

ACÀMICA

---

# ¡Bienvenidas/os a Data Science!



# Agenda

---

Data Science en mi vida (pendiente)

Dónde estamos

Repaso final nb NLP

Explicación: Aprendizaje no supervisado

Break

Hands-On

Cierre



# Actividad: Data Science en mi vida



# Actividad:

## Repaso NB pendiente

## NLP



# PAUSA ACTIVA









# ¿Dónde estamos?





# Cronograma

bloque

entrega

tiempo

## ADQUISICIÓN Y EXPLORACIÓN

## MODELADO

## DEPLOY

Exploración  
de datos

Feature  
Engineering

Regresión

Optimización de  
parámetros

Procesam. del  
lenguaje natural

**Sistema de  
recomendación**

Publicación de  
modelos

SEM 1

SEM 5

SEM 8

SEM 12

SEM 13

SEM 18

SEM 23

SEM 2

SEM 6

SEM 9

SEM 14

SEM 19

SEM 24

SEM 3

SEM 7

SEM 10

SEM 15

SEM 20

SEM 4

SEM 11

SEM 16

SEM 21

SEM 17

SEM 22

APRENDIZAJE SUPERVISADO

**APRENDIZAJE NO  
SUPERVISADO**



# BLOQUE 2 (Parte 1)

Regresión	Semana 8	Machine Learning Clasificación, Árboles de decisión, Train test split
	Semana 9	KNN, métricas para la clasificación Conceptos generales Machine Learning <i>Práctica integradora</i>
	Semana 10	Regresión (Regresión Lineal, Árboles de decisión, KNN, métricas) Validación Cruzada y selección de modelos
	Semana 11	Datasets Desbalanceados + Teorema de Bayes Curva ROC <i>Trabajo en el proyecto</i>
Optimización de parámetros	Semana 12	Optimización de parámetros - Validación cruzada y Gridsearch + lanzamiento entrega 4 <i>Trabajo en el proyecto</i>



# BLOQUE 2 (Parte 2)

## Procesamiento del lenguaje natural

Semana 13	Modelos avanzados - SVM Sesgo y Varianza
Semana 14	Ensamblados, Bagging, Random forest Ensamblados, Boosting
Semana 15	Redes Neuronales: Descenso por gradiente Redes Neuronales: Perceptrón
Semana 16	Redes Neuronales: Perceptrón Multicapa Redes Neuronales: Repaso
Semana 17	Procesamiento del lenguaje natural (NLP)

Semana 18 **Trabajo sobre el proyecto**  
Intro aprendizaje no supervisado + Clustering

## Sistema de recomendación

Semana 19	Métricas de evaluación para clustering Reducción de dimensionalidad: SVD
Semana 20	PCA Sistemas de recomendación
Semana 21	Sistemas de recomendación Ecosistema digital <b>Trabajo sobre el proyecto</b>
Semana 22	Ecosistema digital Puesta en producción



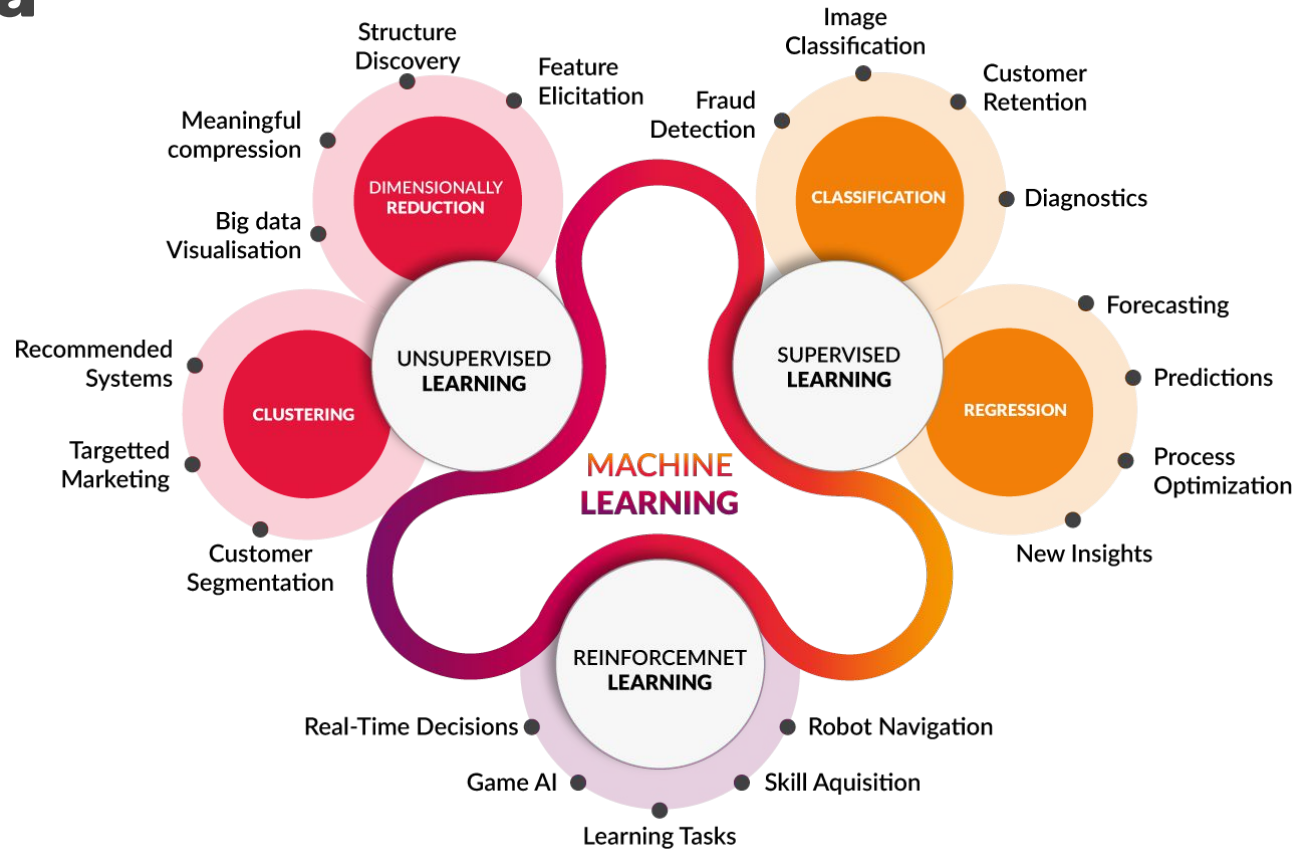
# ¿Cómo anduvieron?



