FER Exploratory Data Analysis

Introduction

This is an R Markdown document contains the exploratory data analysis of the Facial Emotion Recognition dataset.

Setting the working directory of the project

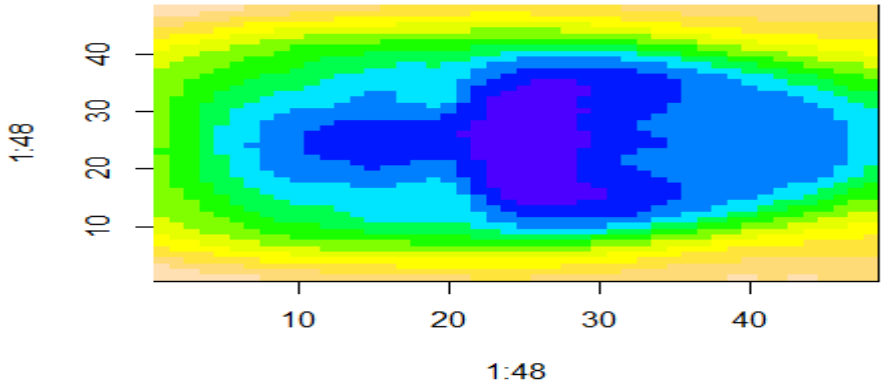
```
setwd("C:/Users/Nishna/Documents/F21DL_CW3")
```

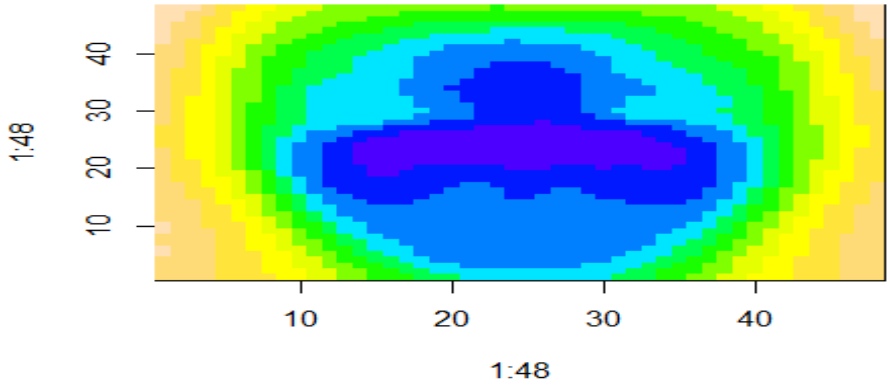
Visualize the distribution of classes in the original dataset

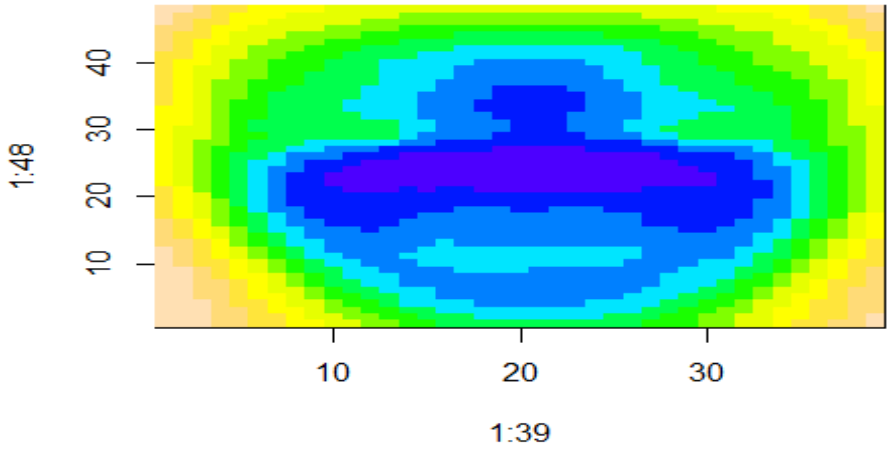
The distribution is not even and the value ranges from 436 to 7215 in the training set and 111 to 1774 in test set

