv", na.strings = c("NA","#DIV/0!",""))
##cleaning data by using only data from accelerometer and dumbbell

NAColumn      <- which(is.na(data[1,]) == TRUE)
data2         <- data[,-NAColumn]
VariableName  <- colnames(data2)
accel         <- grep("accel",VariableName)
dumbbell      <- grep("dumbbell", VariableName)
useVariable   <- sort(c(accel,dumbbell,60))
data2         <- data2[ , useVariable]
data2         <- na.omit(data2)##Remove NA
submit.data   <- submission[,-NAColumn]
submit.data   <- submit.data[ , useVariable]
```

Building Model

Data partitioning and prediction

```
library(caret)
```

```
## Warning: package 'caret' was built under R version 3.2.2

## Loading required package: lattice
## Loading required package: ggplot2
```

```
library(randomForest)
```

```
## Warning: package 'randomForest' was built under R version 3.2.2

## randomForest 4.6-10
## Type rfNews() to see new features/changes/bug fixes.
```

```
##Cross Validation - Create Data Partition by spliting "pml-training.csv" into train set and test set
inTrain       <- createDataPartition(y = data2$class, p = 0.75, list = FALSE)
training      <- data2[inTrain, ]
testing       <- data2[-inTrain, ]
##Preprocessing with PCA
set.seed(1804)
modelFit      <- randomForest(classe ~ .,data=training,mtry=5,importance=TRUE)
modelFit
```

```
##
## Call:
##  randomForest(formula = classe ~ ., data = training, mtry = 5,      importance = TRUE)
##                Type of random forest: classification
##                      Number of trees: 500
## No. of variables tried at each split: 5
##
##         OOB estimate of  error rate: 2.81%
## Confusion matrix:
##      A    B    C    D    E class.error
## A 4158   10    5   10    2 0.006451613
## B   81 2707   54    3    3 0.049508427
## C    9   58 2486    7    7 0.031554344
## D   10    1   95 2296   10 0.048092869
## E    5   12   16   16 2657 0.018107908
```

Cross Validation

```
##Predict testing data set with the model from training data set
predictions    <- predict(modelFit, newdata = testing)
```

The out of sample error rate is shown in the confusion matrix below

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction    A    B    C    D    E
##          A 1380   38    2    3    2
##          B    7  894   20    0   11
##          C    2   13  830   35    9
##          D    6    0    1  764    7
##          E    0    4    2    2  872
##
## Overall Statistics
##
##                Accuracy : 0.9666
##                  95% CI : (0.9611, 0.9714)
##     No Information Rate : 0.2845
##     P-Value [Acc > NIR] : < 2.2e-16
##
##                   Kappa : 0.9577
##  Mcnemar's Test P-Value : NA
##
## Statistics by Class:
##
##                      Class: A Class: B Class: C Class: D Class: E
## Sensitivity            0.9892   0.9420   0.9708   0.9502   0.9678
## Specificity            0.9872   0.9904   0.9854   0.9966   0.9980
## Pos Pred Value         0.9684   0.9592   0.9336   0.9820   0.9909
## Neg Pred Value         0.9957   0.9862   0.9938   0.9903   0.9928
## Prevalence             0.2845   0.1935   0.1743   0.1639   0.1837
## Detection Rate         0.2814   0.1823   0.1692   0.1558   0.1778
## Detection Prevalence   0.2906   0.1900   0.1813   0.1586   0.1794
## Balanced Accuracy      0.9882   0.9662   0.9781   0.9734   0.9829
```