Peer Graded ML

Chetan

10/21/2020

Week 4 PML project

```
library(caret)

## Warning: package 'caret' was built under R version 3.6.3

## Loading required package: lattice

## Loading required package: ggplot2

## Warning: package 'ggplot2' was built under R version 3.6.3

library(knitr)
```

```
library(data.table)

## Warning: package 'data.table' was built under R version 3.6.3

library(rpart.plot)

## Warning: package 'rpart.plot' was built under R version 3.6.3

## Loading required package: rpart

library(rpart)
```

```
library(gbm)

## Warning: package 'gbm' was built under R version 3.6.3

## Loaded gbm 2.1.8

library(ggplot2)
```

```
library(corrplot)

## Warning: package 'corrplot' was built under R version 3.6.3

## corrplot 0.84 loaded
```

cleaning and then exploring the data.

```
Url <- "https://d396qusza40orc.cloudfront.net/predmachlearn/pml-testing.csv"
tra  <- "https://d396qusza40orc.cloudfront.net/predmachlearn/pml-
training.csv"
```

```r
test <- read.csv(url(Url))
dtra <- read.csv(url(tra))
```

cleaning the input of the data

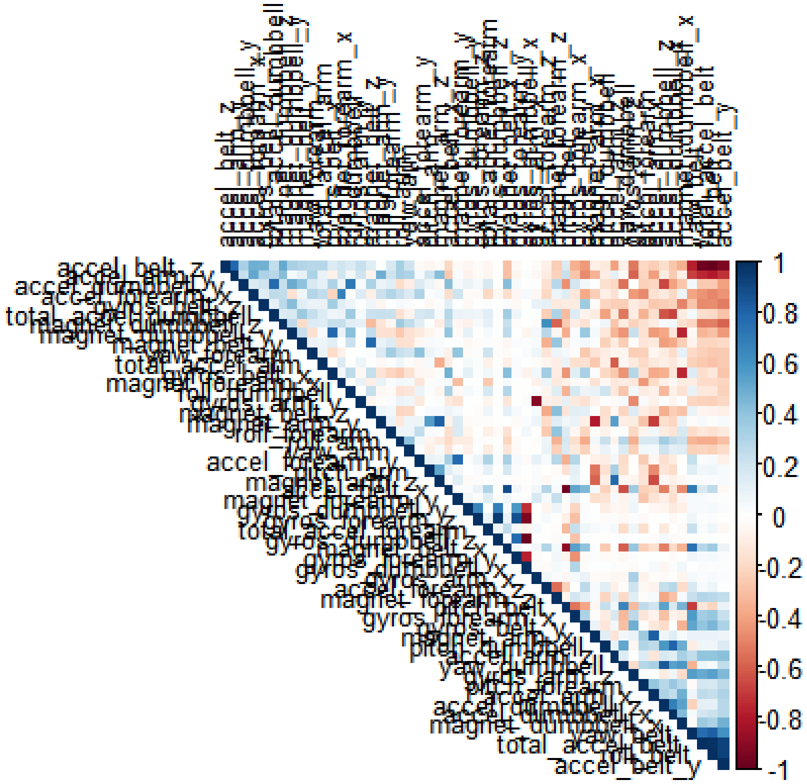
```r
train_data <- dtra[, colSums(is.na(dtra)) == 0]
testing_data <- test[, colSums(is.na(test)) == 0]
```

we will consider seventy percentage of the data for the training set and rest of the thirty percentage of the data for the testing data set

```r
train_data <- train_data[, -c(1:7)]
testing_data <- testing_data[, -c(1:7)]
dim(train_data)

## [1] 19622    86

set.seed(1234)
dtraining <- createDataPartition(dtra$classe, p = 0.7, list = FALSE)
train_data <- train_data[dtraining, ]
testing_data <- train_data[-dtraining, ]
dim(train_data)

## [1] 13737    86

dim(testing_data)

## [1] 4123   86
```

