



DEPARTAMENTO  
DE COMPUTACION

Facultad de Ciencias Exactas y Naturales - UBA

# Laboratorio de datos

## Aprendizaje No Supervisado

Verano 2025

# Aprendizaje Supervisado

Hasta ahora las observaciones tenían variables predictoras y un valor a predecir conocido (etiquetas).

Ejemplo:



The diagram illustrates a supervised learning process. On the left, a table titled "Entrenamiento" contains five rows of data with columns RU and DI. On the right, a table titled "Predicción" contains three rows of data with columns RU and DI. A curved arrow points from the "Entrenamiento" table to the "Predicción" table, indicating that the training data is used to make predictions for new data.

Entrenamiento		
	RU	DI
1	0	104.00
2	50	106.00
3	100	112.30
4	200	117.00
5		

Predicción		
	RU	DI
1	25	
2	600	
3	1500	

# Aprendizaje No Supervisado

No contamos datos de entrenamiento etiquetados. Se trata de descubrir información y estructura implícita en los datos sin ninguna guía externa. Es decir, descubrir patrones o estructuras ocultas en los datos sin la presencia de etiquetas o respuestas predefinidas.

Sirve para entender, resumir, relacionar y visualizar los datos.

Estrategias que veremos:

- Clustering - agrupamiento
- Reducción de la dimensión

# Herramientas de aprendizaje no supervisado

## **Clustering - Agrupamiento**

Métodos para encontrar subgrupos homogéneos dentro del conjunto entero de los datos.

## **Reducción de dimensionalidad**

Métodos para proyectar los datos -en general de dimensiones altas- en un espacio de menor dimensión, que haga posible su manipulación (o visualización) pero preserve las características del conjunto original. Suele usarse también como paso previo al clustering.

# Agrupados

¿Cuáles de esas características tuvieron en cuenta para realizar el agrupamiento?

# Agrupados - Grupo 1

- Color
- Tamaño



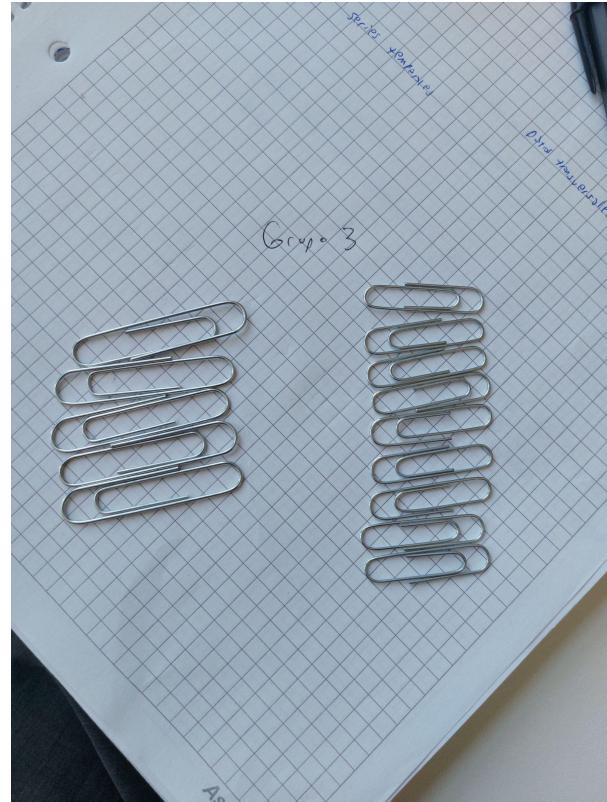
# Agrupados - Grupo 2

- Color
- Tamaño



# Agrupados - Grupo 3

- Tamaño



# Agrupados - Grupo 4

- Color
- Tamaño



# Agrupados - Grupo 5

- Color



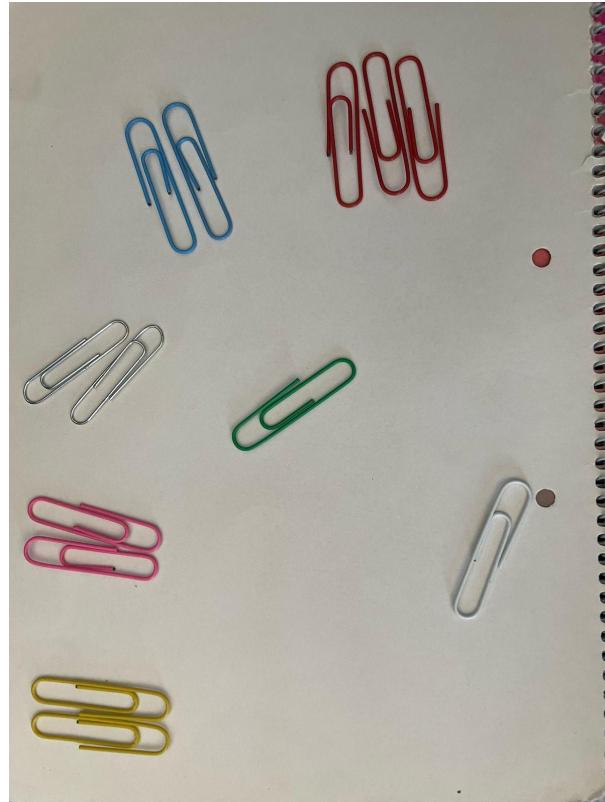
# Agrupados - Grupo 6

- Color
- Tamaño



# Agrupados - Grupo 7

- Color



# Agrupados - Grupo 8

- Color/material
- Tamaño



# ¿Qué tipos de clasificaciones tuvimos?

- por color
- por material
- por tamaño
- por alto

**La selección de atributos nos condiciona el agrupamiento.**

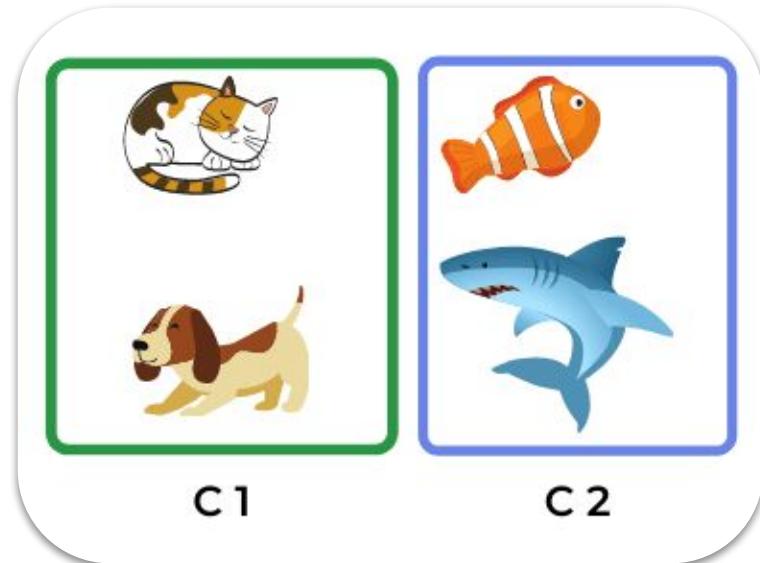
¿CUÁL ES EL AGRUPAMIENTO

CORRECTO?



Objetivo: Agrupar los datos en función de sus características de modo tal que

- Los datos que pertenecen a un mismo grupo son similares.
- Los de distintos grupos no son similares.

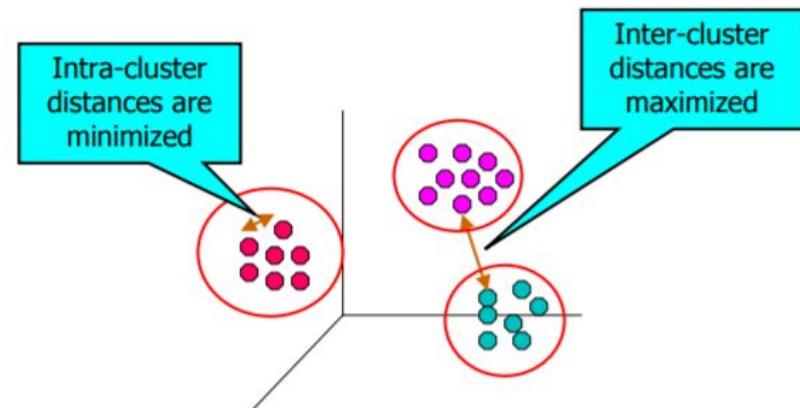


# Clustering (agrupamiento)

Encontrar **grupos de instancias (clusters)** a partir de **información en los datos** que **describan objetos y sus relaciones**.

Instancias de un cluster tienen que ser:

- similares entre sí y
- diferentes a las de otros clusters



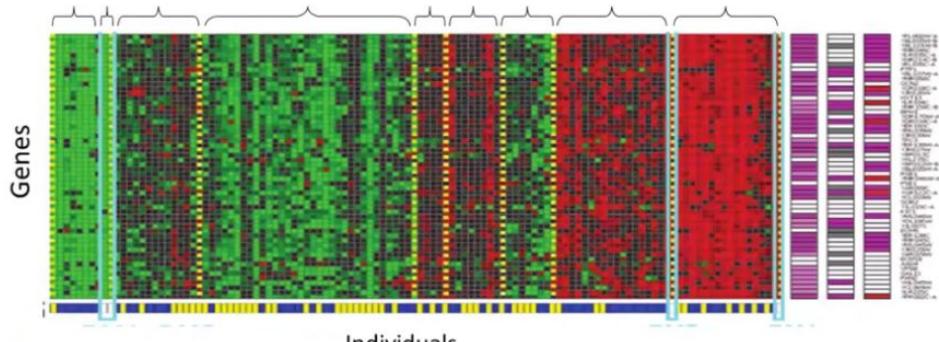
Tan, Steinbach & Kumar, Introduction to Data Mining

[https://www-users.cs.umn.edu/~kumar001/dmbook/dmslides/chap8\\_basic\\_cluster\\_analysis.pdf](https://www-users.cs.umn.edu/~kumar001/dmbook/dmslides/chap8_basic_cluster_analysis.pdf)

# Algoritmos de clustering

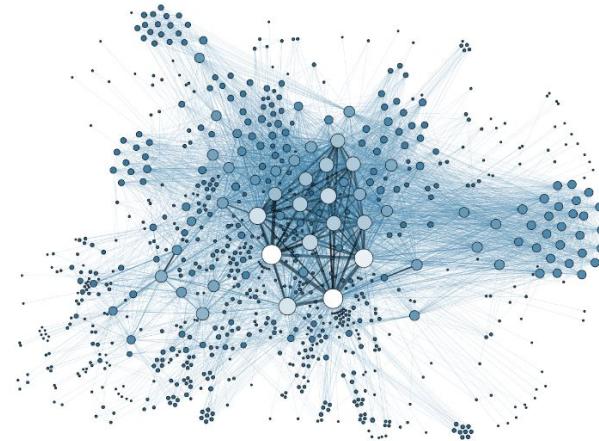
- **De partición:** se clasifican **m datos** en **k clusters**. Cada cluster satisface requerimientos de una partición:
  - cada dato está en un y sólo un cluster
  - cada cluster debe tener al menos un dato
- **Jerárquicos**
  - **Aglomerativos (bottom up):** empiezan con n clusters (cada dato es un cluster) y se combinan grupos.
  - **Divisorios (top down):** comienzan con un cluster de n observaciones y en cada paso se dividen.

# Aplicaciones



Fuente: curso ML Stanford

Análisis de redes sociales



Fuente: Wikimedia commons



Segmentación del mercado.  
Fuente: internet

# K- Means

## K - medias

# Algoritmo K-medias (K-means)

- Es un método iterativo:
  1. **Inicialización:** se elige la localización de los centroides de los K grupos aleatoriamente
  2. **Asignación:** se asigna cada dato al centroide más cercano
  3. **Actualización:** se actualiza la posición del centroide a la media aritmética de las posiciones de los datos asignados al grupo

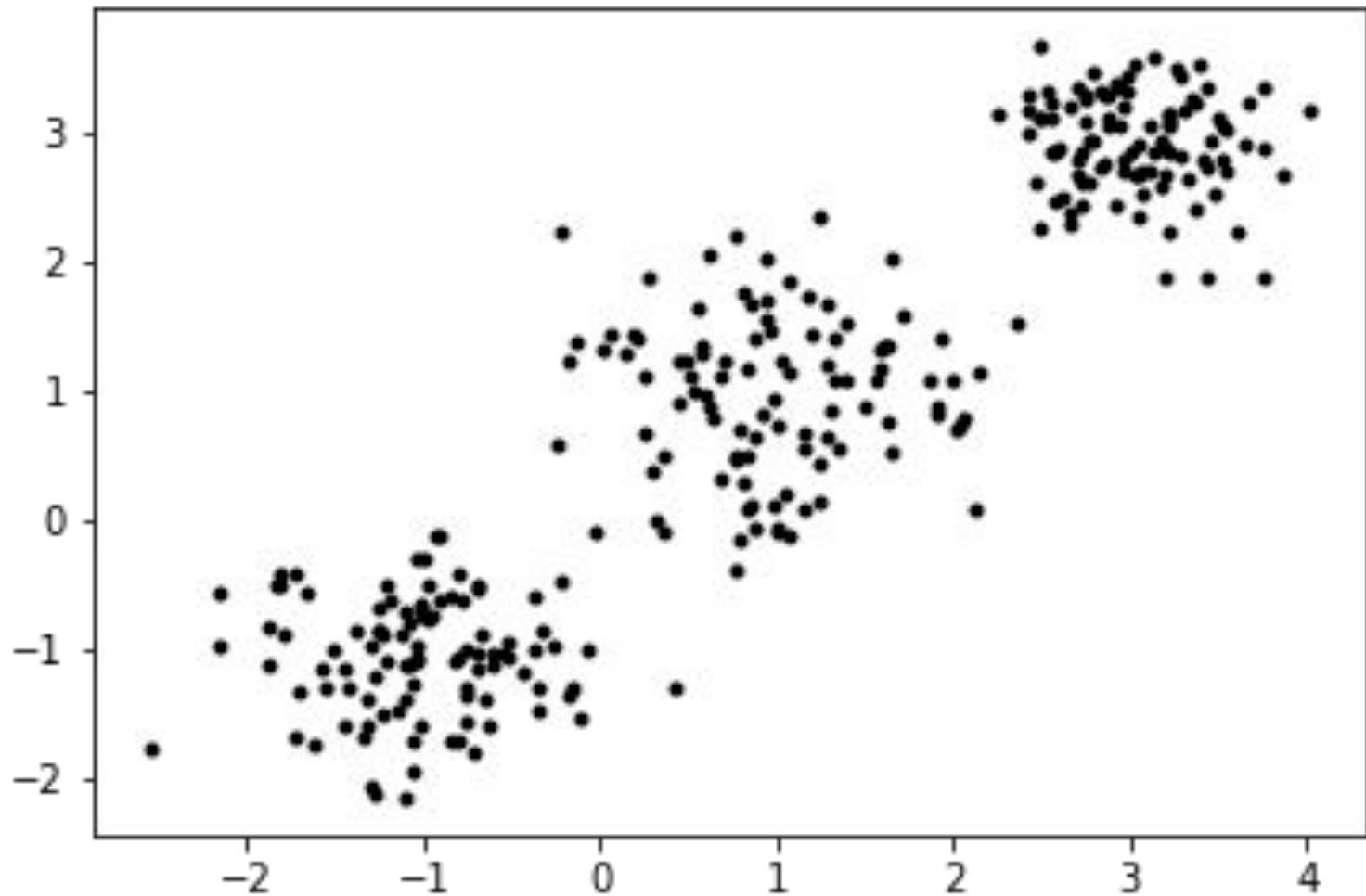
Repite 2 y 3 hasta que la asignación es estable o se agotan las iteraciones permitidas.

