MVA_Ass_9.R

mihikagupta

2020-11-12

```
###### Assignment 9 ##############
## Applying Linear Discriminant Analysis

#Getting working directory
getwd()
```

```
## [1] "/Users/mihikagupta/SEM_2/MVA/Assignments"
```

```
#Setting directory to load data set
setwd("/Users/mihikagupta/SEM_2/MVA/Week10")

#Reading the data into a data frame
#df <- read.csv(file = 'US_Acc_June20.csv')
num <- read.csv(file = 'num.csv')
# Performing analysis on the first 500 records for now to achieve easy and quick results and test the d
num<-num[1:500,]
attach(num)
# Printing first few columns of data set for inference
#head(df)

## Setting random seed to shuffle data before splitting
set.seed(23)

#Checking number of rows
#rows<-sample(nrow(df))

#Shuffling the data
#mva<-df[rows,]

#Taking the required number of instances from the shuffled data to reduce any biases
#mva<-mva[950000:1000000,]

#Checking the structure of the data set
#str(mva)

# Checking the number of rows and columns in the current uncleaned dataset
#ncol(mva)
#nrow(mva)

# Printing all the column names to find and filter the relevant and irrelevant attributes
#names<-names(mva)
#names
```

```
## DATA CLEANING ##

#Dropping the surplus attributes which do not contribute to the analysis
#mva <- mva[-c(1:3,7:10,13,14,19,21:23,33,47:49)]

#Checking for any null values in the present data set
# is.na(mva[,])

#Checking which rows have all the values filled and complete
# complete.cases(mva)

#Making a new dataframe with only the rows that have complete information and all values filled
#Mva<-na.omit(mva)
#Mva<-Mva[!(is.na(Mva$Sunrise_Sunset) | Mva$Sunrise_Sunset==""), ]
#Mva<- Mva[complete.cases(Mva),]
#Verifying for missing values in the new dataframe
#complete.cases(Mva)
#unique(Mva$Sunrise_Sunset)

# Creating new dataframe with only the numerical attributes to perform statistical functions
#num<-Mva[,c(1,4,11:15,17,18)]
#write.csv(num,"/Users/mihikagupta/Desktop/SEM_2/MVA/num.csv", row.names = FALSE)

# Scaling the new data set for better accuracies
# num<-scale(num)

# Checking the dimensions of the data
nrow(num)
```

```
## [1] 500
```

```
ncol(num)
```

```
## [1] 9
```

```
names(num)
```

```
## [1] "Severity"          "Distance.mi."      "Temperature.F."
## [4] "Wind_Chill.F."     "Humidity..."       "Pressure.in."
## [7] "Visibility.mi."    "Wind_Speed.mph."   "Precipitation.in."
```

```
names(num)[names(num) == "Distance.mi."] <- "dist"
names(num)[names(num) == "Temperature.F."] <- "temp"
names(num)[names(num) == "Wind_Chill.F."] <- "windchill"
names(num)[names(num) == "Humidity..."] <- "humidity"
names(num)[names(num) == "Pressure.in."] <- "pressure"
names(num)[names(num) == "Visibility.mi."] <- "visibility"
names(num)[names(num) == "Wind_Speed.mph."] <- "windspeed"
names(num)[names(num) == "Precipitation.in."] <- "precip"
names(num)
```

```
## [1] "Severity"   "dist"       "temp"       "windchill"  "humidity"
## [6] "pressure"   "visibility" "windspeed"  "precip"
```

```
############## LDA ###################
library(MASS)
```

```r
head(num)
```

```
##   Severity dist temp windchill humidity pressure visibility windspeed precip
## 1        2    0   78        78       58    29.16         10        12   0.00
## 2        2    0   96        96       33    29.22         10         7   0.00
## 3        3    0   89        89       59    30.00         10         6   0.00
## 4        2    0   68        68       88    29.40          6         5   0.04
## 5        2    0   53        53       59    29.53         10        12   0.00
## 6        2    0   37        30       96    29.19          2        10   0.02
```

```r
num.data<-as.matrix(num)

# Splitting the dataset into two parts
smp_size_raw<-floor(0.75*nrow(num.data))
train_ind_raw<-sample(nrow(num.data),size = smp_size_raw)
train_raw.df<-as.data.frame(num.data[train_ind_raw,])
test_raw.df<-as.data.frame(num.data[-train_ind_raw,])
# we now have a training and test set .Training is 75% and test is 25%

## Applying lda
num.lda<-lda(formula=train_raw.df$Severity~.,data = train_raw.df)
num.lda
```

```
## Call:
## lda(train_raw.df$Severity ~ ., data = train_raw.df)
##
## Prior probabilities of groups:
##          1          2          3          4
## 0.02133333 0.66933333 0.29066667 0.01866667
##
## Group means:
##        dist     temp windchill humidity pressure visibility windspeed
## 1 1.1135000 70.62500  70.62500 43.25000 28.75875  10.000000 15.000000
## 2 0.1395697 62.37092  61.13386 63.19124 29.12359   9.082470  7.453785
## 3 0.2200642 61.03394  59.56330 66.02752 29.34413   8.755046  7.628440
## 4 0.7680000 66.71429  65.71429 71.71429 29.51857   9.142857  5.571429
##        precip
## 1 0.000000000
## 2 0.006533865
## 3 0.007339450
## 4 0.010000000
##
## Coefficients of linear discriminants:
##                     LD1         LD2          LD3
## dist       -1.12160595  1.00513811  0.146846436
## temp        0.23474890  0.18066456  0.092257940
## windchill  -0.21517971 -0.15179321 -0.060249112
## humidity    0.01293508  0.02476000  0.004200698
## pressure    0.03356420  0.23516608 -0.475098496
## visibility  0.01814974  0.05789129  0.131684221
## windspeed  -0.13233103 -0.07459098 -0.074700341
## precip      3.47082907  1.30403977  2.880775347
##
## Proportion of trace:
##    LD1    LD2    LD3
```

