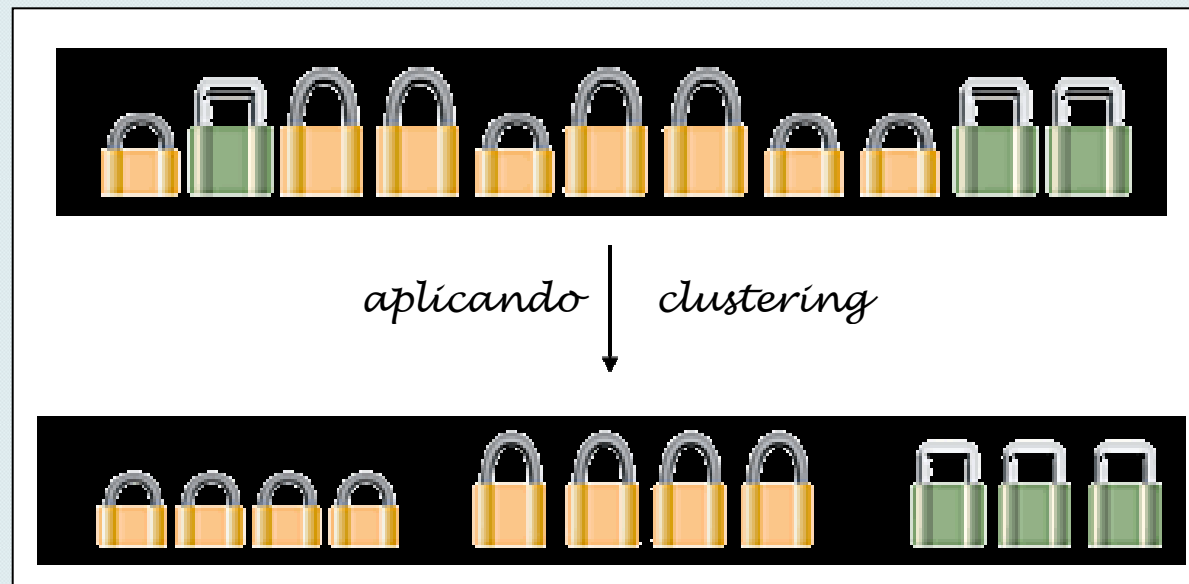


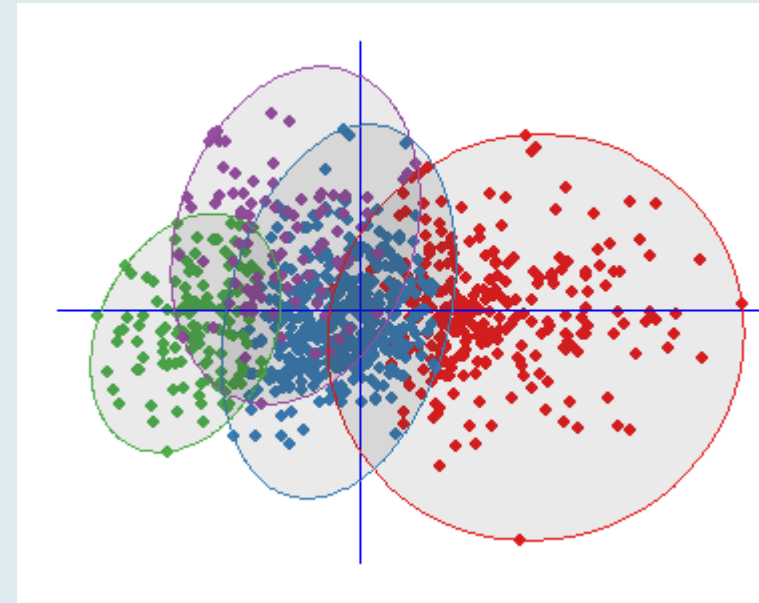
# Clustering

- En muchas aplicaciones la muestra no proviene de un solo grupo, sino que puede provenir de varios grupos.



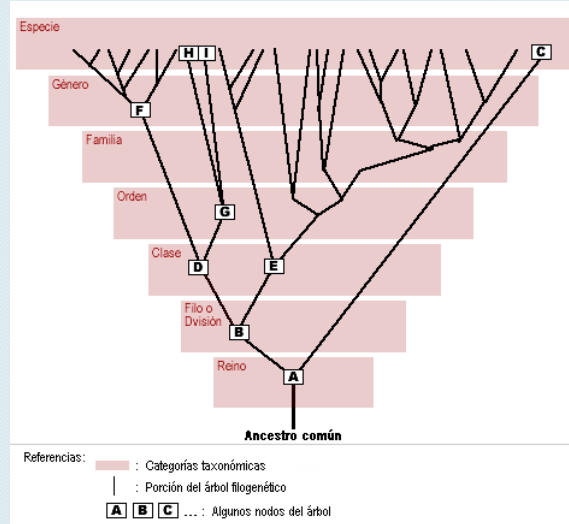
- El análisis de clusters es una técnica para resolver problemas de clasificación.

- Su objetivo consiste en **ordenar objetos en grupos** (conglomerados o clusters) de forma que el **grado de asociación/similitud** entre miembros del mismo cluster sea más fuerte que el grado de asociación/similitud entre miembros de diferentes clusters.



- Cada **cluster** se describe como la clase a la que sus miembros pertenecen.

- Algunos ejemplos:



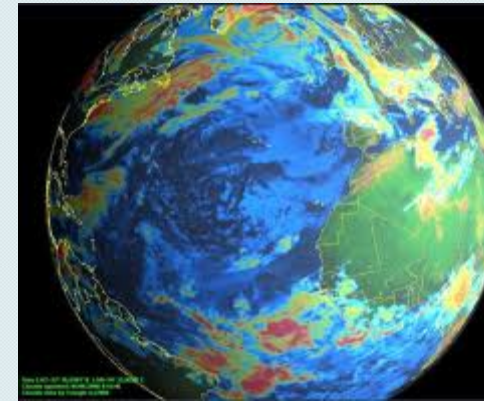
## Biología:

Los biólogos usan la taxonomía para clasificar a las cosas vivientes: reino, división, clase, orden, familia, género y especie

## Clima:

Para entender el clima es necesario encontrar patrones en la atmósfera y los océanos.

