

Aprendizaje No Supervisado

Técnicas de agrupamiento y reducción de la dimensionalidad

Enrique J. Carmona Suárez

Dpto. Inteligencia Artificial, ETS Ingeniería Informática, UNED

ecarmona@dia.uned.es

Second IPARCOS Workshp on Machine Learning and Applications to Physics - 18-20 diciembre 2023

Facultad de Ciencias Físicas, Universidad Complutense de Madrid

Contenido

- Técnicas de agrupamiento
 - Algoritmo k-medias
 - Agrupamiento jerárquico
 - Algoritmo DBSCAN
 - Mapas auto-organizados
- Reducción de la dimensionalidad
 - Análisis de componentes principales

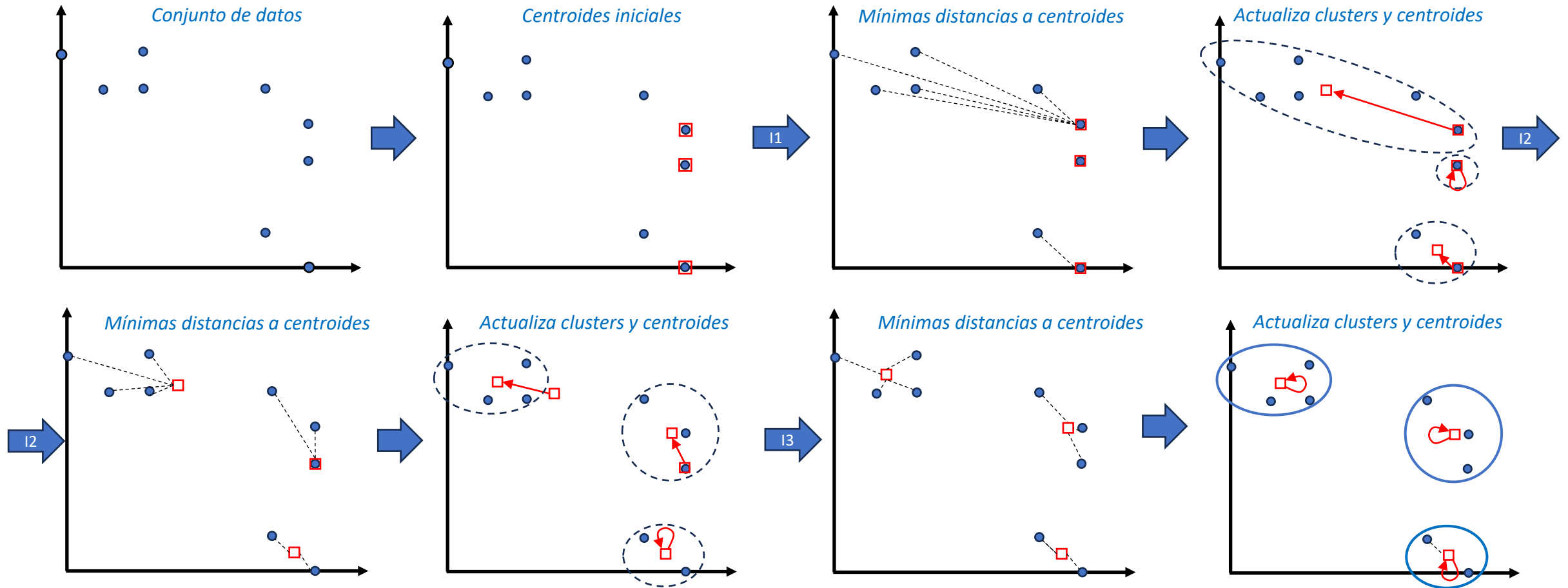
Contenido

- Técnicas de agrupamiento
 - Algoritmo k-medias
 - Características
 - Algoritmo y ejemplo de uso
 - Selección del valor de k
 - Agrupamiento jerárquico
 - Algoritmo DBSCAN
 - Mapas auto-organizados
- Reducción de la dimensionalidad
 - Análisis de componentes principales

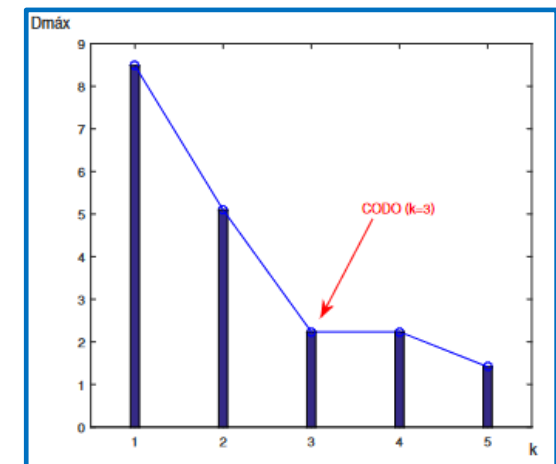
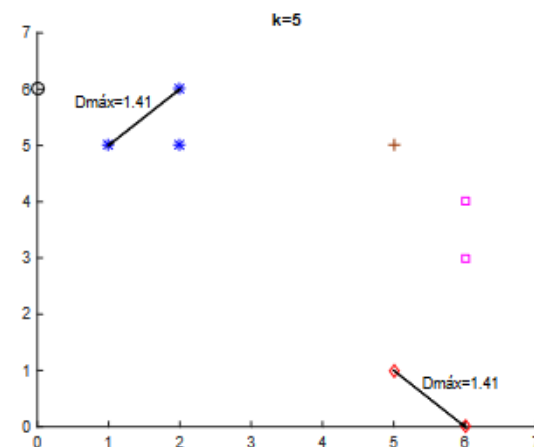
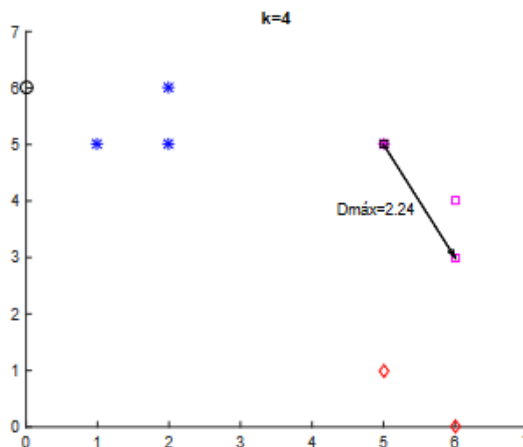
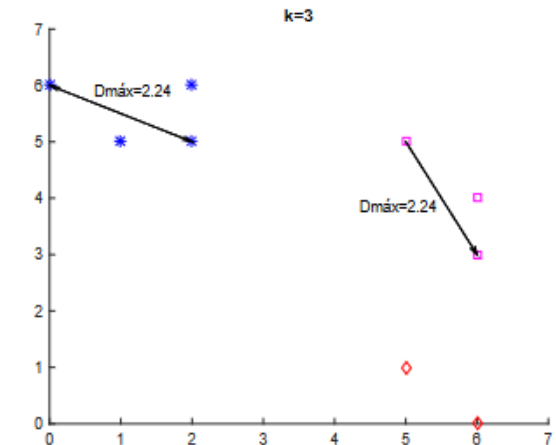
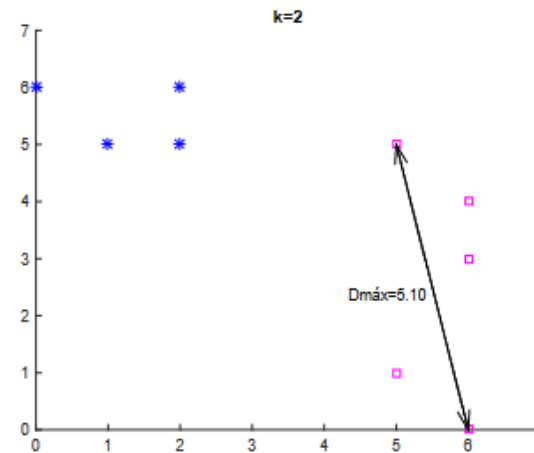
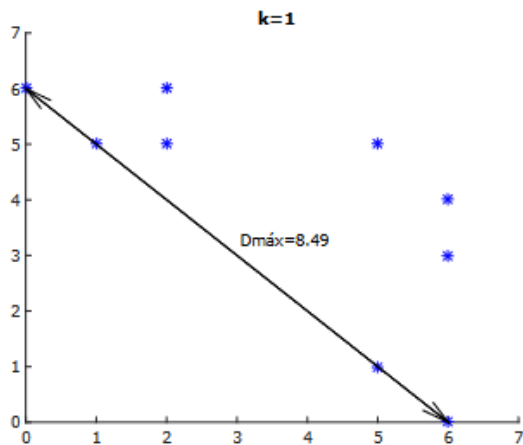
Características

- Basado en la idea de centroide
- Cada ejemplo se agrupa en función de la similitud con cada centroide
- Asume que las agrupaciones tienen forma esférica (hiperesferas)

Ejemplo ($k=3$)



Cómo determinar el valor de k



Contenido

- Técnicas de agrupamiento
 - Algoritmo k-medias
 - Agrupamiento jerárquico
 - Características
 - Estrategias aglomerativas
 - Algoritmo y ejemplo de uso
 - Dendrogramas
 - Algoritmo DBSCAN
 - Mapas auto-organizados
- Reducción de la dimensionalidad
 - Análisis de componentes principales

Características

- Busca construir jerarquías de agrupaciones
- Dos tipos de estrategias:
 - **Aglomerativa**: aproximación ascendente en el que cada ejemplo comienza siendo su propio agrupamiento y, en cada iteración (nivel de jerarquía), los pares de agrupaciones más similares son mezclados.
 - **Divisiva**: opera de forma inversa, aunque es mucho más complejo de implementar.

