1. Trực quan hóa dữ liệu có vai trò gì trong phân tích dữ liệu? Tại sao nó quan trọng trong khám phá dữ liệu (EDA)?

Vai trò:

-Giúp biểu diễn dữ liệu dưới dạng hình ảnh (biểu đồ, đồ thị, bản đồ nhiệt, v.v.) để con người dễ hiểu hơn so với bảng số liệu.

-Hỗ trợ phát hiện xu hướng, mẫu (pattern), mối quan hệ, và điểm bất thường (outlier) trong dữ liệu.

-Là bước quan trọng trong EDA (Exploratory Data Analysis) – giúp nhà phân tích hiểu cấu trúc, phân phối và chất -lượng dữ liệu trước khi xây dựng mô hình.

Tầm quan trọng:

-Tăng tốc độ hiểu dữ liệu.

-Giúp ra quyết định dựa trên dữ liệu (data-driven decision).

-Là nền tảng để chọn chiến lược xử lý và mô hình hóa dữ liệu.

1. Các loại biểu đồ phổ biến (như histogram, scatter plot, boxplot, bar chart) được sử dụng trong các trường hợp nào?

-Histogram (biểu đồ tần suất): Dùng để thể hiện phân phối của dữ liệu số liên tục, giúp ta thấy dữ liệu tập trung ở đâu, có lệch hay không và có giá trị ngoại lai hay không.

-Scatter Plot (biểu đồ phân tán): Dùng để biểu diễn mối quan hệ giữa hai biến số, thường được sử dụng để phát hiện xu hướng, tương quan hoặc các điểm bất thường trong dữ liệu.

-Boxplot (biểu đồ hộp): Thể hiện phân phối dữ liệu và các giá trị ngoại lai, đồng thời cho thấy trung vị, phần tư và độ biến động của dữ liệu. Rất hữu ích khi so sánh nhiều nhóm dữ liệu khác nhau.

-Bar Chart (biểu đồ cột): Dùng để so sánh giá trị giữa các nhóm hoặc danh mục, thích hợp với dữ liệu phân loại như giới tính, quốc gia hoặc loại sản phẩm.

-Line Chart (biểu đồ đường): Dùng để thể hiện xu hướng thay đổi của dữ liệu theo thời gian, ví dụ như doanh thu theo tháng hoặc nhiệt độ theo ngày.

-Pie Chart (biểu đồ tròn): Dùng để thể hiện tỷ lệ phần trăm của từng phần trong tổng thể, ví dụ như cơ cấu doanh thu theo sản phẩm.

1. Làm thế nào để chọn loại biểu đồ phù hợp với đặc điểm của dữ liệu (ví dụ: dữ liệu phân loại, dữ liệu số, dữ liệu thời gian)?

Để chọn loại biểu đồ phù hợp, trước hết cần xác định loại dữ liệu (phân loại, số, hay theo thời gian) và mục tiêu trực quan hóa (so sánh, phân phối, xu hướng, hay mối quan hệ).

-Với dữ liệu phân loại (ví dụ: giới tính, khoa, quốc gia), nên dùng biểu đồ cột (bar chart) hoặc biểu đồ tròn (pie chart) để so sánh hoặc thể hiện tỷ lệ giữa các nhóm.

-Với dữ liệu số liên tục (ví dụ: điểm số, giá trị, tuổi), nên dùng histogram để xem phân phối, hoặc boxplot để phát hiện giá trị ngoại lai và so sánh giữa các nhóm.

-Với dữ liệu thời gian (ví dụ: doanh thu theo tháng, nhiệt độ theo ngày), nên dùng line chart (biểu đồ đường) để thể hiện xu hướng và sự thay đổi theo thời gian.

-Khi muốn xem mối quan hệ giữa hai biến số, có thể dùng scatter plot (biểu đồ phân tán) để thấy rõ tương quan hoặc xu hướng giữa chúng.

