



Análisis de Datos y Big Data

Sesión 5 : Análisis exploratorio de datos para Big Data

Presentan:

Dr. Ulises Olivares Pinto

Joshelyn Yanori Mendoza Alfaro

Fernando Ramírez González



BLOQUE
Innovación, Tecnología
y Creatividad.

Contenido



1. Aplicación y ejercicios de análisis exploratorio de datos



2. Reducción de dimensionalidad



3. Procesamiento con Hadoop

1. Aplicación de análisis exploratorio de datos

Casos de aplicación
Titanic

THE DATA SCIENCE PROCESS



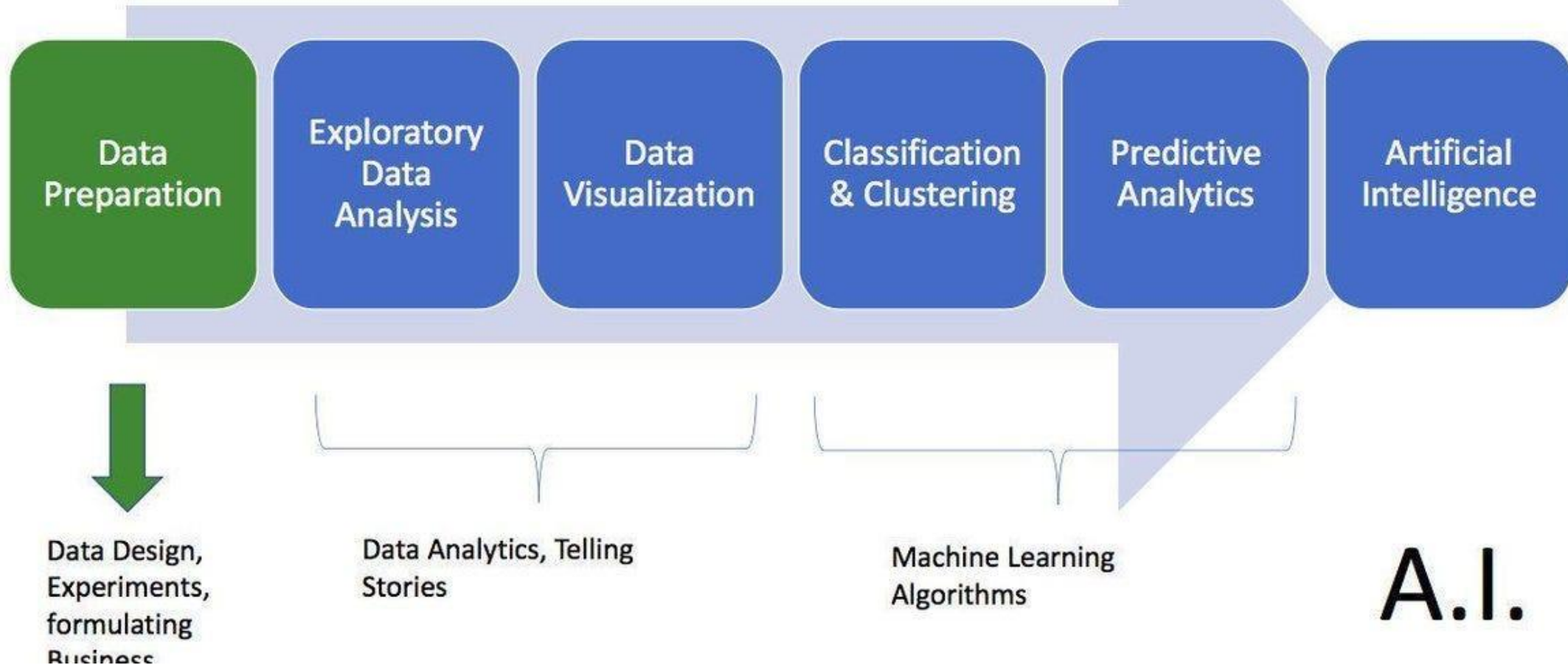
Data Engineers

Data Analysts

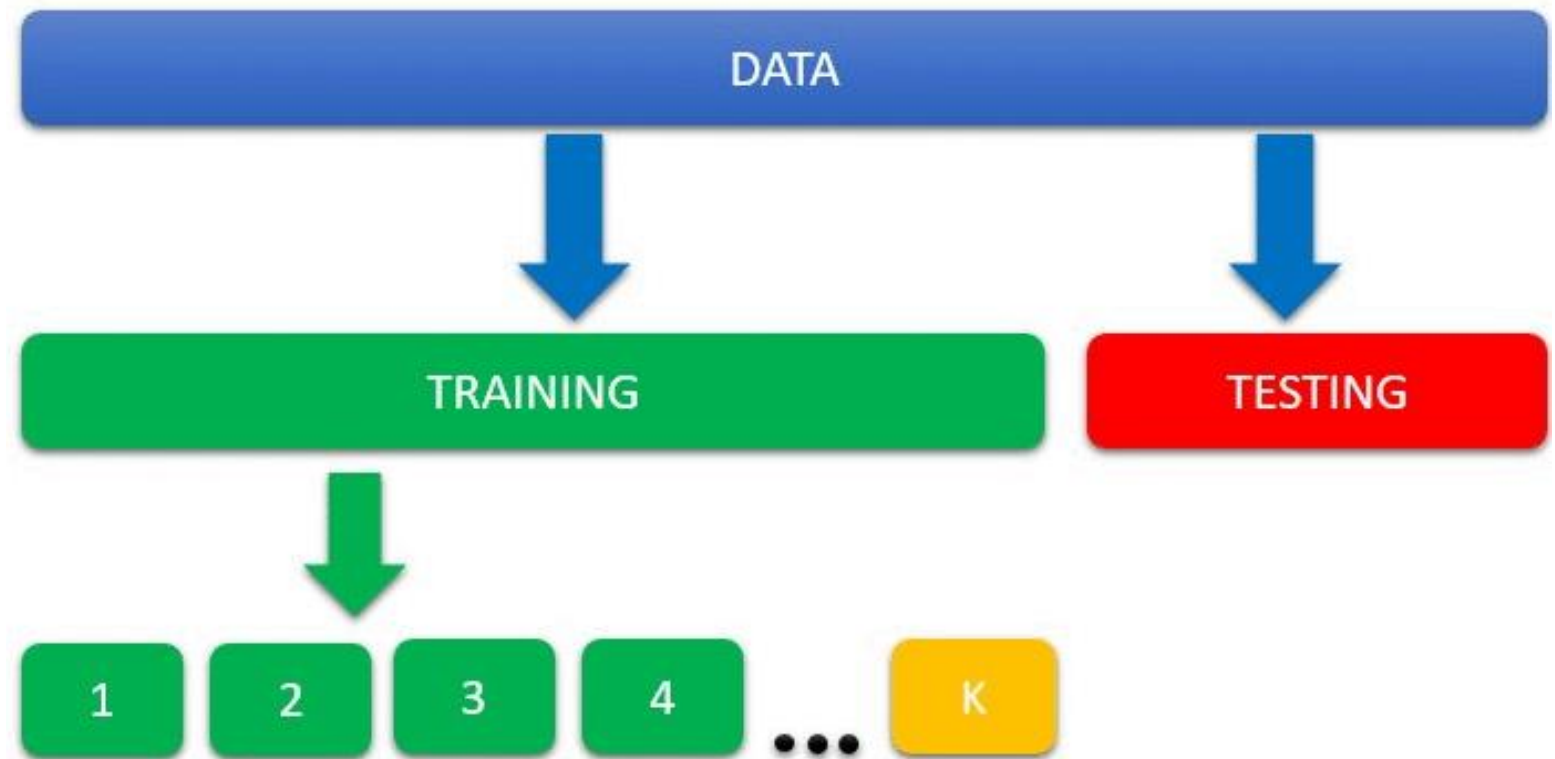
Machine Learning Engineers

Data Scientists

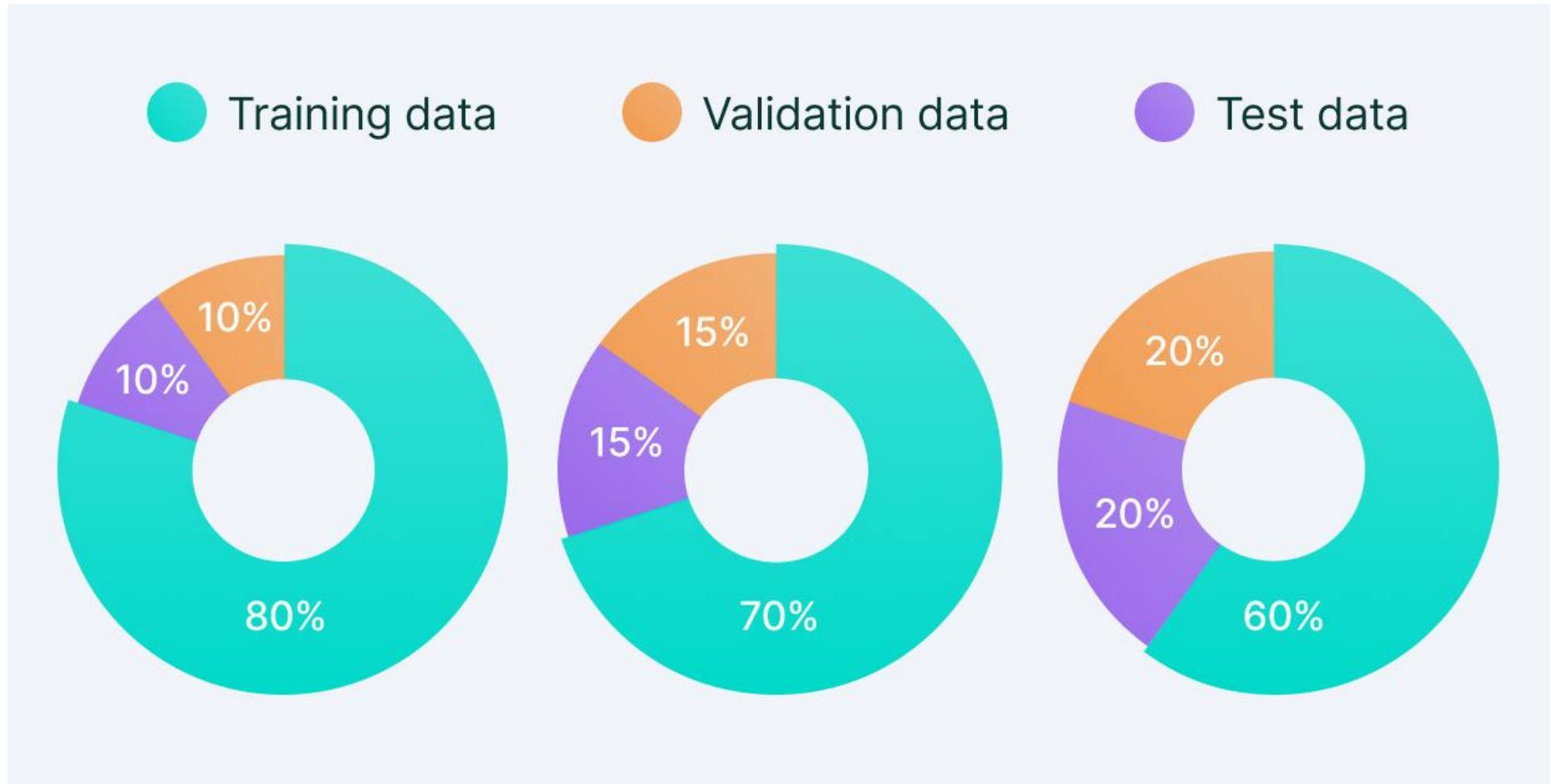
Data Analytics Process



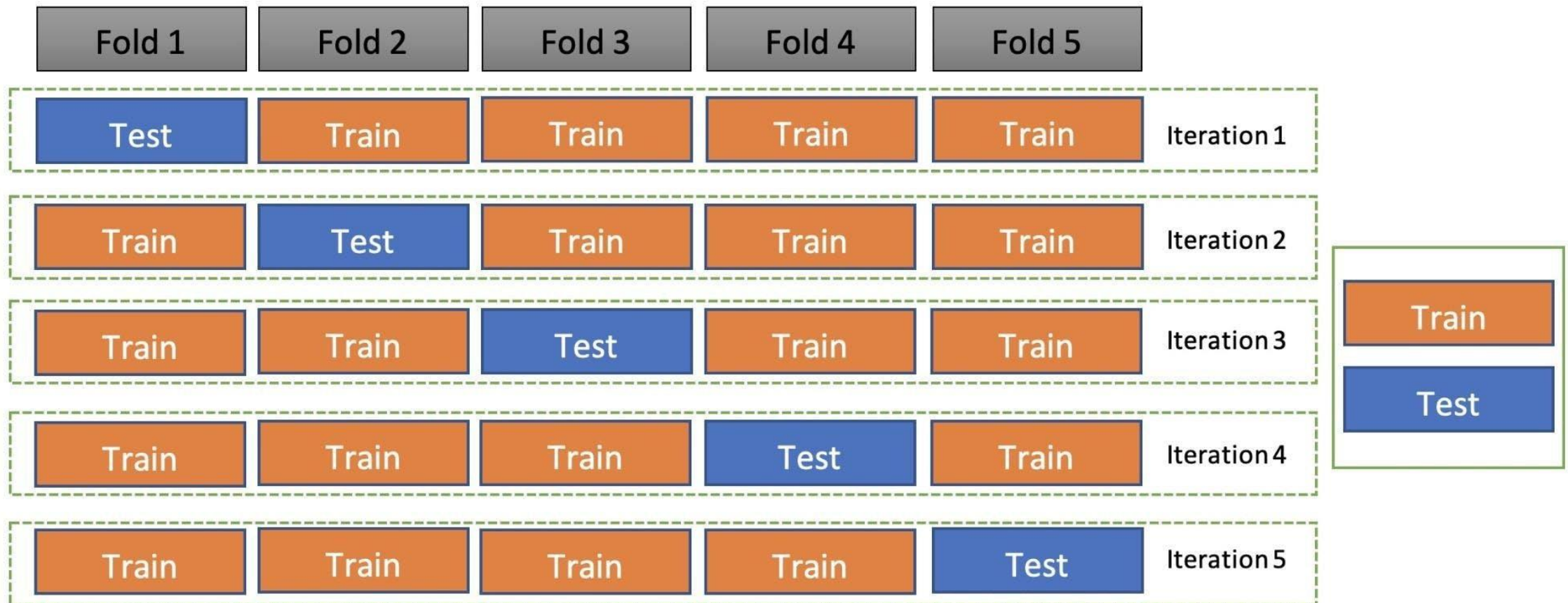
División de datos



¿Cómo dividir mis datos?

