UNIVERSIDAD TECNOLÓGICA DE CHIHUAHUA

Tecnologías de la información

Extracción de Conocimiento en Bases de Datos

Algoritmos de agrupación

IDGS91N

PROFESOR:
Enrique Mascote

Alumno:
Emanuel Chavira

29 de Noviembre de 2025

IV.1. Algoritmos de agrupación y reducción de dimensionalidad

Introducción

La extracción de conocimiento a partir de datos masivos exige técnicas que permitan simplificar y revelar patrones ocultos. Dos enfoques complementarios facilitan este objetivo: **clustering** y **reducción de dimensionalidad**. El *clustering* (agrupación) es un método de aprendizaje no supervisado que identifica similitudes entre observaciones y las agrupa sin necesidad de etiquetas. Esta técnica resulta útil para detectar patrones en conjuntos de datos grandes y desestructurados. La **reducción de dimensionalidad**, por su parte, transforma o selecciona variables para preservar la información esencial, reduciendo el número de dimensiones. Al disminuir la redundancia y el ruido, estas técnicas facilitan la visualización de los datos, mejoran la eficiencia de los algoritmos posteriores y mitigan la maldición de la dimensionalidad, permitiendo que los modelos generalicen mejor.

Algoritmos de agrupación

K-Means

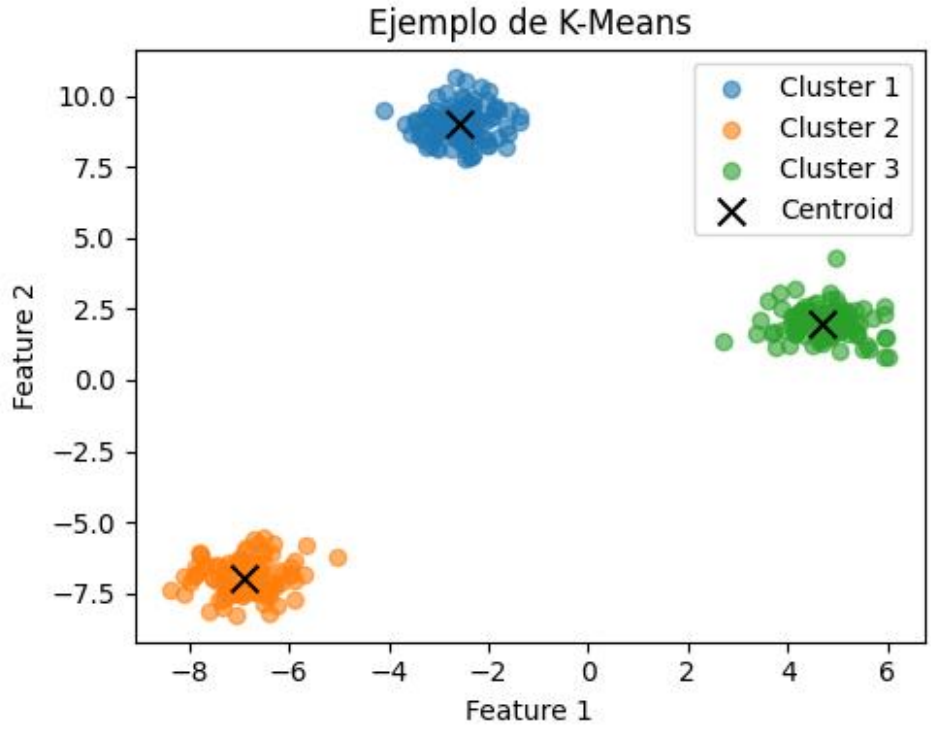
Principio de funcionamiento. K-means divide un conjunto de datos en k grupos que buscan maximizar la similitud interna y minimizar la similitud entre grupos. El algoritmo elige aleatoriamente k centroides iniciales, asigna cada observación al centroide más cercano y actualiza los centroides como la media de los puntos asignados. Estas dos etapas se repiten hasta que las asignaciones dejan de cambiar.

