KIBERNUM
IT Academy

Módulo 6 – Aprendizaje de Máquina No Supervisado

Principal Component Analysis (PCA)

Especialización en Ciencia de Datos

Objetivos

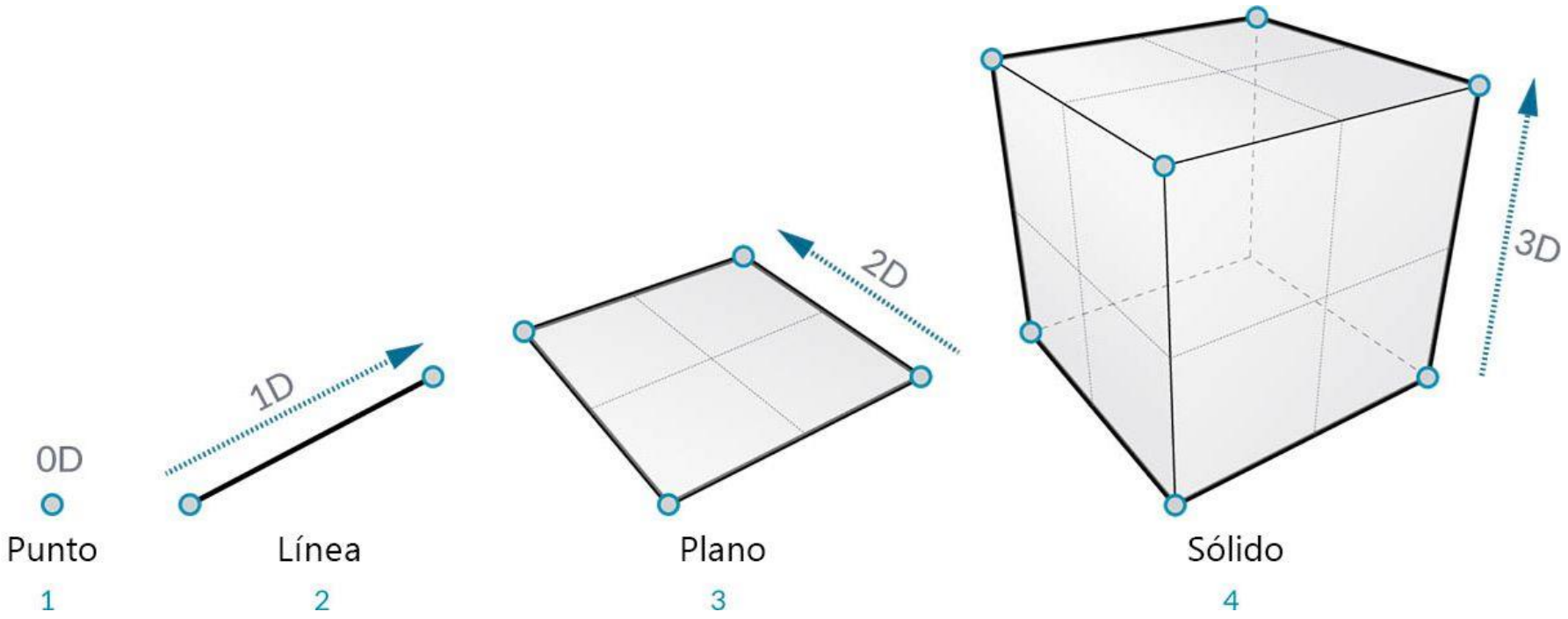
- Describir el concepto de reducción de dimensionalidad.
- Identificar principales algoritmos para la reducción de dimensionalidad.

KIBERNUM

Reducción de Dimensionalidad

¿Qué es la reducción de dimensionalidad?

La reducción de dimensionalidad es un conjunto de técnicas utilizadas para **disminuir el número de variables o dimensiones en un conjunto de datos**. Esto implica eliminar características no relevantes o redundantes en los datos, y/o crear nuevas variables que resuman la información contenida en las variables originales.



KIBERNUM

Intuición sobre la Reducción de Dimensionalidad

Observe la siguiente imagen. ¿Puede armarse una idea de cómo es ese lugar? Este es un ejemplo de cómo la información tridimensional (alto, largo y profundidad) puede ser reducida a dos dimensiones, logrando entregar información suficiente para cumplir con su propósito.

