UNIVERSIDAD TECNOLÓGICA DE CHIHUAHUA

TECNOLOGÍAS DE LA INFORMACIÓN

Extracción de Conocimiento en Bases de Datos

DOCENTE:

Enrique Mascote

Algoritmos de agrupación

PRESENTA:

Iván Eduardo Martínez Martínez

Grupo:

IDGS91N

Chihuahua, Chih., 29 de noviembre de 2025

Introducción

En el proceso de extracción de conocimiento a partir de grandes volúmenes de datos, las técnicas de análisis no supervisado desempeñan un papel fundamental. Entre ellas, el **clustering** permite identificar patrones ocultos y agrupar objetos según su similitud, sin necesidad de etiquetas. Por otra parte, la **reducción de dimensionalidad** simplifica datasets con muchas variables, conservando la estructura esencial de los datos y facilitando su interpretación, visualización y procesamiento.

Este reporte presenta una descripción detallada de los principales algoritmos de agrupación y de reducción de dimensionalidad, explicando su funcionamiento, parámetros clave, ventajas, limitaciones y escenarios de aplicación. El objetivo es comprender cuándo es adecuado utilizar cada técnica y cómo se complementan en procesos reales de minería de datos.

1. Algoritmos de Agrupación (Clustering)

Se seleccionaron tres algoritmos representativos: **K-Means**, **Clustering Jerárquico Aglomerativo** y **DBSCAN**.

1.1 K-Means

Principio de funcionamiento

K-Means es un algoritmo basado en partición que divide los datos en *k* grupos, minimizando la distancia entre los puntos y el centroide de cada cluster. Utiliza un proceso iterativo: inicializa centroides, asigna cada punto al centroide más cercano y actualiza los centroides hasta converger.

Parámetros clave

- **k:** número de clusters.
- **Inicialización:** método para asignar centroides iniciales (ej. *k-means*++).
- **Métrica de distancia:** generalmente Euclidiana.

Ventajas

- Rápido, eficiente y fácil de implementar.
- Funciona bien con clusters esféricos y grandes volúmenes de datos.

Limitaciones

- Requiere definir *k* previamente.
- Sensible a outliers.
- No funciona bien con clusters de forma irregular.

Ejemplo (pseudocódigo simple)

Inicializar k centroides aleatorios

Repetir:

  Asignar cada punto al centroide más cercano

  Recalcular centroides

Hasta que los centroides no cambien

1.2 Clustering Jerárquico Aglomerativo

Principio de funcionamiento

Construye una jerarquía de clusters mediante la fusión progresiva. Cada punto inicia como un cluster separado y, en cada paso, los dos clusters más similares se combinan hasta formar un único cluster o hasta alcanzar el número deseado.

Parámetros clave

- **Método de enlace:** single, complete, average, ward.
- **Métrica de distancia:** Euclidiana, Manhattan, etc.
- **Número de clusters (opcional):** se define cortando el dendrograma.

Ventajas

- No requiere especificar *k* inicialmente.
- Proporciona un dendrograma que facilita el análisis visual.
- Útil para estructuras jerárquicas reales.

Limitaciones

- Computacionalmente costoso para datasets grandes.
- Decisiones de fusión no se pueden revertir.
- Sensible al ruido.

Ejemplo conceptual (diagrama textual)

Datos -> Calcular distancias -> Unir los dos más cercanos

-> Actualizar distancias -> Repetir

-> Generar dendrograma

1.3 DBSCAN (Density-Based Spatial Clustering of Applications with Noise)

Principio de funcionamiento

Identifica clusters como regiones densas separadas por áreas de baja densidad. Clasifica puntos como *core*, *border* o *noise* según su densidad local.

Parámetros clave

- **ε (epsilon):** radio del vecindario.
- **minPts:** mínimo de puntos requeridos para considerar una región densa.

Ventajas

- Detecta clusters de forma arbitraria.
- Maneja outliers naturalmente.
- No requiere definir número de clusters.

Limitaciones

- Sensible a la selección de ε.
- Difícil de aplicar en alta dimensionalidad.
- Parámetros no triviales en datasets heterogéneos.

Ejemplo básico (pseudocódigo)

Para cada punto no visitado:

Marcar como visitado

Obtener vecinos dentro de ε

Si vecinos < minPts: etiquetar como ruido

Si vecinos >= minPts: crear cluster y expandir

2. Algoritmos de Reducción de Dimensionalidad

Se seleccionaron **PCA** y **t-SNE**, dos de los métodos más utilizados tanto en análisis exploratorio como en visualización.

