PMLweek4CourseProject

Haowei Song

September 14, 2017

Introduction

Using data from from this source: http://groupware.les.inf.puc-rio.br/har, 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. Any of the other variables could be used to predict with. Different machine learning model need to be built and tested with cross validation. Sample error will also be estimated. At last, the best prediction model will be used to predict 20 different test cases.

Data Input and Exploratory Analysis

```
## Loading required package: lattice
```

```
## Loading required package: ggplot2
```

```
## Rattle: A free graphical interface for data science with R.
## XXXX 5.1.0 Copyright (c) 2006-2017 Togaware Pty Ltd.
## Type 'rattle()' to shake, rattle, and roll your data.
```

```
## randomForest 4.6-12
```

```
## Type rfNews() to see new features/changes/bug fixes.
```

```
##
## Attaching package: 'randomForest'
```

```
## The following object is masked from 'package:rattle':
##
##     importance
```

```
## The following object is masked from 'package:ggplot2':
##
##     margin
```

Read the training and testing data set

```
train_data <- read.csv("pml-training.csv")
test_data <- read.csv("pml-testing.csv")
```

Exploratory Analysis

```
#Training data set
colnames(train_data)
```

```
##   [1] "X"                       "user_name"
##   [3] "raw_timestamp_part_1"    "raw_timestamp_part_2"
##   [5] "cvtd_timestamp"          "new_window"
##   [7] "num_window"              "roll_belt"
```

```
##   [9] "pitch_belt"               "yaw_belt"
##  [11] "total_accel_belt"         "kurtosis_roll_belt"
##  [13] "kurtosis_picth_belt"      "kurtosis_yaw_belt"
##  [15] "skewness_roll_belt"       "skewness_roll_belt.1"
##  [17] "skewness_yaw_belt"        "max_roll_belt"
##  [19] "max_picth_belt"           "max_yaw_belt"
##  [21] "min_roll_belt"            "min_pitch_belt"
##  [23] "min_yaw_belt"             "amplitude_roll_belt"
##  [25] "amplitude_pitch_belt"     "amplitude_yaw_belt"
##  [27] "var_total_accel_belt"     "avg_roll_belt"
##  [29] "stddev_roll_belt"         "var_roll_belt"
##  [31] "avg_pitch_belt"           "stddev_pitch_belt"
##  [33] "var_pitch_belt"           "avg_yaw_belt"
##  [35] "stddev_yaw_belt"          "var_yaw_belt"
##  [37] "gyros_belt_x"             "gyros_belt_y"
##  [39] "gyros_belt_z"             "accel_belt_x"
##  [41] "accel_belt_y"             "accel_belt_z"
##  [43] "magnet_belt_x"            "magnet_belt_y"
##  [45] "magnet_belt_z"            "roll_arm"
##  [47] "pitch_arm"                "yaw_arm"
##  [49] "total_accel_arm"          "var_accel_arm"
##  [51] "avg_roll_arm"             "stddev_roll_arm"
##  [53] "var_roll_arm"             "avg_pitch_arm"
##  [55] "stddev_pitch_arm"         "var_pitch_arm"
##  [57] "avg_yaw_arm"              "stddev_yaw_arm"
##  [59] "var_yaw_arm"              "gyros_arm_x"
##  [61] "gyros_arm_y"              "gyros_arm_z"
##  [63] "accel_arm_x"              "accel_arm_y"
##  [65] "accel_arm_z"              "magnet_arm_x"
##  [67] "magnet_arm_y"             "magnet_arm_z"
##  [69] "kurtosis_roll_arm"        "kurtosis_picth_arm"
##  [71] "kurtosis_yaw_arm"         "skewness_roll_arm"
##  [73] "skewness_pitch_arm"       "skewness_yaw_arm"
##  [75] "max_roll_arm"             "max_picth_arm"
##  [77] "max_yaw_arm"              "min_roll_arm"
##  [79] "min_pitch_arm"            "min_yaw_arm"
##  [81] "amplitude_roll_arm"       "amplitude_pitch_arm"
##  [83] "amplitude_yaw_arm"        "roll_dumbbell"
##  [85] "pitch_dumbbell"           "yaw_dumbbell"
##  [87] "kurtosis_roll_dumbbell"   "kurtosis_picth_dumbbell"
##  [89] "kurtosis_yaw_dumbbell"    "skewness_roll_dumbbell"
##  [91] "skewness_pitch_dumbbell"  "skewness_yaw_dumbbell"
##  [93] "max_roll_dumbbell"        "max_picth_dumbbell"
##  [95] "max_yaw_dumbbell"         "min_roll_dumbbell"
##  [97] "min_pitch_dumbbell"       "min_yaw_dumbbell"
##  [99] "amplitude_roll_dumbbell"  "amplitude_pitch_dumbbell"
## [101] "amplitude_yaw_dumbbell"   "total_accel_dumbbell"
## [103] "var_accel_dumbbell"       "avg_roll_dumbbell"
## [105] "stddev_roll_dumbbell"     "var_roll_dumbbell"
## [107] "avg_pitch_dumbbell"       "stddev_pitch_dumbbell"
## [109] "var_pitch_dumbbell"       "avg_yaw_dumbbell"
## [111] "stddev_yaw_dumbbell"      "var_yaw_dumbbell"
## [113] "gyros_dumbbell_x"         "gyros_dumbbell_y"
## [115] "gyros_dumbbell_z"         "accel_dumbbell_x"
```

```
## [117] "accel_dumbbell_y"          "accel_dumbbell_z"
## [119] "magnet_dumbbell_x"         "magnet_dumbbell_y"
## [121] "magnet_dumbbell_z"         "roll_forearm"
## [123] "pitch_forearm"             "yaw_forearm"
## [125] "kurtosis_roll_forearm"     "kurtosis_picth_forearm"
## [127] "kurtosis_yaw_forearm"      "skewness_roll_forearm"
## [129] "skewness_pitch_forearm"    "skewness_yaw_forearm"
## [131] "max_roll_forearm"          "max_picth_forearm"
## [133] "max_yaw_forearm"           "min_roll_forearm"
## [135] "min_pitch_forearm"         "min_yaw_forearm"
## [137] "amplitude_roll_forearm"    "amplitude_pitch_forearm"
## [139] "amplitude_yaw_forearm"     "total_accel_forearm"
## [141] "var_accel_forearm"         "avg_roll_forearm"
## [143] "stddev_roll_forearm"       "var_roll_forearm"
## [145] "avg_pitch_forearm"         "stddev_pitch_forearm"
## [147] "var_pitch_forearm"         "avg_yaw_forearm"
## [149] "stddev_yaw_forearm"        "var_yaw_forearm"
## [151] "gyros_forearm_x"           "gyros_forearm_y"
## [153] "gyros_forearm_z"           "accel_forearm_x"
## [155] "accel_forearm_y"           "accel_forearm_z"
## [157] "magnet_forearm_x"          "magnet_forearm_y"
## [159] "magnet_forearm_z"          "classe"
```

```
#Testing data set
colnames(test_data)
```

```
##   [1] "X"                        "user_name"
##   [3] "raw_timestamp_part_1"     "raw_timestamp_part_2"
##   [5] "cvtd_timestamp"           "new_window"
##   [7] "num_window"               "roll_belt"
##   [9] "pitch_belt"               "yaw_belt"
##  [11] "total_accel_belt"         "kurtosis_roll_belt"
##  [13] "kurtosis_picth_belt"      "kurtosis_yaw_belt"
##  [15] "skewness_roll_belt"       "skewness_roll_belt.1"
##  [17] "skewness_yaw_belt"        "max_roll_belt"
##  [19] "max_picth_belt"           "max_yaw_belt"
##  [21] "min_roll_belt"            "min_pitch_belt"
##  [23] "min_yaw_belt"             "amplitude_roll_belt"
##  [25] "amplitude_pitch_belt"     "amplitude_yaw_belt"
##  [27] "var_total_accel_belt"     "avg_roll_belt"
##  [29] "stddev_roll_belt"         "var_roll_belt"
##  [31] "avg_pitch_belt"           "stddev_pitch_belt"
##  [33] "var_pitch_belt"           "avg_yaw_belt"
##  [35] "stddev_yaw_belt"          "var_yaw_belt"
##  [37] "gyros_belt_x"             "gyros_belt_y"
##  [39] "gyros_belt_z"             "accel_belt_x"
##  [41] "accel_belt_y"             "accel_belt_z"
##  [43] "magnet_belt_x"            "magnet_belt_y"
##  [45] "magnet_belt_z"            "roll_arm"
##  [47] "pitch_arm"                "yaw_arm"
##  [49] "total_accel_arm"          "var_accel_arm"
##  [51] "avg_roll_arm"             "stddev_roll_arm"
##  [53] "var_roll_arm"             "avg_pitch_arm"
##  [55] "stddev_pitch_arm"         "var_pitch_arm"
##  [57] "avg_yaw_arm"              "stddev_yaw_arm"
```

```
##  [59] "var_yaw_arm"              "gyros_arm_x"
##  [61] "gyros_arm_y"              "gyros_arm_z"
##  [63] "accel_arm_x"              "accel_arm_y"
##  [65] "accel_arm_z"              "magnet_arm_x"
##  [67] "magnet_arm_y"             "magnet_arm_z"
##  [69] "kurtosis_roll_arm"        "kurtosis_picth_arm"
##  [71] "kurtosis_yaw_arm"         "skewness_roll_arm"
##  [73] "skewness_pitch_arm"       "skewness_yaw_arm"
##  [75] "max_roll_arm"             "max_picth_arm"
##  [77] "max_yaw_arm"              "min_roll_arm"
##  [79] "min_pitch_arm"            "min_yaw_arm"
##  [81] "amplitude_roll_arm"       "amplitude_pitch_arm"
##  [83] "amplitude_yaw_arm"        "roll_dumbbell"
##  [85] "pitch_dumbbell"           "yaw_dumbbell"
##  [87] "kurtosis_roll_dumbbell"   "kurtosis_picth_dumbbell"
##  [89] "kurtosis_yaw_dumbbell"    "skewness_roll_dumbbell"
##  [91] "skewness_pitch_dumbbell"  "skewness_yaw_dumbbell"
##  [93] "max_roll_dumbbell"        "max_picth_dumbbell"
##  [95] "max_yaw_dumbbell"         "min_roll_dumbbell"
##  [97] "min_pitch_dumbbell"       "min_yaw_dumbbell"
##  [99] "amplitude_roll_dumbbell"  "amplitude_pitch_dumbbell"
## [101] "amplitude_yaw_dumbbell"   "total_accel_dumbbell"
## [103] "var_accel_dumbbell"       "avg_roll_dumbbell"
## [105] "stddev_roll_dumbbell"     "var_roll_dumbbell"
## [107] "avg_pitch_dumbbell"       "stddev_pitch_dumbbell"
## [109] "var_pitch_dumbbell"       "avg_yaw_dumbbell"
## [111] "stddev_yaw_dumbbell"      "var_yaw_dumbbell"
## [113] "gyros_dumbbell_x"         "gyros_dumbbell_y"
## [115] "gyros_dumbbell_z"         "accel_dumbbell_x"
## [117] "accel_dumbbell_y"         "accel_dumbbell_z"
## [119] "magnet_dumbbell_x"        "magnet_dumbbell_y"
## [121] "magnet_dumbbell_z"        "roll_forearm"
## [123] "pitch_forearm"            "yaw_forearm"
## [125] "kurtosis_roll_forearm"    "kurtosis_picth_forearm"
## [127] "kurtosis_yaw_forearm"     "skewness_roll_forearm"
## [129] "skewness_pitch_forearm"   "skewness_yaw_forearm"
## [131] "max_roll_forearm"         "max_picth_forearm"
## [133] "max_yaw_forearm"          "min_roll_forearm"
## [135] "min_pitch_forearm"        "min_yaw_forearm"
## [137] "amplitude_roll_forearm"   "amplitude_pitch_forearm"
## [139] "amplitude_yaw_forearm"    "total_accel_forearm"
## [141] "var_accel_forearm"        "avg_roll_forearm"
## [143] "stddev_roll_forearm"      "var_roll_forearm"
## [145] "avg_pitch_forearm"        "stddev_pitch_forearm"
## [147] "var_pitch_forearm"        "avg_yaw_forearm"
## [149] "stddev_yaw_forearm"       "var_yaw_forearm"
## [151] "gyros_forearm_x"          "gyros_forearm_y"
## [153] "gyros_forearm_z"          "accel_forearm_x"
## [155] "accel_forearm_y"          "accel_forearm_z"
## [157] "magnet_forearm_x"         "magnet_forearm_y"
## [159] "magnet_forearm_z"         "problem_id"
```

