

Course Practical Machine Learning Peer Assessment

Oscar Vasquez, Date: 12-Nov-2020

The goal of this project is to predict the manner in which they did the exercise. This is the "classe" variable in the training set. I will use any of the other variables to predict with. I will create a report describing how you built your model, how I used cross validation, what I think the expected out of sample error is, and why I made the choices I did. I will also use your prediction model to predict 20 different test cases.

This study is possible thank to

<http://web.archive.org/web/20161224072740/http://groupware.les.inf.puc-rio.br/har>

(<http://web.archive.org/web/20161224072740/http://groupware.les.inf.puc-rio.br/har>) So thank you very much!

```
#Libraries
library(AppliedPredictiveModeling)
library(caret)
library(corrplot)
library(dplyr)
library(e1071)
library(knitr)
library(ggplot2)
library(randomForest)
library(rpart)
setwd("D:/Trabajo/R/Curso8")
set.seed(2)
#Load Data
TrainUrl<-"https://d396qusza40orc.cloudfront.net/predmachlearn/pml-training.csv"
TestUrl<- "https://d396qusza40orc.cloudfront.net/predmachlearn/pml-testing.csv"
TrainFile <- "./Data/pml-training.csv"
TestFile  <- "./Data/pml-testing.csv"
if (!file.exists("./Data")) {
  dir.create("./Data")}
if (!file.exists(TrainFile)) {
  download.file(TrainUrl, destfile=TrainFile)}
if (!file.exists(TestFile)) {
  download.file(TestUrl, destfile=TestFile)}
Train <- read.csv("./Data/pml-training.csv",header=T,sep="," ,
                 na.strings=c("NA",""))
Test <- read.csv("./Data/pml-testing.csv",header=T,sep="," ,
                 na.strings=c("NA",""))
dim(Train)
```

```
## [1] 19622  160
```

```
dim(Test)
```

```
## [1]  20 160
```

