Clustering

UCR – ECCI

CI-2414 Recuperación de Información

Prof. Kryscia Daviana Ramírez Benavides

Introducción

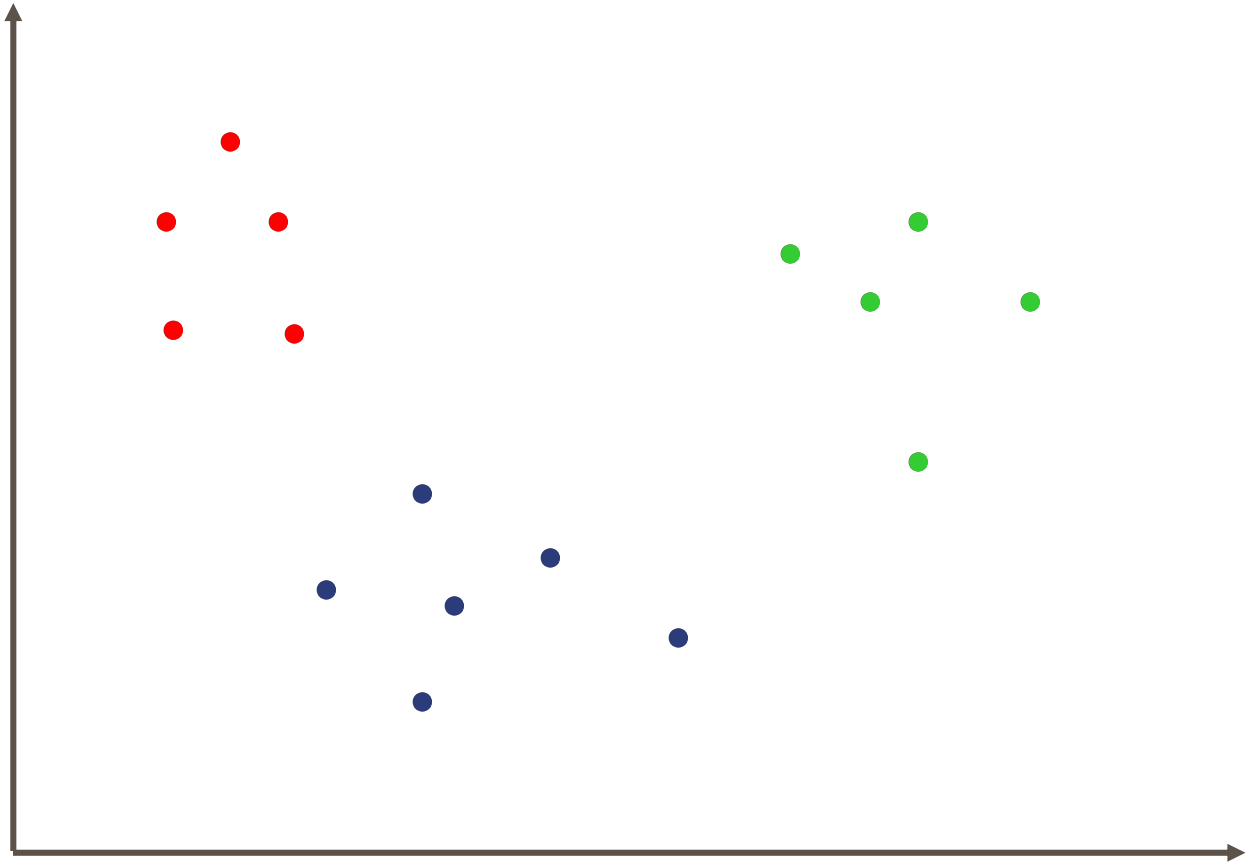
- *Cluster:* Una colección de objetos que son "similares" entre sí.
- *Clustering:* Es la división de los datos en grupos significantes llamados clusters. Ayuda a realizar una agrupación natural o estructural de un conjunto de datos.
  - Es un método alternativo que permite organizar los resultados obtenidos a través de grupos de documentos (*cluster* base) tomando en cuenta algún tópico en especial.
  - Es una técnica para presentar los documentos después de haberlos recuperado en grupos pequeños que están relacionados por un tema en especial.
  - Es una técnica estadística que se usa para generar una estructura de categorías y para agrupar un conjunto de documentos.

Introducción (cont.)

- Divide la colección de documentos en subconjuntos de *clusters* (grupos formados) que cumplen:
  - Los documentos dentro de un *cluster* son similares. Alto grado de asociación entre los miembros de un mismo grupo.
  - Los documentos de *clusters* diferentes son diferentes. Bajo grado de asociación entre miembros de distintos grupos.
- Para realizar clustering se necesita:
  - Ítems de datos (puntos, secuencias).
  - Una función de distancia o similaridad.
  - Un método para evaluar los resultados del *clustering*.

Introducción – Ejemplo (cont.)

Introducción – Ejemplo (cont.)



Introducción (cont.)

- Al *clustering* se le denomina a veces como una asignación automática de clase, esta denominación no es estrictamente precisa:
  - En el *clustering* **no se conocen los grupos antes del proceso** sino que se definen según se asignan los elementos.
  - En la asignación automática de clase **deberán conocerse a priori** las clases.
- Dado que en el *clustering* no es necesario conocer a priori los grupos a formar es muy útil para estructurar grandes conjuntos de datos diversos.
- La Hipótesis de van Rijsbergen postula que "*los documentos estrechamente asociados tienden a ser relevantes para las mismas consultas*".
- Si una colección de documentos está bien agrupada podemos simplemente buscar en los *clusters*, en lugar de buscar en la colección entera.

Introducción (cont.)

- Algunos usos posibles en RI son:
  - Mejorar la precisión o *recall* en un SRI.
  - Organizar y presentar los resultados devueltos por un motor de búsqueda.
  - Procesar la consulta y explorar el corpus.
  - Encontrar documentos similares a un documento dado.
  - Encontrar similaridades entre términos (construir tesauros).
  - Encontrar similaridades entre documentos (clasificación).
  - Escoger temáticas e ignorar otras (filtrado).
- Tiene aplicaciones en muchos campos:
  - Medicina.
  - Taxonomías zoológicas y botánicas.
  - Censos.
  - Imágenes.
  - Entre otros.

Introducción (cont.)

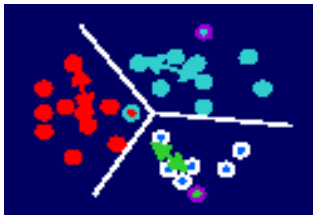
- Problemas por resolver:
  - Decidir las características por las que agrupar los elementos y su representación.
  - Seleccionar el método de agrupación adecuado y, la función de similitud y distancia.
  - Crear los grupos o jerarquías de grupos, lo cual puede ser caro en recursos.
  - Visualizar los resultados, lo cual puede ser caro en recursos.
  - Comprobar la validez de los resultados obtenidos.
  - Seleccionar el método para buscar en la estructura de grupos que se ha formado.
  - Poder nombre a los grupos realizados por el *clustering* (sumarización).

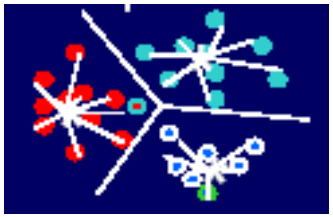
Medidas de Asociación

- A fin de agrupar los documentos de un conjunto de datos se necesita cuantificar de alguna forma el grado de asociación entre ellos:
  - Esto puede hacerse con una medida de distancia o de similitud.
- Algunos métodos de agrupamiento conllevan el uso de una medida específica, pero en general la elección de la medida de asociación queda a la elección del investigador.
- Existen varias medidas de similitud y distancia disponibles, y su elección puede tener efecto sobre los grupos resultantes.
- La determinación de la similitud entre documentos depende de:
  - La representación de los documentos, los pesos asignados a los términos indexados que caracterizan al documento.
  - El coeficiente de similitud que se escoja.

Evaluación del *Clustering*

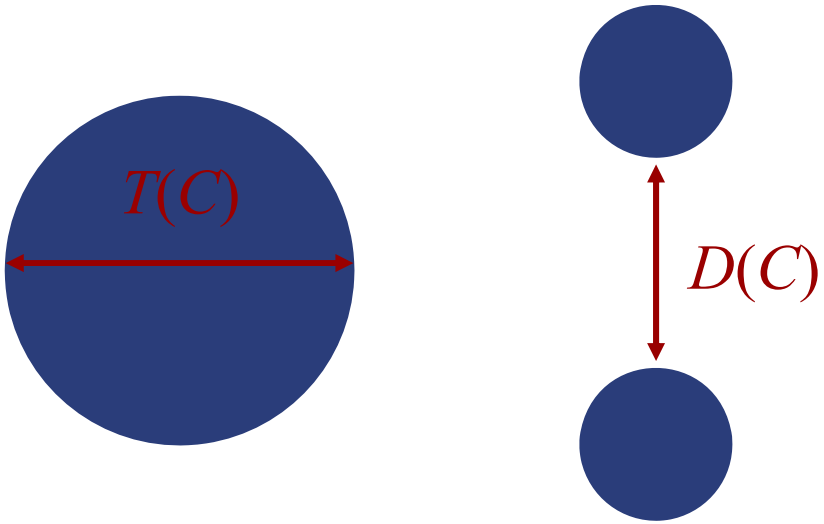
- Un buen método de *clustering* producirá clusters donde:
  - La distancia *intra-cluster* es pequeña.
  - La distancia *inter-cluster* es grande.
- Medidas de calidad de *clustering*:
  - Dado un conjunto S de n puntos de en R^d:
    - *k-Centro:* Buscar k centros tal que el radio máximo de un *cluster* es minimizado.
    - *k-Medio:* Buscar k centros que **Minimizan la Distancia Promedio** de un punto a su centro más cercano.





Evaluación del *Clustering* (cont.)

- La **distancie entre clusters** *D(C)* es la mínima distancia *d(x,y)* para *x* y *y* en *clusters* diferentes.
- La ***estrechez*** *T(C)* de un *clustering* es el máximo diámetro de cualquier *cluster*.
- La **función objetivo** *G(C)* busca **maximizar la distancia entre *clusters* – estrechez**.



Métodos de *Clustering* y Algoritmos Asociados

- Existen muchas formas de agrupar *N* objetos en *M* grupos. El problema aumenta cuando *M* es desconocido a priori.
- La tarea del método de *clustering* será identificar un conjunto de grupos que reflejen alguna estructura subyacente en los datos:
  - Cada método puede producir un agrupamiento diferente y para cada método se podría elegir entre diferentes algoritmos que lleven a cabo los grupos.
  - La elección del método determinará el resultado y la elección del algoritmo determinará la eficacia con que se alcance el resultado.
- Existen dos métodos de *clustering*:
  - No jerárquicos.
  - Jerárquicos.
- También existe el método híbrido, que combinan las dos técnicas anteriores.

Métodos de *Clustering* y Algoritmos Asociados
Métodos de *Clustering* No Jerárquicos

- Los métodos de *clustering* están caracterizados de acuerdo al tipo de estructura de grupos que producen.
- Los métodos simples no-jerárquicos dividen el conjunto de datos de *N* elementos en *M* grupos no solapados:
  - Por ello se les conoce también como **métodos particionantes**.
  - Cada elemento pertenece al grupo que le es más similar.
  - Cada grupo puede representarse por un centroide o representante del grupo, que es representativo de las características de los elementos que contiene.

Métodos de *Clustering* y Algoritmos Asociados
Métodos de *Clustering* No Jerárquicos (cont.)

- Son heurísticos por naturaleza, dado que se requieren decisiones previas sobre:
  - El número de grupos.
  - El tamaño de los grupos.
  - Los criterios de pertenencia a grupos.
  - La forma de representación de los grupos.
- Dado que el alto número de posibles divisiones de N elementos en M grupos hace imposible una solución óptima se intenta encontrar una aproximación particionando el conjunto de datos de alguna forma inicial y, a partir de ahí reubicar los elementos hasta optimizar el criterio empleado.

Métodos de *Clustering* y Algoritmos Asociados
Métodos de *Clustering* No Jerárquicos (cont.)

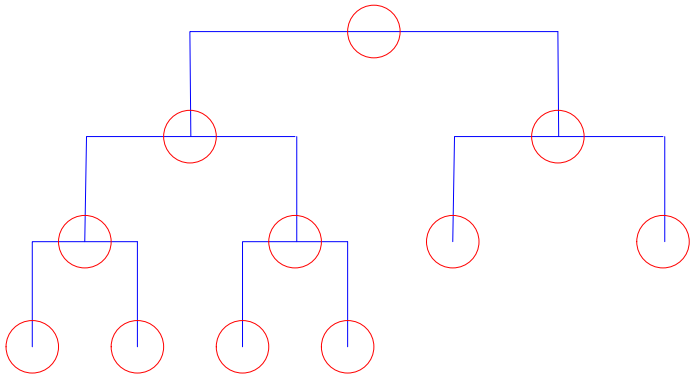
- Algoritmo (Pseudocódigo):
  1. Proveer el número deseado de *cluster*, *k*.
  2. Inicializar la matriz de similitud.
     - Si hay *N* documentos habrá que calcular ***N**(*N*-1)/2** valores del coeficiente de similitud.
  3. Elegir aleatoriamente las *k* instancias como **semillas o centroides**, una por *cluster*.
  4. Formar los *clusters* basados en estos centroides, agregando los elementos más cercanos a este.
  5. Reasignar las *k* instancias (semillas o centroides) de los *clusters*.
  6. Parar cuando el *clustering* converge, después de un número fijo de iteraciones o después de un tiempo dado; sino repetir los pasos 4 y 5.

Métodos de *Clustering* y Algoritmos Asociados
Métodos de *Clustering* Jerárquicos

- Los métodos jerárquicos producen un conjunto de datos anidados en el que los pares de elementos o grupos se van enlazando sucesivamente hasta que todos los elementos quedan conectados.
- Construcción de un árbol basado en una taxonomía jerárquica (dendrograma) de un conjunto de elementos.
- Los métodos jerárquicos pueden ser:
  - *Aglomerativos:* Parten de un conjunto no-agrupado de *N* elementos y realizan *N-1* enlaces de parejas (ascendente) (*bottom-up*).
  - *Divisivos:* Parten de un único grupo de *N* elementos y realizan *N-1* divisiones de grupo en grupos más pequeños (descendente) (*top-down*).
    - Los métodos jerárquicos divisivos son menos usados y disponen de pocos algoritmos para su puesta en servicio
- Se discutirán sólo los métodos aglomerativos.

Métodos de *Clustering* y Algoritmos Asociados
Métodos de *Clustering* Jerárquicos (cont.)

- La estructura de grupos resultante se suele representar como un **dendrograma**:
  - Muestra el orden de emparejamiento de los elementos del conjunto de datos.
  - Muestra el valor o nivel de la función de similitud en el que ocurre cada emparejamiento.
- El *dendrograma* es una representación muy útil para la recuperación en un conjunto de documentos agrupados, ya que indica el camino que debe seguir el proceso de recuperación.



Métodos de *Clustering* y Algoritmos Asociados
Métodos de *Clustering* Jerárquicos (cont.)

- Algoritmo (Pseudocódigo, aglomerativo):
  1. Inicializar la matriz de similitud.
     - Si hay *N* documentos habrá que calcular ***N**(*N*-1)/2** valores del coeficiente de similitud.
  2. Situar a cada uno de los *N* documentos en su propio grupo.
  3. Formar un nuevo grupo combinando la pareja de grupos más similar, *i* y *j*.
  4. Actualizar la matriz de similitud:
     - Eliminar los elementos correspondientes a los antiguos grupos *i* y *j*.
     - Calcular los elementos correspondientes al nuevo grupo *i*+*j*.
  5. Si queda más de 1 grupo entonces repetir los pasos 3 y 4.

Método No-Jerárquico
K-Means

- Asume que las instancias son vectores con valores reales.
- Los clusters están basados en *centroides*, centro de gravedad, o el punto medio del *cluster*. La reasignación de los centroides se hace de la siguiente manera:

$$\mu(C) = \frac{1}{N_C} \sum_{x \in C} x$$

- La asignación de elementos a los *clusters* se basa en la distancia a los centroides actuales del *cluster*:

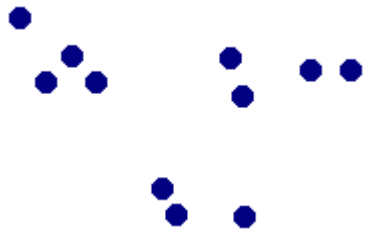
$$d(N, C_j) = \min_{x_i \in N \wedge s_j \in C_j} d(x_i, s_j)$$

Método No-Jerárquico
K-Means (cont.)

- Algoritmo (Pseudocódigo):
  1. Calcular la matriz de similaridad M tal que $M[i, j] = sim(i, j)$ es la similaridad entre el elemento i y el elemento j
  2. Seleccionar k instancias como centroides iniciales de alguna manera.
  3. Asignar cada elemento al centroide que se encuentra a una distancia mínima.
  4. Mover cada centroide hacia el centro del grupo que se le ha asignado.
  5. Repetir los pasos 3 y 4 hasta que ningún centroide cambie, después de un número fijo de iteraciones o después de un tiempo dado.

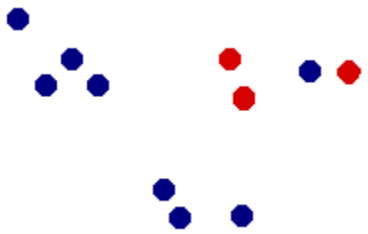
Método No-Jerárquico
K-Means (cont.)

- Sea *d* la medida de la distancia entre las instancias.
- Se selecciona *k* instancias aleatorias $\{s_1, s_2, \ldots s_k\}$ como semillas (centroides iniciales).
- Se repite hasta que el clustering converja:
  - Para cada instancia x_i:
    - Asignar x_i para el *cluster* C_j cuya $d(x_i, s_j)$ es mínima. (*Modifica las semillas para el centroide de cada cluster*).
  - Para cada *cluster* C_j:
    - $s_j = \mu(C_j)$.



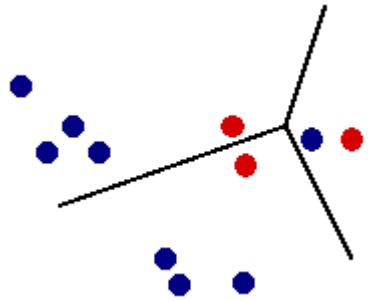
Método No-Jerárquico
K-Means (cont.)

- Sea *d* la medida de la distancia entre las instancias.
- Se selecciona *k* instancias aleatorias $\{s_1, s_2, \ldots s_k\}$ como semillas (centroides iniciales).
- Se repite hasta que el clustering converja:
  - Para cada instancia x_i:
    - Asignar x_i para el *cluster* C_j cuya $d(x_i, s_j)$ es mínima. (*Modifica las semillas para el centroide de cada cluster*).
  - Para cada *cluster* C_j:
    - $s_j = \mu(C_j)$.



Método No-Jerárquico
K-Means (cont.)

- Sea *d* la medida de la distancia entre las instancias.
- Se selecciona *k* instancias aleatorias $\{s_1, s_2, \ldots s_k\}$ como semillas (centroides iniciales).
- Se repite hasta que el clustering converja:
  - Para cada instancia x_i:
    - Asignar x_i para el *cluster* C_j cuya $d(x_i, s_j)$ es mínima. (*Modifica las semillas para el centroide de cada cluster*).
  - Para cada *cluster* C_j:
    - $s_j = \mu(C_j)$.



Método No-Jerárquico
Ejemplo *K-Means*

- Se tiene los siguientes documentos: A, B, C, D, E y F.
- Se utiliza la función coseno como distancia y $k = 2$.
- Los vectores de los documentos y la matriz de similaridad (función coseno) son los siguientes:

$$\begin{array}{l} \\ A = (0.00 \quad 0.09 \quad 0.00 \quad 0.05) \\ B = (0.08 \quad 0.00 \quad 0.00 \quad 0.00) \\ C = (0.00 \quad 0.12 \quad 0.00 \quad 0.00) \\ D = (0.12 \quad 0.00 \quad 0.00 \quad 0.30) \\ E = (0.00 \quad 0.04 \quad 0.00 \quad 0.30) \\ F = (0.30 \quad 0.09 \quad 0.00 \quad 0.00) \end{array} \qquad \begin{array}{c} \begin{array}{cccccc} A & B & C & D & E & F \end{array} \\ \begin{bmatrix} 1 & 0 & 0.9 & 0.5 & 0.6 & 0.2 \\ 0 & 1 & 0 & 0.4 & 0 & 1 \\ 0.9 & 0 & 1 & 0 & 0.1 & 0.3 \\ 0.5 & 0.4 & 0 & 1 & 0.9 & 0.4 \\ 0.6 & 0 & 0.1 & 0.9 & 1 & 0 \\ 0.2 & 1 & 0.3 & 0.4 & 0 & 1 \end{bmatrix} \end{array} \begin{array}{l} A \\ B \\ C \\ D \\ E \\ F \end{array}$$

Método No-Jerárquico
Ejemplo *K-Means* (cont.)

