PML_course_assignment

```
library(caret)
```

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

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

```
library(rattle)
```

```
## Rattle: A free graphical interface for data science with R.
## Version 5.3.0 Copyright (c) 2006-2018 Togaware Pty Ltd.
## Type 'rattle()' to shake, rattle, and roll your data.
```

```
library(ggplot2)
library(dplyr)
```

```
##
## Attaching package: 'dplyr'
```

```
## The following objects are masked from 'package:stats':
##
##     filter, lag
```

```
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
library(stringr)
library(nnet)
```

```
## Warning: package 'nnet' was built under R version 3.6.2
```

```
## load training and testing data
input_train= "/Users/ram5ge/Desktop/R scripts/pml-training.csv"
input_test = "/Users/ram5ge/Desktop/R scripts/pml-testing.csv"
input <- read.csv(input_train)

inTrain<- createDataPartition(y = input$classe, p = 0.7, list = FALSE)
training <- input[inTrain, ]
validation <- input[-inTrain,]
test <- read.csv(input_test)
unique(training$classe)
```

```
## [1] A B C D E
## Levels: A B C D E
```

```
## outcome variable is of ordinal catergory
```

```
## filter features with missinng values >50% and near zero variance
new_train<-training[, which(colMeans(!is.na(training)) > 0.50)]
nzv_cols <- nearZeroVar(new_train)
nzv_cols <- nearZeroVar(new_train)
if(length(nzv_cols) > 0) new_train1 <- new_train[, -nzv_cols]
names(new_train1)
```

```
##  [1] "X"                    "user_name"            "raw_timestamp_part_1"
##  [4] "raw_timestamp_part_2" "cvtd_timestamp"       "num_window"
##  [7] "roll_belt"            "pitch_belt"           "yaw_belt"
## [10] "total_accel_belt"     "gyros_belt_x"         "gyros_belt_y"
## [13] "gyros_belt_z"         "accel_belt_x"         "accel_belt_y"
## [16] "accel_belt_z"         "magnet_belt_x"        "magnet_belt_y"
## [19] "magnet_belt_z"        "roll_arm"             "pitch_arm"
## [22] "yaw_arm"              "total_accel_arm"      "gyros_arm_x"
## [25] "gyros_arm_y"          "gyros_arm_z"          "accel_arm_x"
## [28] "accel_arm_y"          "accel_arm_z"          "magnet_arm_x"
## [31] "magnet_arm_y"         "magnet_arm_z"         "roll_dumbbell"
## [34] "pitch_dumbbell"       "yaw_dumbbell"         "total_accel_dumbbell"
## [37] "gyros_dumbbell_x"     "gyros_dumbbell_y"     "gyros_dumbbell_z"
## [40] "accel_dumbbell_x"     "accel_dumbbell_y"     "accel_dumbbell_z"
## [43] "magnet_dumbbell_x"    "magnet_dumbbell_y"    "magnet_dumbbell_z"
## [46] "roll_forearm"         "pitch_forearm"        "yaw_forearm"
## [49] "total_accel_forearm"  "gyros_forearm_x"      "gyros_forearm_y"
## [52] "gyros_forearm_z"      "accel_forearm_x"      "accel_forearm_y"
## [55] "accel_forearm_z"      "magnet_forearm_x"     "magnet_forearm_y"
## [58] "magnet_forearm_z"     "classe"
```

```
##Since the dubmbell lifts are the crucial variables that classifies qualitative perform
ances of the subjects, variable involving dumbell lifts are used for prediction