*Datos de entrada*

Ejecución ( )

*Datos de entrada*

Ejecución (Dataset)

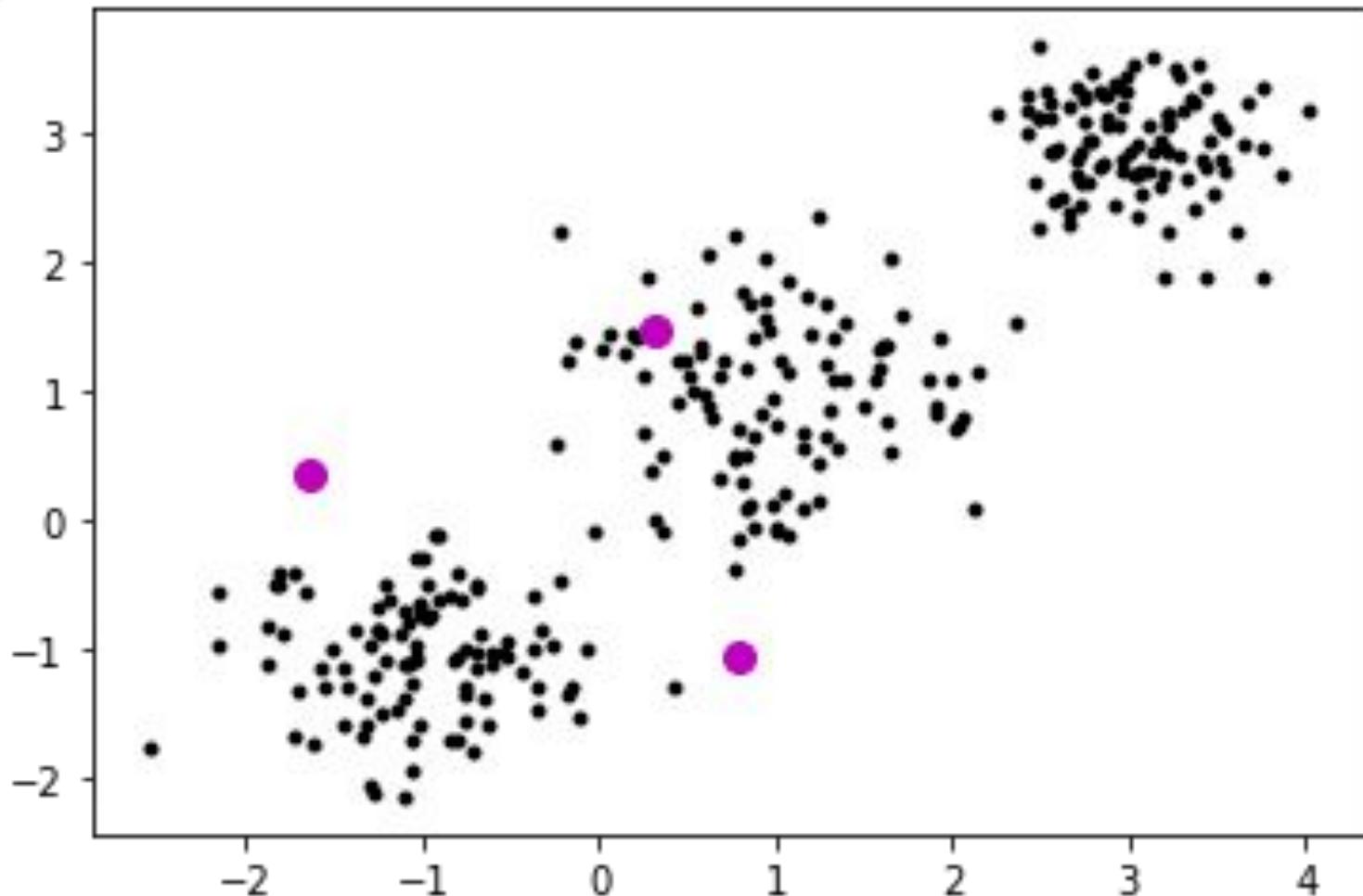


Usamos  $k = 3$

Sorteamos los centroides

Ejecución (Dataset)

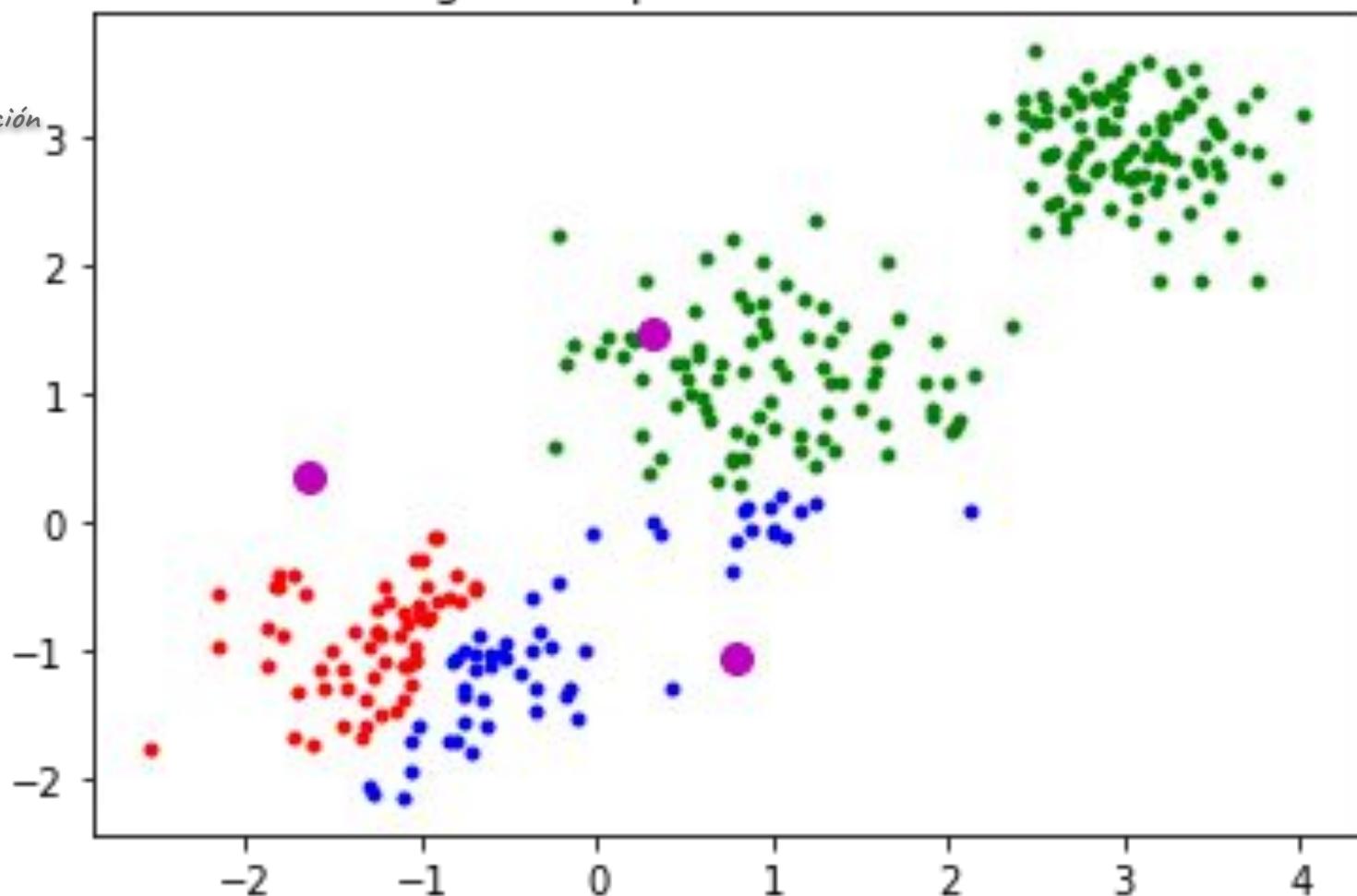
1. Inicialización



## Asignamos puntos a centroides

Ejecución (Dataset)

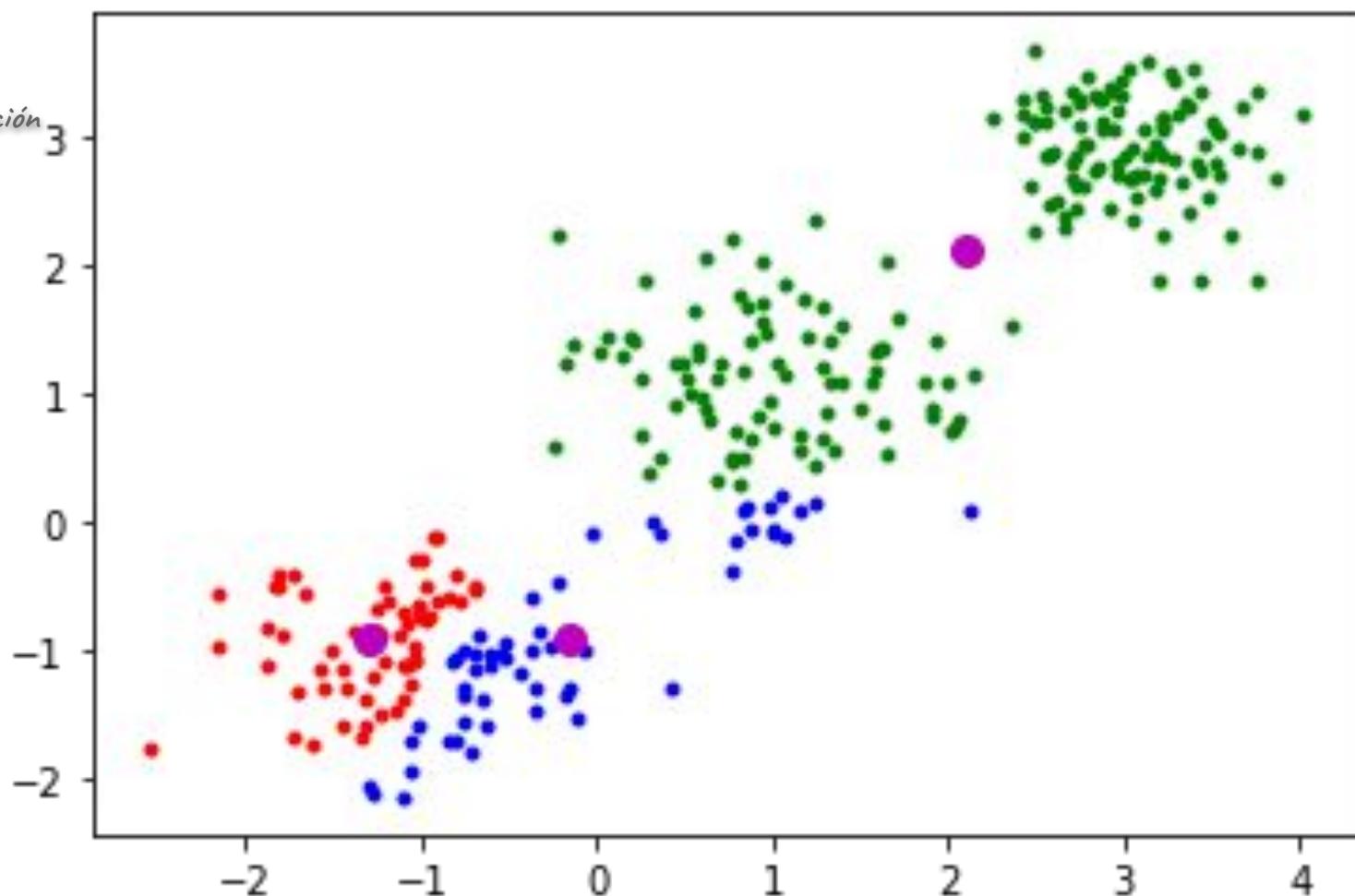
1. Inicialización y Asignación



## Reubicamos centroides

Ejecución (Dataset)

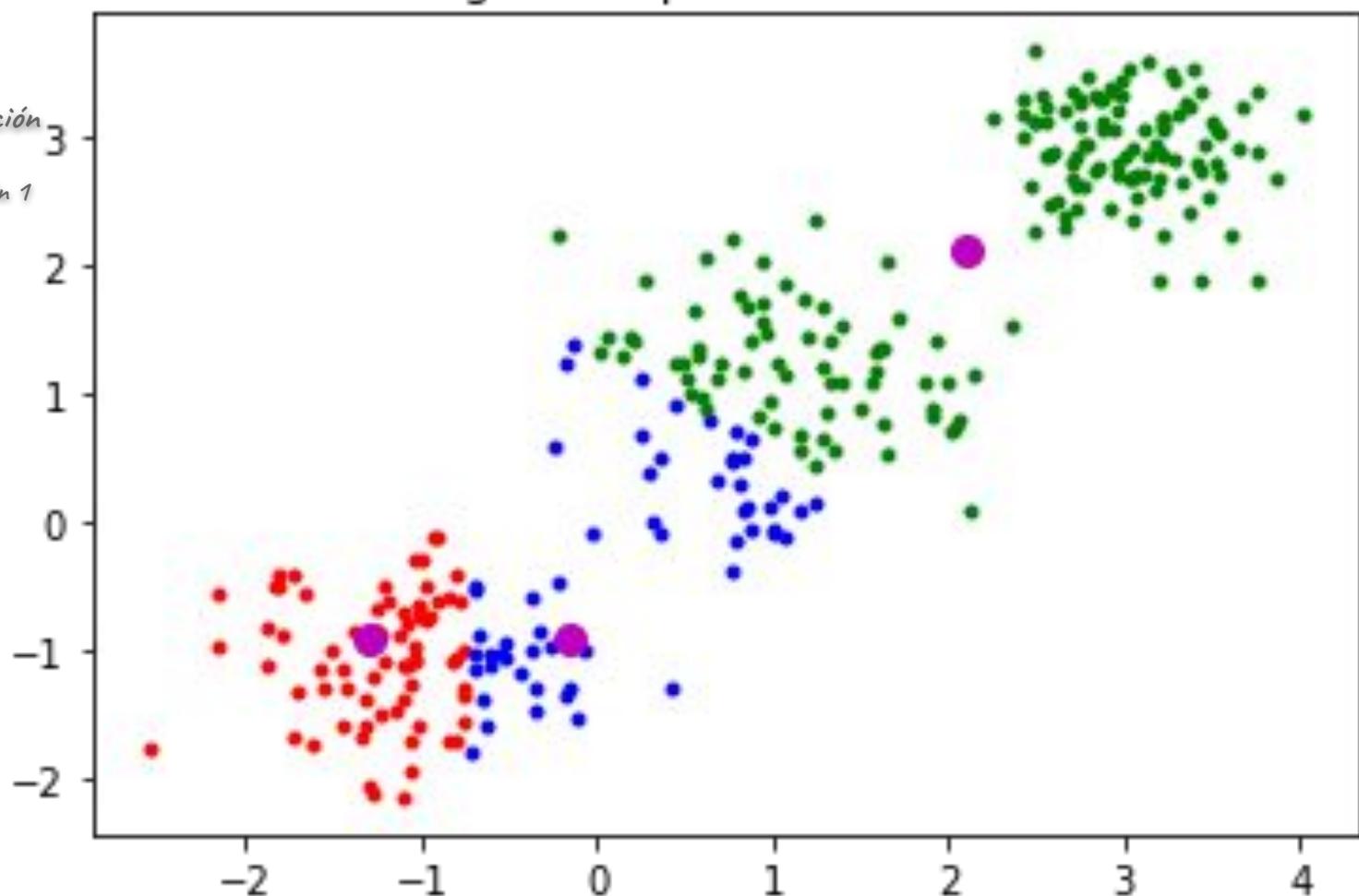
1. Inicialización y Asignación
2. Actualización



## Reasignamos puntos a centroides

Ejecución (Dataset)

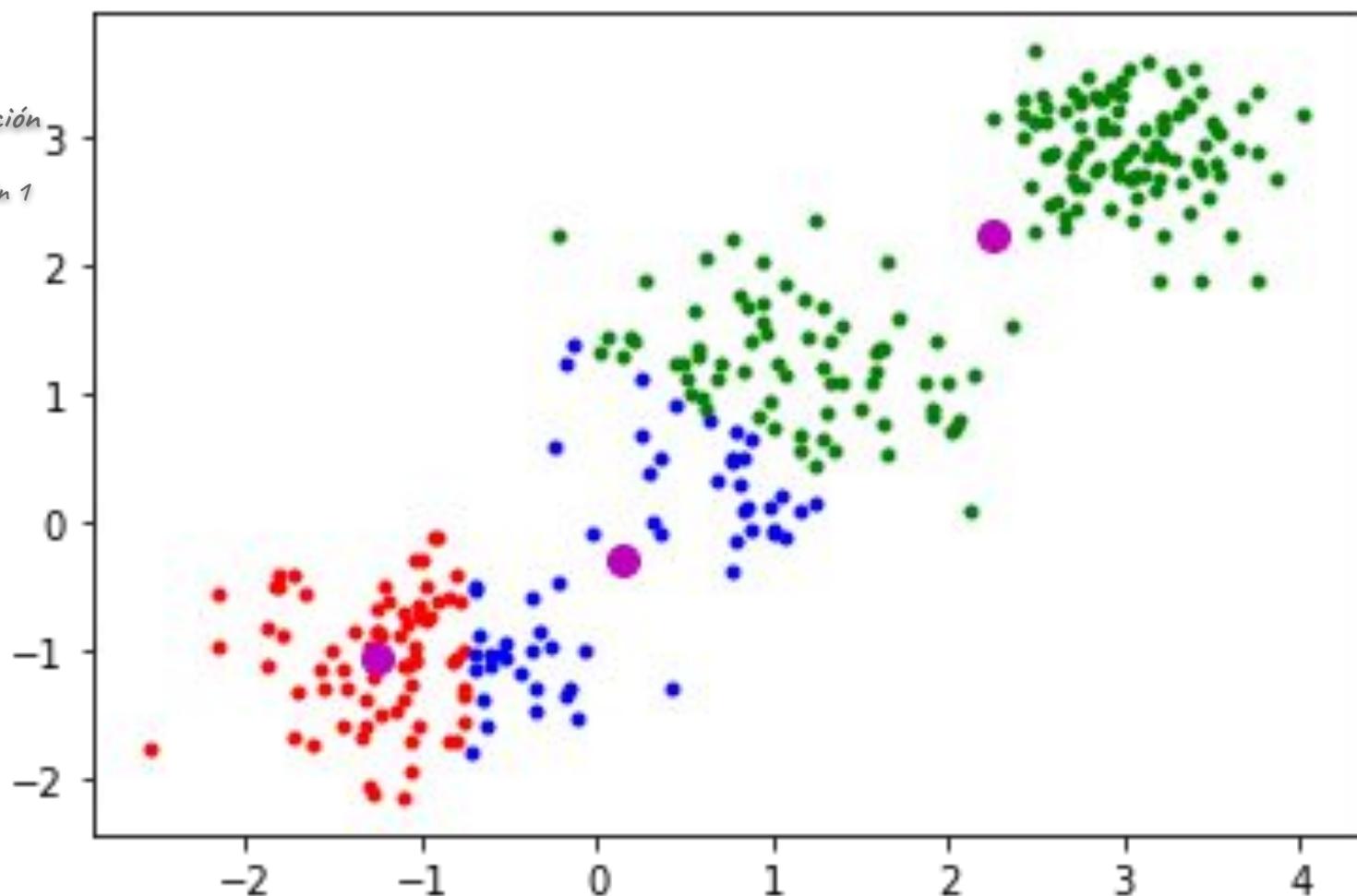
1. Inicialización y Asignación
2. Actualización } Iteración 1
3. Asignación



