

The background features a woman in a white shirt sitting at a desk with a laptop, looking thoughtful with her hand to her chin. The scene is overlaid with a complex digital interface consisting of various data visualization elements: bar charts, line graphs, pie charts with percentages (50%, 85%, 15%), document icons, and a network of blue nodes and lines. A large, glowing white cloud is positioned at the top, partially obscured by the network lines. The overall color palette is dominated by blues and whites, creating a high-tech, data-driven atmosphere.

**iic**

instituto  
de ingeniería  
del conocimiento

# Algoritmos de visualización 2D: PCA y alternativas

**Autoras:**

Cristina Sandoval

Estudiante de Ciencia de Datos e IA en la UPM

Ana Serrano

Científica de datos en el Instituto de  
Ingeniería del Conocimiento

## 1. INTRODUCCIÓN

La visualización de datos desempeña un papel fundamental en la comprensión y el **análisis de información compleja**. En particular, en el ámbito de la reducción de la dimensionalidad, los algoritmos de visualización 2D se convierten en herramientas muy útiles que permiten representar conjuntos de datos de alta dimensión en un espacio bidimensional, lo que facilita la interpretación humana y la **detección de patrones**. A lo largo de este documento se presentarán diversas técnicas para proyectar datos de alta dimensionalidad en representaciones más manejables y significativas.

Estos algoritmos no solo ayudan a simplificar la visualización de datos, sino que también desempeñan un papel importante en la identificación de características relevantes, la detección de patrones no lineales y la interpretación de datos complejos en una variedad de aplicaciones, desde el [procesamiento de imágenes](#) hasta el [análisis genómico](#), por ejemplo, en el caso del sector salud. Además, son valiosos para realizar tareas específicas como determinar si existe separación de datos de distintas clases, evaluar si hay variaciones en función de una variable objetivo en problemas de regresión o detectar clusters o valores atípicos en un conjunto de datos.

A lo largo de esta exploración, llevada a cabo desde el **Instituto de Ingeniería del Conocimiento (IIC)**, examinaremos las técnicas

de Análisis de Componentes Principales (Principal Component Analysis, PCA), Sparse PCA, Kernel PCA, ISOMAP, autocodificadores, t-SNE y UMAP, cada una con sus propios métodos y ventajas para simplificar la visualización de datos y revelar estructuras subyacentes.

Para la comparación de los diferentes algoritmos vamos a usar el **conjunto de datos de flores iris**. Este conjunto de datos tiene originalmente 4 características además de la variable etiqueta y cuenta con 150 filas. Cada dato corresponde a una flor de una especie (etiqueta) y las características son 4 mediciones (largo y ancho del pétalo y del sépalo). La variable etiqueta (clase) es categórica y tiene 3 valores posibles: setosa, versicolour o virginica.

De las 150 observaciones que tenemos, no faltan datos, por lo que no hace falta tratarlos. Además, las clases están balanceadas: hay 50 observaciones en cada una. Al aplicar estos algoritmos, nuestro objetivo será visualizar en 2D cómo se distribuyen los datos y determinar si los datos son distintos en función de su especie.

## 2. COMPARATIVA DE ALGORITMOS

### 2.1. Análisis de Componentes Principales (PCA)

El Análisis de Componentes Principales (Principal Component Analysis, PCA) consiste en transformar un conjunto de variables originales en otro conjunto de variables (componentes principales) que son combinaciones lineales de las primeras. Estas nuevas variables retienen toda la **información sobre la variabilidad del conjunto original**, pero la mayoría de las componentes principales tienen variabilidad pequeña y pueden ser ignoradas sin perder información significativa.

Los componentes principales se construyen en orden de importancia en términos de la variabilidad total que explican. Son combinaciones lineales de las variables originales y son independientes entre sí, así que cada una representa información única. Las condiciones de aplicación incluyen que las variables sean cuantitativas y **las variables originales estén correlacionadas**.

