Executive Summary:

Executive Summary:

The purpose of this exercise is to use predictive machine learning to predict how well a group of individuals performed a specific exercise. Devices such as Jawbone “Up”, Nike “FuelBand” and Fitbit are making it possible to collect a large amount of data about personal activity. People who use these devices regularly quantify how much of a particular activity they do, but they rarely quantify how well they do it. In a study performed by Velloso, Bulling, Gellersen, Ugulino and Fuks, (see references), it does provide a measure, “classe”, (among many other variables) of how well 6 males, each perform one set of 10 repetitions of the Unilateral Dumbbell Bicep Curl in 5 different fashions: exactly according to specification (class A), throwing the elbows to the front (class B), lifting the dumbbell only halfway (class C), lowering the dumbbell only halfway (calls D) and throwing the hips to the front (class E).

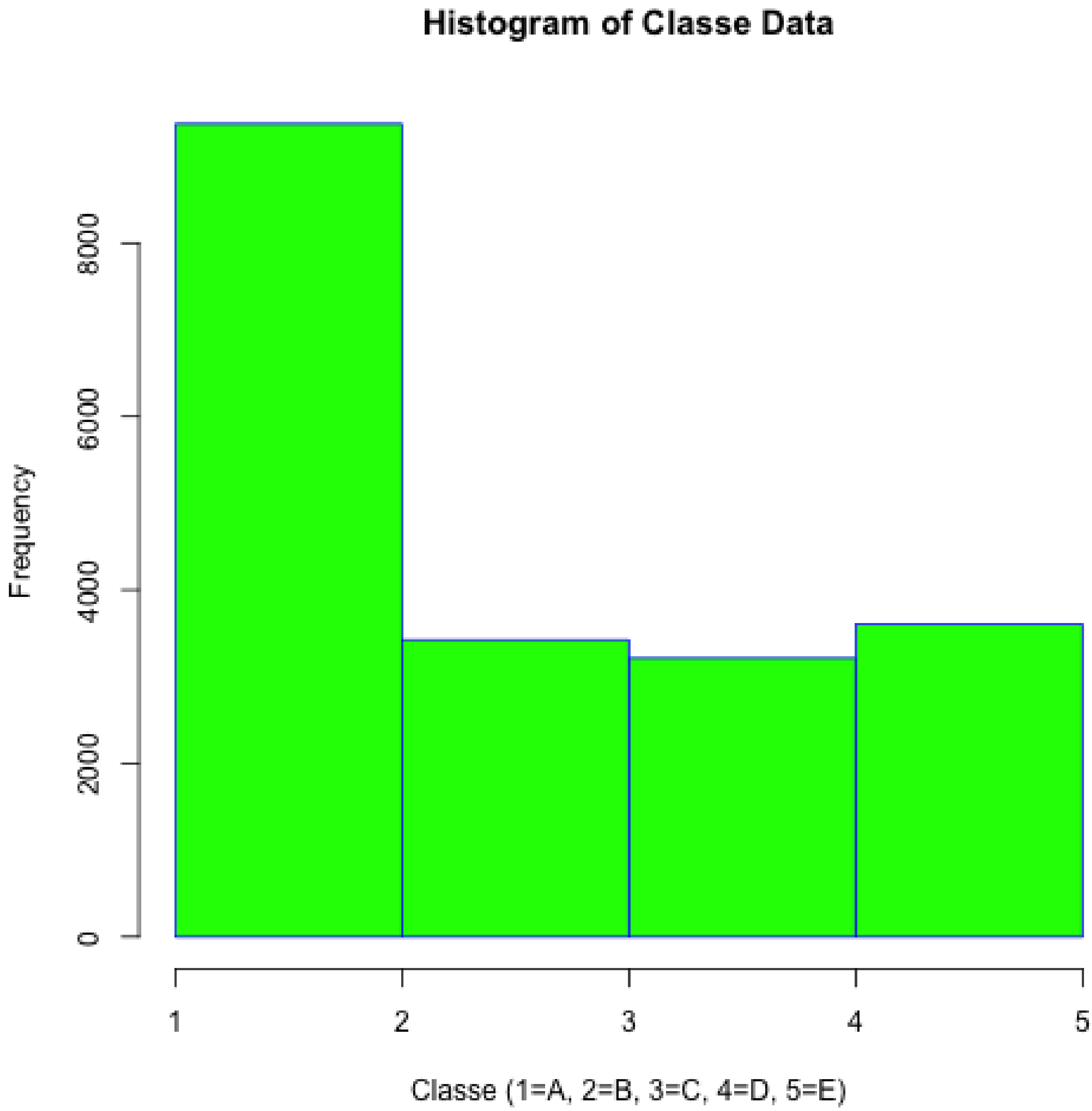
Data:

The dataset referenced above is defined as the “training” dataset which the author will use to train predictive models against. The overall training dataset will be divided into a training dataset and a cross-validation set. Once the models have been trained against the training dataset and the most accurate method selected, then that method will be used against the cross-validation set to determine the predictive accuracy. Given that the accuracy remains high (95% or better) then, and only then, will the model be used against the actual “test” data set.

