Final Project, Practical Machine Learning

Patrick Applegate, applegatepj@gmail.com

1 July 2016

Introduction

The *Practical Machine Learning* course, taught by Jeff Leek at Johns Hopkins University through Coursera, teaches students how to identify patterns in data using the R programming language. These patterns relate the values of predictor variables to their corresponding responses. Once identified, these patterns can be used to estimate the values of the response variable in cases where those values aren't known.

As an example, we can estimate a person's adult height by doubling that person's height at age 2 (Verzani, 2014). In this case, the person's height at age 2 is the predictor variable, and the person's adult height is the response variable. Multiplying by 2 implies that the pattern describing the relationship between child and adult heights is linear.

In this project, we looked at a more complex example having to do with exercise (Velloso et al., 2013). Each of several participants was asked to lift a weight in five different ways, labeled A, B... E. One of these ways represents the correct method of lifting the weight, that is, the method that minimizes the chance of injury if the exercise is repeated many times. The other four methods correspond to common mistakes in weightlifting. The participants were fitted with accelerometers that recorded the movement of different body parts relative to one another during each repetition of the exercise. Here, the accelerometer data represent the predictor variables, whereas the method used to carry out the exercise is the response variable.

The goal of the project was to develop a piece of R code that could correctly identify the method used to lift the weight from the accelerometer data, in cases where the method wasn't reported.

Methods

To meet this goal, I used a training data set that included values for many predictor variables and the response variable, as well as a test data set that omitted the response variable and included fewer records (20 rows vs. 19622 in the training set). The training and test data sets can be downloaded from https://d396qusza40orc.cloudfront.net/predmachlearn/pml-training.csv and https://d396qusza40orc.cloudfront.net/predmachlearn/pml-testing.csv, respectively.

Next, I eliminated most of the predictor variables. Many of the columns in `pml-training.csv` contain little information about the response variable, either because they contain values that have little variance, or because they contain mostly `NAs` (missing values). After eliminating these non-informative columns, I was left with 53 predictor variables out of an initial selection of 152.

I performed the model fitting and cross-validation using a method described by Community TA Leonard Greski at https://github.com/lgreski/datasciencectacontent/blob/master/markdown/pml-randomForestPerformance.md. Essentially, this method uses a random forests approach to develop a classification rule, and estimates the likely out-of-sample accuracy of the fit using k-fold cross-validation. The calculations involved in identifying the classification rule would normally take a long time; I followed Leonard Greski's instructions to speed up the computation using the `parallel` package in R.

Results

The out-of-sample accuracy for the 10 folds performed had a mean of 98.22% with a standard deviation of 0.29%. These accuracies are approximately normally distributed (Fig. 1). This observation suggests that the

mean and standard deviation are good representations of the variability of the k-fold accuracies, and that the true out-of-sample accuracy is close to the mean reported above.

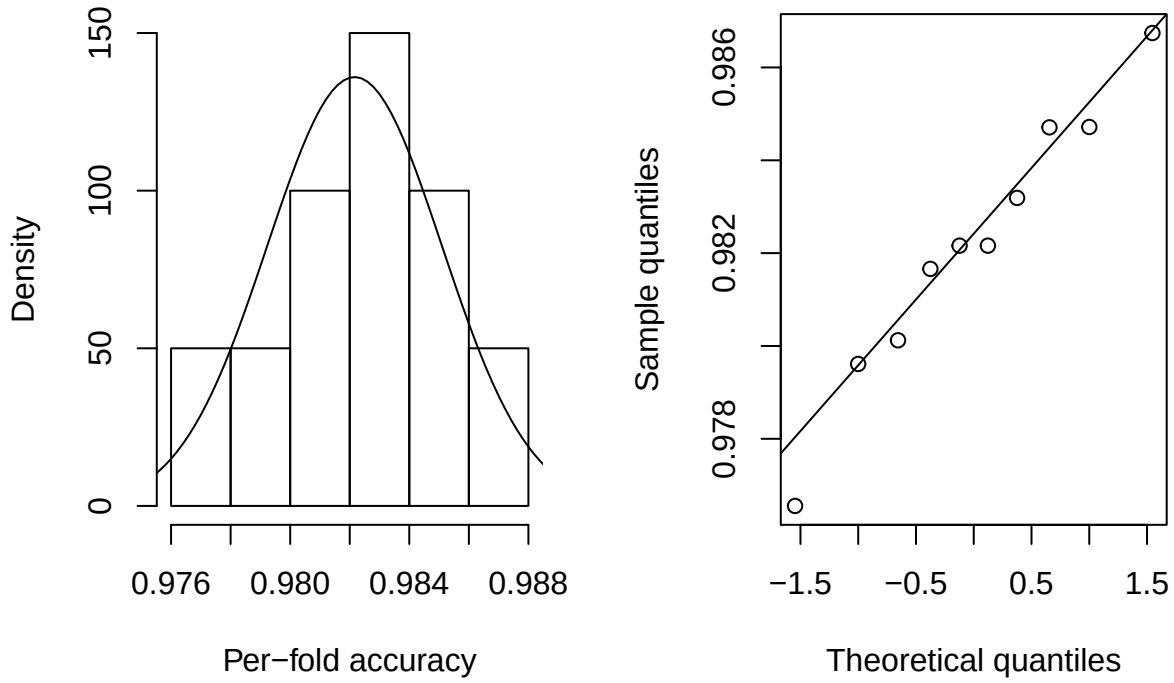

Figure 1: **Figure 1.** Distribution of out-of-sample accuracies for a k-fold cross-validation scheme with 10 folds and a random forest fitting algorithm. Left panel, histogram of accuracies with a superimposed best-fit normal curve. Right panel, quantile-quantile plot of these accuracies. The points in this plot fall close to a straight line, suggesting that the accuracies are approximately normally distributed.

The following confusion matrix shows the frequency with which the algorithm predicts different values of the outcome variable in the rows, vs. the correct answers in the columns. The values on the diagonal correspond to cases in which the calibrated random forests algorithm correctly estimated the response variable; the off-diagonal values represent cases in which the algorithm arrived at the wrong conclusion.

```
## Cross-Validated (10 fold) Confusion Matrix
##
## (entries are percentual average cell counts across resamples)
##
##           Reference
## Prediction    A    B    C    D    E
##          A 28.3  0.3  0.0  0.0  0.0
##          B  0.0 18.9  0.2  0.0  0.1
##          C  0.1  0.2 17.1  0.5  0.1
##          D  0.0  0.0  0.1 15.8  0.1
##          E  0.0  0.0  0.0  0.0 18.1
##
##  Accuracy (average) : 0.9822
```

Discussion

The random forests algorithm seems to be highly satisfactory for this problem. The largest off-diagonal value in the confusion matrix is 0.5, for cases in which the algorithm predicted C but the correct answer was D.

This misclassification represents one area in which the calibration of the algorithm could be improved in future work.

Bibliography

Velloso, E., Bulling, A., Gellersen, H., Ugulino, W., Fuks, H., 2013. Qualitative Activity Recognition of Weight Lifting Exercises. Proceedings of 4th Augmented Human (AH) International Conference in cooperation with ACM SIGCHI (Augmented Human '13) . Available online at http://groupware.les.inf.puc-rio.br/public/papers/2013.Velloso.QAR-WLE.pdf.

Verzani, J., 2014. Using R for Introductory Statistics (2nd ed). CRC Press.