```
## 0.7794 0.1791 0.0416
```

```
summary(num.lda)
```

```
##         Length Class  Mode
## prior    4     -none- numeric
## counts   4     -none- numeric
## means   32     -none- numeric
## scaling 24     -none- numeric
## lev      4     -none- character
## svd      3     -none- numeric
## N        1     -none- numeric
## call     3     -none- call
## terms    3     terms  call
## xlevels  0     -none- list
```

```
# general parameters in lda
num.lda$counts
```

```
##   1   2   3   4
##   8 251 109   7
```

```
num.lda$means
```

```
##        dist     temp windchill humidity pressure visibility windspeed
## 1 1.1135000 70.62500  70.62500 43.25000 28.75875  10.000000 15.000000
## 2 0.1395697 62.37092  61.13386 63.19124 29.12359   9.082470  7.453785
## 3 0.2200642 61.03394  59.56330 66.02752 29.34413   8.755046  7.628440
## 4 0.7680000 66.71429  65.71429 71.71429 29.51857   9.142857  5.571429
##        precip
## 1 0.000000000
## 2 0.006533865
## 3 0.007339450
## 4 0.010000000
```

```
num.lda$scaling
```

```
##                   LD1         LD2          LD3
## dist       -1.12160595  1.00513811  0.146846436
## temp        0.23474890  0.18066456  0.092257940
## windchill  -0.21517971 -0.15179321 -0.060249112
## humidity    0.01293508  0.02476000  0.004200698
## pressure    0.03356420  0.23516608 -0.475098496
## visibility  0.01814974  0.05789129  0.131684221
## windspeed  -0.13233103 -0.07459098 -0.074700341
## precip      3.47082907  1.30403977  2.880775347
```

```
num.lda$prior
```

```
##          1          2          3          4
## 0.02133333 0.66933333 0.29066667 0.01866667
```

```
num.lda$lev
```

```
## [1] "1" "2" "3" "4"
```

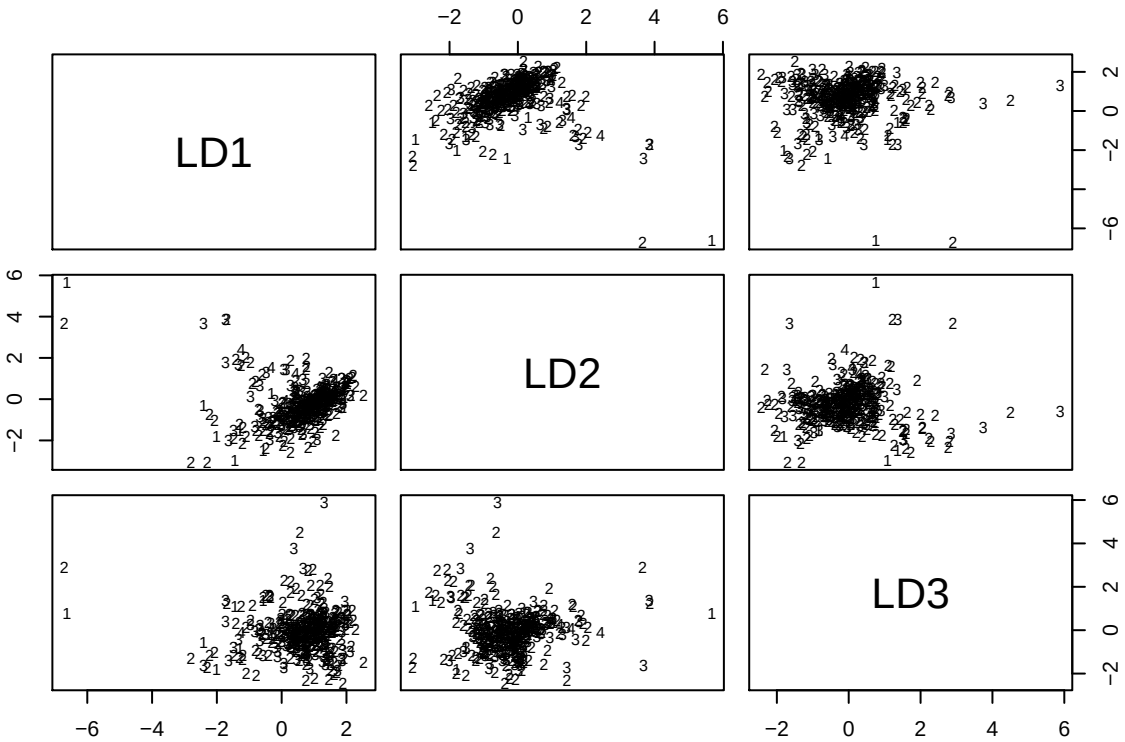
```
num.lda$svd
```

```
## [1] 3.9897350 1.9124438 0.9214469
```

```r
print(num.lda)
```

```
## Call:
## lda(train_raw.df$Severity ~ ., data = train_raw.df)
##
## Prior probabilities of groups:
##          1          2          3          4
## 0.02133333 0.66933333 0.29066667 0.01866667
##
## Group means:
##        dist     temp windchill humidity pressure visibility windspeed
## 1 1.1135000 70.62500  70.62500 43.25000 28.75875  10.000000 15.000000
## 2 0.1395697 62.37092  61.13386 63.19124 29.12359   9.082470  7.453785
## 3 0.2200642 61.03394  59.56330 66.02752 29.34413   8.755046  7.628440
## 4 0.7680000 66.71429  65.71429 71.71429 29.51857   9.142857  5.571429
##        precip
## 1 0.000000000
## 2 0.006533865
## 3 0.007339450
## 4 0.010000000
##
## Coefficients of linear discriminants:
##                     LD1         LD2          LD3
## dist       -1.12160595  1.00513811  0.146846436
## temp        0.23474890  0.18066456  0.092257940
## windchill  -0.21517971 -0.15179321 -0.060249112
## humidity    0.01293508  0.02476000  0.004200698
## pressure    0.03356420  0.23516608 -0.475098496
## visibility  0.01814974  0.05789129  0.131684221
## windspeed  -0.13233103 -0.07459098 -0.074700341
## precip      3.47082907  1.30403977  2.880775347
##
## Proportion of trace:
##    LD1    LD2    LD3
## 0.7794 0.1791 0.0416
```

```r
plot(num.lda)
```




```
#predicting the classes
num.lda.predict<-predict(num.lda,newdata = test_raw.df)
num.lda.predict$class
```

```
##   [1] 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 4 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
##  [38] 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 1 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
##  [75] 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
## [112] 2 2 2 2 2 2 2 2 2 2 2 2 2 2
## Levels: 1 2 3 4
```

```
# View(num.lda.predict)
num.lda.predict$x
```

```
##              LD1          LD2          LD3
## 2    0.287291841 -0.073108518  0.988890901
## 5   -0.869121392 -0.970870097 -0.799052779
## 7    1.352167134 -0.151442020 -1.269369867
## 18   0.663234808  0.233811383 -0.993821439
## 24   0.020384113 -0.006940630  0.188253822
## 26   0.894813356  0.618266450  0.602662701
## 29   0.835449565  0.640282834  0.179138910
## 33   0.585502496 -0.306137430 -1.764743852
## 34   1.559377505  1.272633558  1.132045990
## 49  -0.041204330 -0.161098640  0.282496370
## 54   0.547432191  0.002285577 -1.137223307
## 55  -2.018004792 -1.600918762 -0.841055986
## 63  -0.081131333  0.047052688 -0.736313032
## 65   0.895227407  0.047225993 -1.470326231
## 67   1.217803662  0.364538121 -2.250109132
## 68  -1.244073813 -1.929463828  1.508110306
## 78   1.153837794  0.802885586  0.111455968
```

```