- Al asignar cada elemento al centroide se debe encontrar una distancia mínima, y al mover cada centroide hacia el centro del grupo que se le ha asignado, las fórmulas ha usar son:

$$d(N, C_j) = \min_{\vec{x_i} \in N \wedge \vec{s_j} \in C_j} 1 - \left(\frac{\vec{d_i} \bullet \vec{s_j}}{\left\| \vec{d_i} \right\| \times \left\| \vec{s_j} \right\|} \right) \qquad \mu(C) = \frac{1}{N_C} \sum_{\vec{x} \in C} \vec{x}$$

- La matriz de distancia es:

	A	B	C	D	E	F
A	0	1	0.1	0.5	0.4	0.8
B	1	0	1	0.6	1	0
C	0.1	1	0	1	0.9	0.7
D	0.5	0.6	1	0	0.1	0.6
E	0.4	1	0.9	0.1	0	1
F	0.8	0	0.7	0.6	1	0

Método No-Jerárquico
Ejemplo *K-Means* (cont.)

	t_1	t_2	t_3	t_4
A	0.00	0.09	0.00	0.05
B	0.08	0.00	0.00	0.00
C	0.00	0.12	0.00	0.00
D	0.12	0.00	0.00	0.30
E	0.00	0.04	0.00	0.30
F	0.30	0.09	0.00	0.00

	A	B	C	D	E	F
A	0	1	0.1	0.5	0.4	0.8
B	1	0	1	0.6	1	0
C	0.1	1	0	1	0.9	0.7
D	0.5	0.6	1	0	0.1	0.6
E	0.4	1	0.9	0.1	0	1
F	0.8	0	0.7	0.6	1	0

- Se escoge aleatoriamente dos puntos ($k = 2$), los cuales son A y D.
- Asignar cada elemento al centroide que se encuentra a una distancia mínima:
  - $d(B,A) = 1$ y $d(B,D) = 0.6$, por lo tanto B hace grupo con D.
  - $d(C,A) = 0.1$ y $d(C,D) = 1$, por lo tanto C hace grupo con A.
  - $d(E,A) = 0.4$ y $d(E,D) = 0.1$, por lo tanto E hace grupo con D.
  - $d(F,A) = 0.8$ y $d(F,D) = 0.6$, por lo tanto F hace grupo con D.
- Mover cada centroide hacia el centro del grupo que se le ha asignado.

$$\mu(C_1) = \frac{(0.00 \quad 0.09 \quad 0.00 \quad 0.05) + (0.00 \quad 0.12 \quad 0.00 \quad 0.00)}{2}$$

$$\mu(C_1) = \frac{(0.00 \quad 0.21 \quad 0.00 \quad 0.05)}{2} = (0.00 \quad 0.11 \quad 0.00 \quad 0.03) \Rightarrow C$$

$$\mu(C_2) = \frac{(0.08 \quad 0.00 \quad 0.00 \quad 0.00) + (0.12 \quad 0.00 \quad 0.00 \quad 0.30) + (0.00 \quad 0.04 \quad 0.00 \quad 0.30) + (0.30 \quad 0.09 \quad 0.00 \quad 0.00)}{4}$$

$$\mu(C_2) = \frac{(0.5 \quad 0.13 \quad 0.00 \quad 0.6)}{4} = (0.13 \quad 0.03 \quad 0.00 \quad 0.15) \Rightarrow D$$

Método No-Jerárquico
Ejemplo *K-Means* (cont.)

	t_1	t_2	t_3	t_4
A	0.00	0.09	0.00	0.05
B	0.08	0.00	0.00	0.00
C	0.00	0.12	0.00	0.00
D	0.12	0.00	0.00	0.30
E	0.00	0.04	0.00	0.30
F	0.30	0.09	0.00	0.00

	A	B	C	D	E	F
A	0	1	0.1	0.5	0.4	0.8
B	1	0	1	0.6	1	0
C	0.1	1	0	1	0.9	0.7
D	0.5	0.6	1	0	0.1	0.6
E	0.4	1	0.9	0.1	0	1
F	0.8	0	0.7	0.6	1	0

- Asignar cada elemento al centroide que se encuentra a una distancia mínima:
  - $d(A,C) = 0.1$ y $d(A,D) = 0.5$, por lo tanto A hace grupo con C.
  - $d(B,C) = 1$ y $d(B,D) = 0.6$, por lo tanto B hace grupo con D.
  - $d(E,C) = 0.9$ y $d(E,D) = 0.1$, por lo tanto E hace grupo con D.
  - $d(F,C) = 0.7$ y $d(F,D) = 0.6$, por lo tanto F hace grupo con D.
- Mover cada centroide hacia el centro del grupo que se le ha asignado.

$$\mu(C_1) = \frac{(0.00 \quad 0.09 \quad 0.00 \quad 0.05) + (0.00 \quad 0.12 \quad 0.00 \quad 0.00)}{2}$$

$$\mu(C_1) = \frac{(0.00 \quad 0.21 \quad 0.00 \quad 0.05)}{2} = (0.00 \quad 0.11 \quad 0.00 \quad 0.03) \Rightarrow C$$

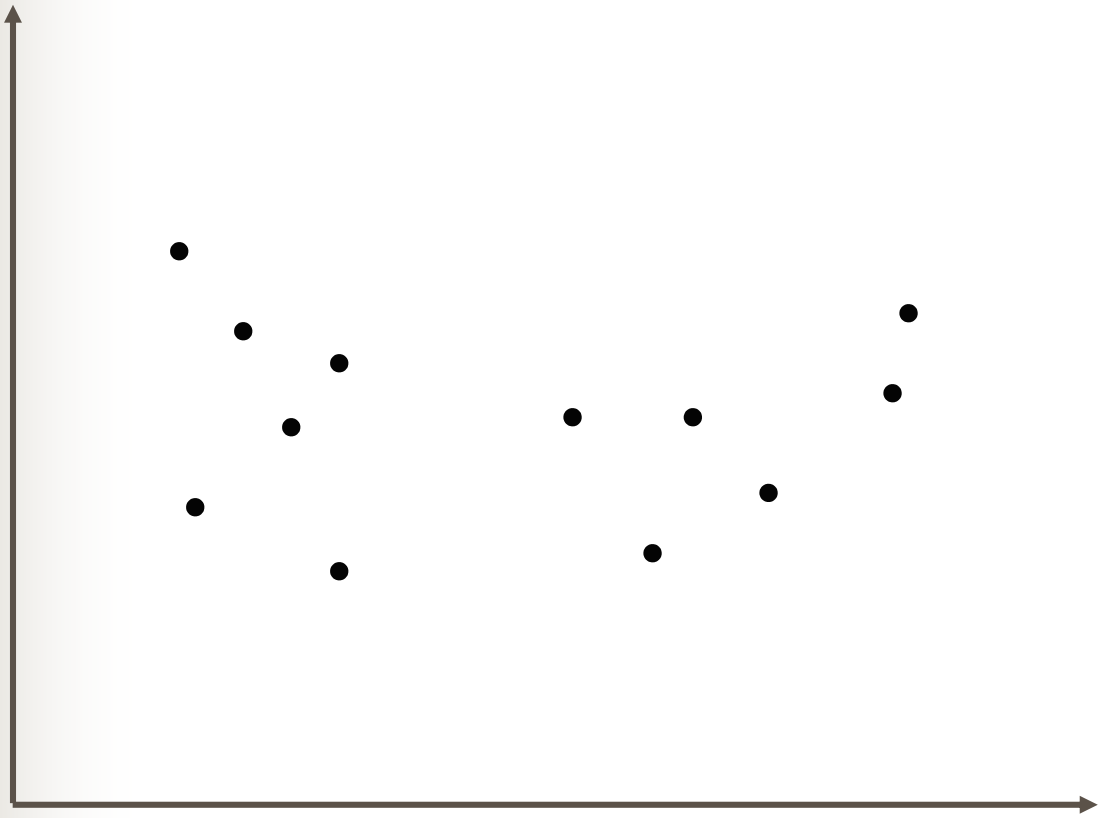
$$\mu(C_2) = \frac{(0.08 \quad 0.00 \quad 0.00 \quad 0.00) + (0.12 \quad 0.00 \quad 0.00 \quad 0.30) + (0.00 \quad 0.04 \quad 0.00 \quad 0.30) + (0.30 \quad 0.09 \quad 0.00 \quad 0.00)}{4}$$

$$\mu(C_2) = \frac{(0.5 \quad 0.13 \quad 0.00 \quad 0.6)}{4} = (0.13 \quad 0.03 \quad 0.00 \quad 0.15) \Rightarrow D$$

Método No-Jerárquico
Ejemplo *K-Means* (cont.)

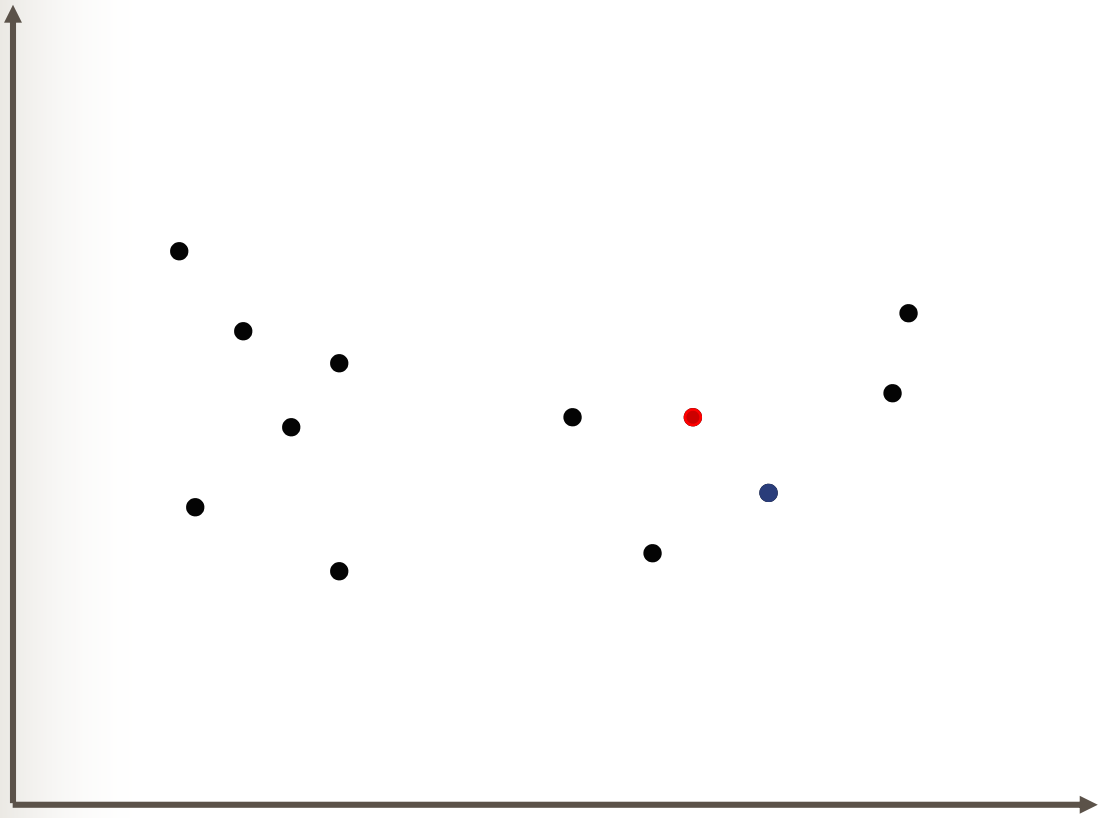
- En las dos últimas iteraciones los centroides no cambiaron, entonces el algoritmo converge y se detiene, por lo que los grupos resultantes son:
  - *A* y *C* – cohesión = (0.00, 0.11, 0.00, 0.03).
  - *B*, *D*, *E* y *F* – cohesión = (0.13, 0.03, 0.00, 0.15).

Método No-Jerárquico
K-Means (k=2) (cont.)

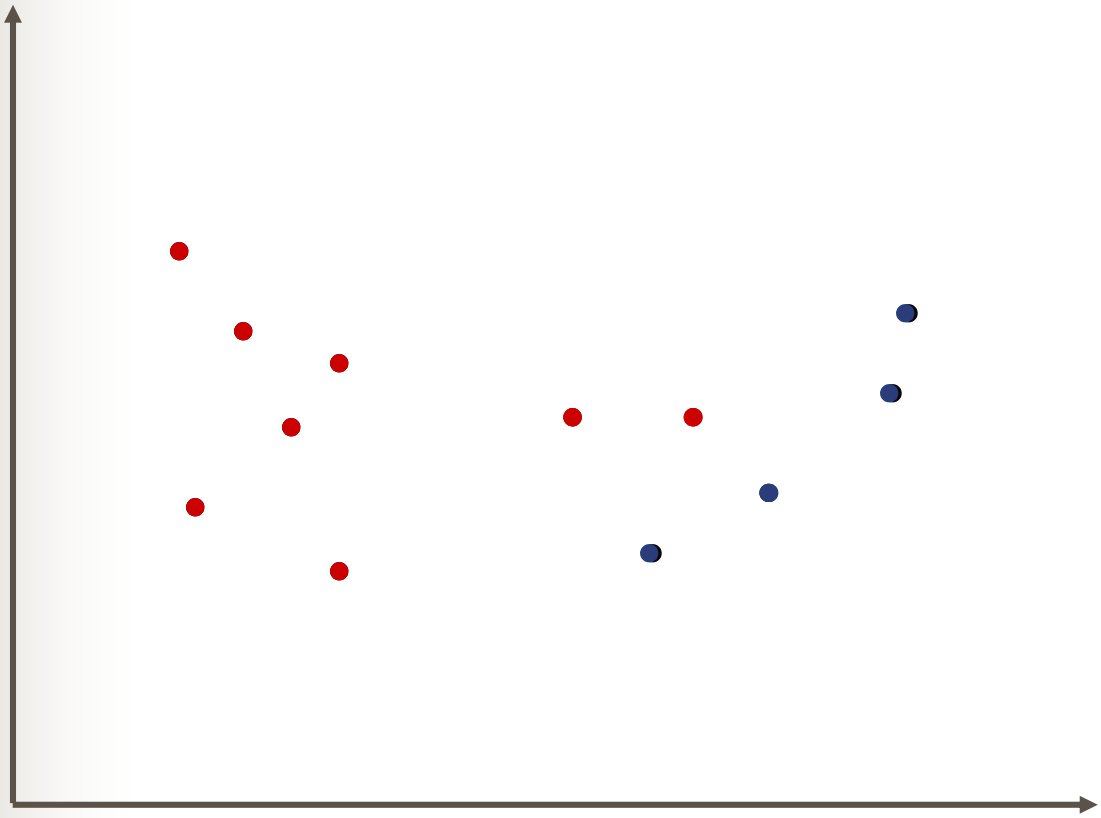


Método No-Jerárquico
K-Means (k=2) (cont.)

Escoger semillas



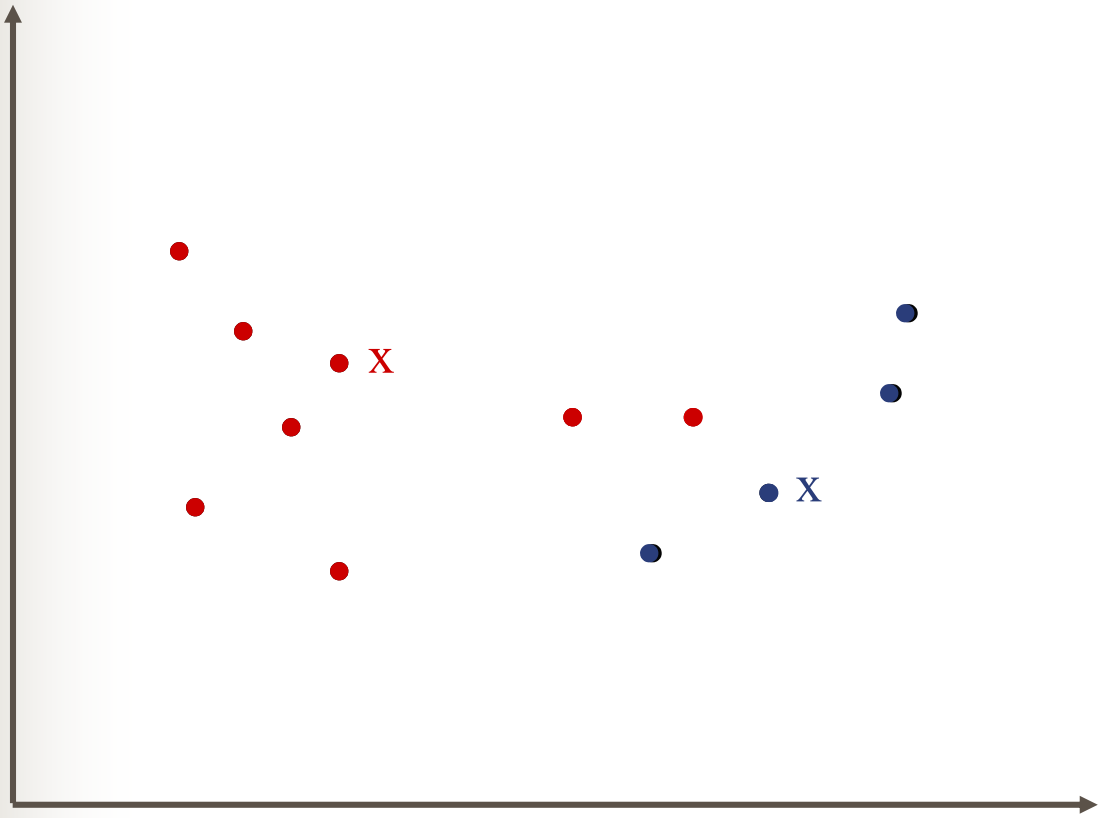
Método No-Jerárquico
K-Means (k=2) (cont.)



Escoger semillas

Reasignar *clusters*

Método No-Jerárquico
K-Means (k=2) (cont.)

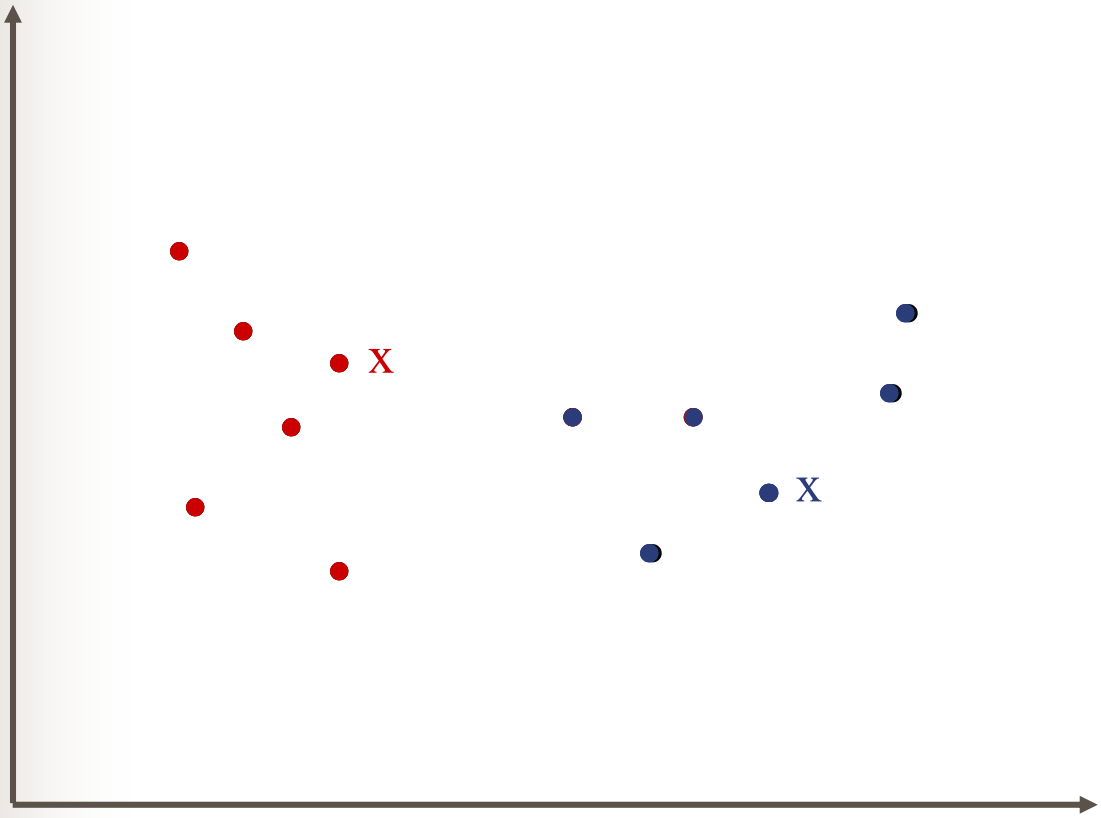


Escoger semillas

Reasignar *clusters*

Computar centroides

Método No-Jerárquico
K-Means (k=2) (cont.)



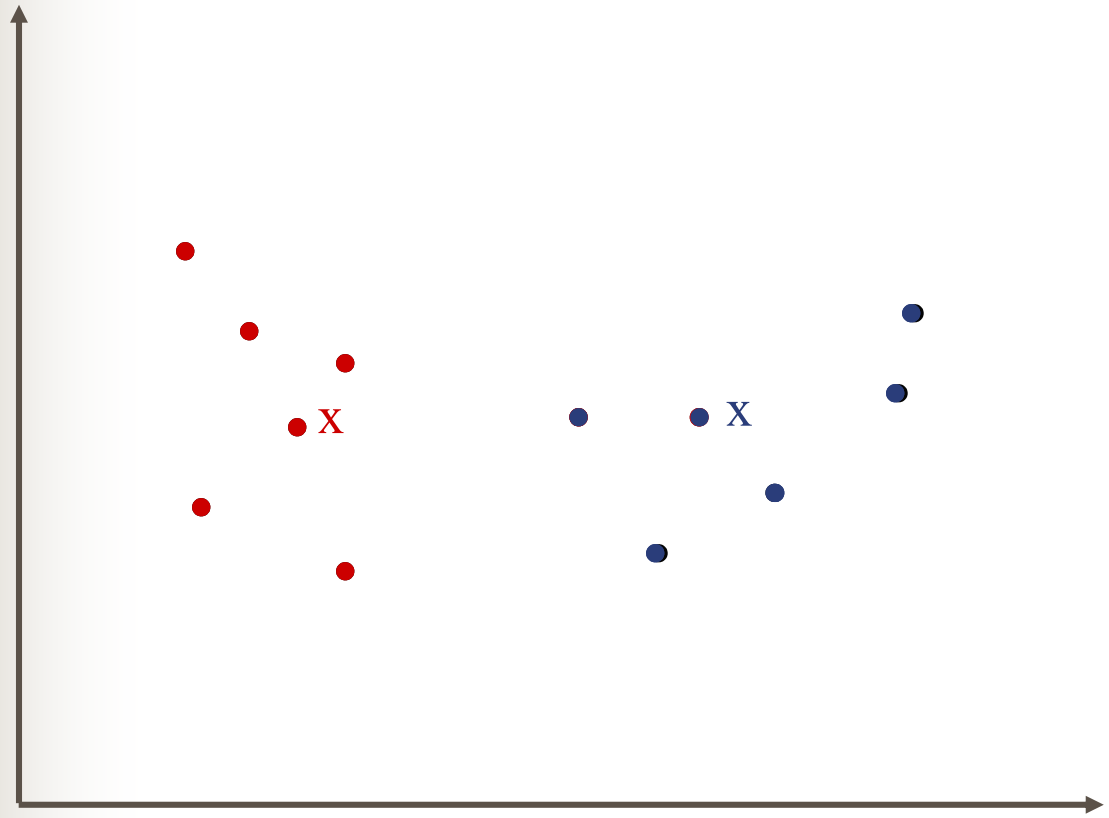
Escoger semillas

Reasignar *clusters*

Computar centroides

Reasignar *clusters*

Método No-Jerárquico
K-Means (k=2) (cont.)



