Peer-Graded Assignment: Prediction Assignment Writeup

Bryan L. Mack

July 3, 2017

```
library(caret) #implicitly loads lattice and ggplot2
```

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

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

```
library(randomForest)
```

```
## randomForest 4.6-12
```

```
## Type rfNews() to see new features/changes/bug fixes.
```

```
##
## Attaching package: 'randomForest'
```

```
## The following object is masked from 'package:ggplot2':
##
##     margin
```

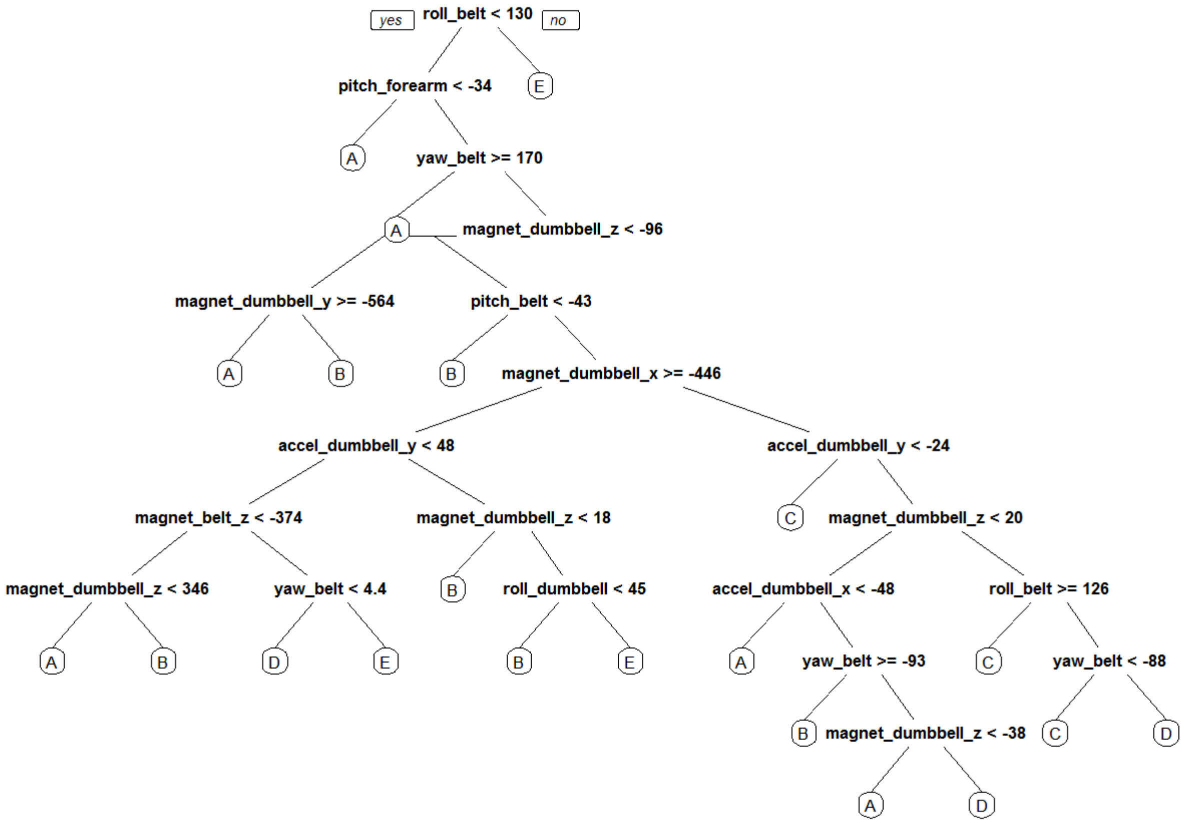
```
library(rpart) #to build decision tree
library(rpart.plot) #for the appendix, display decision tree
library(e1071) #need to build confusionMatrix
```

Cross Validation & Out of Sample Error

Cross validation will be done by splitting the training set into two groups. We can't use the test set when building the model or it becomes part of the training set (out of sample error), so we estimate the test set accuracy with the training set by building a predictor set. This is known as cross-validation. I will test a couple of models against this to determine the best model to use for prediction. Our goal is to capture all of the signal, but none of the noise.

