Librerías de visualización de datos con Python

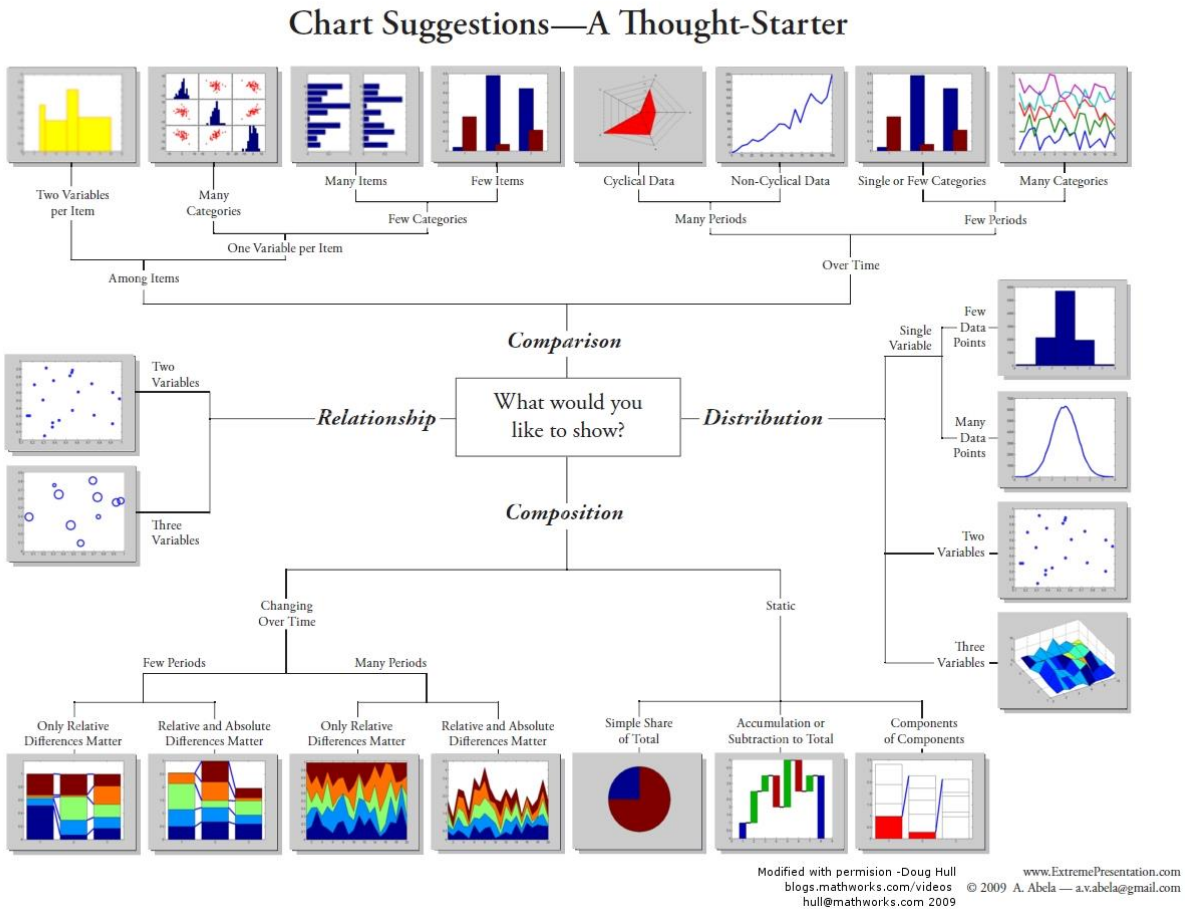
Las visualizaciones son una herramienta fundamental para entender y compartir ideas sobre los datos. La visualización correcta puede ayudar a expresar una idea central, o abrir un espacio para una más profunda investigación; con ella se puede conseguir que todo el mundo hable sobre un conjunto de datos, o compartir una visión sobre lo que los datos nos quieren decir.