## 79  -0.379800347 -1.729145301  2.799392560
## 81  -0.089916348 -0.182306821  0.054435074
## 84   1.199297504  1.250876414  0.553836261
## 85  -0.451346120 -0.368396531 -0.082820780
## 88  -2.340758936  3.554040299 -0.678027106
## 89   0.978572653  0.277255364 -1.330832033
## 92  -0.387838447  0.509511908 -0.339958129
## 104 -0.159845367  0.201010654 -0.004576239
## 107 -0.077170914  1.235951368  0.169413225
## 110 -0.878827840 -1.731150400  1.388491823
## 122  0.255780913  0.214367175 -0.539844066
## 124  0.066286176 -0.285509193 -0.199046482
## 126 -1.542612745 -1.155653080 -1.652359765
## 133  1.558832942  1.447553917  0.633998157
## 134 -0.891146886 -0.401431086 -0.113645209
## 135 -0.642258412 -0.566984150 -1.696738385
## 139  1.696747961  1.632675476  0.910468278
## 142  1.611908306  1.499333688  0.800296648
## 148  0.042867741 -0.207604768  0.918163624
## 157 -0.156211083 -0.638453067  0.496371236
## 158 -0.118301406 -0.282749679 -0.329766646
## 165  0.840615237  0.235188045 -1.636559596
## 173  0.199390725 -0.095252084  1.025020285
## 175 -1.216618584 -0.247749091 -0.918895568
## 176  0.298093022  0.347430312  0.312074786
## 180  0.272671828  0.018495962 -0.736745288
## 186 -0.943207672 -0.716026155 -0.490786219
## 191  1.179194787  0.168365540 -1.996523052
## 197  0.999293403  0.766734572  0.790514970
## 200  0.338863805 -0.453500346 -0.900494194
## 201  0.699237269  0.052301744 -2.244479563
## 203  0.181331711  0.160821842  0.644455785
## 207 -0.599679679  0.815863353  0.531731687
## 208  0.744474562  0.308355520  0.614312116
## 214 -0.304112082  0.052047767  0.070769406
## 216  1.131479640  0.523389468  0.801948312
## 218  0.777817689 -0.186115398 -1.281572031
## 220 -0.478482321  0.073185511 -0.237479950
## 224 -4.322099196  3.921168254  0.710547657
## 227  0.920701434  0.261073920  0.302488534
## 229 -0.322037468 -0.733594960 -0.901072135
## 233  1.373083618  1.199386986  0.455093368
## 237 -0.574414033 -0.733660313 -0.719251994
## 242  0.460922247 -0.134864757  0.226811074
## 246  0.134258395 -1.108201582  4.152759655
## 249  0.559695174  0.669095183 -0.210570384
## 250 -0.184418664  0.255482537  0.183893668
## 252  0.257052558  0.456006615  0.246782625
## 263 -0.558514038 -0.342168799 -0.289178807
## 271  0.094723431  0.484470269  8.293114356
## 273  0.576288314 -0.210720433 -0.687787497
## 278  1.281250409  1.220969557  0.389893452
## 281  0.141105237  0.075581824 -0.067223646
## 283 -0.423500504  0.058883672 -0.168229875
```

```
## 285  1.286980110  0.949939952  0.464934615
## 289  0.239177009 -0.427384159 -0.230250180
## 294  0.368767512 -0.891512327  0.407589729
## 296 -0.088942453 -0.312758949 -0.052865294
## 300  0.419965761 -0.157986769  0.039038177
## 305  0.303634064 -0.081639231  0.617373414
## 308 -0.866929713 -0.501316935 -0.671244276
## 309  0.962358193  0.087164756 -2.670322641
## 315 -0.404472412 -0.137142180  0.498907570
## 325  0.268785320 -0.113954291  0.502087690
## 328 -0.318433084 -0.413669613 -0.634111021
## 331  0.385102270 -0.210025743  0.021507051
## 332 -0.115444164  1.391253661  0.694997380
## 335  0.680714677  0.380780980  1.010806157
## 338 -1.502747430 -0.334894332 -0.424231748
## 342 -0.224428861 -0.105870178  0.506652412
## 351  0.804643865  0.968009249  0.521144530
## 357  1.317984345  0.908320937  0.765803738
## 359  0.302105510 -0.282708467 -0.287330121
## 367 -0.625495746 -0.595904647 -0.436634099
## 368  0.700730865  0.700482357  0.185071068
## 369  1.100974165  0.370534308  1.074232154
## 372  0.104611596  0.321856779 -0.066584130
## 374  0.578200318  0.228083632 -0.774584163
## 375  0.791306740  0.366294359  0.014832204
## 378  1.099692381  0.441802559 -0.996921197
## 379  0.965728570  0.852015981  0.329720765
## 380 -0.567429333  0.834401222 -1.579674265
## 391 -0.335558073 -0.371783419  0.627991292
## 403  0.008872967 -0.287020328 -0.042035116
## 405  1.122676092  0.758660872  0.056242502
## 407  0.266865189 -1.560472982  0.521669377
## 409 -0.046679635 -1.407067444  2.373105226
## 411  0.306522123  0.581506866  0.174014483
## 415 -0.500091974  1.536404932  0.856648393
## 420  0.294556996  0.560587710 -0.406808858
## 421 -0.569758362 -0.260725691 -0.597967162
## 429  0.372771694 -0.490657114 -0.762258280
## 431  0.532498708  0.028032251  0.169033352
## 433  0.494835619 -0.662284964  0.486611823
## 439 -1.005789760 -0.989102766  0.413539364
## 445 -0.115102678 -0.132678698 -0.244301023
## 446  0.272186276 -1.095473541  1.600737175
## 451  0.568443622  0.171207975 -1.605039821
## 454  0.857916432  0.492185638 -0.611475518
## 455 -0.226851945 -0.454012909  0.522314118
## 462  0.581185506  0.186887982  0.986850036
## 474 -1.573431424 -1.643956075 -1.028010193
## 476  1.038823423  0.616085688  0.101465845
## 480  0.947773327  0.002133285 -0.874943060
## 486 -0.935003797 -1.043502050 -0.183027398
## 487  0.054777415 -0.220162766 -0.457412041
## 491  0.469567573  0.191004762  0.310369765
## 492  0.830755596  0.645137505  0.133319607
```

```r
# Getting the posteriors as a dataframe
num.lda.predict.posteriors<-as.data.frame(num.lda.predict$posterior)
library("ROCR")

# creating the ROC/AUC curve
# pred<-prediction(num.lda.predict.posteriors[,4],test_raw.df$Severity)
# ROCR not working here since it supports evaluation of only binary classification taska as of now

#singular values (svd) that gives the ratio of the between- and within-group standard deviations on the
class(num.lda)
```

```
## [1] "lda"
```

```r
num.lda$N
```

```
## [1] 375
```

```r
num.lda$call
```

```
## lda(formula = train_raw.df$Severity ~ ., data = train_raw.df)
```

```r
(prop=num.lda$svd^2/sum(num.lda$svd^2))
```

```
## [1] 0.7793577 0.1790714 0.0415709
```

```r
#we can use the singular values to compute the amount of the between-group variance that is explained b
num.lda2<-lda(formula=train_raw.df$Severity~.,data = train_raw.df,CV=TRUE)
num.lda2