Nishna

Visualizing the distribution of values in each pixel

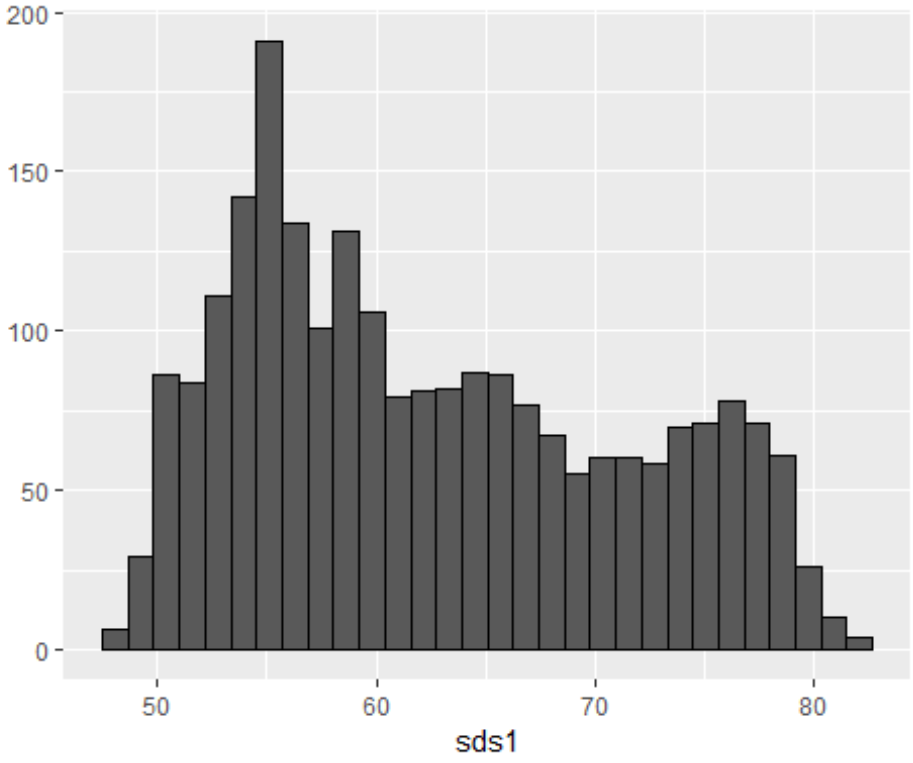


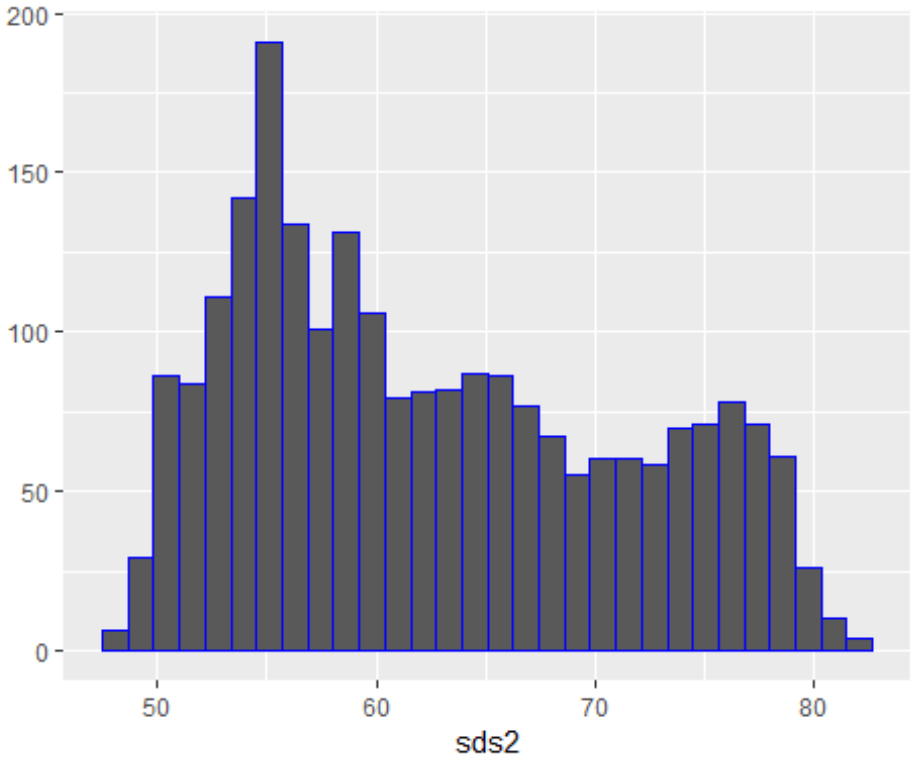




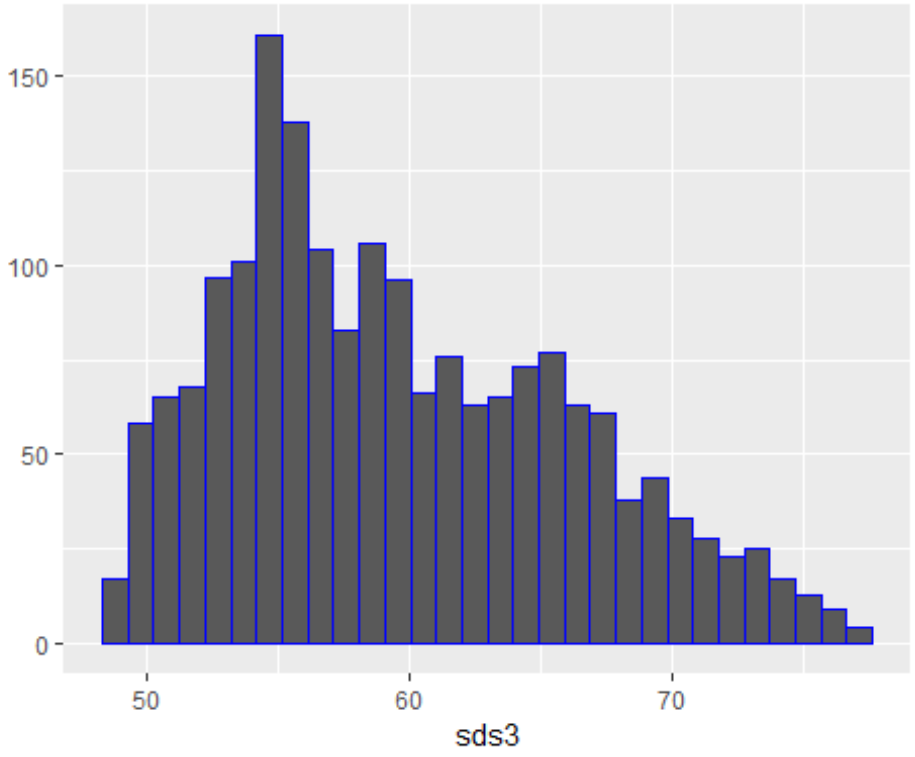
Visualizing the distribution of pixels in rotated images