Se puede ver que los datos se separan bastante bien, sobre todo para las flores de especie setosa (en rojo): se ve una clara separación con las otras dos clases. Con las clases versicolour (en gris) y virginica, (en amarillo) la separación es menor, pero igualmente evidente.

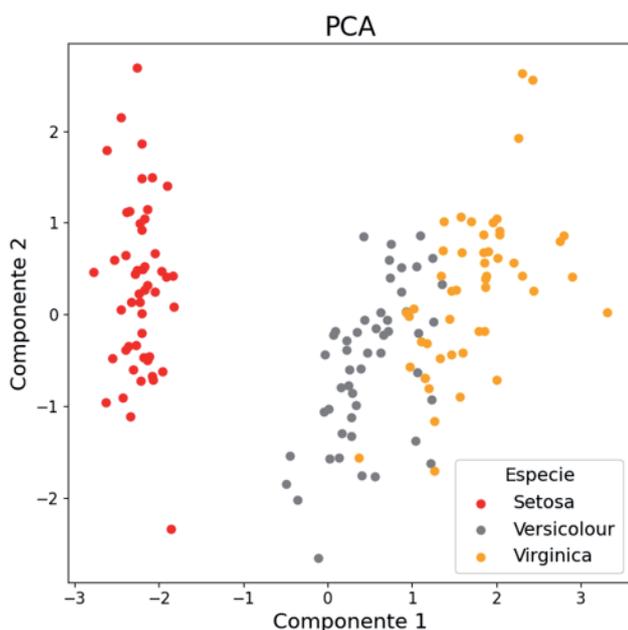
Aunque el PCA se formuló bajo unos supuestos teóricos, es común utilizarlo a pesar de que estos no se cumplan.

#### Ventajas

- Elimina multicolinealidad entre variables.
- Es bastante rápido en comparación con otras técnicas.
- Se puede saber qué características originales tienen más peso en las 2 dimensiones seleccionadas.
- Se puede saber cuánta variabilidad de los datos explican las 2 dimensiones en las que se proyectan los datos (es decir, una estimación de la información original que preservan).

#### Desventajas

- Sensible a la escala, por lo que los datos deben estar estandarizados.
- Sólo captura información distribuida linealmente.
- Se asumen los siguientes supuestos:
- Correlación lineal entre las características originales.
- Normalidad de las características originales.



## 2.2. Sparse PCA

Sparse PCA (Análisis de Componentes Principales con Restricciones de *Esparsidad*) es una variante de PCA diseñada para **reducir la dimensionalidad de datos** de alta dimensión, de manera que las componentes resultantes sean combinaciones lineales de un conjunto limitado de características. A diferencia del PCA estándar, Sparse PCA impone restricciones de *esparsidad*, limitando el número de características significativas en cada componente.

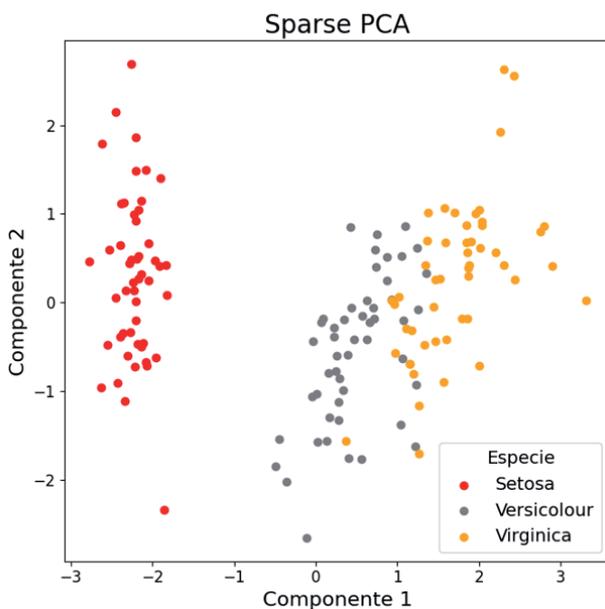
Sparse PCA es útil en situaciones donde se necesita una reducción de dimensionalidad, como en el procesamiento de imágenes o el análisis genómico, para identificar **características más informativas y representativas** en términos de variabilidad de

### Ventajas

- Selección automática de las características más significativas de cada componente.
- Puede ayudar a identificar patrones en las características originales.
- Introduce regularización con la penalización de esparsidad, lo que puede reducir el ruido.

