Predicting how well a dumbbell exercice is performed using machine learning

Summary

Based on the data gathered from accelerometers on the belt, forearm, arm, and dumbell during an expererience involving 6 helathy participants performing the Unilateral Dumbbell Biceps Curl exercice in 5 differentes fashions, a Generalized Boosted Machine (gbm) model has been trained to predict how well the exercice was performed. The model was choosen with cross validation and an out-of-sample error of was estimated, revealing a high performing predictive model.

Data exploraton and pre-processing

Once the data is loaded, we can see that there are many columns with an overwhelming number of NAs.

Here we calculate the share of NAs for each columns, and we get a table that looks like this (we exclude the first 7 columns that are just names, timestamps and measuring window times):

```
##                 colName   class   NA_rate
## 9            pitch_belt numeric 0.0000000
## 10             yaw_belt numeric 0.0000000
## 11     total_accel_belt integer 0.0000000
## 12   kurtosis_roll_belt  factor 0.9793089
## 13  kurtosis_picth_belt  factor 0.9793089
## 14    kurtosis_yaw_belt  factor 0.9793089
## 160              classe  factor 0.0000000
```

It seems that either the column has no NA or has a very big amount of them (the “classe”" column in the end has no NAs, which is great because we will train our model on this attribute !):

```
##     NA_rate Col_count
## 1 0.0000000        52
## 2 0.9793089       100
```

In fact there are 52 columns without NAs, and 100 columns with 97.93% of NAs (here we did not consider the “classe” column as it will be the one we will train the model on).

Let’s exclude those 100 columns and train our model on the 52 others.

Cross-validation and selected model characterisation

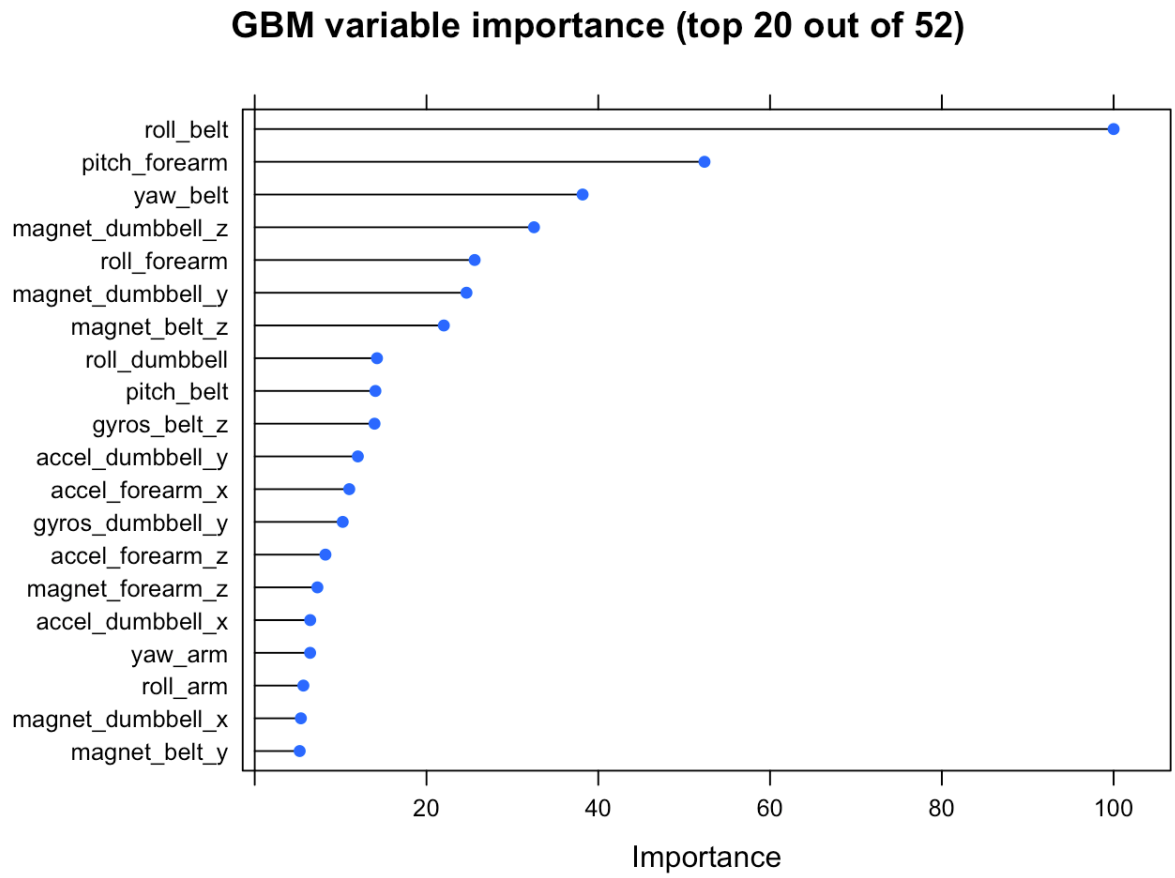
Using the caret and gbm packages, we train Generalized Boosted Machine (gbm) models on the pre-processed dataset. The “trControl” options is set with 10-fold cross validation, and the savePredictions option is set as TRUE, as we need those predictions for the ex-post out-of-sample error estimation.

