UNIVERSIDAD TECNOLÓGICA DE CHIHUAHUA

TECNOLOGÍAS DE LA INFORMACIÓN

Extraccion de Conocimiento de Base de Datos

Algoritmos de Agrupación

IDGS91N

Alumno:

Erick Fabian Terrazas Hernandez

Docente:

Enrique Mascote

Chihuahua, Chih., 30 de noviembre de 2025

INTRODUCCIÓN

En el campo de la minería de datos y la extracción de conocimiento, los algoritmos de agrupación (clustering) y reducción de dimensionalidad desempeñan un papel fundamental al permitir ordenar, simplificar y comprender grandes volúmenes de información. El clustering se utiliza para encontrar patrones naturales en los datos sin necesidad de etiquetas previas, agrupando objetos similares entre sí. Por otro lado, la reducción de dimensionalidad busca condensar un conjunto de variables en un espacio más pequeño, preservando la mayor cantidad posible de información relevante. Ambos procesos ayudan a descubrir estructuras ocultas, reducir ruido, mejorar la visualización y facilitar el entrenamiento de modelos de aprendizaje automático.

ALGORTIMOS DE AGRUPACIÓN

1. **K-means**

2. **Principio de funcionamiento:**

3. K-means divide los datos en *k* clusters asignando cada punto al centroide más cercano. Posteriormente recalcula los centroides y repite el proceso hasta que las asignaciones ya no cambian significativamente. Su objetivo es minimizar la suma de distancias internas dentro de cada cluster.

Parámetros clave:

- *k*: número de clusters deseados.
- Criterio de convergencia.
- Número de iteraciones máximas.
- Método de inicialización de centroides (por ejemplo, k-means++).

Ventajas:

- Fácil de implementar.
- Rápido y escalable para grandes conjuntos de datos.
- Funciona bien con clusters esféricos y bien separados.

Limitaciones:

- El usuario debe elegir *k*.
- Sensible a valores atípicos.
- No detecta bien clusters de formas no esféricas.

Ejemplo de aplicación (pseudocódigo):

1. Elegir *k*.
2. Inicializar centroides aleatoriamente.
3. Asignar cada punto al centroide más cercano.
4. Recalcular centroides.
5. Repetir pasos 3–4 hasta convergencia.

2. **Clustering jerárquico aglomerativo**
   Principio de funcionamiento:
   Inicia considerando cada punto como un cluster independiente. Luego, en cada iteración, une los dos clusters más similares según una medida de distancia (single-link, complete-link o average-link), creando una estructura en forma de árbol llamada dendrograma.

Parámetros clave:

- Métrica de distancia (Euclidiana, Manhattan, etc.).
- Método de enlace (linkage).
- Nivel de corte del dendrograma.

Ventajas:

- No requiere especificar *k* desde el inicio.
- Genera una estructura jerárquica fácil de interpretar.
- Útil para detectar relaciones anidadas.

Limitaciones:

- Computacionalmente costoso para grandes bases de datos.
- Sensible al ruido.
- Una vez combinados clusters, no puede deshacerse la unión.

Ejemplo (pseudocódigo):

1. Cada punto es su propio cluster.
2. Calcular matriz de distancias.
3. Unir clusters más cercanos.
4. Actualizar matriz.
5. Repetir hasta tener un solo cluster.

3. **DBSCAN**

   Principio de funcionamiento:

   DBSCAN identifica clusters basados en densidad. Define un punto como núcleo si tiene suficientes vecinos dentro de un radio específico. Los puntos se agrupan si están densamente conectados; los que no pertenecen a ningún cluster se consideran ruido.

Parámetros clave:

- *eps*: radio de vecindad.
- *minPts*: número mínimo de puntos para formar un núcleo.

Ventajas:

- No requiere especificar el número de clusters.
- Detecta clusters con formas arbitrarias.
- Efectivo contra el ruido.

