Machines Learning Project

Filomena Ciccarelli

4 June 2016

Executive Summary

This report investigates the correct execution of an activity performed by six male participants aged between 20-28 years. In particular, they were asked to perform barbell lifts correctly and incorrectly in 5 different ways (*classe*). The scope of this analysys is to create a machines learnig algorithm that predict the correct execution of the exercise using the predictor variables gathered as part of the research. More information about the background of the the Human Activity Recognition research is available from the website HAR in the *Weight Lifting Exercise Dataset* section.

Based on the available data sets, the report explores the accuracy of different prediction models of the target variable *classe*. The Random Forest model delivered the best accuracy rate 99.4% and therefore the lowest Out Of Sample error. Using this model, the requested 20 predictions were made for the test data set.The report is structured in three sections.

Load data plus Preliminary exploratory data analysis

In this section we load the data and do some exploratory data analysis. The training data set for this project is available here. The testing data set is available here.

```
trainingRaw<-read.csv("pml-training.csv", na.strings = c("","NA","#DIV/0!"))
finaltestRaw<-read.csv("pml-testing.csv",na.strings = c("","NA","#DIV/0!"))
```

```
str(trainingRaw)
```

A preliminary data analysis indicates that the there are *timestamp* variables which will be removed from our prediction model since it has no time dependency. We also remove the columns that have a majority of *NA* values.

```
training<-trainingRaw[,-c(1:7)]
finaltest<- finaltestRaw [, -c(1:7)]
NAindex<-which (colSums(is.na(training))>nrow(training)/2)
training <- training[, -NAindex]
finaltest <- finaltest[, -NAindex]
```

The *training* and *testing* data sets have now respectively 19622 x 53 and 20 x 53 dimensions.

Machines Learning

In this section we carry out the *training* set data partioning and explore different prediction models.

```
library(caret);library(randomForest);library(rpart)
library(rattle);library(scales)
set.seed(1971)
```

Data Partition

```
inTrain<- createDataPartition(training$classe, p=0.7, list= FALSE)
trainingDS<-training[inTrain,]
testing <- training [-inTrain,]
```

The traing data set for our prediction model has 13737 rows with the following *classe* splittings.

```
##    A    B    C    D    E
## 3906 2658 2396 2252 2525
```

Candidate Models

In this section, we fit different models and compare their accuracy results on the *testing* data set.

Linear Discrimininat Analysis

We start our model analysis with a parametric model, LDA. LDA analysis assumes that the data comes from a multivariate normal distribution.

```
ldaMod<-train(trainingDS$classe~.,data=trainingDS,method="lda")
ldaPredict<- predict(ldaMod,newdata=testing[,-53]) #53 is the outcome classe column index
ldaCM <- confusionMatrix(ldaPredict,testing$classe)
```

The LDA model delivers a low accuracy prediction:

```
ldaAcc<-ldaCM$overall[[1]]
percent(ldaAcc)
```

```
## [1] "70.1%"
```

The estimated out of sample (*OOS*) error with the cross validation for this model is 29.9%. LDA method provide a poor predictor model as the *OOS* error is high. In the next sections we explore non-parametric models.

Recursive Partitioning Model

We start with a basic classification tree

```
rpartMod<-train(trainingDS$classe~.,data=trainingDS,method="rpart", tuneLength =30)
```

Predict *classe* for cross validation:

```
rpartPredict<- predict(rpartMod,newdata=testing[,-53])
rpartCM<- confusionMatrix(rpartPredict,testing$classe)
```

The prediction accuracy improves with this model:

```
rpartAcc<-rpartCM$overall[[1]]
percent(rpartAcc)
```

```
## [1] "83.1%"
```

The recursive model delivers a *OOB* equal to 16.9%.

Random Forests

Random Forests should be a better prediction model for our data set.

```
set.seed(1972)
rfMod<-randomForest(classe~.,data=trainingDS,ntree=200)
```

Predicting *classe* for cross validation:

```
rfPredict<- predict(rfMod,newdata=testing[,-53])
rfCM<- confusionMatrix(rfPredict,testing$classe)
```

The accuracy of the random forests model on the testing data set is much higher than the previous models

```
rfAcc<-rfCM$overall[[1]]
percent(rfAcc)
```

```
## [1] "99.4%"
```

The estimated *OBB* with the cross validation data set is equal to 0.646%.

Conclusions

The Random Forest model is very accurate and we select it for the prediction on the *finaltest* data.

```
SubmitPredict <-predict(rfMod,newdata=finaltest,type="class")
SubmitPredict
```

```
##  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
```