#### **ỦY BAN NHÂN DÂN THÀNH PHỐ HỒ CHÍ**

#### **MINH TRƯỜNG ĐẠI HỌC SÀI GÒN**

**KHOA TOÁN - ỨNG DỤNG**

#### 



**BÁO CÁO**

**KHAI PHÁ DỮ LIỆU**

Phân tích khám phá dữ liệu (EDA)

**Giáo viên hướng dẫn:** Đỗ Như Tài

**Thành viên:** Lê Phương Vy

Nguyễn Thị Minh Thùy

Đỗ Ngọc Như Ý

Nguyễn Văn Hậu

*TP. Hồ Chí Minh, Tháng 10, năm 2025*

**DANH SÁCH PHÂN CÔNG THÀNH VIÊN**

|  |  |  |
| --- | --- | --- |
| **Tên thành viên** | **MSSV** | **Công việc** |
| **Lê Phương Vy** | 3123580064 | Nhóm trưởng  Phân công công việc  Phần 1 code  Báo cáo |
| **Nguyễn Văn Hậu** | 3123580012 | Phần 3 code  Báo cáo |
| **Nguyễn Thị Minh Thùy** | 3123580049 | Phần 2 code  Báo cáo |
| **Đỗ Ngọc Như Ý** | 3123580067 | Phần 1 code  Báo cáo  Kết luận |

**LỜI MỞ ĐẦU**

Trong thời đại công nghệ số hiện nay, dữ liệu đóng vai trò vô cùng quan trọng và được xem là “nguồn tài nguyên mới” của thế giới. Việc khai phá dữ liệu (Data Mining) giúp con người phát hiện ra các quy luật, xu hướng và mối quan hệ tiềm ẩn trong dữ liệu, từ đó hỗ trợ việc ra quyết định một cách hiệu quả hơn trong nhiều lĩnh vực như kinh doanh, y tế, giáo dục hay khoa học.

Thông qua môn học Khai phá dữ liệu, chúng em được trang bị kiến thức và kỹ năng cần thiết để hiểu, xử lý và phân tích dữ liệu một cách có hệ thống. Đặc biệt, việc thực hành trên Jupyter Lab giúp chúng em làm quen với các thư viện Python mạnh mẽ như *Pandas, Matplotlib, Seaborn,* và *NumPy*, qua đó dễ dàng thực hiện các bước phân tích khám phá dữ liệu (Exploratory Data Analysis - EDA), trực quan hóa kết quả và rút ra các nhận định có ý nghĩa.

Báo cáo này được thực hiện nhằm tổng hợp lại toàn bộ quá trình nhóm em đã làm trong bài Thực hànhPhân tích khám phá dữ liệu, bao gồm các bước: thống kê mô tả, xử lý và trực quan hóa dữ liệu, cùng với phân tích đơn biến và hai biến. Thông qua bài thực hành, nhóm em mong muốn thể hiện sự hiểu biết và khả năng vận dụng kiến thức đã học vào thực tế.

**LỜI CẢM ƠN**

Nhóm chúng em xin gửi lời cảm ơn chân thành đến thầy **Đỗ Như tài**  – giảng viên phụ trách môn **Khai phá dữ liệu** đã tận tình giảng dạy và hướng dẫn chúng em trong suốt quá trình học.  
Thông qua các buổi học và thực hành trên **Jupyter Lab**, thầy đã giúp chúng em hiểu rõ hơn về quy trình khai phá dữ liệu, từ bước tiền xử lý, phân tích khám phá dữ liệu (EDA) cho đến trực quan hóa và rút ra nhận xét từ dữ liệu thực tế.

Nhờ sự giảng dạy nhiệt tình, dễ hiểu và những ví dụ minh họa cụ thể của thầy, nhóm không chỉ nắm vững kiến thức lý thuyết mà còn biết cách áp dụng vào bài thực hành một cách hiệu quả.  
Môn học này đã giúp chúng em rèn luyện thêm kỹ năng lập trình với Python, tư duy phân tích dữ liệu và cách trình bày kết quả khoa học.

Bài báo cáo tuy đã được nhóm cố gắng hoàn thiện, nhưng chắc chắn vẫn còn những thiếu sót. Nhóm rất mong nhận được những góp ý quý báu từ thầy để rút kinh nghiệm và cải thiện hơn trong các bài làm sau.

Nhóm chúng em xin chân thành cảm ơn thầy!

**MỤC LỤC**

[A. Thống kê mô tả 1](#_Toc211283920)

[**I. Ôn tập lý thuyết** 1](#_Toc211283921)

[**1. Thống kê mô tả là gì? Nó khác gì với thống kê suy luận (inferential statistics)?** 1](#_Toc211283922)

[**2. Các thước đo mô tả chính dùng để làm gì? Khi nào nên dùng trung vị thay vì trung bình?** 1](#_Toc211283923)

[**3. Làm thế nào để xác định phân bố của một tập dữ liệu? Các loại phân bố phổ biến là gì?** 2](#_Toc211283924)

[**4. Độ lệch chuẩn và phạm vi (range) có ý nghĩa gì trong việc đánh giá sự phân tán của dữ liệu** 3](#_Toc211283925)

[**5.** **Sự khác biệt giữa các thước đo Q1, Q2, Q3 trong biểu đồ hộp (boxplot) là gì?** 4](#_Toc211283926)

[**6.** **Làm thế nào để xử lý giá trị thiếu (missing values) trước khi tính toán các chỉ số thống kê mô tả?** 4](#_Toc211283927)

[**7.** **Cách đọc và diễn giải biểu đồ histogram hoặc boxplot từ dữ liệu thực tế?** 5](#_Toc211283928)

[**8.** **Khi gặp dữ liệu có ngoại lai (outliers), nên xử lý thế nào trước khi thống kê mô tả?** 5](#_Toc211283929)

[**II. Bài tập ứng dụng** 6](#_Toc211283930)

[**1.** **Khám phá dữ liệu COVID** 6](#_Toc211283931)

[**2.** **Khám phá và xử lý dữ liệu Marketing Campaign** 14](#_Toc211283932)

[**3.** **Thống kê mô tả dữ liệu về phân loại chất lượng rượu đỏ** 20](#_Toc211283933)

[**4.** **Thống kê mô tả dữ liệu bệnh đái tháo đường** 27](#_Toc211283934)

[**B. Xử lý và trực quan hóa dữ liệu** 36](#_Toc211283935)

[**I. Ôn tập lý thuyết** 36](#_Toc211283936)

[**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)?** 36](#_Toc211283937)

[**2. Các loại biểu đồ phổ biến (như histogram, scatter plot, boxplot, bar chart) được sử dụng trong các trường hợp nào?** 36](#_Toc211283938)

[**3. 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)?** 37](#_Toc211283939)

[**4. Sự khác biệt giữa các thư viện trực quan hóa trong Python như Matplotlib, Seaborn và Plotly là gì?** 37](#_Toc211283940)

[**5. Những nguyên tắc thiết kế nào cần tuân thủ để tạo ra một biểu đồ trực quan hóa dễ hiểu và hiệu quả?** 37](#_Toc211283941)

[**6. 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?** 38](#_Toc211283942)

[**7. Làm thế nào để xuất biểu đồ từ Python ra các định dạng như PNG, PDF hoặc HTML để sử dụng trong báo cáo?** 39](#_Toc211283943)

[**II. Bài tập ứng dụng** 40](#_Toc211283944)

[**1.** **Xử lý và trực quan hóa dữ liệu về giá nhà** 40](#_Toc211283945)

[**2.** **Trực quan hóa dữ liệu về phân loại chất lượng rượu đỏ** 48](#_Toc211283946)

[**3.** **Trực quan hóa dữ liệu về bệnh tiểu đường** 57](#_Toc211283947)

[C. Phân tích đơn biến và hai biến 66](#_Toc211283948)

[**I. Ôn tập lý thuyết** 66](#_Toc211283949)

[**1. Phân tích đơn biến (univariate analysis) là gì? Nó khác gì với phân tích hai biến (bivariate analysis) trong khám phá dữ liệu?** 66](#_Toc211283950)

[**2. Các thước đo thống kê nào thường được sử dụng trong phân tích đơn biến (ví dụ: trung bình, trung vị, mode, độ lệch chuẩn)?** 67](#_Toc211283951)

[**3. Trong phân tích hai biến, làm thế nào để xác định mối quan hệ giữa hai biến (ví dụ: tương quan, nhân quả)?** 67](#_Toc211283952)

[**4. Sự khác biệt giữa tương quan (correlation) và hiệp biến (covariance) trong phân tích hai biến là gì?** 68](#_Toc211283953)

[**5. Khi nào nên sử dụng biểu đồ trực quan hóa trong phân tích đơn biến so với phân tích hai biến?** 68](#_Toc211283954)

[**6. Đoạn code mẫu để tạo biểu đồ scatter plot hoặc heatmap để phân tích mối quan hệ giữa hai biến?** 69](#_Toc211283955)

[**7. Làm thế nào để trực quan hóa mối quan hệ giữa một biến số và một biến phân loại bằng biểu đồ boxplot hoặc violin plot trong Python?** 70](#_Toc211283956)

[**II. Bài làm ứng dụng** 71](#_Toc211283957)

[**1.** **Phân tích dữ liệu đơn biến trên dữ liệu chim cánh cụt** 71](#_Toc211283958)

[**2.** **Phân tích dữ liệu đơn biến trên dữ liệu về giá nhà** 77](#_Toc211283959)

[**3.** **Phân tích dữ liệu hai biến trên dữ liệu chim cánh cụt** 82](#_Toc211283960)

[D. Tóm tắt thực hành 87](#_Toc211283961)

# **A. Thống kê mô tả**

## **I. Ôn tập lý thuyết**

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

Thống kê mô tả (Descriptive Statistics) là phương pháp được sử dụng để tóm tắt, mô tả và trình bày dữ liệu đã thu thập nhằm giúp người đọc hiểu rõ đặc điểm cơ bản của tập dữ liệu. Các chỉ số thường dùng trong thống kê mô tả bao gồm giá trị trung bình, trung vị, mốt, phương sai, độ lệch chuẩn, cũng như các biểu đồ trực quan như biểu đồ cột, biểu đồ tròn hoặc histogram. Thống kê mô tả chỉ phản ánh đặc điểm của dữ liệu mẫu mà không đưa ra bất kỳ suy luận hay kết luận nào cho tổng thể.

Ngược lại, thống kê suy luận (Inferential Statistics) là nhánh thống kê sử dụng dữ liệu mẫu để đưa ra suy luận, ước lượng hoặc kiểm định giả thuyết cho toàn bộ tổng thể. Phương pháp này thường áp dụng các kỹ thuật như ước lượng tham số, kiểm định giả thuyết, phân tích hồi quy và ANOVA. Thông qua thống kê suy luận, người nghiên cứu có thể dự đoán hoặc rút ra kết luận cho quần thể dựa trên thông tin thu thập từ mẫu.

Tóm lại, thống kê mô tả tập trung vào việc trình bày thông tin của dữ liệu hiện có, trong khi thống kê suy luận giúp mở rộng kết quả từ mẫu sang tổng thể, từ đó hỗ trợ quá trình ra quyết định và nghiên cứu một cách khoa học hơn.

### **2. Các thước đo mô tả chính dùng để làm gì? Khi nào nên dùng trung vị thay vì trung bình?**

Các thước đo thống kê mô tả được sử dụng nhằm tóm tắt, mô tả và trình bày đặc điểm cơ bản của dữ liệu trong nghiên cứu. Chúng giúp người phân tích hiểu rõ xu hướng trung tâm, mức độ biến động và hình dạng phân bố của tập dữ liệu, từ đó hỗ trợ việc diễn giải kết quả một cách chính xác hơn.

Cụ thể, có ba nhóm thước đo chính:

* **Thước đo xu hướng trung tâm (Measures of Central Tendency):** Giúp xác định giá trị đại diện cho dữ liệu, bao gồm **trung bình (mean)**, **trung vị (median)** và **mốt (mode)**.
* **Thước đo độ phân tán (Measures of Dispersion):** Dùng để phản ánh mức độ các giá trị phân tán quanh trung tâm, bao gồm khoảng (range), phương sai (variance) và độ lệch chuẩn (standard deviation).
* **Thước đo hình dạng phân bố (Shape Measures):** Cho biết đặc điểm hình dạng của dữ liệu, như **độ lệch (skewness)** cho biết dữ liệu lệch về phía nào, và **độ nhọn (kurtosis)** thể hiện mức độ tập trung của dữ liệu.

Nhờ các thước đo này, nhà nghiên cứu có thể nhanh chóng nhận diện đặc điểm tổng thể của dữ liệu, phát hiện xu hướng hoặc giá trị bất thường, và chuẩn bị cho các bước phân tích sâu hơn như thống kê suy luận hay mô hình hồi quy.

Trung vị nên được sử dụng khi dữ liệu có ngoại lệ hoặc bị lệch (skewed), vì trung bình sẽ bị ảnh hưởng mạnh bởi các giá trị quá lớn hoặc quá nhỏ.

### **3. Làm thế nào để xác định phân bố của một tập dữ liệu? Các loại phân bố phổ biến là gì?**

Có thể xác định phân bố dữ liệu bằng cách:

* Vẽ biểu đồ Histogram hoặc Density Plot để quan sát hình dạng phân phối.
* Dùng các chỉ số thống kê như skewness (độ lệch) và kurtosis (độ nhọn).

Các loại phân bố phổ biến:

* Phân bố chuẩn (Normal distribution): dữ liệu đối xứng quanh trung bình.
* Lệch trái (Left-skewed): đuôi dữ liệu nghiêng về bên trái.
* Lệch phải (Right-skewed): đuôi dữ liệu nghiêng về bên phải.

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

Ý nghĩa của độ lệch chuẩn và phạm vi trong đánh giá sự phân tán của dữ liệu

Độ lệch chuẩn (Standard Deviation) và phạm vi (Range) là hai thước đo quan trọng dùng để đánh giá mức độ phân tán của dữ liệu quanh giá trị trung tâm.