Estrategias aglomerativas

- Algoritmo (N ejemplos):

1. *A cada ejemplo se le asigna un agrupamiento, formando así N agrupamientos diferentes.*
2. *Se determina los dos agrupamientos más similares y éstos se fusionan en un único agrupamiento.*
3. $N \leftarrow N - 1$
4. *Si $N > 1$, se vuelve al paso 2.*

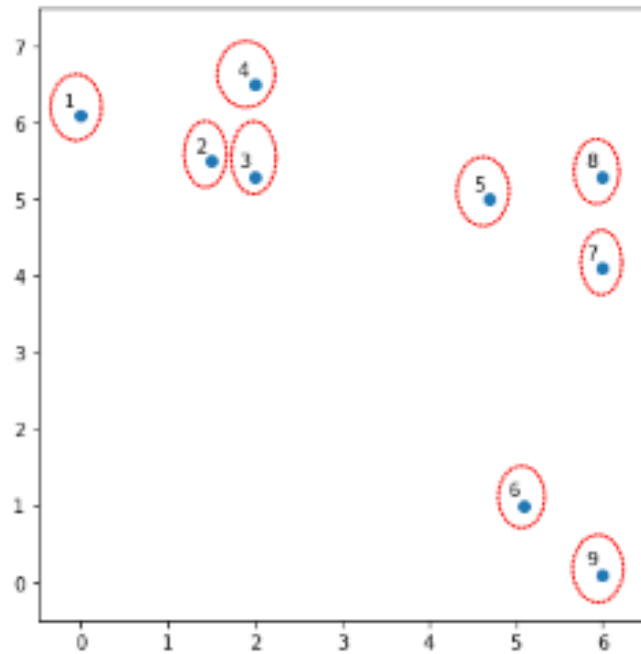
Ejemplo

- Un hostelero tiene la intención de crear un nuevo negocio consistente en abrir diferentes puestos de perritos calientes en la ciudad. Para ello, la información de la que dispone está formada por la ubicación U_i , expresada en términos de latitud y longitud, de la casa de cada una de las familias que conforman su cartera de clientes.
- Para simplificar los cálculos, considérese que las coordenadas de latitud y longitud se transforman a un sistema de coordenadas en el que las unidades del eje-x y eje-y se miden en kilómetros

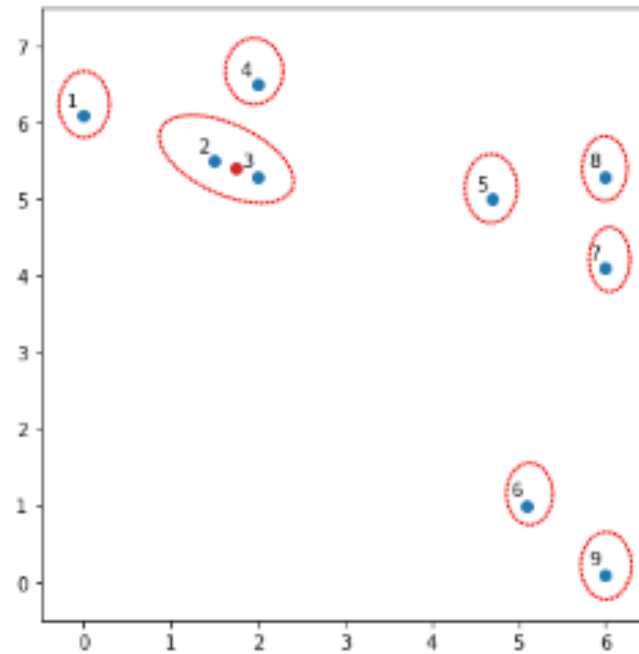
$$\begin{array}{lll} U_1 = (0.0, 6.1) & U_2 = (1.5, 5.5) & U_3 = (2.0, 5.3) \\ U_4 = (2.0, 6.5) & U_5 = (4.7, 5.0) & U_6 = (5.1, 1.0) \\ U_7 = (6.0, 4.1) & U_8 = (6.0, 5.3) & U_9 = (6.0, 0.1) \end{array}$$

- Tres casos de uso:
 1. Determinar el número óptimo de puestos, junto con sus ubicaciones.
 2. Determinar el número máximo de puestos, y sus ubicaciones, si la distancia entre ellos nunca debe ser menor de 4.5Km.
 3. Si sólo se dispone del presupuesto necesario para montar cuatro puestos, determinar sus ubicaciones.

