Practical Machine Learning: Prediction

N. Lakhani

27 January 2018

Executive Summary

This project involves analysis of wearable fitness trackers. "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, my goal is 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. More information is available from the website here: http://groupware.les.inf.puc-rio.br/har (see the section on the Weight Lifting Exercise Dataset)

1. Loading libraries

The libraries needed: caret, rpart, randomForest, reshape2, AppliedPredictiveModeling are loaded in workspace.

2. Data cleaning and preparation

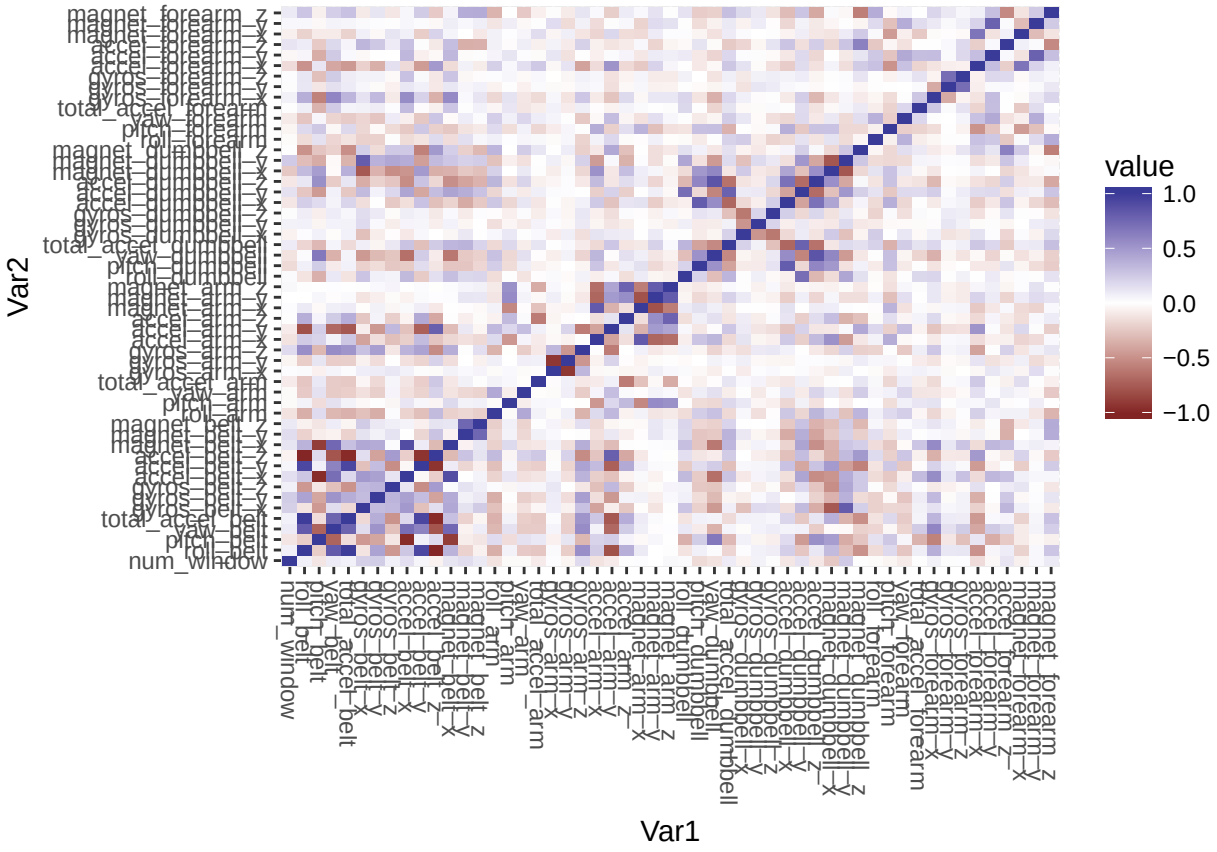
Input data from url's provided. The training data is split (70:30) into training & validation data (to be used as out of sample data) for cross validation. The following observations are made regarding the data:

- Since data from belt,forearm,arm, and dumbell are to be examined - filter out the rest
- There are dummy variables with no measurements for each observation, but these are summary stats for each time sliding window.
- The 'X' variable (row number) and 'new window' (marker for summary data), timestamp are not relevant in the current analysis, hence we drop columns (1:5)
- Several variables have near zero values (NZV) & 'NA's. Variables with NZV's & NA's over 70% are dropped. Interestingly even with 70%, we find that all variables with any NA's are dropped.

3. Cleaned data and features

The cleaned training data set has:

- 54 variables including **classe** with 9,926 observations.The testing dataset has 20 observations and 54 variables.
- The absence of NA's in any variables is also validated.
- The corr plot on the cleaned dataset indicates that very few of the variables have strong correlation (values close to -1 or 1, ignoring the squares along the diagonal in the plot, which shows cor for variables to themselves). *Also looking at the cor values and not too strong relationship, I feel there is no need for PCA and further variable elimination.



