Prediction Assignment Writeup

MisterT1000

8/9/2017

Initial Setup

I'm loading the caret and random forest libraries and setting the working directory to begin.

```
library(caret)
library(randomForest)
setwd("~/test-repo/PracticalMachineLearningProject/")
```

Importing and cleaning data

I imported the data and changed the NA's to 0. The next step is removing the data elements that are not actually predictors, such as the name, window, and timestamp columns. I also made sure to convert data that came up as factors back to the numeric types they should be.

```
dataset <- read.csv("pml-training.csv")
dataset[is.na(dataset)] <- 0
dataset <- dataset[,-c(1:7)]
dataset[,sapply(dataset, class)=="factor"][-34] <- sapply(dataset[,(sapply(data
set, class)=="factor")],as.numeric)[-34]
```

```
## Warning in matrix(value, n, p): data length [667147] is not a sub-multiple
## or multiple of the number of rows [19622]
```

I'm using the X,Y, and Z component predictors because the others are generally calculated from these and, as a result, my hypothesis is that those would not be as signinficant.

```
dataset <- dataset[,grepl("_x$|_y$|_z$|classe",names(dataset))]
```

I split my training data into a new training and validation set.

```
training <- createDataPartition(y = dataset$classe, p = 0.7, list = FALSE)
train <- dataset[training, ]
validation <- dataset[-training, ]
```

I'm using the random forest method to build my model.

```
modRF <- randomForest(classe~.,data=train)
```

Find the out of sample error rate with the validation set.

```
pred <- predict(modRF,validation)
result <- confusionMatrix(pred,validation$classe)
result
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction    A    B    C    D    E
##          A 1667    9    1    7    0
##          B    5 1123   13    0    0
##          C    0    7 1011   23    4
##          D    2    0    1  934    3
##          E    0    0    0    0 1075
##
## Overall Statistics
##
##                Accuracy : 0.9873
##                  95% CI : (0.9841, 0.99)
##     No Information Rate : 0.2845
##     P-Value [Acc > NIR] : < 2.2e-16
##
##                   Kappa : 0.9839
##  Mcnemar's Test P-Value : NA
##
## Statistics by Class:
##
##                      Class: A Class: B Class: C Class: D Class: E
## Sensitivity            0.9958   0.9860   0.9854   0.9689   0.9935
## Specificity            0.9960   0.9962   0.9930   0.9988   1.0000
## Pos Pred Value         0.9899   0.9842   0.9675   0.9936   1.0000
## Neg Pred Value         0.9983   0.9966   0.9969   0.9939   0.9985
## Prevalence             0.2845   0.1935   0.1743   0.1638   0.1839
## Detection Rate         0.2833   0.1908   0.1718   0.1587   0.1827
## Detection Prevalence   0.2862   0.1939   0.1776   0.1597   0.1827
## Balanced Accuracy      0.9959   0.9911   0.9892   0.9838   0.9968
```

Apply the model to the test data set to make a prediction.

```
finalTest <- read.csv("pml-testing.csv")
finalTest[is.na(finalTest)] <- 0
finalPred <- predict(modRF,finalTest)
finalPred
```

```
##  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
```