Machine Learning Course Project

Cesar Garcia

May 11, 2017

Data

The training data for this project are available here: Traning Data (https://d396qusza40orc.cloudfront.net/predmachlearn/pml-training.csv)

The test data are available here: Test Data (https://d396qusza40orc.cloudfront.net/predmachlearn/pml-testing.csv)

Data is courtesy of

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

Load Data and Clean Predictors

Some of the data columns contain a lot of NAs and near zero values. These columns will be removed from the dataset used for training, hence will not be used in the predictions.

```
training <- read.csv("pml-training.csv")
testing <- read.csv("pml-testing.csv")

train <- training[, names(training)[!(nzv(training, saveMetrics = T)[, 4])]]
train <- train[, names(train)[sapply(train, function (x) ! (any(is.na(x) | x == 
""")))]]
train <- train[,-1]
test <- testing[, names(testing)[!(nzv(testing, saveMetrics = T)[, 4])]]
test <- test[, names(test)[sapply(test, function (x) ! (any(is.na(x) | x == """)))]]
testing <- read.csv("pml-testing.csv", na.strings = c("NA","#DIV/0!",""))
```

Separate data for Training and Cross Validation

```
inTrain <- createDataPartition(train$classe,p=0.6,list=FALSE)
subTraining <- train[inTrain,]
SubValidation <- train[-inTrain,]
```

Create a Random Forest Model

The different classe levels are displayed to ensure model address all classe. The model selecte will be a random forest. The data will be saved data to reduce in reprocessing times when project is recalculated.

```
table(subTraining$classe)
```

```
##
##    A    B    C    D    E
## 3348 2279 2054 1930 2165
```

```
# Check if model file exists
model <- "modelFit.RData"
if (!file.exists(model)) {

    date()
    fit <- train(classe ~ ., method = "rf", data = subTraining)
    date()
    save(fit, file = "modelFit.RData")

} else {
    # Good model exists from previous run, load it and use it.
    load(file = "modelFit.RData", verbose = TRUE)
}
```

```
## Loading objects:
##   fit
```

Accuracy and Sample Error

```
predTrain <- predict(fit, subTraining)
confusionMatrix(predTrain,subTraining$classe)
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction    A    B    C    D    E
##          A 3348    2    0    0    0
##          B    0 2277    0    0    0
##          C    0    0 2054    0    0
##          D    0    0    0 1930    2
##          E    0    0    0    0 2163
##
## Overall Statistics
##
##                Accuracy : 0.9997
##                  95% CI : (0.9991, 0.9999)
##     No Information Rate : 0.2843
##     P-Value [Acc > NIR] : < 2.2e-16
##
##                   Kappa : 0.9996
##  Mcnemar's Test P-Value : NA
##
## Statistics by Class:
##
##                      Class: A Class: B Class: C Class: D Class: E
## Sensitivity            1.0000   0.9991   1.0000   1.0000   0.9991
## Specificity            0.9998   1.0000   1.0000   0.9998   1.0000
## Pos Pred Value         0.9994   1.0000   1.0000   0.9990   1.0000
## Neg Pred Value         1.0000   0.9998   1.0000   1.0000   0.9998
## Prevalence             0.2843   0.1935   0.1744   0.1639   0.1838
## Detection Rate         0.2843   0.1934   0.1744   0.1639   0.1837
## Detection Prevalence   0.2845   0.1934   0.1744   0.1641   0.1837
## Balanced Accuracy      0.9999   0.9996   1.0000   0.9999   0.9995
```

The model has a an accuracy of 0.9997 within the training data. The cross validation data set will be processed through the same model to validate the model before we run the test data.

```
predValidation <- predict(fit,SubValidation)
confusionMatrix(predValidation,SubValidation$classe)
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction    A    B    C    D    E
##          A 2232    1    0    0    0
##          B    0 1517    0    0    0
##          C    0    0 1368    0    0
##          D    0    0    0 1286    2
##          E    0    0    0    0 1440
##
## Overall Statistics
##
##                Accuracy : 0.9996
##                  95% CI : (0.9989, 0.9999)
##     No Information Rate : 0.2845
##     P-Value [Acc > NIR] : < 2.2e-16
##
##                   Kappa : 0.9995
##  Mcnemar's Test P-Value : NA
##
## Statistics by Class:
##
##                      Class: A Class: B Class: C Class: D Class: E
## Sensitivity            1.0000   0.9993   1.0000   1.0000   0.9986
## Specificity            0.9998   1.0000   1.0000   0.9997   1.0000
## Pos Pred Value         0.9996   1.0000   1.0000   0.9984   1.0000
## Neg Pred Value         1.0000   0.9998   1.0000   1.0000   0.9997
## Prevalence             0.2845   0.1935   0.1744   0.1639   0.1838
## Detection Rate         0.2845   0.1933   0.1744   0.1639   0.1835
## Detection Prevalence   0.2846   0.1933   0.1744   0.1642   0.1835
## Balanced Accuracy      0.9999   0.9997   1.0000   0.9998   0.9993
```

The model has a 0.9997 accuracy with the validation data and an out of sample error of 0.0003. This provides us a reasonable assurance that the model would be a good predictor of new data.

The important predictors in the model are:

```
varImp(fit)
```

```
## rf variable importance
##
##   only 20 most important variables shown (out of 79)
##
##                                Overall
## raw_timestamp_part_1          100.000
## num_window                     45.487
## roll_belt                      42.954
## pitch_forearm                  26.779
## magnet_dumbbell_z              18.387
## roll_forearm                   13.913
## magnet_dumbbell_y              13.694
## yaw_belt                       13.486
## pitch_belt                     12.571
## cvtd_timestamp30/11/2011 17:12  10.731
## cvtd_timestamp02/12/2011 14:58   9.767
## magnet_dumbbell_x               7.396
## cvtd_timestamp02/12/2011 13:33   7.049
## cvtd_timestamp28/11/2011 14:15   6.853
## accel_belt_z                    6.255
## roll_dumbbell                   5.787
## accel_dumbbell_y                5.404
## cvtd_timestamp05/12/2011 11:24   5.315
## magnet_belt_y                   5.126
## accel_forearm_x                 5.015
```

```
fit$finalModel
```

```
##
## Call:
##  randomForest(x = x, y = y, mtry = param$mtry)
##                Type of random forest: classification
##                      Number of trees: 500
## No. of variables tried at each split: 40
##
##         OOB estimate of  error rate: 0.13%
## Confusion matrix:
##      A    B    C    D    E  class.error
## A 3346    2    0    0    0 0.0005973716
## B    3 2275    1    0    0 0.0017551558
## C    0    3 2051    0    0 0.0014605648
## D    0    0    4 1925    1 0.0025906736
## E    0    0    0    1 2164 0.0004618938
```

Predict Test Data With Model

The out-of-bag (OOB) error rate is 0.13% which provides a highlevel of acuracy. We will use this model to predict classe in the testdata

```
predTest <- predict(fit,testing)
predTest
```

```
##  [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
```

The predictions generated by the model were 100% accurate. The quiz resulted in 100% score.