* **Phạm vi (Range):** Là hiệu giữa giá trị lớn nhất và nhỏ nhất trong tập dữ liệu. Chỉ số này cho biết khoảng biến thiên tổng thể của dữ liệu, giúp nhận biết dữ liệu trải rộng hay tập trung. Tuy nhiên, phạm vi dễ bị ảnh hưởng bởi các giá trị ngoại lai (outliers), nên chỉ phản ánh mức độ phân tán một cách tổng quát.
* **Độ lệch chuẩn (Standard Deviation):** Cho biết mức độ các giá trị trong tập dữ liệu dao động quanh giá trị trung bình. Độ lệch chuẩn càng lớn, dữ liệu càng phân tán; ngược lại, độ lệch chuẩn nhỏ cho thấy dữ liệu tập trung gần trung bình hơn. Đây là thước đo ổn định và được sử dụng phổ biến hơn phạm vi vì nó xem xét toàn bộ các giá trị trong dữ liệu chứ không chỉ hai điểm cực trị.

Tóm lại,phạm vi giúp hình dung nhanh mức độ trải rộng của dữ liệu, trong khi độ lệch chuẩn cung cấp thông tin chính xác hơn về mức độ biến động của các giá trị xung quanh trung bình, giúp nhà phân tích đánh giá được độ ổn định và độ đồng đều của dữ liệu.

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

Trong biểu đồ hộp (boxplot), các thước đo Q1, Q2 và Q3 đại diện cho ba tứ phân vị (quartiles), được dùng để mô tả sự phân bố và mức độ phân tán của dữ liệu.

* Q1 (Quartile 1 – Tứ phân vị thứ nhất): Là giá trị mà 25% số liệu có giá trị nhỏ hơn hoặc bằng Q1. Q1 thể hiện ranh giới của nhóm dữ liệu thấp, giúp nhận biết phần dữ liệu có giá trị nhỏ nhất.
* Q2 (Quartile 2 – Tứ phân vị thứ hai): Còn gọi là trung vị (Median), chia tập dữ liệu thành hai nửa bằng nhau. Q2 cho biết giá trị trung tâm của dữ liệu, là vị trí mà 50% số liệu nằm dưới và 50% nằm trên.
* Q3 (Quartile 3 – Tứ phân vị thứ ba): Là giá trị mà 75% số liệu nhỏ hơn hoặc bằng Q3. Q3 thể hiện ranh giới của nhóm dữ liệu cao, phản ánh phần dữ liệu có giá trị lớn hơn trung vị.

Ba giá trị này giúp thể hiện rõ cấu trúc phân bố, độ lệch và sự tập trung của dữ liệu trong biểu đồ hộp.

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

Giá trị thiếu (missing values) là những ô dữ liệu không có thông tin hoặc bị bỏ trống trong tập dữ liệu. Việc xử lý các giá trị thiếu là bước quan trọng trước khi tính toán các chỉ số thống kê mô tả, vì dữ liệu không đầy đủ có thể làm sai lệch kết quả phân tích.

Một số phương pháp thường được sử dụng để xử lý giá trị thiếu gồm:

* **Loại bỏ (drop):** bỏ hàng hoặc cột chứa nhiều giá trị thiếu.
* **Thay thế (impute):** dùng giá trị trung bình, trung vị hoặc giá trị thường gặp (mode) để điền vào.
* **Nội suy (interpolation):** dự đoán giá trị dựa trên xu hướng dữ liệu xung quanh.

Tóm lại, trước khi tính toán các chỉ số thống kê mô tả, cần xác định nguyên nhân và tỷ lệ giá trị thiếu, sau đó chọn phương pháp xử lý phù hợp. Việc làm sạch dữ liệu giúp đảm bảo độ tin cậy và tính chính xác của các kết quả thống kê sau này.

* 1. **Cách đọc và diễn giải biểu đồ histogram hoặc boxplot từ dữ liệu thực tế?**

Histogram: cho thấy phân bố dữ liệu theo từng khoảng giá trị.

* Nếu hình chuông: dữ liệu phân bố chuẩn.
* Nếu lệch phải:  phần lớn giá trị nhỏ, có vài giá trị lớn.

Boxplot: thể hiện phạm vi dữ liệu, trung vị, các phân bố dữ liệu và ngoại lệ (dấu chấm ngoài “râu”).

* “Hộp” càng cao: dữ liệu càng phân tán.
* Dấu chấm ngoài hộp: giá trị ngoại lai.
  1. **Khi gặp dữ liệu có ngoại lai (outliers), nên xử lý thế nào trước khi thống kê mô tả?**

Một số hướng xử lý:

* Kiểm tra lại tính hợp lý: xem giá trị đó là lỗi nhập liệu hay hợp lệ.
* Loại bỏ (remove): nếu là lỗi hoặc gây sai lệch quá lớn.
* Cắt ngưỡng (winsorizing): thay giá trị ngoại lai bằng giá trị giới hạn (ví dụ: 5% và 95%).
* Phân tích riêng biệt: giữ lại để xem ảnh hưởng đến phân tích.
* Việc xử lý cần cân nhắc để không làm mất thông tin quan trọng.

**II. Bài tập ứng dụng**

* 1. **Khám phá dữ liệu COVID**
  2. Khai báo dữ liệu

A screenshot of a computer code

AI-generated content may be incorrect.

* numpy xử lý mảng số và phép tính thống kê nhanh.
* pandas để đọc và thao tác bảng dữ liệu (DataFrame).
* scipy.stats, skew, kurtosis dùng để tính các chỉ số phân bố (độ lệch, độ nhọn/kurtosis) và hàm thống kê khác.
* pd.read\_csv đọc file CSV (dữ liệu gốc ở đây là bộ dữ liệu OWID covid)

Sau đó giữ lại chỉ những cột cần cho phân tích: iso\_code, continent, location, date, total\_cases, new\_cases. Việc lọc cột giúp giảm độ phức tạp và tập trung vào mục tiêu (phân tích ca mới & tổng ca).

* 1. Đọc và làm sạch dữ liệu
     1. Đọc dữ liệu

A screenshot of a computer

AI-generated content may be incorrect.

* tail(5) cho biết bản cập nhật dữ liệu đến ngày nào
* Dùng để kiểm tra tính cập nhật của dataset và xem có khoảng trống thời gian hay không. Trong báo cáo, nêu rõ mốc thời gian cuối cùng

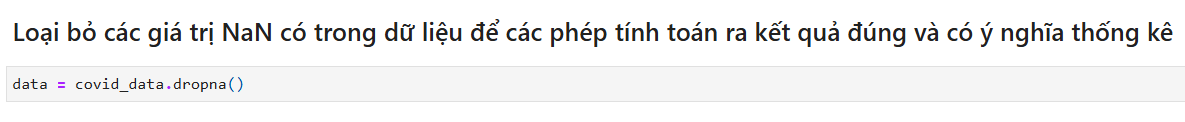
A screenshot of a computer

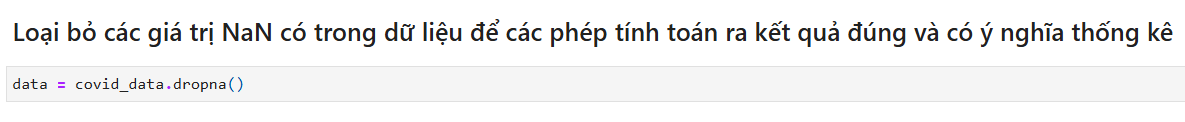
AI-generated content may be incorrect.

* dtypes cho thấy date đang là *object* (chuỗi), total\_cases và new\_cases là float64. Điều này gợi ý cần chuyển date sang datetime nếu muốn phân tích theo thời gian/visualize.
* Dữ liệu có phạm vi thời gian từ tháng 1/2020 đến tháng 8/2024.
* Bao gồm mã quốc gia (iso\_code), châu lục (continent), tên quốc gia (location), ngày (date) và số ca nhiễm (total/new cases).

Tổng số dòng dữ liệu: 429,435 dòng (theo chỉ số dòng cuối cùng là 429,434).

* + 1. Làm sạch dữ liệu



* Lệnh **dropna()** trong **pandas** có chức năng **loại bỏ toàn bộ các hàng (rows) có giá trị bị thiếu** (NaN) trong bất kỳ cột nào của DataFrame **covid\_data**.  
    
  
* Sau khi chạy lệnh này, kết quả được gán vào biến mới data, nghĩa là data là phiên bản sạch hơn của covid\_data.
* Lúc này data chỉ còn những bản ghi đầy đủ thông tin (các cột như total\_cases, new\_cases, continent, location đều có giá trị hợp lệ).

Nhận xét:

* Bộ dữ liệu gốc covid\_data có thể chứa nhiều giá trị trống (NaN) — đặc biệt ở các quốc gia nhỏ hoặc trong giai đoạn đầu dịch.
* Việc dùng dropna() giúp đảm bảo tính chính xác khi tính toán các chỉ số thống kê như mean, median, skew, kurtosis... tránh lỗi hoặc sai lệch do giá trị trống.
  1. Thống kê mô tả cơ bản
     1. Giá trị trung bình ( mean )

**A screenshot of a computer

AI-generated content may be incorrect.**

Kết quả: 1989.2149  
Hàm np.mean() tính giá trị trung bình của toàn bộ cột new\_cases.  
Trung bình mỗi ngày có khoảng 1989 ca nhiễm mới, tuy nhiên giá trị này bị ảnh hưởng bởi các ngày có số ca đột biến rất cao, nên trung bình không phản ánh đúng xu hướng phổ biến.

* + 1. Trung vị ( Median )

**A white text box with black text

AI-generated content may be incorrect.**

Kết quả: 0.0  
Hàm np.median() trả về giá trị ở giữa sau khi sắp xếp dữ liệu.  
Một nửa số quan sát có giá trị ≤ 0, nghĩa là phần lớn các ngày hoặc quốc gia không ghi nhận ca nhiễm mới.

* + 1. Giá trị xuất hiện nhiều nhất ( Mode )

**A white rectangular object with black text

AI-generated content may be incorrect.**

Kết quả: 0.0  
Hàm stats.mode() tìm giá trị xuất hiện nhiều nhất trong dữ liệu.  
0 là giá trị phổ biến nhất, khẳng định rằng đa số dữ liệu có số ca nhiễm mới bằng 0.

* + 1. Phương sai ( Variance )

**A screenshot of a computer

AI-generated content may be incorrect.**

Kết quả: 7.37×10¹⁰  
Hàm np.var() đo độ phân tán của dữ liệu quanh trung bình.  
Phương sai cực lớn → dữ liệu có mức độ biến động rất cao, thể hiện sự khác biệt lớn giữa các quốc gia hoặc thời kỳ dịch.

* + 1. **S**tandard Deviation (Độ lệch chuẩn)

**A screenshot of a computer code

AI-generated content may be incorrect.**

Kết quả: 85,858.16  
Độ lệch chuẩn là căn bậc hai của phương sai.  
Độ lệch chuẩn rất cao cho thấy mức dao động dữ liệu mạnh, phù hợp với thực tế khi dịch có nhiều giai đoạn bùng phát không đều.

* + 1. Min - Max (Giá trị nhỏ nhất - Giá trị lớn nhất)

**A screenshot of a computer code

AI-generated content may be incorrect.**

Kết quả: Min = 0.0 ; Max = 40,475,477.0  
Hai hàm np.min() và np.max() trả về giá trị biên dưới và biên trên của dữ liệu.  
Có ngày không ghi nhận ca mới (0) nhưng cũng có ngày vượt 40 triệu ca → dữ liệu rất phân tán và chênh lệch lớn.

* + 1. Khoảng giá trị ( Range )

**A screenshot of a computer

AI-generated content may be incorrect.**

Kết quả: 40,475,477.0  
Khoảng giá trị là hiệu giữa giá trị lớn nhất và nhỏ nhất.  
Khoảng biến thiên cực lớn chứng tỏ sự dao động mạnh trong số ca mới giữa các quan sát.

* + 1. Percentiles (Phân vị)

**A screenshot of a computer code

AI-generated content may be incorrect.**

Kết quả: 0.0 – 0.0 – 0.0  
Hàm np.percentile() tính các mốc giá trị theo phần trăm phân vị.  
Các phân vị đều bằng 0 → ít nhất 75% dữ liệu bằng hoặc nhỏ hơn 0, cho thấy sự tập trung mạnh ở giá trị thấp.

* + 1. Quartiles (Q1, Q2, Q3)

**A white box with black text

AI-generated content may be incorrect.**

 Kết quả: [0.0, 0.0, 0.0]  
Tương tự phân vị, nhưng được biểu diễn theo tỉ lệ 0–1.  
Các tứ phân vị đều bằng 0 → không có sự biến thiên trong 75% dữ liệu đầu.

* + 1. Interquartile Range (IQR)

**A screenshot of a computer code

AI-generated content may be incorrect.**

Kết quả: 0.0  
 Phân tích: IQR = Q3 - Q1, đo độ phân tán của dữ liệu trung tâm.  
 Nhận xét: IQR bằng 0 cho thấy phần lớn dữ liệu nằm cố định tại giá trị 0, biến thiên hầu như không tồn tại ở vùng giữa.

* + 1. Skewness (Độ lệch)

**A white rectangular object with red and blue text

AI-generated content may be incorrect.**

Kết quả: 340.4787  
 Phân tích: Skewness đo mức độ lệch của phân bố so với trung tâm.  
 Nhận xét: Skew > 0 và rất lớn → phân bố lệch phải cực mạnh, nghĩa là hầu hết giá trị thấp (0) và một số ít giá trị cực cao kéo đuôi bên phải.

* + 1. Kurtosis (Độ lệch)

**A screenshot of a computer

AI-generated content may be incorrect.**

Kết quả: 145,050.17  
 Phân tích: Kurtosis thể hiện độ nhọn của phân bố.  
 Nhận xét: Giá trị rất cao (>3 nhiều lần) → phân bố nhọn hơn chuẩn, dữ liệu chứa nhiều giá trị ngoại lai (outliers).

Kết Luận chung:

* Hơn 90% giá trị bằng 0, chứng tỏ phần lớn quốc gia hoặc thời điểm không ghi nhận ca mới.
* Dữ liệu lệch phải mạnh và rất nhiều outliers, nên không phù hợp với các phương pháp giả định phân phối chuẩn.
* Khi phân tích thống kê suy luận, nên áp dụng phương pháp phi tham số (như Kruskal-Wallis, Mann-Whitney) hoặc biến đổi log để giảm ảnh hưởng của các giá trị cực lớn.
  1. **Khám phá và xử lý dữ liệu Marketing Campaign**

2.1. Khai báo và đọc dữ liệu:

2.1.1. Khai báo dữ liệu

A screenshot of a computer

AI-generated content may be incorrect.

