Tema 7: Clústering

Universidad
Francisco de Vitoria
UFV Madrid

Alberto Nogales

alberto.nogales@ufv.es

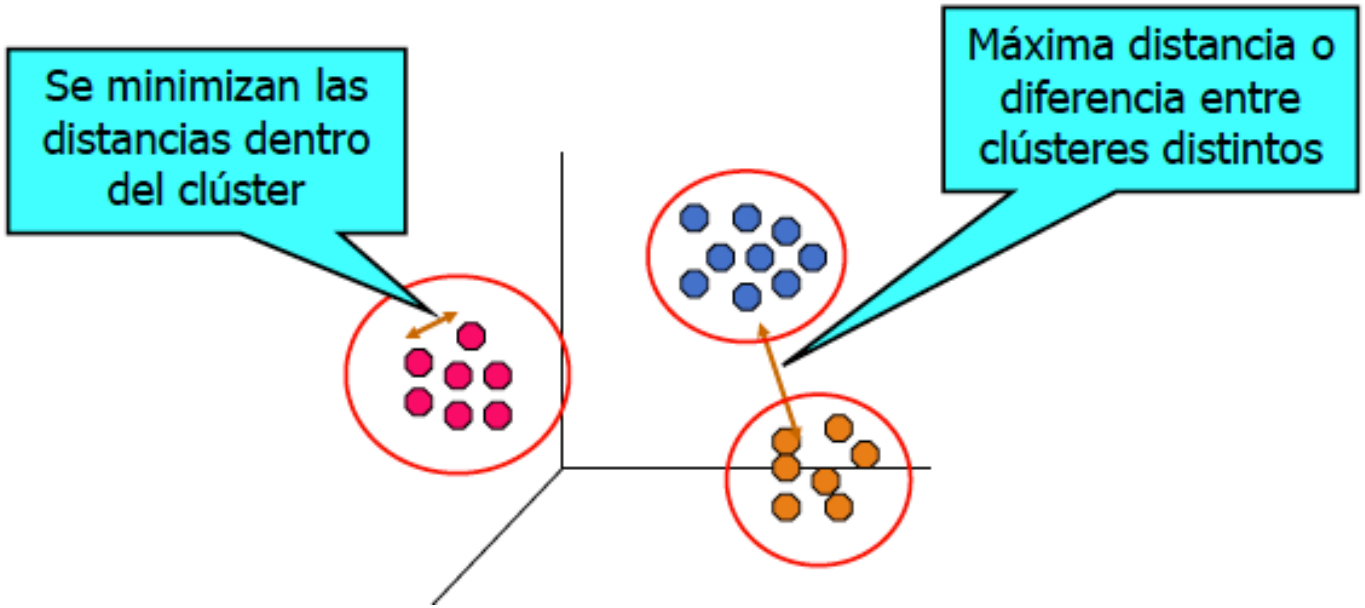
Curso 2020-2021

Índice

- Definición del problema.
- Algoritmos de clústering: k-means, clústering jerárquico aglomerado, DBSCAN.
- Medidas de distancia.
- Calculo de clústers.

Introducción

Objetivo: Encontrar agrupaciones de objetos que sean similares o que estén relacionados entre sí y sean diferentes o no estén relacionados con objetos de otras agrupaciones.



Introducción

1) **Minimizar distancias dentro del clúster:** la dispersión dentro de clústers sea pequeña respecto a la dispersión total.

2) **Maximizar la distancia entre clústers:** refleja el hecho que se esté tratando con el número “correcto” de clústers K.

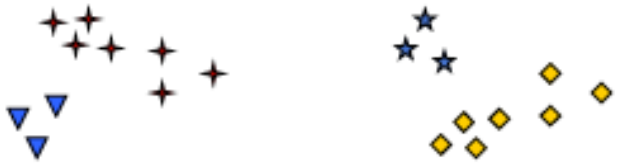
Introducción

Dado un conjunto de vectores X1, . . . , Xn queremos formar con ellos K grupos “homogéneos” que sean “distintos” entre ellos. El tamaño de los clústers es ambiguo.

¿Cuántos clústeres habría?

6 Clústeres

2 Clústers



4 Clústeres

Introducción

Es parecido a los problemas de clasificación pero sin un conjunto de entrenamiento que permita guiar el cómo se generan los clústers. No se clasifica, por lo tanto no están etiquetados.

Utilidad:

1) Comprensión: Agrupar información relacionada entre sí con funciones o características similares.

2) Preprocesamiento: Encontrar outliers o usar el tamaño óptimo de los conjuntos de datos (reducción).

Introducción

Aplicaciones:

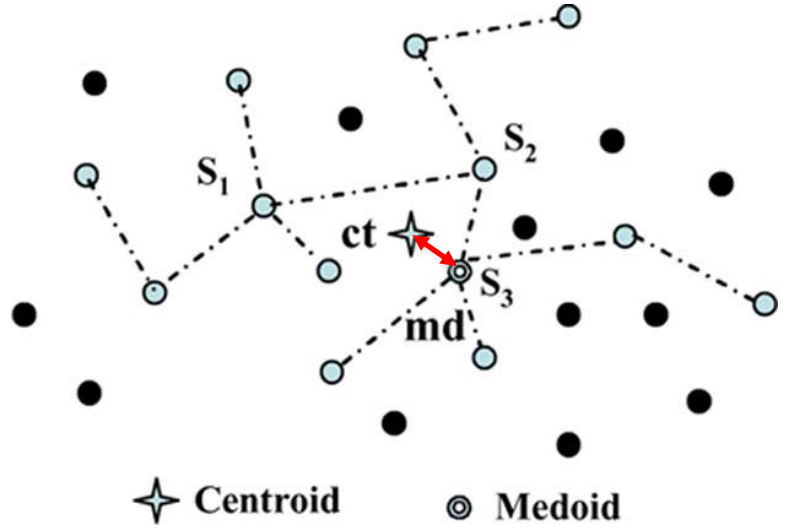
- Biología: para buscar genes que tengan funciones similares.
- Clima: la interpretación del clima requiere encontrar patrones en la atmósfera y en el océano.
- Psicología y medicina: una enfermedad puede tener muchas variantes y con clústering pueden obtenerse esas subcategorías.
- Negocio: la información de clientes o potenciales que se obtiene se puede someter clústering para segmentarlos.

Definiciones

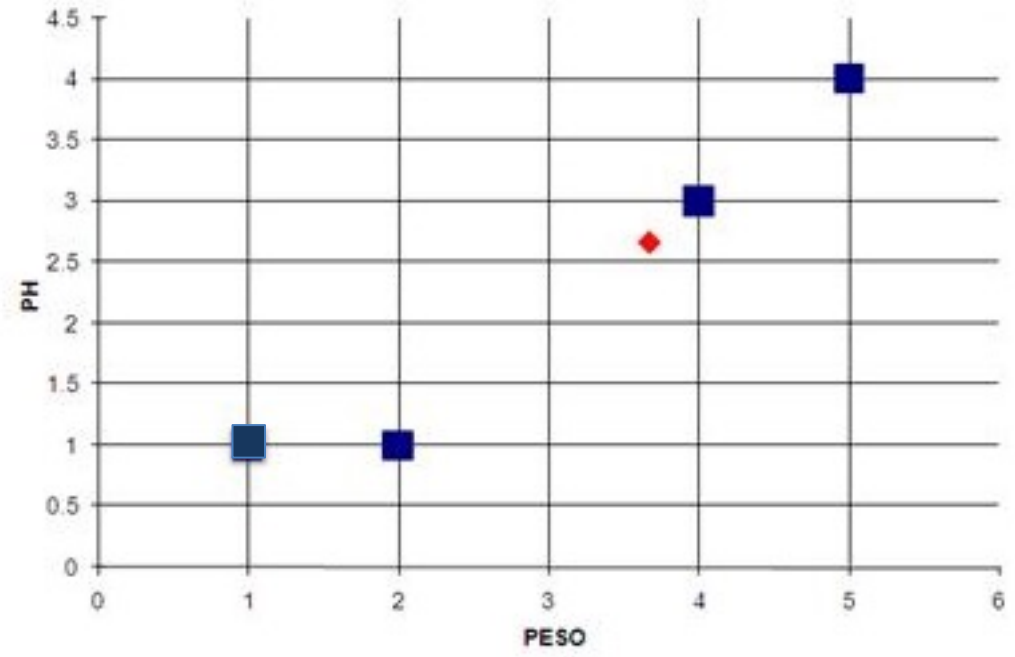
- Centroide: el centro del clúster, es decir, la media de todos los elementos del clúster.

$$centroid = C_m = \frac{\sum_{i=1}^{N}(t_{mi})}{N}$$

- Medoide, es el punto más representativo del clúster.