new_train2<-new_train1[ , grepl( "dumbbell" , names( new_train1) ) ]
new_train2$classe<- new_train1$classe
new_train2$user_name<- new_train1$user_name
names(new_train2)
```

```
##  [1] "roll_dumbbell"        "pitch_dumbbell"       "yaw_dumbbell"
##  [4] "total_accel_dumbbell" "gyros_dumbbell_x"     "gyros_dumbbell_y"
##  [7] "gyros_dumbbell_z"     "accel_dumbbell_x"     "accel_dumbbell_y"
## [10] "accel_dumbbell_z"     "magnet_dumbbell_x"    "magnet_dumbbell_y"
## [13] "magnet_dumbbell_z"    "classe"               "user_name"
```

```
## final list of predictors used fr model building
```

```
##Keep only the features present in the filtered training dataset
cols_to_keep <- intersect(colnames(new_train2), colnames(test))
cols_to_keep
```

```
##  [1] "roll_dumbbell"        "pitch_dumbbell"       "yaw_dumbbell"
##  [4] "total_accel_dumbbell" "gyros_dumbbell_x"     "gyros_dumbbell_y"
##  [7] "gyros_dumbbell_z"     "accel_dumbbell_x"     "accel_dumbbell_y"
## [10] "accel_dumbbell_z"     "magnet_dumbbell_x"    "magnet_dumbbell_y"
## [13] "magnet_dumbbell_z"    "user_name"
```

```
new_validation<- validation[, cols_to_keep, drop=FALSE]
new_validation$classe<- validation$classe

new_test <- test[, cols_to_keep, drop=FALSE]
new_test$problem_id <- test$problem_id
```

```
# since the outcome varibale is a class variable or ordinal variable, this is attempted
 as a classification prediction problem
##Hence methods like classification tree, random forest, KNN are considered as suitable
 prediction methods.
# First attempted with rpart classification is used to build model using training set
fitControl <- trainControl(method = "cv",number = 5)
set.seed(8356)

modFit<-train(classe~., data = new_train2, method = "rpart", trControl =fitControl)
## after building model.It was tested using validation set as intermediate cross validat
ion
pred1<-predict(modFit, new_validation)
summary(pred1)
```

```
##    A    B    C    D    E
## 4291 1037  557    0    0
```

```
confusionMatrix(pred1, new_validation$classe)
```

```
## Confusion Matrix and Statistics
## 
##           Reference
## Prediction    A    B    C    D    E
##          A 1614  587  745  749  596
##          B   43  439   41  191  323
##          C   17  113  240   24  163
##          D    0    0    0    0    0
##          E    0    0    0    0    0
## 
## Overall Statistics
## 
##                Accuracy : 0.3896
##                  95% CI : (0.3771, 0.4022)
##     No Information Rate : 0.2845
##     P-Value [Acc > NIR] : < 2.2e-16
## 
##                   Kappa : 0.1774
## 
##  Mcnemar's Test P-Value : NA
## 
## Statistics by Class:
## 
##                      Class: A Class: B Class: C Class: D Class: E
## Sensitivity            0.9642   0.3854  0.23392   0.0000   0.0000
## Specificity            0.3643   0.8740  0.93476   1.0000   1.0000
## Pos Pred Value         0.3761   0.4233  0.43088      NaN      NaN
## Neg Pred Value         0.9624   0.8556  0.85248   0.8362   0.8161
## Prevalence             0.2845   0.1935  0.17434   0.1638   0.1839
## Detection Rate         0.2743   0.0746  0.04078   0.0000   0.0000
## Detection Prevalence   0.7291   0.1762  0.09465   0.0000   0.0000
## Balanced Accuracy      0.6642   0.6297  0.58434   0.5000   0.5000
```