```

```
## $class
##   [1] 4 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
##  [38] 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 1 2 2 2 2 2 2 2 2 1 2 2 2 2
##  [75] 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 1 2 2 2 2 2 2 2 2 2 2 2 2 2 2
## [112] 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
## [149] 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
## [186] 2 2 2 2 2 2 4 2 1 2 2 2 2 2 2 2 2 2 2 2 2 2 2 3 2 2 2 2 2 2 2 2 2 2 2 2
## [223] 2 1 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
## [260] 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 3 2 2
## [297] 2 2 2 2 2 2 2 1 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
## [334] 2 2 2 2 2 2 2 2 2 2 4 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
## [371] 2 2 1 2 2
## Levels: 1 2 3 4
##
## $posterior
##                1            2            3            4
## 413 7.519504e-02 6.952900e-02 3.249050e-02 8.227855e-01
## 284 1.048764e-04 7.063073e-01 2.808752e-01 1.271262e-02
## 248 7.720173e-04 6.722520e-01 3.124529e-01 1.452309e-02
## 456 1.472728e-04 5.957850e-01 3.987181e-01 5.349592e-03
## 121 1.910645e-03 6.229881e-01 3.569750e-01 1.812629e-02
## 171 1.347669e-02 6.868756e-01 2.953835e-01 4.264196e-03
## 103 1.671273e-03 7.296477e-01 2.645416e-01 4.139500e-03
## 301 8.921714e-04 7.036710e-01 2.920859e-01 3.350868e-03
## 290 4.492009e-02 7.667302e-01 1.874176e-01 9.321372e-04
## 432 1.789357e-04 6.251082e-01 3.640786e-01 1.063428e-02
## 401 2.075277e-03 6.636539e-01 3.134311e-01 2.083969e-02
## 149 6.489273e-04 6.017379e-01 2.964089e-01 1.012042e-01
```

```
## 145 1.932663e-03 7.175656e-01 2.760574e-01 4.444432e-03
## 424 1.562086e-02 7.371387e-01 2.428740e-01 4.366399e-03
## 164 5.393116e-02 6.028176e-01 3.413755e-01 1.875812e-03
## 326 3.836432e-05 6.760999e-01 2.990884e-01 2.477328e-02
## 198 4.571936e-04 7.378255e-01 2.544873e-01 7.229973e-03
## 31  2.398612e-03 7.161280e-01 2.762447e-01 5.228658e-03
## 86  6.060000e-04 7.046441e-01 2.807139e-01 1.403596e-02
## 461 1.736104e-04 6.834215e-01 3.070584e-01 9.346488e-03
## 298 2.496556e-04 6.839157e-01 2.996472e-01 1.618742e-02
## 236 4.230638e-05 6.958798e-01 2.805041e-01 2.357377e-02
## 10  2.573936e-03 6.854924e-01 3.044469e-01 7.486750e-03
## 345 4.382069e-03 7.443024e-01 2.480506e-01 3.264963e-03
## 38  9.493903e-03 7.153529e-01 2.730877e-01 2.065515e-03
## 167 8.462972e-03 7.467374e-01 2.424661e-01 2.333481e-03
## 354 1.370904e-01 5.985133e-01 2.632552e-01 1.141130e-03
## 251 3.259562e-03 5.981425e-01 3.236468e-01 7.495111e-02
## 47  2.312637e-04 7.215858e-01 2.710197e-01 7.163310e-03
## 302 1.815202e-04 6.675656e-01 3.191673e-01 1.308556e-02
## 347 1.394526e-03 7.270680e-01 2.681245e-01 3.413013e-03
## 66  1.207343e-04 6.912328e-01 2.966951e-01 1.195141e-02
## 316 3.129462e-04 7.583392e-01 2.325382e-01 8.809663e-03
## 69  2.106729e-03 6.994542e-01 2.914597e-01 6.979446e-03
## 105 1.851769e-03 7.069939e-01 2.856775e-01 5.476854e-03
## 473 1.620238e-03 6.266362e-01 3.317060e-01 4.003752e-02
## 340 3.392812e-02 6.815446e-01 2.779953e-01 6.532003e-03
## 91  2.044073e-03 6.773649e-01 3.047463e-01 1.584471e-02
## 307 9.837957e-05 6.809801e-01 3.051085e-01 1.381304e-02
## 292 2.893425e-04 6.765502e-01 3.121950e-01 1.096543e-02
## 333 1.743319e-04 7.029056e-01 2.879281e-01 8.991990e-03
## 231 6.392500e-04 7.353209e-01 2.583622e-01 5.677620e-03
## 206 7.624664e-04 7.117860e-01 2.775141e-01 9.937462e-03
## 304 1.428921e-03 6.596793e-01 3.291741e-01 9.717704e-03
## 337 1.655909e-03 6.674409e-01 2.650930e-01 6.581015e-02
## 59  3.416510e-04 7.430691e-01 2.511701e-01 5.419170e-03
## 178 1.669481e-01 5.426759e-01 2.869705e-01 3.405540e-03
## 348 8.736996e-04 6.707601e-01 3.186755e-01 9.690681e-03
## 493 1.103405e-03 7.239986e-01 2.643996e-01 1.049835e-02
## 159 9.346391e-05 6.462772e-01 3.440946e-01 9.534703e-03
## 269 1.281616e-02 6.946143e-01 2.881834e-01 4.386184e-03
## 291 1.206730e-03 7.057828e-01 2.843292e-01 8.681298e-03
## 435 7.082698e-03 7.854657e-01 2.057174e-01 1.734141e-03
## 241 2.292635e-04 6.680731e-01 3.097061e-01 2.199152e-02
## 469 1.272682e-04 6.141182e-01 3.785373e-01 7.217205e-03
## 190 2.943593e-04 6.841674e-01 2.953700e-01 2.016819e-02
## 3   2.342408e-04 6.957319e-01 2.841561e-01 1.987776e-02
## 355 1.392451e-03 7.334623e-01 2.577117e-01 7.433545e-03
## 50  3.502599e-05 6.631129e-01 3.087929e-01 2.805919e-02
## 225 7.755608e-01 1.557377e-01 6.858203e-02 1.194533e-04
## 450 4.042867e-04 6.742663e-01 3.115194e-01 1.381003e-02
## 60  5.341534e-04 7.400519e-01 2.530007e-01 6.413177e-03
## 384 4.647280e-03 7.120170e-01 2.780448e-01 5.290974e-03
## 14  3.119359e-03 7.433129e-01 2.493629e-01 4.204874e-03
## 9   2.104208e-02 5.595460e-01 3.503166e-01 6.909534e-02
## 147 2.357985e-04 6.537026e-01 3.347010e-01 1.136062e-02
```

```
