SISTEMAS INTELIGENTES

T11: Aprendizaje no Supervisado

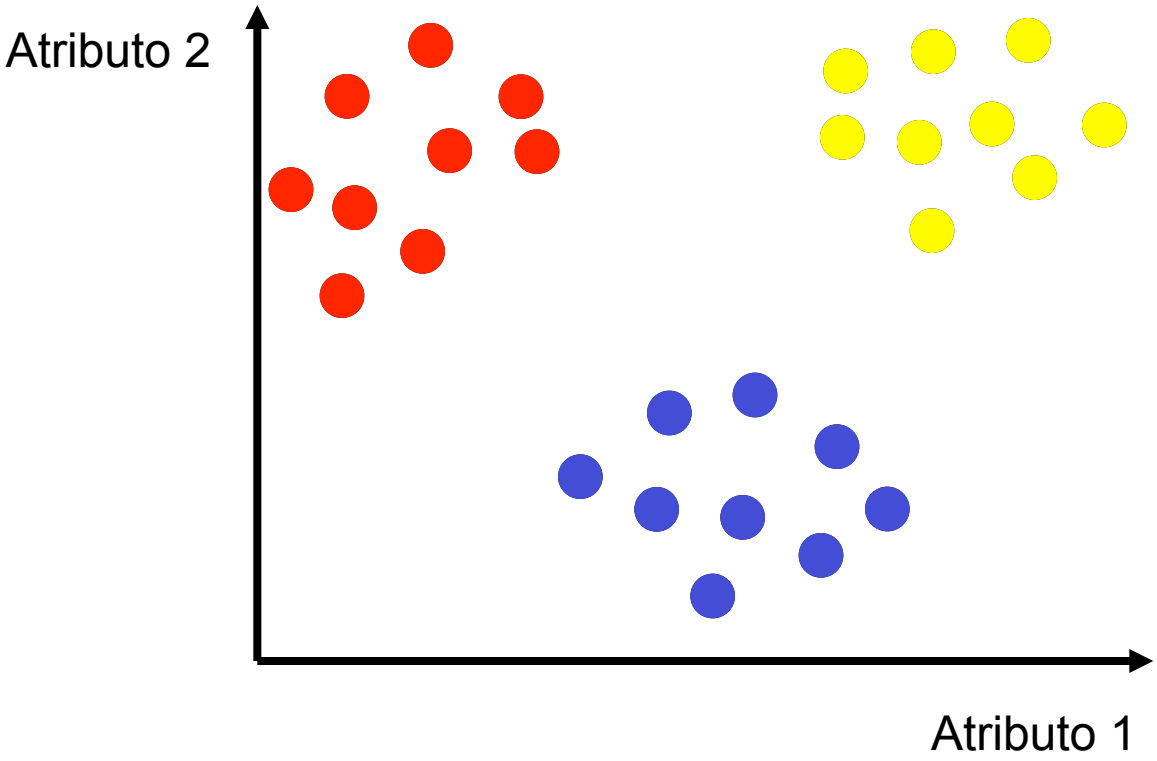
{jdiez, juanjo} @ aic.uniovi.es

Índice

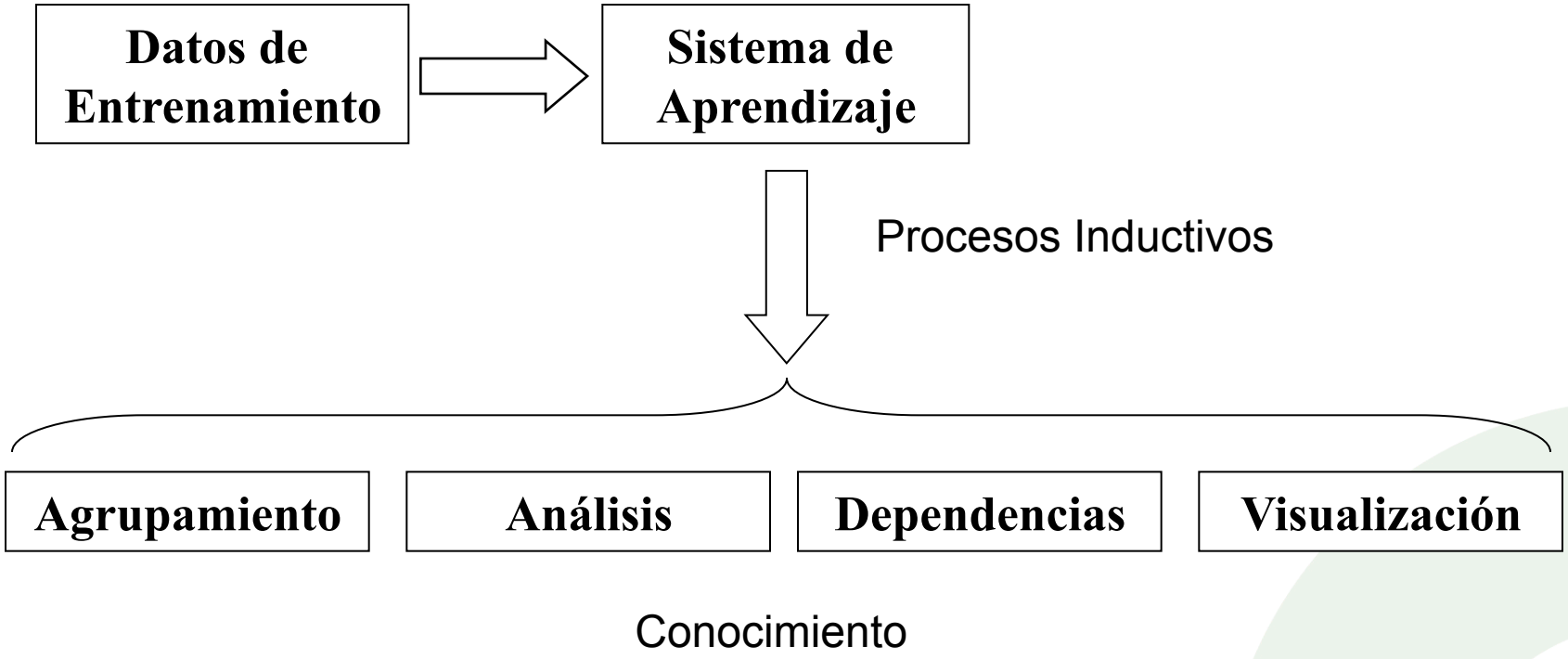
- Aprendizaje no Supervisado
- Clustering
  - Tipos de clustering
  - Algoritmos
    - Dendogramas
    - 1-NN
    - K-means
    - E-M
    - Mapas auto-organizados (SOM)
    - Por estimación de distancias
- Visualización
- Reducción de dimensionalidad
- Extracción de características

Aprendizaje No Supervisado (I)

¿Podemos agrupar los ejemplos en base a sus características?