Parámetros clave. El número de clusters (k), la estrategia de inicialización (aleatoria o *k-means++*), el criterio de convergencia y la métrica de distancia (generalmente euclidiana).

Ventajas. Es simple de entender e implementar, escalable a grandes conjuntos de datos y eficiente en términos computacionales. Produce clusters compactos y funciona bien cuando las agrupaciones son aproximadamente esféricas.

Limitaciones. Requiere fijar k de antemano, es sensible a la selección inicial de los centroides y a los valores atípicos. Tiende a producir clusters de tamaño similar y no maneja bien agrupaciones no esféricas o de distinta densidad.

Ejemplo de aplicación. El siguiente diagrama muestra un conjunto de datos bidimensional dividido en tres grupos mediante K-means; los centroides aparecen marcados con “X”.



Ejemplo de K-Means

El algoritmo asigna cada punto al centroide más cercano y recalcula los centroides hasta que se estabilizan.

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

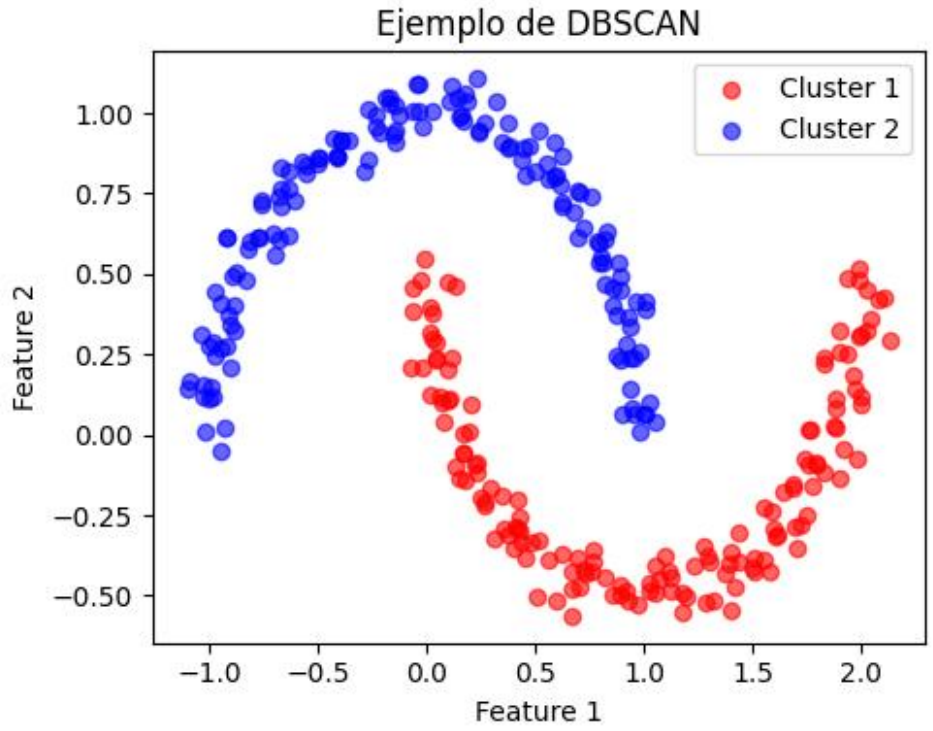
Principio de funcionamiento. DBSCAN se basa en la densidad. Define como **punto núcleo** aquel que posee al menos MinPts vecinos dentro de un radio eps. Los puntos alcanzables a partir de un núcleo forman un cluster; los puntos con menos vecinos son puntos *frontera*, y los que no cumplen estos criterios se consideran **ruido**. Los clusters se forman conectando puntos núcleo adyacentes sin necesidad de predefinir su número.

Parámetros clave. El radio de vecindad (eps) y el número mínimo de puntos necesarios para formar un núcleo (MinPts). La elección de la métrica de distancia también afecta el resultado.

Ventajas. No requiere especificar el número de clusters con antelación; identifica agrupaciones de forma arbitraria y detecta puntos atípicos como ruido. Es robusto frente a outliers y requiere sólo dos parámetros intuitivos.