## 398 1.868431e-04 7.088620e-01 2.717071e-01 1.924402e-02
## 51  5.964538e-03 6.825262e-01 3.022673e-01 9.241912e-03
## 244 4.337193e-03 7.002996e-01 2.858655e-01 9.497754e-03
## 472 9.999879e-01 3.203881e-08 4.607419e-08 1.201969e-05
## 468 3.340302e-04 6.796578e-01 3.102764e-01 9.731839e-03
## 402 2.081106e-02 8.130432e-01 1.634811e-01 2.664645e-03
## 496 1.253177e-03 6.348726e-01 3.547685e-01 9.105714e-03
## 125 3.996431e-04 6.720978e-01 3.225153e-01 4.987257e-03
## 388 1.338337e-03 6.997959e-01 2.930087e-01 5.857042e-03
## 172 3.199287e-04 6.991238e-01 2.846668e-01 1.588948e-02
## 174 4.010261e-05 6.671423e-01 3.072381e-01 2.557953e-02
## 499 1.934737e-03 6.464271e-01 3.272601e-01 2.437811e-02
## 312 2.207936e-04 7.215524e-01 2.614849e-01 1.674185e-02
## 76  7.392069e-04 6.829648e-01 3.133411e-01 2.954888e-03
## 363 3.538720e-03 7.235690e-01 2.695393e-01 3.352936e-03
## 382 3.040956e-03 7.016466e-01 2.914160e-01 3.896462e-03
## 161 1.159924e-04 6.582920e-01 3.339290e-01 7.662957e-03
## 58  2.498651e-04 6.954392e-01 2.971764e-01 7.134553e-03
## 412 2.299780e-01 5.296309e-01 2.388680e-01 1.523087e-03
## 195 8.204855e-04 7.071394e-01 2.851040e-01 6.936028e-03
## 138 8.397253e-04 6.363809e-01 3.535992e-01 9.180183e-03
## 381 6.296527e-04 6.899516e-01 2.988045e-01 1.061427e-02
## 389 7.688345e-04 6.675611e-01 3.116002e-01 2.006986e-02
## 74  2.374802e-03 6.690269e-01 3.153790e-01 1.321930e-02
## 101 8.206423e-05 6.901612e-01 2.850061e-01 2.475070e-02
## 37  2.630430e-04 6.745015e-01 3.031251e-01 2.211036e-02
## 383 9.672795e-05 6.752738e-01 3.108183e-01 1.381120e-02
## 490 1.870414e-03 6.739898e-01 3.052476e-01 1.889217e-02
## 199 9.327008e-01 4.704891e-02 2.020354e-02 4.670922e-05
## 414 2.637456e-04 7.424516e-01 2.510189e-01 6.265723e-03
## 41  2.865133e-03 6.846202e-01 3.030819e-01 9.432742e-03
## 477 8.851621e-05 6.437668e-01 3.533895e-01 2.755121e-03
## 376 7.160927e-05 7.017030e-01 2.815613e-01 1.666413e-02
## 189 3.541536e-03 7.087638e-01 2.799959e-01 7.698843e-03
## 396 2.013114e-05 6.524645e-01 3.114725e-01 3.604285e-02
## 484 7.031860e-06 5.998458e-01 3.933299e-01 6.817254e-03
## 98  3.909320e-03 5.546617e-01 3.930546e-01 4.837443e-02
## 162 5.315878e-05 6.961790e-01 2.832445e-01 2.052338e-02
## 128 3.995849e-05 6.690562e-01 3.122602e-01 1.864368e-02
## 494 1.112237e-01 5.787552e-01 3.086137e-01 1.407410e-03
## 30  5.630015e-04 7.039621e-01 2.829294e-01 1.254552e-02
## 346 1.742054e-04 6.539984e-01 3.247999e-01 2.102749e-02
## 166 9.939537e-04 6.900458e-01 3.032454e-01 5.714849e-03
## 19  7.154008e-02 6.456839e-01 2.798305e-01 2.945540e-03
## 361 7.880959e-04 6.798786e-01 3.136658e-01 5.667520e-03
## 330 2.401083e-04 7.335104e-01 2.586663e-01 7.583189e-03
## 112 4.397467e-04 7.047270e-01 2.871109e-01 7.722393e-03
## 406 2.083615e-03 6.816882e-01 3.127885e-01 3.439717e-03
## 428 1.113583e-03 6.927364e-01 2.982594e-01 7.890647e-03
## 40  3.629881e-04 6.638286e-01 3.148493e-01 2.095912e-02
## 393 4.018845e-04 6.615834e-01 3.237639e-01 1.425084e-02
## 35  2.579816e-04 6.633995e-01 3.138422e-01 2.250030e-02
## 205 4.199092e-04 6.624040e-01 3.309500e-01 6.226172e-03
## 155 2.673880e-02 7.776448e-01 1.938420e-01 1.774426e-03
```

```
## 483 5.350886e-03 6.642438e-01 3.182434e-01 1.216190e-02
## 25  2.783091e-03 8.665103e-01 1.228045e-01 7.902133e-03
## 390 9.911358e-04 6.306898e-01 3.337810e-01 3.453813e-02
## 156 6.875565e-04 8.284585e-01 1.684157e-01 2.438251e-03
## 400 1.304753e-03 6.998309e-01 2.929233e-01 5.941073e-03
## 179 7.705968e-04 5.821195e-01 3.973565e-01 1.975338e-02
## 334 2.196037e-03 6.874658e-01 3.069571e-01 3.381063e-03
## 152 1.946962e-03 7.263509e-01 2.680383e-01 3.663837e-03
## 183 1.518207e-03 7.801933e-01 2.142776e-01 4.010897e-03
## 320 1.747480e-04 7.845285e-01 2.066583e-01 8.638481e-03
## 235 3.430003e-05 6.857383e-01 2.866320e-01 2.759541e-02
## 427 6.235208e-04 6.505923e-01 3.405693e-01 8.214925e-03
## 170 1.402133e-04 6.573709e-01 3.341302e-01 8.358659e-03
## 212 1.142226e-01 4.271591e-01 2.896106e-01 1.690076e-01
## 449 1.278473e-03 7.006319e-01 2.909896e-01 7.099986e-03
## 219 2.218544e-01 5.525453e-01 2.228711e-01 2.729241e-03
## 377 2.204237e-04 6.550362e-01 3.075395e-01 3.720386e-02
## 288 5.475836e-04 7.420303e-01 2.553290e-01 2.093133e-03
## 336 1.721403e-02 7.588060e-01 2.195283e-01 4.451688e-03
## 57  2.231765e-03 7.067893e-01 2.853352e-01 5.643752e-03
## 43  1.852385e-04 6.756831e-01 3.189979e-01 5.133778e-03
## 106 1.829277e-02 7.116420e-01 2.646520e-01 5.413278e-03
## 344 1.243994e-04 6.225679e-01 3.310670e-01 4.624065e-02
## 299 2.669011e-03 6.661234e-01 3.222709e-01 8.936722e-03
## 371 1.868249e-04 6.994293e-01 2.916933e-01 8.690520e-03
## 119 1.033071e-03 6.773607e-01 3.154039e-01 6.202364e-03
## 217 4.437114e-04 7.040282e-01 2.849164e-01 1.061174e-02
## 87  8.246674e-04 6.741083e-01 3.229062e-01 2.160835e-03
## 95  1.752941e-04 6.883197e-01 2.965324e-01 1.497258e-02
## 253 3.759112e-04 7.474454e-01 2.440828e-01 8.095819e-03
## 422 4.715657e-04 6.962019e-01 2.930431e-01 1.028338e-02
## 36  3.693716e-03 8.033210e-01 1.898193e-01 3.165978e-03
## 297 5.168887e-05 6.667341e-01 3.250718e-01 8.142441e-03
## 130 1.813325e-04 6.501507e-01 3.262262e-01 2.344172e-02
## 268 1.216813e-02 7.105545e-01 2.709989e-01 6.278498e-03
## 32  7.867702e-05 6.548010e-01 3.270041e-01 1.811625e-02
## 256 5.698942e-02 6.340971e-01 2.841679e-01 2.474563e-02
## 272 1.117241e-03 6.952798e-01 3.005899e-01 3.013026e-03
## 255 1.011550e-04 7.647584e-01 2.334359e-01 1.704479e-03
## 370 8.758851e-03 7.099931e-01 2.772279e-01 4.020122e-03
## 410 6.696702e-03 7.186631e-01 2.711450e-01 3.495167e-03
## 53  3.325894e-04 6.788913e-01 3.049875e-01 1.578862e-02
## 463 1.541147e-03 6.326694e-01 3.580857e-01 7.703821e-03
## 177 3.409873e-03 6.602211e-01 3.261986e-01 1.017047e-02
## 140 7.462899e-04 7.266063e-01 2.643208e-01 8.326633e-03
## 254 5.849197e-05 6.981494e-01 2.974316e-01 4.360558e-03
## 116 3.822745e-04 6.798812e-01 3.075263e-01 1.221022e-02
## 318 5.241199e-05 6.954809e-01 2.984415e-01 6.025215e-03
## 274 5.301656e-03 7.178611e-01 2.710879e-01 5.749314e-03
## 123 6.133189e-03 6.925091e-01 2.959417e-01 5.416016e-03
## 182 1.321722e-01 5.754391e-01 2.905883e-01 1.800440e-03
## 447 4.250107e-05 7.274776e-01 2.497999e-01 2.268002e-02
## 430 4.676410e-05 6.407317e-01 3.517026e-01 7.518975e-03
## 82  7.148837e-03 6.840805e-01 3.030134e-01 5.757263e-03
```