## Reubicamos centroides

Ejecución (Dataset)

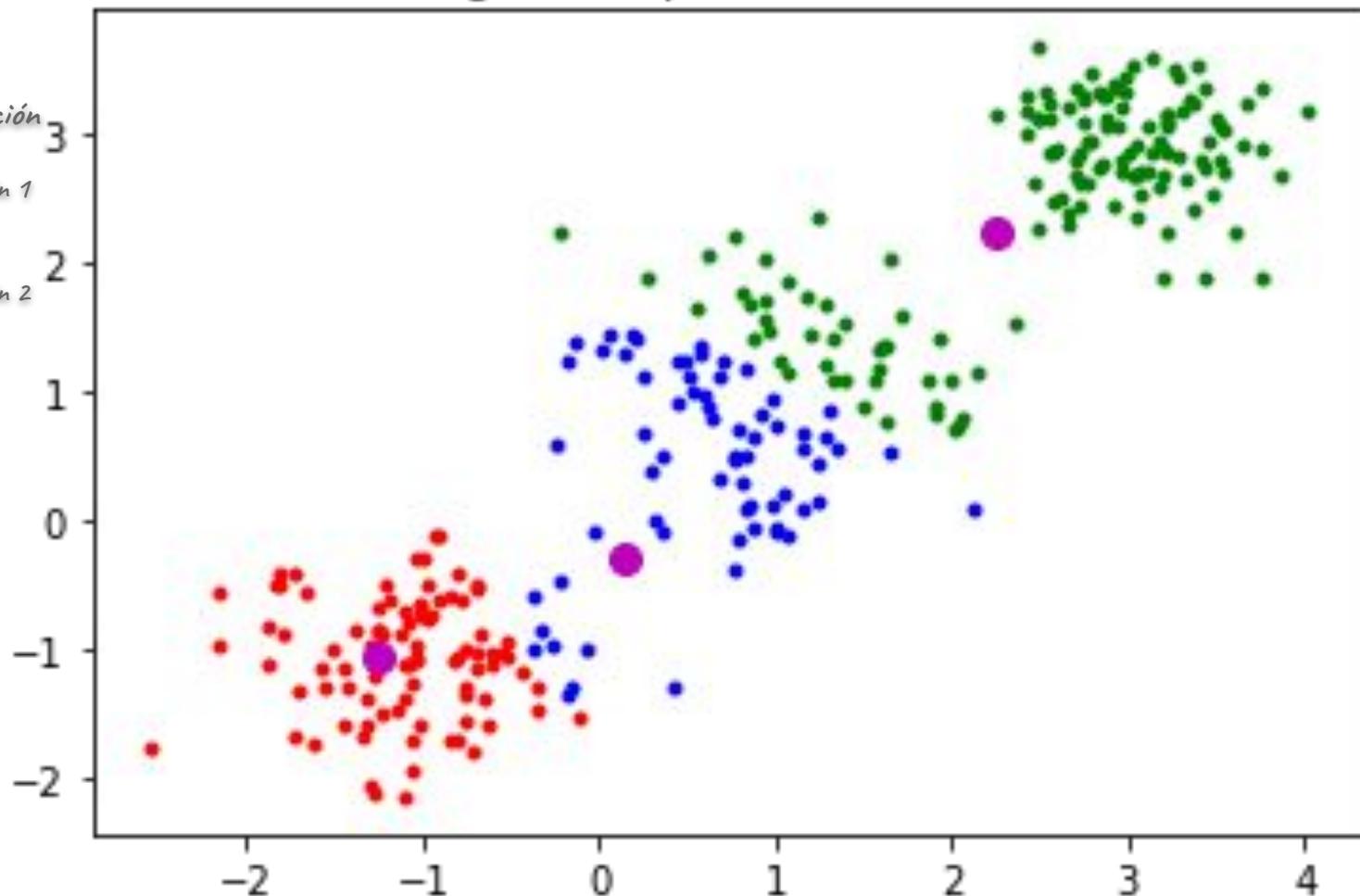
1. Inicialización y Asignación
2. Actualización } Iteración 1
3. Asignación
2. Actualización



## Reasignamos puntos a centroides

Ejecución (Dataset)

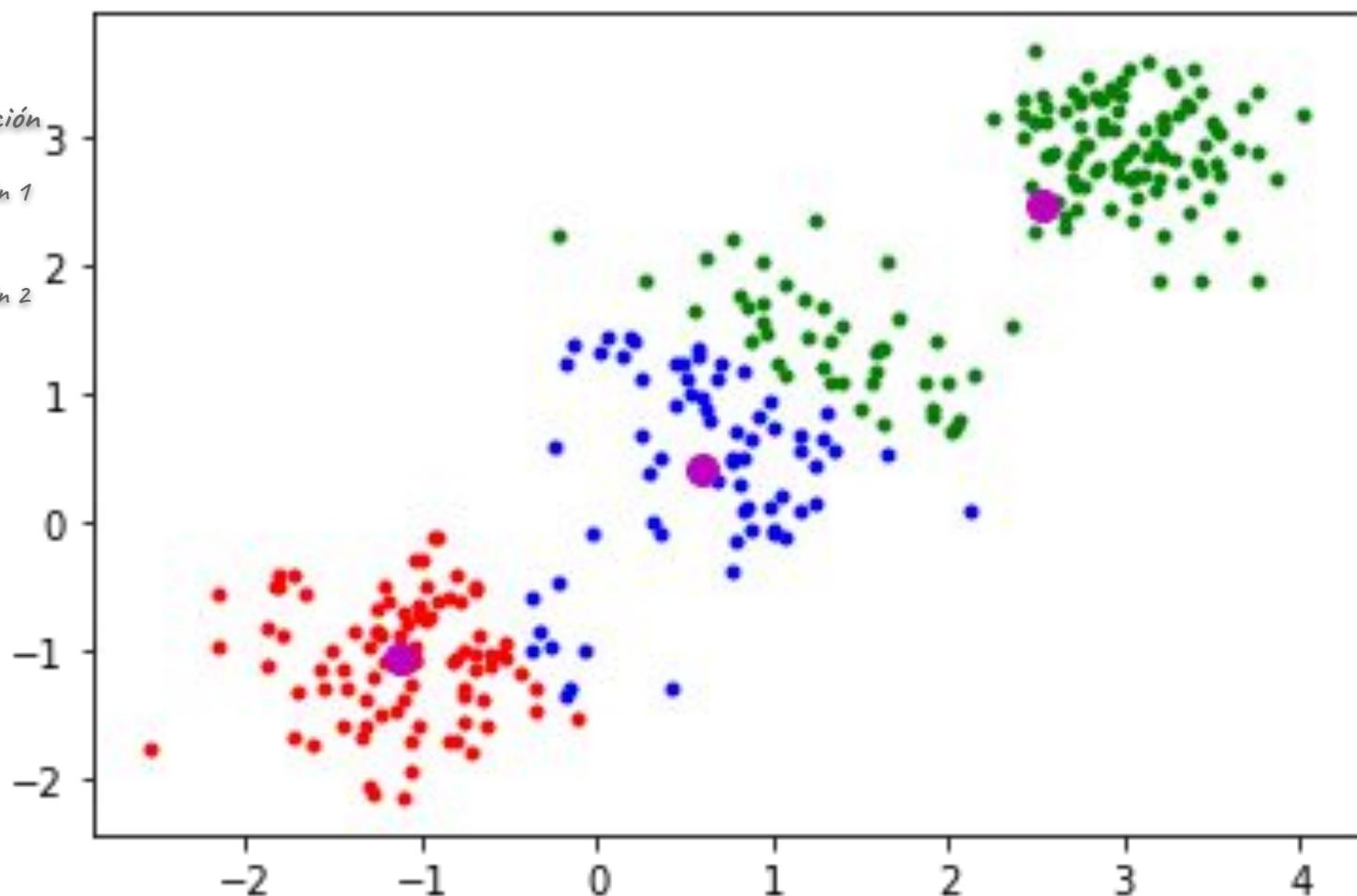
1. Inicialización y Asignación
2. Actualización } Iteración 1
3. Asignación
2. Actualización } Iteración 2
3. Asignación



## Reubicamos centroides

Ejecución (Dataset)

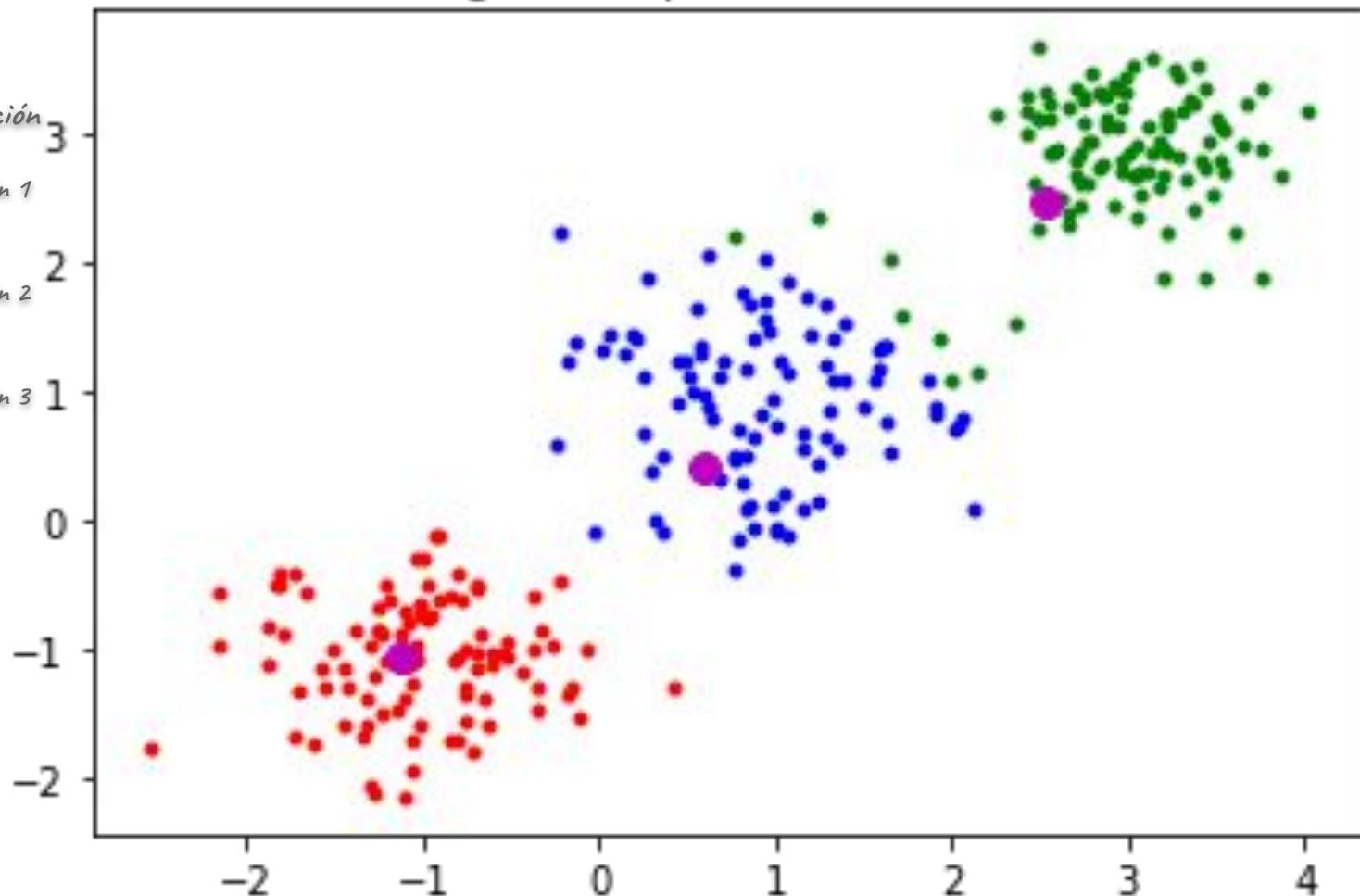
1. Inicialización y Asignación
2. Actualización } Iteración 1
3. Asignación
2. Actualización } Iteración 2
3. Asignación
4. Actualización



## Reasignamos puntos a centroides

Ejecución (Dataset)

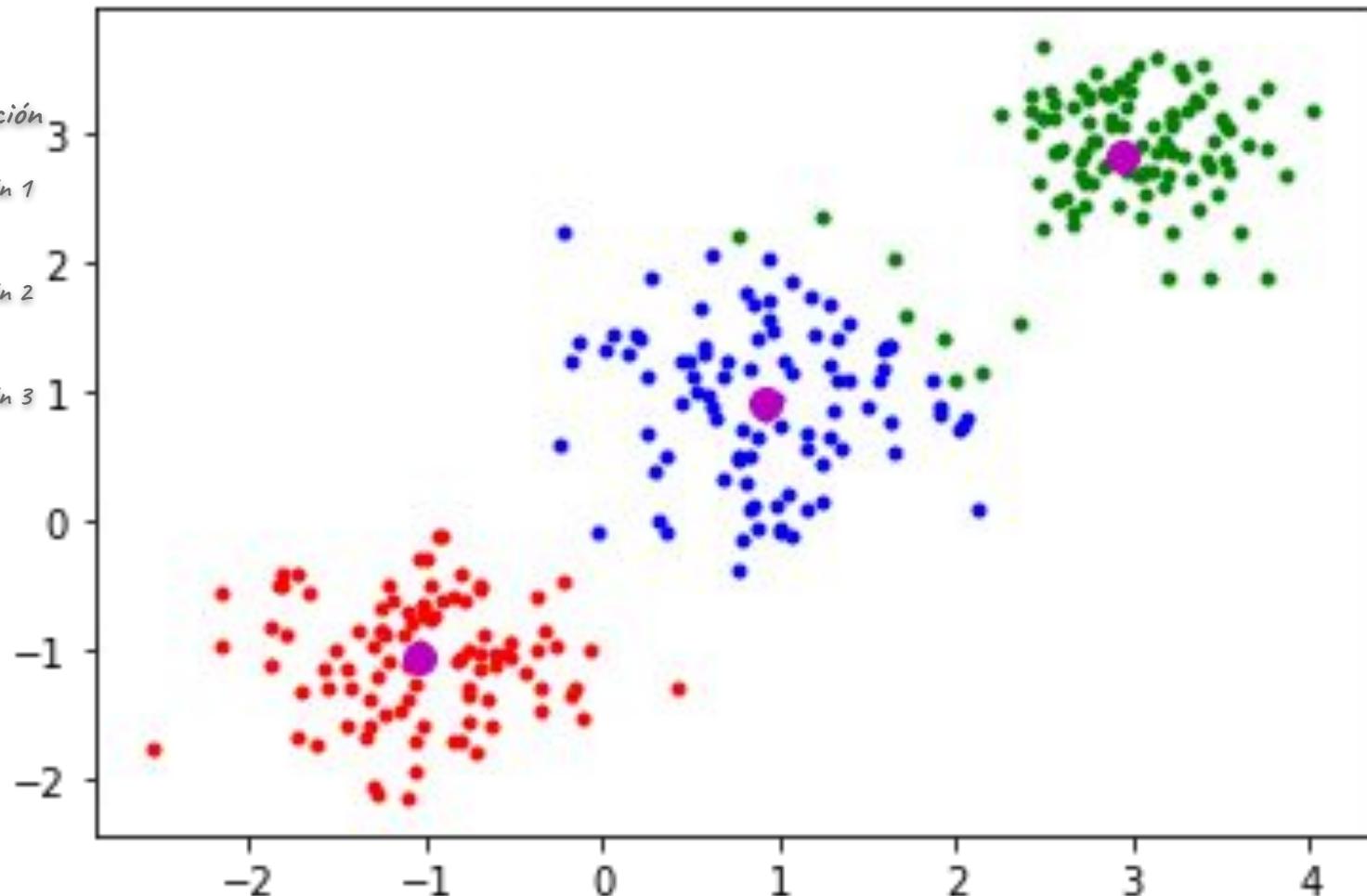
1. Inicialización y Asignación
2. Actualización } Iteración 1
3. Asignación }
2. Actualización } Iteración 2
3. Asignación }
4. Actualización } Iteración 3
5. Asignación }