### Desventajas

- Computacionalmente más costoso que PCA.
- Al implementarlo, hay que elegir hiper-parámetros.
- Interpretación más compleja que PCA.
- Mismas suposiciones que en PCA.



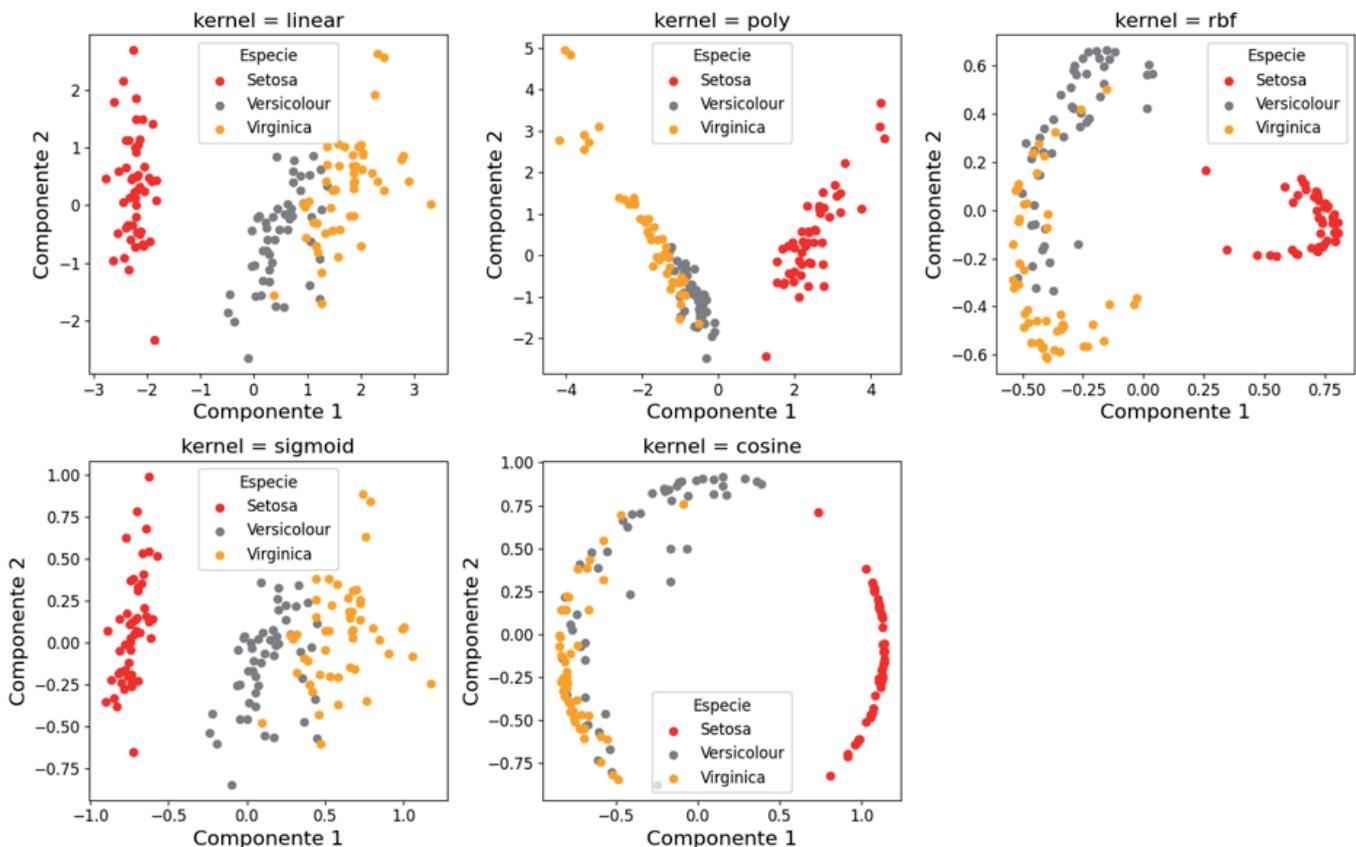
En este caso, el algoritmo da un resultado igual al del PCA, ya que el número de características originales (4) es bajo.

