UNIVERSIDAD TECNOLÓGICA DE CHIHUAHUA

Ingeniería en Desarrollo y Gestión de Software

Extracción de Conocimientos de Bases de Datos

IV.1. Algoritmos de agrupación (25%)

IDGS91N

PRESENTAN:

Giselle Cantú Chávez

NOMBRE DEL DOCENTE:

Ing. Luis Enrique Mascote Cano

Chihuahua, Chih., 29 de noviembre de 2025

Universidad Tecnológica de Chihuahua

Índice

Universidad Tecnológica de Chihuahua

Índice

1. Introducción

En extracción de conocimiento, el análisis no supervisado permite descubrir estructuras ocultas sin utilizar etiquetas. Es útil cuando tenemos datos complejos, mezclados o demasiado numerosos para analizarlos manualmente. Dos técnicas clave son el **clustering**, que agrupa datos similares, y la **reducción de dimensionalidad**, que simplifica conjuntos con muchas variables sin perder información esencial.

El clustering sirve para organizar, segmentar, perfilar y entender patrones, mientras que la reducción de dimensionalidad ayuda a visualizar mejor, acelerar algoritmos e incluso eliminar ruido. En este reporte describo tres algoritmos de agrupación y dos de reducción de dimensionalidad, con sus principios, parámetros, ventajas, limitaciones y ejemplos claros.

2. Algoritmos de agrupación

A continuación, describo **K-Means**, **DBSCAN** y **Clustering Jerárquico**, tres de los métodos más utilizados en minería de datos.

2.1 K-Means

Principio de funcionamiento

K-Means divide los datos en *k* grupos, asignando cada punto al centroide más cercano. Después recalcula los centroides hasta que las distancias internas de cada grupo se estabilizan.

Parámetros clave

- **k:** número de clusters.
- **Iteraciones máximas:** límite del algoritmo.

- **Inicialización:** método para colocar centroides (k-means++, aleatoria).
- **Distancia:** normalmente euclidiana.

Ventajas

- Muy rápido y escalable para grandes conjuntos de datos.
- Fácil de interpretar y de implementar.
- Funciona bien cuando los clusters son esféricos y del mismo tamaño.

Limitaciones

- Requiere fijar *k* antes de empezar.
- Sensible a valores atípicos.
- No sirve bien con clusters de forma irregular.

Ejemplo simple (pseudocódigo)

Elegir k centroides iniciales

Repetir:

Asignar cada punto al centroide más cercano

Recalcular centroides

Hasta que no cambien asignaciones

2.2 DBSCAN

Principio de funcionamiento

DBSCAN agrupa puntos basándose en densidad. Identifica zonas densas y separa regiones dispersas como ruido. No requiere especificar cuántos clusters habrá.

Parámetros clave

- **eps:** distancia máxima entre puntos para considerarlos vecinos.
- **min_samples:** mínimo de puntos para crear una región densa.

