



UNIVERSITAT  
POLITÈCNICA  
DE VALÈNCIA

Escola Tècnica Superior d'Enginyeria Informàtica



# Práctica 1. Aplicación de PCA y LDA a OCR

Percepción (PER)

Curso 2018/2019

Departamento de Sistemas Informáticos y Computación

# Índice

1. Objetivos
2. Trabajo previo, carga y visualización de datos
3. Implementación PCA
4. Implementación LDA
5. Integración de PCA y LDA en un clasificador k-NN
6. Evaluación

# Objetivos

- Implementar las técnicas de reducción de dimensionalidad PCA y LDA estudiadas en teoría.
- Visualizar los vectores de proyección PCA y LDA.
- Realizar proyecciones lineales de los datos utilizando las matrices de proyección obtenidas mediante PCA y LDA.
- Estudiar el efecto que tienen las técnicas de reducción de dimensionalidad PCA y LDA sobre la tasa de error de un clasificador basado en vecinos más cercanos.

# Implementación PCA

- Algoritmo en transparencia 3.35 de teoría
- $X_{d \times n} = \{\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_n\}$  son vectores columna
- $\bar{\mathbf{x}} = \frac{1}{n} \sum_{i=1}^n \mathbf{x}_i$  es el vector media
- La matriz de covarianzas se puede expresar como

$$\Sigma = \frac{1}{n} A A^t$$

donde  $A_{d \times n} = (\mathbf{x}_1 - \bar{\mathbf{x}}, \mathbf{x}_2 - \bar{\mathbf{x}}, \dots, \mathbf{x}_n - \bar{\mathbf{x}})$

- Utiliza la función `eig()`, `sort()` y `diag()` para obtener los vectores propios ordenados por valor propio
- Visualiza los primeros vectores propios para comprobar que son correctos

# Implementación LDA

- Algoritmo en transparencia 3.53 de teoría
- Calcula la media (global) de los datos  $\bar{\mathbf{x}}$
- Inicializa  $S_b = S_w = 0$
- Para cada clase  $c$ :
  - Calcula la media  $\bar{\mathbf{x}}_c$
  - Calcula  $S_b = S_b + n_c(\bar{\mathbf{x}}_c - \bar{\mathbf{x}})(\bar{\mathbf{x}}_c - \bar{\mathbf{x}})^t$
  - Calcula la matriz de covarianzas  $\Sigma_c$
  - Calcula  $S_w = S_w + \Sigma_c$
- Utiliza  $\text{eig}(S_b, S_w)$  para calcular valores propios generalizados
- Obtén los vectores propios ordenados al igual que en PCA
- Visualiza los primeros vectores propios para comprobar que son correctos

# Integración de PCA y LDA en un clasificador k-NN (I)

- Script octave con la siguiente interfaz

```
#!/usr/bin/octave -qf

if (nargin!=7)
    printf("Usage: pcaexp.m <trdata> <trlabs> <tedata> <telabs> <mink> <stepk> <maxk>\n");
    exit(1);
end

arg_list=argv();
trdata=arg_list{1};
trlabs=arg_list{2};
tedata=arg_list{3};
telabs=arg_list{4};
mink=str2num(arg_list{5});
stepk=str2num(arg_list{6});
maxk=str2num(arg_list{7});

load(trdata);
load(trlabs);
load(tedata);
load(telabs);
```

# Integración de PCA y LDA en un clasificador k-NN (y II)

- Llamada a PCA o LDA con training para obtención de vectores propios
- Proyección de training  $XR$  y test  $YR$  con los  $k$  primeros vectores propios
- Llamada a `err=knn(XR,xl,YR,y1,kk)` con  $kk=1$
- Imprime valor  $k$  y tasa de error de clasificación en formato de tabla
- Representación gráfica (gnuplot)

# Evaluación

- Esta práctica supone un 15 % del total de la nota de la asignatura (1.5 puntos)
- Para la evaluación se tendrán en cuenta dos actividades:
  1. Memoria del trabajo realizado:
    - Código comentado de `pca.m` y `lda.m` (0.5 puntos)
    - Gráficas de resultados comparativos que aparecen en el boletín (0.25 puntos)
    - Comentario de los resultados (0.25 puntos)
    - Entrega hasta el **29 de abril** por tarea en PoliformaT
  2. “Competición” (0.5 puntos máximo, dependiendo del resultado obtenido)
    - Se proporciona un nuevo conjunto de datos en el mismo formato
    - Objetivo: Buscar la combinación de reducción de dimensionalidad (PCA, LDA o PCA+LDA) para un clasificador k-NN que minimice el error
    - Se realizará durante la última sesión de prácticas (**16 de abril**)