I will use 70 percent for training, and 30 percent for validation

Train two different models, I will attempt a random forest and a decision tree. I will use whichever predicts most accurately.


yes
roll_belt < 130
no
pitch_forearm < -34
E
A
yaw_belt >= 170
A
magnet_dumbbell_z < -96
magnet_dumbbell_y >= -564
pitch_belt < -43
A
B
B
magnet_dumbbell_x >= -446
accel_dumbbell_y < 48
accel_dumbbell_y < -24
magnet_belt_z < -374
magnet_dumbbell_z < 18
C
magnet_dumbbell_z < 20
magnet_dumbbell_z < 346
yaw_belt < 4.4
B
roll_dumbbell < 45
accel_dumbbell_x < -48
roll_belt >= 126
A
B
D
E
B
E
A
yaw_belt >= -93
C
yaw_belt < -88
B
magnet_dumbbell_z < -38
C
D
A
D

```
## Confusion Matrix and Statistics
## 
##           Reference
## Prediction    A    B    C    D    E
##          A 1431  123   24   50   51
##          B   89  662   99  101   72
##          C   37  156  665   74  122
##          D  107  118  207  673   92
##          E   10   80   31   66  745
## 
## Overall Statistics
## 
##                Accuracy : 0.7096          
##                  95% CI : (0.6978, 0.7212)
##     No Information Rate : 0.2845          
##     P-Value [Acc > NIR] : < 2.2e-16       
## 
##                   Kappa : 0.6331          
##  Mcnemar's Test P-Value : < 2.2e-16       
## 
## Statistics by Class:
## 
##                      Class: A Class: B Class: C Class: D Class: E
## Sensitivity            0.8548   0.5812   0.6481   0.6981   0.6885
## Specificity            0.9411   0.9239   0.9199   0.8935   0.9611
## Pos Pred Value         0.8523   0.6471   0.6309   0.5622   0.7994
## Neg Pred Value         0.9422   0.9019   0.9253   0.9379   0.9320
## Prevalence             0.2845   0.1935   0.1743   0.1638   0.1839
## Detection Rate         0.2432   0.1125   0.1130   0.1144   0.1266
## Detection Prevalence   0.2853   0.1738   0.1791   0.2034   0.1584
## Balanced Accuracy      0.8980   0.7526   0.7840   0.7958   0.8248
## Confusion Matrix and Statistics
## 
##           Reference
## Prediction    A    B    C    D    E
##          A 1674    8    0    0    0
##          B    0 1128    5    0    0
##          C    0    3 1021   12    1
##          D    0    0    0  952    3
##          E    0    0    0    0 1078
## 
## Overall Statistics
##                                           
##                Accuracy : 0.9946          
##                  95% CI : (0.9923, 0.9963)
##     No Information Rate : 0.2845          
##     P-Value [Acc > NIR] : < 2.2e-16       
##                                           
##                   Kappa : 0.9931          
##  Mcnemar's Test P-Value : NA              
## 
## Statistics by Class:
## 
##                      Class: A Class: B Class: C Class: D Class: E
## Sensitivity            1.0000   0.9903   0.9951   0.9876   0.9963
## Specificity            0.9981   0.9989   0.9967   0.9994   1.0000
## Pos Pred Value         0.9952   0.9956   0.9846   0.9969   1.0000
## Neg Pred Value         1.0000   0.9977   0.9990   0.9976   0.9992
## Prevalence             0.2845   0.1935   0.1743   0.1638   0.1839
## Detection Rate         0.2845   0.1917   0.1735   0.1618   0.1832
## Detection Prevalence   0.2858   0.1925   0.1762   0.1623   0.1832
## Balanced Accuracy      0.9991   0.9946   0.9959   0.9935   0.9982
```

Out of Sample Errors (from OOSE variables): Decision Tree: 29.04% Random Forest: 0.54%

Accuracy (from Confusion Matrix): Decision Tree: 70.1% Random Forest: 99.46%

I am going to use the Random forest due to its low generalization error, and high accuracy.

```
##  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
```