```
## 46  2.411257e-04 6.847438e-01 3.100078e-01 5.007289e-03
## 500 2.033358e-04 6.530815e-01 3.349152e-01 1.179994e-02
## 311 2.747553e-04 7.687363e-01 2.268609e-01 4.128057e-03
## 127 7.446625e-04 7.937832e-01 2.011543e-01 4.317824e-03
## 303 3.233401e-04 6.459755e-01 3.463336e-01 7.367596e-03
## 187 4.330213e-04 6.963132e-01 2.994188e-01 3.834908e-03
## 287 6.482937e-04 7.445023e-01 2.491116e-01 5.737818e-03
## 181 3.186281e-04 7.593911e-01 2.348326e-01 5.457736e-03
## 259 2.056022e-03 5.522228e-01 3.168876e-01 1.288336e-01
## 115 2.249416e-04 6.874485e-01 2.919120e-01 2.041457e-02
## 257 1.329156e-04 6.477826e-01 3.443758e-01 7.708769e-03
## 489 9.392532e-05 6.514311e-01 3.158756e-01 3.259939e-02
## 353 6.831374e-04 7.287532e-01 2.581772e-01 1.238649e-02
## 358 3.192841e-03 6.887483e-01 3.042247e-01 3.834177e-03
## 64  7.528021e-03 6.748913e-01 3.109286e-01 6.652127e-03
## 83  5.423775e-05 7.018378e-01 2.900876e-01 8.020411e-03
## 247 3.215344e-05 6.732791e-01 2.974763e-01 2.921250e-02
## 438 6.421587e-02 9.124539e-02 8.612014e-02 7.584186e-01
## 266 2.698790e-05 6.561141e-01 3.122439e-01 3.161502e-02
## 71  6.678331e-01 2.056923e-01 1.232664e-01 3.208223e-03
## 313 1.171389e-04 6.810418e-01 3.066092e-01 1.223185e-02
## 6   1.046305e-04 6.385819e-01 3.548940e-01 6.419496e-03
## 222 8.685419e-02 4.856104e-01 2.734821e-01 1.540533e-01
## 264 3.179932e-04 6.775689e-01 3.118625e-01 1.025063e-02
## 270 5.887375e-04 6.758066e-01 3.144580e-01 9.146726e-03
## 62  1.345669e-03 6.787267e-01 3.166840e-01 3.243617e-03
## 232 2.881394e-04 6.787909e-01 3.106226e-01 1.029837e-02
## 276 8.926199e-04 6.694628e-01 3.242430e-01 5.401652e-03
## 397 4.550571e-04 7.091204e-01 2.858476e-01 4.576951e-03
## 465 1.726770e-04 6.846923e-01 3.057783e-01 9.356780e-03
## 102 1.873743e-03 7.053284e-01 2.852854e-01 7.512475e-03
## 188 3.227749e-01 4.837327e-01 1.923204e-01 1.171991e-03
## 440 1.018231e-04 7.014617e-01 2.927934e-01 5.643136e-03
## 408 4.142262e-03 7.010108e-01 2.905079e-01 4.339107e-03
## 153 1.072991e-04 6.877967e-01 2.914333e-01 2.066268e-02
## 457 3.565532e-06 4.765222e-01 4.881437e-01 3.533052e-02
## 295 1.184915e-02 6.902493e-01 2.929577e-01 4.943831e-03
## 314 9.779471e-03 5.884375e-01 3.533653e-01 4.841774e-02
## 339 1.641717e-04 6.627133e-01 3.127047e-01 2.441778e-02
## 210 2.556132e-04 6.981260e-01 2.832890e-01 1.832934e-02
## 16  1.768703e-03 6.468936e-01 3.413025e-01 1.003524e-02
## 100 1.093142e-03 6.806834e-01 3.069359e-01 1.128752e-02
## 230 8.887936e-04 6.483034e-01 3.419701e-01 8.837683e-03
## 12  2.973771e-03 6.828817e-01 3.084779e-01 5.666615e-03
## 394 2.066443e-03 7.096531e-01 2.845492e-01 3.731283e-03
## 97  6.324827e-04 7.137211e-01 2.774125e-01 8.233923e-03
## 497 8.188505e-05 6.733043e-01 3.108153e-01 1.579846e-02
## 436 2.170112e-02 7.535943e-01 2.206029e-01 4.101735e-03
## 479 1.294664e-04 6.579422e-01 3.329985e-01 8.929776e-03
## 137 4.879809e-01 8.369974e-02 7.946865e-02 3.488507e-01
## 498 3.540163e-04 6.892583e-01 2.928973e-01 1.749038e-02
## 202 2.491433e-04 6.469211e-01 3.450924e-01 7.737355e-03
## 93  6.722059e-04 6.864328e-01 3.015703e-01 1.132470e-02
## 160 3.025459e-04 6.784631e-01 3.068923e-01 1.434206e-02
```

```
## 17  7.467834e-05 6.910446e-01 2.953676e-01 1.351310e-02
## 392 7.102377e-05 7.049960e-01 2.783456e-01 1.658733e-02
## 169 5.984594e-04 6.795359e-01 3.108788e-01 8.986878e-03
## 452 7.421918e-05 6.437447e-01 3.517726e-01 4.408435e-03
## 448 4.636358e-04 6.584935e-01 3.317435e-01 9.299394e-03
## 478 2.115234e-03 6.891002e-01 3.028050e-01 5.979547e-03
## 317 2.051429e-04 6.690819e-01 3.222305e-01 8.482468e-03
## 113 7.556962e-04 7.094840e-01 2.779957e-01 1.176463e-02
## 151 9.943495e-04 7.023272e-01 2.867908e-01 9.887611e-03
## 215 8.583466e-05 6.708302e-01 3.092253e-01 1.985865e-02
## 13  7.172058e-03 7.323710e-01 2.558899e-01 4.566990e-03
## 61  4.671310e-04 8.345861e-01 1.631215e-01 1.825251e-03
## 488 9.376654e-04 7.064247e-01 2.863569e-01 6.280776e-03
## 150 5.611020e-03 6.759990e-01 3.142536e-01 4.136375e-03
## 387 1.917085e-03 6.979105e-01 2.914098e-01 8.762634e-03
## 385 4.267716e-02 4.994289e-01 3.131752e-01 1.447188e-01
## 416 4.641324e-02 6.073057e-01 2.846620e-01 6.161905e-02
## 343 1.360625e-03 6.726636e-01 3.019517e-01 2.402409e-02
## 193 1.993249e-04 6.723411e-01 3.082245e-01 1.923511e-02
## 23  2.199083e-04 6.980608e-01 2.939742e-01 7.745094e-03
## 293 4.013570e-02 6.672099e-01 2.907242e-01 1.930149e-03
## 360 1.898569e-03 