Definiciones



Centroide: [(2+4+5)/3 , (1+3+4)/3] = [3,66 , 2,66]

Medoide (el más cercano al centroide): [4 , 3]

Definiciones

Tipos de clústering:

1) Clústering particionado

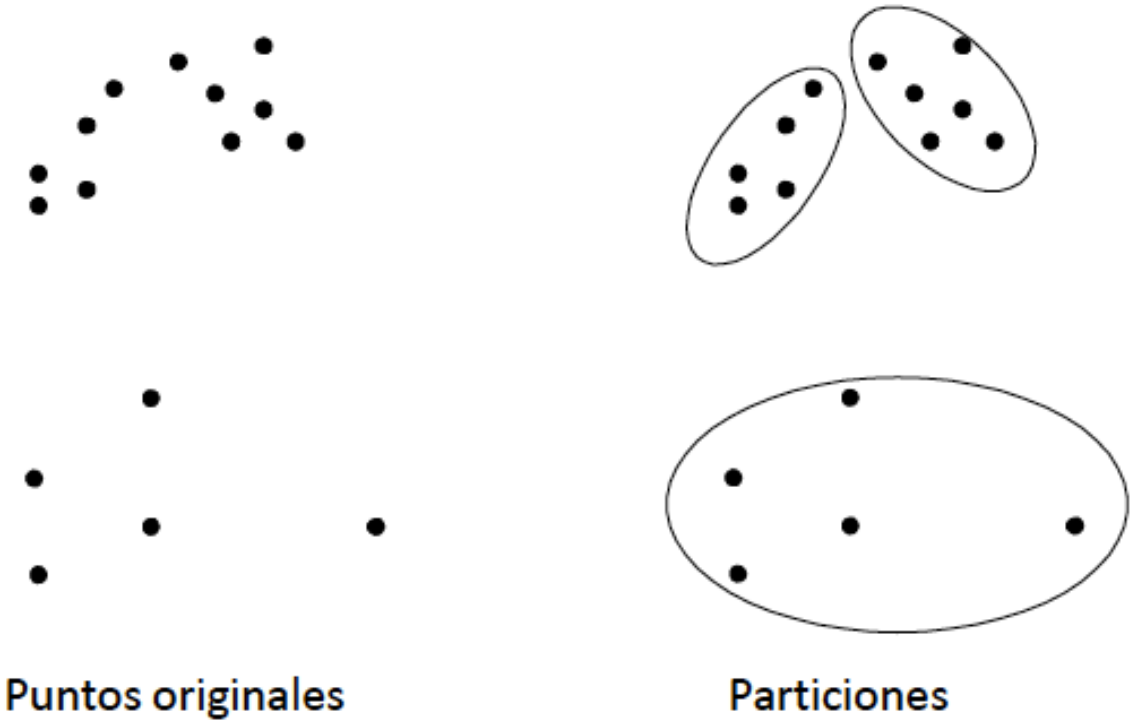
Los grupos no se superponen uno a otro, de modo que cada elemento pertenece únicamente a un clúster.

2) Clústering jerárquico

Conjunto de clústeres anidados y organizados como un árbol jerárquico.

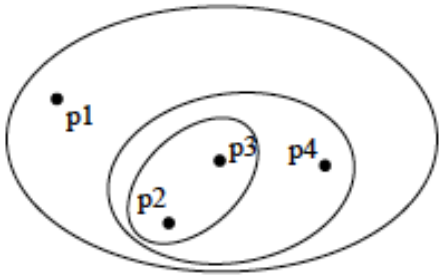
Definiciones

Clústering particionado

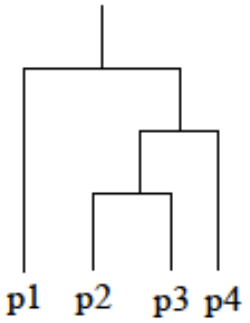


Definiciones

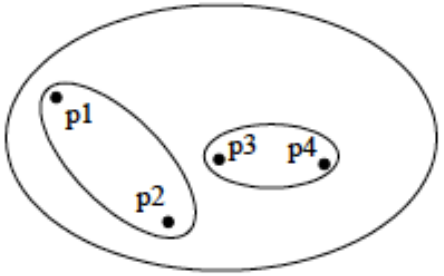
Clústering jerárquico



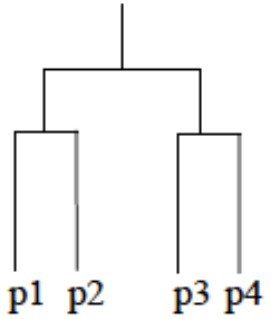
Clúster jerárquico tradicional



Dendrograma tradicional



Clúster jerárquico no tradicional



Dendrograma no tradicional

Definiciones

Diferencias entre clústering

1) Exclusivo vs no-exclusivo: Los no-exclusivos tienen elementos que pueden pertenecer a varios clústeres. Esto permite interpretar múltiples clases o elementos de frontera, es decir que admiten varios comportamientos.

2) Difuso vs no-difuso: En los difusos, los puntos pertenecen a mas clústeres, pero con un peso determinado entre 0 y 1. La suma de los pesos de un individuo es 1. En inglés fuzzy.

Definiciones

3) Parcial vs completo: Es parcial cuando sólo se quiere agrupar algunos datos.

4)Heterogéneo vs homogéneo: Son heterogéneos cuando hay agrupaciones de distinta naturaleza, como diferentes formas, tamaños o densidades.

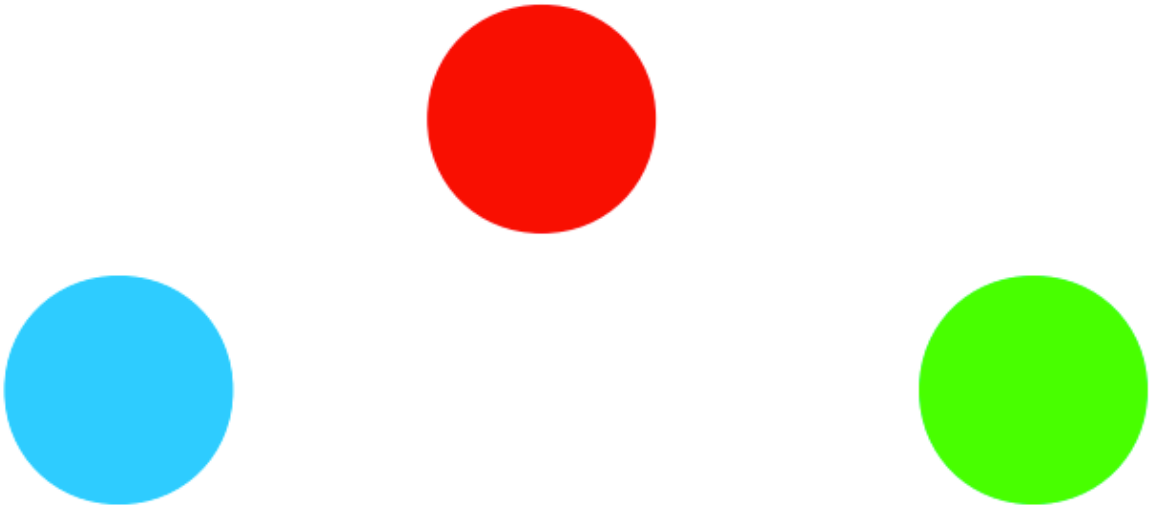
Definiciones

Tipos de clústers:

- Bien definidos/separados.
- Centrados.
- Contiguos.
- Según su densidad.
- De propiedades o conceptuales.
- Descritos por una función objetivo.

Definiciones

Bien definidos o separados: cualquier punto en el clúster es más cercano a cualquier otro punto en el clúster que a cualquier otro punto que no esté en el clúster.



3 clusters bien separados

Definiciones

Centrados: un elemento está más cerca al centro del clúster, que al centro de otro clúster. El centro del clúster será un medoide.

4 clusters basados en el centro

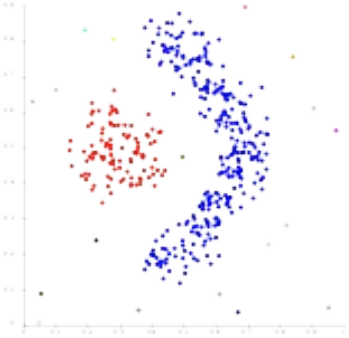
Definiciones

Contiguos: los puntos del mismo clúster están cercanos o son similares a otro punto (o conjunto) del mismo clúster en comparación con otros puntos que no pertenecen al clúster en cuestión.

Definiciones

Según densidad: cuando están separadas regiones de baja densidad, de otras regiones de alta densidad.

Se usan cuando los clústers son irregulares o entrelazados, y cuando se presenta ruido y datos atípicos



Definiciones

Conceptuales: Son clústers que tienen alguna propiedad en común o representan un concepto particular.

Encuentra características comunes entre los elementos de los clústers. Por ejemplo tipos de textos.

Los jerárquicos son conceptuales.

Definiciones

Basados en una función objetivo: Son clústers que minimizan o maximizan una función objetivo.

Enumeran todas las posibles formas de dividir los puntos dentro de un clúster y evalúan la "bondad" de cada conjunto potencial de clústers usando una función objetivo dada.

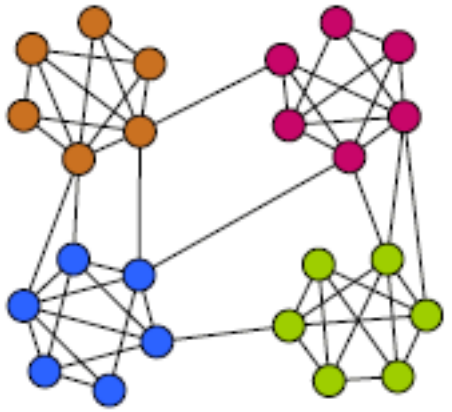
Definiciones

Si es necesario se puede aplicar una función de transformación para mapear el problema de clústering a un dominio diferente y resolver el problema en dicho dominio:

- Con una matriz de adyacencia se define un grafo con pesos en el que los nodos son los puntos a agrupar y las conexiones son la distancia entre puntos.
- Así el clústering equivale a descomponer el grafo en componentes conectadas, una de cada clúster.
- Se busca minimizar las conexiones entre clústeres y maximizar la distancia dentro del clúster.

Definiciones

$$\begin{pmatrix} d11 & d12 & \cdots & d1n \\ d21 & d22 & \cdots & d2n \\ \vdots & \cdots & \cdots & d3n \\ d2n & \cdots & \cdots & dnn \end{pmatrix}$$



Algoritmo: k-means

Sirve para clústering particionado.