## Reubicamos centroides

### Ejecución (Dataset)

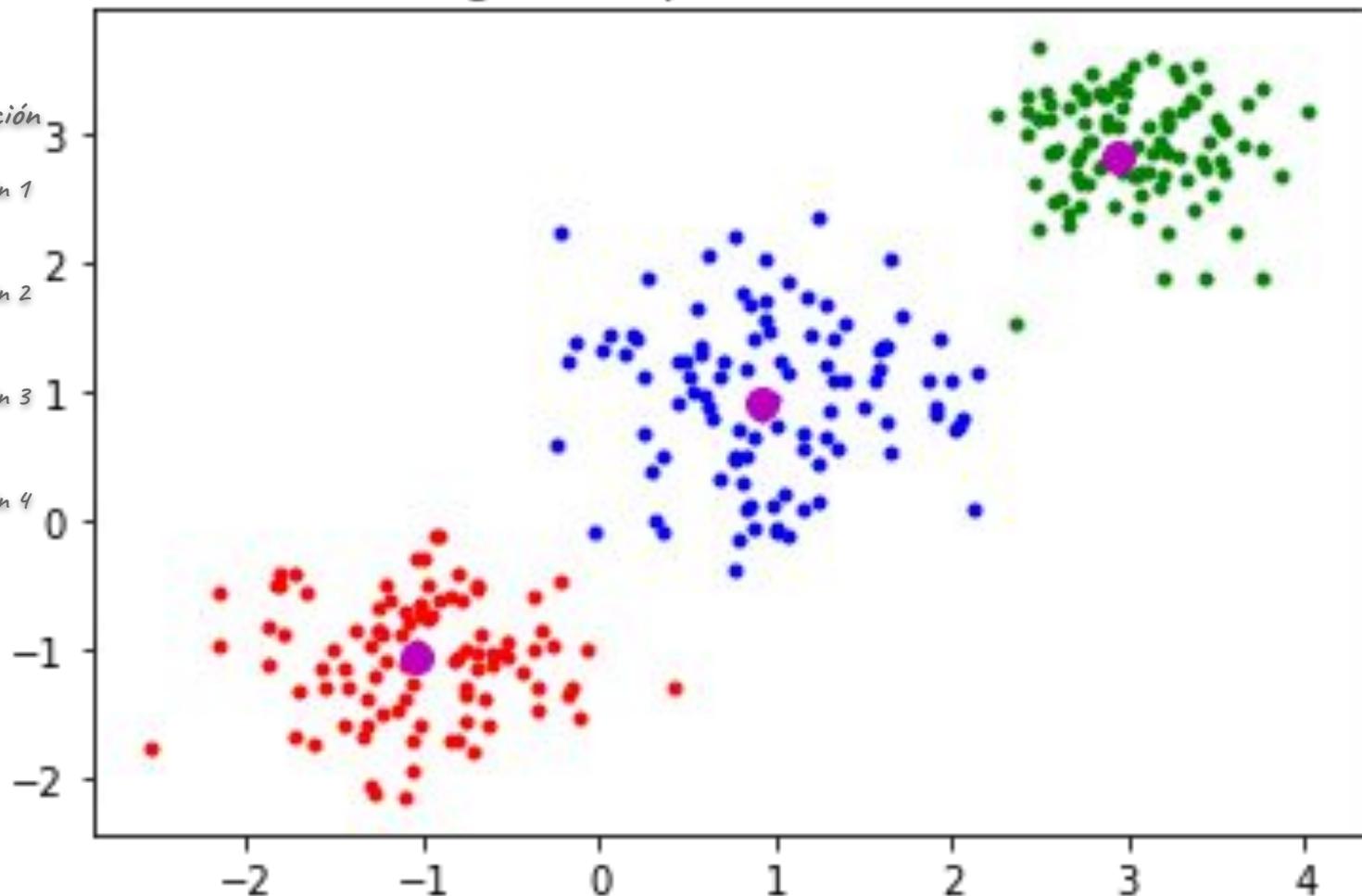
1. Inicialización y Asignación
2. Actualización } Iteración 1
3. Asignación }
2. Actualización } Iteración 2
3. Asignación }
4. Actualización } Iteración 3
5. Asignación }
6. Actualización



## Reasignamos puntos a centroides

### Ejecución (Dataset)

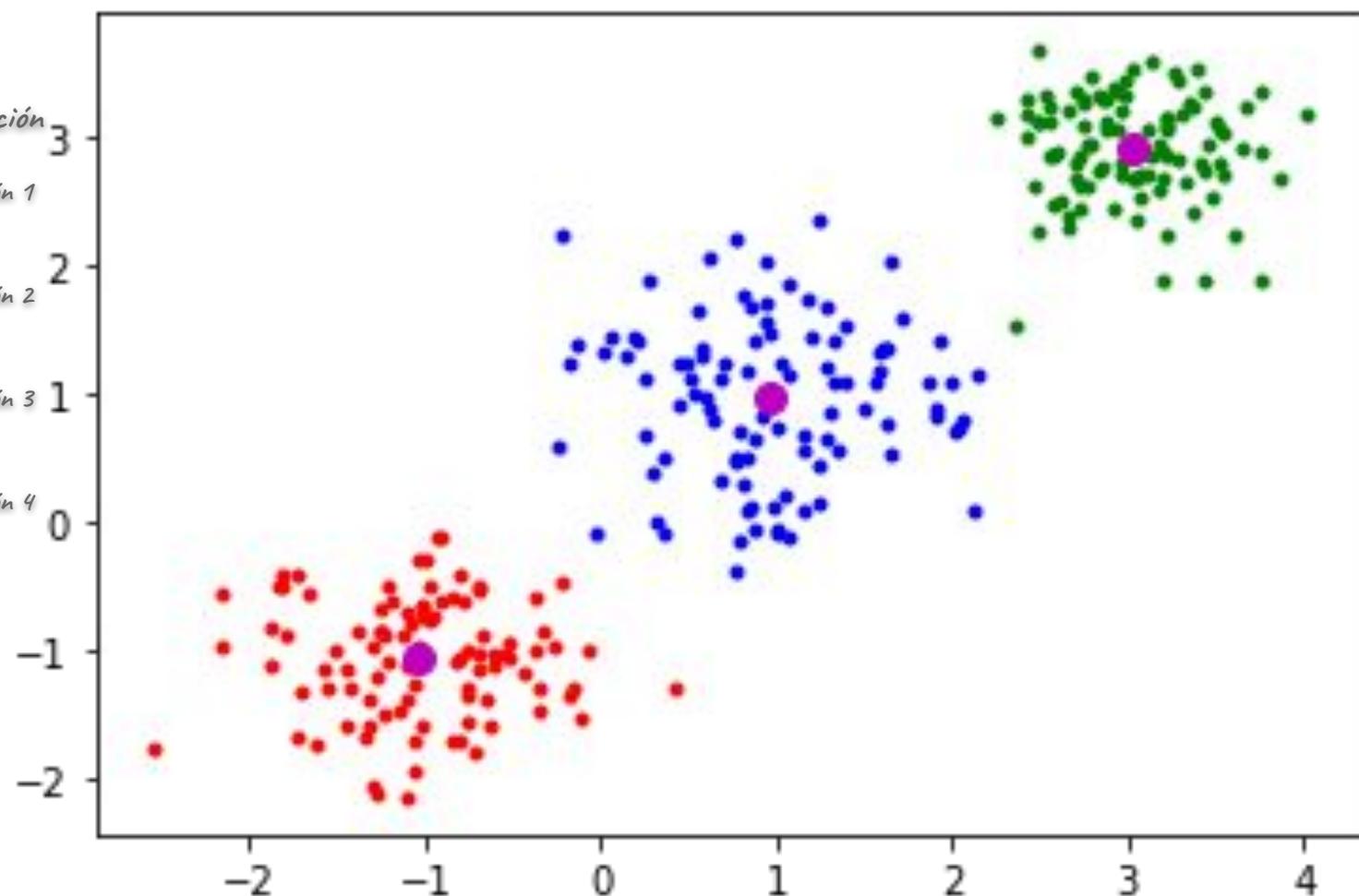
1. Inicialización y Asignación
2. Actualización } Iteración 1
3. Asignación }
2. Actualización } Iteración 2
3. Asignación }
4. Actualización } Iteración 3
5. Asignación }
6. Actualización } Iteración 4
7. Asignación }



# Reubicamos centroides

## Ejecución (Dataset)

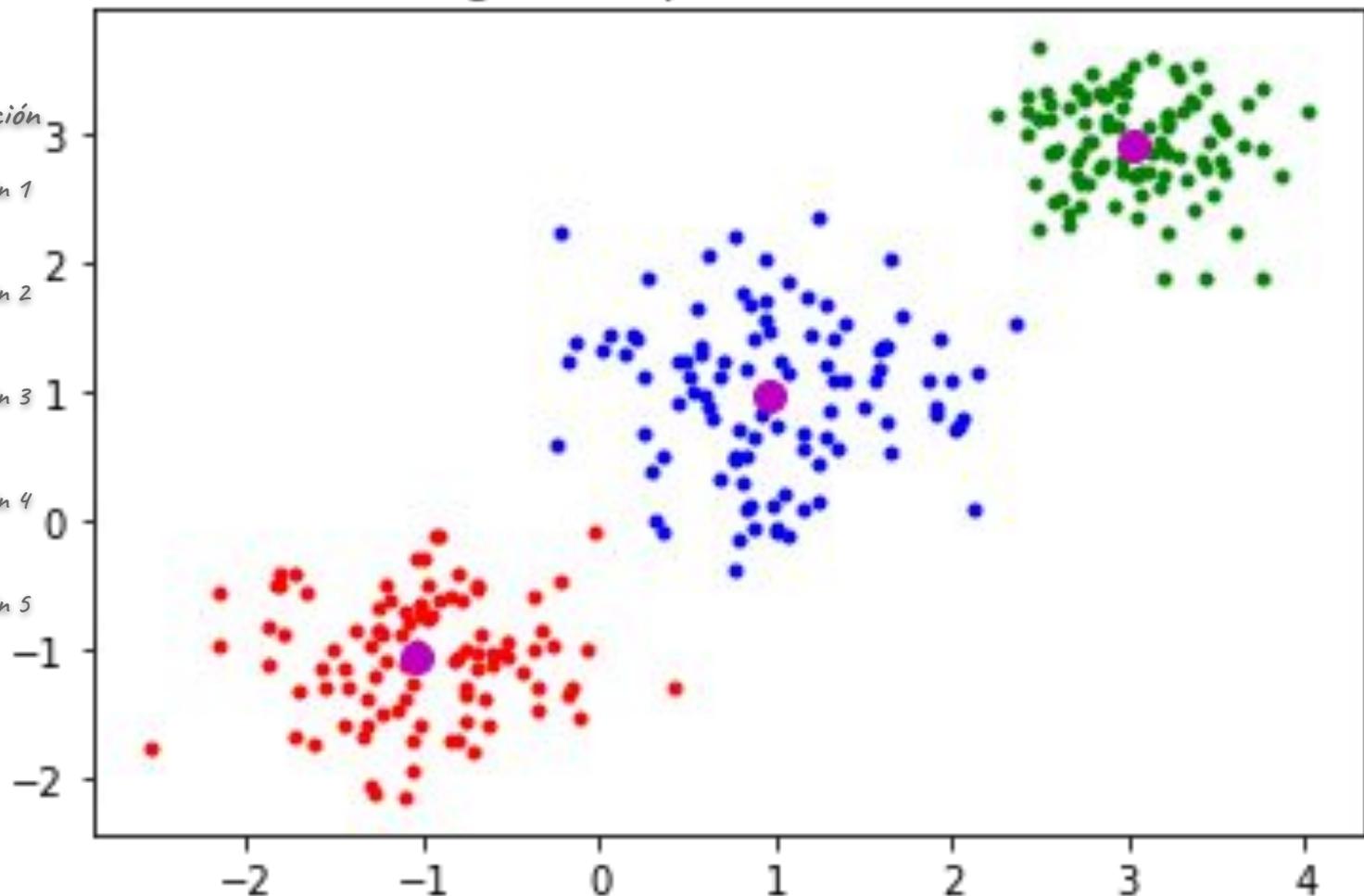
1. Inicialización y Asignación
2. Actualización } Iteración 1
3. Asignación }
2. Actualización } Iteración 2
3. Asignación }
4. Actualización } Iteración 3
5. Asignación }
6. Actualización } Iteración 4
7. Asignación }
8. Actualización



# Reasignamos puntos a centroides

## Ejecución (Dataset)

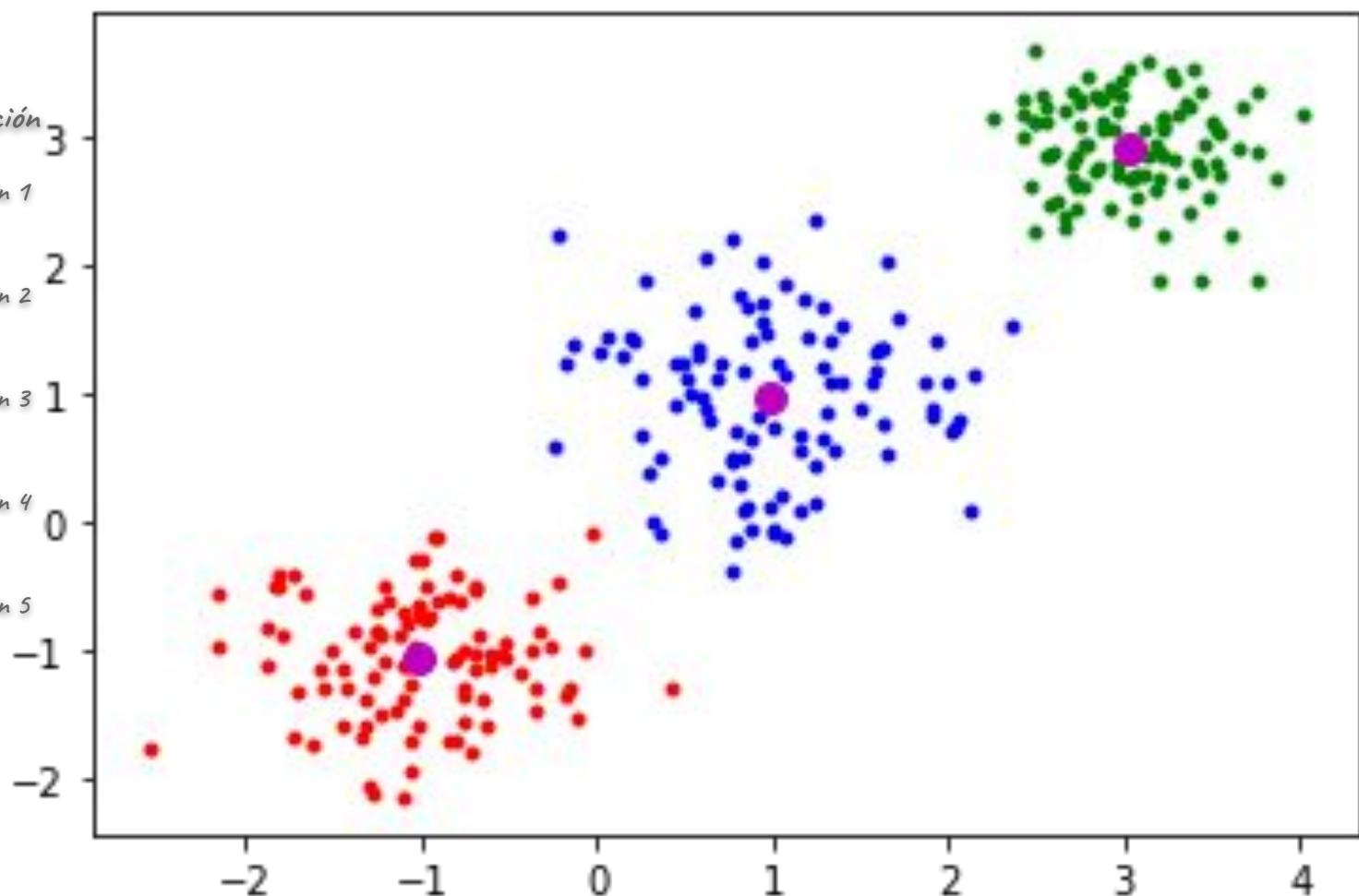
1. Inicialización y Asignación
2. Actualización } Iteración 1
3. Asignación }
2. Actualización } Iteración 2
3. Asignación }
4. Actualización } Iteración 3
5. Asignación }
6. Actualización } Iteración 4
7. Asignación }
8. Actualización } Iteración 5
9. Asignación }



# Reubicamos centroides

## Ejecución (Dataset)

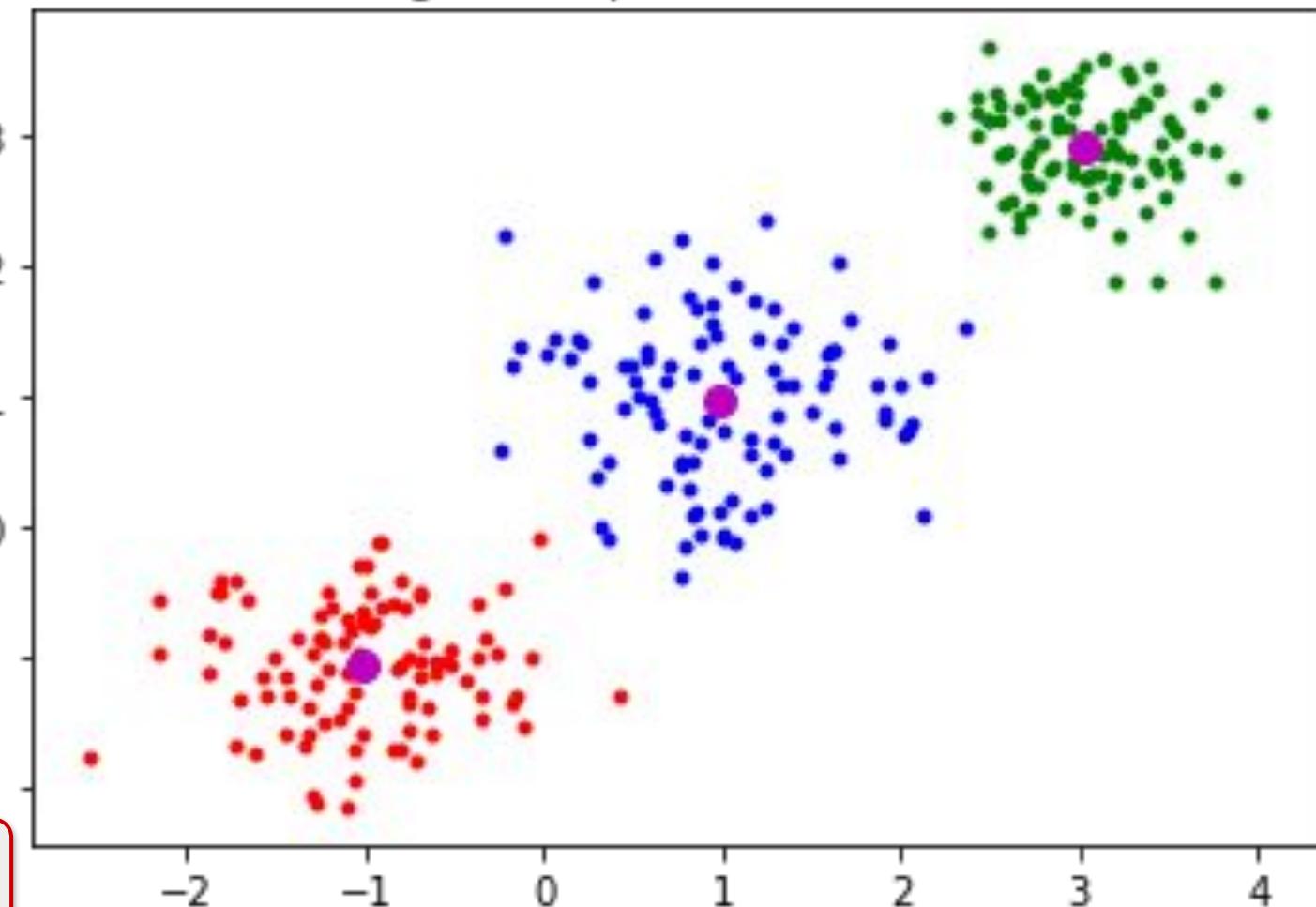
1. Inicialización y Asignación
2. Actualización } Iteración 1
3. Asignación }
2. Actualización } Iteración 2
3. Asignación }
4. Actualización } Iteración 3
5. Asignación }
6. Actualización } Iteración 4
7. Asignación }
8. Actualización } Iteración 5
9. Asignación }
10. Actualización



# Reasignamos puntos a centroides

## Ejecución (Dataset)

1. Inicialización y Asignación
2. Actualización } Iteración 1
3. Asignación }
2. Actualización } Iteración 2
3. Asignación }
4. Actualización } Iteración 3
5. Asignación }
6. Actualización } Iteración 4
7. Asignación }
8. Actualización } Iteración 5
9. Asignación }
10. Actualización } Iteración 6
11. Asignación }



Decidimos cortar por cantidad  
de iteraciones

# K-means

Optimización, función de costo a minimizar. Función de **distorsión**.

$$J = \sum_{i=1}^m \sum_{k=1}^K a_{ik} \cdot \|x^{(i)} - \mu_k\|^2$$

*Within-Cluster Sum of Squares (**WCSS**)*

K: n° de clusters, m: cant. datos

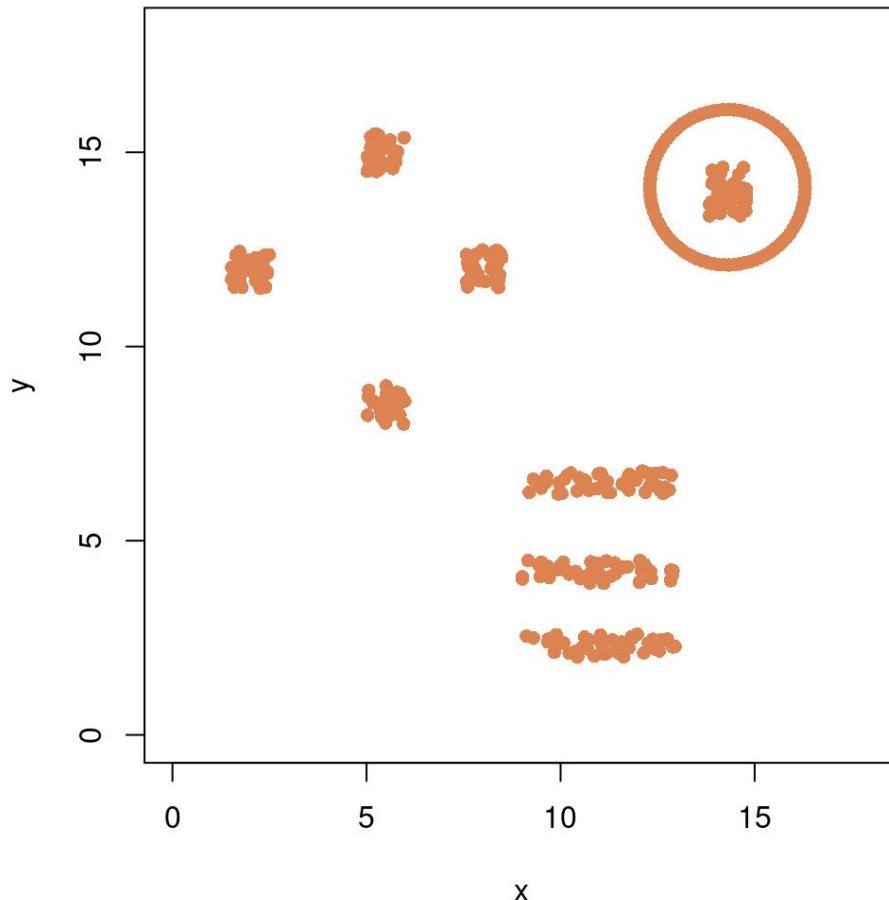
$\mu_k$ : centroide de cluster k

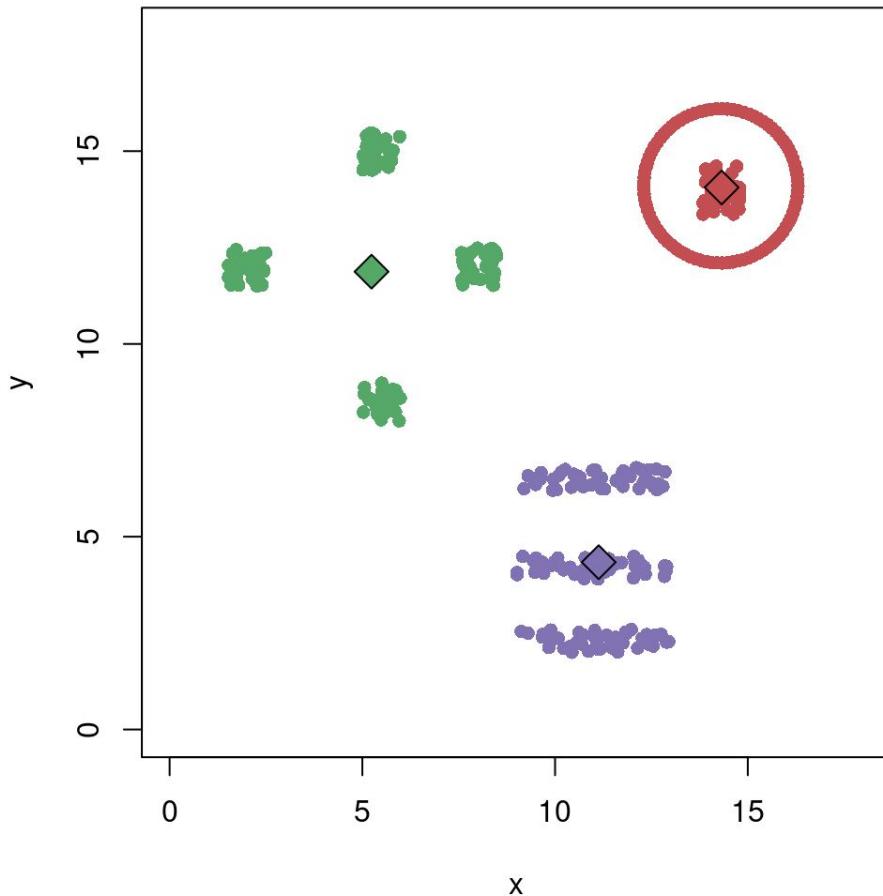
$x^{(i)}$ : dato nro i

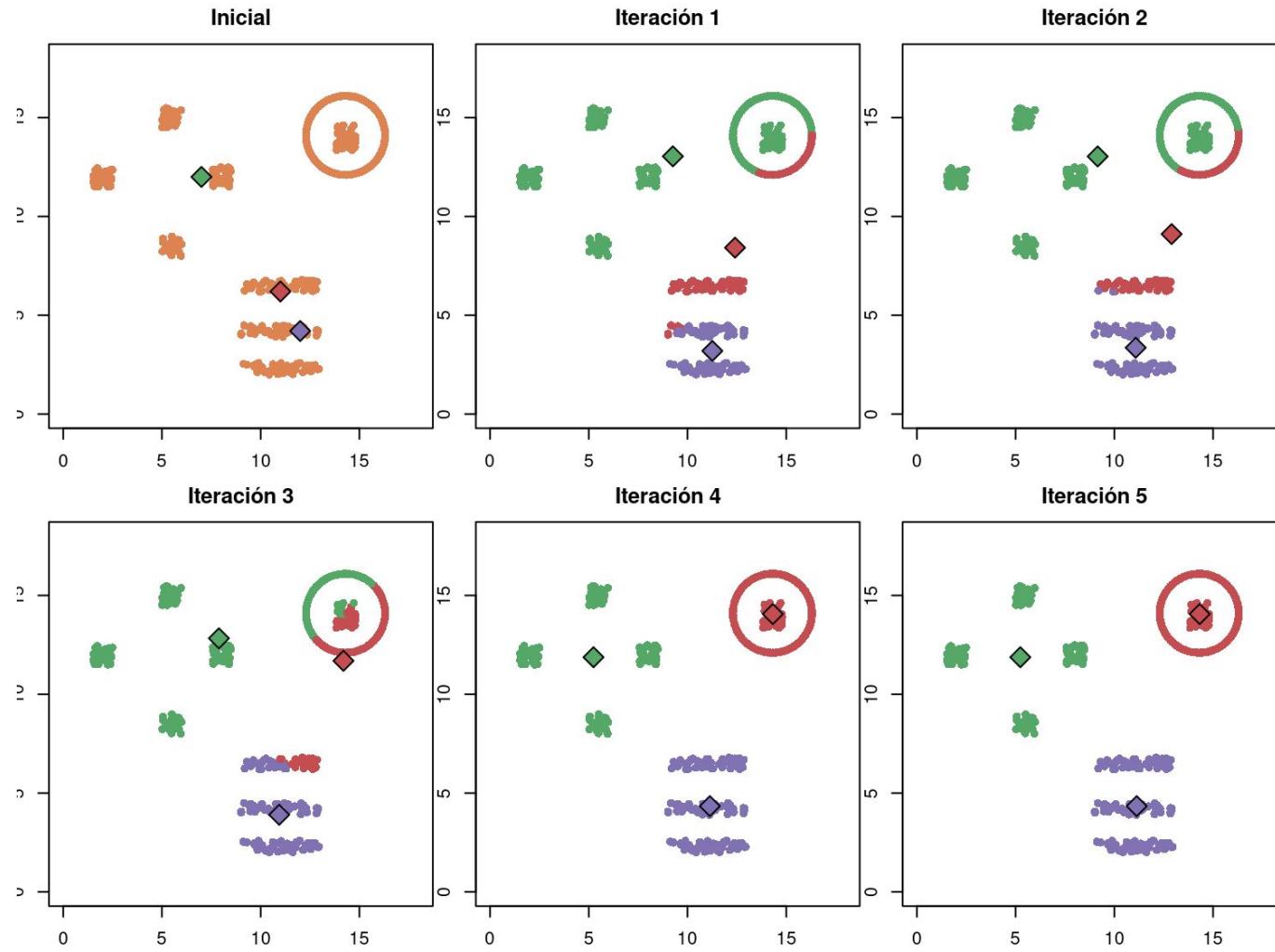
$a_{ik}$   $\begin{cases} 1 & \text{si } x^{(i)} \text{ está asignado al cluster k} \\ 0 & \text{en otro caso} \end{cases}$

K-means intenta encontrar  $\mu_k$  y  $a_{ik}$  que minimicen J

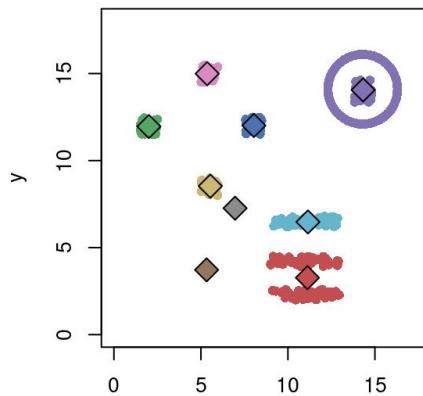
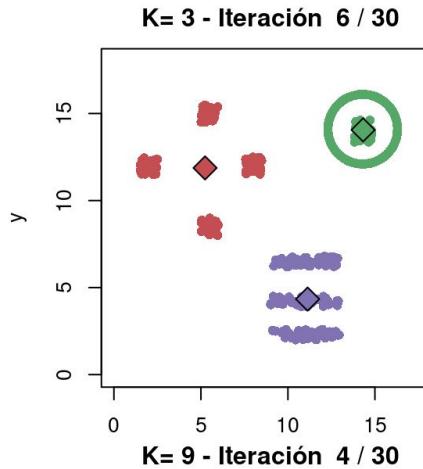
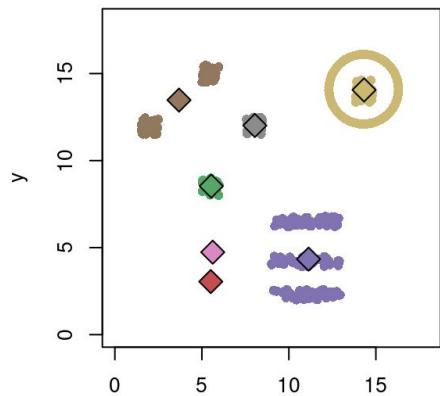
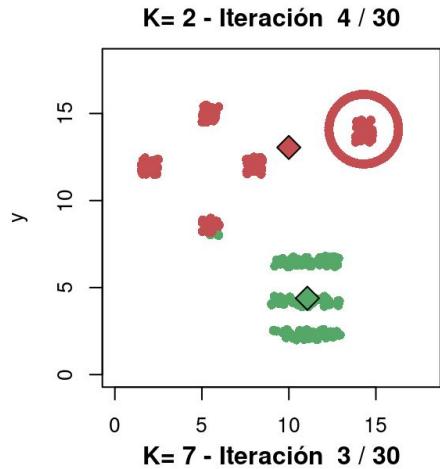
- En **asignación de cluster**: minimiza J con respecto a  $a_{ik}$  (asignando los puntos al centroide más cercano, que está fijo)
- En **actualización de centroide**: minimiza J con respecto a  $\mu_k$  (la ubicación de los centroides)







# ¿Qué sucede si lo ejecutamos con otros valores de k?



# ¿Cómo evaluamos el modelo?

¿Qué sería una métrica de que algo es bueno (o malo)?

Proponemos la distancia de cada punto al centro que le asignamos  
(Within-Cluster Sum of Squares (WCSS)):

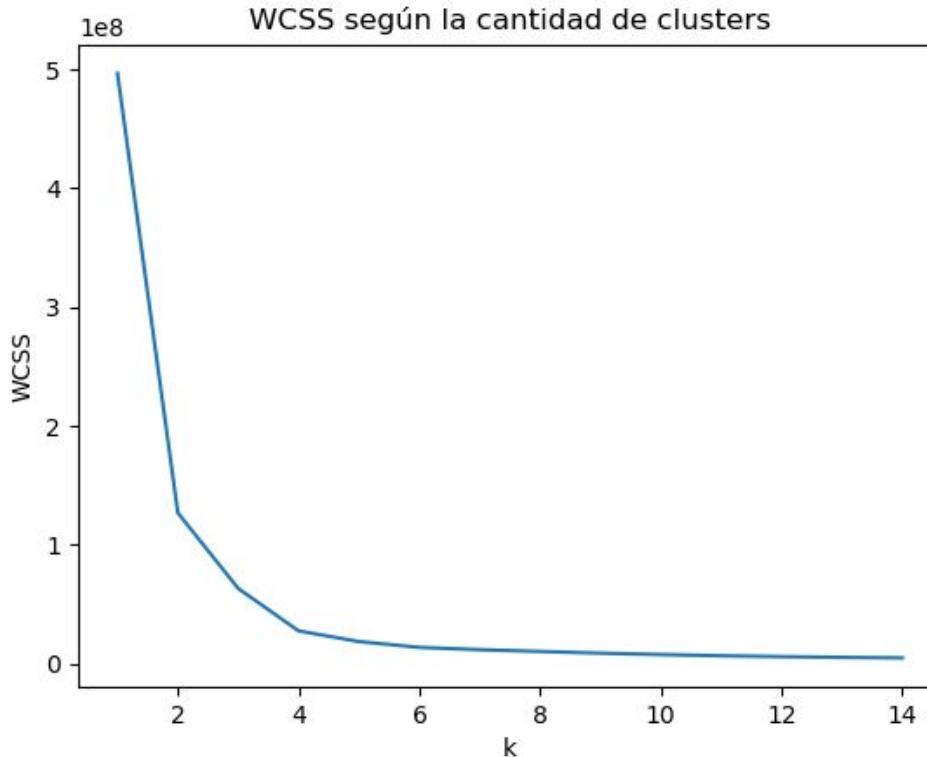
$$WCSS = \sum_{c_i \in C} \sum_{p_{i,j} \in C_i} \text{distancia}(p_{i,j} - c_i)^2$$

Siendo C el conjunto de los centros y C<sub>i</sub> el conjunto de puntos del cluster i.

**¿Cómo elegimos el valor de k?**

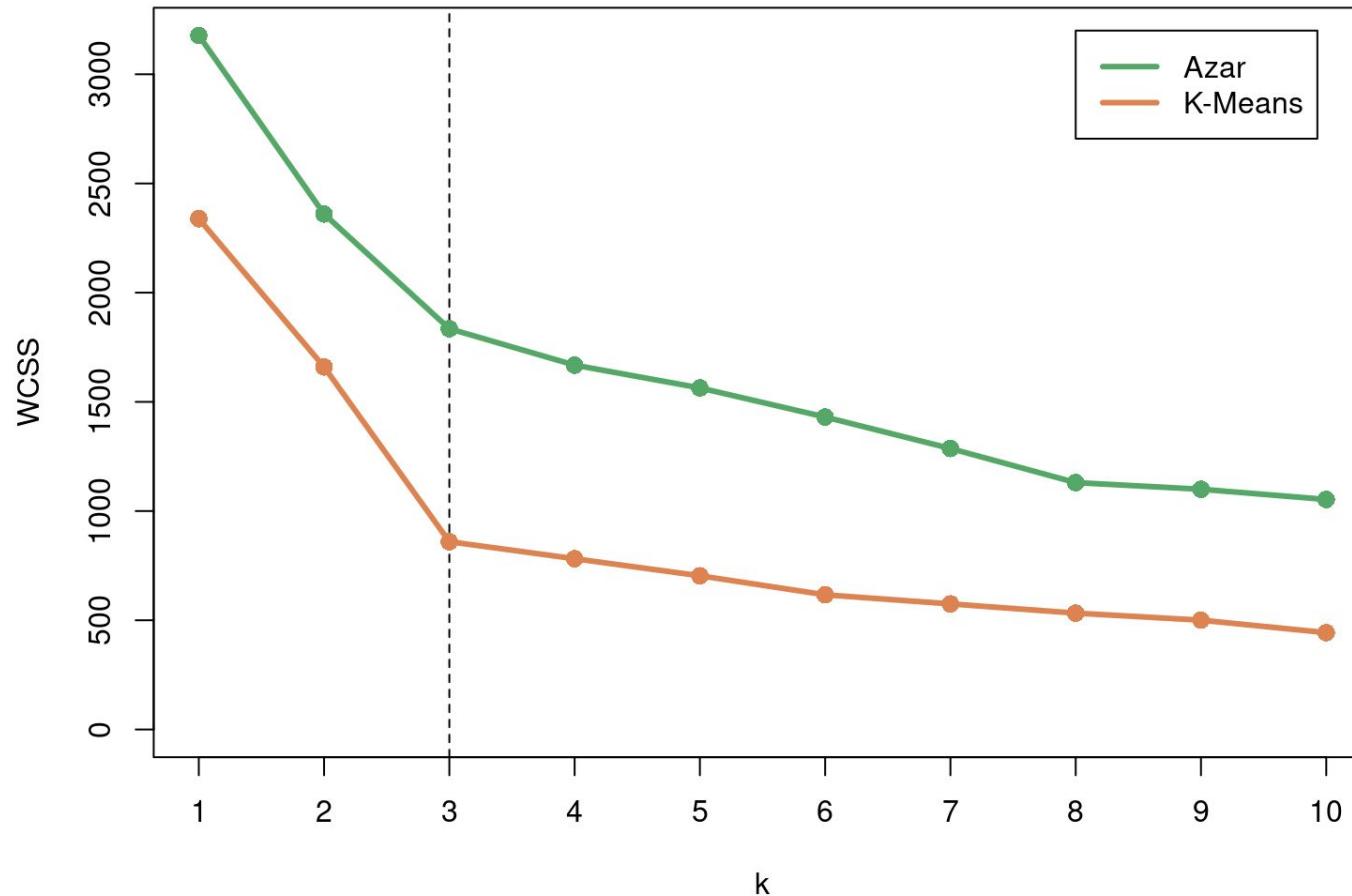
# ¿Cómo elegimos el valor de k?

Graficamos WCSS para un rango de k y usamos el método del codo



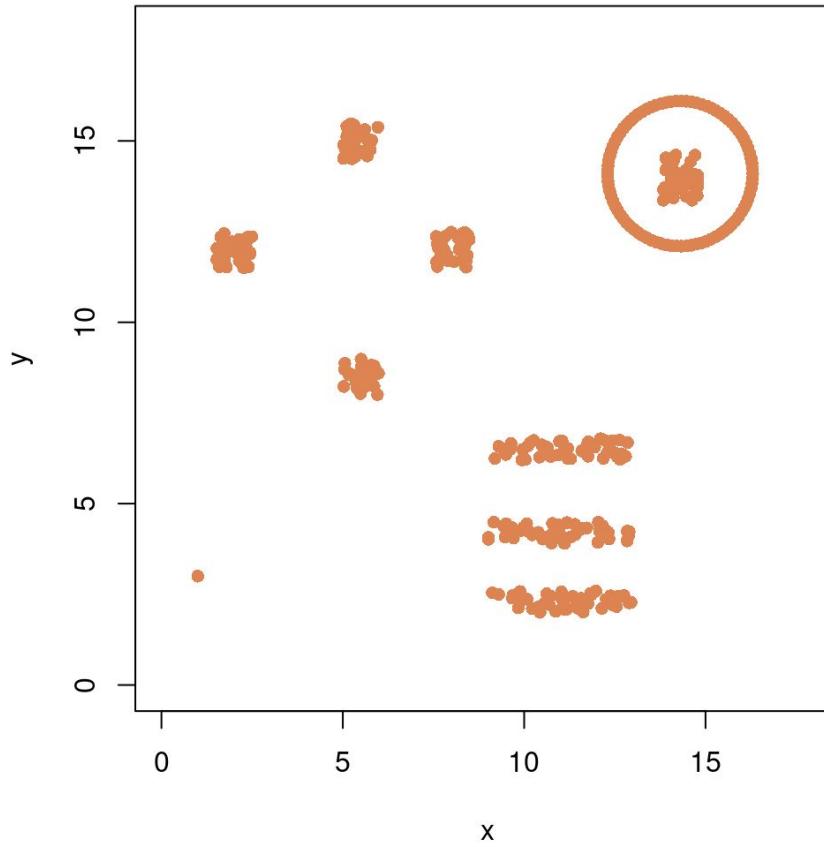
Pero dado un k, el agrupamiento puede depender de los centroides elegidos al momento de inicializar...

## Promedio de 50 repeticiones para cada k

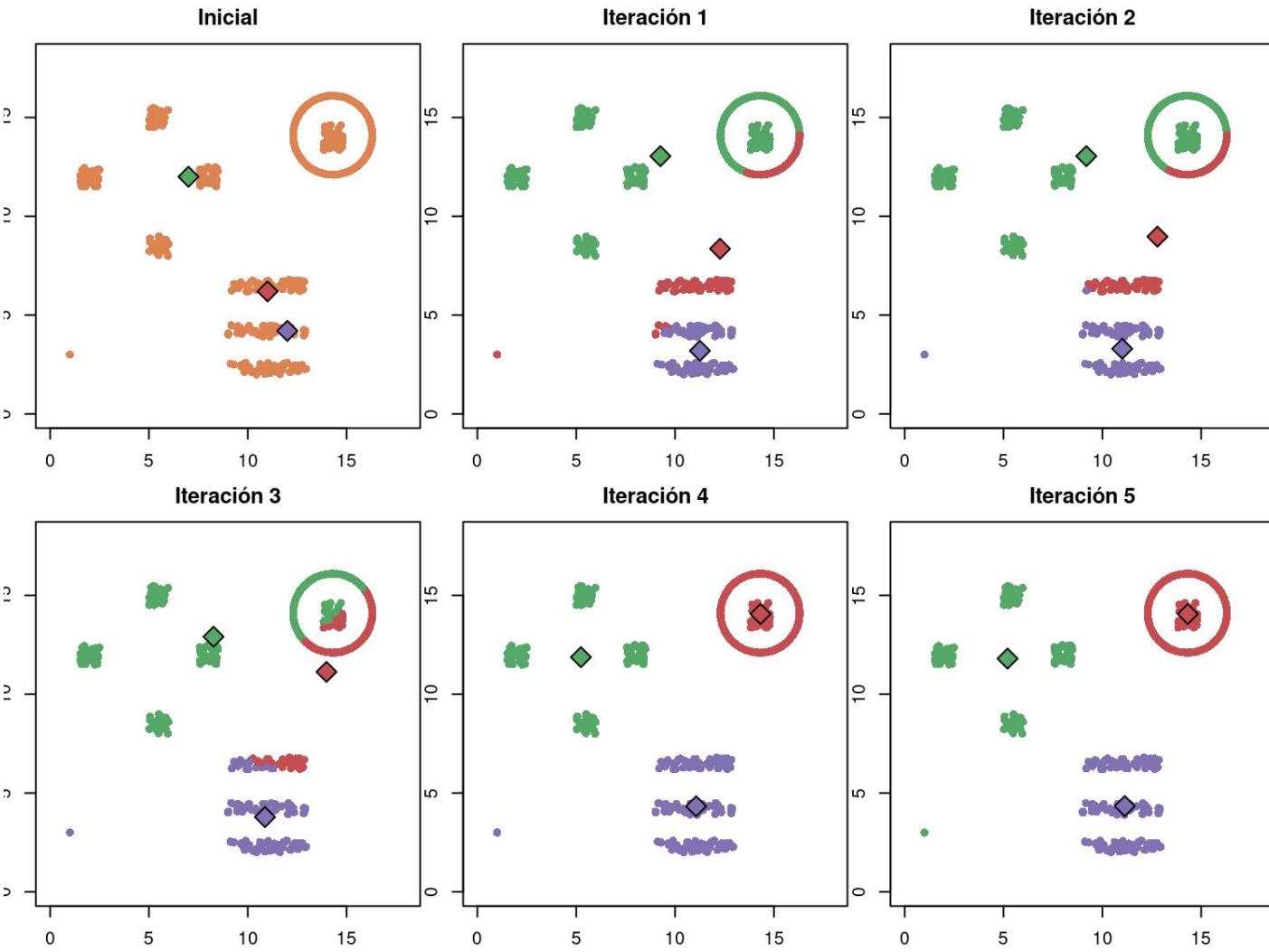


**¿Qué pasa con los datos atípicos  
(outliers)?**

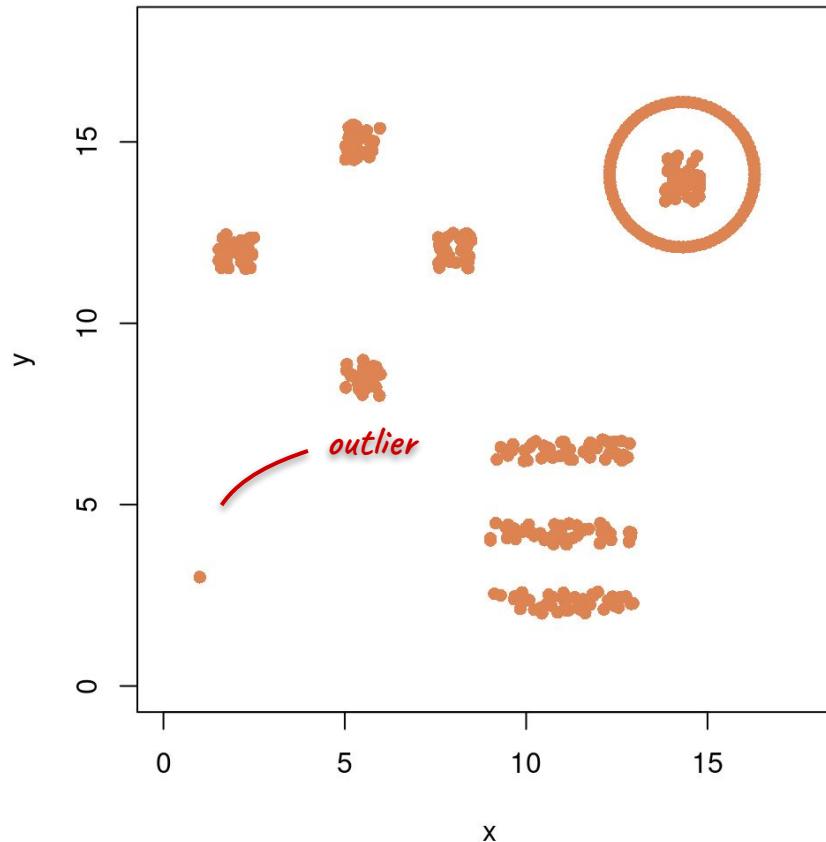
# Outliers



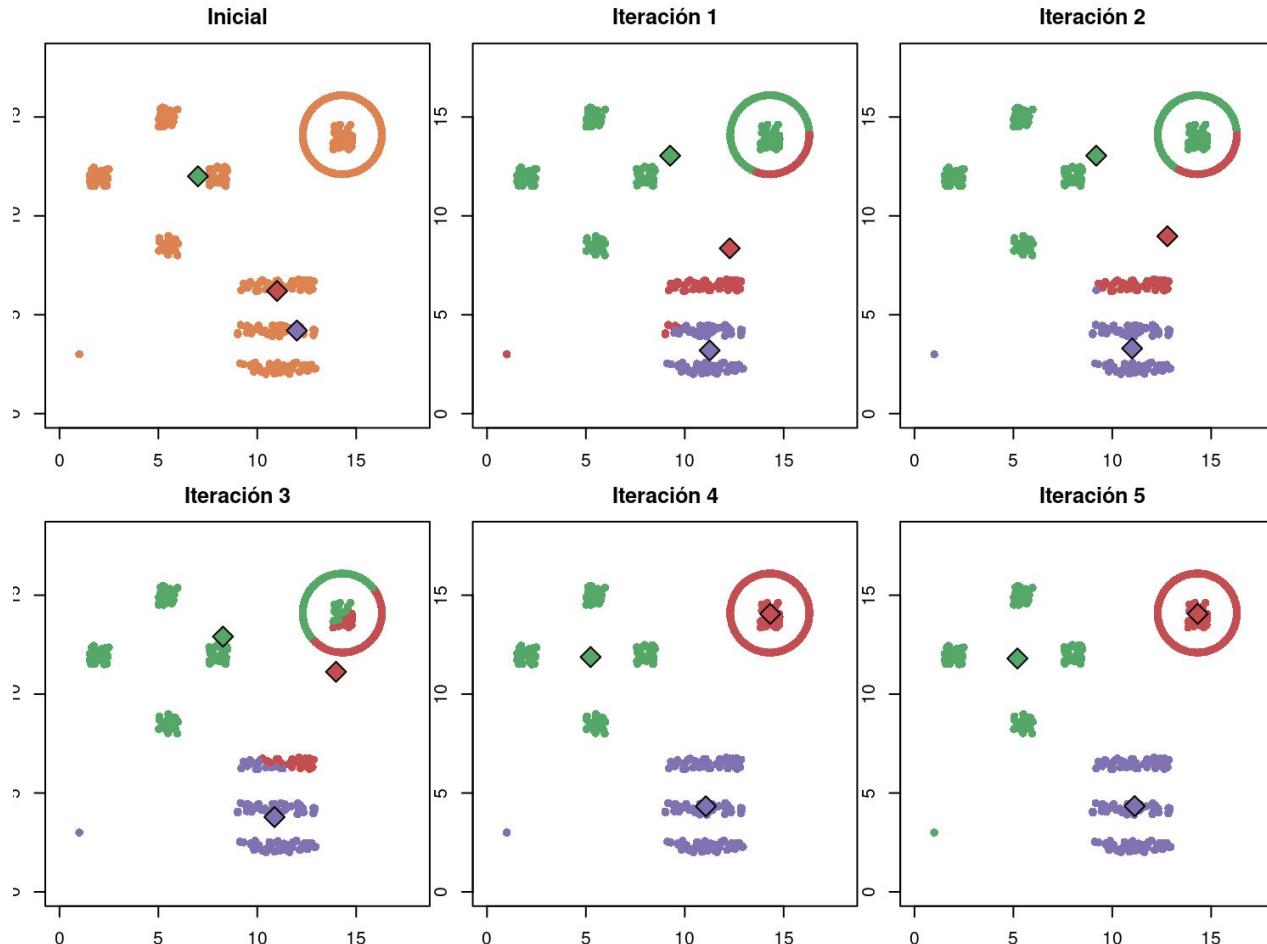
# Outliers



# ¿Qué sucede con los outliers?

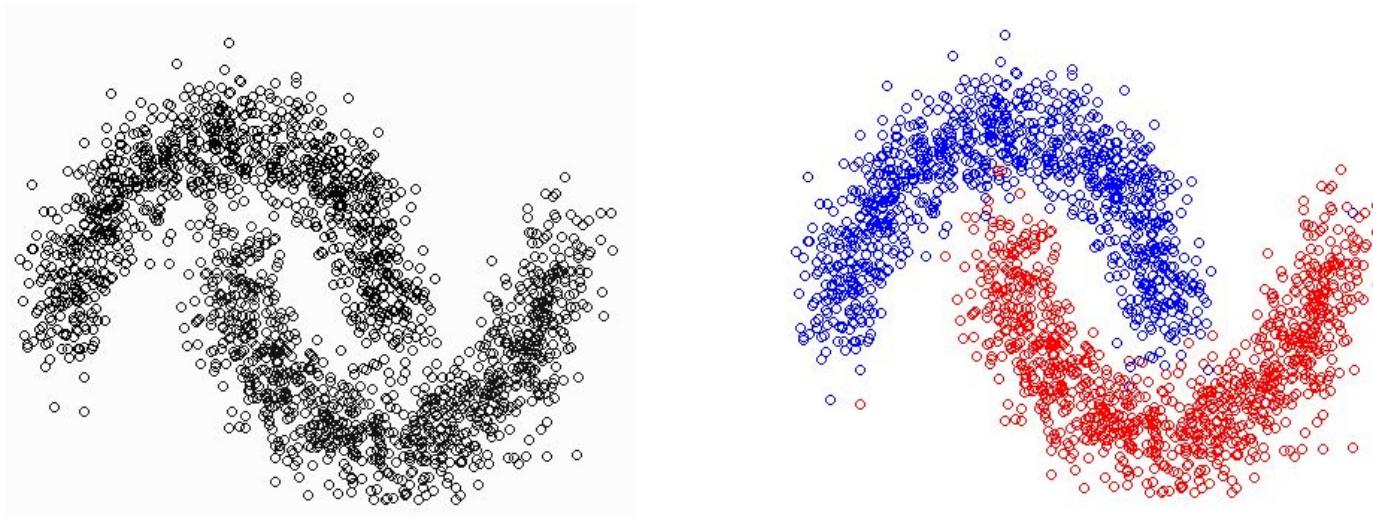


# ¿Qué sucede con los outliers?



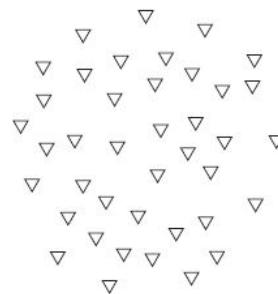
Problemas para encontrar clusters cuando:

- no tienen forma esférica
- tienen tamaños o densidades muy distintas

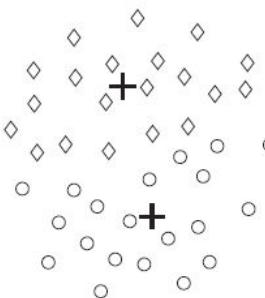


Problemas para encontrar clusters cuando:

- no tienen forma esférica
- tienen tamaños o densidades muy distintas



(a) Original points.



(b) Three K-means clusters.

Figure 8.10. K-means with clusters of different density.



# K-Means con scikit-learn

# K-medias (K-means)

## **Ventajas:**

- Rápido
- Fácil de implementar
- Útil en general

## **Desventajas:**

- Sensible a ruido y outliers
- Necesita especificar el número de clusters de antemano
- No es bueno cuando los datos tienen formas raras, o cuando hay mucha variabilidad en los tamaños de los clusters
- Sensibilidad a los valores iniciales de los centroides

# DBSCAN

Density-based spatial clustering of applications with noise

# Algoritmo DBSCAN

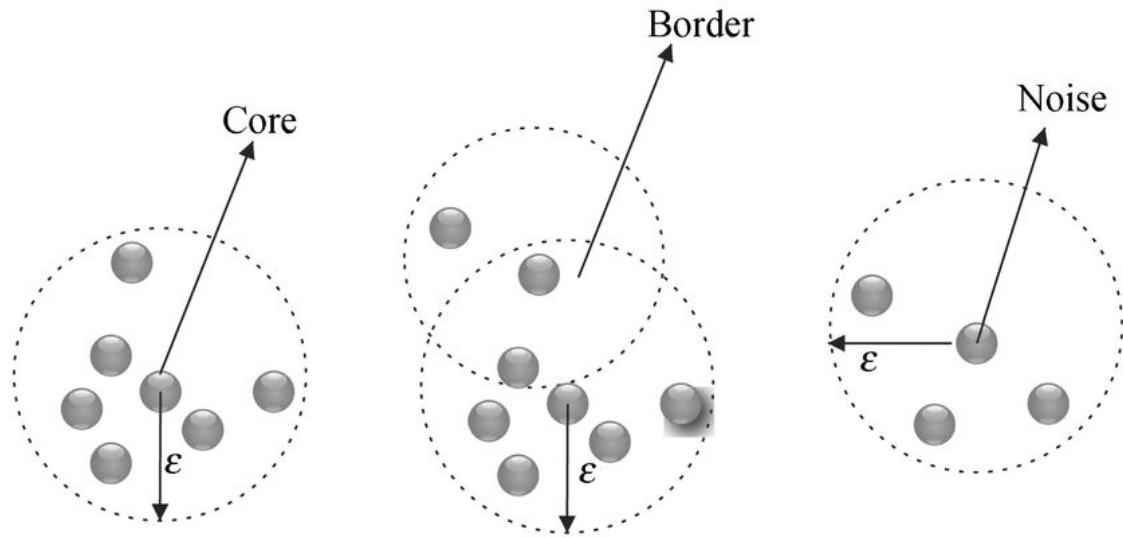
Parámetros:

Eps - Distancia para la vecindad

minPts - Cantidad de vecinos requeridos

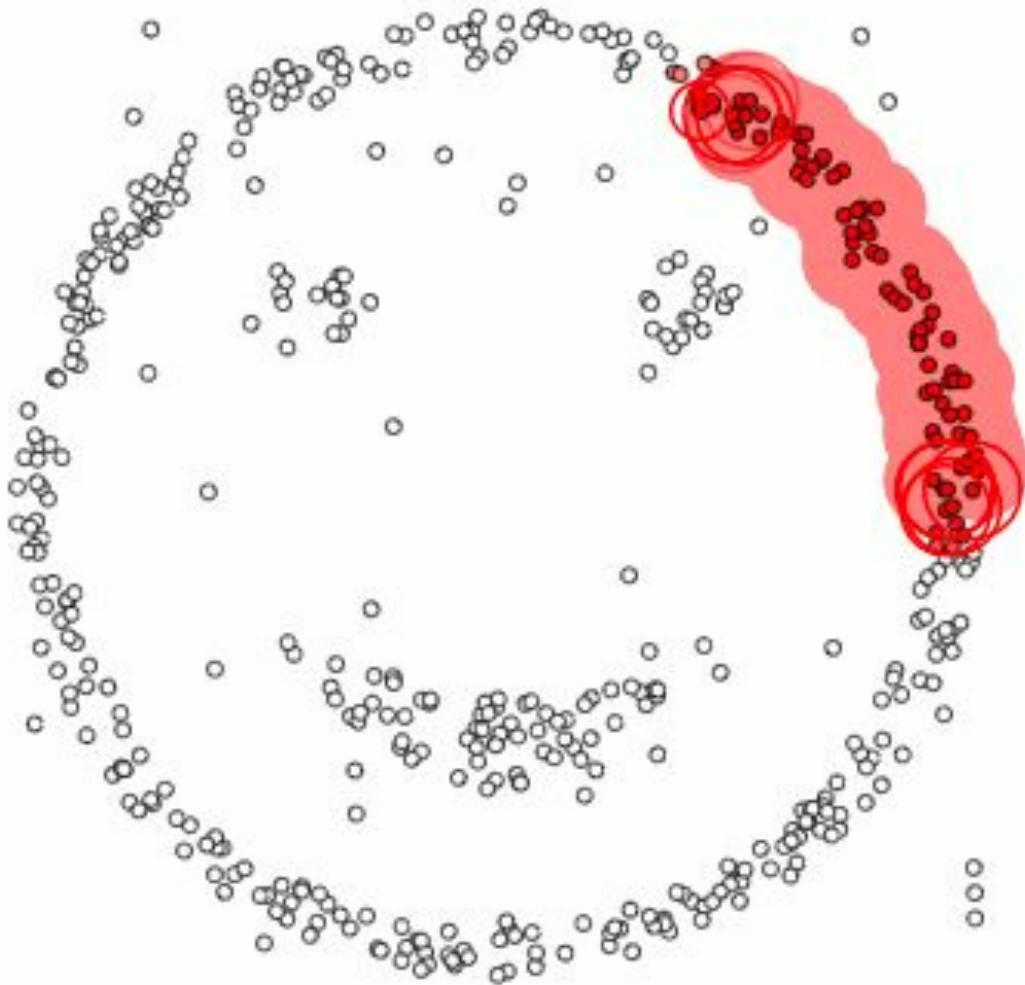
- Un punto p es un punto *núcleo* si al menos minPts puntos están a una distancia menor a Eps de él.
- Un punto q es *alcanzable* desde p si hay un camino de puntos alcanzables que va de p a q.
- Un punto que no sea alcanzable desde al menos minPts es considerado *ruido*.

Ejemplo con MinPts = 7



# Algoritmo DBSCAN

1. Para cada observación miramos el número de puntos a una distancia máxima **Eps** de ella. Esta zona se denomina **Eps**-vecindad de la observación.
2. Si una observación tiene al menos **minPts** vecinos, incluida ella misma, se considera una observación central. En este caso, se ha detectado una observación de alta densidad.
3. Todas las observaciones en la vecindad de una observación central pertenecen al mismo cluster. Puede haber observaciones centrales cercanas entre sí. Por lo tanto, de un paso a otro, se obtiene una larga secuencia de observaciones centrales que constituyen un único cluster.
4. Cualquier observación que no sea una observación central y que no tenga ninguna observación central en su vecindad se considera una anomalía/outlier.





# DBSCAN con scikit-learn

# Algoritmo DBSCAN

## **Ventajas:**

- No asume ningún número de clusters.
- Puede encontrar clusters con formas geométricas arbitrarias.
- Es robusto detectando outliers.
- No es susceptible al orden en que se encuentran los puntos dentro de la base de datos, ni a la inicialización.

## **Desventajas:**

- No puede agrupar bien conjuntos de datos con grandes diferencias en las densidades.
- Es muy sensible a los parámetros ( $\text{minPts}$  y  $\text{Eps}$ ) y a veces es difícil determinarlos.
- No es bueno para datasets muy grandes o en muchas dimensiones.

# Clustering Jerárquico

# Algoritmo Clustering Jerárquico

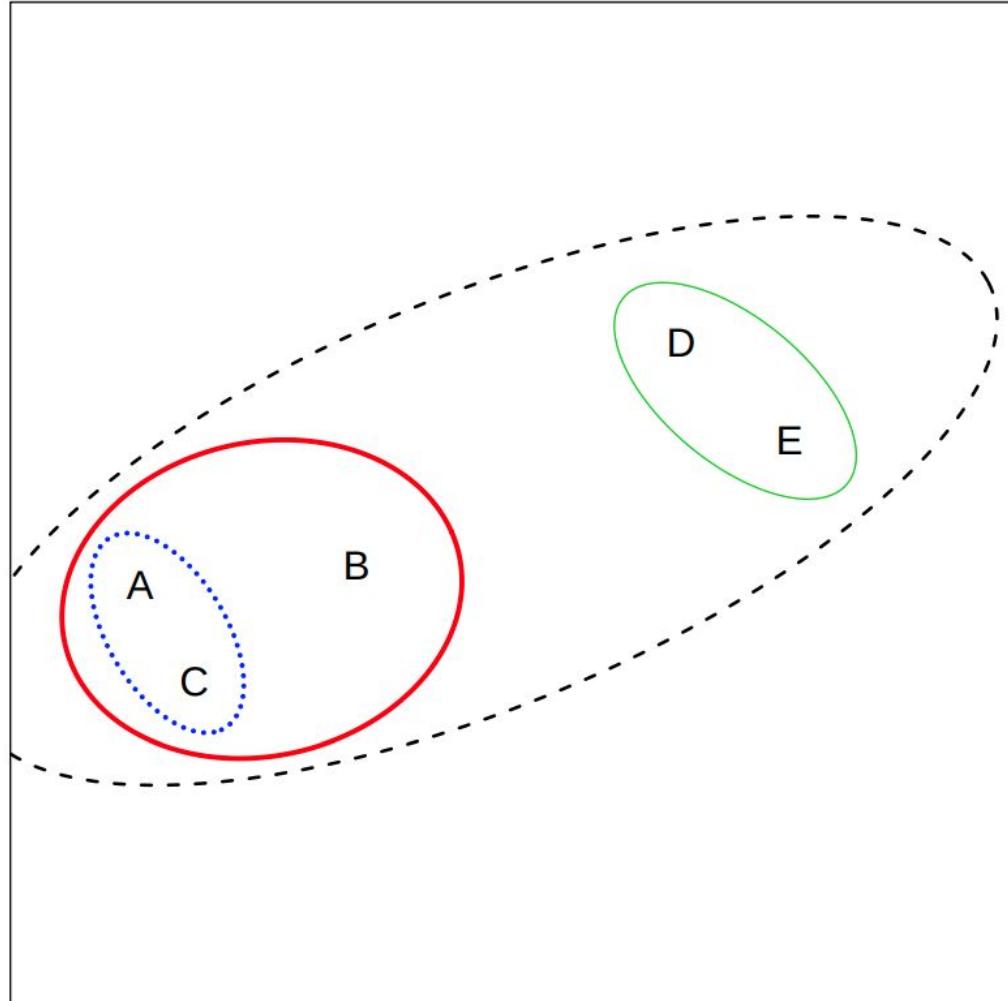
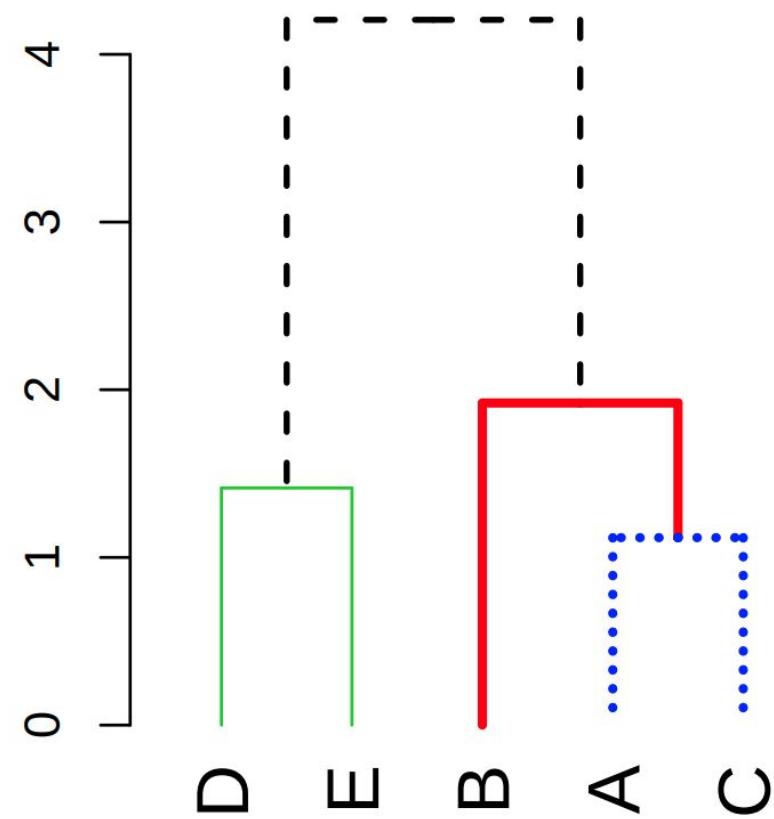
*aglomerativo - bottom up*

1. Cada punto forma un cluster
2. Computar matriz de cercanía
3. Repetir:
  - a. Buscar el par de clusters más similar y hacer un merge
  - b. Actualizar la matriz de proximidadhasta que haya un solo cluster

Este proceso genera un *dendrograma*.