### 2.3. Kernel PCA

Kernel PCA (Análisis de Componentes Principales con función Núcleo) es una extensión de PCA que permite la **reducción de dimensionalidad en espacios de características no lineales**. En lugar de aplicarse directamente a los datos originales, Kernel PCA utiliza funciones núcleo para transformar los datos a un espacio de características de mayor dimensión. Este proceso implica una transformación no lineal que utiliza funciones como el kernel polinomial o coseno. Después de la transformación, se aplica PCA en el nuevo espacio de características para encontrar componentes principales que expliquen la máxima varianza.

Kernel PCA es **especialmente útil cuando los datos no cumplen las condiciones de normalidad y correlación lineal**, siendo beneficioso en aplicaciones como el procesamiento de imágenes, el análisis biológico y la detección de patrones no lineales en datos de alta dimensionalidad. Hay que tener en cuenta que Kernel PCA puede ser más costoso computacionalmente que el PCA estándar debido a la transformación a un espacio de características de mayor dimensionalidad.

Kernel PCA con distintas funciones kernel



Como se puede ver, el resultado cambia bastante en función del kernel a usar. En este caso concreto, la mejor separación podría decirse que es la del kernel lineal. Esto puede deberse a que las relaciones entre las variables en los datos se pueden aproximar bien usando una transformación lineal en un espacio de características más grande.

## 2.4. Autocodificadores

Los autocodificadores son una técnica común en reducción de dimensionalidad y procesamiento de datos. Funcionan mediante una **red neuronal con capas de entrada, de codificación y de decodificación**. En la fase de codificación, reduce la dimensión de los datos a través de transformaciones no lineales, generando una representación más compacta llamada espacio latente o embedding. En la fase de decodificación, intentan reconstruir los datos originales a partir de esta representación reducida.

Durante el entrenamiento, los autocodificadores buscan minimizar la diferencia entre los datos originales y los reconstruidos, ajustando los pesos de la red. Una vez entrenados, la capa de codificación se usa para obtener una **representación de menor dimensión**, que sirve como técnica de reducción de dimensionalidad. La idea es que, si el error entre el dato original y su reconstrucción es pequeño, el espacio latente de pequeña dimensión está manteniendo la información importante.

Los autocodificadores son flexibles, capaces de aprender representaciones útiles incluso en datos no lineales, y se aplican en reconstrucción, generación de datos y detección de anomalías, especialmente cuando se dispone de **datos etiquetados**.