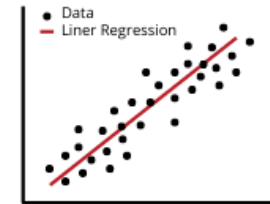


División e Iteraciones

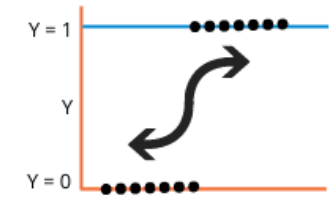


Algoritmos de aprendizaje clásicos

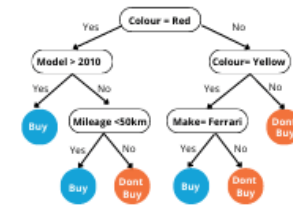
Linear Regression



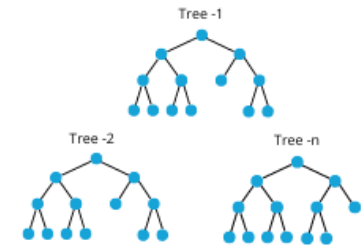
Logistic Regression



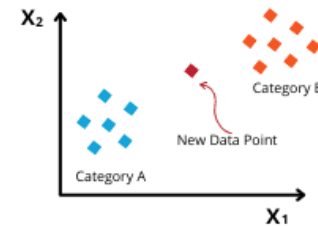
Decision Trees



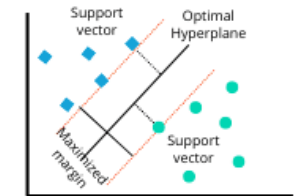
Random Forest



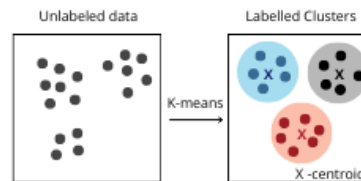
K-Nearest Neighbor



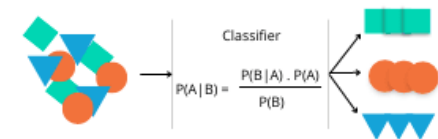
Support Vector Machine



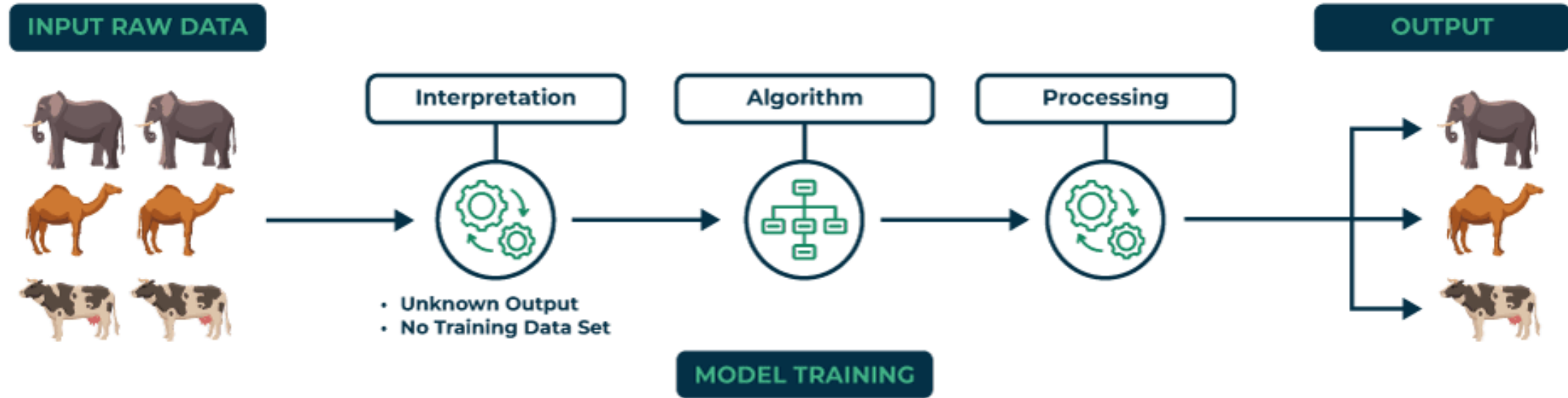
K-Means Clustering

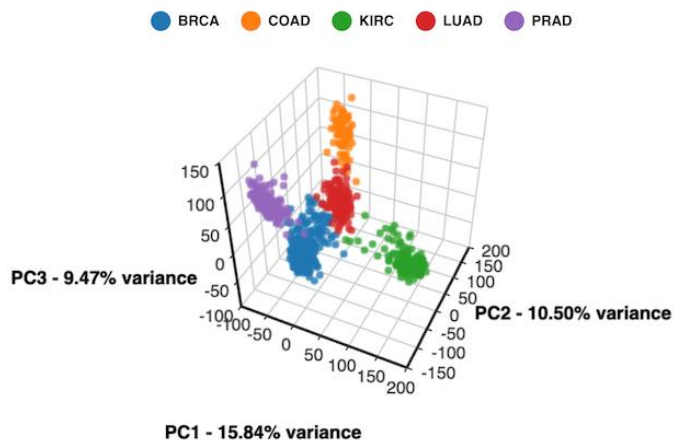


Naïve Bayes



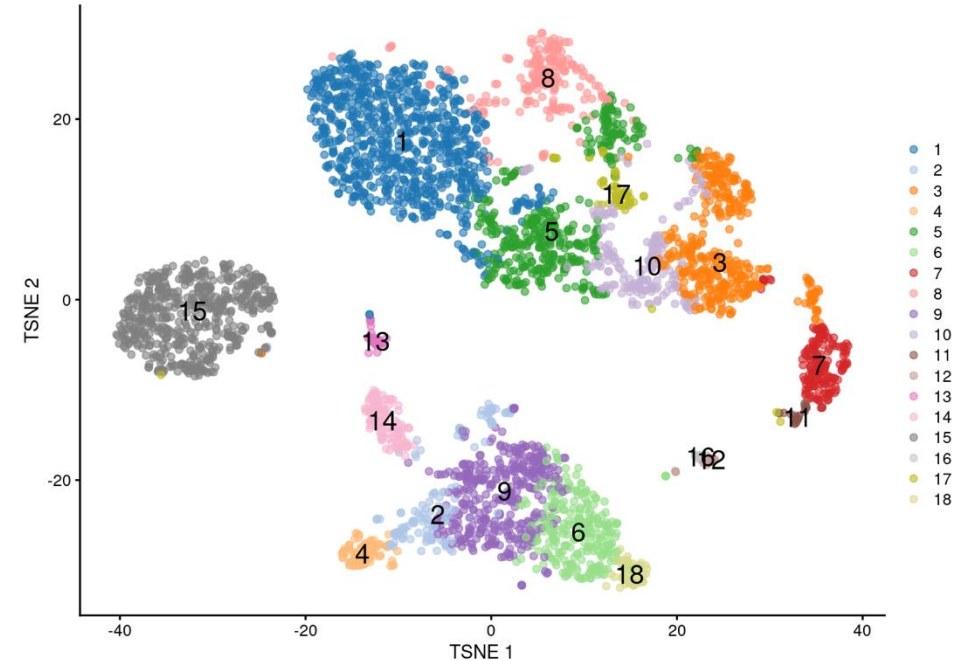
Unsupervised Learning



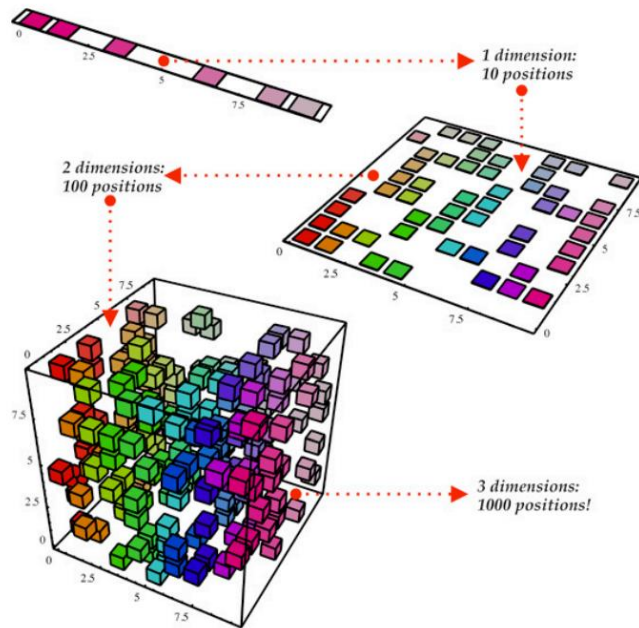


2. Algoritmos de Reducción de Dimensiones

Conceptos, técnicas y aplicaciones prácticas

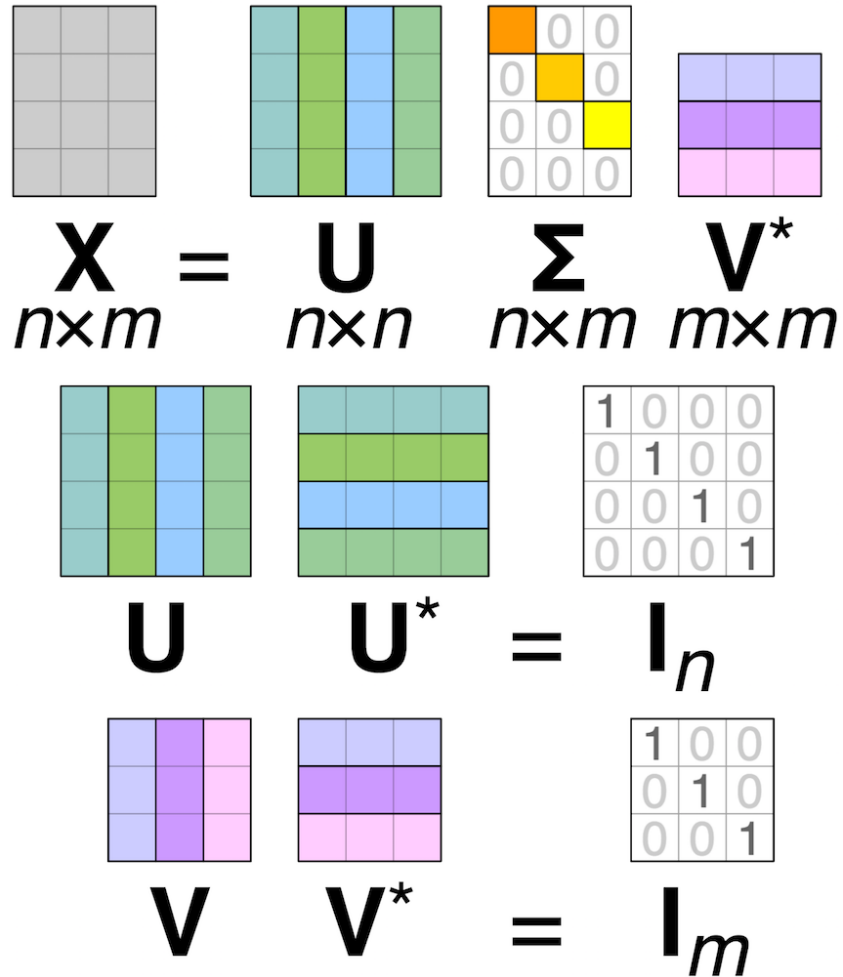


¿Qué es la Disminución de Dimensiones?



Definición e Importancia

- Definición: Proceso de reducir el número de variables aleatorias bajo consideración, obteniendo un conjunto de variables principales.
- Importancia: Facilita la visualización, reduce el costo computacional y mejora la eficiencia de los algoritmos de aprendizaje automático.
- Aplicaciones: Desde la visualización de datos hasta el preprocesamiento en modelos de machine learning.



The diagram illustrates the SVD decomposition of matrix X into matrices U , Σ , and V^* . The dimensions are given as $n \times m$ for X , $n \times n$ for U , $n \times m$ for Σ , and $m \times m$ for V^* .

Matrix X is represented by a 4x4 grid of gray squares.

Matrix U is represented by a 4x4 grid of colored squares (green, blue, green, green).

Matrix Σ is represented by a 4x4 grid of colored squares (orange, yellow, yellow, yellow).

Matrix V^* is represented by a 4x4 grid of colored squares (purple, purple, purple, purple).

The equation is shown as:

$$X = U \Sigma V^*$$

Below this, the orthogonal matrices U and V are shown, along with their products with their transposes, resulting in identity matrices I_n and I_m .

Matrix U is represented by a 4x4 grid of colored squares (green, blue, green, green).

Matrix U^* is represented by a 4x4 grid of colored squares (green, green, blue, green).

Matrix V is represented by a 4x4 grid of colored squares (purple, purple, purple, purple).

