Course Project in Machine Learning

Daniel A. Brodén

September 21st, 2014

Executive Summary

The purpose of this project is to predict the manner in which subjects do exercice. Two data frames were provided consisting of a training and a testing set. There are 5 different types of exercices in the *classe* variable denoted A, B, C, D and E. Our objective is to build and train a prediction model that predicts the type of exercice performed with high accuracy. To ensure this, different machine learning algorithms are compared and cross-validation techniques are used on the training data set. Finally, the prediction model is evaluated on the testing data set for 20 different test cases. Our prediction model successfully predicts the 20 different type of activities from the testing set.

Building and Training a Prediction Model

The training set consist of 19622 observations and 160 variables. To reduce computation time and model complexity the data frames are cleaned, i.e., columns consisting of large numbers of NA values and blanks are removed. Furthermore, timestamps are removed from the data frames as they unecessarily add extra layers of complexity for building the model.

```
testing <- read.csv("pml-testing.csv")
training <- read.csv("pml-training.csv")
training <- training[,c(-1,-3,-4,-5,-6,-7)] #removing irrelevant columns
threshold <- 0.5*dim(training)[1]
training <- training[, colSums(is.na(training)) < threshold] #removing NAs
training <- training[, colSums(training == "") < threshold] #removing blanks
dim(training)
```

```
## [1] 19622    54
```

The training set is reduced to 54 variables.

We train two prediction models using different methods. The first method is based on (i) classification trees while the second method uses (ii) random forest. The outcome of the prediction model is chosen to be the *classe* variable while the predictors are selected as the remaining 53 variables from the training set. The predictors consist of various sensor outputs measured from the belt, arm, dumbell and forearm of the 6 subjects.

```
library(randomForest); library(tree)
set.seed(1)
modFit1 <- tree(as.factor(classe)~., data=training)
modFit2 <- randomForest(as.factor(classe)~., data=training, ntree=50)
modFit2$confusion
```

```
##      A    B    C    D    E class.error
## A 5574    3    2    1    0    0.001075
## B   18 3767   12    0    0    0.007901
```

```
## C    0   12 3405    4    1    0.004968
## D    0    0   28 3185    3    0.009639
## E    0    0    1    4 3602    0.001386
```

NB: The number of trees created from the random forest algorithm is set to 50 to reduce computation time.

The confusion matrix of the prediction shows a very small error rate. The OOB (Out-Of-Bag) error rate estimate is 0.45%. The advantage of selecting the random forest algorithm instead of classification trees is the increase in accuracy. When using random forest there is no need for additional cross-validation such as k-fold, bagging or boosting since this is already executed internally. For this reason we choose method (ii) to build our prediction model.

Evaluating the Prediction Model

We predict the outcomes of the 20 different cases from the testing set.

```
answers <- vector(mode="character",length=dim(testing)[1])
for(i in 1:dim(testing)[1]) {
        answers[i] <- as.character(predict(modFit2, testing[i,]))
}
print(answers)
```

```
##  [1] "B" "A" "B" "A" "A" "E" "D" "B" "A" "A" "B" "C" "B" "A" "E" "E" "A"
## [18] "B" "B" "B"
```

The submission of the results on the course webpage shows that we successfully predict the 20 different outcomes with 100% accuracy.