KIBERNUM

Intuición sobre la Reducción de Dimensionalidad

Ahora, suponga que el ángulo en que se toma la imagen no fuera el adecuado y no logra captar la perspectiva del paisaje. En este caso, la reducción de dimensionalidad no cumplió con su propósito.

KIBERNUM

Intuición sobre la Reducción de Dimensionalidad

Un último ejemplo de cómo reducir 4 dimensiones (largo, ancho, profundidad, tiempo) a 3 dimensiones (largo, ancho y tiempo).

KIBERNUM

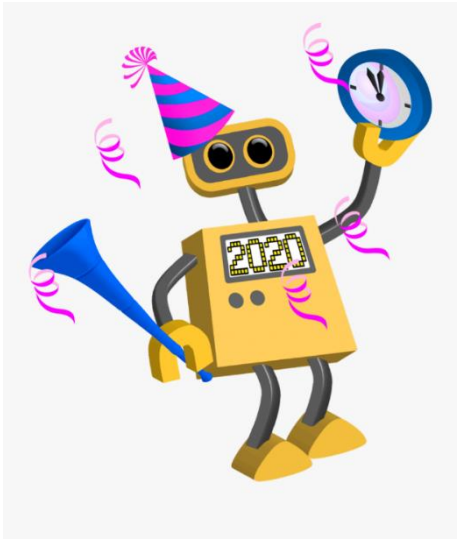
Propósito de la Reducción de Dimensionalidad

- La reducción de dimensionalidad se utiliza para abordar varios problemas en el análisis de datos, incluyendo la complejidad computacional, la visualización de datos y la mejora del rendimiento de los modelos de aprendizaje automático.

- Al reducir la cantidad de variables, es posible **disminuir la complejidad computacional de ciertas tareas**, como la clasificación y la agrupación de datos. Además, la reducción de dimensionalidad puede **ayudar a visualizar datos en espacios de menor dimen**sión, lo que puede ayudar a comprender mejor las relaciones entre las variables. Finalmente, la reducción de dimensionalidad puede mejorar el rendimiento de los modelos de aprendizaje automático, al **reducir el ruido en los datos y mejorar la generalización**.

Menor complejidad

Mejor generalización



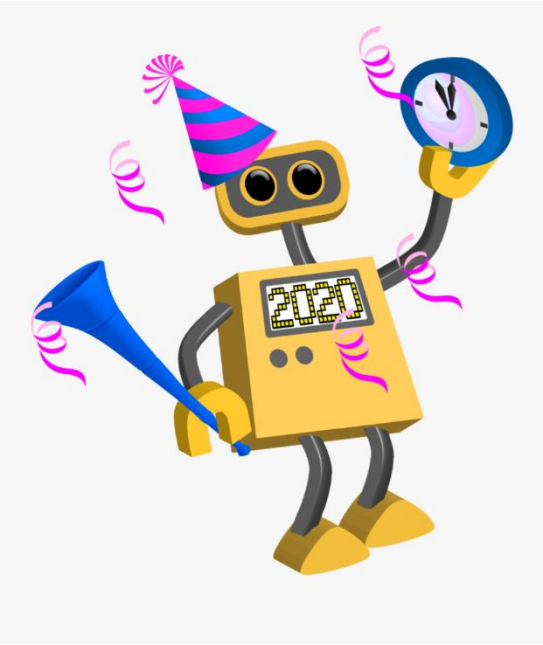
Simplicidad

Mejor visualización

KIBERNUM

Propósito de la Reducción de Dimensionalidad

Hasta ahora se había hablado de que una forma de mejorar el performance de un algoritmo era agregando más variables, lo cual puede sonar contradictorio. Pero hay que pensar que no todas las variables aportan de la misma forma al modelo.



El truco es sacrificar un poco de exactitud para ganar simplicidad en el modelo. Así es más fácil explorar, visualizar y analizar.

Set de datos menos complejos requieren menos recursos computacionales para su procesamiento.

Datos con menos ruido significa un major modelo.

KIBERNUM

Algoritmos para Reducción de Dimensionalidad

Existen varios algoritmos para la reducción de dimensionalidad, los cuales se pueden clasificar en dos categorías principales: la **selección de características** y la **extracción de características**.

La selección de características implica seleccionar un subconjunto de las variables originales para ser utilizadas en el análisis. Algunos de los algoritmos más comunes para la selección de características son:

1. **Filtro**: evalúa la relación entre cada variable y la variable de salida (o entre las variables entre sí) y selecciona las variables con la puntuación más alta.

