UTEC Posgrado

UTEC Posgrado

UTEC Posgrado

Una escuela para
Early Adopters

REDUCCIÓN DE DIMENSIONALIDAD
PCA, t-SNE Y UMAP

Objetivo de la sesión

¿Qué aprenderemos hoy?

Comprender y aplicar técnicas de reducción de dimensionalidad (PCA, t-SNE y UMAP) para transformar datos de alta dimensión en representaciones visualizables, seleccionando la técnica más apropiada según las características del dataset y los objetivos del análisis.

Enfoque práctico: De la teoría a la aplicación en casos reales

Motivación

El problema de alta dimensionalidad **Por qué necesitamos reducir dimensiones**

El problema: Demasiadas dimensiones

Situación típica: Tenemos datos con muchas variables (features)

- Imagen 100×100 pixels $=$ **10,000 dimensiones**

El problema: Demasiadas dimensiones

Situación típica: Tenemos datos con muchas variables (features)

- Imagen 100×100 pixels = **10,000 dimensiones**
- Dataset de ventas con 50 variables = **50 dimensiones**

El problema: Demasiadas dimensiones

Situación típica: Tenemos datos con muchas variables (features)

- Imagen 100×100 pixels = **10,000 dimensiones**
- Dataset de ventas con 50 variables = **50 dimensiones**
- Análisis genómico = **miles de dimensiones**

El problema: Demasiadas dimensiones

Situación típica: Tenemos datos con muchas variables (features)

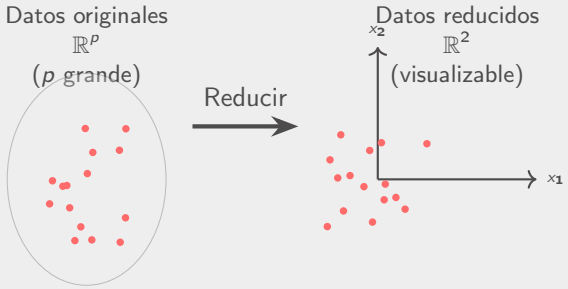
- Imagen 100×100 pixels = **10,000 dimensiones**
- Dataset de ventas con 50 variables = **50 dimensiones**
- Análisis genómico = **miles de dimensiones**

Problema: No podemos visualizar ni entender datos en alta dimensión

- Cerebro humano: máximo 3 dimensiones
- Computadoras: lentas con alta dimensionalidad
- Patrones ocultos en la complejidad

¿Qué es la reducción de dimensionalidad?

Idea central: Transformar datos de **alta dimensión** (p grande) a **baja dimensión** (2-3D) preservando información importante.



Objetivo: Perder la menor cantidad de información posible

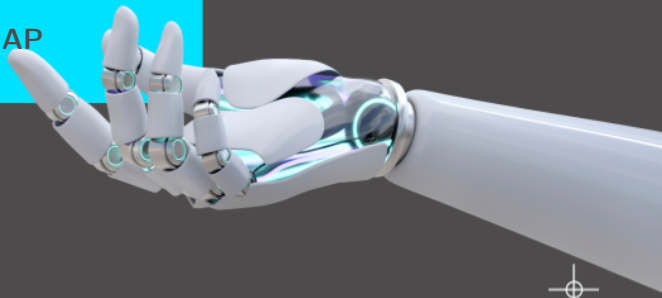
Las tres técnicas principales

Técnica	Tipo	Velocidad	Mejor para
PCA	Lineal	Muy rápida	Estructura global
t-SNE	No lineal	Lenta	Visualizar clusters
UMAP	No lineal	Rápida	Balance global/local

Cada una tiene **ventajas y desventajas** según el caso de uso.

Desarrollo

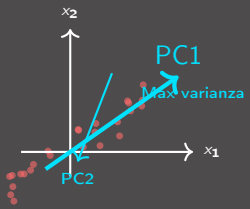
Las tres técnicas **PCA, t-SNE y UMAP**



1. PCA: Principal Component Analysis

Idea simple: Encuentra las **direcciones de máxima variación**

Analogía: Fotografiar un edificio

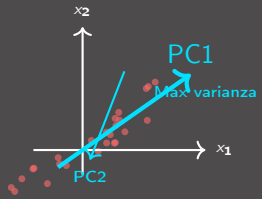


1. PCA: Principal Component Analysis

Idea simple: Encuentra las **direcciones de máxima variación**

Analogía: Fotografiar un edificio

- Hay infinitos ángulos posibles

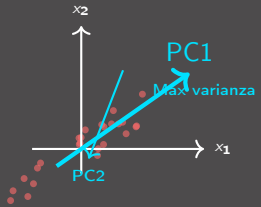


1. PCA: Principal Component Analysis

Idea simple: Encuentra las **direcciones de máxima variación**

Analogía: Fotografiar un edificio

- Hay infinitos ángulos posibles
- Algunos capturan más información

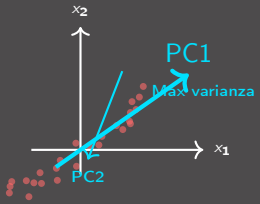


1. PCA: Principal Component Analysis

Idea simple: Encuentra las **direcciones de máxima variación**

Analogía: Fotografiar un edificio

- Hay infinitos ángulos posibles
- Algunos capturan más información
- PCA encuentra el **mejor ángulo**

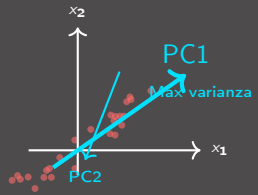


1. PCA: Principal Component Analysis

Idea simple: Encuentra las **direcciones de máxima variación**

Analogía: Fotografiar un edificio

- Hay infinitos ángulos posibles
- Algunos capturan más información
- PCA encuentra el **mejor ángulo**
- Siempre usa proyecciones **lineales**



