Jesús García Herrero

TÉCNICAS DE AGRUPAMIENTO

En esta clase se presentan las técnicas de agrupamiento, también conocidas como clustering, que buscan grupos de instancias con características similares mediante el análisis de "parecido" entre sus atributos. Por tanto, a diferencia de las técnicas anteriores de clasificación y predicción, no se precisa de datos "etiquetados" con categorías o valores objetivo, sino que es un análisis "no supervisado" para encontrar una estructura en los datos.

Se revisan en primer lugar la técnicas basadas en distancias, siendo el más representativo el método "k-medias", que agrupa los datos en k grupos de mínima distancia. Se revisa el problema de definir distancias cuando el espacio de atributos es heterogéneo con técnicas de normalización y transformación de atributos nominales.

A continuación, se presentan las técnicas jerárquicas de agrupamiento, cuyo objetivo no es separar en grupos en un mismo nivel, sino hacer una estructura jerárquica conocida como dendograma. Se detalla el algoritmo COBWEB como técnica representativa de este problema, mostrando las heurísticas de construcción del árbol jerárquico basadas en una función de utilidad de tipo probabilístico que intenta maximizar el parecido entre instancias dentro de cada categoría y maximizar a su vez la separación entre categorías

Por último, se presenta el algoritmo EM (Expectation-Maximization) como técnica que permite estimar grupos de instancias y parámetros de distribuciones de probabilidad que los describen, con un criterio de ajustar éstas a un conjunto de categorías prefijado.

El tema se completa con una presentación de las técnicas "semi-supervisadas", que buscan aunar los clasificadores con las técnicas de clustering con el objetivo de explotar datos no etiquetados, habitualmente mucho más disponibles que los etiquetados que se usan en las técncias habituales de clasificación. Un ejemplo claro de técnica semi-supervisada es la combinación del algoritmo EM con los clasificadores bayesianos.

Agrupamiento

Técnicas y análisis de clustering

Jesús García Herrero
Universidad Carlos III de Madrid

Universidad
Carlos III de Madrid

Agrupamiento

- **Aprendizaje no supervisado.** Objetivo: dados ejemplos sin etiquetar (con clase), determinar un conjunto de grupos (*clusters*) en los que se pueden dividir

- No hay variable a predecir, sino se pretende *descubrir* una estructura en los datos. Ejemplo
  - grupos de clientes con gustos similares en libros, música, etc., para análisis de mercado

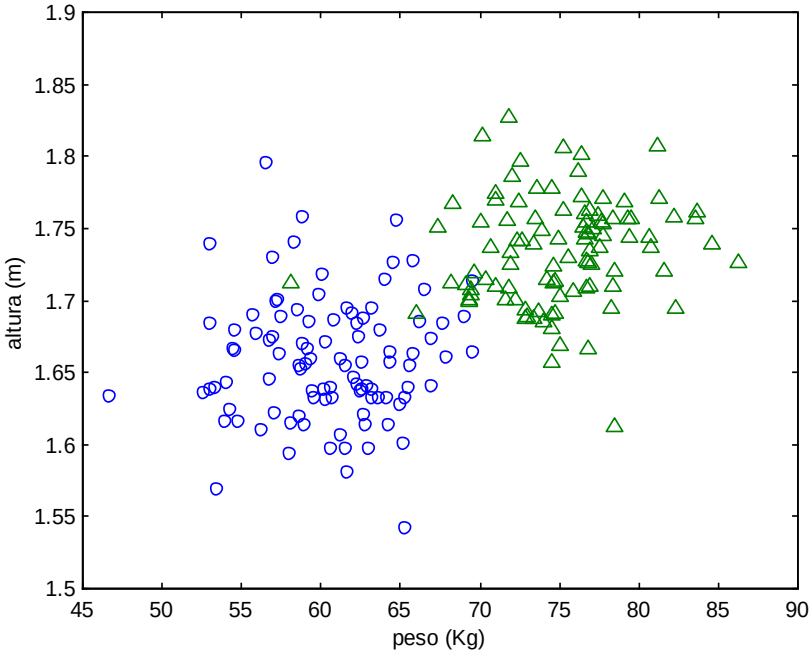
- Suele servir de punto de partida para después hacer un análisis de clasificación sobre los clusters

- Tipos de técnicas:
  - Aglomerativas y divisoras
  - Numéricas: k -medias, EM
  - Conceptuales: cluster/2 y cobweb

Ejemplos de entrada

Sitio de acceso: A_1	1ª cantidad gastada: A_2	Vivienda: A_3	Última compra: A_4
1	0	2	Libro
1	0	1	Disco
1	2	0	Libro
0	2	1	Libro
1	1	1	Libro
2	2	1	Libro

Conceptual

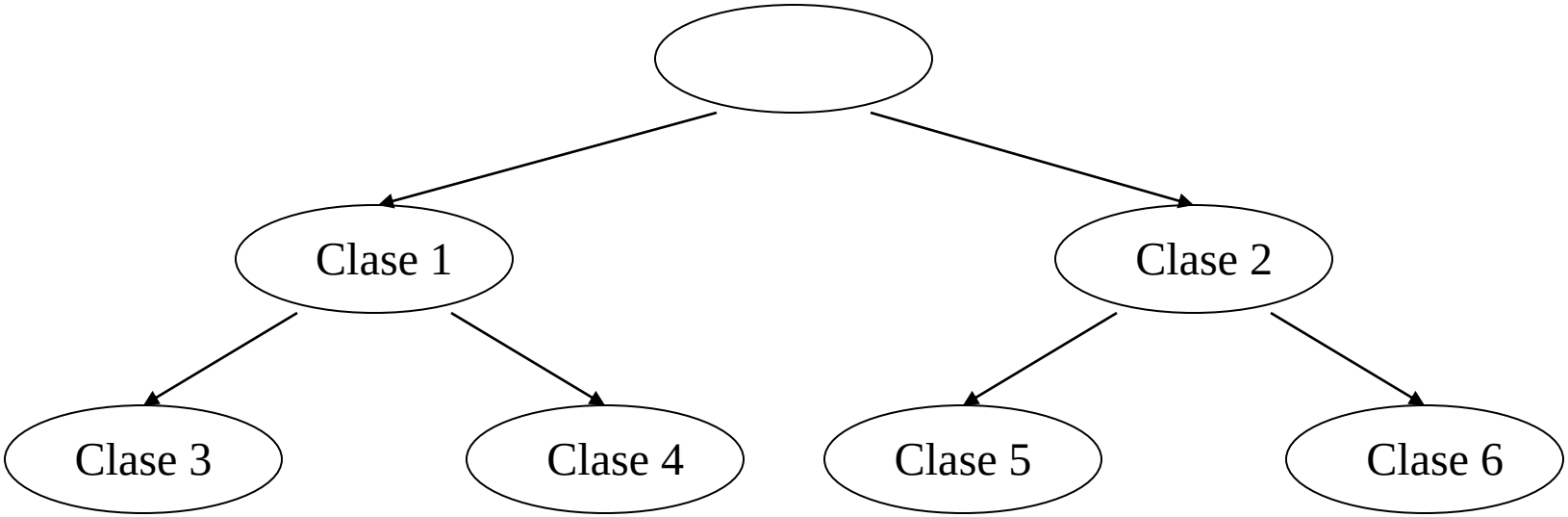


Numérico

Agrupamiento

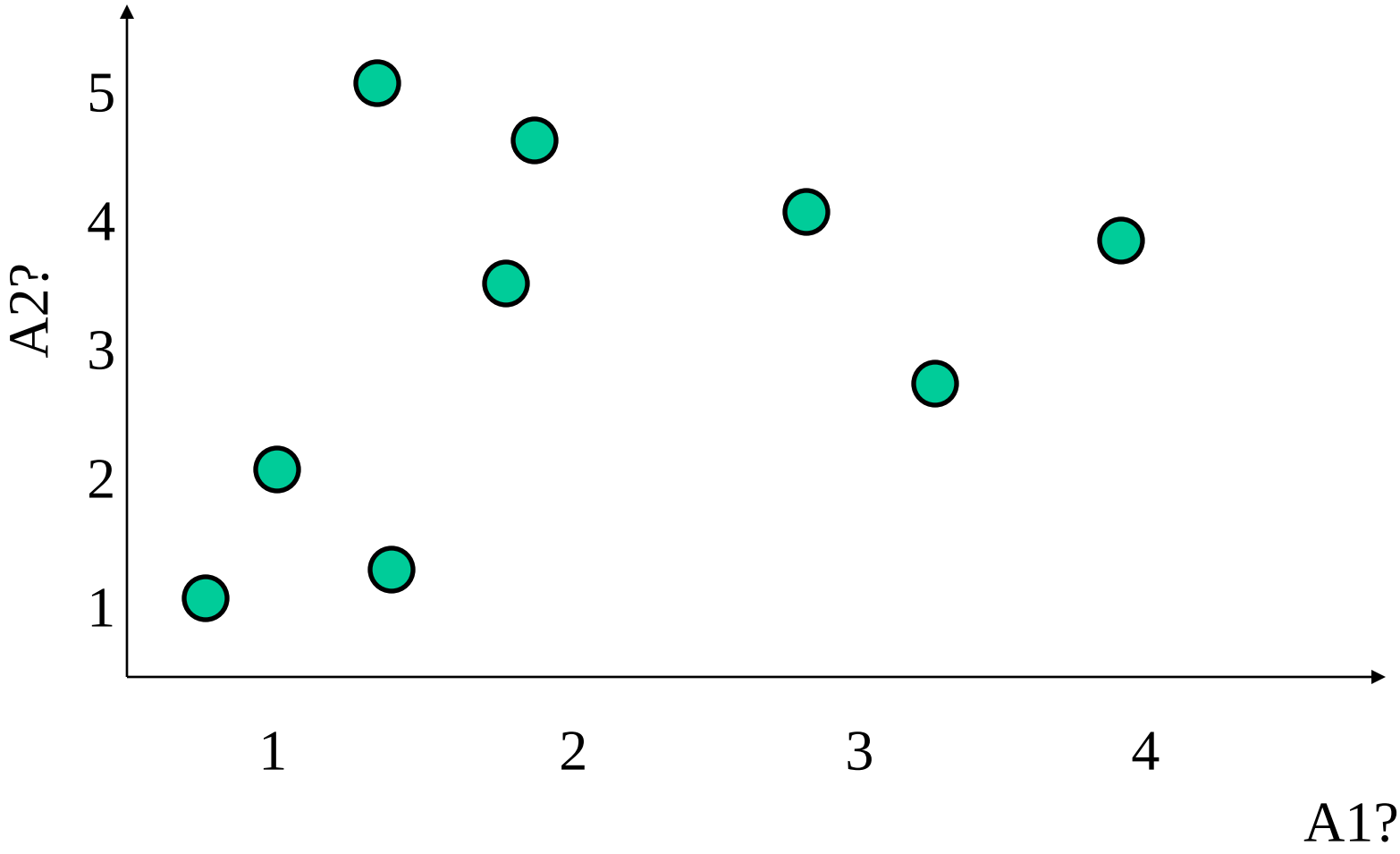
Ejemplo de salidas

- Conjunto de clases
  - **Clase1**: ejemplo4, ejemplo6
  - **Clase2**: ejemplo2, ejemplo3, ejemplo5
  - **Clase3**: ejemplo1
- Jerarquía de clases