Una buena visualización puede dar a quien la observa un sentido rico y amplio de un conjunto de datos. Puede comunicar los datos de manera precisa a la vez que expone los lugares en dónde se necesita más información o dónde una hipótesis no se sostiene. Por otra parte, la visualización nos proporciona un lienzo para aplicar nuestras propias ideas, experiencias y conocimientos cuando observamos y analizamos datos, permitiendo realizar múltiples interpretaciones.

Si como dice el dicho "una imagen vale más que mil palabras", un gráfico interactivo bien elegido entonces podría valer cientos de pruebas estadísticas.

¿Cómo elegir la visualización adecuada?

Una de las primeras preguntas que nos debemos realizar al explorar datos es ¿qué método de visualización es más efectivo?. Para intentar responder esta pregunta podemos utilizar la siguiente guía:



Como podemos ver, la guía se divide en cuatro categorías principales y luego se clasifican los distintos métodos de visualización que mejor representan cada una de esas categorías. Veamos un poco más en detalle cada una de ellas:

Distribuciones: En esta categoría intentamos comprender como los datos se distribuyen.

Se suelen utilizar en el comienzo de la etapa de exploración de datos, cuando queremos comprender las variables. Aquí también nos vamos a encontrar con variables de dos tipos cuantitativas y categóricas. Dependiendo del tipo y cantidad de variables, el método de visualización que vamos a utilizar.

Comparaciones: En esta categoría el objetivo es comparar valores a través de diferentes categorías y con el tiempo (tendencia). Los tipos de gráficos más comunes en esta categoría son los diagramas de barras para cuando estamos comparando elementos o categorías y los diagramas de puntos y líneas cuando comparamos variables cuantitativas.

Relaciones: Aquí el objetivo es comprender la relación entre dos o más variables. La visualización más utilizada en esta categoría es el gráfico de dispersión.

Composiciones: En esta categoría el objetivo es comprender como esta compuesta o distribuida una variable; ya sea a través del tiempo o en forma estática. Las visualizaciones más comunes aquí son los diagramas de barras y los gráficos de tortas.

Analizar y predecir eventos con los datos es una habilidad valiosa, pero también lo es poder transmitir estos hallazgos a otras personas que no tienen una familiaridad con los conjuntos de datos.

El tipo más común de visualización es un gráfico de barras simples, este es un tipo de visualización popular y de uso común para hacer una comparación entre valores y variedad de categorías. Pero hacer esta simple gráfica podemos utilizar hasta Excel, pero que pasa si queremos agregarles más valor a los datos, para ello contamos con varias librerías en Python para crear gráficas detalladas que represente lo que queremos explicar, a continuación, están las librerías más utilizadas:

Matplotlib 

A pesar de tener más de una década, sigue siendo la biblioteca más utilizada para visualización en la comunidad de Python. Debido a que fue la primera biblioteca de visualización de datos de Python, se construyeron muchas otras librerías encima o diseñadas para trabajar en conjunto con ella durante el análisis, algunas de ellas son Pandas y Seaborn.

Si bien matplotlib es bueno para obtener una idea de los datos, no es muy útil para crear gráficos con calidad de publicación rápida y fácilmente, inclusive ha sido criticado por sus estilos predeterminados, que tiene una sensación distintiva de los años 90.

Seaborn 

Seaborn se integra muy bien con Pandas y es otra biblioteca de software de código abierto para análisis y visualización de datos. Es una librería popular para hacer atractivos gráficos de datos estadísticos en Python.

Aprovecha el poder de matplotlib para crear gráficos hermosos en unas pocas líneas de código. La diferencia clave son los estilos predeterminados y las paletas de colores de Seaborn, que están diseñados para ser más estéticos y modernos, entre las características que ofrecen están:

- Varios temas incorporados que mejoran la estética predeterminada de matplotlib.
- Herramientas para elegir paletas de colores para hacer tramas hermosas que revelan patrones en sus datos.
- Funciones para visualizar distribuciones o para compararlas entre subconjuntos de datos.
- Herramientas que se ajustan y visualizan modelos de regresión lineal para diferentes tipos de variables independientes y dependientes.
- Funciones que visualizan matrices de datos y usan algoritmos de agrupamiento para descubrir la estructura en esas matrices.
- Una función para trazar los datos estadísticos de las series temporales con una estimación flexible y la representación de la incertidumbre en torno a la estimación.
- Abstracciones de alto nivel para estructurar grillas de parcelas que le permiten construir fácilmente visualizaciones complejas.

