Final Project for Data Science Course 8 Week 4

NguyenDuy

19 February 2017

Reading data

There are two types of variable in data: raw variable (about 19500 data point and 20 data point for each variable in training and testing data set, respectively) and summary variable (402 data point and 0 data point for each variable in training and testing data set, respectively). There are 60 raw variable and 100 summary variable in both set. Because in testing data set, only raw variables are provided, thus we only pick out raw variable to construct training model.

```
data <- read.csv("pml-training.csv")              #Read training data
data[data == ""] <- NA                             #Clean-up testing data
testingData <- read.csv("pml-testing.csv")        #Read testing data
testingData[testingData == ""] <- NA              #Clean-up testing data

#Cross table of variable and number of variable in training data,
#showing 60 raw variables and 100 summary variables
table(apply(data, 2, function(x){sum(!is.na(x))}))
```

```
##
##   406 19622
##   100    60
```

```
#Cross table of variable and number of variable in testing data,
#showing 60 raw variables and 100 summary variables
table(apply(testingData, 2, function(x){sum(!is.na(x))}))
```

```
##
##   0  20
## 100  60
```

```
#Sample of names of raw variables
apply(data, 2, function(x){sum(!is.na(x))}) %>% .[. == 19622] %>% names %>% head
```

```
## [1] "X"                    "user_name"            "raw_timestamp_part_1"
## [4] "raw_timestamp_part_2" "cvtd_timestamp"       "new_window"
```

```
#Sample of names of summary variables
apply(data, 2, function(x){sum(!is.na(x))}) %>% .[. == 406] %>% names %>% head
```

```
## [1] "kurtosis_roll_belt"   "kurtosis_picth_belt"  "kurtosis_yaw_belt"
## [4] "skewness_roll_belt"   "skewness_roll_belt.1" "skewness_yaw_belt"
```

Extract data into two set: raw variable (set2) and summary variable (set1). We only use set 2 variable in this project.

```
set1 <- names(data)[apply(data, 2, function(x){sum(!is.na(x))})==406]
set2 <- names(data)[apply(data, 2, function(x){sum(!is.na(x))})>406]
```

Cleaning data

From raw data set (set2), we exclude column 1-7, which is only identifier and not real data variable.

```
data[,set2][,c(-1:-7)]  -> extract2
```

Model construction

We use random forest machine learning method to construct identifier model.

```
library(randomForest)
model2 <- randomForest(classe~., extract2)
model2
```

```
##
## Call:
##  randomForest(formula = classe ~ ., data = extract2)
##                Type of random forest: classification
##                      Number of trees: 500
## No. of variables tried at each split: 7
##
##         OOB estimate of  error rate: 0.3%
## Confusion matrix:
##      A    B    C    D    E  class.error
## A 5578    1    0    0    1 0.0003584229
## B    9 3785    3    0    0 0.0031603898
## C    0   11 3409    2    0 0.0037989480
## D    0    0   23 3191    2 0.0077736318
## E    0    0    0    6 3601 0.0016634322
```

Predicting test set

```
testingDataExtract <- testingData[, names(extract2)[-length(names(extract2))]]
result <- predict(model2, testingDataExtract)
tomatch <- c("A", "B", "C", "D", "E")
finalResult <- sapply(result, function(x){tomatch[x]})
finalResult
```

```
##  [1] "B" "A" "B" "A" "A" "E" "D" "B" "A" "A" "B" "C" "B" "A" "E" "E" "A"
## [18] "B" "B" "B"
```

Degree of accuracy

```
k <- 0
for(i in 1:5){k <- k + model2$confusion[i,i]}
k/sum(model2$confusion[,1:5])
```

```
## [1] 0.9970441
```

Finding important variable

List of variable sorted by the degree of imporant are shown below, which the most imporant as row 1

```
varImportance <- importance(model2, type = 2)
varImportance[order(-varImportance),]
```

```
##            roll_belt            yaw_belt        pitch_forearm
##           1260.94087           906.72709            801.18286
##    magnet_dumbbell_z   magnet_dumbbell_y           pitch_belt
##            761.29824           695.57549            694.44510
##         roll_forearm   magnet_dumbbell_x     accel_dumbbell_y
##            622.08928           477.79441            413.17108
##        roll_dumbbell        accel_belt_z         magnet_belt_z
##            409.34824           403.11817            396.22404
##        magnet_belt_y    accel_dumbbell_z      accel_forearm_x
##            370.08028           336.93902            319.70536
##             roll_arm         gyros_belt_z     magnet_forearm_z
##            313.69908           312.08810            286.81908
## total_accel_dumbbell        yaw_dumbbell      gyros_dumbbell_y
##            265.81978           265.27781            256.59256
##        magnet_belt_x         accel_arm_x         magnet_arm_x
##            254.33541           249.39330            244.99918
##     accel_dumbbell_x     accel_forearm_z              yaw_arm
##            244.38144           234.42127            228.88651
##         magnet_arm_y    magnet_forearm_y     total_accel_belt
##            224.59700           221.11397            212.85282
##     magnet_forearm_x        magnet_arm_z            pitch_arm
##            206.75317           187.28522            176.08424
##          yaw_forearm      pitch_dumbbell          accel_arm_y
##            175.79900           175.28250            160.39207
##      accel_forearm_y         gyros_arm_y          gyros_arm_x
##            143.12395           136.74591            133.67342
##         accel_belt_y         accel_arm_z     gyros_dumbbell_x
##            133.12769           129.25058            127.04456
##      gyros_forearm_y        gyros_belt_y         accel_belt_x
##            119.94187           112.79408            111.00654
##   total_accel_forearm     total_accel_arm          gyros_belt_x
##            108.05437           100.15243             94.66080
##      gyros_forearm_z     gyros_dumbbell_z      gyros_forearm_x
##             83.68985            81.38829             73.77335
##          gyros_arm_z
##             58.18472
```