Prediction Assignment Writeup

Srinivasa Rao Burri

October 25, 2018

Using caret and randomForest libraries, I have tried to generate correct answers for each of the 20 test data cases of this assignment. To do this I made use of caret and randomForest, this allowed me to generate correct answers for each of the 20 test data cases provided in this assignment.

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

```
## Loading required package: survival
```

```
## Loading required package: Formula
```

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

```
##
## Attaching package: 'Hmisc'
```

```
## The following objects are masked from 'package:base':
##
##     format.pval, units
```

```
##
## Attaching package: 'caret'
```

```
## The following object is masked from 'package:survival':
##
##     cluster
```

```
## randomForest 4.6-14
```

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

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

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

```
## Loading required package: iterators
```

```
## Loading required package: parallel
```

I have casted all data in 8 columns as numeric values. Displaying out our feature set.

```
feature_set <- colnames(training_data[colSums(is.na(training_data)) == 0])[-(1:7)]
model_data <- training_data[feature_set]
feature_set
```

```
##  [1] "roll_belt"            "pitch_belt"           "yaw_belt"
##  [4] "total_accel_belt"     "gyros_belt_x"         "gyros_belt_y"
##  [7] "gyros_belt_z"         "accel_belt_x"         "accel_belt_y"
## [10] "accel_belt_z"         "magnet_belt_x"        "magnet_belt_y"
## [13] "magnet_belt_z"        "roll_arm"             "pitch_arm"
## [16] "yaw_arm"              "total_accel_arm"      "gyros_arm_x"
## [19] "gyros_arm_y"          "gyros_arm_z"          "accel_arm_x"
## [22] "accel_arm_y"          "accel_arm_z"          "magnet_arm_x"
```

```
## [25] "magnet_arm_y"          "magnet_arm_z"          "roll_dumbbell"
## [28] "pitch_dumbbell"        "yaw_dumbbell"          "total_accel_dumbbell"
## [31] "gyros_dumbbell_x"      "gyros_dumbbell_y"      "gyros_dumbbell_z"
## [34] "accel_dumbbell_x"      "accel_dumbbell_y"      "accel_dumbbell_z"
## [37] "magnet_dumbbell_x"     "magnet_dumbbell_y"     "magnet_dumbbell_z"
## [40] "roll_forearm"          "pitch_forearm"         "yaw_forearm"
## [43] "total_accel_forearm"   "gyros_forearm_x"       "gyros_forearm_y"
## [46] "gyros_forearm_z"       "accel_forearm_x"       "accel_forearm_y"
## [49] "accel_forearm_z"       "magnet_forearm_x"      "magnet_forearm_y"
## [52] "magnet_forearm_z"      "classe"
```

We can split our dataset in 2 models data: training and testing.

```
idx <- createDataPartition(y=model_data$classe, p=0.75, list=FALSE )
training <- model_data[idx,]
testing <- model_data[-idx,]
```

Using parallel processing to build the model, we build 5 random forests with 150 trees each.

```
registerDoParallel()
x <- training[-ncol(training)]
y <- training$classe
rf <- foreach(ntree=rep(150, 6), .combine=randomForest::combine, .packages='randomForest') %dopar% {
randomForest(x, y, ntree=ntree)
}
```

Conclusions and Test Data Submit

This model is accurate as we can see in the consusion matrix. This test data was around 99% accurate and all of test cases are nearly to be correct.

```
pml_write_files = function(x){
  n = length(x)
  for(i in 1:n){
    filename = paste0("problem_id_",i,".txt")
    write.table(x[i],file=filename,quote=FALSE,row.names=FALSE,col.names=FALSE)
  }
}
x <- evaluation_data
x <- x[feature_set[feature_set!='classe']]
answers <- predict(rf, newdata=x)
answers
```

```
##  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
```

```
pml_write_files(answers)
```