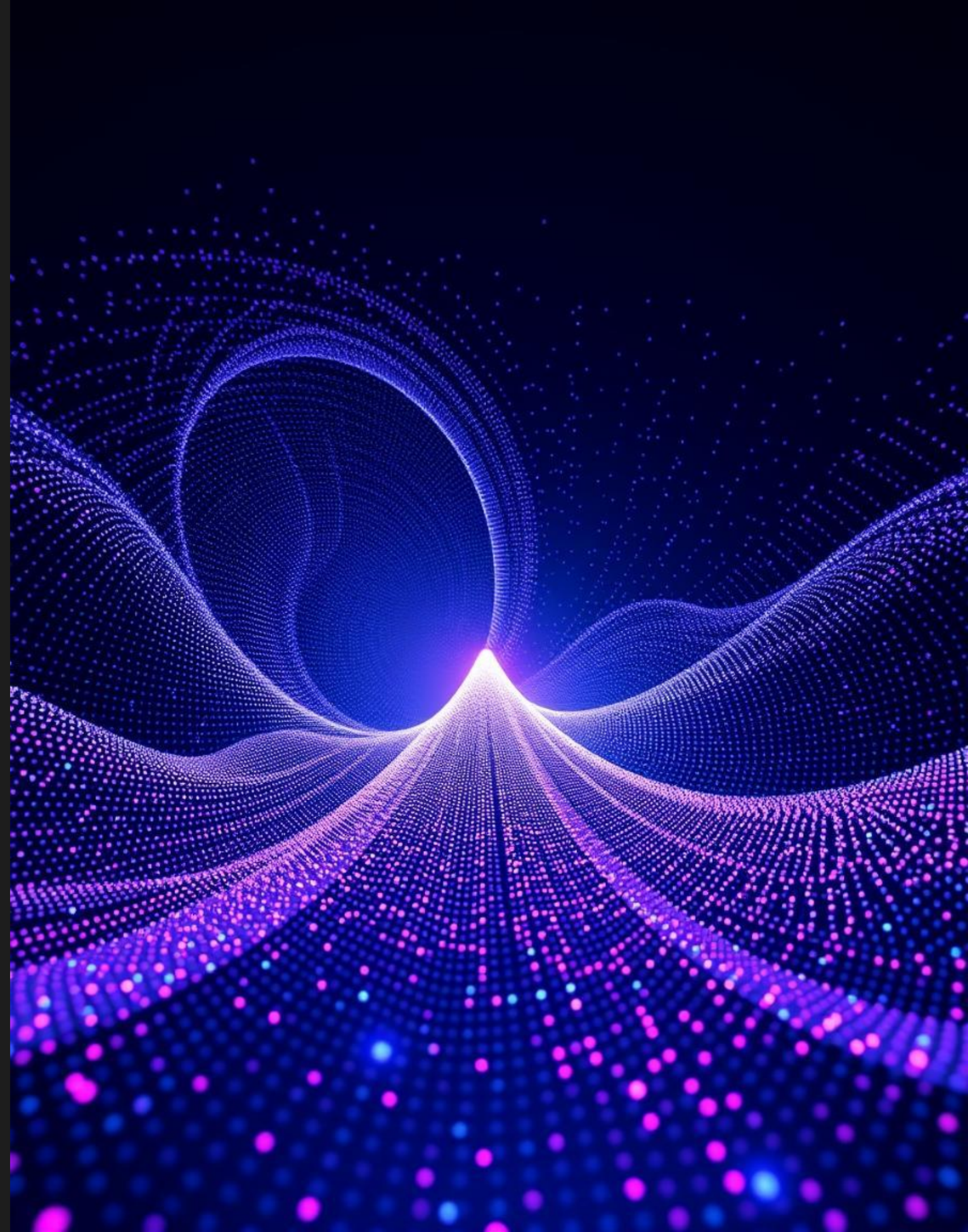
Aprendizaje de Máquina

No Supervisado

Sesión 5

Técnicas de Reducción Dimensional: PCA y t-SNE

En el análisis de datos, los conjuntos de datos de alta dimensión presentan desafíos significativos para la visualización y el procesamiento. Las técnicas de reducción de dimensionalidad, como PCA y t-SNE, ofrecen soluciones efectivas al permitir la representación de datos en espacios de menor dimensión, conservando al mismo tiempo su estructura y relaciones más relevantes.



PCA: Análisis de Componentes Principales

¿Qué es PCA?