Matrix V^* is represented by a 4x4 grid of colored squares (purple, purple, purple, purple).

The equations are shown as:

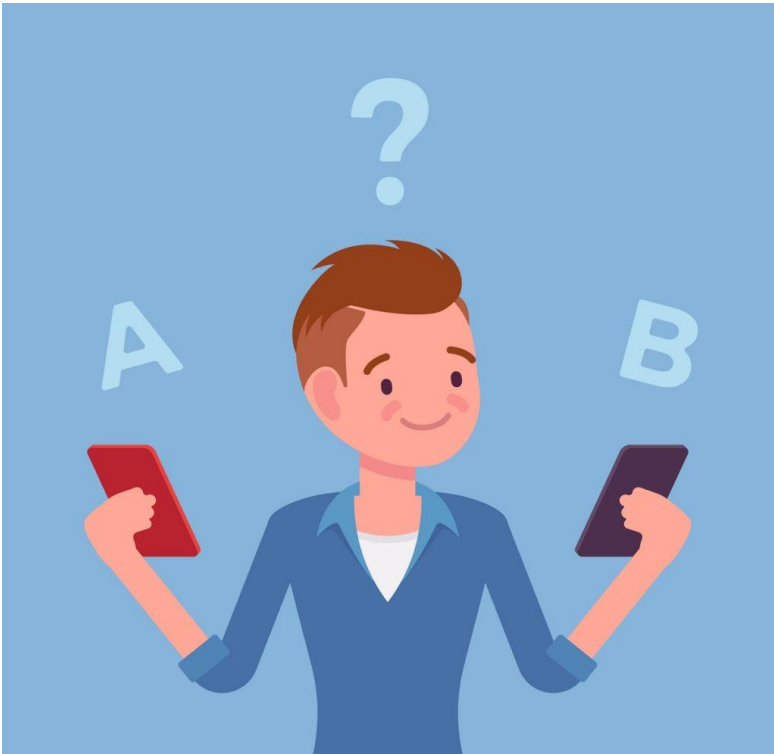
$$U U^* = I_n$$

$$V V^* = I_m$$

Principales Algoritmos de Disminución de Dimensiones

PCA, SVD Truncado, Proyección Aleatoria Aleatoria

- PCA: Transforma variables originales en nuevas variables no variables no correlacionadas, maximizando la varianza retenida retenida en cada componente.
- SVD Truncado: Similar a PCA, aplicado a matrices dispersas, dispersas, descomponiendo una matriz en tres matrices más matrices más pequeñas.
- Proyección Aleatoria: Técnica no lineal que proyecta datos en datos en un espacio de menor dimensión usando matrices matrices aleatorias.
- Comparativa: Precisión, escalabilidad y facilidad de interpretación entre los tres algoritmos.



Comparativa entre Algoritmos de Disminución de Dimensiones



PCA vs SVD vs Proyección Aleatoria

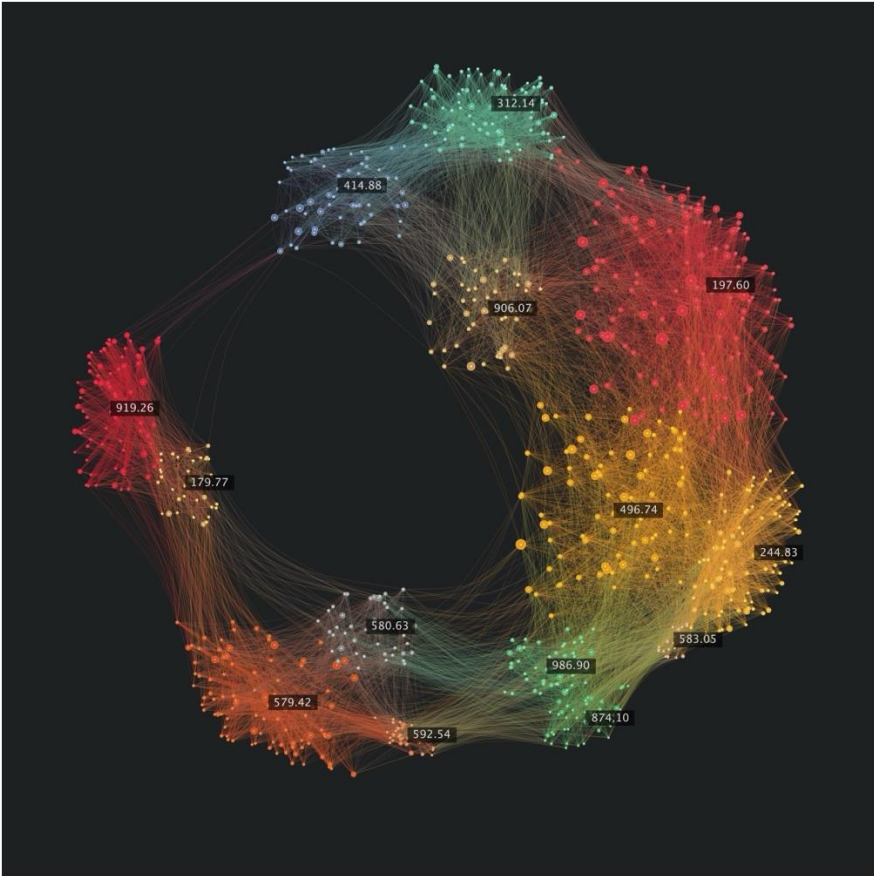
- Precisión: $\text{PCA} > \text{SVD} > \text{Proyección Aleatoria}$.
- Escalabilidad: $\text{Proyección Aleatoria} > \text{SVD} > \text{PCA}$.
- Facilidad de Interpretación: $\text{PCA} > \text{SVD} > \text{Proyección Aleatoria}$.

¿Qué es el Clustering?



Definición e Importancia

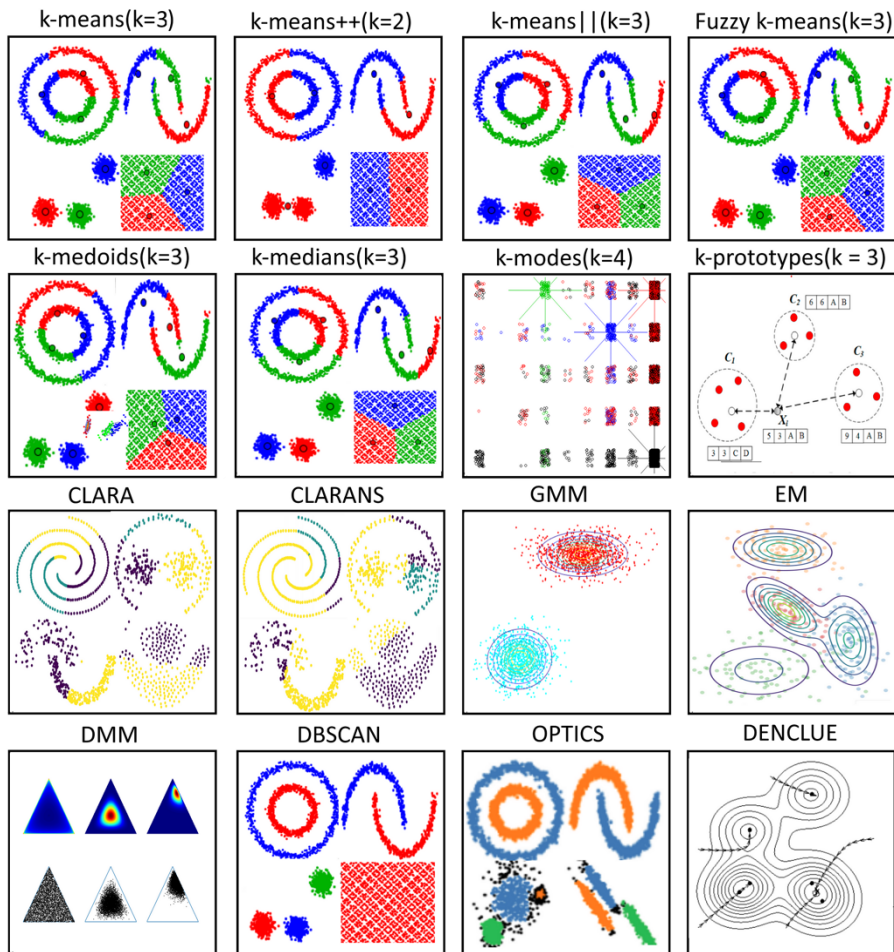
- Definición: Técnica de aprendizaje no supervisado que agrupa datos en clusters, donde los objetos en el mismo cluster son más similares entre sí que a los de otros clusters.
- Importancia: Útil para el análisis exploratorio, segmentación de clientes, detección de anomalías.
- Aplicaciones: Desde la segmentación de clientes hasta la detección de anomalías en datos de sensores.
-

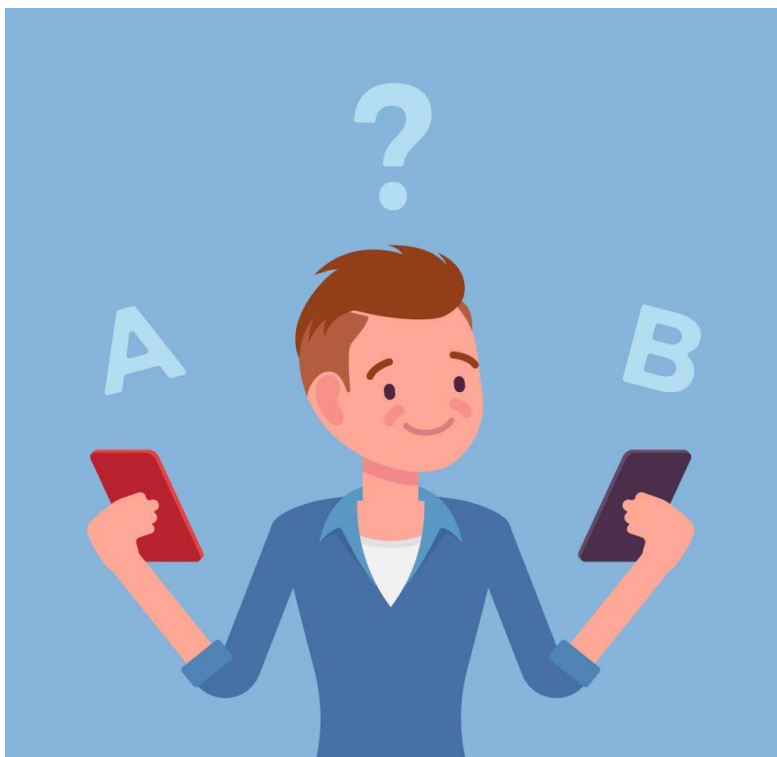


Principales Algoritmos de Clustering

K-Means, Clustering Jerárquico, DBSCAN

- K-Means: Particiona los datos en k clusters, minimizando la suma de las distancias al centroide.
- Clustering Jerárquico: Construye una jerarquía de clusters utilizando una matriz de distancias, con enfoque aglomerativo (bottom-up) o divisivo (top-down).
- DBSCAN: Agrupa puntos que están densamente conectados, identificando outliers como ruido.
- Comparativa: Ventajas y limitaciones en diferentes contextos de aplicación.





Comparativa entre Algoritmos de Clustering

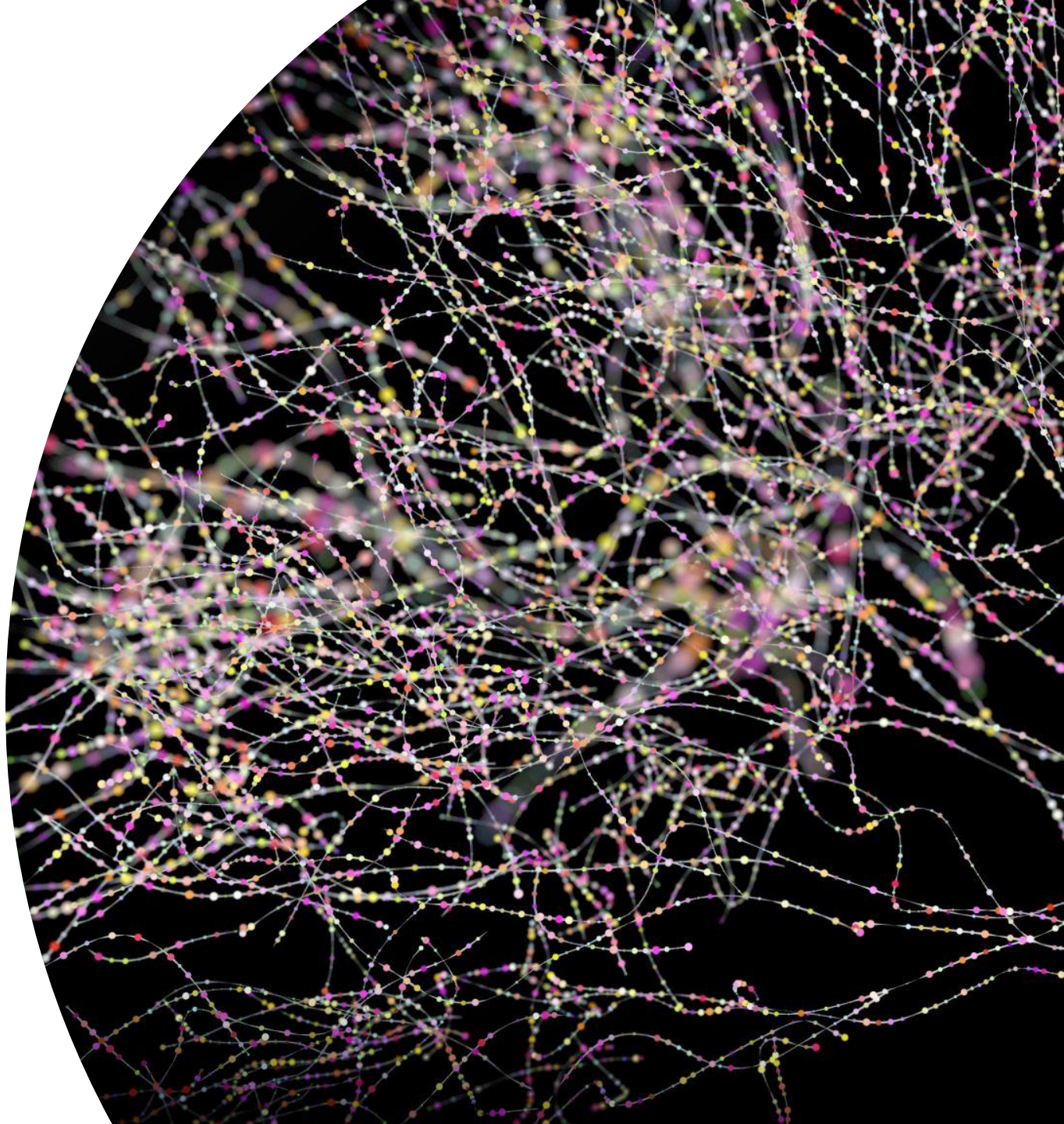


K-Means vs Clustering Jerárquico vs DBSCAN

- Escalabilidad: K-Means > DBSCAN > Clustering Jerárquico.
- Detección de Outliers: DBSCAN > K-Means, Clustering Jerárquico.
- Flexibilidad en la Forma de los Clusters: DBSCAN > Clustering Jerárquico > K-Means.
- Ejemplo: Aplicación de cada algoritmo en diferentes escenarios de datasets.