2. **Wrapper**: utiliza un modelo de aprendizaje automático para seleccionar el conjunto de características que maximiza el rendimiento del modelo.

3. **Incrustado**: las características se seleccionan durante el entrenamiento del modelo de aprendizaje automático.

Algoritmos para Reducción de Dimensionalidad

La extracción de características, por otro lado, implica transformar las variables originales en un nuevo conjunto de variables que resuman la información contenida en las variables originales. Algunos de los algoritmos más comunes para la extracción de características son:

4. **Análisis de componentes principales** (PCA): transforma las variables originales en un conjunto de variables no correlacionadas que expliquen la mayor cantidad posible de la varianza en los datos.

5. **Análisis discriminante lineal** (LDA): encuentra la combinación lineal de variables que maximiza la separación entre las clases en los datos.

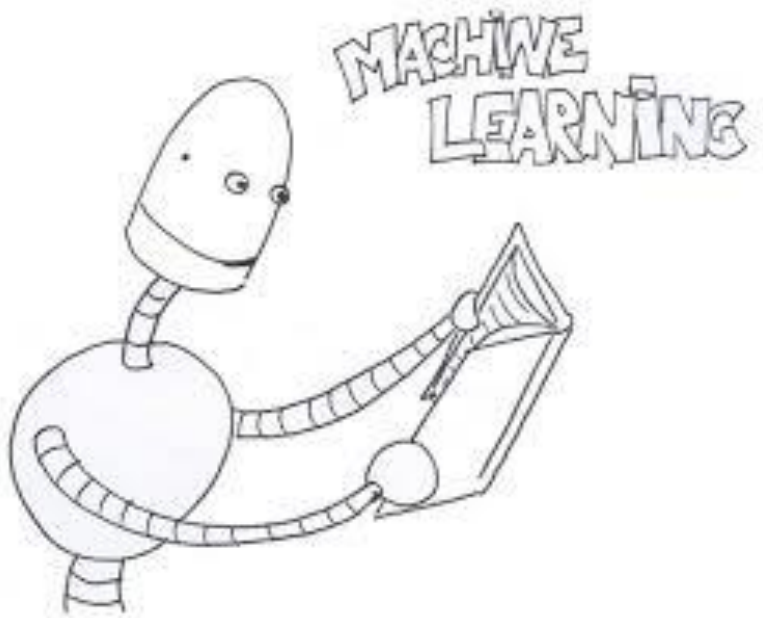
6. **T-distributed Stochastic Neighbor Embedding** (t-SNE): utiliza una función de probabilidad para transformar las variables originales en un conjunto de variables de menor dimensión que preserven la estructura de similitud entre los datos.

Estos son solo algunos ejemplos de algoritmos para la reducción de dimensionalidad. La elección del algoritmo depende del problema específico que se esté abordando y del tipo de datos que se estén utilizando.

Principal Component Analysis

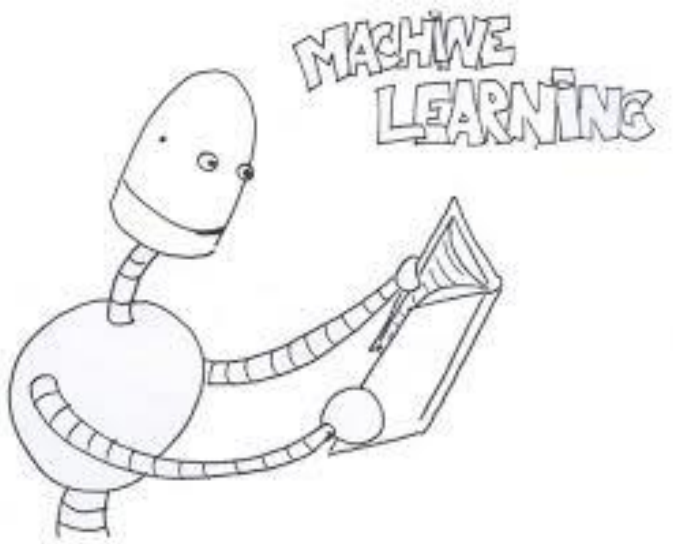
¿Qué es PCA?