Đoạn mã này thực hiện hai nhiệm vụ chính: nạp dataset Marketing Campaign vào DataFrame marketing\_data, và thực hiện một loạt kiểm tra ban đầu để hiểu nhanh cấu trúc dữ liệu

* import pandas as pd: nhập thư viện pandas — thư viện nền tảng để thao tác bảng dữ liệu trong Python.
* pd.read\_csv("marketing\_campaign.csv", sep=None, engine="python"): đọc file CSV. sep=None cho phép pandas dò ký tự phân chia (phù hợp nếu file dùng ‘,’ hoặc ‘;’). engine="python" dùng parser Python (thường giúp đọc file có format phức tạp hơn).
* Việc chọn marketing\_data = marketing\_data[[...]] là chọn lọc cột — giữ lại những biến quan trọng cho phân. Điều này làm cho dataset gọn hơn và tập trung vào các biến phục vụ phân tích hành vi mua hàng.

2.1.2. Đọc dữ liệu

A screenshot of a computer

AI-generated content may be incorrect.A screenshot of a computer

AI-generated content may be incorrect.

* head() và tail(): hiển thị mẫu dữ liệu ở đầu và cuối file để kiểm tra việc đọc file có bị lỗi
* dtypes và info(): kiểm tra kiểu dữ liệu và số lượng giá trị không-null cho từng cột → rất quan trọng để phát hiện cột cần chuyển kiểu, hoặc phát hiện cột có nhiều giá trị thiếu
* shape: cho biết số lượng bản ghi và biến, nền tảng để báo cáo quy mô nghiên cứu
* describe(): thống kê mô tả cho các biến số (mean, std, min, percentiles, max) — dùng để nắm phân phối sơ bộ của biến số.

2.2. Loại bỏ dữ liệu trùng lặp

A screenshot of a computer

AI-generated content may be incorrect.

Kết quả “0 dòng trùng lặp” phản ánh chất lượng dữ liệu ban đầu tốt, dữ liệu khách hàng được nhập chính xác và không bị trùng ID.

Tuy nhiên, trong các tập dữ liệu thực tế khác, cần thận trọng vì:

* Một số bản ghi có thể trùng về thông tin nhưng khác ID → vẫn cần xem xét bằng tay.
* Một số bản ghi trùng chỉ ở vài cột (ví dụ tên, năm sinh) → có thể là lỗi nhập liệu.

Khi phát hiện trùng lặp, thông thường có hai hướng xử lý:

* Xóa hoàn toàn bằng drop\_duplicates(inplace=True) nếu xác định bản ghi trùng là lỗi.
* Giữ lại bản ghi duy nhất nhất định dựa theo tiêu chí cụ thể (ví dụ: giữ bản có ngày cập nhật gần nhất).

Việc thực hiện bước này trước các bước phân tích mô tả hoặc huấn luyện mô hình là rất quan trọng, vì dữ liệu trùng có thể:

* Làm sai lệch thống kê trung bình, tần suất, hoặc tỷ lệ phản hồi (Response).
* Khi xây dựng mô hình dự đoán, có thể khiến mô hình học trùng lặp dữ liệu → giảm khả năng khái quát (generalization).

2.3. Thay thế và thay đổi định dạng dữ liệu

2.3.1. Thay thế các giá trị

A screenshot of a computer

AI-generated content may be incorrect.

Việc gộp các giá trị 1 và 2 cùng mang ý nghĩa “has teen” thể hiện chủ đích đơn giản hóa dữ liệu — biến Teenhome trở thành biến nhị phân phân loại (has teen / has no teen).  
Điều này rất hữu ích khi thực hiện:

* Thống kê mô tả theo nhóm,
* Trực quan hóa (ví dụ: so sánh mức chi tiêu giữa khách hàng có và không có con tuổi teen),
* Hoặc mô hình hóa dữ liệu (giảm số lượng lớp trong biến phân loại).

Thao tác .replace() của pandas là phương pháp trực quan, hiệu quả và dễ đọc khi thay giá trị cho toàn bộ cột. Sau khi thay thế, nên kiểm tra lại kiểu dữ liệu (marketing\_data['Teenhome\_replaced'].dtype) để đảm bảo cột được lưu ở dạng object (chuỗi), phù hợp với các bước xử lý tiếp theo như one-hot encoding nếu cần đưa vào mô hình dự đoán.

Ngoài ra, có thể sử dụng phương pháp .map() thay thế nếu muốn thay giá trị theo dictionary (ví dụ: {0:'has no teen', 1:'has teen', 2:'has teen'}).

2.3.2. Đổi kiểu dữ liệu

A screenshot of a computer program

AI-generated content may be incorrect.

Đoạn code trên chuyển đổi kiểu dữ liệu của cột Income từ float sang int bằng lệnh .astype(int), nhằm loại bỏ phần thập phân không cần thiết và đồng nhất định dạng dữ liệu. Việc tạo thêm cột Income\_changed giúp lưu giá trị sau khi chuyển đổi mà không làm mất dữ liệu gốc.

Kết quả cho thấy cột mới có kiểu dữ liệu int64 và giá trị thu nhập được hiển thị dưới dạng số nguyên, phù hợp cho các phép thống kê và phân tích tiếp theo. Việc đổi kiểu này giúp dữ liệu gọn gàng, chính xác và thuận lợi cho việc mô hình hóa.

2.4. Xử lý giá trị thiếu

Kiểm tra và xử lý các giá trị bị thiếu (NaN) trong bộ dữ liệu Marketing Campaign để đảm bảo dữ liệu hoàn chỉnh, không gây sai lệch cho các bước phân tích tiếp theo.

A screenshot of a computer

AI-generated content may be incorrect.

Bộ dữ liệu hoàn toàn đầy đủ, không có giá trị bị thiếu.

Điều này giúp quá trình phân tích, thống kê mô tả và mô hình hóa sau này được thực hiện chính xác, không cần thực hiện các kỹ thuật thay thế (như điền trung bình, trung vị hoặc mô phỏng dữ liệu).

Việc kiểm tra và xác nhận dữ liệu sạch là bước quan trọng trong giai đoạn tiền xử lý (Data Cleaning), đảm bảo chất lượng và độ tin cậy của các kết quả phân tích tiếp theo.

2.5. Kết quả

A screenshot of a computer

AI-generated content may be incorrect.

Sau khi xử lý dữ liệu (bao gồm kiểm tra và loại bỏ giá trị thiếu, thay thế biến, chuẩn hóa thông tin), bộ dữ liệu Marketing Campaign đã sẵn sàng cho các bước phân tích sâu hơn như mô tả thống kê, phân cụm khách hàng hoặc xây dựng mô hình dự đoán.

Kích thước dữ liệu vẫn giữ nguyên (2240, 13), chứng tỏ quá trình xử lý không làm mất mát thông tin.

Việc hiển thị 5 dòng đầu tiên giúp đảm bảo tính toàn vẹn và hợp lý của dữ liệu sau khi làm sạch, đồng thời xác nhận các biến được giữ lại và tái cấu trúc hợp lý cho mục tiêu phân tích marketing.

* 1. **Thống kê mô tả dữ liệu về phân loại chất lượng rượu đỏ**

3.1. Giới thiệu chung về dữ liệu

Bộ dữ liệu *Red Wine Quality* được lấy từ nghiên cứu của Cortez et al. (2009), thuộc Đại học Minho (Bồ Đào Nha).

Dữ liệu mô tả các đặc tính hóa học của rượu vang đỏ vùng Vinho Verde, và chất lượng cảm quan của chúng (được đánh giá bởi chuyên gia nếm rượu).

Bộ dữ liệu gồm 11 biến độc lập (đặc tính hóa lý) + 1 biến phụ thuộc (quality)

3.2. Khai báo và đọc dữ liệu

3.2.1. Nạp dữ liệu

**A screenshot of a computer

AI-generated content may be incorrect.**

Đoạn mã trên nhập các thư viện cần thiết cho phân tích dữ liệu:

* numpy và pandas hỗ trợ xử lý dữ liệu dạng bảng.
* scipy.stats dùng để tính các chỉ số thống kê như độ lệch (Skewness) và độ nhọn (Kurtosis).

3.2.2. Đọc dữ liệu

A screenshot of a computer

AI-generated content may be incorrect.

* + Fixed acidity: Độ axit cố định
  + Volatile acidity: Độ axit bay hơi
  + Citric acid: Axit citric
  + Residual sugar: Lượng đường còn sót lại
  + Chlorides: Hàm lượng muối trong rượu
  + Free sulfur dioxide: SO₂ tự do
  + Total sulfur dioxide: Tổng SO₂
  + Density: Khối lượng riêng
  + pH: Độ axit – kiềm của rượu
  + Sulphates: Hàm lượng sunfat
  + Alcohol: Nồng độ cồn
  + Quality: Chất lượng

3.2.3. Kiểm tra thông tin và chất lượng dữ liệu

a) Kiểm tra dữ liệu từng biến

A screenshot of a computer

AI-generated content may be incorrect.

DataFrame: cho thấy hầu hết các biến là float64, ngoại trừ quality là int64.

Dữ liệu ở dạng số thực rất thuận lợi cho các phép tính thống kê, mô hình hồi quy, hoặc phân tích tương quan.

Không có biến dạng chuỗi (string) hoặc dạng ngày tháng:  không cần chuyển đổi kiểu dữ liệu.

b) Kiểm tra kích thước dữ liệu

A screenshot of a computer

AI-generated content may be incorrect.

Lệnh data.shape trả về một tuple (1599, 12), nghĩa là tập dữ liệu có 1.599 hàng và 12 cột.

c) Kiểm tra thông tin tổng quát và giá trị thiếu

A screenshot of a computer

AI-generated content may be incorrect.

Việc kiểm tra giá trị thiếu cho thấy tất cả 12 cột đều không chứa bất kỳ giá trị NaN/Null nào (1599 giá trị non-null trên tổng số 1599 quan sát.

d) Kiểm tra giá trị null chi tiết

A screenshot of a computer

AI-generated content may be incorrect.

Kết quả cho thấy:

* Dữ liệu gồm 12 cột (11 đặc tính + 1 biến mục tiêu quality).
* Không có giá trị thiếu (NaN).
* Tất cả biến đều là dạng số (numeric).

3.3. Thống kê mô tả

A screenshot of a computer

AI-generated content may be incorrect.

* + Fixed acidity (Độ axit cố định)

Mean = 8.32, Std = 1.74 → độ lệch chuẩn trung bình, dữ liệu phân tán vừa phải.

Giá trị dao động từ 4.6 đến 15.9 → có một số mẫu axit cao.

**Nhận xét:** Dải giá trị khá rộng, có thể ảnh hưởng đến vị rượu và nên được chuẩn hóa trước khi mô hình hóa.

* + Volatile acidity (Axit bay hơi)

Mean = 0.53, Std = 0.18 → phân bố khá đồng đều.

Dao động từ 0.12 đến 1.58.  
**Nhận xét:** Giá trị cao của biến này thường làm giảm chất lượng rượu( biến này quan trọng trong dự đoán quality).

* + Citric acid (Axit citric)

Mean = 0.27, Std = 0.19, Min = 0 → có mẫu không chứa axit citric.  
**Nhận xét:** Độ axit này thấp, ảnh hưởng nhẹ đến hương vị tổng thể.

* + Residual sugar (Đường còn sót lại)

Mean = 2.54, Std = 1.41 → dao động từ 0.9 đến 15.5.  
**Nhận xét:** Có vài mẫu ngọt hơn mức trung bình (có thể là ngoại lệ).

* + Chlorides (Hàm lượng muối)

Mean = 0.087, Std = 0.047 → giá trị khá nhỏ, biến này ổn định.  
**Nhận xét:** Giá trị muối thấp là tốt, vì muối cao có thể làm giảm chất lượng.

* + Free sulfur dioxide (SO₂ tự do)

Mean = 15.87, Std = 10.46, dao động 1–72.  
**Nhận xét:** Biến có độ phân tán cao, cần xem xét khả năng có ngoại lệ.

* + Total sulfur dioxide (Tổng SO₂)

Mean = 46.47, Std = 32.89, dao động 6–289.  
**Nhận xét:** Biến có phạm vi rất rộng, có thể có nhiều giá trị bất thường.

* + Density (Khối lượng riêng)

Mean ≈ 0.9967, Std = 0.0019.  
**Nhận xét:** Biến này ổn định, ít biến động — phù hợp với đặc tính vật lý của rượu.

* + pH (Độ axit – kiềm của rượu)

Mean = 3.31, Std = 0.15, dao động 2.74–4.01.  
**Nhận xét:** Độ pH trung bình nằm trong khoảng axit nhẹ → hợp lý với rượu vang.

* + Sulphates (Hàm lượng sunfat)

Mean = 0.66, Std = 0.17, dao động 0.33–2.00.  
**Nhận xét:** Một vài mẫu có giá trị khá cao → có thể làm tăng độ bền rượu.

* + Alcohol (Nồng độ cồn)

Mean = 10.42%, Std = 1.07, dao động 8.4–14.9%.  
**Nhận xét:** Mức độ cồn cao thường liên quan đến rượu chất lượng tốt hơn.

* + Quality (Chất lượng rượu)

Mean = 5.64, Std = 0.81, dao động từ 3–8.  
**Nhận xét:** Phần lớn rượu có chất lượng trung bình (khoảng 5–6 điểm). Dữ liệu khá cân bằng, phù hợp cho mô hình phân tích.

A screenshot of a computer

AI-generated content may be incorrect.

Kết quả cho thấy hầu hết các biến trong bộ dữ liệu có độ lệch phải (Skewness dương), nghĩa là phần lớn giá trị tập trung ở mức thấp và có một số giá trị cao bất thường.

* Các biến như fixed acidity, volatile acidity, free sulfur dioxide, total sulfur dioxide, sulphates và alcohol có độ lệch phải nhẹ đến vừa, cho thấy phân phối hơi nghiêng về phía các giá trị nhỏ.
* Biến residual sugar và chlorides có độ lệch phải rất cao (Skewness > 4 và > 5), đồng thời có Kurtosis lớn. Điều này nghĩa là dữ liệu của hai biến này có nhiều giá trị ngoại lệ, phân phối không đều và rất nhọn.
* Các biến citric acid, pH và density có Skewness gần 0 và Kurtosis thấp, nên phân phối gần chuẩn, khá cân đối và ổn định.
* Biến quality (chất lượng rượu) có Skewness nhỏ (0.21) và Kurtosis thấp, cho thấy phân phối điểm chất lượng khá đồng đều, không bị lệch nhiều.
  1. **Thống kê mô tả dữ liệu bệnh đái tháo đường**

4.1. Giới thiệu thông tin dữ liệu

Bộ dữ liệu *Pima Indians Diabetes* là một trong những tập dữ liệu kinh điển trong lĩnh vực y học và học máy (machine learning). Dữ liệu được thu thập từ phụ nữ thuộc bộ tộc Pima (Mỹ) nhằm dự đoán khả năng mắc bệnh đái tháo đường type 2 dựa trên các chỉ số sinh lý và xét nghiệm y khoa.

