HENRY

Data Science

M4L7 | Aprendizaje no supervisado I

→ soyhenry.com

Objetivos

- Aplicar técnicas de clustering como K-Means y DBSCAN para segmentar datos y descubrir patrones dentro de conjuntos de datos no etiquetados.
- Evaluar la calidad de los clusters mediante el método del codo y el índice de silueta, determinando la estructura óptima de agrupación.
- Explorar métodos de reducción de dimensionalidad como PCA y t-SNE, analizando la varianza explicada y su impacto en la simplificación de datos

#TEMAS

Agenda

COMENCEMOS →

.01 Definición de clustering

.02 Familias de clustering particional (e.g. k-means)

.03 Familias de clustering por densidad (e.g. DBSCAN)

.04 Evaluación de algoritmos no supervisados

.05 Reducción de la dimensionalidad

.06 Homework

<-->

¿Qué vimos en la lecture?

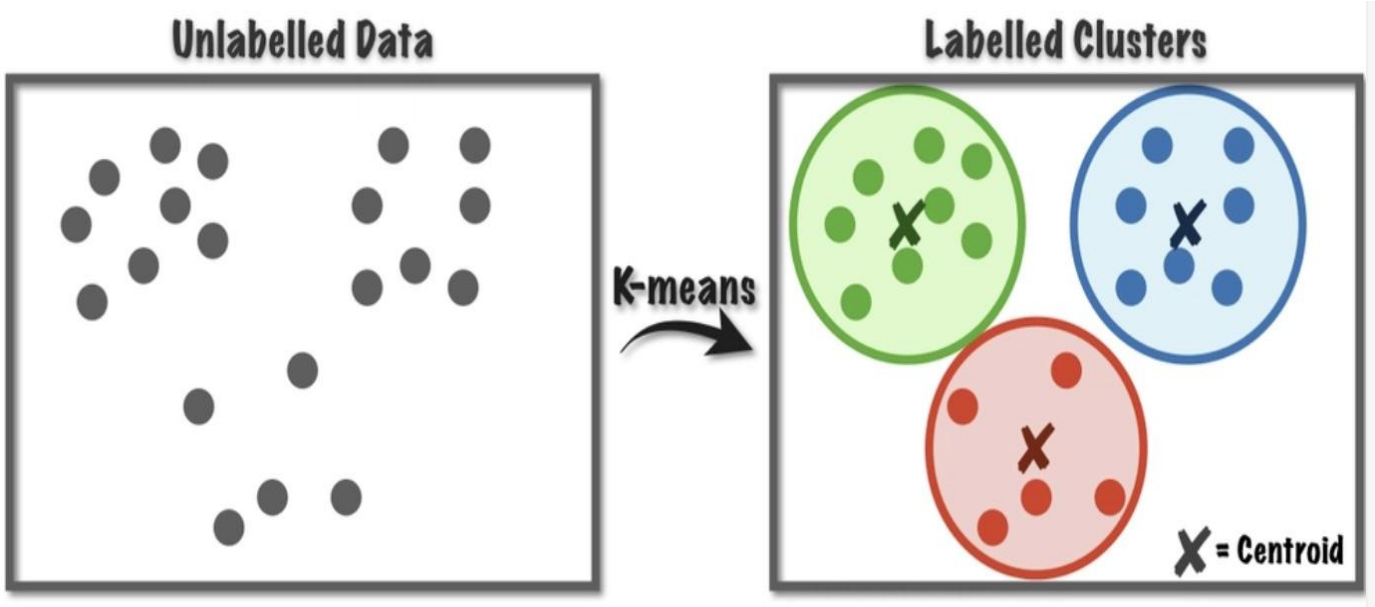
<01>

Definición de clustering

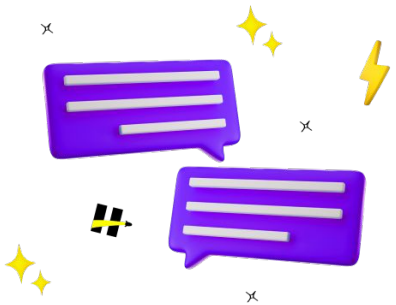
¿Qué es clustering?

- Aprendizaje **no supervisado**: no hay variable objetivo ni etiquetas previas.
- Agrupa observaciones por **similitud** (distancias/similitudes).
- Produce **clusters**: internos homogéneos, entre sí heterogéneos.
- **Reduce complejidad**: sintetiza miles de registros en pocos grupos.
- Descubre **estructura latente** para análisis y decisiones.