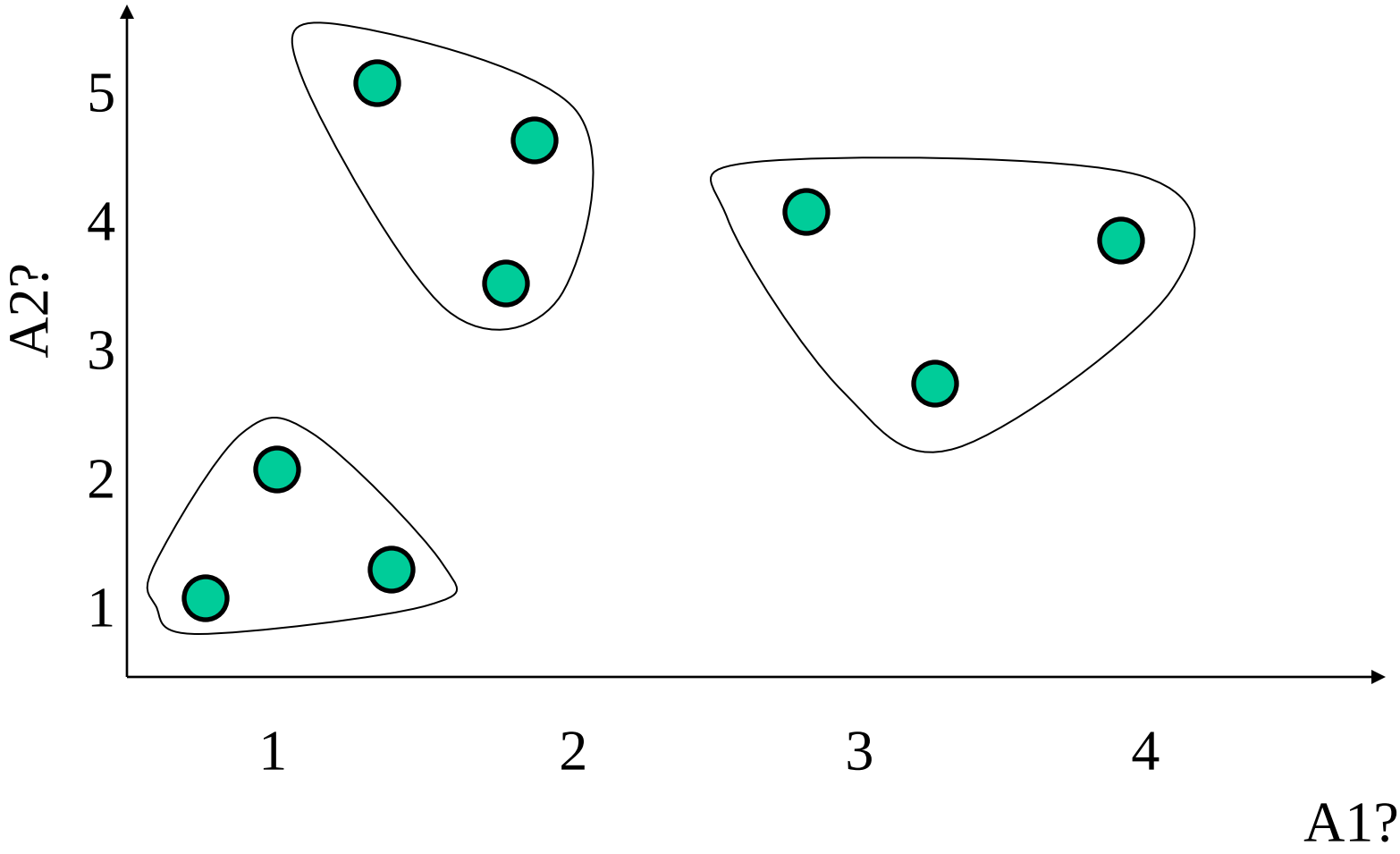


Agrupamiento

Ej. Problema a resolver, k=3

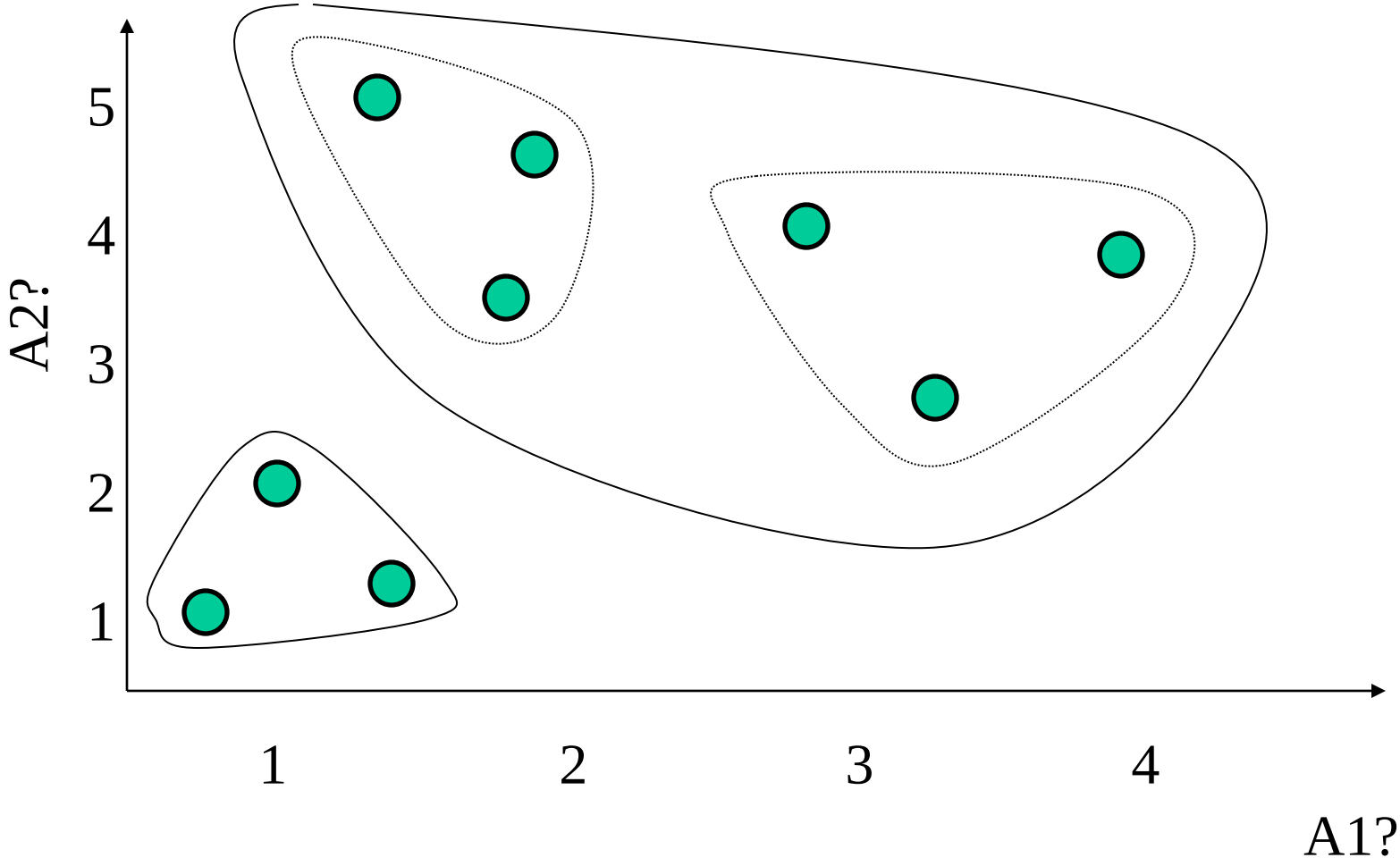


Solución 1



Agrupamiento

Solución 2



Agrupamiento

Agrupamiento con k-medias

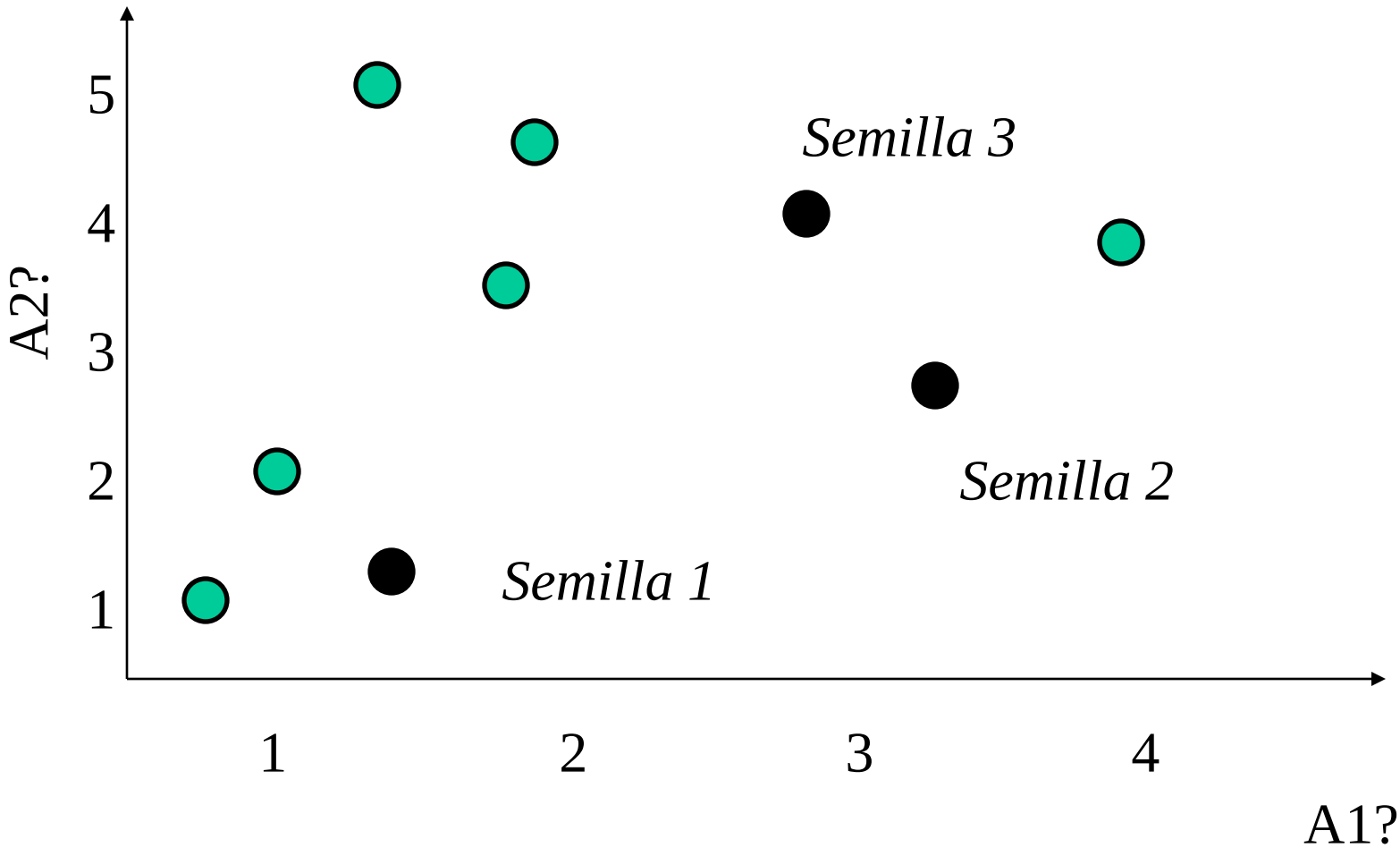
Problema a resolver: distribuir los ejemplos en k conjuntos, minimizando las distancias entre elementos dentro de cada grupo

Algoritmo

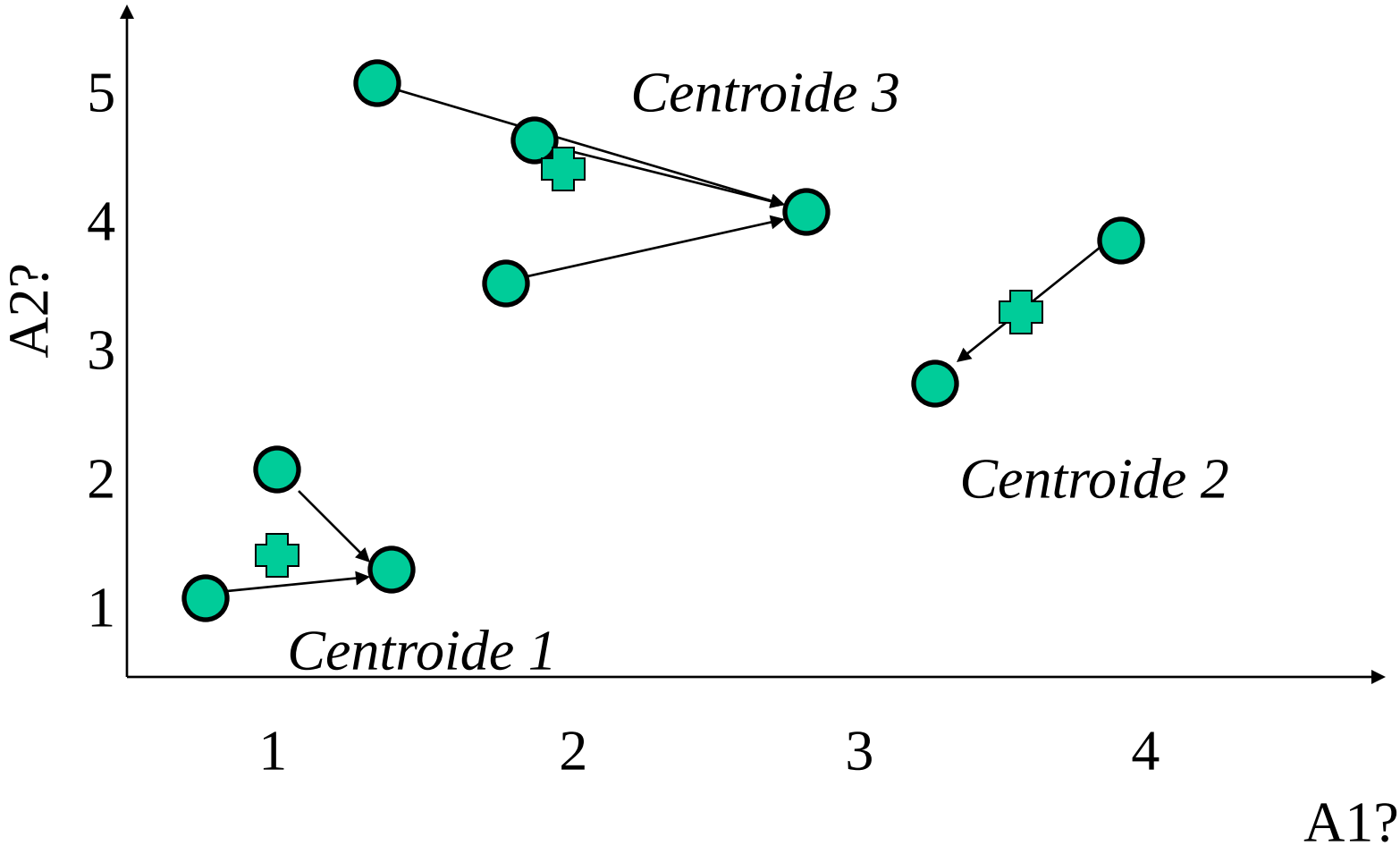
- Si se tienen que formar k clases, se eligen k ejemplos que actúan como semillas
- Cada ejemplo se añade a la clase mas similar
- Cuando se termina, se calcula el centroide de cada clase, que pasan a ser las nuevas semillas
- Se repite hasta que se llega a un criterio de convergencia (p.e. Dos iteraciones no cambian las clasicaciones de los ejemplos)

- Es un método de optimización local. Depende de la elección de la semilla inicial
  - Puede iterarse con varias semillas y seleccionar el mejor
  - Variantes para agrupamiento jerárquico