Por ejemplo, el análisis de cluster puede aplicarse para encontrar patrones en la presión atmosférica de las regiones polares y las áreas del océano que tienen impacto significativo sobre el clima los continentes

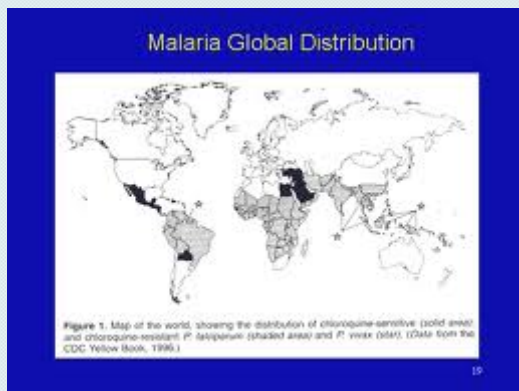


## Recuperación de información

En la Web hay millones de páginas y al hacer una consulta se pueden obtener miles de páginas.

Se puede utilizar clustering para agrupar los resultados de la búsqueda en un pequeño número de clusters, cada uno de los cuales captura un aspecto particular de la consulta.

Por ejemplo una consulta sobre películas puede devolver páginas agrupadas por categorías tales como reseñas, trailers, protagonistas, género.



## Psicología y medicina

Una enfermedad frecuentemente tiene un número de variaciones, y el análisis de cluster puede usarse identificar estas distintas subcategorías. Por ejemplo, para detectar patrones en la distribución en el espacio o el tiempo de una enfermedad

## Economía:

En marketing se recolecta gran cantidad de información acerca de los actuales y de los potenciales clientes.

Se puede usar clustering para segmentar a los clientes pequeños grupos para un análisis adicional.





# Distintos tipos de clusters

## Clusters bien separados

- Un cluster es un conjunto de objetos en el cual cada objeto es más **cercano** (o más similar) a todos los objetos del cluster que a cualquier otro objeto que no esté en el cluster
- Algunas veces se utiliza un **umbral** para especificar que todos los objetos en el cluster deben estar suficientemente cerca (o ser similares) a cualquier otro.
- Esta definición “idealista” de los clusters se satisface sólo cuando los datos contienen clusters “naturales” que están bastante lejos unos de otros.



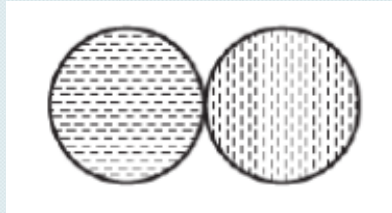
- En la figura se da un ejemplo de clusters bien separados que consiste en dos grupos de puntos en un espacio bidimensional.
- La **distancia** entre cualquier par de puntos de distintos grupos es mayor que la distancia entre cualquier par de puntos dentro del mismo grupo.
- Los clusters bien separados no necesariamente son globulares, pueden tener cualquier forma.

## Clusters basados en prototipos

- Un cluster es un conjunto de objetos en el cual cada objeto está más cerca (o es más similar) al **prototipo** que define al cluster que al prototipo que define cualquier otro cluster.
- Para datos con atributos **continuos**, el **prototipo** de un cluster es usualmente el **centroide**, es decir la media de todos los puntos del cluster.
- Cuando el centroide no es significativo, por ejemplo en el caso de datos con atributos **categoricos**, el prototipo es frecuentemente un **mediode**, es decir el punto más representativo del cluster.



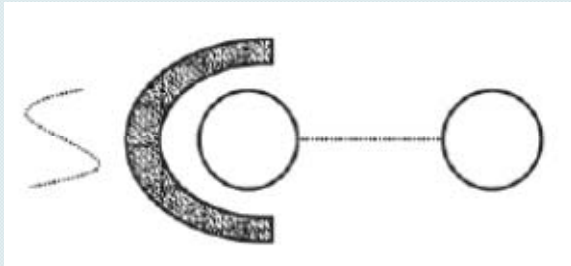
- Para datos de distinto tipo, el prototipo puede verse como el punto más central.



- Generalmente tal tipo de clusters son globulares.  
En la figura se da un ejemplo de clusters basados en el centro.

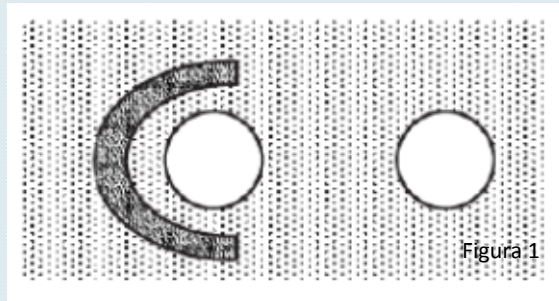
## Clusters basados en grafos

- Si los datos están representados como un **grafo**, donde los **nodos** son los **objetos** y las **ramas** representan las **conexiones** entre objetos, entonces un **cluster** puede definirse como una **componente conexa**, es decir grupo de objetos que están conectados entre sí, pero no tienen conexión con otros objetos fuera del grupo.
- Un ejemplo importante de este tipo de cluster son los **clusters basados en contigüidad**, en los cuales dos de los objetos están conectados sólo si están a una distancia específica uno de otro.

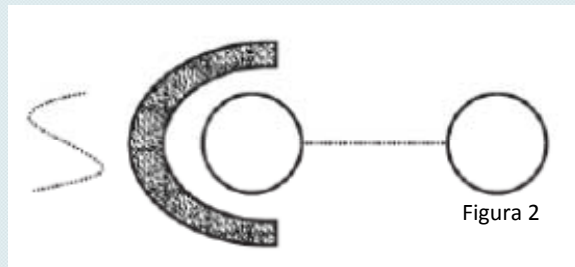


- En la figura se muestra un ejemplo de tales clusters para el caso bidimensional.
- Esta definición de cluster es útil cuando los cluster son **irregulares** o **entrecruzados**, pero pueden tener problemas cuando aparece ruido, como en el ejemplo de la figura donde un pequeño puente de datos pueden mezclar dos clases distintas.

## Clusters basados en densidad



- Un cluster es una región densa de objetos que puede estar rodeada por otra región de baja densidad.