6.766901e-01 3.136045e-01 7.806888e-03
## 362 1.362160e-04 6.625304e-01 3.290792e-01 8.254152e-03
## 239 7.094346e-04 6.863651e-01 3.050037e-01 7.921790e-03
## 485 3.635541e-03 7.387653e-01 2.523257e-01 5.273481e-03
## 22  1.856873e-03 7.064166e-01 2.863188e-01 5.407808e-03
## 495 4.828312e-03 7.292113e-01 2.628306e-01 3.129819e-03
## 184 1.748350e-03 6.434321e-01 3.360066e-01 1.881295e-02
## 15  1.512667e-03 7.631449e-01 2.329966e-01 2.345811e-03
## 144 3.572737e-04 6.922297e-01 2.927176e-01 1.469542e-02
## 275 2.183420e-04 6.802387e-01 2.950187e-01 2.452428e-02
## 459 2.577924e-03 7.121220e-01 2.810270e-01 4.273032e-03
## 306 7.962593e-04 6.985340e-01 2.935318e-01 7.137956e-03
## 277 1.761739e-04 6.709789e-01 3.205881e-01 8.256910e-03
## 48  3.220751e-04 7.311028e-01 2.569166e-01 1.165852e-02
## 319 5.213961e-02 6.959143e-01 2.477818e-01 4.164345e-03
## 434 2.290520e-03 7.000809e-01 2.928703e-01 4.758253e-03
## 323 8.966418e-04 6.735777e-01 3.133458e-01 1.217978e-02
## 141 5.526174e-03 7.976124e-01 1.953708e-01 1.490720e-03
## 146 6.611375e-04 7.211915e-01 2.687865e-01 9.360827e-03
## 444 3.995920e-04 7.061137e-01 2.801552e-01 1.333154e-02
## 404 2.823120e-04 6.913540e-01 2.934888e-01 1.487491e-02
## 286 9.496556e-03 6.499256e-01 3.282970e-01 1.228084e-02
## 471 8.572356e-04 6.840475e-01 3.100591e-01 5.036161e-03
## 425 1.160699e-03 6.798781e-01 3.148528e-01 4.108377e-03
## 341 9.414610e-02 5.034331e-01 3.852639e-01 1.715692e-02
## 262 6.827967e-03 6.438523e-01 3.270533e-01 2.226652e-02
## 310 9.097326e-03 6.782001e-01 3.045623e-01 8.140239e-03
## 75  4.664830e-04 5.817651e-01 4.126549e-01 5.113504e-03
## 226 2.045334e-04 6.918093e-01 2.996987e-01 8.287448e-03
## 470 8.261073e-04 6.916690e-01 2.939921e-01 1.351280e-02
## 364 7.104245e-04 6.942946e-01 2.972072e-01 7.787747e-03
## 213 1.365755e-03 8.495232e-01 1.449958e-01 4.115284e-03
## 417 1.265812e-04 6.896946e-01 2.986828e-01 1.149602e-02
```

```
## 329 1.779194e-02 6.455338e-01 3.318642e-01 4.810102e-03
## 386 1.990167e-04 6.512052e-01 3.420568e-01 6.539037e-03
## 143 3.826829e-02 5.441640e-01 3.771778e-01 4.038991e-02
## 209 1.136245e-03 6.825869e-01 3.128597e-01 3.417184e-03
## 260 7.643665e-05 6.486899e-01 3.449109e-01 6.322717e-03
## 56  1.663681e-04 9.174350e-01 5.068874e-02 3.170987e-02
## 129 3.353977e-02 6.638757e-01 2.916696e-01 1.091492e-02
## 108 8.895973e-04 7.214724e-01 2.730357e-01 4.602247e-03
## 28  3.400132e-05 6.821760e-01 2.916822e-01 2.610785e-02
## 117 2.270072e-04 6.587476e-01 3.336432e-01 7.382251e-03
## 466 2.050926e-05 7.524857e-01 2.384772e-01 9.016633e-03
## 322 7.128358e-04 4.739877e-01 4.866551e-01 3.864432e-02
## 464 1.803792e-03 6.550325e-01 3.327923e-01 1.037135e-02
## 467 1.754134e-03 7.356260e-01 2.597629e-01 2.856975e-03
## 460 5.297334e-05 6.807046e-01 3.027018e-01 1.654061e-02
## 366 7.258573e-03 6.935912e-01 2.897794e-01 9.370834e-03
## 441 4.747185e-04 6.937694e-01 2.936442e-01 1.211167e-02
## 45  1.240858e-04 6.990426e-01 2.893781e-01 1.145521e-02
## 168 2.707278e-02 6.646711e-01 3.037423e-01 4.513806e-03
## 258 9.247050e-04 6.983389e-01 2.903039e-01 1.043257e-02
## 52  7.560497e-04 6.803568e-01 3.137161e-01 5.171001e-03
## 349 5.463645e-01 2.946459e-01 1.558991e-01 3.090437e-03
## 321 1.476928e-02 5.797852e-01 3.355310e-01 6.991450e-02
## 70  2.308639e-03 7.413447e-01 2.512251e-01 5.121638e-03
## 8   3.953551e-03 7.052778e-01 2.835356e-01 7.233097e-03
## 11  1.878151e-03 7.065237e-01 2.862067e-01 5.391512e-03
## 72  5.100740e-05 6.559362e-01 3.087148e-01 3.529800e-02
## 442 9.776786e-04 7.307851e-01 2.624461e-01 5.791134e-03
## 90  8.744962e-04 6.722959e-01 3.200748e-01 6.754735e-03
## 481 9.801783e-05 7.132327e-01 2.735133e-01 1.315601e-02
## 1   3.529832e-03 7.036898e-01 2.849844e-01 7.795971e-03
## 243 3.488193e-03 6.717994e-01 3.217660e-01 2.946395e-03
## 132 4.494490e-04 6.656015e-01 3.208249e-01 1.312415e-02
## 27  1.538901e-04 6.885495e-01 2.945410e-01 1.675563e-02
## 4   1.674026e-04 6.755449e-01 3.102845e-01 1.400318e-02
## 279 4.670762e-05 5.952645e-01 4.015602e-01 3.128637e-03
## 265 4.657138e-04 7.211336e-01 2.711119e-01 7.288742e-03
## 111 6.502162e-03 8.048664e-01 1.858221e-01 2.809324e-03
## 324 2.636694e-02 6.226259e-01 3.121412e-01 3.886589e-02
## 73  2.588184e-03 6.398408e-01 3.277097e-01 2.986138e-02
## 131 2.761604e-02 6.847049e-01 2.828009e-01 4.878164e-03
## 196 1.208331e-01 5.875886e-01 2.884858e-01 3.092499e-03
## 228 4.911776e-04 6.993033e-01 2.929845e-01 7.220998e-03
## 419 6.488261e-04 6.720772e-01 3.154053e-01 1.186874e-02
## 94  6.238074e-04 5.101993e-01 3.797900e-01 1.093870e-01
## 44  2.879258e-02 7.476202e-01 2.187186e-01 4.868669e-03
## 327 1.025741e-04 6.234131e-01 3.721263e-01 4.358018e-03
## 120 2.880120e-04 6.789133e-01 3.034009e-01 1.739789e-02
## 280 2.196854e-02 6.880808e-01 2.858106e-01 4.140066e-03
## 418 2.522026e-04 7.031767e-01 2.852016e-01 1.136954e-02
## 373 3.927403e-04 7.790548e-01 2.077796e-01 1.277278e-02
## 99  4.499937e-03 8.064696e-01 1.878722e-01 1.158239e-03
## 234 4.218710e-04 7.008987e-01 2.906981e-01 7.981274e-03
## 282 2.006168e-04 6.705308e-01 3.244937e-01 4.774857e-03
```

```
