

# CLUSTERING

## MINERÍA DE DATOS

Grado en Ingeniería Informática de Gestión y Sistemas de Información  
Escuela de Ingeniería de Bilbao (EIB)

Alicia Pérez

[alicia.perez@ehu.eus](mailto:alicia.perez@ehu.eus)

Departamento de Lenguajes y Sistemas Informáticos



## Índice

- 1 Introducción
- 2 Taxonomía
- 3 Clustering particional k-medias
- 4 Clustering particional: Algoritmo EM
- 5 Clustering jerárquico
- 6 Clustering basado en densidad
- 7 Evaluación
- 8 Conclusiones

## Introducción

Clasificación Supervisada vs. No-Supervisada

### Clasificación

Muestras:  $\{(\mathbf{x}^{(i)}, \mathbf{c}^{(i)})\}_{i=1}^N$

Clases:  $\mathcal{C} = \{C_1 \dots C_m\}$

Técnica predictiva

### Clustering

Muestras:  $\{\mathbf{x}^{(i)}\}_{i=1}^N$

Clusters:  $\mathcal{G} = \{G_1 \dots G_m\}$

Técnica descriptiva (exploratoria)

## Introducción

Clasificación Supervisada vs. No-Supervisada

### Clasificación supervisada:

- Se dispone de un conjunto de instancias previamente clasificadas
- A partir de ese conjunto de muestras se desea inferir un modelo que permite **predecir** la clase asociada a una nueva instancia

### Clasificación no-supervisada:

- Se dispone de un conjunto de instancias
- Se desea **agrupar** las instancias en clusters (sub-conjuntos), de modo que ...
  - ▶ ... las instancias del mismo cluster presenten similitud interna fuerte
  - ▶ ... las instancias de distintos clusters presenten disimilitud fuerte

## Introducción

Objetivos

Objetivo de la clasificación no-supervisada [Bandyopadhyay and Saha, 2013, Chap. 1]:

- Descubrir agrupamientos naturales para un conjunto de instancias no clasificadas
- Ofrecer una descripción de los datos, en términos de similitud/disimilitud buscando:
  - ▶ Homogeneidad intra-cluster
  - ▶ Separabilidad inter-cluster

GIIGSI (EIB)

Minería de Datos

Clustering

6 / 18

## Introducción

Objetivos

Observaciones:

- **Clustering:** dado un conjunto de  $N$  instancias no-etiquetadas, determinar  $k$  agrupaciones naturales (la mejor agrupación según un criterio dado).
- ¿Es un problema computable? ¿Cuántas agrupaciones posibles consideraría una búsqueda exhaustiva? EJERCICIO: calcular cardinalidad.
  - ▶ Una búsqueda exhaustiva no es computacionalmente abordable.
  - ▶ Recurriremos a heurísticos.
- ¿Cómo evaluar la calidad de las agrupaciones obtenidas?
  - ▶ ¿Esta agrupación ayuda a estudiar los datos mejor que si no estuvieran agrupados?
  - ▶ ¿Ayuda para detectar instancias atípicas (posiblemente erróneas, outliers)?
- Factores determinantes:
  - ▶ La métrica de similitud/disimilitud
  - ▶ Número de clusters

GIIGSI (EIB)

Minería de Datos

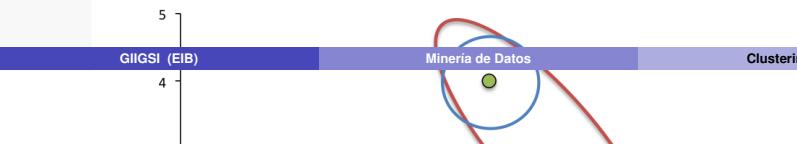
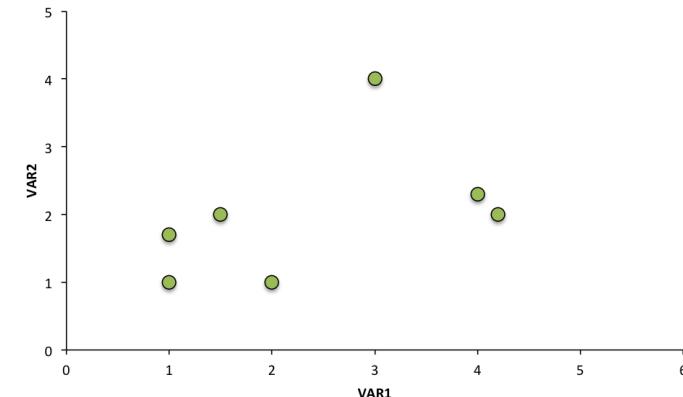
Clustering

8 / 18

## Introducción

Objetivos

Hay que definir el criterio: **similitud/disimilitud**.



## Introducción

Aplicaciones

**Aplicaciones:** [Han et al., 2011]

**Biología:** descubrir taxonomías

**Marketing:** encontrar segmentos de clientes con intereses/objetivos similares; identificar fraude; tasación de viviendas

**Imágenes:** reconstrucción, cuantificación vectorial, identificación de áreas terrestres

**Web:** clasificación de documentos; identificar grupos de usuarios con comportamiento similar.

**Meteorología:** buscar patrones atmosféricos

GIIGSI (EIB)

Minería de Datos

Clustering

9 / 18

## Taxonomía

### Estrategias de clustering: [Bandyopadhyay and Saha, 2013, Chap. 4]

- Jerárquico: (*Hierarchical*) asume una topología (dendograma en particular o árbol en general) que define dependencias entre instancias.
- Particional: (*Partitioning*) agrupar los datos en un número de conjuntos pre-determinado. Se van moviendo las instancias de un cluster a otro y en base a las medidas de similitud/disimilitud. Metodología subyacente: optimización.
- Probabilístico: (*Density-based*) se asume que la densidad de probabilidad condicionada de pertenencia de las instancias a los clusters sigue un tipo de función conocida (e.g. gaussiana) de la cual se desconocen los parámetros que la caracterizan (e.g. media, varianza), los parámetros se pueden estimar (e.g. técnicas como máxima verosimilitud).
- Model-based
- Grid-based
- Soft-computing

Lecturas relacionadas: [Maimon and Rokach, 2005, Bandyopadhyay and Saha, 2013]

## Taxonomía

### Variantes:

- Exclusivo (*exclusive clustering*): una instancia pertenece a un cluster y sólo a uno.

$$\bigcup_{j=1}^m \mathcal{G}_j = \mathcal{X}$$
$$\mathcal{G}_j \cap \mathcal{G}_i = \emptyset \quad \forall i \neq j$$

- Probabilístico (*probabilistic clustering*): se define una función (distribución de probabilidad) de pertenencia de una instancia a cada cluster. Casos particulares:
  - Solapado (*overlapping clustering*): una instancia puede pertenecer a varios clusters (a algunos clusters con probabilidad 0 y a los demás con la misma probabilidad)
  - Difuso (*fuzzy clustering*): la función de pertenencia no es una distribución de probabilidad (no tiene por qué estar definida en el rango [0, 1] y no tiene por qué cumplir el teorema de completitud)

## Clustering particional k-medias

Motivación: cuantificación vectorial

### Motivación

#### Problema-Ejemplo:

- Imagen en fichero codificada con resolución: 24 bits/pixel
  - $2^{24} \approx 16 \times 10^6$  colores diferentes
- Pantalla con resolución: 8 bits/pixel
  - $2^8 = 256$  colores diferentes

Solución: cuantificación vectorial [Alpaydin, 2010, Sec. 7.3]

