

ACÀMICA

---

# ¡Bienvenidas/os a Data Science!



# Agenda

---

¿Cómo anduvieron?

Actividad: Data Science en mi vida

Repaso: Aprendizaje no supervisado

Explicación: Evaluación de clusters

Break

Hands-On

Cierre



# ¿Dónde estamos?



# ¿Cómo anduvieron?



# Actividad: Data Science en mi vida



# Data Science en mi vida

¡Preparen sus charlas relámpago!

En 7 minutos con 7 slides comparte con tus compañeros:

En qué problemas estás aplicando lo aprendido en Data Science y cómo lo estás haciendo.

O bien, en qué problemas te gustaría aplicar Data Science y cómo lo harías.

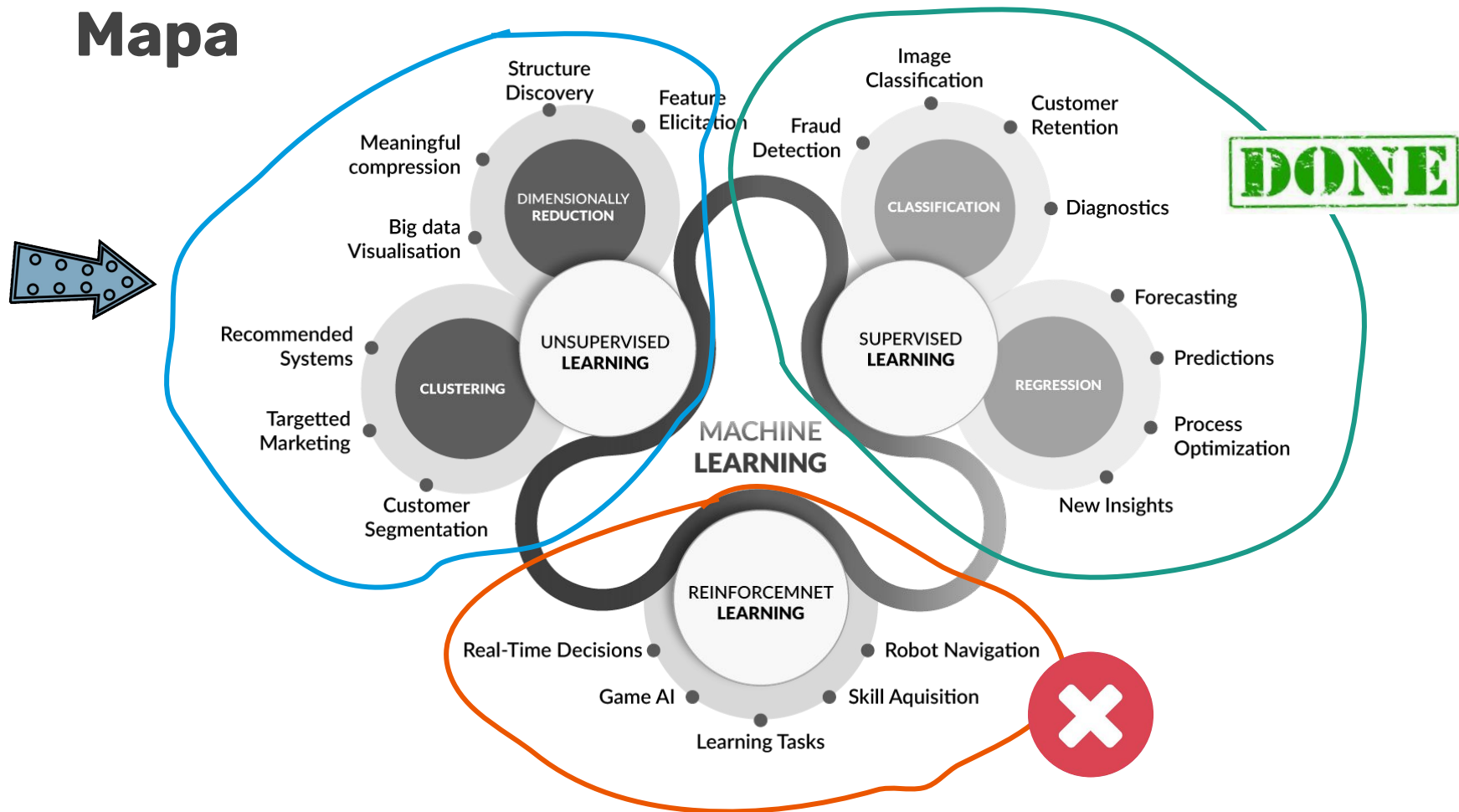
¡Elige algún tema o proyecto que te interese y relaciónalo con lo aprendido!