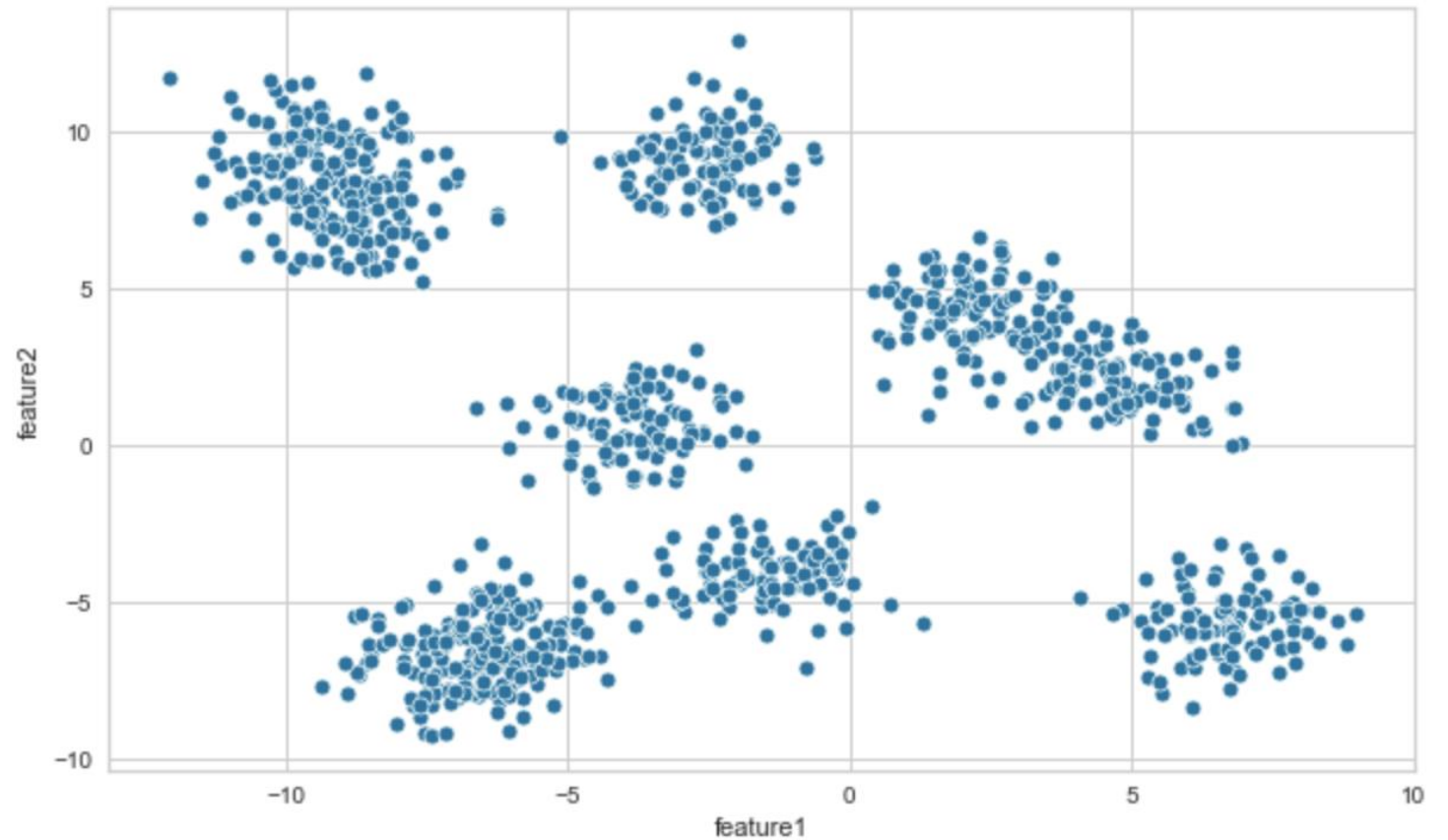
Algoritmos de agrupamiento

Kmeans



Encontrar los k clústeres
que mejor describan a
los datos

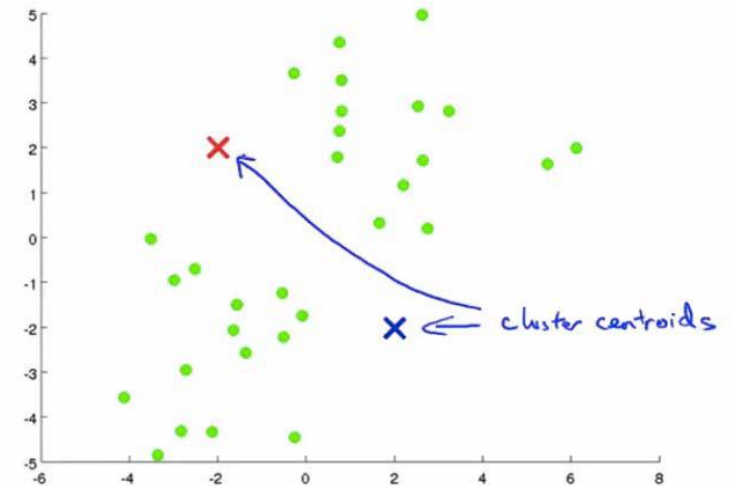
Algoritmo K- means



Algoritmo K-means

Número de clusters $k = 2$

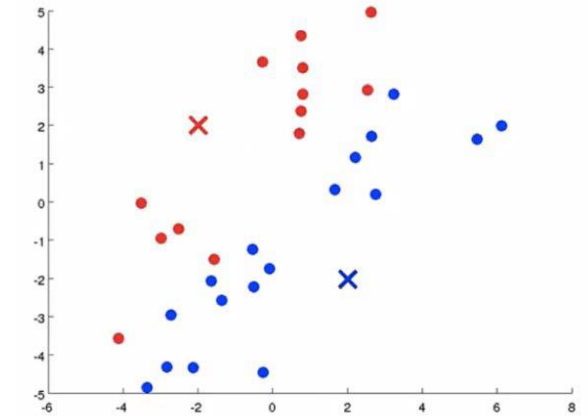
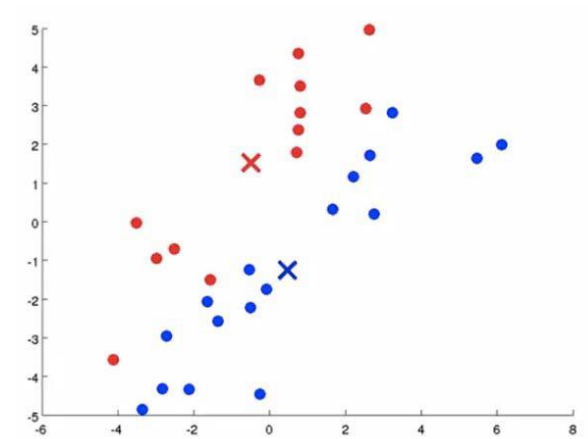
- Se inicializan los centroides de forma aleatoria



Algoritmo K-means

Número de clusters $k = 2$

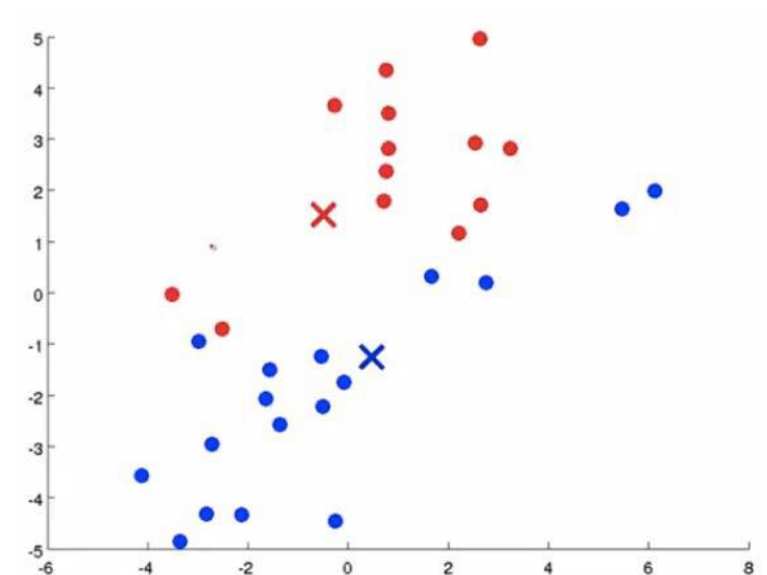
- Asignar mebresia para cada clúster
- Actualizar el centroide de cada cluster (promedio de los puntos para cada cluster)



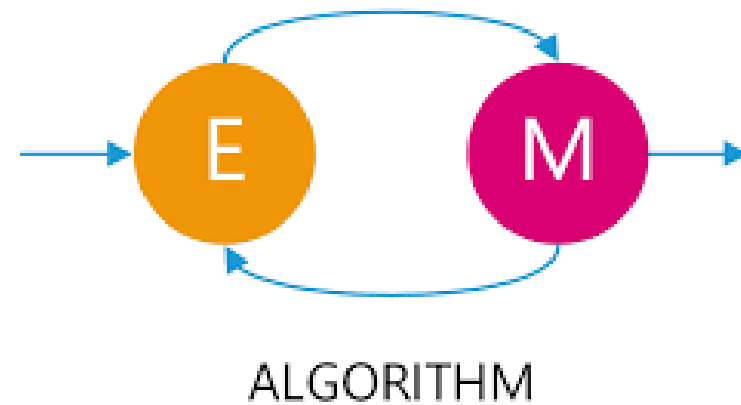
Algoritmo K-means

Número de clústeres $k = 2$

- Actualizar la membresía del cluster
- Actualizar los valores hasta que no existan cambios en la membresía

