Practical Machine learning project

samaa essa

12/10/2020

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 data for this project come from this source: http://groupware.les.inf.puc-rio.br/har.

The training data for this project are available here

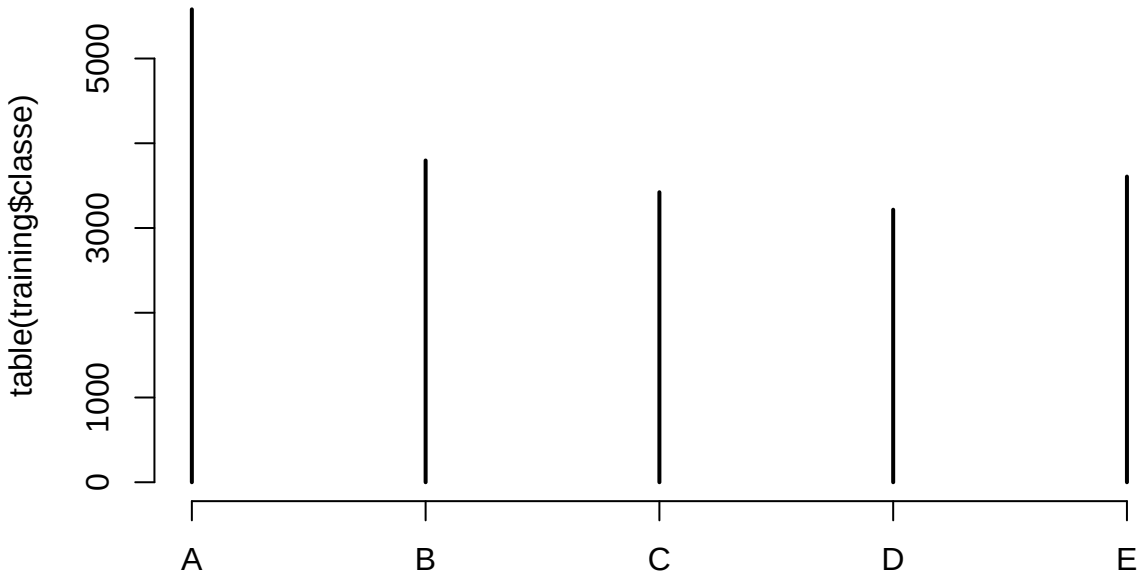
The test data are available here

Load Data

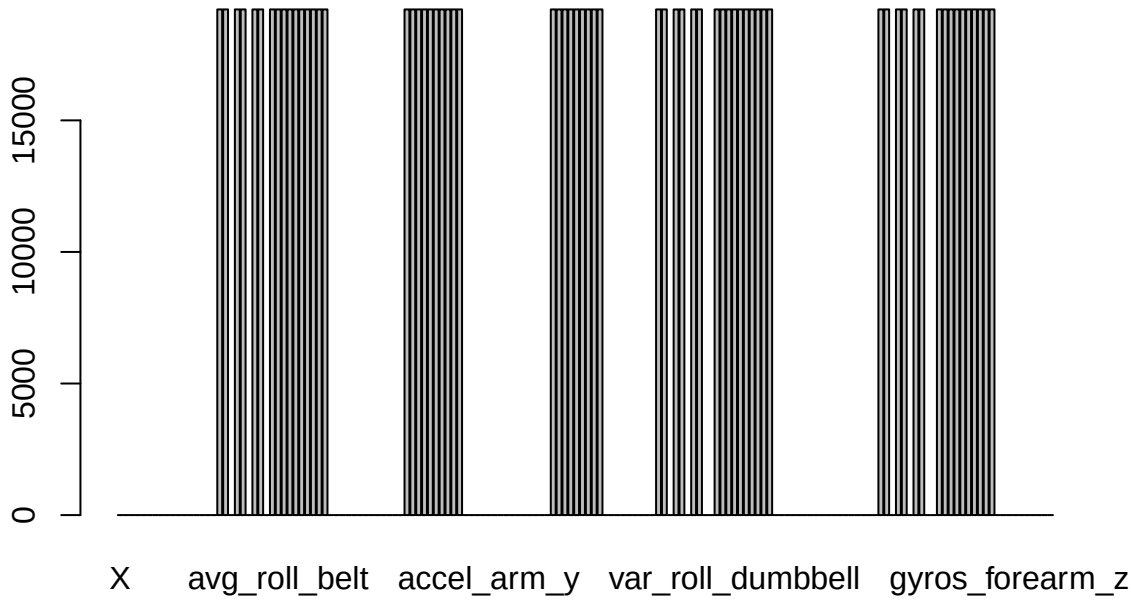
```
training <- read.csv("/home/samaaessa/Desktop/DS coursera/course 8/pml-training.csv")
testCases <- read.csv("/home/samaaessa/Desktop/DS coursera/course 8/pml-testing.csv")
```

Data Exploration and Cleaning

```
plot(table(training$classe))
```




```
barplot(colSums(is.na(training)))
```



So we'll remove those NA columns

```
NAcolumns <- names(which(colSums(is.na(training))>0))
training <- training[,!names(training) %in% NAcolumns]
```

Data Splitting

we'll split the cleaned training set into a pure training data set (70%) and a validation data set (30%). We will use the validation data set to estimate model accuracy.

```
library(caret)
```

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

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

```
set.seed(22519) # For reproducibile purpose
inTrain <- createDataPartition(training$classe, p=0.70, list=F)
trainData <- training[inTrain, ]
testData <- training[-inTrain, ]
```

Feature Selection

```
nzvcol <- nearZeroVar(trainData)
trainData <- trainData[, -nzvcol]
```

Train Model

We fit a predictive model for activity recognition using **Random Forest** algorithm because it automatically selects important variables and is robust to correlated covariates & outliers in general. We will use 10-fold cross validation when applying the algorithm.

```
model <- train(classe ~ ., data=trainData, method="rf", trControl=trainControl(method = "cv",10),ntree=
```

Then, we estimate the performance of the model on the test set.

```
predictions <- predict(model,testData)
confusionMatrix(testData$classe,predictions)
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction    A    B    C    D    E
##          A 1674    0    0    0    0
##          B    0 1139    0    0    0
##          C    0    0 1026    0    0
##          D    0    0    1  963    0
##          E    0    0    0    0 1082
##
## Overall Statistics
##
##                Accuracy : 0.9998
##                  95% CI : (0.9991, 1)
##     No Information Rate : 0.2845
##     P-Value [Acc > NIR] : < 2.2e-16
##
##                   Kappa : 0.9998
##
##  Mcnemar's Test P-Value : NA
##
## Statistics by Class:
##
##                      Class: A Class: B Class: C Class: D Class: E
## Sensitivity            1.0000   1.0000   0.9990   1.0000   1.0000
## Specificity            1.0000   1.0000   1.0000   0.9998   1.0000
## Pos Pred Value         1.0000   1.0000   1.0000   0.9990   1.0000
## Neg Pred Value         1.0000   1.0000   0.9998   1.0000   1.0000
## Prevalence             0.2845   0.1935   0.1745   0.1636   0.1839
## Detection Rate         0.2845   0.1935   0.1743   0.1636   0.1839
## Detection Prevalence   0.2845   0.1935   0.1743   0.1638   0.1839
## Balanced Accuracy      1.0000   1.0000   0.9995   0.9999   1.0000
```