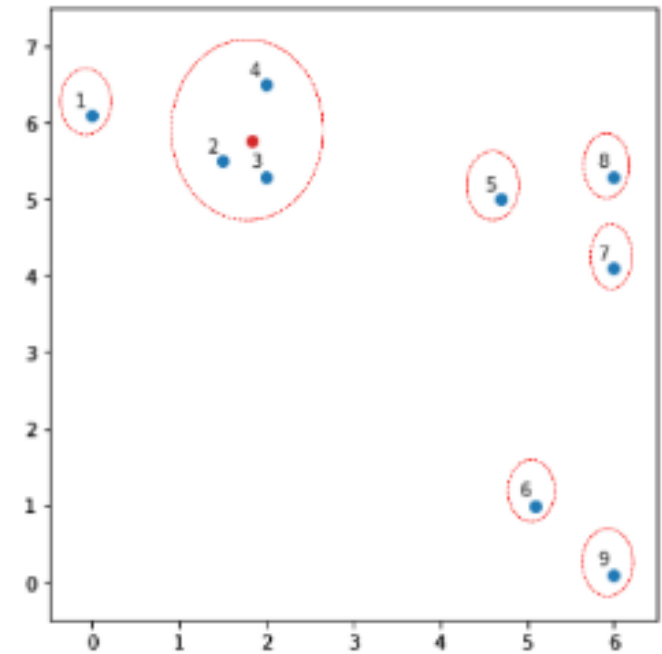
Ejemplo



Asignación inicial

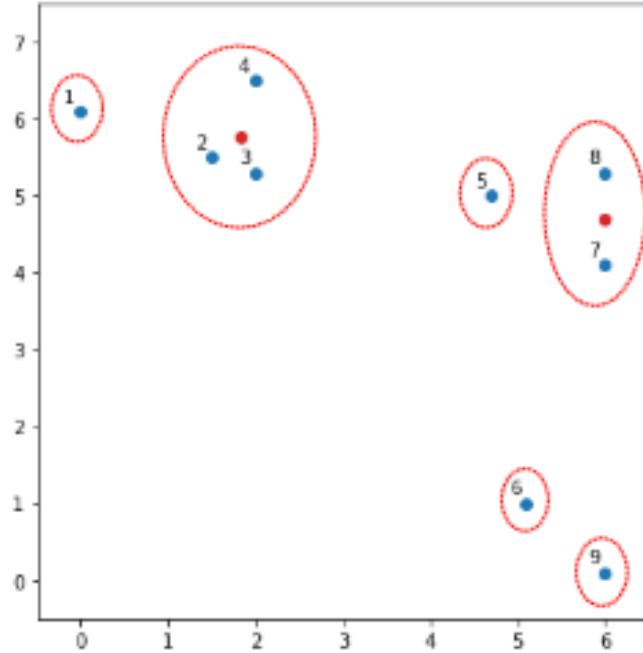


iteración 1

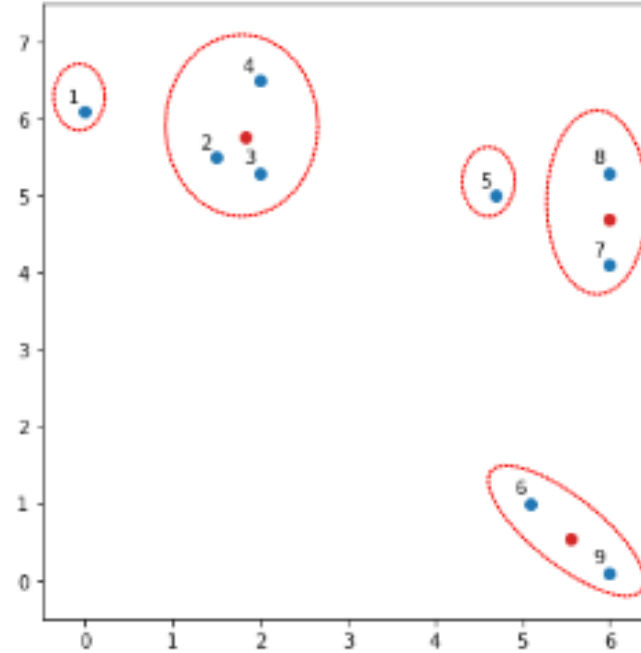


iteración 2

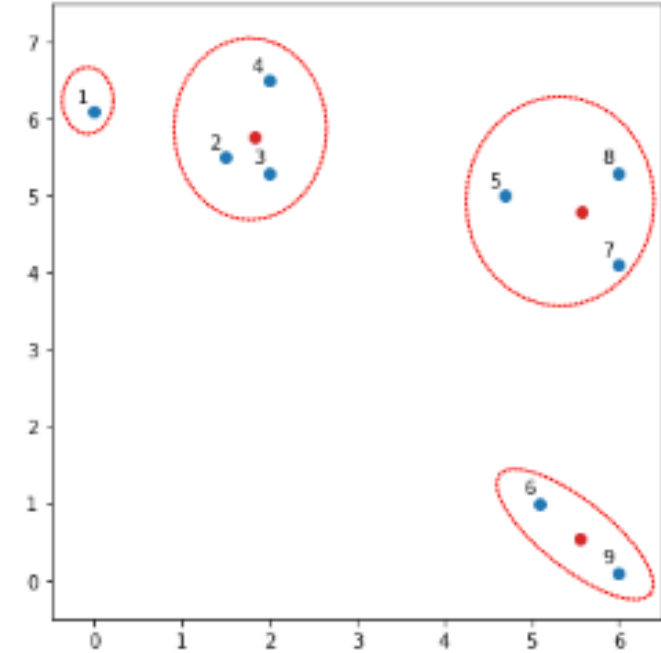
Ejemplo



iteración 3

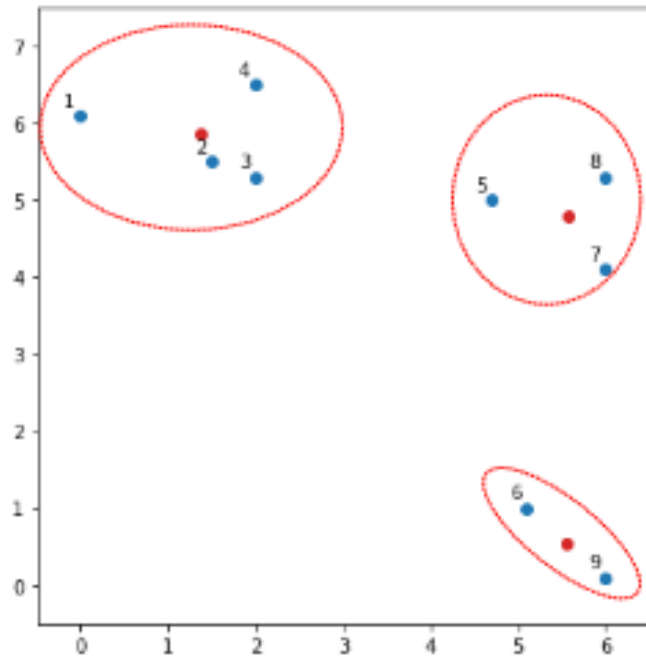


iteración 4

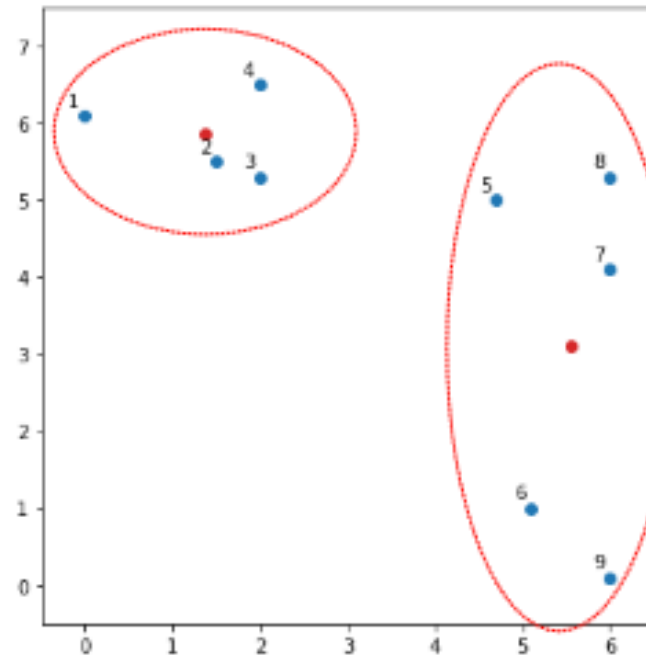


iteración 5

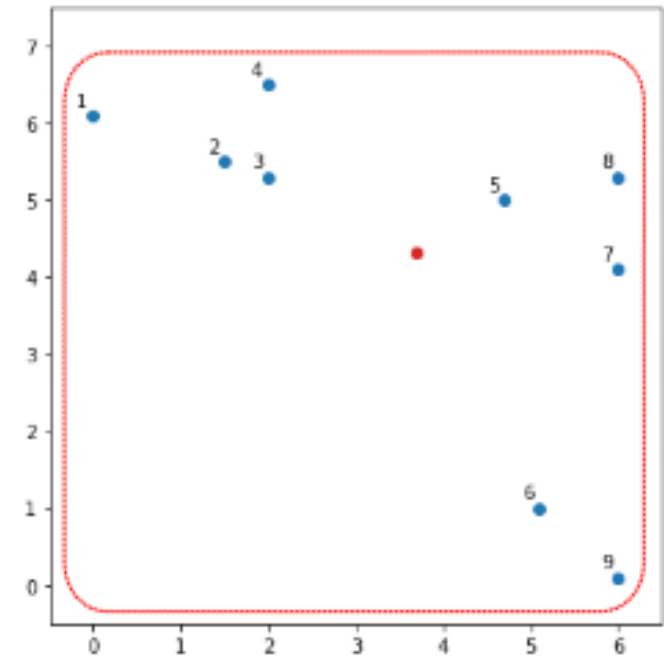
Ejemplo



iteración 6

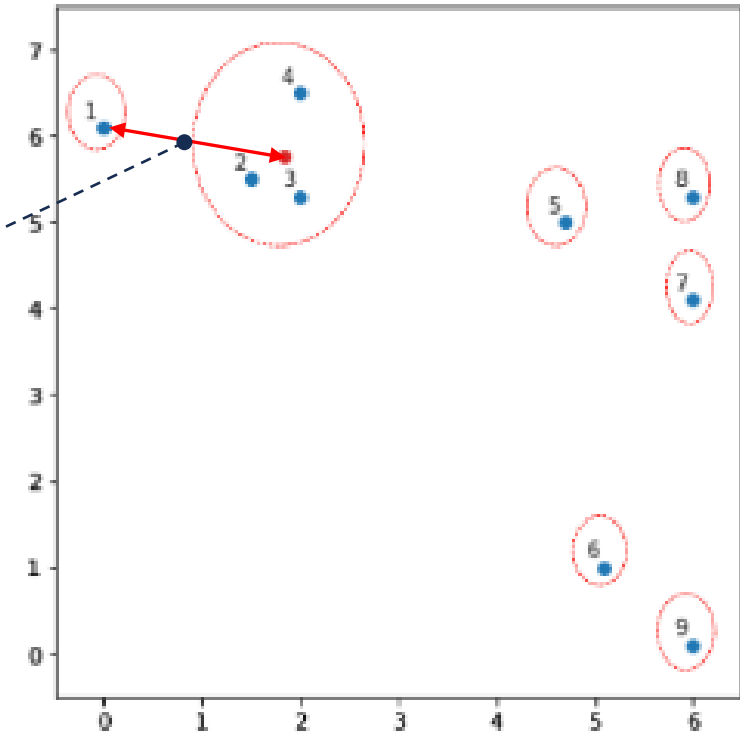
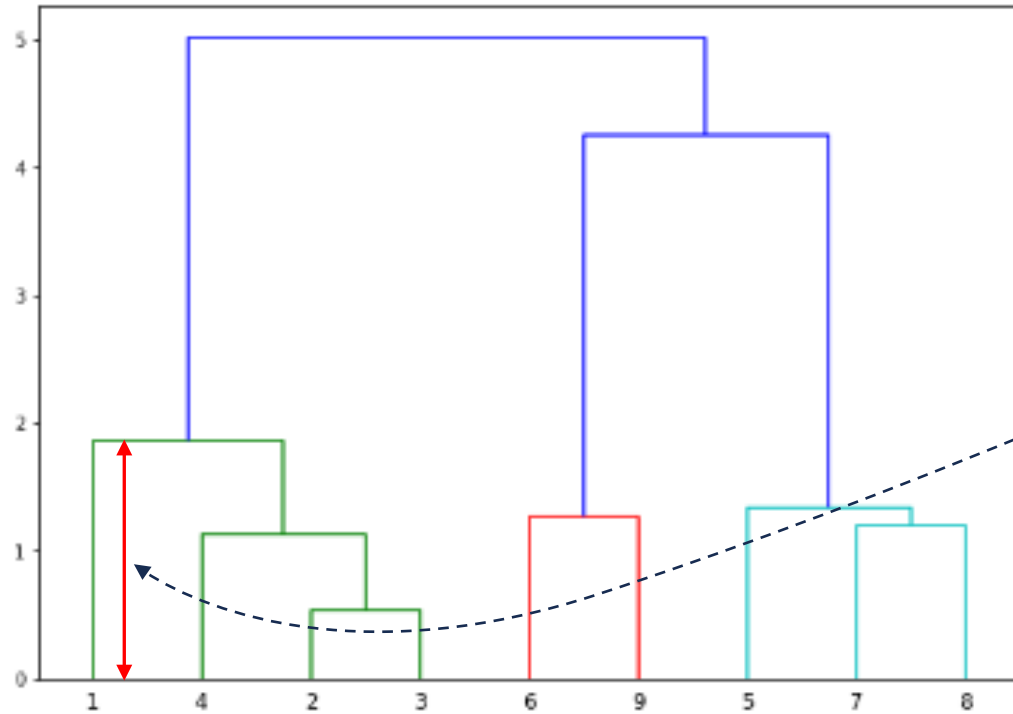


iteración 7



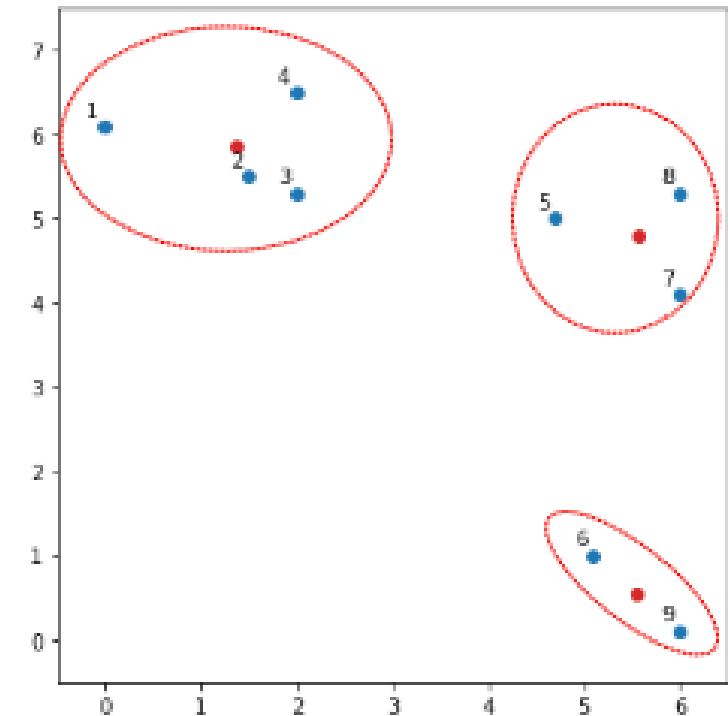
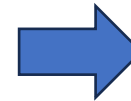
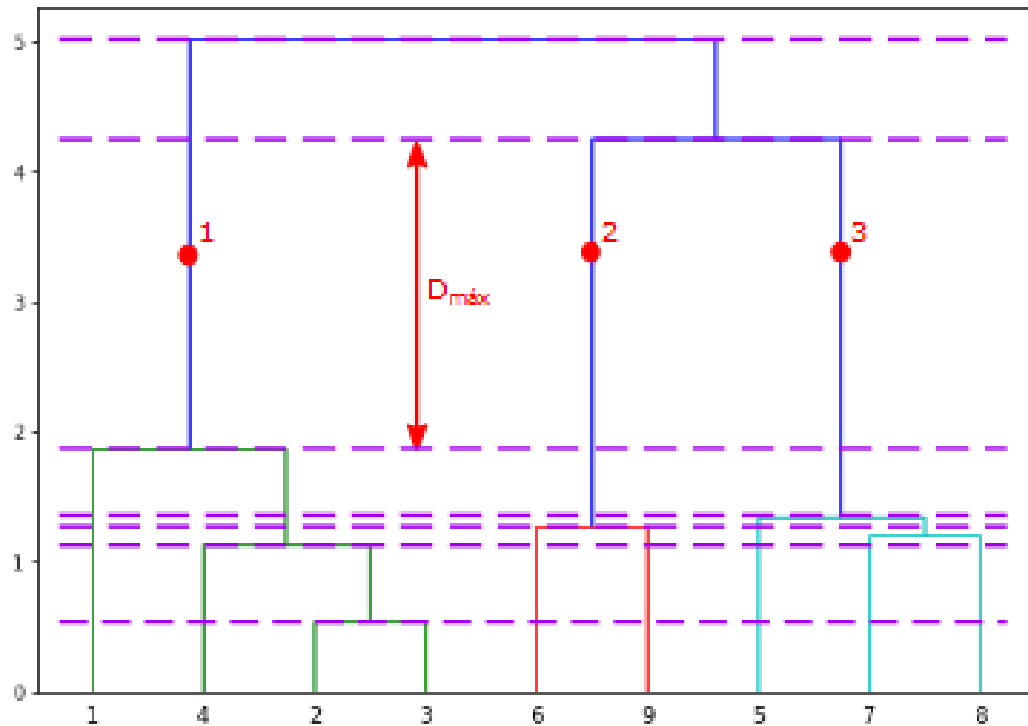
iteración 8