Limitaciones. La selección de eps y MinPts puede ser complicada y sensible al dominio del problema. No se desempeña bien cuando existen clusters con densidades muy diferentes y puede producir resultados diferentes según el orden de exploración de los puntos.

Ejemplo de aplicación. A continuación se ilustra DBSCAN aplicado a un conjunto de datos con forma de "dos medias lunas". El algoritmo agrupa las regiones densas de datos y descarta el ruido.



Ejemplo de DBSCAN

Gaussian Mixture Models (GMM)

Principio de funcionamiento. Un modelo de mezcla gaussiana asume que los datos se generan a partir de la combinación de varias distribuciones normales multivariadas. A diferencia de K-means, que asigna cada observación de forma rígida, GMM realiza **agrupamiento suave**

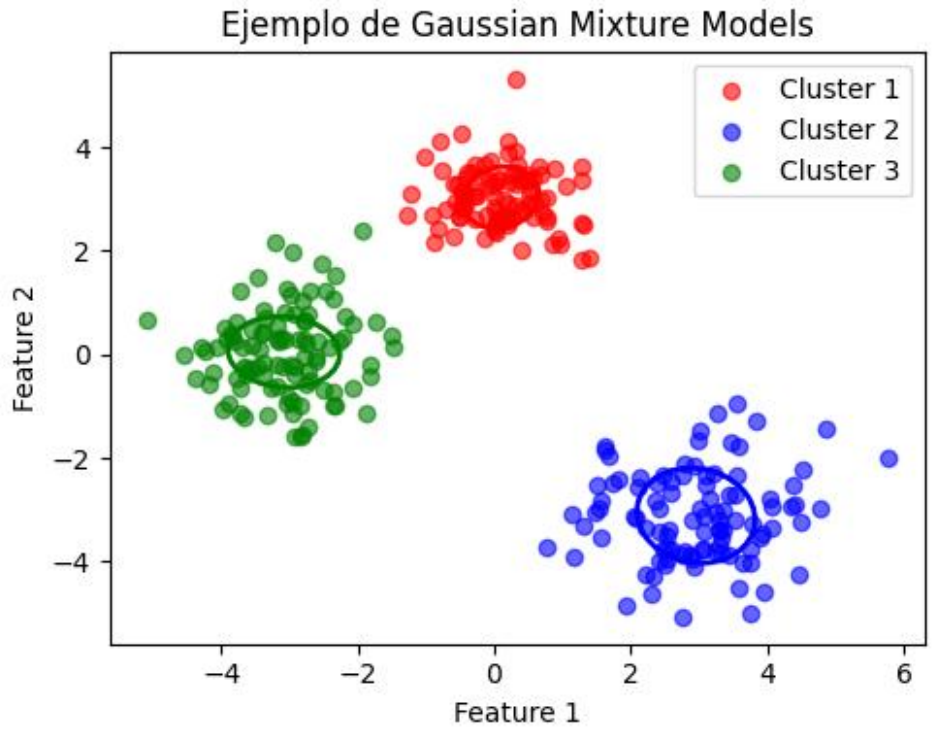
asignando a cada punto una probabilidad de pertenencia a cada componente. Se entrena mediante el algoritmo de Expectación-Maximización (EM): en la etapa de **expectación** se calculan las responsabilidades o probabilidades de pertenencia, y en la etapa de **maximización** se actualizan los parámetros de cada componente (medias, matrices de covarianza y pesos de mezcla).

Parámetros clave. Número de componentes, tipo de matriz de covarianza (completa, diagonal, esférica), valores iniciales de las medias y covarianzas, y criterios de convergencia del algoritmo EM.

Ventajas. Puede modelar clusters de forma elíptica y superpuestos gracias a las matrices de covarianza, lo que ofrece más flexibilidad que K-means. Proporciona probabilidades de pertenencia que permiten analizar la incertidumbre en la asignación de puntos.