Ventajas

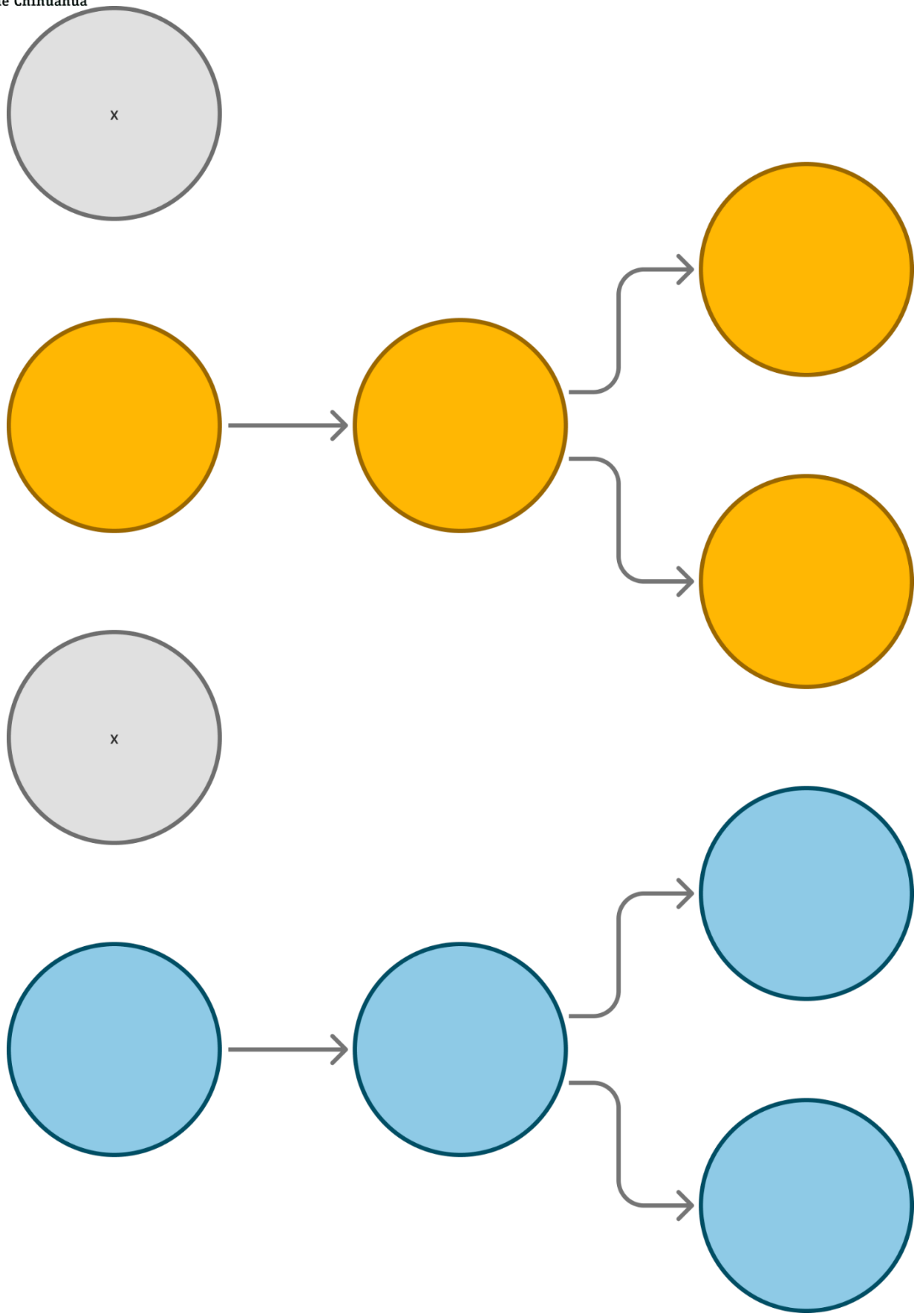
- Detecta clusters de forma arbitraria.
- Identifica outliers de manera natural.
- No necesita un número de clusters predefinido.

Limitaciones

- *eps* es difícil de ajustar.
- No es ideal cuando las densidades varían mucho entre grupos.

Ejemplo simple

Un gráfico donde regiones densas se colorean juntas y puntos aislados quedan como “ruido”.


x
x

2.3 Clustering Jerárquico (Aglomerativo)

Principio de funcionamiento

Empieza considerando cada punto como un cluster individual. Luego agrupa los más cercanos progresivamente hasta formar una estructura jerárquica. Se representa con un dendrograma.

Parámetros clave

- **Métrica de distancia:** euclidiana o Manhattan.
- **Linkage:** single, complete, average, ward.