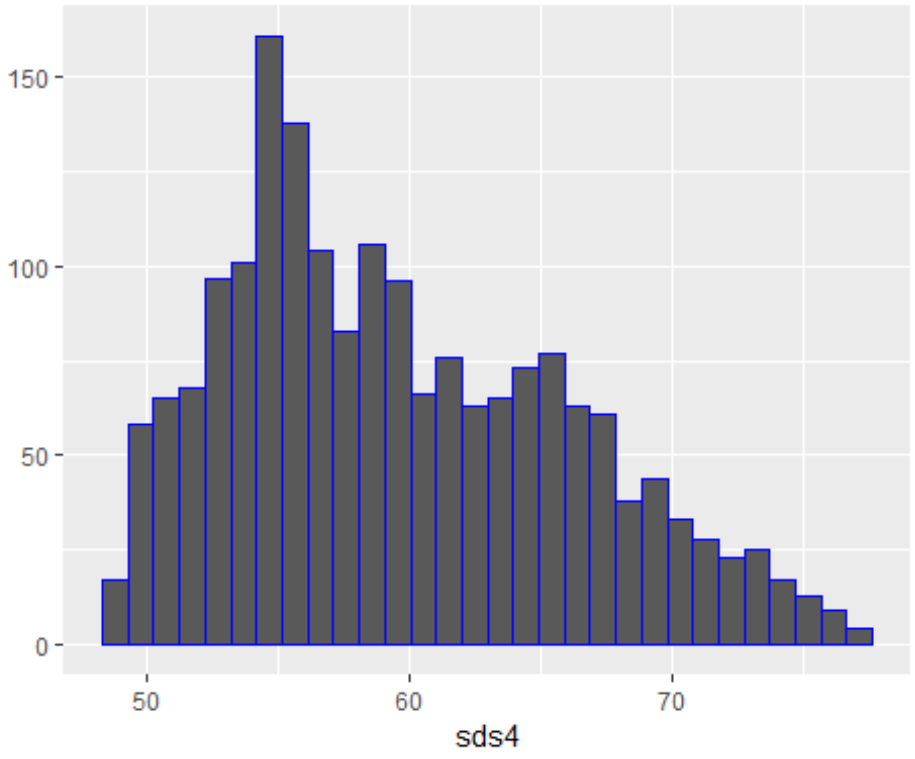
Nishna



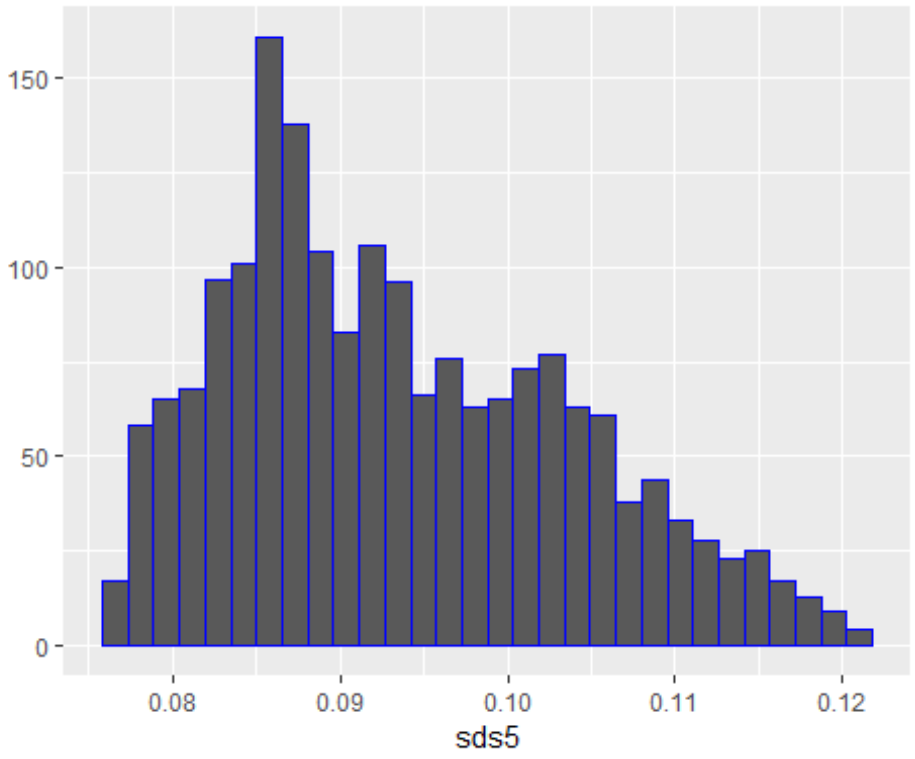


Nishna



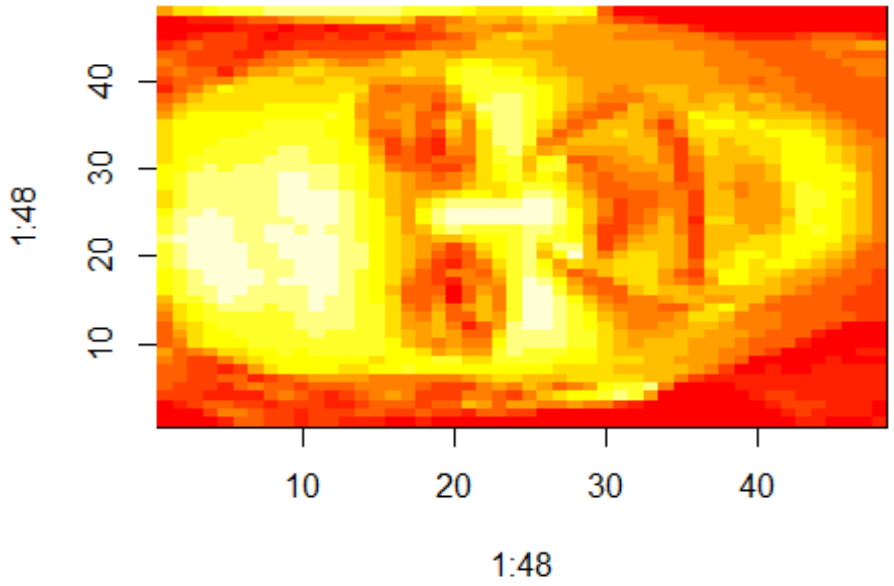