Các biến trong bộ dữ liệu phản ánh các yếu tố nguy cơ chính của bệnh tiểu đường, như nồng độ glucose, chỉ số BMI, huyết áp, và các thông số sinh học khác. Việc sử dụng tập dữ liệu này giúp minh họa quy trình phân tích dữ liệu y tế và xây dựng mô hình dự đoán khả năng mắc bệnh dựa trên đặc điểm sinh học của từng cá nhân.

Bộ dữ liệu gồm **768 quan sát (mẫu dữ liệu)**, tương ứng với từng người phụ nữ trong nhóm khảo sát.  
Mỗi mẫu chứa 9 biến (features), trong đó có:

* 8 biến độc lập (đặc trưng y sinh) – thể hiện các yếu tố ảnh hưởng đến sức khỏe và nguy cơ mắc bệnh.
* 1 biến phụ thuộc (Outcome) – thể hiện tình trạng bệnh lý (0 = không mắc tiểu đường, 1 = mắc tiểu đường).

A screenshot of a computer

AI-generated content may be incorrect.

Đây là tập dữ liệu điển hình để nghiên cứu mối quan hệ giữa yếu tố sinh học và khả năng mắc bệnh tiểu đường, đồng thời giúp kiểm chứng hiệu quả của các mô hình dự báo y học.

4.2. Nạp và xem trước dữ liệu

4.2.1. Import thư viện

A white rectangular object with black text

AI-generated content may be incorrect.

* NumPy (np): Hỗ trợ xử lý dữ liệu dạng mảng và thực hiện các phép toán số học, thống kê nhanh chóng.
* Pandas (pd): Dùng để xử lý dữ liệu dạng bảng (DataFrame), cho phép đọc, truy cập và phân tích dữ liệu một cách thuận tiện.
* SciPy (stats, skew, kurtosis): Cung cấp các hàm thống kê nâng cao, bao gồm tính độ lệch (skewness) và độ nhọn (kurtosis) của phân phối dữ liệu, phục vụ cho việc mô tả đặc điểm của các biến sau này.

4.2.2. Đọc dữ liệu

A screenshot of a medical chart

AI-generated content may be incorrect.

Bảng dữ liệu đầu tiên cho thấy cấu trúc bao gồm các biến chính:

* Pregnancies: số lần mang thai.
* Glucose: nồng độ glucose huyết tương.
* BloodPressure: huyết áp tâm trương.
* SkinThickness: độ dày nếp gấp da.
* Insulin: nồng độ insulin huyết thanh.
* BMI: chỉ số khối cơ thể (Body Mass Index).
* DiabetesPedigreeFunction: chỉ số nguy cơ di truyền tiểu đường.
* Age: tuổi của người tham gia khảo sát.
* Outcome: biến mục tiêu (0 = không mắc bệnh, 1 = mắc bệnh).

4.2.3. Kiểm tra thông tin và chất lượng dữ liệu

a) Kiểm tra dữ liệu từng biến

A screenshot of a computer

AI-generated content may be incorrect.

Kết quả cho thấy:

* Hầu hết các biến là kiểu số nguyên (int64), ví dụ: Pregnancies, Glucose, BloodPressure, Insulin, Age, Outcome.
* Biến BMI và DiabetesPedigreeFunction là số thực (float64) vì có thể chứa giá trị thập phân.

Việc xác định đúng kiểu dữ liệu giúp đảm bảo các phép tính toán và thống kê thực hiện chính xác ở các bước tiếp theo.

b) Kiểm tra kích thước dữ liệu

A screenshot of a computer

AI-generated content may be incorrect.

* Lệnh data.shape trả về một tuple (768, 9) — nghĩa là dữ liệu có 768 dòng và 9 cột.
* Mỗi dòng tương ứng với một người phụ nữ trong bộ dữ liệu, và 9 cột biểu diễn các đặc trưng y học cùng biến mục tiêu Outcome.

c) Kiểm tra thông tin tổng quát và giá trị thiếu

A screenshot of a computer

AI-generated content may be incorrect.

Hàm data.info() cung cấp thông tin tổng quan:

* Tổng số cột: 9
* Tổng số dòng: 768
* Số lượng giá trị non-null trong mỗi cột là 768, nghĩa là không có giá trị bị thiếu (NaN).
* Ngoài ra, hàm cũng hiển thị kiểu dữ liệu và dung lượng bộ nhớ (54.1 KB), giúp đánh giá quy mô và tính đầy đủ của dữ liệu.

d) Kiểm tra giá trị null chi tiết

A screenshot of a computer

AI-generated content may be incorrect.

* Lệnh này đếm số lượng giá trị null trong từng cột.
* Kết quả cho thấy tất cả các cột đều có giá trị bằng 0, xác nhận rằng không có dữ liệu bị thiếu.
* Đây là một tín hiệu tích cực, giúp tránh việc phải áp dụng các phương pháp bù dữ liệu (như thay thế bằng trung bình, trung vị hoặc mô phỏng).

4.3. Thống kê mô tả

A screenshot of a graph

AI-generated content may be incorrect.

* Pregnancies (Số lần mang thai)

Mean = 3.8, Median = 3, Max = 17  
→ Dữ liệu khá cân đối, không bị lệch mạnh.  
→ Tuy nhiên, có một vài giá trị cao (ví dụ 17 lần mang thai) có thể được xem là ngoại lai nhẹ, nhưng vẫn hợp lý trong bối cảnh y học.

* Glucose (Nồng độ glucose huyết tương)

Mean ≈ 121, Median = 117, Min = 0, Max = 199  
→ Dữ liệu hơi lệch phải, tức có nhiều người có mức glucose thấp hơn trung bình, và một số người có giá trị cao bất thường.  
→ Giá trị Min = 0 không hợp lý về mặt y sinh học, vì nồng độ glucose không thể bằng 0. Đây là dấu hiệu của dữ liệu thiếu bị mã hóa bằng 0.  
→ Giá trị Max = 199 phản ánh nhóm bệnh nhân có khả năng mắc tiểu đường nặng.

* BloodPressure (Huyết áp tâm trương)

Mean ≈ 69, Median = 72, Min = 0, Max = 122  
→ Phân bố dữ liệu khá ổn định quanh giá trị trung bình, không quá phân tán.  
→ Tuy nhiên, giá trị 0 là không hợp lý (vì huyết áp không thể bằng 0), chứng tỏ có lỗi hoặc dữ liệu thiếu.  
→ Giá trị cao nhất 122 mmHg vẫn nằm trong giới hạn hợp lý của người bị tăng huyết áp nhẹ.

* SkinThickness (Độ dày nếp gấp da)

Mean ≈ 20.5, Median = 23, Std ≈ 16, Min = 0, Max = 99  
→ Có độ lệch chuẩn cao, chứng tỏ dữ liệu phân tán mạnh.  
→ Min = 0 là không hợp lý, thể hiện giá trị bị thiếu.  
→ Max = 99 là ngoại lệ cao, nhưng vẫn chấp nhận được với người có độ dày da lớn.

* Insulin (Nồng độ insulin huyết thanh)

Mean ≈ 79.8, Std = 115.2, Min = 0, Max = 846  
→ Biến này có độ lệch chuẩn rất cao, cho thấy dữ liệu phân tán mạnh và có nhiều ngoại lai.  
→ Min = 0 không hợp lý về mặt y học, gợi ý dữ liệu bị thiếu hoặc nhập sai.  
→ Max = 846 là giá trị cực lớn, có thể là ngoại lai do sai lệch phép đo hoặc đặc thù bệnh nhân nặng.

* BMI (Chỉ số khối cơ thể)

Mean ≈ 32, Median = 32, Min = 0, Max = 67  
→ Dữ liệu khá cân đối, không lệch.  
→ Tuy nhiên, BMI = 0 không hợp lý (vì BMI luôn > 0).  
→ Max = 67 phản ánh người béo phì nặng, nhưng vẫn trong phạm vi có thể gặp thực tế.

* DiabetesPedigreeFunction (Chỉ số nguy cơ di truyền)

Mean ≈ 0.47, Median = 0.37, Max = 2.42  
→ Dữ liệu có phân phối lệch phải (nhiều giá trị nhỏ, ít giá trị cao).  
→ Các giá trị cao thể hiện nguy cơ di truyền cao nhưng không bị cực đoan.

* Age (Tuổi)

Mean ≈ 33, Median = 29, Min = 21, Max = 81  
→ Phân phối lệch phải, phần lớn người tham gia còn trẻ (khoảng 20–40 tuổi).  
→ Một số ít người lớn tuổi (max = 81), thể hiện dữ liệu bao quát được nhiều nhóm tuổi.

* Outcome (Biến mục tiêu)

Mean ≈ 0.35, nghĩa là khoảng 35% người tham gia bị chẩn đoán mắc bệnh tiểu đường.

Sử dụng data["Outcome"].value\_counts() cho thấy:

* + 0: 500 người (65%) không mắc bệnh.
  + 1: 268 người (35%) mắc bệnh.

→ Dữ liệu mất cân bằng nhẹ, nhưng vẫn đủ để thực hiện mô hình phân loại.

1. Phân tích tỷ lệ lớp

A screenshot of a computer code

AI-generated content may be incorrect.

**Nhận xét**

* Có 500 trường hợp không bị tiểu đường (Outcome = 0) chiếm khoảng 65% dữ liệu.
* Có 268 trường hợp bị tiểu đường (Outcome = 1) chiếm khoảng 35% dữ liệu.
* Dữ liệu không hoàn toàn cân bằng: lớp 0 nhiều hơn lớp 1. Tuy nhiên, sự chênh lệch chưa quá lớn (tỉ lệ ~2:1).

1. Phân tích hình dạng phân phối

A screenshot of a computer

AI-generated content may be incorrect.

* Pregnancies: Lệch phải nhẹ, phân phối hơi nhọn hơn chuẩn
* Glucose: Gần chuẩn, phân phối khá cân đối
* BloodPressure: Lệch trái mạnh, có thể có nhiều giá trị thấp bất thường
* SkinThickness: Phân phối gần chuẩn, hơi bẹt
* Insulin: Lệch phải mạnh và rất nhọn, nhiều giá trị cực lớn, nên chuẩn hóa hoặc log-transform
* BMI: Lệch trái nhẹ, hơi nhọn
* DiabetesPedigreeFunction: Lệch phải mạnh, có thể chứa outlier
* Age: Lệch phải nhẹ, khá ổn định
* Outcome: Lệch phải nhẹ ( do tỷ lệ người mắc bệnh cao hơn )

### **B. Xử lý và trực quan hóa dữ liệu**

## **I. Ôn tập lý thuyết**

### **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)?**

Trực quan hóa dữ liệu (Data Visualization) là quá trình chuyển đổi dữ liệu thành hình ảnh, biểu đồ hoặc đồ thị giúp người xem hiểu nhanh các mẫu, xu hướng và mối quan hệ trong dữ liệu.

Trong phân tích khám phá dữ liệu (EDA), trực quan hóa đóng vai trò rất quan trọng vì:

* Giúp phát hiện giá trị ngoại lai, dữ liệu thiếu, hoặc xu hướng bất thường.
* Giúp so sánh các biến, kiểm tra giả thuyết và rút ra nhận định ban đầu mà con số đơn thuần khó thể hiện được.

Nói cách khác, trực quan hóa giúp “thấy được câu chuyện ẩn sau dữ liệu”.

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

Các loại biểu đồ phổ biến và cách sử dụng:

* Histogram: Dùng để thể hiện phân bố của một biến liên tục (như điểm số, thu nhập, chiều cao). Giúp nhận biết dữ liệu có phân bố chuẩn hay lệch.
* Boxplot: Dùng để so sánh sự phân bố và phát hiện ngoại lai giữa các nhóm hoặc biến.
* Scatter plot: Dùng để thể hiện mối quan hệ giữa hai biến liên tục, giúp phát hiện xu hướng hoặc tương quan.
* Bar chart: Dùng cho dữ liệu phân loại (categorical) để so sánh tần suất hoặc giá trị trung bình giữa các nhóm.

### **3. 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 chọn biểu đồ phụ thuộc vào loại dữ liệu:

* Với dữ liệu phân loại (categorical): dùng biểu đồ cột (bar chart) hoặc biểu đồ tròn (pie chart) để so sánh tần suất giữa các nhóm.
* Với dữ liệu số (numeric/continuous): dùng histogram để xem phân bố, boxplot để phát hiện ngoại lai, scatter plot để xem mối quan hệ giữa hai biến.
* Với dữ liệu theo thời gian (time series): dùng biểu đồ đường (line chart) để quan sát xu hướng thay đổi theo thời gian.

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

Sự khác biệt giữa Matplotlib, Seaborn và Plotly là:

* Matplotlib: Là thư viện cơ bản và mạnh mẽ nhất để trực quan hóa dữ liệu trong Python. Nó cho phép tùy chỉnh chi tiết biểu đồ, nhưng mã lệnh thường dài và ít thẩm mỹ.
* Seaborn: Được xây dựng trên nền Matplotlib, cung cấp biểu đồ đẹp, dễ tạo và hỗ trợ trực tiếp phân tích thống kê như boxplot, heatmap, hay pairplot.
* Plotly: Hỗ trợ biểu đồ tương tác (interactive), giúp người dùng phóng to, di chuột xem giá trị, và phù hợp cho dashboard hoặc trình bày dữ liệu trực quan hiện đại.

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

Khi thiết kế biểu đồ, cần tuân thủ các nguyên tắc sau để đảm bảo tính dễ hiểu, hiệu quả và trực quan:

* Chọn biểu đồ phù hợp với loại dữ liệu (phân loại, số, thời gian).
* Giữ biểu đồ đơn giản, rõ ràng, tránh quá nhiều màu sắc hoặc chi tiết rối mắt.
* Gắn nhãn trục, tiêu đề và chú thích đầy đủ, giúp người xem dễ đọc và diễn giải.
* Sử dụng tỉ lệ trục hợp lý để không gây hiểu lầm về quy mô dữ liệu.
* Làm nổi bật thông tin chính bằng màu sắc hoặc ký hiệu khác biệt.