Cómo funciona PCA

Proceso paso a paso:

1. **Centrar los datos**: restar la media a cada variable

Cómo funciona PCA

Proceso paso a paso:

1. **Centrar los datos**: restar la media a cada variable
2. **Calcular covarianza**: qué variables están correlacionadas

Cómo funciona PCA

Proceso paso a paso:

1. **Centrar los datos**: restar la media a cada variable
2. **Calcular covarianza**: qué variables están correlacionadas
3. **Encontrar direcciones principales**: autovectores de la matriz de covarianza

Cómo funciona PCA

Proceso paso a paso:

1. **Centrar los datos**: restar la media a cada variable
2. **Calcular covarianza**: qué variables están correlacionadas
3. **Encontrar direcciones principales**: autovectores de la matriz de covarianza
4. **Proyectar**: transformar datos a las nuevas direcciones

Cómo funciona PCA

Proceso paso a paso:

1. **Centrar los datos**: restar la media a cada variable
2. **Calcular covarianza**: qué variables están correlacionadas
3. **Encontrar direcciones principales**: autovectores de la matriz de covarianza
4. **Proyectar**: transformar datos a las nuevas direcciones
5. **Retener componentes**: quedarse con los primeros 2-3 que capturan más varianza

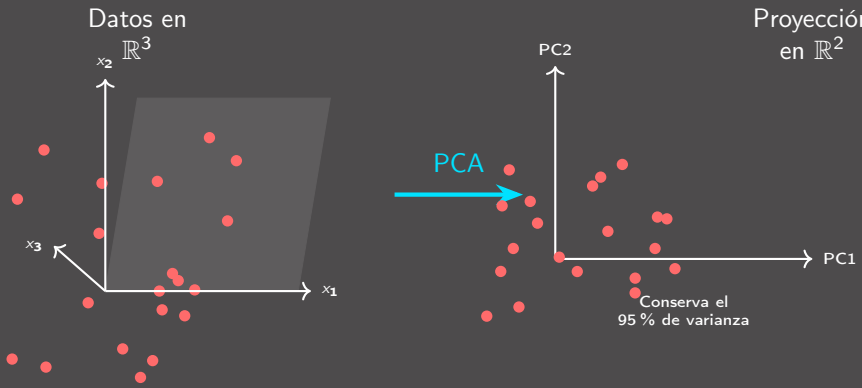
Cómo funciona PCA

Proceso paso a paso:

1. **Centrar los datos**: restar la media a cada variable
2. **Calcular covarianza**: qué variables están correlacionadas
3. **Encontrar direcciones principales**: autovectores de la matriz de covarianza
4. **Proyectar**: transformar datos a las nuevas direcciones
5. **Retener componentes**: quedarse con los primeros 2-3 que capturan más varianza

Característica clave: Es **determinista** (siempre da el mismo resultado)

PCA: Visión geométrica



¿Cuándo usar PCA?

Casos ideales:

- **Datos linealmente correlacionados**
  - Ejemplo: indicadores financieros (precio, volumen, volatilidad)

¿Cuándo usar PCA?

Casos ideales:

- **Datos linealmente correlacionados**
  - Ejemplo: indicadores financieros (precio, volumen, volatilidad)
- **Necesitas velocidad**
  - Millones de puntos, cientos de variables

¿Cuándo usar PCA?

Casos ideales:

- **Datos linealmente correlacionados**
  - Ejemplo: indicadores financieros (precio, volumen, volatilidad)
- **Necesitas velocidad**
  - Millones de puntos, cientos de variables
- **Quieres interpretabilidad**
  - Cada componente tiene significado: “factor de crecimiento”, “factor de riesgo”

¿Cuándo usar PCA?

Casos ideales:

- **Datos linealmente correlacionados**
  - Ejemplo: indicadores financieros (precio, volumen, volatilidad)
- **Necesitas velocidad**
  - Millones de puntos, cientos de variables
- **Quieres interpretabilidad**
  - Cada componente tiene significado: "factor de crecimiento", "factor de riesgo"
- **Necesitas reconstruir datos**
  - Compresión de imágenes, eliminación de ruido

¿Cuándo usar PCA?

Casos ideales:

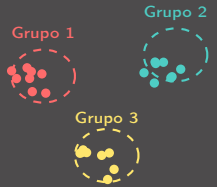
- **Datos linealmente correlacionados**
  - Ejemplo: indicadores financieros (precio, volumen, volatilidad)
- **Necesitas velocidad**
  - Millones de puntos, cientos de variables
- **Quieres interpretabilidad**
  - Cada componente tiene significado: "factor de crecimiento", "factor de riesgo"
- **Necesitas reconstruir datos**
  - Compresión de imágenes, eliminación de ruido

Limitación: No captura bien relaciones **no lineales**

2. t-SNE: t-Distributed Stochastic Neighbor Embedding

Idea simple: Agrupa puntos **similares** y separa puntos **diferentes**

Analogía: Organizar tu escritorio



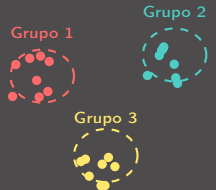
Clusters bien separados

2. t-SNE: t-Distributed Stochastic Neighbor Embedding

Idea simple: Agrupa puntos **similares** y separa puntos **diferentes**

Analogía: Organizar tu escritorio

- Tienes papeles dispersos



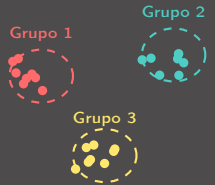
Clusters bien separados

2. t-SNE: t-Distributed Stochastic Neighbor Embedding

Idea simple: Agrupa puntos **similares** y separa puntos **diferentes**

Analogía: Organizar tu escritorio

- Tienes papeles dispersos
- Agrupas por tema: facturas juntas, recibos juntos



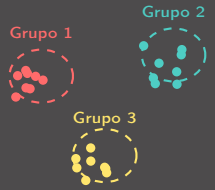
Clusters bien separados

2. t-SNE: t-Distributed Stochastic Neighbor Embedding

Idea simple: Agrupa puntos **similares** y separa puntos **diferentes**

Analogía: Organizar tu escritorio

- Tienes papeles dispersos
- Agrupas por tema: facturas juntas, recibos juntos
- Cada grupo **bien separado**



