Prediction Assignment Writeup

Nicolás Rivera Garzón

11/1/2021

Machine Learning Course Project:

A. 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, your 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).

B. Environment set up

The first step is to load the neccesary libraries and set up a wd

```
rm(list=ls())
setwd("C:/Users/Nicolás Rivera/OneDrive/Documentos/Data Science Johns Hopkins University/Practical Machi
library(caret)
library(rpart)
library(rpart.plot)
library(randomForest)
library(corrplot)
library(rattle)
```

C. Data Loading and cleaning

Define the url containing the data and create a 70% training data set and a 30% test data set.

```
UrlTrain <- "http://d396qusza40orc.cloudfront.net/predmachlearn/pml-training.csv"
UrlTest  <- "http://d396qusza40orc.cloudfront.net/predmachlearn/pml-testing.csv"
training <- read.csv(url(UrlTrain))
testing  <- read.csv(url(UrlTest))
inTrain  <- createDataPartition(training$classe, p=0.7, list=FALSE)
TrainSet <- training[inTrain, ]
TestSet  <- training[-inTrain, ]
dim(TrainSet)
```

```
## [1] 13737   160
```

```
dim(TestSet)
```

```
## [1] 5885  160
```

Remove variables with zero variance

```
NZV <- nearZeroVar(TrainSet)
TrainSet <- TrainSet[, -NZV]
TestSet  <- TestSet[, -NZV]
dim(TrainSet)
```

```
## [1] 13737   105
```

```
dim(TestSet)
```

```
## [1] 5885  105
```

Remove variables that are mostly NA

```
AllNA    <- sapply(TrainSet, function(x) mean(is.na(x))) > 0.95
TrainSet <- TrainSet[, AllNA==FALSE]
TestSet  <- TestSet[, AllNA==FALSE]
dim(TrainSet)
```

```
## [1] 13737    59
```

```
dim(TestSet)
```

```
## [1] 5885   59
```

```
dim(TestSet)
```

```
## [1] 5885   59
```

D. Data Modeling

Random Forest

```
set.seed(12345)
controlRF <- trainControl(method="cv", number=3, verboseIter=FALSE)
modFitRandForest <- train(classe ~ ., data=TrainSet, method="rf",
                          trControl=controlRF)
modFitRandForest$finalModel
```

```
##
## Call:
##  randomForest(x = x, y = y, mtry = param$mtry)
##                Type of random forest: classification
##                      Number of trees: 500
## No. of variables tried at each split: 41
##
##         OOB estimate of  error rate: 0.01%
## Confusion matrix:
##      A    B    C    D    E  class.error
## A 3906    0    0    0    0 0.0000000000
## B    1 2657    0    0    0 0.0003762227
## C    0    0 2396    0    0 0.0000000000
## D    0    0    0 2252    0 0.0000000000
## E    0    0    0    0 2525 0.0000000000
```

E. Predict

```
predictTEST <- predict(modFitRandForest, newdata=testing)
predictTEST
```

```
##  [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
```