Cada clúster está asociado con un centroide (valor de la media del clúster).

Cada punto es asignado al clúster más cercano al centroide.

El número de clústers “K” debe ser especificado.

Estandarizar los datos si es necesario.

Algoritmos: k-means

Pasos:

Eliminar outliers

Seleccionar K puntos diferentes como los **centroides iniciales** (puede ser aleatoriamente)

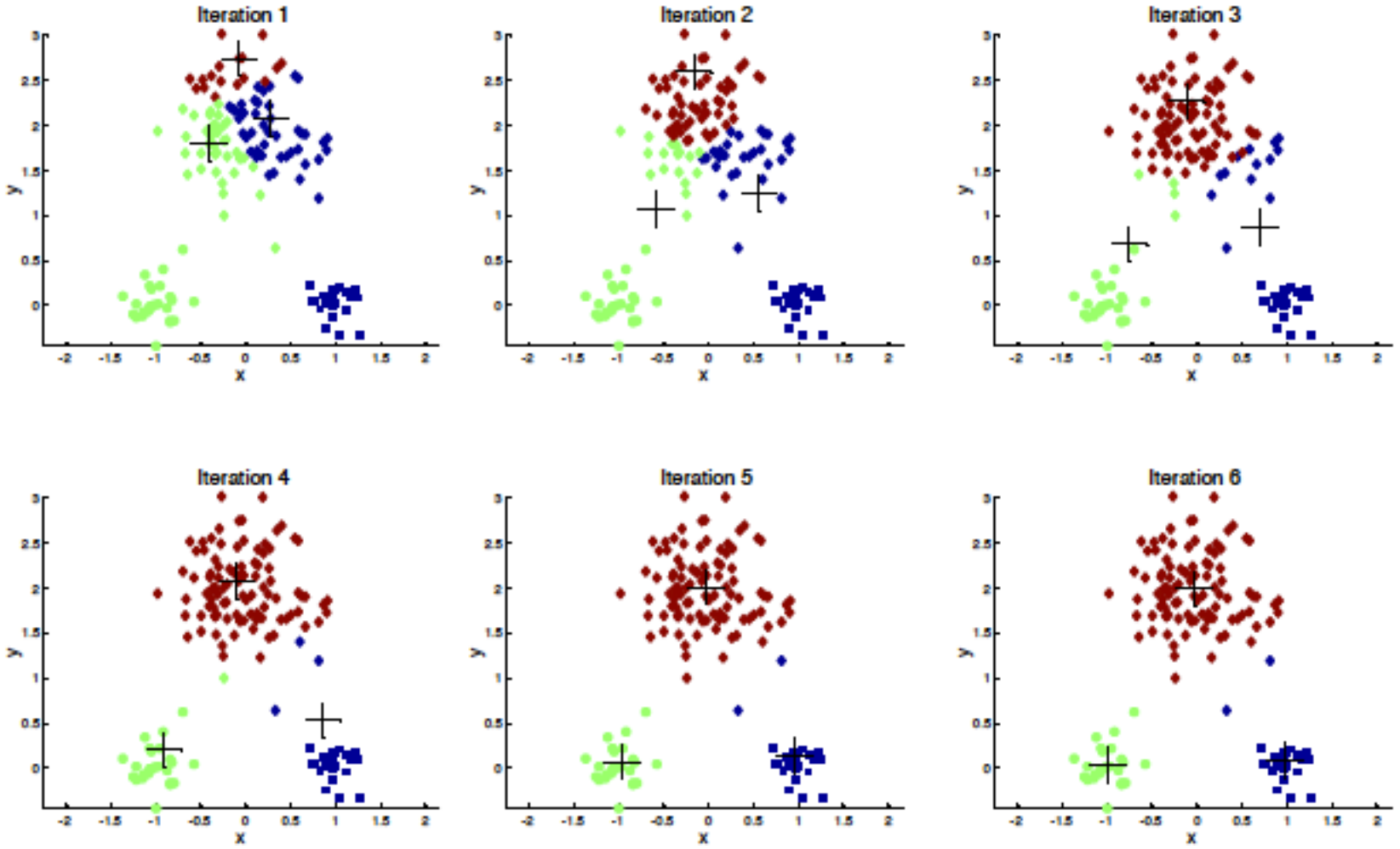
Repetir

Desde **K clústers asignar** todos los puntos al **centroide más cercano**

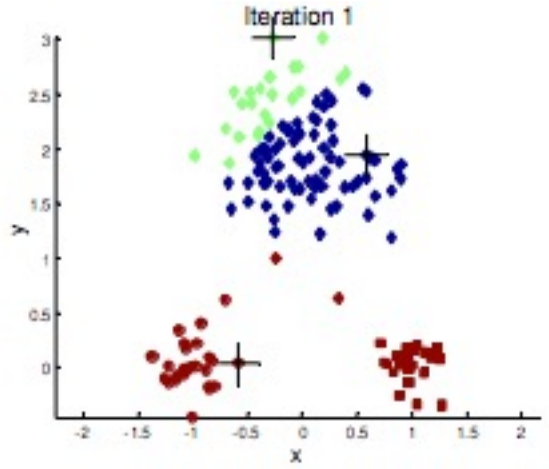
Recalcular el **centroide** de **cada clúster**

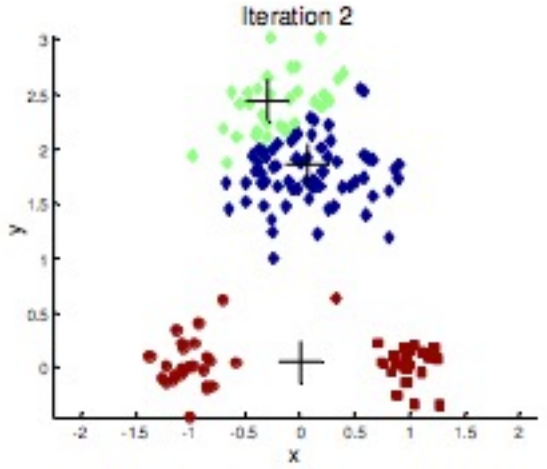
Hasta que el **centroide no cambia**

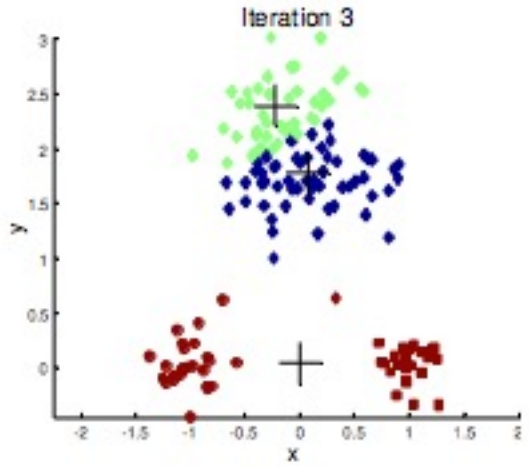
Algoritmos: k-means

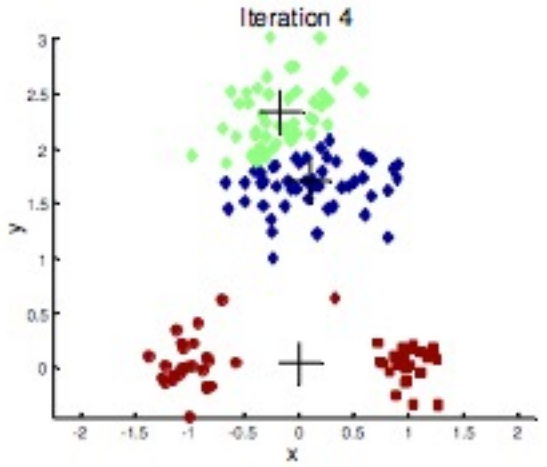


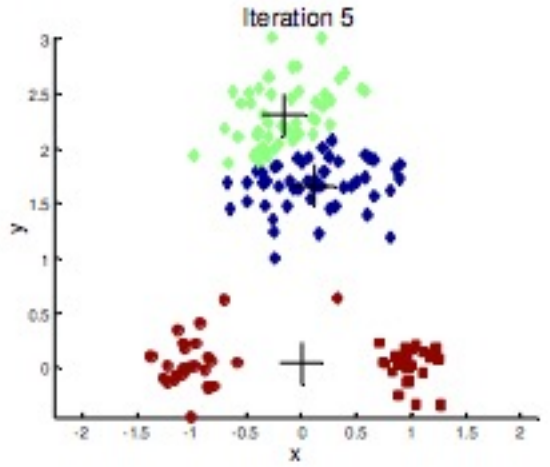
Algoritmos: k-means











Algoritmos: k-means

¿Cómo se eligen los centroides?