Files have 160 variables so its large to see data structure, so I will see just one data structure, lets say Train, Test should be the same structure.

```
str(Train)
```

```

## 'data.frame':    19622 obs. of  160 variables:
## $ X                : int  1 2 3 4 5 6 7 8 9 10 ...
## $ user_name         : Factor w/ 6 levels "adelmo","carlitos",...: 2 2 2 2 2 2 2 2 2 2
...
## $ raw_timestamp_part_1 : int  1323084231 1323084231 1323084231 1323084232 1323084232 1323
084232 1323084232 1323084232 1323084232 1323084232 ...
## $ raw_timestamp_part_2 : int  788290 808298 820366 120339 196328 304277 368296 440390 484
323 484434 ...
## $ cvtd_timestamp      : Factor w/ 20 levels "02/12/2011 13:32",...: 9 9 9 9 9 9 9 9 9 9
...
## $ new_window          : Factor w/ 2 levels "no","yes": 1 1 1 1 1 1 1 1 1 1 ...
## $ num_window          : int  11 11 11 12 12 12 12 12 12 12 ...
## $ roll_belt           : num  1.41 1.41 1.42 1.48 1.48 1.45 1.42 1.42 1.43 1.45 ...
## $ pitch_belt          : num  8.07 8.07 8.07 8.05 8.07 8.06 8.09 8.13 8.16 8.17 ...
## $ yaw_belt            : num  -94.4 -94.4 -94.4 -94.4 -94.4 -94.4 -94.4 -94.4 -94.4 -94.4
...
## $ total_accel_belt    : int  3 3 3 3 3 3 3 3 3 3 ...
## $ kurtosis_roll_belt  : Factor w/ 396 levels "-0.016850","-0.021024",...: NA NA NA NA NA
NA NA NA NA NA ...
## $ kurtosis_pitch_belt : Factor w/ 316 levels "-0.021887","-0.060755",...: NA NA NA NA NA
NA NA NA NA NA ...
## $ kurtosis_yaw_belt   : Factor w/ 1 level "#DIV/0!": NA NA NA NA NA NA NA NA NA NA ...
## $ skewness_roll_belt  : Factor w/ 394 levels "-0.003095","-0.010002",...: NA NA NA NA NA
NA NA NA NA NA ...
## $ skewness_roll_belt.1 : Factor w/ 337 levels "-0.005928","-0.005960",...: NA NA NA NA NA
NA NA NA NA NA ...
## $ skewness_yaw_belt   : Factor w/ 1 level "#DIV/0!": NA NA NA NA NA NA NA NA NA NA ...
## $ max_roll_belt       : num  NA NA NA NA NA NA NA NA NA NA ...
## $ max_pitch_belt      : int  NA NA NA NA NA NA NA NA NA NA ...
## $ max_yaw_belt        : Factor w/ 67 levels "-0.1","-0.2",...: NA NA NA NA NA NA NA NA NA
NA ...
## $ min_roll_belt       : num  NA NA NA NA NA NA NA NA NA NA ...
## $ min_pitch_belt      : int  NA NA NA NA NA NA NA NA NA NA ...
## $ min_yaw_belt        : Factor w/ 67 levels "-0.1","-0.2",...: NA NA NA NA NA NA NA NA NA
NA ...
## $ amplitude_roll_belt : num  NA NA NA NA NA NA NA NA NA NA ...
## $ amplitude_pitch_belt : int  NA NA NA NA NA NA NA NA NA NA ...
## $ amplitude_yaw_belt   : Factor w/ 3 levels "#DIV/0!","0.00",...: NA NA NA NA NA NA NA NA
NA NA ...
## $ var_total_accel_belt : num  NA NA NA NA NA NA NA NA NA NA ...
## $ avg_roll_belt       : num  NA NA NA NA NA NA NA NA NA NA ...
## $ stddev_roll_belt    : num  NA NA NA NA NA NA NA NA NA NA ...
## $ var_roll_belt       : num  NA NA NA NA NA NA NA NA NA NA ...
## $ avg_pitch_belt      : num  NA NA NA NA NA NA NA NA NA NA ...
## $ stddev_pitch_belt   : num  NA NA NA NA NA NA NA NA NA NA ...
## $ var_pitch_belt      : num  NA NA NA NA NA NA NA NA NA NA ...
## $ avg_yaw_belt        : num  NA NA NA NA NA NA NA NA NA NA ...
## $ stddev_yaw_belt     : num  NA NA NA NA NA NA NA NA NA NA ...
## $ var_yaw_belt        : num  NA NA NA NA NA NA NA NA NA NA ...
## $ gyros_belt_x        : num  0 0.02 0 0.02 0.02 0.02 0.02 0.02 0.02 0.03 ...
## $ gyros_belt_y        : num  0 0 0 0 0.02 0 0 0 0 0 ...
## $ gyros_belt_z        : num  -0.02 -0.02 -0.02 -0.03 -0.02 -0.02 -0.02 -0.02 -0.02 0 ...
## $ accel_belt_x        : int  -21 -22 -20 -22 -21 -21 -22 -22 -20 -21 ...