Escoger semillas

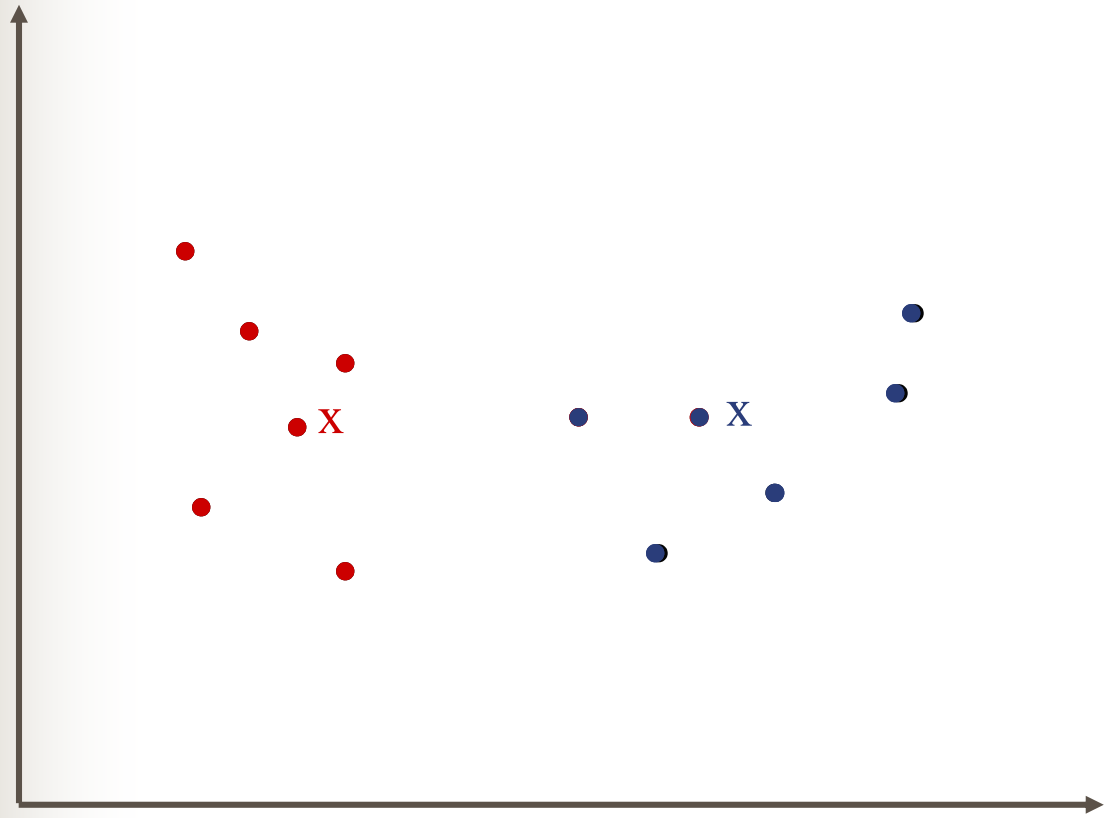
Reasignar *clusters*

Computar centroides

Reasignar *clusters*

Computar centroides

Método No-Jerárquico
K-Means (k=2) (cont.)



Escoger semillas

Reasignar *clusters*

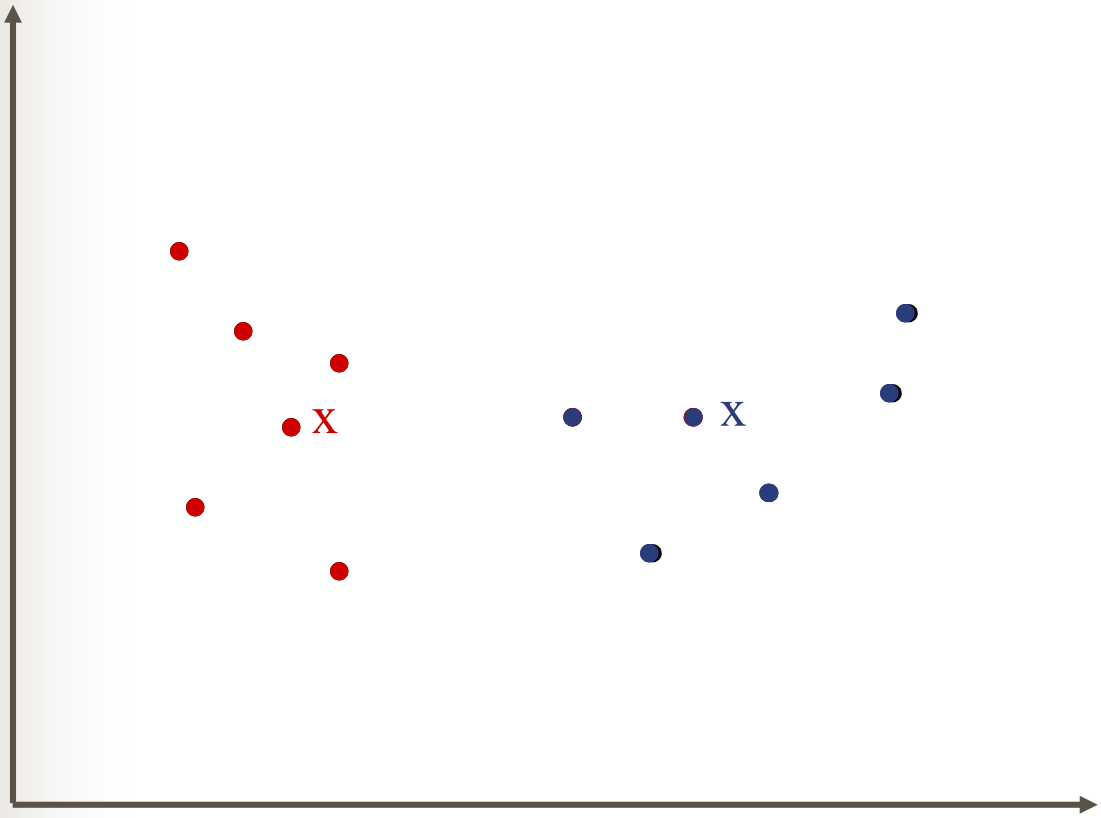
Computar centroides

Reasignar *clusters*

Computar centroides

Reasignar *clusters*

Método No-Jerárquico
K-Means (k=2) (cont.)



Escoger semillas

Reasignar *clusters*

Computar centroides

Reasignar *clusters*

Computar centroides

Reasignar *clusters*

Converge!

Método No-Jerárquico
K-Means – Complejidad Computacional (cont.)

- El tiempo para calcular la distancia entre dos instancias es O(*m*), donde *m* es la dimensionalidad de los vectores.
- La reasignación de *clusters* tiene un tiempo de O(*kn*), la distancia entre *clusters* O(*knm*).
- El computar centroides tiene un tiempo de O(*nm*), o sea, cada instancia del vector debe agregarse una vez a algún centroide.
- Se debe asumir estos dos pasos, cada uno, están hechos una vez para las iteraciones de I, O(*Iknm*).

Método No-Jerárquico
K-Means (cont.)

- Ventajas:
  - Asintóticamente es más rápido que HAC.
  - Puede alcanzar complejidad lineal tanto en tiempo como en espacio.
- Desventajas:
  - En algunas implementaciones, un centro puede sufrir de *inanición*.
  - No se conoce a priori el valor óptimo de *k*.
  - Puede no converger (problema del vector pegado).
  - No es determinista, con la misma colección y el mismo valor de *k* da resultados diferentes para diferentes ejecuciones.

Método No-Jerárquico
K-Means (cont.)

- Problemas:
  - Necesidad de saber *k* por adelantado.
    - La calidad del *clustering* depende de los *k* puntos iniciales.
    - Aunque se podría probar varios valores para *k*.
    - La mejor estrechez *intra-cluster* ocurre cuando *k=n* (cada punto en su propio *cluster*).
  - El algoritmo tiende ir a los mínimos locales que son sensibles a los centroides con los que comienza.
    - Un área local densa puede atrapar un centroide.
  - Disjunto y exhaustivo.
  - Asume que los clusters son esféricos en el espacio vectorial, sensible a los cambios coordinados, el pesado, etc.

Método Jerárquico
Hierarchical Agglomerative Clustering (HAC)

- Asume una función de similaridad para determinar la similaridad de dos instancias.
- Se utiliza una función de similaridad que determina la similaridad de dos instancias: *sim(x,y)*. Por ejemplo: la función coseno.
- Tipos de HAC:
  - *Enlace Simple:* Similaridad de dos miembros más similares.
  - *Enlace Completo:* Similaridad de dos miembros menos similares.
  - *Promedio de Grupos:* Promedio de similaridad entre miembros.
  - *Ward's Method:* Análisis del acercamiento de la variación para evaluar las distancias entre los *clusters*.

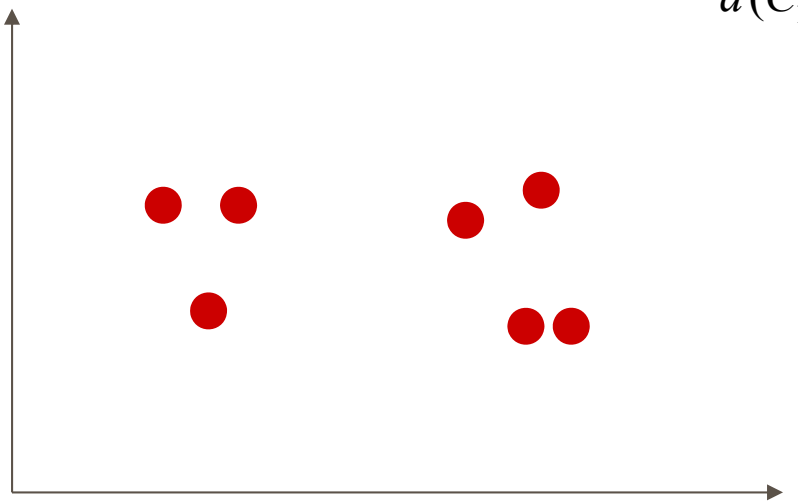
Método Jerárquico
Hierarchical Agglomerative Clustering (HAC) (cont.)

- Algoritmo (Pseudocódigo):
  1. Calcular una matriz de similaridad M tal que $M[i, j] = sim(i, j)$ es la similaridad entre el grupo i y el grupo j.
  2. Inicialmente, colocar cada documento en un grupo propio.
  3. Unir los dos grupos más similares en uno solo.
  4. Actualizar M para reflejar el cambio producido por la unión de grupos.
  5. Repetir los pasos 3 y 4 hasta que haya un único grupo.

Método Jerárquico
Hierarchical Agglomerative Clustering (HAC) (cont.)

- Inicialmente, todos los puntos están en un cluster propio.
- Se repite hasta que queda 1 grupo:
  - Se agrupa los dos *clusters* más cercanos de acuerdo a la similaridad.
  - Se actualiza *M* para reflejar el cambio producido por la agrupación de *clusters*.

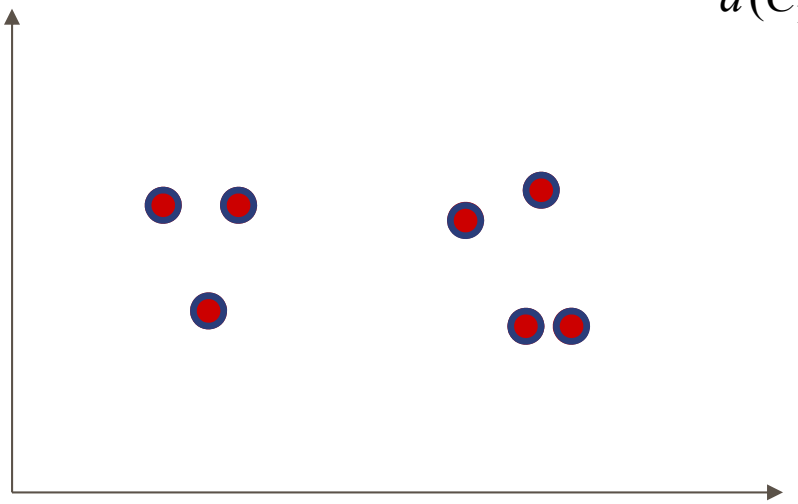
$$d(C_1, C_2) = \min_{x \in C_1, y \in C_2} d(x, y)$$



Método Jerárquico
Hierarchical Agglomerative Clustering (HAC) (cont.)

- Inicialmente, todos los puntos están en un cluster propio.
- Se repite hasta que queda 1 grupo:
  - Se agrupa los dos *clusters* más cercanos de acuerdo a la similaridad.
  - Se actualiza *M* para reflejar el cambio producido por la agrupación de *clusters*.

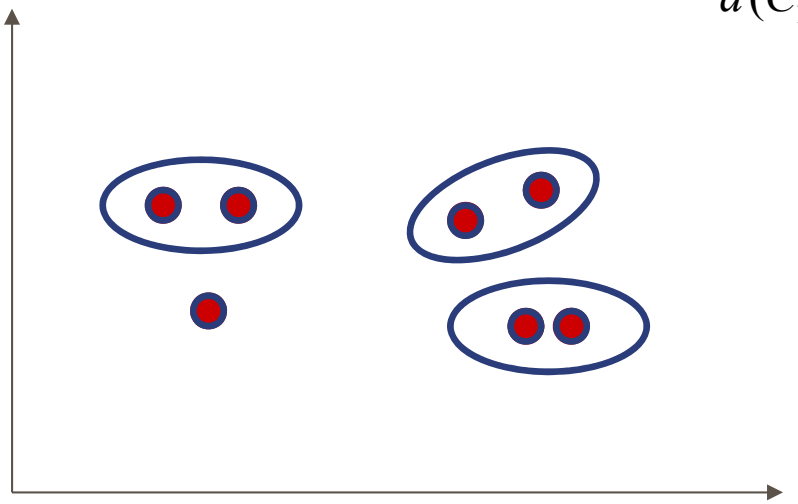
$$d(C_1, C_2) = \min_{x \in C_1, y \in C_2} d(x, y)$$



Método Jerárquico
Hierarchical Agglomerative Clustering (HAC) (cont.)

- Inicialmente, todos los puntos están en un cluster propio.
- Se repite hasta que queda 1 grupo:
  - Se agrupa los dos *clusters* más cercanos de acuerdo a la similaridad.
  - Se actualiza *M* para reflejar el cambio producido por la agrupación de *clusters*.

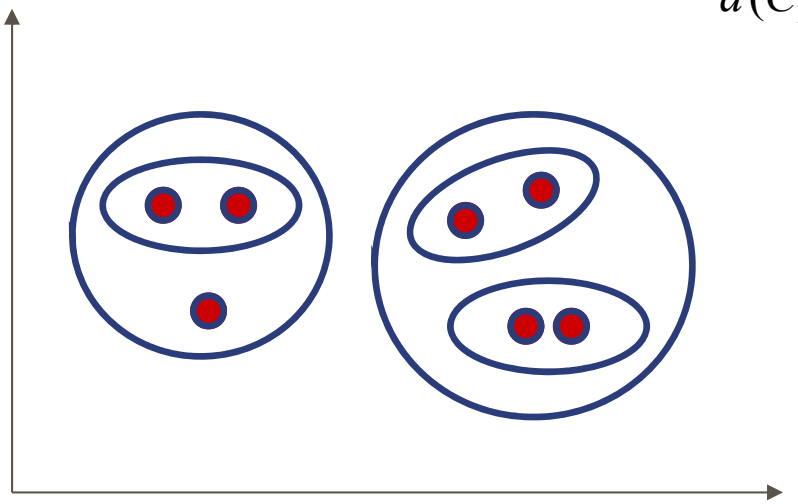
$$d(C_1, C_2) = \min_{x \in C_1, y \in C_2} d(x, y)$$



Método Jerárquico
Hierarchical Agglomerative Clustering (HAC) (cont.)

- Inicialmente, todos los puntos están en un cluster propio.
- Se repite hasta que queda 1 grupo:
  - Se agrupa los dos *clusters* más cercanos de acuerdo a la similaridad.
  - Se actualiza *M* para reflejar el cambio producido por la agrupación de *clusters*.

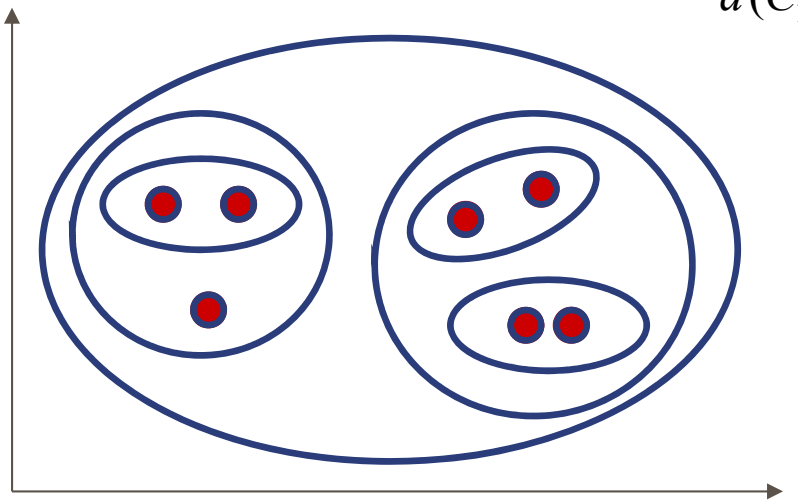
$$d(C_1, C_2) = \min_{x \in C_1, y \in C_2} d(x, y)$$



Método Jerárquico
Hierarchical Agglomerative Clustering (HAC) (cont.)

- Inicialmente, todos los puntos están en un cluster propio.
- Se repite hasta que queda 1 grupo:
  - Se agrupa los dos *clusters* más cercanos de acuerdo a la similaridad.
  - Se actualiza *M* para reflejar el cambio producido por la agrupación de *clusters*.

$$d(C_1, C_2) = \min_{x \in C_1, y \in C_2} d(x, y)$$



Método Jerárquico
Hierarchical Agglomerative Clustering (HAC) (cont.)

- *Enlace simple:* La similitud entre grupos es la similitud entre la pareja de elementos más similar.
  - De esta forma cada miembro de un grupo será, al menos, más similar a un elemento de su mismo grupo que a cualquier miembro de otro grupo.

$$sim(C_i, C_j) = \max_{x \in C_i, y \in C_j} sim(x, y)$$

- *Enlace completo:* La similitud entre grupos es la similitud entre la pareja de elementos menos similar.
  - De esta forma cada miembro de un grupo será más similar al elemento menos similar de su mismo grupo que al elemento menos similar de cualquier otro grupo.

$$sim(C_i, C_j) = \min_{x \in C_i, y \in C_j} sim(x, y)$$

Método Jerárquico
Hierarchical Agglomerative Clustering (HAC) (cont.)

- *Enlace de promedio de grupo:* Es un compromiso entre los extremos que representan el sistema de enlace simple y enlace completo. Cada miembro de un grupo tiene una similitud promedio con los restantes miembros de su grupo y es mayor que la similitud promedio con los miembros de cualquier otro grupo.

$$sim(C_i, C_j) = \frac{1}{N_{C_i} * N_{C_j}} * \sum_{x \in C_i} \sum_{y \in C_j} sim(x, y)$$

Donde:

- $sim(x,y)$: Similaridad del elemento x del *cluster* C_i con el elemento y del *cluster* C_j.
- N_{Ci}: Número de elementos del *cluster* C_i.
- N_{Cj}: Número de elementos del *cluster* C_j.

Método Jerárquico
Hierarchical Agglomerative Clustering (HAC) (cont.)

- *Método Ward's:* Es distinto al resto de métodos porque utiliza un análisis del acercamiento de la variación para evaluar las distancias entre los *clusters*. En resumen, este método procura reducir al mínimo la suma de los cuadrados (SS) de cualquier dos *clusters* (hipotéticos) que se pueda formar en cada paso. En general:
  - Es muy eficiente.
  - Tiende a crear *clusters* del tamaño pequeño.
  - Tiende a producir *clusters* homogéneos y una jerarquía simétrica.

  El algoritmo comienza con un *cluster* grande que abarca todos los elementos que se agruparán. En este caso, la suma del error de cuadrados es 0. El programa busca los elementos que pueden ser agrupados juntos mientras que reducen al mínimo el aumento en la suma del error de cuadrados. La suma del error de cuadrados se computa como:

$$SS_e = x_i^2 - \frac{1}{N}\left(\sum_{i=1}^{N} x_i\right)^2$$

Hierarchical Agglomerative Clustering (HAC)
Ejemplo Enlace Simple y Enlace Completo

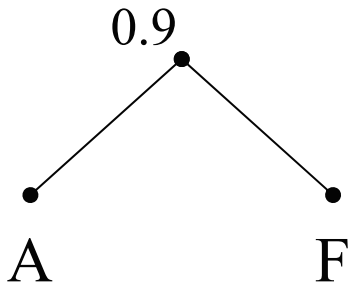
- Se tiene los siguientes documentos: A, B, C, D, E y F.
- La siguiente tabla tiene la similaridad ordenada de forma decreciente:
- La matriz de similaridad entre ellos es:

$$\begin{array}{c} \\ A \\ B \\ C \\ D \\ E \\ F \end{array} \begin{array}{c} \begin{array}{cccccc} A & B & C & D & E & F \end{array} \\ \begin{bmatrix} 1 & 0.3 & 0.5 & 0.6 & 0.8 & 0.9 \\ 0.3 & 1 & 0.4 & 0.5 & 0.7 & 0.8 \\ 0.5 & 0.4 & 1 & 0.3 & 0.5 & 0.2 \\ 0.6 & 0.5 & 0.3 & 1 & 0.4 & 0.1 \\ 0.8 & 0.7 & 0.5 & 0.4 & 1 & 0.3 \\ 0.9 & 0.8 & 0.2 & 0.1 & 0.3 & 1 \end{bmatrix} \end{array}$$

Pareja	Similitud
AF	0,9
AE	0,8
BF	0,8
BE	0,7
AD	0,6
AC	0,5
BD	0,5
CE	0,5
BC	0,4
DE	0,4
AB	0,3
CD	0,3
EF	0,3
CF	0,2
DF	0,1

Hierarchical Agglomerative Clustering (HAC)
Ejemplo Enlace Simple (cont.)