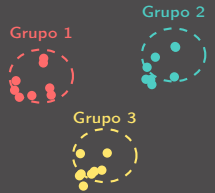
Clusters bien separados

2. t-SNE: t-Distributed Stochastic Neighbor Embedding

Idea simple: Agrupa puntos **similares** y separa puntos **diferentes**

Analogía: Organizar tu escritorio

- Tienes papeles dispersos
- Agrupas por tema: facturas juntas, recibos juntos
- Cada grupo **bien separado**
- No importa **dónde** quede cada grupo



Clusters bien separados

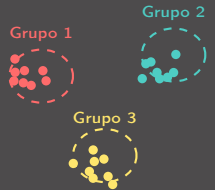
2. t-SNE: t-Distributed Stochastic Neighbor Embedding

Idea simple: Agrupa puntos **similares** y separa puntos **diferentes**

Analogía: Organizar tu escritorio

- Tienes papeles dispersos
- Agrupas por tema: facturas juntas, recibos juntos
- Cada grupo **bien separado**
- No importa **dónde** quede cada grupo

▷ Preserva **vecindad local**, no distancias globales



Clusters bien separados

Cómo funciona t-SNE

Proceso (simplificado):

1. **Medir similitudes en alta dimensión**
   - Para cada punto: ¿quiénes son sus vecinos cercanos?

Cómo funciona t-SNE

Proceso (simplificado):

1. **Medir similitudes en alta dimensión**
   - Para cada punto: ¿quiénes son sus vecinos cercanos?
2. **Inicializar aleatoriamente en 2D**
   - Puntos empiezan en posiciones al azar

Cómo funciona t-SNE

Proceso (simplificado):

1. **Medir similitudes en alta dimensión**
   - Para cada punto: ¿quiénes son sus vecinos cercanos?
2. **Inicializar aleatoriamente en 2D**
   - Puntos empiezan en posiciones al azar
3. **Mover puntos iterativamente**
   - Acercar vecinos que estaban cerca en alta dimensión
   - Alejar puntos que estaban lejos

Cómo funciona t-SNE

Proceso (simplificado):

1. **Medir similitudes en alta dimensión**
   - Para cada punto: ¿quiénes son sus vecinos cercanos?
2. **Inicializar aleatoriamente en 2D**
   - Puntos empiezan en posiciones al azar
3. **Mover puntos iterativamente**
   - Acercar vecinos que estaban cerca en alta dimensión
   - Alejar puntos que estaban lejos
4. **Iterar hasta convergencia**
   - Puede tomar cientos o miles de iteraciones

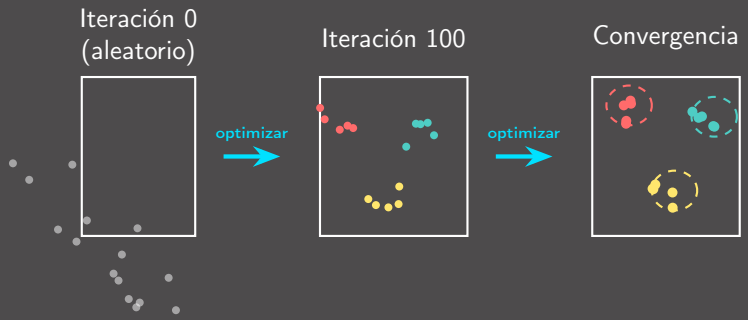
Cómo funciona t-SNE

Proceso (simplificado):

1. **Medir similitudes en alta dimensión**
   - Para cada punto: ¿quiénes son sus vecinos cercanos?
2. **Inicializar aleatoriamente en 2D**
   - Puntos empiezan en posiciones al azar
3. **Mover puntos iterativamente**
   - Acercar vecinos que estaban cerca en alta dimensión
   - Alejar puntos que estaban lejos
4. **Iterar hasta convergencia**
   - Puede tomar cientos o miles de iteraciones

Característica clave: Es **estocástico** (resultados pueden variar)

t-SNE: Proceso iterativo



¿Cuándo usar t-SNE?

Casos ideales:

- **Visualización exploratoria de clusters**
  - Ejemplo: ver grupos de clientes, tipos de productos

¿Cuándo usar t-SNE?

Casos ideales:

- **Visualización exploratoria de clusters**
  - Ejemplo: ver grupos de clientes, tipos de productos
- **Datasets pequeños/medianos** ($< 10{,}000$ puntos)
  - Con más datos se vuelve muy lento

¿Cuándo usar t-SNE?

Casos ideales:

- **Visualización exploratoria de clusters**
  - Ejemplo: ver grupos de clientes, tipos de productos
- **Datasets pequeños/medianos** ($< 10{,}000$ puntos)
  - Con más datos se vuelve muy lento
- **Estructuras no lineales complejas**
  - Datos de imágenes, texto, biología

¿Cuándo usar t-SNE?

Casos ideales:

- **Visualización exploratoria de clusters**
  - Ejemplo: ver grupos de clientes, tipos de productos
- **Datasets pequeños/medianos** ($< 10{,}000$ puntos)
  - Con más datos se vuelve muy lento
- **Estructuras no lineales complejas**
  - Datos de imágenes, texto, biología
- **Solo necesitas visualizar** (no usar los datos después)
  - No se puede aplicar a nuevos datos

¿Cuándo usar t-SNE?

