

Chương 8: Các Kỹ thuật Trực quan hóa Nâng cao

Contents

- Các Kỹ thuật Trực quan hóa Nâng cao
- 8.1 Giới thiệu về Kỹ thuật Trực quan hóa Nâng cao
- 8.2 Trực quan hóa Dữ liệu 3D
- 8.4 Network Graphs (Biểu đồ Mạng)
- 8.5 Tùy chỉnh và Kết hợp Biểu đồ
- 8.6 Lưu trữ và Chia sẻ Biểu đồ
- 8.7 Bài tập Thực hành
- 8.8 Tổng kết chương 8

Các Kỹ thuật Trực quan hóa Nâng cao

Chương này sẽ giúp bạn tìm hiểu các kỹ thuật trực quan hóa nâng cao, cho phép khai thác sâu hơn các công cụ và phương pháp để phân tích và hiển thị dữ liệu phức tạp. Nội dung của chương được chia thành các phần sau:

8.1 Giới thiệu về Kỹ thuật Trực quan hóa Nâng cao

Mục tiêu: Hiểu lý do và bối cảnh sử dụng các kỹ thuật trực quan hóa nâng cao.

Lý thuyết:

- Trực quan hóa nâng cao giúp ta phân tích các dữ liệu lớn và phức tạp, từ đó nhận diện các mối quan hệ, xu hướng hoặc bất thường mà các phương pháp phân tích truyền thống không thể làm được.
- Các công cụ và thư viện như Plotly, Seaborn, Bokeh, Dash rất thích hợp cho việc tạo ra các đồ thị nâng cao với khả năng tương tác và tính linh hoạt cao.
- Các loại đồ thị nâng cao bao gồm các đồ thị 3D, Heatmaps, và Network Graphs. Những công cụ này giúp hiển thị các mối quan hệ trong dữ liệu từ nhiều góc độ khác nhau.

8.2 Trực quan hóa Dữ liệu 3D

Ứng dụng: Biểu đồ 3D giúp hiển thị các mối quan hệ giữa ba hoặc nhiều biến số. Đây là một công cụ quan trọng khi bạn cần phân tích các dữ liệu không gian hoặc mối quan hệ đa chiều.

Công cụ sử dụng: Plotly, Matplotlib (Axes 3D).

Ví dụ mã code:

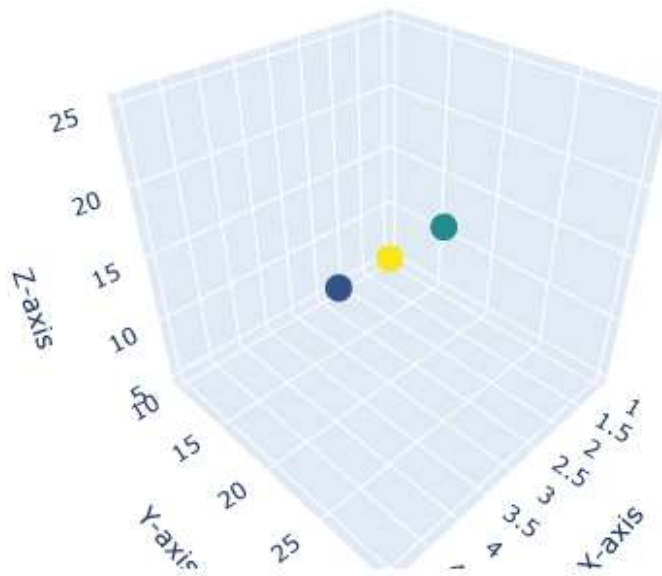
```
import plotly.graph_objects as go

x = [1, 2, 3, 4, 5]
y = [10, 20, 15, 25, 30]
z = [5, 15, 10, 20, 25]

fig = go.Figure(data=[go.Scatter3d(x=x, y=y, z=z, mode='markers',
                                   marker=dict(size=8, color=z, colorscale='Viridis'))])
fig.update_layout(title="Biểu đồ 3D", scene=dict(xaxis_title="X-axis", yaxis_title="Y-axis", zaxis_title="Z-axis"))
fig.write_image("chart.png")

from IPython.display import Image
Image("chart.png")
```

Biểu đồ 3D



8.3 Heatmaps và Biểu đồ Ma trận

Ứng dụng: Heatmaps rất hữu ích trong việc phân tích mối tương quan giữa các biến số trong dữ liệu ma trận, ví dụ như các dữ liệu hành vi người dùng hoặc số liệu kinh tế theo tháng, năm.