-Nếu dữ liệu có nhiều biến, có thể dùng heatmap hoặc pairplot để thể hiện mối quan hệ phức tạp giữa các biến.

1. Sự khác biệt giữa các thư viện trực quan hóa trong Python như Matplotlib, Seaborn và Plotly là gì?

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Thư viện | Đặc điểm chính | Ưu điểm | Nhược điểm | Khi nên sử dụng |
| Matplotlib | Thư viện cơ bản, mạnh mẽ và linh hoạt nhất trong Python. | - Tùy chỉnh cao, hỗ trợ nhiều loại biểu đồ.  - Có thể xuất ra nhiều định dạng (PNG, PDF...). | - Cú pháp dài, cần nhiều lệnh để tạo biểu đồ đẹp.  - Giao diện không tự động tối ưu. | Khi cần tạo biểu đồ tĩnh hoặc xuất báo cáo in ấn. |
| Seaborn | Xây dựng trên Matplotlib, tối ưu cho trực quan hóa thống kê. | - Tạo biểu đồ nhanh, màu sắc đẹp.  - Tự động xử lý dữ liệu và thêm yếu tố thống kê (mean, median...). | - Ít linh hoạt hơn Matplotlib khi tùy chỉnh chi tiết. | Khi cần khám phá dữ liệu (EDA) nhanh và đẹp mắt. |
| Plotly | Hỗ trợ biểu đồ tương tác và trình bày trên web. | - Biểu đồ động, có thể zoom, hover, ẩn/hiện dữ liệu.  - Xuất HTML dễ dàng. | - Cấu hình phức tạp hơn khi cần tinh chỉnh. | Khi cần biểu đồ tương tác cho dashboard hoặc báo cáo web. |

1. Những nguyên tắc thiết kế nào cần tuân thủ để tạo ra một biểu đồ trực quan hóa dễ hiểu và hiệu quả?

Để biểu đồ dễ hiểu và chính xác, cần tuân thủ các nguyên tắc sau:

-Đơn giản, rõ ràng: Tránh chi tiết rối mắt, chỉ thể hiện nội dung chính.

-Chính xác: Giữ đúng tỷ lệ, không bóp méo dữ liệu.

-Đầy đủ nhãn và tiêu đề: Có tên biểu đồ, nhãn trục và chú thích rõ ràng.

-Màu sắc hợp lý: Dễ phân biệt, hài hòa, nhất quán giữa các nhóm dữ liệu.

-Tập trung thông điệp chính: Mỗi biểu đồ chỉ nên thể hiện một ý quan trọng.

1. Làm thế nào để tạo một biểu đồ đơn giản như histogram hoặc bar chart bằng Matplotlib? Bạn có thể chia sẻ đoạn code mẫu không?

Để tạo biểu đồ bằng Matplotlib, ta cần:

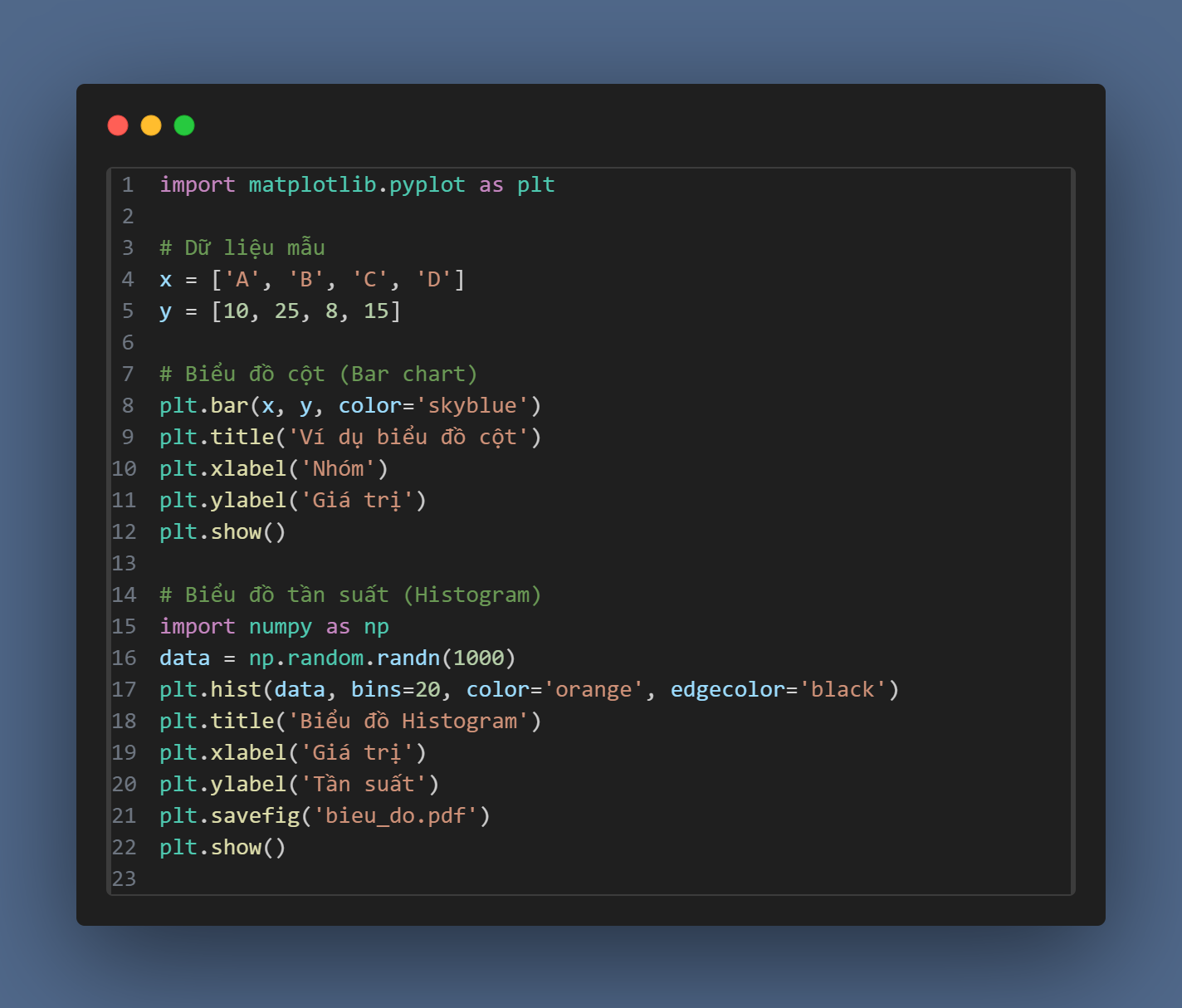
1.Nhập thư viện matplotlib.pyplot.

2.Chuẩn bị dữ liệu (danh sách hoặc mảng số).

3.Gọi hàm vẽ biểu đồ như plt.bar() cho biểu đồ cột hoặc plt.hist() cho biểu đồ tần suất.

4.Thêm tiêu đề, nhãn trục và hiển thị bằng plt.show().

Ví dụ code:



|  |  |
| --- | --- |
| output | output1 |

1. Làm thế nào để xuất biểu đồ từ Python ra các định dạng như PNG, PDF hoặc HTML để sử dụng trong báo cáo?

Để xuất biểu đồ từ Python, ta dùng:

Matplotlib: sử dụng lệnh plt.savefig("tenfile.png") hoặc plt.savefig("tenfile.pdf") để lưu biểu đồ dưới dạng hình ảnh hoặc file PDF.

Code ví dụ:



Plotly: sử dụng fig.write\_html("tenfile.html") để lưu biểu đồ tương tác dưới dạng file HTML.

Code ví dụ:

