Machine Learning Final Project

Philip Mayfield

November 17, 2017

Background

Using the data from HAR (Human Activity Recognition) I am attempting to identify if a person has performed an exercise correctly using data from accelerometers worn during exercise. The humans were asked to lift weights correctly and with give misfunctions. From this, the exercise was placed into five classifications. For more information see this link. http://groupware.les.inf.puc-rio.br/har

This analysis will predict the classifiction based on the acceleromter data.

Model Building

The model was created by splitting the data into two datasets: training and crossvalidation. Note: the test dataset (20 rows) is a third dataset provided by the instructor. After splitting, I used a random forest to create a classifier.

```
library(caret)

## Warning: package 'caret' was built under R version 3.4.2

## Loading required package: lattice

## Loading required package: ggplot2

## Warning: package 'ggplot2' was built under R version 3.4.2

library(rattle)

## Rattle: A free graphical interface for data science with R.
## Version 5.1.0 Copyright (c) 2006-2017 Togaware Pty Ltd.
## Type 'rattle()' to shake, rattle, and roll your data.

library(parallel)
library(doParallel)

## Warning: package 'doParallel' was built under R version 3.4.2

## Loading required package: foreach

## Loading required package: iterators

#Read the data from disk
AllData <- read.csv("pml-training.csv")
TestData <- read.csv("pml-testing.csv")
```

```
#use only columns with data suitable for analysis (remove time and blank rows
)
AllData <- AllData[,c(2,6:11,37:48,60:68,84:86,102,113:124,140,151:160)]
##create training and crossvalidation datasets
inTrain <- createDataPartition(y=AllData$classe, p=0.7, list=FALSE)
training <- AllData[inTrain,]
crossvalidation <- AllData[-inTrain,]


##setup parallel processing, note this didn't work but the R developer said i
t would be fixed this week
cluster <- makeCluster(8)
registerDoParallel(cluster)
#Configure trainControl object
fitControl <- trainControl(method = "cv",number = 2,allowParallel = FALSE)  #
I ended up running this serial not parallel due to R bug

##rpart is R's method for Classification and regression trees (with parallel)
modFit <- caret::train(classe  ~ .,method="rf",data=training,trControl = fitC
ontrol)

## randomForest 4.6-12

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

##
## Attaching package: 'randomForest'

## The following object is masked from 'package:rattle':
##
##     importance

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

##return to single core processing
stopCluster(cluster)
registerDoSEQ()
```

Cross Validation

I kept 30% of the data separate from the training data for cross validation. The confusion matrix below shows that the model is predicting quite well. The model has 99.85% accuracy and a Kappa = .9981 which are both amazingly accurate.

```
#Predicting new values
pred<-predict(modFit,newdata=crossvalidation)
result <- confusionMatrix(crossvalidation$classe, pred)
result
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction    A    B    C    D    E
##          A 1673    0    0    0    1
##          B    3 1136    0    0    0
##          C    0    0 1026    0    0
##          D    0    0    2  961    1
##          E    0    0    0    2 1080
##
## Overall Statistics
##
##                Accuracy : 0.9985
##                  95% CI : (0.9971, 0.9993)
##     No Information Rate : 0.2848
##     P-Value [Acc > NIR] : < 2.2e-16
##
##                   Kappa : 0.9981
##  Mcnemar's Test P-Value : NA
##
## Statistics by Class:
##
##                      Class: A Class: B Class: C Class: D Class: E
## Sensitivity            0.9982   1.0000   0.9981   0.9979   0.9982
## Specificity            0.9998   0.9994   1.0000   0.9994   0.9996
## Pos Pred Value         0.9994   0.9974   1.0000   0.9969   0.9982
## Neg Pred Value         0.9993   1.0000   0.9996   0.9996   0.9996
## Prevalence             0.2848   0.1930   0.1747   0.1636   0.1839
## Detection Rate         0.2843   0.1930   0.1743   0.1633   0.1835
## Detection Prevalence   0.2845   0.1935   0.1743   0.1638   0.1839
## Balanced Accuracy      0.9990   0.9997   0.9990   0.9987   0.9989
```

Prediction of Testing Dataset

Below are the predictions for the 20 values in the test dataset.

```
pred<-predict(modFit,newdata=TestData)
pred

##  [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
```