The model has the following attributes:

```
## Stochastic Gradient Boosting
##
## 19622 samples
##    52 predictor
##     5 classes: 'A', 'B', 'C', 'D', 'E'
##
## No pre-processing
## Resampling: Cross-Validated (10 fold)
## Summary of sample sizes: 17660, 17659, 17660, 17660, 17661, 17660, ...
## Resampling results across tuning parameters:
##
##   interaction.depth  n.trees  Accuracy   Kappa      Accuracy SD
##   1                   50      0.7541533  0.6883012  0.008934982
##   1                  100      0.8222388  0.7750427  0.007818834
##   1                  150      0.8559765  0.8177390  0.008010237
##   2                   50      0.8578618  0.8199445  0.009076143
##   2                  100      0.9091311  0.8850248  0.007233971
##   2                  150      0.9336948  0.9160964  0.006485377
##   3                   50      0.8989375  0.8720556  0.010314373
##   3                  100      0.9436330  0.9286800  0.007346711
##   3                  150      0.9625406  0.9526110  0.006161470
##   Kappa SD
##   0.011520443
##   0.009893179
##   0.010095147
##   0.011507641
##   0.009175571
##   0.008215973
##   0.013096995
##   0.009304616
##   0.007803249
##
## Tuning parameter 'shrinkage' was held constant at a value of 0.1
##
## Tuning parameter 'n.minobsinnode' was held constant at a value of 10
## Accuracy was used to select the optimal model using  the largest value.
## The final values used for the model were n.trees = 150,
##  interaction.depth = 3, shrinkage = 0.1 and n.minobsinnode = 10.
```

So the final model choosen based on Accuracy has 150 trees, and interaction depth of 3, a shrinkage value set as 0.1 and finally a minimum observation in each node set as 10 observations.

Here we look at the top 20 most important variables in the model:



We then estimate the out-of-sample error based on the saved predictions from the 10-fold cross-validation.

Out-of-sample error estimation

For each of the 10 folds of the cross-validation, we look at the number of misclassified observation of the hold-out dataset for the specific model that has been eventually selected and that we described above :

```
##     Folds Misclass Total
## 1  Fold01       68  1962
## 2  Fold02       70  1963
## 3  Fold03       52  1962
## 4  Fold04       78  1962
## 5  Fold05       69  1961
## 6  Fold06       80  1962
## 7  Fold07       80  1963
## 8  Fold08       62  1964
## 9  Fold09       80  1963
## 10 Fold10       96  1960
```

Now to compute and estimation of the out-of-sample error, we just average the error rate across those 10 folds.

The out-of-sample error is estimated as around 3.75%, which is quite low, revealing a relatively good model performance for this type of setup.

Conclusion

The model trained on the dataset seems to be particularly good at predicting the way the Unilateral Dumbbell Biceps Curl exercice is performed, but we have to underline that this is based on a relatively small dataset out of only 6 participants who were all young and healthy men. These results are only valid for new data generated from them. To be generalized (or not), this model should be trained on more data from more participants of different age, size and health conditions.