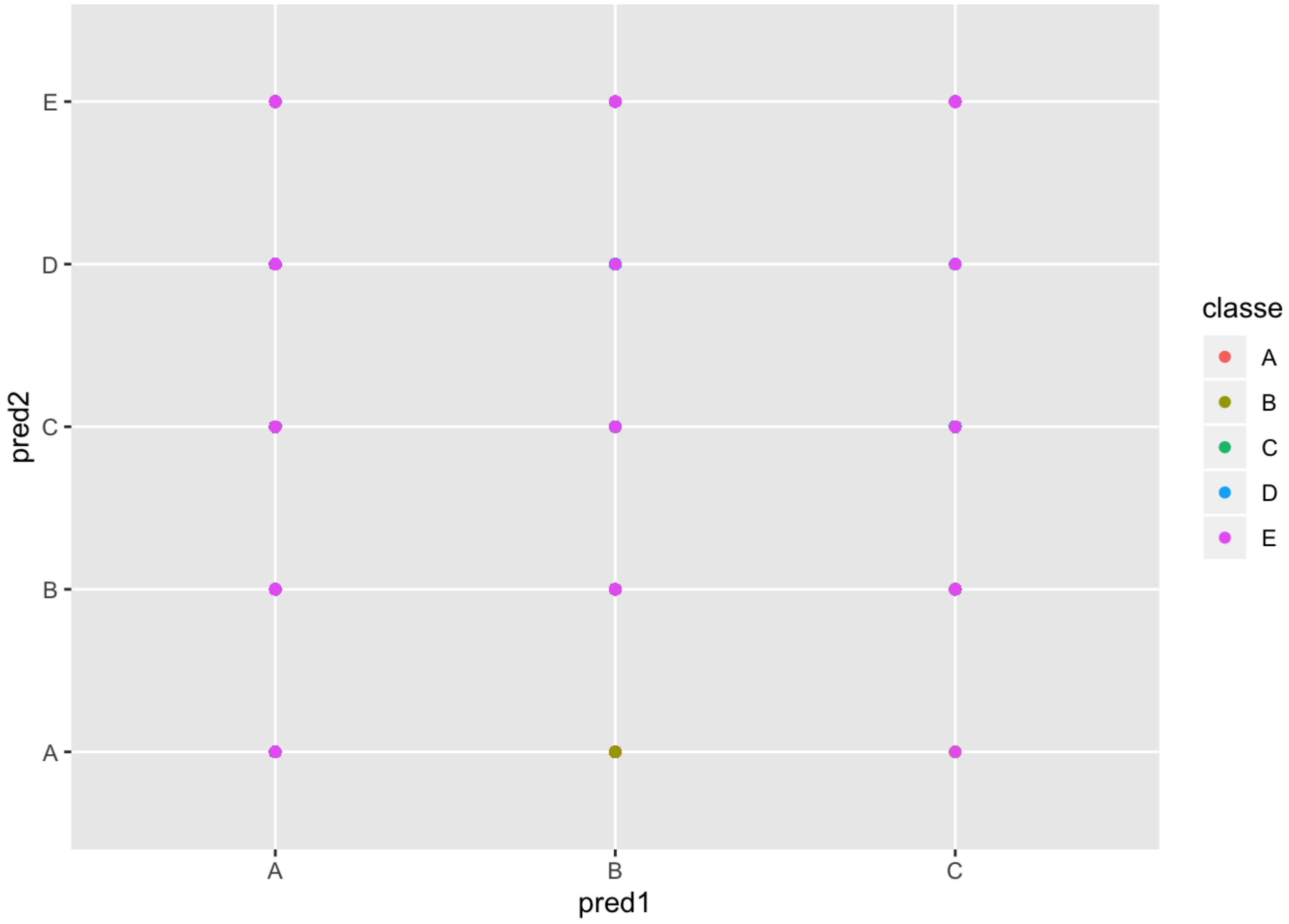
```
### The accuracy of this method is not great, 0.3. with significant p-value
# Secondly the same data is classified using K-nearest neighbor method is used to build another model for comparison and the classes are compared
modFit2 <- train(classe~., data=new_train2, method="knn", trControl=fitControl)
pred2 <- predict(modFit2, new_validation)
pred2
```

```
##    [1] A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A
##   [38] A A A A A A A A A A A A A A A A A A A B A B A E A A A A A A A A A A A A A
##   [75] A A A A A A A A A A A A A A A A A A A A A A A A A C C A A A A A A A A A A
##  [112] B A D D D E B A C C A A A D A B C B A A A A A A A C A A A A A B B A D A C
##  [149] A A C C C A D D D A A A A A A A B A A C A D D B A A A A A A B B C D A A A
##  [186] A A A A A A A B A A A A A A A A A A A A A B A A A A A A A A A A A A A A C
##  [223] A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A
##  [260] A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A
##  [297] A A A A A A A A A A A A A A A A A A C A A C C A D A A A A D A A A A A A A
##  [334] A A A A A A C D A C A A D D D D D D A A C D A D A A A A A B C A A C A D D
##  [371] D D A A A E A A A A A A A A B A A A A A A A D A A A A A A A A A A A C A A
##  [408] A A A A A A A A A A A A A A A D A A A A A A A A A A A A A A A A A A C A A
##  [445] A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A B A
##  [482] A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A
##  [519] A A A A A A D D A A A A A A A A A A A A A A A A A A A A A A A A A A A A A
##  [556] A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A
##  [593] A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A
##  [630] A A A A A A A A A B B B A B A B B A A B A B A A B B B A A A A A A A A A A
##  [667] A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A
##  [704] A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A
##  [741] A A A A A A A A A A A A A A A A A A A A A D A A D D A A A A A A A A A A A
##  [778] A A A A A A A D D A A D A A A A A A A A A A A A A A A A A A A A A A A A A
##  [815] A A A A A A D D D A A A D A A A A A A A A A A A A A A A A A A A A A A A A
##  [852] A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A
##  [889] A A A A A A A A A A A A A A A A A A A A A A D A A A A A A A A A A A A A A
##  [926] A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A
##  [963] A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A
## [1000] A A A A A A A A A A A A A A A A A A A A A A A A C A A A A A A A A A A A A
## [1037] A A A A A A A A A A A E A E E A A A A A A A A A A A A A A A A A A A A A A
## [1074] A A A A A A B A A A B A A B A A A A A A A A A A A A A A A A A A A A A A A
