Project Report on Human Activity Recognition, as part of the Coursera Practical Machine Learning course

Alejandro G. Bedregal

DATA SET & EXPLORATORY DATA ANALYSIS

In this study, we use data from the WLE dataset [1] to build a model that predicts if a person is doing barbell lifts correctly. Briefly, 6 male participants were asked to perform barbell lifts correctly and incorrectly in 5 different ways (A,B,C,D and E). The data was aquired through 4 accelerometers located in the 'belt', 'forearm', 'arm' and 'dumbbell'. The dataset consists of 38 variables measured for each of the 4 accelerometers and including 14,718 measurements. More information is available from the WLE dataset website: http://groupware.les.inf.puc-rio.br/har

We started by splitting the dataset in a randomly selected trainning (3/4 of total dataset) and testing (1/4) subsamples

```
inTrain <- createDataPartition(y=Data0$classe, p=0.75, list=FALSE)
trainData0 <- Data0[inTrain,]
testData0 <- Data0[-inTrain,]
```

In what follows, we put aside the testing sample and worked with the trainning sample. In a **first step of pre-processing** the data, we remove variables that were not properly defined in the dataset (i.e., NA), leaving **13 different measurements for each accelerometer**. These parameters include: roll, pitch, yaw and total acceleration; 'gyro', 'magnet' and acceleration in each of the 3 cartesian spatial dimensions

We perform an **exploratory data analysis**. In Figures 1 and 2 we show examples of dispersion plots for measuremets from the accelerometers in the arm and dumbbell, respectively.