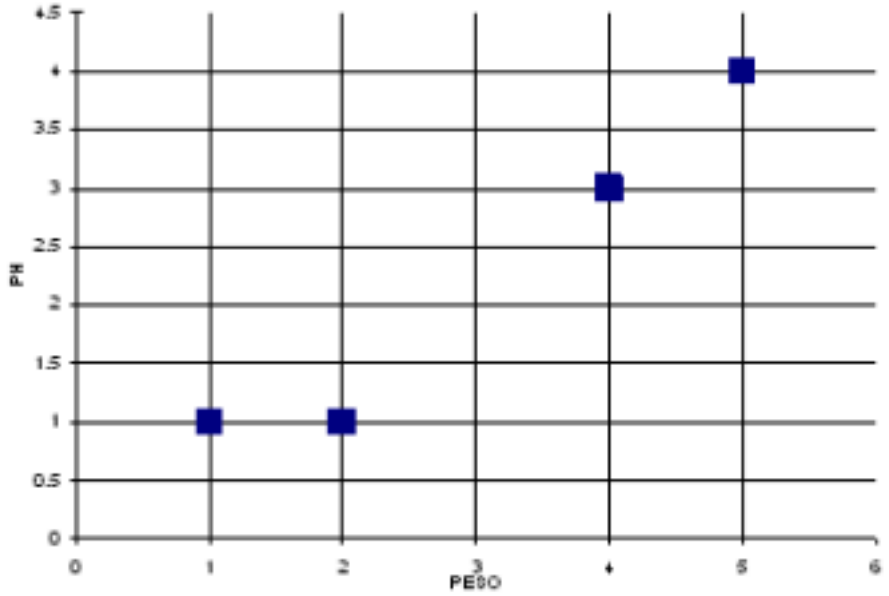
- Repetir la ejecución varias veces por fuerza bruta.
- Seleccionar más de K centroides iniciales y entonces elegir de entre esos donde se estabiliza la distancia media al centroide. Método del codo.
- Con la diferencia entre la distancia media al centroide (a) y al cluster más cercano (b). S= (b-a)/max(a,b)
- Postprocesamiento.

Algoritmos: k-means

Ejemplo (medicamentos):

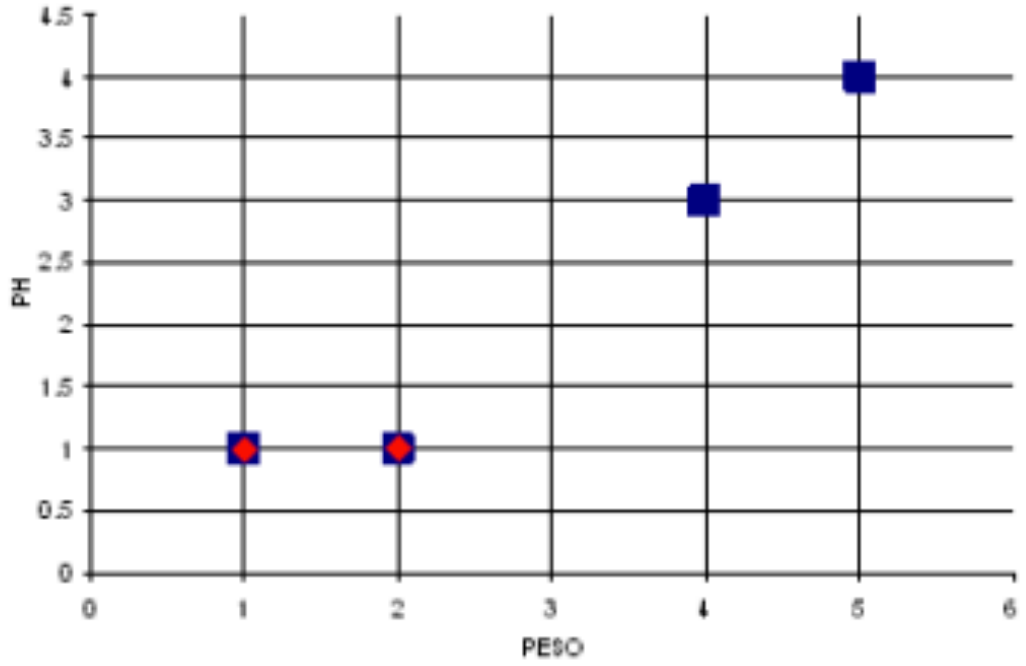
K=2

Peso	Índice ph
1	1
2	1
4	3
5	4



Algoritmos: k-means

Se eligen los centroides



Algoritmos: k-means

Calculo la distancia entre los puntos y los centroides:

Usando la distancia Euclídea, la distancia entre el punto (4,3) y el centroide (1,1) es:

$$\sqrt{(4-1)^2+(3-1)^2}=3{,}61$$

La distancia del mismo punto con el otro centroide es 2,83

Se plantea una matriz de distancias entre los puntos y los centroides (las columnas son los puntos y las filas los centroides)

$$\begin{pmatrix} 0{,}00 & 1{,}00 & 3{,}61 & 5{,}00 \\ 1{,}00 & 0{,}00 & 2{,}83 & 4{,}24 \end{pmatrix}$$

Algoritmos: k-means

A partir de la matriz anterior y teniendo en cuenta las distancias mínimas a los centroides. Se genera otra matriz con las asignaciones de los clústers (0 no pertenece al clúster y 1 pertenece). Dependiendo de la fila en la que se encuentre será el clúster asociado a un centroide o al otro.

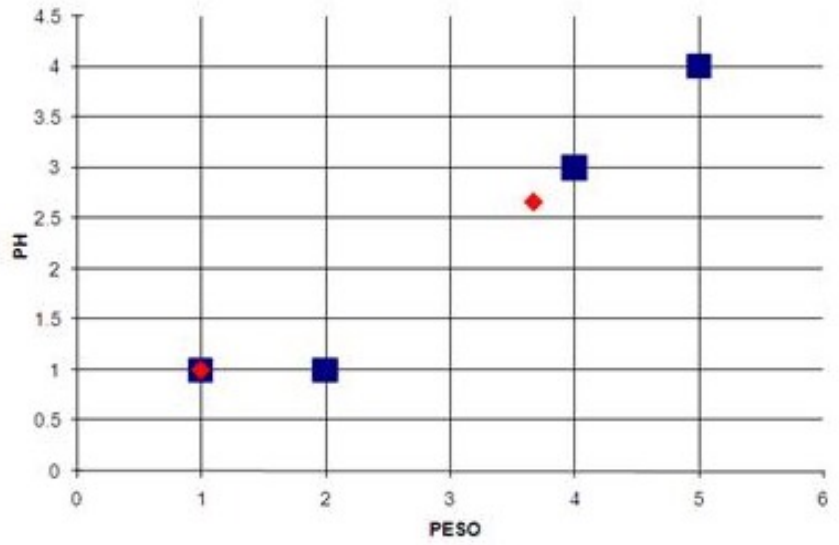
$$\begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 \end{pmatrix}$$

Algoritmos: k-means

Por último se calculan los nuevos centroides. Si no se han movido se para. En caso contrario se repite el proceso.

C1 = [1,0]

C2 = [(2+4+5)/3 , (1+3+4)/3] = [3,66 ,2,66]



Algoritmos: k-means

Se obtiene la matriz de distancias (centroides/filas vs individuos/columnas):

$$\begin{pmatrix} 0{,}00 & 1{,}00 & 3{,}61 & 5{,}00 \\ 3{,}14 & 2{,}36 & 0{,}47 & 1{,}89 \end{pmatrix}$$

La matriz para los clústers:

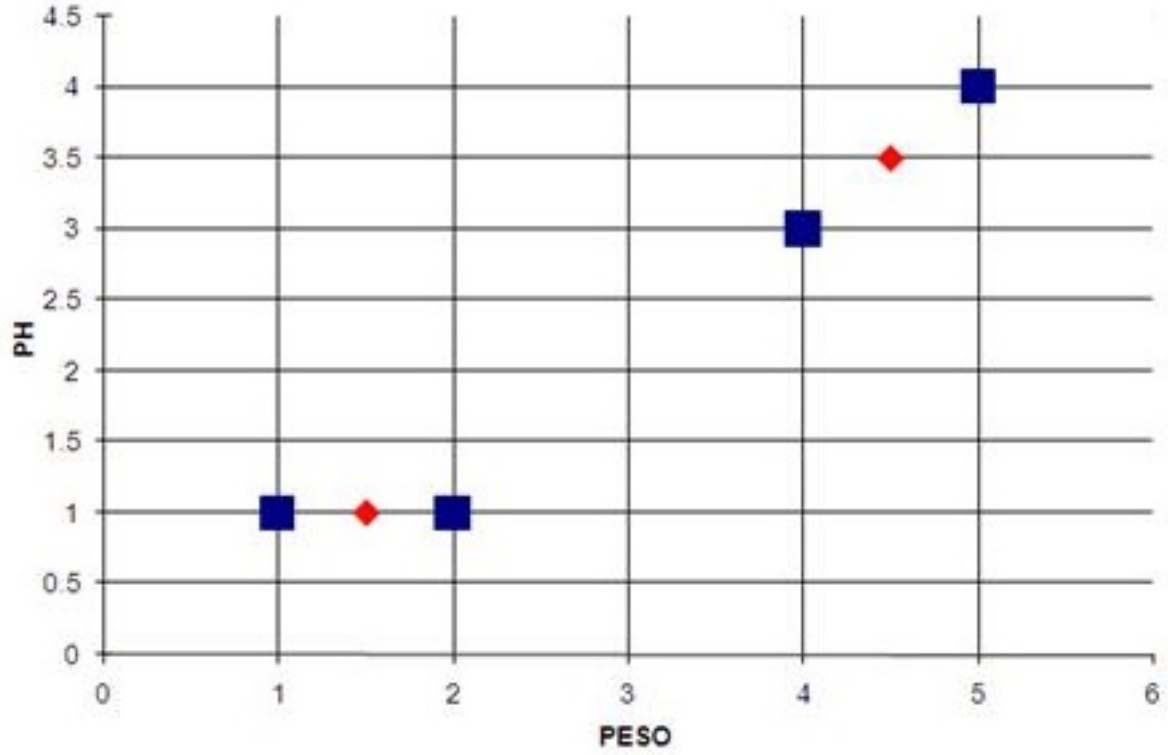
$$\begin{pmatrix} 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \end{pmatrix}$$

Los valores de los centroides:

C1=[1,5 , 1] C2=[4,5 , 3,5]

Algoritmos: k-means

Se dibujan los nuevos centroides



Algoritmos: k-means

Otra iteración

Se obtiene la matriz de distancias :

$$\begin{pmatrix} 0{,}50 & 0{,}50 & 3{,}20 & 4{,}61 \\ 4{,}30 & 3{,}54 & 0{,}71 & 0{,}71 \end{pmatrix}$$

La matriz para los clústers :

$$\begin{pmatrix} 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \end{pmatrix}$$

Cómo no se han movido, se puede decir que se ha llegado al final.