Ventajas

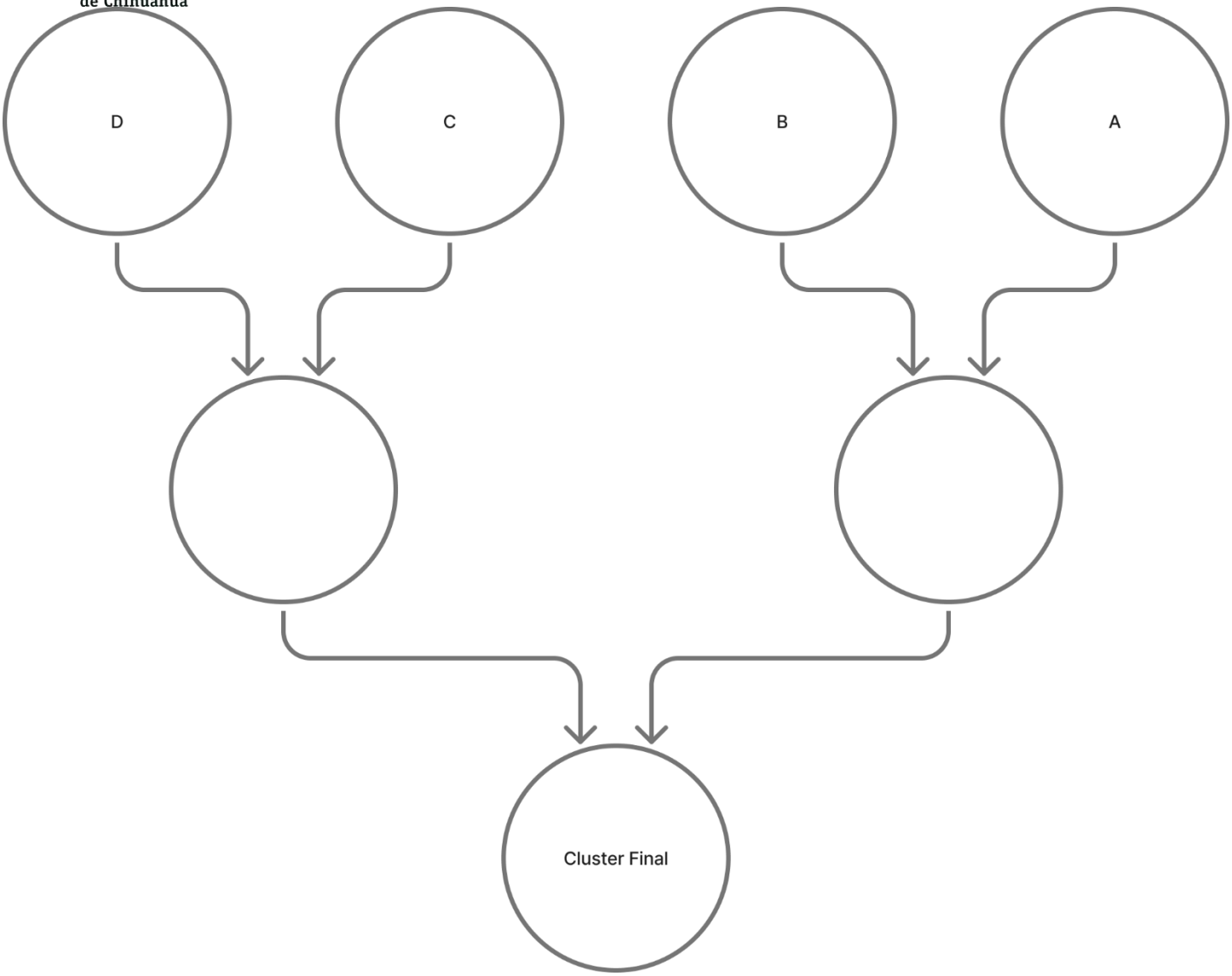
- Permite visualizar la relación entre clusters con dendrogramas.
- No requiere especificar *k* de inicio.
- Identifica estructuras más complejas.

Limitaciones

- Computacionalmente costoso.
- Difícil de escalar a millones de puntos.

Ejemplo

Un dendrograma que muestra cómo se fusionan los puntos hasta formar grupos.


D
C
B
A
Cluster Final

3. Algoritmos de reducción de dimensionalidad

A continuación, explico **PCA** y **t-SNE**, dos herramientas fundamentales para simplificar y visualizar datos complejos.