Figure 1: Example of 4 Arm Accel. param. correlations

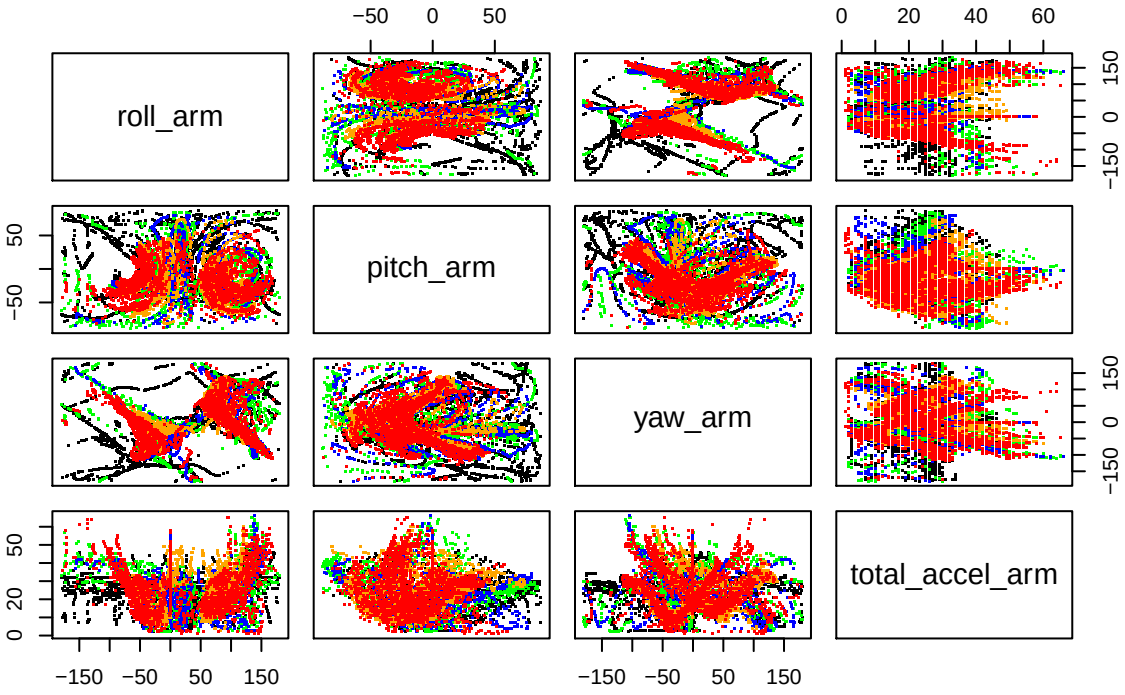
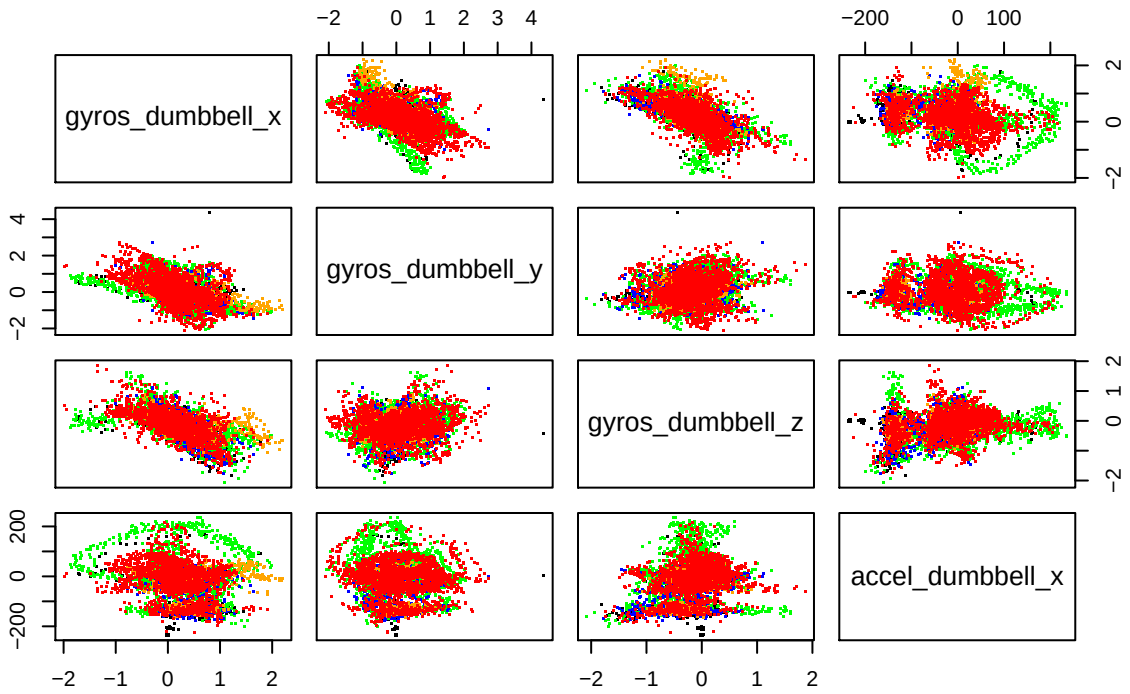


Figure 2: Example of 4 Dumbbell Accel. param. correlations



In 5 different colors we flag the data according to the 5 different ways to do the excercise. As we see, complex patterns emerge from some of the different parameters measured in a given accelerator. Also, some of them seem to be highly correlated. By calculating **correlation matrices** for the 13 parametrs of each accelerometer we found that for several parameter pairs the correlations are >0.8. Instead, such large numbers of highly correlated pairs do not appear as often if parameters from different accelerator are compared.

PRINCIPAL COMPONENT ANALYSIS (PCA)

The exploratory data analysis described above suggests that we can reduce the number of parameters for each accelerometer as many of those parameters are strongly correlated. We decided to continue our data pre-processing by performing a **Principal Component Analysis (PCA)** for each of the accelertor parameter-sets. For example, here we use the CARET package in the 13-parameter data from the arm accelerometer:

```
pcArm <-preProcess(trainData_Arm[,2:14], method="pca", thresh=0.9)
pcPred_Aarm <- predict(pcArm,trainData_Arm[,2:14])
```

For each accelerometer we capture >90% of the total variance. In this way we reduce the number of model parameters and, at the same time, we allowed our model results to be interpretable in function of each of the 4 accelerometers. As a result of our PCA for each accelerometer we reduce from 13 to 7 the total number of parameters for the arm accelerometer. In a similar way, we reduce the number of parameters to 4, 6 and 8 for the belt, dumbbell and forearm accelerometers, respectively. In summary, we reduced from 52 to 25 the number of parameters to fit with our model and still retain >90% of the variability between parameters.

