FitBitML

Sotero Alvarado

Sunday, May 17, 2015

In this part we download the training and test sets.

```
setwd("~/ML/assignment")


library(RCurl)
```

```
## Loading required package: bitops
```

```
datad <- getURL("https://d396qusza40orc.cloudfront.net/predmachlearn/pml-training.csv",
                ssl.verifypeer=0L, followlocation=1L)

datatr<- read.csv(text=datad)



datat <- getURL("https://d396qusza40orc.cloudfront.net/predmachlearn/pml-testing.csv",ssl.verifypeer=0L

data2 <- read.csv(text=datat)
```

Below we download all required packages and remove all columns that are all NAs. We also remove the X column since it was causing the model to return erroneous predictions.

```
# Removing all NA columns
library("caret")
```

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

```
library("e1071")


data2 <-  Filter(function(x)!all(is.na(x)), data2)

data2 <- data2[ ,-1  ]


coln<-" "

coln<- colnames(data2)

coln<-c(coln,"classe")



datatr <-  datatr[ ,colnames(datatr)%in%coln]
```

Here we use cross validation.

```
inTrain <- createDataPartition( y = datatr$classe, p = 0.6, list= F,   )

training <- datatr[inTrain, ]
testing <- datatr[-inTrain, ]

modFit <- train( classe ~ . , method = "gbm", data = training, verbose = F )
```

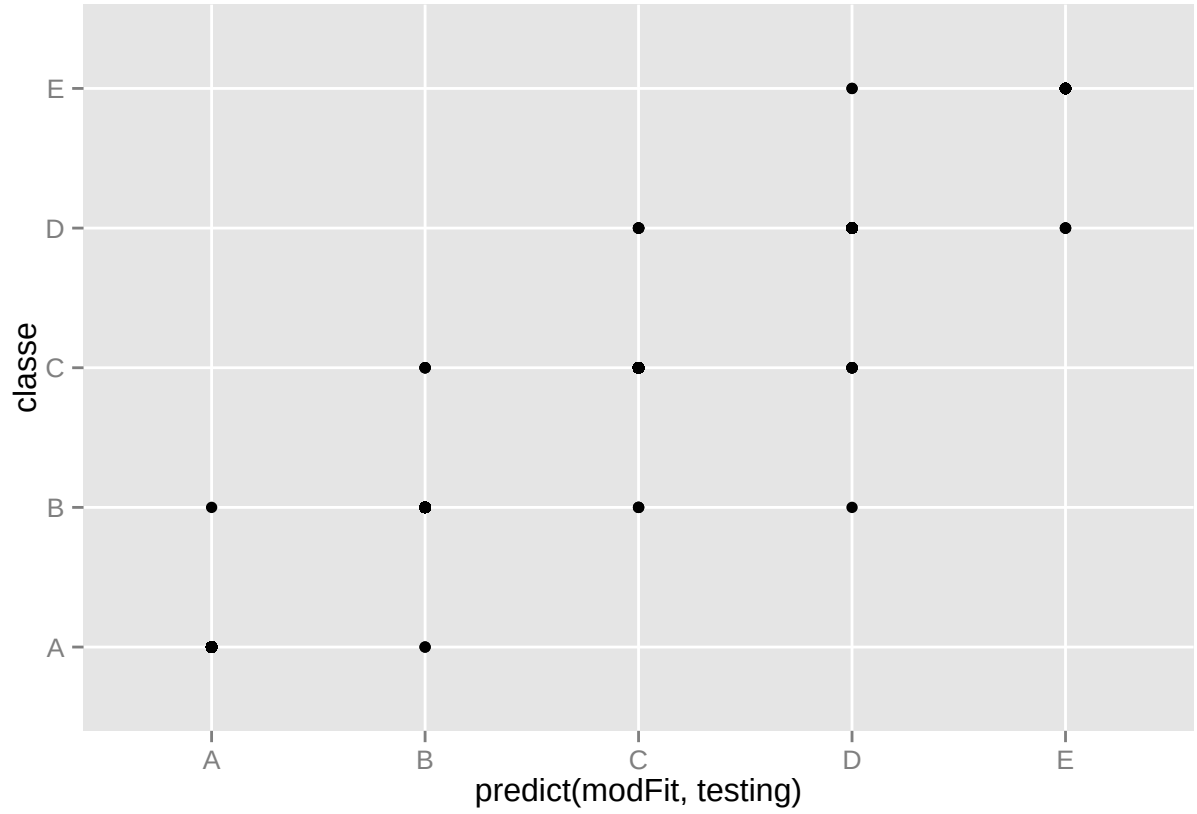
```
## Loading required package: gbm
## Loading required package: survival
##
## Attaching package: 'survival'
##
## The following object is masked from 'package:caret':
##
##     cluster
##
## Loading required package: splines
## Loading required package: parallel
## Loaded gbm 2.1.1
## Loading required package: plyr
```

```
pred<-predict(modFit, newdata = testing )




confusionMatrix(testing$classe, predict(modFit, newdata = testing ) )
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction    A    B    C    D    E
##          A 2230    2    0    0    0
##          B    1 1513    3    1    0
##          C    0    4 1357    7    0
##          D    0    0    7 1275    4
##          E    0    0    0    2 1440
##
## Overall Statistics
##
##                Accuracy : 0.996
##                  95% CI : (0.9944, 0.9973)
##     No Information Rate : 0.2843
##     P-Value [Acc > NIR] : < 2.2e-16
##
##                   Kappa : 0.995
##  Mcnemar's Test P-Value : NA
##
## Statistics by Class:
```

```
##
##                      Class: A Class: B Class: C Class: D Class: E
## Sensitivity            0.9996   0.9961   0.9927   0.9922   0.9972
## Specificity            0.9996   0.9992   0.9983   0.9983   0.9997
## Pos Pred Value         0.9991   0.9967   0.9920   0.9914   0.9986
## Neg Pred Value         0.9998   0.9991   0.9985   0.9985   0.9994
## Prevalence             0.2843   0.1936   0.1742   0.1638   0.1840
## Detection Rate         0.2842   0.1928   0.1730   0.1625   0.1835
## Detection Prevalence   0.2845   0.1935   0.1744   0.1639   0.1838
## Balanced Accuracy      0.9996   0.9976   0.9955   0.9953   0.9985
```

```
qplot(predict(modFit,testing),classe, data=testing)
```



Since our model proved quite accurate we are now able to make our predictions. I was expecting the error to be about 0.1 however it turned out that it was a lot my accuracy. I am guessing because I used the most accurate methond according the book and Intruduction to Statistical Learning.

We have our predictions below.

```
pred2<-predict(modFit, newdata = data2 )


pred2
```

```
##  [1] 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
```