

# Reconhecimento de Padrões-

## Exercício prático de reconhecimento facial e extração de características

2 de outubro de 2019

### 1 Introdução

Este exercício consiste avaliar o desempenho de um classificador na tarefa de reconhecer imagens de faces humanas. A avaliação deve ser feita através da taxa de acerto do classificador *Bayesiano*. A base de dados a ser utilizada será a de faces *Olivetti*, neste exercício também será utilizada uma técnica de redução de dimensionalidade na base de dados, o PCA, apresentado em sala de aula.

### 2 Base de faces Olivetti

A base de dados Olivetti pode ser encontrada no pacote *RnavGraphImageData*, disponível do repositório do R. Esta base é composta por imagens frontais de faces de 40 pessoas diferentes, para cada pessoa foram capturadas 10 imagens com variações leves de expressões faciais e posicionamento. No total a base contém 400 imagens de resolução de 64x64 pixels. A base de dados é representada por uma matriz de 4096 colunas, onde os pixels das imagens 2D estão dispostos de forma unidimensional nas linhas da matriz da base de dados. No código abaixo a base de dados é carregada para a variável *faces*.

```
> rm(list = ls())
> require(RnavGraphImageData)
> # Carregando a Base de dados
> data( faces )
> faces <- t( faces )
```

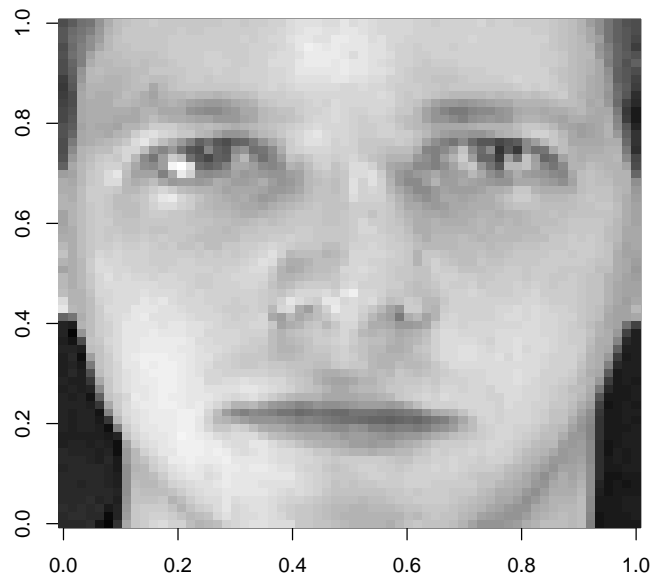
Para visualizar a imagem de cada amostra da base utilize a função *MostraImagem*, codificada abaixo:

```
> MostraImagem <- function( x )
+ {
+   rotate <- function(x) t( apply(x, 2, rev) )
```

```

+   img <- matrix( x, nrow=64 )
+   cor <- rev( gray(50:1/50) )
+   image( rotate( img ), col=cor )
+ }
> MostraImagem( faces[1,] )

```



Esta base de dados não possui o vetor de rótulos, porém sabemos que cada classe possui 10 amostras e estão dispostas sequencialmente. O código abaixo gera o vetor de rótulos  $y$ . Como existem 40 classes o vetor contém valores de rótulos variando entre 1 e 40.

```

> #Gerando os rótulos
> y <- NULL
> for(i in 1:nrow(faces) )
+ {
+   y <- c( y, ((i-1) %% 10) + 1 )
+ }

```

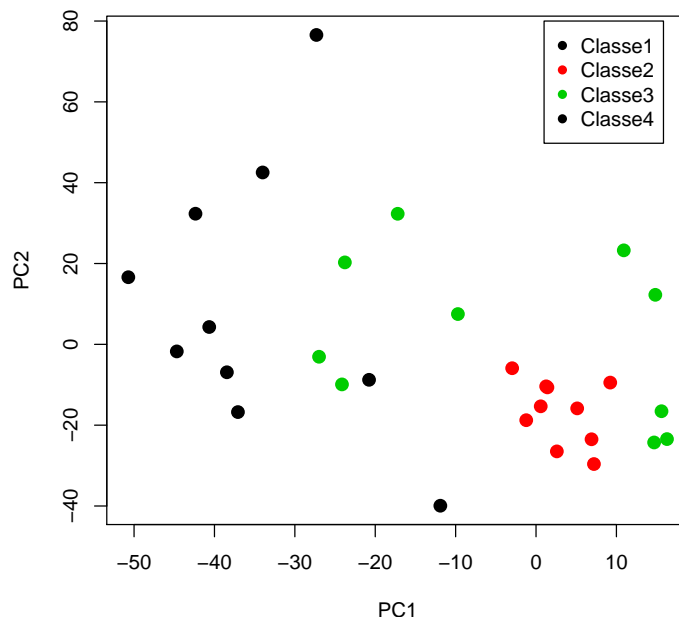
### 3 Redução de dimensionalidade

As imagens da base de dados Olivetti possuem dimensão de 64 x 64 pixels, ou seja, cada padrão possuem 4096 atributos. Para facilitar a classificação deve-se

realizar a redução de dimensionalidade dos dados. Para isto utilize o algoritmo de Análise de Componentes Principais (PCA - *Principal Component Analysis*). No exercício, o aluno deverá encontrar a menor dimensão obtida e gerar os resultados do classificador Bayesiano Gaussiano Multivariado para ela. Algumas rotinas que implementam o PCA podem precisar que as colunas de atributos sejam nomeadas. Se for preciso, para gerar nomes para cada coluna utilize o código abaixo:

```
> # Nomeando os atributos
> nomeColunas <- NULL
> for(i in 1:ncol(faces) )
+ {
+   nomeColunas <- c(nomeColunas, paste("a", as.character(i), sep=".") )
+ }
> colnames(faces) <- nomeColunas
> rownames(faces) <- NULL
```

A Figura abaixo apresenta as 30 primeiras amostras da base de dados representadas em 2 dimensões PCA. Estas dimensões correspondem aos 2 primeiros componentes principais após a aplicação da redução de dimensionalidade com o PCA. Observe que as amostras da classe 3 estão mais espalhadas no espaço do que as das Classes 1 e 2. Por inspeção visual observa-se que a pessoa observada nas imagens aparece em algumas imagens com óculos e em outras sem óculos.



## 4 Projeto do Experimento

Os padrões da base de dados devem ser divididos em 2 partes: dados de treinamento e dados de teste. Recomenda-se utilizar 50% para treinamento e 50% para testes. Esta divisão dos dados deve ser feita de forma aleatória. Cada experimento deve ser repetido 10 vezes e em cada execução uma nova divisão aleatória dos dados deve ser feita. O resultado de classificação deve ser apresentado como a média das 10 execuções junto com o desvio padrão da média.

## 5 Análise de Resultados

Deve-se gerar uma tabela com a taxa de acerto do classificador apresentado em percentual de acerto. Apresente resultados também na forma de matriz de confusão. Como a base de dados possui 40 classes, a matriz de confusão deve conter dimensão 40x40. Analise a matriz de confusão e discuta qual fator da imagem pode estar afetando o desempenho dos classificadores.