Limitaciones. Requiere especificar el número de componentes y es sensible a la inicialización. El algoritmo EM puede converger a mínimos locales y su costo computacional aumenta con la dimensionalidad. Además, asume que los datos se distribuyen de forma aproximadamente gaussiana.

Ejemplo de aplicación. La figura siguiente muestra tres clusters modelados mediante mezclas gaussianas. Las elipses indican las covarianzas de los componentes y reflejan cómo GMM captura estructuras elípticas superpuestas.



Ejemplo de Gaussian Mixture Models

Algoritmos de reducción de dimensionalidad

Análisis de Componentes Principales (PCA)

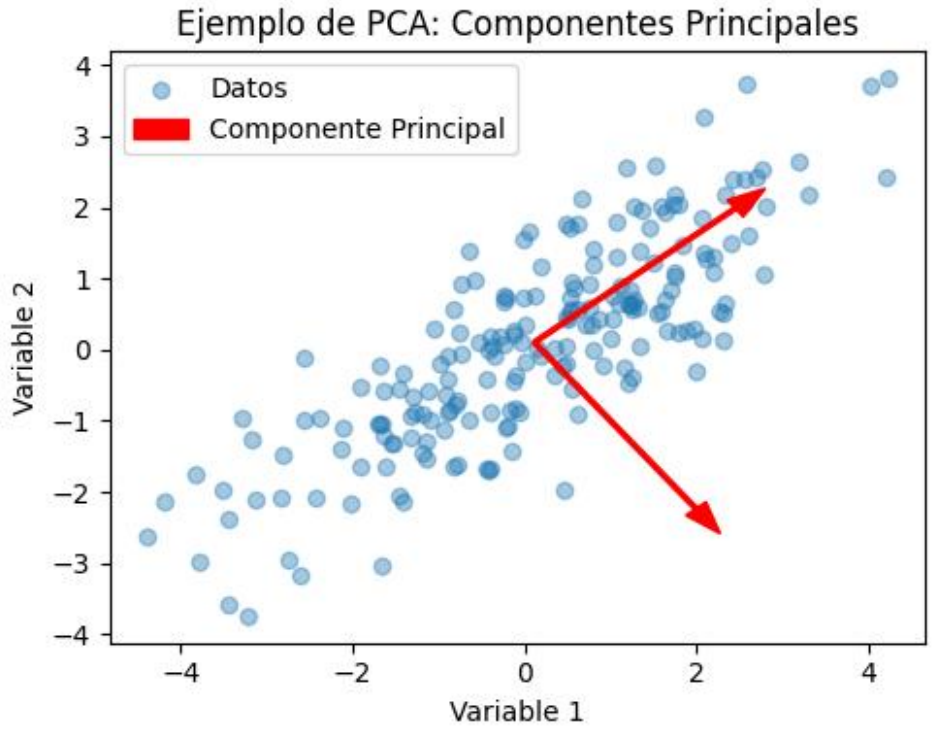
Fundamento conceptual. PCA es una técnica lineal que transforma un conjunto de variables correlacionadas en un nuevo conjunto de variables no correlacionadas denominadas **componentes principales**. Para ello estandariza los datos, calcula la matriz de covarianza y obtiene los vectores propios (direcciones) y valores propios (importancia) de dicha matriz. Los componentes principales se ordenan según la varianza que explican y el conjunto se proyecta sobre los primeros componentes para reducir dimensiones manteniendo la mayor variabilidad posible.

Parámetros clave. El número de componentes a retener, el método de escalado de las variables y, en ciertas implementaciones, el tipo de algoritmo de descomposición (por ejemplo, *svd* o *eigen*).

Ventajas. Maneja multicolinealidad creando variables no correlacionadas, reduce el ruido y comprime los datos al conservar sólo los componentes importantes, mejora la eficiencia de entrenamiento y permite detectar outliers en el espacio reducido.

Limitaciones. Los componentes son combinaciones lineales de las variables originales, lo que dificulta su interpretación; requiere escalar los datos; puede perder información si se mantienen pocos componentes; asume relaciones lineales y es sensible a valores atípicos. La reducción dimensional puede ser costosa en grandes conjuntos de datos.