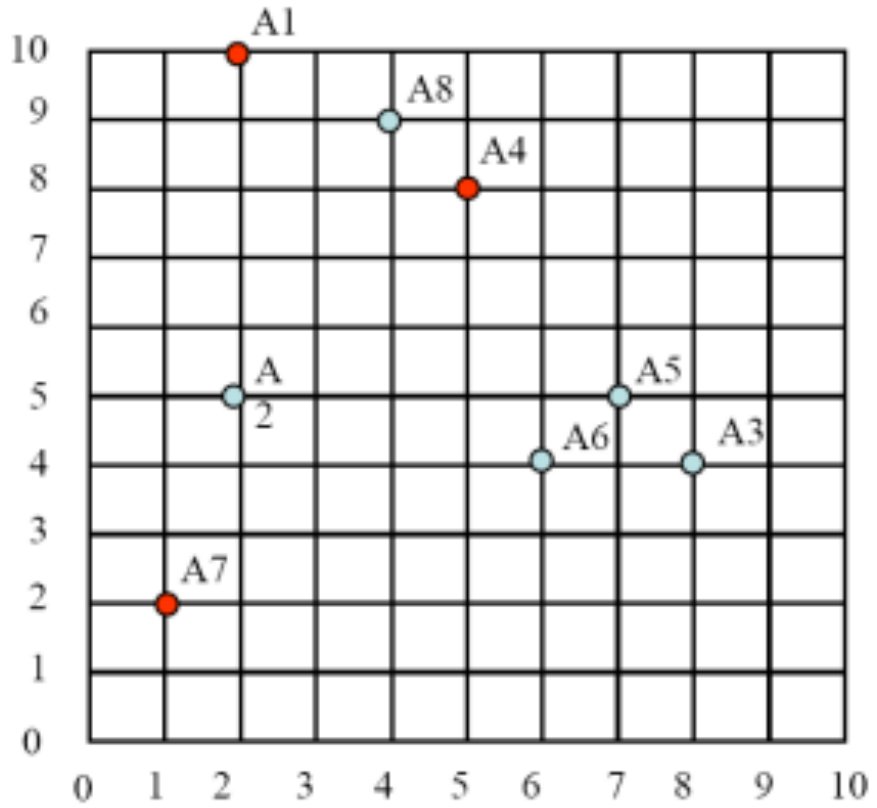
Algoritmos: k-means

Ejercicio:

Para k=3

Centroides: (1,2), (2,10), (5,8)

X	Y
2	5
4	9
6	4
7	5
8	4
1	2
2	10
5	8



Algoritmos: k-means

Paso 1:

Matriz de distancias

$$\begin{pmatrix} 3{,}16 & 7{,}61 & 5{,}38 & 6{,}70 & 7{,}28 & 0 & 8{,}06 & 7{,}21 \\ 5 & 2{,}23 & 7{,}21 & 7{,}07 & 8{,}48 & 8{,}66 & 0 & 3{,}60 \\ 4{,}24 & 1{,}41 & 4{,}12 & 3{,}60 & 5 & 7{,}21 & 3{,}60 & 0 \end{pmatrix}$$

Clústers

$$\begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 1 & 1 & 1 & 0 & 0 & 1 \end{pmatrix}$$

Nuevos centroides: (1,5 3,5) (2 10) (6 6)

Algoritmos: k-means

Paso 2:

Matriz de distancias

$$\begin{pmatrix} 1{,}58 & 6{,}04 & 4{,}52 & 5{,}70 & 6{,}51 & 1{,}58 & 6{,}51 & 5{,}70 \\ 5 & 2{,}23 & 7{,}21 & 7{,}07 & 8{,}48 & 8{,}66 & 0 & 3{,}60 \\ 4{,}12 & 3{,}60 & 2 & 1{,}41 & 2{,}82 & 6{,}40 & 5{,}65 & 2{,}23 \end{pmatrix}$$

Clústers

$$\begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 1 & 1 & 0 & 0 & 1 \end{pmatrix}$$

Nuevos centroides: (1,5 3,5) (3 9,5) (6,5 5,25)

Algoritmos: k-means

Paso 3:

Matriz de distancias

$$\begin{pmatrix} 1{,}58 & 6{,}04 & 4{,}52 & 5{,}70 & 6{,}51 & 1{,}58 & 6{,}51 & 5{,}70 \\ 4{,}6 & 1{,}11 & 6{,}26 & 6{,}02 & 7{,}43 & 7{,}76 & 1{,}11 & 2{,}5 \\ 4{,}5 & 4{,}5 & 1{,}34 & 0{,}55 & 1{,}95 & 6{,}38 & 6{,}54 & 3{,}13 \end{pmatrix}$$

Clústers

$$\begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 \end{pmatrix}$$

Nuevos centroides: (1,5 3,5) (3,66 9) (7 4,33)

Algoritmos: k-means

Paso 4:

Matriz de distancias

$$\begin{pmatrix} 1{,}58 & 6{,}04 & 4{,}52 & 5{,}70 & 6{,}51 & 1{,}58 & 6{,}51 & 5{,}70 \\ 4{,}33 & 0{,}34 & 5{,}52 & 5{,}21 & 6{,}62 & 7{,}48 & 1{,}93 & 1{,}67 \\ 5{,}04 & 5{,}55 & 1{,}05 & 0{,}67 & 1{,}05 & 6{,}43 & 7{,}55 & 4{,}17 \end{pmatrix}$$

Clústers

$$\begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 \end{pmatrix}$$

Nuevos centroides: No hay. **PARAMOS!!!!**

Algoritmos: k-means

Ejercicio (anchura y altura minerales)

K=2 inicializado en (1,3) y (3,7)

Anchura	Altura
1	1
3	2
4	4
6	4

Algoritmos: k-means

Evaluación de los clústers:

Suma del error cuadrático (SSE)

Para cada punto, el error es la distancia al clúster más cercano :

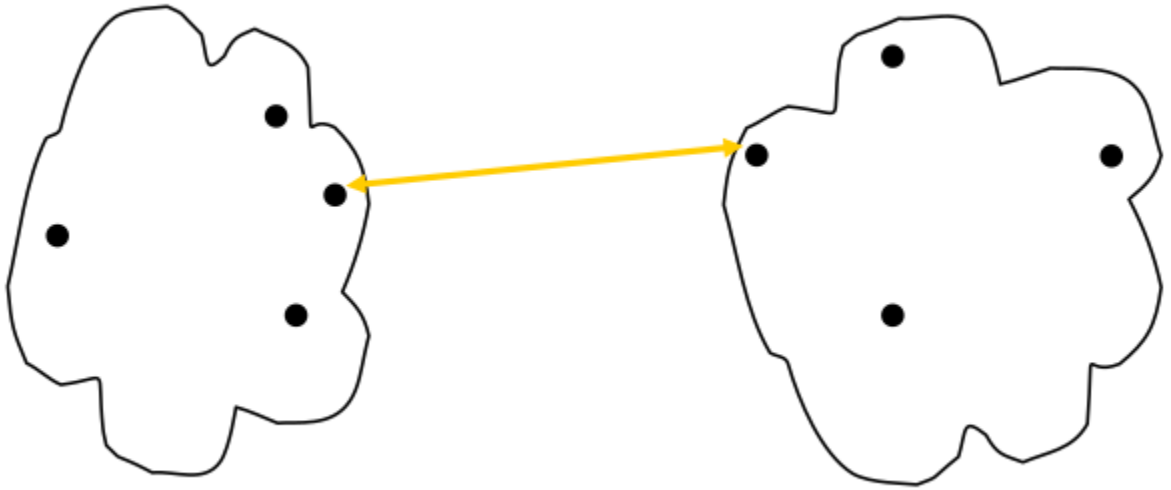
$$SSE = \sum_{i=1}^{K} \sum_{x \in C_i} dist^2(m_i, x)$$

x es un punto del clúster C_i y m_i es el centroide

Se puede reducir el SSE incrementando el número de clústeres K. Hay que minimizar la función.