# Repaso: Aprendizaje no supervisado - Clustering





# Mapa



# Solo datos

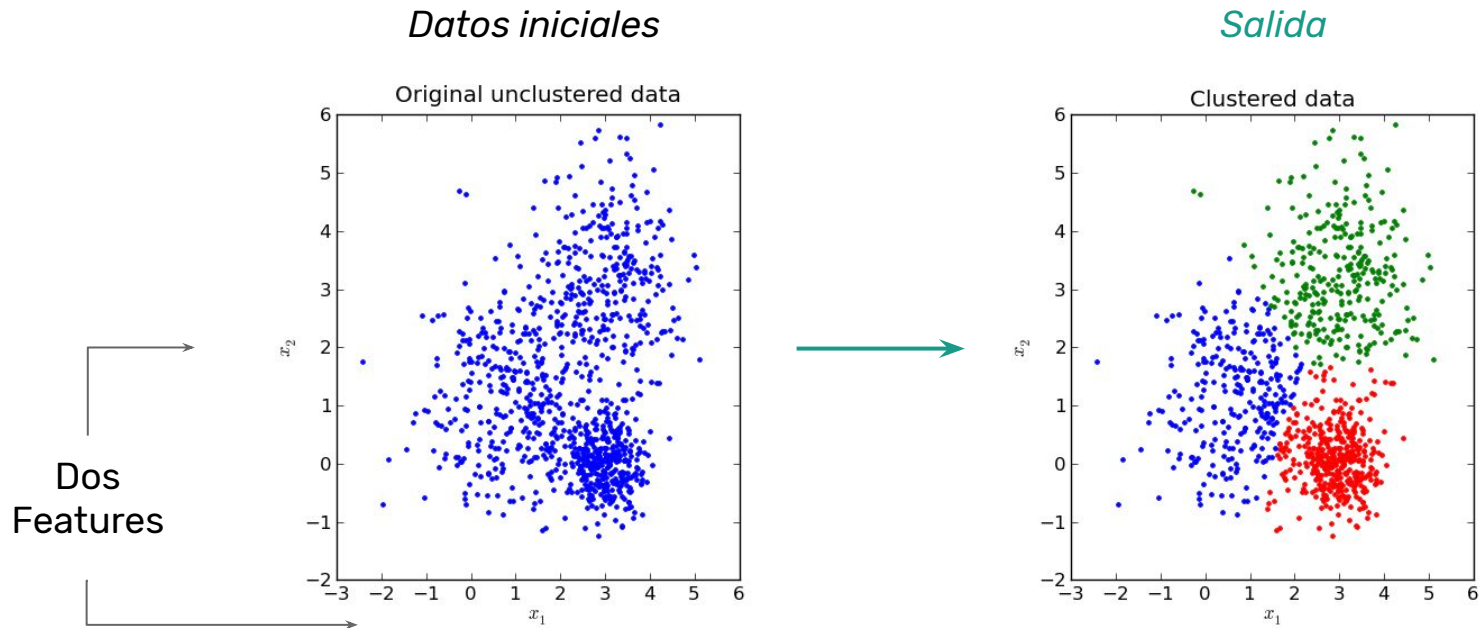
Llamamos **Aprendizaje No Supervisado** a los métodos para trabajar con datos (instancias) que no tienen asociados una etiqueta (una clase o un valor).

Los objetivos principales en Aprendizaje No Supervisado son:

- Clustering
- Reducción de dimensionalidad

# Aprendizaje No Supervisado • Clustering

Dado un set de datos, nuestro objetivo será encontrar grupos (clusters) en los cuales las instancias pertenecientes sean parecidas (estén “cerca”).



# Aprendizaje No Supervisado • Clustering

## ¿Para qué sirve?

Encontrar grupos en los datos puede ayudar en problemas de:

- Investigación de mercado
- Sistemas de recomendación
- Medicina
- Biología (genética y especies)
- Muchísimas mas cosas

## ¿Cómo se hace?

Algunos de los algoritmos para hacer clustering son:

- K-means
- DBSCAN
- Hierarchical Clustering (aglomerativo)
- Fuzzy C-Means (como K-means pero permite overlap)
- GMM: Gaussian Mixture Models (supone distribucion gaussiana)

