Machine Learning Coursera Project

bcoke

October 25, 2015

```
require(caret)
```

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

```
require(randomForest)
```

```
## Loading required package: randomForest
## randomForest 4.6-12
## Type rfNews() to see new features/changes/bug fixes.
```

Introduction

Human Activity Recognition (HAR) has the potential for many powerful applications monitoring physicial activity in health and disease. Machine learning techniques allow for classification of such human activity.The purpose of this analysis is to use machine learning techniques in R to predict five types of dumbell lifts (correct and incorrect techniques). Such a technique has the potential to allow for closed loop classification of dumbell curls, thus alerting the user if their technique is correct or incorrect.

Methods

First we need to read in the data. Click for the training data and test data.

The data is also available in this github repo.

Once the data is downloaded, it can read into R with the following command.

```
trainSet<- read.csv("pml-training.csv")
testSet <- read.csv("pml-testing.csv")
```

Cleaning the Data

First we will set the seed so the analysis is reproducible. Then we will split the training data into a training and testing set.

```
set.seed(8484)
trainIndex = sample(1:dim(trainSet)[1],size=dim(trainSet)[1]/8,replace=F)
train = trainSet[trainIndex,]
test = trainSet[-trainIndex,]
```

Model Building

To find the columns in the data set that have near zero variance (and thus are of little value for our training), I first use the nearZeroVar function. I then set a variable called classIndex to remove the unwanted columns.

```
nsv <- nearZeroVar(train, saveMetrics = TRUE)
x <- nsv[4]
classIndex <- which(x$nzv == FALSE)
# remove columns with near zero variance
trainClassBig <- trainSet[, classIndex]
# remove first 6 columns that don't add to the model
trainClassBig <- trainClassBig[, -(1:6)]
```

Next I look in the testSet to see what rows contain no data to remove columns that will not contribute to the model.

```
testSetIndexed <- testSet[, classIndex]
testSetIndexed <- testSetIndexed[, -(1:6)]

naindex <- which(is.na(testSetIndexed[1, ]))

# Creating test and training datasets without the unneeded columns
testSetIndexed <- testSetIndexed[, -naindex]
trainClassBig <- trainClassBig[, -naindex]
```

Next I create the model using the randomForest package.

```
bestFit <- randomForest(classe ~ ., data = trainClassBig)
```

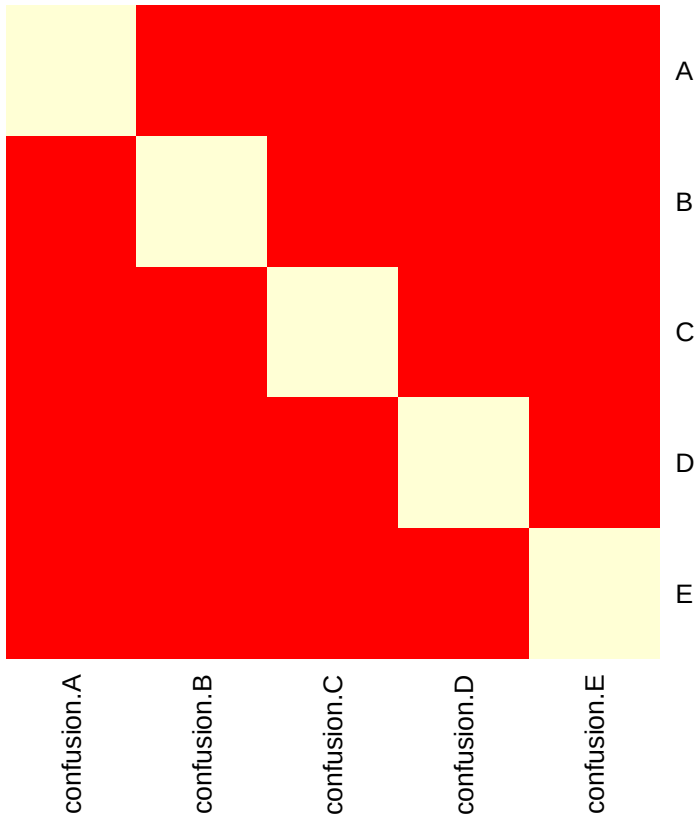
And we can take a look at the how well the model works.

```
print(bestFit)
```

```
##
## Call:
##  randomForest(formula = classe ~ ., data = trainClassBig)
##                Type of random forest: classification
##                      Number of trees: 500
## No. of variables tried at each split: 7
##
##         OOB estimate of  error rate: 0.26%
## Confusion matrix:
##      A    B    C    D    E  class.error
## A 5578    1    0    0    1 0.0003584229
## B    9 3785    3    0    0 0.0031603898
## C    0    8 3412    2    0 0.0029222677
## D    0    0   21 3194    1 0.0068407960
## E    0    0    1    4 3602 0.0013861935
```

The model has an out-of-bag (OOB) estimate of error rate of 0.26%. We can also visualize the confusion matrix, which confirms what we see in the quantitative measurements above.

```
x <- as.matrix(as.data.frame(bestFit[5]))
x <- x[, -6]
heatmap(x, Rowv = NA, Colv = NA, revC = TRUE,
        main = "Heatmap of Confusion Matrix",
        cexCol = 1, cexRow = 1)
```



Lastly, we can use our model to predict the test set.

```
predict(bestFit, testSetIndexed)
```

```
##  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
```

Discussion

Using relatively simple commands, I was able to develop an impressively accurate, random forest machine learning algorithtim to classify HAR data. Such an approach is generally useful for many large datasets and applications beyond HAR.

References

Ugulino, W.; Cardador, D.; Vega, K.; Velloso, E.; Milidiu, R.; Fuks, H. Wearable Computing: Accelerometers' Data Classification of Body Postures and Movements. Proceedings of 21st Brazilian Symposium on Artificial Intelligence. Advances in Artificial Intelligence - SBIA 2012. In: Lecture Notes in Computer Science. , pp. 52-61. Curitiba, PR: Springer Berlin / Heidelberg, 2012. ISBN 978-3-642-34458-9. DOI: 10.1007/978-3-642-34459-6_6.

Read more about the dataset here.