¿Cómo se ve el clustering?



Similitud y distancias



- Distancias típicas: **Euclidiana** y **Manhattan**.
- Similitud: **coseno** (convertible a distancia).
- La métrica elegida condiciona las **fronteras** entre grupos.
- Requiere **estandarización** cuando hay escalas distintas.
- Objetivo: aproximar “cercanía” **significativa** al problema.

<02>

Familias de clustering particional (e.g. k-means)

Clustering particional: familia

- Divide el dataset en **k** particiones excluyentes.
- Minimiza **variabilidad interna** y maximiza **separación**.
- Representantes: **K-Means**, **K-Medoids**, **CLARA**, **MiniBatch K-Means**.
- Supone clusters relativamente **compactos**.
- Útil para **segmentación operativa**.

K-Means: idea central

- Cada cluster se representa por su **centroide** (media).
- Minimiza la **SSE/inercia**: suma de distancias cuadráticas al centroide.
- Alterna **asignación** (al centroide más cercano) y **actualización** (recalcular centroides).
- Converge cuando las asignaciones **se estabilizan**.
- Fronteras son **hiperplanos** (lineales).

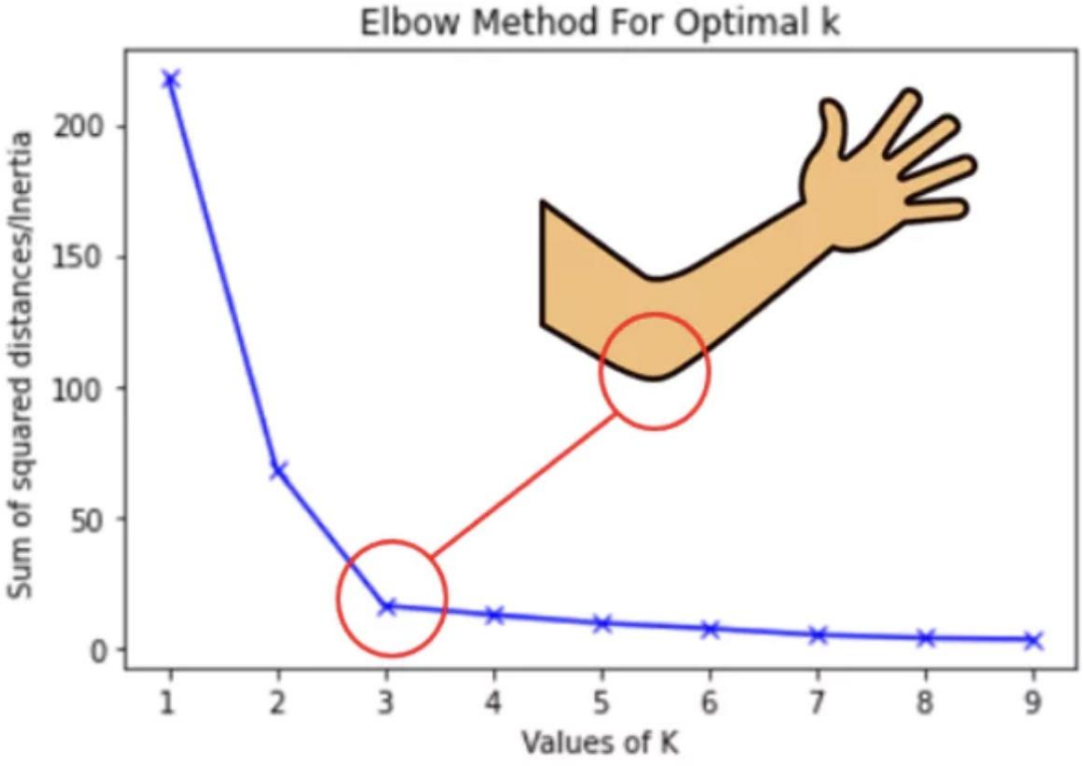
K-Means: hiperparámetros y k

- **k** (número de clusters) define la granularidad.
- **k-means++** mejora la **inicialización** y evita mínimos locales.
- Más **iteraciones** no garantizan mejor estructura.
- Elegir **k**: método del codo + **silueta** + criterio de negocio.
- Reportar **estabilidad** entre corridas.