```

```

## $ accel_belt_y      : int  4 4 5 3 2 4 3 4 2 4 ...
## $ accel_belt_z      : int  22 22 23 21 24 21 21 21 24 22 ...
## $ magnet_belt_x     : int  -3 -7 -2 -6 -6 0 -4 -2 1 -3 ...
## $ magnet_belt_y     : int  599 608 600 604 600 603 599 603 602 609 ...
## $ magnet_belt_z     : int  -313 -311 -305 -310 -302 -312 -311 -313 -312 -308 ...
## $ roll_arm          : num  -128 -128 -128 -128 -128 -128 -128 -128 -128 -128 ...
## $ pitch_arm         : num  22.5 22.5 22.5 22.1 22.1 22 21.9 21.8 21.7 21.6 ...
## $ yaw_arm           : num  -161 -161 -161 -161 -161 -161 -161 -161 -161 -161 ...
## $ total_accel_arm    : int  34 34 34 34 34 34 34 34 34 34 ...
## $ var_accel_arm     : num  NA NA NA NA NA NA NA NA NA NA ...
## $ avg_roll_arm      : num  NA NA NA NA NA NA NA NA NA NA ...
## $ stddev_roll_arm   : num  NA NA NA NA NA NA NA NA NA NA ...
## $ var_roll_arm      : num  NA NA NA NA NA NA NA NA NA NA ...
## $ avg_pitch_arm     : num  NA NA NA NA NA NA NA NA NA NA ...
## $ stddev_pitch_arm  : num  NA NA NA NA NA NA NA NA NA NA ...
## $ var_pitch_arm     : num  NA NA NA NA NA NA NA NA NA NA ...
## $ avg_yaw_arm       : num  NA NA NA NA NA NA NA NA NA NA ...
## $ stddev_yaw_arm    : num  NA NA NA NA NA NA NA NA NA NA ...
## $ var_yaw_arm       : num  NA NA NA NA NA NA NA NA NA NA ...
## $ gyros_arm_x       : num  0 0.02 0.02 0.02 0 0.02 0 0.02 0.02 0.02 ...
## $ gyros_arm_y       : num  0 -0.02 -0.02 -0.03 -0.03 -0.03 -0.03 -0.02 -0.03 -0.03 ...
## $ gyros_arm_z       : num  -0.02 -0.02 -0.02 0.02 0 0 0 0 -0.02 -0.02 ...
## $ accel_arm_x       : int  -288 -290 -289 -289 -289 -289 -289 -289 -288 -288 ...
## $ accel_arm_y       : int  109 110 110 111 111 111 111 111 109 110 ...
## $ accel_arm_z       : int  -123 -125 -126 -123 -123 -122 -125 -124 -122 -124 ...
## $ magnet_arm_x      : int  -368 -369 -368 -372 -374 -369 -373 -372 -369 -376 ...
## $ magnet_arm_y      : int  337 337 344 344 337 342 336 338 341 334 ...
## $ magnet_arm_z      : int  516 513 513 512 506 513 509 510 518 516 ...
## $ kurtosis_roll_arm : Factor w/ 329 levels "-0.02438","-0.04190",...: NA NA NA NA NA NA
NA NA NA NA ...
## $ kurtosis_pitch_arm : Factor w/ 327 levels "-0.00484","-0.01311",...: NA NA NA NA NA NA
NA NA NA NA ...
## $ kurtosis_yaw_arm   : Factor w/ 394 levels "-0.01548","-0.01749",...: NA NA NA NA NA NA
NA NA NA NA ...
## $ skewness_roll_arm  : Factor w/ 330 levels "-0.00051","-0.00696",...: NA NA NA NA NA NA
NA NA NA NA ...
## $ skewness_pitch_arm : Factor w/ 327 levels "-0.00184","-0.01185",...: NA NA NA NA NA NA
NA NA NA NA ...
## $ skewness_yaw_arm   : Factor w/ 394 levels "-0.00311","-0.00562",...: NA NA NA NA NA NA
NA NA NA NA ...
## $ max_roll_arm       : num  NA NA NA NA NA NA NA NA NA NA ...
## $ max_pitch_arm      : num  NA NA NA NA NA NA NA NA NA NA ...
## $ max_yaw_arm        : int  NA NA NA NA NA NA NA NA NA NA ...
## $ min_roll_arm       : num  NA NA NA NA NA NA NA NA NA NA ...
## $ min_pitch_arm      : num  NA NA NA NA NA NA NA NA NA NA ...
## $ min_yaw_arm        : int  NA NA NA NA NA NA NA NA NA NA ...
## $ amplitude_roll_arm : num  NA NA NA NA NA NA NA NA NA NA ...
## $ amplitude_pitch_arm : num  NA NA NA NA NA NA NA NA NA NA ...
## $ amplitude_yaw_arm  : int  NA NA NA NA NA NA NA NA NA NA ...
## $ roll_dumbbell      : num  13.1 13.1 12.9 13.4 13.4 ...
## $ pitch_dumbbell     : num  -70.5 -70.6 -70.3 -70.4 -70.4 ...
## $ yaw_dumbbell       : num  -84.9 -84.7 -85.1 -84.9 -84.9 ...
## $ kurtosis_roll_dumbbell : Factor w/ 397 levels "-0.0035","-0.0073",...: NA NA NA NA NA NA
A NA NA NA ...