THE MODEL

We decided to use *Generalized Boosted Models (gbm)* to perform our modeling. For our dataset, it provided better accuracy compared to other techniques like the Linear Discriminant Analysis (LDA).

```
trainPC <- data.frame(pcPred_Arm, pcPred_Belt, pcPred_Dumbblee, pcPred_Forearm)
modelFit_GBM <- train(trainData$classe ~., method="gbm", data=trainPC)
```

The final values used for the best gbm model were n.trees = 150, interaction.depth = 3, shrinkage = 0.1 and n.minobsinnode = 10. Accuracy (= 0.884 for the best model) was used to select the optimal model using the largest value.

OUT OF SAMPLE ERROR AND CROSS-VALIDATION

We evaluate the **out of sample error** using **cross-validation**. We quantify our errors through 'Accuracy' (i.e., the probability of getting a correct outcome) and 'Concordance' (ideal for multi-class data like ours), parametrized with *kappa* parameter.

First, we use our trainning dataset and split it in a sub-trainning set (3/4 of original training dataset) and a sub-testing set (1/4). Then we build our gbm model in an analogous way as described in the previous section, and we evaluate our best gmb model in the sub-testing set. We repeat this process 30 times. In each iteration we randomly selected our sub-training and sub-testing datasets, and store the retrieved accuracy and kappa. Finally, we estimated bi-weight means for the resulting distributions in accuracy and kappa. Our mean out-of-sample error extimations are

$Accuracy = 0.872$ (lower than the 0.884 in-sample estimation)

$Kappa = 0.841$

TESTING OUR MODEL

Finally, we test our best gmb model in the testing dataset. We applied the same data pre-processing used for the trainning set to the testing set (i.e., selecting the 13 relevant parameters for each accelerator, PCA using the results found for the trainning dataset).

```
predTest <- predict(modelFit_GBM,testPC)
```

```
confusionMatrix(predTest,testData$classe)
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction    A    B    C    D    E
##          A 1327   92   28   21   12
##          B   22  777   48    7   27
##          C   14   63  757   65   38
##          D   26    8   19  695   25
##          E    6    9    3   16  799
##
## Overall Statistics
##
##                Accuracy : 0.8881
##                  95% CI : (0.8789, 0.8967)
##     No Information Rate : 0.2845
##     P-Value [Acc > NIR] : < 2.2e-16
##
##                   Kappa : 0.8581
##  Mcnemar's Test P-Value : < 2.2e-16
##
## Statistics by Class:
##
##                      Class: A Class: B Class: C Class: D Class: E
## Sensitivity            0.9513   0.8188   0.8854   0.8644   0.8868
## Specificity            0.9564   0.9737   0.9555   0.9810   0.9915
## Pos Pred Value         0.8966   0.8820   0.8079   0.8991   0.9592
```

```
## Neg Pred Value         0.9801   0.9572   0.9753   0.9736   0.9749
## Prevalence             0.2845   0.1935   0.1743   0.1639   0.1837
## Detection Rate         0.2706   0.1584   0.1544   0.1417   0.1629
## Detection Prevalence   0.3018   0.1796   0.1911   0.1576   0.1699
## Balanced Accuracy      0.9538   0.8962   0.9205   0.9227   0.9391
```

By comparing the predicted ways of doing the excercise (A, B, C, D and E) with the real values, the 'confussionMatrix' task shows us our best gmb model recovers 89% of the results correctly.

REFERENCES

[1] Velloso, E.; Bulling, A.; Gellersen, H.; Ugulino, W.; Fuks, H. Qualitative Activity Recognition of Weight Lifting Exercises. Proceedings of 4th International Conference in Cooperation with SIGCHI (Augmented Human '13) . Stuttgart, Germany: ACM SIGCHI, 2013.