## 352 2.350881e-01 3.935268e-01 2.250001e-01 1.463850e-01
## 42  2.543375e-05 6.569398e-01 3.339676e-01 9.067145e-03
## 395 2.571193e-03 7.219135e-01 2.713324e-01 4.182832e-03
## 154 1.392198e-03 6.990222e-01 2.902172e-01 9.368453e-03
## 109 1.435763e-03 6.876305e-01 3.051549e-01 5.778866e-03
## 21  3.520128e-03 6.760232e-01 3.127074e-01 7.749253e-03
## 96  3.316844e-04 6.994416e-01 2.915922e-01 8.634454e-03
## 80  2.544388e-04 3.776414e-04 8.195013e-04 9.985484e-01
## 39  5.543948e-02 4.468124e-01 3.153932e-01 1.823549e-01
## 136 5.575138e-03 6.990782e-01 2.797374e-01 1.560926e-02
## 453 6.919264e-03 7.416467e-01 2.467688e-01 4.665287e-03
## 223 2.782902e-03 6.892223e-01 2.967379e-01 1.125688e-02
## 365 4.826355e-04 7.385043e-01 2.566561e-01 4.356891e-03
## 211 4.965534e-05 6.723760e-01 2.933722e-01 3.420207e-02
## 114 3.011809e-02 6.796805e-01 2.794697e-01 1.073173e-02
## 261 4.444521e-04 6.643713e-01 3.209339e-01 1.425033e-02
## 118 5.235339e-04 7.655089e-01 2.279970e-01 5.970532e-03
## 221 1.598054e-04 7.744316e-01 2.172628e-01 8.145784e-03
## 192 6.984189e-03 7.176765e-01 2.697316e-01 5.607661e-03
## 163 1.050105e-03 6.941409e-01 3.004884e-01 4.320557e-03
## 443 3.603895e-04 7.892671e-01 2.060204e-01 4.352049e-03
## 458 1.464811e-03 6.913763e-01 3.005576e-01 6.601293e-03
## 475 2.155354e-05 6.685332e-01 2.933405e-01 3.810474e-02
## 426 1.044432e-03 8.334255e-01 1.637524e-01 1.777653e-03
## 482 1.061439e-04 6.711871e-01 2.989538e-01 2.975296e-02
## 194 8.509494e-03 6.675826e-01 3.184104e-01 5.497470e-03
## 77  1.673133e-04 6.763382e-01 3.019869e-01 2.150760e-02
## 399 4.429927e-03 6.898631e-01 3.007341e-01 4.972847e-03
## 356 4.669476e-04 6.807026e-01 3.133390e-01 5.491452e-03
## 437 1.191190e-04 6.796258e-01 3.081055e-01 1.214961e-02
## 240 8.218090e-04 7.029621e-01 2.892831e-01 6.932945e-03
## 238 2.216197e-03 7.819883e-01 1.804562e-01 3.533929e-02
## 204 6.643238e-05 6.745484e-01 3.161932e-01 9.191966e-03
## 245 5.615035e-04 7.032549e-01 2.855426e-01 1.064102e-02
## 267 1.166161e-03 6.772471e-01 2.989274e-01 2.265936e-02
## 20  9.366253e-04 7.162059e-01 2.783972e-01 4.460351e-03
## 423 5.992643e-01 2.555004e-01 1.379783e-01 7.257069e-03
## 350 1.304056e-03 6.753286e-01 3.147990e-01 8.568278e-03
## 185 1.046879e-01 4.751703e-01 3.068925e-01 1.132493e-01
##
## $terms
## train_raw.df$Severity ~ dist + temp + windchill + humidity +
##     pressure + visibility + windspeed + precip
## attr(,"variables")
## list(train_raw.df$Severity, dist, temp, windchill, humidity,
##     pressure, visibility, windspeed, precip)
## attr(,"factors")
##                       dist temp windchill humidity pressure visibility
## train_raw.df$Severity    0    0         0        0        0          0
## dist                     1    0         0        0        0          0
## temp                     0    1         0        0        0          0
## windchill                0    0         1        0        0          0
## humidity                 0    0         0        1        0          0
## pressure                 0    0         0        0        1          0
```

```
## visibility               0    0         0        0        0          1
## windspeed                0    0         0        0        0          0
## precip                   0    0         0        0        0          0
##                        windspeed precip
## train_raw.df$Severity          0      0
## dist                           0      0
## temp                           0      0
## windchill                      0      0
## humidity                       0      0
## pressure                       0      0
## visibility                     0      0
## windspeed                      1      0
## precip                         0      1
## attr(,"term.labels")
## [1] "dist"       "temp"       "windchill"  "humidity"   "pressure"
## [6] "visibility" "windspeed"  "precip"
## attr(,"order")
## [1] 1 1 1 1 1 1 1 1
## attr(,"intercept")
## [1] 1
## attr(,"response")
## [1] 1
## attr(,".Environment")
## <environment: R_GlobalEnv>
## attr(,"predvars")
## list(train_raw.df$Severity, dist, temp, windchill, humidity,
##     pressure, visibility, windspeed, precip)
## attr(,"dataClasses")
## train_raw.df$Severity                 dist                 temp
##             "numeric"             "numeric"             "numeric"
##             windchill              humidity              pressure
##             "numeric"             "numeric"             "numeric"
##            visibility             windspeed                precip
##             "numeric"             "numeric"             "numeric"
##
## $call
## lda(formula = train_raw.df$Severity ~ ., data = train_raw.df,
##     CV = TRUE)
##
## $xlevels
## named list()
```

```
head(num.lda2$class)
```

```
## [1] 4 2 2 2 2 2
## Levels: 1 2 3 4
```

```
#the Maximum a Posteriori Probability (MAP) classification (a factor)
#posterior: posterior probabilities for the classes.

head(num.lda2$posterior,3)
```

```
##                1         2         3         4
## 413 0.0751950424 0.0695290 0.0324905 0.82278545
## 284 0.0001048764 0.7063073 0.2808752 0.01271262
```

```
## 248 0.0007720173 0.6722520 0.3124529 0.01452309
```

```r
# Partition plots
library("klaR")
#partimat(Severity~.,data=train_raw.df,method="lda")
```