Limitaciones:

- Sensible a la selección de *eps* y *minPts*.
- Difícil de aplicar en datos de muy alta dimensión.
- Puede fallar cuando hay variaciones grandes de densidad.

Ejemplo (pseudocódigo):

1. Para cada punto no visitado:
   a. Marcarlo como visitado.
   b. Obtener sus vecinos.
   c. Si el número de vecinos ≥ minPts, crear un cluster y expandirlo.
2. Si no cumple, marcar como ruido.

ALGORITMOS DE REDUCCIÓN DE DIMENSIONALIDAD

1. **Análisis de Componentes Principales (PCA)**

Fundamento matemático:

PCA utiliza descomposición de valores propios sobre la matriz de covarianza para identificar las direcciones (componentes principales) que explican la mayor varianza. Proyecta los datos en un nuevo espacio reducido, maximizando la información retenida.

Parámetros clave:

- Número de componentes a conservar.
- Método de estandarización previa.
- Tipo de descomposición (SVD o covarianza).

Ventajas:

- Reduce ruido y redundancia.
- Mejora visualización en 2D o 3D.
- Útil para acelerar algoritmos supervisados.

Limitaciones:

- Es lineal; no captura relaciones no lineales.
- Difícil de interpretar componentes.
- Sensible a escalamiento de datos.

Ejemplo sencillo:

1. Estandarizar datos.
2. Calcular matriz de covarianza.
3. Obtener vectores propios.
4. Elegir componentes principales.
5. Proyectar los datos.

2. **t-SNE**

   Fundamento conceptual:

   t-SNE transforma distancias altas en probabilidades de similitud y luego crea un

espacio reducido donde estas similitudes se preservan. Minimiza la divergencia entre distribuciones en alta y baja dimensión, generando visualizaciones muy claras para datos complejos.

Parámetros clave:

- *perplexity* (controla densidad local).
- Número de iteraciones.
- *learning rate*.

Ventajas:

- Excelente visualización de clústeres complejos.
- Captura relaciones no lineales.
- Útil en datos multidimensionales como imágenes o embeddings.

Limitaciones:

- No es adecuado para reducción con propósito de modelado.
- Alto costo computacional.
- Resultados no determinísticos sin semilla fija.

Ejemplo (conceptual):

1. Calcular similitudes en alta dimensión.
2. Inicializar posiciones en 2D o 3D.
3. Ajustar posiciones para minimizar divergencia.
4. Mostrar mapa resultante.

COMPARATIVA Y CONCLUSIONES

Clustering vs. Reducción de dimensionalidad (comparativa):

Aspecto	Clustering	Reducción de dimensionalidad
Objetivo	Agrupar datos según similitud	Simplificar representación de variables
Tipo	No supervisado	No supervisado (generalmente)
Resultados	Etiquetas de grupos	Nuevas variables o representación compacta
Uso principal	Descubrir patrones	Visualización, preprocesamiento
Sensibilidad al ruido	Alta (según algoritmo)	PCA reduce ruido; t-SNE sensible

Situaciones prácticas:

- Se prioriza clustering cuando se desea dividir una base en grupos similares, detectar patrones o segmentar clientes.
- Se prioriza reducción de dimensionalidad cuando existe un número muy grande de variables y se necesita visualizar, acelerar o eliminar redundancia.

Conclusiones generales:

Los algoritmos de agrupación permiten identificar patrones en datos sin etiquetas y revelar estructuras ocultas. Por su parte, la reducción de dimensionalidad facilita simplificar la información manteniendo características relevantes. Ambos enfoques se complementan y, en muchos proyectos de minería de datos, suelen usarse de manera conjunta para mejorar análisis y modelos posteriores.

REFERENCIAS

Jolliffe, I. T. (2002). *Principal Component Analysis*. Springer.

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

Scikit-learn. (2024). *User Guide: Clustering and Dimensionality Reduction*. https://scikit-learn.org