# Aprendizaje no supervisado

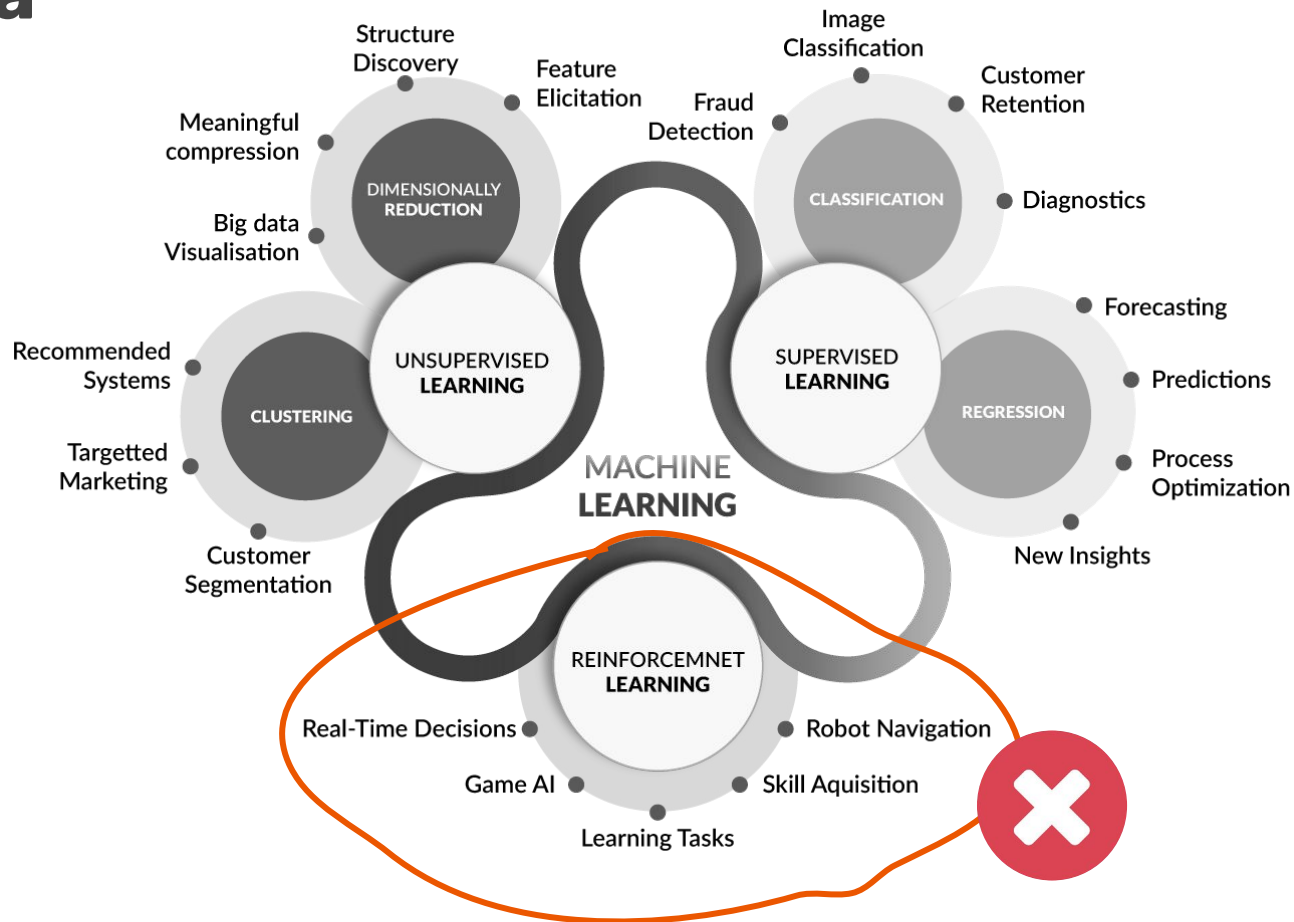


# Mapa

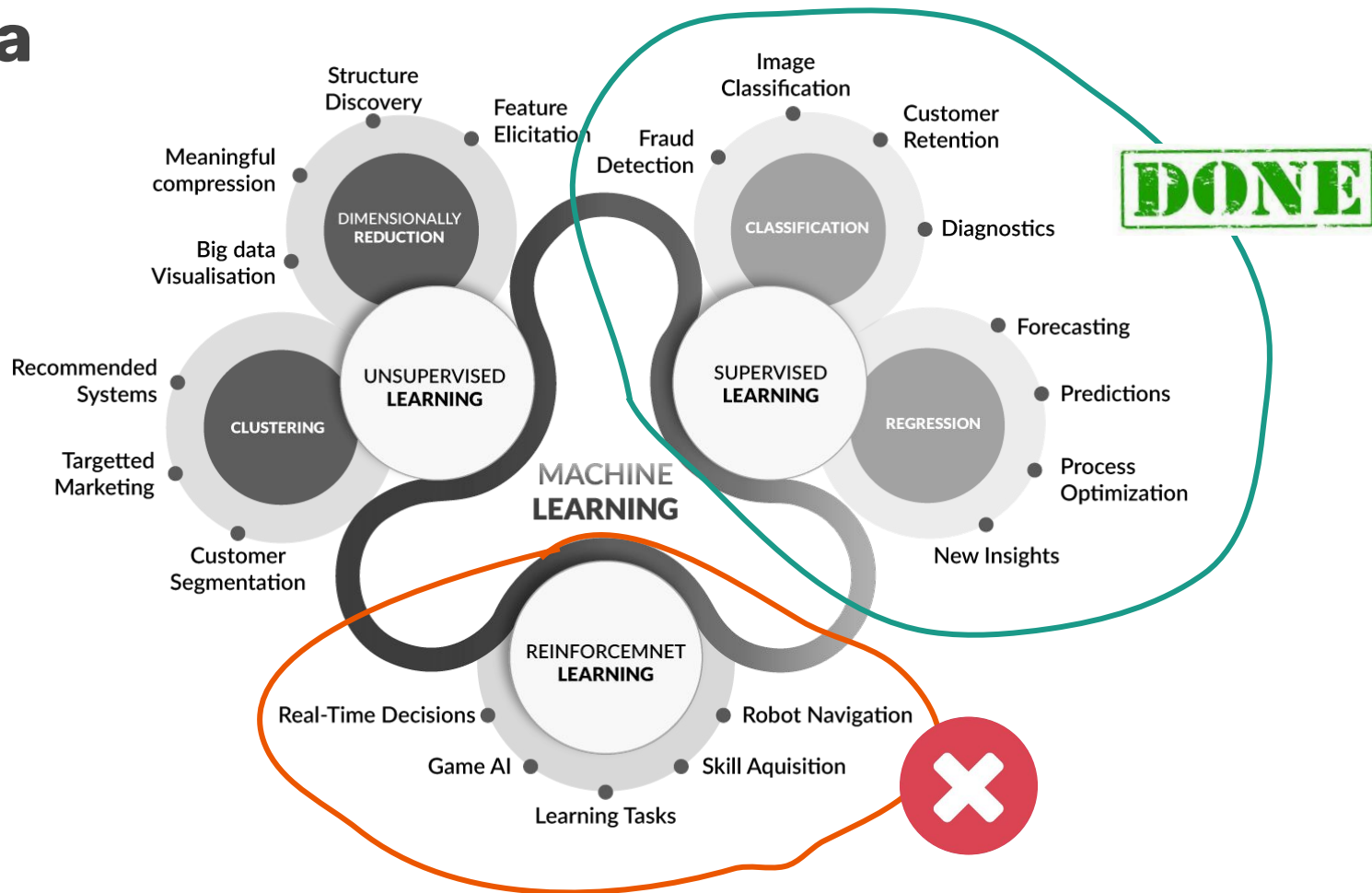




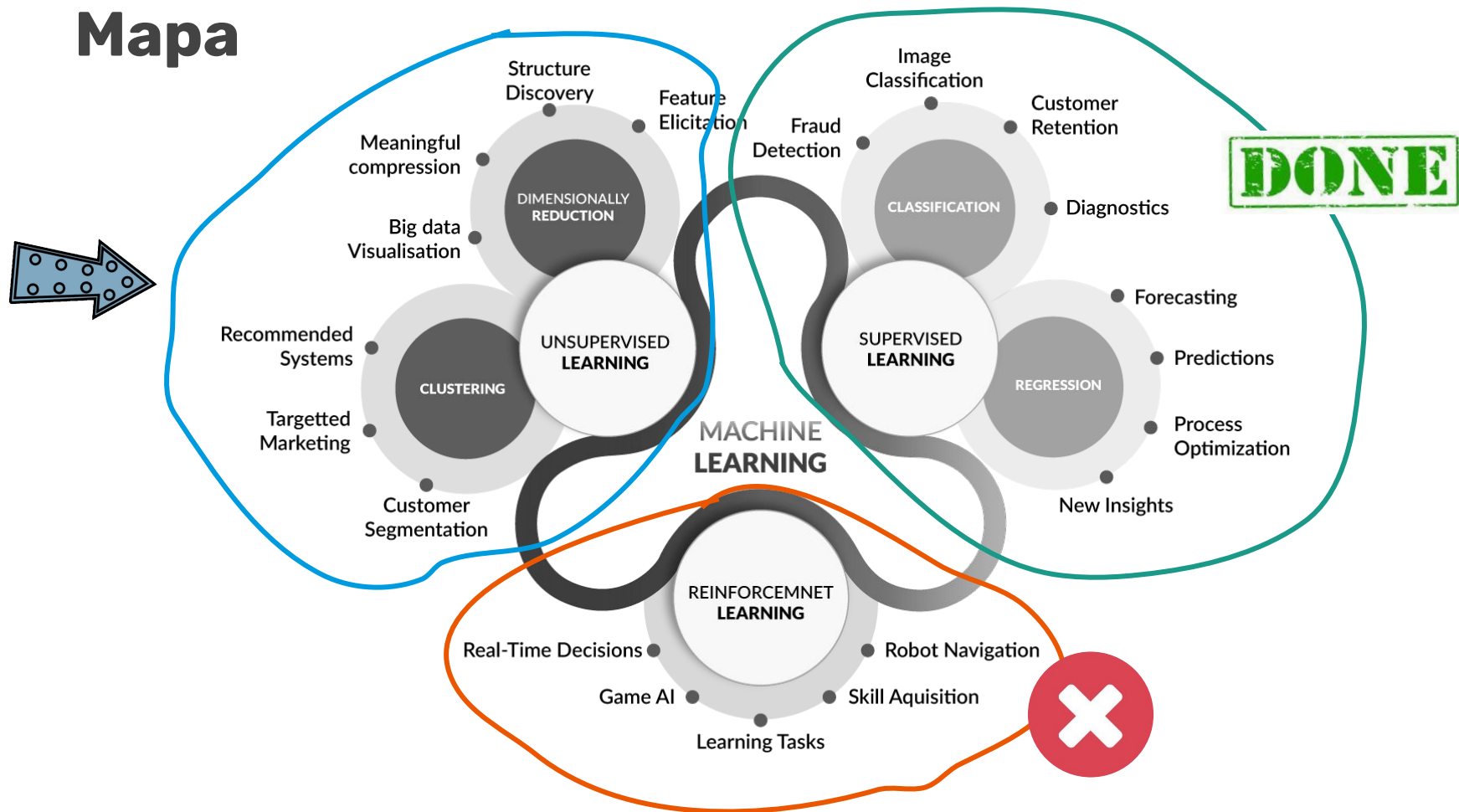
# Mapa



# Mapa



# Mapa



Si venían un poco perdidos con Redes Neuronales,

Si venían un poco perdidos hasta ahora,  
**¡bienvenidos al barco de  
Aprendizaje No Supervisado!**



## 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).



# 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).

A diferencia del **Aprendizaje Supervisado**, el objetivo ya no pasa por predecir la etiqueta, sino por encontrar **patrones en el set de datos.**

# 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).

Las principales herramientas en Aprendizaje No Supervisado son:

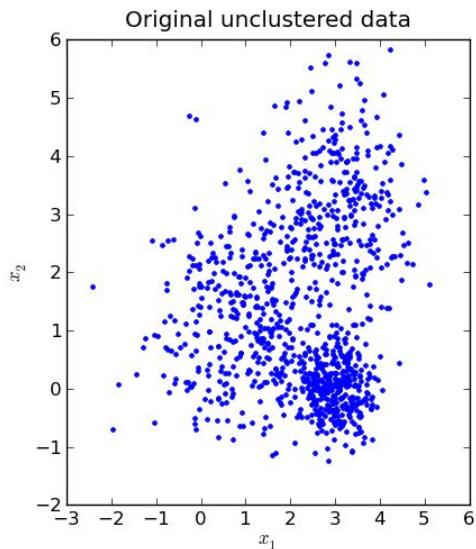
- Clustering
- Reducción de dimensionalidad

- 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”).

