

Contenido

[Historia y Fundamentos de Matplotlib II](#_Toc168975290)

[Tipos Básicos de Gráficos II](#_Toc168975291)

[Personalización de gráficos. II](#_Toc168975292)

[Introducción a Seaborn y Comparación con Matplotlib II](#_Toc168975293)

[Gráficos de Densidad y de Violín para Distribuciones III](#_Toc168975294)

[Gráficos de Catplot y Factorplot para Categorización de Datos III](#_Toc168975295)

[Creación de Gráficos Interactivos III](#_Toc168975296)

[Introducción a GeoPandas para Manipulación de Datos Geoespaciales IV](#_Toc168975297)

[Creación de Mapas Coropléticos y Puntos de Interés IV](#_Toc168975298)

[Integración con Folium para Visualizaciones Interactivas IV](#_Toc168975299)

[Bibliografia: IV](#_Toc168975300)

# Historia y Fundamentos de Matplotlib

Matplotlib es una biblioteca de visualización de datos para el lenguaje de programación Python, creada por John D. Hunter en 2003. Hunter se inspiró en MATLAB, un entorno de programación numérica, para desarrollar una herramienta similar que pudiera integrarse con Python, permitiendo a los usuarios crear gráficos de alta calidad y personalizables. Matplotlib es ampliamente utilizado en la comunidad científica y de análisis de datos debido a su versatilidad y capacidad para generar una amplia variedad de gráficos.

# Tipos Básicos de Gráficos

Gráficos de Líneas: Útiles para mostrar datos continuos a lo largo de un eje, como el tiempo. Se utilizan para identificar tendencias.

Gráficos de Barras: Ideales para comparar diferentes grupos o categorías. Pueden ser verticales u horizontales.

Histogramas: Muestran la distribución de un conjunto de datos al dividirlos en intervalos (bins), permitiendo visualizar la frecuencia de los valores dentro de cada intervalo.

Gráficos de Dispersión: Utilizados para mostrar la relación entre dos variables continuas. Cada punto representa un par de valores, permitiendo observar correlaciones y patrones.

# Personalización de gráficos.

Títulos y Etiquetas: Se pueden añadir títulos y etiquetas a los ejes para proporcionar contexto (plt.title(), plt.xlabel(), plt.ylabel()).

Leyendas: Se pueden añadir leyendas para identificar diferentes elementos en el gráfico (plt.legend()).

Estilos: Matplotlib ofrece varios estilos predefinidos y la capacidad de crear estilos personalizados (plt.style.use()).

# Introducción a Seaborn y Comparación con Matplotlib

Seaborn es una biblioteca de visualización de datos basada en Matplotlib que ofrece una interfaz de alto nivel para dibujar gráficos estadísticos atractivos y fáciles de interpretar. Seaborn simplifica la creación de gráficos complejos y estéticamente agradables mediante sus funciones preconfiguradas y paletas de colores. A diferencia de Matplotlib, Seaborn se centra más en la visualización de datos estadísticos.

# Gráficos de Densidad y de Violín para Distribuciones

Gráficos de Densidad: Utilizan la estimación de densidad de kernel (KDE) para mostrar la distribución de datos de manera suave.

Gráficos de Violín: Combinan la visualización de la distribución de datos y su resumen estadístico, mostrando densidades y cuartiles.

# Gráficos de Catplot y Factorplot para Categorización de Datos

Catplot: Es una función de Seaborn que facilita la creación de gráficos categóricos como box plots, bar plots, y violin plots con una API unificada.

Factorplot: Similar a Catplot, pero más antiguo y menos flexible. Se recomienda el uso de Catplot en versiones recientes de Seaborn.

Plotly es una biblioteca de gráficos interactivos que permite la creación de gráficos de alta calidad y completamente interactivos. Se integra fácilmente con Python y ofrece soporte para gráficos 2D y 3D, así como gráficos específicos para aplicaciones de ciencia de datos.

# Creación de Gráficos Interactivos

Gráficos de Dispersión y Líneas: Plotly facilita la creación de gráficos interactivos de dispersión y líneas utilizando su clase Scatter.

Mapas: Plotly proporciona herramientas para crear mapas interactivos utilizando datos geoespaciales con Scattergeo y Choropleth.

Uso de Dash para Crear Aplicaciones Web Interactivas

Dash es un framework de Python para construir aplicaciones web analíticas. Permite integrar gráficos de Plotly en aplicaciones web interactivas de manera sencilla, utilizando componentes React y Flask en su backend.

# Introducción a GeoPandas para Manipulación de Datos Geoespaciales

GeoPandas extiende las capacidades de pandas para permitir operaciones en datos geoespaciales. Facilita el manejo de datos de series temporales y la realización de operaciones espaciales como uniones, cálculos de áreas, y más.

# Creación de Mapas Coropléticos y Puntos de Interés

Mapas Coropléticos: Utilizan variaciones de color para mostrar datos geoespaciales agregados por regiones o áreas.

Puntos de Interés: Visualización de puntos específicos sobre un mapa que representan ubicaciones de interés.

# Integración con Folium para Visualizaciones Interactivas

Folium es una biblioteca de Python que facilita la creación de mapas interactivos utilizando Leaflet.js. Permite integrar datos geoespaciales en mapas interactivos fácilmente.

# Bibliografia:

Folium Contributors. (2021). Folium Documentation. <https://python-visualization.github.io/folium/>

Plotly Technologies Inc. (2015). Collaborative data science. Montreal, QC. <https://plotly.com>

Jordahl, K., et al. (2020). GeoPandas: Python tools for geographic data. Journal of Open Source Software, 5(20), 2237. <https://doi.org/10.21105/joss.02237>

Plotly Technologies Inc. (2018). Dash User Guide & Documentation. <https://dash.plotly.com/>

Plotly Technologies Inc. (2015). Collaborative data science. Montreal, QC. <https://plotly.com>

Waskom, M. L., et al. (2014). seaborn: V0.5.0 (November 2014). Zenodo. <http://doi.org/10.5281/zenodo.12710>