K-Means: método del codo

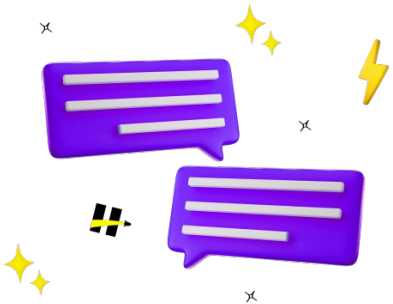
- Inercia decrece de **k=2: 27.133** a **k=10: 14.590**.
- **Sin codo marcado** → estructura **continua** (no divisiones naturales claras).
- Elegir **k=3–5** sirve operativamente.
- El “codo difuso” **es información**, no un fallo.
- Triangular con **silueta** y lectura de **negocio**.

Método del Codo



K-Means: limitaciones y variantes

- Asume clusters **esféricos** y tamaños **similares**.
- **Sensibilidad** a outliers y a la **escala**.
- K-Medoids: usa **representante real** (robusto a outliers).
- **CLARA** y **MiniBatch**: eficiencia en **grandes volúmenes**.
- Preprocesar: **estandarizar** y revisar **outliers**.



<03>

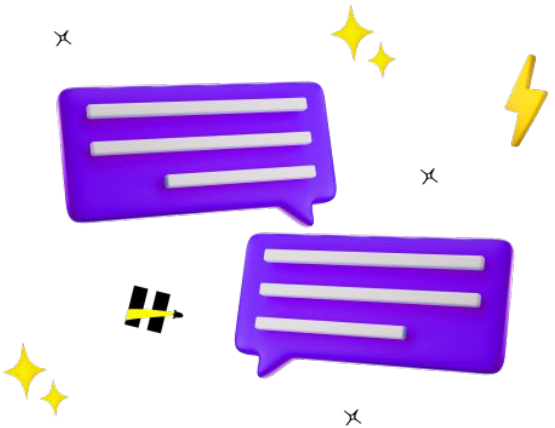
Familias de clustering por densidad (e.g. DBSCAN)

Clustering por densidad: familia

- Define clusters como **regiones densas** separadas por baja densidad.
- No requiere **k** a priori.
- Maneja **formas arbitrarias** y **ruido**.
- Representantes: **DBSCAN** (y **HDBSCAN**).
- Útil cuando la estructura **no es esférica**.

DBSCAN: conceptos clave

- Parámetros: **ε (radio)** y **min_samples** (mínimos vecinos).
- Puntos: **centrales**, **borde**, **ruido**.
- Expansión desde puntos **centrales** por conectividad densa.
- Etiqueta **−1** para ruido/outliers.
- Detecta **anomalías** sin borrarlas.



DBSCAN: selección de parámetros

- **ε** pequeño → muchos clusters minúsculos y mucho **ruido**.
- **ε** grande → **fusiones** y pérdida de detalle.
- Usar **k-distance plot** como guía (codo puede ser **difuso**).
- Densidades muy distintas → considerar **HDBSCAN**.
- La “dificultad” para encontrar varios clusters puede **ser el resultado**.

DBSCAN: ventajas y límites

- Captura **formas no lineales** y **ruido** naturalmente.
- No exige **k** ni centros **esféricos**.
- Sensible a **ε/min_samples** y a densidades **heterogéneas**.
- Coste computacional en **datasets muy grandes** (variantes optimizadas).
- Complementa a K-Means para **auditar** estructura real.

<04>

Evaluación de algoritmos no supervisados

soyhenry.com

Evaluación: inercia (SSE) y “codo”

- **Inercia** ↓ al aumentar **k**; mirar **descenso marginal**.
- Codo = **equilibrio** compacidad vs simplicidad.
- Sin codo claro → **homogeneidad** o densidades mixtas.
- Es **específico** de K-Means (no aplica a DBSCAN).

Evaluación: silueta (-1 a 1)

- s(i) = (b(i) – a(i)) / max{a(i), b(i)}.
- 0.5: fuerte; 0.25–0.5: razonable; <0.25: **débil/solapada**.

- ShopSense: **máx en k=4 = 0.240** → solapamiento **significativo**.
- DBSCAN con un solo cluster: **no** se calcula silueta.
- Lectura conjunta = **historia coherente** de homogeneidad.