## [1111] A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A
## [1148] A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A
## [1185] A A A A A A A A A A A A A A A A A A A A A A A A A A A A B A A A A A A A A
## [1222] A A A A A A A A A C A A A A A A A A A A A A A A D A A A A A A A C C B C A
## [1259] A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A E A A A A
## [1296] A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A
## [1333] A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A
## [1370] A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A
## [1407] A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A B A A A A
## [1444] A A A B A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A
## [1481] A A A C A A A A A A A A A A A A A A A A B A A A A A A A A A A A A A A A A
## [1518] A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A
## [1555] A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A
## [1592] A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A A
## [1629] A A A C A A A A A A A B C A A A D A D A A A D A A A A A B A D D B A A A A
## [1666] A A A A A A A A A A A A B A A B A A A A A D B B B D B B B B B B B B B B B
## [1703] B B B B B D A B C B B B B B B B C B B B C B B B B B B B B B B B B D C C B
## [1740] C D B B B C B B A A D B B B B B B B B B B B E B B D D D D B B B B B B B B
## [1777] C B B B B B B D B E E C C C B D B B B B B B B B B C C B B B B B B B B B B
## [1814] B B B B B B B B B B B B B B A B B B B A B B A B B B B B B B B B B A A A B
## [1851] D B B D B B B B B B B B B B B B B B B B B A A B B B B B B B B B B B B B B
## [1888] B B B B B A B B B B B B B B B B B B B B B B B B B B B B B B C B A A B A B
## [1925] B B B B B B B B B A B B B B B B B B B B B B B A A A A B A D B B B B B B B
## [1962] B B B B B B B B B B B B B B B A A A B B B B A A D C A B B B A A B E E B C
## [1999] C B C B B B B B B B B B B B B B B B D A B B B B B B A A B E B B B E B B B
## [2036] E B B B B B B B B B B B B B E B B B B B B B B B B B B B B B B B B B B B B
## [2073] B B B B A B B B B B B A B A B B B B B B B B B B A A B B B B B B B B B B B
## [2110] B B B B B B D C C B B B B B B B B B B B B D E E E B B B B B B B B B B B B
## [2147] B B E B B D B D B B B B B B B B D C B E D B B B B B C B B D D B B E B E B
## [2184] B B B B B B C A B B E B C C B B A B E B B B B B B B B B B B B B B B B B B
## [2221] C B B E B B B B B B B D B B B B C B B C C E B B E B B B B B B B B D B B B
## [2258] C E B B B B B B B B B B B B B E B B B B B B B B B B B B B B B B B B B B B
## [2295] B C C C B B B B B B B B B B B B B B B B B B B B B B B C C B B C C B B B B
## [2332] B E B E B B B B B B B B C D E B B B B B B B B B B B B B B B B B B B A A B
## [2369] A A A A C A A A A B B A B B B B B B B B B B B B B B A A A A A A B B B B A
## [2406] A A A A A B B B B B B B B B B B B B B B B B B B B B B B B B B B B B B B B
## [2443] B B B B B B B B B B B B B B B B B B B B B B B B B B B B B B B B B B B B B
## [2480] B B B B B B B B B B B B B B B B B B B B B B B B B B B