```

```
## $ kurtosis_pitch_dumbbell : Factor w/ 400 levels "-0.0163","-0.0233",...: NA NA NA NA NA NA N
A NA NA NA ...
## $ kurtosis_yaw_dumbbell : Factor w/ 1 level "#DIV/0!": NA NA NA NA NA NA NA NA NA NA NA ...
## $ skewness_roll_dumbbell : Factor w/ 400 levels "-0.0082","-0.0096",...: NA NA NA NA NA NA NA N
A NA NA NA ...
## $ skewness_pitch_dumbbell : Factor w/ 401 levels "-0.0053","-0.0084",...: NA NA NA NA NA NA NA N
A NA NA NA ...
## $ skewness_yaw_dumbbell : Factor w/ 1 level "#DIV/0!": NA NA NA NA NA NA NA NA NA NA NA ...
## $ max_roll_dumbbell : num NA NA NA NA NA NA NA NA NA NA NA ...
## $ max_pitch_dumbbell : num NA NA NA NA NA NA NA NA NA NA NA ...
## $ max_yaw_dumbbell : Factor w/ 72 levels "-0.1","-0.2",...: NA NA NA NA NA NA NA NA NA NA
NA ...
## $ min_roll_dumbbell : num NA NA NA NA NA NA NA NA NA NA NA ...
## $ min_pitch_dumbbell : num NA NA NA NA NA NA NA NA NA NA NA ...
## $ min_yaw_dumbbell : Factor w/ 72 levels "-0.1","-0.2",...: NA NA NA NA NA NA NA NA NA NA
NA ...
## $ amplitude_roll_dumbbell : num NA NA NA NA NA NA NA NA NA NA NA ...
## [list output truncated]
```

Now I will clean data, for random forest I will select record with more than 60% of data and Variable "X" has a N row indicator so we should not use it to predict classes.

```
Train_clean <- Train[, -1]
Test_clean <- Test[, -1]
goodColumns <- c((colSums(!is.na(Train_clean[, ncol(Train_clean)])) >= 0.6*nrow(Train_clean)))
Train_clean <- Train_clean[, goodColumns]
Test_clean <- Test_clean[, goodColumns]
dim(Train)
```

```
## [1] 19622 160
```

```
dim(Train_clean)
```

```
## [1] 19622 59
```

```
dim(Test)
```

```
## [1] 20 160
```

```
dim(Test_clean)
```

```
## [1] 20 59
```

Lets select training and test for modeling, 80/20.

```
TestIndex = createDataPartition(Train_clean$classe, p=0.80, list=F)
Training = Train_clean[TestIndex,]
Testing = Train_clean[-TestIndex,]
dim(Training)
```

```
## [1] 15699    59
```

```
dim(Testing)
```

```
## [1] 3923    59
```

Lest Run a Random Forest

```
RF1 <- randomForest(classe~.,data=Training)
RF1
```

```
##
## Call:
## randomForest(formula = classe ~ ., data = Training)
##           Type of random forest: classification
##           Number of trees: 500
## No. of variables tried at each split: 7
##
##           OOB estimate of  error rate: 0.1%
## Confusion matrix:
##      A      B      C      D      E  class.error
## A 4464      0      0      0      0 0.0000000000
## B   23036      0      0      0 0.0006583278
## C      0      42733      1      0 0.0018261505
## D      0      0      32567      3 0.0023319083
## E      0      0      0      22884 0.0006930007
```

```
importance(RF1)
```