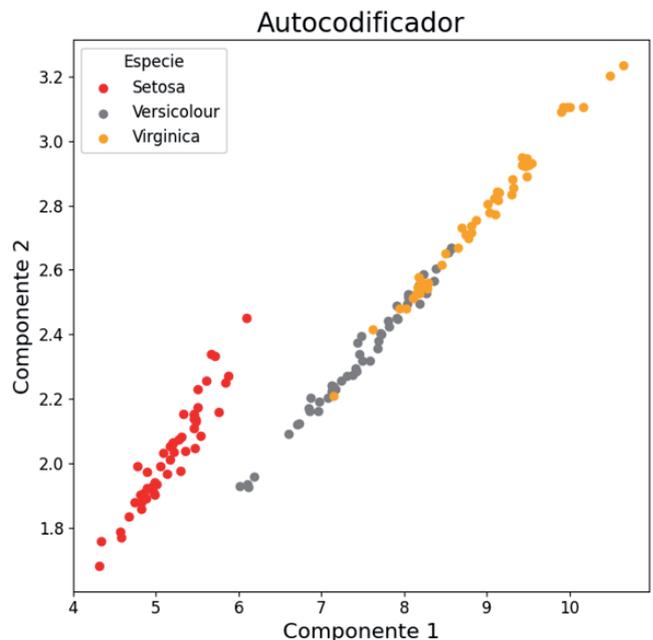
Este método muestra los datos de especie versicolour (en gris) y virginica (en amarillo) más solapados. Además, parece que las dos dimensiones de su espacio latente están bastante correlacionadas lineal y positivamente.

Ventajas

- Captura estructuras no lineales.

Desventajas

- Se debe elegir la función kernel.
- Mayor complejidad computacional.



Ventajas:

- Eliminación de ruido.
- Pueden servir para generación de datos.
- Permiten la transferencia de conocimiento (transfer learning).
- Conocimiento de cuanta información se mantiene (mediante el error de entrenamiento).

Desventajas:

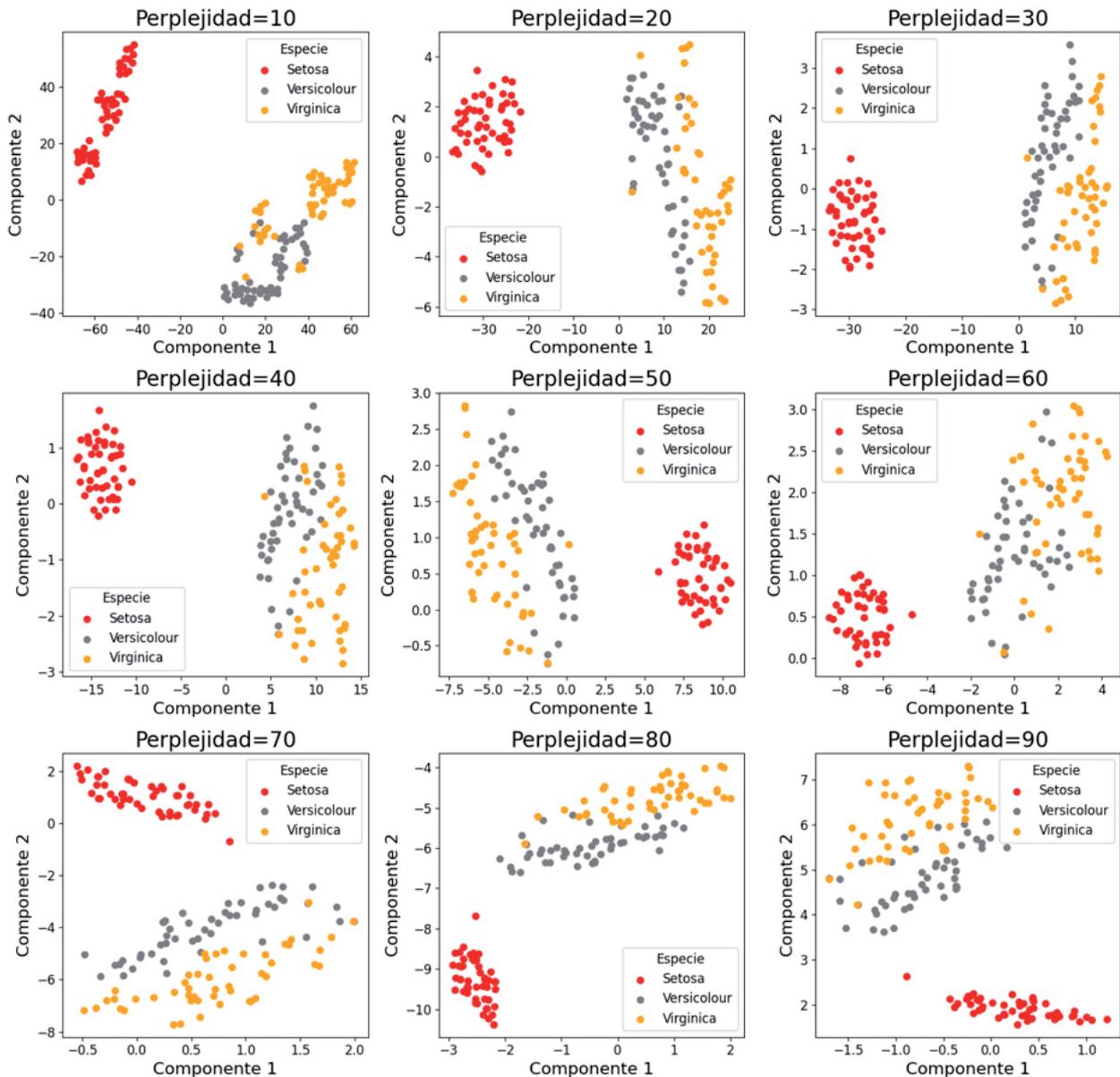
- Supervisión limitada.
- Sensibilidad a hiper-parámetros y arquitectura de la red.
- Complejidad computacional.
- Limitaciones en datos con estructuras complejas.