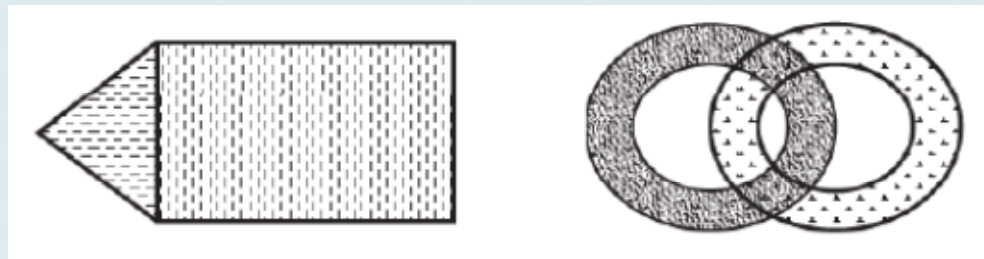
- En la figura 1 se muestra algunos clusters basados en densidad para datos artificiales a los cuales se le agregó ruido a los de la figura 2.

- Los dos clusters circulares no están mezclados porque el puente entre ellos se desvaneció en el ruido.

- Además la curva en forma de “S” también se desvaneció el ruido.
- Este tipo de definición de cluster se utiliza cuando los cluster son irregulares o entrecruzados y presentan ruidos.
- Los cluster basados en contigüidad no funcionarían bien en el caso de la figura 1, porque ruido tiene formar puentes entre los cluster.

## Clusters conceptuales

- Más generalmente, se puede definir un **cluster** como un **conjunto de objetos que comparten una propiedad**.
- Esta definición engloba a todas las definiciones anteriores de cluster y además incluye nuevos tipos de cluster.
- Consideremos los cluster en la figura. Un área triangular (cluster) es adyacente a otra rectangular, y hay otros dos círculos entrecruzados (clusters).

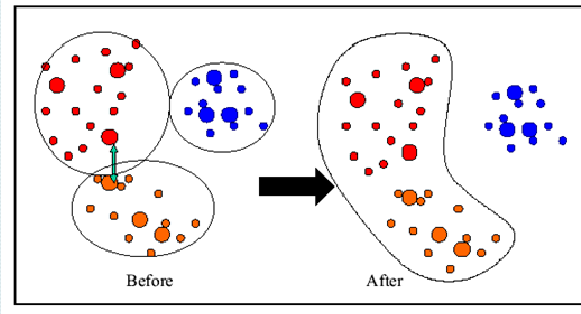




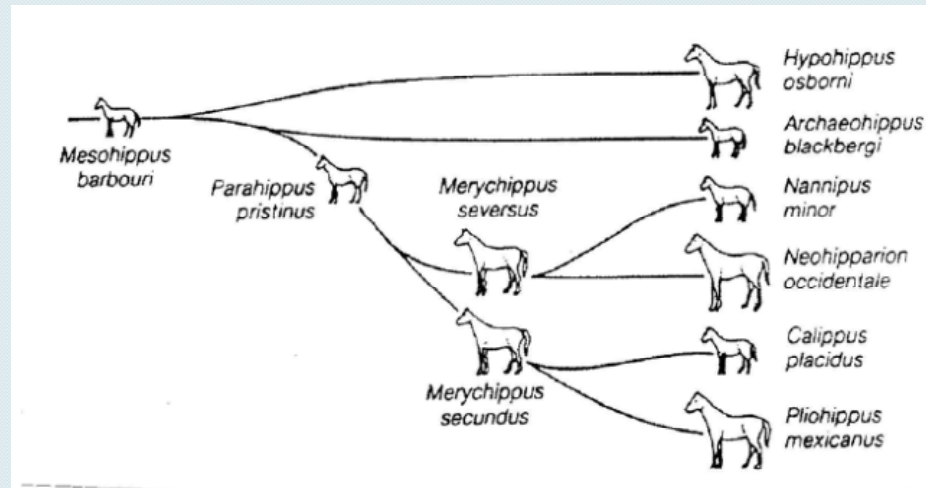
- En ambos casos, el algoritmo de clustering necesitaría un concepto muy específico de cluster para detectar exitosamente estos clusters.
- El proceso para encontrar estos clusters se denomina clustering conceptual
  - Algunos ejemplos este tipo de cluster aparecen en el reconocimiento de imágenes.

## Otra clasificación de los métodos de clustering

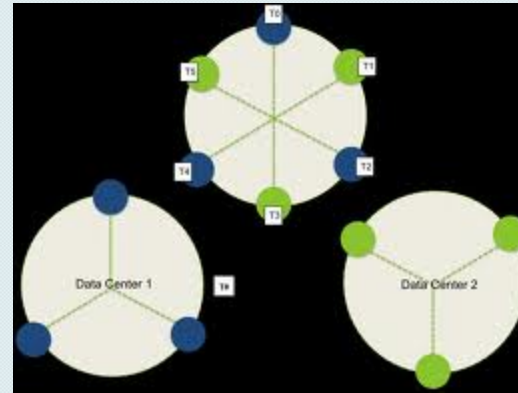
- Basados en distancias



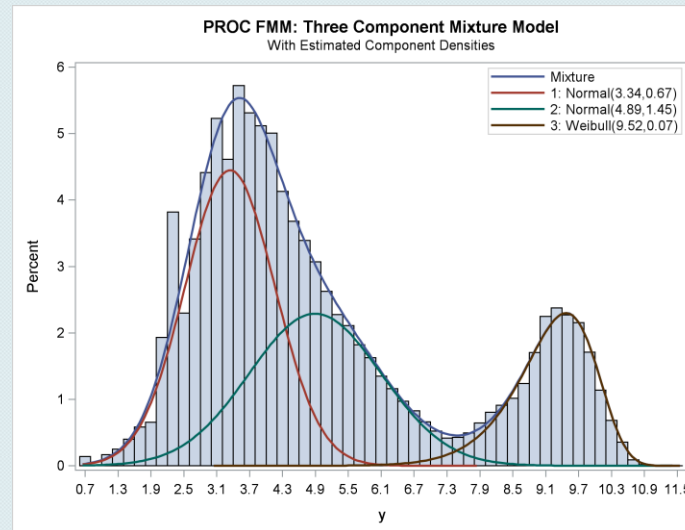
- Jerárquicos



- Basados en particiones



- Probabilísticos



## Mezcla de densidades

- La mezcla de densidades puede escribirse como:

$$p(\mathbf{x}) = \sum_{i=1}^k p(\mathbf{x}|\mathcal{G}_i)p(\mathcal{G}_i)$$

donde:

- $\mathcal{G}_i$  son las **componentes** de la mezcla, que también se denominan *grupos* o *clusters*
  - $p(\mathbf{x}|\mathcal{G}_i)$  son las **densidades** de las componentes
  - $p(\mathcal{G}_i)$  son las **proporciones** de la mezcla.
- El **número** de componentes  $k$  es un parámetro que debe ser especificado de antemano.