Nishna

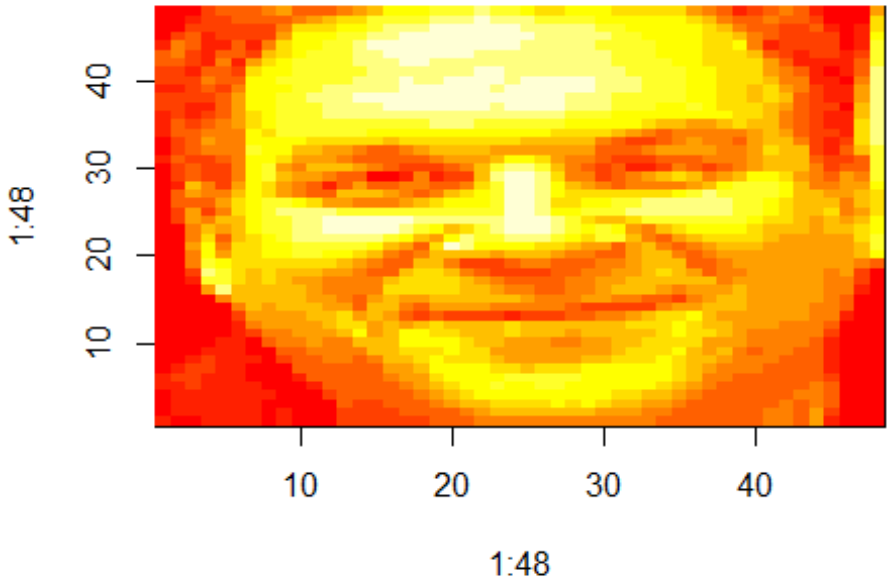


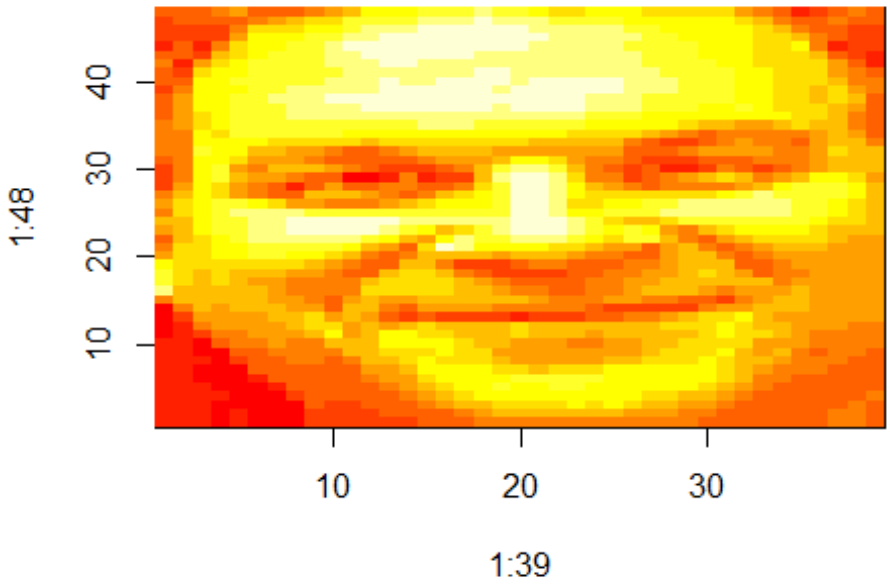
Visualizing transformations - Sample image