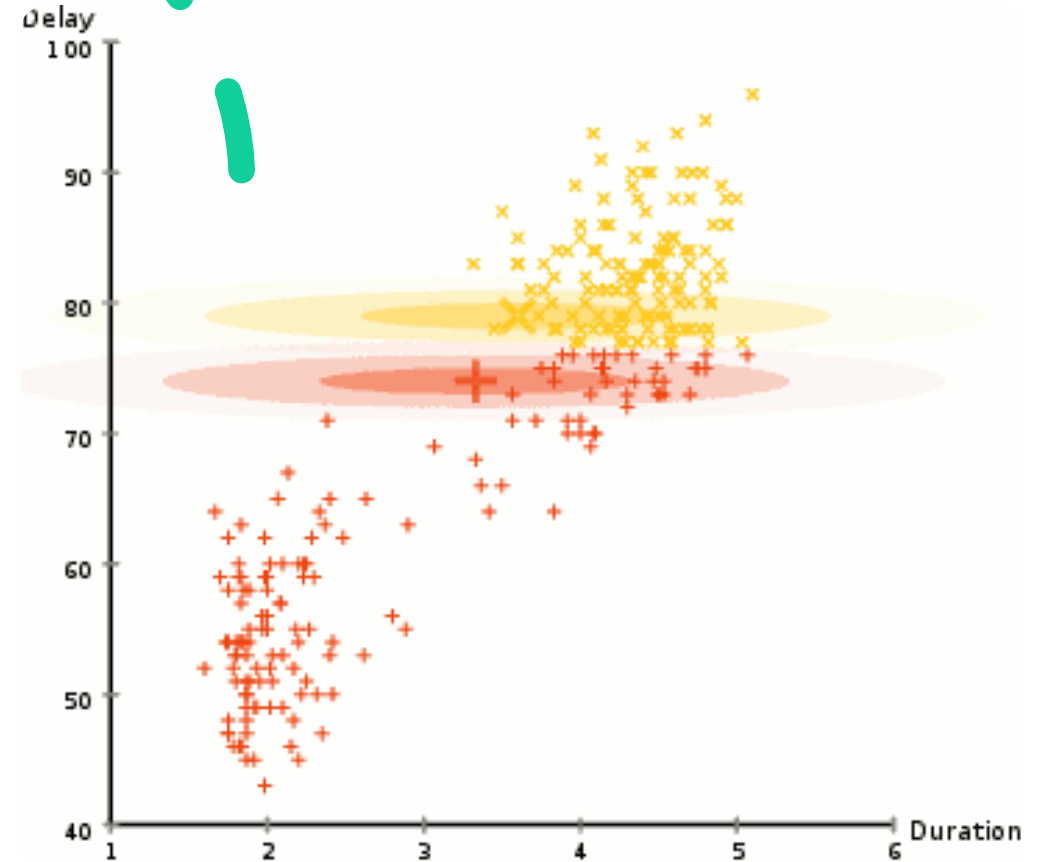


Expectation Maximization



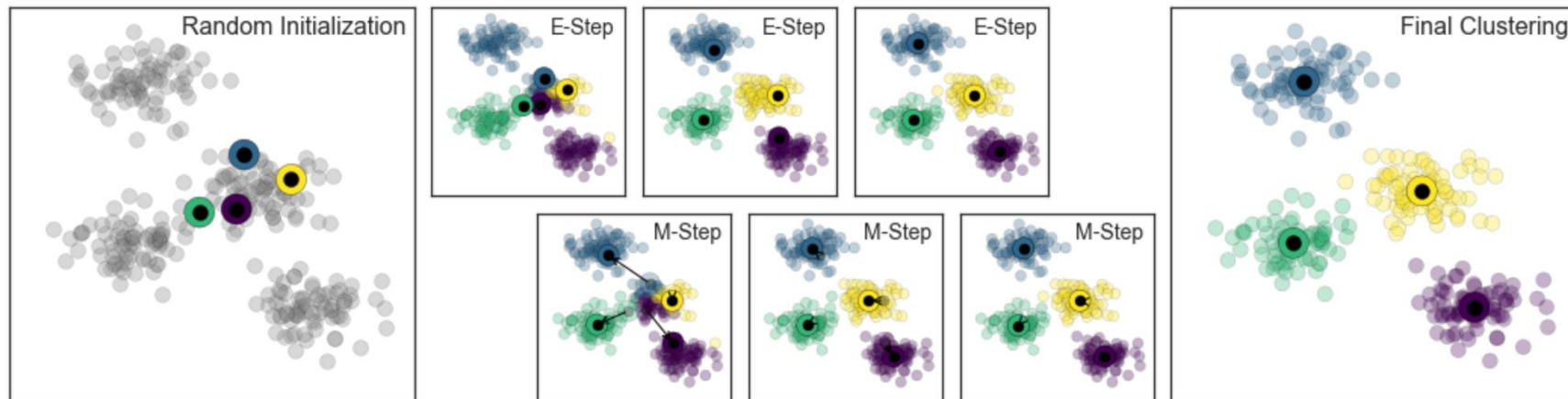
Expectation Maximization (EM)

- Es un algoritmo poderoso que se puede emplear en múltiples contextos.
1. Designa puntos iniciales de centroides (aleatorios).
 2. Este algoritmo consiste de dos partes
 - **Paso E:** Asignar puntos al centroide más cercano.
 - **Paso M:** Establecer los centroides usando promedios.

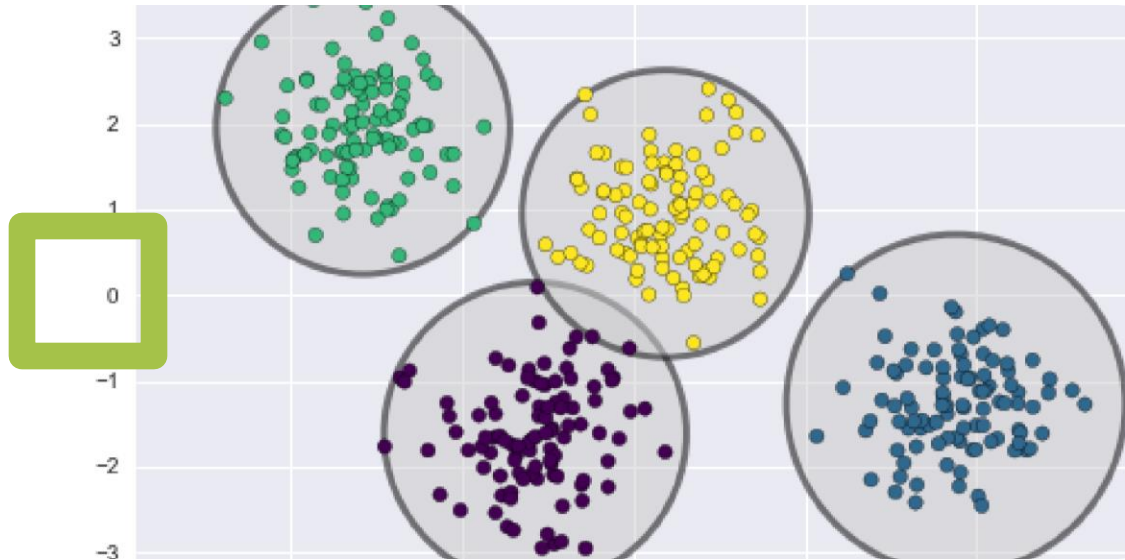
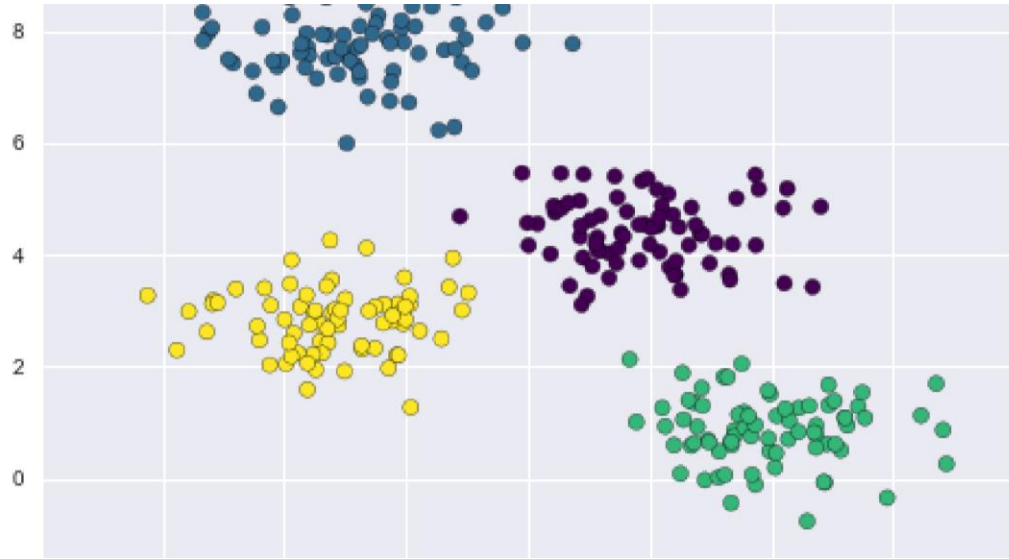


Expectation Maximization (EM)

- En el paso E (Expectation), consiste en actualizar la pertenencia de un punto a un centroide.
- El paso M (Maximization) consiste en maximizar una función fitness, la cual define la ubicación de los centroides.

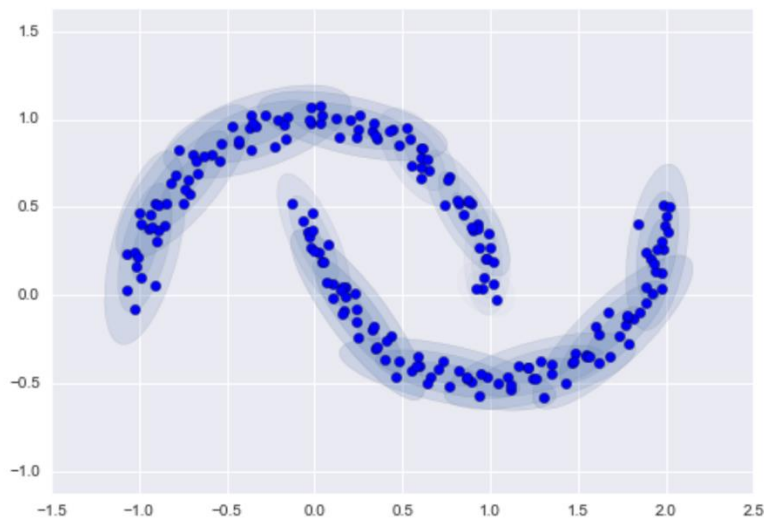
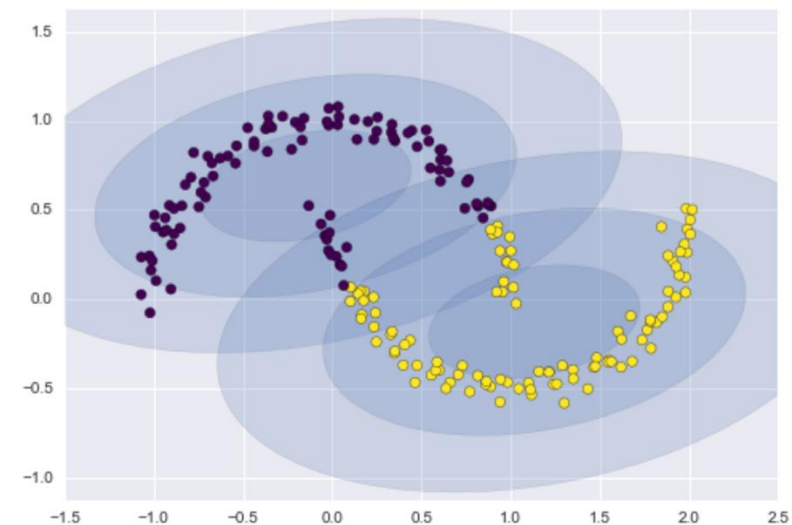
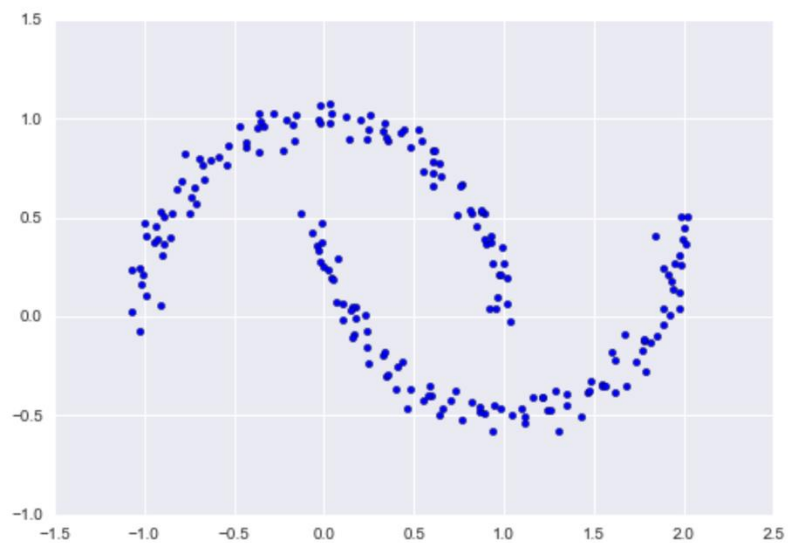