PCA es un método estadístico para reducir la dimensionalidad de un conjunto de datos, conservando la mayor cantidad de información posible. Transforma un conjunto de datos con muchas variables en un conjunto más pequeño que aún captura la mayor parte de la variabilidad de los datos originales.

Componentes Principales

PCA encuentra nuevas variables llamadas componentes principales, que son combinaciones lineales de las variables originales. Estas componentes son ortogonales entre sí y se ordenan por importancia, capturando la mayor parte de la variabilidad de los datos.

Aplicaciones de PCA

Reconocimiento Facial

Reducción de datos en imágenes y extracción de características relevantes.

Compresión de Imágenes

Almacenamiento de imágenes con menos información sin perder demasiada calidad.

Análisis de Datos Financieros

Reducción del número de indicadores usados en modelos de inversión o análisis de riesgo.

Ventajas y Desventajas de PCA

Ventajas	Desventajas
Reduce la dimensionalidad de los datos, facilitando el procesamiento y almacenamiento.	Puede ser difícil interpretar los componentes principales, ya que son combinaciones de variables originales.
Elimina la multicolinealidad al generar componentes ortogonales.	Al reducir la dimensionalidad, se pierde algo de información.
Puede mejorar el rendimiento de algoritmos de aprendizaje automático.	Supone que las relaciones entre las variables son lineales, lo que no siempre es cierto.
Facilita la visualización de datos cuando se reduce a 2D o 3D.	

Implementación en Python: Reducción de Dimensionalidad en Datos de Imágenes de Dígitos

Para aplicar PCA en un caso práctico, usaremos el conjunto de datos *Digits* de Scikit-learn, que contiene imágenes de números escritos a mano (del 0 al 9) con dimensiones de 8x8 píxeles (64 características por imagen).

Pasos que seguiremos en Python

1. Cargar el **dataset Digits** de **Scikit-learn**.
2. Visualizar algunas imágenes originales.
3. Aplicar PCA para reducir la dimensionalidad de 64 a 2 dimensiones.
4. Visualizar los datos reducidos en un gráfico de dispersión.
5. Evaluar la cantidad de varianza retenida.

El detalle de la actividad se encuentra en la guía de estudio de la sesión.

T-SNE: T-Distributed Stochastic Neighbor Embedding

¿Qué es t-SNE?

t-SNE es un método de reducción de dimensionalidad no lineal utilizado principalmente para la visualización de datos de alta dimensión en espacios de 2D o 3D. Es especialmente útil cuando los datos tienen estructuras complejas que los métodos lineales (como PCA) no pueden captar.

¿Cómo Funciona?

t-SNE está diseñado para preservar la relación de proximidad entre puntos similares. Calcula la probabilidad de que dos puntos sean vecinos en el espacio original y luego optimiza para minimizar la divergencia entre ambas distribuciones en el espacio reducido.

Aplicaciones de t-SNE

Análisis de Datos Genómicos

Identificación de relaciones entre distintos tipos de células o genes.

Procesamiento de Imágenes

Visualización de la organización de imágenes en función de características aprendidas.

Procesamiento de Texto (NLP)

Visualización de embeddings de palabras como word2vec o BERT.

Ventajas y Desventajas de t-SNE

Ventajas	Desventajas
Es muy eficaz para encontrar patrones y estructuras ocultas en datos de alta dimensionalidad.	Es computacionalmente costoso en comparación con otros métodos como PCA.
Mantiene relaciones locales entre los puntos, lo que facilita la agrupación de datos similares.	Puede ser difícil de interpretar, ya que la escala de los ejes en t-SNE no tiene un significado claro.
Es muy útil para visualizar datos en 2D o 3D sin perder demasiada información.	Los resultados pueden variar en cada ejecución debido a su inicialización aleatoria.
	No es adecuado para reducción de dimensionalidad antes de aplicar modelos de aprendizaje automático.

Implementación en Python: Visualización de datos de dígitos con t-SNE

Vamos a aplicar t-SNE al mismo conjunto de datos ***Digits*** de **Scikit-learn** para visualizar cómo se agrupan los números escritos a mano.

Pasos que seguiremos en Python

1. Cargar el dataset Digits.
2. Estandarizar los datos (para mejorar el rendimiento de t-SNE).
3. Aplicar t-SNE para reducir la dimensionalidad a 2D.
4. Visualizar los datos reducidos en un gráfico de dispersión.

El detalle de la actividad se encuentra en la guía de estudio de la sesión.

Preguntas

Sección de preguntas

Aprendizaje de Máquina

No Supervisado

Continúe con las actividades