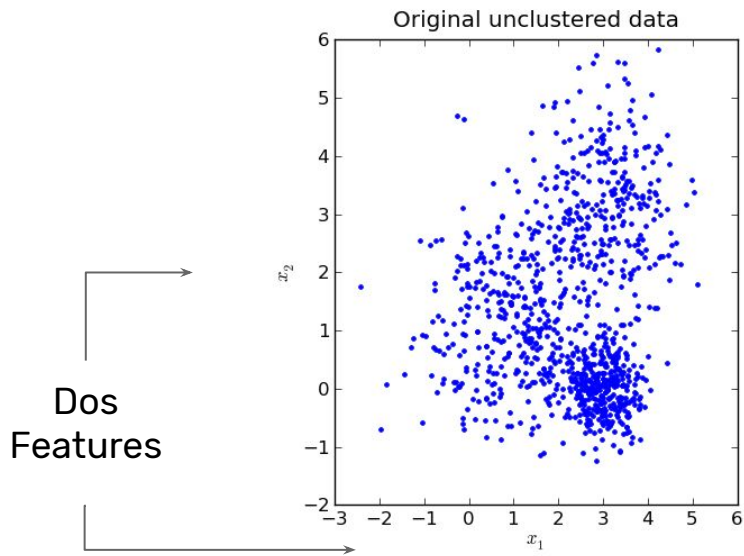
*Datos iniciales*



# Aprendizaje No Supervisado • Clustering

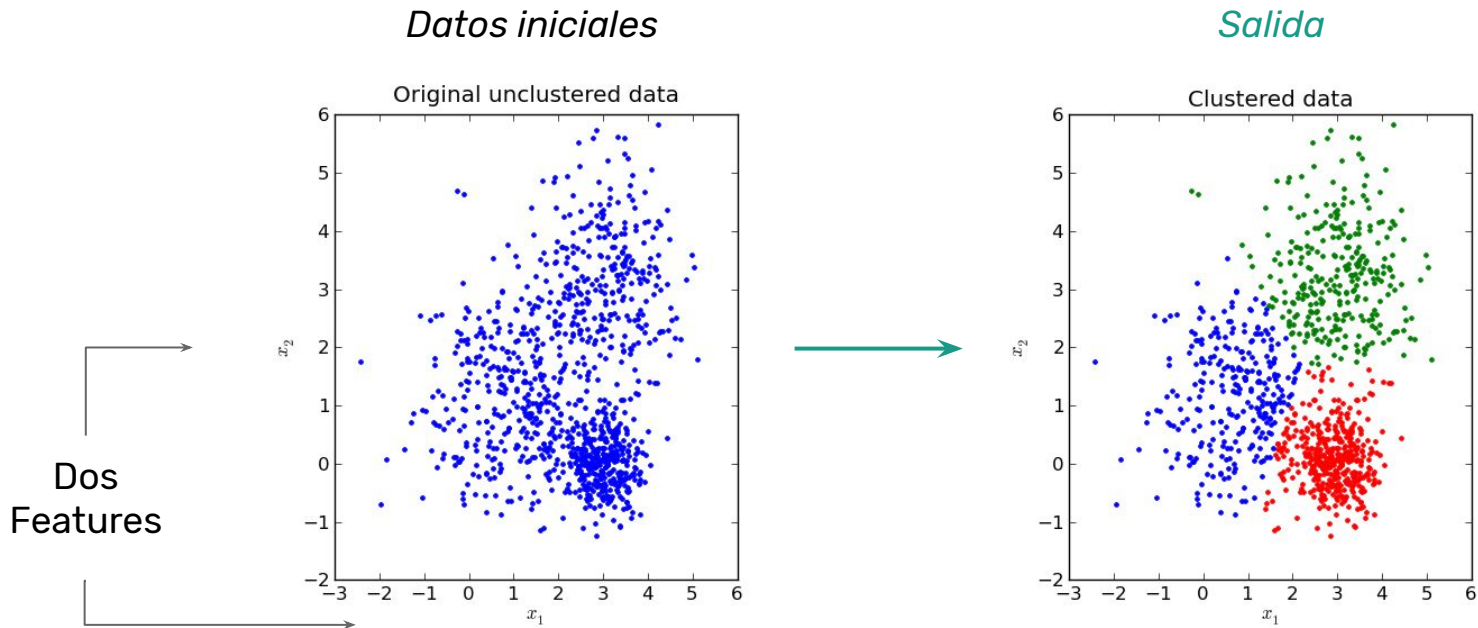
Dado un set de datos, nuestro objetivo será encontrar grupos (clusters) en los cuales las instancias pertenecientes sean parecidas (estén “cerca”).