Expectation Maximization (EM)



- Proceso de clusterizado

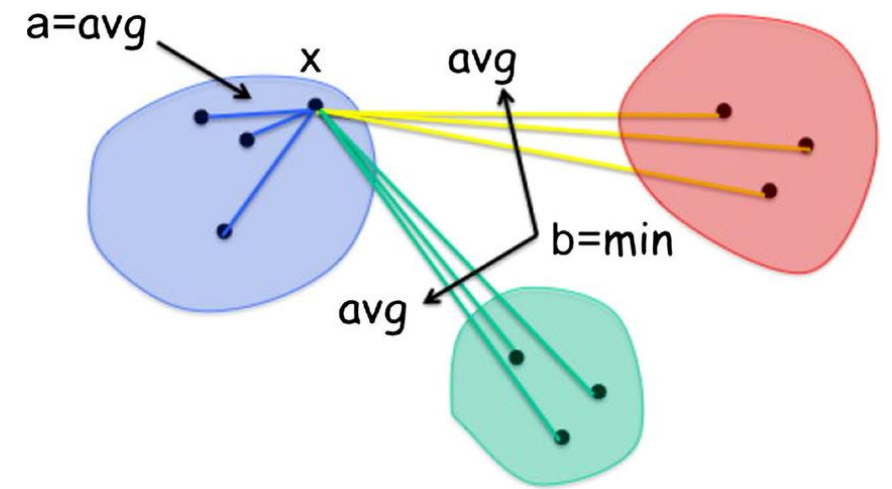
Expectation Maximization (EM)



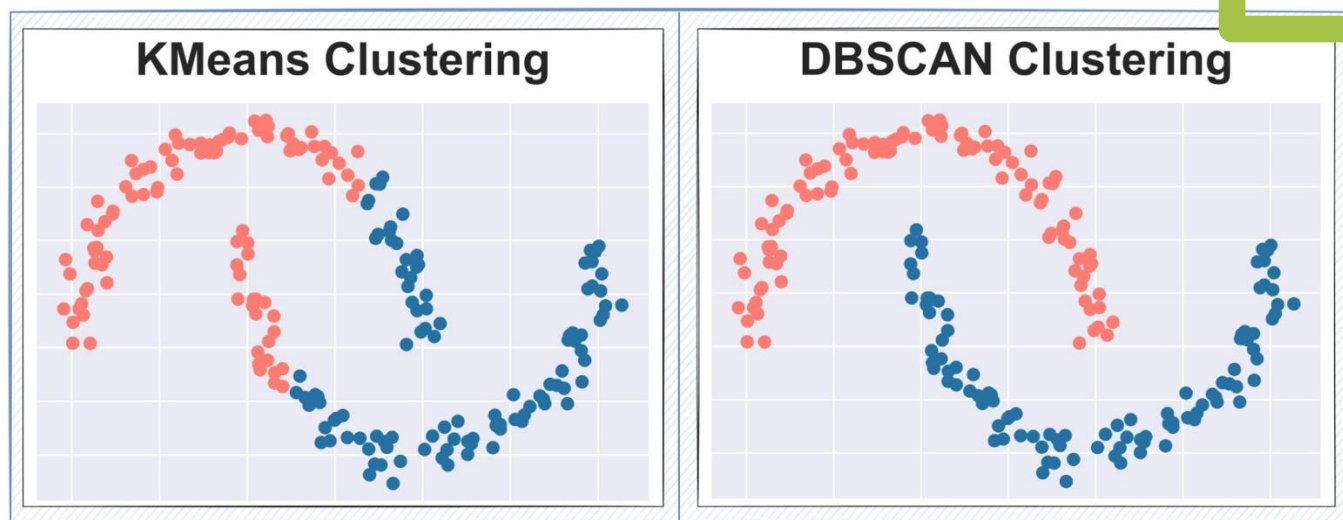
Métricas de evaluación

- El **Silhouette Score** es una métrica utilizada para evaluar la calidad de los clusters creados por un algoritmo de clustering. Esta métrica toma en cuenta tanto la cohesión dentro de los clusters como la separación entre los clusters.

$$s = \frac{b - a}{\max(a, b)}$$



¿Cómo evaluar algoritmos de clustering?



✗ Silhouette score

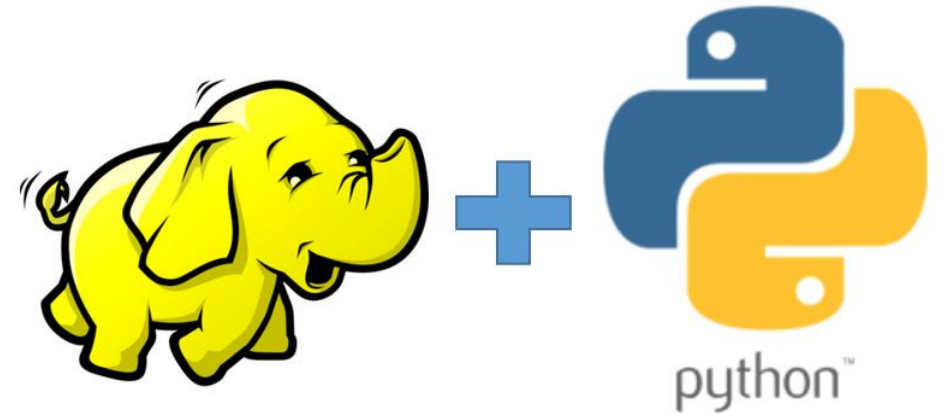
0.49 (Worse clustering but better score)	0.31 (Better clustering but worse score)
---	---

✓ DBCV score
(Density-based clustering validation)

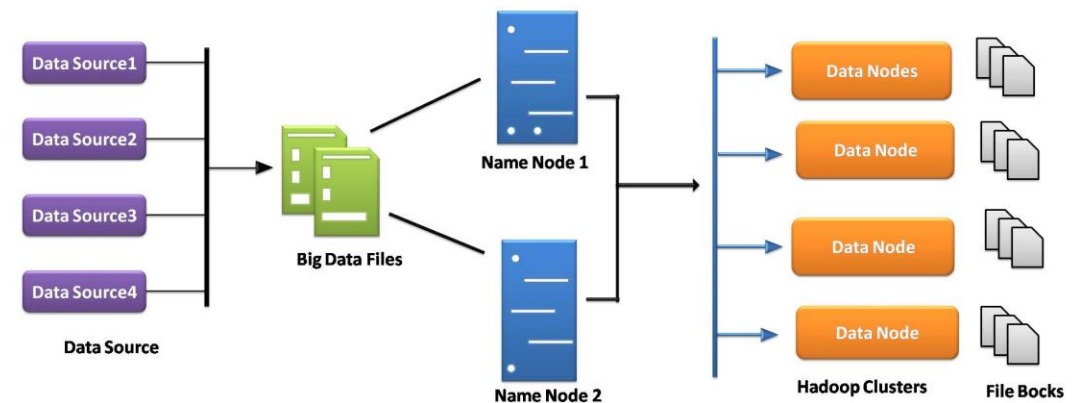
-0.63 (Worse clustering and worse score)	0.45 (Better clustering and better score)
---	--

3. Hadoop y Python

Aplicaciones



Hadoop System Architecture



HDFS

Apache Hadoop 2.0 and YARN