3.1 PCA (Análisis de Componentes Principales)

Fundamento matemático / conceptual

PCA convierte variables originales en nuevas variables llamadas componentes principales, ordenadas por la cantidad de varianza que explican. Se basa en descomposición en valores propios de la matriz de covarianza.

Parámetros clave

- **Número de componentes** a retener.
- **Escalado de datos:** recomendado antes de aplicar PCA.

Ventajas

- Reduce dimensionalidad manteniendo mayor varianza posible.
- Favorece visualizaciones en 2D o 3D.
- Acelera algoritmos dependientes de distancia.

Limitaciones

- Solo captura relaciones lineales.
- Puede ser difícil de interpretar.

Ejemplo sencillo

Tomar un dataset con 20 variables y conservar solo 2 componentes para visualizarlo.

3.2 t-SNE

Fundamento matemático / conceptual

t-SNE convierte distancias de alta dimensión en probabilidades y las proyecta a un espacio de menor dimensión maximizando similitudes locales. Se usa para visualización más que para modelado.

Parámetros clave

- **Perplexity:** equilibrio entre vecindarios pequeños y grandes.
- **Learning rate.**
- **Número de iteraciones.**

Ventajas

- Excelente para visualizar clusters en datos de alta dimensión.
- Revela estructuras que PCA no detecta.

Limitaciones

- No conserva relaciones globales.
- Alto costo computacional.
- No sirve para predicción futura (no transforma nuevos datos fácilmente).

Ejemplo

Reducir un dataset de imágenes MNIST a 2D para visualizar grupos de dígitos.

4. Comparativa y conclusiones

Clustering vs. reducción de dimensionalidad (comparación general)

Técnica	¿Qué hace?	¿Cuándo usarla?
Clustering	Agrupa datos por similitud	Segmentación, detección de patrones, perfiles
Reducción de dimensionalidad	Simplifica variables sin perder demasiada información	Visualización, eliminación de ruido, acelerar algoritmos

¿Cuándo usar cada una?

- **Clustering** tiene prioridad cuando el objetivo es descubrir grupos naturales, segmentar usuarios o encontrar patrones sin etiquetas.
- **Reducción de dimensionalidad** es mejor cuando el dataset tiene demasiadas variables, cuando hay ruido o cuando necesito visualizar datos de forma más clara antes de aplicar clustering.
  De hecho, muchos pipelines combinan ambas técnicas: primero reducen dimensiones con PCA y luego aplican clustering como K-Means para mejorar resultados.

Conclusión general

El análisis no supervisado abre camino para explorar datos sin etiquetas y descubrir estructuras ocultas. K-Means, DBSCAN y Clustering Jerárquico ofrecen maneras diferentes de agrupar datos según la forma, densidad o jerarquía. Por otro lado, PCA y t-SNE facilitan simplificar y visualizar datasets con muchas variables. Comprender cómo funcionan, sus parámetros y sus limitaciones permite elegir la mejor herramienta según la naturaleza del problema.

5. Fuentes de apoyo

Aggarwal, C. C. (2023). *Data clustering: Algorithms and applications*. Springer. https://doi.org/10.1007/978-3-031-11265-8

McInnes, L., Healy, J., & Melville, J. (2018). *UMAP: Uniform Manifold Approximation and Projection*. arXiv. https://arxiv.org/abs/1802.03426

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., et al. (2011). *Scikit-learn: Machine learning in Python*. Journal of Machine Learning Research, 12, 2825–2830. https://jmlr.org/papers/v12/pedregosa11a.html

Van der Maaten, L., & Hinton, G. (2008). *Visualizing data using t-SNE*. Journal of Machine Learning Research, 9, 2579–2605. https://www.jmlr.org/papers/v9/vandermaaten08a.html

Wold, S., Esbensen, K., & Geladi, P. (1987). *Principal Component Analysis*. Chemometrics and Intelligent Laboratory Systems, 2(1-3), 37–52. https://doi.org/10.1016/0169-7439(87)80084-9