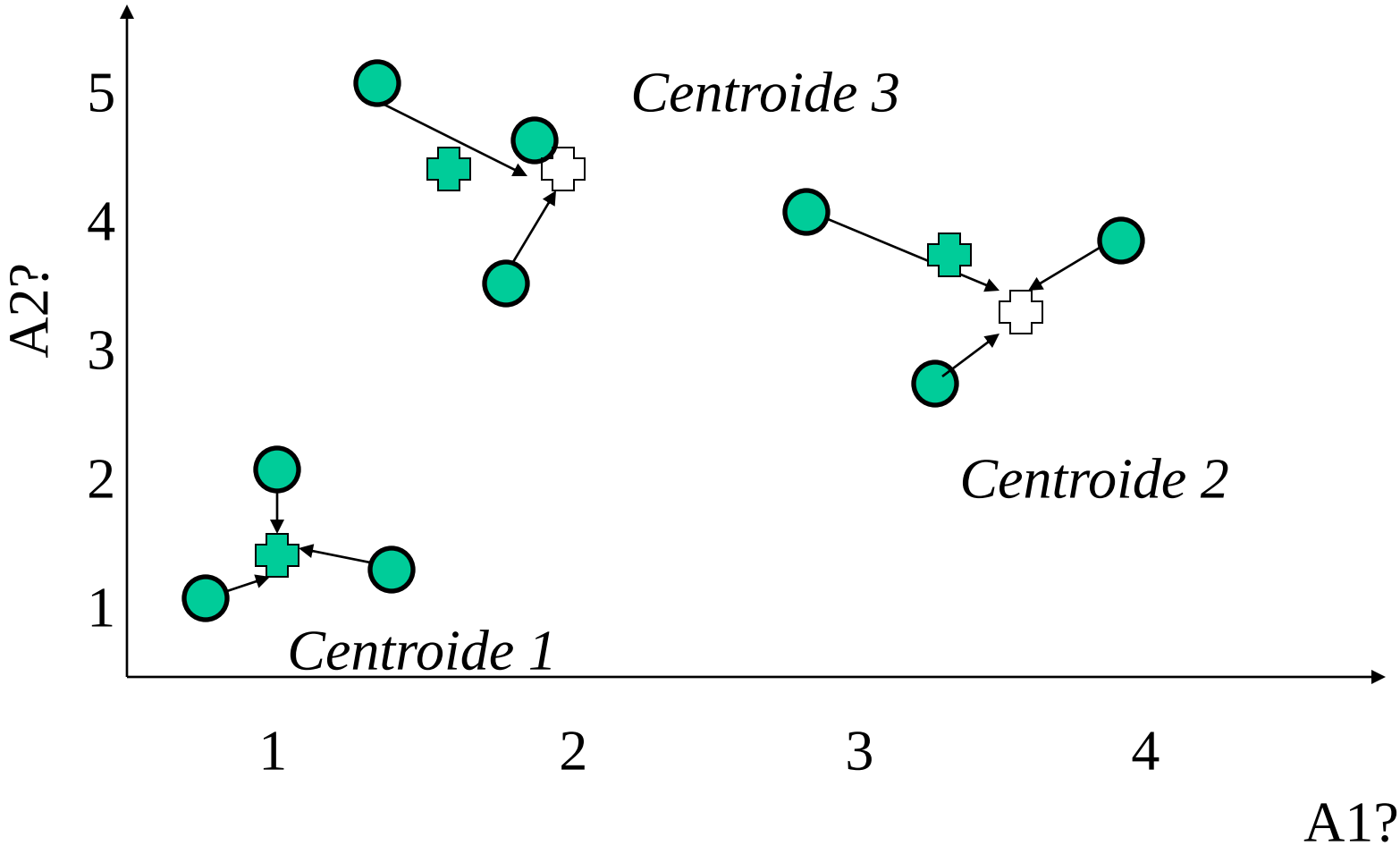
Ejemplo de k-medias. Inicio



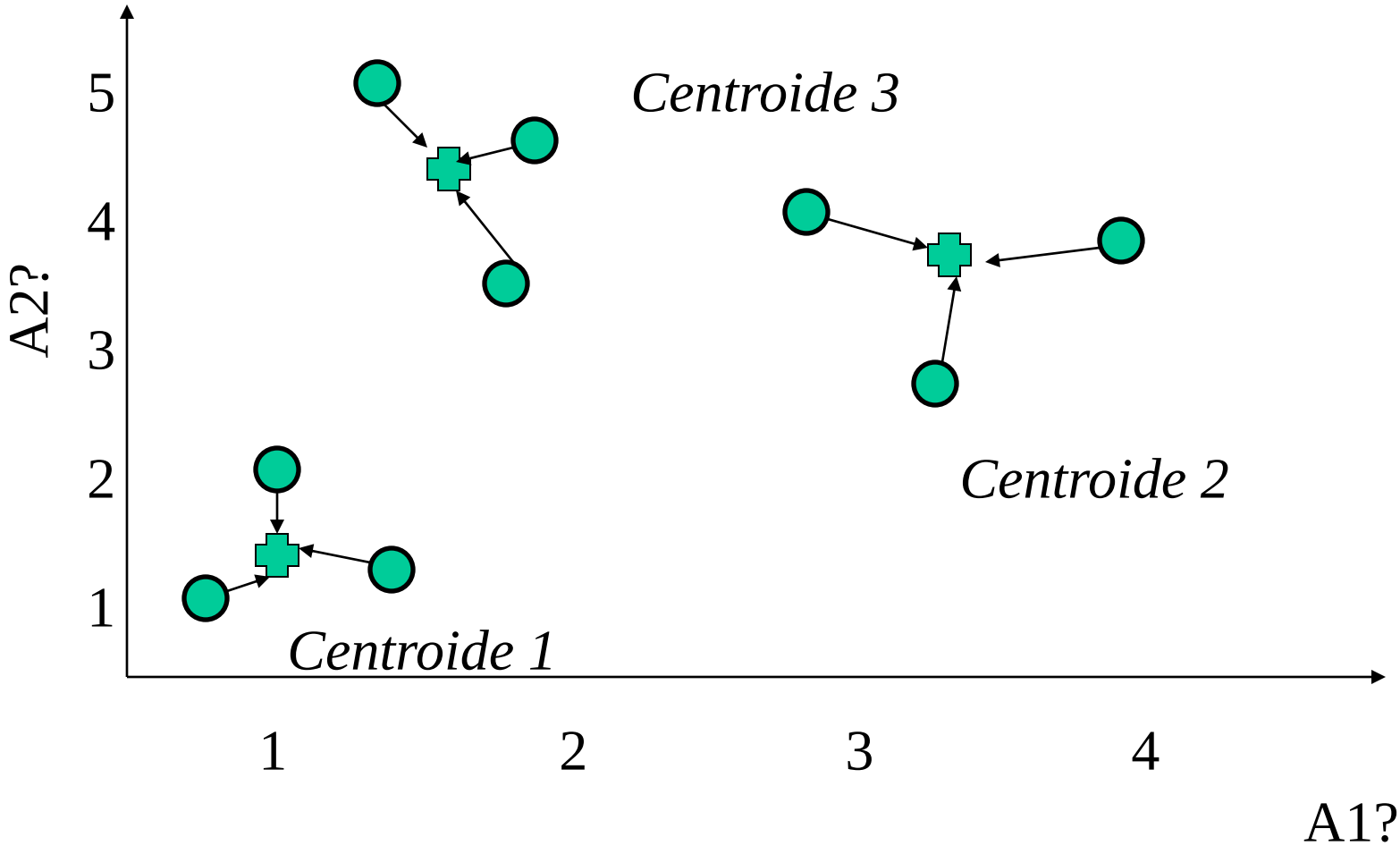
Ejemplo de k-medias. Iteración 1



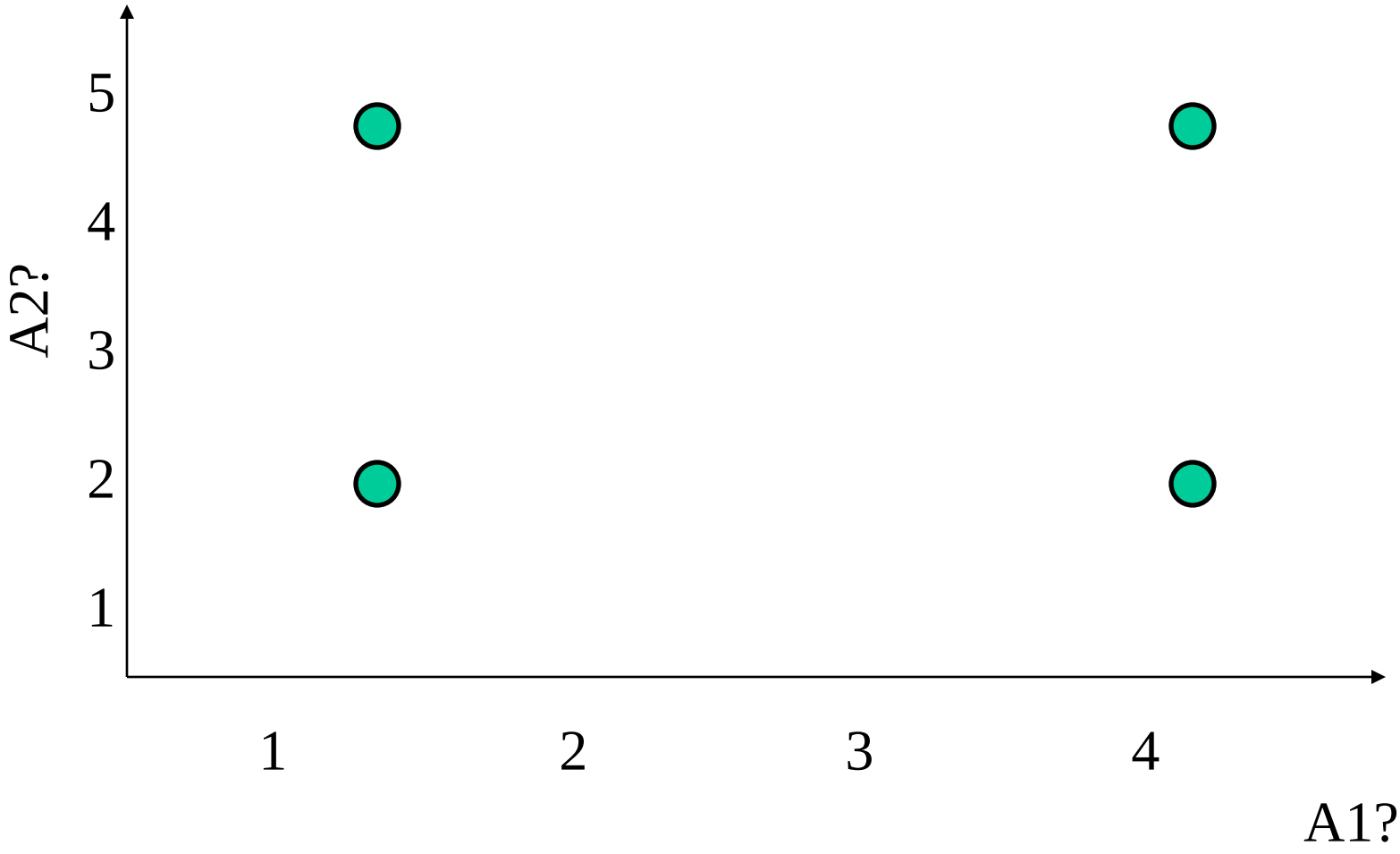
Ejemplo de k-medias. Iteración 2



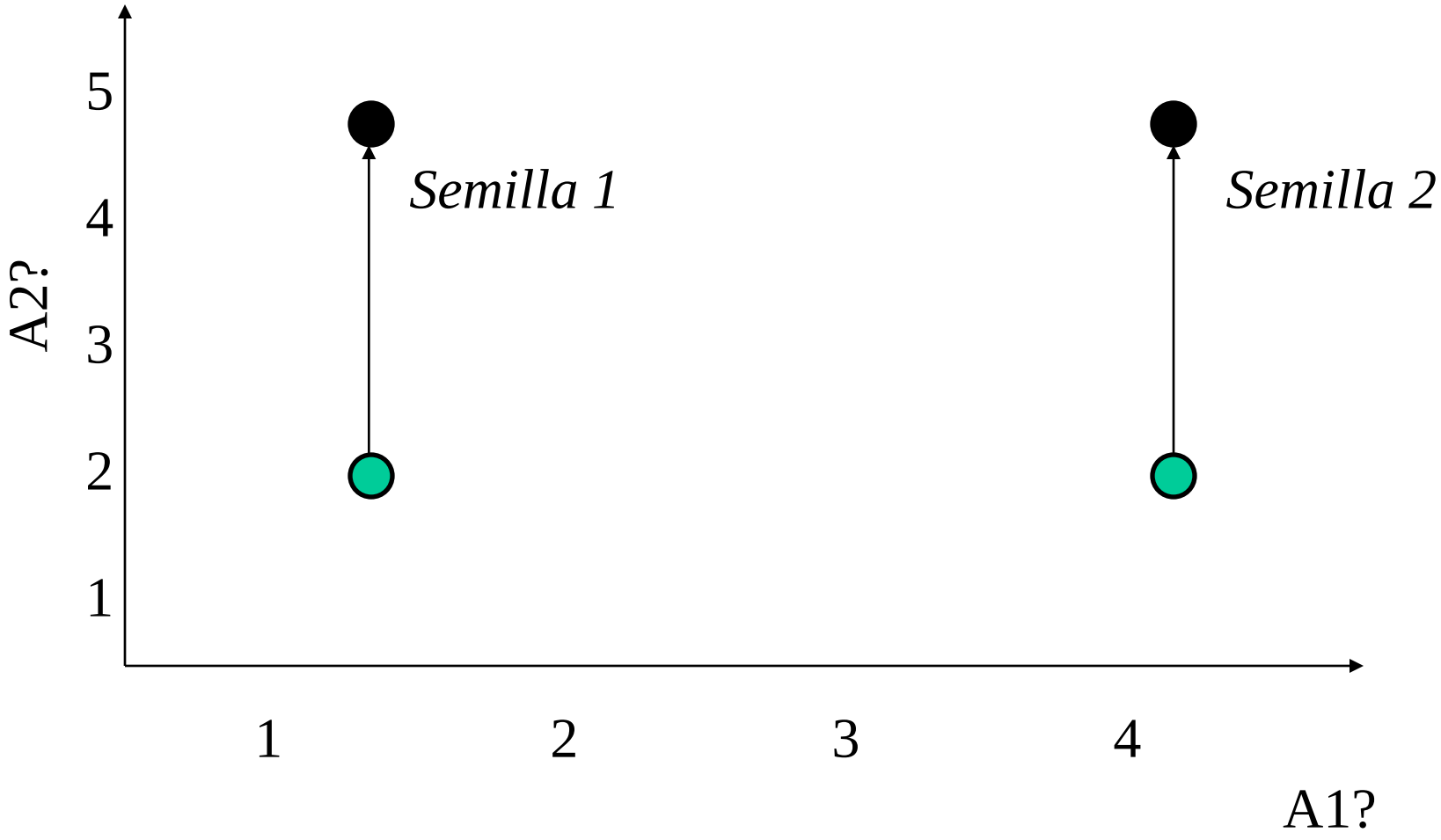
Ejemplo de k-medias. Iteración 3



Ej2. Problema a resolver, k=2

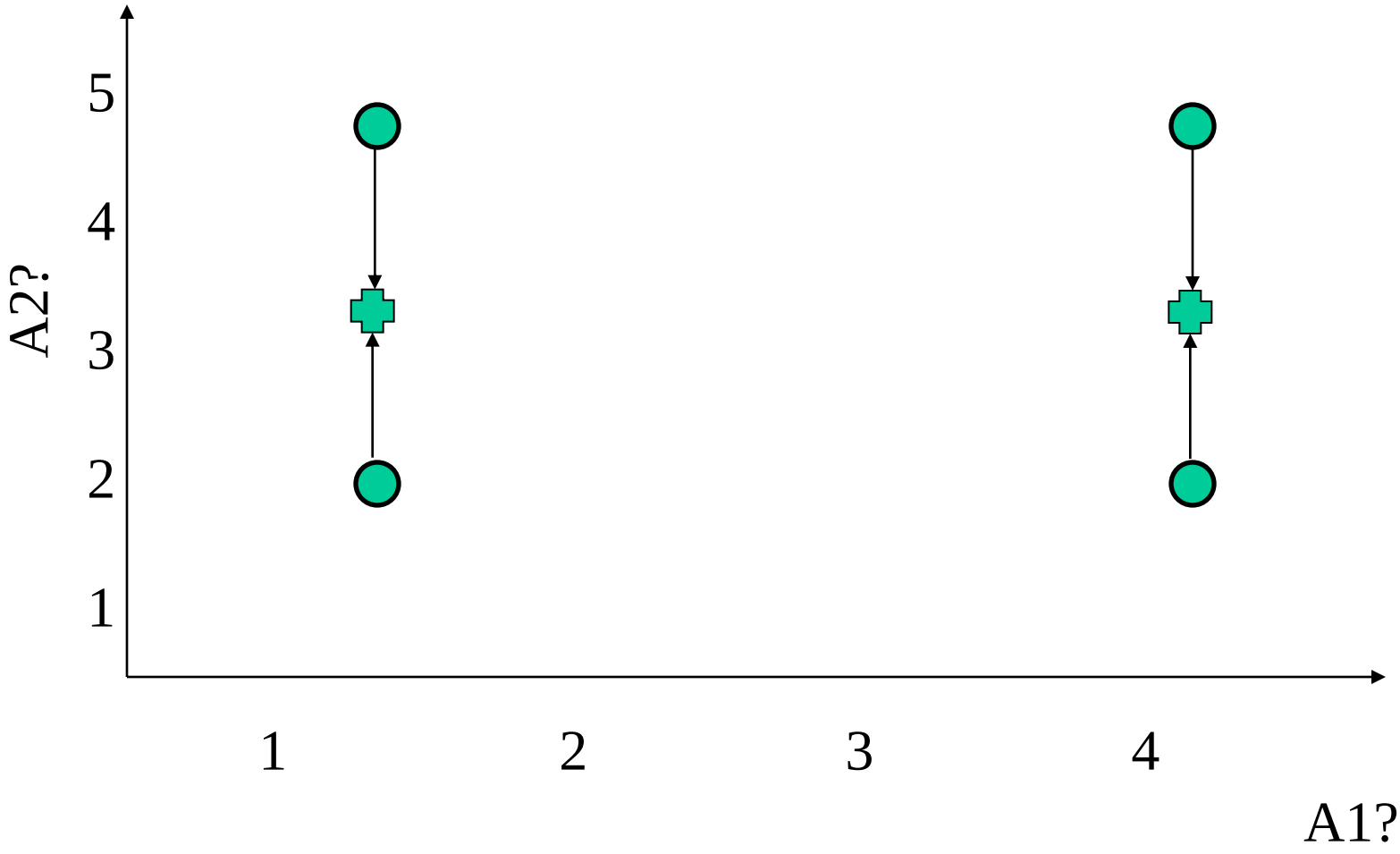


Ej2. Inicio



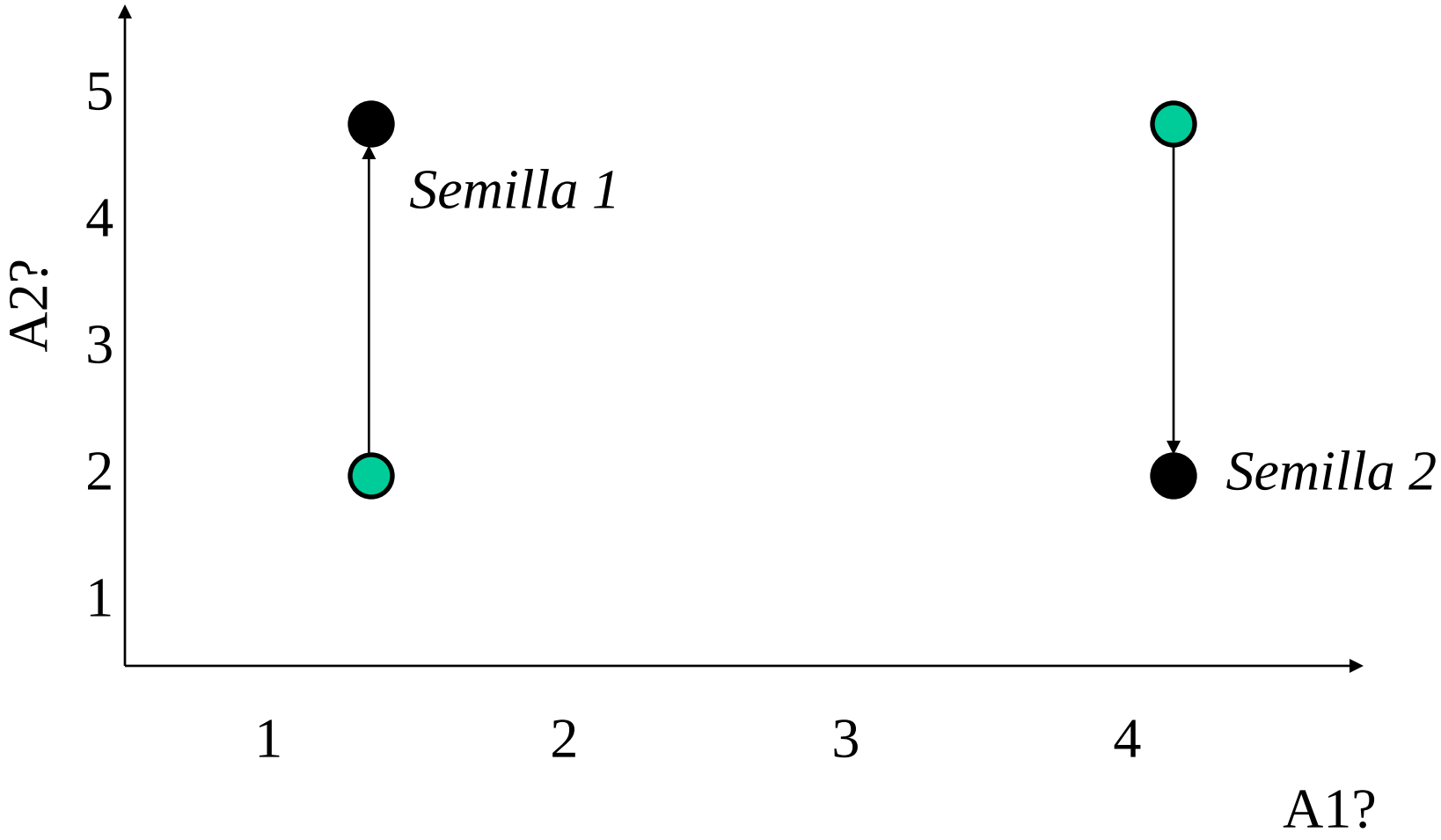
Agrupamiento

Ej2. Situación estable



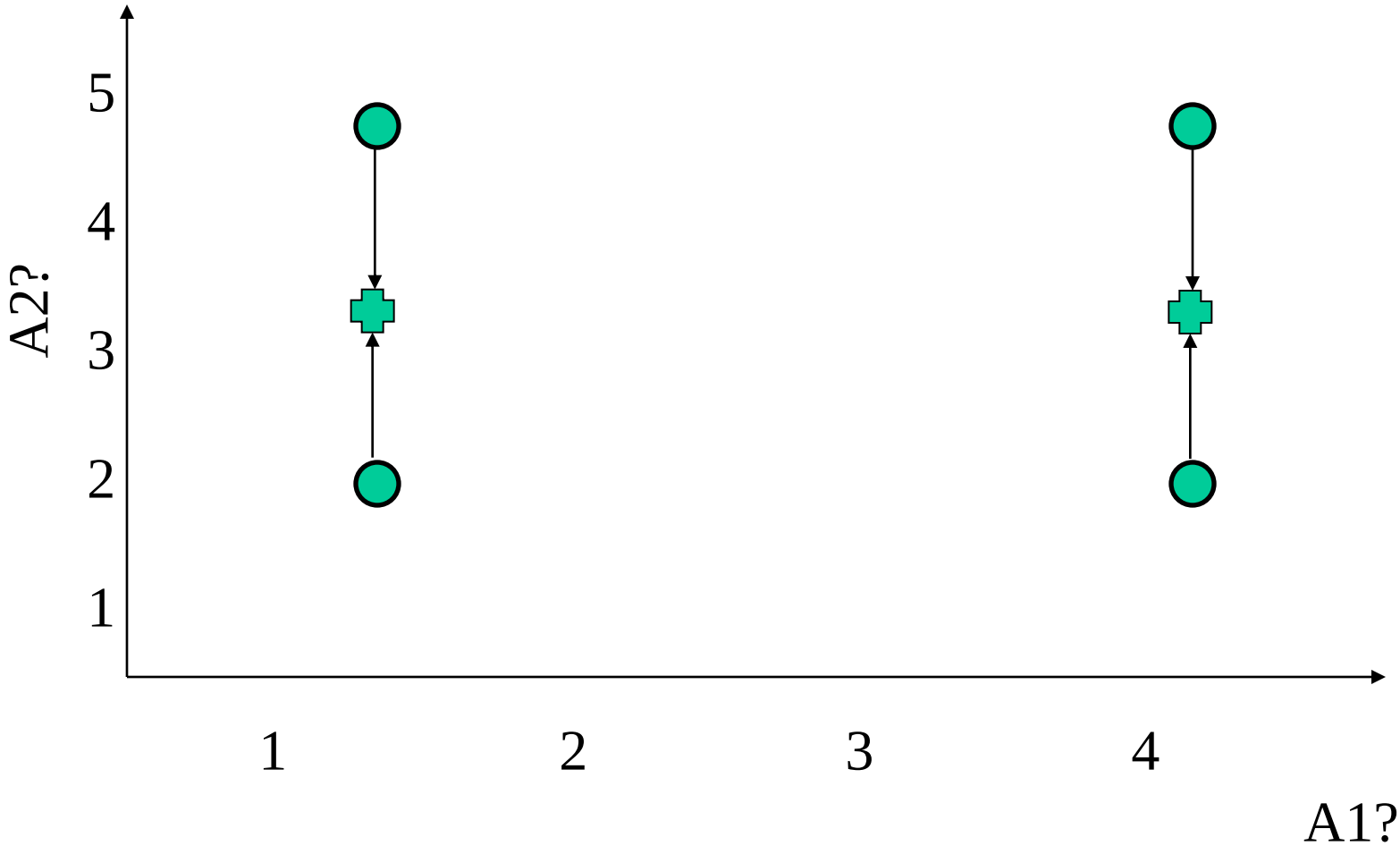
Agrupamiento

Ej2. Inicio



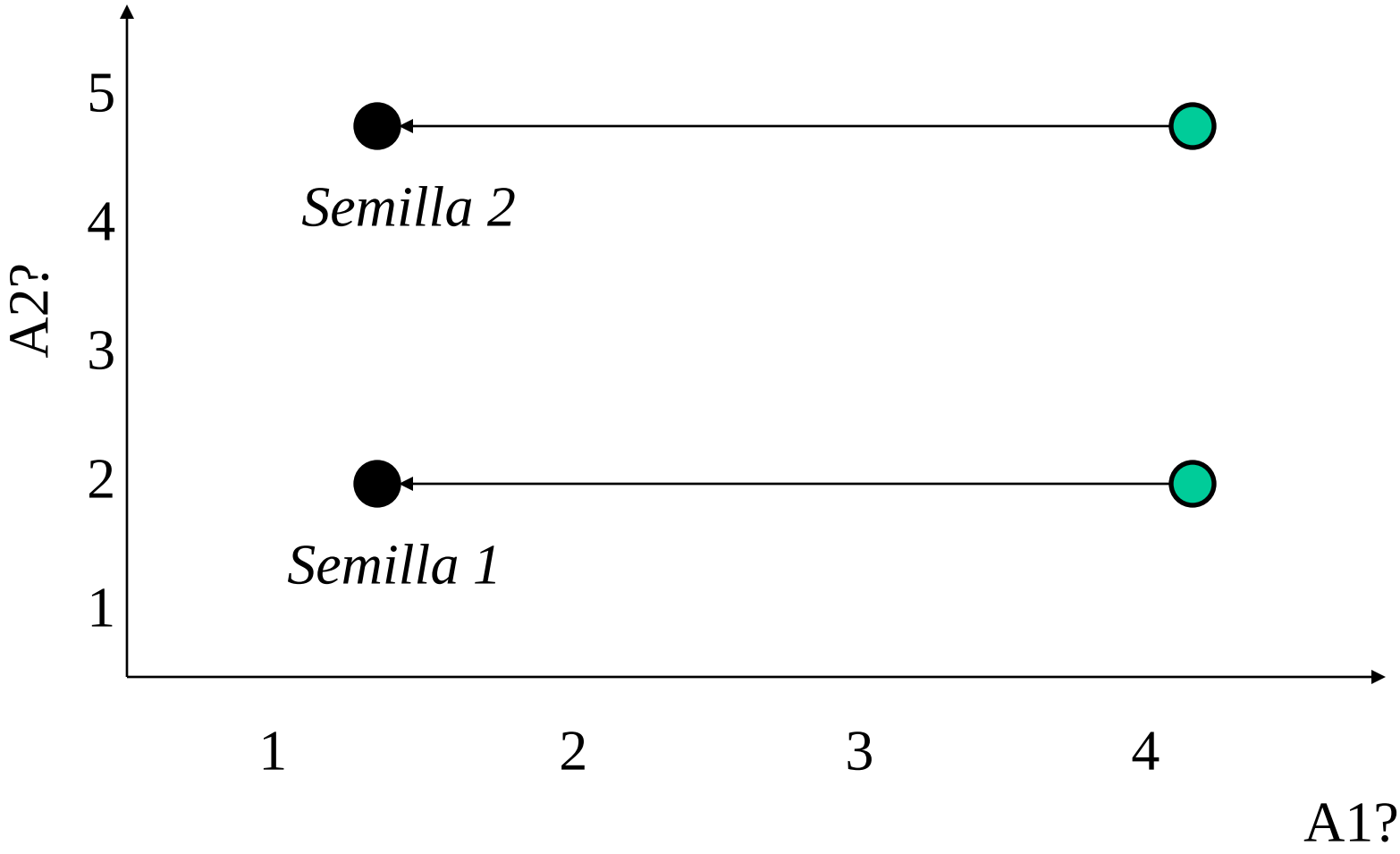
Agrupamiento

Ej2. Situación estable



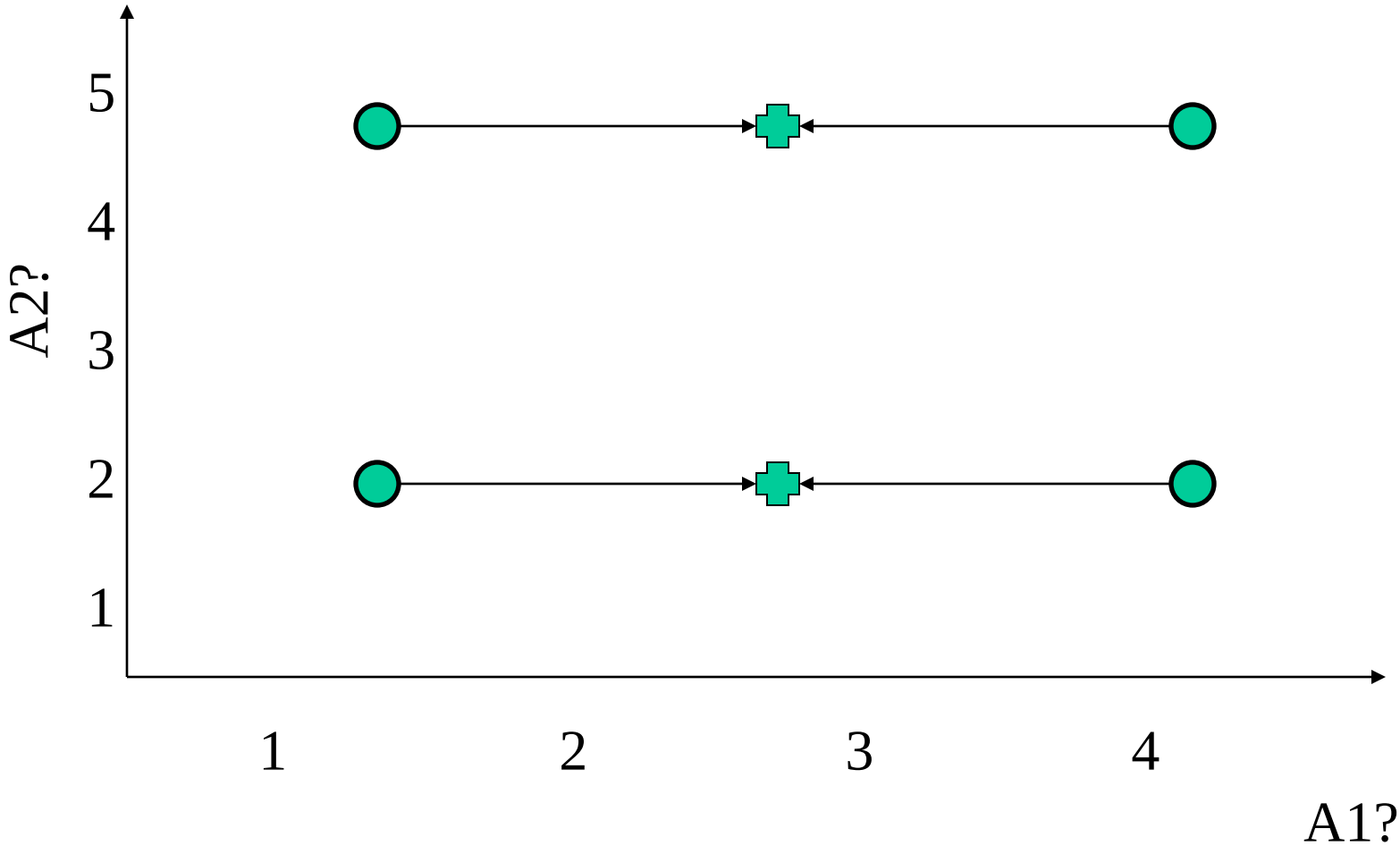
Agrupamiento

Ej2. Inicio



Agrupamiento

Ej2. Situación estable



Agrupamiento

Cálculo de la distancia

- Dados dos ejemplos $\mathbf{X_i}$, $\mathbf{X_j}$, con atributos x_{il}, x_{jl}, l=1,…, F

$$d(X_i, X_j) = \sqrt{\sum_{l=1}^{F} (x_{il} - x_{jl})^2}$$

- La similitud puede estimarse con el inverso de la distancia
- Problemas:
  - atributos con diferentes rangos o importancias:
    - normalizar valores
    - distancia estadística (Mahalanobis):

$$d(\vec{X}_i, \vec{X}_j) = \sqrt{\sum_{l=1}^{F} \frac{(x_{il} - x_{jl})^2}{\sigma_l^2}}, \quad d(X_i, X_j) = \sqrt{(\vec{X}_i - \vec{X}_j)^t S^{-1} (\vec{X}_i - \vec{X}_j)}$$

  - atributos nominales:

$$d(x_{il}, x_{jl}) = \begin{cases} 1, & \text{si } x_{il} \neq x_{jl} \\ 0, & \text{si } x_{il} = x_{jl} \end{cases}$$

