UNIVERSIDAD TECNOLÓGICA DE CHIHUAHUA

DESARROLLO Y GESTIÓN DE SOFTWARE

IV.1. Algoritmos de agrupación

EXTRACCIÓN DE CONOCIMIENTO EN BASES DE DATOS

PRESENTA:

KARLA ALEJANDRA DE LA CRUZ ZEA

DOCENTE:

ING. LUIS ENRIQUE MASCOTE CANO

29 de noviembre de 2025

Contenido

Introducción

La extracción de conocimiento en grandes volúmenes de datos requiere técnicas que permitan organizar, simplificar y comprender estructuras complejas. Dos de los enfoques más utilizados en esta área son el clustering y la reducción de dimensionalidad.

El clustering agrupa datos similares sin necesidad de etiquetas, permitiendo descubrir patrones, segmentos o comportamientos ocultos. Por otro lado, la reducción de dimensionalidad transforma conjuntos de datos con muchas variables en representaciones más compactas, sin perder la mayor parte de la información relevante. Esto facilita la visualización, acelera los modelos de machine learning y elimina ruido.

Este reporte analiza los principales algoritmos de agrupación y reducción de dimensionalidad, explicando su funcionamiento, parámetros clave y aplicaciones prácticas dentro del análisis de datos moderno.

Algoritmos de agrupación (Clustering)

K-Means

Principio de funcionamiento

K-Means divide los datos en *k* grupos minimizando la distancia entre los puntos y el centro del cluster (centroide).
El algoritmo sigue estos pasos:

1. Se eligen *k* centroides iniciales.
2. Cada punto se asigna al centroide más cercano.
3. Se recalculan los centroides con el promedio de los puntos asignados.
4. Se repite hasta que ya no cambien las asignaciones.

Parámetros clave

- **k**: número de clusters.

- **Método de inicialización** (k-means++, random).
- **Número máximo de iteraciones.**

Ventajas

- Simple, rápido y escalable.
- Funciona bien con datos grandes y bien separados.

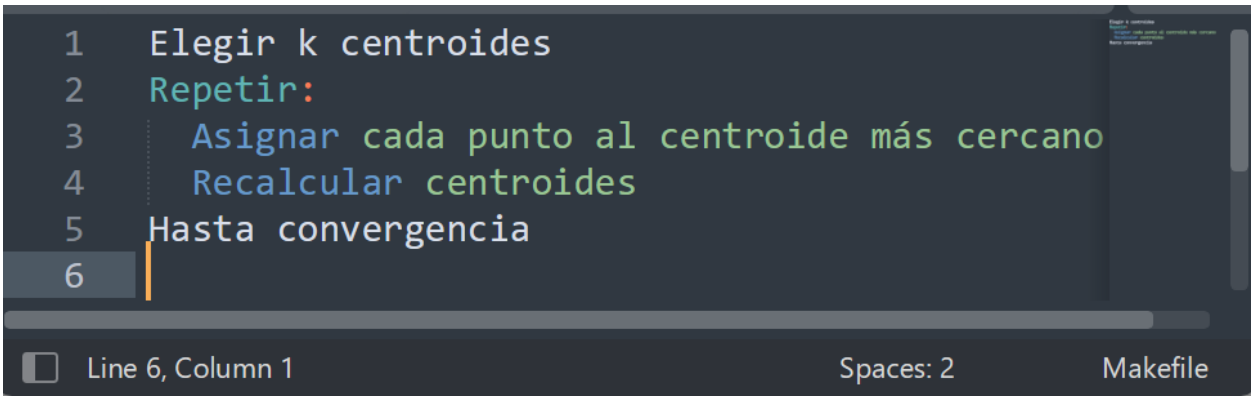
Limitaciones

- Requiere definir *k* previamente.
- No identifica formas no esféricas.
- Sensible a outliers.

Ejemplo de uso

Segmentación de clientes por edad e ingresos.

Pseudocódigo:



Clustering jerárquico aglomerativo

Principio de funcionamiento

Construye jerarquías de agrupación uniendo pares de puntos o clusters cercanos.
Etapas:

1. Cada punto inicia como un cluster individual.
2. Se unen los dos clusters más cercanos.

3. Se repite hasta obtener un único cluster o la cantidad deseada.

Parámetros clave

- **Método de enlace**: single, complete, average, Ward.
- **Métrica de distancia**: Euclídea, Manhattan, Coseno, etc.