(i) original (ii)rotation (iii)cropped



Nishna





Nishna

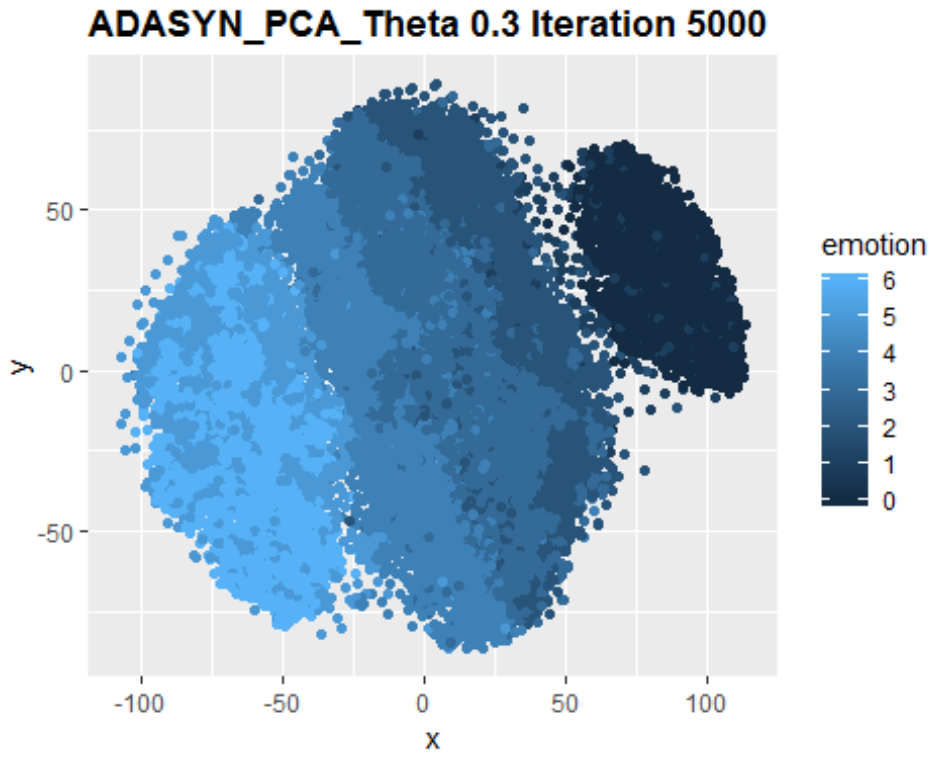
PCA -train data

From the PCA summary its clear that the first 25 principal components holds more than 80% of the data.

```
load(file = "C:/Users/Nishna/Documents/F21DL_CW3/Datasets/RDA/pca_train_original.rda")
library(dplyr)
summary(pca_train_original)$importance[,c(2,5,10,15,20,25,50)] %>% knitr::kable()
```

	PC2	PC5	PC10	PC15	PC20	PC25	PC50
Standard deviation	1.88251	1.028855	0.528177 2	0.384529 5	0.32008 3	0.282354 4	0.174971 3
Proportion of Variance	0.17644	0.052700	0.013890 0	0.007360 0	0.00510 0	0.003970 0	0.001520 0
Cumulative Proportion	0.42813	0.622170	0.712710 0	0.759380 0	0.78946 0	0.810980 0	0.871240 0

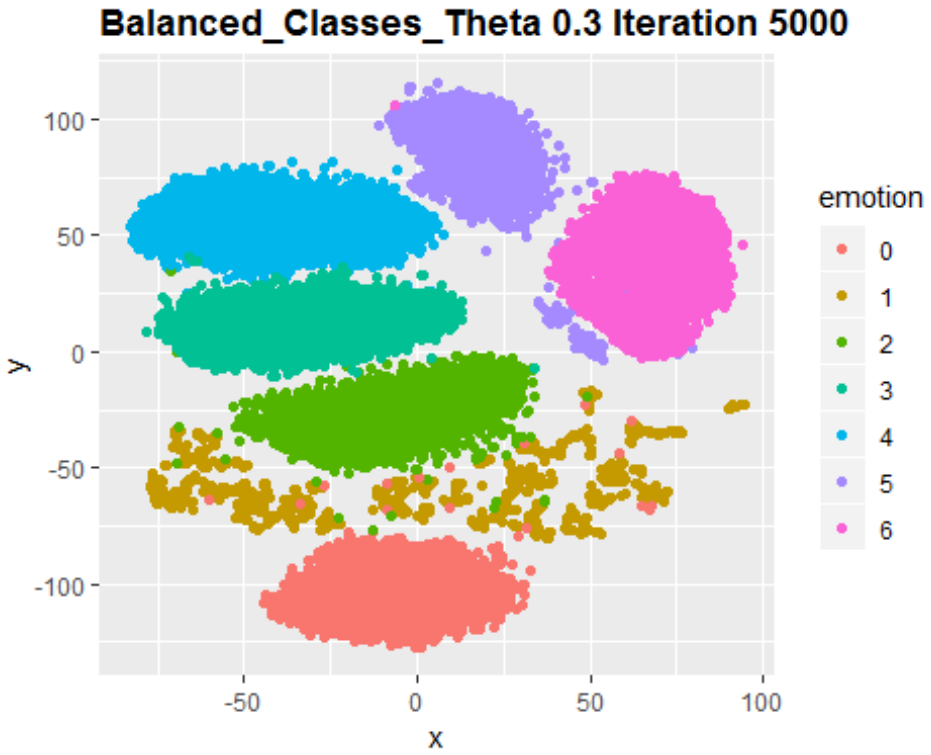
tSNE - orig-train



Nishna

tSNE - balanced train

Now its easier for the ML algorithms to classify the images




```
                DT CLASSIFIERS
```

