Forecasting

Jose Ruiz

21/8/2020

Summary

uman Activity Recognition (HAR) is a key research area that is gaining increasing attention, especially for the development of context-aware systems. There are many potential applications for HAR, like: elderly monitoring, life log systems for monitoring energy expenditure and for supporting weight-loss programs, and digital assistants for weight lifting exercises. Using devices such as Jawbone Up, Nike FuelBand, and Fitbit it is now possible to collect a large amount of data about personal activity relatively inexpensively.

Six participants were asked to perform one set of 10 repetitions of the Unilateral Dumbbell Biceps Curl in five different fashions: exactly according to the specification (Class A), throwing the elbows to the front (Class B), lifting the dumbbell only halfway (Class C), lowering the dumbbell only halfway (Class D) and throwing the hips to the front (Class E).

This report will describe how the data captured are used to identify the parameters involved in predicting the movement involved based on the above classification, and then to predict the movement for 20 test cases.

The training data were divided into two groups, a training data and a validation data (to be used to validate the data), to derived the prediction model by using the training data, to validate the model where an expected out-of-sample error rate of less than 0.5%, or 99.5% accuracy, would be acceptable before it is used to perform the prediction on the 20 test cases - that must have 100% accuracy (to obtain 20 points awarded).

The training model developed using Random Forest was able to achieve over 99.99% accuracy, or less than 0.03% out-of-sample error, and was able to predict the 20 test cases with 100% accuracy.

Download Data

```r
library(lattice)
library(ggplot2)
library(caret)
library(parallel)
library(doParallel)
```

```
## Loading required package: foreach
```

```
## Loading required package: iterators
```

```r
library(rpart)
library(rpart.plot)
training <- read.csv("D:/Users/1019049157/Escritorio/Positiva 2020/Machine Learning/pml-training.csv", n
testing <- read.csv("D:/Users/1019049157/Escritorio/Positiva 2020/Machine Learning/pml-testing.csv", na
```

Cleaning data

Remove variables with near zero variance Remove columns that are not predictors, which are the the seven first columns

```
training<-training[,colSums(is.na(training)) == 0]
testing <-testing[,colSums(is.na(testing)) == 0]

training   <-training[,-c(1:7)]
testing <-testing[,-c(1:7)]
```

Create cross VAlidation

In order to get out-of-sample errors, split the training data in training (75%) and testing (25%) data) subsets:

```
inTrain <- createDataPartition(y=training$classe, p=0.75, list=FALSE)
NEOTraining <- training[inTrain, ]
NEOTesting <- training[-inTrain, ]
dim(NEOTraining)
```

```
## [1] 14718    53
```

```
dim(NEOTesting)
```

```
## [1] 4904   53
```

Predicting Model

DECISION TREE Fit model on NEOTraining data Use model to predict class in validation set (NEOTesting)

```
fitDT <- rpart(classe ~ ., data=NEOTraining, method="class")
predictionDT <- predict(fitDT, NEOTesting, type = "class")
```

Estimate the errors of the prediction algorithm in the Decision Tree model confusionMatrix(NEOTesting$classe, predictionDT)

RANDOM FOREST Fit model on NEOTraining data fitRF <- randomForest(classe ~ ., data=NEOTraining, method="class")

Use model to predict class in validation set (NEOTesting) predictionRF <- predict(fitRF, NEOTesting, type = "class")

Estimate the errors of the prediction algorithm in the Random Forest confusionMatrix(NEOTesting$classe, predictionRF)

predictSubmission <- predict(fitRF, testing, type="class") predictSubmission

Conclusion

The model predicted the 20 test cases with 100% accuracy. All 20 points were awarded after submitting the 20 test files.