Peer-graded Assignment

R Markdown

Click the **Knit** button to generate the document

Introduction

This project tries to determine the manner in which a specific exercise **Unilateral Dumbbell Biceps Curl** is done. There are five possible outcomes:

- **A:** correct execution
- **B:** incorrect execution by throwing the elbows to the front
- **C:** incorrect execution by lifting the dumbbell only halfway
- **D:** incorrect execution by lowering the dumbbell only halfway
- **E:** incorrect execution by throwing the hips to the front

The training data is taken from a set of measurements made with accelerometers placed on the belt, forearm, arm, and dumbell of 6 participants, who were asked to perform one set of 10 repetitions of the Unilateral Dumbbell Biceps Curl in the five different ways listed above (A, B, C, D and E).

The procedure to construct and select the final model is the following.

Load packages, data and set seed

Load the required packages

```r
library(lattice); library(ggplot2); library(caret)
```

Data was loaded using "read.csv" function.

```r
x <- read.csv("pml-training.csv")
```

The seed for the random number generator was set at 1969.

```r
set.seed(1969)
```

Divide data into training and testing sets

Data was separated into two groups: training and testing, with 70% of the data in the training set and 30% in the testing set.

```r
inTrain <- createDataPartition(y=x$classe,p=0.7,list=F)
training <- x[inTrain,]
testing <- x[-inTrain,]
```

Preparing and clearning data

From the training data, four groups were formed corresponding to the type of accelerometer used, i.e. belt, arm, dumbbell and forearm. These were selected using the grep function. For example for the "belt" accelerometer the following was excecuted.

```
belt <- grep("belt",names(training))
```

and similarly for the other accelerometers. This was used later to find separate models that use the data of only some combination of accelerometers.

Before constructing the model, the data was cleaned. Several variables contained a large fraction of NA's (~98%) and these variables were removed from the training and testing data before attempting any model building.

The follwing code was used to identify the variables with large numbers of NA's. This was done for all the accelerometers.

```
nabelt <- sapply(training[,belt],function(u) length(which(is.na(u)))/length(u))
nabelt
```

```
##            roll_belt           pitch_belt             yaw_belt
##            0.0000000            0.0000000            0.0000000
##     total_accel_belt   kurtosis_roll_belt  kurtosis_picth_belt
##            0.0000000            0.0000000            0.0000000
##    kurtosis_yaw_belt   skewness_roll_belt skewness_roll_belt.1
##            0.0000000            0.0000000            0.0000000
##    skewness_yaw_belt        max_roll_belt       max_picth_belt
##            0.0000000            0.9798355            0.9798355
##         max_yaw_belt        min_roll_belt       min_pitch_belt
##            0.0000000            0.9798355            0.9798355
##         min_yaw_belt  amplitude_roll_belt amplitude_pitch_belt
##            0.0000000            0.9798355            0.9798355
##   amplitude_yaw_belt var_total_accel_belt        avg_roll_belt
##            0.0000000            0.9798355            0.9798355
##     stddev_roll_belt        var_roll_belt       avg_pitch_belt
##            0.9798355            0.9798355            0.9798355
##    stddev_pitch_belt       var_pitch_belt         avg_yaw_belt
##            0.9798355            0.9798355            0.9798355
##      stddev_yaw_belt         var_yaw_belt         gyros_belt_x
##            0.9798355            0.9798355            0.0000000
##         gyros_belt_y         gyros_belt_z         accel_belt_x
##            0.0000000            0.0000000            0.0000000
##         accel_belt_y         accel_belt_z        magnet_belt_x
##            0.0000000            0.0000000            0.0000000
##        magnet_belt_y        magnet_belt_z
##            0.0000000            0.0000000
```

The variables with output equal to zero were selected for the final data.

Additionally, several variables related to the accelerometer measurements were "factor" variables. These variables were also removed from the training and testing sets, as they did not show predictive value.