Construcción de un dendrograma



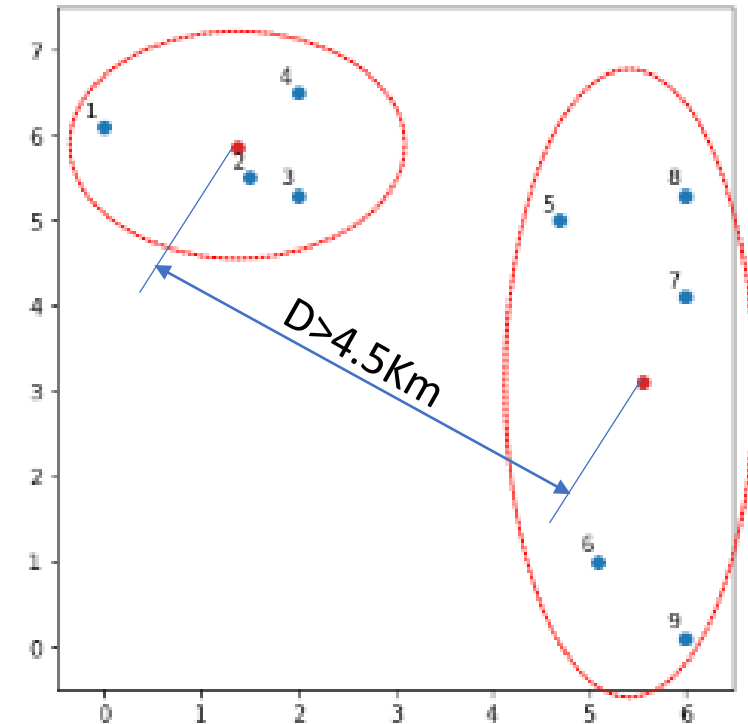
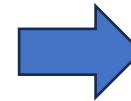
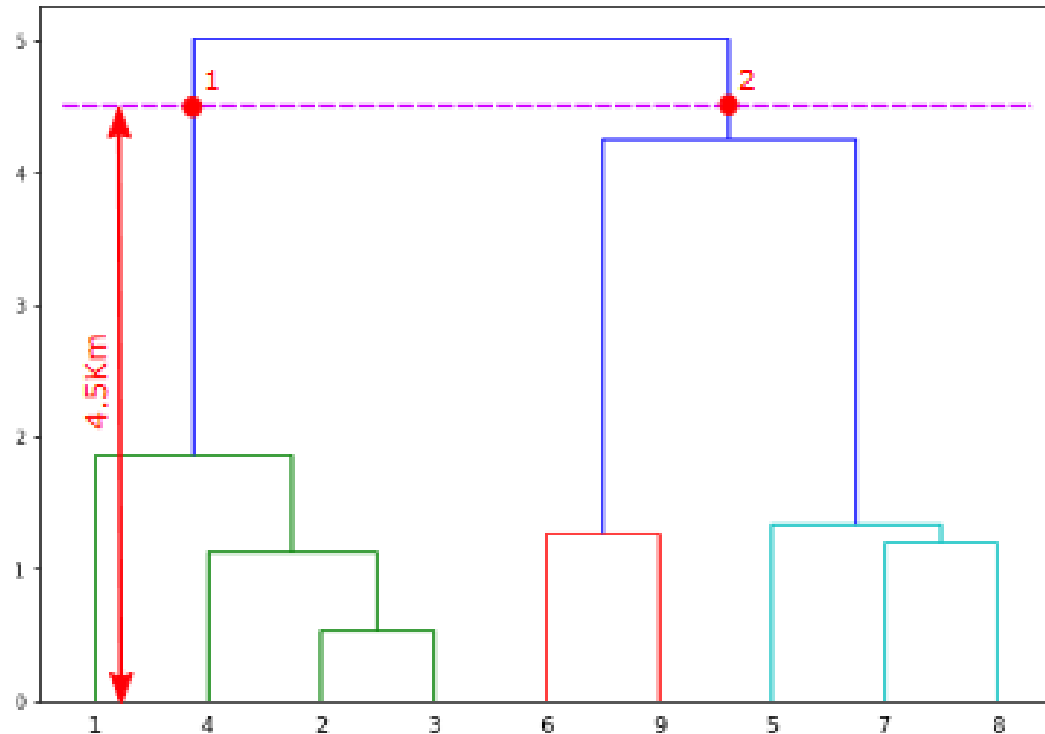
Uso del dendrograma (caso 1)

Número óptimo de puestos de perritos calientes



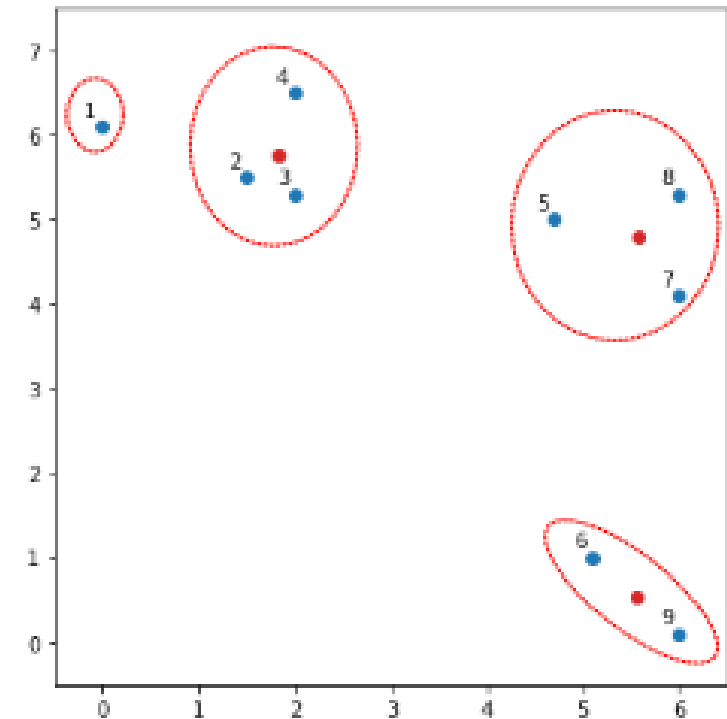
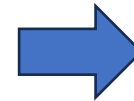
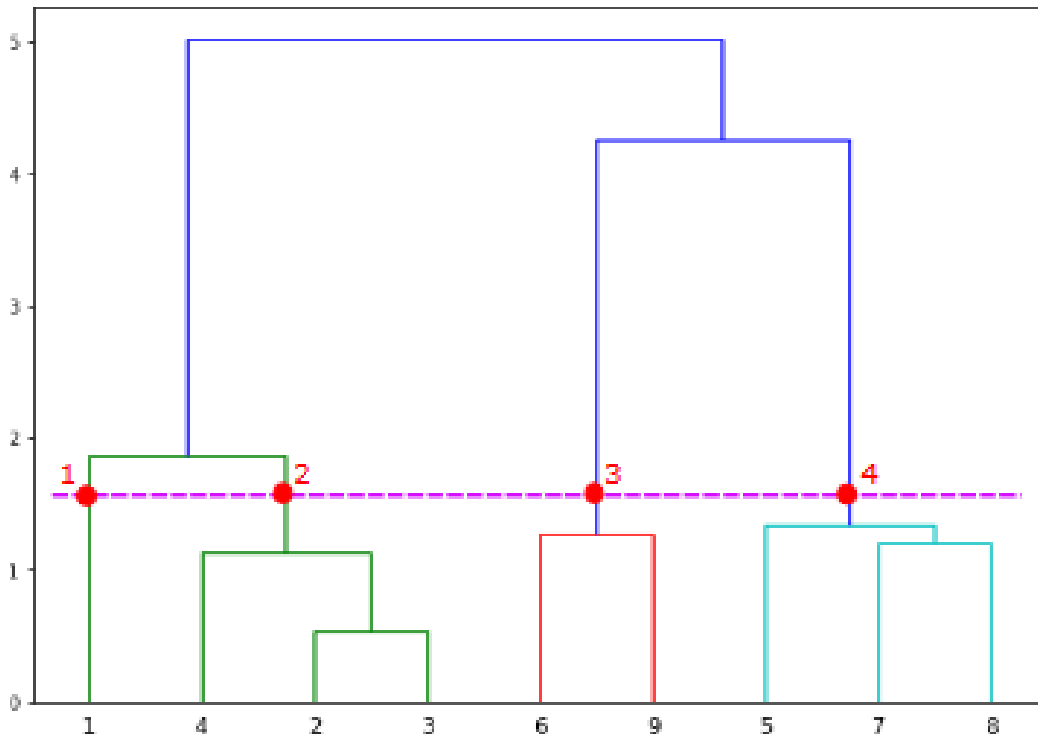
Uso del dendrograma (caso 2)

Número máximo de puestos, sabiendo que la distancia entre ellos nunca debe ser menor de 4.5Km.



Uso del dendrograma (caso 3)

El presupuesto sólo permite montar cuatro puestos



Contenido

- Técnicas de agrupamiento
 - Algoritmo k-medias
 - Agrupamiento jerárquico
 - Algoritmo DBSCAN
 - Características
 - Algoritmo y ejemplo de uso
 - Mapas auto-organizados
- Reducción de la dimensionalidad
 - Análisis de componentes principales

Características

- DBSCAN (*Density-Based Spatial Clustering of Applications with Noise*) define los agrupamientos como regiones continuas de alta densidad.
- Separa las regiones considerando las áreas de baja densidad.
- En DBSCAN existen tres tipos de puntos:
 - Un punto P es un **punto-núcleo** si al menos N_{min} puntos (incluido él mismo) están en su vecindad- ϵ . Se dice que esos puntos son directamente alcanzables desde P.
 - Un punto F es un **punto-frontera** si no es un punto núcleo, pero tiene al menos un punto-núcleo en su vecindad- ϵ .
 - Un punto R es un **punto-ruidoso** si no tiene al menos un punto-núcleo en su vecindad- ϵ .