- Dada una muestra y  $k$ , el aprendizaje consiste en estimar las densidades y las proporciones de las componentes.
- Vamos a suponer que las densidades siguen un modelo paramétrico, por lo cual es necesario sólo **estimar** los **parámetros**.
- Si las densidades de las componentes son normales multivariadas, tenemos que  $p(\mathbf{x}|\mathcal{G}_i) \sim \mathcal{N}(\boldsymbol{\mu}_i, \Sigma_i)$  y los parámetros que deben estimarse a partir de la muestra aleatoria  $\mathcal{X}$  son  $\Phi = \{p(\mathcal{G}_i), \boldsymbol{\mu}_i, \Sigma_i\}_{i=1}^k$

- La **clasificación paramétrica** es un tipo de modelo con mezcla, donde:
  - los **grupos**  $\mathcal{G}_i$  corresponden a las **clases**  $\mathcal{C}_i$ ,
  - las densidades de las componentes  $p(\mathbf{x}|\mathcal{G}_i)$  corresponden a las **densidades** de las **clases**  $p(\mathbf{x}|\mathcal{C}_i)$
  - $p(\mathcal{G}_i)$  corresponden a las **probabilidades a priori** de las **clases**  $p(\mathcal{C}_i)$

$$p(\mathbf{x}) = \sum_{i=1}^k p(\mathbf{x}|\mathcal{G}_i)p(\mathcal{G}_i)$$



- En este *caso supervisado*, sabemos *cuántos grupos hay* y el aprendizaje de los parámetros es trivial, porque están dados los rótulos, es decir qué instancias pertenecen a cada clase (componente).
- Habíamos visto que dada la muestra  $\mathcal{X} = \{\mathbf{x}^j, \mathbf{r}^j\}_{j=1}^n$ , donde  $r_i^j = \begin{cases} 1 & \text{si } \mathbf{x}^j \in C_i \\ 0 & \text{sino} \end{cases}$ , suponiendo que cada clase tiene distribución gaussiana, los estimadores de máxima verosimilitud separados por cada clase son:

$$\hat{P}(C_i) = \frac{1}{n} \sum_{j=1}^n r_i^j$$

$$\hat{\mu}_i = \frac{\sum_{j=1}^n r_i^j x^j}{\sum_{j=1}^n r_i^j}$$

$$\hat{\Sigma}_i = \frac{\sum_{j=1}^n r_i^j (x^j - \hat{\mu}_i)(x^j - \hat{\mu}_i)^t}{\sum_{j=1}^n r_i^j}$$

- La diferencia ahora está en que la muestra es  $\mathcal{X} = \{x^j\}_{j=1}^n$ , y por lo tanto tenemos un problema de aprendizaje *no supervisado*.
- Ahora sólo conocemos  $x^j$  y no los rótulos  $r^j$ , es decir no conocemos cuál  $x^j$  proviene de cada componente.

- Por lo tanto debemos estimar ambas:
  - primero estimamos los rótulos  $r_i^j$ , la componente a la cual una instancia dada pertenece,
  - una vez que estimamos el rótulo, debemos estimar los parámetros de la componente dado el conjunto de instancias que pertenecen a ella.

## K-medias

- Supongamos que tenemos una imagen que se guarda con 24 bits/pixel y puede tener hasta 16 millones de colores y que tenemos una pantalla con 8 bits/pixel que puede usar sólo 256 colores.



- Queremos encontrar los mejores 256 colores entre los 16 millones de modo tal que la imagen representada sólo con esos 256 colores se parezca tanto como sea posible a la original. Esto es una cuantificación del color, donde se transforma de una alta resolución a una baja resolución.

- Supongamos que tenemos una muestra  $\mathcal{X} = \{\mathbf{x}^j\}_{j=1}^n$ .  
Tenemos  $k$  vectores de referencia  $\mathbf{m}_j$ , con  $j = 1, \dots, k$
- En nuestro ejemplo de la cuantificación del color,  $\mathbf{x}^j$  son los valores de los píxeles en 24 bits y  $\mathbf{m}_j$  son los colores de entrada de la función, con  $k = 256$ .
- Supongamos por ahora que conocemos de alguna manera los valores de  $\mathbf{m}_j$ .

Entonces para representar la imagen, dado el pixel  $\mathbf{x}^j$ , se lo representa con el valor más similar  $\mathbf{m}_i$  en la función de color de modo que:

$$\|\mathbf{x}^j - \mathbf{m}_i\| = \min_{1 \leq l \leq k} \|\mathbf{x}^j - \mathbf{m}_l\|$$

- Veamos cómo calcular  $\mathbf{m}_i$ : al representar  $\mathbf{x}^j$  con  $\mathbf{m}_i$ , hay un error proporcional a la distancia  $\|\mathbf{x}^j - \mathbf{m}_i\|$ .

Para que la nueva imagen luzca parecida a la original, se quiere que estas distancias sean lo más pequeñas posibles para todos los píxeles.

El error está dado por:

$$E(\{\mathbf{m}_i\}_{i=1}^k | \mathcal{X}) = \sum_{j=1}^n \sum_{i=1}^k b_i^j \|\mathbf{x}^j - \mathbf{m}_i\|^2 \quad (1)$$

donde

$$b_i^j = \begin{cases} 1 & \text{si } \|\mathbf{x}^j - \mathbf{m}_i\| = \min_{1 \leq l \leq k} \|\mathbf{x}^j - \mathbf{m}_l\| \\ 0 & \text{en otro caso} \end{cases} \quad (2)$$

