**Trả lời các câu hỏi ôn tập lý thuyết**

**1.1: Thống kê mô tả:**

*+ Thống kê mô tả là gì? Nó khác gì với thống kê suy luận (inferential statistics)?*

* **Thống kê mô tả (Descriptive Statistics)** là tập hợp các phương pháp được sử dụng để tóm tắt, tổ chức và trình bày các đặc điểm chính của một tập dữ liệu, thường là bằng cách sử dụng các chỉ số như trung bình, độ lệch chuẩn và các biểu đồ. Mục đích của nó là mô tả dữ liệu như nó vốn có.
* Ngược lại, **Thống kê suy luận (Inferential Statistics)** sử dụng dữ liệu mẫu để dự đoán hoặc suy luận về các đặc điểm của một quần thể lớn hơn. Nó bao gồm các kỹ thuật như kiểm định giả thuyết và xây dựng khoảng tin cậy.

+ *Các thước đo thống kê mô tả chính (ví dụ: trung bình, trung vị, phương sai, độ lệch chuẩn) được sử dụng để làm gì? Trong trường hợp nào thì nên dùng trung vị thay vì trung bình?*

Các thước đo thống kê mô tả chính được sử dụng để tóm tắt các khía cạnh khác nhau của dữ liệu:

* Các thước đo xu hướng trung tâm (như trung bình và trung vị) cho biết giá trị điển hình hoặc trung tâm của dữ liệu.
* Các thước đo độ phân tán (như phương sai và độ lệch chuẩn) cho biết mức độ phân tán hoặc biến thiên của dữ liệu xung quanh giá trị trung tâm.

Nên sử dụng trung vị (median) thay vì trung bình (mean) khi tập dữ liệu có giá trị ngoại lai (outliers) hoặc có phân bố bị lệch (skewed distribution) đáng kể. Trung vị ít bị ảnh hưởng bởi các giá trị cực trị hơn so với trung bình, do đó nó là thước đo tốt hơn cho xu hướng trung tâm trong những trường hợp này.

*+ Làm thế nào để xác định phân bố của một tập dữ liệu? Các loại phân bố phổ biến là gì (ví dụ: phân bố chuẩn, lệch trái, lệch phải)?*

Phân bố của một tập dữ liệu có thể được xác định bằng cách trực quan hóa dữ liệu (sử dụng biểu đồ histogram hoặc biểu đồ Q-Q) và tính toán các chỉ số thống kê như độ lệch (skewness) và độ nhọn (kurtosis).

Các loại phân bố phổ biến bao gồm:

* Phân bố chuẩn (Normal/Gaussian Distribution): Dữ liệu đối xứng, hình chuông, trung bình, trung vị và mode xấp xỉ nhau.
* Phân bố lệch phải (Right-Skewed/Positive Skew): Đuôi dài hơn về phía bên phải (giá trị lớn), trung bình > trung vị.
* Phân bố lệch trái (Left-Skewed/Negative Skew): Đuôi dài hơn về phía bên trái (giá trị nhỏ), trung bình < trung vị.

*+ Độ lệch chuẩn và phạm vi (range) có ý nghĩa gì trong việc đánh giá sự phân tán của dữ liệu?*

**Độ lệch chuẩn (Standard Deviation)** là thước đo độ phân tán phổ biến nhất, cho biết trung bình một điểm dữ liệu phân tán khỏi giá trị trung bình là bao nhiêu. Giá trị nhỏ cho thấy dữ liệu tập trung gần trung bình; giá trị lớn cho thấy dữ liệu phân tán rộng.

**Phạm vi (Range)** là sự khác biệt giữa giá trị lớn nhất và giá trị nhỏ nhất trong tập dữ liệu. Nó cung cấp một đánh giá nhanh về tổng độ rộng của dữ liệu, nhưng rất nhạy cảm với các giá trị ngoại lai vì nó chỉ phụ thuộc vào hai điểm dữ liệu.

*+ Sự khác biệt giữa các thước đo như Q1, Q2, Q3 trong biểu đồ hộp (boxplot) là gì?*

Trong biểu đồ hộp (boxplot), các thước đo Q1, Q2, Q3 là các tứ phân vị (quartiles) chia tập dữ liệu đã sắp xếp thành bốn phần bằng nhau:

* *Q1 (Tứ phân vị thứ nhất):* Điểm mà 25% dữ liệu nằm bên dưới nó.
* *Q2 (Tứ phân vị thứ hai) hay Trung vị (Median)*: Điểm mà 50% dữ liệu nằm bên dưới nó. Đây là đường kẻ ở giữa hộp.
* *Q3 (Tứ phân vị thứ ba)*: Điểm mà 75% dữ liệu nằm bên dưới nó.

Hộp trong boxplot biểu thị Phạm vi giữa các tứ phân vị (Interquartile Range - IQR), tức là khoảng từ Q1 đến Q3, bao gồm 50% dữ liệu ở giữa.

*+ Làm thế nào để xử lý giá trị thiếu (missing values) trước khi tính toán các chỉ số thống kê mô tả?*

Trước khi tính toán các chỉ số thống kê mô tả, việc xử lý giá trị thiếu (missing values) là cần thiết vì chúng có thể làm sai lệch kết quả hoặc gây ra lỗi tính toán. Các phương pháp phổ biến bao gồm:

1. **Loại bỏ (Deletion)**: Xóa các bản ghi (hàng) chứa giá trị thiếu (chỉ nên làm khi số lượng thiếu ít).
2. **Điền khuyết (Imputation)**: Thay thế giá trị thiếu bằng một ước tính, chẳng hạn như trung bình hoặc trung vị (cho dữ liệu số) hoặc mode (cho dữ liệu phân loại) của cột đó.
3. **Sử dụng mô hình (Model-based Imputation)**: Sử dụng các mô hình học máy để dự đoán giá trị thiếu.

*+ Bạn có thể giải thích cách đọc và diễn giải một biểu đồ histogram hoặc boxplot từ dữ liệu thực tế không?*

* **Histogram (Biểu đồ tần suất)**: Diễn giải hình dạng, xu hướng trung tâm và độ phân tán của một biến số. Hình dạng cho biết phân bố (chuẩn, lệch), vị trí đỉnh cao nhất cho biết Mode (giá trị phổ biến nhất), và độ rộng của biểu đồ cho biết sự biến thiên của dữ liệu.
* **Boxplot (Biểu đồ hộp)**: Diễn giải trung vị (đường kẻ giữa), độ phân tán (chiều dài hộp và râu), và giá trị ngoại lai (các điểm nằm ngoài râu). Một hộp dài cho thấy sự biến thiên cao hơn. Nếu trung vị không nằm ở giữa hộp, phân bố có thể bị lệch.

*+ Khi gặp một tập dữ liệu có giá trị ngoại lai (outliers), bạn sẽ xử lý chúng như thế nào trước khi thực hiện thống kê mô tả?*

Khi gặp giá trị ngoại lai (outliers), việc xử lý cần cân nhắc kỹ vì chúng có thể làm sai lệch các thước đo như trung bình và độ lệch chuẩn. Các bước xử lý phổ biến là:

1. **Xác định nguyên nhân**: Kiểm tra xem chúng có phải là lỗi nhập liệu hay một hiện tượng có ý nghĩa thực tế không.
2. **Giữ nguyên (nếu có ý nghĩa)**: Nếu giá trị ngoại lai là hợp lệ và quan trọng (ví dụ: một giao dịch lớn), bạn có thể chọn sử dụng trung vị và IQR (vốn ít nhạy cảm với ngoại lai) thay vì trung bình và độ lệch chuẩn.
3. **Biến đổi**: Sử dụng biến đổi toán học (ví dụ: logarit) để giảm tác động của ngoại lai.
4. **Loại bỏ/Giới hạn (winsorization/capping)**: Chỉ nên làm khi xác định chúng là lỗi hoặc khi chúng làm sai lệch nghiêm trọng mô hình. Giới hạn bằng cách thay thế ngoại lai bằng giá trị Q3 + 1.5\*IQR hoặc Q1 - 1.5\*IQR là cách tiếp cận phổ biến và ít gây mất thông tin hơn so với loại bỏ.

**1.2: Xử lý và trực quan hóa dữ liệu**

*+ 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)?*

**Trực quan hóa dữ liệu (Data Visualization)** đóng vai trò là "cầu nối" giúp chuyển đổi dữ liệu thô, phức tạp thành các hình ảnh trực quan dễ hiểu (biểu đồ, đồ thị).

Trong Khám phá dữ liệu (EDA - Exploratory Data Analysis), trực quan hóa đặc biệt quan trọng vì nó giúp:

1. Phát hiện xu hướng, mô hình, và mối quan hệ mà các con số thống kê đơn thuần có thể bỏ sót.
2. Nhận diện nhanh chóng giá trị ngoại lai, giá trị thiếu và sai sót trong dữ liệu.
3. Xác định hình dạng phân bố của các biến số.
4. Truyền đạt kết quả một cách hiệu quả và thuyết phục.

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

|  |  |
| --- | --- |
| **Biểu đồ** | **Ứng dụng** |
| Histogram | Phân bố của một biến số định lượng (số). |
| Bar Chart (Biểu đồ cột) | So sánh tần suất hoặc giá trị của các dữ liệu phân loại (categorical). |
| Scatter Plot (Biểu đồ phân tán) | Trực quan hóa mối quan hệ/tương quan giữa hai biến số định lượng. |
| Boxplot (Biểu đồ hộp) | So sánh phân bố (trung vị, độ phân tán, ngoại lai) của một biến số trên các nhóm phân loại khác nhau. |
| Line Chart (Biểu đồ đường) | Hiển thị xu hướng theo thời gian (dữ liệu chuỗi thời gian). |

*+ 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)?*

Việc lựa chọn biểu đồ phụ thuộc vào loại dữ liệu và mục tiêu phân tích:

* Dữ liệu Phân loại (Categorical): Dùng Bar Chart (so sánh nhóm) hoặc Pie Chart (thể hiện tỷ lệ phần trăm của tổng thể).
* Dữ liệu Số (Numerical):
* Phân bố: Dùng Histogram hoặc Boxplot.
* Mối quan hệ 2 biến số: Dùng Scatter Plot.
* Dữ liệu Thời gian (Time Series): Dùng Line Chart để theo dõi sự thay đổi theo thời gian.

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

|  |  |  |
| --- | --- | --- |
| **Thư viện** | **Đặc điểm** | **Phù hợp với** |
| Matplotlib | Nền tảng cơ bản và mạnh mẽ nhất. Cung cấp sự kiểm soát chi tiết ở cấp độ thấp (low-level). Đòi hỏi nhiều code để tạo biểu đồ phức tạp. | Tạo biểu đồ tùy chỉnh, điều chỉnh mọi yếu tố. |
| Seaborn | Xây dựng trên Matplotlib. Cung cấp giao diện cấp cao (high-level) để tạo các biểu đồ thống kê hấp dẫn, phức tạp một cách dễ dàng. Tích hợp tốt với Pandas. | Trực quan hóa thống kê chuyên sâu (boxplots, violin plots, heatmaps). |
| Plotly | Tạo biểu đồ tương tác (interactive), có thể nhúng vào web hoặc báo cáo HTML. Không chỉ là công cụ thống kê mà còn là công cụ trực quan hóa dữ liệu lớn. | Báo cáo, dashboard cần tính năng tương tác (zoom, hover). |

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

Để tạo một biểu đồ dễ hiểu và hiệu quả, hãy tuân thủ các nguyên tắc thiết kế sau:

1. **Tiêu đề rõ ràng**: Mô tả biểu đồ là gì và tại sao nó quan trọng.
2. **Nhãn trục rõ ràng**: Đảm bảo các trục X và Y được gắn nhãn và có đơn vị đo lường cụ thể.
3. **Loại bỏ sự lộn xộn**: Tránh các hiệu ứng 3D, bóng, hoặc lưới không cần thiết.
4. **Sử dụng màu sắc có ý nghĩ**a: Chỉ sử dụng màu để nhấn mạnh điểm chính hoặc phân biệt các danh mục, không nên dùng quá nhiều màu ngẫu nhiên.
5. **Bắt đầu trục Y từ 0 (cho Bar Chart)**: Nếu không, sự khác biệt có thể bị phóng đại sai lệch.

*+ 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?*

Đoạn code mẫu:

import matplotlib.pyplot as plt

import numpy as np

import pandas as pd

# Dữ liệu mẫu

data\_num = np.random.normal(loc=50, scale=10, size=1000) # Dữ liệu số

data\_cat = ['A', 'B', 'C', 'A', 'B', 'A', 'C', 'C', 'B', 'A'] # Dữ liệu phân loại

counts = pd.Series(data\_cat).value\_counts() # Đếm tần suất

# --- 1. Tạo HISTOGRAM ---

plt.figure(figsize=(8, 5))

plt.hist(data\_num, bins=30, color='skyblue', edgecolor='black')

plt.title('Histogram Phân bố Dữ liệu Số')

plt.xlabel('Giá trị')

plt.ylabel('Tần suất')

plt.grid(axis='y', alpha=0.5)

plt.show()

# --- 2. Tạo BAR CHART ---

plt.figure(figsize=(7, 5))

plt.bar(counts.index, counts.values, color=['red', 'green', 'blue'])

plt.title('Bar Chart Tần suất Dữ liệu Phân loại')

plt.xlabel('Danh mục')

plt.ylabel('Số lượng')

plt.show()

*+ 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 lưu biểu đồ đã tạo từ Matplotlib sang các định dạng tệp khác nhau (PNG, PDF, HTML), bạn sử dụng hàm plt.savefig() ngay sau khi tạo biểu đồ và trước khi gọi plt.show().

VD:

# Ví dụ xuất biểu đồ Histogram từ phần 14

plt.figure(figsize=(8, 5))

plt.hist(data\_num, bins=30, color='skyblue', edgecolor='black')

plt.title('Histogram Phân bố Dữ liệu Số')

plt.xlabel('Giá trị')

plt.ylabel('Tần suất')

# Xuất ra các định dạng khác nhau:

plt.savefig('histogram.png') # Định dạng PNG (Chất lượng cao, thường dùng)

plt.savefig('histogram.pdf') # Định dạng PDF (Thường dùng cho báo cáo)

# Đối với HTML, bạn thường dùng thư viện Plotly vì nó tạo biểu đồ tương tác,

# sau đó dùng plot.write\_html('biểu\_đồ.html') thay vì Matplotlib.

plt.show()

**1.3: Phân tích đơn biến và đa biến**

*+ Phân tích đơn biến (univariate analysis) là gì? Nó khác gì với phân tích hai biến (bivariate analysis) trong khám phá dữ liệu?*

**Phân tích đơn biến (Univariate Analysis)**

Định nghĩa: Quá trình phân tích một biến duy nhất để hiểu đặc điểm, phân bố, và xu hướng của nó mà không liên quan đến các biến khác.

Mục đích: Xác định phân bố (chuẩn, lệch), phát hiện ngoại lai, tính các thống kê cơ bản.

Ví dụ: Phân tích cột "Income" trong marketing\_campaign.csv để xem phân bố thu nhập.

**Phân tích hai biến (Bivariate Analysis)**

Định nghĩa: Quá trình phân tích mối quan hệ giữa hai biến để khám phá xu hướng hoặc tương quan.

Mục đích: Đánh giá mức độ liên kết (dương, âm, hoặc không có) giữa hai biến.

Ví dụ: Phân tích mối quan hệ giữa "Income" và "MntWines" để xem thu nhập ảnh hưởng đến chi tiêu cho rượu.

**Sự khác biệt**

Số lượng biến: Đơn biến (1 biến), hai biến (2 biến).

Mục tiêu: Đơn biến mô tả, hai biến khám phá mối quan hệ.

Phương pháp: Đơn biến dùng histogram/boxplot; hai biến dùng scatter plot/heatmap.

## *+ Các thước đo thống kê nào thường được sử dụng trong phân tích đơn biến?*

* **Trung bình (Mean)**: .Nhạy với ngoại lai (ví dụ: trung bình "Income").
* **Trung vị (Median)**: Giá trị giữa khi sắp xếp dữ liệu, không bị ảnh hưởng bởi ngoại lai (ví dụ: trung vị "Income").
* **Mode (Mốt)**: Giá trị xuất hiện nhiều nhất (ví dụ: mode của "Education").
* **Độ lệch chuẩn (Standard Deviation)**: . Đo phân tán (ví dụ: độ lệch chuẩn "MntWines").
* **Phạm vi (Range)**: Range= max(x) - min(x) (ví dụ: phạm vi "culmen\_length\_mm").
* **Tứ phân vị (Quartiles) và IQR**: Q1, Q2, Q3; IQR = Q3 - Q1 (ví dụ: IQR "new\_cases").
* **Phân vị (Percentiles)**: Giá trị dưới một phần trăm nhất định (ví dụ: phân vị 75 của "Income").

**Ứng dụng**: Sử dụng pandas (.mean(), .median(), .std()) hoặc numpy/scipy.stats như trong tài liệu (trang 2).

## *+ Trong phân tích hai biến, làm thế nào để xác định mối quan hệ giữa hai biến?*

### Tương quan (Correlation)

* **Định nghĩa**: Đo lường mức độ tuyến tính (hệ số Pearson: -1 đến +1).
* **Công thức**: wps
* **Ví dụ**: Tương quan giữa "Income" và "MntWines".

### Hiệp biến (Covariance)

* **Định nghĩa**: Đo hướng và cường độ mối quan hệ tuyến tính, phụ thuộc đơn vị.
* **Công thức**: wps​.
* **Ví dụ**: Hiệp biến giữa "culmen\_length\_mm" và "body\_mass\_g".

### Phân tích nhân quả (Causality)

* **Cách xác định**: Không thể từ dữ liệu quan sát (chỉ tương quan); cần thử nghiệm hoặc mô hình nhân quả (DAG).
* **Ví dụ**: "Income" cao có dẫn đến "MntWines" cao không? Cần kiểm tra yếu tố khác.

**Phương pháp**: Biểu đồ (scatter), thống kê (tương quan Pearson), mô hình (hồi quy).

## *+ Sự khác biệt giữa tương quan (correlation) và hiệp biến (covariance) trong phân tích hai biến?*

* **Định nghĩa**:
  + **Hiệp biến**: Đo hướng và cường độ, giá trị phụ thuộc đơn vị.
  + **Tương quan**: Chuẩn hóa hiệp biến, giá trị từ -1 đến +1.
* **Công thức**:
  + Covariance: wps
  + Correlation:wps
* **Sự khác biệt**:
  + **Quy mô**: Hiệp biến thay đổi theo đơn vị (ví dụ: 500,000); tương quan cố định [-1, 1].
  + **So sánh**: Tương quan dễ so sánh giữa cặp biến; hiệp biến không.
  + **Dễ hiểu**: Tương quan (0.7) rõ ràng hơn hiệp biến.
* **Ví dụ**: Cov(Income, MntWines) = 500,000, r = 0.6.

## *+ Khi nào nên sử dụng biểu đồ trực quan hóa trong phân tích đơn biến so với phân tích hai biến?*

### Phân tích đơn biến

* **Khi nào dùng**: Để hiểu phân bố, xu hướng trung tâm, hoặc ngoại lai của một biến.
* **Biểu đồ**:
  + **Histogram**: Phân bố (ví dụ: "Income").
  + **Boxplot**: Ngoại lai và tứ phân vị (ví dụ: "new\_cases").
  + **Pie chart/Bar chart**: Biến phân loại (ví dụ: "species").

### Phân tích hai biến

* **Khi nào dùng**: Để khám phá mối quan hệ hoặc xu hướng giữa hai biến.
* **Biểu đồ**:
  + **Scatter plot**: Mối quan hệ tuyến tính (ví dụ: "Income" vs "MntWines").
  + **Heatmap**: Ma trận tương quan.
  + **Boxplot/Violin với hue**: So sánh nhóm (ví dụ: "Income" theo "Education").

### Quy tắc

* Đơn biến mô tả, hai biến liên kết. Chọn dựa trên loại biến và câu hỏi.

## *+ Đoạn code mẫu để tạo biểu đồ scatter plot hoặc heatmap?*

**Scatter Plot:**

import pandas as pd

import seaborn as sns

import matplotlib.pyplot as plt

marketing\_data = pd.read\_csv("marketing\_campaign.csv", skipinitialspace=True)

marketing\_data["Income"] = pd.to\_numeric(marketing\_data["Income"], errors='coerce')

marketing\_data["Income"] = marketing\_data["Income"].fillna(marketing\_data["Income"].median())

plt.figure(figsize=(10, 6))

sns.scatterplot(data=marketing\_data, x="Income", y="MntWines", hue="Education")

plt.title("Mối quan hệ giữa Income và MntWines theo Education")

plt.xlabel("Thu nhập (Income)")

plt.ylabel("Chi tiêu cho rượu (MntWines)")

plt.legend(title="Trình độ học vấn")

plt.show()

**Heatmap:**

plt.figure(figsize=(10, 8))

numeric\_data = marketing\_data.select\_dtypes(include=['int64', 'float64'])

correlation\_matrix = numeric\_data.corr()

sns.heatmap(correlation\_matrix, annot=True, cmap="coolwarm", vmin=-1, vmax=1)

plt.title("Ma trận tương quan giữa các biến số")

plt.show()

## *+ Làm thế nào để trực quan hóa mối quan hệ giữa một biến số và một biến phân loại?*

### Boxplot:

plt.figure(figsize=(12, 6))

sns.boxplot(data=marketing\_data, x="Education", y="Income")

plt.title("Phân bố thu nhập theo trình độ học vấn")

plt.xlabel("Trình độ học vấn")

plt.ylabel("Thu nhập (Income)")

plt.xticks(rotation=45)

plt.show()

**Violin Plot:**

plt.figure(figsize=(12, 6))

sns.violinplot(data=marketing\_data, x="Education", y="Income")

plt.title("Phân bố thu nhập theo trình độ học vấn (Violin Plot)")

plt.xlabel("Trình độ học vấn")

plt.ylabel("Thu nhập (Income)")

plt.xticks(rotation=45)

plt.show()