# Crear una clase PCA

1. Define una clase PCA que tenga métodos para fit(), fit\_transform() y transform(), así como parámetros hiperámetros en la inicialización.
2. Implementar el método fit():
   1. Calcula la matriz de covarianza de los datos.
   2. Calcula los vectores y valores propios de la matriz de covarianza.
   3. Ordena los vectores propios según los valores propios en orden descendente.
3. Implementar el método fit\_transform():
   1. Utiliza los resultados del método fit() para proyectar los datos originales en el nuevo espacio de características reducido.
4. Implementar el método transform():
   1. Proyecta los datos originales en el nueso espacio de características reducido utilizando los resultados del método fit().
   2. Agregar parámetros hiperparámetros:
5. Permitir que la clase PCA tome parámetros como el número de componentes principales a mantener.

Esta implementación básica de PCA proporciona funcionalidad para ajustar el modelo a los datos (fit()), transformar los datos originales en el nuevo espacio de características reducido (fit\_transform()), y proyectar nuevos datos en el mismo espacio de características reducido (transform()).

<https://github.com/rushter/MLAlgorithms/blob/master/mla/pca.py>

# Crear una clase SVD:

1. Define una clase SVD que contenga métodos para calcular la descomposición de valores singulares y para realizar la transformación de los datos.
2. **Implementar el método fit()**:

Calcula la descomposición de valores singulares de la matriz de datos.

La función np.linalg.svd es parte de la biblioteca NumPy y se utiliza para calcular la descomposición de valores singulares (SVD) de una matriz. La SVD es una herramienta fundamental en el ámbito del análisis de datos y se utiliza en una variedad de aplicaciones, como la reducción de dimensionalidad, la compresión de datos y la reconstrucción de matrices.

Cuando llamas a np.linalg.svd(X), estás calculando la descomposición de valores singulares de la matriz X, y obtienes tres matrices como resultado: U, S y Vt. Estas matrices representan la factorización de la matriz original X de la siguiente manera:

U es una matriz unitaria que contiene los vectores singulares izquierdos.

S es una matriz diagonal que contiene los valores singulares.

Vt es la matriz transpuesta de la matriz unitaria que contiene los vectores singulares derechos.

1. **Implementar el método transform()**:

Realiza la transformación de los datos originales utilizando los resultados de la descomposición de valores singulares.

En este ejemplo, la clase SVD tiene métodos para fit() y transform(). El método fit() calcula la descomposición de valores singulares de la matriz de datos X y almacena los componentes U, S y Vt. El método transform() realiza la transformación de los datos originales X utilizando los componentes Vt.