Machine Learning: Motion Sensors

IvLL

July 11, 2016

Project: Analysing Data From Wearable Devices

Synopsis

The purpose of this study is to create a model to predict the movements of a particular exercise, the Unilateral Dumbbell Biceps Curls. The models do this by analysing datasets from kinetic sensors which can be found here: Training: https://d396qusza40orc.cloudfront.net/predmachlearn/pml-training.csv (https://d396qusza40orc.cloudfront.net/predmachlearn/pml-training.csv) Testing: https://d396qusza40orc.cloudfront.net/predmachlearn/pml-testing.csv (https://d396qusza40orc.cloudfront.net/predmachlearn/pml-testing.csv)

The results of this study are that a Random Forest model was able to predict the test set near 100% accuracy.

DownLoad Data

The Data was downloaded at the source and stored locally

```
#train data
trainURL <- "https://d396qusza40orc.cloudfront.net/predmachlearn/pml-training.csv?accessType = D
OWNLOAD"
download.file(trainURL, destfile="train.csv")

#test data
testURL <- "https://d396qusza40orc.cloudfront.net/predmachlearn/pml-testing.csv?accessType = DOW
NLOAD"
download.file(testURL, destfile="test.csv")
```

PrePreprocess the data

The raw csv are processed by removing the first seven columns which do not contian information relevant to the analysis and removing rows with NAs.

```
Train1<- read.csv("train.csv",na.strings = c("","#DIV/0!","NA"),header = TRUE)
Test2<- read.csv('test.csv',na.strings=c('','NA'))
Train2<-Train1[,!apply(Train1,2,function(x) any(is.na(x)) )]
Train<-Train2[,-c(1:7)]
Train$classe <- as.factor(Train$classe)
dim(Train)
```

```
## [1] 19622    53
```

Analysis of data and Variables

```
#Load required libraries
library(caret)
```

```
## Loading required package: lattice
```

```
## Loading required package: ggplot2
```

```
library(randomForest)
```

```
## randomForest 4.6-12
```

```
## Type rfNews() to see new features/changes/bug fixes.
```

```
##
## Attaching package: 'randomForest'
```

```
## The following object is masked from 'package:ggplot2':
##
##     margin
```

```
library(e1071)
```

Data Paritioning

We begin by partition the training set into a 60/40 training and validation set. The dataset is large enough (19622 entries) to allow for this split.

```
# partition the train set
set.seed(1994)
partrain <- createDataPartition(y=Train$classe,p=0.6, list=FALSE)
train <- Train[partrain,]
test <- Train[-partrain,]
dim(test)
```

```
## [1] 7846   53
```

```
dim(train)
```

```
## [1] 11776    53
```

```
# samtrain<-train[sample(nrow(train),5000),] # for testing code
```

Modeling the Data

The random forest tree algorithm was chosen to model the data. The model is first trained on the training set we see the the accuracy on the training set is a robust 98.9%.

```
rf<-randomForest(classe~., data=train, method='class')
trainpreds<-predict(rf,Train, type='class')
trainconf<-confusionMatrix(trainpreds,Train$classe)
save(trainconf,file='test.RData')
```