### **6. 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?**

Cách tạo biểu đồ đơn giản bằng Matplotlib là:

Ta sử dụng thư viện Matplotlib – công cụ trực quan hóa dữ liệu phổ biến và dễ dùng.

* Dùng **plt.hist()** để vẽ biểu đồ histogram, thể hiện phân bố của dữ liệu liên tục như điểm số hoặc chiều cao.
* Dùng **plt.bar()** để vẽ biểu đồ cột (bar chart), thể hiện so sánh giữa các nhóm hoặc danh mục.

Hai loại biểu đồ này giúp nhận biết xu hướng, phân bố và sự khác biệt giữa các nhóm dữ liệu một cách trực quan.

Đoạn code mẫu:

A screenshot of a computer program

AI-generated content may be incorrect.

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

Cách xuất biểu đồ ra các định dạng PNG, PDF hoặc HTML để sử dụng trong báo cáo là:

Biểu đồ sau khi tạo bằng Python có thể được xuất ra các định dạng hình ảnh như PNG, PDF hoặc HTML để chèn vào báo cáo. Trong Matplotlib, dùng lệnh **plt.savefig("tenfile.png")** hoặc **plt.savefig("tenfile.pdf")** để lưu biểu đồ dưới dạng ảnh hoặc file PDF. Với Plotly, có thể sử dụng **fig.write\_html("tenfile.html")** để xuất biểu đồ tương tác. Việc lưu biểu đồ giúp trình bày kết quả trực quan, chuyên nghiệp và dễ chia sẻ trong báo cáo phân tích dữ liệu.

## **II. Bài tập ứng dụng**

* 1. **Xử lý và trực quan hóa dữ liệu về giá nhà**
  2. Khai báo và đọc dữ liệu
* Input:

A computer code with text

AI-generated content may be incorrect.

* Output:

A screenshot of a table

AI-generated content may be incorrect.

* Giá nhà (Price) dao động khá lớn (từ 475.000 đến
* 850.000 euro chỉ trong 5 mẫu đầu) → thị trường có sự chênh lệch đáng kể giữa các khu vực.
* Diện tích (Area) cũng biến thiên mạnh, từ 60 m² đến 138 m².
* Số phòng (Room) dao động từ 3 đến 6 → có thể tương quan tích cực với diện tích và giá.
* Mã vùng (Zip) giúp phân nhóm nhà theo khu vực, có thể dùng sau này để phân tích không gian (spatial analysis).
  1. Tạo biến mới

A white background with black text

AI-generated content may be incorrect.

* Input:



* Output:

A screenshot of a table

AI-generated content may be incorrect.

* Chênh lệch giá lớn: Giá €/m2 dao động mạnh từ ≈4.500 (1060 TH) đến ≈10.700 (1091 CR), cho thấy sự khác biệt rõ rệt về vị trí hoặc chất lượng nhà ở giữa các khu vực.
* Giá trị vị trí: Khu vực 1091 CR có giá €/m2 cao nhất, ngụ ý đây là khu trung tâm/đắc địa hoặc có nhà ở chất lượng cao.
* Không tuyến tính: Không có mối quan hệ tuyến tính trực tiếp giữa Diện tích và Giá €/m2 . Diện tích càng lớn (1060 TH), giá €/m2 thường có xu hướng thấp hơn.
* Số phòng không chi phối: Số phòng không phải là yếu tố chi phối chính đến €/m2 (căn 6 phòng có giá thấp hơn căn 3 phòng), mà Vị trí (mã bưu chính) là yếu tố quan trọng hơn.
  1. Trực quan hóa dữ liệu với thư viện Matplotlib
     1. Case 1: cơ bản
* Input:

A computer screen shot of a code

AI-generated content may be incorrect.

* Output:

A graph of blue bars

AI-generated content may be incorrect.

* Hai khu vực 1017 EL và 1075 AH có giá cao nhất, gần 6 triệu euro, cho thấy đây là những khu vực đắt đỏ nhất trong dữ liệu.
* Các khu vực như 1076 XP, 1075 BM, 1015 AG có giá thấp hơn rõ rệt, dao động quanh mức 3 triệu euro.  
  Sự chênh lệch giữa khu vực cao nhất và thấp nhất khá lớn → giá nhà ở Amsterdam không đồng đều, phụ thuộc mạnh vào vị trí địa lý và giá trị khu vực
  + 1. Case 2: Thêm tiêu đề, nhãn trục và điều chỉnh kích cỡ chữ
* Input:

A screenshot of a computer code

AI-generated content may be incorrect.

* Output:

A graph of blue bars

AI-generated content may be incorrect.

* Biểu đồ cho thấy hai khu vực 1017 EL và 1075 AH có giá nhà cao nhất, đều xấp xỉ 6 triệu euro.  
   Các khu vực tiếp theo như 1016 AE, 1017 ZP, và 1012 JS có giá thấp hơn một chút, dao động quanh 4,5–5 triệu euro.  
   Các khu vực còn lại như 1031 KV, 1015 AG, 1075 BM, và đặc biệt là 1076 XP, có giá thấp hơn rõ rệt, chỉ khoảng 3 triệu euro.
* Điều này cho thấy giá nhà ở Amsterdam chênh lệch khá lớn giữa các khu vực, trong đó những khu nằm gần trung tâm hoặc có vị trí đắc địa (như 1017, 1075) thường có giá cao hơn nhiều so với các khu ngoại vi.
  + 1. Case 3: So sánh Price và PriceperSqm
* Input:

A screenshot of a computer code

AI-generated content may be incorrect.

* Output:

A graph showing a bar graph

AI-generated content may be incorrect.

* Giá nhà phân bố không đồng đều giữa các khu vực, thể hiện sự khác biệt về vị trí, cơ sở hạ tầng và mức sống.
* Các khu vực có mã bưu chính bắt đầu bằng 1017 và 1075 thường nằm gần trung tâm thành phố, nên giá cao hơn đáng kể.
* Biểu đồ giúp dễ dàng nhận ra vùng có giá nhà cao nhất và thấp nhất, phục vụ tốt cho việc phân tích thị trường hoặc đưa ra quyết định đầu tư.
* Đáng chú ý, khu 1012 JS có giá cao vượt trội, hơn 25.000 euro/m², cao gấp đôi các khu còn lại. Điều này cho thấy đây có thể là khu trung tâm đắt đỏ nhất, nơi có vị trí thuận lợi hoặc nguồn cung nhà ở hạn chế.
* Các khu vực như 1017 EL và 1075 AH cũng có giá cao, khoảng 12.000–15.000 euro/m², thể hiện là khu vực cao cấp.  
   Ngược lại, các khu 1076 XP, 1075 BM hay 1031 KV có giá thấp hơn nhiều, chỉ khoảng 9.000–10.000 euro/m².
  1. Trực quan hóa dữ liệu với thư viện Seaborn
     1. Case 1: Cơ bản
* Input:

A white rectangular object with text

AI-generated content may be incorrect.

* Output:

A graph of blue bars

AI-generated content may be incorrect.

* Biểu đồ cho thấy hai khu vực 1017 EL và 1075 AH có giá nhà cao nhất, xấp xỉ 6 triệu euro, cho thấy đây là những khu vực đắt đỏ nhất của Amsterdam.  
   Các khu vực như 1016 AE, 1017 ZP, và 1012 JS có giá thấp hơn một chút (khoảng 4,5–5 triệu euro), trong khi 1076 XP là khu có giá thấp nhất trong nhóm top 10, chỉ khoảng 2,8 triệu euro.
* Điều này thể hiện sự chênh lệch lớn về giá nhà giữa các khu vực, thường do vị trí địa lý, tiện ích xung quanh, và mức độ phát triển của từng vùng.
  + 1. Case 2: Thêm nhãn và tiêu đề
* Input:

A computer screen shot of a math equation

AI-generated content may be incorrect.

* Output:

A graph of blue rectangular bars

AI-generated content may be incorrect.

* Biểu đồ cho thấy hai khu vực 1017 EL và 1075 AH có giá nhà cao nhất, đều xấp xỉ 6 triệu euro, thể hiện đây là những khu vực đắt đỏ nhất của Amsterdam.  
   Các khu vực 1016 AE, 1017 ZP, và 1012 JS cũng có giá cao nhưng thấp hơn một chút (khoảng 4,5–5 triệu euro).  
   Trong khi đó, 1076 XP có giá thấp nhất trong nhóm top 10, chỉ khoảng 2,8 triệu euro.
* Điều này chứng tỏ giá nhà ở Amsterdam có sự chênh lệch rõ rệt giữa các khu vực, và vị trí địa lý là yếu tố ảnh hưởng mạnh nhất đến giá bán.
  + 1. Case 3: So sánh Price và PriceperSqm
* Input:

A screenshot of a computer code

AI-generated content may be incorrect.

* Output:

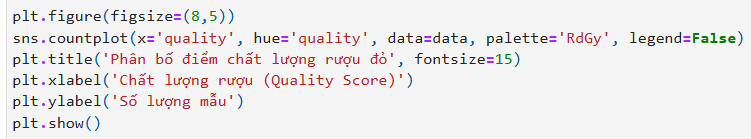
A comparison of a graph

AI-generated content may be incorrect.

* + Khu vực “JS012” nổi bật rõ rệt với giá vượt trội trên 25.000/m².
  + Cho thấy đây là khu vực đắt đỏ nhất, có thể nằm ở trung tâm thành phố hoặc khu phát triển cao cấp.
  + Hoặc do diện tích nhà ở trung bình nhỏ nhưng giá tổng thể cao, nên giá trên m² tăng mạnh.
  + Các khu vực như “1017” và “E1075” cũng có giá trên m² khá cao (khoảng 13.000–15.000/m²). Có thể là vùng có nhu cầu cao, cơ sở hạ tầng tốt, hoặc gần trung tâm.
  + Những khu vực còn lại (ví dụ: AE017, KV015, BM076, XP...) có giá trên m² dao động quanh 8.000–10.000. Cho thấy mức giá trung bình hoặc thấp hơn, có thể là vùng ngoại ô hoặc ít sôi động hơn.
  + Sự chênh lệch rõ rệt giữa khu vực JS012 và các khu khác chứng tỏ thị trường nhà ở phân hóa mạnh. Giá trị bất động sản phụ thuộc nhiều vào vị trí địa lý và tiềm năng khu vực.
  1. **Trực quan hóa dữ liệu về phân loại chất lượng rượu đỏ**

2.1. Biểu đồ Histogram ( Phân bố điểm chất lượng rượu )

* + Input:



* + Output:

A graph of different colored squares

AI-generated content may be incorrect.

* Chất lượng Dữ liệu Tổng quát: Cả hai tập dữ liệu đều có chất lượng cao về tính đầy đủ, vì không chứa bất kỳ giá trị thiếu nào
* Thách thức về Phân phối Biến: Một số biến quan trọng (như Insulin, BloodPressure, và DiabetesPedigreeFunction) thể hiện sự lệch phân phối mạnh.
* Vấn đề Không cân bằng Lớp (Tiểu đường): Tỷ lệ giữa "Không tiểu đường" (Outcome = 0) và "Bị tiểu đường" (Outcome = 1) là xấp xỉ 2:1. Sự không cân bằng này cần được xử lý để tránh mô hình thiên vị lớp đa số.
* Vấn đề Không cân bằng Lớp (Rượu): Sự mất cân bằng lớp là rất nghiêm trọng; gần 90% số mẫu rượu tập trung ở điểm chất lượng 5 và 6. Các lớp chất lượng kém (3, 4) và xuất sắc (7, 8) là lớp thiểu số cực đoan, đòi hỏi các kỹ thuật xử lý không cân bằng nâng cao hoặc chuyển đổi bài toán thành phân loại nhị phân.

2.2. Phân bố một số đặc trưng hóa học

* Input:

A computer code with colorful text

AI-generated content may be incorrect.

* Output:

A graph of a graph

AI-generated content may be incorrect.

Phân tích phân bố của các đặc trưng hóa học cho thấy sự lệch phân phối phổ biến, là yếu tố cần thiết phải xử lý trước khi huấn luyện mô hình:

* Lệch Phải Nghiêm trọng: Các biến residual sugar và sulphates thể hiện sự lệch phải rất mạnh, cho thấy hầu hết các mẫu rượu có giá trị thấp và chỉ một số ít có giá trị ngoại lệ rất cao. Các biến này rất cần được biến đổi dữ liệu (ví dụ: transformation) để đạt được phân phối đối xứng hơn.
* Phân phối Đặc biệt: Biến citric acid có hình dạng phân bố hai đỉnh (bimodal), gợi ý rằng có thể có hai nhóm mẫu rượu khác biệt tồn tại trong tập dữ liệu.
* Phân phối Tương đối Tốt: Biến volatile acidity có phân bố gần như đối xứng, cho thấy dữ liệu này có thể được sử dụng trực tiếp mà ít cần biến đổi.

Tóm lại: Hầu hết các đặc trưng hóa học cần được xử lý tiền mô hình hóa bằng các kỹ thuật biến đổi để giảm độ xiên và cải thiện hiệu suất của mô hình.

2.3. Ma trận tương quan giữa các biến hóa học và chất lượng

* Input:

A computer code with colorful text

AI-generated content may be incorrect.

* Output:

A chart with different colored squares

AI-generated content may be incorrect.

* Về tình trạng tiểu đường (Outcome):
* Glucose là yếu tố dự báo mạnh nhất: Nồng độ Glucose có mối tương quan dương mạnh nhất với Outcome (0.47), và giá trị trung vị ở nhóm mắc bệnh (≈140) cao hơn hẳn nhóm không mắc (≈105).
* Các yếu tố quan trọng khác: BMI (0.29) và Age (0.24) cũng là những yếu tố có tương quan dương mạnh mẽ với nguy cơ mắc bệnh.
* Về chất lượng rượu (Quality):
* Tác động tích cực: Alcohol (0.48) là yếu tố hóa học có mối tương quan dương mạnh nhất, cho thấy nồng độ cồn cao thường đi kèm với chất lượng rượu tốt hơn.
* Tác động tiêu cực: Volatile Acidity (-0.39) có mối tương quan âm mạnh nhất, ngụ ý độ axit dễ bay hơi cao làm giảm chất lượng rượu.
* Về mối quan hệ giữa các biến độc lập:
  + Pregnancies và Age có mối tương quan rất mạnh (0.54).
  + Fixed Acidity có mối liên hệ mạnh với Density (0.67) và pH (-0.68), cho thấy sự phụ thuộc phức tạp của các yếu tố axit.