| ## | MeanDecreaseGini |
|-------------------------|------------------|
| ## user_name | 124.714187 |
| ## raw_timestamp_part_1 | 1287.563554 |
| ## raw_timestamp_part_2 | 11.843285 |
| ## cvtd_timestamp | 1885.858647 |
| ## new_window | 0.187356 |
| ## num_window | 767.900905 |
| ## roll_belt | 732.000079 |
| ## pitch_belt | 388.931283 |
| ## yaw_belt | 474.136224 |
| ## total_accel_belt | 157.993650 |
| ## gyros_belt_x | 50.489938 |
| ## gyros_belt_y | 68.157357 |
| ## gyros_belt_z | 169.695910 |
| ## accel_belt_x | 82.041923 |
| ## accel_belt_y | 83.395782 |
| ## accel_belt_z | 254.460873 |
| ## magnet_belt_x | 139.866665 |
| ## magnet_belt_y | 268.587345 |
| ## magnet_belt_z | 239.380200 |
| ## roll_arm | 167.775408 |
| ## pitch_arm | 74.271181 |
| ## yaw_arm | 104.317037 |
| ## total_accel_arm | 36.625711 |
| ## gyros_arm_x | 49.902705 |
| ## gyros_arm_y | 54.568427 |
| ## gyros_arm_z | 21.463844 |
| ## accel_arm_x | 121.894913 |
| ## accel_arm_y | 63.910969 |
| ## accel_arm_z | 50.710504 |
| ## magnet_arm_x | 116.687239 |
| ## magnet_arm_y | 92.265414 |
| ## magnet_arm_z | 72.152481 |
| ## roll_dumbbell | 250.156992 |
| ## pitch_dumbbell | 110.176982 |
| ## yaw_dumbbell | 155.343968 |
| ## total_accel_dumbbell | 155.685764 |
| ## gyros_dumbbell_x | 55.656434 |
| ## gyros_dumbbell_y | 127.874520 |
| ## gyros_dumbbell_z | 29.377996 |
| ## accel_dumbbell_x | 153.958842 |
| ## accel_dumbbell_y | 238.581992 |
| ## accel_dumbbell_z | 176.035494 |
| ## magnet_dumbbell_x | 302.730618 |
| ## magnet_dumbbell_y | 444.111376 |
| ## magnet_dumbbell_z | 406.984073 |
| ## roll_forearm | 316.435284 |
| ## pitch_forearm | 401.846014 |
| ## yaw_forearm | 70.262309 |
| ## total_accel_forearm | 44.296294 |
| ## gyros_forearm_x | 29.386091 |
| ## gyros_forearm_y | 50.865507 |
| ## gyros_forearm_z | 33.091312 |

```
## accel_forearm_x      157.200388
## accel_forearm_y      58.863882
## accel_forearm_z     122.728484
## magnet_forearm_x     95.585967
## magnet_forearm_y     90.915598
## magnet_forearm_z    118.879014
```

It seems pretty good, really high accuracy, let's classify Test, for this a trick, predict command will not work with a difference in data types, here we can use R coercion. I will add one case from Training, in order to give same types to all our variables in Test_clean, and when I call predict I will exclude this record

```
Test_clean <- rbind(Training[100,1:58], Test_clean[1:58])

Pred_RF1 <- predict(RF1,newdata=Test_clean[-1,])

Pred_RF1
```

```
##  1  2  3  4  5  6  7  8  9 10 11 12 13 14 15 16 17 18 19 20
##  B  A  B  A  A  E  D  B  A  A  B  C  B  A  E  E  A  B  B  B
## Levels: A B C D E
```

Now a little piece of code to create files with outcomes, remember Knit has its own Knit directory, so set it properly

```
for(i in 1:20){
  filename = paste0("./Data/Answer/Output_id_",i,".txt")
  write.table(Pred_RF1[i],file=filename,quote=FALSE,row.names=FALSE,col.names=FALSE)
}
```