Practical Machine Learning Project

Frank Goeddeke

Friday, July 25, 2014

Executive Summary

This project is in partial fulfillment of a Coursera Practical Machine Learning MOOC course taught by Jeff Leek et al., of Johns Hopkins University. Using data from 6 participants wearing electronic accelerometer devices on the belt, forearm, arm, and dumbell, the goal of this project is to build a machine learning algorithm to predict how well a sample of participants are doing their exercises. The participants are rated on how well they do exercises in 5 categories, labelled "A", "B", "C", "D", and "E". Data (19622 observations, 160 variables) was randomly divided into a training data set of 13737 observations (p = 0.70), and a cross validation testing dataset of 5885 observations. Non-numeric, missing data, and NA data were removed, leaving 52 numeric independent variables. For simplictiy's sake, no other pre-processing of the data was performed. A random forest tree was fitted to the training data and cross validated with an accuracy of 0.9963. Predictions were made on the 20 observations in the testing data set.

Load Packages

```
library(randomForest)
```

```
## Warning: package 'randomForest' was built under R version 3.1.1
```

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

```
library(caret)
```

```
## Warning: package 'caret' was built under R version 3.1.1
```

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

```
library(AppliedPredictiveModeling)
```

```
## Warning: package 'AppliedPredictiveModeling' was built under R version
## 3.1.1
```

```
set.seed(975)
```

Read in the Data

```
#Dont forget to set the working directory!
trainingDataSet <- read.csv(file="pml-training.csv", as.is = TRUE, stringsAsFactors = FALSE, sep=',', na
# Then remove the non-numeric columns
trainingDataSet <- trainingDataSet[,-seq(1:7)]
trainingDataSet$classe <- as.factor(trainingDataSet$classe)
testingDataSet <- read.csv(file="pml-testing.csv", as.is = TRUE, stringsAsFactors = FALSE, sep=',', na.
# Then remove the non-numeric columns
testingDataSet <- testingDataSet[,-seq(1:7)]
```

Clean the data of NA's

```
NAtraining <- apply(trainingDataSet,2,function(x) {sum(is.na(x))})
trainingDataSetClean <- trainingDataSet[,which(NAtraining == 0)]
NAtesting <- apply(testingDataSet,2,function(x) {sum(is.na(x))})
testingDataSetClean <- testingDataSet[,which(NAtesting == 0)]
```

NA's and missing data columns need to be removed, as the random forest algorithm won't work with them.

Create data into training and cross validation data sets

```
TrainPart = createDataPartition(trainingDataSetClean$classe, p = 0.7, list=FALSE)
trainingDataSetClean2 = data.frame(trainingDataSetClean[TrainPart,])
crossValidation = trainingDataSetClean[-TrainPart,]
```

Splitting a data set into a training data set of 70% and a cross validation data set of 30% is fairly common.

Fit the random forest model

```
modelFitRFnoPCA <- randomForest(trainingDataSetClean2$classe ~., data=trainingDataSetClean2, importance=
predictRFnoPCA <- predict(modelFitRFnoPCA, crossValidation)
confusionMatrix(predictRFnoPCA, crossValidation$classe)
```

```
## Warning: package 'e1071' was built under R version 3.1.1

## Confusion Matrix and Statistics
##
##           Reference
## Prediction    A    B    C    D    E
##          A 1673    4    0    0    0
##          B    0 1134    2    0    0
##          C    1    1 1024   13    0
##          D    0    0    0  951    1
##          E    0    0    0    0 1081
```

```
##
## Overall Statistics
##
##                Accuracy : 0.996
##                  95% CI : (0.994, 0.998)
##     No Information Rate : 0.284
##     P-Value [Acc > NIR] : <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             0.999    0.996    0.998    0.987    0.999
## Specificity             0.999    1.000    0.997    1.000    1.000
## Pos Pred Value          0.998    0.998    0.986    0.999    1.000
## Neg Pred Value          1.000    0.999    1.000    0.997    1.000
## Prevalence              0.284    0.194    0.174    0.164    0.184
## Detection Rate          0.284    0.193    0.174    0.162    0.184
## Detection Prevalence    0.285    0.193    0.177    0.162    0.184
## Balanced Accuracy       0.999    0.998    0.997    0.993    1.000
```

For the sake of simplicity (Occam's Razor!), a simple random forest model was fitted to the training data set (13,737 observations) with 52 numeric variables with no pre-processing or missing data.

The random forest model was validated with the cross validation data set of 5,885 observations, randomly split from the original data set. The accuracy of this model is 0.9963. Of the 5,885 predictions made in the cross validation sample, only 22 wrong predictions were made.

Out of Sample Error

Accuracy is one measure of out of sample error for non-continuous data. The accuracy of the random forest model on the cross validation data set is 0.996. Alternatively, the Kappa Index in the cross validation sample is 0.995. Both measures indicate a very low error rate.

Make predictions for testing data set

```
prediction <- predict(modelFitRFnoPCA, testingDataSetClean)
prediction
```

```
##  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
```

References

Note: The data can be downloaded from here: http://groupware.les.inf.puc-rio.br/har

The citation for the dataset is: 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.

Read more: http://groupware.les.inf.puc-rio.br/har#ixzz38U5DPb3J