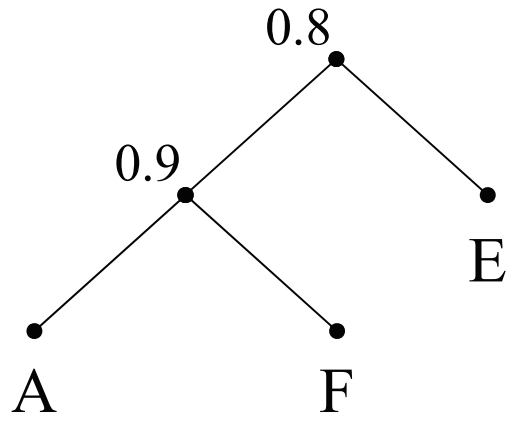
	A	*B*	*C*	*D*	*E*	*F*
A	1	0.3	0.5	0.6	0.8	0.9
B	0.3	1	0.4	0.5	0.7	0.8
C	0.5	0.4	1	0.3	0.5	0.2
D	0.6	0.5	0.3	1	0.4	0.1
E	0.8	0.7	0.5	0.4	1	0.3
F	0.9	0.8	0.2	0.1	0.3	1



Pareja	Similitud
AF	0,9
AE	0,8
BF	0,8
BE	0,7
AD	0,6
AC	0,5
BD	0,5
CE	0,5
BC	0,4
DE	0,4
AB	0,3
CD	0,3
EF	0,3
CF	0,2
DF	0,1

Hierarchical Agglomerative Clustering (HAC)
Ejemplo Enlace Simple (cont.)

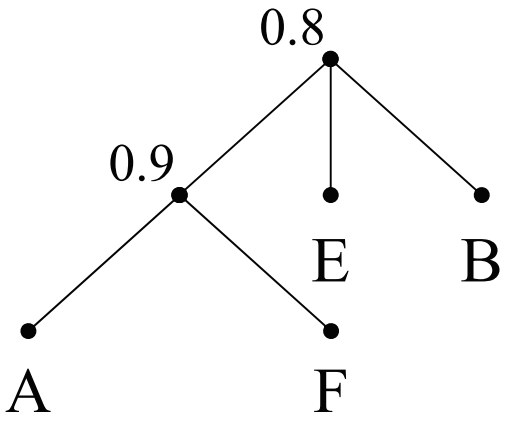
	AF	*B*	*C*	*D*	*E*
AF	1	0.8	0.5	0.6	0.8
B	0.8	1	0.4	0.5	0.7
C	0.5	0.4	1	0.3	0.5
D	0.6	0.5	0.3	1	0.4
E	0.8	0.7	0.5	0.4	1



Pareja	Similitud
AF	0,9
AE	0,8
BF	0,8
BE	0,7
AD	0,6
AC	0,5
BD	0,5
CE	0,5
BC	0,4
DE	0,4
AB	0,3
CD	0,3
EF	0,3
CF	0,2
DF	0,1

Hierarchical Agglomerative Clustering (HAC)
Ejemplo Enlace Simple (cont.)

$$\begin{array}{c} \\ AEF \\ B \\ C \\ D \end{array} \begin{array}{c} \begin{array}{cccc} AEF & B & C & D \end{array} \\ \begin{bmatrix} 1 & 0.8 & 0.5 & 0.6 \\ 0.8 & 1 & 0.4 & 0.5 \\ 0.5 & 0.4 & 1 & 0.3 \\ 0.6 & 0.5 & 0.3 & 1 \end{bmatrix} \end{array}$$

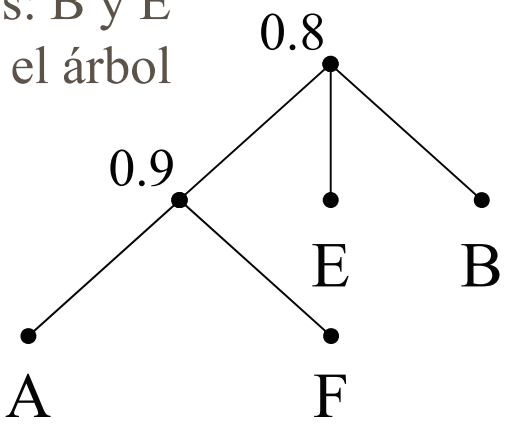


Pareja	Similitud
AF	0,9
AE	0,8
BF	0,8
BE	0,7
AD	0,6
AC	0,5
BD	0,5
CE	0,5
BC	0,4
DE	0,4
AB	0,3
CD	0,3
EF	0,3
CF	0,2
DF	0,1

Hierarchical Agglomerative Clustering (HAC)
Ejemplo Enlace Simple (cont.)

$$\begin{array}{c} \\ ABEF \\ C \\ D \end{array} \begin{array}{c} \begin{array}{ccc} ABEF & C & D \end{array} \\ \begin{bmatrix} 1 & 0.5 & 0.6 \\ 0.5 & 1 & 0.3 \\ 0.6 & 0.3 & 1 \end{bmatrix} \end{array}$$

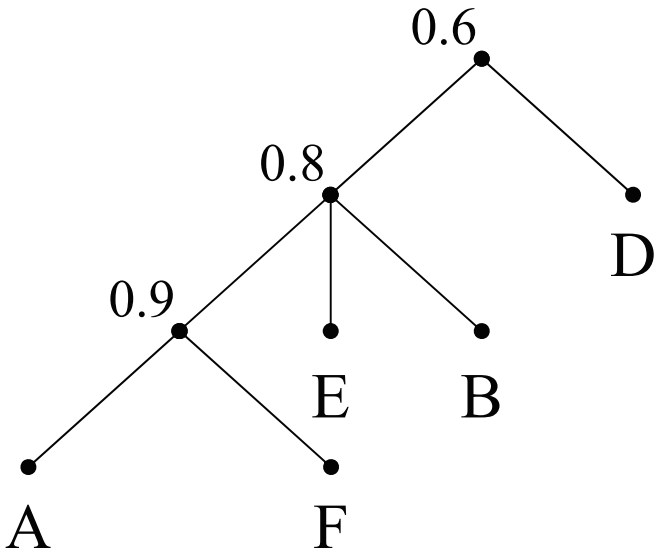
Sin cambios: B y E ya están en el árbol



Pareja	Similitud
AF	0,9
AE	0,8
BF	0,8
BE	0,7
AD	0,6
AC	0,5
BD	0,5
CE	0,5
BC	0,4
DE	0,4
AB	0,3
CD	0,3
EF	0,3
CF	0,2
DF	0,1

Hierarchical Agglomerative Clustering (HAC)
Ejemplo Enlace Simple (cont.)

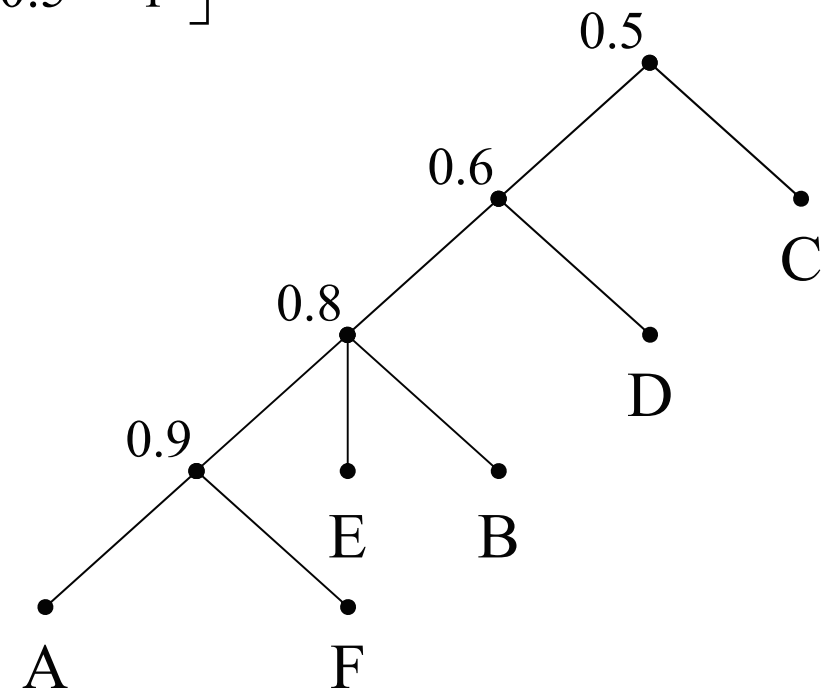
$$\begin{array}{c} \\ ABEF \\ C \\ D \end{array} \begin{array}{c} \begin{array}{ccc} ABEF & C & D \end{array} \\ \begin{bmatrix} 1 & 0.5 & 0.6 \\ 0.5 & 1 & 0.3 \\ 0.6 & 0.3 & 1 \end{bmatrix} \end{array}$$



Pareja	Similitud
AF	0,9
AE	0,8
BF	0,8
BE	0,7
AD	0,6
AC	0,5
BD	0,5
CE	0,5
BC	0,4
DE	0,4
AB	0,3
CD	0,3
EF	0,3
CF	0,2
DF	0,1

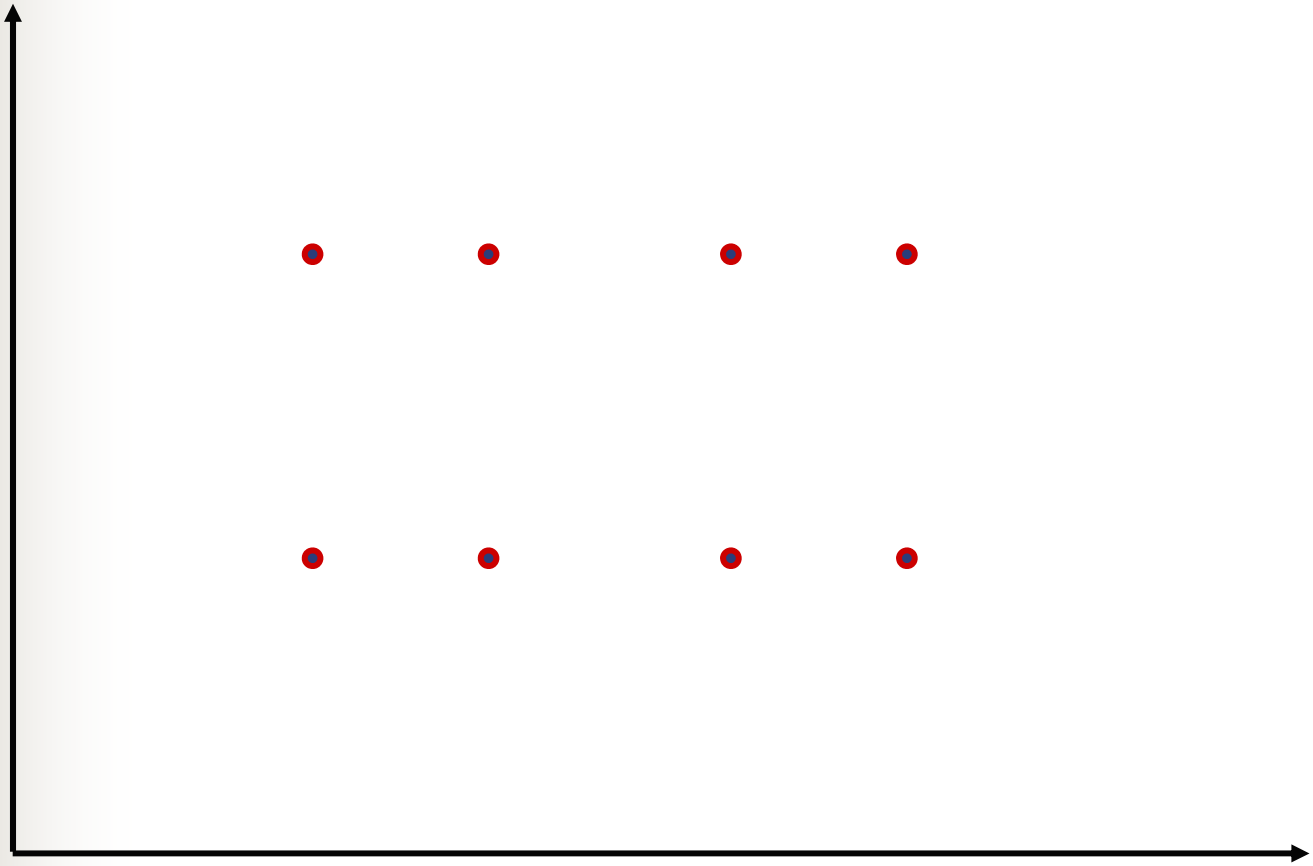
Hierarchical Agglomerative Clustering (HAC)
Ejemplo Enlace Simple (cont.)

$$\begin{array}{c} \\ ABDEF \\ C \end{array} \begin{array}{c} \begin{array}{cc} ABDEF & C \end{array} \\ \begin{bmatrix} 1 & 0.5 \\ 0.5 & 1 \end{bmatrix} \end{array}$$

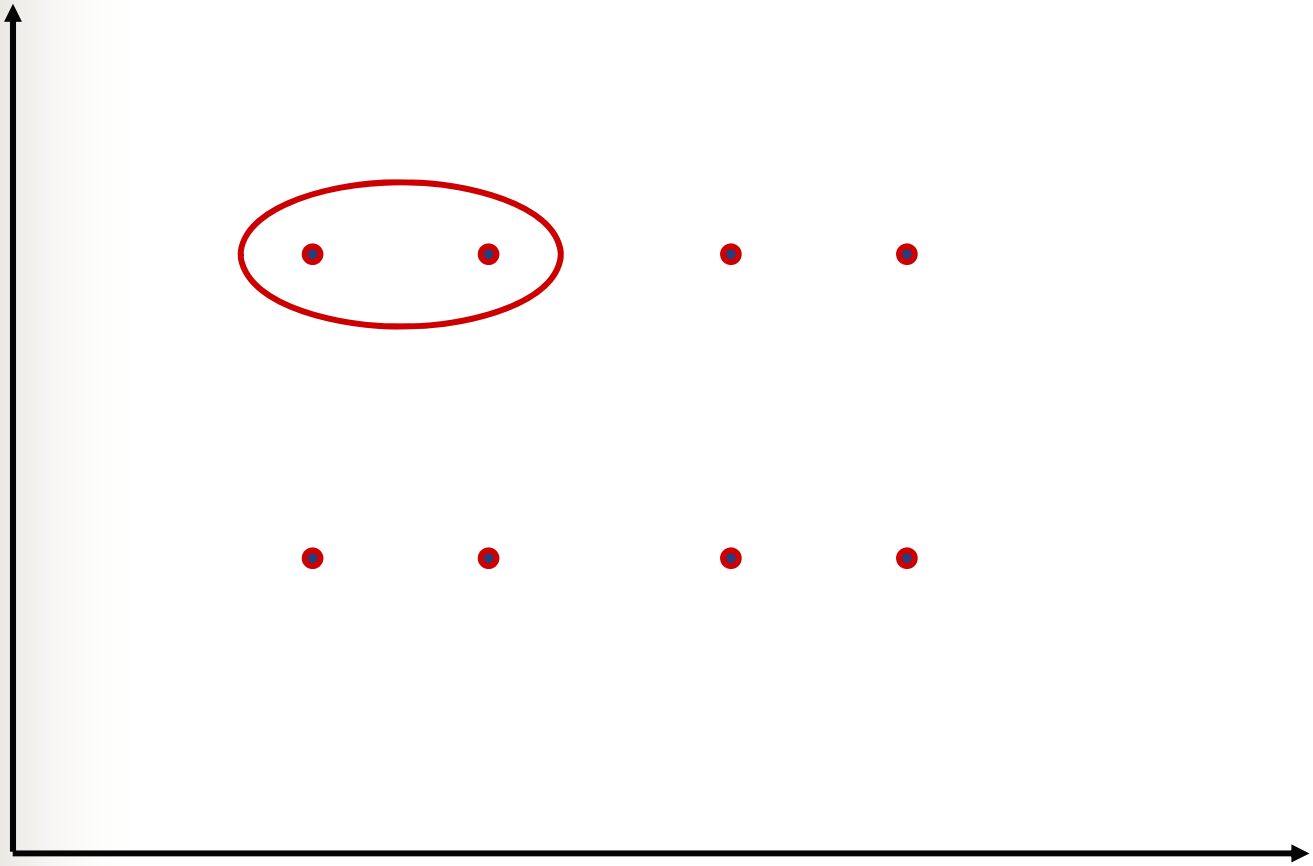


Pareja	Similitud
AF	0,9
AE	0,8
BF	0,8
BE	0,7
AD	0,6
AC	0,5
BD	0,5
CE	0,5
BC	0,4
DE	0,4
AB	0,3
CD	0,3
EF	0,3
CF	0,2
DF	0,1

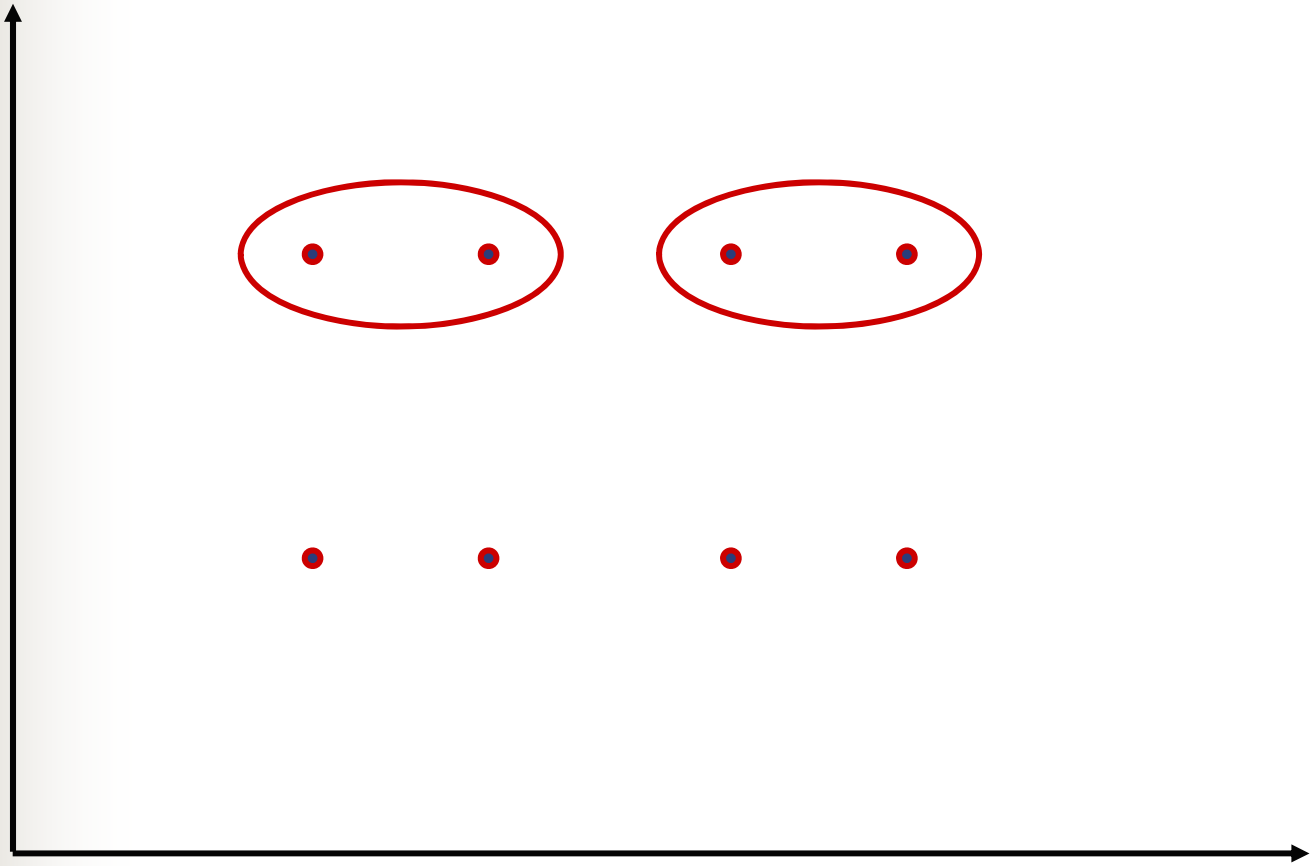
Hierarchical Agglomerative Clustering (HAC)
Ejemplo Enlace Simple (cont.)



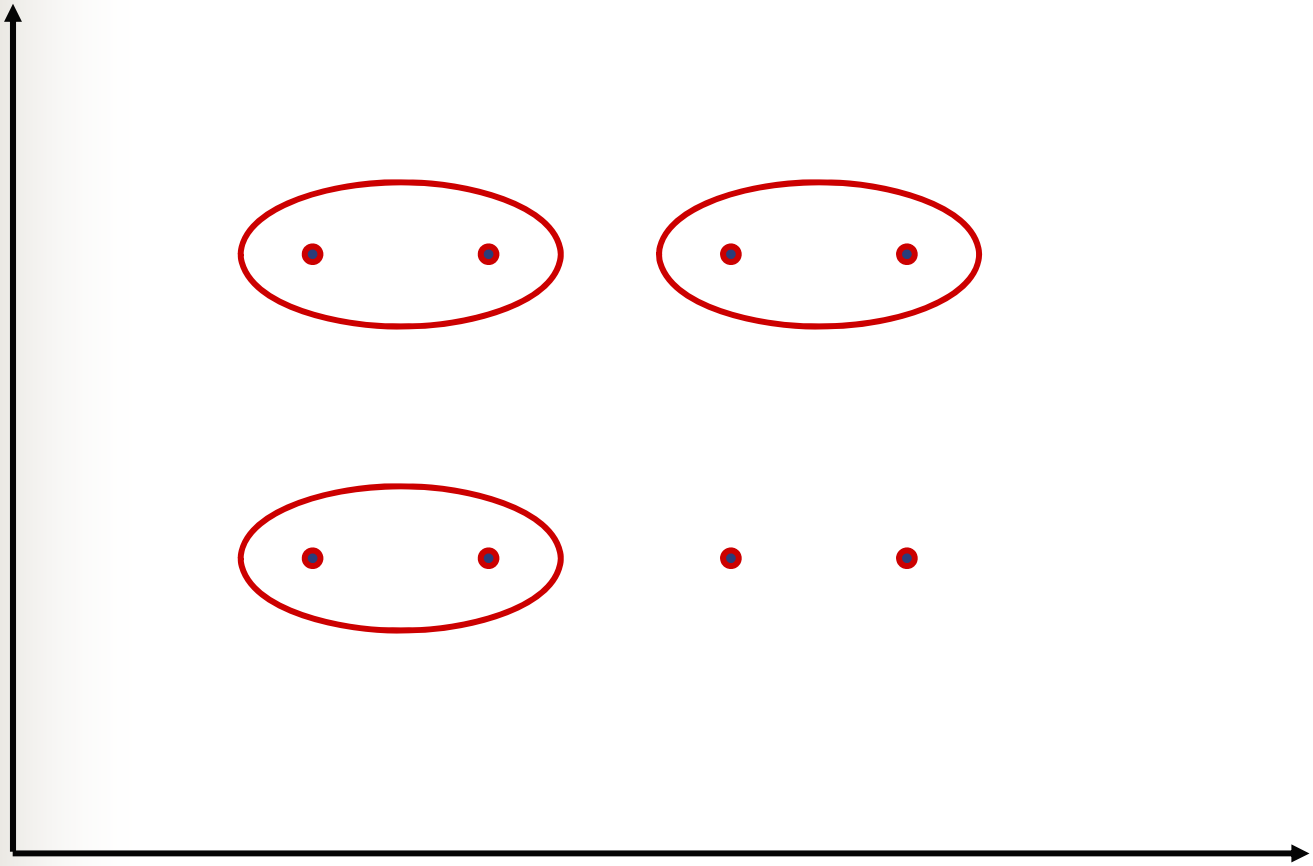
Hierarchical Agglomerative Clustering (HAC)
Ejemplo Enlace Simple (cont.)