4. Building the model and parameters

I have taken the following approach:

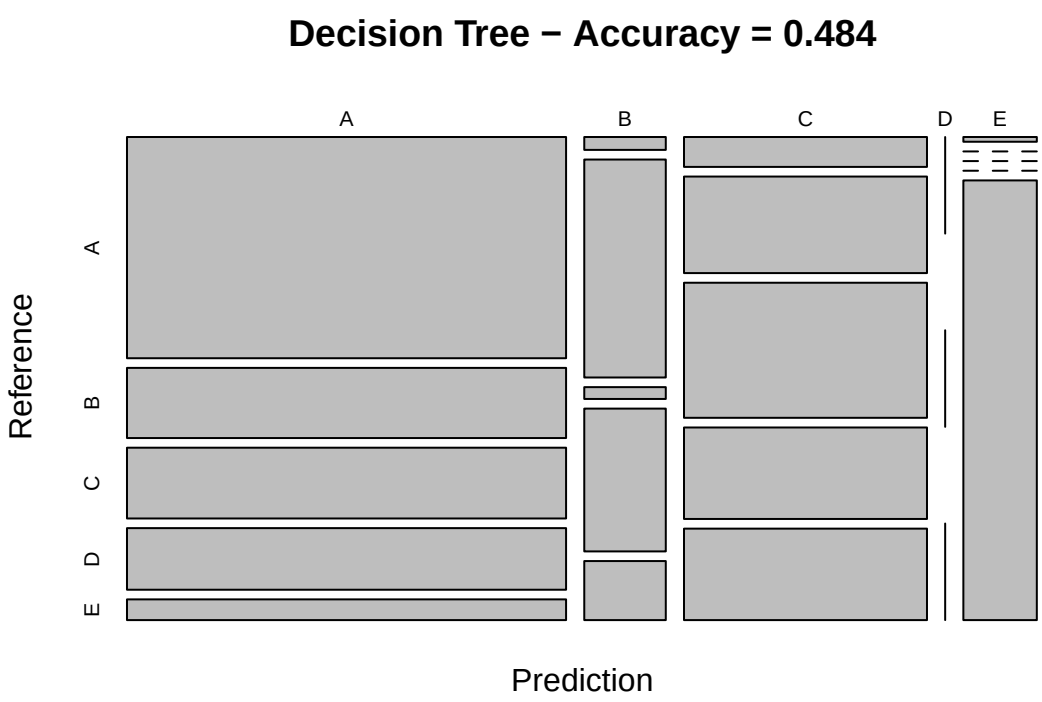
- The variable being predicted **classe** is a factor with 5 levels, so this is a classification problem.
- The 3 models evaluated for best accuracy are ++a) Decision Trees (rpart), ++b) Stochastic gradient boosting trees (gbm), ++c) Random forest decision trees (rf).
- I do a 3-fold cross validation using the function train to build the model

4a. Decision trees model results

The first model we explore is Decision Trees. As can be seen visually from the plot and confusion matrix (for classe values):

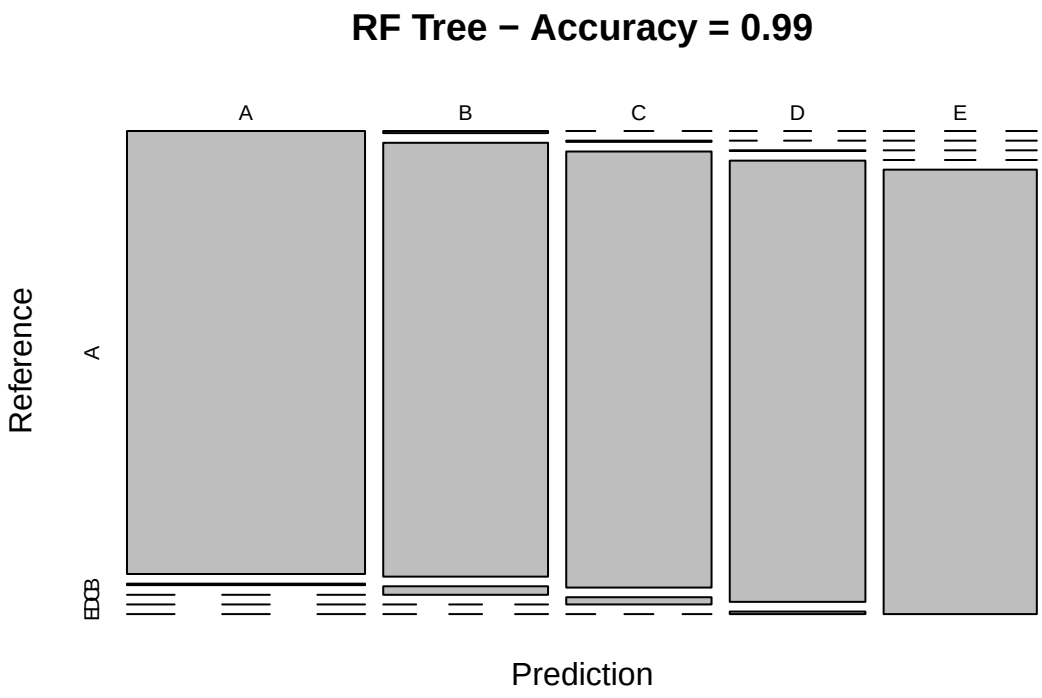
- the accuracy is quite low at 52% and
- the confusion matrix is highly populated across the matrix indicating many false postives/negatives; not a great model.

```
## Warning: In mosaicplot.default(x, xlab = xlab, ylab = ylab, ...) :
##  extra argument 'fill' will be disregarded
```