Benchmark classifier - Original dataset with no preprocessing

```
##       C M  Accuracy     Kappa  AccuracySD     KappaSD
## 1 0.010 1 0.3061417 0.1197156 0.007121018 0.009542731
## 2 0.010 2 0.3050964 0.1182839 0.006145504 0.008312891
## 3 0.010 3 0.3027281 0.1149507 0.006011270 0.007742939
## 4 0.255 1 0.3378382 0.1898444 0.009979800 0.014482260
## 5 0.255 2 0.3230346 0.1713132 0.007876186 0.011798731
## 6 0.255 3 0.3162424 0.1620172 0.007102503 0.010489010
## 7 0.500 1 0.3379430 0.1906786 0.010650180 0.015211839
## 8 0.500 2 0.3226868 0.1715594 0.007734124 0.011075260
## 9 0.500 3 0.3149190 0.1615090 0.007561889 0.010566149
```

Classifier II - After Data Cleaning, Dimensionality Reduction & Stratification

```
library(caret)
dtree_2$results
```

```
##          cp  Accuracy     Kappa   AccuracySD      KappaSD
## 1 0.1639936 0.6534853 0.5900293 0.0560036122 0.0666284762
```

Nishna

```
## 2 0.1795008 0.4802465 0.3826450 0.0001799339 0.0001880994
## 3 0.1916091 0.1650778 0.0000000 0.0001772847 0.0000000000

#Prediction
load(file = "C:/Users/Nishna/Documents/F21DL_CW3/Datasets/RDA/pca_validation_
original.rda")
y_hat <- predict(dtree_2, pca_validation, type = "raw")

#Results
load("C:/Users/Nishna/Documents/F21DL_CW3/Datasets/RDA/orig_validation.rda")
confusionMatrix(y_hat, factor(orig_validation[,1]))
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction    0    1    2    3    4    5    6
##          0    0    0    0    0    0    0    0
##          1 1013  110    0    0    0    0    0
##          2    0    0    0    0    0    0    0
##          3    0    0 1009 1804    0    0    0
##          4    0    0    0    0 1222  779    0
##          5    0    0    0    0    0    0    0
##          6    0    0    0    0    0    0 1242
##
## Overall Statistics
##
##                Accuracy : 0.6098
##                  95% CI : (0.5984, 0.6211)
##     No Information Rate : 0.2513
##     P-Value [Acc > NIR] : < 2.2e-16
##
##                   Kappa : 0.5252
##  Mcnemar's Test P-Value : NA
##
## Statistics by Class:
##
##                      Class: 0 Class: 1 Class: 2 Class: 3 Class: 4 Class: 5
## Sensitivity            0.0000  1.00000   0.0000   1.0000   1.0000   0.0000
## Specificity            1.0000  0.85670   1.0000   0.8123   0.8692   1.0000
## Pos Pred Value            NaN  0.09795      NaN   0.6413   0.6107      NaN
## Neg Pred Value         0.8589  1.00000   0.8595   1.0000   1.0000   0.8915
## Prevalence             0.1411  0.01532   0.1405   0.2513   0.1702   0.1085
## Detection Rate         0.0000  0.01532   0.0000   0.2513   0.1702   0.0000
## Detection Prevalence   0.0000  0.15643   0.0000   0.3918   0.2787   0.0000
## Balanced Accuracy      0.5000  0.92835   0.5000   0.9061   0.9346   0.5000
##                      Class: 6
## Sensitivity             1.000
```

```
## Specificity             1.000
## Pos Pred Value          1.000
## Neg Pred Value          1.000
## Prevalence              0.173
## Detection Rate          0.173
## Detection Prevalence    0.173
## Balanced Accuracy       1.000
```

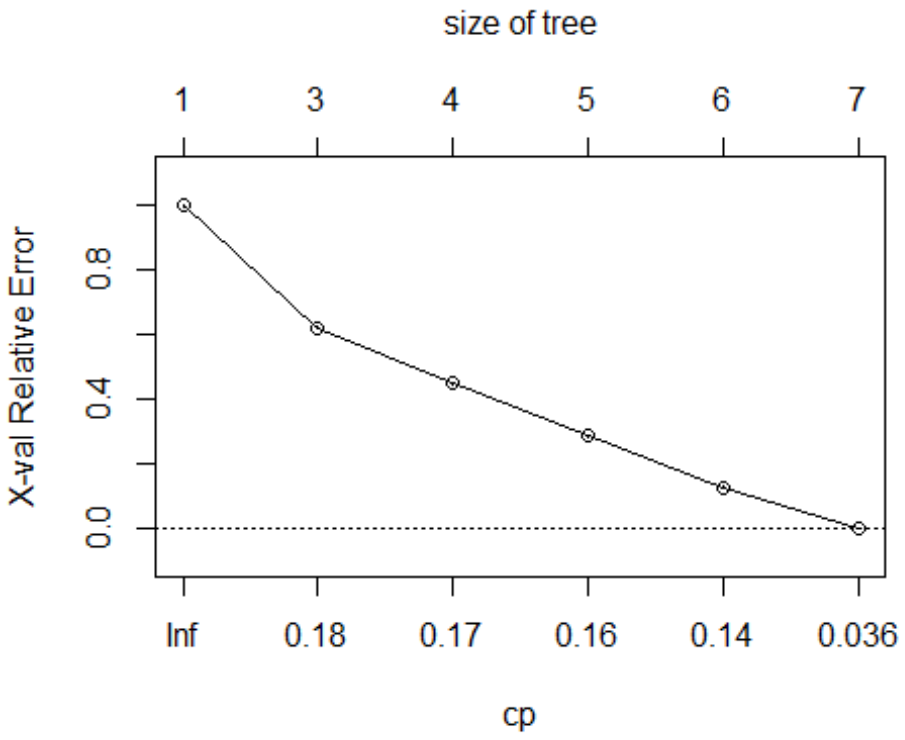
From the table , we get a max accuracy of 65.5% for cp = 0.16399