Casos ideales:

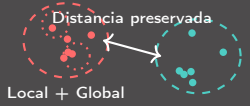
- **Visualización exploratoria de clusters**
  - Ejemplo: ver grupos de clientes, tipos de productos
- **Datasets pequeños/medianos** ($< 10{,}000$ puntos)
  - Con más datos se vuelve muy lento
- **Estructuras no lineales complejas**
  - Datos de imágenes, texto, biología
- **Solo necesitas visualizar** (no usar los datos después)
  - No se puede aplicar a nuevos datos

Limitaciones: Lento, no preserva distancias globales, no determinista

3. UMAP: Uniform Manifold Approximation & Projection

Idea simple: Preserva **estructura local Y global** simultáneamente

Analogía: Mapa del metro

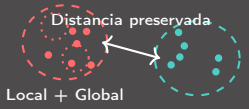


3. UMAP: Uniform Manifold Approximation & Projection

Idea simple: Preserva **estructura local Y global** simultáneamente

Analogía: Mapa del metro

- Simplifica la ciudad real

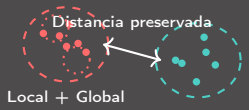


3. UMAP: Uniform Manifold Approximation & Projection

Idea simple: Preserva **estructura local Y global** simultáneamente

Analogía: Mapa del metro

- Simplifica la ciudad real
- Mantiene **estaciones cercanas** juntas (local)

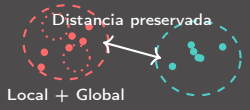


3. UMAP: Uniform Manifold Approximation & Projection

Idea simple: Preserva **estructura local Y global** simultáneamente

Analogía: Mapa del metro

- Simplifica la ciudad real
- Mantiene **estaciones cercanas** juntas (local)
- Muestra **qué líneas conectan** (global)



3. UMAP: Uniform Manifold Approximation & Projection

Idea simple: Preserva **estructura local Y global** simultáneamente

Analogía: Mapa del metro

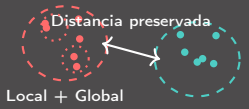
- Simplifica la ciudad real
- Mantiene **estaciones cercanas** juntas (local)
- Muestra **qué líneas conectan** (global)
- Es útil aunque no sea geográficamente exacto

3. UMAP: Uniform Manifold Approximation & Projection

Idea simple: Preserva **estructura local Y global** simultáneamente

Analogía: Mapa del metro

- Simplifica la ciudad real
- Mantiene **estaciones cercanas** juntas (local)
- Muestra **qué líneas conectan** (global)
- Es útil aunque no sea geográficamente exacto



▷ "Lo mejor de ambos mundos": velocidad de PCA + calidad de t-SNE

Cómo funciona UMAP

Proceso (simplificado):

1. **Construir grafo de vecindad en alta dimensión**
   - Conectar cada punto con sus k vecinos más cercanos

Cómo funciona UMAP

Proceso (simplificado):

1. **Construir grafo de vecindad en alta dimensión**
   - Conectar cada punto con sus k vecinos más cercanos
2. **Asignar pesos a las conexiones**
   - Vecinos muy cercanos: peso alto
   - Vecinos lejanos: peso bajo

Cómo funciona UMAP

Proceso (simplificado):

1. **Construir grafo de vecindad en alta dimensión**
   - Conectar cada punto con sus k vecinos más cercanos
2. **Asignar pesos a las conexiones**
   - Vecinos muy cercanos: peso alto
   - Vecinos lejanos: peso bajo
3. **Optimizar grafo en 2D**
   - Recrear estructura de vecindad en baja dimensión
   - Mantener conexiones fuertes

Cómo funciona UMAP

Proceso (simplificado):

1. **Construir grafo de vecindad en alta dimensión**
   - Conectar cada punto con sus k vecinos más cercanos
2. **Asignar pesos a las conexiones**
   - Vecinos muy cercanos: peso alto
   - Vecinos lejanos: peso bajo
3. **Optimizar grafo en 2D**
   - Recrear estructura de vecindad en baja dimensión
   - Mantener conexiones fuertes
4. **Balance automático**
   - Preserva tanto vecindad local como relaciones globales

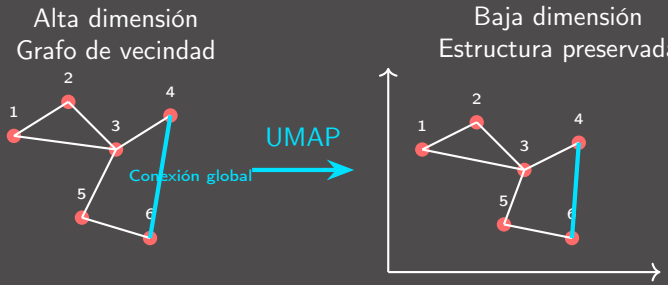
Cómo funciona UMAP

Proceso (simplificado):

1. **Construir grafo de vecindad en alta dimensión**
   - Conectar cada punto con sus k vecinos más cercanos
2. **Asignar pesos a las conexiones**
   - Vecinos muy cercanos: peso alto
   - Vecinos lejanos: peso bajo
3. **Optimizar grafo en 2D**
   - Recrear estructura de vecindad en baja dimensión
   - Mantener conexiones fuertes
4. **Balance automático**
   - Preserva tanto vecindad local como relaciones globales

Ventaja: Más rápido que t-SNE, mejor estructura que PCA

UMAP: Grafo de vecindad



¿Cuándo usar UMAP?

Casos ideales:

- **Datasets grandes** ($>$ 10,000 puntos)
  - Escala mejor que t-SNE

¿Cuándo usar UMAP?

Casos ideales:

- **Datasets grandes** ($>$ 10,000 puntos)
  - Escala mejor que t-SNE
- **Necesitas estructura local Y global**
  - Ejemplo: datos genómicos, redes sociales

¿Cuándo usar UMAP?