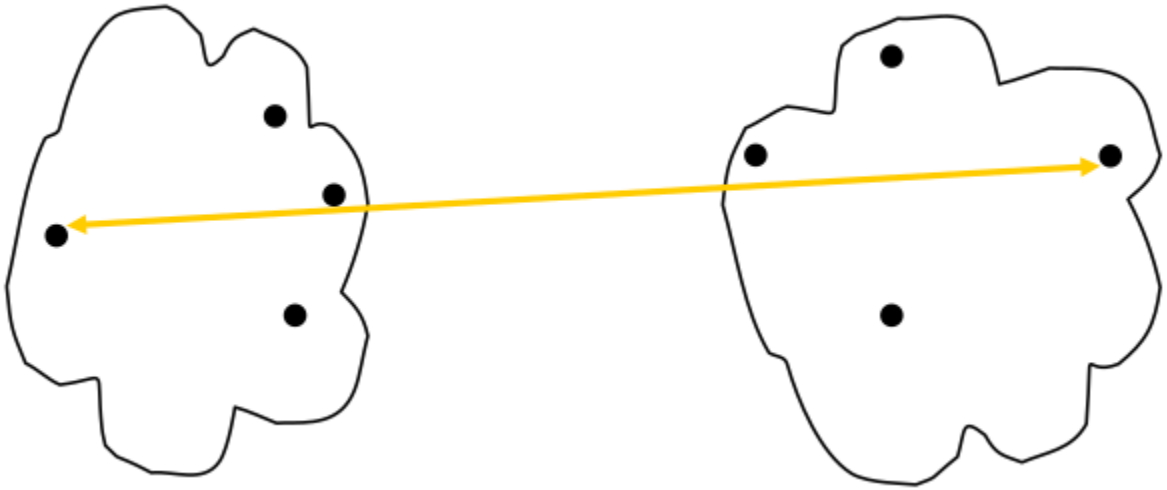
Similitud entre clústers

Distancia mínima



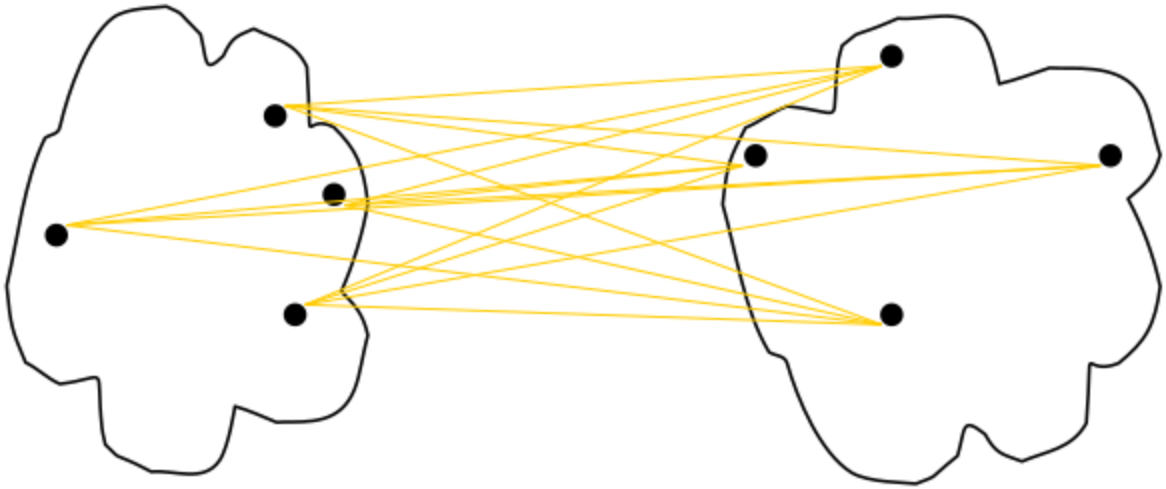
Similitud entre clústers

Distancia máxima



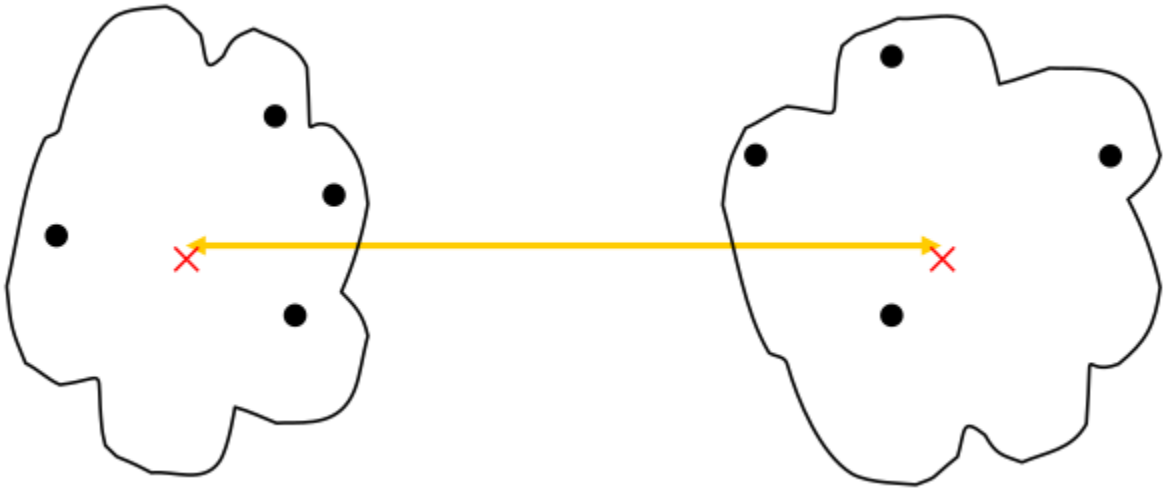
Similitud entre clústers

Distancia media



Similitud entre clústers

Distancia centroide



Validación de los clústers

1. Determinación de la tendencia del clústering de un conjunto de datos, es decir, si se da algún tipo de tendencia que no sea al azar.

2. Comparar los resultados del análisis con resultados externos conocidos, alguna variable clase.

3. Evaluar lo bien que se ajusta el resultado del análisis sin referencias a información externa.

4. Comparar los resultados de dos conjuntos de datos para ver cuál es mejor.

Algoritmos: k-means

Estrategias de postprocesamiento:

- Elimina los clústeres de posibles datos atípicos.
- Divide clústeres dudosos con alto SSE.
- Mezcla clústeres cercanos con un bajo SSE.
- A veces se pueden añadir estos procesos dentro del procesamiento básico de clústering.

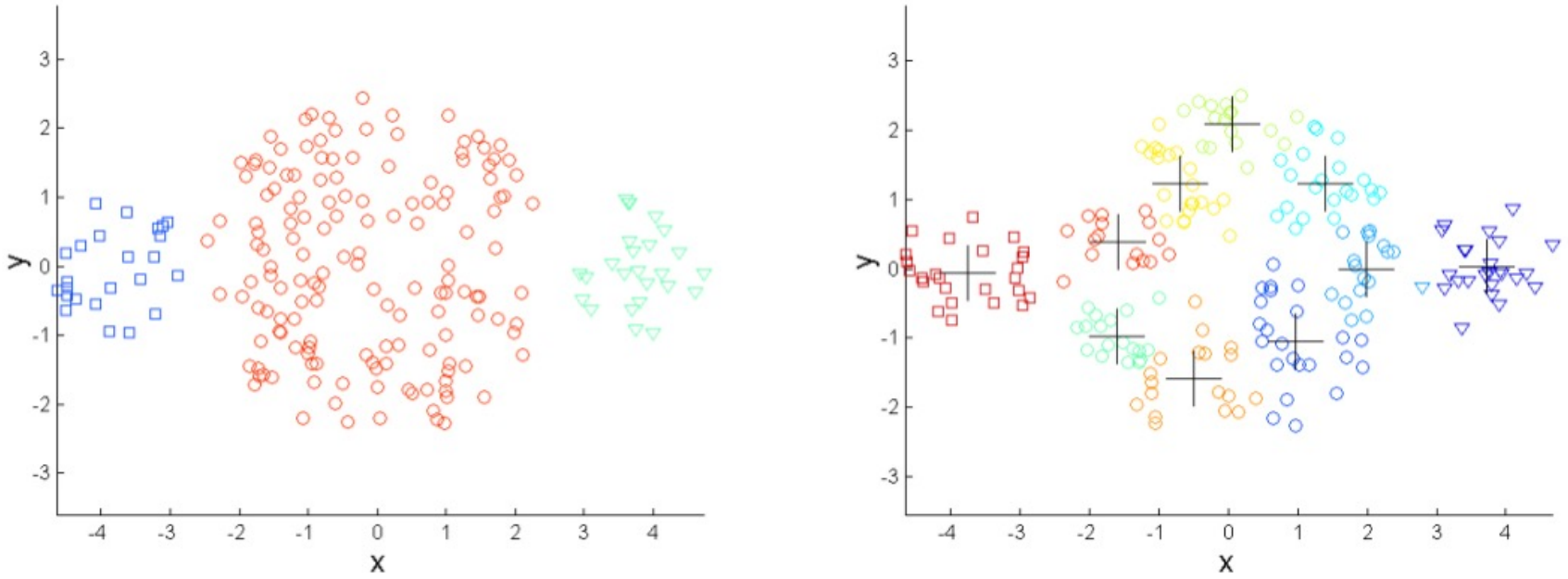
Algoritmos: k-means

El algoritmo K-means tiene problemas cuando los clústeres son muy diferentes en:

- Tamaño.
- Densidad.
- Forma (formas irregulares).
- Cuando hay datos atípicos.

Algoritmos: k-means

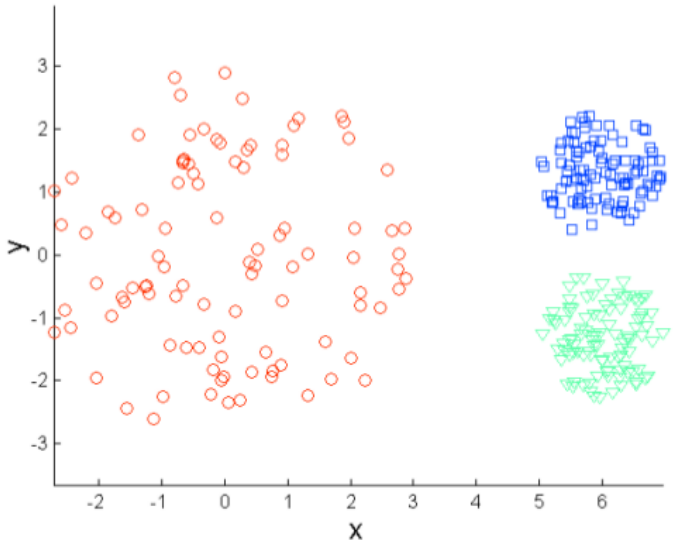
Tamaño



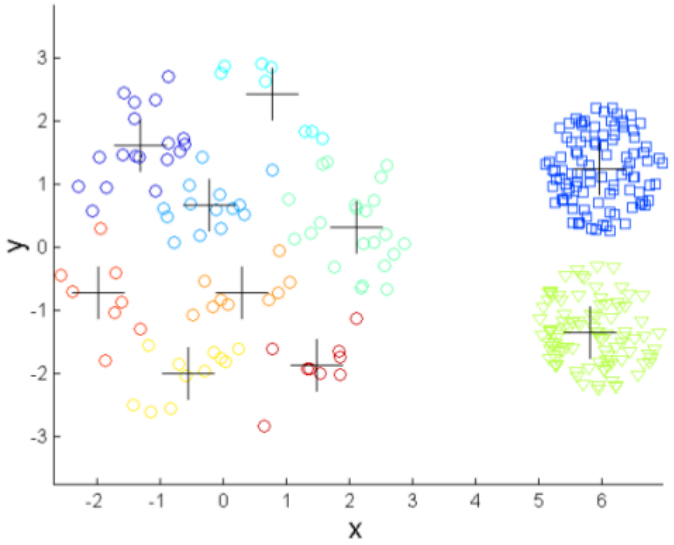
Generar muchos clústers y juntar los más cercanos.