Algoritmo

Algoritmo “ingenuo”

FOR cada punto P del conjunto original
Etiqueta P como punto núcleo, frontera o ruidoso

cluster_actual := 0

FOR cada punto-núcleo Q

IF Q no tiene etiqueta_cluster **THEN**

cluster_actual := cluster_actual + 1;

cluster_Q := cluster_actual

FOR cada punto V de la vecindad(Q)

IF cluster_V = $\emptyset$ **THEN**

cluster_V := cluster_actual

IF V es un punto-núcleo **THEN**

vecindad(Q) := vecindad(Q) $\cup$ vecindad(V)

Algoritmo

Algoritmo “ingenuo”

FOR cada punto P del conjunto original
Etiqueta P como punto núcleo, frontera o ruidoso

cluster_actual := 0

FOR cada punto-núcleo Q

IF Q no tiene etiqueta_cluster **THEN**

cluster_actual := cluster_actual + 1;

cluster_Q := cluster_actual

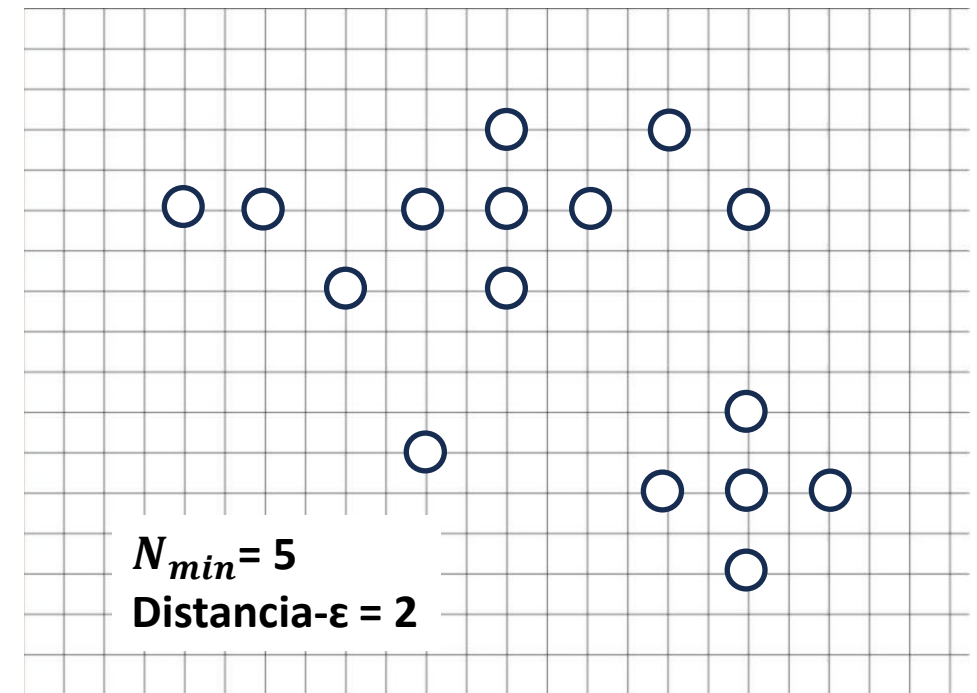
FOR cada punto V de la vecindad(Q)

IF cluster_V = $\emptyset$ **THEN**

cluster_V := cluster_actual

IF V es un punto-núcleo **THEN**

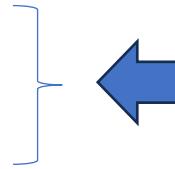
vecindad(Q) := vecindad(Q) $\cup$ vecindad(V)



Ejemplo de uso

Algoritmo “ingenuo”

FOR cada punto P del conjunto original
Etiqueta P como punto núcleo, frontera o ruidoso



cluster_actual := 0

FOR cada punto-núcleo Q

IF Q no tiene etiqueta_cluster **THEN**

cluster_actual := cluster_actual + 1;

cluster_Q := cluster_actual

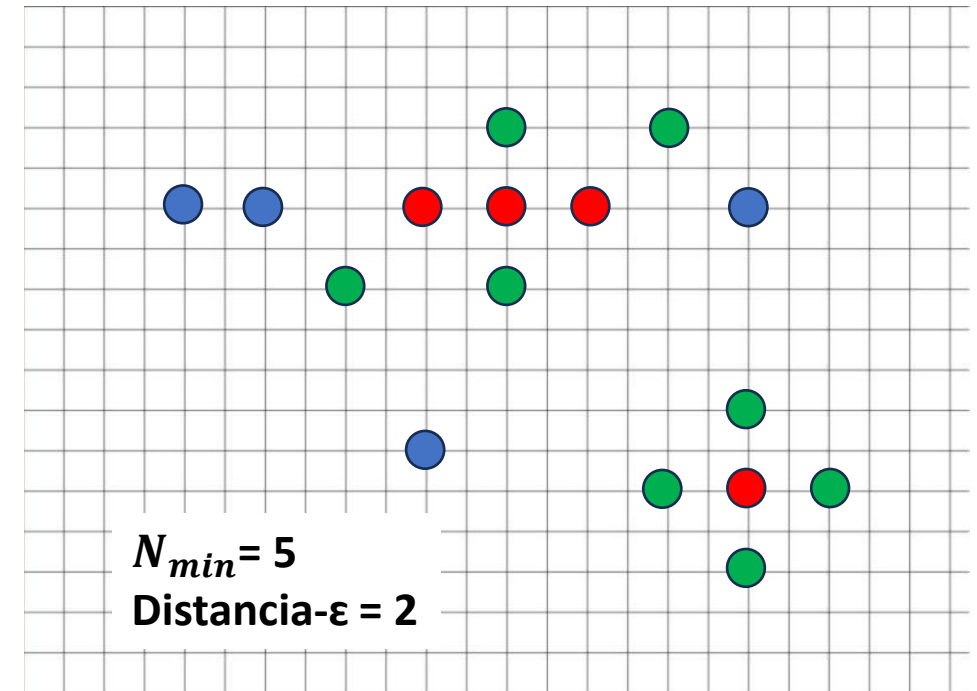
FOR cada punto V de la vecindad(Q)

IF cluster_V = $\emptyset$ **THEN**

cluster_V := cluster_actual

IF V es un punto-núcleo **THEN**

vecindad(Q) := vecindad(Q) $\cup$ vecindad(V)



- Punto núcleo
- Punto frontera
- Punto ruidoso

Ejemplo de uso

Algoritmo “ingenuo”

FOR cada punto P del conjunto original
Etiqueta P como punto núcleo, frontera o ruidoso

cluster_actual := 0

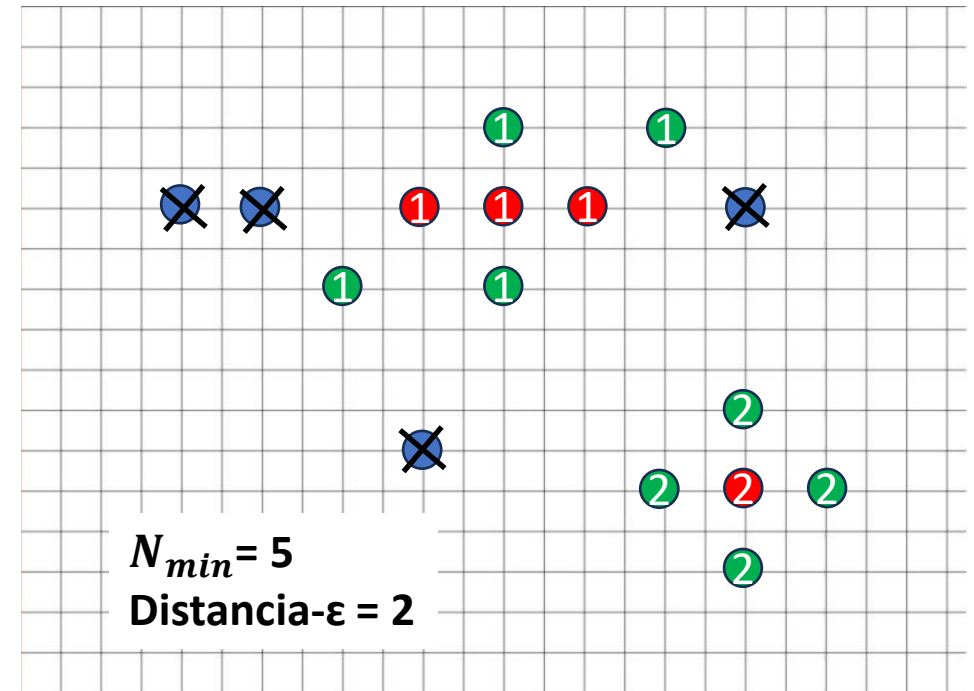
FOR cada punto-núcleo Q

IF Q no tiene etiqueta_cluster **THEN**
cluster_actual := cluster_actual + 1;
cluster_Q := cluster_actual

FOR cada punto V de la vecindad(Q)

IF cluster_V = $\emptyset$ **THEN**
cluster_V := cluster_actual

IF V es un punto-núcleo **THEN**
vecindad(Q) := vecindad(Q) $\cup$ vecindad(V)



Ejemplo de uso

Algoritmo “ingenuo”

FOR cada punto P del conjunto original
Etiqueta P como punto núcleo, frontera o ruidoso

cluster_actual := 0

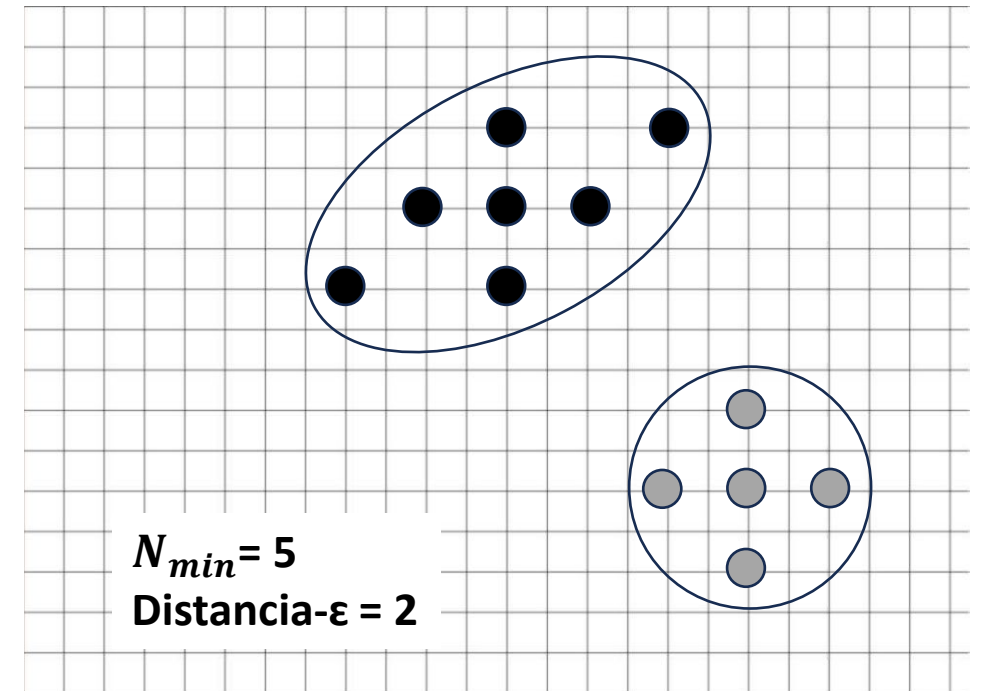
FOR cada punto-núcleo Q

IF Q no tiene etiqueta_cluster **THEN**
cluster_actual := cluster_actual + 1;
cluster_Q := cluster_actual

