Practical Machine Learning project

Using devices such as Jawbone Up, Nike FuelBand, and Fitbit, we could collect data about personal activities, but they rarely quantify how well we do it. In this project, we use data from accelerometers on the belt, forearm, arm, and dumbell of 6 participants to predict their manners of their exercises.

Data processing

Load data

```
setwd("~/GIT_REPO/PMachineL")
trainData <- read.csv("~/GIT_REPO/PMachineL/pml-training.cvs", na.strings=c("NA","#DIV/0!")
)
testData <- read.csv("~/GIT_REPO/PMachineL/pml-testing.cvs", na.strings=c("NA","#DIV/0!"))
```

Clean data

We remove columns with all missing values and unnessary columns for predicting such as: user_name, raw_timestamp, window.

```
for(i in c(8:(ncol(trainData)-1))) {trainData[,i] = as.numeric(as.character(trainData[,i]))
}
ftrainData = trainData [, colSums(is.na(trainData))==0]
finalTrain = ftrainData[, c(8:ncol(ftrainData))]
for(i in c(8:(ncol(testData)-1))) {testData[,i] = as.numeric(as.character(testData[,i]))}
ftestData = testData [, colSums(is.na(testData))==0]
finalTest = ftestData[,8:ncol(ftestData)]
```

Build models

```
library(caret)
```

```
## Warning: package 'caret' was built under R version 3.1.3
```

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

```
## Warning: package 'ggplot2' was built under R version 3.1.2
```

```
library(randomForest)
```

```
## Warning: package 'randomForest' was built under R version 3.1.2
```

```
## randomForest 4.6-10
## Type rfNews() to see new features/changes/bug fixes.
```

```
inTrain = createDataPartition(y=finalTrain$classe, p=0.6, list=FALSE )
training = finalTrain[inTrain,]
testing = finalTrain[-inTrain,]
```

I chose random forests model because it is more accuracy than other methods such as boosting, trees. ### Random forests

```
set.seed (123)
modelFit = randomForest (classe~., data = training, ntree =150)
modelFit
```

```
##
## Call:
##  randomForest(formula = classe ~ ., data = training, ntree = 150)
##                Type of random forest: classification
##                      Number of trees: 150
## No. of variables tried at each split: 7
##
##         OOB estimate of  error rate: 0.74%
## Confusion matrix:
##      A    B    C    D    E  class.error
## A 3346    2    0    0    0 0.0005973716
## B   17 2253    9    0    0 0.0114085125
## C    0   17 2032    5    0 0.0107108082
## D    0    0   26 1902    2 0.0145077720
## E    0    0    3    6 2156 0.0041570439
```

As the result, the error rate is just less than 1%, and the accuracy is around 99%

Predictions

```
prediction <- as.character(predict(modelFit, finalTest))
prediction
```

```
##  [1] "B" "A" "B" "A" "A" "E" "D" "B" "A" "A" "B" "C" "B" "A" "E" "E" "A"
## [18] "B" "B" "B"
```