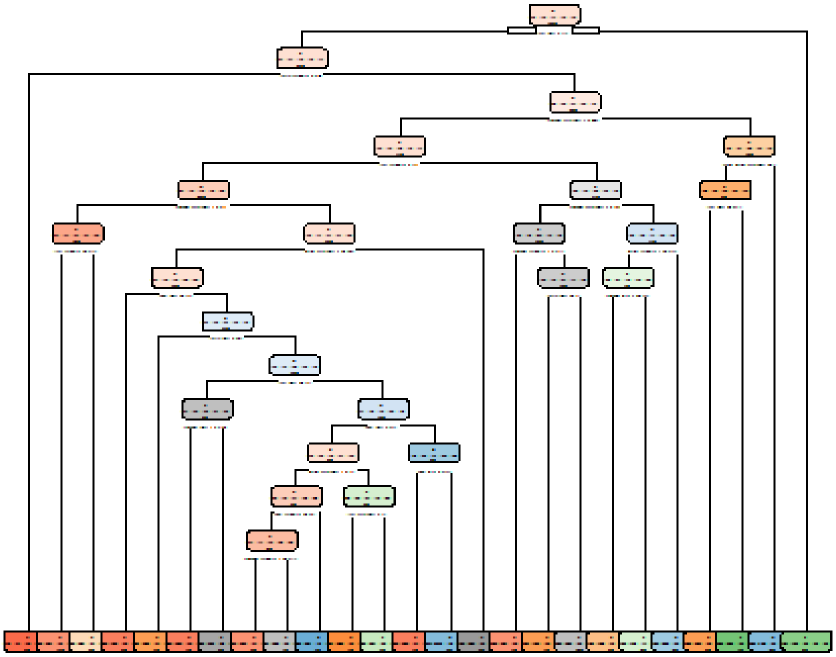
removing the variables that are non zero .

```r
nZero <- nearZeroVar(train_data)
train_data <- train_data[, -nZero]
testing_data <- testing_data[, -nZero]
dim(train_data)

## [1] 13737    53

dim(testing_data)

## [1] 4123   53

plot_cor <- cor(train_data[, -53])
corrplot(plot_cor, order = "FPC", method = "color", type = "upper", tl.cex =
0.8, tl.col = rgb(0, 0, 0))
```



Algorithms used: trees and random forests n

```
set.seed(20000)
tredec <- rpart(classe ~ ., data=train_data, method = "class")
rpart.plot(tredec)

## Warning: labs do not fit even at cex 0.15, there may be some overplotting
```

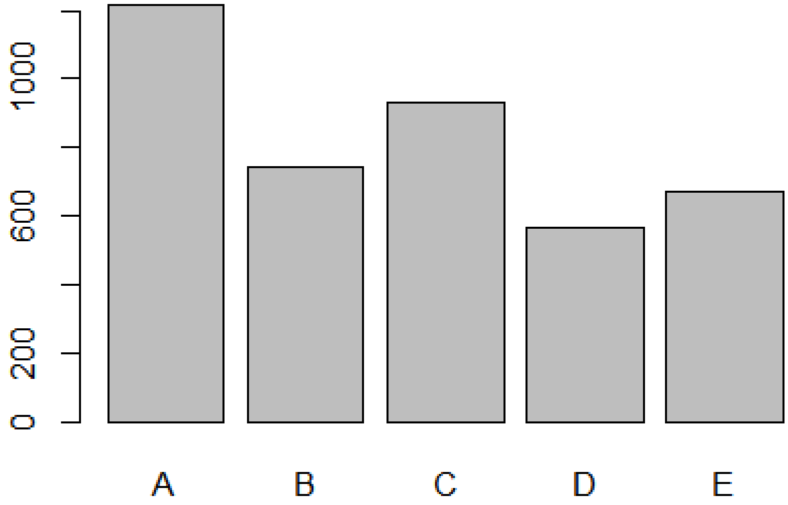


we will be validate the model

```
modelpre <- predict(tredec, testing_data, type = "class")
ab <- confusionMatrix(modelpre, testing_data$classe)
ab

## Confusion Matrix and Statistics
## 
##           Reference
## Prediction    A    B    C    D    E
##          A 1067  105    9   24    9
##          B   40  502   59   63   77
##          C   28   90  611  116   86
##          D   11   49   41  423   41
##          E   19   41   18   46  548
## 
## Overall Statistics
## 
##                Accuracy : 0.7642
##                  95% CI : (0.751, 0.7771)
##     No Information Rate : 0.2826
##     P-Value [Acc > NIR] : < 2.2e-16
## 
##                   Kappa : 0.7015
## 
##  Mcnemar's Test P-Value : < 2.2e-16
## 
```

```
## Statistics by Class:
##
##                      Class: A Class: B Class: C Class: D Class: E
## Sensitivity            0.9159   0.6379   0.8279   0.6295   0.7201
## Specificity            0.9503   0.9284   0.9055   0.9589   0.9631
## Pos Pred Value         0.8789   0.6775   0.6563   0.7487   0.8155
## Neg Pred Value         0.9663   0.9157   0.9602   0.9300   0.9383
## Prevalence             0.2826   0.1909   0.1790   0.1630   0.1846
## Detection Rate         0.2588   0.1218   0.1482   0.1026   0.1329
## Detection Prevalence   0.2944   0.1797   0.2258   0.1370   0.1630
## Balanced Accuracy      0.9331   0.7831   0.8667   0.7942   0.8416
```

```
plot(modelpre)
```



```
set.seed(10000)
ctr_gbm <- trainControl(method = "repeatedcv", number = 5, repeats = 1)
valid_gbm <- train(classe ~ .,data=train_data, method = "gbm", trControl =
ctr_gbm, verbose = FALSE)
valid_gbm$finalModel
```

```
## A gradient boosted model with multinomial loss function.
## 150 iterations were performed.
## There were 52 predictors of which 52 had non-zero influence.
```

So finally i was able to do this project with the help of the videos i watched on coursera. Basically we predicted how many did the exercise and the order in which they did it.
Thankyou