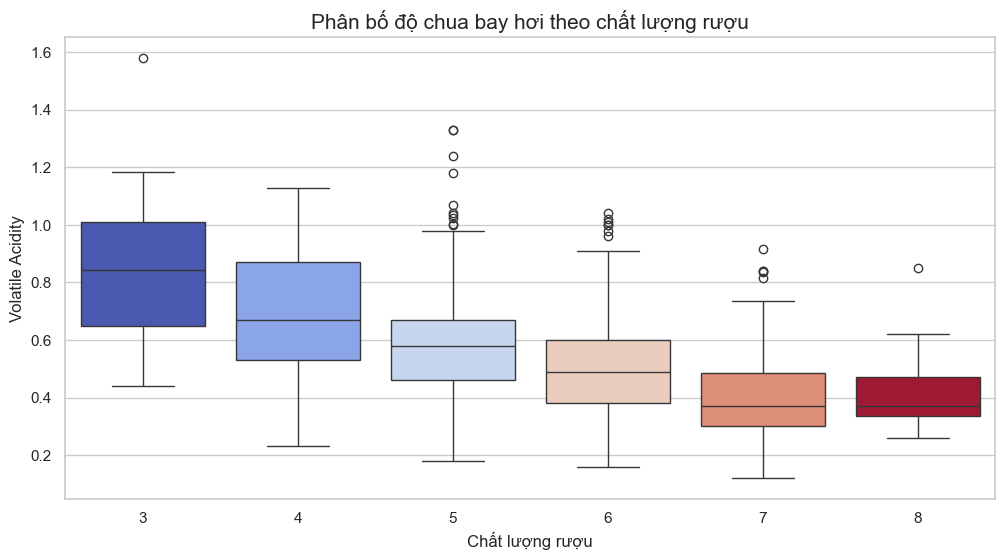
2.4. Biểu đồ boxplot: So sánh alcohol và volatile acidity theo chất lượng

* Input:

A computer screen shot of a computer code

AI-generated content may be incorrect.

* Output:



A graph of different colored squares

AI-generated content may be incorrect.

Phân tích xác nhận và là hai yếu tố phân biệt chất lượng mạnh nhất:

* Tương quan Ngược Dấu: có tương quan dương với chất lượng (trung vị tăng từ 3 đến 8), trong khi có tương quan âm (trung vị giảm từ 3 đến 8).
* Đặc trưng Chất lượng Cao: Rượu đạt chất lượng cao nhất (điểm 7, 8) được đặc trưng bởi tổ hợp Độ cồn cao nhất và Độ chua bay hơi thấp nhất với sự chồng chéo ít nhất với các lớp thấp hơn.
* Thách thức Mô hình: Sự chồng chéo lớn giữa các hộp của lớp trung bình (điểm 5 và 6) cho thấy và sẽ khó phân biệt chính xác các mức chất lượng này, nhưng rất hiệu quả trong việc phân tách các thái cực (rượu rất tốt và rất kém).

2.5. Biểu đồ Scatter plot: Mối quan hệ giữa độ cồn và độ axit bay hơi

* Input:

A close-up of a computer screen

AI-generated content may be incorrect.

* Output:

A graph with red and blue dots

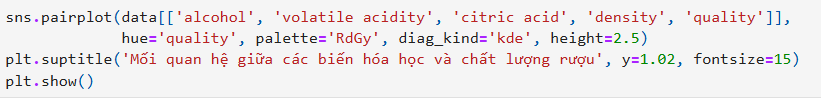
AI-generated content may be incorrect.

Phân tích biểu đồ phân tán xác nhận hai xu hướng dự báo mạnh về chất lượng rượu:

* Rượu Chất lượng Cao (Điểm 7, 8): Tập trung ở vùng Alcohol cao () và Volatile Acidity thấp (). Tổ hợp này là dấu hiệu dự báo mạnh cho chất lượng vượt trội.
* Rượu Chất lượng Kém (Điểm 3, 4): Tập trung ở vùng Alcohol thấp và Volatile Acidity cao.
* Thách thức Lớp Trung bình: Các lớp chất lượng trung bình (Điểm 5, 6) có sự trùng lặp và chồng chéo lớn trên toàn bộ biểu đồ. Điều này chỉ ra rằng hai biến này không đủ để phân biệt chính xác chất lượng rượu ở mức trung bình, nhưng rất hiệu quả để phân tách rượu rất tốt và rất kém.

2.6. Biểu đồ Pairplot: Mối quan hệ tổng quát

* Input:



* Output:

A graph of a number of dots

AI-generated content may be incorrect.

Phân tích tương quan cho thấy các yếu tố hóa học chính ảnh hưởng đến chất lượng rượu () là:

* Tương quan Dương Mạnh: Alcohol (nồng độ cồn) có mối quan hệ tương quan dương mạnh nhất với chất lượng. Rượu chất lượng cao (điểm 7, 8) thường có nồng độ cồn cao hơn (thường ).
* Tương quan Âm Mạnh: Volatile acidity (axit dễ bay hơi) và density (mật độ) có mối quan hệ tương quan âm mạnh. Rượu chất lượng cao thường có hàm lượng axit dễ bay hơi và mật độ thấp hơn.
* Thách thức Mô hình hóa: Mặc dù có xu hướng rõ ràng, sự phân tách lớp không hoàn hảo và sự pha trộn lớn giữa các lớp chất lượng trung bình (5, 6, 7) cho thấy cần phải sử dụng các thuật toán học máy phi tuyến tính để đạt được độ chính xác dự đoán cao.
* Cộng tuyến Tiềm tàng: Tồn tại mối tương quan âm mạnh giữa và , cảnh báo về khả năng cộng tuyến (multicollinearity) cần được xử lý trong bước tiền xử lý.
  1. **Trực quan hóa dữ liệu về bệnh tiểu đường**

3.1. Biểu đồ Histogram: Phân bố của các biến tình trạng mắc bệnh tiểu đường

* Input:

A close-up of a computer screen

AI-generated content may be incorrect.

* Output:

A graph of a graph

AI-generated content may be incorrect.

Tập dữ liệu đang trong tình trạng mất cân bằng (Imbalanced):

* Lớp 0 (Không tiểu đường): Có số lượng mẫu lớn hơn, khoảng 500 quan sát.
* Lớp 1 (Có tiểu đường): Có số lượng mẫu nhỏ hơn, khoảng 270 quan sát.

3.2. Phân phối các biến đặc tửng nổi bật

* Input:

A white background with red and blue text

AI-generated content may be incorrect.

* Output:



* Glucose: Phân phối gần Chuẩn (Normal), tập trung nhiều nhất quanh 100-125 .
* BMI: Phân phối gần Chuẩn (Normal), hơi lệch phải, tập trung chủ yếu quanh 30-35.
* Age (Tuổi): Phân phối lệch phải mạnh (Strongly Right-Skewed), đa số đối tượng nằm trong độ tuổi 20-30.
* Insulin: Phân phối lệch phải cực kỳ mạnh (Extremely Right-Skewed), với số lượng lớn quan sát tập trung ở giá trị rất thấp (gần 0).

3.3. Ma trận tương quan

* Input:

A white background with black and red text

AI-generated content may be incorrect.

* Output:

A screenshot of a graph

AI-generated content may be incorrect.

* Mối tương quan với Tình trạng Tiểu đường (Outcome)
* Glucose (0.47): Có tương quan mạnh nhất và dương với Outcome. Đây là yếu tố dự đoán quan trọng nhất.
* BMI (0.29): Tương quan dương tương đối mạnh với Outcome.
* Age (Tuổi) (0.24): Tương quan dương tương đối mạnh với Outcome.
* BloodPressure (0.065): Tương quan rất yếu với Outcome, cho thấy ít có vai trò dự đoán.
* Mối tương quan giữa các Biến độc lập
* Pregnancies và Age (0.54): Có tương quan rất mạnh và dương (hợp lý về mặt sinh học).
* SkinThickness và Insulin (0.44): Tương quan mạnh và dương.
* SkinThickness và BMI (0.39): Tương quan trung bình đến mạnh và dương.
* Glucose và Insulin (0.33): Tương quan trung bình và dương.

3.4. Biểu đồ Boxplot: Phân bố Glucode theo tình trạng tiểu đường

* Input:

A white background with red text

AI-generated content may be incorrect.

* Output:

A diagram of a graph

AI-generated content may be incorrect.

* Giá trị trung vị (Median): Trung vị Glucose ở nhóm Có tiểu đường (≈140mg/dL) cao hơn đáng kể so với nhóm Không tiểu đường (≈105mg/dL).
* Phân bố: Toàn bộ phạm vi tứ phân vị (IQR) của nhóm Có tiểu đường (≈120 đến ≈170mg/dL) nằm ở mức cao hơn so với nhóm Không tiểu đường (≈95 đến ≈125mg/dL), cho thấy Glucose là một yếu tố dự báo mạnh mẽ cho tình trạng tiểu đường.

3.5. Phân phối của một biến quan trọng theo Outcome

* Input:

A computer screen shot of a person

AI-generated content may be incorrect.

* Output:

A graph of blood glucose

AI-generated content may be incorrect.

A graph of age and age

AI-generated content may be incorrect.A graph of different colored lines

AI-generated content may be incorrect.

Phân tích ban đầu trên cả hai tập dữ liệu xác định ba thách thức chính về chất lượng và phân phối dữ liệu cần được xử lý trong giai đoạn tiền xử lý:

a) Vấn đề Không cân bằng Lớp (Class Imbalance)

* Rượu: Không cân bằng nghiêm trọng. Gần 90% số mẫu tập trung ở điểm chất lượng trung bình (5 và 6). Các lớp chất lượng kém (3, 4) và xuất sắc (7, 8) là lớp thiểu số cực đoan.
* Tiểu đường: Không cân bằng rõ rệt với tỷ lệ xấp xỉ 2:1 (65% không bệnh, 35% có bệnh).
* Hành động*:* Cần áp dụng kỹ thuật xử lý không cân bằng (SMOTE hoặc ADASYN ) hoặc chuyển đổi sang bài toán nhị phân.

b) Dữ liệu Lệch Phân phối (Skewness)

* Nhiều biến quan trọng trong cả hai tập dữ liệu bị lệch phân phối mạnh
  + Tiểu đường: Insulin (≈2.27) và DiabetesPedigreeFunction (≈1.92) lệch phải.
  + Rượu: Residual Sugar và Sulphates lệch phải nghiêm trọng.
* Hành động: Cần thực hiện biến đổi dữ liệu để làm cho phân phối đối xứng hơn, cải thiện hiệu suất của các mô hình tuyến tính.

c) Các Biến Dự báo Chủ đạo

* Rượu: Các biến có khả năng phân biệt mạnh nhất là (tương quan dương) và (tương quan âm). Rượu chất lượng cao (Quality≥7) có cao và thấp.
* Tiểu đường: Các biến phân biệt mạnh nhất là và . Nhóm bệnh nhân tiểu đường tập trung ở vùng cao (≥140) và cao (≥30)
* Thách thức*:* Sự chồng chéo lớn của dữ liệu ở mức trung bình trong cả hai bài toán cho thấy cần sử dụng các thuật toán học máy phi tuyến tính để đạt được độ chính xác dự đoán cao.

3.6. Biểu đồ Scatter plot: Mối quan hệ giữa glucose và BMI

* Input:

A screen shot of a computer

AI-generated content may be incorrect.

* Output:

A diagram of a substance

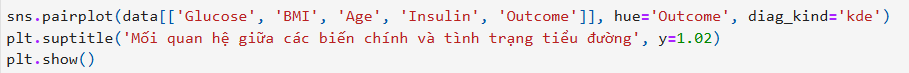
AI-generated content may be incorrect.

Phân tích và xác nhận đây là hai đặc trưng dự báo chính cho bệnh tiểu đường:

* Vùng Nguy cơ Cao: Nhóm Bị tiểu đường (Outcome = 1) tập trung rõ rệt ở khu vực cao (>=140) và cao (>=30)
* Vấn đề Dữ liệu Ẩn: Các giá trị và bằng 0 là giá trị thiếu được mã hóa sai và cần được xử lý bằng kỹ thuật thay thế (imputation) trước khi mô hình hóa.
* Thách thức Phân loại: Tồn tại sự chồng chéo đáng kể giữa hai lớp trong khu vực trung bình (Glucose: 100-140, BMI: 25-35 ). Điều này chỉ ra rằng và không đủ để phân loại hoàn hảo, cần phải kết hợp với các biến khác (như và ) trong mô hình.

3.7. Biểu đồ Pairplot: Mối quan hệ giữa các biến

* Input:



* Output:

A group of blue and orange dots

AI-generated content may be incorrect.

Phân tích tập dữ liệu cho thấy các điểm chính sau:

* Không cân bằng Lớp: Biến mục tiêu () bị không cân bằng với tỷ lệ khoảng 2:1 (65% không bệnh, 35% có bệnh). Cần xử lý để tránh sai lệch mô hình.
* Phân phối Lệch và Ngoại lai: Nhiều biến (đặc biệt là Insulin (Skewness ) và DiabetesPedigreeFunction) bị lệch phải rất mạnh. Điều này yêu cầu biến đổi dữ liệu () để cải thiện hiệu suất mô hình.
* Các Biến Dự báo Chính:
  + Glucose và BMI là các yếu tố dự báo mạnh nhất. Nhóm bệnh nhân tiểu đường () có xu hướng tập trung ở khu vực có nồng độ Glucose và chỉ số BMI cao hơn.
  + Age cũng có mối quan hệ dương với khả năng mắc bệnh.
* Thách thức Mô hình: Dù có xu hướng rõ ràng, sự chồng chéo lớn giữa hai nhóm bệnh và không bệnh trong các biểu đồ phân tán cho thấy không có ngưỡng đơn lẻ nào có thể phân loại hoàn hảo. Cần sử dụng các mô hình học máy phức tạp (phi tuyến tính) để kết hợp các đặc trưng.

