Machine Learning Course Project

Hubert LEVIEL

November 18th, 2015

Summary

The goal of this project is to build a machine learning algorithm to predict activity quality from activity monitors

Loading and parting the data

We load the data, then immediately part it into training and testing

```
data <- read.csv(url('https://d396qusza40orc.cloudfront.net/predmachlearn/pml-training.csv'))
testing_set <- read.csv(url('https://d396qusza40orc.cloudfront.net/predmachlearn/pml-testing.csv'))

library(caret)
```

```
## Warning: package 'caret' was built under R version 3.2.2
```

```
## Loading required package: lattice
## Loading required package: ggplot2
```

```
set.seed(555)
trainingIndex <- createDataPartition(data$classe, p=0.7, list=FALSE)
training <- data[trainingIndex,]
testing <- data[-trainingIndex,]
```

Preprocessing

The summary of training data shows many NAs or unset values, so I choose to remove the columns with to many NAs or unset values. Also we can see that some variables like X or window or date are ordered and shouldn't be used to build a model, neither the username. So I also remove the first 7 columns. And then I remove the exact same columns from the testing.

```
training.complete <- training[, colSums(is.na(training) | training=='' ) < nrow(training) * 0.5]
training.complete <- training.complete[, -c(1:7)]
testing.complete <- testing[,names(training.complete)]
testing_set.complete <- testing_set[,names(training.complete)[names(training.complete)!="classe"]]
dim(training.complete)
```

```
## [1] 13737    53
```

This leaves us with 53 columns (out of 160)

Training

To predict the classe outcome, I am going to use Quadratic Discriminant Analysis (qda) method of caret package, and train on other variables.

```
mymod <- train(classe~., data=training.complete, method='qda', show=FALSE)
```

```
## Loading required package: MASS
```

We can see that there is 90% accuracy with the training data

```
table(predict(mymod, training.complete),training.complete$classe)
```

```
##
##        A    B    C    D    E
##   A 3754  194    2    8    0
##   B  102 2189  118    8   65
##   C   19  246 2259  296  102
##   D   25    8   10 1919   63
##   E    6   21    7   21 2295
```

```
1-sum(predict(mymod, training.complete)!=training.complete$classe)/dim(training.complete)[1]
```

```
## [1] 0.9038364
```

Testing

We can see that there is 90% accuracy with the test data

```
table(predict(mymod, testing.complete),testing.complete$classe)
```

```
##
##        A    B    C    D    E
##   A 1597   86    0    3    0
##   B   50  940   48    5   35
##   C   14   95  973  134   43
##   D   11    6    3  812   25
##   E    2   12    2   10  979
```

```
1-sum(predict(mymod, testing.complete)!=testing.complete$classe)/dim(testing.complete)[1]
```

```
## [1] 0.9007647
```