

APPLICATIONS :

① Compresión individual de imágenes.



Obs: $\text{rank}(A) \leq 1500$.

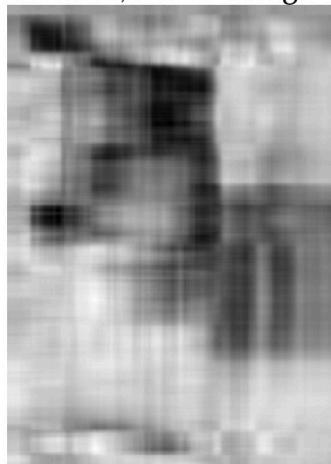
$$\begin{aligned}
 B_k &= U_k \Sigma_k V_k^T = \left[\sum_{j=1}^k a_i^T \cdot v_j \cdot v_j^T \right] \\
 &= A V_k \cdot V_k^T \\
 &= \sum_{j=1}^k \begin{bmatrix} a_1 \cdot v_j \cdot v_j^T \\ \vdots \\ a_n \cdot v_j \cdot v_j^T \end{bmatrix} \\
 &= \sum_{j=1}^k A \cdot v_j \cdot v_j^T
 \end{aligned}$$

Idee: utilligan B_k como imagen comprimida:

Original



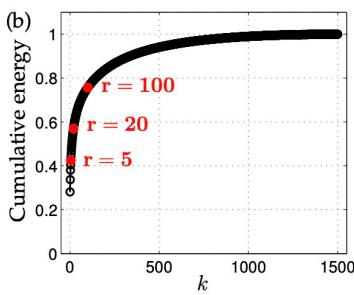
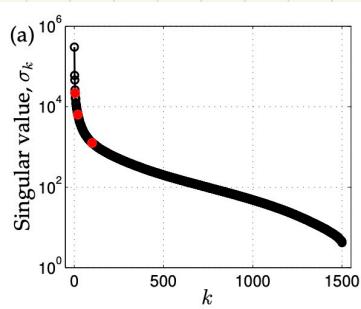
$r = 5, 0.57\% \text{ storage}$



$r = 20, 2.33\% \text{ storage}$



$r = 100, 11.67\% \text{ storage}$



PCA (Principal Component Analysis)

Input: un dataset of "feature vectors":

$$\mathbf{x} = (\dots \quad \quad)$$

$$\mathbf{x}_2 = (\dots \quad \quad \quad)$$

⋮

$$\mathbf{x}_n = (\dots \quad \quad \quad \quad)$$

$\star \rightarrow$ caracteriza "individuo" "ítem";

Objetivo. Encuentra una "base adaptada a los datos":

- los datos son "sparse" en esta base
- la base está ordenada por relevancia.

- los primeros elementos de la base (los CP) "explican" bien los datos.

Sol: 1) Centrar los datos $\hat{\mathbf{x}}_i = \mathbf{x}_i - \bar{\mathbf{x}}$

$$\bar{\mathbf{x}} = \frac{1}{N} \sum \mathbf{x}_i$$

2) $A = \begin{bmatrix} \hat{\mathbf{x}}_1 \\ \vdots \\ \hat{\mathbf{x}}_N \end{bmatrix} \rightarrow A = U \Sigma V^T$

base adaptada: $\{v_1, v_2, \dots, v_r\}$
 ↳ primera CP.
 ↳ segunda CP.

- 3) Calcular los coeficientes de los dígitos en "base V".
- 4) Utilizar las componentes principales para clasificar.

Ej: 1. Cancer y genética.

Data set: (x_1, l) \rightarrow etiqueta

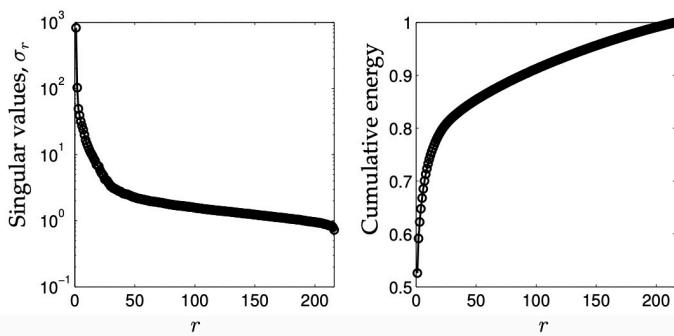
$$\begin{matrix} & & & \\ \vdots & & & \\ (x_{216}, l) & & & l \in \{\text{oncovo, sano}\} \end{matrix}$$

$$x_i = (g_1, g_2, \dots, g_{4000})$$

$$g_i = \begin{cases} 1 & \text{si el gen } i \text{ se expresa} \\ 0 & \text{si no.} \end{cases}$$