- $\mathcal{X} = \{\mathbf{x}^{(1)}, \dots, \mathbf{x}^{(N)}\}$  conjunto de pixels de la imagen del fichero
  - $\mathbf{x}^{(t)}$ : valor (de 24 bits) que toma el pixel  $(t)$
- $\mathcal{M} = \{\mathbf{m}_1, \mathbf{m}_2, \dots, \mathbf{m}_k\}$  conjunto de colores permitidos en la pantalla
  - $\mathbf{m}_i$ : valor (de 8 bits) permitido por la resolución de la pantalla
- Mapeo:  $\mathbf{x}^{(t)} \rightarrow \mathbf{m}_i$

## Clustering particional k-medias

Motivación: cuantificación vectorial



Fuente de la figura:[Bishop, 2006]

## Clustering particional k-medias

Motivación: cuantificación vectorial

Método de cuantificación vectorial:  $f_{cv} : \mathcal{X} \rightarrow \mathcal{M}$

- ¿cuantificación uniforme?
  - ▶ se infra-utiliza el rango permitido con valores no existentes
  - ▶ no permite distinguir valores cercanos frecuentes asignados al mismo intervalo
- ¡buscar mapeo mejor! máximo local, sub-óptimo

- ① Seleccionar code-book: conjunto de vectores de referencia (conjunto de k colores)

\*  $\mathcal{M} = \{\mathbf{m}_1, \mathbf{m}_2, \dots, \mathbf{m}_k\}$  (k=256) ¿Cómo seleccionar el code-book?

- ② Asignar a  $\mathbf{x}^{(t)}$  el code-word más cercano ( $\mathbf{m}_j$ )

$$f_{cv} : \mathcal{X} \rightarrow \mathcal{M} \quad f_{cv}(\mathbf{x}^{(t)}) = \arg \min_{\mathbf{m}_j \in \mathcal{M}} d(\mathbf{x}^{(t)}, \mathbf{m}_j) \quad (1)$$

Distancia Euclídea:  $\mathbf{x}^{(t)} = (x_1^{(t)}, \dots, x_n^{(t)})$ ,  $\mathbf{m}_j = (m_{j1}, \dots, m_{jn})$

$$d(\mathbf{x}^{(t)}, \mathbf{m}_j) = \|\mathbf{x}^{(t)} - \mathbf{m}_j\| = \left[ \sum_{r=1}^n (x_r^{(t)} - m_{jr})^2 \right]^{\frac{1}{2}} \quad (2)$$

## Clustering particional k-medias

Algoritmo: Clustering k-medias

Selección racional del code-book:

- Solución analítica: **irresoluble!**
- Solución heurística de optimización local: **clustering k-medias**

- ▶ Inicialización: seleccionar  $\mathcal{M} = \emptyset$  y  $\mathcal{M}^{nuevo} = \{\mathbf{m}_1, \mathbf{m}_2, \dots, \mathbf{m}_k\}$  aleatorios
- ▶ Proceso iterativo: repetir hasta que  $\mathcal{M} \approx \mathcal{M}^{nuevo}$

- ① Actualizar:  $\mathcal{M} \leftarrow \mathcal{M}^{nuevo}$
- ② Estimar conjunto de etiquetas de pertenencia  $b_j^t$   $1 \leq j \leq k$  para cada uno de las instancias  $\mathbf{x}^{(1)}, \dots, \mathbf{x}^{(t)}, \dots, \mathbf{x}^{(N)}$ :  $b_j^t = \text{¿es } \mathbf{m}_j \text{ el vector de referencia más cercano a } \mathbf{x}^{(t)}?$

$$b_j^t = \begin{cases} 1 & \text{si } \mathbf{m}_j = \arg \min_{\mathbf{m}_r \in \mathcal{M}} d(\mathbf{x}^{(t)}, \mathbf{m}_r) \\ 0 & \text{en otro caso} \end{cases} \quad (6)$$

- ③ Minimizar error: buscar para qué nuevo conjunto  $\mathcal{M}^{nuevo} = \{\mathbf{m}_1^{nuevo}, \dots, \mathbf{m}_k^{nuevo}\}$  se obtiene el mínimo del error total representado en la ecuación (5).

$$\frac{\partial E}{\partial \mathbf{m}_j} = 0 \implies \mathbf{m}_j^{nuevo} = \frac{\sum_{t=1}^N [b_j^t \cdot \mathbf{x}^{(t)}]}{\sum_{t=1}^N b_j^t} \quad (7)$$

Observación:  $\mathbf{m}_j^{nuevo}$  promedio de las instancias ( $\mathbf{x}^{(t)}$   $1 \leq t \leq N$ ) que pertenecen a  $\mathbf{m}_j$

- ▶ Retornar:  $\mathcal{M}^{nuevo}$

## Clustering particional k-medias

Motivación: cuantificación vectorial

- ① **Seleccionar code-book:** de modo que la imagen reconstruida con baja resolución se parezca a la original tanto como sea posible

▶ Error al representar  $\mathbf{x}^{(t)}$  mediante  $\mathbf{m}_j$ : cuadrado de la menor distancia (varianza)

$$e_i^t = \sum_{j=1}^k b_j^t \cdot [d(\mathbf{x}^{(t)}, \mathbf{m}_j)]^2 \quad \text{donde } b_j^t = \delta(f_{cv}(\mathbf{x}^{(t)}), \mathbf{m}_j) \quad (3)$$

$$= [d(\mathbf{x}^{(t)}, f_{cv}(\mathbf{x}^{(t)}))]^2 \quad (4)$$

▶ Error total de reconstrucción: (suma para todos los pixels)

$$E(\mathcal{M} | \mathcal{X}) = \sum_{t=1}^N e_i^t \quad (5)$$

- ② Selección racional del code-book:

- ★ seleccionar el conjunto  $\mathcal{M} = \{\mathbf{m}_j\}_{j=1}^k$  que minimice el error total
- ★ para calcular el error necesitamos conocer el conjunto  $\mathcal{M}$  (porque  $b_j^t$  depende de  $\mathbf{m}_j$ )
- ★ **⇒ problema de optimización irresoluble analíticamente!**

## Clustering particional k-medias

Algoritmo: Clustering k-medias

### Algoritmo: Clustering k-medias

Initialize  $\mathbf{m}_i, i = 1, \dots, k$ , for example, to  $k$  random  $\mathbf{x}^t$

Repeat

For all  $\mathbf{x}^t \in \mathcal{X}$

$$b_i^t \leftarrow \begin{cases} 1 & \text{if } \|\mathbf{x}^t - \mathbf{m}_i\| = \min_j \|\mathbf{x}^t - \mathbf{m}_j\| \\ 0 & \text{otherwise} \end{cases}$$

For all  $\mathbf{m}_i, i = 1, \dots, k$

$$\mathbf{m}_i \leftarrow \sum_t b_i^t \mathbf{x}^t / \sum_t b_i^t$$

Until  $\mathbf{m}_i$  converge

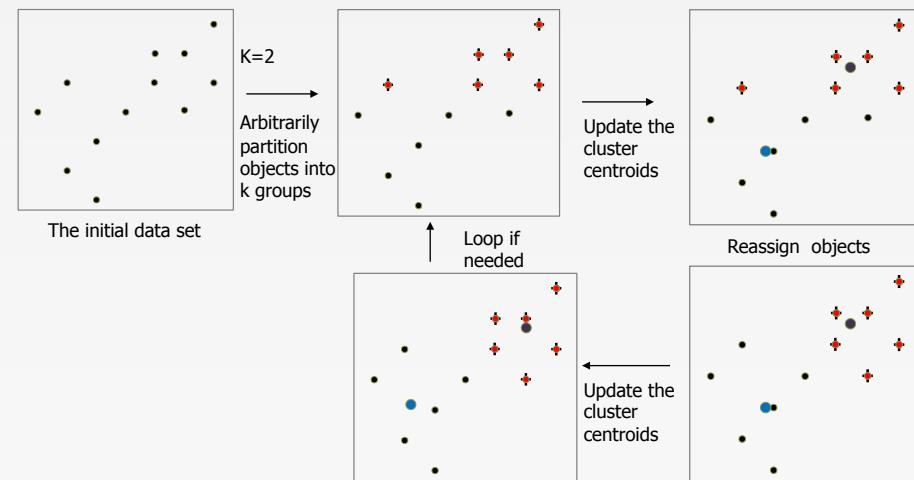
Figura: k-means clustering algorithm

Fuente de la figura: [Alpaydin, 2010, Chap. 7] Ver:

[http://commons.wikimedia.org/wiki/File%3AEM\\_Clustering\\_of\\_Old\\_Faithful\\_data.gif](http://commons.wikimedia.org/wiki/File%3AEM_Clustering_of_Old_Faithful_data.gif)

## Clustering particional k-medias

Algoritmo: Clustering k-medias



Fuente de la figura: [Han et al., 2011]

## Clustering particional k-medias

Conclusiones

Conclusiones algoritmo clustering k-medias:

- **Objetivo:** buscar *code-book* ( $\mathcal{M}$ ) que minimice el error total de reconstrucción

$$\widehat{\mathcal{M}} = \arg \min_{\mathcal{M}} E(\mathcal{M} | \mathcal{X}) \quad (8)$$

- **Algoritmo sub-óptimo:** garantiza que se minimiza el error localmente pero no globalmente. Las soluciones exploradas y el resultado final dependen fuertemente de:

- ▶ **Inicialización**  $\mathcal{M} = \{\mathbf{m}_1, \mathbf{m}_2, \dots, \mathbf{m}_k\}$ 
  - ★ Selección aleatoria: de  $k$  instancias del conjunto o de  $k$  puntos aleatorios del espacio
  - ★ Seleccionar  $\mathbf{m}_0$  como la media de  $\mathcal{X}$ ,  $\mathbf{m}_j = \mathbf{m}_0 + \mathbf{r}_j$  donde  $\mathbf{r}_j$  es un vector aleatorio
  - ★ Calcular las componentes principales, dividir el rango en  $k$  intervalos, inicializar con las medias de los grupos de datos en esos intervalos
- ▶ **Distancia** definición de la función distancia

- **Interpretación intuitiva:** los *clusters* obtenidos se caracterizan por sus medias (*centroídes*)

- **Aplicaciones:** cuantificación vectorial

- ▶ imágenes (OCR)
- ▶ voz (ASR)
- ▶ transmisión de señal (digitalización)
- ▶ ...

## Clustering particional k-medias

Aplicación

### Aplicación: cuantificación vectorial

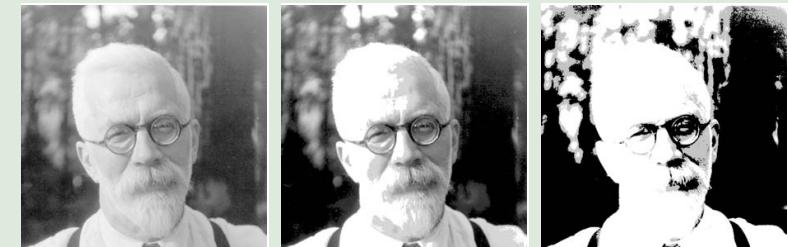


Figura: Imagen de R. A. Fisher (a) original: 8.0 bits/pixel; (b) cuantificación vectorial con *codebook* de 200 vectores de referencia, ratio de compresión: 1.9 bits/pixel; (c) cuantificación vectorial con *codebook* de 4 vectores de referencia, ratio de compresión: 0.50 bits/pixel;

Fuente de la figura: [Hastie et al., 2003]. Sir Ronald A. Fisher (1890-1962) was one of the founders of modern day statistics, to whom we owe maximum likelihood, sufficiency, and many other fundamental concepts.

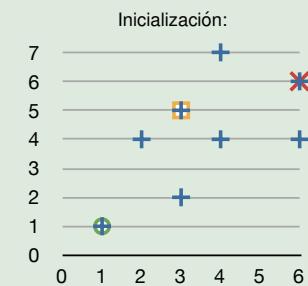
## Clustering particional k-medias

Ejercicio

### Ejercicio

Dado un conjunto de 8 datos caracterizados con 2 atributos, se desea agrupar los datos mediante clustering k-medias con  $k=3$ .

Atributos		
	X <sub>1</sub>	X <sub>2</sub>
1	1	1
2	2	4
3	3	2
4	3	5
5	4	4
6	4	7
7	6	4
8	6	6

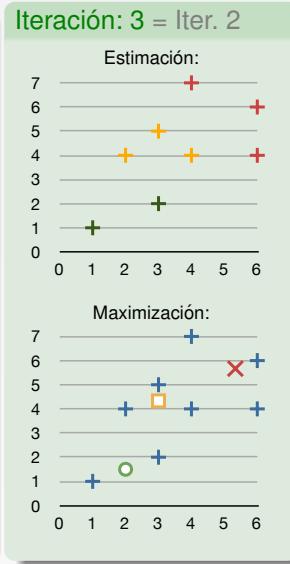
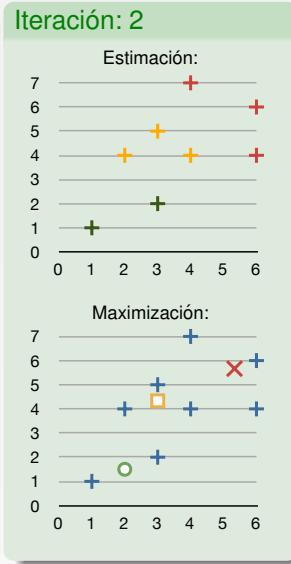
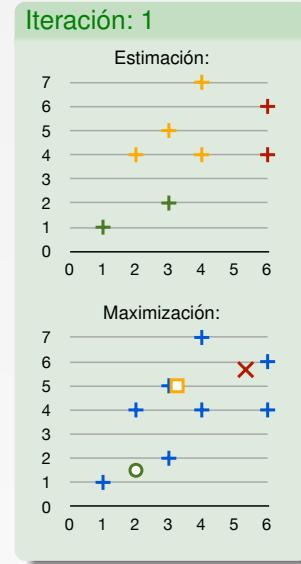


Se pide realizar los siguientes cálculos hasta convergencia:

- 1 Paso E: calcular la matriz de pertenencia y representar gráficamente el resultado
- 2 Paso M: calcular el nuevo codebook y representar gráficamente el resultado

## Clustering particional k-medias

Ejercicio



## Clustering particional: Algoritmo EM

Mixtura de dos Gaussianas

Caso particular: [mixtura de dos gaussianas](#)  
[Hastie et al., 2003] [Bishop, 2006, Chap. 9]

$$Y_1 \approx N(\mu_1, \sigma_1^2) \quad \theta_1 = (\mu_1, \sigma_1^2) \quad (9)$$

$$Y_2 \approx N(\mu_2, \sigma_2^2) \quad \theta_2 = (\mu_2, \sigma_2^2) \quad (10)$$

$$Y = (1 - \Delta) \cdot Y_1 + \Delta \cdot Y_2 \quad \theta = (\mu_1, \sigma_1^2, \mu_2, \sigma_2^2) \quad (11)$$

Donde:  $\Delta \in \{0, 1\}$  siendo  $Pr(\Delta = 1) = \pi$

Adaptar el modelo a las N muestras mediante el algoritmo EM para una mixtura de dos gaussianas.