Casos ideales:

- **Datasets grandes** ($>$ 10,000 puntos)
  - Escala mejor que t-SNE
- **Necesitas estructura local Y global**
  - Ejemplo: datos genómicos, redes sociales
- **Quieres aplicar a nuevos datos**
  - UMAP puede transformar datos no vistos en entrenamiento

¿Cuándo usar UMAP?

Casos ideales:

- **Datasets grandes** ($>$ 10,000 puntos)
  - Escala mejor que t-SNE
- **Necesitas estructura local Y global**
  - Ejemplo: datos genómicos, redes sociales
- **Quieres aplicar a nuevos datos**
  - UMAP puede transformar datos no vistos en entrenamiento
- **Buscas balance velocidad/calidad**
  - Más rápido que t-SNE, mejor que PCA para datos no lineales

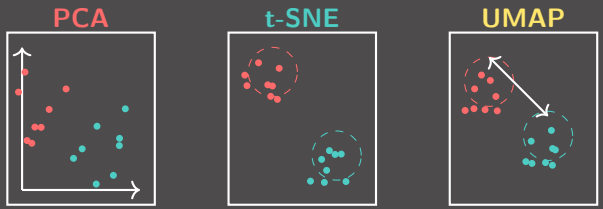
¿Cuándo usar UMAP?

Casos ideales:

- **Datasets grandes** ($>$ 10,000 puntos)
  - Escala mejor que t-SNE
- **Necesitas estructura local Y global**
  - Ejemplo: datos genómicos, redes sociales
- **Quieres aplicar a nuevos datos**
  - UMAP puede transformar datos no vistos en entrenamiento
- **Buscas balance velocidad/calidad**
  - Más rápido que t-SNE, mejor que PCA para datos no lineales

Ventaja principal: Versatilidad - funciona bien en muchos casos

Comparación visual: Mismo dataset



Comparación detallada

Aspecto	PCA	t-SNE	UMAP
Tipo	Lineal	No lineal	No lineal
Velocidad	Muy rápida	Lenta	Rápida
Preserva local	No	Sí	Sí
Preserva global	Sí	No	Sí
Determinista	Sí	No	No
Nuevos datos	Sí	No	Sí
Interpretable	Sí	No	Parcial
Escalabilidad	Excelente	Pobre	Buena

Aclaración importante: Aplicación a nuevos datos

¿Qué técnica se asemeja a PCA?

Pregunta clave: ¿Puedo aplicar el modelo a datos nuevos sin reentrenar?

Técnica	Nuevos datos?	Cómo funciona
PCA	SÍ	Aplica componentes aprendidas
t-SNE	NO	Debe reentrenar todo
UMAP	SÍ	Aplica modelo aprendido

Aclaración importante: Aplicación a nuevos datos

¿Qué técnica se asemeja a PCA?

Pregunta clave: ¿Puedo aplicar el modelo a datos nuevos sin reentrenar?

Técnica	Nuevos datos?	Cómo funciona
PCA	SÍ	Aplica componentes aprendidas
t-SNE	NO	Debe reentrenar todo
UMAP	SÍ	Aplica modelo aprendido

Respuesta: UMAP se asemeja a PCA en este aspecto.

Ambas pueden usar `transform()` para aplicar el modelo a nuevos datos.
t-SNE requiere re-ejecutar `fit_transform()` con todos los datos.

Aclaración importante: Aplicación a nuevos datos

¿Qué técnica se asemeja a PCA?

Pregunta clave: ¿Puedo aplicar el modelo a datos nuevos sin reentrenar?

Técnica	Nuevos datos?	Cómo funciona
PCA	SÍ	Aplica componentes aprendidas
t-SNE	NO	Debe reentrenar todo
UMAP	SÍ	Aplica modelo aprendido

Respuesta: UMAP se asemeja a PCA en este aspecto.

Ambas pueden usar `transform()` para aplicar el modelo a nuevos datos.
t-SNE requiere re-ejecutar `fit_transform()` con todos los datos.

Implicación práctica: Para sistemas en **producción** (ej: streaming de datos), usa PCA o UMAP, no t-SNE.

Ejemplos de código: Nuevos datos

PCA - Puede aplicarse a nuevos datos:

```
pca = PCA(n_components=2)
X_train_pca = pca.fit_transform(X_train)
X_test_pca = pca.transform(X_test) ✓
```

t-SNE - NO puede aplicarse a nuevos datos:

```
tsne = TSNE(n_components=2)
X_train_tsne = tsne.fit_transform(X_train)
# X_test_tsne = tsne.transform(X_test) × ERROR
```

UMAP - Puede aplicarse a nuevos datos:

```
reducer = umap.UMAP(n_components=2)
X_train_umap = reducer.fit_transform(X_train)
X_test_umap = reducer.transform(X_test) ✓
```

Conclusión: UMAP y PCA comparten esta ventaja clave sobre t-SNE

Estructura Local vs Global: Comparación

¿Qué estructura preserva cada técnica?

Técnica	Local	Global	Aplicaciones ideales
PCA	No	Sí	• Análisis exploratorio inicial • Compresión de datos • Identificar tendencias generales • Features para ML
t-SNE	Sí	No	• Descubrir subgrupos ocultos • Segmentación de clientes • Visualización de embeddings • Detección de outliers locales
UMAP	Sí	Sí	• Estructuras jerárquicas • Clasificación de tipos celulares • Análisis de redes sociales • Datasets con múltiples escalas

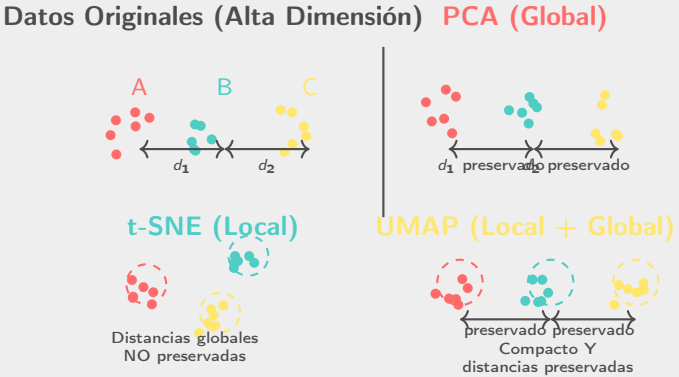
Estructura Local:

- Vecinos cercanos permanecen juntos
- Clusters compactos bien definidos

Estructura Global:

- Distancias relativas entre grupos
- Relaciones jerárquicas

Visualización: Local vs Global



Escenarios: ¿Cuál usar?