Data Processing

Cleaning the training and testing data set

In this section, removing those variables with nearly zero variance, variables that are almost always NA, and variables that don't make intuitive sense for prediction.

```
# Removing variables with nearly zero variance
nzv <- nearZeroVar(train_data)
train_data <- train_data[, -nzv]
test_data <- test_data[, -nzv]

# Removing variables that are almost always NA
na <- sapply(train_data, function(x) mean(is.na(x))) > 0.95
train_data <- train_data[, na==F]
test_data <- test_data[, na==F]

# removing variables (X, user_name, raw_timestamp_part_1, raw_timestamp_part_2, cvtd_timestamp) that is
train_data <- train_data[, -(1:5)]
test_data <- test_data[, -(1:5)]
```

Data Preparation

In this section, training data set will be splited into a smaller training data set and a validation data set.

```
inTrain<-createDataPartition(y=train_data$classe, p=0.7, list=F)
trainingSmall <- train_data[inTrain, ]
validation <- train_data[-inTrain, ]
```

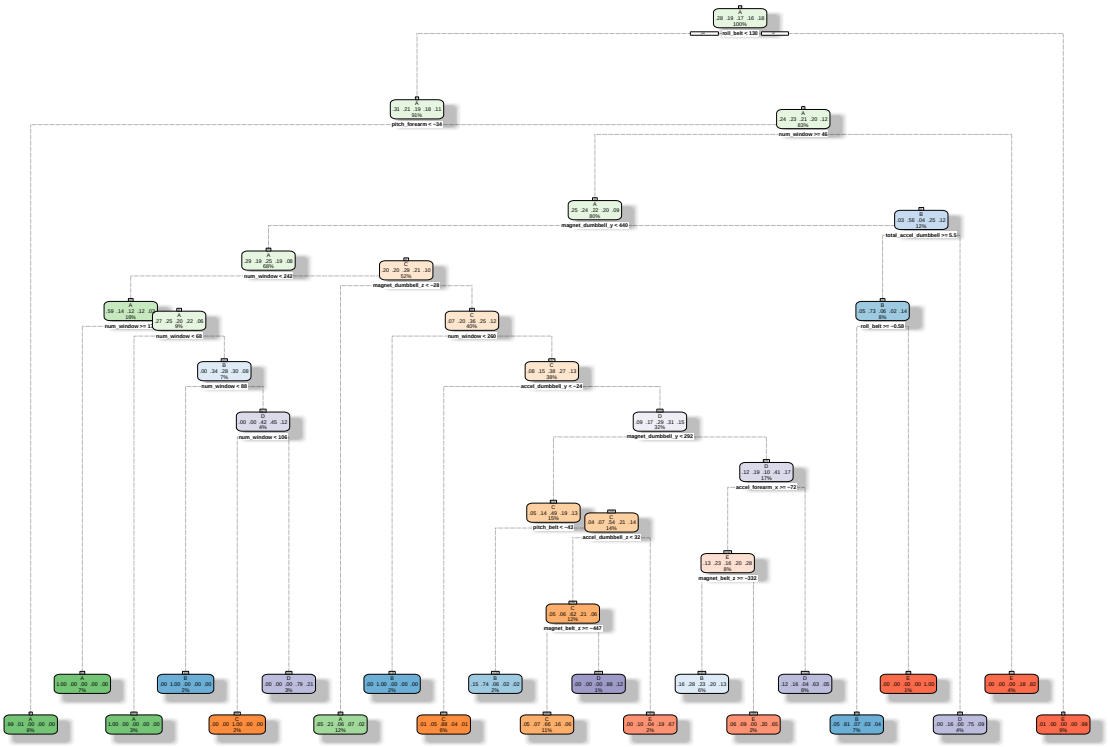
Machine Learning Models

Classification Tree

Building the model

```
set.seed(12345)
modFit_tree <- rpart(classe ~ ., data=trainingSmall, method="class")
fancyRpartPlot(modFit_tree)
```

```
## Warning: labs do not fit even at cex 0.15, there may be some overplotting
```