*Datos iniciales*



# 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

# 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?

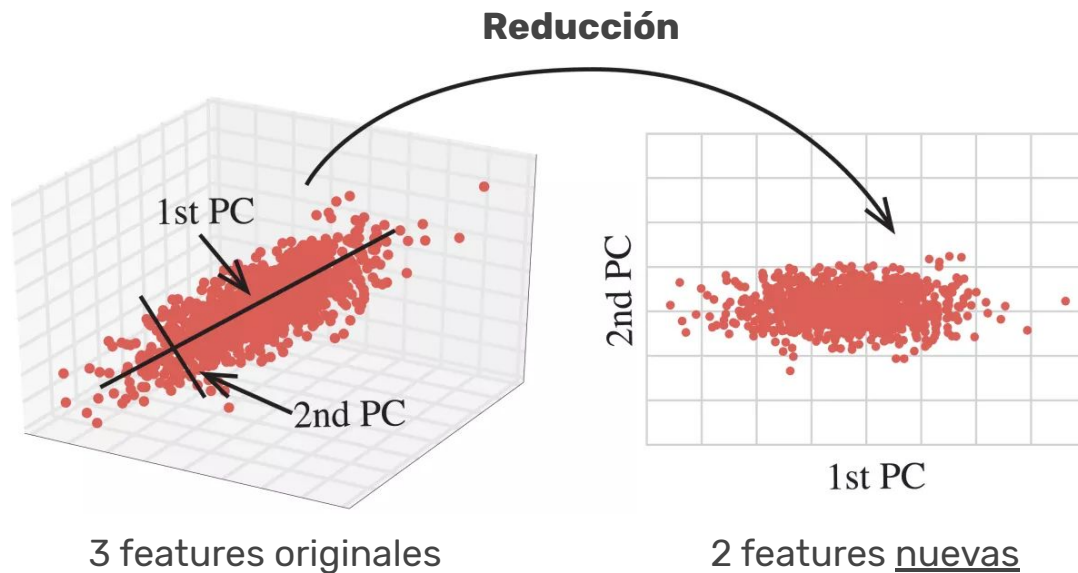
Algunos de los algoritmos para hacer clustering son:

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

- Clustering
- Reducción de dimensionalidad

## Aprendizaje No Supervisado • Reducción de la dimensionalidad

Buscamos reducir la cantidad de features de un dataset, pero reteniendo la mayor cantidad de “información” posible.



# Aprendizaje No Supervisado • Reducción de la dimensionalidad

## ¿Para qué sirve?

Reducir la cantidad de features en un dataset puede servir para:

- Reducir el input en un modelo de regresión o clasificación
- Compresión de archivos
- Visualización
- Detectar features relevantes en datasets
- Muchísimas mas cosas

# Aprendizaje No Supervisado • Reducción de la dimensionalidad

## ¿Para qué sirve?

Reducir la cantidad de features en un dataset puede servir para:

- Reducir el input en un modelo de regresión o clasificación
- Compresión de archivos
- Visualización
- Detectar features relevantes en datasets
- Muchísimas mas cosas

## ¿Cómo se hace?

Algunos de los métodos de reducción de dimensionalidad son:

- PCA: Principal Component Analysis (usa SVD)
- MDS: Multidimensional scaling
- t-SNE: t-distributed Stochastic Neighbor Embedding
- Auto-Encoders (Se hace con Redes Neuronales)
- LDA: Linear Discriminant Analysis (si hay etiquetas de clases)