Công cụ sử dụng: Seaborn, Plotly.

Ví dụ mã code:

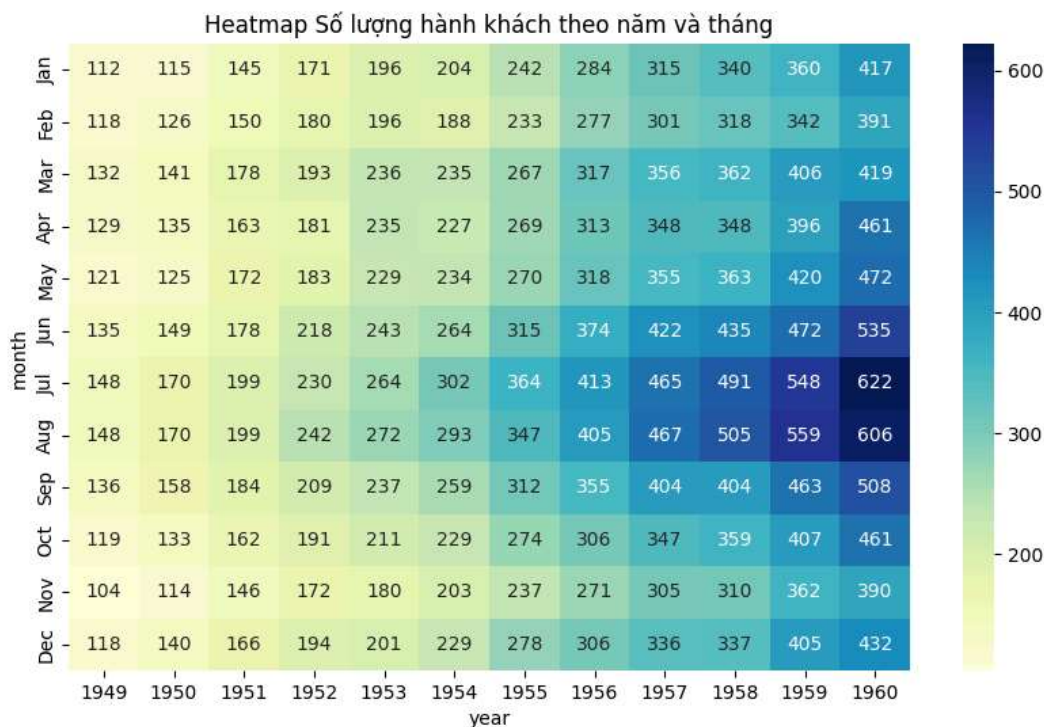
```
import seaborn as sns
import matplotlib.pyplot as plt
from IPython.display import Image

# Tải dữ liệu và tạo bảng pivot
data = sns.load_dataset("flights")
pivot_table = data.pivot(index="month", columns="year", values="passengers")

# Tạo heatmap
plt.figure(figsize=(10, 6)) # Tạo kích thước cho hình vẽ
sns.heatmap(pivot_table, annot=True, fmt="d", cmap="YlGnBu")
plt.title("Heatmap Số lượng hành khách theo năm và tháng")

# Lưu biểu đồ dưới dạng hình ảnh
plt.savefig("chart.png")

# Hiển thị ảnh đã lưu
plt.close() # Đóng biểu đồ sau khi lưu để không bị hiển thị lại
Image("chart.png")
```



Chú thích:

- `sns.heatmap` tạo một heatmap với bảng màu được chọn, cùng với giá trị số hành khách hiển thị trên từng ô.
- `plt.figure(figsize=(10, 6))` thay đổi kích thước của biểu đồ.
- `plt.title("Heatmap...")` tạo tiêu đề cho biểu đồ.