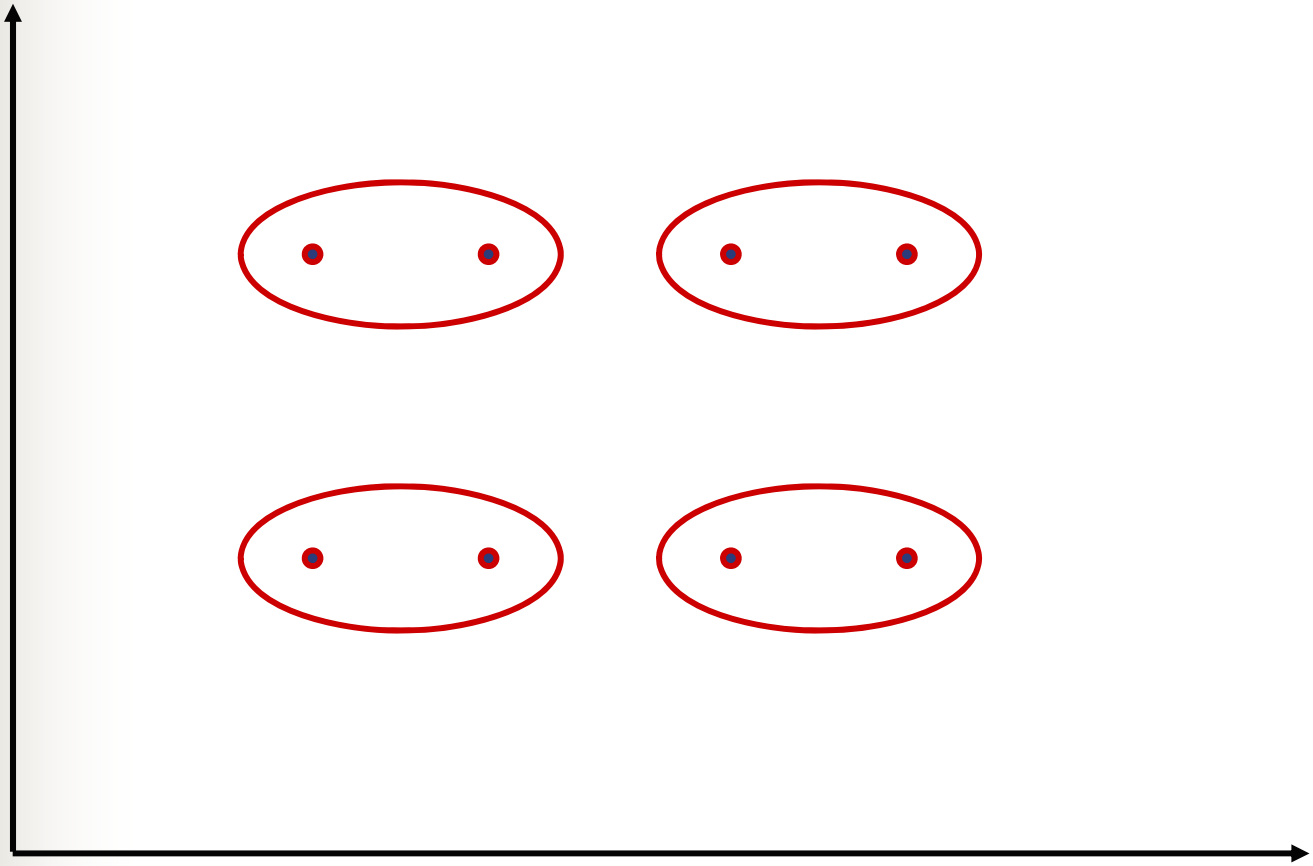
Hierarchical Agglomerative Clustering (HAC)
Ejemplo Enlace Simple (cont.)



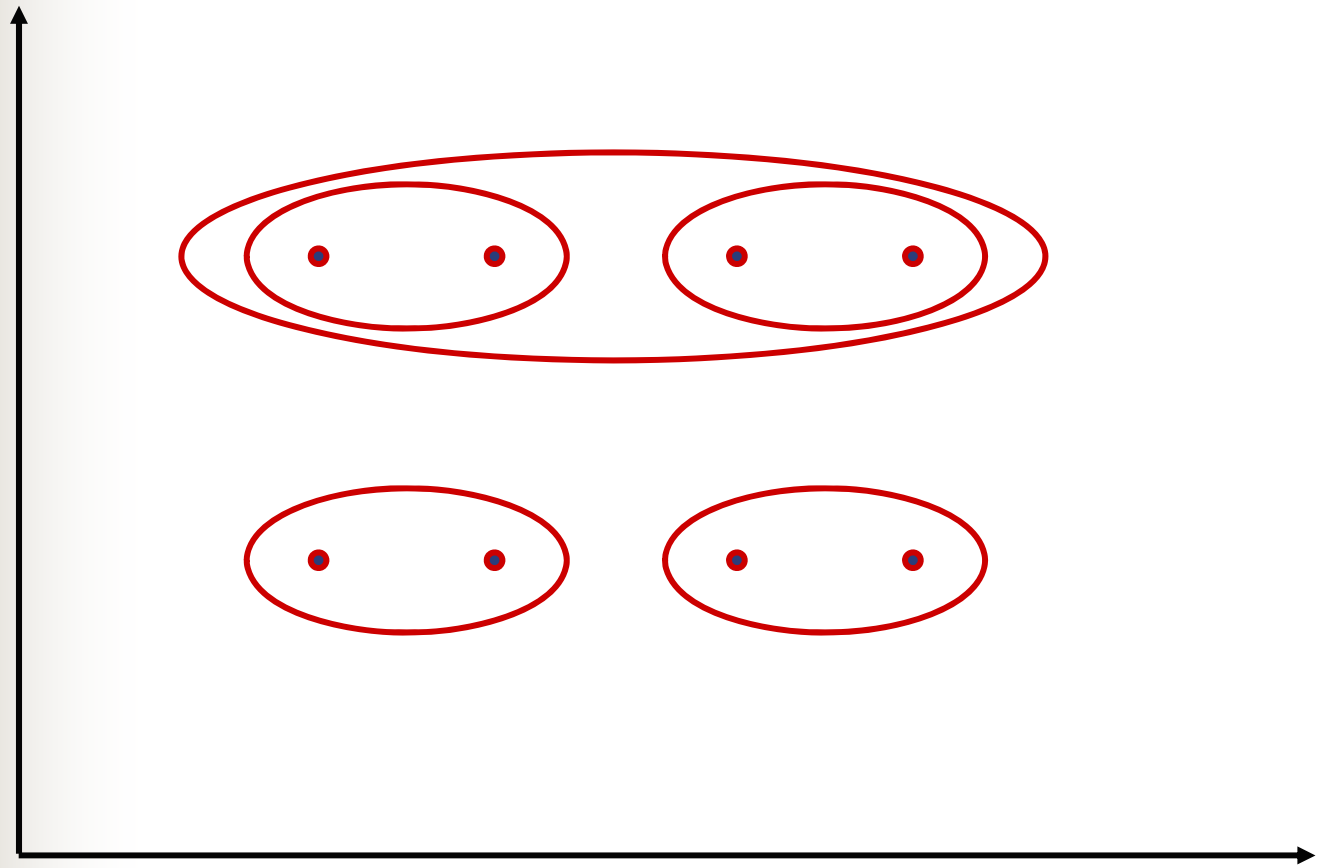
Hierarchical Agglomerative Clustering (HAC)
Ejemplo Enlace Simple (cont.)



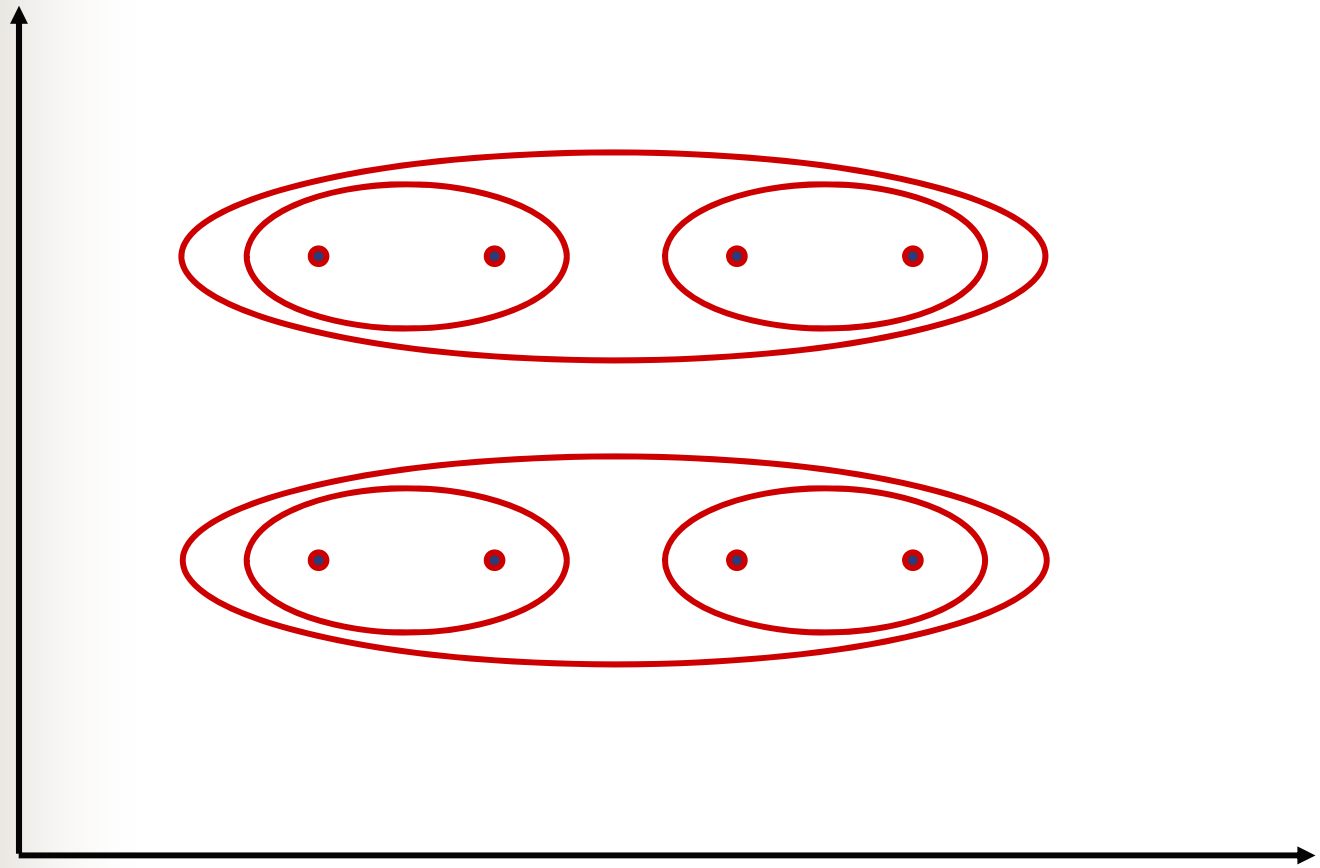
Hierarchical Agglomerative Clustering (HAC)
Ejemplo Enlace Simple (cont.)



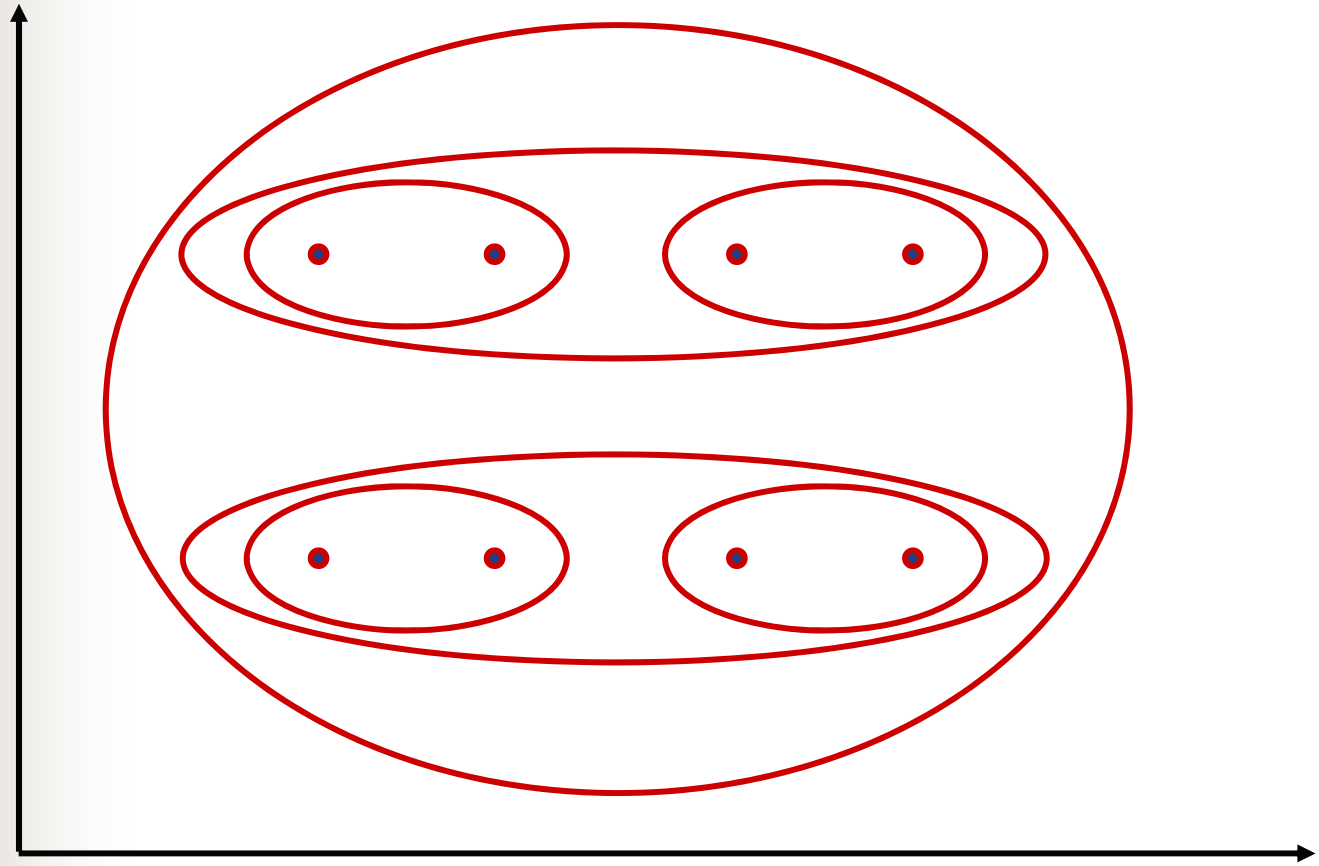
Hierarchical Agglomerative Clustering (HAC)
Ejemplo Enlace Simple (cont.)



Hierarchical Agglomerative Clustering (HAC)
Ejemplo Enlace Simple (cont.)

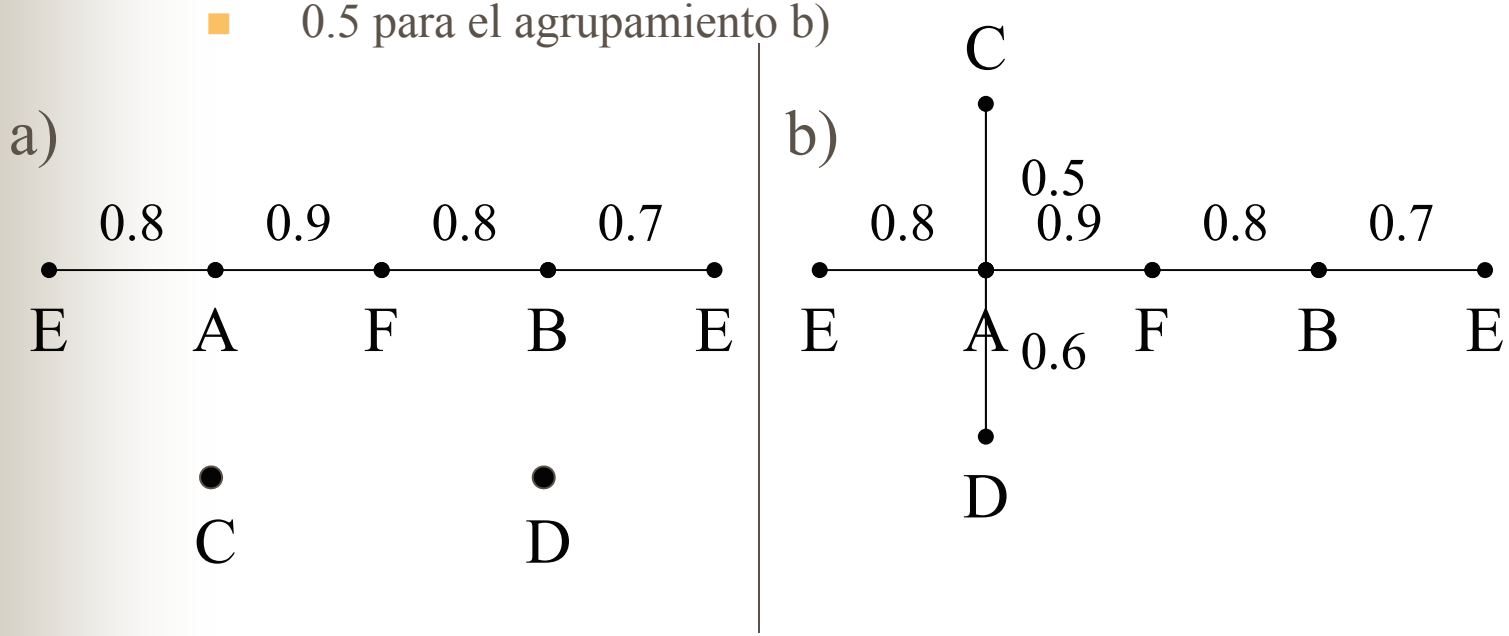


Hierarchical Agglomerative Clustering (HAC)
Ejemplo Enlace Simple (cont.)



Hierarchical Agglomerative Clustering (HAC)
Ejemplo Enlace Simple (cont.)

- Se tienen dos estructuras de agrupamiento para los datos de la tabla, en las que se estableció un umbral de similitud a partir del cual se considera el agrupamiento:
  - 0.7 para el agrupamiento a)
  - 0.5 para el agrupamiento b)



Pareja	Similitud
AF	0,9
AE	0,8
BF	0,8
BE	0,7
AD	0,6
AC	0,5
BD	0,5
CE	0,5
BC	0,4
DE	0,4
AB	0,3
CD	0,3
EF	0,3
CF	0,2
DF	0,1

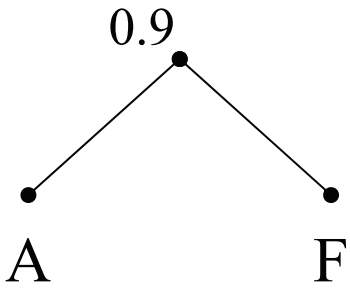
Hierarchical Agglomerative Clustering (HAC)
Ejemplo Enlace Simple (cont.)

- El método de enlace simple tiende a formar un pequeño número de grupos grandes, que se caracterizan por un efecto de encadenamiento en el que cada elemento está unido normalmente a tan sólo otro elemento del grupo.
- Los datos del ejemplo propician que los elementos se sumen consecutivamente a grupos que incluyen al grupo anterior:
  - Con la única alternativa de dejar elementos independientes en el caso a).
  - Sin embargo, podrían generarse dos o más grupos disjuntos si la ordenación decreciente de similitudes lo permite.

Hierarchical Agglomerative Clustering (HAC)
Ejemplo Enlace Completo (cont.)

	A	*B*	*C*	*D*	*E*	*F*
A	1	0.3	0.5	0.6	0.8	0.9
B	0.3	1	0.4	0.5	0.7	0.8
C	0.5	0.4	1	0.3	0.5	0.2
D	0.6	0.5	0.3	1	0.4	0.1
E	0.8	0.7	0.5	0.4	1	0.3
F	0.9	0.8	0.2	0.1	0.3	1

Nuevo



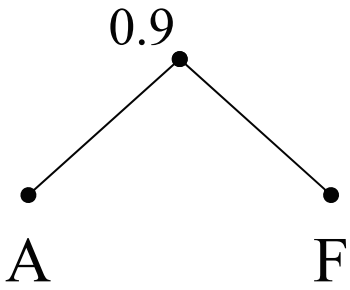
Pareja	Similitud
AF	0,9
AE	0,8
BF	0,8
BE	0,7
AD	0,6
AC	0,5
BD	0,5
CE	0,5
BC	0,4
DE	0,4
AB	0,3
CD	0,3
EF	0,3
CF	0,2
DF	0,1

Hierarchical Agglomerative Clustering (HAC)
Ejemplo Enlace Completo (cont.)

	AF	*B*	*C*	*D*	*E*
AF	1	0.3	0.2	0.1	0.3
B	0.3	1	0.4	0.5	0.7
C	0.2	0.4	1	0.3	0.5
D	0.1	0.5	0.3	1	0.4
E	0.3	0.7	0.5	0.4	1

Comprobar EF? EF = 0.3 < AE = 0.8

No hay cambios



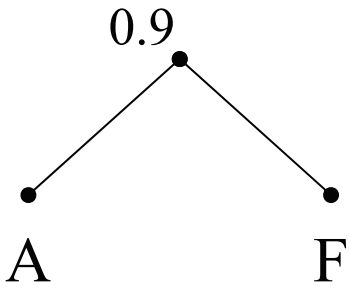
Pareja	Similitud
AF	0,9
AE	0,8
BF	0,8
BE	0,7
AD	0,6
AC	0,5
BD	0,5
CE	0,5
BC	0,4
DE	0,4
AB	0,3
CD	0,3
EF	0,3
CF	0,2
DF	0,1

Hierarchical Agglomerative Clustering (HAC)
Ejemplo Enlace Completo (cont.)