B B B B B B B B B B
## [2517] B B B B B B B B B B B B B B B B B B B B B B B B E B B B B B B B B B B B B
## [2554] B B B B B D D B D E B B B D B B B C C C C B B B B B B B B B B B B B B B A
## [2591] A D B B B B A C C C A B B B B B B B B B B B B B B B D C C C B B B B B B B
## [2628] B B B B B B B B B B B B B B B B B B B B B B A B B B B B D D D B B B B B B
## [2665] D D E C C B B B B B B B B B B C E C B A B B B B B B B B B B B B B B B B B
## [2702] B B B B B B B B B B B B B B B B B B B B B A B B B B B B A C B B B B B B B
## [2739] B B B B B D B B C B B B B B E E B E B B B B B B B B B B A D A A A B B B E
## [2776] B B B B B B B B B B B C C E B B B B D B B B B B B B B B B B E C E B B B B
## [2813] B B D B C B B B D B B B B C C D D B C C E E C C C C C C B C E E B B E C C
## [2850] C D C C A B C C C C C C C A C D C C D A C C C C C C C C C B A C D D C C D
## [2887] C D C C C C C C C C D C B D C C C C D C D C D D C C C C C C C D D D E C D
## [2924] C C D B B C B C C C C C C C C B C C C C C C C C C C C C C C C C C C C C C
## [2961] C C C C C C C C C C C C C C C C C C C C C C C C C C C C C C C C C C C C C
## [2998] C C C C C C C C C C C C C C C C C C C C C C C C D C C C C C C C C E D C C
## [3035] C C C C C C C D C C C D C C B C C B B C C C C C C C C C C C C C C C E C E
## [3072] E E C E E C C C C C C C C C C C C B E C C C C C C C C D C C C C C C C C C
## [3109] C C C C C C C C C B C C C C C C C C C C C C C C C C C C C C C C C C C C C
## [3146] C C C C C C C C C C C C C C C C C C C C C C C C C C C C C C C B C C C C C
## [3183] C C C C C C C C C C B B B B D D D C C C C D C C C C C C C C C C C C C C C
## [3220] E C E C C C E E E C D C C C C C C E C C E C C C C C C C A C A C C B E C C
## [3257] E C D C D D C C C C D D D A C C C C B B A A A A C C C C C C A B B B E C C
## [3294] C C C C E C E C B A C C C E C C C D C A A C C C C C C C C C C C C C C C C
## [3331] C C C C C C C C C C C C B C C C C C C C C E E E E B C C C C C C C C D B C
## [3368] C C B B B B B B B E C B B B C C C C C C C C C B C D C C C C C C C C C B D
## [3405] D C D C C B C C C C B C C C D E D D C C C C C C B D B C D C C C C C D C D
## [3442] C C C C C C C C C C D D C C C C C C C C C C C C C B C C C B C B A D A D C
## [3479] C E C A D E D C C A A C C C C C C C C C C C C C C C C B E C C C B B C D C
## [3516] C C C C C C C A C C C B D C C C C C C C C C C C A C B D C C A C C C C C C
## [3553] C C C C A C C C C A C C D D B D C C C B E A E C C C C C B B B C C B B B B
## [3590] C B C C B C C B C C C C C C C C C C C C C C C C C C C C C C C C C C C C C
## [3627] C C C C C C C C C C C C C C C C C C C C C C C C C C C C C C C C C C C C C
## [3664] C C C C C C C C C C C C C C C C C C C C C C C C C C C C C C C C C C C C C
## [3701] C C C C C C C C C C C C C C C C C C C C C C C C C C C C C C C C C C C C C
## [3738] C C C C C C C C C C C C C C C C C C C C C C C C C C C C C C C C C C C C C
## [3775] D C C C C E C C C C C C C C D D C C C C C C C C D D C C A C D D C B C D C
## [3812] A C A D C C D C C C C C C A A A C