Aprendizaje Inductivo No Supervisado (II)



Aprendizaje Inductivo No Supervisado (III)

- No hay clase

	Atributos		
Atr-1	**...**	**Atr-n**	**Clase**
	ejemplo 1		clase ej-1
	...		...
	ejemplo m		clase ej-m

- Tratamos de encontrar algún tipo de regularidad en los datos de entrada

Técnicas de aprendizaje no supervisado

- Clustering: agrupan objetos en regiones donde la similitud mutua es elevada
- Visualización: permiten observar el espacio de instancias en un espacio de menor dimensión
- Reducción de la dimensionalidad: los datos de entrada son agrupados en subespacios de una dimensión más baja que la inicial
- Extracción de características: construyen nuevos atributos (pocos) a partir de los atributos originales (muchos)

Clustering

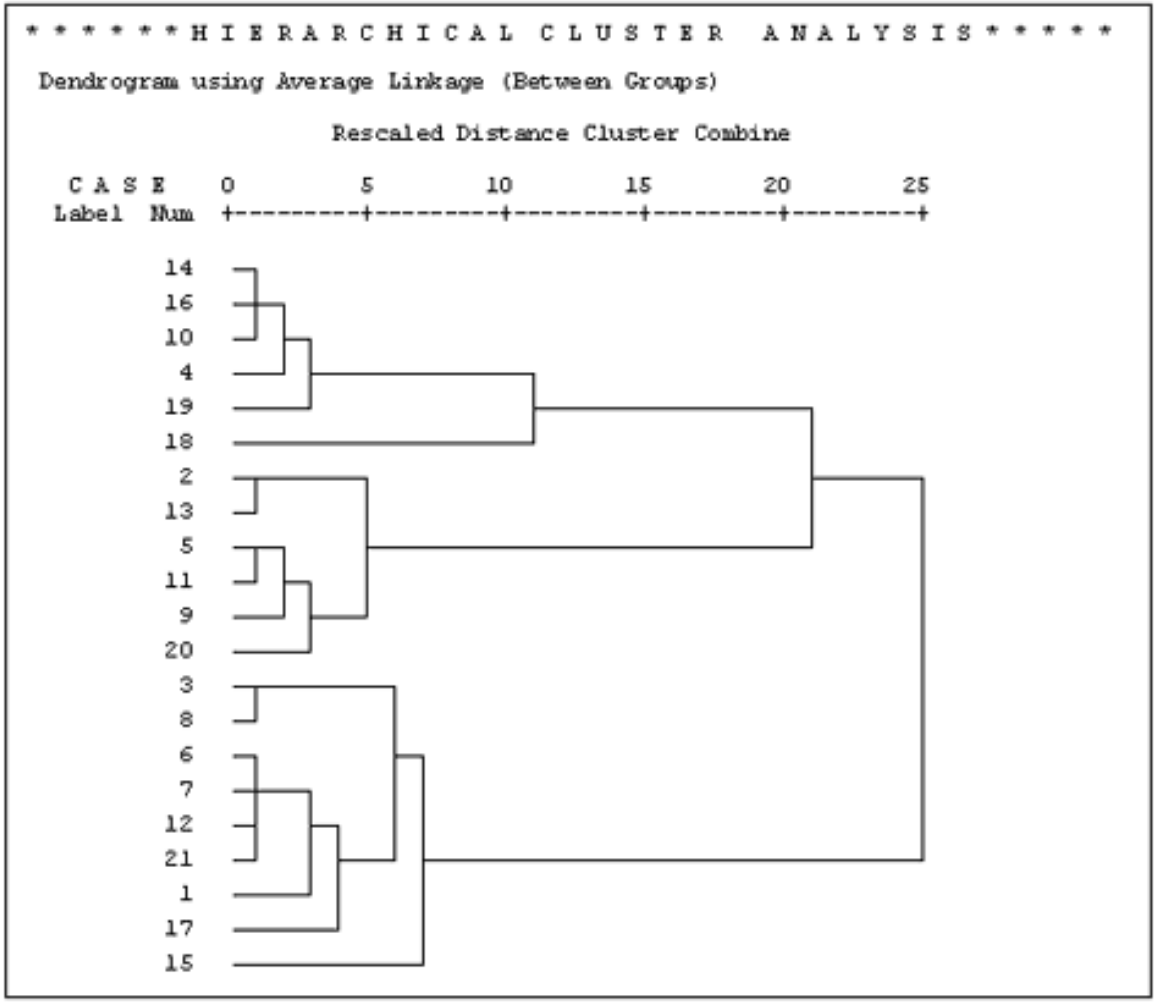
- Objetivo: descubrir estructura en los datos de entrada
  - Busca agrupamientos entre los ejemplos de forma que cada grupo sea homogéneo y distintos de los demás
  - ¿Cuántos grupos?
- Métodos para descubrir estas estructuras
  - Representar la estructura: formar los grupos
  - Describir la estructura: indicar las fronteras que separan los clusters
  - Definir la estructura: asignar nombres útiles a los clusters
- Definición “formal” de cluster:
  - agregación de puntos en el espacio de entrada donde la “distancia” entre cada par de objetos es menor que la distancia de cualquiera de ellos a otro objeto que no pertenece al cluster