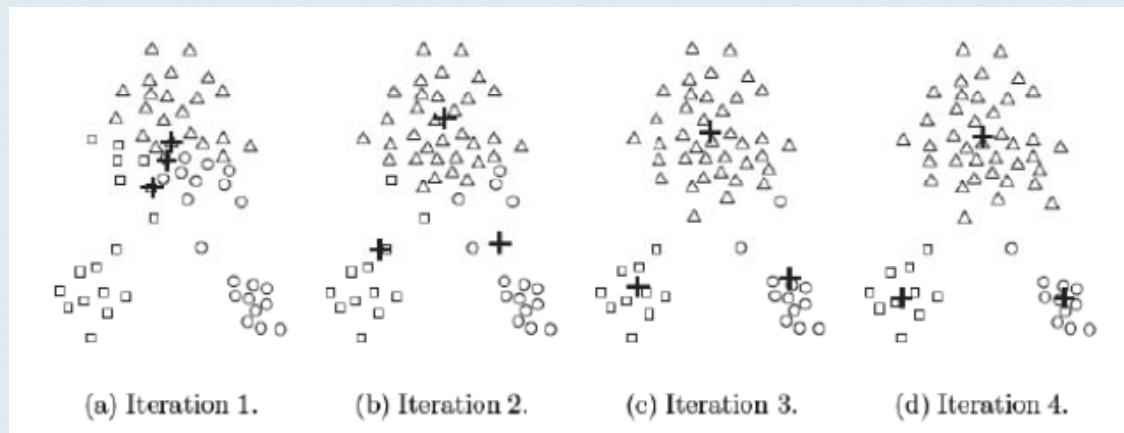
- Como los  $b_i^j$  dependen de  $\mathbf{m}_i$  este problema de optimización no se puede resolver analíticamente.



- Vamos a aplicar un proceso iterativo llamado k-medias.
  - Primero comenzamos con valores  $\mathbf{m}_i$  Inicializados.
  - Luego en cada iteración utilizamos la ecuación (2) y calculamos los  $b_i^j$  para todos los  $\mathbf{x}^j$ , que son los rótulos estimados; si  $b_i^j = 1$  diremos que  $\mathbf{x}^j$  pertenece al grupo de  $\mathbf{m}_i$ .
  - Entonces, una vez obtenidos los rótulos, se minimiza la ecuación (1).
  - Derivando dicho ecuación con respecto a  $\mathbf{m}_i$  e igualando a cero, se obtiene

$$\mathbf{m}_i = \frac{\sum_{j=1}^n b_i^j \mathbf{x}^j}{\sum_{j=1}^n b_i^j}$$

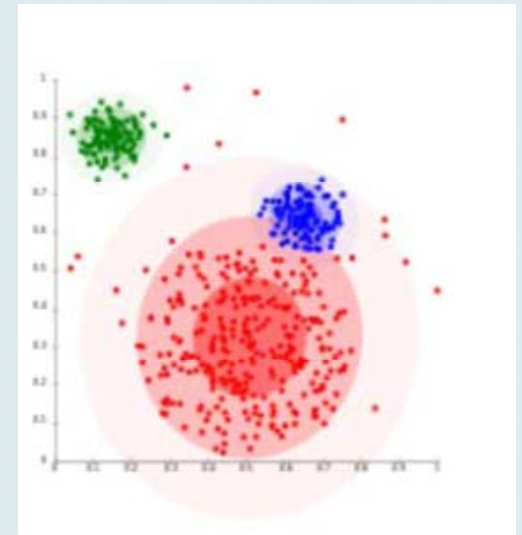
- Este es un procedimiento iterativo porque una vez que calculamos el nuevo  $\mathbf{m}_i$ ,  $b_i^j$  cambia y debe ser calculado, lo cual afecta a  $\mathbf{m}_i$ .
- Estos dos pasos se repiten hasta que se estabilice  $\mathbf{m}_i$  como se ve en la figura.



- Este método es una aplicación del algoritmo EM.

- Si los datos tienen distribución gaussiana, este método funciona bien

- Se puede aplicar sólo cuando están definidas las medias



- Se necesita especificar por adelantado el número  $k$  de clusters
- Es muy inestable en presencia de ruido y de datos anómalos
- No es apropiado para aplicar en el caso de que los clusters no tengan forma convexa

## Algoritmo K-medias

1. Inicializar  $\mathbf{m}_i$  para  $i = 1, \dots, k$ , por ejemplo con los valores de  $k$   $\mathbf{x}^j$  elegidas al azar
2. Repetir hasta que las  $\mathbf{m}_i$  converjan:

- Para todo  $\mathbf{x}^j \in \mathcal{X}$

$$b_i^j \leftarrow \begin{cases} 1 & \text{si } \|\mathbf{x}^j - \mathbf{m}_i\| = \min_{1 \leq l \leq k} \|\mathbf{x}^j - \mathbf{m}_l\| \\ 0 & \text{en otro caso} \end{cases}$$

- Para todo  $\mathbf{m}_i$ ,  $i = 1, \dots, k$

$$\mathbf{m}_i \leftarrow \frac{\sum_{j=1}^n b_i^j \mathbf{x}^j}{\sum_{j=1}^n b_i^j}$$

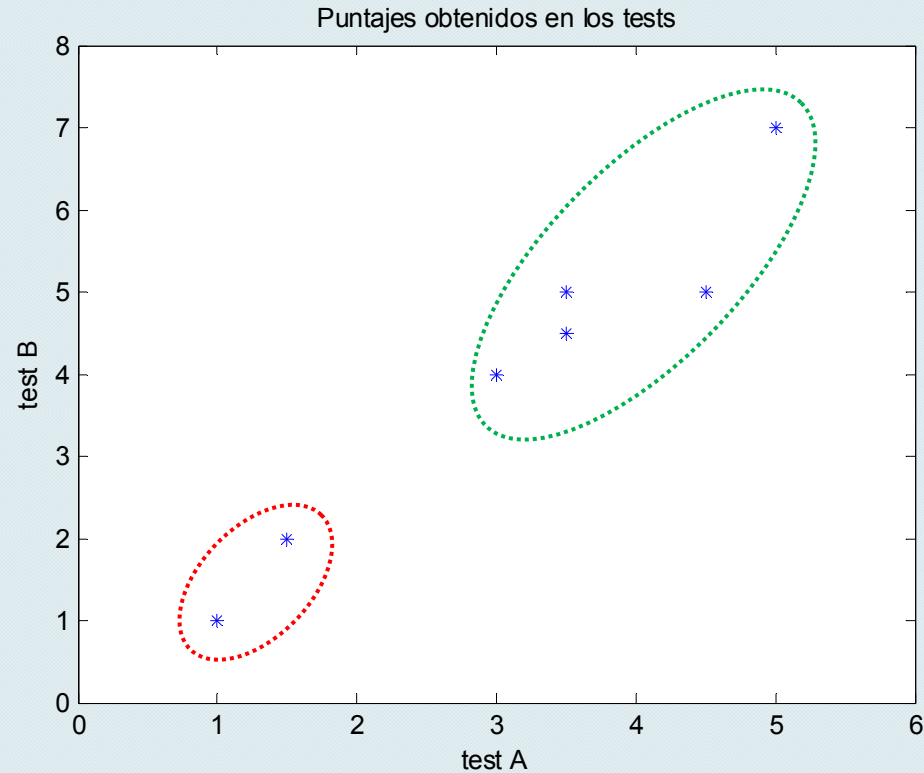
- Una de las desventajas de este método es que es un procedimiento de búsqueda local, y los valores finales de  $m_i$  dependen en gran medida los valores iniciales de  $m_i$ .
- Hay varios métodos para elegir los valores iniciales de  $m_i$ :
  - elegir aleatoriamente  $k$  instancias
  - calcular la media de todos los datos
  - calcular la componente principal, dividir su rangos en  $k$  intervalos iguales, particionar los datos en  $k$  grupos, y tomar las medias de grupos como valores iniciales.