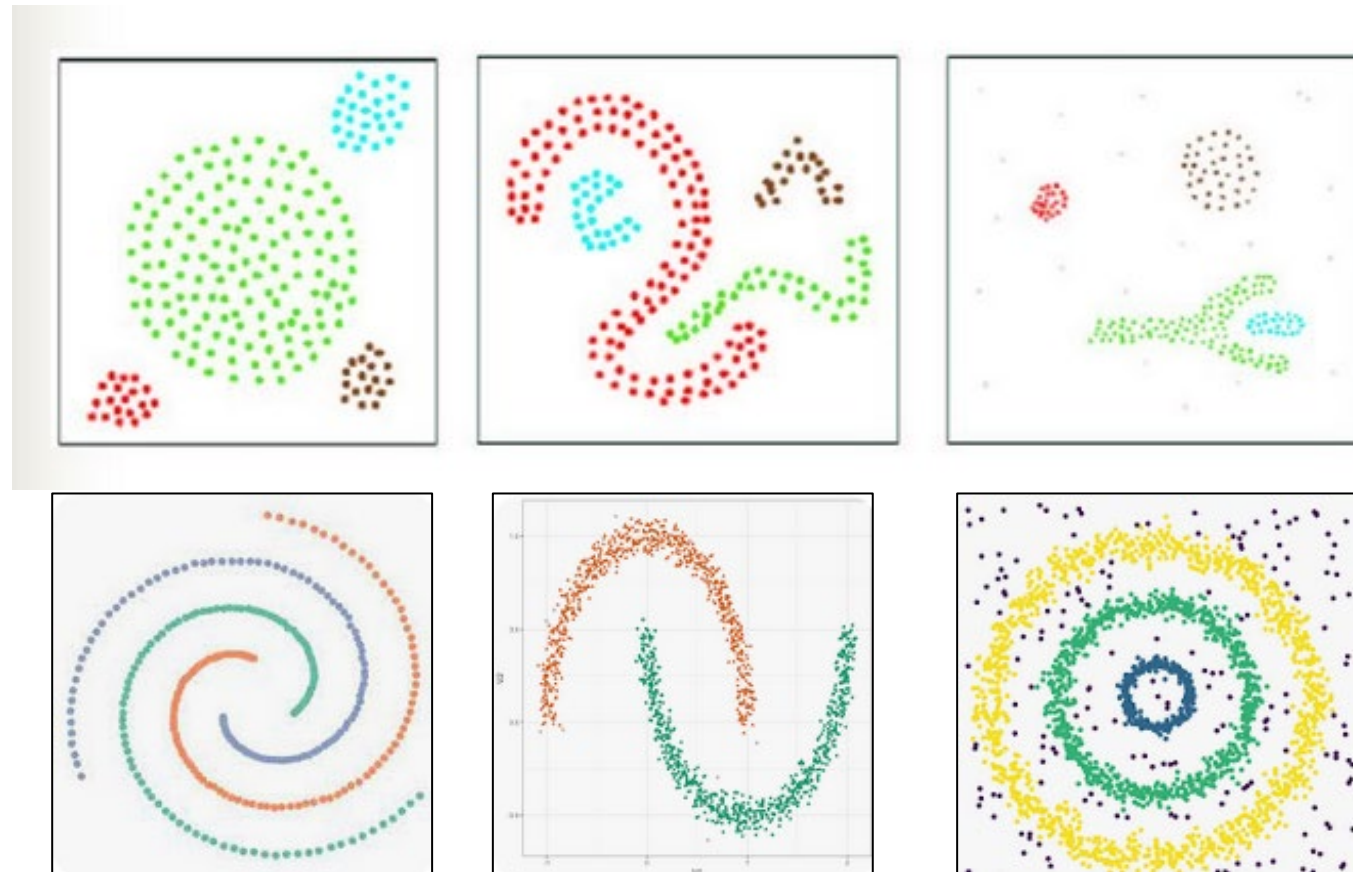
FOR cada punto V de la vecindad(Q)

IF cluster_V = $\emptyset$ **THEN**
cluster_V := cluster_actual

IF V es un punto-núcleo **THEN**
vecindad(Q) := vecindad(Q) $\cup$ vecindad(V)



Puede modelar grupos de formas muy diversas

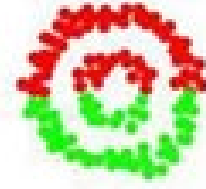
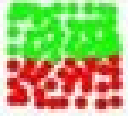


Estrateg. aglomerativas vs. k-medias vs. DBSCAN

Características	E. AGLOMERATIVAS	K-MEDIAS	DBSCAN
Hay que especificar el número de grupos	No 	Sí 	No 
Encontrar grupos de formas muy diversas	No 	No 	Sí 
Maneja el ruido (outliers)	No 	No 	Sí 
Número de hiperparámetros	0 	k 	(N_{min}, ϵ) 
Es determinista	Sí 	No	No (puntos-frontera)
Sensible a la métrica de distancia	Sí	Sí	Sí
Escala bien con el tamaño de datos	No 	Sí 	Sí 
Es intuitivo	No	Sí 	No

Estrateg. aglomerativas vs. k-medias vs. DBSCAN

Características	E. AGLOMERATIVAS	K-MEDIAS	DBSCAN
Hay que especificar el número de grupos			
Encontrar grupos de formas muy diversas			
Maneja el ruido (outliers)			
Número de hiperparámetros			
Es determinista			
Sensible a la métrica de distancia			
Escala bien con el tamaño de datos			

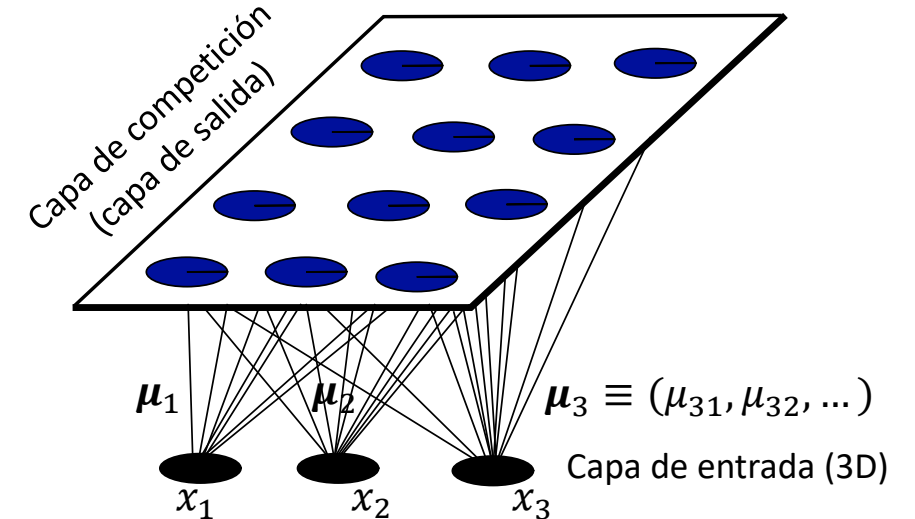
			The original set
			DBSCAN
			K-means

Contenido

- Técnicas de agrupamiento
 - Algoritmo k-medias
 - Agrupamiento jerárquico
 - Algoritmo DBSCAN
 - Mapas auto-organizados
 - Características
 - Algoritmo y ejemplo de uso
- Reducción de la dimensionalidad
 - Análisis de componentes principales

Mapas auto-organizados

- También llamados redes de Kohonen
- Un mapa auto-organizado es un tipo de RNA de dos capas
- Produce una representación discreta del espacio de ejemplos de entrada (mapa), que preserva las propiedades topológicas de dicho espacio



Algoritmo básico y ejemplo de uso

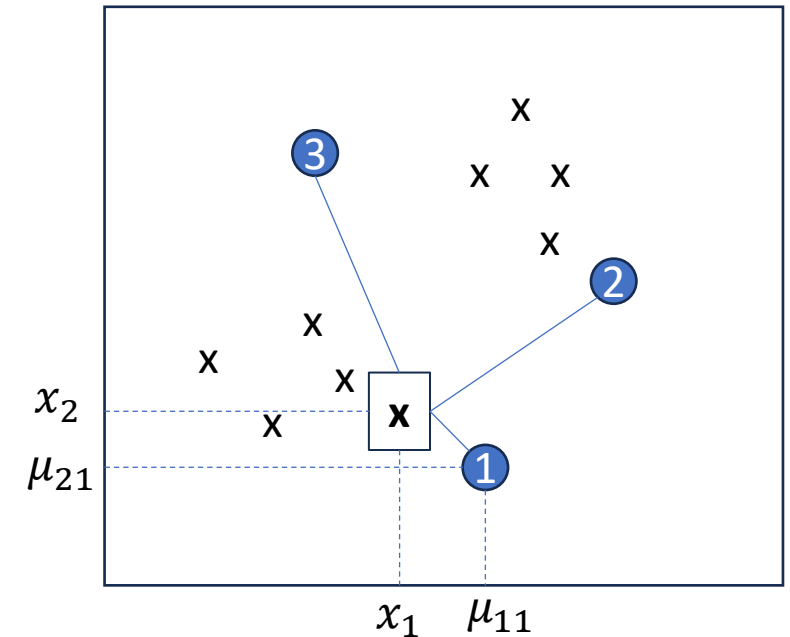
FOR cada ejemplo $\mathbf{x} \equiv (x_i, \dots, x_d)$ del conjunto de entrenamiento

➡ Calcular la distancia de $\mathbf{x}$ a cada neurona de la capa de salida
Elegir como ganadora la neurona C_j que está a menor distancia

Actualizar sólo los pesos de C_j :

$$\Delta\mu_{ij} = \alpha(t)(x_i - \mu_{ij}) \quad \text{Ley de Hebb}$$

Ejemplo en 2D con 3 neuronas



Algoritmo básico y ejemplo de uso

FOR cada ejemplo $\mathbf{x} \equiv (x_i, \dots, x_d)$ del conjunto de entrenamiento