2.1 Análisis de Componentes Principales (PCA)

Fundamento matemático

PCA transforma el espacio original mediante una descomposición en valores propios de la matriz de covarianza. Las nuevas componentes son combinaciones lineales ortogonales que explican la mayor varianza posible.

Parámetros clave

- **Número de componentes:** cuántas dimensiones mantener.
- **Estandarización:** decisión de escalar o no los datos.
- **Varianza explicada:** porcentaje objetivo (ej. 90%).

Ventajas

- Reduce ruido y redundancia.
- Facilita visualización y procesamiento.
- Computacionalmente eficiente.

Limitaciones

- Solo captura relaciones lineales.
- Difícil interpretación de componentes.
- Sensible a escalamiento.

Ejemplo conceptual

1. Estándarizar datos
2. Calcular matriz de covarianza
3. Obtener eigenvalues y eigenvectors
4. Ordenar por varianza explicada
5. Proyectar datos en componentes seleccionadas

2.2 t-SNE (t-distributed Stochastic Neighbor Embedding)

Fundamento conceptual

t-SNE transforma datos de alta dimensión en un espacio de menor dimensión preservando la relación de proximidad entre puntos. Modela probabilidades de similitud y minimiza la divergencia Kullback–Leibler entre distribuciones de alta y baja dimensión.

Parámetros clave

- **Perplexity:** controla balance entre estructura local y global.
- **Learning rate:** velocidad de actualización.
- **Iteraciones:** número de pasos de optimización.

Ventajas

- Excelente para visualizar clústeres no lineales.
- Capta estructuras complejas.
- Muy usado en imágenes y embeddings.

Limitaciones

- Alto costo computacional.
- No es reproducible sin fijar semilla.
- Difícil interpretación cuantitativa.

Ejemplo conceptual

1. Calcular similitudes en alta dimensión
2. Crear distribución en baja dimensión al azar
3. Minimizar KL-divergence iterativamente
4. Obtener mapa visual 2D/3D

3. Comparativa y Conclusiones

3.1 Comparativa general

Aspecto	Clustering	Reducción de Dimensionalidad
Objetivo	Agrupar datos según similitud	Simplificar variables manteniendo estructura
Tipo de salida	Etiquetas o grupos	Nuevas variables o proyecciones
Uso principal	Descubrir patrones	Visualizar, reducir ruido, acelerar modelos
Dependencia de etiquetas	No requiere	No requiere
Problemas que resuelve	Segmentación, detección de patrones	Alta dimensionalidad, visualización

3.2 ¿Cuándo usar cada uno?

- **Clustering** se usa cuando el objetivo es agrupar clientes, documentos, imágenes o patrones sin etiquetas previas.

- **Reducción de dimensionalidad** se prioriza cuando el dataset tiene demasiadas variables, lo cual dificulta la visualización, afecta el rendimiento o genera ruido.

- En muchos casos, ambas técnicas se complementan: por ejemplo, aplicar PCA antes de K-Means mejora estabilidad y velocidad del clustering.

Conclusiones

El análisis no supervisado ofrece herramientas poderosas para descubrir la estructura subyacente de los datos. Los algoritmos de clustering permiten segmentar y analizar patrones sin necesidad de etiquetas, mientras que los métodos de reducción de dimensionalidad permiten simplificar la información manteniendo su esencia.

La elección de la técnica adecuada depende del objetivo analítico, el tipo de datos y las limitaciones computacionales. En la práctica profesional, ambos enfoques se usan de manera conjunta para explorar, visualizar y extraer conocimiento de datasets complejos.

Referencias (formato APA)

Abdi, H., & Williams, L. J. (2010). *Principal component analysis*. Wiley Interdisciplinary Reviews: Computational Statistics, 2(4), 433–459.

Van der Maaten, L., & Hinton, G. (2008). *Visualizing data using t-SNE*. Journal of Machine Learning Research, 9, 2579–2605.

Han, J., Kamber, M., & Pei, J. (2012). *Data Mining: Concepts and Techniques* (3rd ed.). Morgan Kaufmann.

Scikit-Learn. (2024). *Clustering and dimensionality reduction documentation*. https://scikit-learn.org (omite el enlace en Word si gustas)