## Ejemplo

- Consideremos los siguientes datos que consisten en las puntuaciones obtenidas en 2 tests de 7 individuos.

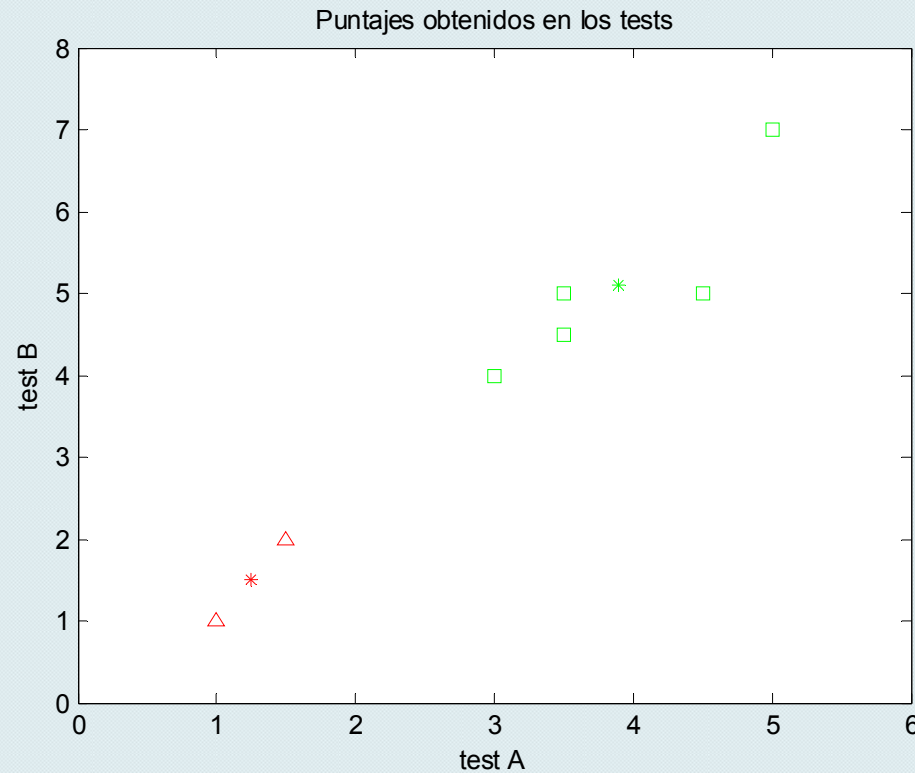
Sujeto	A	B
1	1.0	1.0
2	1.5	2.0
3	3.0	4.0
4	5.0	7.0
5	3.5	5.0
6	4.5	5.0
7	3.5	4.5





- Graficando los datos se observa que se podrían separar en 2 grupos: los que obtuvieron puntajes “**bajos**” y los que obtuvieron puntajes “**altos**”

- Aplicando el método de las k-medias se obtiene



Centroides:

- grupo 1: (3.9,5.1)
- grupo 2: (1.25,1.5)

	distancias a la media del grupo	
Sujeto	1	2
1	25.22	0.3125
2	15.37	0.3125
3	2.02	9.3125
4	4.82	44.3125
5	0.17	17.3125
6	0.37	22.8125
7	0.52	14.0625

## ¿Cómo elegir el número de clusters?

- Una regla usual es hacer

$$k \approx \sqrt{\frac{n}{2}}$$

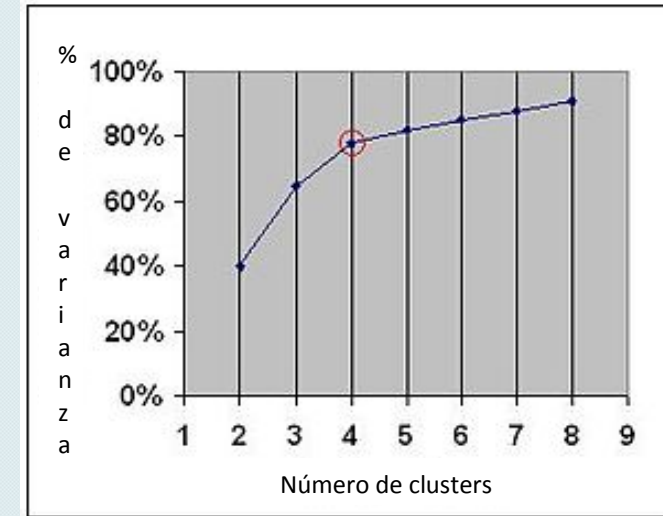
donde  $n$  es el número de datos

- En algunos casos, la validación de los grupos puede hacerse manualmente controlando si los clusters realmente separan grupos significativos de los datos.  
Por ejemplo en la cuantificación de color, se puede inspeccionar la imagen para controlar su calidad.

- Hacer un plot de los datos en dos dimensiones usando componentes principales puede ser útil para descubrir la estructura de los datos y el número de clusters.
- También puede ayudar una aproximación incremental: probar diferentes posibilidades, incrementando el valor de  $k$  y ver cuál es la mejor, esto es ver cuál minimiza el cuadrado de la distancia de todos los puntos a sus centros de clases.

- Método del “codo” (elbow):

- Se puede graficar el porcentaje de varianza explicado por los clusters (cociente entre la varianza entre los grupos y la varianza total, test  $\mathcal{F}$ ) vs. el número de clusters.