Bokeh 

Bokeh es una librería de fuente abierta y de uso gratuito para cualquier tipo de proyecto, es versátil y se integra muy bien con javascript. Esta es una librería interactiva que fue creada para los navegadores web modernos para visualizar parcelas altamente interactivas y aplicaciones de datos.

Con Bokeh se puede crear cualquier tipo de diagrama gráfico, incluidos tableros de instrumentos y variedad de gráficos. Por lo tanto, su fuerza reside en la capacidad de crear gráficos interactivos listos para la web, lo que hace que sea diferente a las dos librerías mencionadas anteriormente.

Bokeh proporciona tres interfaces con distintos niveles de control para adaptarse a diferentes tipos de usuarios:

- El nivel más alto es para crear gráficos rápidamente, incluye métodos para crear gráficos comunes como diagramas de barras, diagramas de cajas e histogramas.
- El nivel medio tiene la misma especificidad de matplotlib y le permite controlar los bloques de construcción básicos de cada gráfico.
- El nivel más bajo está dirigido a desarrolladores e ingenieros de software, no tiene valores predeterminados preestablecidos y requiere que defina cada elemento del gráfico.

Pygal

Es una parte de la librería de Python que exporta gráficos vectoriales en diferentes formas y estilos, es gratuita de código abierto y ha sido ampliamente utilizado debido a sus altas opciones de personalización y simplicidad al mismo tiempo. Las opciones para crear visualizaciones están muy abiertas e incluyen gráficos circulares, gráficos de barras, histogramas, mapas, entre otros, todo depende del aspecto y las sensaciones requeridos del gráfico.

Su principal diferenciador es la capacidad de generar gráficos SVG o gráficos vectoriales escalables. Mientras trabajes con conjuntos de datos pequeños, SVG le hará bien, pero si estás creando gráficos con cientos de miles de puntos de datos, tendrás problemas para renderizar y se volverán lentos.

Plotly

A esta librería también se le conoce como Plot.ly debido a su plataforma en línea. Es una herramienta de visualización en línea que se utiliza para el análisis de datos, gráficos científicos y otras visualizaciones, hay muchas visualizaciones interactivas de calidad profesional en línea que se crearon con este módulo.

Es diferente a las otras librerías de Python ya que es una herramienta interactiva en línea que crea las representaciones, por lo tanto, lo que se está creando con ella se publica en la web. Los gráficos creados son altamente interactivos con consejos de herramientas y variedad de otras opciones, como efecto de zoom, panorámica, selección escala automática, movimiento, reinicio, entre otros. Se modifica fácilmente haciendo clic en diferentes partes y parámetros del gráfico sin conocimiento de código.

Machine Learning sin una visualización adecuada son extremadamente difíciles de analizar. Python es una de las herramientas más innovadoras y populares para la visualización de datos, la buena noticia es que no hace falta mucho para crear una visualización en Python, ya que este lenguaje ha existido por más de veinte años y ha acumulado librerías exclusivas.

Hay múltiples herramientas y opciones para visualizar los datos, sin embargo, tener variedad de opciones complica el asunto y crea confusión para los usuarios. Identificar el método apropiado que se debe usar depende de los requisitos y expectativas del proyecto, la forma correcta es probar diferentes técnicas y entender cuál es apropiada:

- Matplotlib: es el método más simple para las representaciones básicas.
- Seaborn: es ideal para crear gráficos estadísticos visualmente atractivos que incluyen color.
- Bokeh: funciona muy bien para visualizaciones más complicadas e ideal para presentaciones interactivas basadas en web.
- Pygal: funciona bien para generar vectores y archivos interactivos, sin embargo, no tiene flexibilidad como otros métodos.
- Plotly: es la opción más útil y fácil para crear visualizaciones altamente interactivas basadas en la web.