Ejemplo ilustrativo. En la siguiente figura se representan los dos componentes principales obtenidos a partir de un conjunto de datos bivariado; los vectores rojos indican las direcciones de mayor varianza utilizadas para proyectar los datos.



Ejemplo de PCA

t-Distributed Stochastic Neighbor Embedding (t-SNE)

Fundamento conceptual. t-SNE es una técnica no lineal que transforma datos de alta dimensión en un espacio de dos o tres dimensiones conservando las distancias locales entre puntos. El

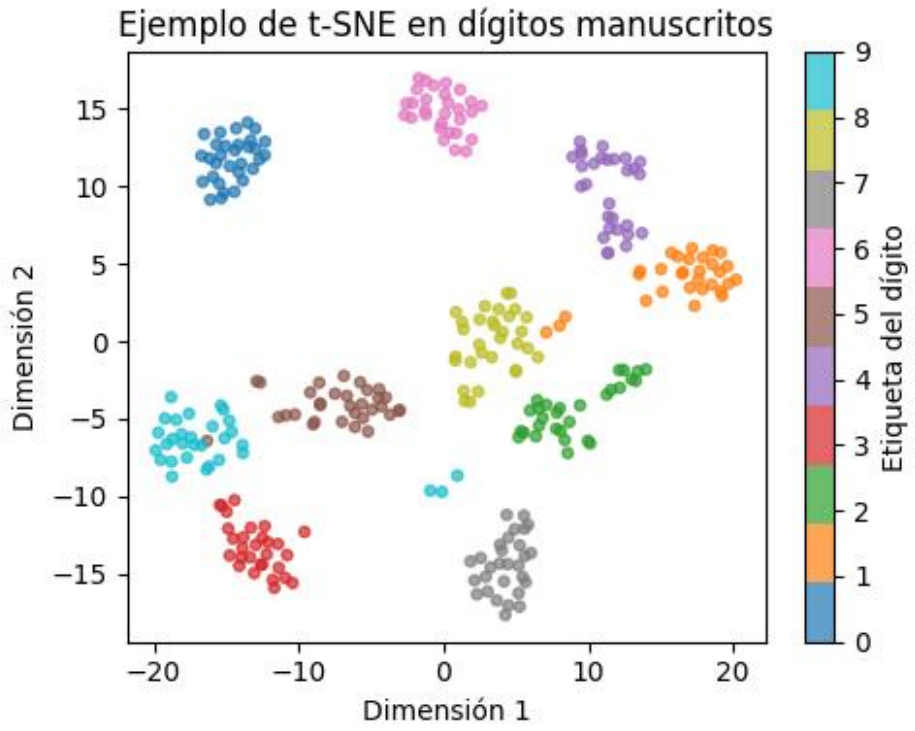
algoritmo calcula, en el espacio original, una distribución de probabilidades que mide la similitud entre pares de puntos usando un núcleo gaussiano. En el espacio reducido, define una distribución basada en la t de Student y optimiza las posiciones de los puntos minimizando la divergencia entre ambas distribuciones mediante descenso de gradiente. De este modo, puntos cercanos en alta dimensión permanecen juntos en la visualización, permitiendo discernir agrupaciones complejas.

Parámetros clave. El número de dimensiones de salida, la **perplexidad** (aproximadamente el número de vecinos considerados), la tasa de aprendizaje (learning rate), el número de iteraciones y el parámetro de aleatoriedad que afecta a la reproducibilidad.

Ventajas. Preserva la estructura local de los datos mejor que los métodos lineales y es idóneo para visualizar datos complejos y no lineales, como imágenes o texto. No necesita normalidad ni relaciones lineales.

Limitaciones. Elevada complejidad computacional; no produce resultados deterministas, por lo que ejecuciones sucesivas pueden diferir; la elección de parámetros afecta significativamente el resultado y no permite proyectar fácilmente nuevos puntos sin reentrenar el modelo.