- Los primeros clusters dan mucha información (explican un alto porcentaje de varianza), pero en algún punto, el aumento del porcentaje comienza a decrecer, produciendo un ángulo en el gráfico.

El número de clusters se elige en ese punto, el “codo”

- En el ejemplo de la figura, el “codo” se produce en 4, y por lo tanto se hace  $k = 4$ .

## Clusters jerárquicos

- Existen métodos para clustering en los cuales sólo se usan las similitudes de las instancias, sin ningún otro requerimiento sobre los datos.
- El objetivo es encontrar grupos tales que las instancias en un grupo son más similares entre sí que las instancias en diferentes grupos. Esta aproximación se denomina clusters jerárquicos.

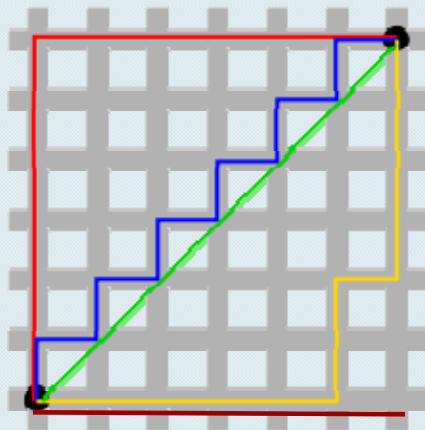


- Estos métodos se pueden clasificar en dos grupos:
- **Aglomerativos:**  
comienza con  $n$  grupos, cada uno conteniendo una instancia de entrenamiento, y va uniendo grupos similares para formar grupos más grandes, hasta que quede un solo grupo
- **Divisivos:**  
comienza con un solo grupo y va dividiendo los grupos grandes en pequeños hasta que cada grupo contenga una sola instancia.

- Para decir cuáles cluster deben **combinarse** (para los métodos aglomerativos) o cuál cluster debe **separarse** (para los métodos divisivos) es necesaria una **medida de similitud** entre los conjuntos de observaciones.
- Esto se consigue usando una **métrica** apropiada, es decir una medida de la distancia entre pares de observaciones, y un **criterio de unión** que especifica los conjuntos similares como función de las distancias entre pares de observaciones en los conjuntos.

## Métrica

- La elección de la métrica influirá en la forma de los clusters.
- Además algunos pares de elementos puede estar cercanos de acuerdo a una distancia pero lejos usando otra.



Por ejemplo, la distancia entre  $(1,0)$  y  $(0,0)$  es generalmente 1 usando las distancias habituales, pero la distancia entre  $(1,1)$  y  $(0,0)$  pueden ser 2,  $\sqrt{2}$  ó 1 dependiendo si se usan la distancia de **Manhattan**, la **euclídea** o la **máxima distancia**

- Formalmente, una **distancia** o **métrica** es una función

$$d: \mathbb{R}^n \times \mathbb{R}^n \rightarrow \mathbb{R}$$

que verifica las siguientes propiedades:

- $d(\mathbf{a}, \mathbf{b}) \geq 0 \quad \forall \mathbf{a}, \mathbf{b} \in \mathbb{R}^n$

- $d(\mathbf{a}, \mathbf{b}) = d(\mathbf{b}, \mathbf{a}) \quad \forall \mathbf{a}, \mathbf{b} \in \mathbb{R}^n$

- $d(\mathbf{a}, \mathbf{b}) \leq d(\mathbf{a}, \mathbf{c}) + d(\mathbf{c}, \mathbf{b}) \quad \forall \mathbf{a}, \mathbf{b}, \mathbf{c} \in \mathbb{R}^n$

- $d(\mathbf{a}, \mathbf{a}) = 0 \quad \forall \mathbf{a} \in \mathbb{R}^n$

- Si  $\mathbf{a} \in \mathbb{R}^n$  y  $\mathbf{b} \in \mathbb{R}^n$  son tales que  $d(\mathbf{a}, \mathbf{b}) = 0$ , entonces  $\mathbf{a} = \mathbf{b}$

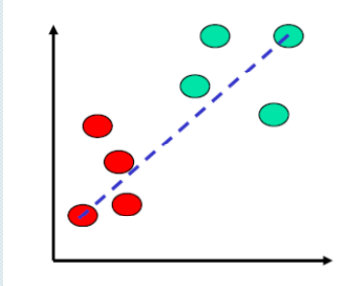
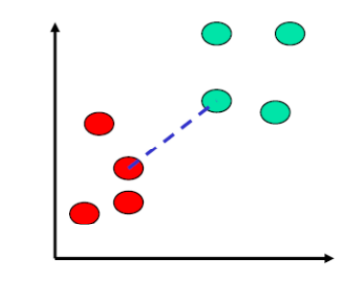
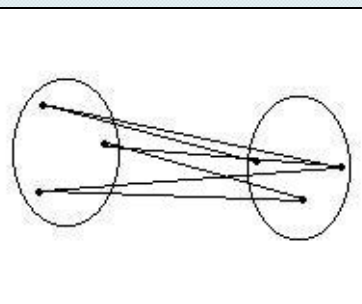
Algunas distancias usuales son:

distancia euclídea	$\ \mathbf{a} - \mathbf{b}\ _2 = \sqrt{\sum_{i=1}^n (a_i - b_i)^2}$
cuadrado de distancia euclídea	$\ \mathbf{a} - \mathbf{b}\ _2^2 = \sum_{i=1}^n (a_i - b_i)^2$
distancia de Manhattan	$\ \mathbf{a} - \mathbf{b}\ _1 = \sum_{i=1}^n  a_i - b_i $
distancia de Minkowski	$\ \mathbf{a} - \mathbf{b}\ _m = \left( \sum_{i=1}^n  a_i - b_i ^m \right)^{1/m}$ $m \geq 2$
distancia máxima	$\ \mathbf{a} - \mathbf{b}\ _\infty = \max_{1 \leq i \leq n}  a_i - b_i $
distancia promedio	$\frac{1}{n} \sum_{i=1}^n (a_i - b_i)^2$

## Criterio de enlace

- El criterio de enlace determina la distancia entre los conjuntos de observaciones como una función de la distancia entre pares de observaciones.