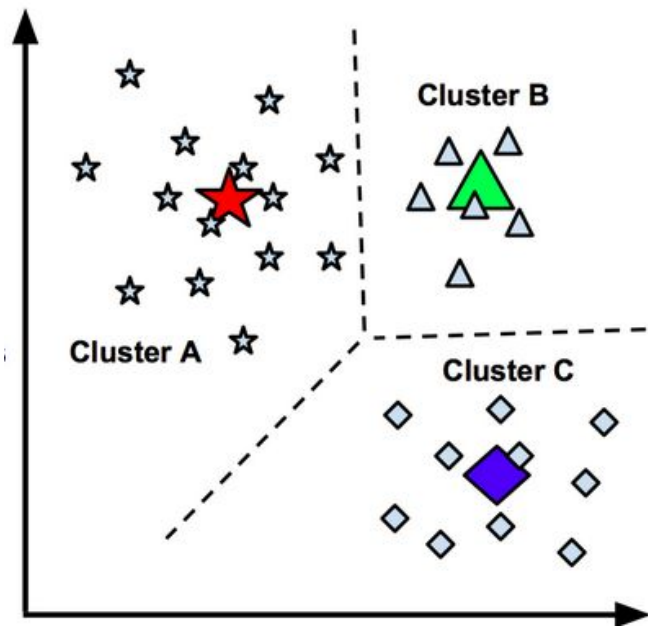
Aprendizaje No Supervisado

# Clustering: K-Means



# Clustering • K-Means

Objetivo: Separar los datos en  $k$  clusters (número dado) ubicando a las instancias que estén dentro de una región cercana dentro de un mismo cluster.



**Idea:** encontrar un número  $k$  de centros (centroids), uno por cada cluster, de manera tal que la distancia entre los centros y los datos más cercanos sea la mínima posible.

Luego cada instancia se identifica en el grupo del centroide más cercano.

# Clustering • K-Means

**¿Cómo se hace?** Se utiliza un algoritmo iterativo hasta llegar al resultado.

## **1) Se inicializan los k Centroides.**

La ubicación inicial puede ser aleatoria o con algún criterio.

## **2) Encontrar el centroide más cercano.**

Se asigna a cada instancia al centroide más cercano (el significado de “cercano” puede cambiar, es un hiperparámetro)

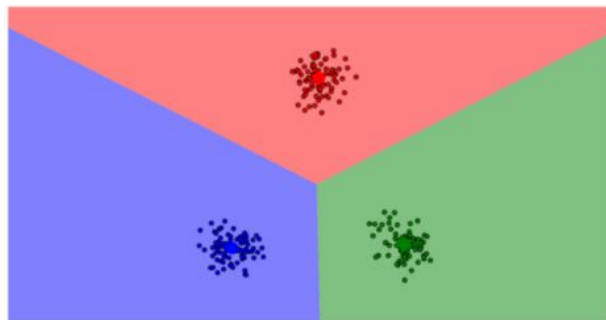
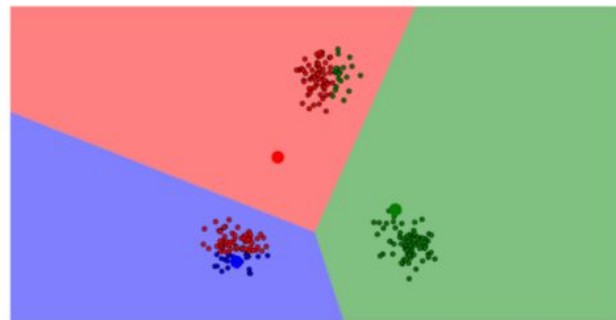
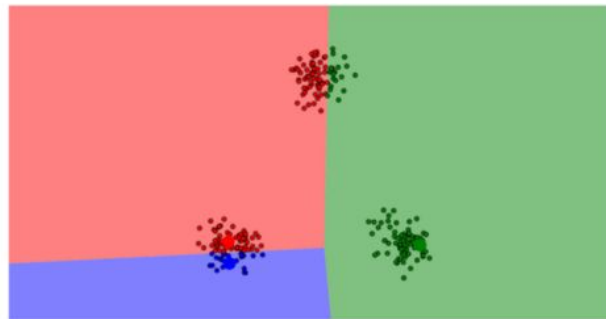
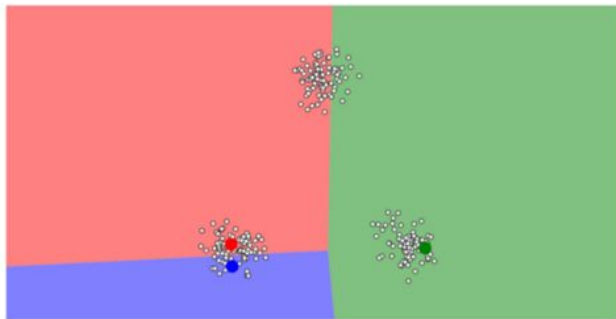
## **3) Actualizar los centroides.**

La nueva posición del centroide es el promedio de las posiciones de las instancias en ese cluster (de acá viene el means).

## **4) Repetir pasos 2 y 3.**

Se repiten los updates hasta que la posición del centroide ya no varíe

# Clustering • K-Means





Aprendizaje No Supervisado

# Clustering: DBSCAN

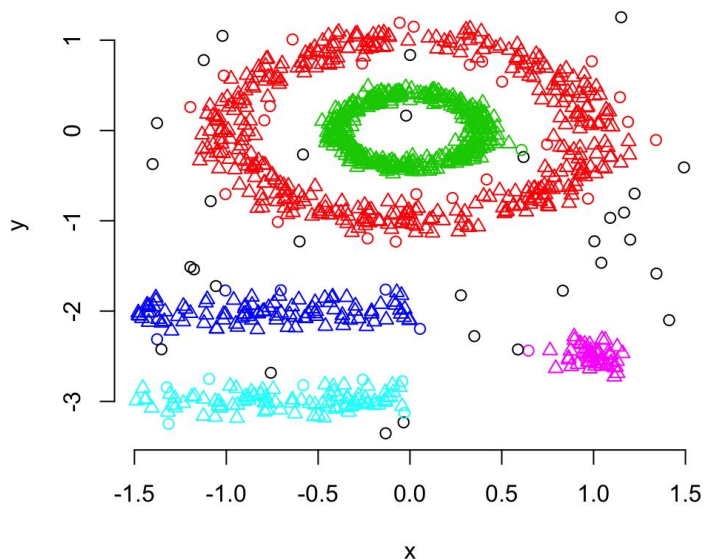


# Clustering • DBSCAN

DBSCAN = **Density-Based Spatial Clustering of Applications with Noise**

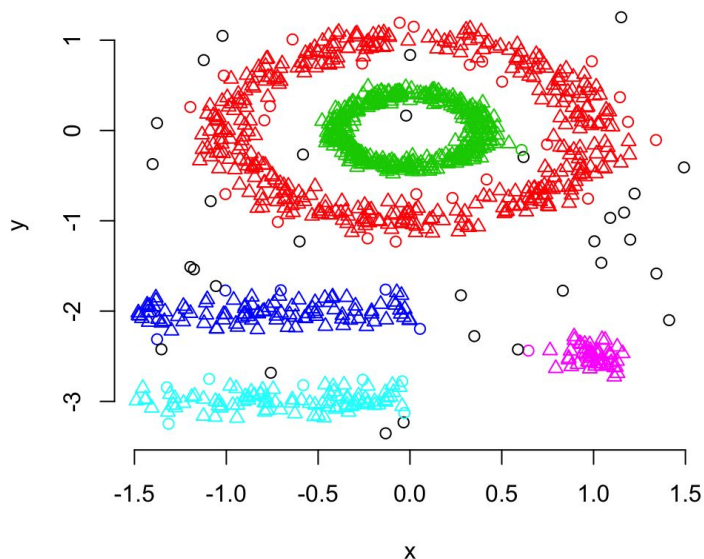
# Clustering • DBSCAN

Objetivo: Identificar un número arbitrario de clusters. Los clusters estarán definidos por densidad de puntos. Puede haber puntos que no pertenezcan a ningún cluster (noise).



# Clustering • DBSCAN

Objetivo: Identificar un número arbitrario de clusters. Los clusters estarán definidos por densidad de puntos. Puede haber puntos que no pertenezcan a ningún cluster (**noise = OUTLIERS**).



**Idea:** recorrer todo el dataset e ir identificando las zonas de puntos densamente pobladas como pertenecientes a un mismo cluster.

Los puntos aislados serán reconocidos como ruido.



