Practical Machine Learning

Davide Liperoti

03 luglio 2015

Brief background

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. These type of devices are part of the quantified self movement - a group of enthusiasts who take measurements about themselves regularly to improve their health, to find patterns in their behavior, or because they are tech geeks. One thing that people regularly do is quantify how much of a particular activity they do, but they rarely quantify how well they do it. In this project, my goal will be to use data from accelerometers on the belt, forearm, arm, and dumbell of 6 participants. They were asked to perform barbell lifts correctly and incorrectly in 5 different ways. More information is available from the website here: http://groupware.les.inf.puc-rio.br/har (see the section on the Weight Lifting Exercise Dataset).

Code

Loading packets and data

For this analysis I'll need to load basic packages as follow:

```
library(caret)
```

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

```
library(ggplot2)
library(randomForest)
```

```
## randomForest 4.6-10
## Type rfNews() to see new features/changes/bug fixes.
```

```
library(rpart)
library(rpart.plot)
```

As described above, data consist in two different files, one is training dataset and it is available here [https://d396qusza40orc.cloudfront.net/predmachlearn/pml-training.csv]. Instead testing dataset is available here [https://d396qusza40orc.cloudfront.net/predmachlearn/pml-testing.csv].

Loading the training dataset, we discover that it is made by 19622 measures and 160 variables. In the training dataset there are many missing value, coded as *NA*, *#DIV/0!* or blank space.

```
# to check the presence of the NA and #DIV/0! character
# for (i in 1:160){print(table(grepl("#DIV/0", train[, i]))); print(i) }

if(!exists("training") && !exists("testing"))
```

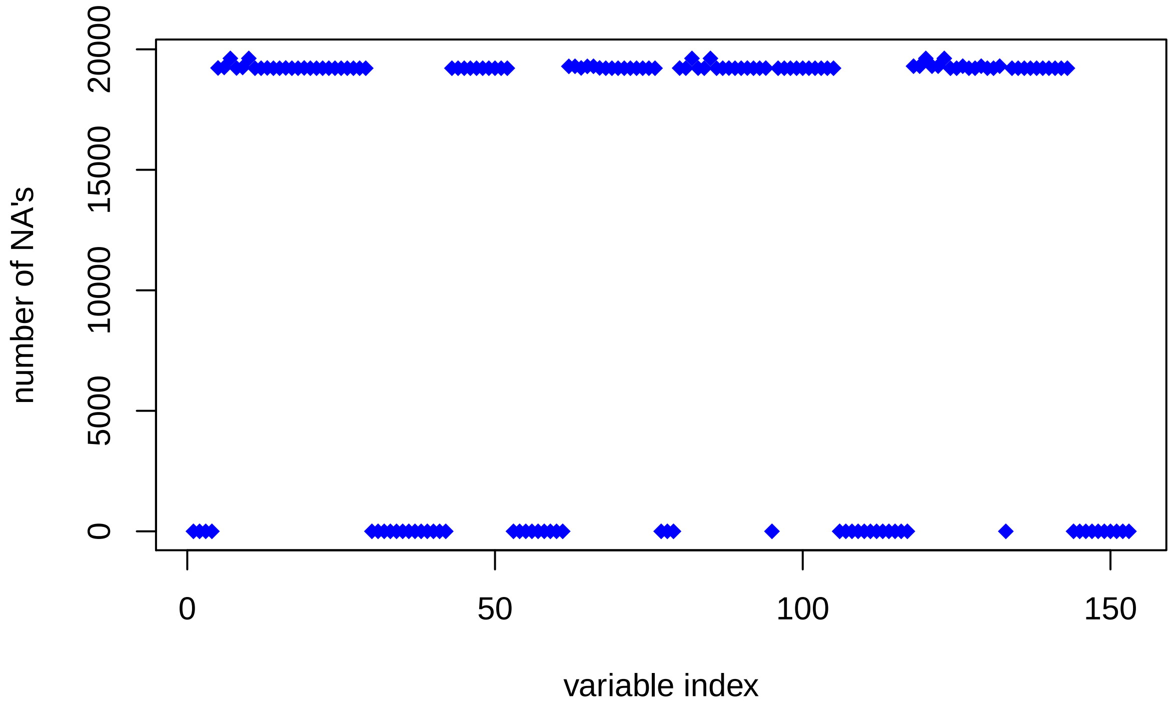
```
    {
    training <- read.csv(file = "pml-training.csv", na.strings = c("NA","#DIV/0!", ""))
    testing <- read.csv(file = "pml-testing.csv", na.strings = c("NA","#DIV/0!", ""))
}

# useless first 7 columns
training <- training[, -c(1:7)]
```