Es un **método de reducción de dimensionalidad** que a menudo se utiliza para reducir la dimensionalidad de grandes sets de datos, mediante la **transformación** de una gran cantidad de variable en una menor cantidad de ellas, **reteniendo la mayor parte de la información**.



KIBERNUM

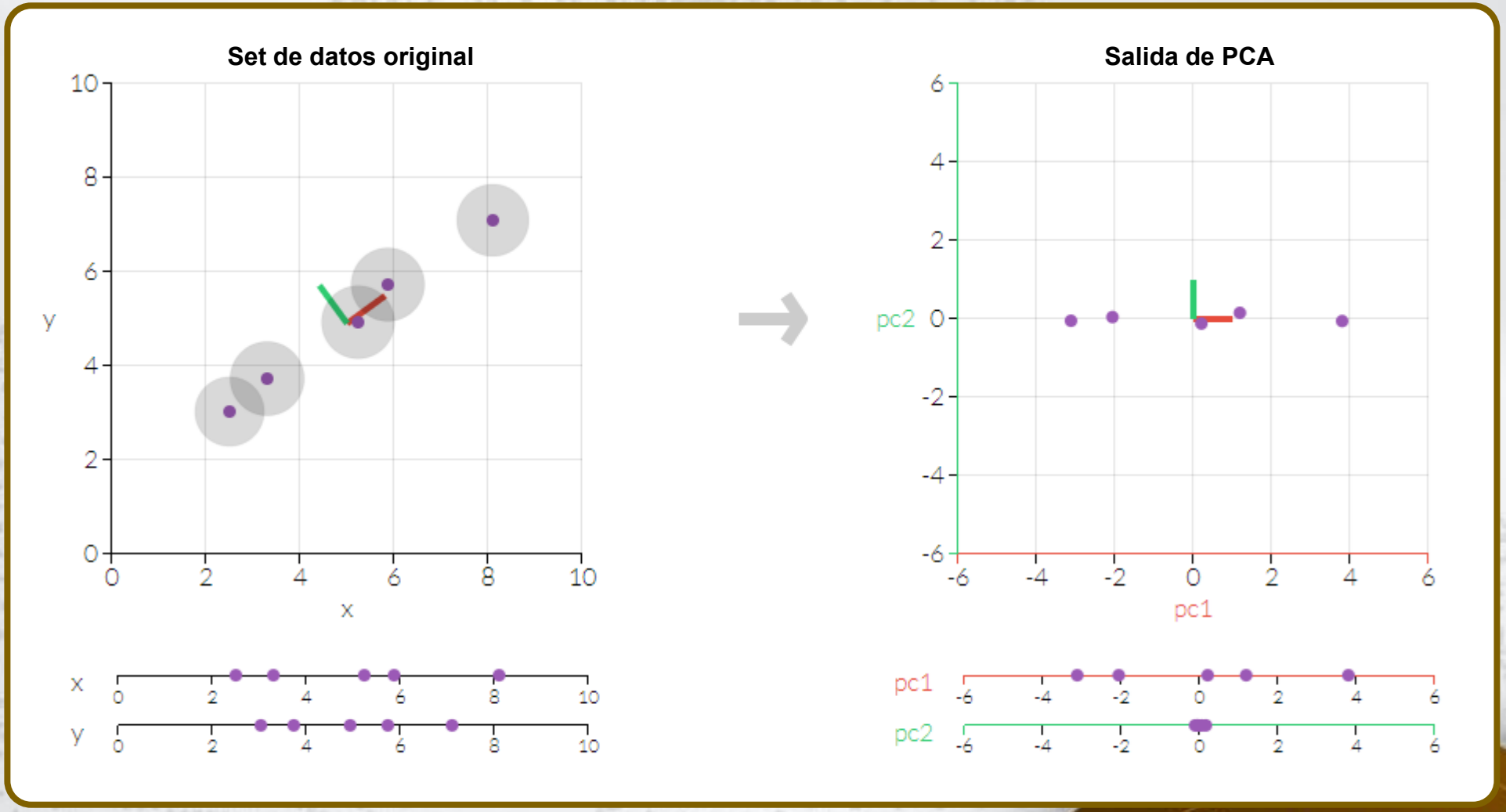
¿Cómo hace la reducción?