Métodos de Clustering

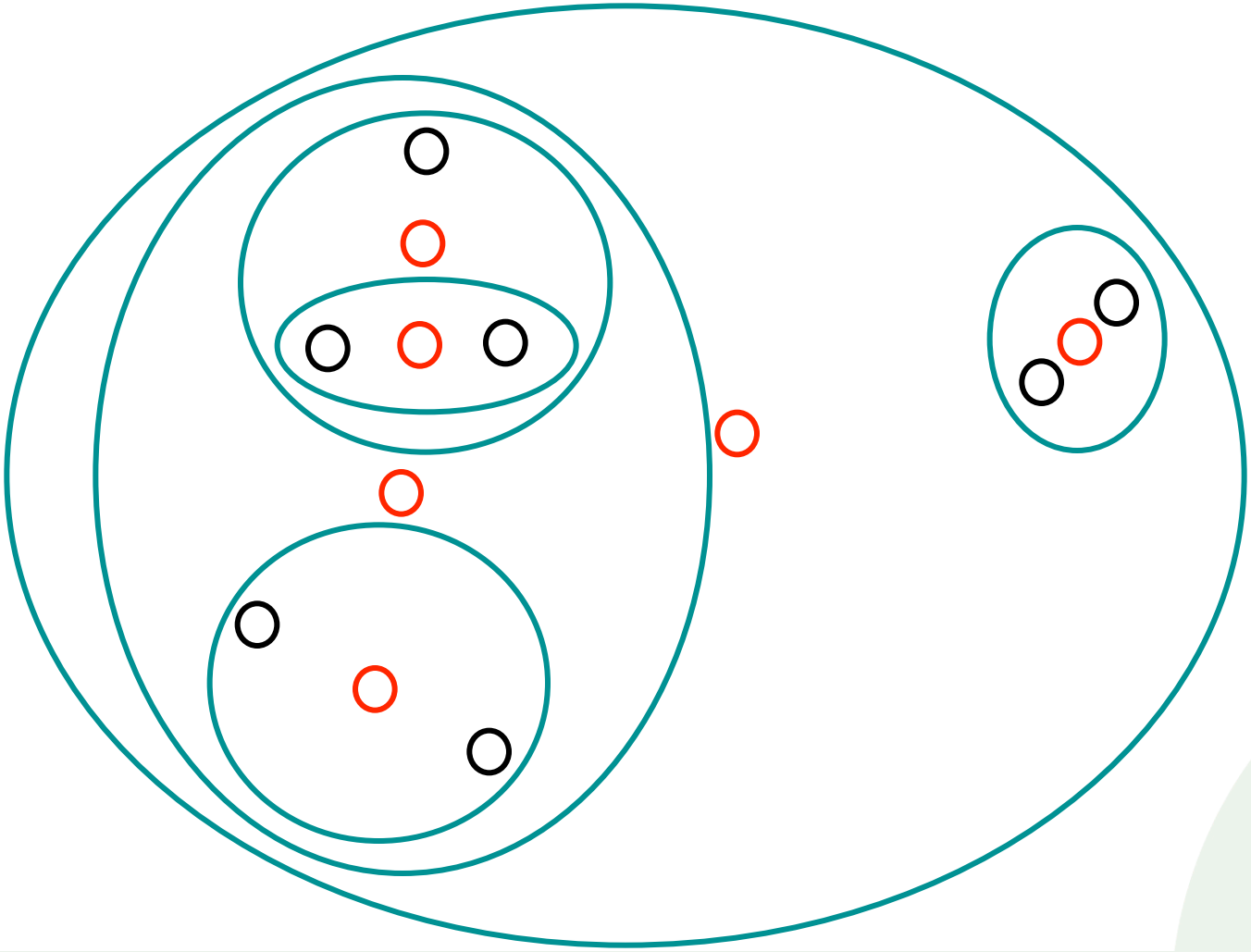
- Jerárquicos: los datos se agrupan de manera arborescente
  - Pueden ser top-down o bottom-up
  - Ej: Dendogramas, Por estimación de distancias
- No jerárquicos: generar particiones a un solo nivel
  - Ej: k-means
- Paramétricos: asumen que las densidades condicionales de los grupos tienen cierta forma paramétrica conocida (p.e. Gaussiana), y se reduce a estimar los parámetros
  - Ej: Algoritmo EM
- No paramétricos: no asumen nada sobre el modo en el que se agrupan los objetos
  - Ej: 1NN, Por estimación de distancias

Dendograma (I)

Un método sencillo consiste en ir agrupando pares de individuos según su *similitud*. Se va aumentando el límite de distancia para hacer grupos. Esto nos da diferentes agrupaciones a distintos niveles, de una manera jerárquica

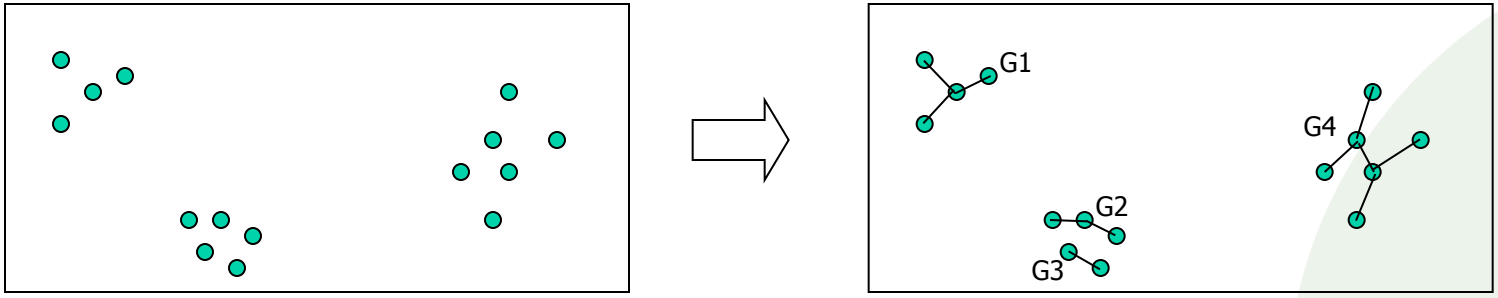


Dendograma (II)



1-NN (Nearest Neighbour)

- Dada una serie de ejemplos en un espacio, se conecta cada ejemplo con su ejemplo más cercano
- La conectividad entre ejemplos genera los grupos
- Puede generar muchos grupos pequeños
- Existen variantes: k-NN o como el spanning tree que para de agrupar cuando llega a un número de grupos



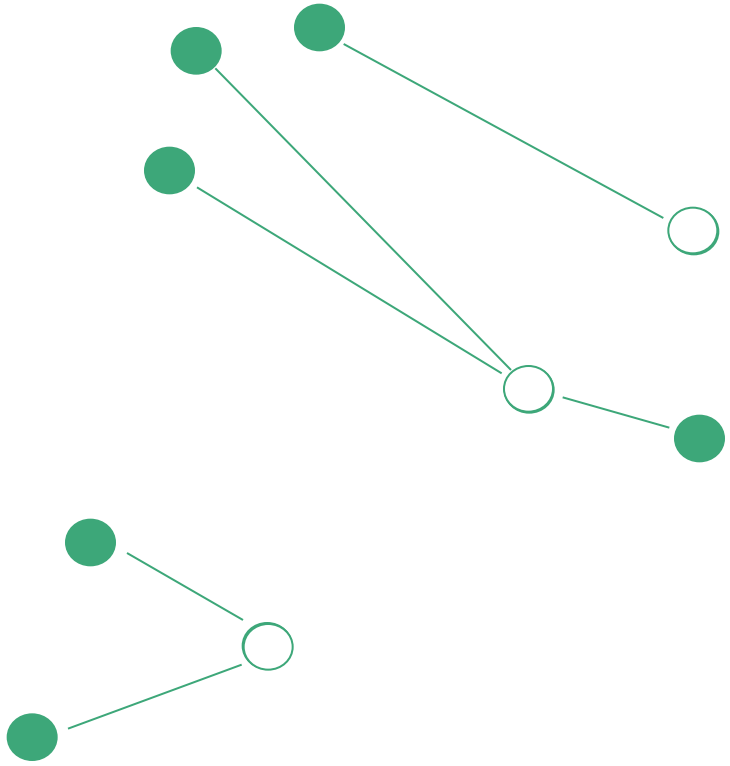
k means (I)