In this dataset there are indeed a lot of **NA** values. We could figure this fact by inspecting the occurance of the **NA** values in a plot. Code inside **for-loop** is able to check how many columns in the dataset have an high % of NA-values inside them. It is now clear that when NAs occur in a variable of the dataset, then almost the entire column is full of NAs: infact, the edge is at 97%.

We can inspect it by using a scatter plot.

```
plot(colSums(is.na(training)),xlab="variable index", ylab="number of NA's", pch=18, col="blue", cex=1.1
```



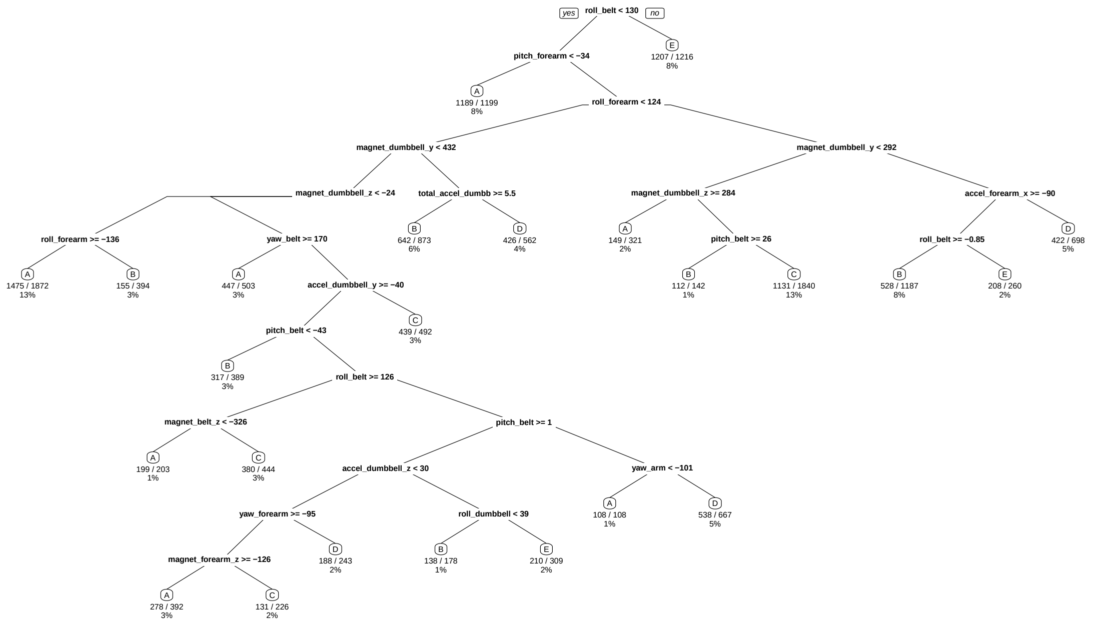
To build a predictive model, I must subset the training dataset into two subset, one called **subTrain** and the other **subTest**.

```
partition <- createDataPartition(y=training$classe, p=0.75, list=FALSE)
subTrain <- training[partition, ]
subTest <- training[-partition, ]
```

```
model1 <- rpart(classe ~ ., data=subTrain, method="class")

# Predicting:
prediction1 <- predict(model1, subTest, type = "class")

# Plot of the Decision Tree
rpart.plot(model1, main="Classification Tree", extra=102, under=TRUE, faclen=0)
```



Classification Tree