C C C C C C C C C C C D D D D D D D D D
## [3849] D D D D D D D D D D D D D D D A D D D A D A D D A D D D D D D D C A C D C
## [3886] C D C D C C C C C D D C C D D C C A C C D C D D C D A D C C C C C A C A A
## [3923] B D D D C D D D A D A A A D A D A B B A A D A C A D D D B D A D D D D D D
## [3960] D D D D D D D D D D B D B D D D D D D D B D D B D D D D D D D D D D D D D
## [3997] D D E D B D D D D D E D D D D D D D D D D D D D D D D D D D D D D D D D D
## [4034] D D D D D D D D D D D D D D D D D D D D D D D B D B B D D D D D D D D D D
## [4071] D D D D D D D D D D D D D D D D D D D D D D D D D E E E E D C C C C C C D
## [4108] D D D D D D D D D D D D D D D D D D C B B D D D D D D D D D E D D D E E B
## [4145] B D D D D D D D D D D D D D B B D E D D D D D D D D D D D D D D D D D D D
## [4182] D D D D D D D D D D D D D D D D D D D D D D D D D D D D D D D D D D D E D
## [4219] D D C C C C D D D D D D D D D D D D D D A A A D D D D D D D A D D D D D A
## [4256] D D A A A A D D D D A A D A D A D A D A D D D D D A A D D D D D D D D D A
## [4293] D D D D D D D D D D D D D D D D D D C D D D D C E E C D D D D D D C D D E
## [4330] D D D D C D D D D C C D D D D D D D D D D D D D D D D D D D D D D C D C C
## [4367] C E D D B D D D D C E E E D D D D D D D D D E E D D D D D D D D D D D D D
## [4404] D D D D D D D D D D D D D D D D D D D D A C C C C C C D D C C D D C C B C
## [4441] C C C C D D A D D A A D D D D D D D D D E D D D D D D D D D E D D D D D D
## [4478] D D D D D D D D D D D D D D D D D D E D D D D D D D D A D D D D D D D E E
## [4515] E D D D E D D D D D C D D E D D B E D D D D E E D D E D D D A D D D D E D
## [4552] D D D D A D A A D D D E E D D D D C A C C D D B C D D A D C D C C C C D D
## [4589] D D D D D D D D D D D D D D D D D D D D D D D D D D D D D D D D D D D D D
## [4626] D D D D D D D D D D D D D D D E B B D D D D D D D D D D D D D D D D D D D
## [4663] D D D D D D D D D D D D D D D D D D D D D D D D D D D D D D C D D D D D D
## [4700] D D D D D D D D D D D D D D D D C D D D D D D E E D D D D D D D E E D D D
## [4737] D D D D D D D D D D D D D D D D D D D D D D D D D E E D D D E E D D D D D
## [4774] C C D A A C A C C C C C C D D E D D D A D D A D D D D D D D E E E E E E E
## [4811] E E E D D E E E E E E E B B B A C B E C B E E E B B B B B B C B D C E E C
## [4848] B E E C D B E E E E E E E E D B A A C C C C E E E E E E E E D B B B B B E
## [4885] E E E C E E E E E E E E E E E E E E E E E D B B C E E E E E E E E E B B E
## [4922] B E E E B E E E E E E B C B C E E C E E E E C B E E B C E B E E E E E E E
## [4959] E E E E E D A A B E E E E E E E D D D E E D E E E E E E E E E E B E E E E
## [4996] E E E E E E E E D D E D E E E E A E E D E E E D E D E E A D E E D D E D B
## [5033] D D D D E C E E C E C E E E E E E E E E E E E E E E E E E E E E E E E E E
## [5070] E E E E E E E E E E E E E E E E E E E E E E E E E E E E E E E E E E E E E
## [5107] E E E E E E E E E E E E E E E E E E E E E E E E E E E E E E E E E E E E E
## [5144] E E E E E B E