UNIVERSIDAD TECNOLÓGICA DE CHIHUAHUA

Desarrollo y Gestión de Software

Extracción de Conocimiento en Bases de Datos

Algoritmos de agrupación

IDGS 91N

PRESENTA:

T.S.U. HUGO URIEL CHAPARRO ESTRADA

DOCENTE:

Enrique Mascote

Chihuahua, Chih., 14 oct 2025

Introducción

En el ámbito de la ciencia de datos y la minería de datos, el análisis no supervisado desempeña un papel fundamental al descubrir patrones, tendencias o estructuras ocultas en conjuntos de datos sin etiquetas. Dos técnicas clave de este enfoque son el clustering (agrupamiento) y la reducción de dimensionalidad. El clustering permite identificar subconjuntos de datos similares entre sí, lo que resulta útil en segmentación de clientes, análisis de imágenes, bioinformática y más. Por otro lado, la reducción de dimensionalidad busca simplificar los datos eliminando características redundantes o irrelevantes, lo cual mejora la eficiencia del procesamiento, la visualización y reduce el riesgo de sobreajuste.

Este reporte describe tres de los algoritmos de agrupación más utilizados: K-means, DBSCAN y Gaussian Mixture Models, así como dos técnicas relevantes de reducción de dimensionalidad: PCA y Autoencoders. Se analizarán sus principios, parámetros, ventajas, limitaciones y se incluirán ejemplos ilustrativos para facilitar su comprensión.

Algoritmos de agrupación

1. K-means

Principio de funcionamiento:
K-means divide los datos en *k* grupos (clusters) al minimizar la distancia intra-cluster entre los puntos y sus respectivos centroides. Inicialmente, selecciona *k* centroides aleatorios, asigna cada punto al centro más cercano y recalcula los centroides. Este proceso se repite hasta que no hay cambios significativos en las asignaciones.

Parámetros clave:

- *k*: número de clusters.
- Métrica de distancia (generalmente euclidiana).
- Número máximo de iteraciones.

Ventajas:

- Rápido y fácil de implementar.

- Escalable a grandes conjuntos de datos.

Limitaciones:

- Requiere especificar *k* previamente.
- Sensible a valores atípicos.
- Supone clusters esféricos y de tamaño similar.

Ejemplo (pseudocódigo):

1. Inicializar k centroides aleatorios
2. Repetir:
   a. Asignar cada punto al centroide más cercano
   b. Recalcular centroides

Hasta que las asignaciones no cambien

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

Principio de funcionamiento:
DBSCAN agrupa puntos densamente conectados. Identifica regiones de alta densidad (clusters) separadas por regiones de baja densidad (ruido). Define puntos como: núcleo (core), frontera o ruido, con base en su densidad local.

Parámetros clave:

- *eps*: radio de vecindad.
- *minPts*: número mínimo de puntos en una vecindad para formar un cluster.

Ventajas:

- No necesita especificar el número de clusters.

- Detecta clusters de formas arbitrarias.
- Maneja bien el ruido y outliers.

Limitaciones:

- Difícil ajustar parámetros óptimos.
- Bajo rendimiento en datos con densidades muy variables.

Ejemplo de aplicación:

Imaginemos un conjunto de datos geoespaciales. DBSCAN puede identificar aglomeraciones urbanas como clusters, y zonas despobladas como ruido.

3. Gaussian Mixture Models (GMM)

Principio de funcionamiento:

GMM asume que los datos provienen de una combinación de distribuciones gaussianas. A través del algoritmo Expectation-Maximization (EM), estima la probabilidad de que cada punto pertenezca a cada cluster.

Parámetros clave:

- Número de componentes gaussianos.
- Media y covarianza de cada componente.
- Peso de cada distribución.

Ventajas:

- Capta mejor la forma elíptica de los clusters.
- Proporciona probabilidades de pertenencia.

Limitaciones:

- Mayor complejidad computacional.
- Puede converger a óptimos locales.
- Requiere que los datos se ajusten bien a distribuciones gaussianas.