### 2.5. t-SNE

t-SNE es una técnica de reducción de dimensiones. Opera en dos etapas: **construcción de distribuciones de probabilidad y optimización de la proyección**. En la primera etapa, calcula similitudes condicionales en el espacio de alta dimensión mediante funciones de probabilidad condicional basadas en distribuciones Gaussianas. Luego, construye distribuciones de probabilidad en un espacio de baja dimensión y optimiza la proyección para que las similitudes sean lo más parecidas posible entre ambos

espacios, utilizando la divergencia de Kullback-Leibler. Este proceso iterativo mejora la semejanza entre las probabilidades condicionales en cada iteración.

t-SNE es efectivo para visualizar conjuntos de datos de dimensionalidad pequeña y mediana, pero puede ser costoso en conjuntos de datos de gran dimensionalidad original. **Se recomienda su uso después de PCA en conjuntos grandes**. Es importante mencionar que t-SNE es un algoritmo estocástico, y los resultados pueden variar según la inicialización.



En la literatura se dice que los hiper-parámetros elegidos en este algoritmo influyen con gran peso en la representación que se obtiene. Los gráficos anteriores también indican esto mismo. El algoritmo de t-SNE también se basa en el concepto de “vecinos”, y el número de estos está determinado por la perplejidad. A mayor perplejidad, mayor número de “vecinos” se empleará en las estimaciones.

Ventajas:

- Preservación de estructuras locales.
- Robusto a escalas.