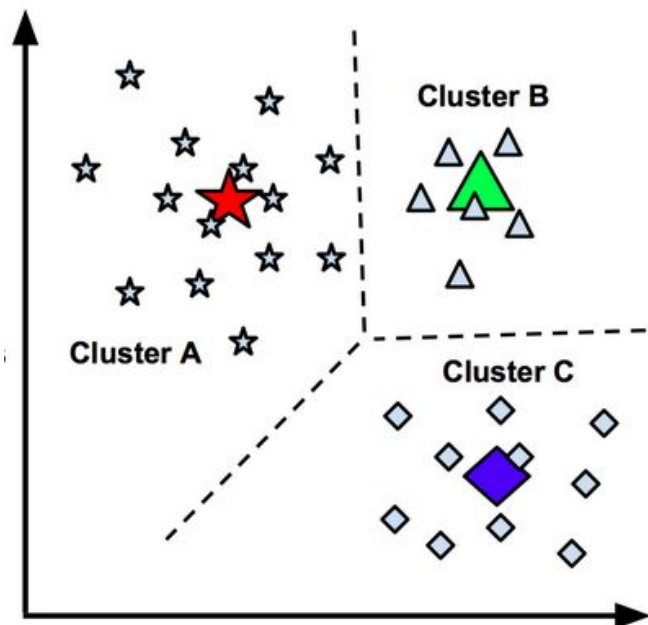
Aprendizaje No Supervisado

# 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)

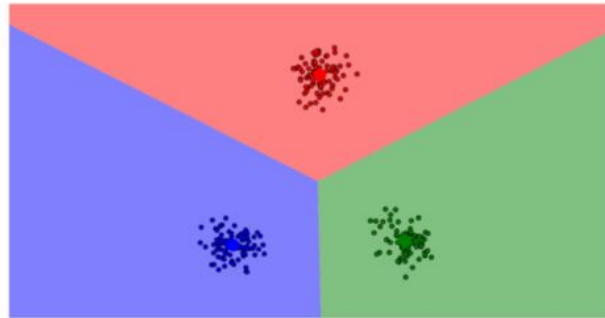
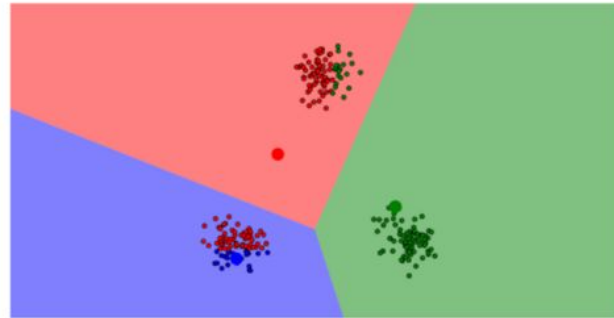
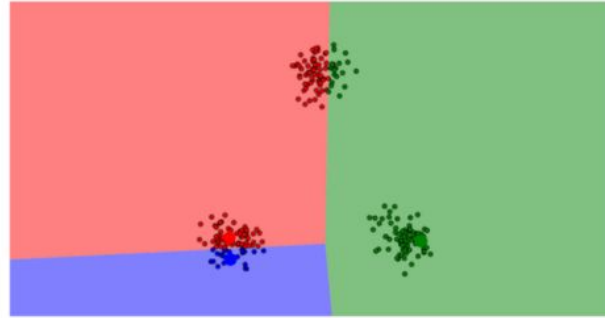
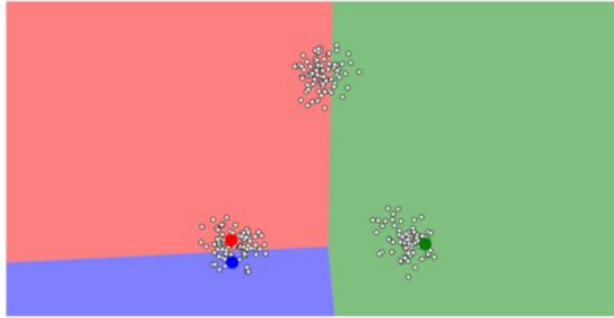
## **3) Actualizar los centroides.**

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

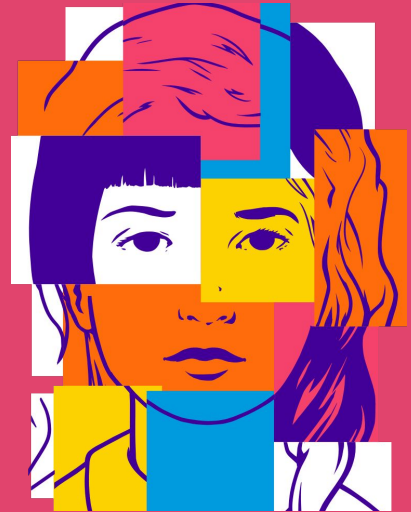


# Clustering • K-Means

## Visualización

Aprendizaje No Supervisado

# Clustering: DBSCAN

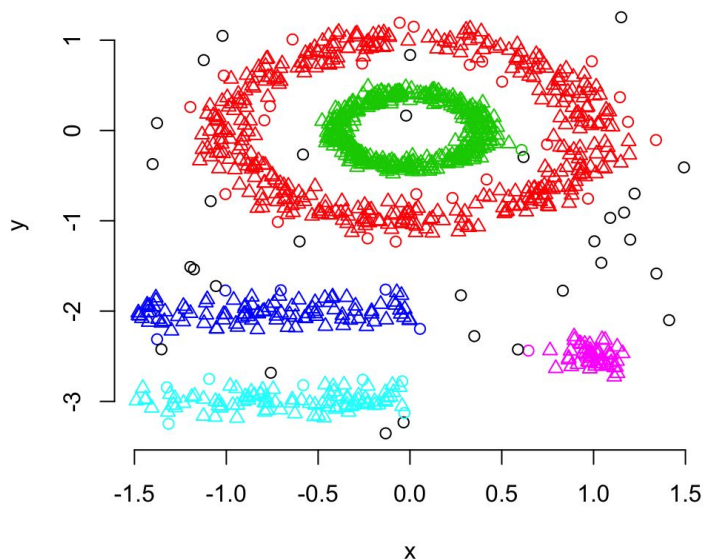


# Clustering • DBSCAN

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

# Clustering • DBSCAN

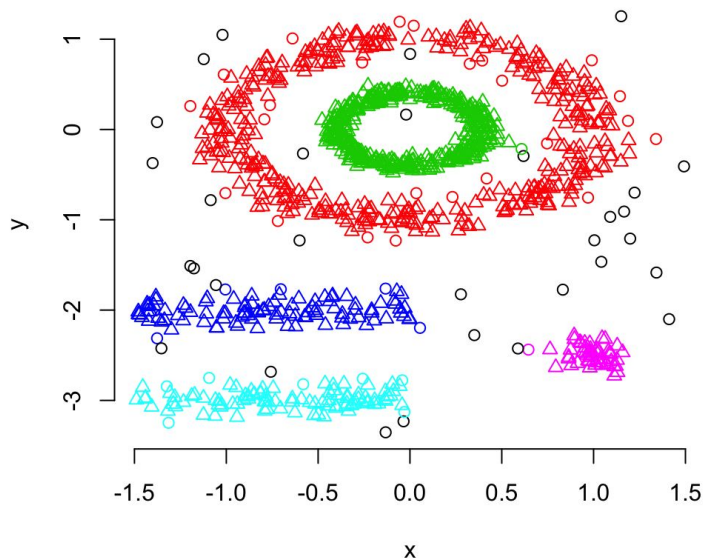
Objetivo: Identificar un número arbitrario de clusters. Los clusters estarán definidos por densidad de puntos. Puede haber puntos que no pertenezcan a ningún cluster (noise).





