Predict Exercise Pattern Using Automatic Activity Detection Sensor Data

Summary

Wearable sensors like Jawbone Up, Nike FuelBand, and Fitbit enables large amount of data collection about personal activity relatively inexpensively. The goal of this analysis is to automatically recognize the activity type by machine learning model using this data.Random forest model models are built using training dataset. Cross validation is used to validate the results.

Data Source

- The training data : https://d396qusza40orc.cloudfront.net/predmachlearn/pml-training.csv
- The test data : https://d396qusza40orc.cloudfront.net/predmachlearn/pml-testing.csv
- Source : http://groupware.les.inf.puc-rio.br/har

Exploratory Data Analysis

Load data from CSV files.

```
trainData <- read.csv("pml-training.csv", stringsAsFactor=FALSE)
testData <- read.csv("pml-testing.csv", stringsAsFactor=FALSE)
```

Check the data quality of taining data

```
sum(complete.cases(trainData))
```

```
## [1] 406
```

```
sum(complete.cases(testData))
```

```
## [1] 0
```

There are few complete cases. The data set is reduced by applying filters to remove empty colums and non significant values.

```
trainData1 <- trainData[,sapply(trainData, is.numeric)]
trainData1 <- trainData1[,sapply(trainData1, function(x) sum(is.na(x)) == 0)]
trainData1$classe <- as.factor(trainData$classe)
sum(complete.cases(trainData1))
```

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

```
testData1 <- testData[,sapply(testData, is.numeric)]
testData1 <- testData1[,sapply(testData1, function(x) sum(is.na(x)) == 0)]
#testData1$problem_id <- as.factor(testData$problem_id)
sum(complete.cases(testData1))
```

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

Create test/validation partitions

```
trainIndex <- createDataPartition(trainData1$classe, p = 0.6, list = FALSE)
training <- trainData1[trainIndex, ]
validation <- trainData1[-trainIndex, ]
```

Cross-Validation

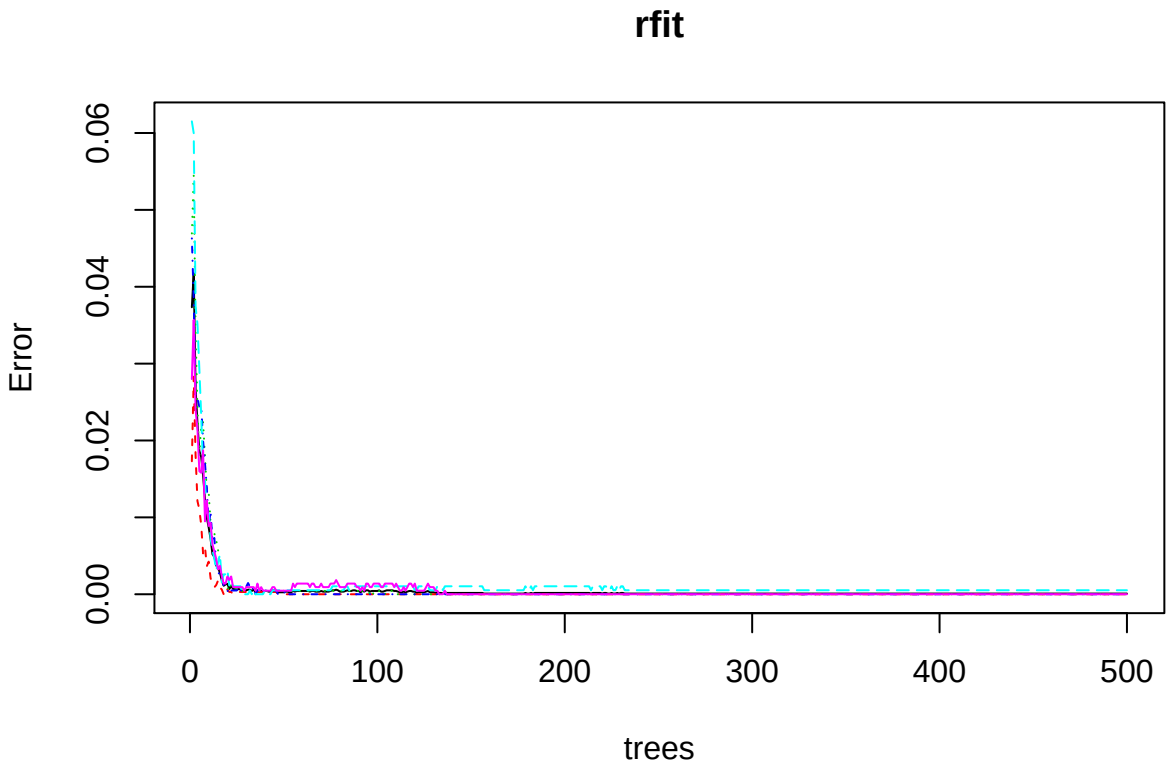
```
rfit <- randomForest(classe ~ .,data = training, importance = TRUE); rfit
```

```
##
## Call:
##  randomForest(formula = classe ~ ., data = training, importance = TRUE)
##                Type of random forest: classification
##                      Number of trees: 500
## No. of variables tried at each split: 7
##
##         OOB estimate of  error rate: 0.01%
## Confusion matrix:
##      A    B    C    D    E class.error
## A 3348    0    0    0    0   0.0000000
## B    0 2279    0    0    0   0.0000000
## C    0    0 2054    0    0   0.0000000
## D    0    0    0 1929    1   0.0005181
## E    0    0    0    0 2165   0.0000000
```

```
cv <- confusionMatrix(predict(rfit, validation), validation$classe)
```

The cross-validation accuracy is 99.9873%.

```
plot(rfit)
```


rfit
Error
0.00
0.02
0.04
0.06
0
100
200
300
400
500
trees