Human activity recognition

Summary

The *Weight Lifting Exercises Dataset* described and analyzed in the article *Qualitative Activity Recognition of Weight Lifting Exercises* by Velloso, E. et al. (see Human Activity Recognition (http://groupware.les.inf.puc-rio.br/har)) contains data about 5 difference barbell lifting exercises performed by 5 young people. These exercises have been classified in five categories (A, B, C, D, E). Category A corresponds to correct execution while B, C, D ,E are different errors of execution. The objective of our analysis is to assess the capability of the measured variables to predict the category to which exercises belong.

Analysis

Our analysis will not be based on the original dataset that can be downloaded at WLE dataset (http://groupware.les.inf.puc-rio.br/static/WLE/WearableComputing_weight_lifting_exercises_biceps_curl_variations.csv) but on files *pml-training.csv* and *pml-testing* files that can be downloaded at Practical machine learning (https://class.coursera.org/predmachlearn-002/human_grading/view/courses/972090/assessments/4/submissions).

Data preparation

We start by reading the data:

```
training<-read.csv("pml-training.csv")
testing<-read.csv("pml-testing.csv")
```

The subsequent step is to transform the output variable classe in a factor

```
training$classe<-as.factor(training$classe)
```

The training data set contains 160 variables and 19622 observations. The test set contains 160 variables and 20 observations.

To reduce the dataset we first remove near zero values:

```
options(warn=-1)
suppressPackageStartupMessages(library(caret))
nzv <- nearZeroVar(training, saveMetrics=TRUE)
omit <- which(nzv$nzv==TRUE)
training <- training[,-omit]
testing <- testing[,-omit]
```

This reduces the number of variables from 160 to 100. Variables can be further reduced by removing those that contain a high percentage of null values

```
notNullColumns<-colSums(is.na(training)) < 19000
training<-training[,notNullColumns]
testing<-testing[,notNullColumns]
```

This reduces the number of variables to 59.

The last step we perform to make computation faster is the random selection of 3000 rows:

```
numberOfRows<-3000
trainInds <- sample(nrow(training), numberOfRows)
training <- training[trainInds,]
```

We point out that the number 3000 is arbitrary and can be modified if desired.

Data analysis

Since out testing dataset is too small (20 observations) and does not contain the classe variable we split our training dataset into a training subset and a test subset:

```
trainIndex <-createDataPartition(training$classe,p=0.6,list=FALSE)
training.train<-training[trainIndex,]
training.test<-training[-trainIndex,]
```

We are now ready to analyze our data using random forests:

```
suppressMessages(library(randomForest))
modFit<-train(classe~.,data=training.train,method="rf",prox=TRUE,preProcess=c("center", "scale"
))
modFit$results
```

```
##   mtry Accuracy  Kappa AccuracySD  KappaSD
## 1    2   0.9551 0.9432   0.008801 0.011108
## 2   41   0.9983 0.9978   0.001455 0.001847
## 3   80   0.9980 0.9974   0.001463 0.001857
```

We can now test our model on our test set:

```
prediction<-predict(modFit,training.test)
table(prediction,training.test$classe)
```

```
##
## prediction   A   B   C   D   E
##          A 342   0   0   0   0
##          B   0 217   0   0   0
##          C   0   0 217   0   0
##          D   0   0   0 201   0
##          E   0   0   0   0 220
```

```
confusionMatrix<-confusionMatrix(prediction,training.test$classe)
confusionMatrix$byClass[,c(1,2,8)]
```

```
##          Sensitivity Specificity Balanced Accuracy
## Class: A           1           1                 1
## Class: B           1           1                 1
## Class: C           1           1                 1
## Class: D           1           1                 1
## Class: E           1           1                 1
```

The accuracy provides us with the desired out-of-sample error estimate. Last but not least we apply our prediction to the original test set

```
prediction<-predict(modFit,testing)
prediction
```

```
##  [1] A A A A A A A A A A A A A A A A A A A A
## Levels: A B C D E
```