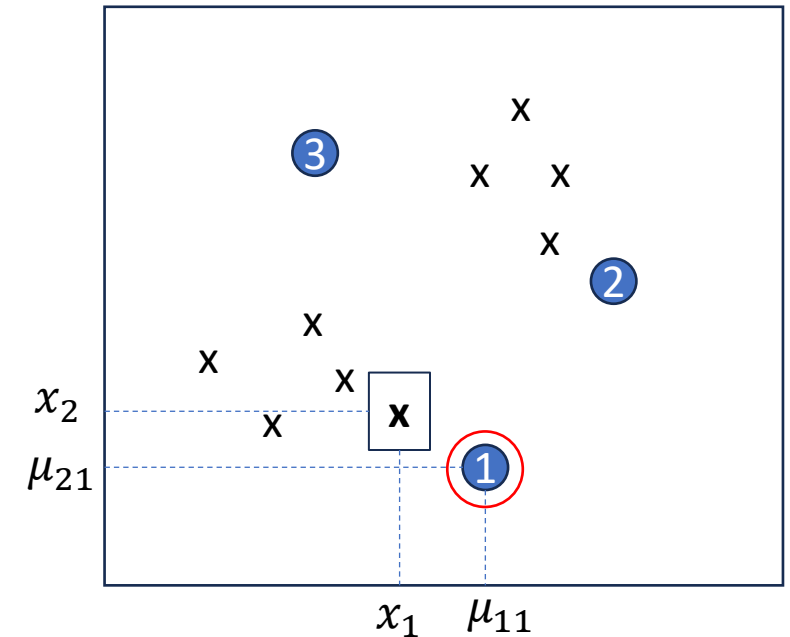
Calcular la distancia de $\mathbf{x}$ a cada neurona de la capa de salida

➡ Elegir como ganadora la neurona C_j que está a menor distancia

Actualizar sólo los pesos de C_j :

$$\Delta\mu_{ij} = \alpha(t)(x_i - \mu_{ij}) \quad \text{Ley de Hebb}$$

Ejemplo en 2D con 3 neuronas

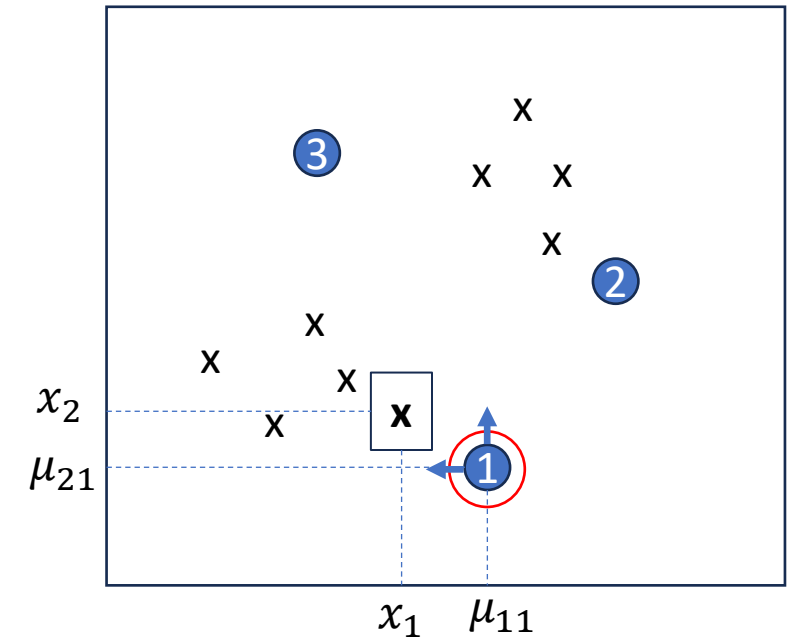


Algoritmo básico y ejemplo de uso

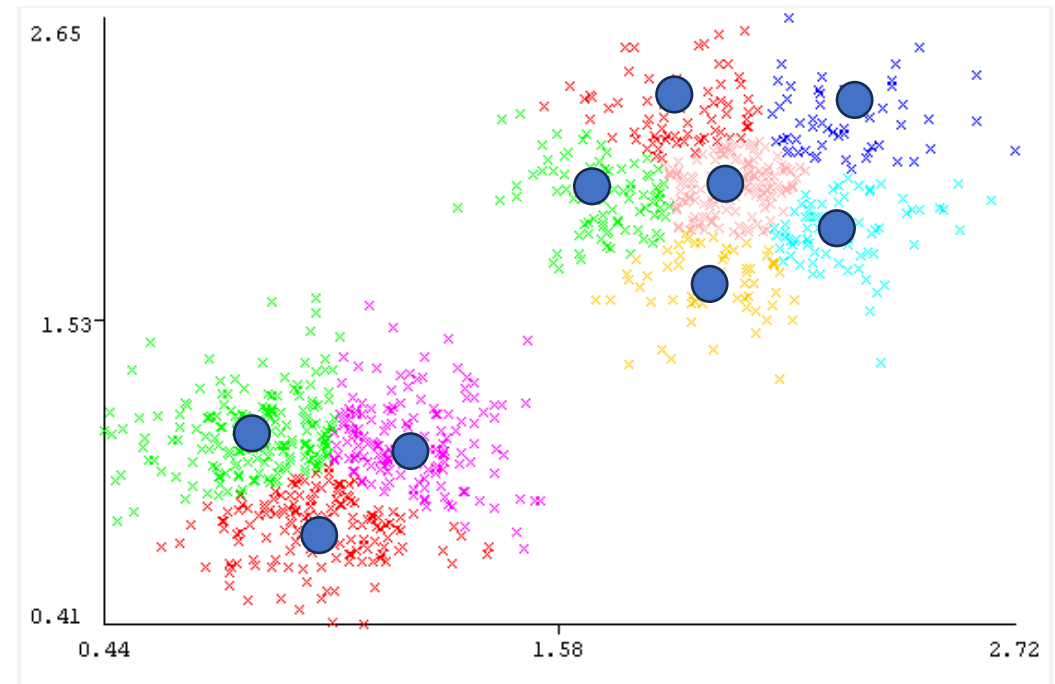
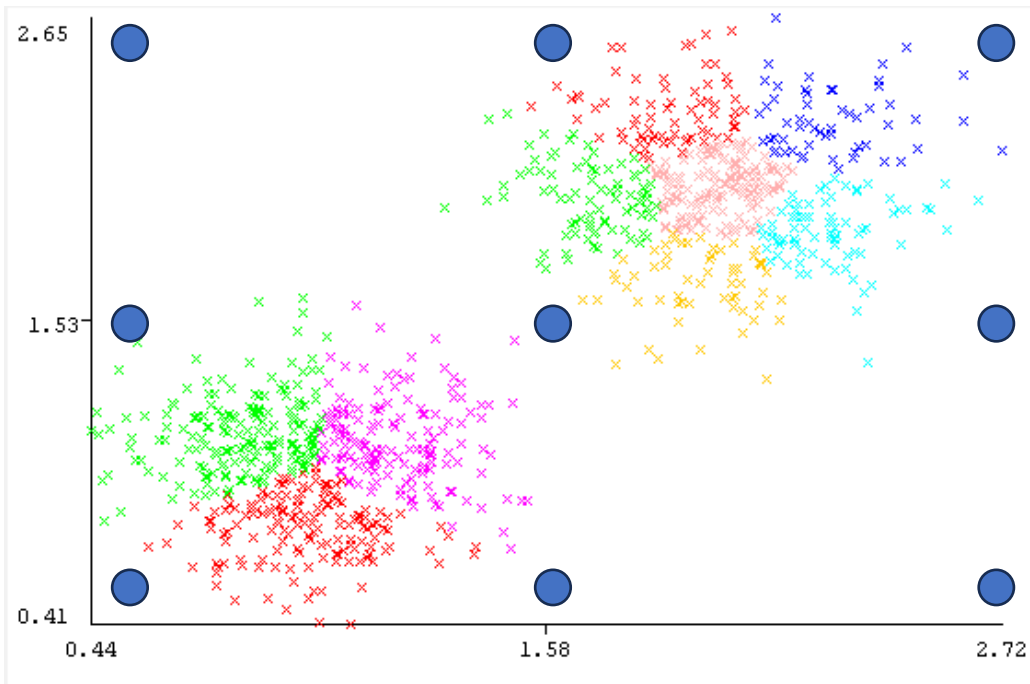
FOR cada ejemplo $\mathbf{x} \equiv (x_i, \dots, x_d)$ del conjunto de entrenamiento
 Calcular la distancia de $\mathbf{x}$ a cada neurona de la capa de salida
 Elegir como ganadora la neurona C_j que está a menor distancia
➡ Actualizar sólo los pesos de C_j :

$$\Delta\mu_{ij} = \alpha(t)(x_i - \mu_{ij}) \quad \text{Ley de Hebb}$$

Ejemplo en 2D con 3 neuronas



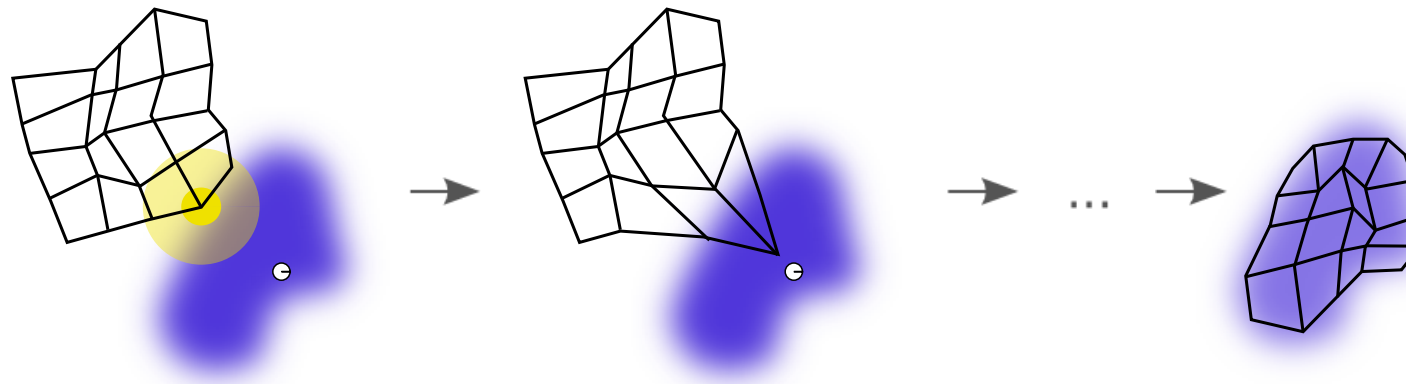
Ejemplo: mapa de 3x3 neuronas y 2 clases



Idea de vecindario

Se puede añadir la idea de vecindario de neuronas, siguiendo algún tipo de estructura espacial (por ejemplo, una malla):

$$\Delta\mu_{ij} = \frac{\alpha(t)}{1+d(C_i, C_j)} (x_i - \mu_{ij}) \quad \forall C_j \text{ tq } d(C_i, C_j) \leq \theta \text{ y } C_i \text{ la neurona ganadora}$$