8.4 Network Graphs (Biểu đồ Mạng)

Ứng dụng: Biểu đồ mạng dùng để thể hiện các mối liên kết hoặc mạng lưới, ví dụ như mối quan hệ trong mạng xã hội hoặc trong các hệ thống dữ liệu phức tạp.

Công cụ sử dụng: NetworkX, Plotly.

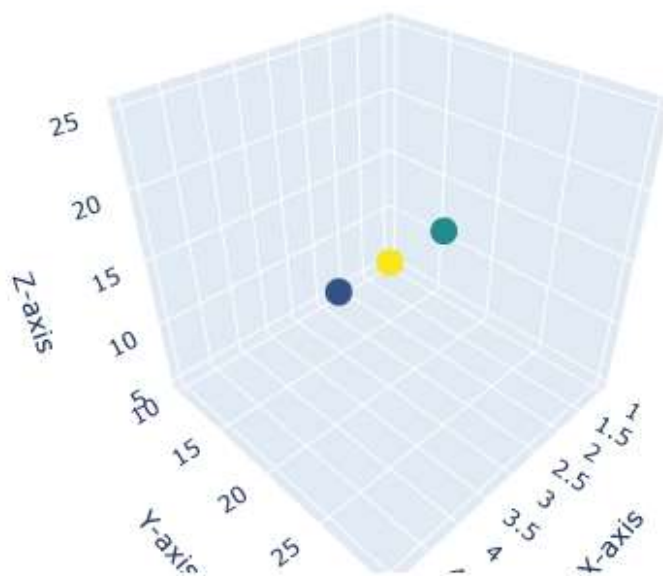
Ví dụ mã code:

```
import networkx as nx
import matplotlib.pyplot as plt

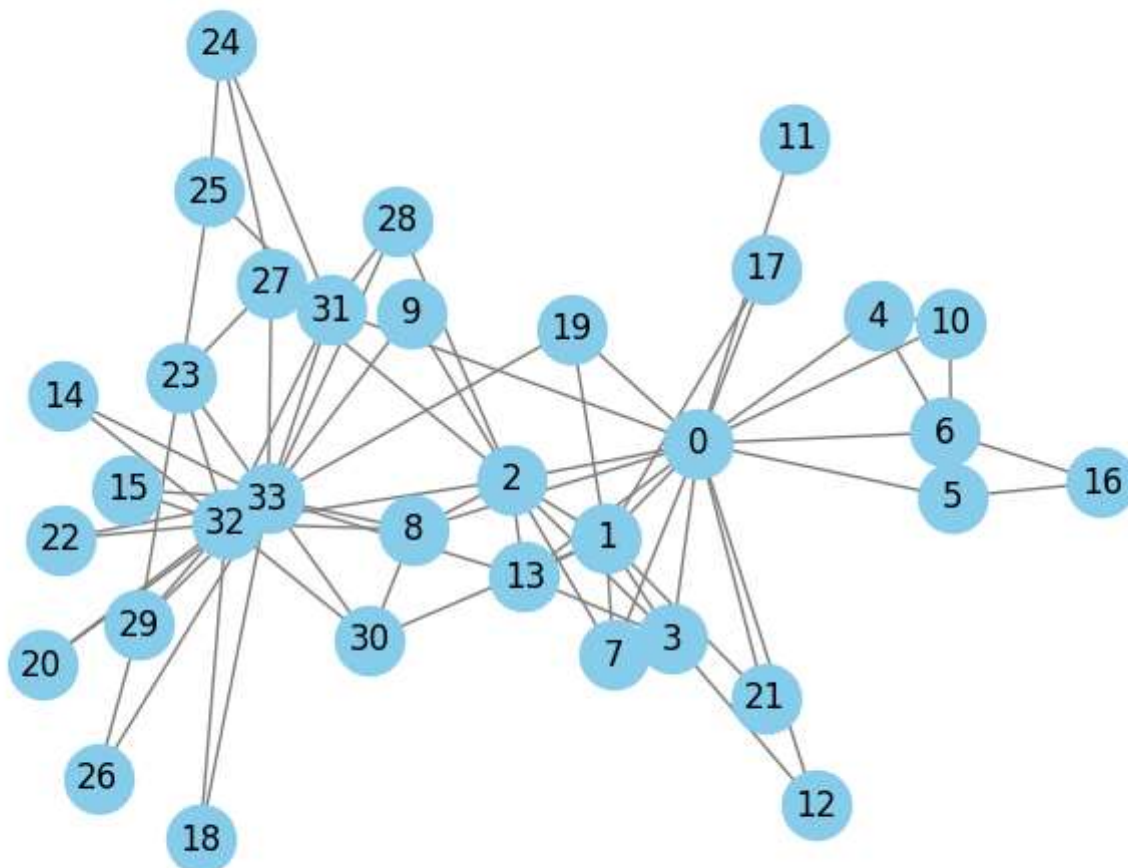
G = nx.karate_club_graph()
nx.draw(G, with_labels=True, node_color='skyblue', node_size=600, edge_color='gray')
plt.title("Network Graph - Karate Club")
fig.write_image("chart.png")

from IPython.display import Image
Image("chart.png")
```