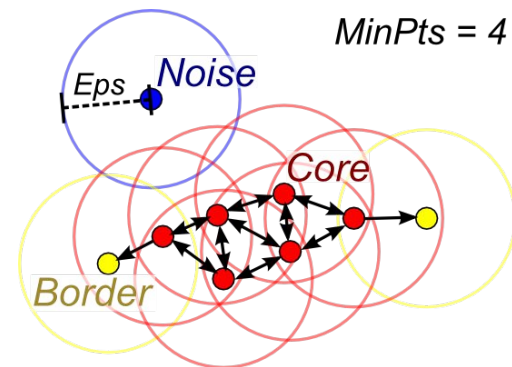
# Clustering • DBSCAN

## ¿Cómo se hace?

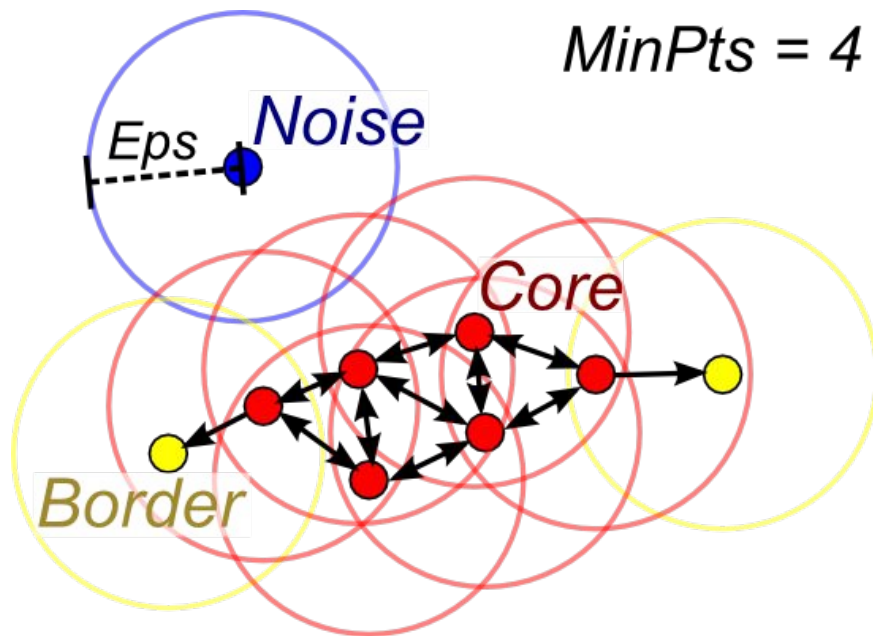
**1) Se define una distancia epsilon (parámetro)** como la vecindad de un punto. Se elige un número de puntos mínimos de para considerar un cluster minPoints (parámetro).

**2) Luego se realiza el siguiente proceso sobre todos los puntos del dataset:**

- 1) Se toma un punto no visitado random. Se identifica si el punto es un 'core', es decir, si tiene minPoints en su vecindario. Si no tiene, se lo llama 'noise'. Este punto se marca como visitado.
- 2) Si es un core, se le asigna un nuevo cluster y todos los puntos de su vecindario se consideran dentro de su cluster. Si alguno de estos puntos también son cores, este proceso se repite. A los puntos asignados a un cluster que no son core, se los llama 'border'. Todos se marcan como visitados.
- 3) Este proceso se repite hasta que todos los puntos hayan sido visitados.



# Clustering • DBSCAN



# ¿Cómo se comparan K-Means y DBSCAN?



# K-Means



- Rápido, muy mucho.  $O(n)$ .
- No tiene parámetros
- Fácil asignar nuevas instancias



- Hay que definir el número de clusters
- Solo funciona bien con clusters tipo esferas
- Sensible a outliers (afectan el promedio)

# DBSCAN

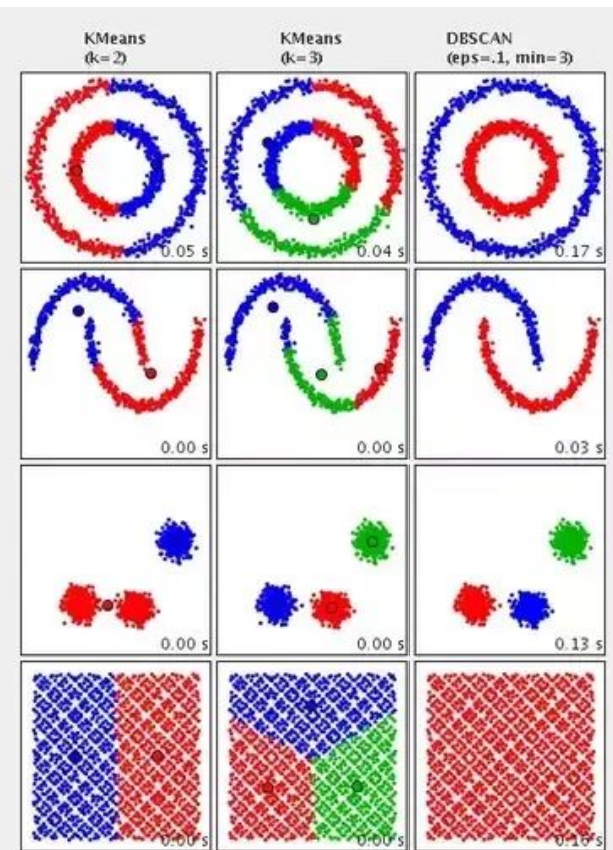
- No hay que elegir el número de clusters
- Detecta cualquier forma de clusters
- Determina automáticamente datos outliers
- Hay que elegir bien los parámetros
- No anda bien si hay clusters de diferentes densidades
- Es computacionalmente más costoso (tarda más)



# Clustering • K-Means vs. DBSCAN

## ¿Cómo funcionan?

Es muy común ver este tipo de representaciones de los métodos, donde muestra cómo categorizan distintos datasets y cuanto tardan en hacerlo.



Aprendizaje No Supervisado

# Evaluación de clusters



# Evaluación: Distancia al centroide

Buscamos una medida para la validación e interpretación de clusters en un dataset.