E E E E C E E E E E E E E E E E E E E E E E E E E E E E E E
## [5181] E E E E E E E E E E E E E E E E E E E E E E E E E E E E E E D E E E E E E
## [5218] E E C E E E E E E E E E E E E E E E E E E D D E E E E E E E E E E E E E E
## [5255] E E E E E E E E E E E E E E E E E E E E E E E E E E E E E E E E E E E E E
## [5292] E E E E E E E E E E E E E E E E E E E E E E E E E E E E E E E E E E E E E
## [5329] E E E E E E E E E E E E E E E E E E E E E E E E E E E E E E E E E E D D D
## [5366] E E E E E D E C E C E E E E E E E E E E E E E E E E E E E E E E D C C C C
## [5403] E E E E E E E B E E E E E E E E E E E E E C A A C E E D E E E E E D B E E
## [5440] A E E A C E E C E B B B C C E E E E E E E C E E E D E E C C C E E D E E E
## [5477] E E E E A E E E E E E E B E E E E D D D E E D E E D D D D E D E D D D C E
## [5514] E B E E E E D E D D D E E E E C C D E D D D E E E E D D D D E D D E D D D
## [5551] D E E E E E E E E E E E D D A E E E A E E E E E E E E A A A E C E E E E E
## [5588] E E B B E D C E E E E E E E E E C E E E E E C E E B B B B D E B E E E C E
## [5625] B E E B B D D B B E E B E E B B B B B B E A A A A A A A A E E E E E E E E
## [5662] E E E E E E C E E E E E E E A C E E E E E E E E E E E E E E E E E E E B B
## [5699] B E E E E E E E C E C E E E E E E E C E E E E E C E E E E E E E E E E E E
## [5736] E E E E E E E E E E D B C E E E E E C C E E B B D E D D D B E E E E E E E
## [5773] E E E E E E E E E E E E E E E E E E E E E E E E E E E E E E A E C A E B E
## [5810] A A A E C C E C C E E E E E E E E E C E E E D C E C E E E B B B E E E E E
## [5847] E E E E E E E E E E B C B B E D B E E E E E D E E C B B C E E E E E E B A
## [5884] D E
## Levels: A B C D E
```

```
table(pred2, new_validation$classe)
```

```
##
## pred2    A    B    C    D    E
##     A 1550   84   32   55   32
##     B   39  918   77   24   79
##     C   31   58  805   85   66
##     D   47   42   73  755   83
##     E    7   37   39   45  822
```

```
##Here the accuracy of the prediction was greater than 30%. So it was used for predictio
n in the test data
qplot(pred1, pred2, color = classe, data = new_validation)
```



```
##Note:application of random forest method would have been a appropriate method as well.
But it was computationally expensive and causes the system to break.
```

```
pred_test <- predict(modFit2, new_test)
pred_test
```

```
##  [1] A A B A A B D B A A A C B A B E A B B B
## Levels: A B C D E
```

R Markdown

This is an R Markdown document. Markdown is a simple formatting syntax for authoring HTML, PDF, and MS Word documents. For more details on using R Markdown see http://rmarkdown.rstudio.com (http://rmarkdown.rstudio.com).

When you click the **Knit** button a document will be generated that includes both content as well as the output of any embedded R code chunks within the document. You can embed an R code chunk like this:

Note that the `echo = FALSE` parameter was added to the code chunk to prevent printing of the R code that generated the plot.