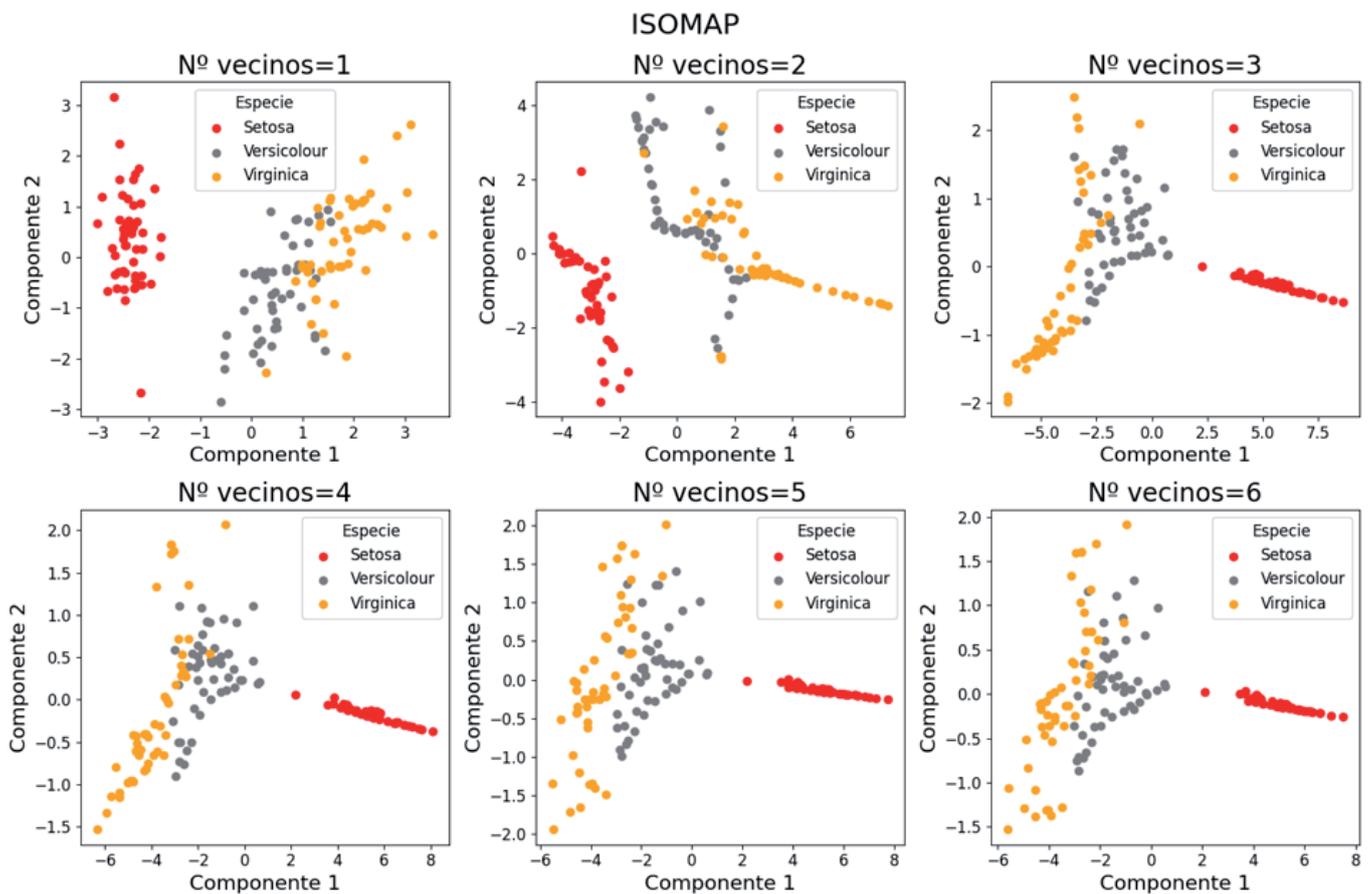
Desventajas:

- Sensibilidad a la inicialización.
- Elección de hiper-parámetros.

## 2.6. ISOMAP

ISOMAP (Isometric Feature Mapping) es un algoritmo usado para simplificar y entender datos complejos. Su objetivo principal es **conservar las relaciones de distancia entre puntos** cuando se reduce la dimensionalidad de los datos. El proceso incluye la construcción de un grafo basado en la vecindad de los puntos, el

cálculo de distancias especiales entre ellos y, finalmente, la representación de los datos en un espacio más simple que conserva estas relaciones de distancia. ISOMAP es útil para **datos con estructuras no lineales**, como conjuntos de datos con formas o variedades no lineales, facilitando su visualización y análisis.



Como se puede ver en la imagen anterior, se obtienen resultados bastante diferentes si usamos diferentes números de “vecinos” de cada punto al construir el grafo. Aunque hay una separación clara en casi todos los gráficos, podría decirse que donde mejor se ve es para 5 o 6 “vecinos”.