**Idea:** medimos cuál es la distancia media de cada dato al centroide más cercano.

# Evaluación: Distancia al centroide

Buscamos una medida para la validación e interpretación de clusters en un dataset.

$$d(i) = \|\mathbf{X}_i - \mathbf{C}_j\|^2$$

Distancia de  
cada dato

Posición  
del dato  $i$

Posición del centroide  
más cercano

**Idea:** medimos cuál es la distancia media de cada dato al centroide más cercano.

# Evaluación: Distancia al centroide

## (K-Means)

Buscamos una medida para evaluar qué tan bien resulta el clustereo en K-Means.

**Idea:** medimos cuál es la distancia media de cada dato al centroide más cercano.

# Evaluación: Distancia al centroide

## (K-Means)

Buscamos una medida para evaluar qué tan bien resulta el clustereo en K-Means.

$$D = \frac{1}{N} \sum_{i=1}^N d(i)$$

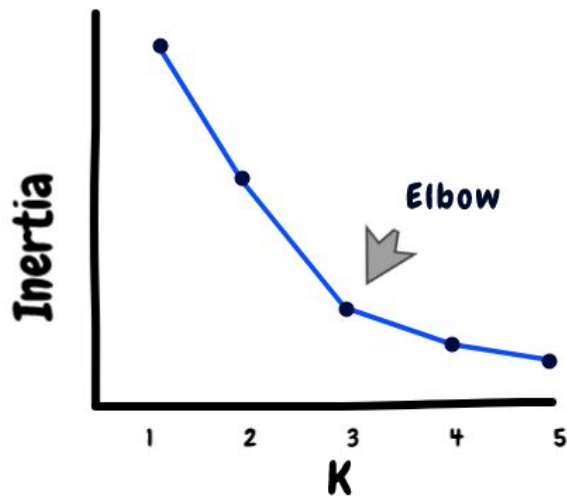
**Distancia  
media total**

**Idea:** medimos cuál es la distancia media de cada dato al centroide más cercano.

# Evaluación: Distancia al centroide

## (K-Means)

Buscamos una medida para evaluar qué tan bien resulta el clustereo en K-Means.



**Idea:** medimos cuál es la distancia media de cada dato al centroide más cercano.

• **En sklearn:** Luego de fitear el modelo, la variable 'model.inertia\_' tiene esta información.

**K ÓPTIMO:** Buscar donde esta el 'codo' de la curva. El valor de inercia siempre descende con el número de clusters.

# Evaluación: Silhouette

**Silhouette:** es una medida para la validación e interpretación de clusters en un dataset (para cualquier método de clustering que hayan usado).

Definición:

---

$$s(i) = \begin{cases} 1 - a(i)/b(i), & \text{if } a(i) < b(i) \\ 0, & \text{if } a(i) = b(i) \\ b(i)/a(i) - 1, & \text{if } a(i) > b(i) \end{cases}$$

**Idea:** medimos qué tan parecidos son los datos con su propio cluster (cohesión) en comparación con qué tan parecidos son a otros clusters (separación).

**s(i):** es el valor de silhouette para el dato i.

**a(i):** distancia media del dato i con el resto de su cluster

**b(i):** distancia media del dato i con el cluster más cercano



# Evaluación: Silhouette

Definición:

---

$$s(i) = \begin{cases} 1 - a(i)/b(i), & \text{if } a(i) < b(i) \\ 0, & \text{if } a(i) = b(i) \\ b(i)/a(i) - 1, & \text{if } a(i) > b(i) \end{cases}$$