# **C. Phân tích đơn biến và hai biến**

## **I. Ôn tập lý thuyết**

### **1. 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) là bước đầu tiên trong quá trình khám phá dữ liệu, khi ta chỉ xem xét một biến duy nhất để hiểu rõ đặc điểm, phân bố và xu hướng của nó. Mục đích chính của phân tích đơn biến là mô tả dữ liệu bằng các chỉ số thống kê cơ bản, giúp ta nhận biết giá trị điển hình, mức độ phân tán cũng như hình dạng của phân bố dữ liệu. Ví dụ, khi ta xem điểm thi trung bình của sinh viên hoặc chiều cao trung bình của một nhóm người, đó chính là phân tích đơn biến.

Trong khi đó, phân tích hai biến (Bivariate Analysis) được dùng để xem xét mối quan hệ giữa hai biến với nhau — có thể là giữa một biến độc lập và một biến phụ thuộc, hoặc giữa hai biến định lượng. Phân tích hai biến giúp ta trả lời những câu hỏi như: “Số giờ học có ảnh hưởng đến điểm thi không?”, hay “Chiều cao và cân nặng có liên quan với nhau không?”. Nói cách khác, nếu phân tích đơn biến chỉ tập trung vào đặc điểm của một biến, thì phân tích hai biến giúp ta hiểu mức độ liên hệ hoặc tác động lẫn nhau giữa hai yếu tố trong dữ liệu.

### **2. Các thước đo thống kê nào thường được sử dụng trong phân tích đơn biến (ví dụ: trung bình, trung vị, mode, độ lệch chuẩn)?**

Để mô tả một biến, ta thường dùng các thước đo thống kê cơ bản.

* Trung bình (Mean) cho biết giá trị đại diện của dữ liệu, bằng cách cộng tất cả các giá trị rồi chia cho số lượng quan sát.
* Trung vị (Median) là giá trị nằm giữa khi sắp xếp các số theo thứ tự, phản ánh xu hướng trung tâm mà ít bị ảnh hưởng bởi các giá trị ngoại lai.
* Mốt (Mode) là giá trị xuất hiện nhiều nhất trong tập dữ liệu.
* Độ lệch chuẩn (Standard deviation) và phương sai (Variance) cho biết dữ liệu có phân tán hay không, tức là các giá trị có gần trung bình hay bị trải rộng.
* Độ lệch (Skewness) và độ nhọn (Kurtosis) giúp ta hiểu hình dạng của phân bố dữ liệu — ví dụ, dữ liệu có lệch về bên trái hay bên phải, hoặc có nhọn hơn hay dẹt hơn so với phân phối chuẩn.

Những thước đo này giúp ta nắm bắt được bản chất của dữ liệu, phát hiện điểm bất thường, và chuẩn bị cho bước phân tích sâu hơn ở giai đoạn tiếp theo.

### **3. Trong phân tích hai biến, làm thế nào để xác định mối quan hệ giữa hai biến (ví dụ: tương quan, nhân quả)?**

Khi phân tích hai biến, ta thường muốn biết liệu có mối quan hệ nào giữa chúng hay không, và nếu có thì mối quan hệ đó mạnh hay yếu. Một trong những cách phổ biến nhất là sử dụng hệ số tương quan (Correlation Coefficient).

Hệ số tương quan, thường được ký hiệu là rrr, nằm trong khoảng từ -1 đến 1.

* Nếu rrr gần 1, hai biến có mối quan hệ tuyến tính cùng chiều (một tăng thì biến kia cũng tăng).
* Nếu rrr gần -1, hai biến có mối quan hệ ngược chiều (một tăng thì biến kia giảm).
* Nếu rrr gần 0, điều đó cho thấy không có mối quan hệ tuyến tính rõ ràng giữa hai biến.

Bên cạnh đó, trong một số trường hợp, ta có thể quan tâm đến mối quan hệ nhân quả (Causation), tức là xác định xem biến này có phải là nguyên nhân gây ra thay đổi ở biến kia hay không. Tuy nhiên, cần lưu ý rằng tương quan không có nghĩa là nhân quả — hai biến có thể cùng thay đổi theo thời gian nhưng không nhất thiết có mối quan hệ nguyên nhân – kết quả. Để xác định quan hệ nhân quả, ta cần những phương pháp phân tích sâu hơn như hồi quy, thí nghiệm hoặc kiểm định giả thuyết.

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

Cả tương quan và hiệp biến đều là công cụ dùng để đo mức độ liên hệ giữa hai biến, nhưng chúng có những điểm khác biệt rõ ràng.

Hiệp biến (Covariance) thể hiện hai biến thay đổi cùng nhau như thế nào. Nếu hiệp biến dương, hai biến có xu hướng tăng hoặc giảm cùng chiều; nếu âm, một biến tăng thì biến kia giảm. Tuy nhiên, giá trị của hiệp biến phụ thuộc vào đơn vị đo của dữ liệu, nên rất khó so sánh giữa các cặp biến khác nhau.

Tương quan (Correlation) là phiên bản “chuẩn hóa” của hiệp biến. Nó loại bỏ ảnh hưởng của đơn vị đo, vì vậy kết quả luôn nằm trong khoảng từ -1 đến 1, giúp ta dễ dàng diễn giải và so sánh hơn. Nói đơn giản, hiệp biến cho biết hướng thay đổi, còn tương quan cho biết mức độ chặt chẽ của mối quan hệ đó.

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

Trong phân tích dữ liệu, việc trực quan hóa bằng biểu đồ là cách rất hiệu quả để hiểu dữ liệu nhanh hơn và phát hiện xu hướng tiềm ẩn.

Khi chỉ có một biến, tức là phân tích đơn biến, ta có thể sử dụng:

* Histogram (biểu đồ tần suất) để xem phân bố của dữ liệu.
* Boxplot (biểu đồ hộp) để phát hiện các giá trị ngoại lai hoặc mức độ lệch.
* Bar chart (biểu đồ cột) khi dữ liệu là dạng phân loại.

Còn khi phân tích hai biến, việc chọn biểu đồ phụ thuộc vào loại biến:

* Nếu cả hai biến là số, nên dùng scatter plot (biểu đồ phân tán) hoặc heatmap (bản đồ nhiệt) để thấy mối tương quan.
* Nếu một biến là số và biến còn lại là phân loại, nên dùng boxplot hoặc violin plot để so sánh sự khác biệt giữa các nhóm.

### **6. Đoạn code mẫu để tạo biểu đồ scatter plot hoặc heatmap để phân tích mối quan hệ giữa hai biến?**

Dưới đây là một số ví dụ cơ bản minh họa cho cách trực quan hóa mối quan hệ giữa các biến bằng Python:

* Biểu đồ Scatter Plot ( mối quan hệ giữa hai biến )

**A computer screen shot of a computer code

AI-generated content may be incorrect.**

* Biểu đồ Heatmap ( hiển thị mức độ tương quan )

**A close-up of a computer screen

AI-generated content may be incorrect.**

* Biểu đồ Boxplot và Violin Plot ( giữa biến số và biến phân loại )

**A screenshot of a computer code

AI-generated content may be incorrect.**

### **7. Làm thế nào để trực quan hóa mối quan hệ giữa một biến số và một biến phân loại bằng biểu đồ boxplot hoặc violin plot trong Python?**

Khi muốn phân tích mối quan hệ giữa một biến định lượng (biến số) và một biến định tính (biến phân loại), hai loại biểu đồ thường được sử dụng là boxplot (biểu đồ hộp) và violin plot (biểu đồ hình violin).

Boxplot giúp thể hiện phân bố của dữ liệu thông qua các giá trị thống kê như trung vị, tứ phân vị và các điểm ngoại lai. Nó cho thấy rõ mức độ phân tán và sự khác biệt giữa các nhóm trong biến phân loại. Ví dụ, ta có thể dùng boxplot để so sánh điểm thi trung bình giữa các nhóm giới tính hoặc giữa các lớp học khác nhau.

Trong khi đó, violin plot cũng thể hiện phân bố dữ liệu tương tự boxplot nhưng bổ sung thêm đường cong mật độ (density), giúp ta thấy rõ hơn hình dạng phân bố của dữ liệu trong từng nhóm. Nhờ đó, violin plot trực quan và sinh động hơn khi muốn mô tả sự khác biệt về phân bố giữa các nhóm.

Cả hai loại biểu đồ này đều rất hữu ích trong việc nhận diện sự khác biệt giữa các nhóm và phát hiện những giá trị bất thường.

Việc sử dụng boxplot hoặc violin plot không chỉ giúp hiểu sâu hơn về mối quan hệ giữa các biến mà còn hỗ trợ trong việc phát hiện sự khác biệt, xu hướng hoặc những giá trị bất thường trong dữ liệu.

## **II. Bài làm ứng dụng**

* 1. **Phân tích dữ liệu đơn biến trên dữ liệu chim cánh cụt**
  2. Phân tích các biến định tính (Categorical Variables)
* Input:

A screenshot of a computer code

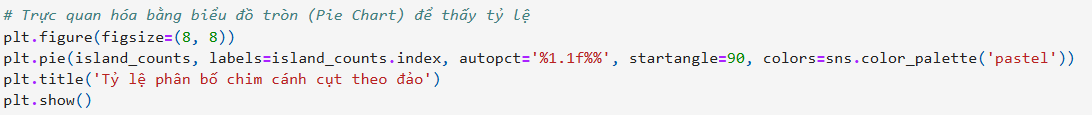
AI-generated content may be incorrect.

* Output:

A graph with different colored squares

AI-generated content may be incorrect.

* Tập dữ liệu này là mất cân bằng (imbalanced) về số lượng mẫu giữa các loài.
* Sự chênh lệch lớn nhất nằm giữa loài Adelie và Chinstrap, với số lượng mẫu của Adelie gấp hơn hai lần số lượng mẫu của Chinstrap.
* Khi xây dựng mô hình phân loại (Classification Model) để phân biệt các loài chim cánh cụt, mô hình có thể thiên vị (bias) các loài có số lượng lớn hơn (Adelie và Gentoo) và dự đoán kém chính xác hơn đối với loài Chinstrap.
  1. Phân tích biến Island
* Input:



* Output:

A pie chart with numbers and text

AI-generated content may be incorrect.

* Tập dữ liệu đang nghiêng về phía các cá thể chim cánh cụt được thu thập từ Đảo Biscoe, chiếm gần một nửa tổng số mẫu.
* Sự phân bố này cho thấy tập dữ liệu không đồng đều về mặt địa lý.
* Khi phân tích sâu hơn, cần lưu ý đến yếu tố này vì các đặc điểm của chim cánh cụt có thể bị ảnh hưởng bởi môi trường sống khác nhau trên ba đảo.
  1. Phân tích biến sex ( giới tính )
* Input:

A screenshot of a computer code

AI-generated content may be incorrect.

* Output:

A graph of a person and person

AI-generated content may be incorrect.

* Tập dữ liệu này cân bằng tốt (well-balanced) về giới tính.
* Số lượng chim cánh cụt nam và nữ gần như ngang bằng nhau.
* Sự cân bằng này rất tốt cho việc xây dựng mô hình học máy, giúp mô hình không bị thiên vị về một giới tính cụ thể nào.
  1. Phân tích biến culmen\_length\_mm (Độ dài mỏ)
* Input:

A screenshot of a computer code

AI-generated content may be incorrect.

* Output:

A blue rectangle with numbers

AI-generated content may be incorrect.

* Trung vị: Đường kẻ trong Box Plot nằm ở khoảng 44mm.
* Phạm vi Chính: Phần lớn dữ liệu (IQR - 50% ở giữa) nằm trong khoảng từ 40mm đến 49mm.
* Giá trị Ngoại lai: Biểu đồ Hộp không hiển thị rõ ràng các điểm ngoại lai (outliers) riêng lẻ, cho thấy dữ liệu phân bố tương đối tốt trong phạm vi tối thiểu và tối đa.
* Giá trị tối thiểu khoảng 32mm.
* Giá trị tối đa khoảng 60mm.

A graph with blue lines

AI-generated content may be incorrect.

* Hình dạng: Biểu đồ Histogram và đường KDE cho thấy phân bố là đa đỉnh (Multimodal), cụ thể là hai đỉnh rõ rệt (Bimodal).
* Đỉnh thứ nhất tập trung quanh 40mm.
* Đỉnh thứ hai tập trung quanh 50mm.
* Nguyên nhân: Sự xuất hiện của hai đỉnh này gợi ý rằng biến này có thể được tạo thành từ dữ liệu của hai nhóm (loài) khác nhau với các giá trị trung bình về độ dài mỏ khác biệt.
  1. Phân tích biến body\_mass\_g (Khối lượng cơ thể)
* Input:

A screenshot of a computer code

AI-generated content may be incorrect.

* Output:

A green rectangular object with numbers

AI-generated content may be incorrect.

* Trung vị: Đường kẻ trong Box Plot nằm ở khoảng 4.000g.
* Phạm vi Chính: Phần lớn dữ liệu (IQR - 50% ở giữa) nằm trong khoảng từ 3.500g đến 4.750g.
* Giá trị Ngoại lai: Biểu đồ Hộp không hiển thị rõ ràng các điểm ngoại lai (outliers) riêng lẻ, cho thấy dữ liệu phân bố tương đối tốt trong phạm vi tối thiểu và tối đa.
* Giá trị tối thiểu khoảng 2.700g.
* Giá trị tối đa khoảng 6.200g.

A green and white graph

AI-generated content may be incorrect.

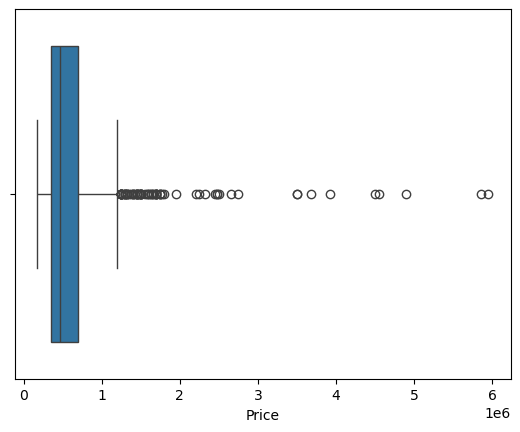
* Hình dạng: Tương tự như độ dài mỏ, phân bố của khối lượng cơ thể là đa đỉnh (Multimodal), cụ thể là hai đỉnh rõ rệt (Bimodal).
* Đỉnh thứ nhất tập trung quanh 3.500 – 4.000g.
* Đỉnh thứ hai, ít rõ rệt hơn, tập trung quanh 4.750 – 5.500g.
* Nguyên nhân: Sự xuất hiện của hai đỉnh này gợi ý rằng biến này được tạo thành từ dữ liệu của hai nhóm (loài) khác nhau với các giá trị trung bình về khối lượng cơ thể khác biệt.