Algoritmos: k-means

Densidad



Puntos originales

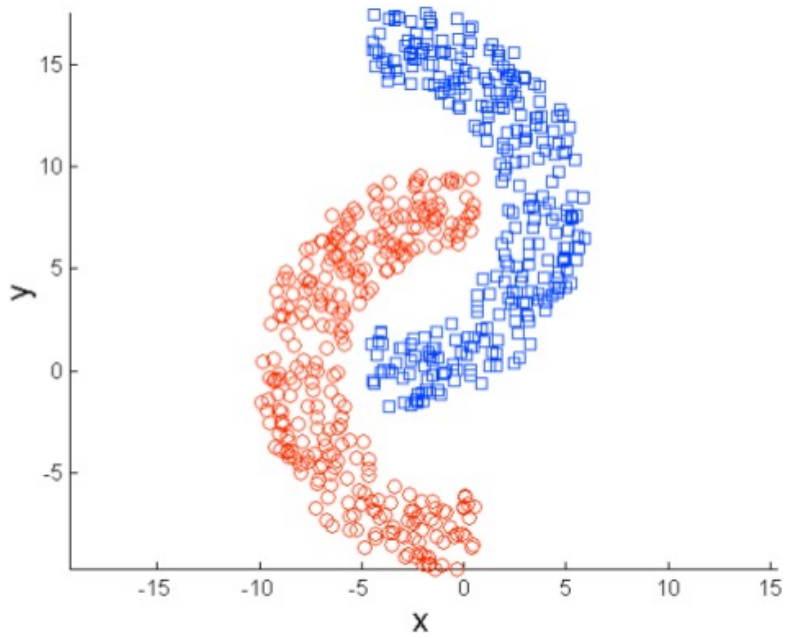


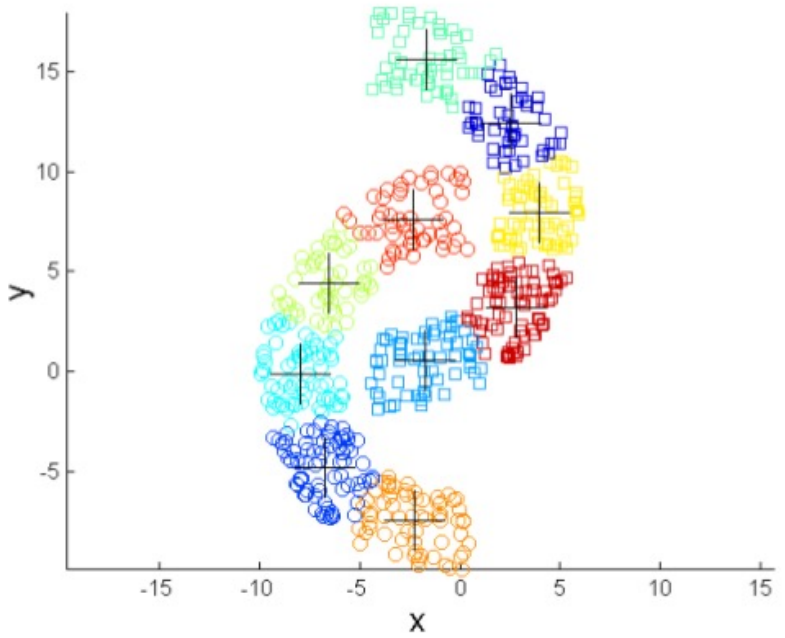
Clústeres K-means

Poner más centroides donde menos densidad hay.

Algoritmos: k-means

Forma (formas irregulares).





Generar los centroides a lo largo o ancho de las formas.

Algoritmos: Clustering jerárquico

Produce un conjunto de clústeres anidados organizados con un árbol jerárquico:

- Se puede emplear un dendrograma.
- Un diagrama arbóreo que representa las divisiones o mezclas.
- No se necesita asumir un número previo de clústeres.

Algoritmos: Clustering jerárquico

2 tipos principales:

- Aglomerativo: Al inicio, cada punto es un clúster. En cada paso, se mezcla los pares de clústeres más cercanos entre sí en un único elemento.

- Disociativo: Empezar con un clúster que incluya todo. En cada paso, dividir un clúster hasta que tengamos k clústeres o solo haya un punto por clúster.

Algoritmos: Clustering jerárquico

Algoritmo de **single linkage aglomerativo**:

Calcula la matriz de proximidad

Cada punto es un clúster

Repetir

- Mezclar dos clústeres que estén lo más cerca posible.
- Actualizar la matriz de proximidad

Hasta solo quede un clúster

Algoritmos: Clustering jerárquico

Ejemplo:

1) Se calcula la matriz de adyacencia

	A	B	C	D	E	F	G
A	0						
B	2,15	0					
C	0,78	1,53	0				
D	1,07	1,14	0,43	0			
E	0,85	1,38	0,21	0,29	0		
F	1,16	1,01	0,55	0,22	0,41	0	
G	1,56	2,83	1,856	2,04	2,02	2,05	0

Algoritmos: Clustering jerárquico

2) Los más cercanos son C y E con distancia 0,21. Se crea un clúster con ellos.

3) Se calcula la nueva matriz de adyacencia con las distancias menores de los nodos que forman el clúster al resto de nodos que tengan en común.

Algoritmos: Clustering jerárquico

	A	B	C	D	E	F	G
A	0						
B	2,15	0					
C	**0,78**	1,53	0				
D	1,07	1,14	0,43	0			
E	0,85	**1,38**	0,21	**0,29**	0		
F	1,16	1,01	0,55	0,22	**0,41**	0	
G	1,56	2,83	**1,86**	2,04	2,02	2,05	0

Con A
Con B
Con D
Con F
Con G

Algoritmos: Clústering jerárquico

3) Se calcula la matriz de adyacencia otra vez hasta que no haya más grupos que hacer.

	A	B	(C,E)	D	F	G
A	0					
B	2,15	0				
(C,E)	0,78	1,38	0			
D	**1,07**	1,14	**0,29**	0		
F	1,16	**1,01**	0,41	0,22	0	
G	1,56	2,83	1,86	**2,04**	2,05	0

Algoritmos: Clústering jerárquico

4) Se calcula la matriz de adyacencia otra vez hasta que no haya nada más. Para D y F que es la menor distancia.

	A	B	(C,E)	(D,F)	G
A	0				
B	2,15	0			
(C,E)	**0,78**	1,38	0		
(D,F)	1,07	**1,01**	0,29	0	
G	1,56	2,83	**1,86**	2,04	0

Algoritmos: Clústering jerárquico

5)

	A	B	((C,E), (D,F))	G
A	0			
B	2,15	0		
((C,E), (D,F))	0,78	1,01	0	
G	1,56	2,83	1,86	0

6)

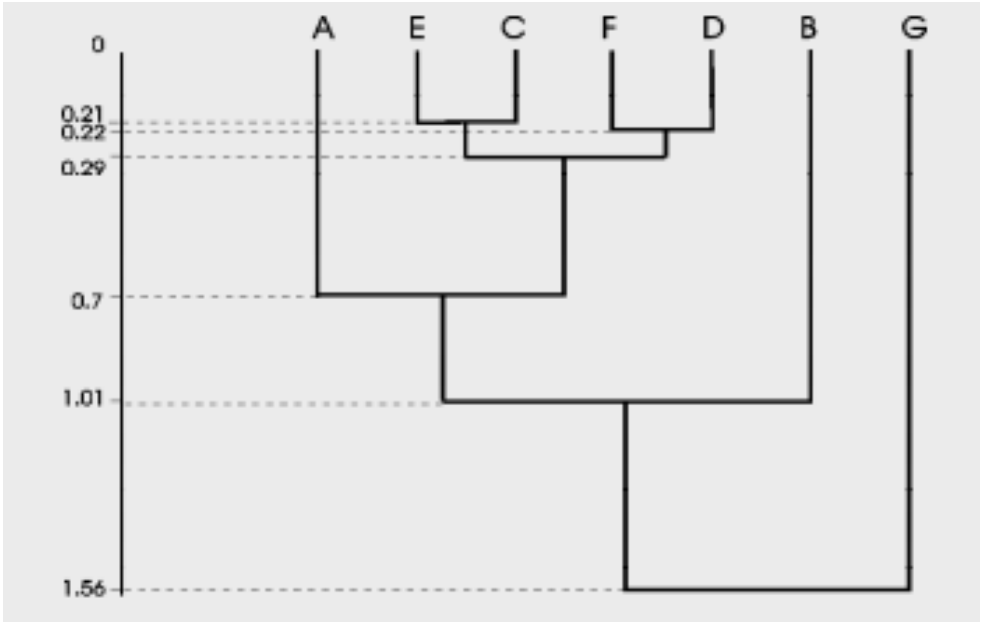
	(A, ((C,E), (D,F))	B	G
(A, ((C,E), (D,F)))	0		
B	1,01	0	
G	1,56	2,83	0

7)

	(B,(A, ((C,E), (D,F))))	G
(B,(A, ((C,E), (D,F))))	0	
G	1,56	0

Algoritmos: Clústering jerárquico

7) Se obtiene el dendrograma desde dentro hacia afuera.