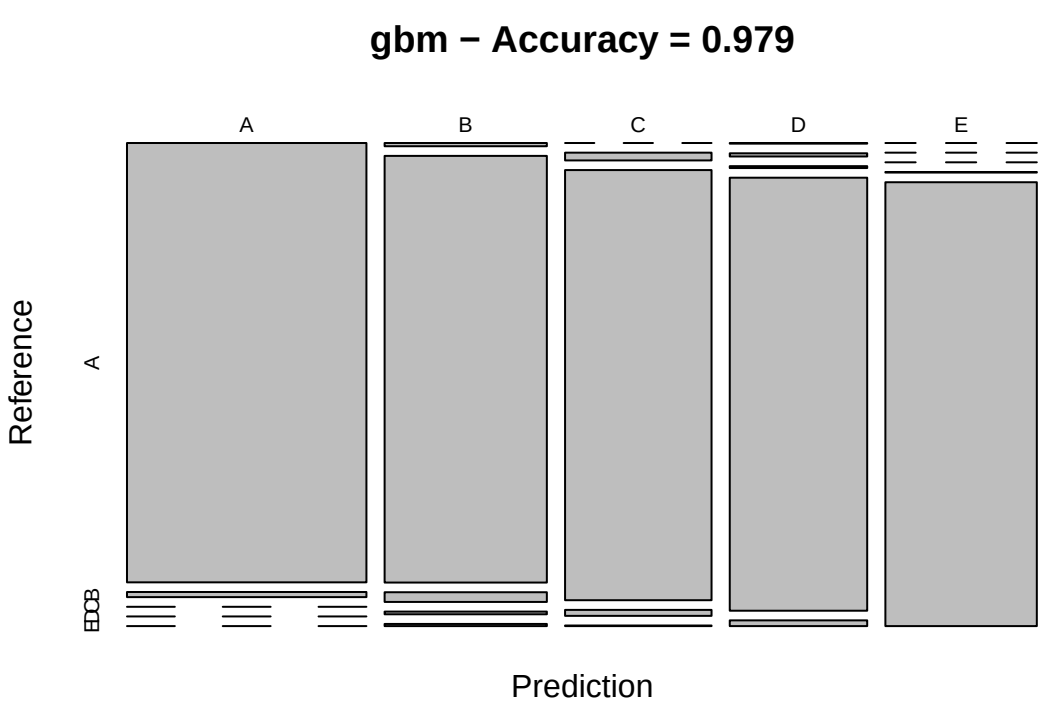
4b. Random Forest model results

The random forest model shows a significant improvement over the rpart model. As can be seen in output below, both from the plot and confusion matrix (for classe values): *The accuracy is above 99%.* The confusion matrix is quite clean with the diagonal of the matrix having majority of the matches. *The accuracy also peaks at about 27 predictors. The variables are listed in the order of importance.



4c. gbm model results

The gbm model shows: *Significant improvement over the rpart model and is slightly better than the rf model.* As can be seen in the output below, both from the plot and confusion matrix (for classe values), the accuracy is 98%. Of the 53 predictors, only 12 have influence



4. Final model and evaluation

Based on the results, I have picked the **rf** model as the final model with accuracy of 0.996. The top 5 features in order of inluence are shown below along with the accuracy predictions for the test dataset

```
##   Model  Accuracy
## 1  CART 0.4838861
## 2   GBM 0.9788285
## 3    RF 0.9898847
```

5. Prediction on test cases and output submission

```
##        FeatureName Importance
## 1       num_window  100.00000
## 2        roll_belt   65.86912
## 3       pitch_belt   41.85809
## 4         yaw_belt   34.19024
## 5 total_accel_belt   31.12868
```

```
##  [1] "A" "B" "A" "A" "E" "D" "B" "A" "A" "B" "C" "B" "A" "E" "E" "A" "B"
## [18] "B" "B"
```