	AF	*B*	*C*	*D*	*E*
AF	1	0.3	0.2	0.1	0.3
B	0.3	1	0.4	0.5	0.7
C	0.2	0.4	1	0.3	0.5
D	0.1	0.5	0.3	1	0.4
E	0.3	0.7	0.5	0.4	1

Comprobar AB? AB = 0.3 < BF = 0.8

No hay cambios

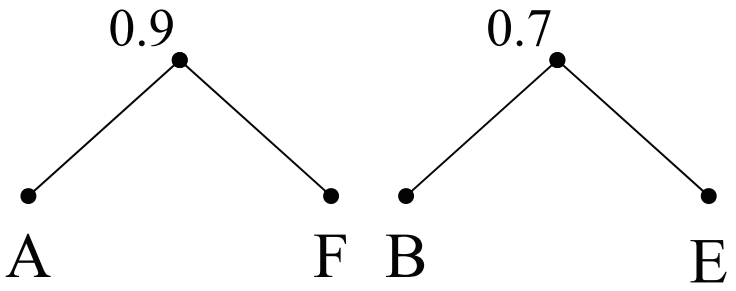


Pareja	Similitud
AF	0,9
AE	0,8
BF	0,8
BE	0,7
AD	0,6
AC	0,5
BD	0,5
CE	0,5
BC	0,4
DE	0,4
AB	0,3
CD	0,3
EF	0,3
CF	0,2
DF	0,1

Hierarchical Agglomerative Clustering (HAC)
Ejemplo Enlace Completo (cont.)

	AF	*B*	*C*	*D*	*E*
AF	1	0.3	0.2	0.1	0.3
B	0.3	1	0.4	0.5	0.7
C	0.2	0.4	1	0.3	0.5
D	0.1	0.5	0.3	1	0.4
E	0.3	0.7	0.5	0.4	1

Nuevo

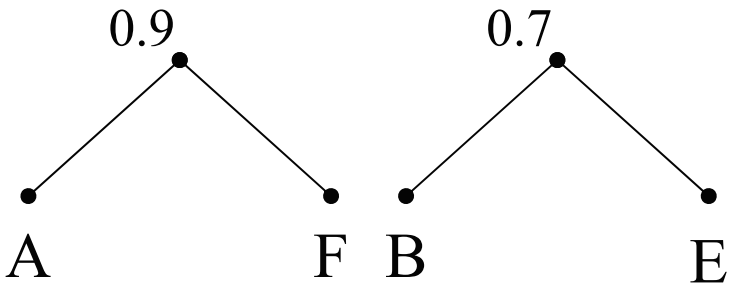


Pareja	Similitud
AF	0,9
AE	0,8
BF	0,8
BE	0,7
AD	0,6
AC	0,5
BD	0,5
CE	0,5
BC	0,4
DE	0,4
AB	0,3
CD	0,3
EF	0,3
CF	0,2
DF	0,1

Hierarchical Agglomerative Clustering (HAC)
Ejemplo Enlace Completo (cont.)

$$\begin{array}{c} \\ AF \\ BE \\ C \\ D \end{array} \begin{array}{c} \begin{array}{cccc} AF & BE & C & D \end{array} \\ \begin{bmatrix} 1 & 0.3 & 0.2 & 0.1 \\ 0.3 & 1 & 0.4 & 0.4 \\ 0.2 & 0.4 & 1 & 0.3 \\ 0.1 & 0.4 & 0.3 & 1 \end{bmatrix} \end{array}$$

Comprobar DF? DF = 0.1 < AD = 0.6
No hay cambios



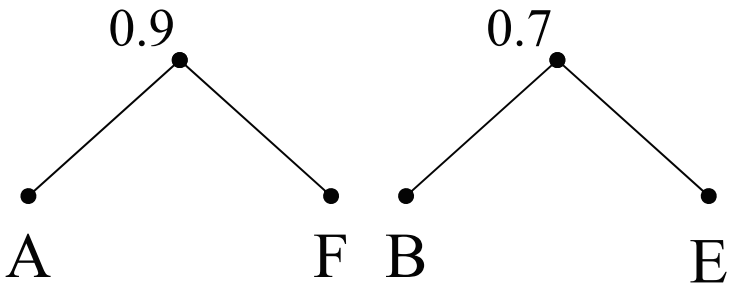
Pareja	Similitud
AF	0,9
AE	0,8
BF	0,8
BE	0,7
AD	0,6
AC	0,5
BD	0,5
CE	0,5
BC	0,4
DE	0,4
AB	0,3
CD	0,3
EF	0,3
CF	0,2
DF	0,1

Hierarchical Agglomerative Clustering (HAC)
Ejemplo Enlace Completo (cont.)

	AF	*BE*	*C*	*D*
AF	1	0.3	0.2	0.1
BE	0.3	1	0.4	0.4
C	0.2	0.4	1	0.3
D	0.1	0.4	0.3	1

Comprobar CF? CF = 0.2 < AC = 0.5

No hay cambios

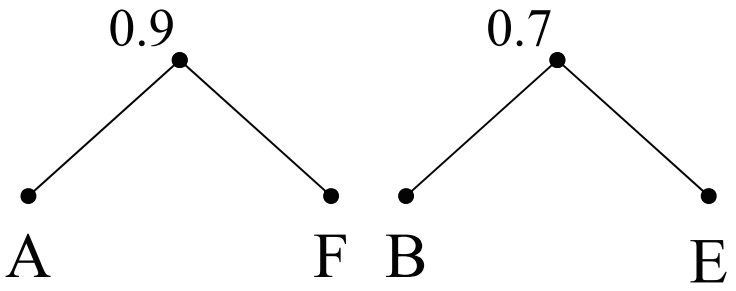


Pareja	Similitud
AF	0,9
AE	0,8
BF	0,8
BE	0,7
AD	0,6
AC	0,5
BD	0,5
CE	0,5
BC	0,4
DE	0,4
AB	0,3
CD	0,3
EF	0,3
CF	0,2
DF	0,1

Hierarchical Agglomerative Clustering (HAC)
Ejemplo Enlace Completo (cont.)

	AF	*BE*	*C*	*D*
AF	1	0.3	0.2	0.1
BE	0.3	1	0.4	0.4
C	0.2	0.4	1	0.3
D	0.1	0.4	0.3	1

Comprobar DE? DE = 0.4 < BD = 0.5
No hay cambios



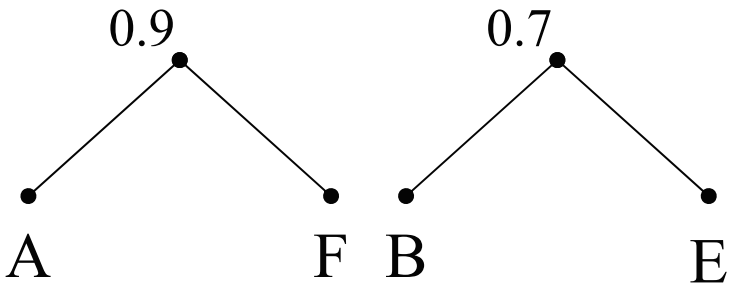
Pareja	Similitud
AF	0,9
AE	0,8
BF	0,8
BE	0,7
AD	0,6
AC	0,5
BD	0,5
CE	0,5
BC	0,4
DE	0,4
AB	0,3
CD	0,3
EF	0,3
CF	0,2
DF	0,1

Hierarchical Agglomerative Clustering (HAC)
Ejemplo Enlace Completo (cont.)

$$\begin{array}{c} \\ AF \\ BE \\ C \\ D \end{array} \begin{array}{c} \begin{array}{cccc} AF & BE & C & D \end{array} \\ \begin{bmatrix} 1 & 0.3 & 0.2 & 0.1 \\ 0.3 & 1 & 0.4 & 0.4 \\ 0.2 & 0.4 & 1 & 0.3 \\ 0.1 & 0.4 & 0.3 & 1 \end{bmatrix} \end{array}$$

Comprobar BC? $BC = 0.4 < CE = 0.5$

No hay cambios

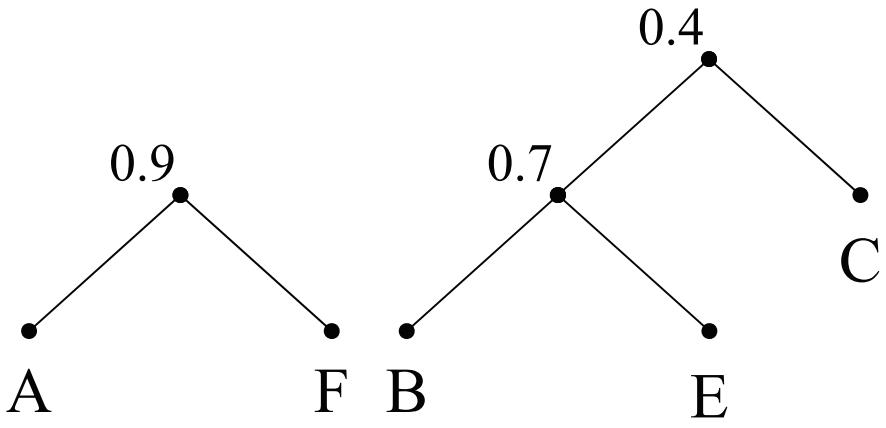


Pareja	Similitud
AF	0,9
AE	0,8
BF	0,8
BE	0,7
AD	0,6
AC	0,5
BD	0,5
CE	0,5
BC	0,4
DE	0,4
AB	0,3
CD	0,3
EF	0,3
CF	0,2
DF	0,1

Hierarchical Agglomerative Clustering (HAC)
Ejemplo Enlace Completo (cont.)

	AF	*BE*	*C*	*D*
AF	1	0.3	0.2	0.1
BE	0.3	1	0.4	0.4
C	0.2	0.4	1	0.3
D	0.1	0.4	0.3	1

Comprobar CE? CE = 0.5 > BC = 0.4



Pareja	Similitud
AF	0,9
AE	0,8
BF	0,8
BE	0,7
AD	0,6
AC	0,5
BD	0,5
CE	0,5
BC	0,4
DE	0,4
AB	0,3
CD	0,3
EF	0,3
CF	0,2
DF	0,1

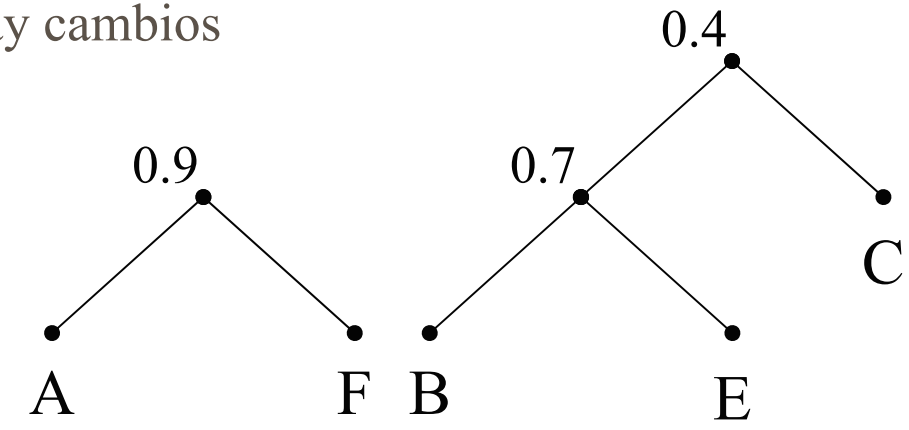
Hierarchical Agglomerative Clustering (HAC)
Ejemplo Enlace Completo (cont.)

$$\begin{array}{c} \\ AF \\ BCE \\ D \end{array} \begin{array}{c} \begin{array}{ccc} AF & BCE & D \end{array} \\ \begin{bmatrix} 1 & 0.2 & 0.1 \\ 0.2 & 1 & 0.3 \\ 0.1 & 0.3 & 1 \end{bmatrix} \end{array}$$

Comprobar BD? BD = 0.5 > DE = 0.4

Comprobar CD? CD = 0.3 < DE = 0.4

No hay cambios

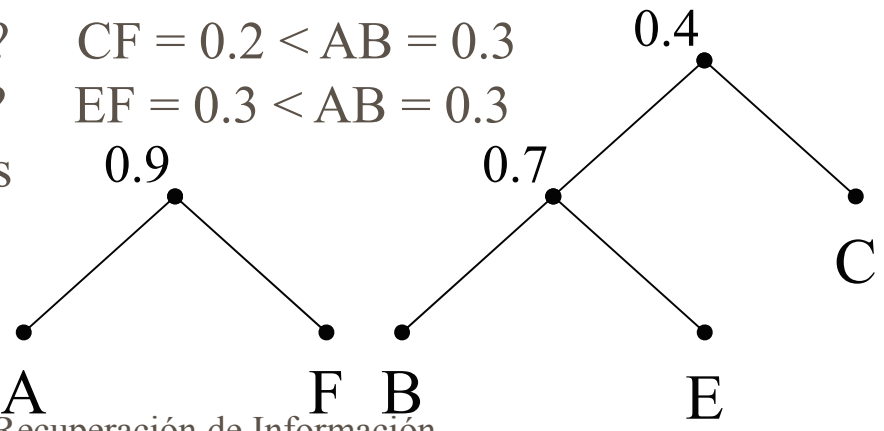


Pareja	Similitud
AF	0,9
AE	0,8
BF	0,8
BE	0,7
AD	0,6
AC	0,5
BD	0,5
CE	0,5
BC	0,4
DE	0,4
AB	0,3
CD	0,3
EF	0,3
CF	0,2
DF	0,1

Hierarchical Agglomerative Clustering (HAC)
Ejemplo Enlace Completo (cont.)

$$\begin{array}{c} \\ AF \\ BCE \\ D \end{array} \begin{array}{c} \begin{array}{ccc} AF & BCE & D \end{array} \\ \begin{bmatrix} 1 & 0.2 & 0.1 \\ 0.2 & 1 & 0.3 \\ 0.1 & 0.3 & 1 \end{bmatrix} \end{array}$$

Comprobar AC? AC = 0.5 > AB = 0.3
Comprobar AE? AE = 0.8 > AB = 0.3
Comprobar BF? BF = 0.8 > AB = 0.3
Comprobar CF? CF = 0.2 < AB = 0.3
Comprobar EF? EF = 0.3 < AB = 0.3
No hay cambios



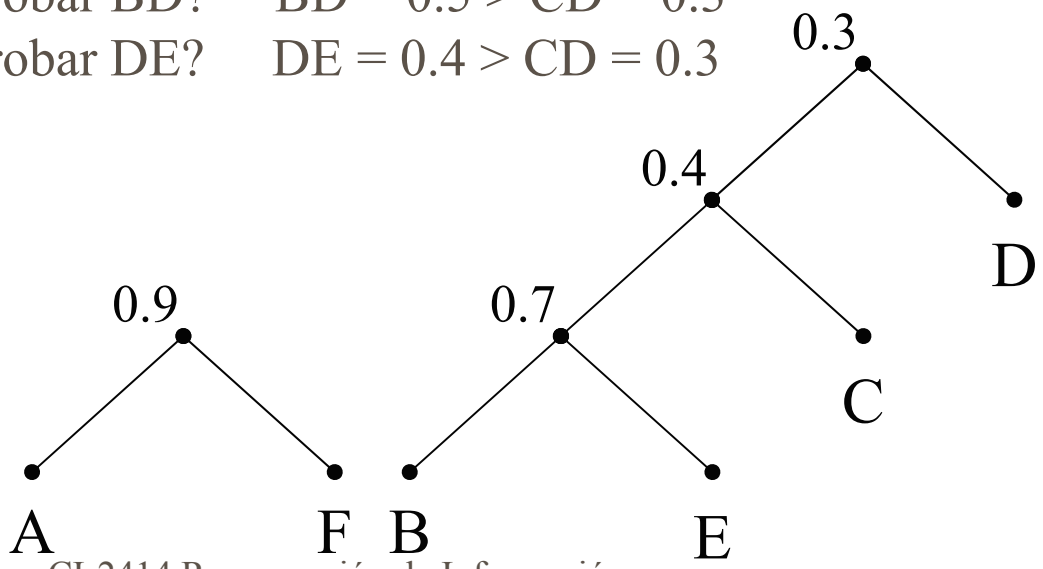
Pareja	Similitud
AF	0,9
AE	0,8
BF	0,8
BE	0,7
AD	0,6
AC	0,5
BD	0,5
CE	0,5
BC	0,4
DE	0,4
AB	0,3
CD	0,3
EF	0,3
CF	0,2
DF	0,1

Hierarchical Agglomerative Clustering (HAC)
Ejemplo Enlace Completo (cont.)

$$\begin{array}{c} \\ AF \\ BCE \\ D \end{array} \begin{array}{c} \begin{array}{ccc} AF & BCE & D \end{array} \\ \begin{bmatrix} 1 & 0.2 & 0.1 \\ 0.2 & 1 & 0.3 \\ 0.1 & 0.3 & 1 \end{bmatrix} \end{array}$$

Comprobar BD? BD = 0.5 > CD = 0.3

Comprobar DE? DE = 0.4 > CD = 0.3

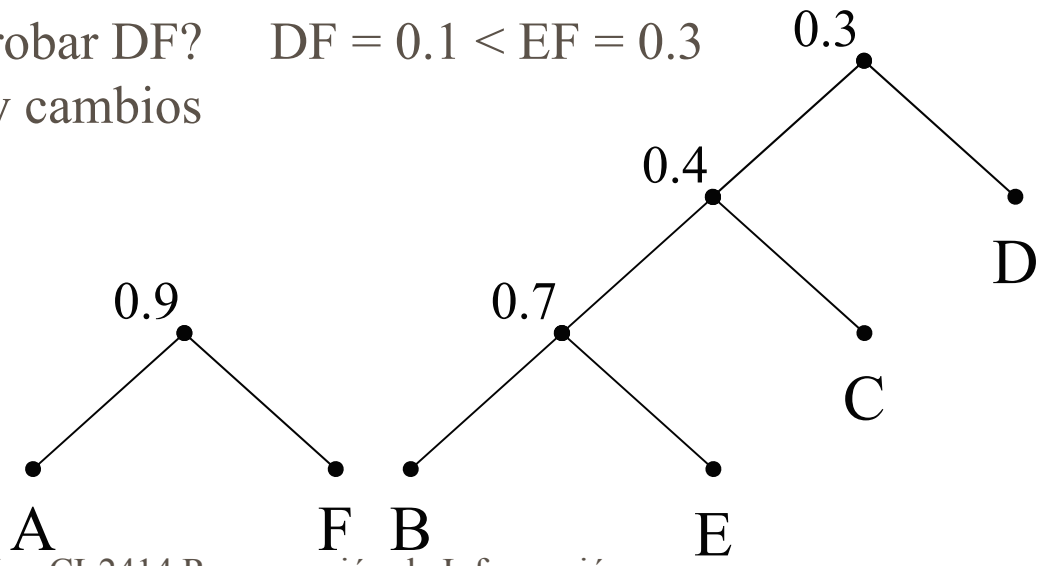


Pareja	Similitud
AF	0,9
AE	0,8
BF	0,8
BE	0,7
AD	0,6
AC	0,5
BD	0,5
CE	0,5
BC	0,4
DE	0,4
AB	0,3
CD	0,3
EF	0,3
CF	0,2
DF	0,1

Hierarchical Agglomerative Clustering (HAC)
Ejemplo Enlace Completo (cont.)

$$\begin{array}{c} \\ AF \\ BCDE \end{array} \begin{array}{c} \begin{array}{cc} AF & BCDE \end{array} \\ \begin{bmatrix} 1 & 0.1 \\ 0.1 & 1 \end{bmatrix} \end{array}$$

Comprobar BF? BF = 0.8 > EF = 0.3
Comprobar CF? CF = 0.2 < EF = 0.3
Comprobar DF? DF = 0.1 < EF = 0.3
No hay cambios

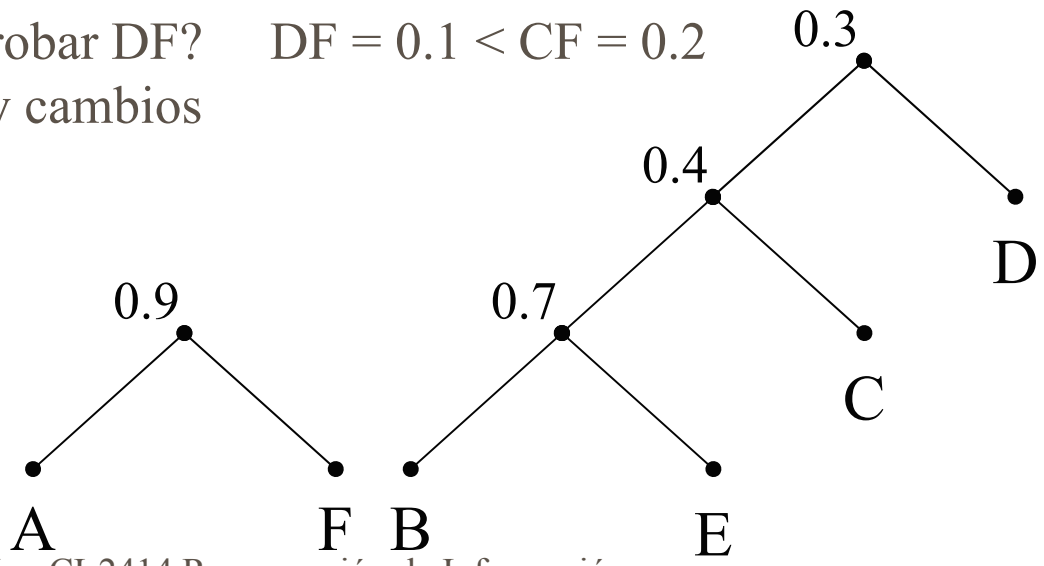


Pareja	Similitud
AF	0,9
AE	0,8
BF	0,8
BE	0,7
AD	0,6
AC	0,5
BD	0,5
CE	0,5
BC	0,4
DE	0,4
AB	0,3
CD	0,3
EF	0,3
CF	0,2
DF	0,1

Hierarchical Agglomerative Clustering (HAC)
Ejemplo Enlace Completo (cont.)

$$\begin{array}{c} \\ AF \\ BCDE \end{array} \begin{array}{c} \begin{array}{cc} AF & BCDE \end{array} \\ \begin{bmatrix} 1 & 0.1 \\ 0.1 & 1 \end{bmatrix} \end{array}$$

Comprobar BF? BF = 0.8 > CF = 0.2
Comprobar EF? CF = 0.3 > CF = 0.2
Comprobar DF? DF = 0.1 < CF = 0.2
No hay cambios

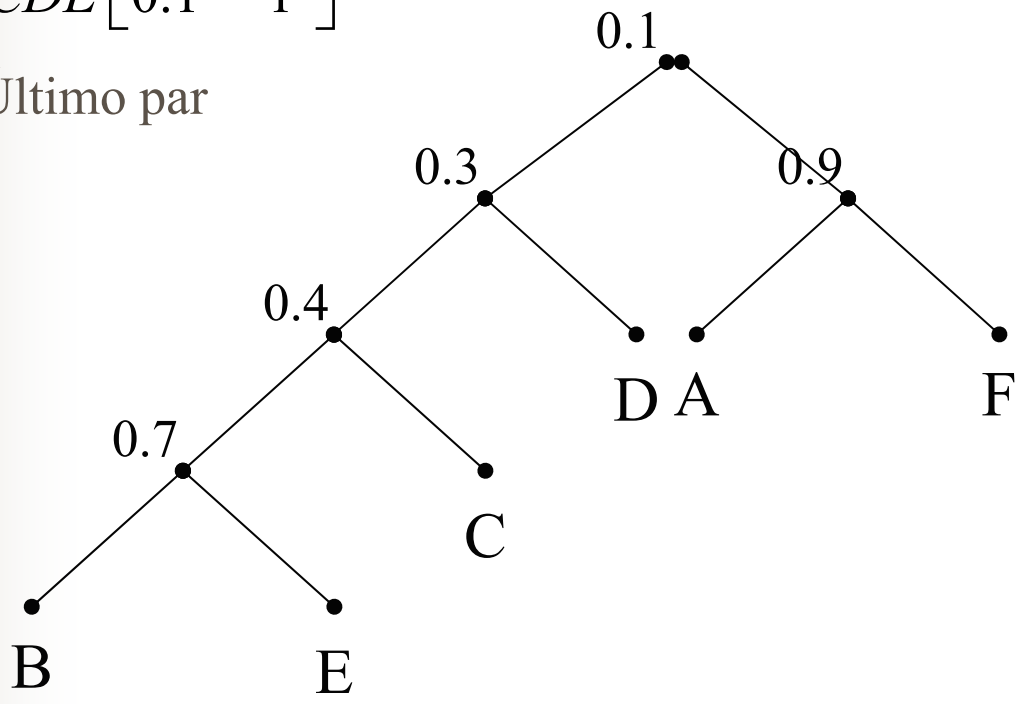


Pareja	Similitud
AF	0,9
AE	0,8
BF	0,8
BE	0,7
AD	0,6
AC	0,5
BD	0,5
CE	0,5
BC	0,4
DE	0,4
AB	0,3
CD	0,3
EF	0,3
CF	0,2
DF	0,1

Hierarchical Agglomerative Clustering (HAC)
Ejemplo Enlace Completo (cont.)