Ejemplo ilustrativo. A continuación se muestra un mapa de dos dimensiones obtenido mediante t-SNE a partir de imágenes de dígitos manuscritos; cada color representa una clase distinta y las agrupaciones reflejan la similitud local.



Ejemplo de t-SNE en dígitos manuscritos

Comparativa y conclusiones

La siguiente tabla resume las principales diferencias entre las técnicas de **clustering** y de **reducción de dimensionalidad**:

Aspecto	Clustering	Reducción de dimensionalidad
Objetivo	Agrupar observaciones en subconjuntos homogéneos sin etiquetas.	Transformar o seleccionar variables para reducir el número de dimensiones preservando la información esencial.
Tipo de aprendizaje	No supervisado.	No supervisado (PCA, t-SNE) o supervisado (LDA).
Resultado	Asignación de cada punto a un cluster (duro o probabilístico).	Nueva representación de los datos en menor dimensión o subconjunto de características.
Uso principal	Descubrir patrones, segmentar	Visualización, compresión, mejora de

Aspecto	Clustering	Reducción de dimensionalidad
	clientes, detección de anomalías.	eficiencia y de rendimiento de modelos.
Interacción	Puede usarse después de reducción de dimensionalidad para mejorar resultados en datos de alta dimensión.	Suele aplicarse antes de algoritmos de clustering o clasificación para mejorar su rendimiento.

Situaciones prácticas

- **Cuando predomina la exploración de agrupaciones.** Si el objetivo es segmentar clientes, detectar grupos de comportamiento o identificar patrones en datos sin etiquetar, los algoritmos de clustering son prioritarios. En datos de alta dimensionalidad suele emplearse primero una técnica de reducción, como PCA, para eliminar ruido y acelerar el algoritmo de agrupación.
- **Cuando se requiere visualizar o preparar datos para modelos.** Si se buscan representaciones compactas para visualización, compresión o mejora de la generalización, las técnicas de reducción de dimensionalidad son prioritarias. Posteriormente pueden aplicarse algoritmos de agrupación o clasificación sobre el espacio reducido.

Conclusiones generales

El clustering y la reducción de dimensionalidad son herramientas esenciales en la extracción de conocimiento. K-means ofrece una solución simple y rápida para agrupar datos, aunque su desempeño depende del número de clusters y de la forma de los datos. DBSCAN permite identificar agrupaciones de forma arbitraria y distinguir ruido, pero su resultado depende de parámetros sensibles y densidades homogéneas. Las mezclas gaussianas proporcionan una aproximación probabilística más flexible, capaz de modelar clusters de forma elíptica, aunque con mayor complejidad y necesidad de especificar el número de componentes.

En reducción de dimensionalidad, PCA aporta un método lineal robusto para comprimir datos y eliminar redundancia, conservando la mayor varianza posible. t-SNE, en cambio, permite visualizar estructuras no lineales al mantener la similitud local, pero requiere más tiempo de cómputo y cuidados en la selección de parámetros. La elección de cada técnica depende del

problema: para explorar estructuras globales o preparar datos para algoritmos supervisados, PCA es una opción adecuada; para visualizar grupos intrincados en datos complejos, t-SNE revela patrones que métodos lineales no capturan.

Referencias

- Hartigan, J. A., & Wong, M. A. (1979). *Algorithm AS 136: A K-Means Clustering Algorithm*. Applied Statistics, 28(1), 100-108.
- Ester, M., Kriegel, H.-P., Sander, J., & Xu, X. (1996). *A density-based algorithm for discovering clusters in large spatial databases with noise*. In *Proceedings of the Second International Conference on Knowledge Discovery and Data Mining* (pp. 226-231).
- Reynolds, D. A. (2009). *Gaussian Mixture Models*. Encyclopedia of Biometrics, 659-663.
- Jolliffe, I. T., & Cadima, J. (2016). *Principal component analysis: A review and recent developments*. Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences, 374(2065), 20150202.
- van der Maaten, L., & Hinton, G. (2008). *Visualizing data using t-SNE*. Journal of Machine Learning Research, 9(Nov), 2579-2605.