En palabras simples, esto es de lo que se trata PCA:

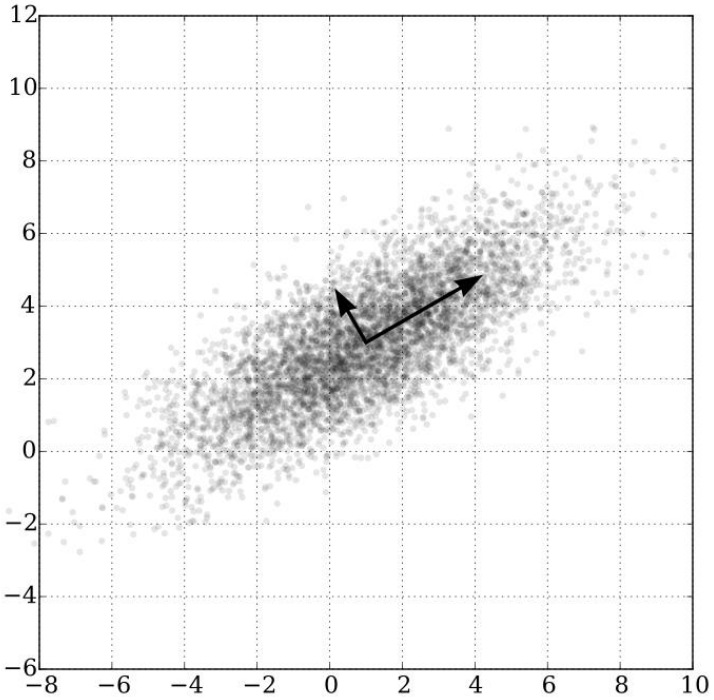
Encontrar las direcciones de variación máxima en datos de alta dimensión y proyectarlo en un subespacio de dimensiones más pequeñas mientras se conserva la mayor parte de la información.

¿Cómo hace la reducción?



Eigenvectors y Eigenvalues

Los **eigenvectors** (componentes principales) determinan las direcciones del Nuevo espacio de features, y los **eigenvalues** determinan su magnitud. En otras palabras, eigenvalues explican la varianza de la data a lo largo de los nuevos ejes de features.



KIBERNUM

PCA Paso a Paso

El procedimiento de determinación de los componentes principales es el siguiente:

Paso 1: Estandarización.

Paso 2: Calcular matriz de Covarianza (o bien realizar una Descomposición de Vector Singular).

Paso 3: Obtener los Eigenvectors y Eigenvalues desde la matriz de covarianza o de correlación.