## Clustering particional: Algoritmo EM

### Expectation Maximization

Algoritmo EM: [Alpaydin, 2010, Sec. 7.4] [Duda et al., 2000, Sec. 3.9] [Hastie et al., 2003, Sec. 8.5]

- Objetivo del [algoritmo k-medias](#): buscar *code-book* ( $\mathcal{M}$ ) que minimice el error total de reconstrucción
- Objetivo del [algoritmo EM](#): maximizar la log-verosimilitud de la muestra
  - ▷ Técnica probabilística
  - ▷ Contexto de aplicación: intervienen dos conjuntos de variables aleatorias:
    - \*  $\mathcal{X}$ : observables (muestras)
    - \*  $\mathcal{Z}$ : ocultas (atributos perdidos y parámetros del modelo subyacente a las muestras)
    - Maximizar log-verosimilitud:  $\mathcal{L}(\Phi | \mathcal{X}, \mathcal{Z})$
  - ▷ Técnica que permite inferir parámetros de la distribución de probabilidad que da cuenta de las instancias (aproximación paramétrica)

## Clustering particional: Algoritmo EM

Mixtura de dos Gaussianas



## Clustering particional: Algoritmo EM

Mixtura de dos Gaussianas

### Algorithm 8.1 EM Algorithm for Two-component Gaussian Mixture.

1. Take initial guesses for the parameters  $\hat{\mu}_1, \hat{\sigma}_1^2, \hat{\mu}_2, \hat{\sigma}_2^2, \hat{\pi}$  (see text).
2. *Expectation Step*: compute the responsibilities

$$\hat{\gamma}_i = \frac{\hat{\pi}\phi_{\hat{\theta}_2}(y_i)}{(1 - \hat{\pi})\phi_{\hat{\theta}_1}(y_i) + \hat{\pi}\phi_{\hat{\theta}_2}(y_i)}, \quad i = 1, 2, \dots, N. \quad (8.42)$$

3. *Maximization Step*: compute the weighted means and variances:

$$\begin{aligned} \hat{\mu}_1 &= \frac{\sum_{i=1}^N (1 - \hat{\gamma}_i)y_i}{\sum_{i=1}^N (1 - \hat{\gamma}_i)}, & \hat{\sigma}_1^2 &= \frac{\sum_{i=1}^N (1 - \hat{\gamma}_i)(y_i - \hat{\mu}_1)^2}{\sum_{i=1}^N (1 - \hat{\gamma}_i)}, \\ \hat{\mu}_2 &= \frac{\sum_{i=1}^N \hat{\gamma}_i y_i}{\sum_{i=1}^N \hat{\gamma}_i}, & \hat{\sigma}_2^2 &= \frac{\sum_{i=1}^N \hat{\gamma}_i(y_i - \hat{\mu}_2)^2}{\sum_{i=1}^N \hat{\gamma}_i}, \end{aligned}$$

and the mixing probability  $\hat{\pi} = \sum_{i=1}^N \hat{\gamma}_i/N$ .

4. Iterate steps 2 and 3 until convergence.

## Clustering particional: Algoritmo EM

Mixtura de dos Gaussianas

Conclusiones: [Alpaydin, 2010, Sec. 7.4] [Bishop, 2006, Sec. 9.3.2]

- Pertenencia al cluster:
  - ▶ K-means clustering:  $b_i^k \in \{0, 1\}$  (bits de pertenencia, *hard label*)
  - ▶ EM:  $\gamma_i \in [0, 1] \in \mathbb{R}$  (probabilidad de pertenencia, *soft label*)
- K-means clustering es un caso particular del EM
  - ▶ aplicado a mixturas de gaussianas
  - ▶ asumiendo que las entradas son idénticas e igualmente distribuidas
  - ▶ con igual varianza
  - ▶ todas las componentes gaussianas tienen la misma probabilidad a-priori
- K-means vs. EM con mixtura de gaussianas general:
  - ▶ K-means clustering: densidad inicial círculo
  - ▶ EM: densidad inicial elipse de forma y orientación arbitraria
- EM es más general que K-means clustering

## Clustering particional: Algoritmo EM

Mixtura de dos Gaussianas

### Ejemplo: mixtura de dos gaussianas

Agrupar los datos en dos clusters asumiendo que han sido generados por dos gaussianas. Datos:  $\mathcal{X} = \{-0.39, 0.12, 0.94, 1.67, 1.76, 2.44, 3.72, 4.28, 4.92, 5.53, 0.06, 0.48, 1.01, 1.68, 1.80, 3.25, 4.12, 4.60, 5.28, 6.22\}$

Iteración	$\hat{\pi}$
1	0.485
5	0.493
10	0.523
15	0.544
20	0.546

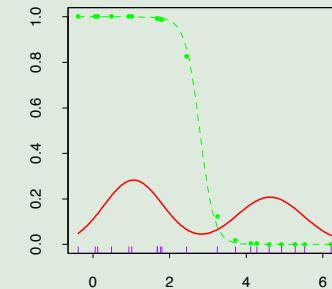
$$\hat{\pi} = 0.546$$

$$\hat{\mu}_1 = 4.62$$

$$\hat{\sigma}_1^2 = 0.87$$

$$\hat{\mu}_2 = 1.06$$

$$\hat{\sigma}_2^2 = 0.77$$



Fuente de la figura: [Hastie et al., 2003]

## Clustering particional: Algoritmo EM

EM caso general

### Algoritmo EM

```

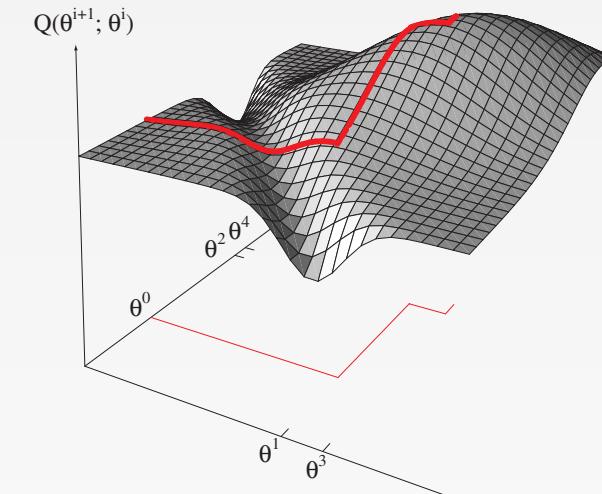
1 begin initialize  $\theta^0, T, i = 0$ 
2   do  $i \leftarrow i + 1$ 
3     E step : compute  $Q(\theta; \theta^i)$ 
4     M step :  $\theta^{i+1} \leftarrow \arg \max_{\theta} Q(\theta; \theta^i)$ 
5     until  $Q(\theta^{i+1}; \theta^i) - Q(\theta^i; \theta^{i-1}) \leq T$ 
6   return  $\hat{\theta} \leftarrow \theta^{i+1}$ 
7 end

```

Fuente de la figura: [Duda et al., 2000]

## Clustering particional: Algoritmo EM

EM caso general



Fuente de la figura: [Duda et al., 2000]

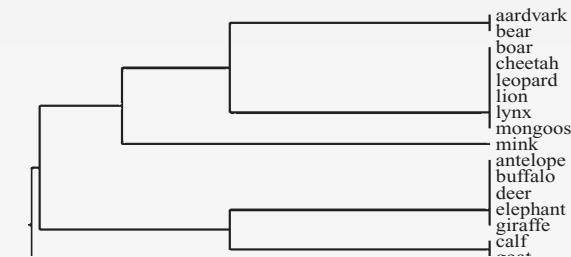
## Clustering jerárquico

### Clustering jerárquico:

[Hastie et al., 2003, Sec. 14.3.12] [Alpaydin, 2010, Sec. 7.7]

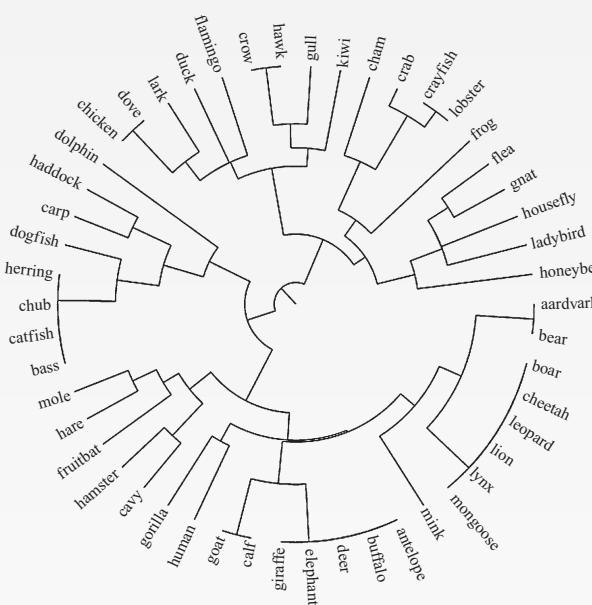
- Objetivo: descubrir agrupamientos naturales en un conjunto  $\mathcal{X}$  de N instancias ( $|\mathcal{X}| = N$ )
- Número de clusters: no se pre-define
  - ▷ Descubre agrupamientos con un número de grupos que va desde 1 hasta N
  - ▷ Ofrece relación entre particiones con distinto número de clusters según una jerarquía
    - \* En el nivel inferior hay N clusters cada uno con una sola muestra
    - \* Los clusters de un nivel se forman agrupando clusters del nivel inmediatamente inferior
    - \* En el nivel superior hay un solo cluster que incluye todas las muestras
- Métodos:
  - ▷ Clustering aglomerativo (e.g. algoritmo Sahn)
  - ▷ Clustering divisivo

## Clustering jerárquico



Fuente de la figura: [Witten et al., 2011]

### Clustering jerárquico



Aplicación: Taxonomía de especies. Fuente de la figura: [Witten et al., 2011]

## Clustering jerárquico

	mpg	cyl	disp	hp	drat	wt	qsec	vs	am	gear	carb
Mazda RX4	21.0	6	160.0	110	3.90	2.620	16.46	0	1	4	4
Mazda RX4 Wag	21.0	6	160.0	110	3.90	2.875	17.02	0	1	4	4
Datsun 710	22.8	4	108.0	93	3.85	2.320	18.61	1	1	4	1
Hornet 4 Drive	21.4	6	258.0	110	3.08	3.215	19.44	1	0	3	1
Hornet Sportabout	18.7	8	360.0	175	3.15	3.440	17.02	0	0	3	2
Valiant	18.1	6	225.0	105	2.76	3.460	20.22	1	0	3	1
Duster 360	14.3	8	360.0	245	3.21	3.570	15.84	0	0	3	4
Merc 240D	24.4	4	146.7	62	3.69	3.190	20.00	1	0	4	2
Merc 230	22.8	4	140.8	95	3.92	3.150	22.90	1	0	4	2
Merc 280	19.2	6	167.6	123	3.92	3.440	18.30	1	0	4	4
Merc 280C	17.8	6	167.6	123	3.92	3.440	18.90	1	0	4	4
Merc 450SE	16.4	8	275.8	180	3.07	4.070	17.40	0	0	3	3
Merc 450SL	17.3	8	275.8	180	3.07	3.730	17.60	0	0	3	3
Merc 450SLC	15.2	8	275.8	180	3.07	3.780	18.00	0	0	3	3
Cadillac Fleetwood	10.4	8	472.0	205	2.93	5.250	17.98	0	0	3	4
Lincoln Continental	10.4	8	460.0	215	3.00	5.424	17.82	0	0	3	4
Chrysler Imperial	14.7	8	440.0	230	3.23	5.345	17.42	0	0	3	4

Figura: mtcars dataset

## Clustering jerárquico

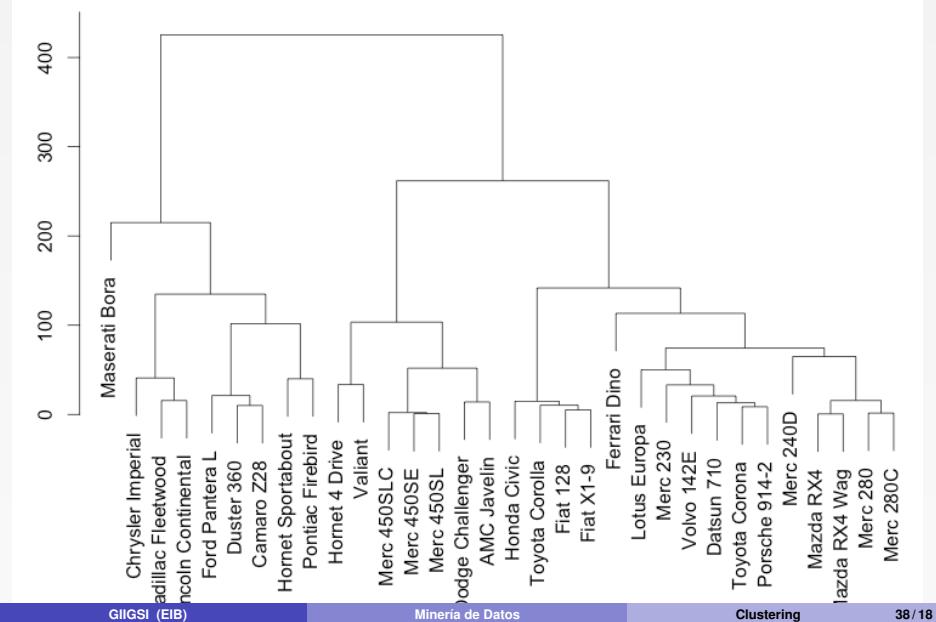


Figura: Dendograma asociado a `mtcars` dataset.

## Clustering jerárquico

Distancia inter-grupal

### Definición distancia inter-grupal:

Distintas formas de definir la **distancia inter-grupal**:

- **Single-link:** la distancia entre dos clusters,  $\mathcal{G}_i$  y  $\mathcal{G}_j$ , se define como la distancia menor entre pares de elementos de los dos clusters

$$d(\mathcal{G}_i, \mathcal{G}_j) = \min_{\forall \mathbf{x}^{(r)} \in \mathcal{G}_i, \forall \mathbf{x}^{(s)} \in \mathcal{G}_j} d(\mathbf{x}^{(r)}, \mathbf{x}^{(s)}) \quad (12)$$

- **Complete-link:** la distancia entre dos clusters,  $\mathcal{G}_i$  y  $\mathcal{G}_j$ , se define como la distancia mayor entre pares de elementos de los dos clusters

$$d(\mathcal{G}_i, \mathcal{G}_j) = \max_{\forall \mathbf{x}^{(r)} \in \mathcal{G}_i, \forall \mathbf{x}^{(s)} \in \mathcal{G}_j} d(\mathbf{x}^{(r)}, \mathbf{x}^{(s)}) \quad (13)$$

- **Average-link:** la distancia entre dos clusters,  $\mathcal{G}_i$  y  $\mathcal{G}_j$ , se define como la distancia entre sus centroides

$$d(\mathcal{G}_i, \mathcal{G}_j) = d(\mathbf{m}_i, \mathbf{m}_j) \quad \text{siendo } \mathbf{m}_i = \frac{1}{|\mathcal{G}_i|} \sum_{\forall \mathbf{x}^{(r)} \in \mathcal{G}_i} \mathbf{x}^{(r)} \quad (14)$$

## Clustering jerárquico

Métodos de clustering jerárquico

### Clustering aglomerativo (*bottom-up*)

- Comienza con N clusters asociando una sola instancia a cada uno de los clusters.
- En cada iteración se obtiene un cluster menos mezclando los dos clusters más cercanos (a menor distancia inter-grupal).
- Termina cuando todos los clusters se han mezclado y se tiene un solo cluster con N instancias.

### Clustering divisorio (*top-down*)

Comienza con un solo cluster con las N instancias y va dividiendo hasta que da lugar a N clusters donde cada uno de ellos contiene sólo una instancia

GLIGSI (EIB)

Minería de Datos

Clustering

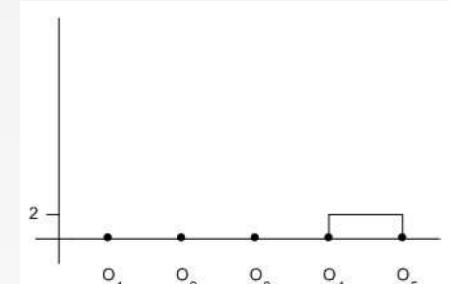
39 / 18

## Clustering jerárquico

Dendograma

Dendograma: creación mediante un ejemplo

	$O_1$	$O_2$	$O_3$	$O_4$	$O_5$
$O_1$	3	14	62	40	
$O_2$		29	61	41	
$O_3$			94	72	
$O_4$				2	
$O_5$					



GLIGSI (EIB)

Minería de Datos

Clustering

40 / 18

GLIGSI (EIB)

Minería de Datos

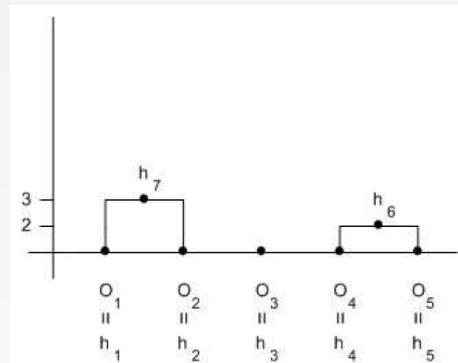
Clustering

41 / 18

## Clustering jerárquico

Dendograma

	$O_1$	$O_2$	$O_3$	$h_6$
$O_1$		3	14	51
$O_2$			29	51
$O_3$				83
$h_6$				



GlGSI (EIB)

Minería de Datos

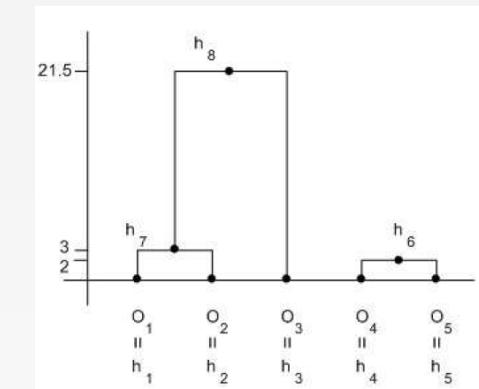
Clustering

42/18

## Clustering jerárquico

Dendograma

	$h_7$	$O_3$	$h_6$
$h_7$		21,5	
$O_3$			83
$h_6$			



GlGSI (EIB)

Minería de Datos

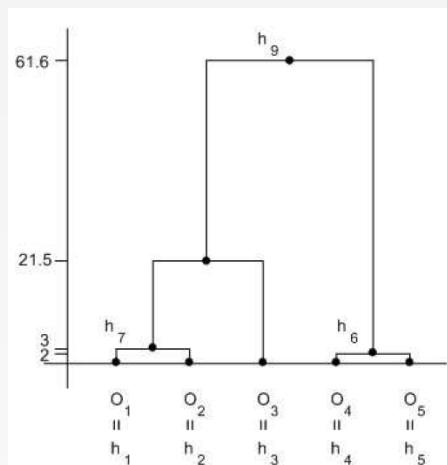
Clustering

43/18

## Clustering jerárquico

Dendograma

	$h_8$	$h_6$
$h_8$		61,6
$h_6$		



GlGSI (EIB)

Minería de Datos

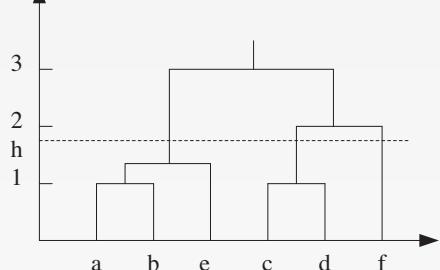
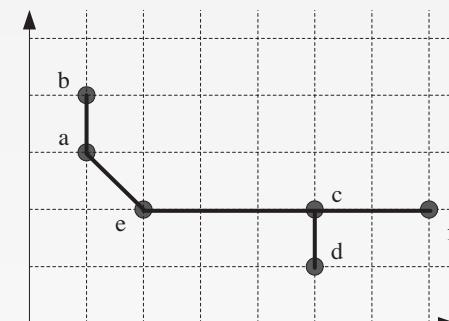
Clustering

44/18

## Clustering jerárquico

Dendograma

**Ejercicio:** Dado los datos (a,b,..., f) representados en espacio bi-dimensional, agruparlos según algoritmo de jerárquico y representar el clustering mediante un dendrograma. ¿Qué representa h en la solución mostrada?



Fuente de la figura: [Alpaydin, 2010, Chap. 7]

GlGSI (EIB)

Minería de Datos

Clustering

45/18

## Clustering jerárquico

Dendograma

Fuente de la figura:  
wikimedia-commons

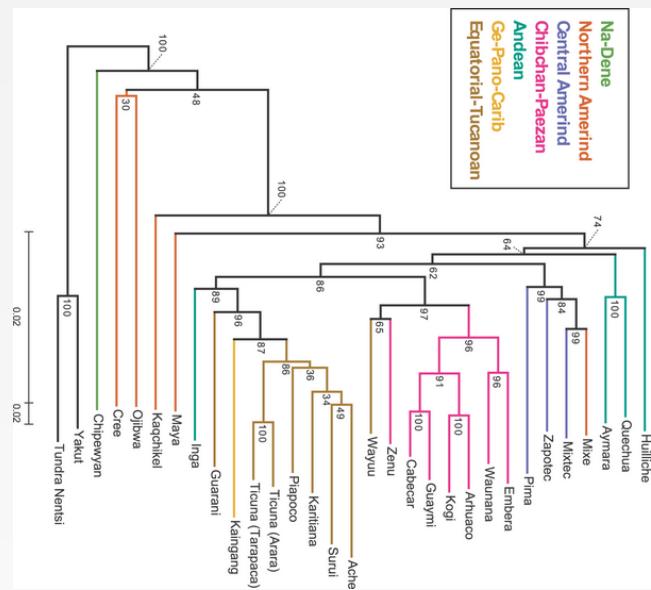


Figura: Hierarchical clustering of languages.

## Clustering jerárquico

### Ejercicio:

Escribir, en pseudo-código, el algoritmo de clustering aglomerativo empleando *complete-link* como distancia inter-grupal.

### Ejercicio:

- ① Descargar *mtcars* dataset
- ② Realizar clustering jerárquico de los datos mediante Weka (u otras herramientas)
- ③ ¿Cómo se ha calculado la distancia entre instancias?
- ④ ¿Cómo se ha calculado la distancia inter-grupal?
- ⑤ Visualizar dendograma
- ⑥ Interpretar el dendograma. ¿Qué coches que pertenecen al mismo cluster? (define los clusters de modo que sean separables según una distancia umbral dada)

## Clustering jerárquico

Dendograma

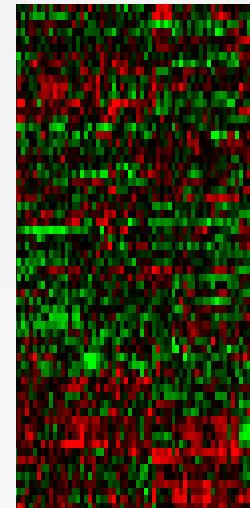


Figura: Hierarchical clustering of DNA

Fuente de la figura: [Hastie et al., 2003]

## Clustering jerárquico

### Ejercicio

Dado un conjunto de 5 datos caracterizados con 3 atributos, se desea representar gráficamente el agrupamiento de los datos mediante clustering jerárquico aglomerativo empleando como distancia inter-grupal *average-link* y la siguiente métrica para la distancia entre instancias:

$$d(\mathbf{x}_i, \mathbf{x}_j) = \sum_{r=1}^3 (x_{i,r} - x_{j,r})^2$$

Atributos		
	$X_1$	$X_2$
1	2	4
2	3	5
3	1	1
4	3	10
5	3	9

## Ejercicio: Clustering basado en densidad

Sesión científica (póster/demo) ¿voluntarios?

## Evaluación

Índices externos

Dadas dos particiones,  $\mathcal{C}$  (real) y  $\mathcal{G}$  (system), contamos los **pares de instancias** (*pair-wise comparison*) que:

- **N11** = pertenecen al mismo cluster en  $\mathcal{C}$  y al mismo grupo en  $\mathcal{G}$
- **N10** = pertenecen al mismo cluster en  $\mathcal{C}$  y a distinto grupo en  $\mathcal{G}$
- **N01** = pertenecen a distinto grupo en  $\mathcal{C}$  y al mismo grupo en  $\mathcal{G}$
- **N00** = pertenecen a distinto grupo en  $\mathcal{C}$  y a distinto grupo en  $\mathcal{G}$

		$\mathcal{C}$	
		=	$\neq$
$\mathcal{G}$	=	N11	N01
	$\neq$	N10	N00

Figuras de mérito o índices para evaluar algoritmos de clasificación no-supervisada: [Bandyopadhyay and Saha, 2013, 6]

- **Externos:** comparar el resultado que proporciona un algoritmo respecto de un conjunto de datos etiquetado (benchmark externo, datos clasificados típicamente por expertos), también se conoce como *cluster vs class evaluation*. Ejemplo: Jaccard index, Rand-index, ...
- **Internos:** exploran la estructura intrínseca de los datos como compromiso de dos métricas:
  - 1 *Cohesión (intra-cluster)*: cercanía de los elementos dentro de un cluster mediante e.g. varianza
  - 2 *Separabilidad (inter-cluster)*: distancia inter-grupal.

Índices internos: BIC-index, CH-index, [Silhouette-index](#), DB-index, Dunn-index, XB-index, PS-index, I-index, ... [Bandyopadhyay and Saha, 2013, 6]

## Evaluación

Índices externos

## Coeficientes

$$\text{Jaccard}(\mathcal{G}, \mathcal{C}) = \frac{N11}{N11 + N10 + N01}$$

$$\text{RandIndex}(\mathcal{G}, \mathcal{C}) = \frac{N11 + N00}{N11 + N10 + N01 + N00}$$

$$\text{FolkesMalows}(\mathcal{G}, \mathcal{C}) = \sqrt{\frac{N11}{N11 + N01} \cdot \frac{N11}{N11 + N10}}$$

## Evaluación

índices externos

### Ejercicio: Pair-wise comparison

Sea  $\{1, 2, \dots, 6\}$  un conjunto de instancias. Las instancias han sido agrupadas de dos formas distintas mediante dos algoritmos P y Q respectivamente:

- $P = \{1, 2, 3\}, \{4, 5, 6\}$
- $Q = \{1, 3, 4\}, \{2, 5, 6\}$

Se pide:

- 1 Comparar, cuantitativamente, las particiones mediante figuras de validación externa: índice Rand, coeficiente Jaccard y Folkes&Malows.
- 2 ¿Son simétricas las figuras de validación externa empleadas? 

## Evaluación

índices internos

### Cohesión interna

- 1 Cohesión del cluster  $\mathcal{G}_i$ : se puede medir la disimilitud interna como la suma de errores cuadráticos (*Sum of Squared Error* ó SSE) respecto al centroide  $\mathbf{c}_i$ :

$$SSE(\mathcal{G}_i) = \sum_{\mathbf{x} \in \mathcal{G}_i} d^2(\mathbf{x}, \mathbf{c}_i)$$

- 2 Cohesión de la partición  $\mathcal{G} = \{\mathcal{G}_1, \dots, \mathcal{G}_i, \dots, \mathcal{G}_m\}$ :

$$SSE(\mathcal{G}) = \sum_{i=1}^m SSE(\mathcal{G}_i) = \sum_{i=1}^m \sum_{\mathbf{x} \in \mathcal{G}_i} d^2(\mathbf{x}, \mathbf{c}_i)$$

Observación: cuanto menor sea SSE, mayor será la cohesión.



## Evaluación

índices internos

### Ejercicio: cohesión interna

- 1 El objetivo del clustering consiste en obtener similitud intra-cluster alta, por lo tanto,  $SSE(\mathcal{G}_i)$  debería ser ... ¿alta o baja?
- 2 Dado que deseamos obtener ¿alta o baja?  $SSE(\mathcal{G}_i) \forall i$ , entonces  $SSE(\mathcal{G})$  debería ser ... ¿alta o baja?



## Evaluación

índices internos

### Separabilidad externa

- Disimilitud externa de la partición:

$$ExtSSE(\mathcal{G}) = \sum_{i=1}^m \sum_{\mathbf{x} \notin \mathcal{G}_i} d^2(\mathbf{x}, \mathbf{c}_i)$$

Observación: coste computacional alto.



- Separabilidad de la partición

$$BSS(\mathcal{G}) = \sum_{i=1}^m |\mathcal{G}_i| d^2(\mathbf{c}, \mathbf{c}_i) \quad (\text{separation})$$

donde  $\mathbf{c}_i$  es el centroide del cluster  $\mathcal{G}_i$ , y  $\mathbf{c}$  es el centroide del conjunto completo de datos  $\mathcal{X}$

## Evaluación

índices internos

### Ejercicio:

- ➊ Incrementando el número de clusters se obtiene una medida SSE ¿mayor o menor?
- ➋ Incrementando el número de clusters se obtiene una medida BSS ¿mayor o menor?
- ➌ Conclusión: compromiso entre SSE y BSS

## Evaluación

índices internos

Dada la instancia  $x_i \in \mathcal{G}_j$ , entonces  $-1 \leq \text{Silhouette}(x_i) \leq 1$

$\text{Silhouette}(x_i) \approx 1$	$\Rightarrow x_i$ está bien agrupada
$\text{Silhouette}(x_i) \approx -1$	$\Rightarrow x_i$ está mal agrupada
$\text{Silhouette}(x_i) \approx 0$	caso intermedio

## Evaluación

índices internos

### Silhouette:

- Índice interno que combina cohesión intra-cluster y separabilidad inter-cluster.
- La anchura silueta de la instancia  $x_i \in \mathcal{G}_j$  se define:

$$\text{Silhouette}(x_i) = \frac{b(x_i) - a(x_i)}{\max\{b(x_i), a(x_i)\}}$$

►  $a(x_i)$ : distancia media entre  $x_i$  y el resto de instancias del cluster  $\mathcal{G}_j$

$$a(x_i) = \frac{1}{|\mathcal{G}_j| - 1} \sum_{x_j \in \mathcal{G}_j} d(x_i, x_j)$$