From this we get the model which has the most important features

```
dat <- readRDS(file = "C:/Users/Nishna/Documents/F21DL_CW3/Datasets/RDS/orig_
train_balanced.Rds")
```

1. cp - complexity parameter
   ## Hyperparameter tuning - cp

```
library(rpart)
n <- ncol(dat)
dtree_3 <- rpart(class ~., data = dat[,c(1:25,n)],
                 method = "class",
                 parms = list(split = "information")
                     )

printcp(dtree_3)

##
## Classification tree:
## rpart(formula = class ~ ., data = dat[, c(1:25, n)], method = "class",
##     parms = list(split = "information"))
##
## Variables actually used in tree construction:
## [1] PC2
##
## Root node error: 18830/22553 = 0.83492
##
## n= 22553
##
##         CP nsplit rel error  xerror      xstd
## 1 0.18874      0   1.00000 1.00000 0.0029609
## 2 0.17313      2   0.62252 0.62252 0.0039846
## 3 0.16399      3   0.44939 0.44939 0.0038615
## 4 0.15836      4   0.28540 0.28540 0.0033978
## 5 0.12703      5   0.12703 0.12703 0.0024557
## 6 0.01000      6   0.00000 0.00000 0.0000000

plotcp(dtree_3)
```



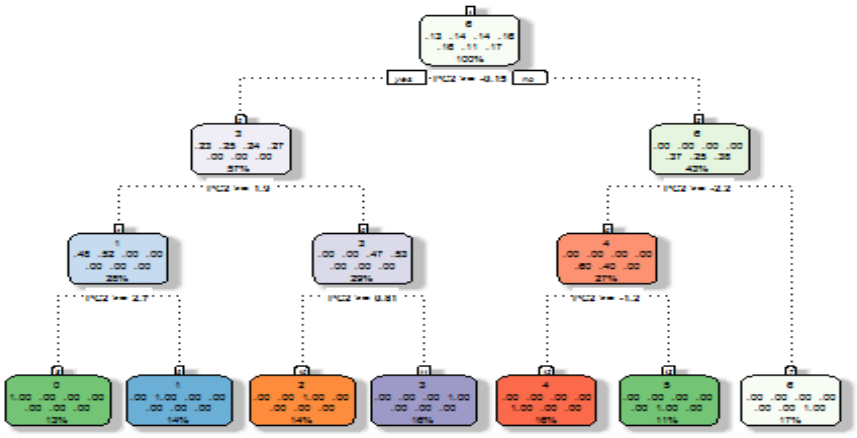

```
#Prediction
load(file = "C:/Users/Nishna/Documents/F21DL_CW3/Datasets/RDA/pca_validation_
original.rda")
y_hat <- predict(dtree_3, as.data.frame(pca_validation), type = "class")

#Results
load("C:/Users/Nishna/Documents/F21DL_CW3/Datasets/RDA/orig_validation.rda")
confusionMatrix(y_hat, factor(orig_validation[,1]))

## Confusion Matrix and Statistics
##
##           Reference
## Prediction    0    1    2    3    4    5    6
##          0 1013    0    0    0    0    0    0
##          1    0  110    0    0    0    0    0
##          2    0    0 1009    0    0    0    0
##          3    0    0    0 1804    0    0    0
##          4    0    0    0    0 1222    0    0
##          5    0    0    0    0    0  779    0
##          6    0    0    0    0    0    0 1242
##
## Overall Statistics
##
##                Accuracy : 1
##                  95% CI : (0.9995, 1)
##     No Information Rate : 0.2513
##     P-Value [Acc > NIR] : < 2.2e-16
```

Nishna

```
##
##                   Kappa : 1
##  Mcnemar's Test P-Value : NA
##
## Statistics by Class:
##
##                      Class: 0 Class: 1 Class: 2 Class: 3 Class: 4 Class: 5
## Sensitivity            1.0000  1.00000   1.0000   1.0000   1.0000   1.0000
## Specificity            1.0000  1.00000   1.0000   1.0000   1.0000   1.0000
## Pos Pred Value         1.0000  1.00000   1.0000   1.0000   1.0000   1.0000
## Neg Pred Value         1.0000  1.00000   1.0000   1.0000   1.0000   1.0000
## Prevalence             0.1411  0.01532   0.1405   0.2513   0.1702   0.1085
## Detection Rate         0.1411  0.01532   0.1405   0.2513   0.1702   0.1085
## Detection Prevalence   0.1411  0.01532   0.1405   0.2513   0.1702   0.1085
## Balanced Accuracy      1.0000  1.00000   1.0000   1.0000   1.0000   1.0000
##                      Class: 6
## Sensitivity             1.000
## Specificity             1.000
## Pos Pred Value          1.000
## Neg Pred Value          1.000
## Prevalence              0.173
## Detection Rate          0.173
## Detection Prevalence    0.173
## Balanced Accuracy       1.000
```