Rattle 2017−Sep−17 20:17:28 songhw

Cross Validation and Out of sample error

```
modFit_tree_validation <- predict(modFit_tree, validation, type = "class")
Validation_tree <- confusionMatrix(modFit_tree_validation, validation$classe)
Validation_tree
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction    A    B    C    D    E
##          A 1481  153   42   45   13
##          B   88  790  127   73   62
##          C   31   60  830  123   52
##          D   62  110   19  622   93
##          E   12   26    8  101  862
##
## Overall Statistics
##
##                Accuracy : 0.7791
##                  95% CI : (0.7683, 0.7896)
##     No Information Rate : 0.2845
##     P-Value [Acc > NIR] : < 2.2e-16
##
##                   Kappa : 0.7202
##  Mcnemar's Test P-Value : < 2.2e-16
##
```

```
## Statistics by Class:
##
##                      Class: A Class: B Class: C Class: D Class: E
## Sensitivity            0.8847   0.6936   0.8090   0.6452   0.7967
## Specificity            0.9399   0.9263   0.9453   0.9423   0.9694
## Pos Pred Value         0.8541   0.6930   0.7573   0.6865   0.8543
## Neg Pred Value         0.9535   0.9264   0.9591   0.9313   0.9549
## Prevalence             0.2845   0.1935   0.1743   0.1638   0.1839
## Detection Rate         0.2517   0.1342   0.1410   0.1057   0.1465
## Detection Prevalence   0.2946   0.1937   0.1862   0.1540   0.1715
## Balanced Accuracy      0.9123   0.8099   0.8771   0.7938   0.8830
```

Building the model

Random Forest

```
set.seed(123)
modFit_rf <- randomForest(classe ~ ., data=trainingSmall)
modFit_rf
```

```
##
## Call:
##  randomForest(formula = classe ~ ., data = trainingSmall)
##                Type of random forest: classification
##                      Number of trees: 500
## No. of variables tried at each split: 7
##
##         OOB estimate of  error rate: 0.31%
## Confusion matrix:
##      A    B    C    D    E  class.error
## A 3904    1    0    0    1 0.0005120328
## B    5 2650    3    0    0 0.0030097818
## C    0   10 2384    2    0 0.0050083472
## D    0    0   14 2237    1 0.0066607460
## E    0    0    0    6 2519 0.0023762376
```

Cross Validation and Out of sample error

```
modFit_rf_validation <- predict(modFit_rf, validation, type = "class")
Validation_rf <- confusionMatrix(modFit_rf_validation, validation$classe)
Validation_rf
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction    A    B    C    D    E
##          A 1674    1    0    0    0
##          B    0 1138    3    0    0
##          C    0    0 1022    7    0
##          D    0    0    1  957    0
##          E    0    0    0    0 1082
##
```

```
## Overall Statistics
##
##                Accuracy : 0.998
##                  95% CI : (0.9964, 0.9989)
##     No Information Rate : 0.2845
##     P-Value [Acc > NIR] : < 2.2e-16
##
##                   Kappa : 0.9974
##  Mcnemar's Test P-Value : NA
##
## Statistics by Class:
##
##                      Class: A Class: B Class: C Class: D Class: E
## Sensitivity            1.0000   0.9991   0.9961   0.9927   1.0000
## Specificity            0.9998   0.9994   0.9986   0.9998   1.0000
## Pos Pred Value         0.9994   0.9974   0.9932   0.9990   1.0000
## Neg Pred Value         1.0000   0.9998   0.9992   0.9986   1.0000
## Prevalence             0.2845   0.1935   0.1743   0.1638   0.1839
## Detection Rate         0.2845   0.1934   0.1737   0.1626   0.1839
## Detection Prevalence   0.2846   0.1939   0.1749   0.1628   0.1839
## Balanced Accuracy      0.9999   0.9992   0.9973   0.9963   1.0000
```

Prediction

Model random forest has better accuracy than decision tree, therefore, we are using random forest to predict the result of the testing data set.

```
prediction_rf_test_data <- predict(modFit_rf, test_data, type = "class")
prediction_rf_test_data
```

```
##  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
```