Exercise Quantification

jgblouin

Wednesday, February 11, 2015

Introduction

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 Processing

First, load the required libraries and acquire the data.

```
library(caret)
```

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

```
library(randomForest)
```

```
## randomForest 4.6-10
## Type rfNews() to see new features/changes/bug fixes.
```

```
## if (!file.exists("Machine_Learning/pml-training.csv")) {
##   download.file("http://d396qusza40orc.cloudfront.net/predmachlearn/pml## -training.csv", destfile =
## }
## if (!file.exists("Machine_Learning/pml-testing.csv")) {
##   download.file("http://d396qusza40orc.cloudfront.net/predmachlearn/pml## -testing.csv", destfile =
## }
tempData <-read.csv("pml-training.csv")
testing <-read.csv("pml-testing.csv")
```

We now remove columns with more than 100 NA values and variance is less than 10.

```
##Remove columns with more than 100 NAs.
data <- tempData[, colSums(is.na(tempData)) < 100]

##Keep variables with variance > 10.
colvar <- apply(data, 2, var)
selectBigVar <- which(colvar > 10)
selectBigVar <- as.vector(selectBigVar)
```

We can now restrict our study to relevant features. Examining the documentation concerning the data reveals that the statistical variables are not needed for our purpose.

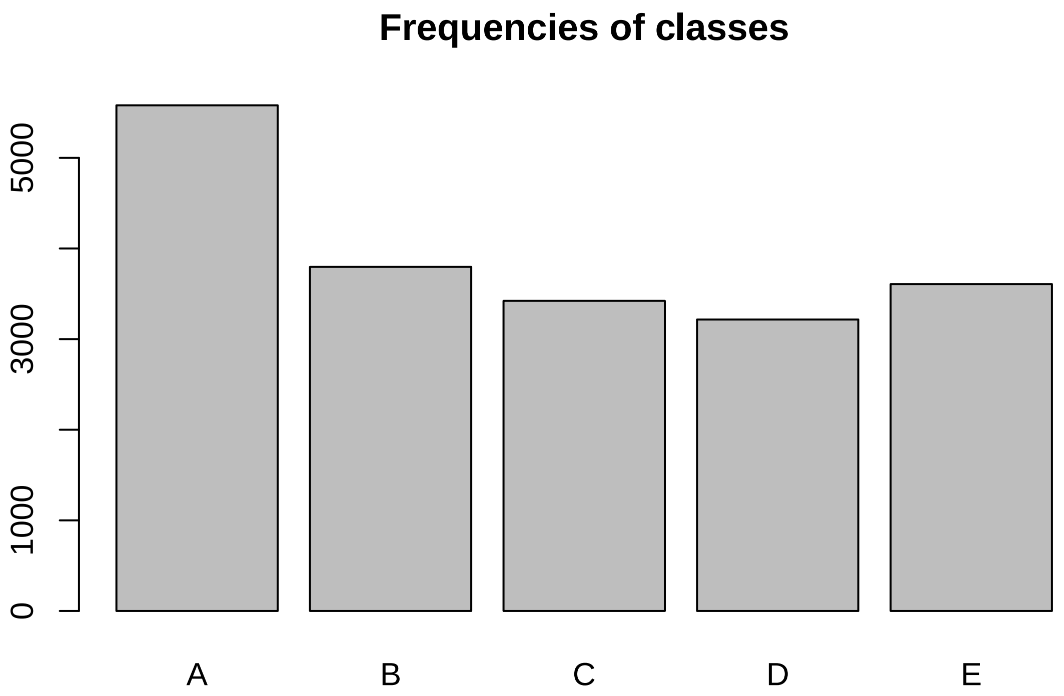
```
selectManual <- c(2, 8, 9, 24:26, 30:33, 37:39, 49, 50, 65:67, 71, 72, 83, 87:89, 93)
combine <- unique(as.vector(rbind(selectBigVar, selectManual)))
```

```
## Warning in rbind(selectBigVar, selectManual): number of columns of result
## is not a multiple of vector length (arg 2)
```

```
select <- combine[-1]
data <- data[select]
```