- Se utiliza para encontrar los k puntos *más densos* en el conjunto de datos

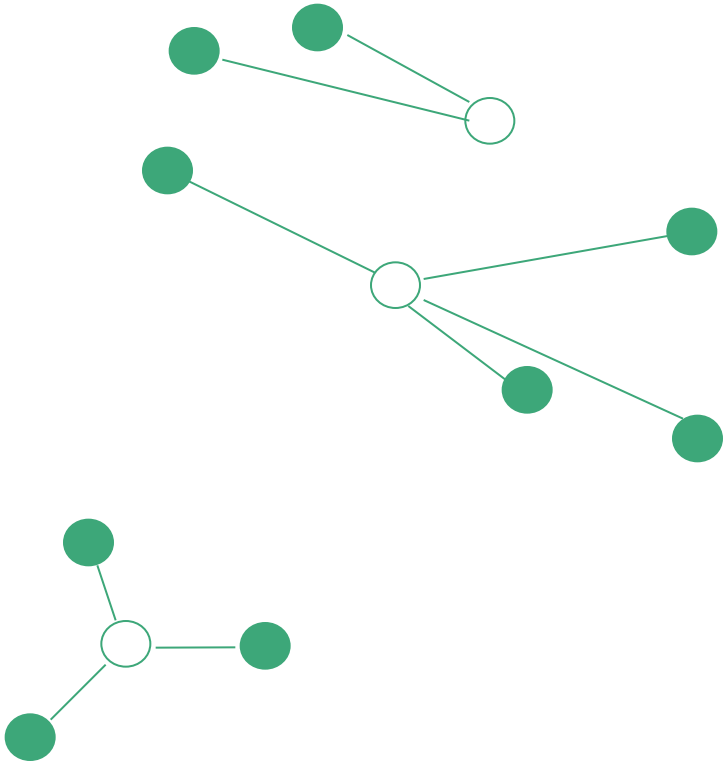
```
Algoritmo k-means es
   Se seleccionan aleatoriamente k centros
   Repetir
        Asignar cada ejemplo al conjunto con el centro más cercano
        Calcular los puntos medios de los k conjuntos
    Hasta que los conjuntos no varíen
    Devolver los k centros
Fin Algoritmo
```

- Determinar k no es fácil
  - Si k se elige muy pequeño, se agruparían grupos “distintos”
  - Si k se elige muy grande, hay centros que se quedan huérfanos
  - Incluso con k exacto, puede haber algún centro que quede huérfano
  - El valor de k se suele determinar heurísticamente
- Depende de los centros iniciales
  - Se puede ejecutar con distintas semillas

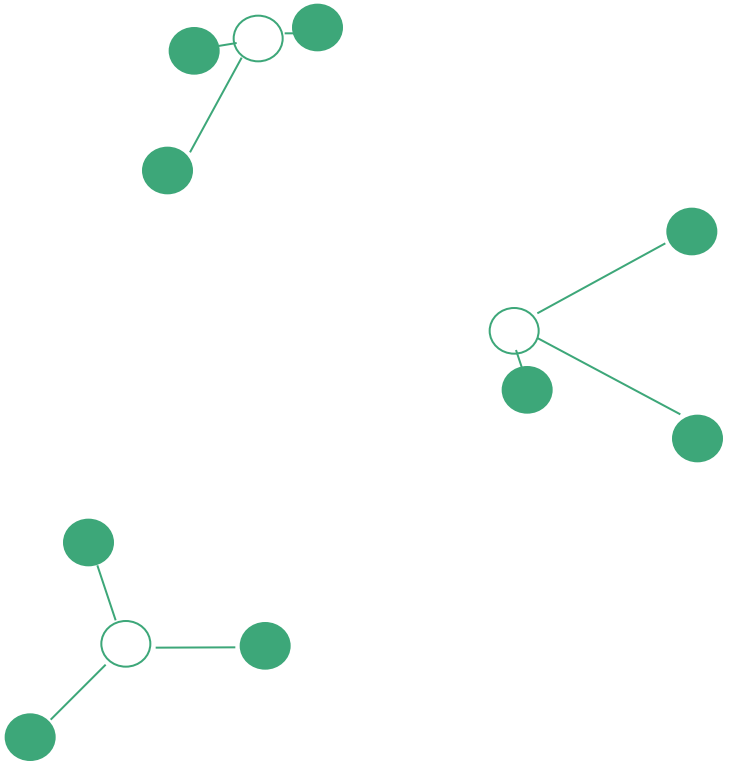
k means (II)



k means (III)



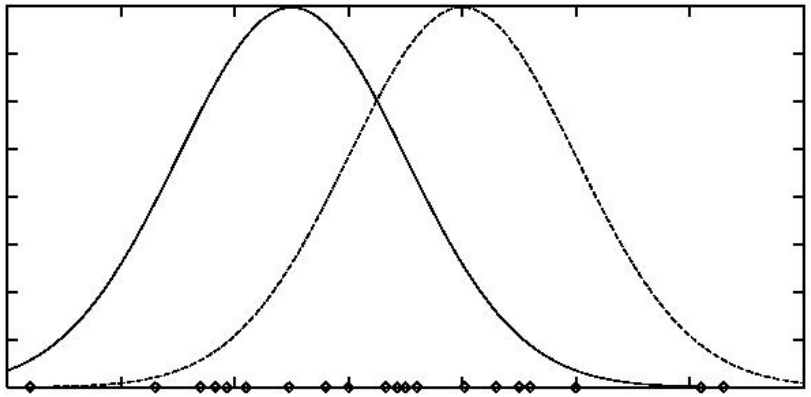
k means (IV)



Algoritmo EM (I)

- Expectation Maximization, Maximum Likelihood Estimate (Dempster et al. 1977)
- Es la base de muchos algoritmos: Autoclass, HMM,…
- Se utiliza cuando tenemos variables ocultas o latentes
- Se adapta al clustering: <x, ¿grupo?>
- Idea intuitiva, consta de dos pasos:
  - Paso de Estimación: dada la hipótesis actual h, estimamos las variables ocultas
  - Paso de Maximización: una vez estimadas las variables ocultas, generamos una nueva hipótesis h'
  - Se repiten los pasos EM hasta que no cambia la hipótesis

Algoritmo EM (II)



- Generamos ejemplos en base a dos distribuciones normales
- Pretendemos asignar cada ejemplo a uno de los dos grupos
- Si conocemos la desviación de las dos distribuciones normales (igual), el problema pasa por descubrir la media de cada una de ellas

Partimos de un conjunto { $<x_1>$, …, $<x_n>$ }

Queremos saber a qué grupo pertenece cada ejemplo

$<x_i$, grupo1 o grupo2 }

La hipótesis que buscamos son $h=<\mu_1,\mu_2>$ y tratamos de hacerlo a través de dos variables ocultas $<x, z_1, z_2>$

z_i será la probabilidad de que pertenezca al grupo i

Algoritmo EM (III)

Algoritmo:

Inicialización: generamos aleatoriamente h=<μ_1,μ_2>

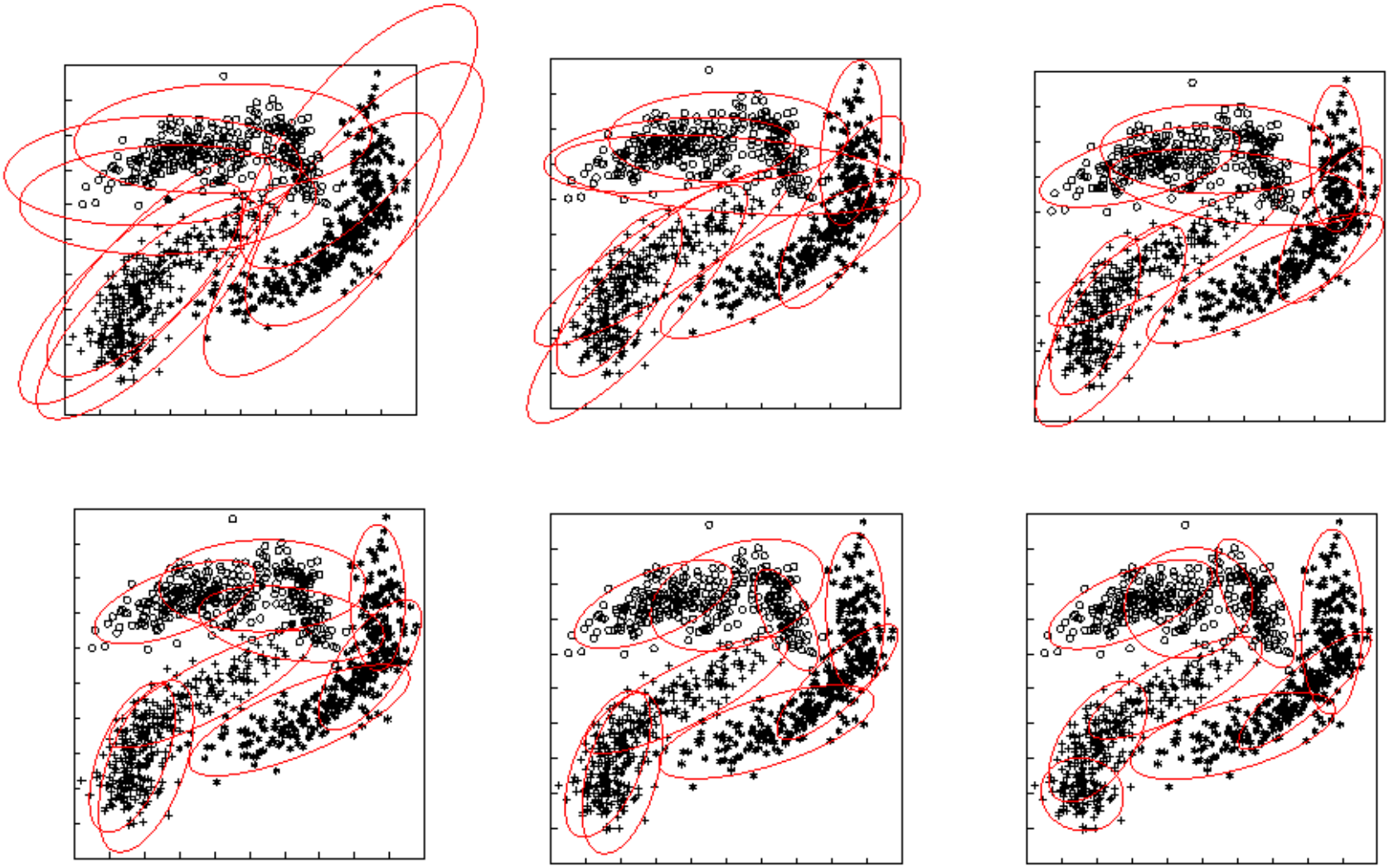
Paso E: **E**stimamos los valores z_1,z_2 para todos los ejemplos usando h

Paso M: En base a las estimaciones **M**aximizamos la hipótesis h, generando una nueva hipótesis h' =<μ_1',μ_2'> (mover las medias)

Paso E: $$E[z_{ij}] = \frac{p(x = x_i \mid \mu = \mu_j)}{\sum_{n=1}^{2} p(x = x_i \mid \mu = \mu_n)} = \frac{e^{-\frac{1}{2\sigma^2}(x_i - \mu_j)^2}}{\sum_{n=1}^{2} e^{-\frac{1}{2\sigma^2}(x_i - \mu_n)^2}}$$

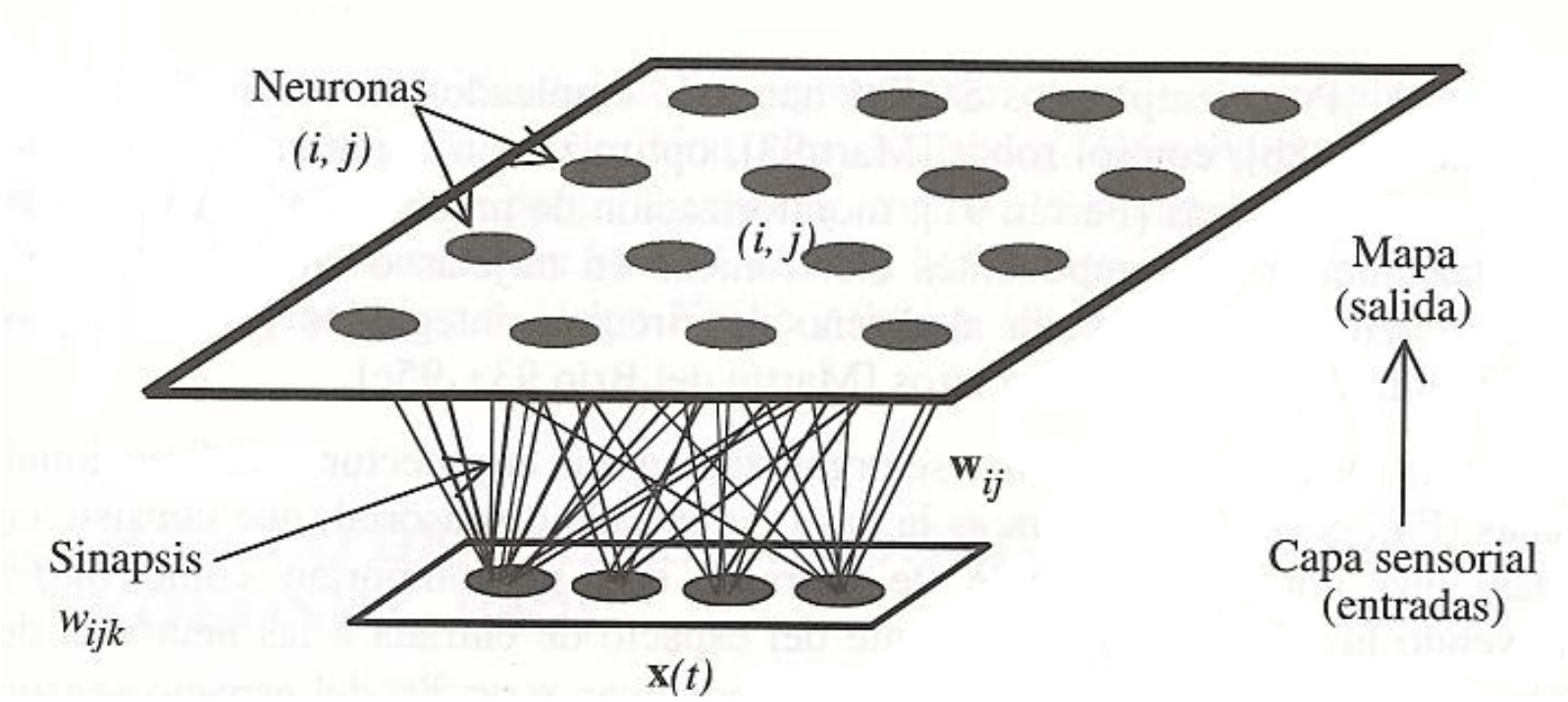
Paso M: $$\mu_j = \frac{\sum_{i=1}^{m} E[z_{ij}]\, x_i}{\sum_{i=1}^{m} E[z_{ij}]}$$

Algoritmo EM (IV)



Mapas auto-organizados (I)

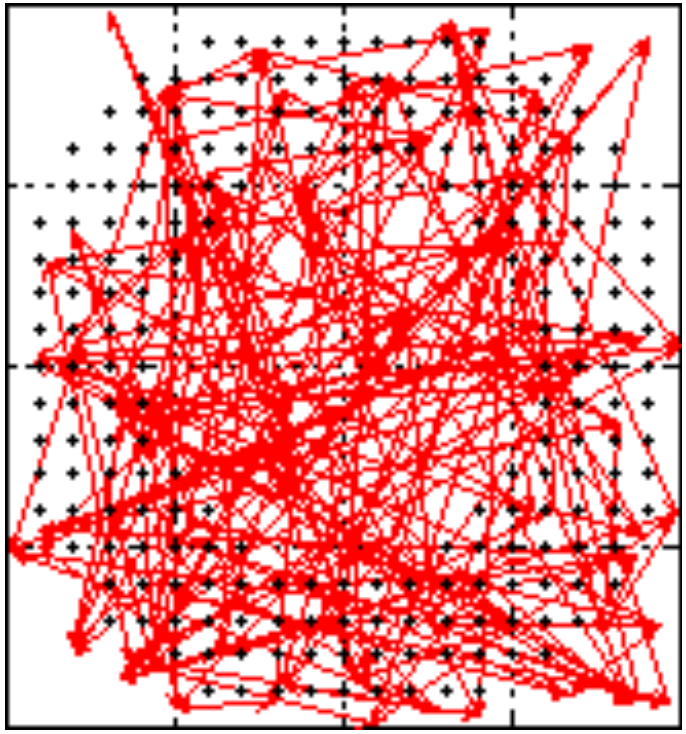
- Self-Organizing Maps (SOM) o redes de Kohonen, o de memoria asociativa, LVQ (linear-vector quantization) [Kohonen]



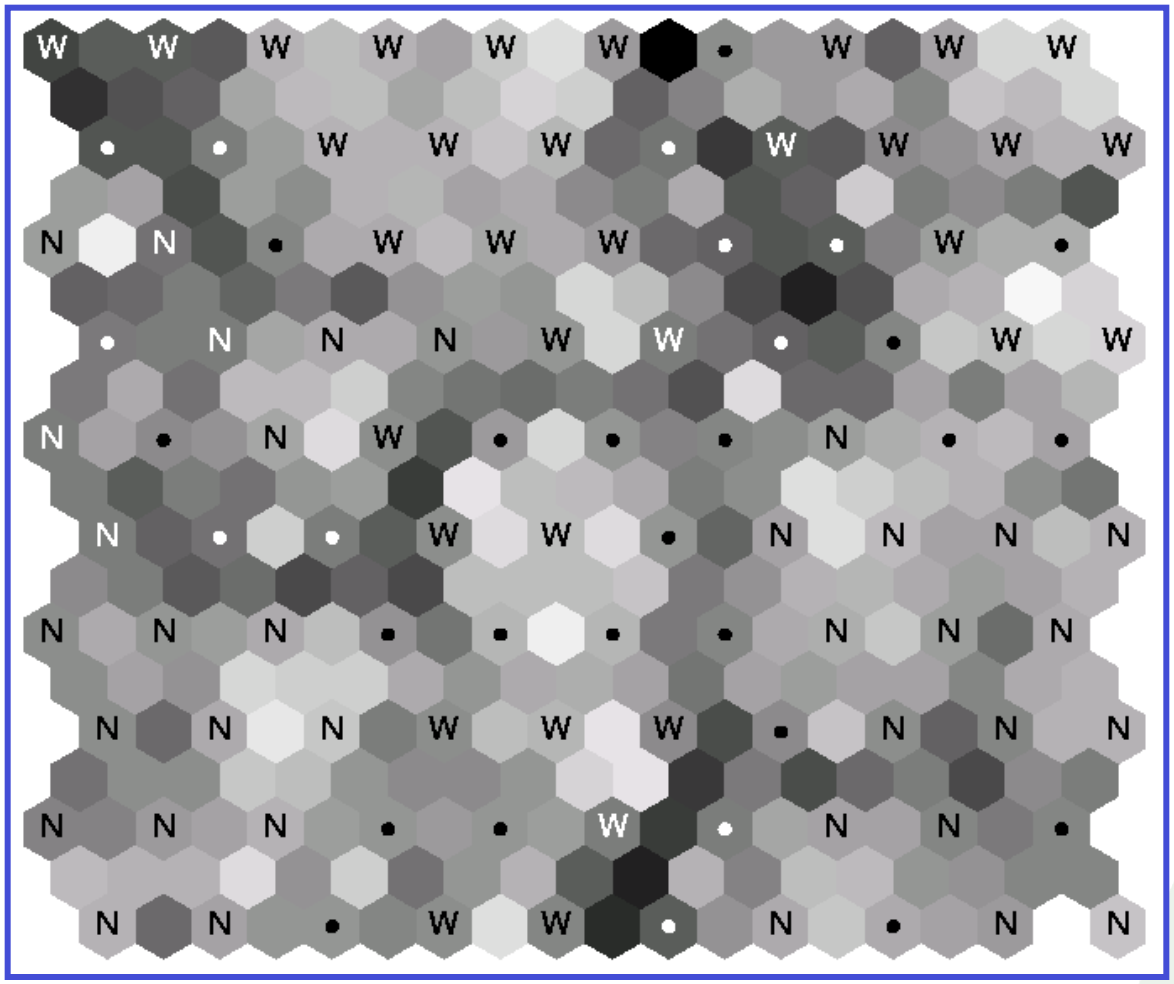
Mapas auto-organizados (II)

- Se dispone de un mapa o retícula finita formada por nodos con un vector de la misma dimensión que los ejemplos de entrada
- Durante el entrenamiento, para cada ejemplo se calcula el nodo más próximo (nodo vencedor)
- Ese nodo, y sus vecinos, se adaptan para aproximarse al vector de entrada. Este proceso mantiene la topología
- El mapa final, a través de sus nodos, agrupa los ejemplos en clusters. Cada nodo agrupa un cierto número de ejemplos
- El mapa final se puede etiquetar, asignando a cada nodo la clase mayoritaria de los ejemplos que representa
- Gracias a que mantiene la topología, el mapa muestra la distancia entre los distintos grupos
- Permite visualizar los datos de entrada en un plano de dos dimensiones

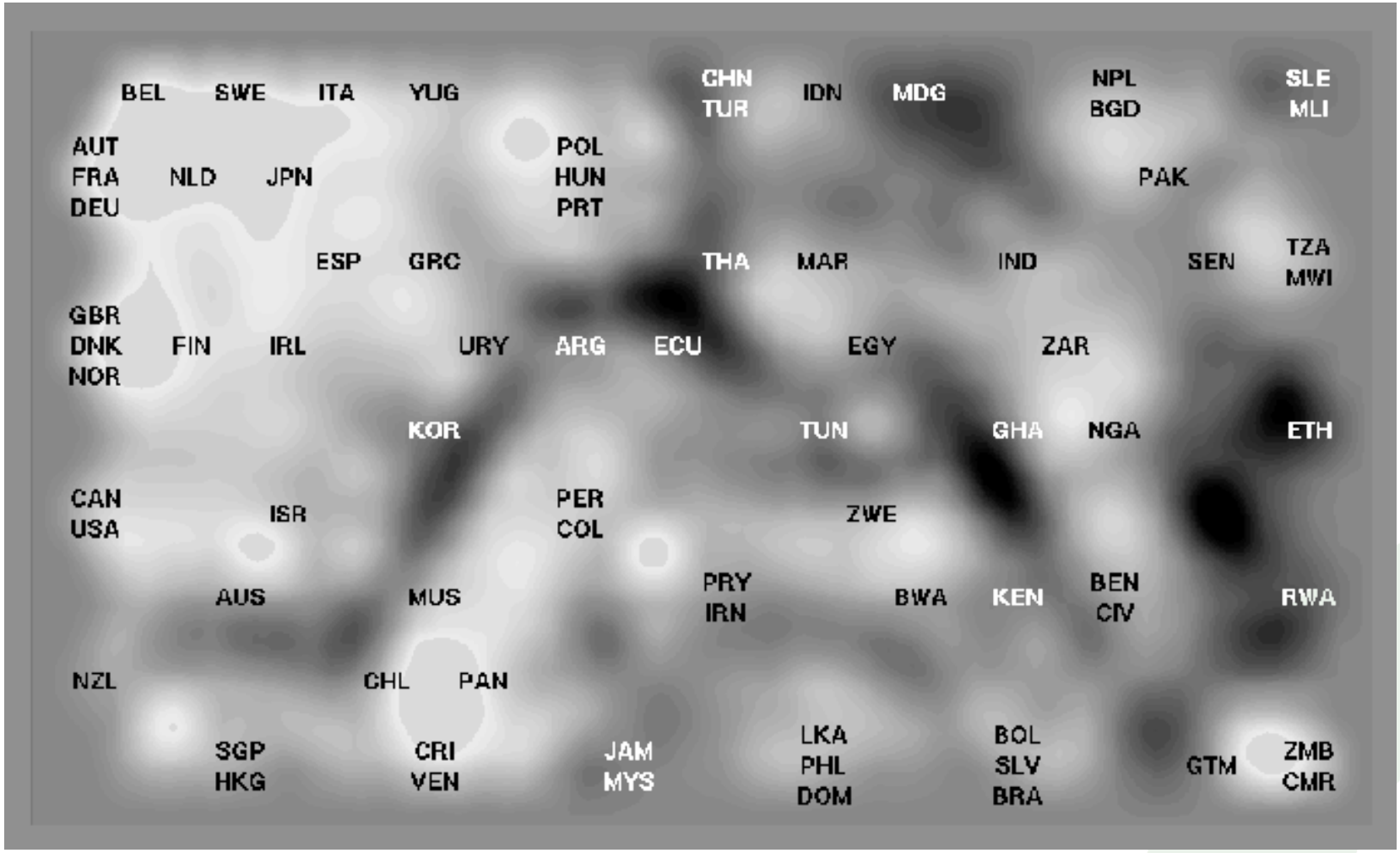
Mapas auto-organizados (III)



Visualización: SOM (I)

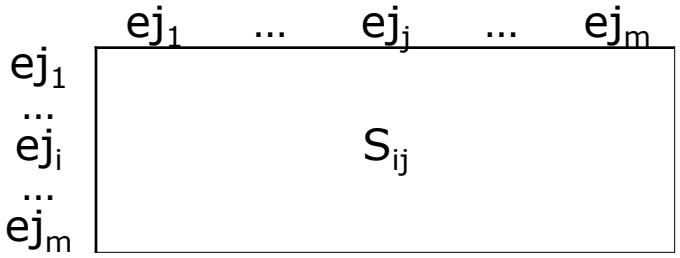


Visualización: SOM (II)



Por estimación de distancias (I)

- Parte de una matriz de similitud (o de distancias) entre cada uno de los objetos



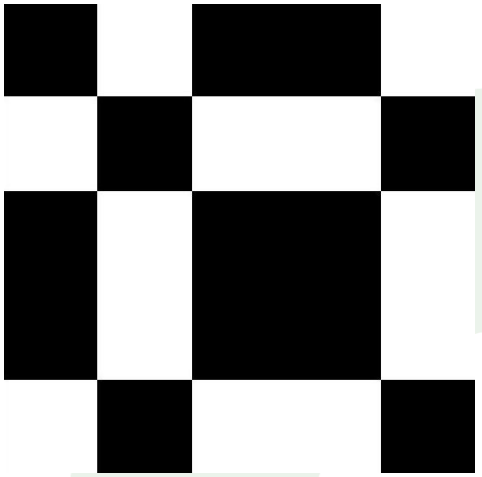
- Se considera que dos objetos están próximos si son similares entre si y, además, diferentes de los mismos objetos
- Mediante un proceso iterativo, se transforma la matriz hasta que finalmente converge a una matriz binaria:
  - El valor 0 indican que esos dos objetos pertenecen a un mismo cluster
  - El valor distinto de 0, que pertenecen a distintos grupos
- Si se repite el proceso con cada submatriz resultante, tenemos un cluster jerárquico