Interpretación honesta de métricas

- Silhouette baja ≠ inútil: **segmentos operativos** válidos.
- DBSCAN confirma **continuo** con outliers valiosos.
- Decisiones: **personalización continua** > cortes rígidos.
- Métricas + negocio → **expectativas realistas**.
- Usar ambos enfoques para **validar** la narrativa de datos.

<05>

Reducción de la dimensionalidad

Reducción de dimensionalidad: propósito

- Resolver **alta dimensionalidad** y **redundancia**.
- **Visualizar** estructura, **explorar** relaciones, **mejorar** modelos.
- Proyecciones **2D/3D** para validar clusters y outliers.
- Prepara features **compactas** para pipelines.

PCA (ShopSense): qué revela

- Encuentra **componentes** que maximizan **varianza** (SVD/covarianzas).
- **PC1** ≈ actividad/valor (frecuencia + gasto).
- **PC2** ≈ preferencia de **canal** (digital vs físico).
- **45–50%** de varianza explicada (estructura **distribuida**).
- Visualización: **superposición** entre clusters de K-Means, acorde a **0.240**.

t-SNE y uso práctico

- t-SNE preserva **vecindad local**; puede **exagerar** separaciones.
- Útil para **explorar** variaciones sutiles; no para fronteras globales.
- Con DBSCAN: confirma **grupo denso** + **outliers** periféricos.
- PCA/t-SNE como **mapas narrativos** para negocio.
- Las componentes sirven como **features** en recomendadores.

<DATA SCIENCE/>

Vayamos a la práctica

Homework

Consigna

En esta actividad pondrás en práctica los conceptos vistos sobre **clustering** y **reducción de la dimensionalidad**, utilizando un dataset real de clientes minoristas.

El objetivo es reproducir un flujo analítico completo que permita segmentar a los clientes según sus características de compra, visualizar los grupos en un espacio reducido y reflexionar cómo esta segmentación puede servir como **proxy o paso previo a un sistema de recomendación**.

Tu tarea consiste en realizar un análisis no supervisado que incluya:

1. Exploración del dataset y tratamiento de variables numéricas relevantes.
2. Aplicación de **K-Means** para generar clusters representativos de clientes.
3. Evaluación del modelo mediante el **método del codo** y la **silueta**.
4. Representación visual de los clusters usando **PCA** y/o **t-SNE**.
5. Breve reflexión sobre cómo los clusters obtenidos podrían alimentar un sistema de recomendación (por similitud entre clientes o perfiles de compra).

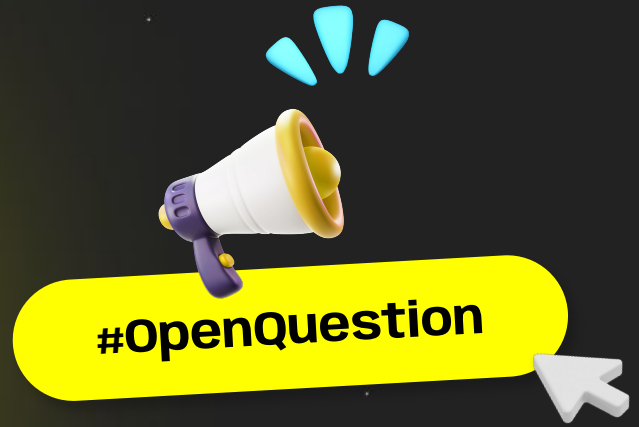
Tareas a realizar

1. Cargar el dataset y realizar limpieza básica (eliminar nulos, normalizar si es necesario).
2. Implementar **K-Means** con al menos tres valores distintos de *k*.
3. Evaluar con **método del codo (inercia/SSE)** y **silueta (−1 a 1)** para seleccionar el k óptimo.
4. Generar una visualización 2D con **PCA** mostrando los clusters.
5. Incluir un breve párrafo de interpretación (1–2 líneas por cluster) describiendo patrones observados.

Extra credits optativo

- Agregar una visualización con **t-SNE** para comparar la estructura local de los clusters.
- Aplicar un algoritmo alternativo (**DBSCAN** o **Agglomerative Clustering**) y comparar resultados.
- Generar una columna adicional con una etiqueta interpretativa de cada grupo (por ejemplo: *High Value*, *Occasional Buyer*, *Digital User*).
- Escribir una reflexión corta sobre cómo los clusters podrían servir de base para un **sistema de recomendación tipo "usuarios similares"**.

HENRY



¿Preguntas?

HENRY

¡Muchas gracias!

→ soyhenry.com