Test set performance

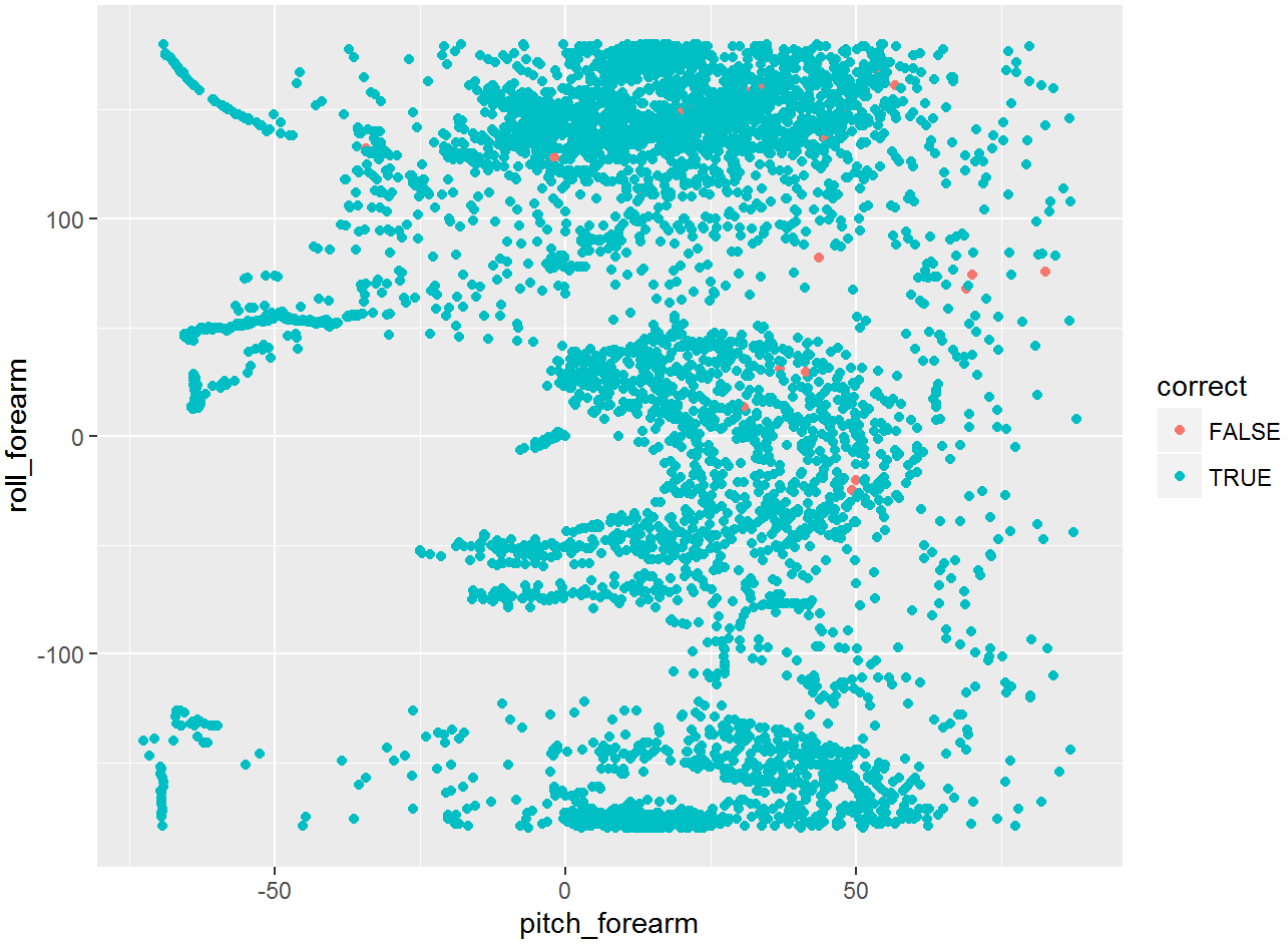
Using the model we trained on the training set we make predictions on the test set and find that it 99.5% accurate. The plot shows the points that were predicted incorrectly.

```
testpreds <-predict(rf,test,type='class')
testconf <-confusionMatrix(testpreds,test$classe)

sum(testpreds!=test$classe)/length(test$classe)
```

```
## [1] 0.005098139
```

```
correct<-testpreds == test$classe
tpreds<-qplot(pitch_forearm, roll_forearm, colour=correct,data=test)
tpreds
```



Prediction on the 20 observations

The model is used to predict the values for the quality of lifts for the 20 observations.

```
Test1 <- Test2[,!apply(Test2,2,function(x) any(is.na(x)) )]
Test <- Test1[,-c(1:7)]
pred_final<-predict(rf,Test,type='class')
save(pred_final,file='testpredictedc.RData')
```