Hyperparameter tuning - Pruning

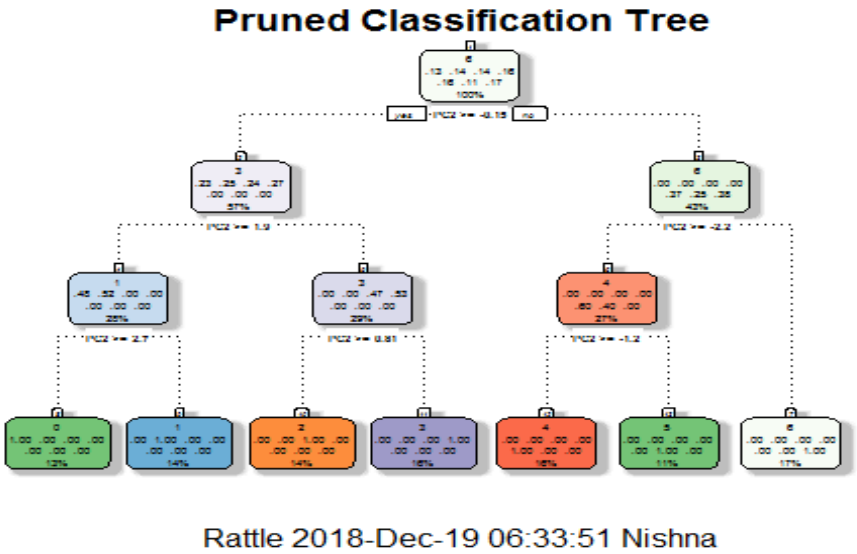
```
library(rattle)
library(RColorBrewer)
library(rpart.plot)
fancyRpartPlot(dtree_3)
```



Rattle 2018-Dec-19 06:33:48 Nishna

Nishna

```
prun_tree_orig<- prune(dtree_3,
                  cp = dtree_3$cptable[which.min(dtree_3$cptable[,"xerror"]),
"CP"])
fancyRpartPlot(prun_tree_orig, uniform=TRUE,
               main="Pruned Classification Tree")
```



The final model - After tuning the parameters

```
#Prediction
load(file = "C:/Users/Nishna/Documents/F21DL_CW3/Datasets/RDA/pca_test_origin
al.rda")
y_hat <- predict(prun_tree_orig, as.data.frame(pca_test), type = "class")
save(prun_tree_orig, file = "C:/Users/Nishna/Documents/F21DL_CW3/Classifiers/
prun_tree_orig.rda")
#Results
load("C:/Users/Nishna/Documents/F21DL_CW3/Datasets/RDA/orig_test.rda")
print('                CONFUSION MATRIX-FINAL MODEL-ORIGINAL TEST')

## [1] "                CONFUSION MATRIX-FINAL MODEL-ORIGINAL TEST"

confusionMatrix(y_hat, factor(orig_test[,1]))

## Confusion Matrix and Statistics
##
##           Reference
## Prediction    0    1    2    3    4    5    6
##          0  954   15    0    0    0    0    0
##          1    4   89   42    0    0    0    0
##          2    0    7  969  162    0    0    0
##          3    0    0   13 1557   82    0    0
##          4    0    0    0   55 1160  155    0
##          5    0    0    0    0    5  658   80
##          6    0    0    0    0    0   18 1153
```

```
##
## Overall Statistics
##
##                Accuracy : 0.9111
##                  95% CI : (0.9043, 0.9176)
##     No Information Rate : 0.2471
##     P-Value [Acc > NIR] : < 2.2e-16
##
##                   Kappa : 0.8928
##  Mcnemar's Test P-Value : NA
##
## Statistics by Class:
##
##                      Class: 0 Class: 1 Class: 2 Class: 3 Class: 4 Class: 5
## Sensitivity            0.9958  0.80180   0.9463   0.8777   0.9302  0.79182
## Specificity            0.9976  0.99349   0.9725   0.9824   0.9646  0.98661
## Pos Pred Value         0.9845  0.65926   0.8515   0.9425   0.8467  0.88560
## Neg Pred Value         0.9994  0.99688   0.9909   0.9607   0.9850  0.97312
## Prevalence             0.1335  0.01546   0.1427   0.2471   0.1737  0.11577
## Detection Rate         0.1329  0.01240   0.1350   0.2169   0.1616  0.09167
## Detection Prevalence   0.1350  0.01881   0.1585   0.2301   0.1909  0.10351
## Balanced Accuracy      0.9967  0.89765   0.9594   0.9300   0.9474  0.88921
##                      Class: 6
## Sensitivity            0.9351
## Specificity            0.9970
## Pos Pred Value         0.9846
## Neg Pred Value         0.9867
## Prevalence             0.1718
## Detection Rate         0.1606
## Detection Prevalence   0.1631
## Balanced Accuracy      0.9660
```