# 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?

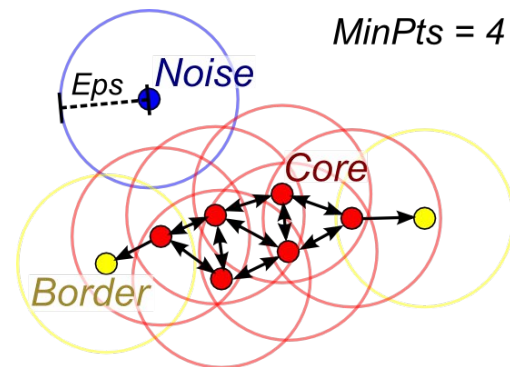
**1) Se define una distancia epsilon (parámetro)** como la vecindad de un punto. Se elige un número de puntos mínimos de para considerar un cluster minPoints (parámetro).

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

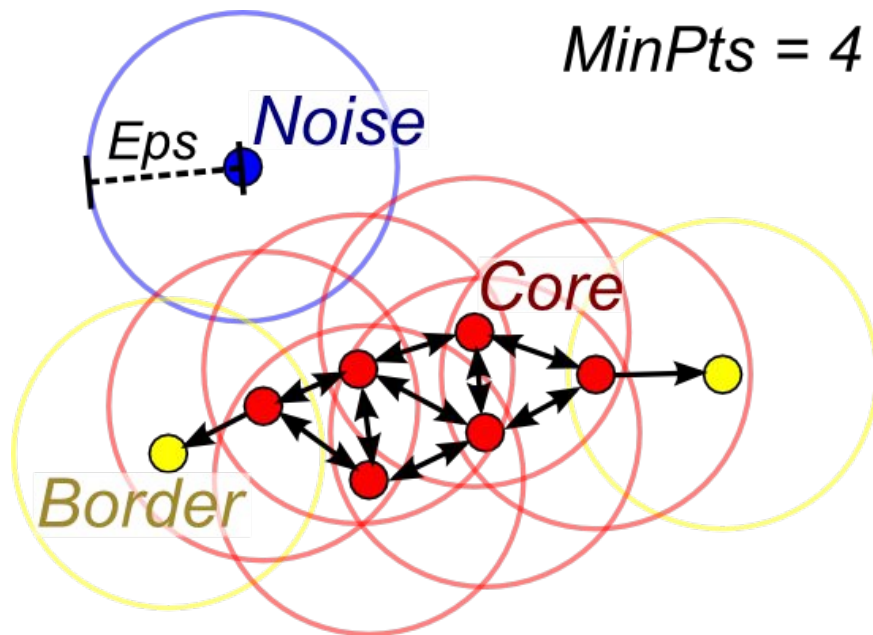
1) Se toma un punto no visitado random. Se identifica si el punto es un 'core', es decir, si tiene minPoints en su vecindario. Si no tiene, se lo llama 'noise'. Este punto se marca como visitado.

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)

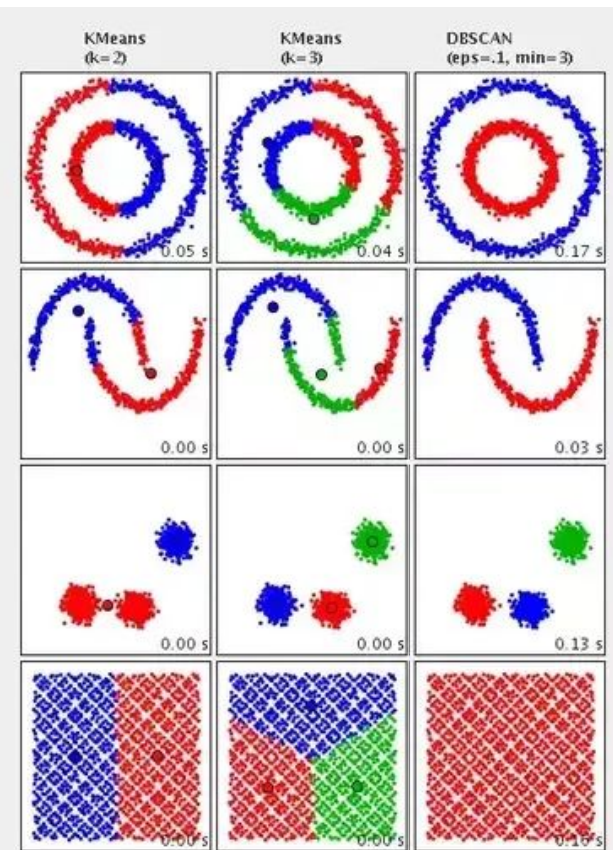
# DBSCAN

- 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.



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!**

---



# Hands-on training





## Hands-on training

DS\_Encuentro\_36\_Clustering.ipynb



# Para la próxima: 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!

# Para la próxima

---

1. Terminar de ver los videos de “Aprendizaje No Supervisado”.
2. Completar los notebooks de hoy y atrasados.
3. Terminar la entrega 05.
4. Preparar el relato “Data Science en mi vida”.

ACÀMICA