►  $b(x_i)$ : mínima distancia media entre  $x_i$  y las instancias de los otros clusters  $\mathcal{G}_k$  con  $1 \leq k \leq m \wedge k \neq j$

$$\begin{aligned} b(x_i) &= \min_{\mathcal{G}_k \neq \mathcal{G}_j} d_*(x_i, \mathcal{G}_k) \\ &= \min_{\mathcal{G}_k \neq \mathcal{G}_j} \frac{1}{|\mathcal{G}_k|} \sum_{x_k \in \mathcal{G}_k} d(x_i, x_k) \end{aligned}$$

## Evaluación

índices internos

Índice silueta de un conjunto de instancias: extrapolar el índice silueta a una partición  $\mathcal{G} = \{G_1 \dots G_m\}$  tomando la anchura Silhouette media global del conjunto de instancias  $\mathcal{X} = \{\mathbf{x}^{(i)}\}_{i=1}^N$ :

$$\begin{aligned} \text{silhouette}(\mathcal{X}) &= \overline{\text{Silhouette}(x_i)} \\ &= \frac{1}{|\mathcal{X}|} \sum_{x_i \in \mathcal{X}} \text{Silhouette}(x_i) \end{aligned}$$

## Evaluación

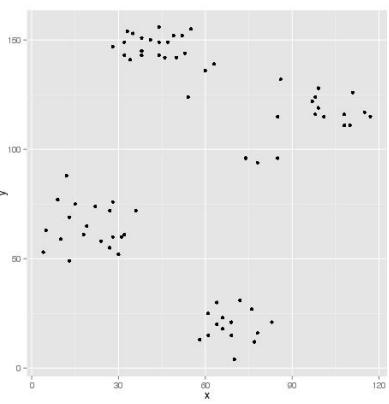
### Ejercicio: estimación de k mediante Silhouette index

Ruspini data (descargar): 75 instancias, 2 atributos.

Se pide:

- ➊ Calcular particiones para distintos valores de  $k$ .
- ➋ Calcular la 'calidad' de cada partición con la anchura media global de silhouette.
- ➌ Elegir el valor de  $k$  ( $2 \leq k \leq M$ ) que maximice silhouette global

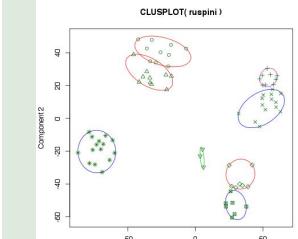
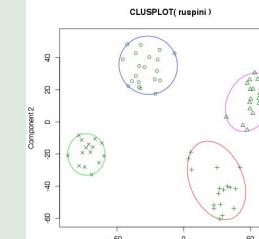
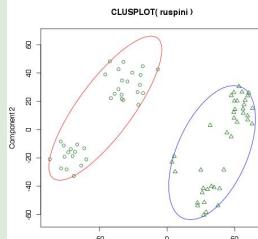
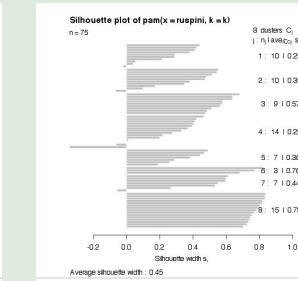
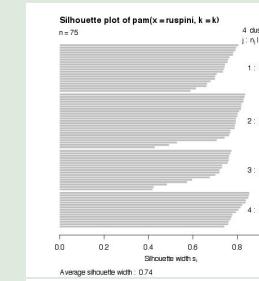
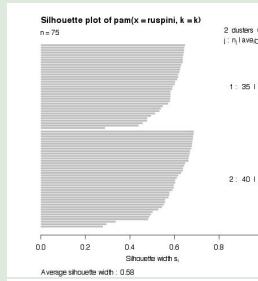
$$\hat{k} = \arg \max_{2 \leq k \leq M} \{\bar{s}_k\}$$



## Evaluación

### Ejercicio: estimación de k mediante Silhouette index

Solución: particiones en  $k \in \{2, 4, 8\}$  clusters



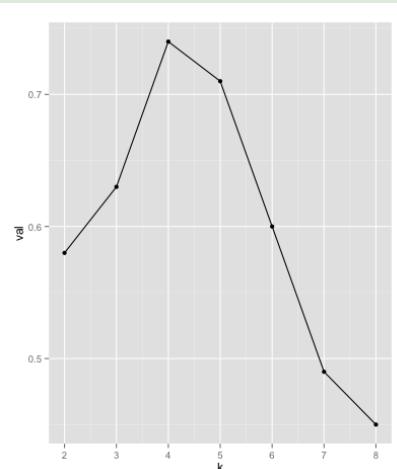
## Evaluación

### Ejercicio: estimación de k mediante Silhouette index

Solución: resultados k-means con particiones de  $k \in \{2, \dots, 8\}$  clusters.

$k$	$\bar{s}_k$
2	0.58
3	0.63
4	0.74
5	0.71
6	0.60
7	0.49
8	0.45

$$\hat{k} = 4$$



Conclusión: hemos deducido el número de clusters ( $\hat{k}$ ) que optimiza el valor silhouette

## Evaluación

### Índices internos

Otros índices de validación interna:

- ➊ Modified Hurbert  $\Gamma$  statistic
- ➋ Dunn's indices
- ➌ Davies-Bouldin index
- ➍ Cophenetic correlation index (clustering jerárquico)
- ➎ ...

## Conclusiones

- Técnica de clasificación no-supervisada
- Objetivo: ayudar a explorar los datos desvelando agrupamientos naturales:
  - ▶ Homogeneidad intra-cluster
  - ▶ Separabilidad inter-cluster
- Fundamental: medida de similitud/disimilitud. Criterios:
  - ▶ Minimizar el error
  - ▶ Maximizar la verosimilitud
- Aproximaciones:
  - ▶ Probabilístico
    - ★ k-means clustering: buscar el *codebook* que minimiza el error de reconstrucción
    - ★ Expectation Maximization: ajustar un modelo de mixturas a los datos
  - ▶ Jerárquico
  - ▶ ...
- Factores determinantes:
  - ▶ Métrica similitud
  - ▶ Número de clusters

## Conclusiones

- Técnicas descriptivas:
  - ▶ Distintas aproximaciones de clustering pueden ayudar a revelar distintas estructuras internas en los datos
- ¿Y si los datos no tienen estructura de clustering?
  - ▶ Estos algoritmos siempre producen una agrupación de las instancias.
- Una buena comprensión e interpretación de los resultados y métricas de evaluación es crucial.

## Bibliografía I

- Alpaydin, E. (2010). *Introduction to Machine Learning*. MIT Press.
- Bandyopadhyay, S. and Saha, S. (2013). *Unsupervised Classification*. Springer.
- Bishop, C. (2006). *Pattern Recognition and Machine Learning*. Springer.
- Duda, R. O., Hart, P. E., and Stork, D. G. (2000). *Pattern Classification*. Wiley-Interscience.
- Han, J., Kamber, M., and Pei, J. (2011). *Data Mining: Concepts and Techniques*. Morgan Kaufmann, 3rd edition.
- Hastie, T., Tibshirani, R., and Friedman, J. (2003). *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. Springer Series in Statistics.

## Bibliografía II

- Maimon, O. and Rokach, L. (2005). *The Data Mining and Knowledge Discovery Handbook*. Springer.
- Witten, I. H., Frank, E., and Hall, M. A. (2011). *Data Mining: Practical Machine Learning Tools and Techniques*. The Morgan Kaufmann Series in Data Management Systems, 3rd edition.