

Matplotlib



1. El problema del dongle

Matplotlib proporciona una interfaz de scripting con estado para una generación rápida y fácil de gráficos similares a los de MATLAB. Debido a que matplotlib es una biblioteca, los usuarios tienen acceso a todas las ricas estructuras de datos integradas de Python, como listas, diccionarios, conjuntos y más.



2. Descripción general de la arquitectura de matplotlib

El objeto matplotlib de nivel superior que contiene y administra todos los elementos de un gráfico determinado se denomina Figura. Una de las principales tareas arquitectónicas que matplotlib debe resolver es implementar un marco para representar y manipular la figura que se separe del acto de representar la figura en una ventana de interfaz de usuario o en una copia impresa. La arquitectura para lograr esto está lógicamente separada en tres capas, son: backend, artista y scripting.



3. Refactorización de backend

Los backends se refactorizaron para requerir solo la funcionalidad mínima necesaria en los propios backends, con todo lo demás movido al núcleo.

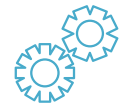


4. Transforma

Matplotlib pasa mucho tiempo transformando las coordenadas de un sistema a otro. Estos sistemas de coordenadas incluyen: datos, ejes, figura y visualización.

5. El oleoducto de polilínea

Al trazar trazos de líneas, hay una serie de pasos que se realizan para pasar de los datos sin procesar a la línea dibujada en la pantalla. Como Transformación, Handle missing data, Recorte, Ajuste y Simplificación.



6. Texto de matemáticas

Matplotlib proporciona dos formas de representar expresiones matemáticas: usetex y mathtext.



7. Pruebas de regresión

Algunas regresiones todavía caen a través de las grietas, pero en general la calidad de las versiones ha mejorado considerablemente desde que se implementó el marco de pruebas.



8. Lecciones aprendidas

Matplotlib podría haberse beneficiado en el desarrollo temprano de gastar más esfuerzo en la integración con módulos y API existentes, como los kits de herramientas Kiva y Enable. Sin embargo, la integración con la funcionalidad existente, puede hacer que las compilaciones y lanzamientos sean más complejas y reducir la flexibilidad en el desarrollo interno.



Es una biblioteca de trazado basada en Python con soporte completo para 2D y soporte limitado para gráficos 3D, ampliamente utilizado en la comunidad de computación científica de Python.

Origen: John Hunter

POR QUÉ USAMOS GRÁFICOS?

- Son una forma de tomar información y hacerla más comprensible.
- El punto de los gráficos es facilitar la comparación de diferentes conjuntos de datos.
- Cuanta más información pueda transmitir un gráfico sin aumentar la complejidad, mejor.

EL PEOR GRAFICO DEL MUNDO

El gráfico circular

PARA QUÉ SE USA EL GRÁFICO CIRCULAR?

- Para mostrar la relación de partes de un todo.
- Cuando se comparan 2-3 puntos de datos diferentes con cantidades muy diferentes de información
- Su uso real es hacer saber a la gente cómo se ve una fracción.

QUÉ LE PASA AL GRÁFICO CIRCULAR Y POR QUÉ SE DEBE DEJAR DE USAR?

- El gráfico solo es útil si podemos comparar todos y cada uno de los elementos dentro de él.
- La realidad es que los humanos no son muy buenos comparando trozos de círculo cuando se trata de tamaño. Es algo a tener en cuenta cuando se trata de articular la información de la manera más completa y comprensible posible.
- Los gráficos más elegantes no requieren etiquetado de datos. No deberías necesitar tener ningún número extraño en tus datos para expresar tu punto de vista. Si lo haces, estás usando el gráfico equivocado.

TOMAR EN CONSIDERACION ANTES DE ELEGIR UN GRAFICO CIRCULAR

- Siempre que haya similitud en la información disponible, un gráfico circular no es el gráfico correcto para usar.
- Siempre que hay varios (3 o más) puntos de datos diferentes, un gráfico circular no es el gráfico correcto para usar.
- Los gráficos circulares son muy fáciles de abusar.
- Un gráfico circular no es el gráfico correcto para usar si necesitas etiquetar cada porcentaje.

ANÁLISIS DE DATOS EXPLORATORIOS (EDA): PYTHON

1. Adquisición de datos

Es el proceso de encontrar y cargar los datos en nuestro sistema. Hay dos formas en las que podemos encontrar datos:

1. Datos privados
2. Datos públicos

2. Limpieza de datos

Es importante deshacerse de las irregularidades y limpiar los datos después de obtenerlos en nuestro sistema.

Las irregularidades son de diferentes tipos de datos: Faltan valores, Formato incorrecto, Encabezados incorrectos Y Anomalías/Outliers.

Por lo que se debe: Arreglar las filas y columnas, manejar valores faltantes, soltar los valores que faltan, manejo de valores atípicos, y estandarizar de valores.

3. Análisis univariable

Es cuando analizamos datos sobre una sola variable/columna de un conjunto de datos.

Análisis univariable categórico sin ordenar:

Una variable no ordenada es una variable categórica que no tiene orden definido.

Análisis Univariable Ordenado Categórico:

Las variables ordenadas son aquellas variables que tienen un rango natural de orden.

5. Análisis multivariante

Es cuando analizamos los datos tomando en cuenta más de dos variables/columnas de un conjunto de datos

4. Análisis bivariado

Es cuando analizamos los datos teniendo en cuenta dos variables/columnas de un conjunto de datos.

a) Análisis numérico-numérik:

Es el análisis de las dos variables numéricas de un conjunto de datos. Podemos analizarlo de tres maneras diferentes: Parcela de dispersión, Parcela de pares y Matriz de correlación

b) Numérico - Análisis categórico

Es el análisis de una variable numérica y una variable categórica de un conjunto de datos. Los analizamos principalmente utilizando gráficos medios, medianos y de caja.

c) Categórico - Análisis categórico

Es un enfoque para analizar conjuntos de datos para resumir sus principales características, a menudo con métodos visuales. Se puede usar un modelo estadístico o no, pero principalmente la EDA es para ver lo que los datos pueden decirnos más allá de la tarea formal de modelado o prueba de hipótesis.

EDA en Python utiliza la visualización de datos para dibujar patrones e información significativas. También implica la preparación de conjuntos de datos para su análisis mediante la eliminación de irregularidades en los datos.