Cleaning the data:

By looking at the training data there are multiple columns which either do not contain numerical data (ie “N/A”, " ", or 0) or data that are not applicable to the model (the first 7 columns). All of these columns need to be eliminated from the final training data frame. The testing data will require the same cleaning method. The following histogram is a snap-shot of the training data:

```
#Clean training data
trainnew <- data.frame(dlTrain)
trainnew <-trainnew[ , ! apply(trainnew, 2, function(x) any(is.na(x)))]
trainnew <-trainnew[ , ! apply(trainnew, 2, function(x) any(x == ""))]
trainnew <-trainnew[ , ! apply(trainnew, 2, function(x) all(x == 0))]
trainnew <-trainnew[ , -(1:7), drop=FALSE]
g<-as.numeric(trainnew$classe)
hist(g, main="Histogram of Classe Data", xlab="Classe (1=A, 2=B, 3=C, 4=D, 5=E)", border="blue", col="green",
```


Histogram of Classe Data
Frequency
0
2000
4000
6000
8000
1
2
3
4
5
Classe (1=A, 2=B, 3=C, 4=D, 5=E)

Conduct Data Splitting:

In order to begin the model-training process we need to randomly split the training data set into training ("training") and cross-validation ("cross_val"). I use a 75/25% split.

Fit the model/ Train the model:

This step utilizes both "set.seed" and cross-validation processes, as well as "train" and "predict" functions to determine the most accurate model to utilize. The models which are evaluated against each other here are the "Rpart", "Random Forest" and "GBM".

The accuracies of the models are as follows:

R Part: 0.4978937

Random Forest: 1

GBM: 0.9734339

Therefore, the Random Forest model appears to be the most accurate and will be used from here on.

Predict the Out-of-Sample Error:

This prediction entails using the RF model, cross-validation and the "predict" function to determine the model's accuracy and out-of-sample error against the cross-validation dataset.

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction    A    B    C    D    E
##          A 1395    0    0    0    0
##          B    0  949    0    0    0
##          C    0    0  855    0    0
##          D    0    0    0  804    0
##          E    0    0    0    0  901
##
## Overall Statistics
##
##                Accuracy : 1
##                  95% CI : (0.9992, 1)
##     No Information Rate : 0.2845
##     P-Value [Acc > NIR] : < 2.2e-16
##
##                   Kappa : 1
##  Mcnemar's Test P-Value : NA
##
## Statistics by Class:
##
##                      Class: A Class: B Class: C Class: D Class: E
## Sensitivity            1.0000   1.0000   1.0000   1.0000   1.0000
## Specificity            1.0000   1.0000   1.0000   1.0000   1.0000
## Pos Pred Value         1.0000   1.0000   1.0000   1.0000   1.0000
## Neg Pred Value         1.0000   1.0000   1.0000   1.0000   1.0000
## Prevalence             0.2845   0.1935   0.1743   0.1639   0.1837
## Detection Rate         0.2845   0.1935   0.1743   0.1639   0.1837
## Detection Prevalence   0.2845   0.1935   0.1743   0.1639   0.1837
## Balanced Accuracy      1.0000   1.0000   1.0000   1.0000   1.0000
```

The accuracy of the prediction of out-of-sample error is 1.

The out-of-sample error is 0

Importance of variables in the model must be known for further regressive analysis and to determine the highest level of correlation between which variables and the variable (classe) that we want to predict.

```
## [1] "The top 5 variables in importance:"
```

```
##                     Overall
## roll_belt         100.00000
## pitch_forearm      67.32039
## yaw_belt           52.64770
## magnet_dumbbell_z  51.69843
## pitch_belt         44.90532
```

These results show high enough accuracy that we can use this same model/ method against the original test data set. This test data set does not contain any "Classe" variables and the predictive machine learning model will give predicted values for classe. These predicted values are as follows:

```
##  [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
```

References:

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 SIGHI. Stuttgart, GE, 2013