Weight Lifting Machine Learning Project

Synopsys

The goal of your project is to predict the manner in which they did the exercise. This is the "classe" variable in the training set. You may use any of the other variables to predict with. You should create a report describing how you built your model, how you used cross validation, what you think the expected out of sample error is, and why you made the choices you did. You will also use your prediction model to predict 20 different test cases.

Data

The data for this project come from this source: http://groupware.les.inf.puc-rio.br/har.

```
download.file("https://d396qusza40orc.cloudfront.net/predmachlearn/pml-training.csv", destfile="./ML1Pr
DataTrain<-read.csv("./ML1Pro/pml-training.csv")
download.file("https://d396qusza40orc.cloudfront.net/predmachlearn/pml-testing.csv", destfile="./ML1Pro
DataTest<-read.csv("./ML1Pro/pml-testing.csv")
```

PreProcessing

```
library(caret)
```

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

"' There is some personal in data set which is does not carry any useful info for prediction. We are going to eliminate that Colomns

```
DataTrain2 <- DataTrain[, -seq(from = 1, to = 8, by = 1)]
```

In this data a lot of columns which have most zeros.So we are removing Coulumns which NAs means equal to zero.

```
DataTrain3<-subset(DataTrain2, select=colMeans(is.na(DataTrain2)) == 0)
```

I would like to remove all zeros columns, then I tried using nearZeroVar(dataset).

```
removeColumns <-nearZeroVar(DataTrain3)
DataTrain3 <-DataTrain3[, -removeColumns]
```

As resul we get

```
dim(DataTrain3)
```

```
## [1] 19622    52
```

Get training/Test set

```
set.seed(1832)
asd <- createDataPartition(y = DataTrain3$classe, p = 0.1, list = F)
asd1<- DataTrain3[asd, ]
asd2<- DataTrain3[-asd, ]
```

Model training

Let's use Random forest classifier for our model

```
asd4 <- train(classe ~ ., data = asd1, method = "rf")
```

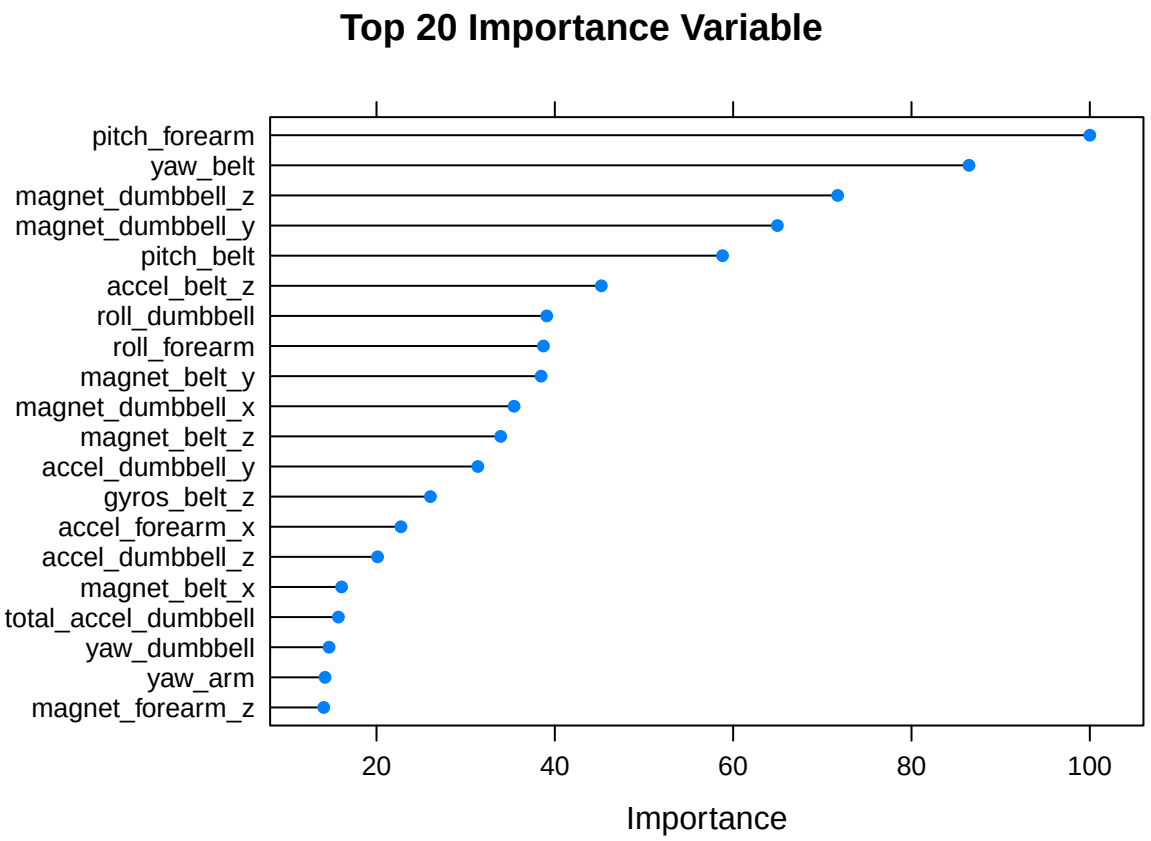
```
## Loading required package: randomForest

## Warning: package 'randomForest' was built under R version 3.1.1

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

let's choose 20 important variebles

```
IObj<-varImp(asd4)
plot(IObj, main = "Top 20 Importance Variable", top = 20)
```



Cross-Validation

Let‘s mesuare Cross-Validation error

```
asd3<-asd2[sample(1:nrow(asd2),500),]
error<-confusionMatrix(asd3$classe, predict(asd4,asd3))
error
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction   A   B   C   D   E
##          A 130   1   1   0   0
##          B   3  81   4   0   1
##          C   0   1  80   0   0
##          D   0   1   3  82   3
##          E   1   3   0   2 103
##
## Overall Statistics
##
##                Accuracy : 0.952
##                  95% CI : (0.929, 0.969)
##     No Information Rate : 0.268
##     P-Value [Acc > NIR] : <2e-16
##
##                   Kappa : 0.939
##  Mcnemar's Test P-Value : NA
##
## Statistics by Class:
##
##                      Class: A Class: B Class: C Class: D Class: E
## Sensitivity             0.970    0.931    0.909    0.976    0.963
## Specificity             0.995    0.981    0.998    0.983    0.985
## Pos Pred Value          0.985    0.910    0.988    0.921    0.945
## Neg Pred Value          0.989    0.985    0.981    0.995    0.990
## Prevalence              0.268    0.174    0.176    0.168    0.214
## Detection Rate          0.260    0.162    0.160    0.164    0.206
## Detection Prevalence    0.264    0.178    0.162    0.178    0.218
## Balanced Accuracy       0.982    0.956    0.953    0.980    0.974
```

Test-estimated out-of-sample error

```
SampleError <- 1 - error$overall[1];
names(SampleError) <- "Out of Sample Error"
SampleError
```

```
## Out of Sample Error
##               0.048
```

Test set

Now we can apply machine learning algorithm for the 20 test cases

```
DataTest$classe <- 1:nrow(DataTest);
predict(asd4,DataTest)
```

```
##  [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
```