We can now plot the frequencies of the classes.

```
barplot(table(data$classe), main = "Frequencies of classes")
```



Splitting the data set

The data set is still large. As seen below, it has 19622 observations and 45 variables.

```
dim(data)
```

```
## [1] 19622    45
```

This data set can be partitioned into a training set and into a cross validation set with a 60:40 split.

```
inTrain <- createDataPartition(data$classe, p = 0.6, list = FALSE)
training <- data[inTrain,]
crossValidation <- data[-inTrain,]
```

Model selection

We are now in possession of a training, testing and cross validation set, so we can now fit a model using random forest.

```
fit <- randomForest(classe ~., data = training)
fit
```

```
## 
## Call:
##  randomForest(formula = classe ~ ., data = training) 
##                Type of random forest: classification
##                      Number of trees: 500
## No. of variables tried at each split: 6
## 
##         OOB estimate of  error rate: 0.2%
## Confusion matrix:
##      A    B    C    D    E  class.error
## A 3347    1    0    0    0 0.0002986858
## B    3 2275    1    0    0 0.0017551558
## C    0    3 2048    3    0 0.0029211295
## D    0    0    6 1922    2 0.0041450777
## E    0    0    0    4 2161 0.0018475751
```

This model can now be used to predict on the cross validation set.

```
predcv <- predict(fit, crossValidation)
confusionMatrix(predcv, crossValidation$classe)
```

```
## Confusion Matrix and Statistics
## 
##           Reference
## Prediction    A    B    C    D    E
##          A 2232    0    0    0    0
##          B    0 1518    1    0    0
##          C    0    0 1367    3    0
##          D    0    0    0 1282    0
##          E    0    0    0    1 1442
## 
## Overall Statistics
##                                           
##                Accuracy : 0.9994          
##                  95% CI : (0.9985, 0.9998)
##     No Information Rate : 0.2845          
##     P-Value [Acc > NIR] : < 2.2e-16       
##                                           
##                   Kappa : 0.9992          
##  Mcnemar's Test P-Value : NA
##
## Statistics by Class:
##
##                      Class: A Class: B Class: C Class: D Class: E
## Sensitivity            1.0000   1.0000   0.9993   0.9969   1.0000
## Specificity            1.0000   0.9998   0.9995   1.0000   0.9998
## Pos Pred Value         1.0000   0.9993   0.9978   1.0000   0.9993
## Neg Pred Value         1.0000   1.0000   0.9998   0.9994   1.0000
## Prevalence             0.2845   0.1935   0.1744   0.1639   0.1838
## Detection Rate         0.2845   0.1935   0.1742   0.1634   0.1838
## Detection Prevalence   0.2845   0.1936   0.1746   0.1634   0.1839
## Balanced Accuracy      1.0000   0.9999   0.9994   0.9984   0.9999
```

The model proves very acurate, with an accuracy of 99.8% on the training set.

Prediction results for the training set

Here are the predictions on the testing set.

```
predictedResults <- as.character(predict(fit, testing))
predictedResults
```

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

```
##Tests <- cbind(testing, predictedResults)
##subset(Tests, select=names(Tests)[grep("belt|[^(fore)]arm|dumbbell|forearm", names(Tests), invert=TRU
```

Submit to Coursera

```
pml_write_files = function(x){
  n = length(x)
  for(i in 1:n){
    filename = paste0("problem_id_",i,".txt")
    write.table(x[i],file=filename,quote=FALSE,row.names=FALSE,col.names=FALSE)
  }
}

pml_write_files(predictedResults)
```

References

- Velloso, E.; Bulling, A.; Gellersen, H.; Ugulino, W.; Fuks, H. Qualitative Activity Recognition of Weight Lifting Exercises. Proceedings of 4th International Conference in Cooperation with SIGCHI (Augmented Human '13) . Stuttgart, Germany: ACM SIGCHI, 2013
- http://groupware.les.inf.puc-rio.br/har