Ejemplo gráfico:
En una nube de datos bidimensionales con forma elíptica, GMM ajusta cada componente gaussiano a una subregión de los datos, permitiendo una segmentación más natural que K-means.

Algoritmos de reducción de dimensionalidad

1. Análisis de Componentes Principales (PCA)

Fundamento conceptual:
PCA transforma los datos originales en un nuevo conjunto de variables ortogonales llamadas componentes principales. Estas capturan la máxima varianza de los datos, ordenadas de mayor a menor contribución.

Parámetros clave:

- Número de componentes a conservar.
- Centrado de datos.

Ventajas:

- Muy útil para visualización.
- Reduce ruido y mejora eficiencia.
- Conserva la mayor información posible con menos dimensiones.

Limitaciones:

- Lineal: no captura relaciones no lineales.

- Los componentes no son fácilmente interpretables.

Ejemplo (pseudocódigo):

1. Centrar los datos
2. Calcular la matriz de covarianza
3. Obtener los eigenvectores y valores propios
4. Seleccionar los principales componentes
5. Proyectar los datos en las nuevas dimensiones

2. Autoencoders

Fundamento conceptual:
Los autoencoders son redes neuronales que aprenden a comprimir y descomprimir datos. Su capa oculta central (bottleneck) representa una versión de baja dimensionalidad del conjunto original.

Parámetros clave:

- Número de neuronas en la capa oculta.
- Función de activación.
- Optimización del error de reconstrucción.

Ventajas:

- Captura relaciones no lineales complejas.
- Flexible y extensible a grandes volúmenes de datos.

Limitaciones:

- Requiere más recursos y tiempo de entrenamiento.

- Puede sobreajustarse si no se regula correctamente.

Ejemplo ilustrativo:
Una imagen de 28x28 píxeles puede comprimirse a un vector de 32 dimensiones, y luego reconstruirse con mínima pérdida, conservando las características visuales más importantes.

Comparativa y conclusiones

Clustering vs. Reducción de Dimensionalidad

Criterio	**Clustering**	**Reducción de Dimensionalidad**
Objetivo	Agrupar datos similares	Simplificar representación de datos
Supervisión	supervisado	
No supervisado	No	
Salida	Etiquetas de cluster	Nuevas variables o dimensiones
Aplicación	Segmentación, análisis exploratorio	Preprocesamiento, visualización, mejora de modelos

Situaciones prácticas:

- En análisis de clientes, se puede usar reducción de dimensionalidad (como PCA) antes de aplicar clustering (como K-means), para reducir el ruido y acelerar el agrupamiento.
- Si se desea visualizar datos de alta dimensión en 2D o 3D, primero se aplica PCA o t-SNE.

- En casos donde se desconoce la estructura de los datos, DBSCAN puede revelar agrupaciones naturales sin requerir el número de clusters.

Conclusiones

Tanto los algoritmos de clustering como los de reducción de dimensionalidad son herramientas esenciales en la minería de datos moderna. K-means es eficiente y simple, pero limitado en flexibilidad, mientras que DBSCAN y GMM ofrecen soluciones más adaptativas y probabilísticas. En cuanto a la reducción de dimensionalidad, PCA ofrece una solución lineal efectiva y rápida, mientras que los autoencoders permiten capturar estructuras más complejas gracias al poder de las redes neuronales. La elección entre estos algoritmos depende del tipo de datos, sus dimensiones y los objetivos específicos del análisis.

Referencias

- Aggarwal, C. C. (2015). *Data Mining: The Textbook*. Springer. ● Bishop, C. M. (2006). *Pattern Recognition and Machine Learning*. Springer.
- Géron, A. (2019). *Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow*. O'Reilly Media.
- Scikit-learn documentation. (n.d.). *Clustering and Dimensionality Reduction*. https://scikit-learn.org/stable/user_guide.html
- Chollet, F. (2018). *Deep Learning with Python*. Manning Publications.