The following code was used to achieve this.

```
fbelt <- as.numeric(sapply(training[,belt],is.factor))
bad_belt <-unique(c(which(nabelt!=0),which(fbelt!=0)))
bbelt <- belt[bad_belt]
gbelt <- belt[-bad_belt]
```

From these, the vector of indices bbelt (for "bad" belt data), was used to remove the corresponding columns from the "belt" subgroup, while gbelt (for "good" belt data) contained the indices of the columns to keep.

The remaining variables associated with "belt" were

```
names(training[,gbelt])
```

```
##  [1] "roll_belt"        "pitch_belt"       "yaw_belt"
##  [4] "total_accel_belt" "gyros_belt_x"     "gyros_belt_y"
##  [7] "gyros_belt_z"     "accel_belt_x"     "accel_belt_y"
## [10] "accel_belt_z"     "magnet_belt_x"    "magnet_belt_y"
## [13] "magnet_belt_z"
```

Similar variables remained for the other accelerometer groups.

The final data set (both training and testing sets) was cleaned in the same way. Here we removed all the "bad" data and the first 7 columns which did not contain information about the acceleromets.

```
training <- training[,-c(bbelt,barm,bdumbbell,bforearm)]
training <- training[,-c(1:7)]

testing <- testing[,-c(bbelt,barm,bdumbbell,bforearm)]
testing <- testing[,-c(1:7)]
```

The accelerometer groups were recalcualted for the final model training.

```
belt <- grep("belt",names(training))
arm <- grep("_arm",names(training))
dumbbell <- grep("dumbbell",names(training))
forearm <- grep("forearm",names(training))
```

Training data

The initial model used to train the data was "rpart".

Several models were trained using all variables listed above or combinations of subsets of variables according to the accelerometer (i.e. "belt", "arm", "dumbbell" and "forearm"). In each case the in-sample error (accuracy) was calculated.

Example of the code used is the following

```
modFit_all <- train(classe~.,data=training,method="rpart")
modFit_ba <- train(y=training$classe,x=training[,c(belt,arm)],method="rpart")
```

The accuracy of this model using all variables was **0.4963**.

The following table shows the calculated accuracy for combinations of pairs of accelerometer location.

acc	belt	arm	dumbell	forearm
belt	0.39	0.5494	0.4269	0.4369
arm	x	0.375	0.3662	0.3837
dumbbell	x	x	0.3859	0.439
forearm	x	x	x	0.3837

From these data it seems that the best models would be either the **all variables** or the **belt-arm variables**.

Model improvement

The lack of good accuracy results prompted the use of bagging techniques. In this case the model was retrained using the method "treebag". The follwing code was used.

```r
modFit_all <- train(classe~.,data=training,method="treebag")
modFit_ba <- train(y=training$classe,x=training[,c(belt,arm)],method="treebag")
```

This produced the following accuracy values for the in-sample error

acc	belt	arm	dumbell	forearm
belt	0.9993	0.9999	0.9999	0.9997
arm	x	0.9996	0.9992	0.9999
dumbbell	x	x	0.9975	0.9995
forearm	x	x	x	0.9994

Using all variables the accuracy is **0.9993**

From this data it is clear that there is significant improvement using the *bagging* method. The final model selection will depend on the performance on the testing data.

Validation

To determine the out-of-sample error the models were used to predict the "classe" variable from the testing set (corrected as described above). An example of the code used was

```r
p_ba <- predict(modFit_ba,testing)
confusionMatrix(p_ba,testing$classe)
```

The accuracy of all these models using the "testing" data is given in the following table.

acc	belt	arm	dumbell	forearm
belt	0.9137	0.9679	0.9584	0.9747
arm	x	0.8637	0.955	0.9526
dumbbell	x	x	0.8702	0.9363
forearm	x	x	x	0.8856

Using all variables the accuracy is **0.982**

From these data we estimate that the out-of-sample accuracy no lower than **0.86**, which approximately correspond to modeling using only the “arm” accelerometer data.

Model selection

From the previous result the best model seems to be the one using **all variables** and that is the one used to predic the 20 testing cases.