Biểu đồ 3D



Network Graph - Karate Club



Chú thích:

- `nx.draw()` dùng để vẽ đồ thị. Các tham số điều chỉnh giao diện của đồ thị:
- `with_labels=True` giúp hiển thị các nhãn của các nút.
- `node_color='skyblue'` màu sắc của các nút.
- `node_size=600` điều chỉnh kích thước của các nút.
- `edge_color='gray'` xác định màu sắc của các cạnh nối giữa các nút.

8.5 Tùy chỉnh và Kết hợp Biểu đồ

Ứng dụng: Kết hợp các loại biểu đồ khác nhau trong cùng một hình giúp bạn phân tích dữ liệu từ nhiều góc độ. Ví dụ: kết hợp biểu đồ đường và biểu đồ cột để minh họa dữ liệu theo danh mục và theo thời gian

Ví dụ mã code:

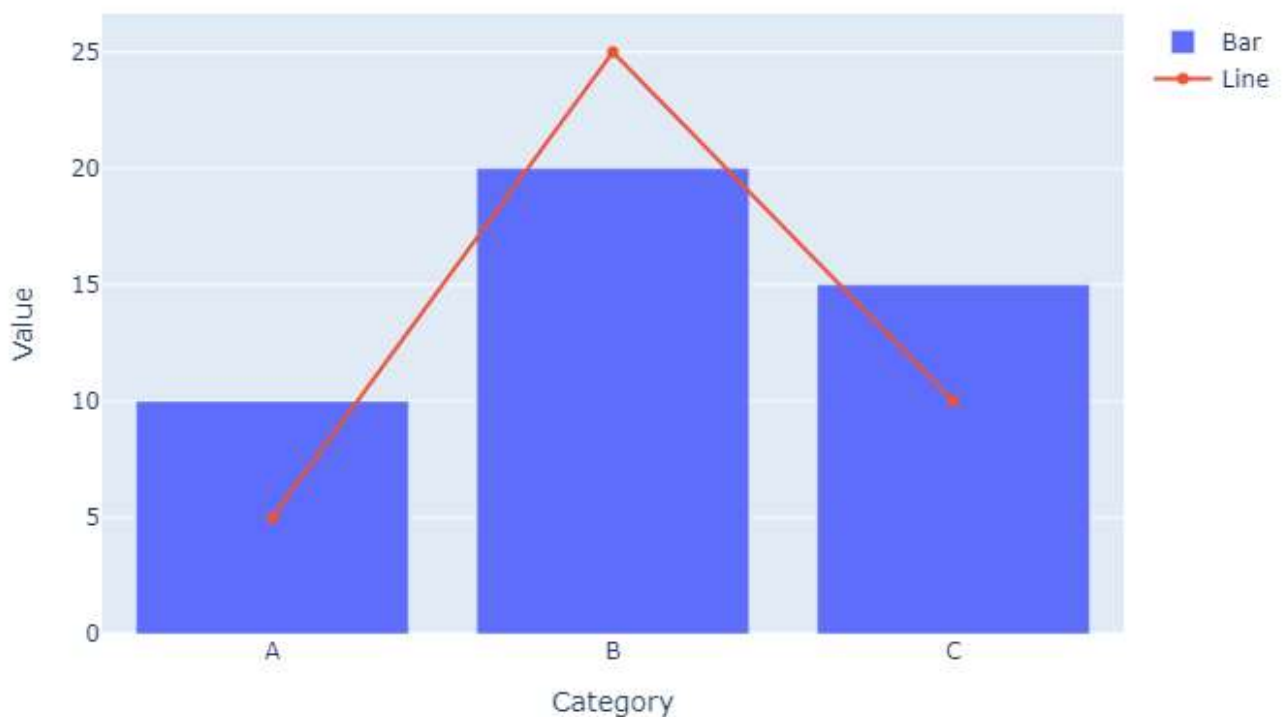
```
import plotly.graph_objects as go

fig = go.Figure()
fig.add_trace(go.Bar(x=["A", "B", "C"], y=[10, 20, 15], name="Bar"))
fig.add_trace(go.Scatter(x=["A", "B", "C"], y=[5, 25, 10], mode="lines+markers", name="Line"))

fig.update_layout(title="Kết hợp Biểu đồ", xaxis_title="Category", yaxis_title="Value")
fig.write_image("chart.png")

from IPython.display import Image
Image("chart.png")
```

Kết hợp Biểu đồ



Chú thích

- `fig = go.Figure()` tạo một đối tượng đồ thị mới để chứa các yếu tố đồ họa.
- `fig.add_trace(go.Bar(x=["A", "B", "C"], y=[10, 20, 15], name="Bar"))`: Thêm một biểu đồ cột, nơi các danh mục "A", "B", "C" sẽ có giá trị tương ứng là 10, 20 và 15.
- `name="Bar"` giúp gán tên cho biểu đồ cột trong phần chú thích.
- `fig.add_trace(go.Scatter(x=["A", "B", "C"], y=[5, 25, 10], mode="lines+markers", name="Line"))`: Thêm một biểu đồ đường, với các điểm dữ liệu tại "A", "B", "C" có giá trị 5, 25 và 10.

25, 10 tương ứng. Các điểm này sẽ được nối với nhau thành một đường, và các điểm cũng được đánh dấu bằng các ký tự (markers).

- mode="lines+markers" kết hợp cả đường và dấu chấm.

8.6 Lưu trữ và Chia sẻ Biểu đồ

Lưu trữ biểu đồ: Biểu đồ có thể được lưu dưới các định dạng phổ biến như PNG, HTML, PDF hoặc SVG, giúp bạn dễ dàng chia sẻ hoặc nhúng vào các báo cáo, trang web, hoặc tài liệu.

Chia sẻ trên nền tảng web:

Plotly cung cấp khả năng chia sẻ biểu đồ trực tuyến, cho phép nhúng biểu đồ vào các website hoặc chia sẻ qua các nền tảng như Dash.

8.7 Bài tập Thực hành

Bài tập 1: Tạo một biểu đồ 3D minh họa dữ liệu thời gian.

Bài tập 2: Tạo heatmap thể hiện mối tương quan giữa các biến số trong bộ dữ liệu.

Bài tập 3: Xây dựng một Dashboard nhỏ kết hợp hai biểu đồ (cột và đường).

8.8 Tổng kết chương 8

- **Nội dung:** Trong chương 8 này, chúng ta đã khám phá các kỹ thuật trực quan hóa nâng cao, bao gồm biểu đồ 3D, heatmaps và biểu đồ mạng, cùng với cách tùy chỉnh và kết hợp các biểu đồ để phân tích dữ liệu phức tạp. Chương 9 sẽ tiếp tục với việc tích hợp các kỹ thuật này vào các bài tập thực tế, giúp bạn hiểu rõ hơn cách áp dụng trực quan hóa vào các bài toán thực tế.