# Ejemplo

del libro ISLP

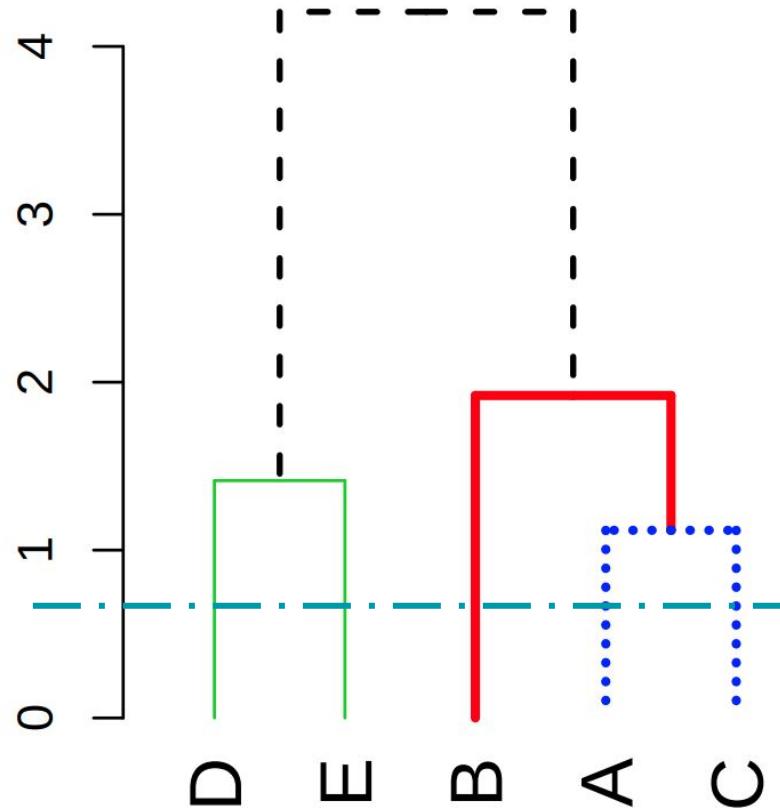


## Dendrograma

La *altura* representa la escala a la que se unen los clusters.

Una vez hecho el dendrograma, se lo puede cortar en una altura y así generar el clustering.

## Ejemplo del libro ISLP



# Criterios de cercanía

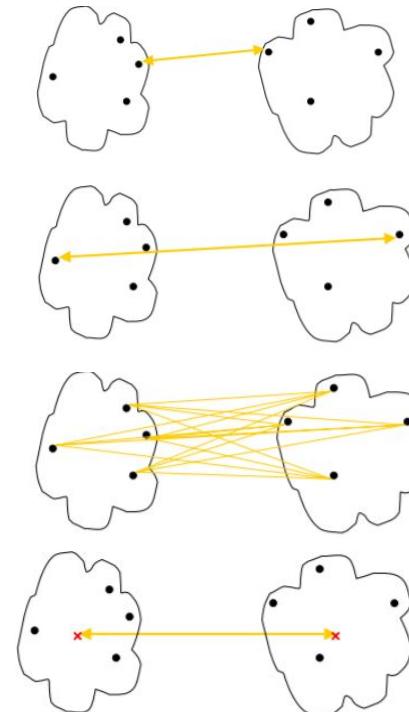
Entre dos puntos - la cercanía es la distancia (que depende de los atributos).  
Puede ser la distancia euclídea u otra.

¿Pero entre dos clusters que tienen varios puntos?

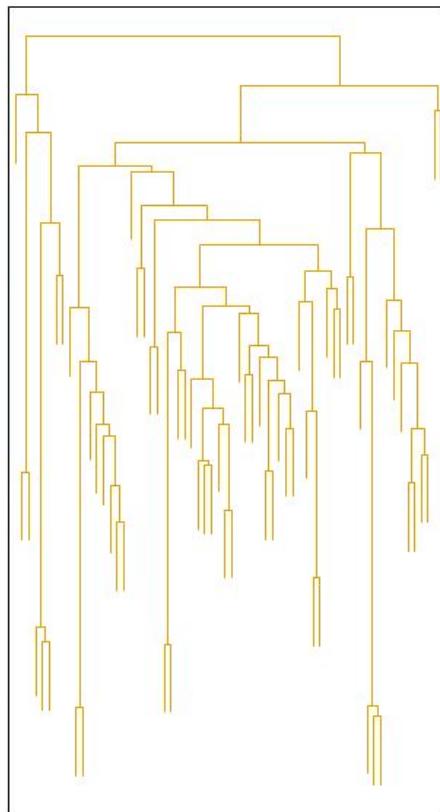
Hay que extender la noción de distancia entre puntos, a conjuntos de puntos.

# Medición de similaridad entre clusters

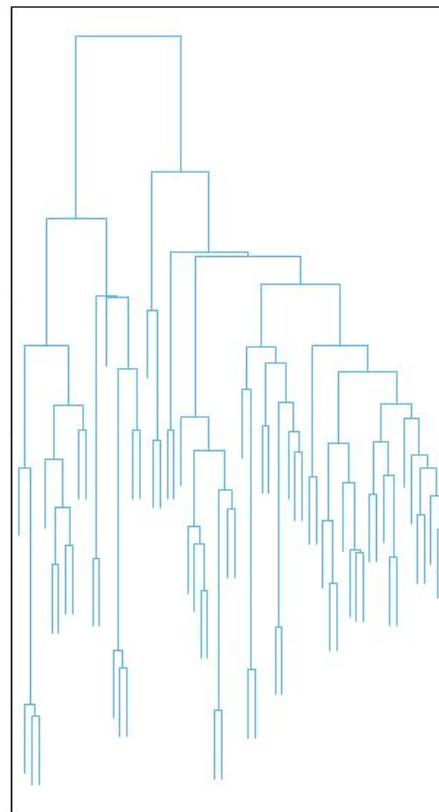
- **Single linkage:** distancia mínima entre dos puntos de los dos distintos clusters.
- **Complete linkage:** distancia máxima entre dos puntos de los distintos clusters
- **Average linkage:** promedio de la distancia entre los puntos de los clusters
- **Centroid linkage:** distancia entre centroides



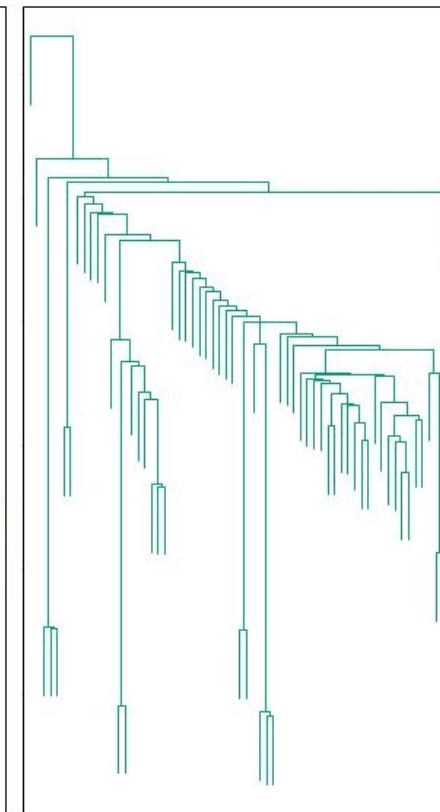
Average Linkage



Complete Linkage



Single Linkage



**FIGURE 12.14.** Average, complete, and single linkage applied to an example data set. Average and complete linkage tend to yield more balanced clusters.

## Ejemplo con single linkage

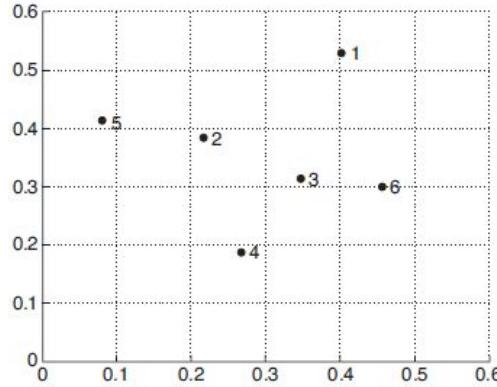


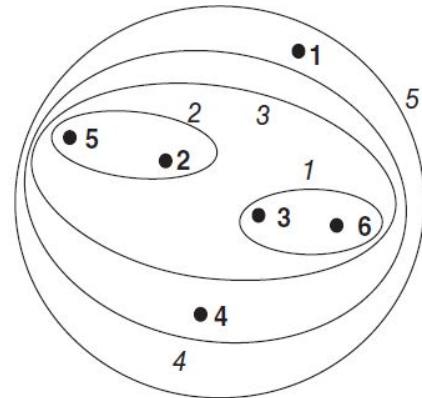
Figure 8.15. Set of 6 two-dimensional points.

	p1	p2	p3	p4	p5	p6
p1	0.00	0.24	0.22	0.37	0.34	0.23
p2	0.24	0.00	0.15	0.20	0.14	0.25
p3	0.22	0.15	0.00	0.15	0.28	0.11
p4	0.37	0.20	0.15	0.00	0.29	0.22
p5	0.34	0.14	0.28	0.29	0.00	0.39
p6	0.23	0.25	0.11	0.22	0.39	0.00

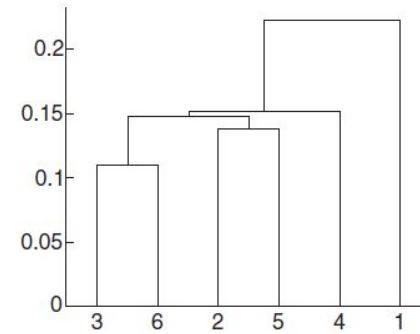
Table 8.4. Euclidean distance matrix for 6 points.

Point	x Coordinate	y Coordinate
p1	0.40	0.53
p2	0.22	0.38
p3	0.35	0.32
p4	0.26	0.19
p5	0.08	0.41
p6	0.45	0.30

Table 8.3.  $xy$  coordinates of 6 points.



(a) Single link clustering.



(b) Single link dendrogram.

$$\begin{aligned}
 dist(\{3, 6\}, \{2, 5\}) &= \min(dist(3, 2), dist(6, 2), dist(3, 5), dist(6, 5)) \\
 &= \min(0.15, 0.25, 0.28, 0.39) \\
 &= 0.15.
 \end{aligned}$$

# Clustering jerárquico

## **Ventajas:**

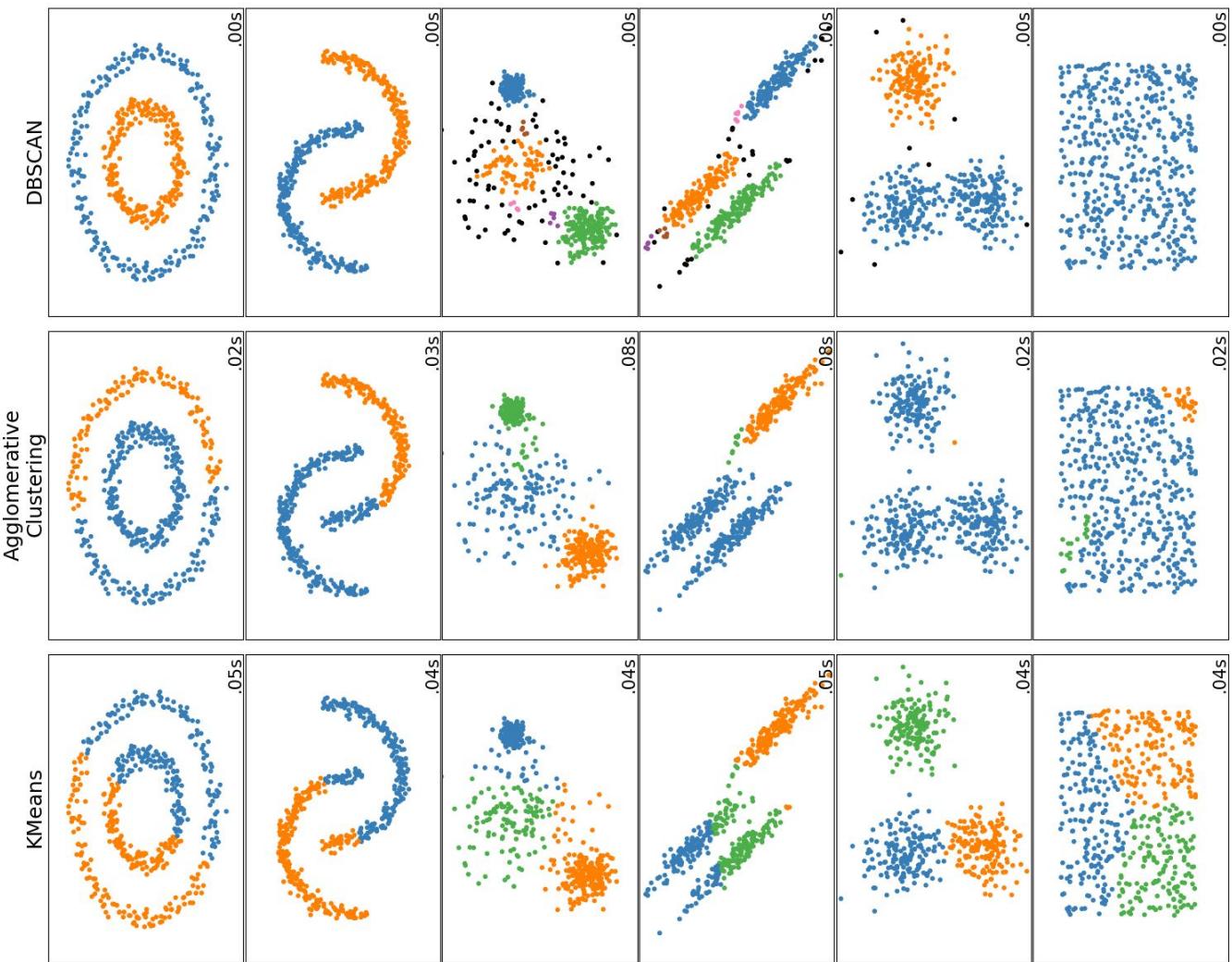
- No asume ningún número de clusters (se pueden obtener cortando el dendograma en el nivel deseado)
- Genera un dendrograma: puede ser útil para interpretar
- Pueden corresponder a taxonomías (ej. reino animal)

## **Desventajas:**

- Sensible a ruido y outliers
- Computacionalmente caro en tiempo y en espacio
- No siempre la estructura jerárquica es la más adecuada
- Optimiza localmente, no de manera global



# Clustering jerárquico con scikit-learn



# Cierre

1. Aprendizaje No Supervisado
2. KMeans
3. DBSCAN
4. Jerárquico