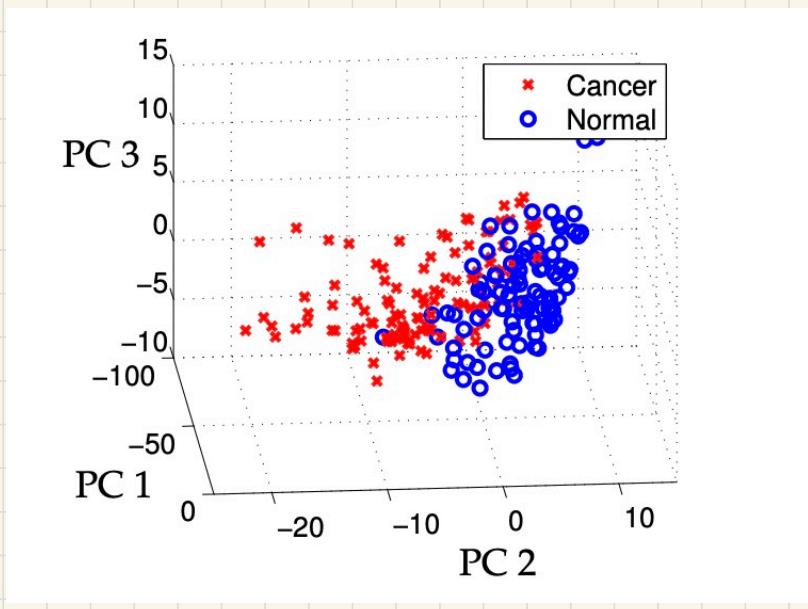
¿Cómo extraer la info relevante?

$$A = U \Sigma V^T$$



clustering (classification)

con $\{v_1, v_2, v_3\}$.



Eigen faces :



Construimos una matriz X

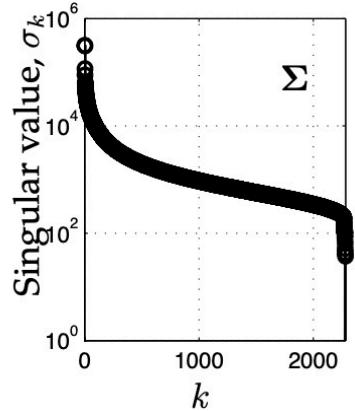
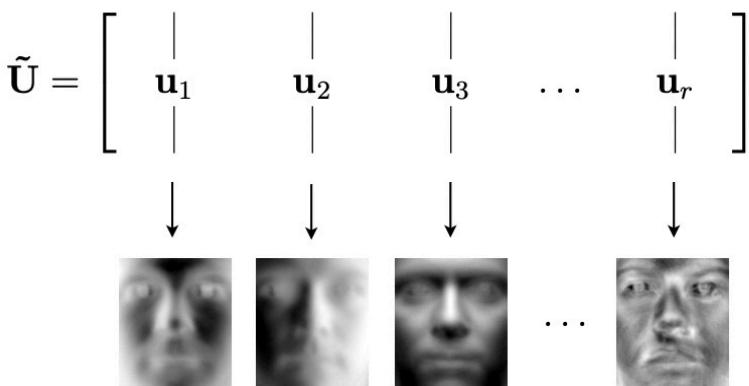
$$\begin{array}{cccc}
 \text{Person 1} & \text{Person 2} & \text{Person 3} & \dots \\
 \downarrow & \downarrow & \downarrow & \dots \\
 \mathbf{x}_1 & \dots & \mathbf{x}_{k_2} & \dots & \mathbf{x}_{k_3} & \dots & \mathbf{x}_m \\
 \mathbf{X} = \left[\begin{array}{c|c|c|c|c|c}
 \hline
 & & & & & \\
 \hline
 \end{array} \right] & & & & & & \\
 \end{array}$$

Average face

→ Imágenes centradas.

(También podrían ser las imágenes con los filtros).

$$X = U \Sigma V^T$$



Eigenfaces

Reconstruction :

$$\hat{x}_i(k) = U_k U_k^T \cdot \tilde{x}_i + \bar{x}$$

Test image



$r = 25$



$r = 50$



$r = 100$



$r = 200$



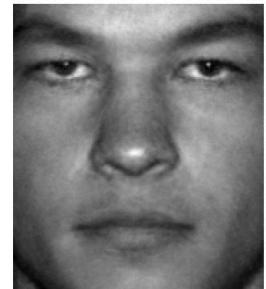
$r = 400$



$r = 800$



$r = 1600$



i) ¿Podemos reconstruir imágenes de otro tipo?

Test image



$r = 25$



$r = 50$



$r = 100$



$r = 200$



$r = 400$



$r = 800$



$r = 1600$



i Podemos usarlo para distinguir entre cara ?