Escenario	Técnica recomendada
Datos financieros correlacionados	PCA - Rápido, interpretable
Visualizar segmentación de clientes	t-SNE - Clusters claros
Análisis genómico (millones de células)	UMAP - Escala bien
Compresión de imágenes	PCA/SVD - Reversible
Explorar dataset nuevo (desconocido)	PCA primero - Rápido
Preparar datos para ML	PCA - Nuevos datos
Visualizar embeddings de texto	t-SNE o UMAP
Datos con jerarquía multinivel	UMAP - Local + global

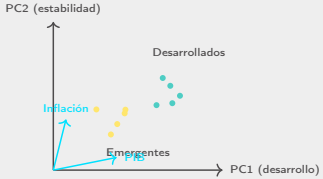
Caso 1: PCA es superior

Situación: Indicadores económicos de países

Variables:

- PIB per cápita
- Inflación
- Desempleo
- Deuda pública
- Crecimiento

Por qué PCA?



Caso 1: PCA es superior

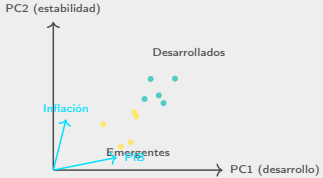
Situación: Indicadores económicos de países

Variables:

- PIB per cápita
- Inflación
- Desempleo
- Deuda pública
- Crecimiento

Por qué PCA?

- Variables **linealmente correlacionadas**



Caso 1: PCA es superior

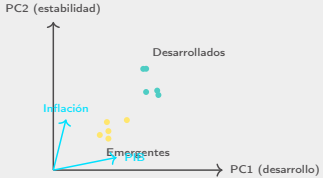
Situación: Indicadores económicos de países

Variables:

- PIB per cápita
- Inflación
- Desempleo
- Deuda pública
- Crecimiento

Por qué PCA?

- Variables **linealmente correlacionadas**
- Interpretación: PC1 = "desarrollo", PC2 = "estabilidad"



Caso 1: PCA es superior

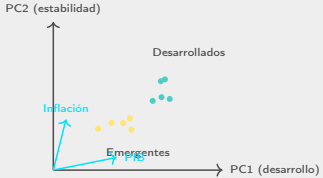
Situación: Indicadores económicos de países

Variables:

- PIB per cápita
- Inflación
- Desempleo
- Deuda pública
- Crecimiento

Por qué PCA?

- Variables **linealmente correlacionadas**
- Interpretación: PC1 = "desarrollo", PC2 = "estabilidad"
- Muy rápido (cientos de países)



Caso 2: t-SNE es superior

Situación: Clasificación de imágenes de dígitos (0-9)

Datos:

- 5000 imágenes 28×28
- 784 dimensiones por imagen
- 10 clases (dígitos)

Por qué t-SNE?

10 grupos
bien separados

Caso 2: t-SNE es superior

Situación: Clasificación de imágenes de dígitos (0-9)

Datos:

- 5000 imágenes 28×28
- 784 dimensiones por imagen
- 10 clases (dígitos)

Por qué t-SNE?

- Relaciones **altamente no lineales**

10 grupos
bien separados

Caso 2: t-SNE es superior

Situación: Clasificación de imágenes de dígitos (0-9)

Datos:

- 5000 imágenes 28×28
- 784 dimensiones por imagen
- 10 clases (dígitos)

Por qué t-SNE?

- Relaciones **altamente no lineales**
- Queremos **visualizar** grupos claramente

10 grupos
bien separados

Caso 2: t-SNE es superior

Situación: Clasificación de imágenes de dígitos (0-9)

Datos:

- 5000 imágenes 28×28
- 784 dimensiones por imagen
- 10 clases (dígitos)

Por qué t-SNE?

- Relaciones **altamente no lineales**
- Queremos **visualizar** grupos claramente
- Dataset tamaño moderado (5k puntos)

10 grupos
bien separados

Caso 2: t-SNE es superior

Situación: Clasificación de imágenes de dígitos (0-9)

Datos:

- 5000 imágenes 28×28
- 784 dimensiones por imagen
- 10 clases (dígitos)

Por qué t-SNE?

- Relaciones **altamente no lineales**
- Queremos **visualizar** grupos claramente
- Dataset tamaño moderado (5k puntos)
- No necesitamos aplicar a nuevas imágenes

10 grupos
bien separados

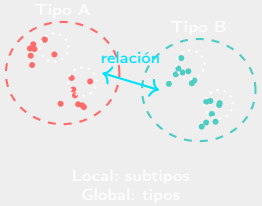
Caso 3: UMAP es superior

Situación: Secuenciación de ARN de células (single-cell RNA-seq)

Datos:

- 100,000 células
- 20,000 genes por célula
- Jerarquía: tipos → subtipos

Por qué UMAP?

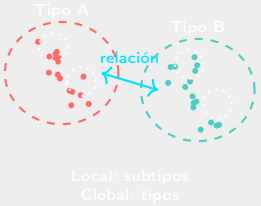


Caso 3: UMAP es superior

Situación: Secuenciación de ARN de células (single-cell RNA-seq)

Datos:

- 100,000 células
- 20,000 genes por célula
- Jerarquía: tipos → subtipos



Por qué UMAP?

