Practical machine learning assignment

Hena

Wednesday, March 18, 2015

**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.

**Data**

The training data for this project are available here:

<https://d396qusza40orc.cloudfront.net/predmachlearn/pml-training.csv>

The test data are available here:

<https://d396qusza40orc.cloudfront.net/predmachlearn/pml-testing.csv>

**Libraries**

I used following libraries:

library(Hmisc)  
library(caret)  
library(randomForest)  
library(foreach)  
library(doParallel)

**Loading Training Set**

read.csv ("trainingset.csv") read.csv("testingtwo.csv")

**Cleaning the data set**

First all blank('""'), '#DIV/0' and 'NA' values are converted to 'NA'. Any Columns containing 'NA' are removed from both downloaded data sets. The features user\_name raw\_timestamp\_part\_1 raw\_timestamp\_part\_2 cvtd\_timestamp new\_window num\_window are not related to calculations and are removed form the downloaded data.

testingsetnew <-read.csv("trainingset.csv", na.strings=c("NA","#DIV/0!", "")) testingtwonew <-read.csv("testingtwo.csv" , na.strings=c("NA", "#DIV/0!", "")) testingsetfinal<-testingsetnew[,colSums(is.na(testingsetnew)) == 0] testingtwofinal <-testingtwonew[,colSums(is.na(testingtwonew)) == 0] testingsetfinal <-testingsetfinal[,-c(1:7)] testingtwofinal <-testingtwofinal[,-c(1:7)]

# Create Test and Training Data set

set.seed(990)  
  
inTraining.matrix <- createDataPartition(testingsetfinal$classe, p = 0.35, list = FALSE)  
training.data <- testingsetfinal[inTraining.matrix, ]  
testing.data <- testingsetfinal[-inTraining.matrix, ]

**Random Forest**

Run Random Forest to develop prediction model

r}rf\_model<-train(classe~.,data=training.data,method="rf",  
 trControl=trainControl(method="cv",number=5),  
 prox=TRUE,allowParallel=TRUE)  
print(rf\_model)

Random Forest

6869 samples 52 predictor 5 classes: 'A', 'B', 'C', 'D', 'E'

No pre-processing Resampling: Cross-Validated (5 fold)

Summary of sample sizes: 5497, 5495, 5494, 5494, 5496

Resampling results across tuning parameters:

|  |
| --- |
| mtry | Accuracy | Kappa | Accuracy SD | Kappa SD |
| 2 | 0.9784566 | 0.9727407 | 0.004662349 | 0.005898092 |
| 27 | 0.9809298 | 0.9758719 | 0.002072385 | 0.002620205 |
| 52 | 0.9702986 | 0.9624213 | 0.004667295 | 0.005898576 |

Accuracy was used to select the optimal model using the largest value. The final value used for the model was mtry = 27. >

print(rf\_model$finalModel)

Call: randomForest(x = x, y = y, mtry = param$mtry, proximity = TRUE, allowParallel = TRUE) Type of random forest: classification Number of trees: 500 No. of variables tried at each split: 27

OOB estimate of error rate: 1.79%

Confusion matrix:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| A | B | C | D | E class.error |
| A | 1944 | 5 | 2 | 0 | 2 0.004608295 |
| B | 27 | 1289 | 11 | 2 | 0 0.030097818 |
| C | 0 | 23 | 1165 | 10 | 0 0.027545910 |
| D | 1 | 0 | 18 | 1104 | 3 0.019538188 |
| E | 2 | 3 | 7 | 1244 | 0.015043547 |

save(rf\_model, file="rf\_model$finalModel.RData")

**Run model against TEST SET and Predict outcome**

load(file="rf\_model$finalModel.RData", verbose=TRUE)  
Loading objects:  
 rf\_model  
  
newdata <-testingtwofinal  
final <- predict(rf\_model, newdata)  
  
newdata <- cbind(final , newdata)  
subset(newdata, select=names(newdata)[grep("belt|[^(fore)]arm|dumbbell|forearm", names(newdata), invert=TRUE)])

|  |  |  |
| --- | --- | --- |
| final problem\_id | Value | Test no: |
| 1 | B | 1 |
| 2 | A | 2 |
| 3 | B | 3 |
| 4 | A | 4 |
| 5 | A | 5 |
| 6 | E | 6 |
| 7 | D | 7 |
| 8 | D | 8 |
| 9 | A | 9 |
| 10 | A | 10 |
| 11 | B | 11 |
| 12 | C | 12 |
| 13 | B | 13 |
| 14 | A | 14 |
| 15 | E | 15 |
| 16 | E | 16 |
| 17 | A | 17 |
| 18 | B | 18 |
| 19 | B | 19 |
| 20 | B | 20 |

**Write answers to seperate text files**

```pml\_write\_files = function(x){ n = length(x) path <- "ML\_caret" for(i in 1:n){ filename = paste0("problem\_id\_",i,".txt")

write.table(x[i],file=file.path(path, filename),quote=FALSE,row.names=FALSE,col.names=FALSE)

} }

pml\_write\_files(final)```