Algoritmos: Clústering jerárquico

Ejemplo:

Animal	Orejas	Movilidad	Hábitat	Pico
Gato	Punta	Cuadrúpedo	Doméstico	No
Tigre	Punta	Cuadrúpedo	Salvaje	No
Oso	Redonda	Cuadrúpedo	Salvaje	No
Cuervo	No	Bípedo	Salvaje	Si
Ornitorrinco	No	Cuadrúpedo	Salvaje	Si

Algoritmos: Clústering jerárquico

Ejemplo:

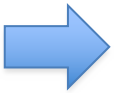
Animal	Orejas	Movilidad	Hábitat	Pico
Gato	1	1	1	1
Tigre	1	1	2	1
Oso	2	1	2	1
Cuervo	3	2	2	2
Ornitorrinco	3	1	2	2

	G	T	O	C	Or
G	0				
T	1	0			
O	1,41	1	0		
C	2,46	2,44	3	0	
Or	2,44	2,23	1,73	1,41	0

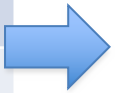
Algoritmos: Clústering jerárquico

	G	T	O	C	Or
G	0				
T	1	0			
O	1,41	1	0		
C	2,46	2,44	3	0	
Or	2,44	2,23	1,73	1,41	0

	(G,T)	O	C	Or
(G,T)	0			
O	1	0		
C	2,44	3	0	
Or	2,23	1,73	1,41	0



	(G,T),O	C	Or
(G,T),O	0		
C	2,44	0	
Or	1,73	1,41	0



	((G,T),O), C	Or
((G,T),O), C	0	
Or	1,41	0

Algoritmos: Clústering jerárquico

Interpretación:

	(((G,T),O),C)	Or
(((G,T),O),C)	0	
Or	1,41	0

El gato, el tigre se parecen mucho (uno es doméstico y otro salvaje) y el oso se parece a ellos en parte. Luego jerárquicamente encontramos al cuervo y al ornitorrinco.

Algoritmos: Clústering jerárquico

Ej:

	A	B	C	D
A	0			
B	9	0		
C	4	5	0	
D	7	3	11	0

Algoritmos: DBSCAN

Tiene un enfoque basado en la densidad, modelando los clústers como cúmulos de alta densidad de puntos. Por lo cual, si un punto pertenece o no a un clúster, debe estar cerca de un montón de otros puntos de dicho clúster.

Algoritmos: DBSCAN

Pasos:

A partir de épsilon positivo (radio del cluster) y un minPoints (individuos mínimos en el clúster)

Repetir hasta que no se generen nuevos clusters:

Se elige un punto arbitrario en el conjunto de datos.

Creo un cluster que cumpla con minPoints a una distancia épsilon.

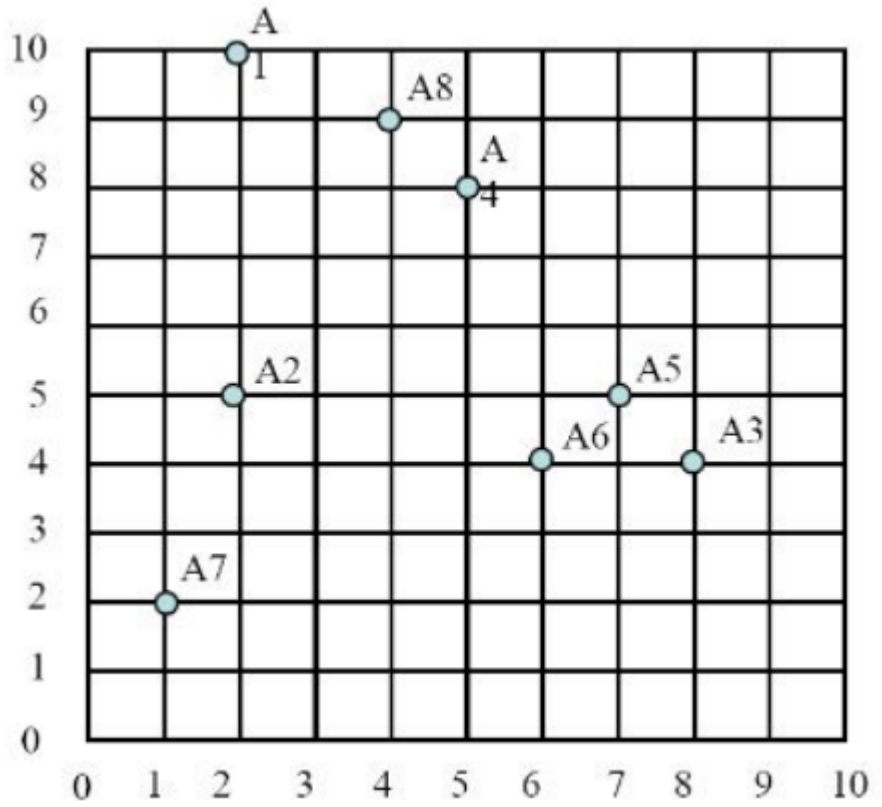
Si un cluster no cumple con la condición de minPoints se considera un outlier.

Algoritmos: DBSCAN

Ejemplo:

minPoints=2

Epsilon=$\sqrt{2}$



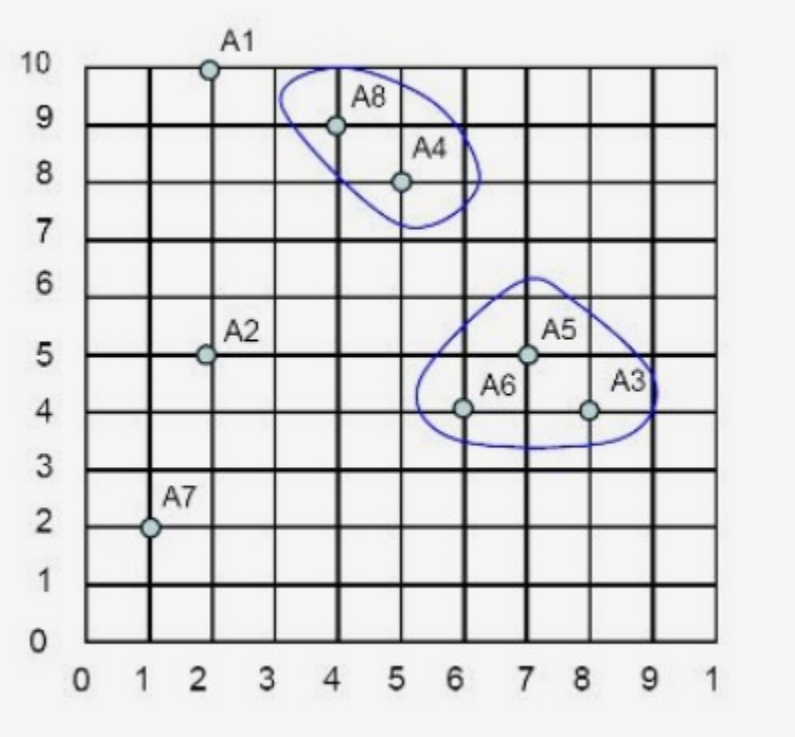
Algoritmos: DBSCAN

1) Se calcula la matriz de adyacencia con la distancia Euclídea

	A1	A2	A3	A4	A5	A6	A7	A8
A1	0							
A2	Sqrt(25)	0						
A3	Sqrt(72)	Sqrt(37)	0					
A4	Sqrt(13)	Sqrt()18	Sqrt(25)	0				
A5	Sqrt(50)	Sqrt(25)	Sqrt(2)	Sqrt(13)	0			
A6	Sqrt(52)	Sqrt(17)	Sqrt(4)	Sqrt(17)	Sqrt(2)	0		
A7	Sqrt(65)	Sqrt(10)	Sqrt(53)	Sqrt(52)	Sqrt(45)	Sqrt(29)	0	
A8	Sqrt(5)	Sqrt(20)	Sqrt(41)	Sqrt(2)	Sqrt(25)	Sqrt(29)	Sqrt(58)	0

Algoritmos: DBSCAN

2) Los que cumplan con la regla de épsilon se agrupan



Algoritmos: DBSCAN

3) Al haber outilers se aumenta la épsilon a $\sqrt{10}$

	A1	A2	A3	A4	A5	A6	A7	A8
A1	0							
A2	Sqrt(25)	0						
A3	Sqrt(72)	Sqrt(37)	0					
A4	Sqrt(13)	Sqrt(18)	Sqrt(25)	0				
A5	Sqrt(50)	Sqrt(25)	Sqrt(2)	Sqrt(13)	0			
A6	Sqrt(52)	Sqrt(17)	Sqrt(4)	Sqrt(17)	Sqrt(2)	0		
A7	Sqrt(65)	Sqrt(10)	Sqrt(53)	Sqrt(52)	Sqrt(45)	Sqrt(29)	0	
A8	Sqrt(5)	Sqrt(20)	Sqrt(41)	Sqrt(2)	Sqrt(25)	Sqrt(29)	Sqrt(58)	0

Algoritmos: DBSCAN

4) Los clusters cumplen con la condición mínima. Se para.