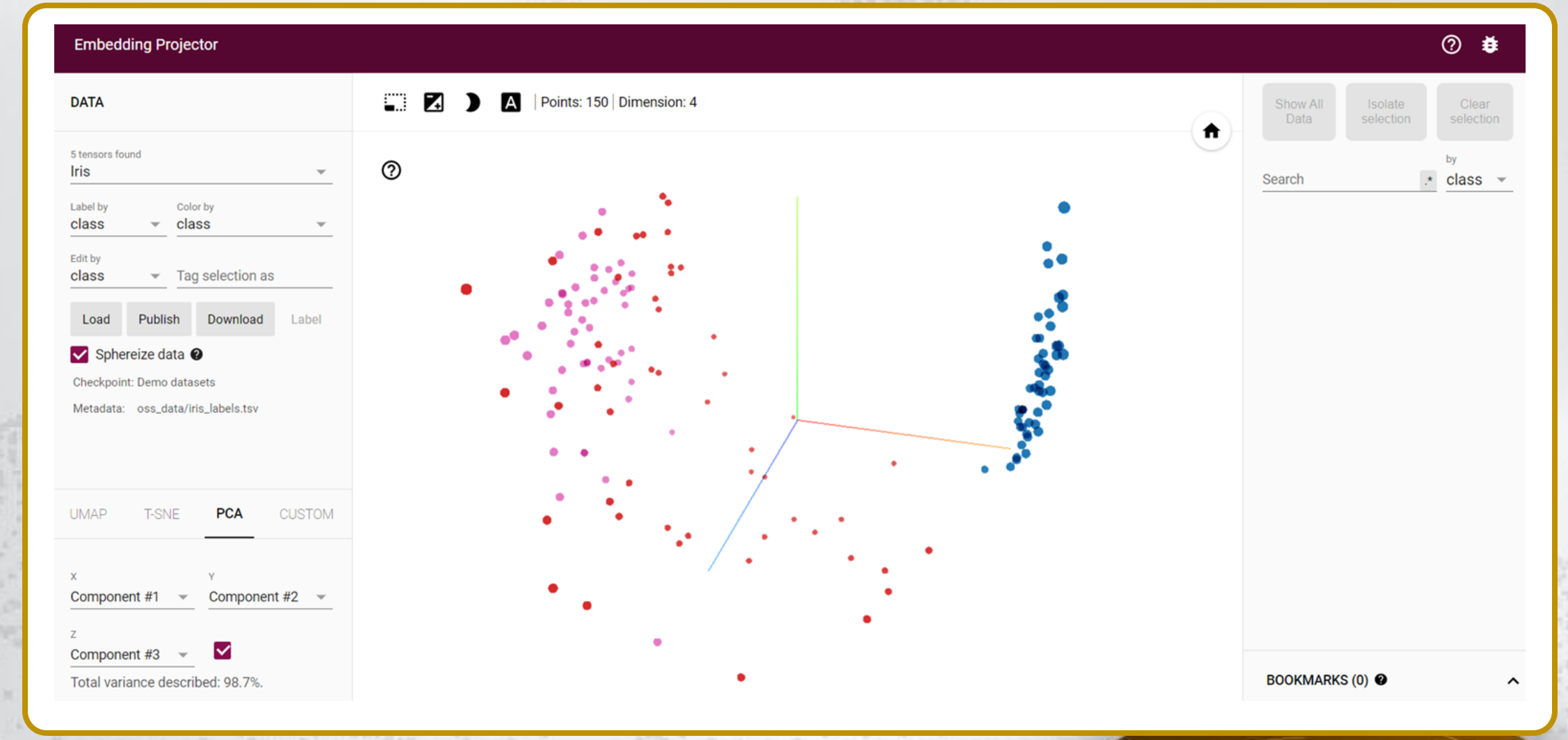
Paso 4: Ordenar los eigenvalues en orden descendientes y elegir los k eigenvectores que corresponden al k más largo eigenvalue, donde k es el número de dimensiones del nuevo subespacio.

Paso 5: Construir la matriz de proyección W a partir de los k eigenvectors seleccionados.

Paso 6: Transformar el dataset original X via W para obtener el subespacio Y de features k-dimensionales.

https://plot.ly/python/v3/ipython-notebooks/principal-component-analysis/

Playground



https://projector.tensorflow.org/

KIBERNUM

Ventajas

1. Reduce la dimensionalidad de los datos: PCA puede transformar un conjunto de datos de alta dimensionalidad en un conjunto de datos de baja dimensionalidad sin perder demasiada información.

2. Identifica patrones en los datos: PCA puede ayudar a identificar patrones y relaciones entre las variables originales.

3. Facilita la visualización de los datos: la reducción de dimensionalidad a través de PCA puede facilitar la visualización de los datos en espacios de menor dimensión.

4. Elimina la correlación entre variables: PCA transforma las variables originales en un conjunto de variables no correlacionadas, lo que puede ayudar a reducir el ruido en los datos.

Desventajas

1. La interpretación de los resultados puede ser difícil: los componentes principales generados por PCA pueden ser difíciles de interpretar y asignar a variables específicas en el conjunto de datos original.

2. Sensible a los valores atípicos: PCA es sensible a los valores atípicos en los datos, lo que puede afectar negativamente la calidad de los resultados.

3. Requiere una cantidad significativa de memoria y procesamiento: PCA puede requerir una cantidad significativa de memoria y procesamiento, especialmente para conjuntos de datos grandes.

4. Requiere que los datos estén estandarizados: para que PCA funcione correctamente, es importante que los datos estén estandarizados para que las variables tengan una media de cero y una desviación estándar de uno.

Resumen

- PCA no es un algoritmo regresivo.
- PCA aprende acerca de la relación entre los valores X e Y, encontrando la lista de los principales ejes.
- Permite reducir la dimensionalidad de la data, para:
  - Mejorar el performance de procesamiento de los algoritmos.
  - O bien, facilitar el análisis visual de los datos.
- PCA no se comporta muy bien cuando hay valores atípicos (outliers).
- No olvidar estandarizar los datos.
- A pesar de lo anterior, es el algoritmo más popular para la reducción de dimensiones.

Dudas y consultas

KIBERNUM

Fin Presentación

KIBERNUM