Machine Learning Project

Joseph Fann

Executive Summary

In this report we will analyze the data to predict the type of exercise that a participant is performing.

Analysis

First we load the required libraries

```r
library(caret)
library(dplyr)
```

And then we load the data from source. The source data were saved locally during the writing of this report. The source data were also been analyzed to conclude that we need to set na.strings variable to include both word "Div/0" and "NA"

```r
url <- "https://d396qusza40orc.cloudfront.net/predmachlearn/pml-training.csv"
original <- read.table(url, header=TRUE, sep=",",na.strings = c("#DIV/0!","NA"))
validationurl <- "https://d396qusza40orc.cloudfront.net/predmachlearn/pml-testing.csv"
validation <- read.table(validationurl, header=TRUE, sep=",", na.strings = c("DIV/0!","NA"))
```

From original data, create training and testing data for cross validation.

```r
set.seed(10)
inTrain <- createDataPartition(y=original$classe,p=0.75, list=FALSE)
training <- original[inTrain,]
testing <- original[-inTrain,]
```

We will create a function to determine the amouunt of NAs in each column, and then reduce the original data to the data we need to work with.
From "temp" table we can see there are a lot of variables with a lot of NAs. We remove those variables and only focus on variable with complete data set. Variables 1 through 6 are identifying variables and we will remove as well.

```r
nana <- function(x){
        sum(is.na(x))
}
temp <- apply(training,2,nana)
table(temp)
```

```
## temp
##     0 14408 14409 14410 14412 14413 14414 14415 14416 14432 14465 14466
##    60    67     1     1     1     4     1     4     2     2     1     4
## 14467 14468 14469 14470 14718
##     2     1     1     2     6
```

```
temp <- temp<10000
training2 <- training[,temp]
training3 <- training2[,7:60]
```

Use random forest method to model the data This is computer expensive and will take some time.

```
model <- train(classe~., data=training3, method="rf")
```

We use the model to cross validate with the testing data.

```
pred <- predict(model, newdata=testing[,1:159])
confusionMatrix(pred, testing$classe)$overall['Accuracy']
```

```
##  Accuracy
## 0.9967374
```

We achieved the accuracy of 99%!?

```
validationpredict <- predict(model,newdata=validation[,1:159])
```

From the model, the prediction is:

```
validationpredict
```

```
##  [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
```