Agrupamiento

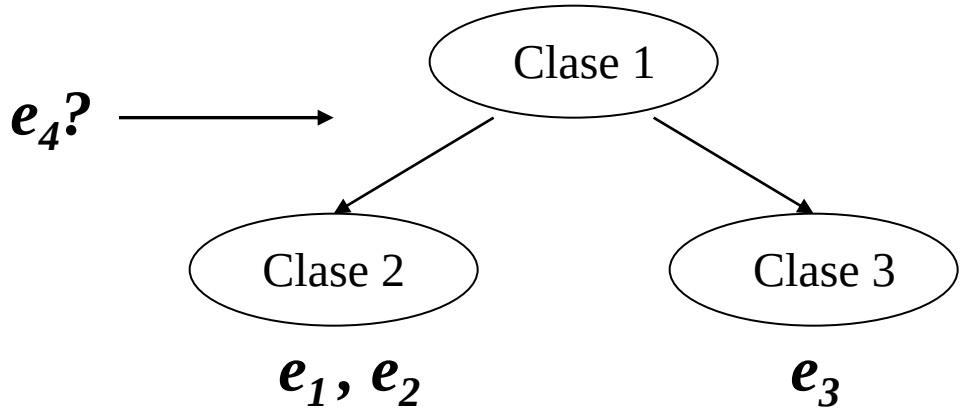
Clustering conceptual jerárquico

- Problema de los enfoques numéricos: distancia cuando atributos son no numéricos
- En el agrupamiento conceptual una partición de los datos es buena si cada clase tiene una buena interpretación conceptual (modelo cognitivo de jerarquías)
- Hay dos tareas (Fisher y Langley, 85 y 86):
  - **agrupamiento**: determinación de subconjuntos útiles de un conjunto de datos (métodos numéricos)
  - **caracterización**: determinación de un concepto por cada subconjunto descrito extensionalmente (métodos conceptuales)

- COBWEB es un algoritmo incremental de agrupamiento que forma un árbol añadiendo ejemplos uno por uno
  - La actualización con un nuevo ejemplo puede suponer ubicarlo en su hoja más adecuada, o bien una re-estructuración con la nueva información

Agrupamiento

Ejemplo



- Varias alternativas de particiones:
  - P_1=[{e_1, e_2},{e_3, e_4}]
  - P_2=[{e_1, e_2, e_3,}, {e_4}]

- Calidad de cada partición?
- Creación de una nueva clase?: [{e_1, e_2},{e_3}, {e_4}]

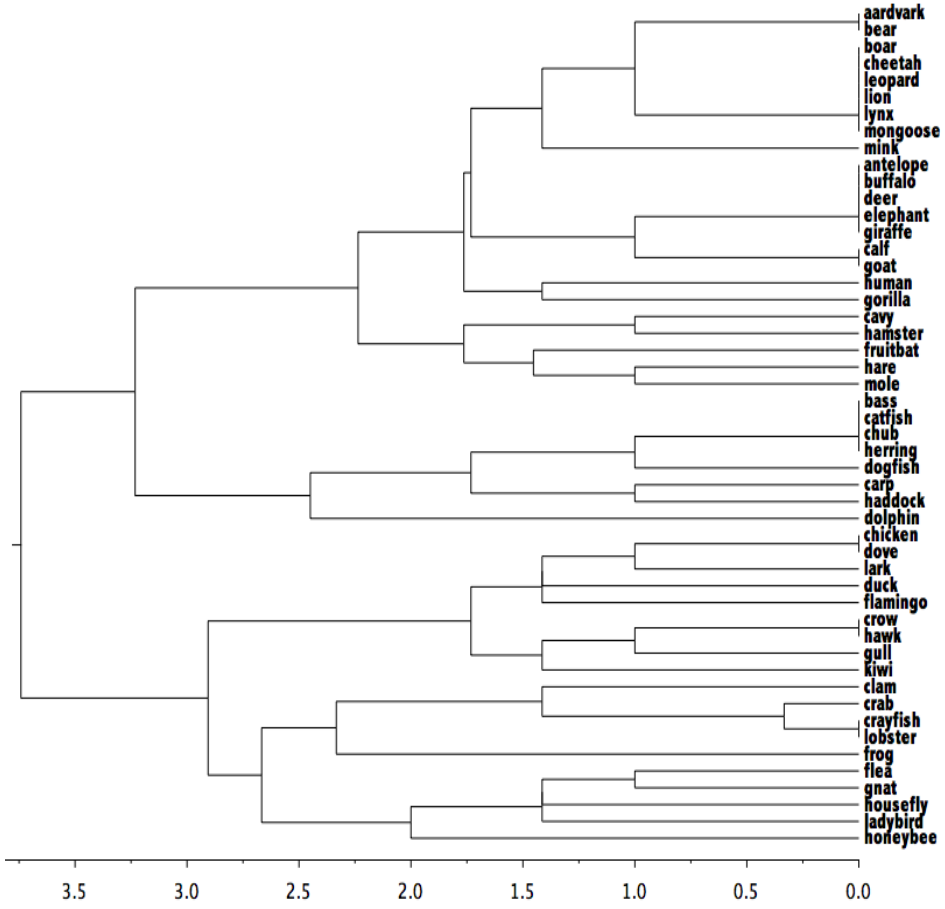
Agrupamiento

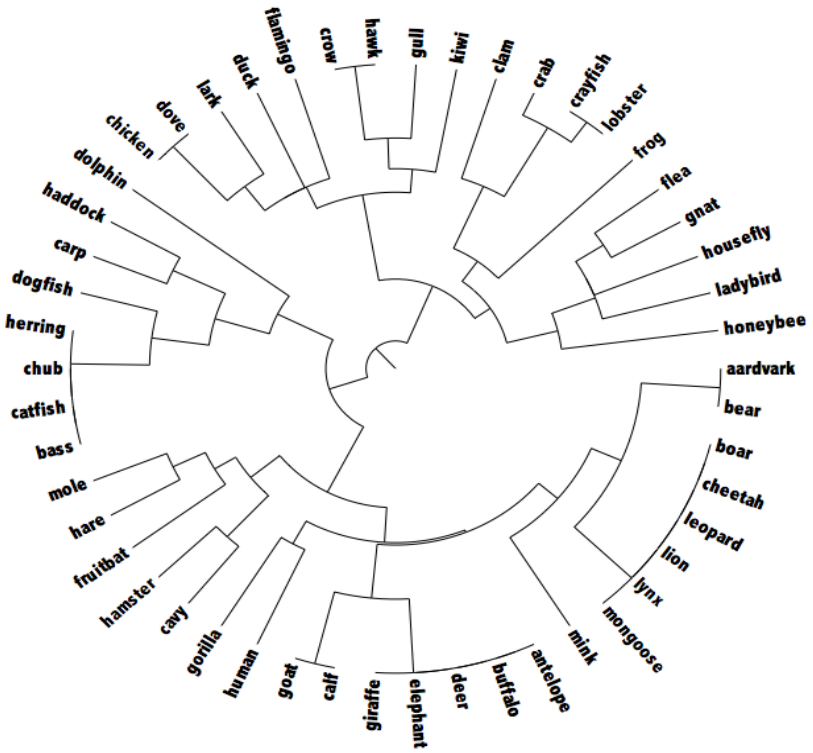
Clustering jerárquico

- División recursiva: jerarquía conocida como *dendograma*
  - La profundidad es proporcional a la disimilaridad entre sus hijos

Example hierarchical clustering

- 50 examples of different creatures from the zoo data





Creación del árbol

- COBWEB crea incrementalmente un árbol cuyos nodos son conceptos probabilísticos
- La clasificación consiste en descender por las ramas cuyos nodos se equiparen mejor, basándose en los valores de varios atributos al mismo tiempo
- Realiza búsqueda en escalada en el espacio de árboles probabilísticos de clasificación
- Operadores:
  - Clasificar una instancia en una clase (nodo)
  - Crear una nueva clase
  - Combinar dos clases en una
  - Separar una clase en varias
  - Promocionar un nodo
- En cada nodo se guarda:
  - Número de instancias por debajo
  - En cada atributo Aj, valor l, número de ejemplos con $A_j = V_{jl}$

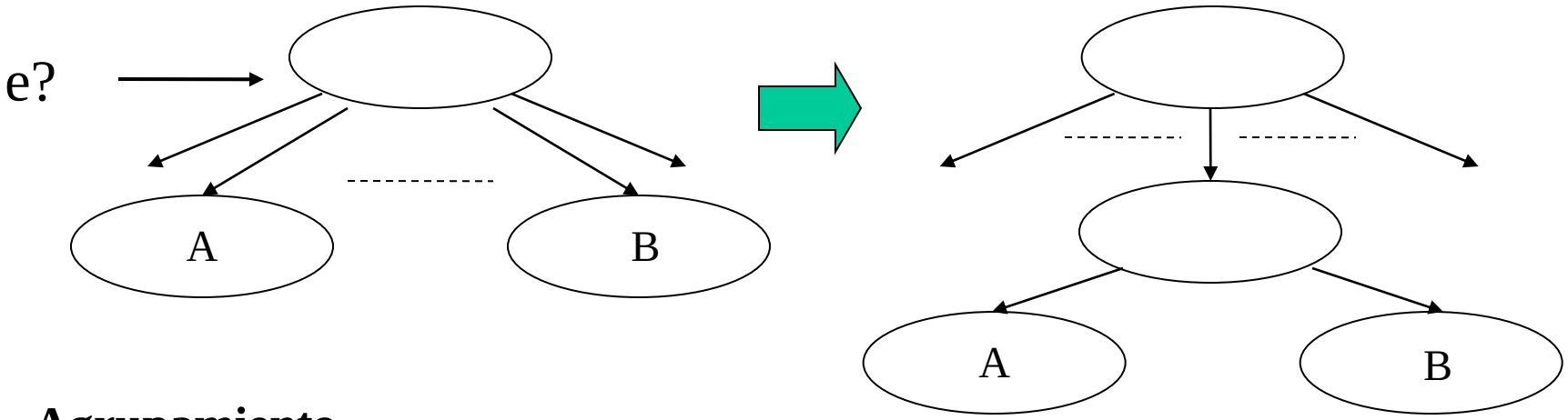
Agrupamiento

Operadores de agrupamiento

- Clasificar una instancia:
  - Se introduce la instancia en cada una de las clases sucesoras de la actual y se evalúan las distintas categorías
  - Si el mejor es un nodo hoja, se introduce en el. Si no, se llama recursivamente al algoritmo con él
- Crear una clase:
  - Se comparan las calidades de:
    - la partición creada por añadir la instancia a la mejor clase
    - la partición resultante de crear una nueva clase para esa instancia sola
  - En caso de ser mejor que este s ola, se crea un nuevo sucesor de la clase actual

Operadores de agrupamiento

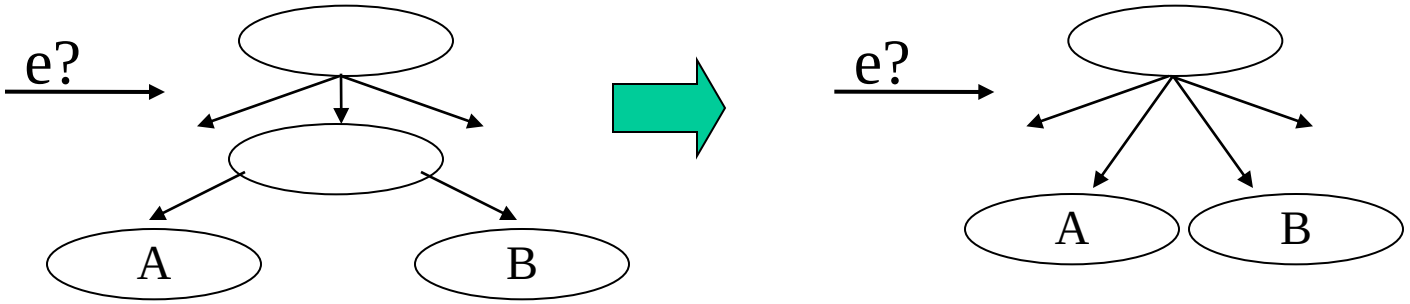
- Los dos operadores anteriores dependen mucho del orden de los ejemplos.
  - Un análisis completo de reestructuración no es viable
  - Se incorporan dos operadores básicos para compensar:
- **Combinar dos nodos**: Cuando se intenta clasificar una instancia en un nivel, se intentan mezclar las dos mejores clases y se evalúan las categorías de si están mejor solas o juntas en una misma categoría



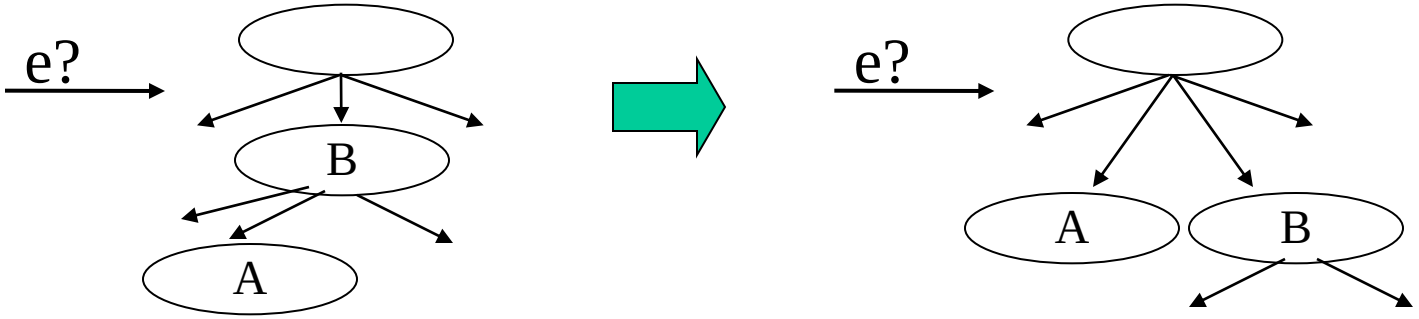
Agrupamiento

Operadores de agrupamiento

- **Separar dos nodos**: Cuando se intenta introducir una instancia en una clase que tenga sucesores, se estudia si se pueden subir los sucesores al mismo nivel que la clase



- **Promocionar un nodo**: El mismo operador anterior, pero individualizado para un s olo nodo



Agrupamiento

Heurística de búsqueda

- Define el nivel básico (aspecto cognitivo)
- Ayuda a considerar al mismo tiempo la similitud intra-clase y la disimilitud inter-clases
  - **Intra-clase** ($p(A_i = V_{ij} | C_k)$): cuanto mas grande, más ejemplos en la clase comparten el mismo valor (clases homogéneas)
  - **Inter-clases** ($p(C_k | A_i = V_{ij})$): cuanto mas grande, menos ejemplos de distintas clases comparten el mismo valor (separación entre clases)
- Calidad de una partición $\{C_1, C_2,...,C_M\}$, C_k mutuamente excluyentes, es un compromiso entre las dos:

$$\sum_{k=1}^{M} \sum_{i} \sum_{j} p(A_i = V_{ij}) p(C_k \mid A_i = V_{ij}) p(A_i = V_{ij} \mid C_k)$$

donde $p(A_i = V_{ij})$ prefiere valores más frecuentes

Agrupamiento

Utilidad de una categoría

- Como, según Bayes,

  $p(A_i=V_{ij})p(C_k|A_i=V_{ij}) = p(C_k)p(A_i=V_{ij}|C_k)$

  sustituyendo:

$$\sum_{k=1}^{M} p(C_k)\sum_i\sum_j p(A_i = V_{ij} \mid C_k)^2$$

  donde $\sum_i\sum_j p(A_i = V_{ij} \mid C_k)^2$ es el numero esperado de valores de atributos que se pueden predecir correctamente para un miembro cualquiera de C_k

- La utilidad de una categoría C_k es la mejora con respecto a no tener información de la partición:

$$p(C_k)\left[\sum_i\sum_j p(A_i = V_{ij} \mid C_k)^2 - \sum_i\sum_j p(A_i = V_{ij})^2\right]$$

Agrupamiento

Utilidad de una partición

- La utilidad de una partición P={C_1, C_2, ..., C_M} es:

$$CU(P) = \frac{1}{M} \sum_{k=1}^{M} p(C_k) \left[\sum_i \sum_j p(A_i = V_{ij} \mid C_k)^2 - \sum_i \sum_j p(A_i = V_{ij})^2 \right]$$

- Para evitar sobreadecuamiento (un grupo por ejemplo)
  - Factor 1/M:
  - Factor de poda (*cut-off*): mejora en utilidad por una subdivisión
- Si el valor V_{ij} de un atributo A_i es independiente de la pertenencia a la clase C_k:

  $p(A_i=V_{ij}|C_k) = p(A_i=V_{ij})$

  para ese valor la utilidad de la partición es nula
- Si lo anterior es cierto para todos los valores l, el atributo A_j es irrelevante

Agrupamiento

Utilidad de una partición

- Caso extremo: cada instancia en una categoría $\Rightarrow$ el numerador toma el valor máximo:

$$CU(P) = \frac{1}{M}\sum_{k=1}^{M} p(C_k)\left[n - \sum_i \sum_j p(A_i = V_{ij})^2 \right]$$

Numero de atributos

Atributos numéricos

- Los atributos numéricos se modelan con una distribución normal

$$f(a_i) = \frac{1}{\sqrt{2\pi}\sigma} \exp\left[-\frac{1}{2} \frac{(a_i - \mu)^2}{\sigma^2} \right]$$

- Equivalente a suma cuadrática de probabilidades:

$$\sum_j p(A_i = V_{ij} \mid C_k)^2 \leftrightarrow \int_{-\infty}^{+\infty} f^2(a_i) da_i = \frac{1}{2\sqrt{\pi}\sigma_i}$$

$$CU(C_1, C_2, ..., C_M) = \frac{1}{M} \sum_{k=1}^{M} p(C_k) \frac{1}{2\sqrt{\pi}} \sum_i \left(\frac{1}{\sigma_{ik}} - \frac{1}{\sigma_i} \right)$$

- Un grupo con un ejemplo tendría utilidad inifinita:
  - Agudeza (*acuity*): mínima varianza en un grupo

Agrupamiento

Algoritmo de agrupamiento

Nodo COBWEB (Instancia,Nodo) {
Actualizar los contadores de Nodo;
Si Nodo es hoja
Entonces IncluirInstancia (Instancia,Nodo);
    Devolver Nodo
Si no MejorNodo= MejorClase (Instancia,Nodo);
    Si es apropiado crear una nueva clase
    Entonces Nuevo-nodo:= CrearNodo (Instancia,Nodo);
        Devolver Nuevo-nodo
    Si es apropiado combinar dos nodos
    Entonces Nuevo-nodo:= CombinarNodos (Instancia,Nodo);
        Devolver COBWEB (Instancia,Nuevo-nodo)
    Si es apropiado separar dos nodos
    Entonces Nuevo-nodo:= SepararNodos (Instancia,Nodo);
        Devolver COBWEB (Instancia,Nodo)
    Si es apropiado promocionar un nodo
    Entonces Nuevo-nodo:= PromocionarNodo (Instancia,Nodo);
        Devolver COBWEB (Instancia,Nodo)
    Si no, Devolver COBWEB (Instancia,MejorNodo) }

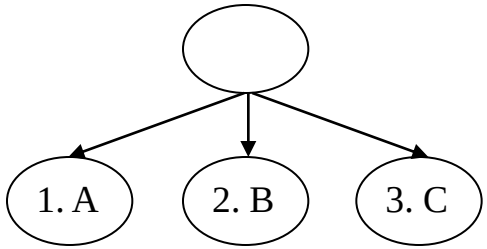
Ejemplo COBWEB

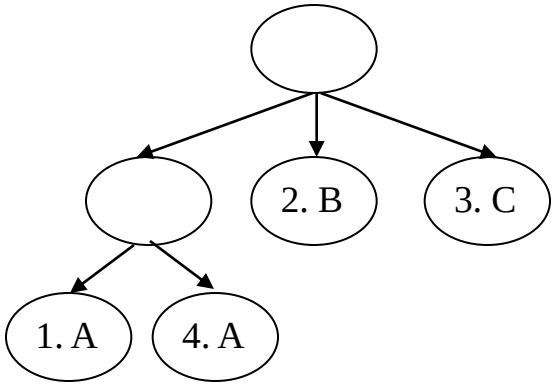
- **Reconocimiento de caracteres (OCR)**
  - **Identificar letras a partir de 20 fuentes, con distorsiones aleatorias**
  - **Parámetros numéricos extraídos de las imágenes**

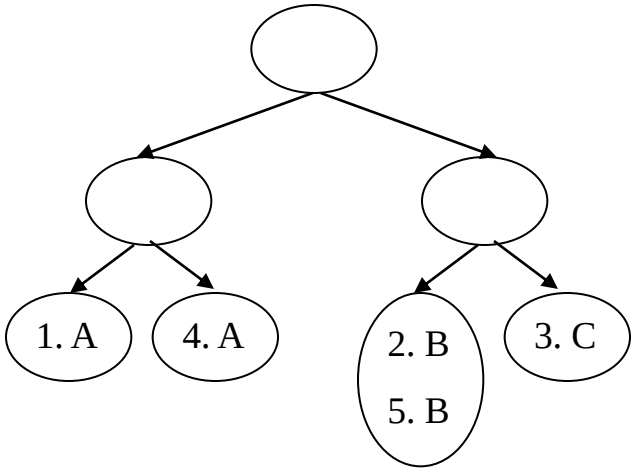
x-box	y-box	width	high	onpix	x-bar	y-bar	x2bar	y2bar	xybar	x2ybar	xy2bar	x-ege	xegvy	y-ege	yegvx	letter
3	11	5	8	3	13	4	5	3	12	1	8	2	6	4	9	A
2	4	4	3	3	9	7	2	6	11	4	7	4	7	5	9	B
4	6	5	4	3	6	7	5	6	11	8	13	2	10	3	9	C
4	10	6	7	2	9	4	3	2	8	1	8	3	7	3	8	A
3	4	5	3	3	8	7	2	6	11	5	7	2	8	4	9	B
5	5	6	8	2	6	7	7	10	8	6	15	1	9	4	9	C
3	5	5	3	2	6	2	2	2	5	2	8	2	6	3	6	A
3	6	4	4	3	11	6	3	6	11	3	7	2	8	4	11	B
4	9	5	6	4	5	8	7	6	9	8	14	2	10	4	10	C
4	10	7	7	2	8	7	3	0	7	0	8	3	7	2	8	A
9	14	7	8	4	9	6	6	6	11	4	9	6	7	7	10	B
3	10	5	7	3	4	9	6	6	6	8	14	1	8	4	10	C
5	9	7	7	5	8	3	1	2	6	2	7	3	5	4	7	A
2	7	3	5	2	6	6	9	7	6	7	7	2	8	8	10	B
8	13	5	8	2	8	6	7	7	12	5	9	2	10	5	9	C

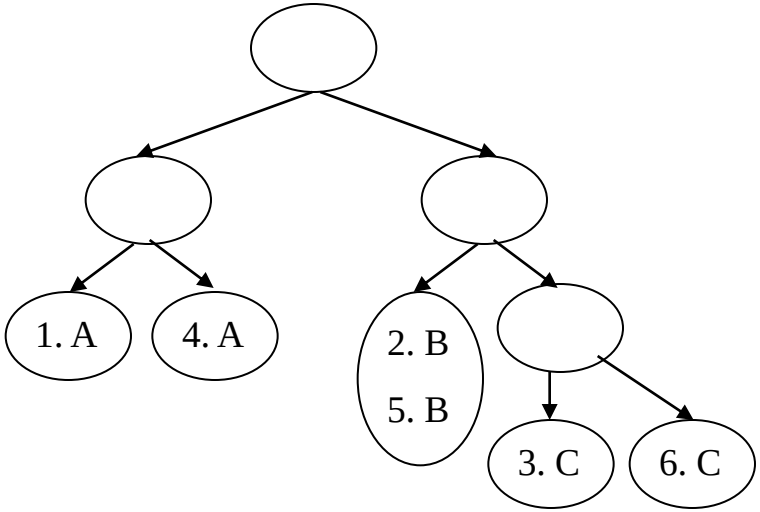
Agrupamiento

Ejemplo (cont)



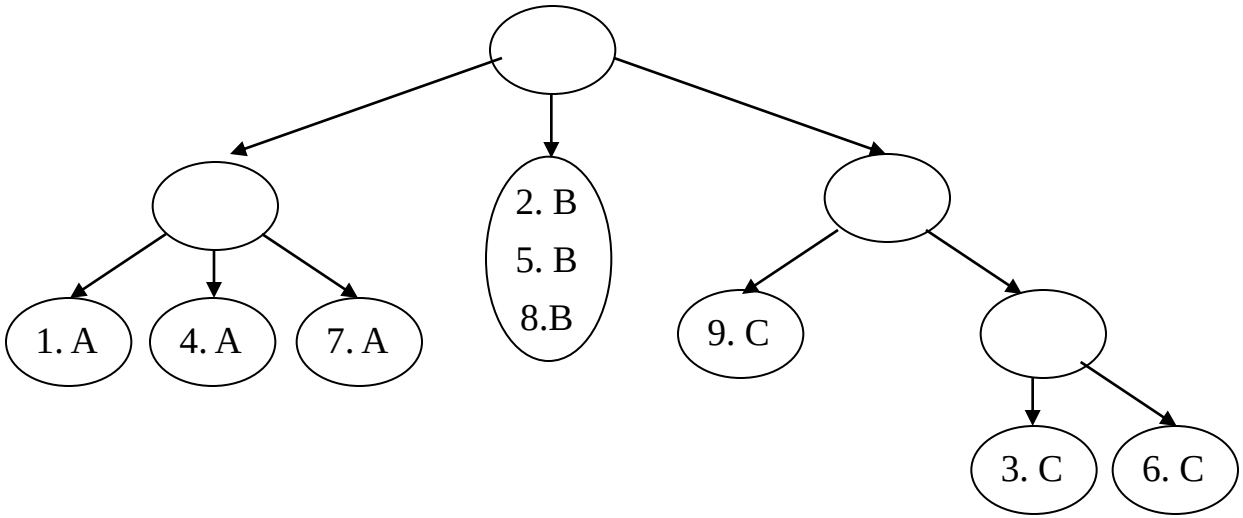




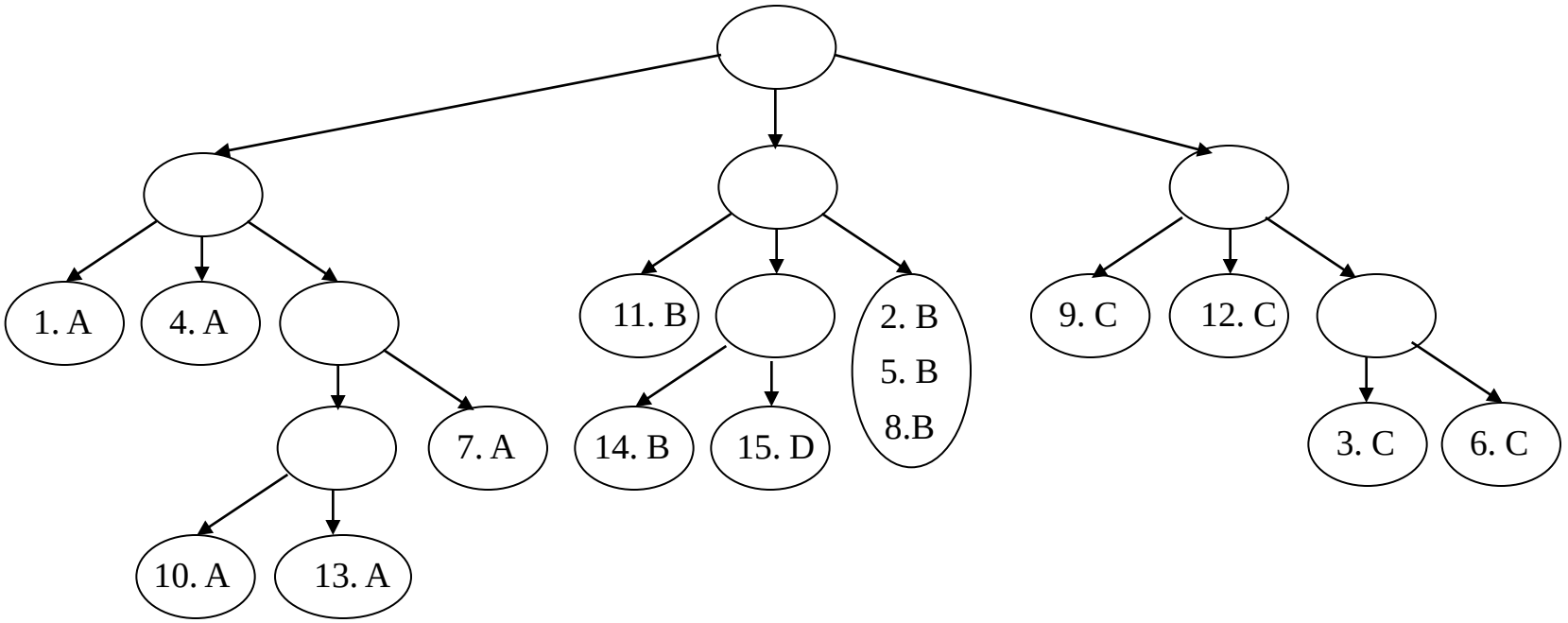


Agrupamiento

Ejemplo (cont)



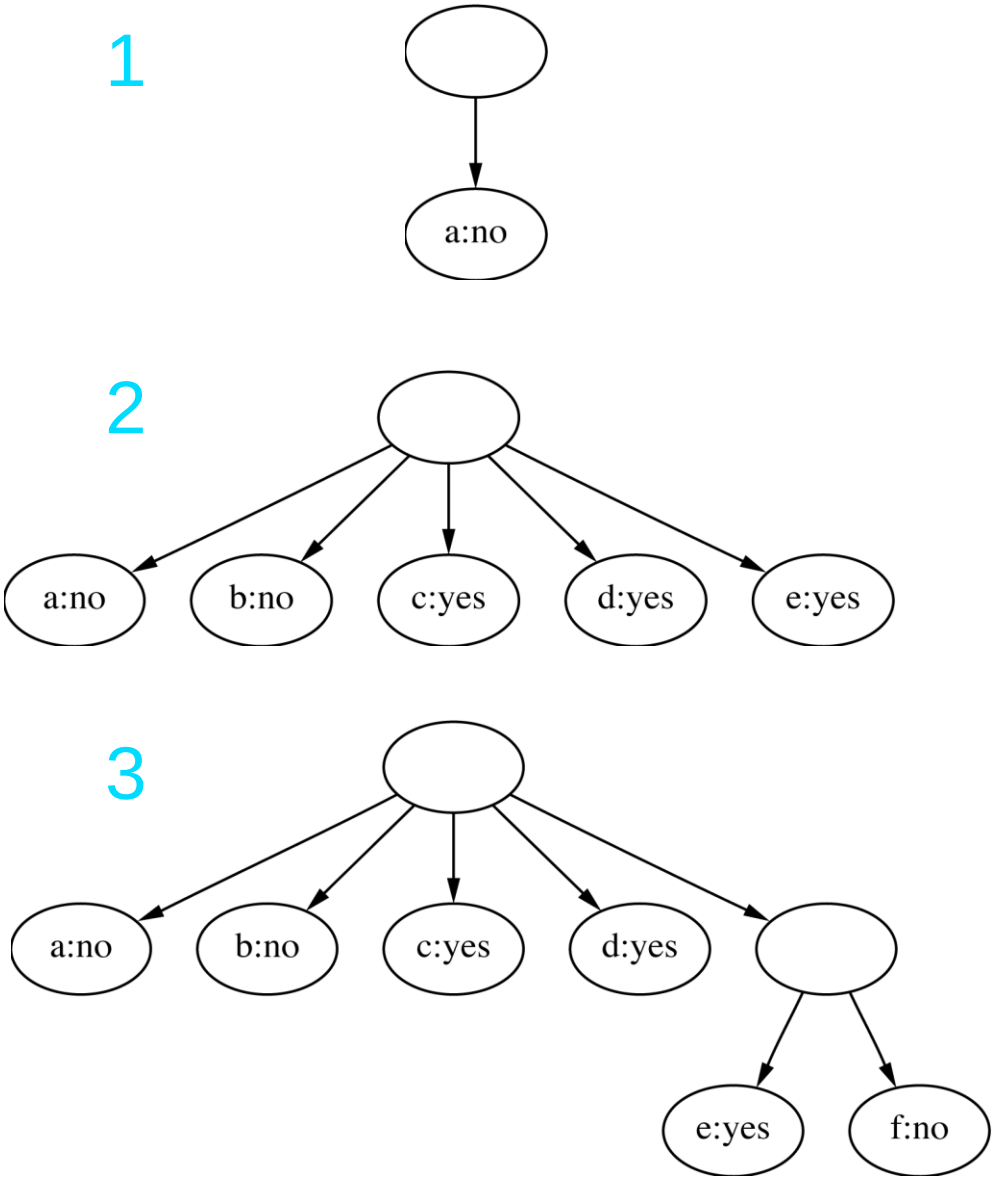
Ejemplo (cont)