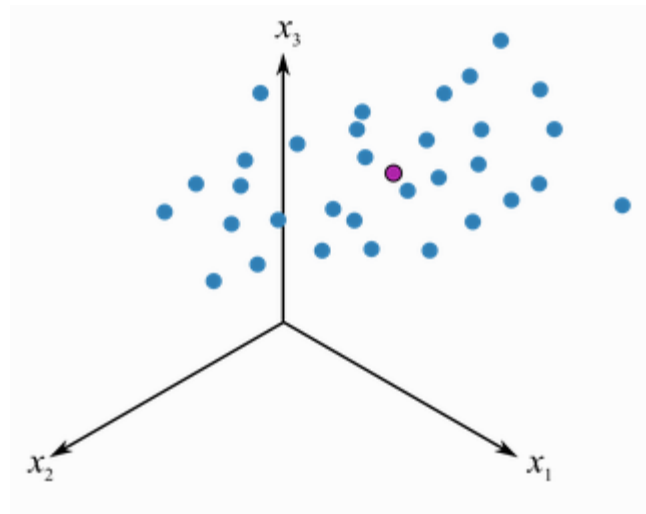
Contenido

- Técnicas de agrupamiento
 - Algoritmo k-medias
 - Agrupamiento jerárquico
 - Algoritmo DBSCAN
 - Mapas auto-organizados
- Reducción de la dimensionalidad
 - Análisis de componentes principales (PCA)

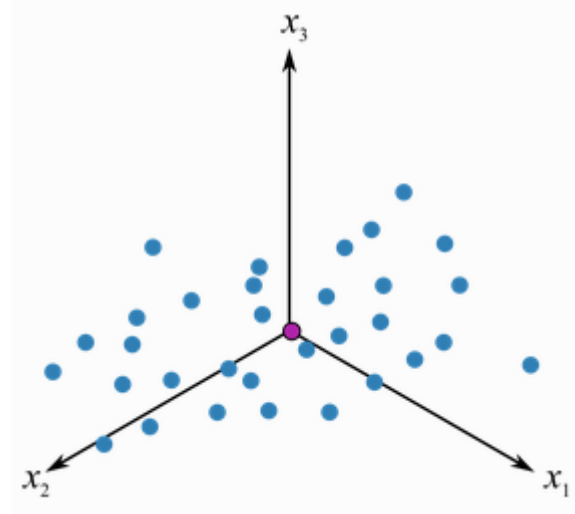
Análisis de componentes principales (PCA)

- Es una técnica para reducir la dimensionalidad de un conjunto de datos
- La idea es aplicar una transformación lineal que lleve los datos originales a un nuevo sistema de coordenadas donde la mayor parte de la variación en los datos se puede describir con menos dimensiones
- Cada nueva dimensión recibe el nombre de *componente principal i-ésima*.
- Utilidad:
 - Se pueden usar las dos (tres) primeras componentes principales para visualizar los datos en dos (tres) dimensiones e identificar visualmente grupos de datos relacionados
 - Para mejorar las prestaciones de algoritmos cuya operativa está basada en el uso de distancias

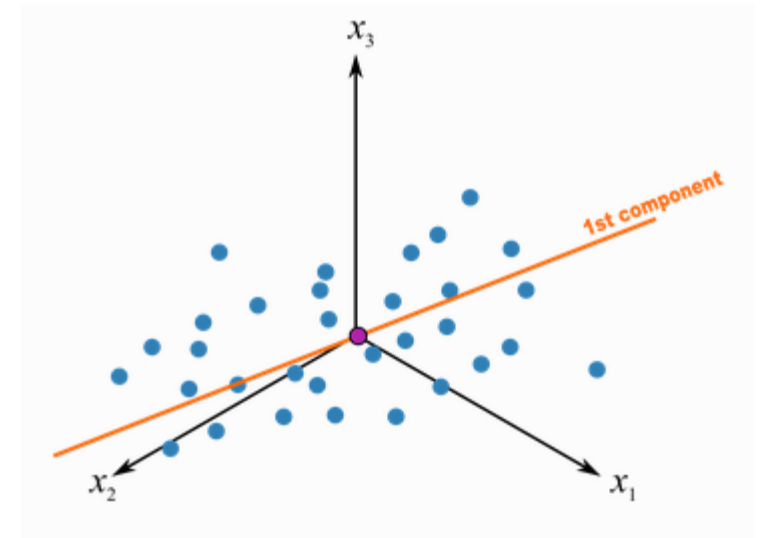
Interpretación geométrica de PCA



Datos originales

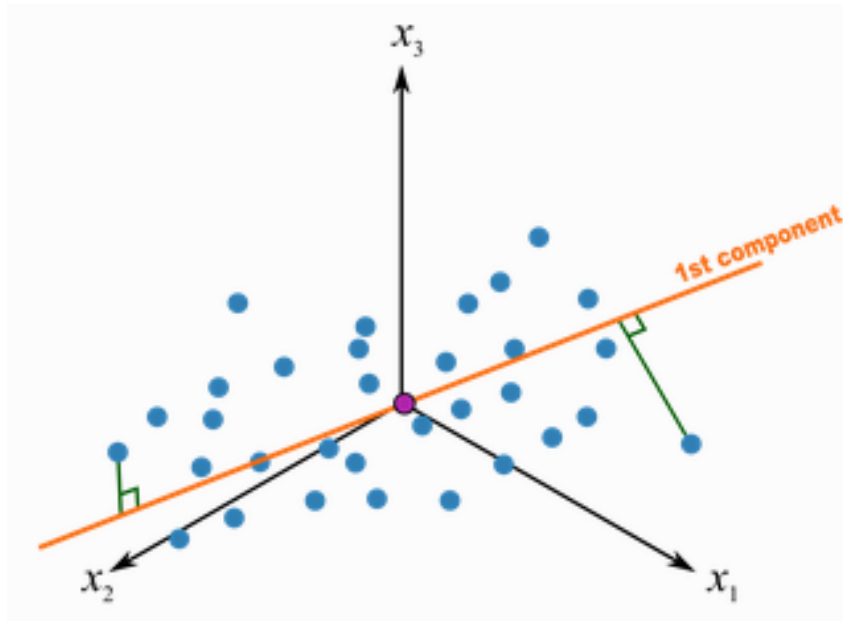


1. Se centran los datos usando la media

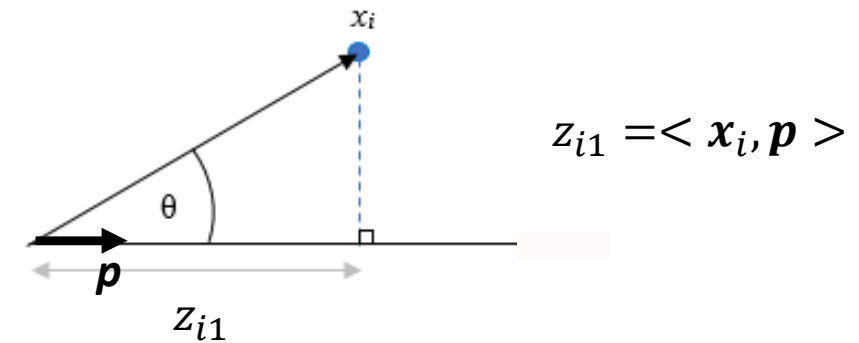


2. Se calcula la 1ª componente (Z1)
Se busca la línea que mejor se ajusta a los datos y se representa por el vector unitario $\mathbf{p}$ que la define

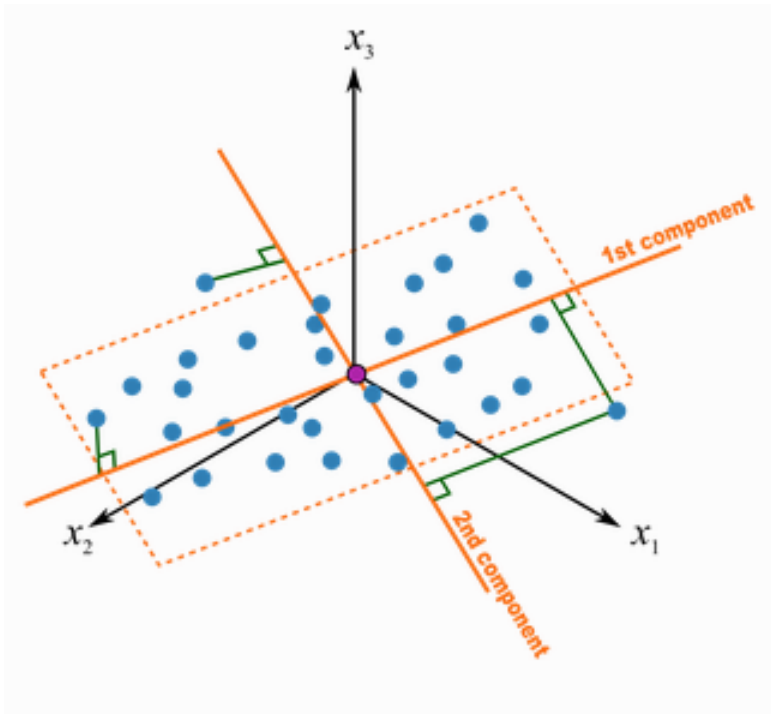
Interpretación geométrica de PCA



3. Se proyecta cada punto en el nuevo eje y se calcula la distancia desde el origen a cada proyección



Interpretación geométrica de PCA



4. Se calcula la 2ª componente (Z2)

- Pasará por el origen y es perpendicular a la primera
- Se proyectan los puntos sobre el nuevo eje y se calculan las distancias de estas proyecciones al origen

5. Los ejes Z1 y Z2 definen un nuevo sistema de coordenadas

Los pares de puntos (z_{i1}, z_{i2}) representan las nuevas coordenadas de los puntos originales en ese espacio de dimensión menor.

PCA, autovalores y autovectores

- Se calcula la matriz **V** de autovectores que diagonaliza la matriz de covarianza **C** asociada a la matriz de datos previamente centrados **B**:

$$\mathbf{V}^{-1}\mathbf{C}\mathbf{V} = \mathbf{D}$$

donde:

- **D** es la matriz diagonal de autovalores de la matriz de covarianza
- Cada uno de los vectores columna de **V** representa un autovector de **C**
- Cada autovalor en la columna i-ésima de **D** corresponde al autovector de la misma columna en **V**
- Se ordenan las columnas de **V** y **D** en orden de autovalor decreciente
 - Los autovectores forman una nueva base para los datos (las componentes principales)
 - Cada autovalor normalizado representará la proporción de varianza explicada por su respectivo autovector
- Se selecciona un subconjunto de autovectores como vectores base en la matriz **W**, tantos como varianza total se quiera explicar (suma de autovalores normalizados asociados a los autovectores elegidos)
- Finalmente, se proyectan los datos originales centrados, **B**, sobre la nueva base:

$$\mathbf{T} = \mathbf{B}\mathbf{W}$$