**s(i):** es el valor de silhouette para el dato i.

**a(i):** distancia media del dato i con el resto de su cluster

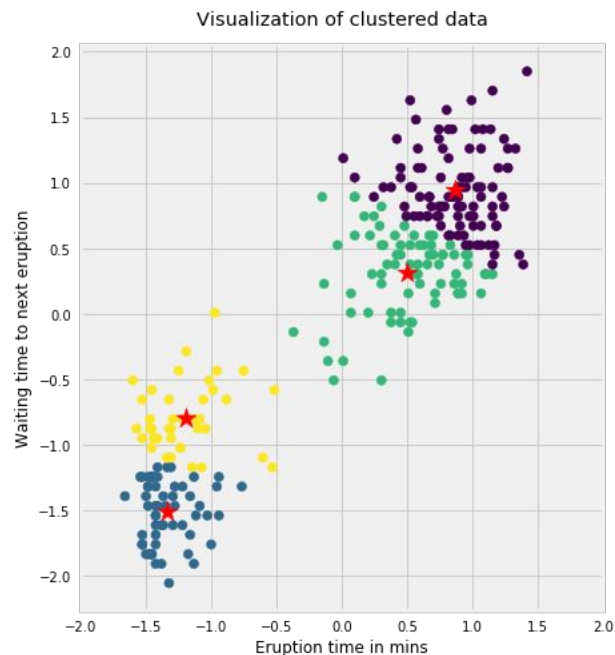
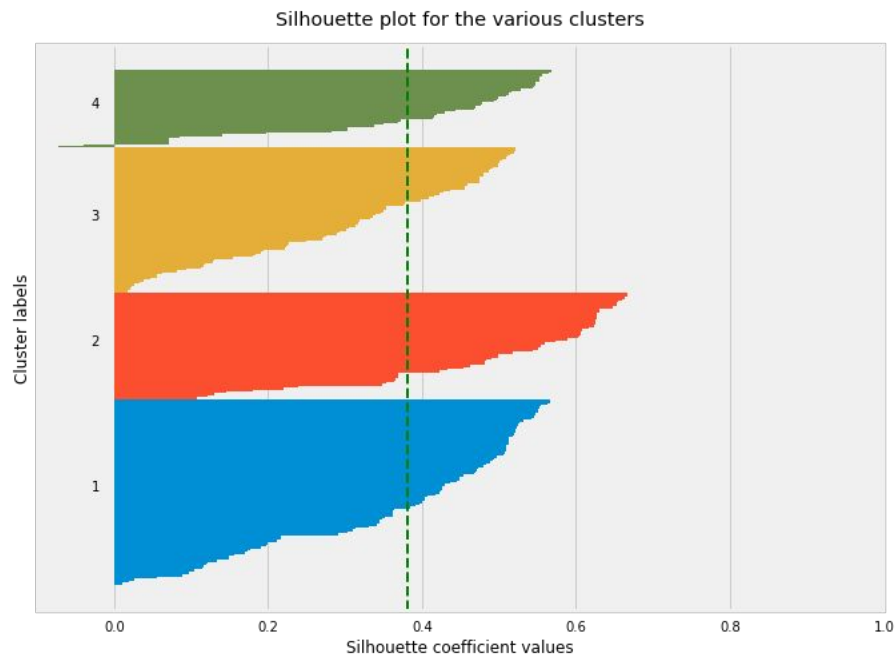
**b(i):** distancia media del dato i con el cluster más cercano

Esto nos da una medida para cada dato de qué tan bien está ubicado en los clusters. Para una medida de todo el conjunto, se toma la media de todos los s(i).