Ventajas

- No requiere definir *k* inicialmente.
- Genera dendrogramas interpretables.

Limitaciones

- Costoso en datasets grandes.
- Sensible al ruido según el método de enlace.

Ejemplo de aplicación

Agrupación de documentos por similitud temática.

DBSCAN

Principio de funcionamiento

Agrupa puntos que están densamente conectados y marca como ruido los puntos aislados.
Clasifica puntos como:

- **Core:** suficientes vecinos dentro de un radio.
- **Border:** cerca de un core, pero no con suficientes vecinos.
- **Noise:** aislados.

Parámetros clave

- **eps:** distancia máxima para considerar vecinos.
- **min_samples:** número mínimo de puntos para formar un cluster.

Ventajas

- Detecta clusters de forma arbitraria.
- Maneja bien outliers.
- No requiere *k*.

Limitaciones

- Sensible a eps y min_samples.
- Difícil para datos con densidades muy distintas.

Ejemplo de aplicación

Detección de zonas con actividad anómala en GPS o sensores.

Algoritmos de reducción de dimensionalidad

Análisis de Componentes Principales (PCA)

Fundamento conceptual

Reduce el número de variables transformándolas en nuevas componentes que capturan la mayor varianza de los datos.
Se basa en descomposición en valores propios (eigenvalues).

Parámetros clave

- **n_components:** cantidad de componentes deseadas.
- **Escalado previo:** recomienda estandarizar.

Ventajas

- Conserva la estructura global del dataset.
- Rápido y ampliamente usado.

Limitaciones

- Solo captura relaciones lineales.
- Difícil interpretar las nuevas componentes.

Ejemplo

Reducir 10 variables biométricas a 2 para graficarlas.

t-SNE

Fundamento conceptual

Embebe datos de alta dimensión en 2D o 3D manteniendo relaciones locales (vecinos). Funciona modelando probabilidades de cercanía entre puntos.

Parámetros clave

- **perplexity:** define el número aproximado de vecinos.
- **learning_rate:** velocidad de optimización.
- **n_iter:** iteraciones del algoritmo.

Ventajas

- Excelente visualización de clusters.
- Maneja relaciones no lineales.

Limitaciones

- Computacionalmente costoso.
- No mantiene estructura global.
- No sirve para predicción futura ni reconstrucción.

Ejemplo

Visualizar grupos de clientes o tipos de imágenes en 2D.

Comparativa

Tipo de método	Cuándo usarlo	Ventajas	Limitaciones
Clustering	Descubrir grupos ocultos, segmentar clientes, detectar comportamientos similares.	No requiere etiquetas, útil para exploración.	Sensible a parámetros, puede fallar con ruido o alta dimensión.
Reducción de dimensionalidad	Visualizar datos, eliminar ruido, acelerar modelos, preparar clustering.	Permite simplificar datos y mejorar modelos.	Puede perder información relevante.

Conclusión

Los algoritmos de clustering revelan patrones naturales dentro de los datos, mientras que los métodos de reducción de dimensionalidad facilitan la visualización y el procesamiento de información compleja. Ambos son herramientas complementarias: uno organiza los datos y el otro los simplifica para analizarlos mejor.
El uso estratégico de ambos procesos permite mejorar tareas como segmentación, análisis exploratorio y preparación de datos para machine learning avanzado.

Referencias

Géron, A. (2022). *Hands-On Machine Learning with Scikit-Learn, Keras & TensorFlow* (3rd ed.). O'Reilly Media.

Han, J., Pei, J., & Kamber, M. (2022). *Data Mining: Concepts and Techniques* (4th ed.). Morgan Kaufmann.

Scikit-learn. (2024). *Clustering and Manifold Learning Documentation*. https://scikit-learn.org/stable/modules/clustering.html

Van der Maaten, L., & Hinton, G. (2008). *Visualizing data using t-SNE*. Journal of Machine Learning Research.

OpenAI. (2025). *ChatGPT Technical Guidance for Data Science Applications*.