Agrupamiento

Ejemplo: weather data

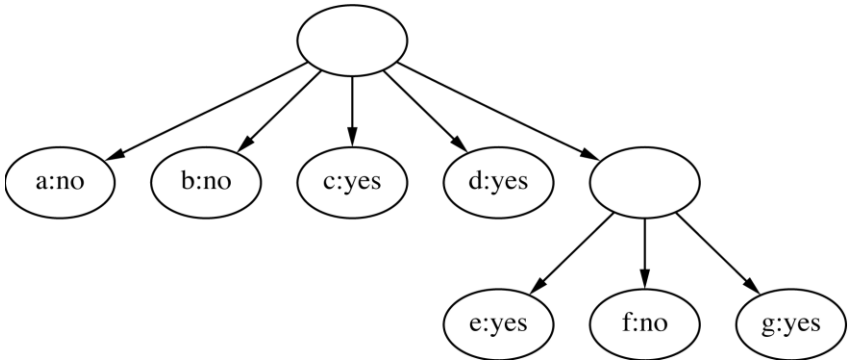
ID	Outlook	Temp.	Humidity	Windy
A	Sunny	Hot	High	False
B	Sunny	Hot	High	True
C	Overcast	Hot	High	False
D	Rainy	Mild	High	False
E	Rainy	Cool	Normal	False
F	Rainy	Cool	Normal	True
G	Overcast	Cool	Normal	True
H	Sunny	Mild	High	False
I	Sunny	Cool	Normal	False
J	Rainy	Mild	Normal	False
K	Sunny	Mild	Normal	True
L	Overcast	Mild	High	True
M	Overcast	Hot	Normal	False
N	Rainy	Mild	High	True



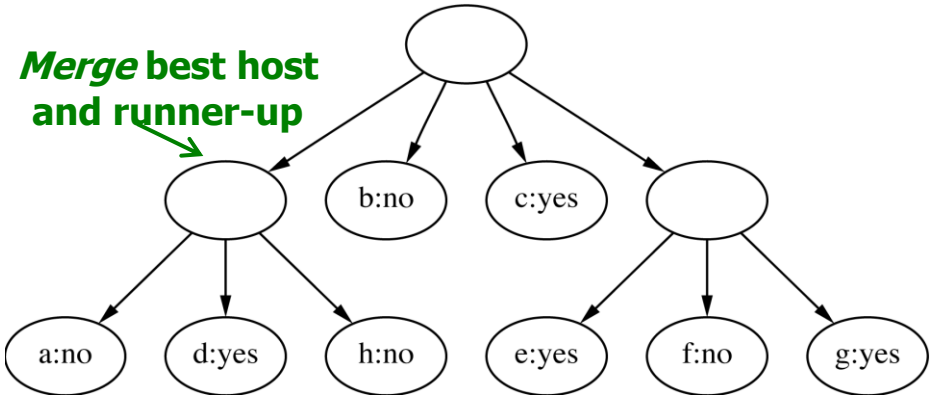
Clustering weather data

ID	Outlook	Temp.	Humidity	Windy
A	Sunny	Hot	High	False
B	Sunny	Hot	High	True
C	Overcast	Hot	High	False
D	Rainy	Mild	High	False
E	Rainy	Cool	Normal	False
F	Rainy	Cool	Normal	True
G	Overcast	Cool	Normal	True
H	Sunny	Mild	High	False
I	Sunny	Cool	Normal	False
J	Rainy	Mild	Normal	False
K	Sunny	Mild	Normal	True
L	Overcast	Mild	High	True
M	Overcast	Hot	Normal	False
N	Rainy	Mild	High	True

4

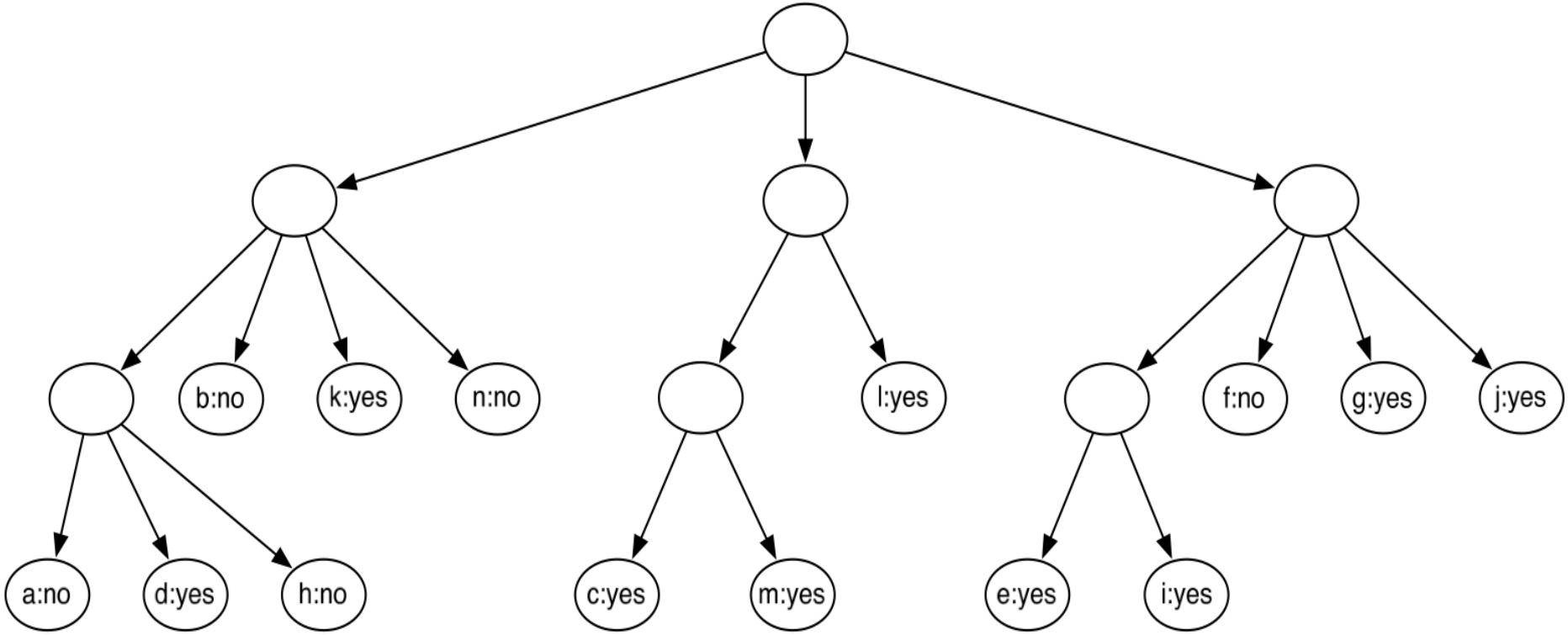


5

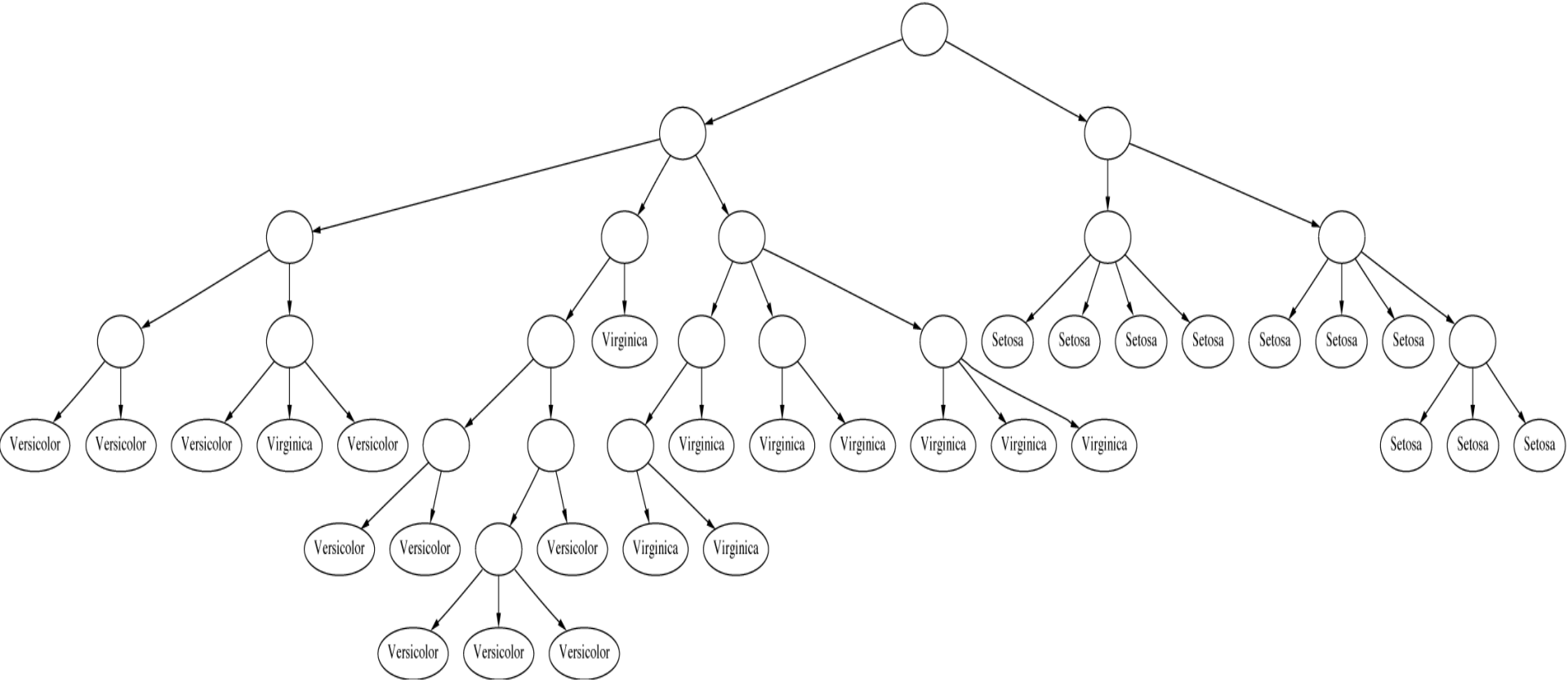


Consider *splitting* the best host if merging doesn't help

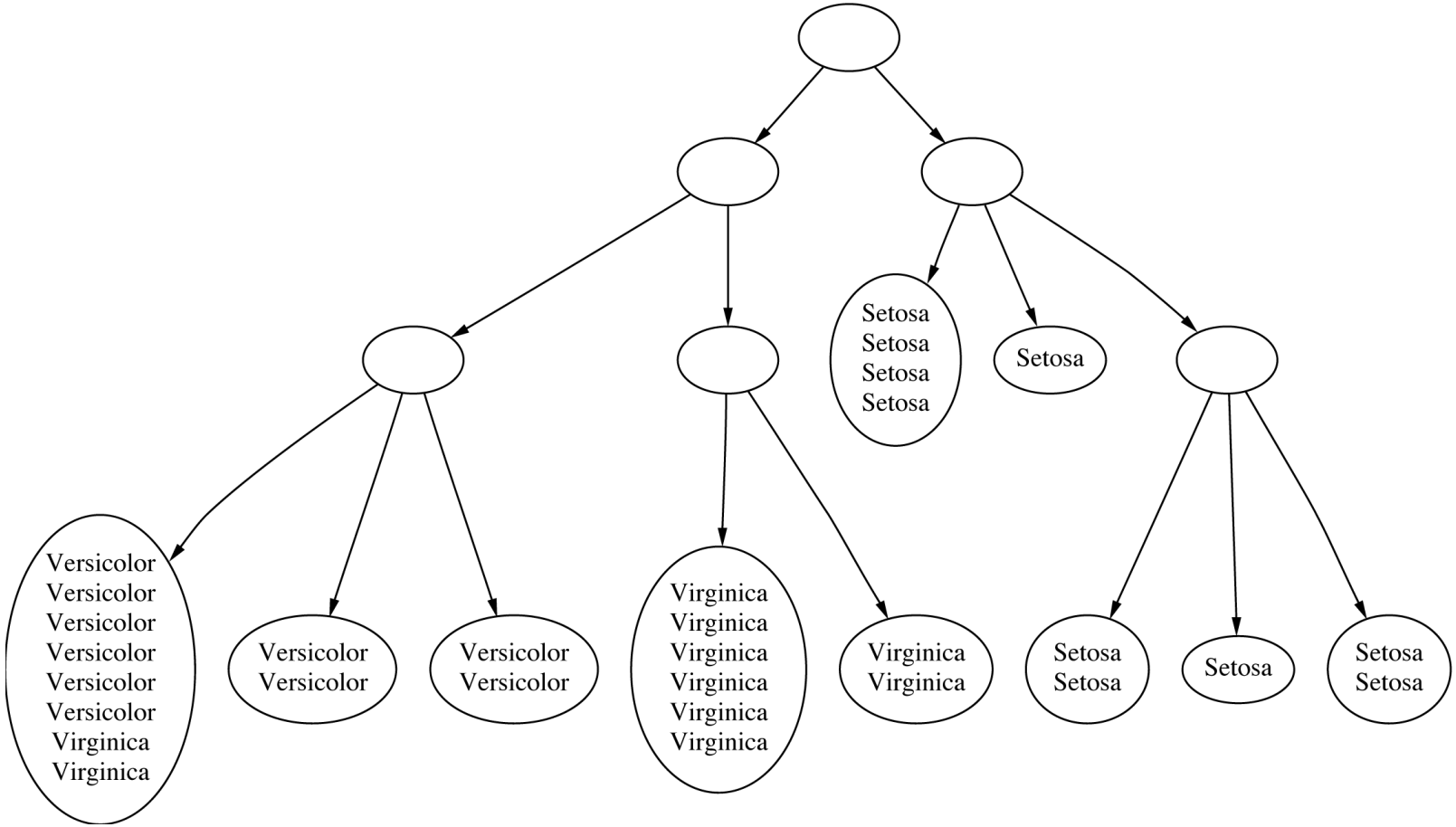
Final hierarchy



Example: the iris data (subset)

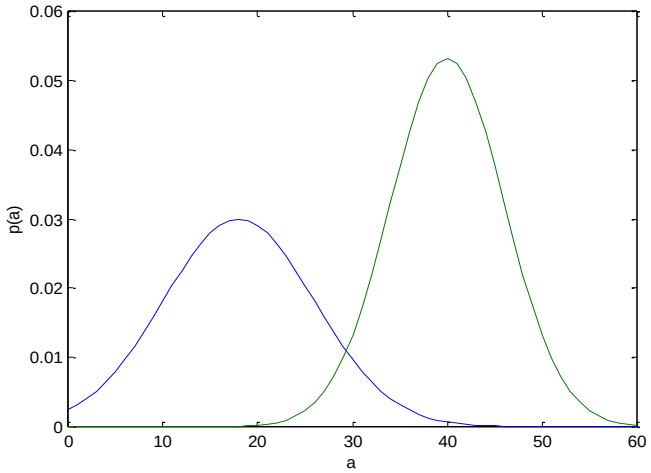


Clustering with cutoff



Agrupamiento probabilístico (EM)

- El agrupamiento con categorías presenta algunos problemas
  - Dependencia del orden
  - Tendencia al sobreajuste
- Esta aproximación evita las divisiones prematuras, asignando parámetros a un modelo de mezcla de distribuciones
- Mezcla de k distribuciones de probabilidad, representando los k clusters.



$\mu 1=18$; $\mu 2=40$;
$\sigma 1=8$; $\sigma 2=6$;
$p1=0.6$
$p2=0.4$

- Se determina la probabilidad de que cada instancia proceda de cada grupo

Agrupamiento

Algoritmo EM

- Se determina el número de grupos a ajustar: k
- En cada grupo, parámetros de las distribuciones de cada atributo

  $p_i, \mu_i, \sigma_i \; i=1...k$

  Ej.: 2 grupos (A, B) y un atributo: 5 parámetros
- En cada iteración hay dos etapas:
- "***Expectation***"**:** se calculan las probabilidades de cada ejemplo en cada uno de los grupos

$$\Pr(A \mid x) = \frac{f(x \mid A) p_A}{f(x)}; \qquad f(x \mid A) = \frac{1}{\sqrt{2\pi}\sigma_A} \exp\left[-\frac{1}{2} \frac{(x-\mu_A)^2}{\sigma_A^2} \right]$$

- "***Maximization***"**:** se calculan los parámetros de las distribuciones que maximicen la verosimilitud de las distribuciones

$$w_i = \Pr(A \mid x_i);$$

$$\mu_A = \frac{w_1 x_1 + ... w_n x_n}{w_1 + ... w_n}; \quad \sigma_A^2 = \frac{w_1 (x_1 - \mu_A) + ... w_n (x_n - \mu_A)^2}{w_1 + ... w_n};$$

Agrupamiento

Algoritmo EM

- **Condición de parada:**
  - se estima la verosimilitud sobre todo el conjunto de instancias:

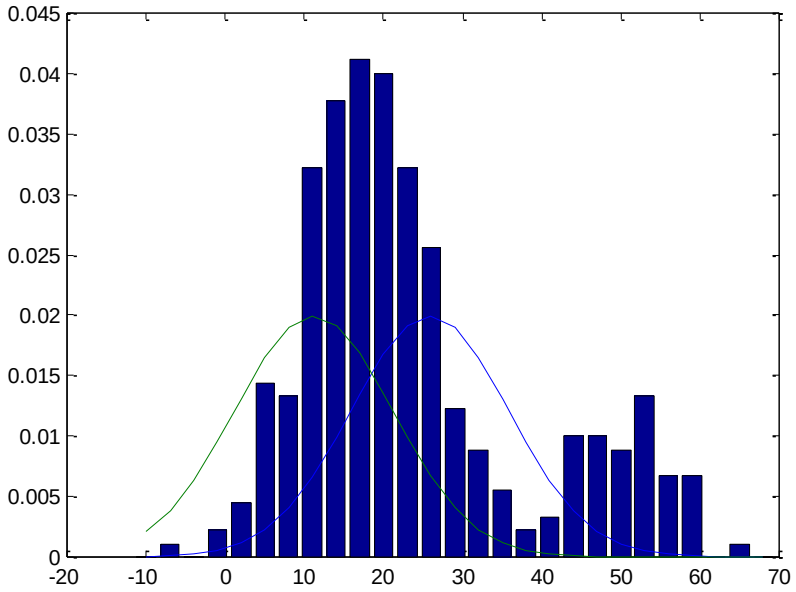
$$\prod_{i=1}^{n}\left[f(x_i \mid A)p_A + f(x_i \mid B)p_B\right]$$

  - se para cuando la mejora en sucesivas iteraciones está debajo de ε

- **Extensión:**
  - Ejemplos con varios atributos y clases
    - Puede asumirse independencia (Naive Bayes) o estimar covarianzas (problema de sobreajuste)
  - Atributos nominales:
    - Se determinan las probabilidades condicionales para cada valor
  - No asegura mínimo global: varias iteraciones
  - Se pueden estimar el número de grupos automáticamente, mediante validación cruzada

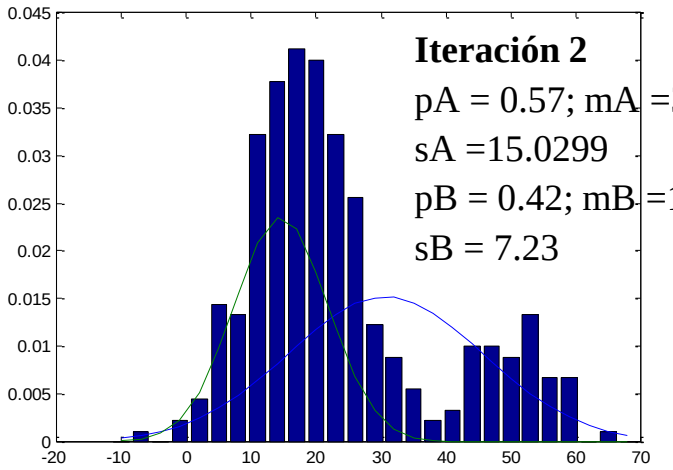
Ejemplo

- 300 datos procedentes de dos distribuciones (no conocidas)
  - pA=0.8; μA=18; σA=8; pB=0.2; μB=50; σB=6;
- Inicialización:
  - pA=pB=0.5; σA=σB=10.0;μA,μB en dos ejemplos al azar

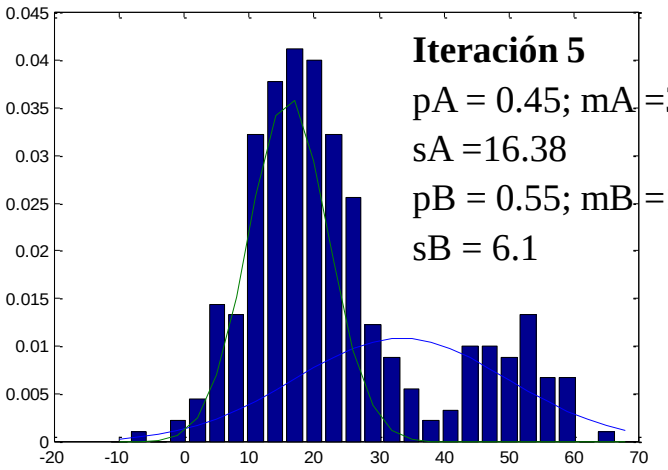


Agrupamiento

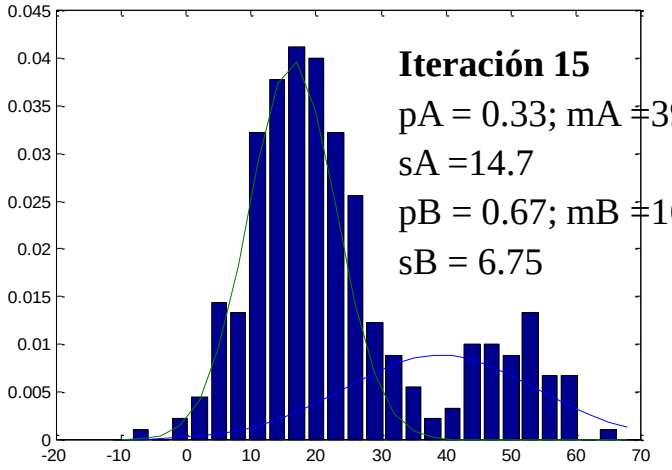
Ejemplo



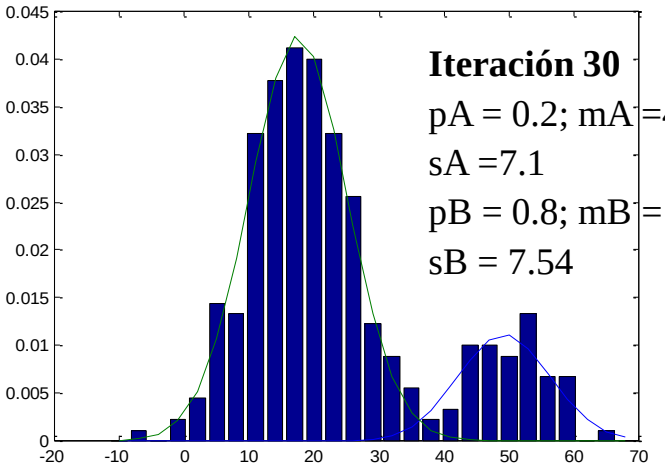
Iteración 2
pA = 0.57; mA =30.68;
sA =15.0299
pB = 0.42; mB =14.61
sB = 7.23



Iteración 5
pA = 0.45; mA =33.42;
sA =16.38
pB = 0.55; mB =16.05
sB = 6.1



Iteración 15
pA = 0.33; mA =39.0;
sA =14.7
pB = 0.67; mB =16.37
sB = 6.75



Iteración 30
pA = 0.2; mA =49.2;
sA =7.1
pB = 0.8; mB =17.54
sB = 7.54

Agrupamiento

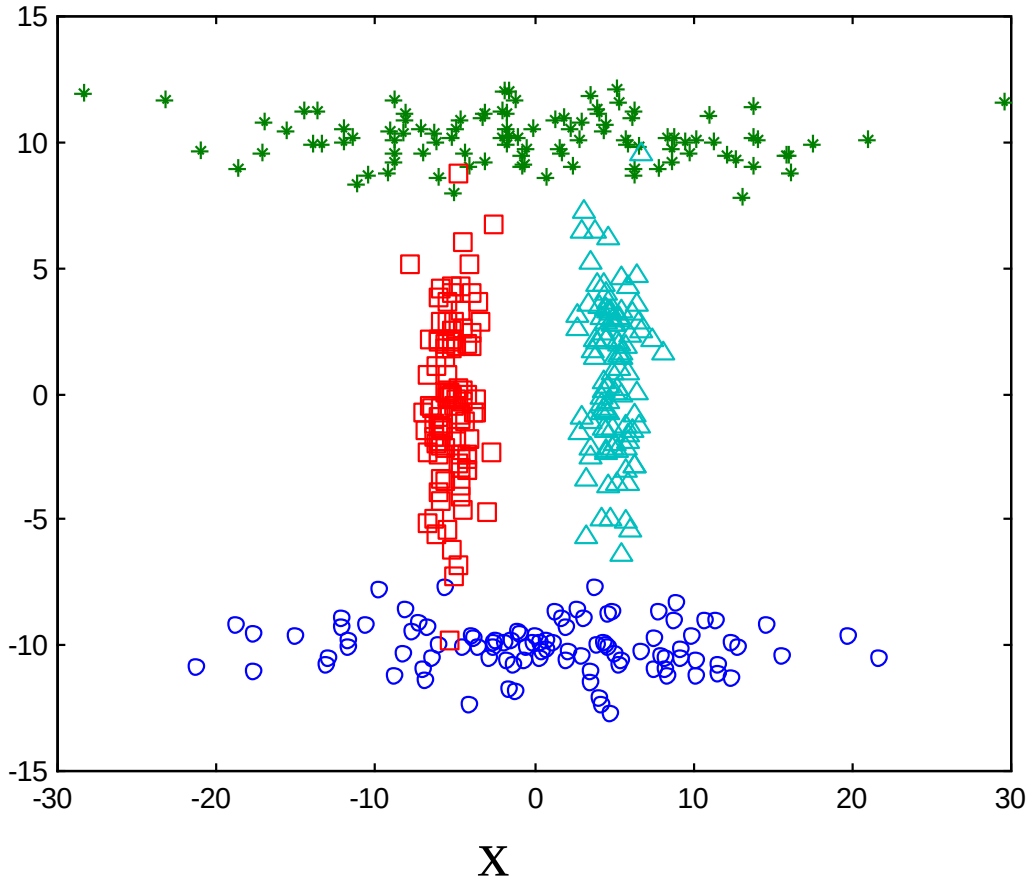
Ejemplo

Agrupamiento con diferentes rangos y varianzas en los atributos:

c_1=[0;-10];c_2=[0;10];c_3=[-5;0];c_4=[5;0]; 100 muestras de cada distribución

$\sigma_{x1}=\sigma_{x2}=10$; $\sigma_{y1}=\sigma_{y2}=1$

$\sigma_{x3}=\sigma_{x4}=1$; $\sigma_{y3}=\sigma_{y4}=3$;



Agrupamiento

Salida k-medias (k=4)

c_1=[0;-10];c_2=[0;10];c_3=[-5;0];c_4=[5;0]; 100 muestras de cada distribución

$\sigma_{x1}=\sigma_{x2}=10$; $\sigma_{y1}=\sigma_{y2}=1$;

$\sigma_{x3}=\sigma_{x4}=1$; $\sigma_{y3}=\sigma_{y4}=3$;

kMeans

======

Cluster centroids:

Cluster 0

0.2959815217391304 5 0.4362347826086957 4

Cluster 1

6.1586733333333346 -10.12866 1

Cluster 2

-0.4669076190476191 9.952375238095238 2

Cluster 3

-7.0927921568627434 -9.091737254901963 1Clustered Instances

0	**184 (46%)**
1	**60 (15%)**
2	**105 (26%)**
3	**51 (13%)**

Agrupamiento

Salida EM

c_1=[0;-10];c_2=[0;10];c_3=[-5;0];c_4=[5;0]; 100 muestras de cada distribución

$\sigma_{x1}=\sigma_{x2}=10$; $\sigma_{y1}=\sigma_{y2}=1$;

$\sigma_{x3}=\sigma_{x4}=1$; $\sigma_{y3}=\sigma_{y4}=3$;

Number of clusters selected by cross validation: 4

Cluster: 0 Prior probability: 0.2487

Attribute: X
Normal Distribution. Mean = 4.9001 StdDev = 1.0596
Attribute: Y
Normal Distribution. Mean = 0.7223 StdDev = 3.0122

Cluster: 1 Prior probability: 0.2505

Attribute: X
Normal Distribution. Mean = 0.6498 StdDev = 8.4748
Attribute: Y
Normal Distribution. Mean = -10.0672 StdDev = 0.9735

Cluster: 2 Prior probability: 0.2514

Attribute: X
Normal Distribution. Mean = -0.2195 StdDev = 10.195
Attribute: Y
Normal Distribution. Mean = 10.1292 StdDev = 0.9554

Cluster: 3 Prior probability: 0.2494

Attribute: X
Normal Distribution. Mean = -5.1877 StdDev = 0.9635
Attribute: Y
Normal Distribution. Mean = -0.2764 StdDev = 3.1796

Agrupamiento

Número de clusters, k?

- Valor k que minimiza distancia a los centros de clusters con validación cruzada
- Utilización de penzliación en la distancia a los datos de entranamiento (criterio MDL)
- Aplicar k-medias recursivamente con k = 2 y usar criterio de parada (eg. based on MDL)
  - Semillas de subclusters en la dirección de mayor varianza en el cluster (un sigma en cada dirección desde el centro del cluster padre)

Aprendizaje semisupervisado

- *Semisupervised learning*: utilizar datos etiquetados y no etiquetados
  - Objetivo: mejorar la clasificación
- Razón: datos no etiquetados son más disponibles y baratos
  - Web mining: classifying web pages
  - Text mining: identifying names in text
  - Video mining: classifying people in the news
- Aprovechar amplia disponibilidad de ejemplos no etiquetados

Clustering y clasificación

- Idea: naïve Bayes sobre ejemplos etiquetados y después EM
  1. Constuir modelo naïve Bayes con datos etiquetados
  2. Etiquetar datos no etiquetados con probabilidaes ("expectation" step)
  3. Entrenar nuevo modelo naïve Bayes con todos los datos ("maximization" step)
  4. Repetir 2 y 3 hasta convergencia
- Como EM fijando los cluster a las clases y comnzando con probabilidades de etiquetados

Clustering y clasificación

- Aplicado con éxito en textos
  - Ciertas frases son indicativas de clases
  - Algunas frases ocrren solo en textos no etiquetados, otras en los dos
- EM puede generalizar tomando la ocurrencia de ambos tipos
  - Variación: reducir el peso de datos no etiquetados
  - Variación: permitir más de un cluster por clase