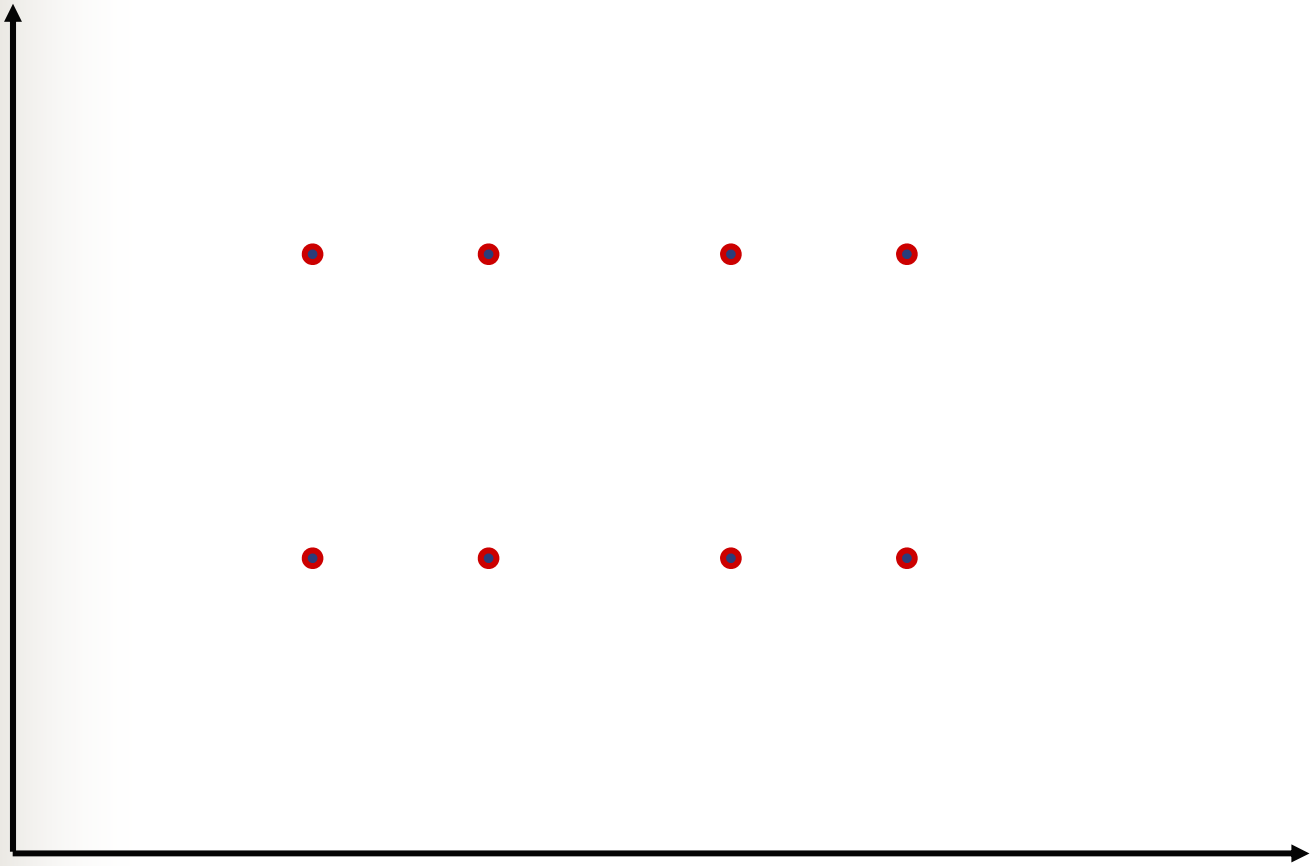
$$\begin{array}{c} \\ AF \\ BCDE \end{array} \begin{array}{c} \begin{array}{cc} AF & BCDE \end{array} \\ \begin{bmatrix} 1 & 0.1 \\ 0.1 & 1 \end{bmatrix} \end{array}$$

Último par

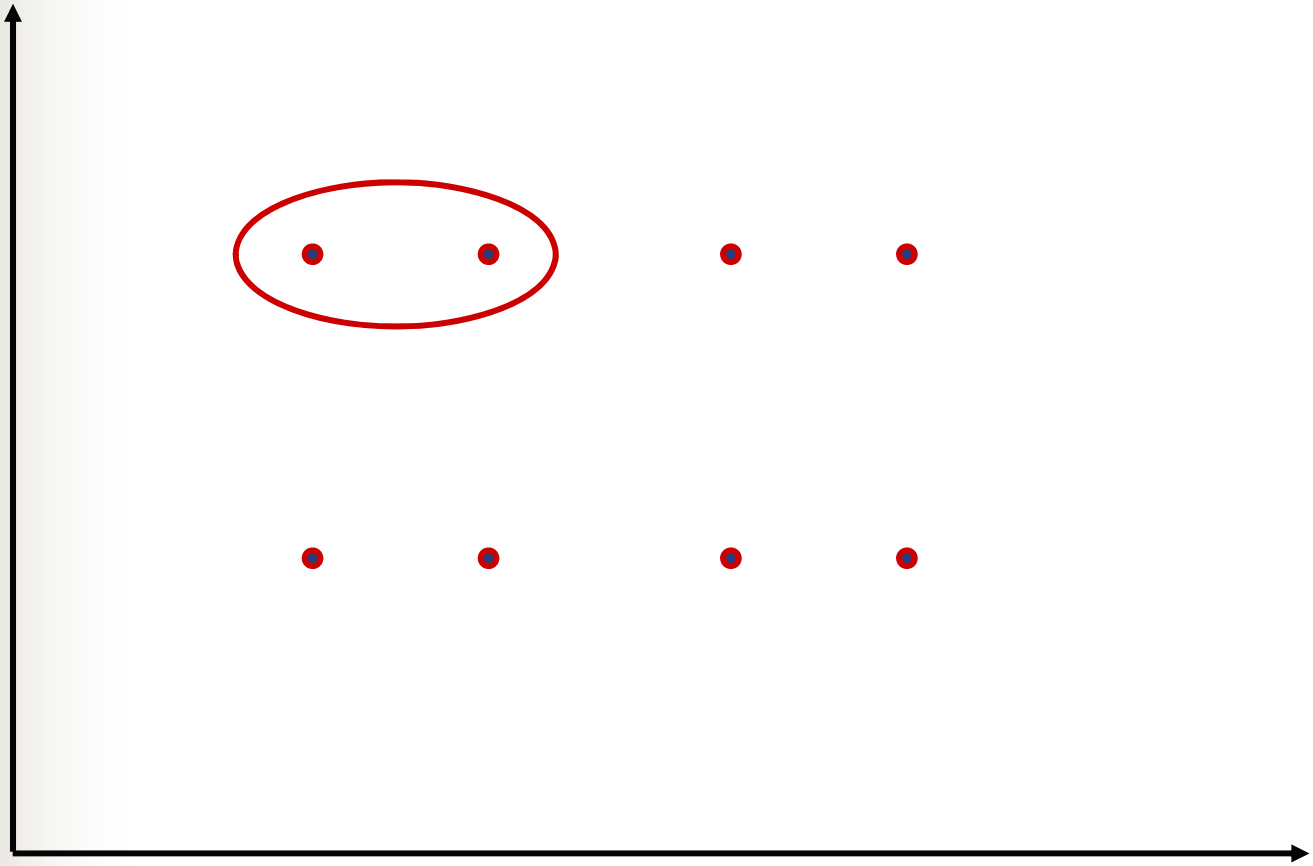


Pareja	Similitud
AF	0,9
AE	0,8
BF	0,8
BE	0,7
AD	0,6
AC	0,5
BD	0,5
CE	0,5
BC	0,4
DE	0,4
AB	0,3
CD	0,3
EF	0,3
CF	0,2
DF	0,1

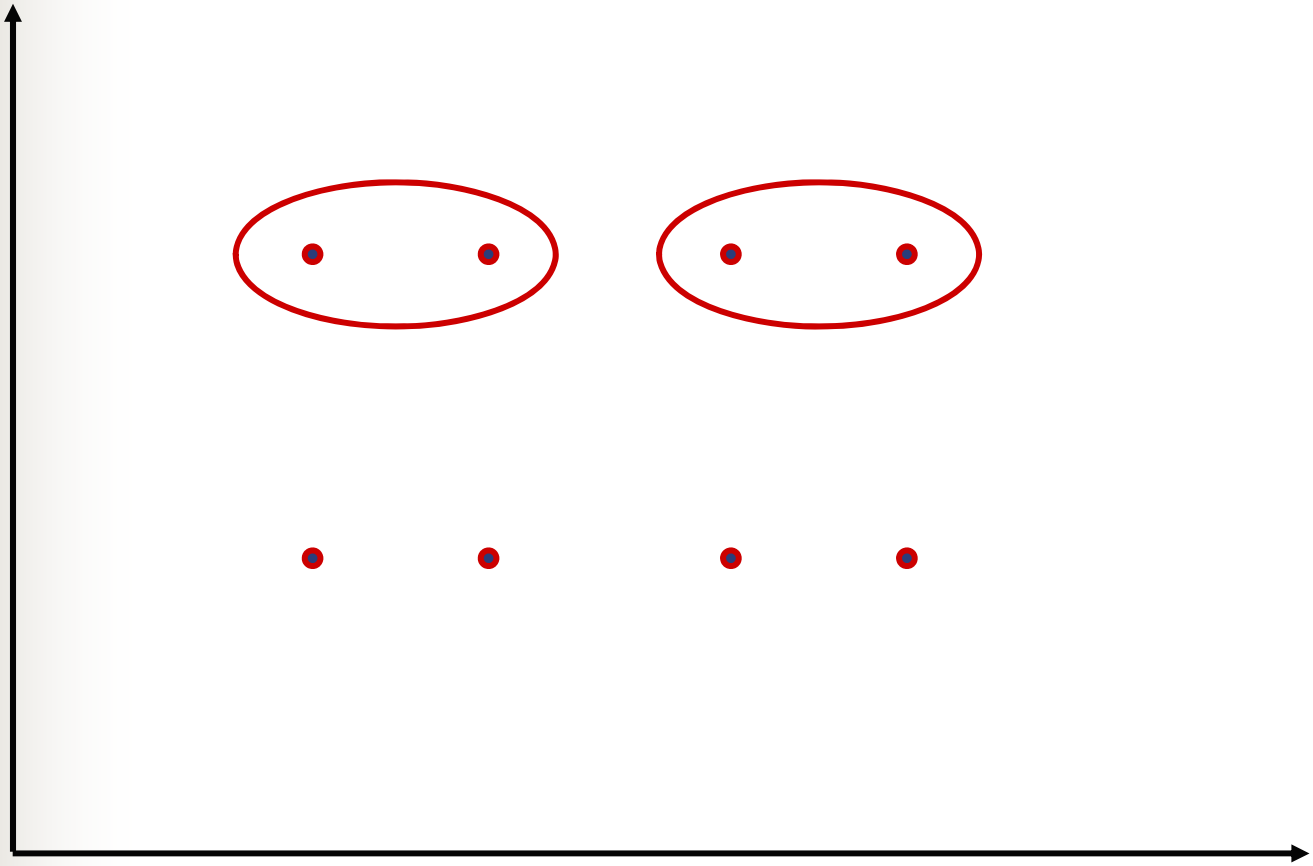
Hierarchical Agglomerative Clustering (HAC)
Ejemplo Enlace Completo (cont.)



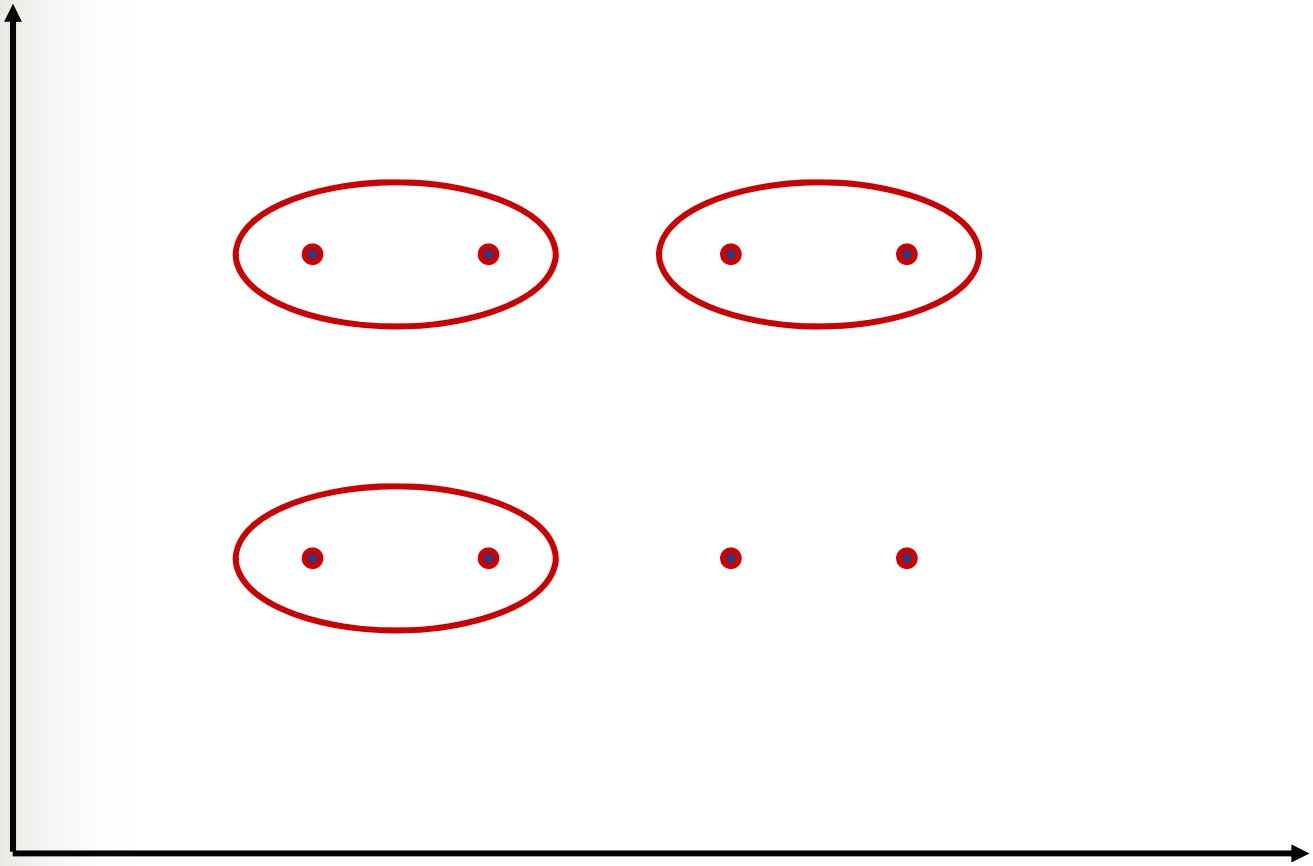
Hierarchical Agglomerative Clustering (HAC)
Ejemplo Enlace Completo (cont.)



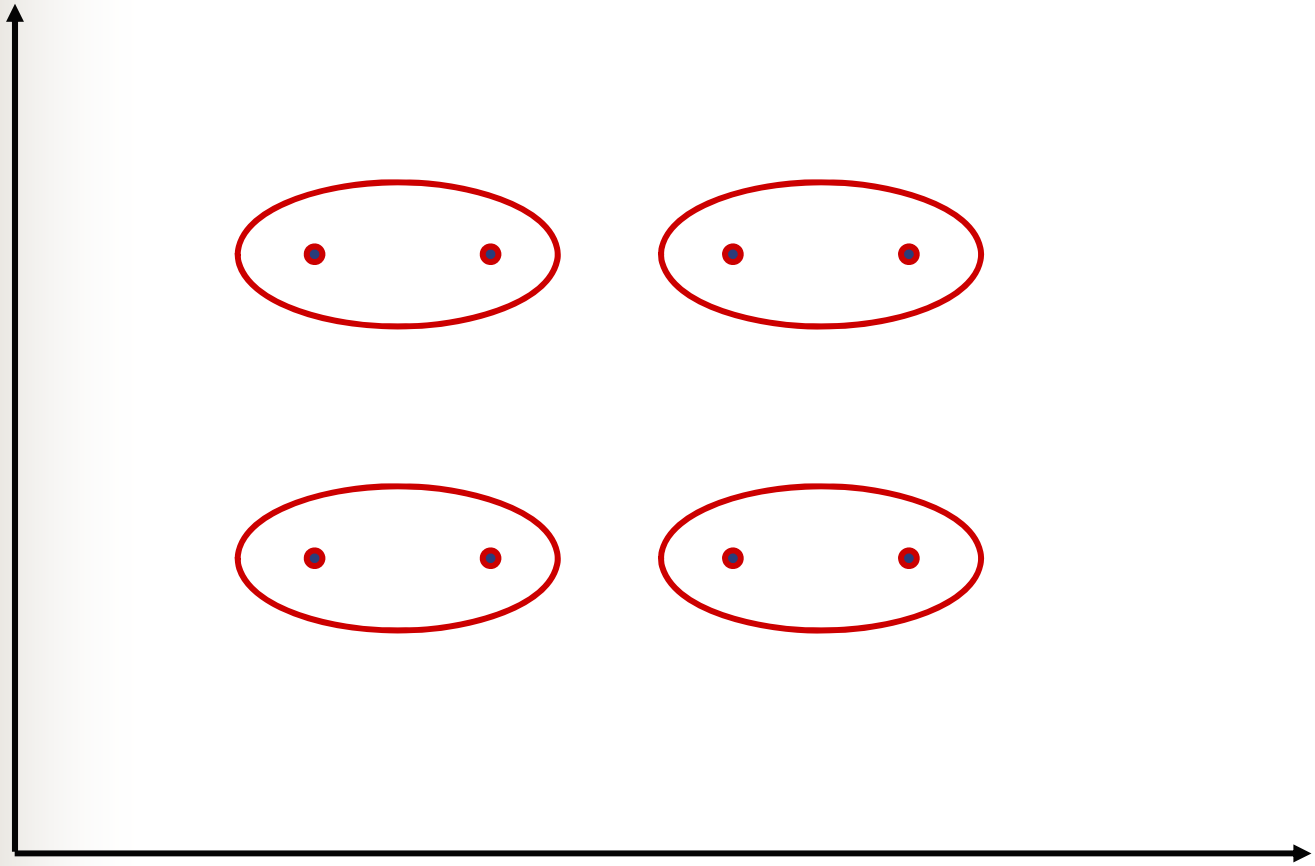
Hierarchical Agglomerative Clustering (HAC)
Ejemplo Enlace Completo (cont.)