Por estimación de distancias (II)

- En cada iteración se transforma la matriz en dos pasos
- Normalización: usando una norma L_X
- Se vuelven a calcular las similitudes: divergencia Jensen-Shannon

$$p_{ij}(t+1) = \frac{s_{ij}(t)}{max\{|s_{ik}(t)| : k\}}$$

$$s_{ij}(t+1) = \frac{1}{2}\left(\sum_k p_{ik}(t+1) \log \frac{p_{ik}(t+1)}{\frac{1}{2}\left(p_{ik}(t+1) + p_{jk}(t+1)\right)} + \sum_k p_{jk}(t+1) \log \frac{p_{jk}(t+1)}{\frac{1}{2}\left(p_{jk}(t+1) + p_{ik}(t+1)\right)} \right)$$



Análisis Estadísticos

- Se pueden utilizar como paso previo para determinar el método más apropiado para un aprendizaje supervisado
- Se utilizan como preprocesos para la limpieza y preparación de datos para el uso de métodos supervisados
- Ejemplos:
  - Estudio de la distribución de los datos
  - Estimación de densidad
  - Detección datos anómalos
  - Análisis de dispersión (p.ej. las funciones de separabilidad pueden considerarse como técnicas muy simples no supervisadas)

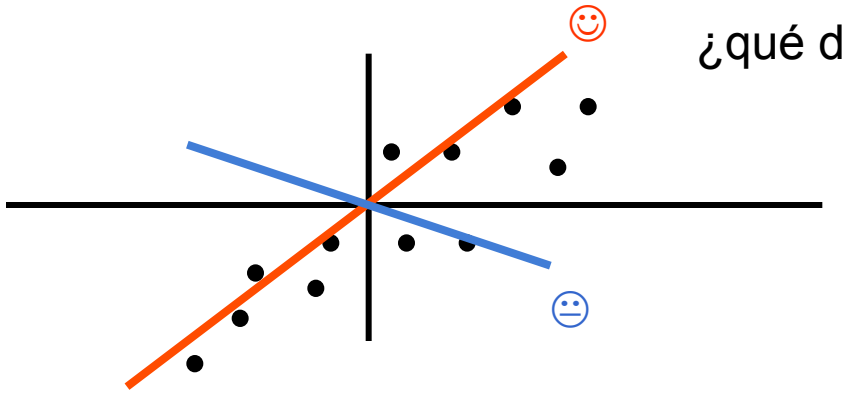
Reducción de dimensionalidad

- ¿Por qué es interesante?

  Hay atributos que resultan nocivos para el aprendizaje:
  - Irrelevancia: el atributo no ofrece capacidad de discriminación en el aprendizaje
  - Separación de la información: una información interesante puede estar recogida en varios atributos
  - Se trata de aumentar la densidad de la información reduciendo el espacio de entrada
- PCA – Análisis de componentes principales
  - Combinan variables redundantes en una única variable (componente o factor)
- FA – Análisis de factores
  - Clase de algoritmos que incluyen PCA

Análisis de Componentes Principales

- Objetivo: Encontrar un espacio de dimensión menor que preserve la mayor cantidad de información que contiene el espacio original



¿qué dirección contiene más información?

Correlaciones y Asociaciones

- Permiten establecer relevancia/irrelevancia de factores y si ésta es positiva o negativa respecto a otro factor o variable a estudiar
- Coeficiente de correlación y matrices de correlación

$$Cor(\bar{x},\bar{y}) = \frac{Cov(\bar{x},\bar{y})}{\sigma_x \cdot \sigma_y} \qquad Cov(\bar{x},\bar{y}) = \frac{1}{n}\sum_{i=1}^{n}(x_i - \mu_x)(y_i - \mu_y)$$

- Asociaciones (cuando los atributos son discretos)
  - Ejemplo: tabaquismo y alcoholismo están asociados.
- Dependencias funcionales: asociación unidireccional
  - Ejemplo: el nivel de riesgo de enfermedades cardiovasculares depende del tabaquismo y alcoholismo (entre otras cosas)

Asociaciones y dependencias

- Se buscan **asociaciones** de la siguiente forma:

$$(x_1 = a) \leftrightarrow (x_4 = b)$$

- De los n casos del conjunto de entrada que las dos comparaciones sean verdaderas o falsas será cierto en rc casos:

Tc = certeza de la regla = rc/n

- Si consideramos valores nulos, tenemos también un número de casos en los que se aplica satisfactoriamente (diferente de Tc) y denominado Ts
- Se buscan **dependencias** de la siguiente forma (IF Ante THEN Cons):

if $(x_1= a, x_3=c, x_5=d)$ then $(x_4=b, x_2=a)$

- De los n casos de entrada, el antecendente se puede hacer cierto en ra casos y de estos en rc casos se hace también el consecuente
- Tenemos dos parámetros Tc (confidence/accuracy) y Ts (support):

Tc= certeza de la regla =rc/ra, fuerza o confianza P(Cons|Ante)

Ts = mínimo nº de casos o porcentaje en los que se aplica satisfactoriamente (rc o rc /n respectivamente).

Llamado también prevalencia: P(Cons $\wedge$ Ante)

Reglas de asociación y dependencia

DNI	Renta Familiar	Ciudad	Profesión	Edad	Hijos	Obeso	Casado
11251545	5.000.000	Barcelona	Ejecutivo	45	3	S	S
30512526	1.000.000	Melilla	Abogado	25	0	S	N
22451616	3.000.000	León	Ejecutivo	35	2	S	S
25152516	2.000.000	Valencia	Camarero	30	0	S	S
23525251	1.500.000	Benidorm	Animador	30	0	N	N

- Asociaciones:
  - Casado e (Hijos > 0) están asociados (66%, 3 casos).
  - Obeso y casado están asociados (80%, 4 casos)
- Dependencias:
  - (Hijos > 0) → Casado (100%, 2 casos).
  - Casado → Obeso (100%, 3 casos)

Clustering conceptual

- **Una partición de los datos es buena sii cada clase tiene una buena interpretación conceptual**
- Cada grupo queda caracterizado por un concepto
- Tiene en cuenta las relaciones semánticas entre los atributos
- Intenta introducir la mayor información sobre el contexto
- COBWEB
  - Genera un árbol (clasificación)
  - Cada nodo hace referencia a un concepto y contiene la descripción probabilística de los atributos (y de la clase si la hubiera)
  - Recorre el árbol en sentido descendente buscando el mejor lugar en el que colocar cada ejemplo
  - Usa una medida, la utilidad de la categoría, para decidir cual es el mejor punto para añadir
  - No es sencillo explicarlo en una transparencia…

Extracción de características

- Idea intuitiva:
  - Pretenden descubrir variables dependientes (y=f(x)) a partir de variables independientes (x)
  - No necesitan conocer las variables dependientes