Ventajas:

- Captura estructuras no lineales.
- Fácilmente interpretable (dos puntos más cercanos en 2D también lo son en la dimensionalidad original).

Desventajas:

- Computacionalmente costoso.
- Sensibilidad a la densidad de los datos.
- Aunque preserva estructuras no lineales, puede tener problemas con estructuras muy complejas.
- Se debe escoger un número de “vecinos”.

## 2.7. UMAP

UMAP (Uniform Manifold Approximation and Projection) es un método similar a ISOMAP. UMAP destaca por su rapidez y eficacia, especialmente en comparación con otros métodos como t-SNE, siendo **apropiado para conjuntos de datos de gran dimensionalidad**.

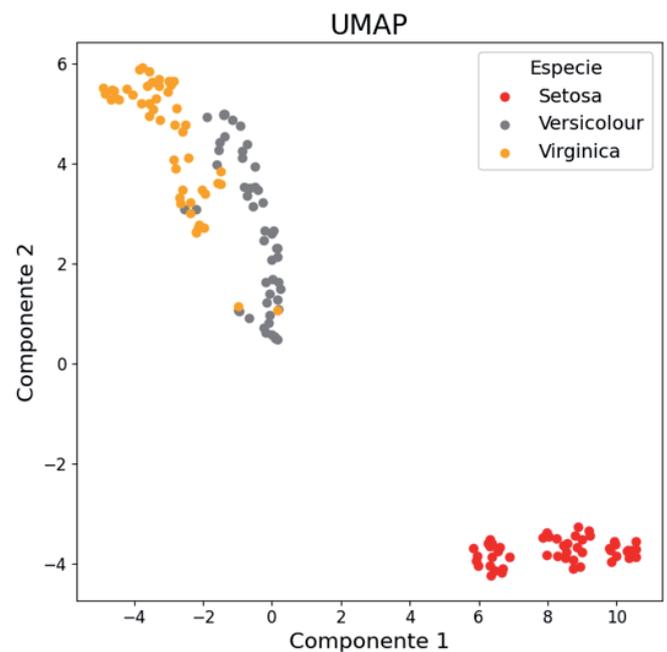
Se puede ver que UMAP ha hecho una separación muy clara de los distintos grupos.

Ventajas:

- Mejora de la preservación de la estructura local.
- Eficiente y relativamente rápido.

Desventajas:

- Requiere más recursos computacionales que otras técnicas.



### 3. CONCLUSIONES

Aunque está muy extendido el uso del Análisis de Componentes Principales (Principal Component Analysis, PCA), existen más opciones de reducción de la dimensionalidad. Con los datos de flores empleados en este ejemplo, la mayoría de los algoritmos han captado las **diferencias entre las distribuciones de los datos de las distintas clases** (en particular, entre setosa y las demás), permitiendo visualizarlas en un plano de 2 dimensiones.

En este contexto, hay que tener en cuenta que, al tratarse de **métodos no supervisados**, la eficacia no puede medirse de antemano. La elección del método debe basarse en las **características del conjunto de datos a analizar**

y en cómo deseamos interpretar los resultados. Por lo tanto, es importante entender conceptualmente las diferentes técnicas y sus fundamentos teóricos para **elegir aquellas más apropiadas para cada situación**. Lo más recomendable cuando se tiene un objetivo específico (determinar si las clases se separan, si la distribución de los datos varía en relación con una variable continua o si hay puntos atípicos) es emplear varios métodos. De esta manera, se podrá evaluar si alguno de ellos captura la asociación que esperas encontrar.



iic

©ADIC

C/ Francisco Tomás y Valiente, nº 11  
EPS, edificio B, 5ª planta  
UAM Cantoblanco  
28049 Madrid, España.

Tel.: (+34) 91 497 2323  
Fax: (+34) 91 497 2334  
iic@iic.uam.es  
www.iic.uam.es