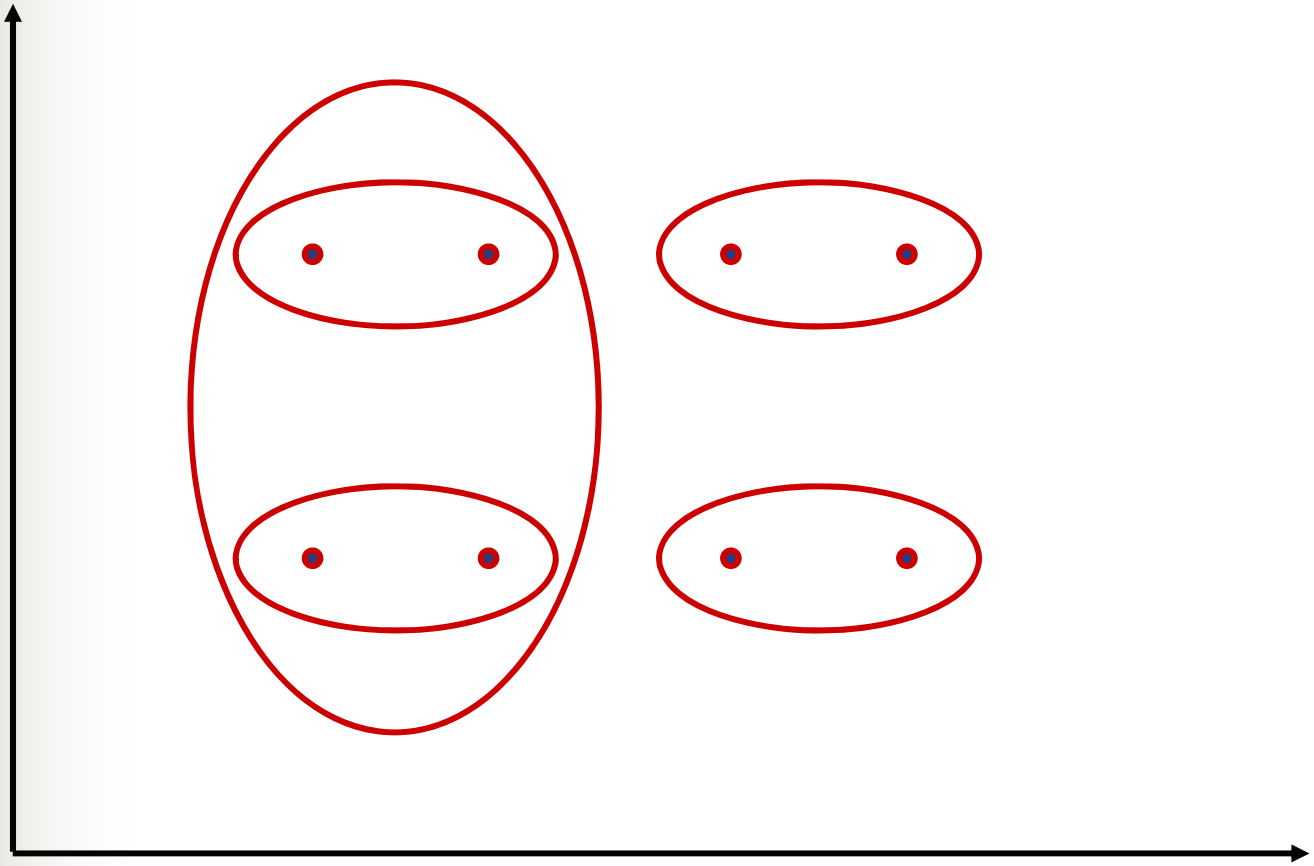
Hierarchical Agglomerative Clustering (HAC)
Ejemplo Enlace Completo (cont.)



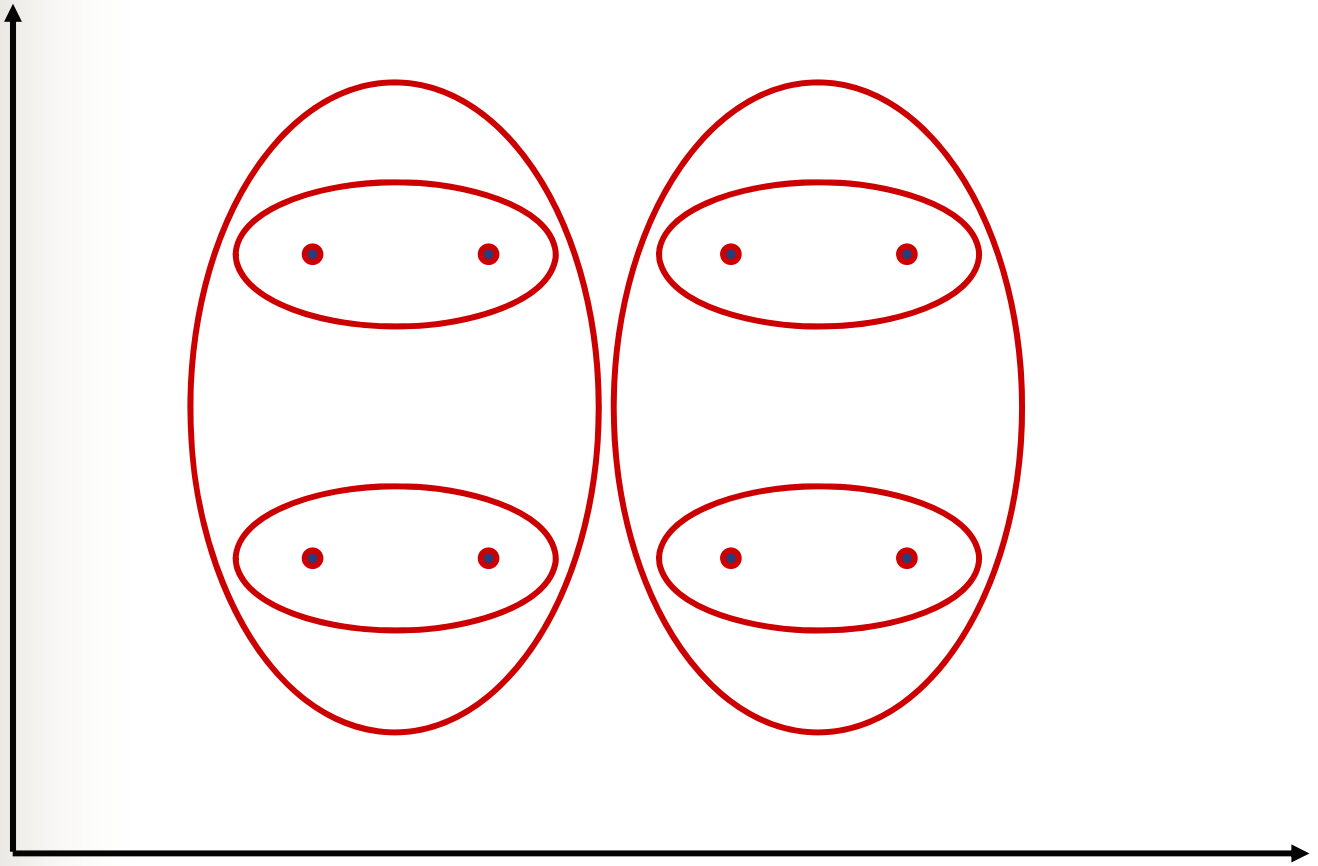
Hierarchical Agglomerative Clustering (HAC)
Ejemplo Enlace Completo (cont.)



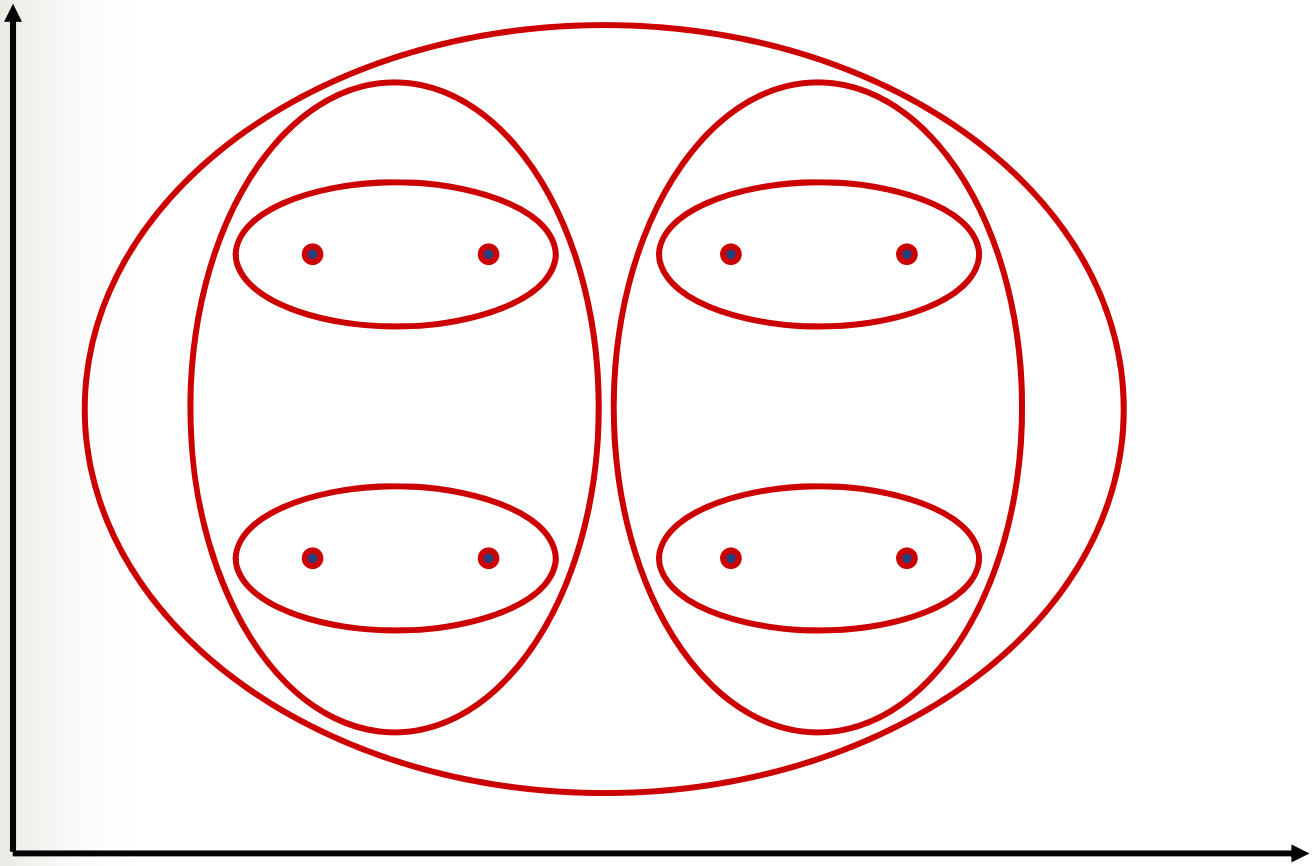
Hierarchical Agglomerative Clustering (HAC)
Ejemplo Enlace Completo (cont.)



Hierarchical Agglomerative Clustering (HAC)
Ejemplo Enlace Completo (cont.)

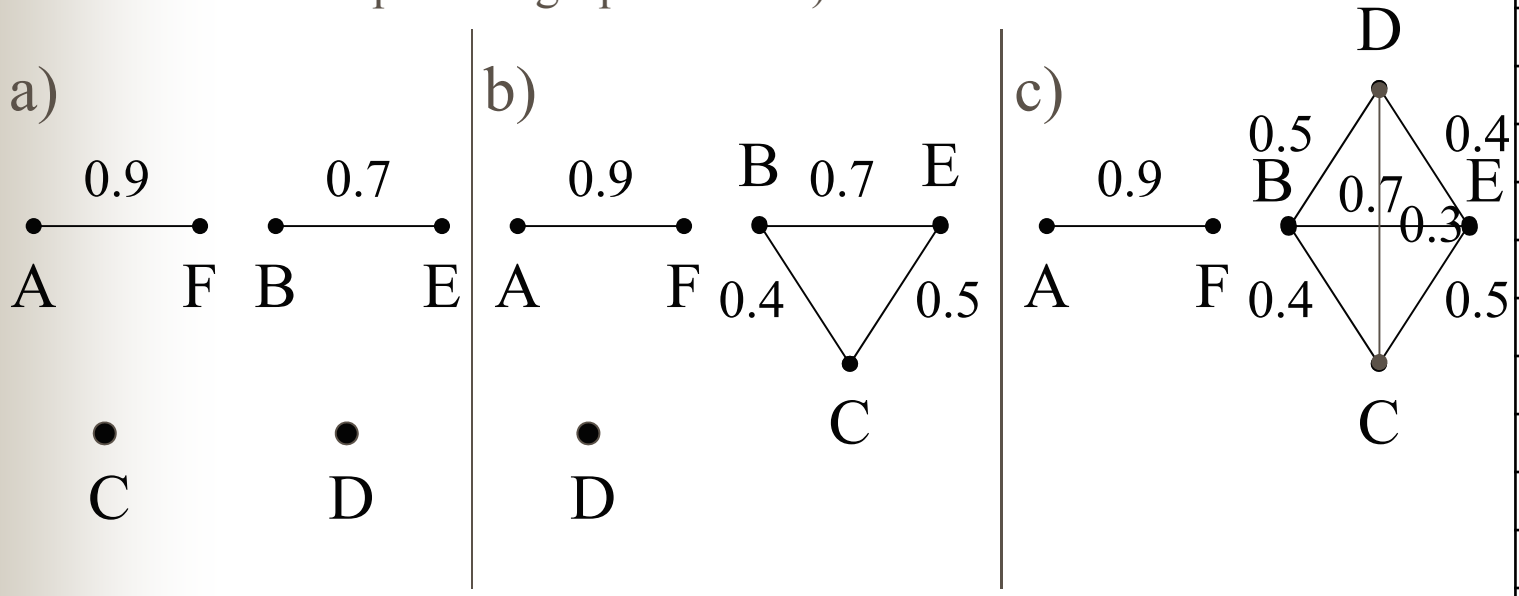


Hierarchical Agglomerative Clustering (HAC)
Ejemplo Enlace Completo (cont.)



Hierarchical Agglomerative Clustering (HAC)
Ejemplo Enlace Completo (cont.)

- Se tienen tres posibles estructuras de agrupamiento de enlace completo para los datos de la tabla:
  - 0.7 para el agrupamiento a)
  - 0.4 para el agrupamiento b)
  - 0.3 para el agrupamiento c)



Pareja	Similitud
AF	0,9
AE	0,8
BF	0,8
BE	0,7
AD	0,6
AC	0,5
BD	0,5
CE	0,5
BC	0,4
DE	0,4
AB	0,3
CD	0,3
EF	0,3
CF	0,2
DF	0,1

Hierarchical Agglomerative Clustering (HAC) Ejemplo Enlace Completo (cont.)

- El método de enlace completo lleva a una estructura muy diferente a la de enlace simple. Tiende a producir:
  - Un número mayor de grupos más pequeños.
  - Densamente enlazados.
- Cada elemento de un grupo de enlace completo garantiza su parecido con todos los demás elementos de su grupo al nivel de similitud establecido, esto lo hace mejor para la recuperación que el método de enlace simple (en el que las similitudes entre elementos del mismo grupo pueden ser muy bajas).
- Desafortunadamente el enlace completo es mucho más costoso que el enlace simple.

Hierarchical Agglomerative Clustering (HAC) Ejemplo (cont.)

- En el proceso de agrupamiento por enlace simple basta con recordar la lista de elementos agrupados previamente. El proceso puede pararse cuando todos los elementos estén incluidos de alguna forma en la estructura de grupos.
- En un sistema de enlace completo los pasos del agrupamiento dependen de las similitudes entre cada elemento y cada uno de los elementos del grupo. Es necesario recordar la lista de todos los pares de elementos considerados previamente en el proceso de agrupamiento.

Método Jerárquico *Hierarchical Agglomerative Clustering* (HAC) – Complejidad Computacional (cont.)

- En la primera iteración, todos los métodos de HAC necesitan encontrar la similaridad con todos los pares de *n* instancias, por lo que tiene una duración de $O(n^2)$.
- En cada una de las *n-2* iteraciones de la segunda iteración, es encontrar la distancia entre el más reciente *cluster* creado y todos los otros *clusters* existentes.
- Para mantener un funcionamiento total de $O(n^2)$, la similaridad de cada *cluster* a otro se debe hacer en tiempo constante.

Método Jerárquico
Hierarchical Agglomerative Clustering (HAC) (cont.)

- Ventajas:
  - Calidad del agrupamiento generado.
  - Tiempo de ejecución en colecciones pequeñas.
  - Es determinista, con la misma colección da los mismos resultados para diferentes ejecuciones.
- Desventajas:
  - En general, tiene un orden de duración más grande que *K-Means*.
  - Gasto de espacio en memoria considerable.

Método Híbrido – Algoritmos Híbridos
Algoritmo Buckshot

- Este algoritmo combina los algoritmos de *clustering K-Means* y *HAC*.
- Pseudocódigo:
  - Tomar aleatoriamente una muestra de instancias de tamaño $\sqrt{kN}$, donde k es el número de *clusters* y N es el número de documentos.
  - Buscar k centros con la muestra usando el Enlace de Promedio de Grupo de *HAC*.
  - Asignar cada documento a un *cluster*, usando la distancia más cercana (aplica *K-Means*).
  - Repetir la asignación una o dos veces, ya que los centros de los *clusters* pueden cambiar de puesto.
- El tiempo total del es O(n) y evita problemas de mala selección de las semillas.

Método Híbrido – Algoritmos Híbridos
Algoritmo del Fraccionamiento

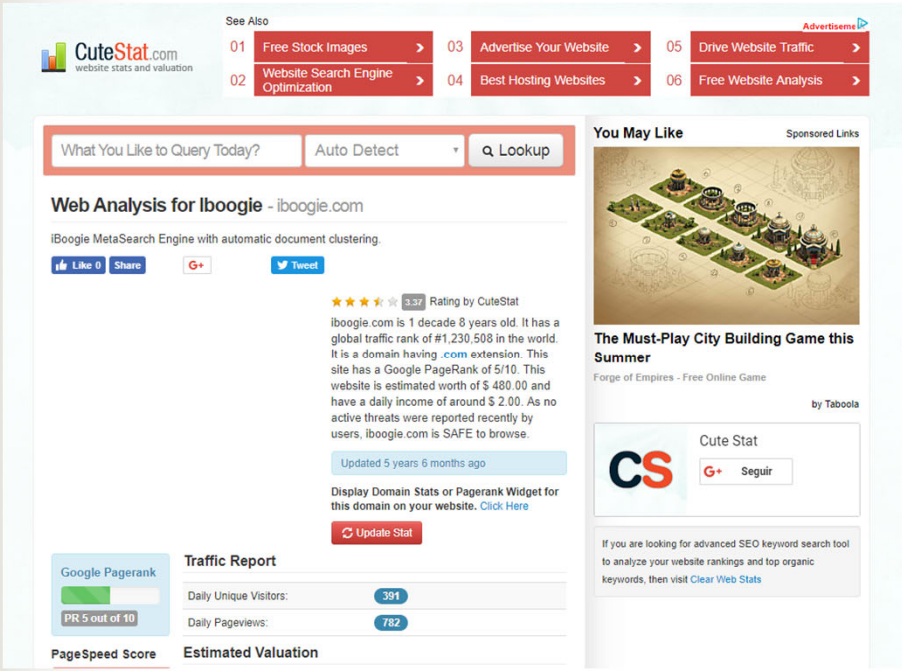
- Este algoritmo divide el conjunto de documentos en N/m grupos de tamaño fijo m, con $m > k$, y combina los algoritmos de *clustering K-Means* y *HAC*, u otro.
- Pseudocódigo:
  - Dividir el conjunto de documentos en N/m *clusters* de tamaño fijo m, con $m > k$, donde k es el número de *clusters* y N es el número de documentos.
  - Agrupar los datos en cada *cluster* usando algún algoritmo de *clustering* (Enlace de Promedio de Grupo de *HAC*). Tratar a estos *clusters* de N/m como individuos, y repita, hasta solamente tener k grupos.
  - Asignar cada documento a uno de los k *clusters*, usando la distancia más cercana (aplica *K-Means*).
  - Repetir la asignación una o dos veces, ya que los centros de los *clusters* pueden cambiar de puesto.
- El tiempo total del es $O(n^2)$ y, evita problemas de mala selección de las semillas y la elección aleatoria de las semillas.

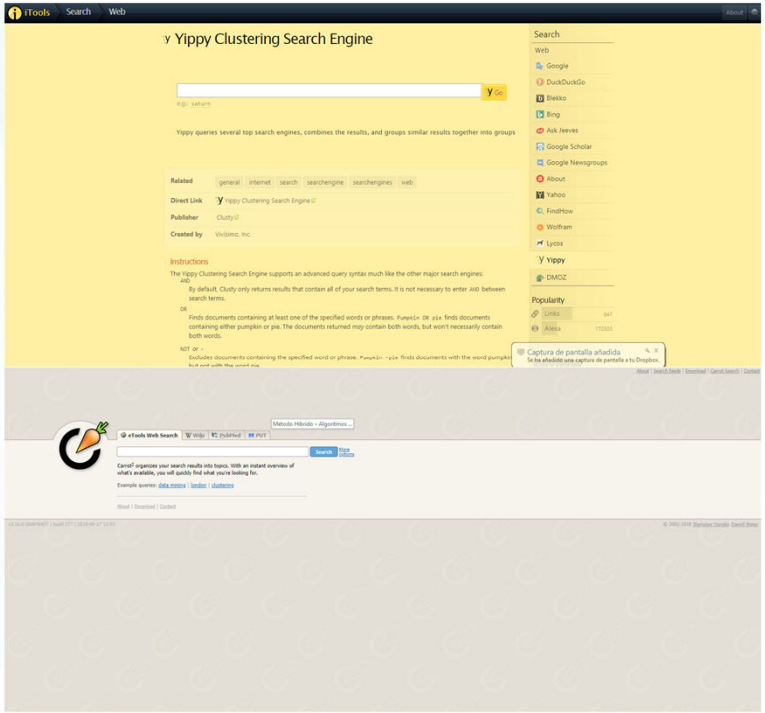
Método Híbrido – Algoritmos Híbridos
Buckshot vs. Fraccionamiento

- El Fraccionamiento dura más que Buckshot, aunque la complejidad asintótica es igual, pero no tiene la aleatoriedad del Buckshot.
- Buckshot se puede funcionar varias veces, con la esperanza de conseguir mejores resultados.
- Según el corte, el Fraccionamiento parece hacer mejores *clusters*.
- La estratificación contra la elección aleatoria es quizás aún más importante para los documentos que para los *widgets*.

Buscadores con *Clustering*

- https://iboogie.com.cutestat.com/
- http://itools.com/tool/yippy-web-search
- http://search.carrot2.org/stable/search





Conclusiones

- El *clustering* es un método de recuperar la información a través del agrupamiento de documentos en base a temas específicos.
- El uso del *clustering* facilitaría al usuario localizar rápidamente la información esperada.
  - Una secuencia ordenada de palabras es mucho más significativas que simples palabras claves.
- Existe el problema de poder nombre a los grupos hechos, para solucionar este problema se necesita de la sumarización.
- El algoritmo *HAC* produce mejores resultados que el algoritmo *K-means*. Sin embargo, estos resultados dependen de los siguientes factores:
  - Colecciones de prueba utilizadas.
  - Criterio de evaluación escogido.
  - Función de distancia escogida.
  - Implementación particular de cada algoritmo.
- Tanto *HAC* como *K-Means* son familias de algoritmos más que algoritmos específicos.

Referencias Bibliográficas

- La información fue tomada de:
  - Libro de texto del curso.
  - http://www.gedlc.ulpgc.es/docencia/seminarios/rit/.
  - http://www.elet.polimi.it/upload/matteucc/Clustering/tutorial_html/index.html.
  - http://www.cs.umbc.edu/~nicholas/clustering.
  - http://www.dcs.gla.ac.uk/%7Eiain/keith/.
  - Presentación de Edward Chang. UC Santa Barbara, 2004.