1. **Phân tích dữ liệu đơn biến trên dữ liệu về giá nhà**
   1. Phân tích dữ liệu đơn biến dựa vào Boxplot
      1. Case 1

* Input:



* Output:



* Phần lớn giá nhà tập trung ở mức thấp, nằm trong khoảng dưới 1 triệu đơn vị tiền tệ.Điều này cho thấy đa số nhà ở trong tập dữ liệu có giá trung bình hoặc thấp.
* Có rất nhiều điểm ngoại lệ ở phía bên phải (các chấm rải rác đến 6 triệu).
* Nghĩa là một số khu vực có giá nhà cao vượt trội, có thể là nhà sang, biệt thự hoặc nằm ở khu trung tâm đắt đỏ.
* Phân bố giá nhà bị lệch phải (right-skewed).Cho thấy đa số giá nhà thấp, chỉ có một số ít có giá rất cao, làm tăng trung bình tổng thể.
* Độ trải rộng của dữ liệu khá lớn (range rộng), thể hiện sự chênh lệch đáng kể về giá giữa các khu vực.
  + 1. Case 2
* Input:

A close-up of a computer code

AI-generated content may be incorrect.

* Output:

A graph of a house price

AI-generated content may be incorrect.

* Phần lớn giá nhà tập trung ở mức thấp, chủ yếu dưới 1 triệu đơn vị tiền tệ.Điều này cho thấy đa số nhà trong tập dữ liệu thuộc phân khúc giá trung bình hoặc thấp.
* Có nhiều điểm dữ liệu nằm xa bên phải, thậm chí đến 6 triệu, cho thấy nhiều giá trị ngoại lệ ở mức rất cao.Đây có thể là những căn nhà sang trọng, biệt thự hoặc ở vị trí trung tâm đắt đỏ.
* Phân bố giá nhà bị lệch phải (Right-skewed distribution).Nghĩa là đa số giá nhà ở mức thấp, chỉ có một số ít căn có giá rất cao kéo trung bình tăng lên.
* Khoảng biến thiên dữ liệu (IQR) nhỏ nhưng có nhiều ngoại lệ xa, phản ánh sự chênh lệch lớn giữa nhà bình dân và cao cấp trên thị trường.
  1. Phân tích dữ liệu đơn biến dựa vào violin plot
     1. Case 1
* Input:



* Output:

A graph of a function

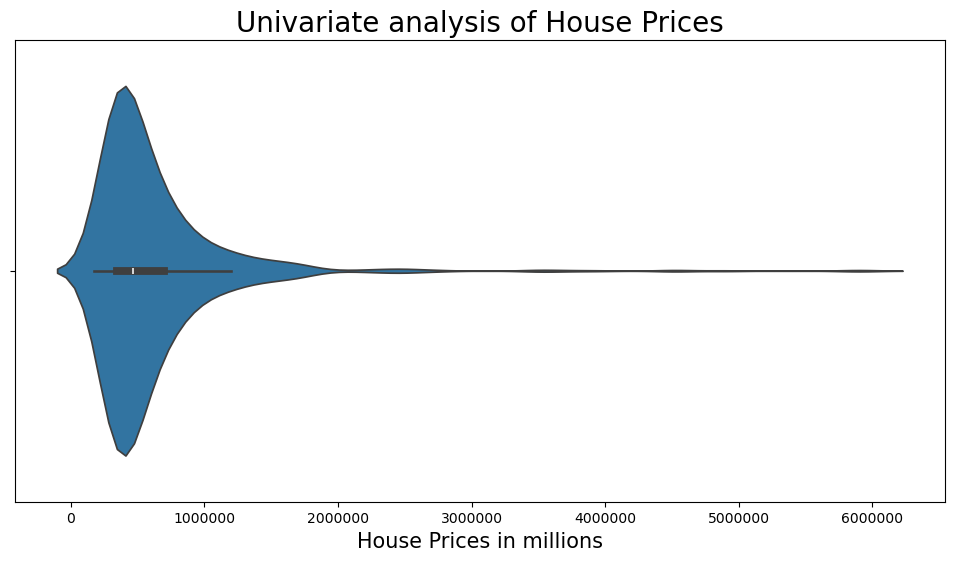
AI-generated content may be incorrect.

* Phần lớn giá nhà tập trung ở mức thấp, chủ yếu dưới 1 triệu đơn vị.  
   → Khu vực này phình to, chứng tỏ đa số bất động sản có giá bình dân hoặc trung bình.
* Phần kéo dài mảnh sang bên phải cho thấy có nhiều giá trị ngoại lệ – một số căn nhà có giá cực cao lên tới trên 6 triệu.  
   → Điều này thể hiện sự chênh lệch lớn giữa các phân khúc thị trường.
* Phân bố bị lệch phải mạnh (Right-skewed distribution).  
   → Nghĩa là phần lớn dữ liệu tập trung ở mức thấp, chỉ có ít giá trị cực cao kéo trung bình tăng lên.
* Độ biến thiên cao, chứng tỏ giá nhà không đồng nhất, chịu ảnh hưởng mạnh từ vị trí, diện tích, và điều kiện khu vực.
  + 1. Case 2
* Input:

A screenshot of a computer code

AI-generated content may be incorrect.

* Output:



* Phần lớn giá nhà tập trung ở mức thấp (phần phình to nằm gần 0 → 1 triệu).  
   → Cho thấy đa số bất động sản trong dữ liệu thuộc nhóm giá thấp hoặc trung bình.
* Phần đuôi dài kéo về bên phải (Right-skewed distribution)  
   → Biểu hiện của một số giá trị ngoại lệ rất cao, có thể lên tới hơn 6 triệu.  
   → Điều này phản ánh thị trường nhà ở có sự chênh lệch rõ rệt, với ít căn giá cực cao so với phần lớn còn lại.
* Giá trị trung vị nhỏ hơn trung bình, do dữ liệu bị lệch phải mạnh.  
   → Khi phân tích thống kê hoặc huấn luyện mô hình, nên chuẩn hóa dữ liệu hoặc lấy log(Price) để giảm ảnh hưởng của các giá trị cực đại.
* Biến động giá khá lớn, thể hiện độ phân tán cao của dữ liệu, cho thấy giá nhà chịu ảnh hưởng bởi nhiều yếu tố như vị trí, diện tích, tiện ích,....
  1. Phân tích dữ liệu đơn biến dựa vào bảng tóm tắt giá trị
* Input:



* Output:



* Bộ dữ liệu gồm 924 mẫu, cho thấy giá nhà trung bình khoảng 622,000, trong khi trung vị thấp hơn (467,000) → dữ liệu lệch phải, tồn tại nhiều giá trị ngoại lệ với giá cao nhất tới 5,950,000.
* Diện tích nhà trung bình 96 m², đa số nằm trong khoảng 60–113 m², nhưng có một số căn rất lớn (623 m²).
* Số phòng trung bình khoảng 3–4, tối đa 14 phòng, thể hiện sự đa dạng về quy mô nhà.
* Nhìn chung, cả ba biến đều lệch phải

1. **Phân tích dữ liệu hai biến trên dữ liệu chim cánh cụt** 
   1. Phân tích dữ liệu 2 biến dựa vào phương pháp scatterplot

3.1.1. Case 1

* Input:



* Output:

A diagram of a number of blue dots

AI-generated content may be incorrect.

* Tương quan dương mạnh: Hai biến có mối tương quan dương mạnh mẽ. Khi Độ dài mỏ (culmen length) tăng lên, Khối lượng cơ thể (body mass) cũng có xu hướng tăng theo.
* Sự tập trung của dữ liệu: Dữ liệu không tạo thành một cụm duy nhất mà có vẻ như chia thành hai hoặc ba nhóm riêng biệt.
* Nhóm 1 (Nhỏ): Tập trung ở phía dưới bên trái (mỏ ngắn, khối lượng nhẹ), khoảng và .
* Nhóm 2 (Lớn): Tập trung ở phía trên bên phải (mỏ dài, khối lượng nặng), khoảng và .
* Khả năng phân loại: Sự phân tách rõ ràng thành các cụm này cho thấy hai biến này kết hợp với nhau tạo ra một khả năng phân biệt rất cao giữa các loài chim cánh cụt. Các nhóm này gần như chắc chắn tương ứng với các loài khác nhau (Adelie, Chinstrap, Gentoo).

3.1.2. Case 2

* Input:

A close-up of a computer code

AI-generated content may be incorrect.

* Output:

A diagram of a number of blue dots

AI-generated content may be incorrect.

* Tương quan dương mạnh: Có một mối tương quan dương mạnh mẽ giữa hai biến. Khi Độ dài mỏ tăng, Khối lượng cơ thể cũng có xu hướng tăng theo.
* Phân cụm rõ rệt: Điểm dữ liệu không phân bố ngẫu nhiên mà có vẻ như tụ lại thành hai hoặc ba cụm (clusters) riêng biệt, gợi ý về sự tồn tại của các nhóm cá thể khác nhau trong tập dữ liệu (rất có thể là các loài chim cánh cụt khác nhau).
* Cụm 1 (Nhỏ): Tập trung ở phía dưới bên trái, đại diện cho những con chim cánh cụt nhỏ, nhẹ (mỏ ngắn, khối lượng thấp, và ).
* Cụm 2 (Lớn): Tập trung ở phía trên bên phải, đại diện cho những con chim cánh cụt lớn, nặng (mỏ dài, khối lượng cao, và )
  1. Phân tích hai biến dựa vào bảng crosstab/two-way
* Input:

A white rectangular object with red text

AI-generated content may be incorrect.

* Output:

A screenshot of a screen

AI-generated content may be incorrect.

* Cân bằng tuyệt đối: Các loài Adelie ( 73 nam / 73 nữ) và Chinstrap (34 nam / 34 nữ) có sự cân bằng hoàn hảo về số lượng mẫu giữa hai giới tính.
* Gần cân bằng: Loài Gentoo gần như cân bằng, với 61 nam và 58 nữ.

Kết luận: Tập dữ liệu này rất cân bằng về phân bố giới tính trong mỗi loài chim cánh cụt.

* 1. Phân tích hai biến sử dụng pivot\_table
* Input:



* Output:

A screenshot of a computer

AI-generated content may be incorrect.

* Độ dài mỏ là một đặc trưng phân biệt quan trọng giữa loài Adelie và hai loài còn lại (Chinstrap và Gentoo) do có sự chênh lệch lớn về giá trị trung bình.
* Tuy nhiên, sự khác biệt về độ dài mỏ trung bình giữa Chinstrap và Gentoo là không đáng kể (≈1.3 mm), gợi ý rằng chỉ dựa vào độ dài mỏ sẽ khó phân biệt hai loài này.
  1. Phân tích hai biến sử dụng pairplot
* Input:

A close up of words

AI-generated content may be incorrect.

* Output:

A group of blue and black graphs

AI-generated content may be incorrect.

* Tương quan dương mạnh: Hai biến có mối tương quan dương mạnh mẽ. Cá thể có mỏ dài hơn thường có khối lượng cơ thể nặng hơn.
* Phân cụm rõ rệt: Điểm dữ liệu tách biệt thành hai hoặc ba cụm (clusters) rõ ràng. Điều này cho thấy sự khác biệt về kích thước và khối lượng giữa các nhóm (loài) chim cánh cụt.
* Khả năng phân biệt: Cả hai biến đều có phân phối đa đỉnh (Multimodal) trên trục đường chéo, xác nhận chúng là những đặc trưng phân biệt cực kỳ mạnh mẽ khi được sử dụng cùng nhau trong mô hình phân loại.

# **D. Tóm tắt thực hành**

Khám phá dữ liệu (Exploratory Data Analysis – EDA) là bước đầu tiên và quan trọng trong phân tích dữ liệu, giúp hiểu rõ đặc điểm, cấu trúc và mối quan hệ giữa các biến trước khi xây dựng mô hình hoặc rút ra kết luận.

Trong thực tế, EDA thường gặp nhiều khó khăn, đặc biệt là về chất lượng dữ liệu. Dữ liệu có thể bị thiếu, sai định dạng hoặc chứa giá trị ngoại lai. Vì vậy, người phân tích cần thực hiện tiền xử lý dữ liệu như: xử lý giá trị thiếu (loại bỏ, điền trung bình hoặc nội suy), phát hiện và loại bỏ ngoại lai, chuẩn hóa biến số và mã hóa biến phân loại. Việc này giúp đảm bảo dữ liệu sạch và đáng tin cậy trước khi phân tích sâu hơn.

Khi làm việc với dữ liệu lớn, EDA còn đối mặt với vấn đề về hiệu năng và khả năng trực quan. Để khắc phục, có thể sử dụng các kỹ thuật như lấy mẫu (sampling), đọc từng phần (chunking) hoặc dùng công cụ hỗ trợ dữ liệu lớn như *Dask* hay *PySpark*. Trong trực quan hóa, nên ưu tiên các biểu đồ tổng hợp như *heatmap, density plot* hay *hexbin plot* để dễ nhận thấy xu hướng tổng thể.

Python là công cụ phổ biến nhất cho EDA nhờ hệ sinh thái mạnh:

* *Pandas, NumPy*: thao tác và làm sạch dữ liệu.
* *Matplotlib, Seaborn*: trực quan hóa dữ liệu.
* *Scikit-learn*: xử lý giá trị thiếu, chuẩn hóa, mã hóa biến.
* *Dask, PySpark*: hỗ trợ dữ liệu lớn.

Để quá trình EDA hiệu quả, cần tuân thủ một số nguyên tắc:

* Hiểu rõ nguồn và mục tiêu của dữ liệu.
* Kiểm tra dữ liệu tổng quan trước khi xử lý (head(), info(), describe()).
* Kết hợp giữa số liệu thống kê và trực quan hóa.
* Ghi chép lại toàn bộ quy trình để đảm bảo tái lập và minh bạch.
* Kiểm tra ảnh hưởng của các bước tiền xử lý lên kết quả phân tích.

Tóm lại, EDA không chỉ là bước làm quen với dữ liệu mà còn quyết định độ chính xác và chất lượng của toàn bộ quá trình phân tích. Làm EDA cẩn thận, khoa học và có hệ thống sẽ giúp tiết kiệm thời gian, phát hiện vấn đề sớm và đảm bảo kết quả cuối cùng đáng tin cậy hơn.