Shiny Application and Reproducible Pitch

JahlaJazz

October 30, 2016

Excutive Summary

The purpose of this presentation, is to create a front-end application for "what-if-analysis". We are going to revisit the dataset, provided in the Practical Machine Learning course and use a random forest model to fit the data. In this approach the following parameters will serve as front-end input items

1. Select the percentage to use, in creating the training and testing datasets. These datasets will be used for model fitting and parameter tuning
2. Select from a list of predictors, the variable to use in the regression model
3. Select the number of folds to used for cross-fold validation
4. Select the depth of the random forest tree

Explanation of the datasets used in this presentation

The focus, of the data, is to predict the manner in which the Unilateral Dumbbell Biceps Curl excercise was performed by a group of six participants, ranging in age from 20 to 28. Each particpants was asked to perform the excercise in five different ways, while wearing electronic devices designed to record various measurements, of the activity. Following is a listing of the ways in which the excercise was performed, of which only the first, is viewed as correct.

- classe A: exactly according to specification (ie, the correct way to do a curl)
- classe B: throwing the elbows to the front
- classe C: lifting the dumbbell only halfway
- classe D: lowering the dumbbell only halfway
- classe E: throwing the hips to the front

The results from the electronic readings and "classe" (the target variable) was recorded and a data set of 19,622 samples with 160 variable was used as the starting point for creating a model. According to the instruction, any combination of variable can be used to create the model and must be validated against another dataset of 20 samples, from which the "classe" variable is to be assigned, based upon the variables for each sample.

Outputs reflected in the shiny application

Given the input items, specified by the user, show the following items:

```
1. Summary information on the fitted model
2. A visual representation of the relationship between the number of predictors and accuracy
3. The relative importance of the selected predictors
4. Confusion Matrix and Statistics
5. Testing the model against the validation dataset
```

Following is a summary of my input assumptions

1. The initial dataset, before any adustment, was 19,622 samples with 160 variable
2. After adjustments for tidiness, the origianl dataset was reduced to 53 variable
3. The percentage used for creating the training dataset was 75%, which resulted 14,718 observation and 53 variable. The testing dataset contained the remaining observation.
4. For the random forest method, 5-fold cross validation and a tree depth of 50 was used.

Outputs resulting from input assumptions

The first output is are the predictors used for regression, the second is the fitted model, the third is the confusion matrix and the last are the predictions based on the validation dataset:

```
## Warning: package 'caret' was built under R version 3.3.1
```

```
## Warning: package 'lattice' was built under R version 3.3.1
```

```
## Warning: package 'randomForest' was built under R version 3.3.1
```

```
##  [1] "roll_belt"            "pitch_belt"           "yaw_belt"
##  [4] "total_accel_belt"     "gyros_belt_x"         "gyros_belt_y"
##  [7] "gyros_belt_z"         "accel_belt_x"         "accel_belt_y"
## [10] "accel_belt_z"         "magnet_belt_x"        "magnet_belt_y"
## [13] "magnet_belt_z"        "roll_arm"             "pitch_arm"
## [16] "yaw_arm"              "total_accel_arm"      "gyros_arm_x"
## [19] "gyros_arm_y"          "gyros_arm_z"          "accel_arm_x"
## [22] "accel_arm_y"          "accel_arm_z"          "magnet_arm_x"
## [25] "magnet_arm_y"         "magnet_arm_z"         "roll_dumbbell"
## [28] "pitch_dumbbell"       "yaw_dumbbell"         "total_accel_dumbbell"
## [31] "gyros_dumbbell_x"     "gyros_dumbbell_y"     "gyros_dumbbell_z"
## [34] "accel_dumbbell_x"     "accel_dumbbell_y"     "accel_dumbbell_z"
## [37] "magnet_dumbbell_x"    "magnet_dumbbell_y"    "magnet_dumbbell_z"
## [40] "roll_forearm"         "pitch_forearm"        "yaw_forearm"
## [43] "total_accel_forearm"  "gyros_forearm_x"      "gyros_forearm_y"
## [46] "gyros_forearm_z"      "accel_forearm_x"      "accel_forearm_y"
## [49] "accel_forearm_z"      "magnet_forearm_x"     "magnet_forearm_y"
## [52] "magnet_forearm_z"     "classe"
```

```
## Random Forest
##
## 14718 samples
##    52 predictor
##     5 classes: 'A', 'B', 'C', 'D', 'E'
##
## No pre-processing
## Resampling: Cross-Validated (5 fold)
## Summary of sample sizes: 11775, 11773, 11775, 11774, 11775
## Resampling results across tuning parameters:
##
##   mtry  Accuracy   Kappa
##    2    0.9889254  0.9859887
##   27    0.9909632  0.9885680
##   52    0.9834898  0.9791109
##
## Accuracy was used to select the optimal model using  the largest value.
```

```
## The final value used for the model was mtry = 27.
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction    A    B    C    D    E
##          A 1391   11    0    0    0
##          B    4  936    5    0    1
##          C    0    2  847    6    2
##          D    0    0    3  798    3
##          E    0    0    0    0  895
##
## Overall Statistics
##
##                Accuracy : 0.9925
##                  95% CI : (0.9896, 0.9947)
##     No Information Rate : 0.2845
##     P-Value [Acc > NIR] : < 2.2e-16
##
##                   Kappa : 0.9905
##  Mcnemar's Test P-Value : NA
##
## Statistics by Class:
##
##                      Class: A Class: B Class: C Class: D Class: E
## Sensitivity            0.9971   0.9863   0.9906   0.9925   0.9933
## Specificity            0.9969   0.9975   0.9975   0.9985   1.0000
## Pos Pred Value         0.9922   0.9894   0.9883   0.9925   1.0000
## Neg Pred Value         0.9989   0.9967   0.9980   0.9985   0.9985
## Prevalence             0.2845   0.1935   0.1743   0.1639   0.1837
## Detection Rate         0.2836   0.1909   0.1727   0.1627   0.1825
## Detection Prevalence   0.2859   0.1929   0.1748   0.1639   0.1825
## Balanced Accuracy      0.9970   0.9919   0.9941   0.9955   0.9967
```

```
##  [1] 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
```