# Evaluación: Silhouette

**Resultados:** se suele mirar el perfil de todos los datos, buscamos que sea parejo.

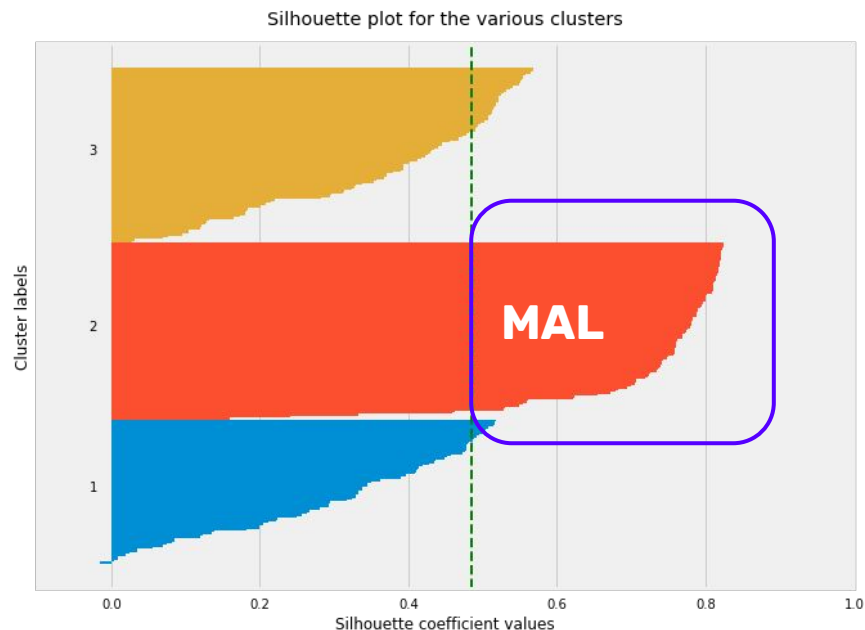
Silhouette analysis using  $k = 4$



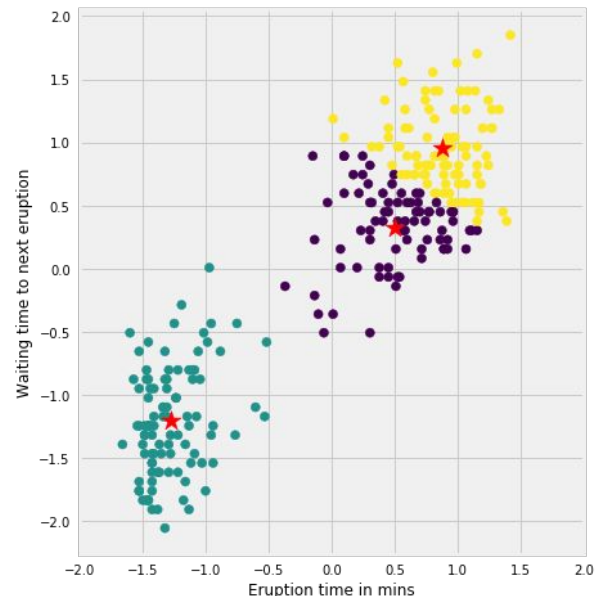
# Evaluación: Silhouette

**Resultados:** se suele mirar el perfil de todos los datos, buscamos que sea parejo.

Silhouette analysis using  $k = 3$



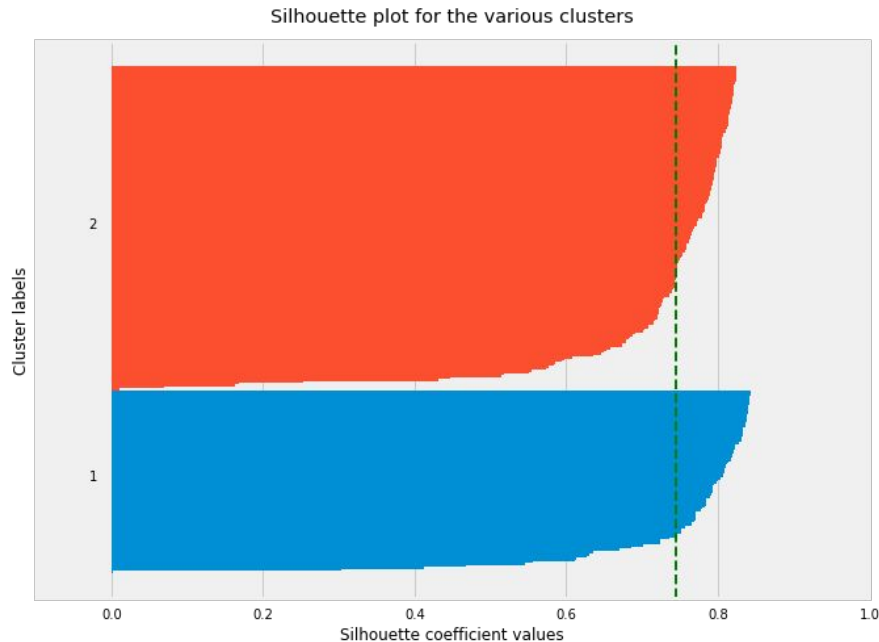
Visualization of clustered data



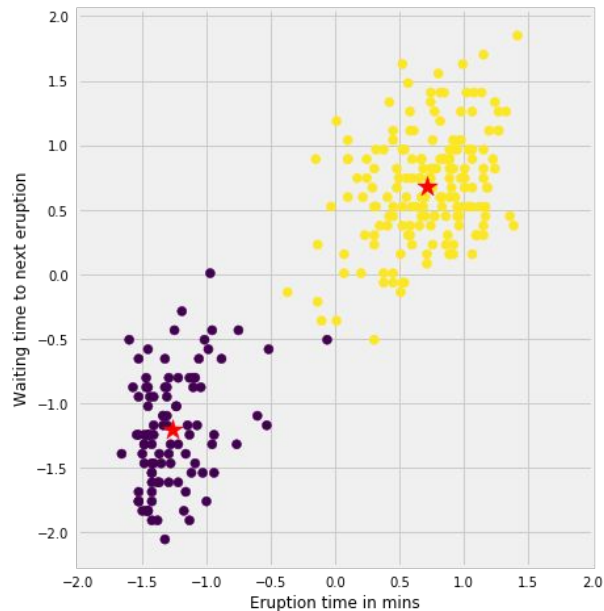
# Evaluación: Silhouette

**Resultados:** se suele mirar el perfil de todos los datos, buscamos que sea parejo.

Silhouette analysis using  $k = 2$



Visualization of clustered data



A close-up photograph of a white ceramic cup filled with a latte. The surface of the milk is decorated with intricate latte art, featuring a central heart shape surrounded by concentric, wavy lines. The cup is placed on a matching white saucer. In the background, a white napkin and a silver fork are visible, though they are out of focus. The overall lighting is soft and even, highlighting the textures of the coffee and the smooth surface of the cup.

**¡BREAK!**

---

Ph. Credit: Drew Coffmann



# Hands-on training



## Hands-on training

DS\_Encuentro\_37\_Clustering\_Metricas.ipynb



# Para la próxima

---

1. Ver los videos de “Aprendizaje No Supervisado: Reducción de Dimensionalidad”.
2. Completar los notebooks de hoy y atrasados.



ACÀMICA