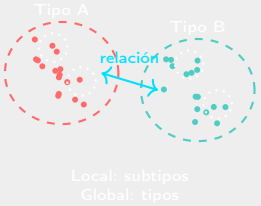
- **Dataset muy grande** (100k células)

Caso 3: UMAP es superior

Situación: Secuenciación de ARN de células (single-cell RNA-seq)

Datos:

- 100,000 células
- 20,000 genes por célula
- Jerarquía: tipos → subtipos



Por qué UMAP?

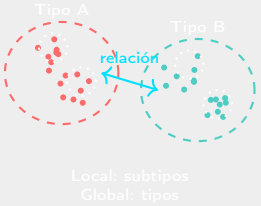
- **Dataset muy grande** (100k células)
- Necesita **estructura jerárquica** (tipos y subtipos)

Caso 3: UMAP es superior

Situación: Secuenciación de ARN de células (single-cell RNA-seq)

Datos:

- 100,000 células
- 20,000 genes por célula
- Jerarquía: tipos → subtipos



Por qué UMAP?

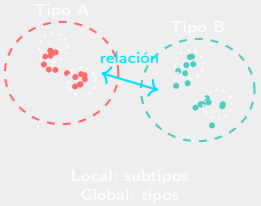
- **Dataset muy grande** (100k células)
- Necesita **estructura jerárquica** (tipos y subtipos)
- Preservar tanto **local como global**

Caso 3: UMAP es superior

Situación: Secuenciación de ARN de células (single-cell RNA-seq)

Datos:

- 100,000 células
- 20,000 genes por célula
- Jerarquía: tipos → subtipos



Por qué UMAP?

- **Dataset muy grande** (100k células)
- Necesita **estructura jerárquica** (tipos y subtipos)
- Preservar tanto **local como global**
- t-SNE sería demasiado lento

Conclusiones

Resumen y recomendaciones **Reflexión final**

Resumen: Las tres técnicas

1. PCA: Proyección lineal, rápida, interpretable
   - Usa cuando: datos lineales, necesitas velocidad o interpretación

Resumen: Las tres técnicas

1. **PCA:** Proyección lineal, rápida, interpretable
   - Usa cuando: datos lineales, necesitas velocidad o interpretación
2. **t-SNE:** Preserva vecindad local, visualiza clusters
   - Usa cuando: quieres ver grupos claramente, datos < 10k

Resumen: Las tres técnicas

1. **PCA:** Proyección lineal, rápida, interpretable
   - Usa cuando: datos lineales, necesitas velocidad o interpretación
2. **t-SNE:** Preserva vecindad local, visualiza clusters
   - Usa cuando: quieres ver grupos claramente, datos < 10k
3. **UMAP:** Balance local/global, rápida y escalable
   - Usa cuando: datos grandes, necesitas ambas estructuras

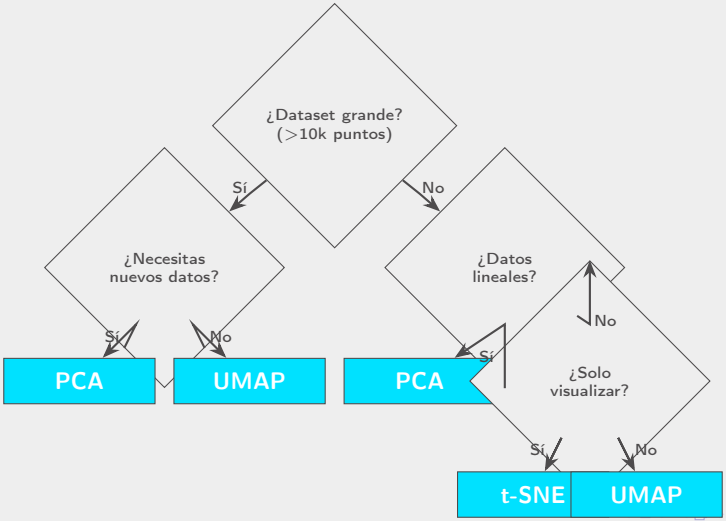
Resumen: Las tres técnicas

1. **PCA:** Proyección lineal, rápida, interpretable
   - Usa cuando: datos lineales, necesitas velocidad o interpretación
2. **t-SNE:** Preserva vecindad local, visualiza clusters
   - Usa cuando: quieres ver grupos claramente, datos < 10k
3. **UMAP:** Balance local/global, rápida y escalable
   - Usa cuando: datos grandes, necesitas ambas estructuras

Regla práctica:

- Empieza con PCA (rápido, simple)
- Si no funciona bien: prueba t-SNE o UMAP
- Para producción: PCA o UMAP (pueden aplicarse a nuevos datos)

Guía de decisión



Consejos prácticos

1. **Preprocesamiento es clave**
   - Estandarizar variables (media 0, desviación 1)
   - Eliminar outliers extremos

Consejos prácticos

1. **Preprocesamiento es clave**
   - Estandarizar variables (media 0, desviación 1)
   - Eliminar outliers extremos
2. **PCA como primer paso**
   - Si tienes 1000+ dimensiones: PCA a 50D, luego t-SNE/UMAP a 2D
   - Ahorra tiempo y mejora resultados

Consejos prácticos

1. **Preprocesamiento es clave**
   - Estandarizar variables (media 0, desviación 1)
   - Eliminar outliers extremos
2. **PCA como primer paso**
   - Si tienes 1000+ dimensiones: PCA a 50D, luego t-SNE/UMAP a 2D
   - Ahorra tiempo y mejora resultados
3. **Experimenta con hiperparámetros**
   - t-SNE: perplexity (5-50)
   - UMAP: n_neighbors (5-50), min_dist (0.0-0.99)

Consejos prácticos

1. **Preprocesamiento es clave**
   - Estandarizar variables (media 0, desviación 1)
   - Eliminar outliers extremos
2. **PCA como primer paso**
   - Si tienes 1000+ dimensiones: PCA a 50D, luego t-SNE/UMAP a 2D
   - Ahorra tiempo y mejora resultados
3. **Experimenta con hiperparámetros**
   - t-SNE: perplexity (5-50)
   - UMAP: n_neighbors (5-50), min_dist (0.0-0.99)
4. **Valida resultados**
   - Usa métricas: Silhouette Score, Trustworthiness
   - Compara múltiples técnicas

Consejos prácticos

1. **Preprocesamiento es clave**
   - Estandarizar variables (media 0, desviación 1)
   - Eliminar outliers extremos
2. **PCA como primer paso**
   - Si tienes 1000+ dimensiones: PCA a 50D, luego t-SNE/UMAP a 2D
   - Ahorra tiempo y mejora resultados
3. **Experimenta con hiperparámetros**
   - t-SNE: perplexity (5-50)
   - UMAP: n_neighbors (5-50), min_dist (0.0-0.99)
4. **Valida resultados**
   - Usa métricas: Silhouette Score, Trustworthiness
   - Compara múltiples técnicas
5. **t-SNE/UMAP no son deterministas**
   - Ejecuta varias veces con diferentes semillas
   - Verifica que los patrones sean consistentes

Limitaciones a considerar

Técnica	Limitación principal	Cuándo es problema
PCA	Solo relaciones lineales	Datos con curvas, espirales, formas complejas
t-SNE	No preserva distancias globales	Cuando importa la separación real entre grupos
t-SNE	Muy lento	Datasets $>$ 10,000 puntos
t-SNE	No aplica a nuevos datos	Sistemas en producción
UMAP	Menos interpretable	Cuando necesitas explicar cada dimensión
UMAP	No determinista	Resultados pueden variar entre ejecuciones

Pregunta de reflexión

¿Pregunta para pensar?

Tienes un dataset de 50,000 transacciones bancarias con 100 variables cada una. Quieres:

1. Detectar patrones de fraude (clusters anómalos)
2. Aplicar el modelo a nuevas transacciones en tiempo real
3. Explicar a auditores qué variables son importantes

¿Qué técnica(s) usarías y por qué?

Pregunta de reflexión

¿Pregunta para pensar?

Tienes un dataset de 50,000 transacciones bancarias con 100 variables cada una. Quieres:

1. Detectar patrones de fraude (clusters anómalos)
2. Aplicar el modelo a nuevas transacciones en tiempo real
3. Explicar a auditores qué variables son importantes

¿Qué técnica(s) usarías y por qué?

Pista: Considera múltiples técnicas en secuencia

Posible solución

Estrategia híbrida:

1. **PCA primero** (100D $\rightarrow$ 20D)
   - Identificar variables importantes (interpretabilidad)
   - Reducir ruido y acelerar siguientes pasos
   - Puede aplicarse a nuevas transacciones

Posible solución

Estrategia híbrida:

1. **PCA primero** (100D $\rightarrow$ 20D)
   - Identificar variables importantes (interpretabilidad)
   - Reducir ruido y acelerar siguientes pasos
   - Puede aplicarse a nuevas transacciones
2. **UMAP para visualización** (20D $\rightarrow$ 2D)
   - Detectar clusters de fraude visualmente
   - Escala bien con 50k transacciones
   - Puede aplicarse a nuevas transacciones

Posible solución

Estrategia híbrida:

1. **PCA primero** (100D $\rightarrow$ 20D)
   - Identificar variables importantes (interpretabilidad)
   - Reducir ruido y acelerar siguientes pasos
   - Puede aplicarse a nuevas transacciones
2. **UMAP para visualización** (20D $\rightarrow$ 2D)
   - Detectar clusters de fraude visualmente
   - Escala bien con 50k transacciones
   - Puede aplicarse a nuevas transacciones
3. **t-SNE opcional** para reportes
   - Solo para visualizar clusters en presentaciones
   - No para el sistema en producción

Posible solución

Estrategia híbrida:

1. **PCA primero** (100D $\rightarrow$ 20D)
   - Identificar variables importantes (interpretabilidad)
   - Reducir ruido y acelerar siguientes pasos
   - Puede aplicarse a nuevas transacciones
2. **UMAP para visualización** (20D $\rightarrow$ 2D)
   - Detectar clusters de fraude visualmente
   - Escala bien con 50k transacciones
   - Puede aplicarse a nuevas transacciones
3. **t-SNE opcional** para reportes
   - Solo para visualizar clusters en presentaciones
   - No para el sistema en producción

Lección: Combinar técnicas suele ser la mejor estrategia

Conclusión final

No existe una técnica perfecta

- **PCA**: Tu herramienta de análisis rápido
  - Simple, rápida, interpretable

Conclusión final

No existe una técnica perfecta

- **PCA:** Tu herramienta de análisis rápido
  - Simple, rápida, interpretable
- **t-SNE:** Tu microscopio de clusters
  - Para ver detalles finos, grupos ocultos

Conclusión final

No existe una técnica perfecta

- **PCA**: Tu herramienta de análisis rápido
  - Simple, rápida, interpretable
- **t-SNE**: Tu microscopio de clusters
  - Para ver detalles finos, grupos ocultos
- **UMAP**: Tu navaja suiza
  - Versátil, rápida, escalable

Conclusión final

No existe una técnica perfecta

- **PCA**: Tu herramienta de análisis rápido
  - Simple, rápida, interpretable
- **t-SNE**: Tu microscopio de clusters
  - Para ver detalles finos, grupos ocultos
- **UMAP**: Tu navaja suiza
  - Versátil, rápida, escalable

La mejor técnica depende de:

Tus datos + Tu objetivo + Tus recursos

UTEC Posgrado

UTEC Posgrado