- Algunos criterios de enlace entre dos conjuntos de observaciones  $A$  y  $B$  son:

máximo o completo (complete)	$\max \{d(\mathbf{a}, \mathbf{b}): \mathbf{a} \in A, \mathbf{b} \in B\}$	
mínimo o individual (single)	$\min \{d(\mathbf{a}, \mathbf{b}): \mathbf{a} \in A, \mathbf{b} \in B\}$	
promedio o centroide (average)	$\frac{1}{ A  B } \sum_{\mathbf{a} \in A} \sum_{\mathbf{b} \in B} d(\mathbf{a}, \mathbf{b})$	



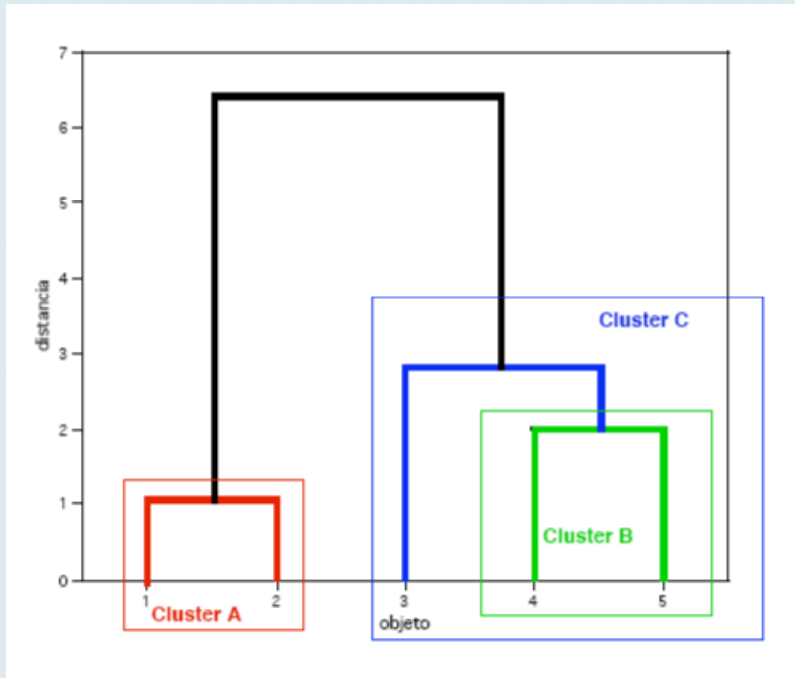
# *Algoritmo* clusters jerárquicos aglomerativos

Dado un conjunto de  $n$  datos:

1. **Asignar** cada dato a un cluster, de modo que se obtienen  $n$  clusters.
2. Calcular las **distancias** entre todos los clusters
3. Encontrar el par de clusters **más cercanos** (más similares) y **unirlos** en un único cluster, de modo que ahora se tiene un cluster menos que antes

Repetir los pasos 2 y 3 hasta obtener  $k$  clusters

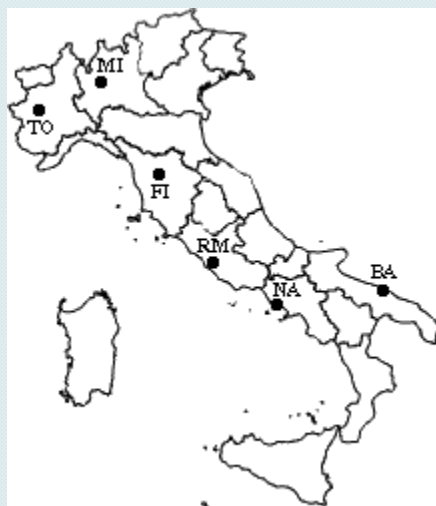
# Dendrograma



- Una vez que se aplica el método aglomerativo, el resultado se grafica en una estructura jerárquica llamada **dendrograma**.
- Es un árbol en el cual las **hojas** corresponden a las **instancias**, las cuales son agrupadas en el orden en que son unidas

## Ejemplo

- En la siguiente tabla se dan las distancias en kilómetros entre 6 ciudades italianas: Milán, Turín, Florencia, Roma, Nápoles, Bari.



	BA	FI	MI	NA	RM	TO
BA	0	662	877	255	412	996
FI	662	0	295	468	268	400
MI	877	295	0	754	564	138
NA	255	468	754	0	219	869
RM	412	268	564	219	0	669
TO	996	400	138	869	669	0

- El par de ciudades más cercanas es Milán y Turín, que están a **138** km. Estas ciudades se unen en un cluster "MI/TO".



- Luego se calcula la distancia de este nuevo cluster al resto de las ciudades.
- Si se usa el **enlace de mínima distancia** (single), se considera que la distancia del objeto compuesto a cualquier otro objeto es igual a la distancia más corta entre cualquier miembro del cluster al objeto fuera de él.

○ Por ejemplo, para calcular la distancia entre MI/TO y Roma, se obtiene:

$$\begin{array}{l} d(\text{MI}, \text{RM}) = 564 \\ d(\text{TO}, \text{RM}) = 669 \end{array} \Rightarrow d(\text{MI/TO}, \text{RM}) = 564$$

- Se obtiene la siguiente tabla de distancias:

	BA	FI	MI/TO	NA	RM
BA	0	662	877	255	412
FI	662	0	295	468	268
MI/TO	877	295	0	754	564
NA	255	468	754	0	219
RM	412	268	564	219	0

- Calculamos las distancias y se obtiene que

$$\min d(i, j) = d(\text{NA}, \text{RM}) = 219$$

- Y por lo tanto se unen Nápoles y Roma en un nuevo cluster llamado “NA/RM”



- La nueva tabla de distancias es:

	BA	FI	MI/TO	NA/RM
BA	0	662	877	255
FI	662	0	295	268
MI/TO	877	295	0	564
NA/RM	255	268	564	0

- Como

$$\min d(i, j) = d(\text{BA}, \text{NA/RM}) = 255$$

se unen Bari y NA/RM en un cluster llamado “BA/NA/RM”



- La nueva tabla de distancias es:

	BA/NA/RM	FI	MI/TO
BA/NA/RM	0	268	564
FI	268	0	295
MI/TO	564	295	0

- Como

$$\min d(i, j) = d(\text{FI}, \text{BA/NA/RM}) = 268$$

se unen Florencia y BA/NA/RM en un nuevo cluster llamado "FI/BA/NA/RM"





- La nueva tabla de distancias es:

	<b>BA/FI/NA/RM</b>	<b>MI/TO</b>
<b>BA/FI/NA/RM</b>	0	295
<b>MI/TO</b>	295	0

- Finalmente, se unen los dos clusters.

- El proceso se puede resumir en el siguiente árbol jerárquico:

