ỦY BAN NHÂN DÂN THÀNH PHỐ HỒ CHÍ MINH

**TRƯỜNG ĐẠI HỌC SÀI GÒN**

****

**BÁO CÁO**

Học phần: Phân tích dữ liệu

**ĐỀ TÀI:  
Phân tích khám phá về bệnh đái tháo đường**

Giảng viên hướng dẫn: ThS. Đỗ Như Tài

Thành viên nhóm:

Nguyễn Vũ Hào – MSSV: 3122410098

Trần Duy Khương – MSSV: 3122410192

Võ Thị Thương – MSSV: 3122410408

Tp.Hồ Chí Minh, 2025

|  |  |  |
| --- | --- | --- |
| **Họ và Tên** | **Mã số sinh viên** | **Phân công công việc** |
| Nguyễn Vũ Hào | 3122410098 | * Xử lý dữ liệu bị thiếu * Xóa dữ liệu trùng lặp * Chuẩn hóa dữ liệu * Kiểm tra dữ liệu sau khi xử lý * Soạn slide trình diễn |
| Trần Duy Khương | 3122410192 | * Phân tích và mô tả vấn đề * Đưa ra hướng giải quyết * Đẩy dữ liệu vào và kiểm tra dữ liệu * Viết báo cáo |
| Võ Thị Thương | 3122410408 | * Trực quan hóa dữ liệu * Biểu diễn mối liên hệ giữa các thuộc tính * Mô phỏng bằng biểu đồ, kết luận về đầu ra (tức là có bị đái tháo đường hay không) |

**Mục lục**

[**CHƯƠNG I: GIỚI THIỆU VỀ PHÂN TÍCH VÀ KHÁM PHÁ DỮ LIỆU** 4](#_Toc209723054)

[**1.** **Tổng quan về Phân tích Khám phá Dữ liệu (EDA)** 4](#_Toc209723055)

[**2.** **Mục tiêu của Phân tích Khám phá Dữ liệu** 5](#_Toc209723056)

[**3.** **Công cụ và Ngôn ngữ Python trong EDA** 5](#_Toc209723057)

[**4.** **Quy trình EDA trong Nghiên cứu Đái tháo đường** 6](#_Toc209723058)

[**CHƯƠNG II: ĐỊNH NGHĨA, CHUẨN ĐOÁN VÀ PHÂN LOẠI BỆNH ĐÁI THÁO ĐƯỜNG VÀ CÁC BIẾN CHỨNG** 7](#_Toc209723059)

[**1.** **Giới thiệu** 7](#_Toc209723060)

[**2.** **Định nghĩa và tiêu chuẩn chuẩn đoán bệnh đái tháo đường và các dạng rối loạn dung nạp glucose** 7](#_Toc209723061)

[**2.1.** **Định nghĩa** 7](#_Toc209723062)

[**2.2.** **Chuẩn đoán và tiêu chuẩn chuẩn đoán** 7](#_Toc209723063)

[**2.3.** **Tiêu chuẩn chuẩn đoán** 8](#_Toc209723064)

[**3.** **Phân loại** 9](#_Toc209723065)

[**3.1.** **Các phân loại trước đó** 9](#_Toc209723066)

[**3.2.** **Phân loại sửa đổi** 9](#_Toc209723067)

[**4.** **Các biến chứng** 10](#_Toc209723068)

[**4.1.** **Đái tháo đường** 11](#_Toc209723069)

[**4.2.** **Rối loạn điều hòa glucose – Rối loạn dung nạp glucose (IGT) và rối loạn đường huyết lúc đói (IFG)** 11](#_Toc209723070)

[**CHƯƠNG II: SỬ DỤNG THUẬT TOÁN HỌC ADAP ĐỂ DỰ ĐOÁN SỰ KHỞI PHÁT CỦA BỆNH TIỂU ĐƯỜNG** 12](#_Toc209723071)

[**1.** **Giới thiệu về thuật toán** 12](#_Toc209723072)

[**2.** **Áp dụng vào mô hình để phân tích** 13](#_Toc209723073)

[**2.1.** **Đối tượng và phương pháp** 13](#_Toc209723074)

[**2.2.** **Lựa chọn các thuộc tính** 13](#_Toc209723075)

[**2.3.** **Lựa chọn các trường hợp** 15](#_Toc209723076)

[**3.** **Mô tả thuật toán** 15](#_Toc209723077)

[**CHƯƠNG III: TỔNG QUAN, PHÂN LOẠI VÀ CHUẨN ĐOÁN CÁC LỚP GLUCOSE VÀ ĐIỀU TRỊ** 17](#_Toc209723078)

[**1.** **Tổng quan** 17](#_Toc209723079)

[**2.** **Phân loại và chuẩn đoán** 17](#_Toc209723080)

[**2.1.** **Phân loại chi tiết** 17](#_Toc209723081)

[**2.2.** **Các danh mục khác về dung nạp glucose** 18](#_Toc209723082)

[**2.3.** **Tiêu chí chuẩn đoán** 18](#_Toc209723083)

[**CHƯƠNG IV: KHÁM PHÁ VÀ PHÂN TÍCH KHÁM PHÁ VỀ BỆNH ĐÁI THÁO ĐƯỜNG** 19](#_Toc209723084)

[**1.** **Định nghĩa vấn đề** 19](#_Toc209723085)

[**2.** **Chuẩn bị vấn đề** 21](#_Toc209723086)

[**2.1.** **Khai báo thư viện** 21](#_Toc209723087)

[**2.2.** **Nạp dữ liệu** 22](#_Toc209723088)

[**2.3.** **Kiểm tra một số thông tin cơ bản** 22](#_Toc209723089)

[**3.** **Chuẩn bị dữ liệu** 26](#_Toc209723090)

[**3.1.** **Mô tả vấn đề** 26](#_Toc209723091)

[**3.2.** **Làm sạch dữ liệu** 26](#_Toc209723092)

[**4.** **Phân tích dữ liệu** 31](#_Toc209723093)

[**4.1.** **Mô tả vấn đề** 31](#_Toc209723094)

[**4.2.** **Thống kê mô tả** 31](#_Toc209723095)

[**4.3.** **Trực quan hóa dữ liệu** 35](#_Toc209723096)

[**4.4.** **Trực quan hóa mối liên hệ giữa các thuộc tính** 39](#_Toc209723097)

[**4.5.** **Thuật toán học ADAP để dự đoán sự khởi phát của bệnh đái tháo đường** 46](#_Toc209723098)

[**4.6.** **Mô phỏng và trực quan hóa** 48](#_Toc209723099)

# **CHƯƠNG I: GIỚI THIỆU VỀ PHÂN TÍCH VÀ KHÁM PHÁ DỮ LIỆU**

1. **Tổng quan về Phân tích Khám phá Dữ liệu (EDA)**

Phân tích khám phá dữ liệu (Exploratory Data Analysis - EDA) là quá trình quan trọng đầu tiên trong bất kỳ dự án khoa học dữ liệu nào, đặc biệt là trong lĩnh vực y tế như nghiên cứu về bệnh đái tháo đường. EDA là phương pháp tiếp cận hệ thống nhằm phân tích, tóm tắt và trực quan hóa dữ liệu để hiểu rõ hơn về đặc điểm, mô hình và xu hướng tiềm ẩn trong tập dữ liệu.

Trong bối cảnh nghiên cứu về bệnh đái tháo đường, EDA giúp chúng ta khám phá các mối quan hệ phức tạp giữa các yếu tố nguy cơ, triệu chứng lâm sàng, và kết quả chẩn đoán. Quá trình này không chỉ giúp làm sạch và chuẩn bị dữ liệu mà còn cung cấp những hiểu biết sâu sắc về bản chất của bệnh lý, từ đó định hướng cho các bước phân tích tiếp theo.

1. **Mục tiêu của Phân tích Khám phá Dữ liệu**

Việc thực hiện EDA trong nghiên cứu bệnh đái tháo đường nhằm đạt được các mục tiêu chính sau:

Hiểu cấu trúc dữ liệu: Xác định kích thước tập dữ liệu, số lượng biến, kiểu dữ liệu của từng biến, và mức độ hoàn chỉnh của dữ liệu. Điều này đặc biệt quan trọng khi làm việc với dữ liệu y tế thường có nhiều giá trị bị thiếu do tính chất nhạy cảm và phức tạp của thông tin sức khỏe.

Phát hiện patterns và trends: Khám phá các mô hình xuất hiện trong dữ liệu như sự phân bố của các yếu tố nguy cơ theo độ tuổi, giới tính, hoặc nhóm dân số. Ví dụ, phân tích sự khác biệt về tỷ lệ mắc bệnh đái tháo đường giữa nam và nữ, hoặc xu hướng gia tăng theo độ tuổi.

Xác định outliers và anomalies: Phát hiện các giá trị bất thường có thể là lỗi nhập liệu hoặc những trường hợp đặc biệt cần được xem xét kỹ lưỡng. Trong dữ liệu y tế, outliers có thể mang ý nghĩa lâm sàng quan trọng hoặc phản ánh sai sót trong quá trình thu thập dữ liệu.

Khám phá correlation: Phân tích mối tương quan giữa các biến để hiểu rõ hơn về mối quan hệ giữa các yếu tố nguy cơ và khả năng mắc bệnh đái tháo đường. Điều này giúp xác định những yếu tố có ảnh hưởng mạnh nhất đến việc phát triển bệnh.

1. **Công cụ và Ngôn ngữ Python trong EDA**

**Python** đã trở thành ngôn ngữ lập trình hàng đầu cho phân tích dữ liệu và khoa học dữ liệu nhờ vào hệ sinh thái thư viện phong phú và cú pháp đơn giản, dễ hiểu. Trong nghiên cứu về bệnh đái tháo đường, Python cung cấp những công cụ mạnh mẽ để thực hiện EDA hiệu quả.

**Pandas** là thư viện cốt lõi cho việc xử lý và phân tích dữ liệu có cấu trúc. Với pandas, chúng ta có thể dễ dàng đọc dữ liệu từ các định dạng khác nhau (CSV, Excel, JSON), thực hiện các phép biến đổi dữ liệu, tính toán thống kê mô tả, và xử lý giá trị thiếu. Đối với dữ liệu y tế về đái tháo đường, pandas giúp chúng ta tổ chức và thao tác với các biến như tuổi, BMI, glucose, insulin một cách hiệu quả.

**NumPy** cung cấp các hàm toán học và thống kê cơ bản, hỗ trợ tính toán số học trên arrays đa chiều. Thư viện này đặc biệt hữu ích khi cần thực hiện các phép tính ma trận hoặc các thuật toán số học phức tạp trong quá trình phân tích.

**Matplotlib** và **Seaborn** là hai thư viện trực quan hóa dữ liệu quan trọng nhất. Matplotlib cung cấp khả năng tạo ra các biểu đồ cơ bản và tùy chỉnh chi tiết, trong khi Seaborn mở rộng khả năng này với các biểu đồ thống kê chuyên sâu và giao diện thân thiện hơn. Trong nghiên cứu đái tháo đường, các biểu đồ như histogram để hiển thị phân phối glucose, scatter plot để xem mối quan hệ giữa BMI và insulin, hoặc heatmap để visualize correlation matrix đều rất hữu ích.

**Scipy** bổ sung các hàm thống kê nâng cao, cho phép thực hiện các test thống kê để kiểm định giả thuyết, phân tích tương quan, và các phương pháp thống kê khác cần thiết trong nghiên cứu y tế.

1. **Quy trình EDA trong Nghiên cứu Đái tháo đường**

Quá trình EDA cho dữ liệu bệnh đái tháo đường thường tuân theo một quy trình có hệ thống bao gồm các bước chính:

**Bước 1** - Tải và khám phá dữ liệu sơ bộ: Import các thư viện cần thiết và đọc dữ liệu vào DataFrame. Sử dụng các hàm như info() , describe() , shape , head() để có cái nhìn tổng quan đầu tiên về cấu trúc và nội dung dữ liệu.

**Bước 2** - Đánh giá chất lượng dữ liệu: Kiểm tra giá trị thiếu ( isnull() ), phát hiện duplicates, và đánh giá tính nhất quán của dữ liệu. Trong dữ liệu y tế, việc xử lý giá trị thiếu cần được thực hiện cẩn trọng vì có thể ảnh hưởng đến tính chính xác của phân tích.

**Bước 3** - Phân tích thống kê mô tả: Tính toán các thống kê cơ bản như mean, median, standard deviation cho các biến liên tục, và frequency distribution cho các biến categorical. Điều này giúp hiểu rõ về range và distribution của các yếu tố như tuổi, BMI, glucose levels.

**Bước 4** - Trực quan hóa phân phối dữ liệu: Sử dụng histograms, box plots, và density plots để hiểu về distribution của từng biến. Việc này giúp phát hiện skewness, outliers, và đặc điểm phân phối của dữ liệu.

**Bước 5** - Phân tích mối quan hệ giữa các biến: Tạo correlation matrix, scatter plots, và cross- tabulations để khám phá mối quan hệ giữa các predictors và target variable (có mắc đái tháo đường hay không). Quy trình EDA này tạo nền tảng vững chắc cho các bước phân tích tiếp theo, bao gồm feature engineering, model selection, và predictive modeling, đồng thời đảm bảo rằng những insight được rút ra từ dữ liệu là đáng tin cậy và có ý nghĩa thực tiễn trong việc hiểu và dự đoán bệnh đái tháo đường.

# **CHƯƠNG II: ĐỊNH NGHĨA, CHUẨN ĐOÁN VÀ PHÂN LOẠI BỆNH ĐÁI THÁO ĐƯỜNG VÀ CÁC BIẾN CHỨNG**

1. **Giới thiệu**

Thuật ngữ “đái tháo đường (diabetes mellitus)” mô tả một rối loạn chuyển hóa có nhiều nguyên nhân, đặc trưng bởi tình trạng tăng đường huyết mạn tính (chronic hyperglycaemia), kèm theo rối loạn chuyển hóa carbohydrate, lipid và protein. Nguyên nhân chủ yếu là do thiếu hụt tiết insulin, giảm tác dụng insulin, hoặc cả hai.

Các hậu quả lâu dài của bệnh bao gồm:  
- Tổn thương, rối loạn chức năng và suy yếu nhiều cơ quan.  
- Các biến chứng đặc hiệu: bệnh võng mạc (có thể gây mù), bệnh thận (dẫn tới suy thận), bệnh thần kinh (tăng nguy cơ loét bàn chân, cắt cụt chi, biến dạng khớp, rối loạn thần kinh tự động, rối loạn tình dục).  
- Nguy cơ tăng cao mắc bệnh tim mạch, bệnh mạch ngoại vi và bệnh mạch máu não.

Bệnh có thể biểu hiện triệu chứng điển hình như: khát nhiều, tiểu nhiều, sụt cân, nhìn mờ. Trong thể nặng có thể xuất hiện nhiễm toan ceton hoặc tình trạng tăng thẩm thấu không ceton, dẫn tới hôn mê, thậm chí tử vong nếu không điều trị. Tuy nhiên, nhiều trường hợp bệnh diễn tiến âm thầm, triệu chứng không rõ ràng, khiến tình trạng tăng đường huyết kéo dài trước khi được chẩn đoán.  
Về cơ chế bệnh sinh, có nhiều tiến trình khác nhau:  
- Quá trình phá hủy tế bào beta tụy → thiếu insulin tuyệt đối.  
- Quá trình gây đề kháng insulin → giảm tác động của insulin ở mô đích.

1. **Định nghĩa và tiêu chuẩn chuẩn đoán bệnh đái tháo đường và các dạng rối loạn dung nạp glucose**
   1. **Định nghĩa**

Đái tháo đường là một rối loạn chuyển hóa do nhiều nguyên nhân, đặc trưng bởi tình trạng tăng đường huyết mạn tính, kèm theo rối loạn chuyển hóa carbohydrate, mỡ và protein. Hậu quả là tổn thương lâu dài và suy chức năng của nhiều cơ quan.

* 1. **Chuẩn đoán và tiêu chuẩn chuẩn đoán**
     1. **Chuẩn đoán**

- Khi chẩn đoán đái tháo đường, bác sĩ cần chắc chắn và xác nhận vì chẩn đoán này ảnh hưởng suốt đời bệnh nhân.  
- Trường hợp bệnh nhân có triệu chứng rõ rệt kèm tăng đường huyết nặng, chẩn đoán đơn giản hơn so với người không triệu chứng.  
- Tăng đường huyết cấp tính do nhiễm trùng, chấn thương, stress tuần hoàn có thể tạm thời, không được coi là chẩn đoán đái tháo đường.  
- Với người không có triệu chứng, không bao giờ được chẩn đoán chỉ dựa trên một xét nghiệm đơn lẻ. Cần có ít nhất một xét nghiệm khác.  
- Nếu chưa rõ ràng, bệnh nhân cần được theo dõi và xét nghiệm định kỳ.  
- Các yếu tố cần cân nhắc: chủng tộc, tiền sử gia đình, tuổi, béo phì, bệnh lý kèm theo.  
- HbA1c có thể phản ánh đường huyết trung bình nhiều tuần, nhưng chưa đủ tiêu chuẩn hóa để thay thế OGTT trong chẩn đoán.

* + 1. **Đái tháo đường ở trẻ em**

- Thường khởi phát với triệu chứng nặng, đường huyết rất cao, nhiều glucose và ceton trong nước tiểu.  
- Thường được chẩn đoán ngay bằng xét nghiệm đường huyết, sau đó cần tiêm insulin khẩn cấp.  
- OGTT không cần thiết trong những trường hợp này.  
- Một số trẻ biểu hiện nhẹ hơn có thể cần OGTT hoặc xét nghiệm đường huyết lúc đói.

* 1. **Tiêu chuẩn chuẩn đoán**

- Chẩn đoán lâm sàng thường dựa vào triệu chứng: khát nhiều, tiểu nhiều, nhiễm trùng tái diễn, sụt cân không rõ nguyên nhân, hôn mê.  
- Đường huyết cao trong một lần xét nghiệm vượt ngưỡng chẩn đoán đã đủ để xác định bệnh.  
- Nếu kết quả nằm trong vùng nghi ngờ, có thể cần làm OGTT.  
- Ở người lớn và trẻ em, ngưỡng chẩn đoán là như nhau.

**2.3.1. Thay đổi đường huyết lúc đói**

- Trước đây: ≥7.8 mmol/L (140 mg/dl).  
 - Mới: ≥7.0 mmol/L (126 mg/dl) với huyết tương, hoặc ≥6.1 mmol/L

(110mg/dl) với máu toàn phần.  
 - Lý do: nhiều nghiên cứu cho thấy nguy cơ biến chứng vi mạch tăng rõ khi đường huyết lúc đói ≥7.0 mmol/L.

**2.3.2. Trong nghiên cứu dịch tễ học**

- Có thể phân loại dựa trên đường huyết lúc đói hoặc sau 2h làm OGTT.  
- Nhiều nghiên cứu chỉ dùng giá trị sau 2h OGTT.  
- Tuy nhiên, hai giá trị này không hoàn toàn trùng nhau.

**2.3.3. Trong chuẩn đoán cá thể**

- Không được chẩn đoán dựa trên một lần đo.  
- Cần xác nhận bằng triệu chứng hoặc một xét nghiệm khác.  
- Tốt nhất nên xét nghiệm cả lúc đói và sau 2h OGTT.

1. **Phân loại**
   1. **Các phân loại trước đó**

Phân loại đầu tiên được chấp nhận rộng rãi về bệnh đái tháo đường được WHO công bố vào năm 1980 (1) và, dưới dạng sửa đổi, vào năm 1985 (3). Phân loại bệnh đái tháo đường năm 1980 và 1985 và các nhóm liên quan đến rối loạn dung nạp glucose bao gồm các nhóm lâm sàng và hai nhóm nguy cơ thống kê. Ủy ban Chuyên gia năm 1980 đã đề xuất hai nhóm chính của bệnh đái tháo đường và đặt tên cho chúng là IDDM hay Loại 1, và NIDDM hay Loại 2. Trong Báo cáo Nhóm Nghiên cứu năm 1985, các thuật ngữ Loại 1 và Loại 2 đã bị loại bỏ, nhưng các nhóm IDDM và NIDDM được giữ lại, và một nhóm Bệnh đái tháo đường liên quan đến suy dinh dưỡng (MRDM) đã được giới thiệu. Trong cả hai báo cáo năm 1980 và 1985, các nhóm bệnh đái tháo đường khác bao gồm các loại khác và rối loạn dung nạp glucose (IGT) cũng như đái tháo đường thai kỳ (GDM). Những điều này đã được phản ánh trong Danh pháp Bệnh tật Quốc tế (IND) sau đó vào năm 1991, và lần sửa đổi thứ mười của Phân loại Bệnh tật Quốc tế (ICD-10) vào năm 1992. Phân loại năm 1985 đã được chấp nhận rộng rãi và được sử dụng trên toàn thế giới. Nó thể hiện sự dung hòa giữa phân loại lâm sàng và phân loại nguyên nhân, cho phép phân loại từng đối tượng và bệnh nhân một cách hữu ích về mặt lâm sàng ngay cả khi nguyên nhân hoặc nguyên nhân cụ thể chưa được biết. Phân loại được khuyến nghị bao gồm cả việc phân giai đoạn bệnh đái tháo đường dựa trên các tiêu chí mô tả lâm sàng và phân loại nguyên nhân bổ sung.

* 1. **Phân loại sửa đổi**

Phân loại này bao gồm cả các giai đoạn lâm sàng và các loại nguyên nhân của bệnh đái tháo đường và các loại tăng đường huyết khác, theo đề xuất của Kuzuya và Matsuda (15).

Phân loại lâm sàng phản ánh rằng bệnh đái tháo đường, bất kể nguyên nhân, đều tiến triển qua nhiều giai đoạn lâm sàng trong diễn biến tự nhiên của bệnh. Hơn nữa, mỗi cá thể có thể chuyển từ giai đoạn này sang giai đoạn khác theo bất kỳ hướng nào. Những người mắc hoặc đang mắc bệnh đái tháo đường có thể được phân loại theo giai đoạn dựa trên các đặc điểm lâm sàng, ngay cả khi không có thông tin liên quan đến nguyên nhân cơ bản.

Phân loại theo loại nguyên nhân là kết quả của việc hiểu biết tốt hơn về nguyên nhân gây bệnh đái tháo đường.

* + - 1. **Các ứng dụng của phân loại mới**

Phân loại mới bao gồm các giai đoạn phản ánh các mức độ tăng đường huyết khác nhau ở từng cá thể mắc bất kỳ quá trình bệnh lý nào có thể dẫn đến bệnh đái tháo đường.

Tất cả các đối tượng mắc bệnh đái tháo đường đều có thể được phân loại theo giai đoạn lâm sàng, và điều này có thể đạt được trong mọi trường hợp. Giai đoạn đường huyết có thể thay đổi theo thời gian tùy thuộc vào mức độ của các quá trình bệnh lý tiềm ẩn (Hình 2). Quá trình bệnh lý có thể hiện diện nhưng có thể chưa tiến triển đủ xa để gây tăng đường huyết.

Phân loại nguyên nhân phản ánh thực tế là khiếm khuyết hoặc Quá trình có thể dẫn đến bệnh tiểu đường có thể được xác định ở bất kỳ giai đoạn nào trong quá trình phát triển bệnh tiểu đường – ngay cả ở giai đoạn đường huyết bình thường. Do đó, sự hiện diện của kháng thể tế bào đảo tụy ở một cá nhân có đường huyết bình thường khiến người đó có khả năng mắc bệnh tự miễn dịch loại 1. Đáng tiếc là hiện nay có rất ít chỉ số nhạy cảm hoặc đặc hiệu cao của bệnh tiểu đường loại 2, mặc dù những chỉ số này có thể được phát hiện khi nguyên nhân được xác định rõ ràng hơn. Các quá trình bệnh lý tương tự có thể gây ra rối loạn đường huyết lúc đói và/hoặc rối loạn dung nạp glucose mà không đáp ứng các tiêu chuẩn chẩn đoán bệnh tiểu đường. Ở một số cá nhân mắc bệnh tiểu đường, việc kiểm soát đường huyết đầy đủ có thể đạt được bằng việc giảm cân, tập thể dục và/hoặc thuốc uống. Do đó, những cá nhân này không cần insulin và thậm chí có thể trở lại IGT hoặc đường huyết bình thường. Những cá nhân khác cần insulin để kiểm soát đường huyết đầy đủ nhưng có thể sống sót mà không cần insulin. Theo định nghĩa, những cá nhân này có một lượng insulin tiết ra còn lại. Những người bị phá hủy tế bào beta rộng rãi, và do đó không còn tiết insulin, cần insulin để sống sót. Mức độ nghiêm trọng của bất thường chuyển hóa có thể thoái triển (ví dụ khi giảm cân), tiến triển (ví dụ khi tăng cân), hoặc giữ nguyên.

1. **Các biến chứng**
   1. **Đái tháo đường**

Đái tháo đường, bất kể nguyên nhân tiềm ẩn, được chia thành: Cần insulin để sống sót (tương ứng với phân loại lâm sàng trước đây là "Đái tháo đường phụ thuộc insulin - IDDM"), ví dụ: thiếu hụt peptide C; Cần insulin để kiểm soát, tức là kiểm soát chuyển hóa, thay vì để sống sót, ví dụ: tiết một lượng insulin nội sinh nhưng không đủ để đạt được đường huyết bình thường nếu không bổ sung insulin ngoại sinh; và Không cần insulin, tức là những người có thể được kiểm soát thỏa đáng bằng các phương pháp không dùng thuốc hoặc thuốc khác ngoài insulin. Hai phân nhóm sau cùng tạo thành phân nhóm trước đây của NIDDM.

* 1. **Rối loạn điều hòa glucose – Rối loạn dung nạp glucose (IGT) và rối loạn đường huyết lúc đói (IFG)**

Rối loạn điều hòa glucose (IGT và IFG) đề cập đến một trạng thái chuyển hóa trung gian giữa cân bằng glucose bình thường và bệnh tiểu đường. Tuy nhiên, cần phải khẳng định rõ ràng rằng IFG và IGT không thể thay thế cho nhau và đại diện cho những bất thường khác nhau về điều hòa glucose, một ở trạng thái đói và một sau bữa ăn.

IGT, thay vì là một nhóm như trong phân loại trước, được phân loại là một giai đoạn trong lịch sử tự nhiên của rối loạn chuyển hóa carbohydrate. Một giai đoạn của IFG cũng được công nhận vì những đối tượng này, giống như những người mắc IGT, có nguy cơ tiến triển thành bệnh tiểu đường và bệnh mạch máu lớn cao hơn, mặc dù dữ liệu triển vọng còn ít và dữ liệu ban đầu cho thấy nguy cơ tiến triển thấp hơn IGT (18), mặc dù hồ sơ yếu tố nguy cơ tim mạch tương tự đã được cho thấy ở những đối tượng mắc IFG và IGT (19). IFG đề cập đến nồng độ glucose lúc đói thấp hơn nồng độ cần thiết để chẩn đoán bệnh tiểu đường nhưng cao hơn phạm vi tham chiếu "bình thường". Giá trị của IFG là nồng độ glucose huyết tương lúc đói là 6,1 mmol l–1 (110 mg dl–1) hoặc cao hơn (máu toàn phần 5,6 mmol l–1; 100 mg dl–1), nhưng thấp hơn 7,0 mmol l–1 (126 mg dl–1) (máu toàn phần 6,1 mmol l–1; 110 mg dl–1). Nếu thực hiện OGTT, một số người mắc IFG sẽ bị IGT hoặc tiểu đường, nhưng điều này không thể xác định được nếu không có OGTT. Nếu nguồn lực cho phép, khuyến cáo tất cả những người mắc IFG nên thực hiện OGTT để loại trừ chẩn đoán tiểu đường. Những người đáp ứng các tiêu chí của IGT hoặc IFG có thể bình thường đường huyết trong cuộc sống hàng ngày, thể hiện qua nồng độ hemoglobin glycated bình thường hoặc gần bình thường. IGT và IFG không phải là thực thể lâm sàng riêng biệt, mà là các nhóm nguy cơ cho bệnh tiểu đường và/hoặc bệnh tim mạch trong tương lai (20,21).

Chúng có thể xảy ra như một giai đoạn trung gian trong bất kỳ quá trình bệnh lý nào được liệt kê trong Bảng 2. IGT thường liên quan đến Hội chứng Chuyển hóa (Hội chứng Kháng Insulin) (22). Do đó, IGT có thể không liên quan trực tiếp đến cơ chế bệnh sinh của bệnh tim mạch, mà có thể đóng vai trò là một chỉ số hoặc dấu hiệu của nguy cơ gia tăng do mối tương quan của nó với các yếu tố khác của Hội chứng Chuyển hóa, vốn là các yếu tố nguy cơ tim mạch. Rõ ràng, những người mắc IGT chỉ biểu hiện tình trạng không dung nạp glucose khi được thử nghiệm với glucose đường uống.

# **CHƯƠNG II: SỬ DỤNG THUẬT TOÁN HỌC ADAP ĐỂ DỰ ĐOÁN SỰ KHỞI PHÁT CỦA BỆNH TIỂU ĐƯỜNG**

1. **Giới thiệu về thuật toán**

Khả năng dự báo và phân biệt đóng vai trò trung tâm trong nhiều tình huống y tế. Các kỹ thuật thống kê tiêu chuẩn như phân tích phân biệt, phân tích hồi quy và phân tích nhân tố đã được sử dụng để cung cấp khả năng này. Tuy nhiên, ngay cả khi tồn tại các mối quan hệ hàm ẩn có thể cung cấp khả năng dự báo, các kỹ thuật thống kê tiêu chuẩn vẫn có thể không thành công. Các phương pháp thống kê tiêu chuẩn có thể mang lại kết quả không như mong đợi khi:

1. Kích thước mẫu nhỏ.
2. Dạng của mối quan hệ hàm cơ bản chưa được biết rõ.
3. Các mối quan hệ hàm cơ bản bao gồm các tương tác và tương quan phức tạp giữa một số biến.

Những tình trạng này không phải là hiếm gặp trong các vấn đề y tế. Trong những tình huống như vậy, một số phương pháp tiếp cận mạng nơ-ron nhân tạo mang lại nhiều hứa hẹn.

Mô hình mạng nơ-ron là một lớp thuật toán học tập được mô tả lần đầu vào cuối những năm 1950.[2] Những thuật toán ban đầu này đã được mở rộng gần đây, và sự quan tâm đến chúng đã tăng lên đáng kể.[3] Các thuật toán này có khả năng sử dụng một tập dữ liệu huấn luyện để khám phá các mẫu trong dữ liệu. Một thuật toán như vậy, được gọi là ADAP, đã được phát triển bởi hai tác giả [Smith, Dickson] vào 1961.[4,5] Chúng tôi đã chọn kiểm tra thuật toán ADAP và thử nghiệm việc sử dụng nó trong việc dự báo sự khởi phát của bệnh tiểu đường không phụ thuộc insulin trong khoảng thời gian năm năm. Dữ liệu được sử dụng trong nghiên cứu này được lấy từ quần thể người da đỏ Pima gần Phoenix, Arizona. Sau khi thuật toán được huấn luyện bằng 576 trường hợp, ADAP đã được sử dụng để dự báo liệu 192 trường hợp thử nghiệm khác có phát triển bệnh tiểu đường trong vòng năm năm hay không. Việc buộc ADAP đưa ra kết luận về tất cả các trường hợp thử nghiệm đã tạo ra độ nhạy và độ đặc hiệu là 76%. Đường cong đặc tính hoạt động của máy thu (ROC) đã được xác định. Những nỗ lực đang được tiến hành để kiểm tra cùng một tập dữ liệu bằng cách sử dụng mô hình hồi quy logistic và perceptron tuyến tính.

1. **Áp dụng vào mô hình để phân tích**
   1. **Đối tượng và phương pháp**

Dân số trong nghiên cứu này là dân số người da đỏ Pima gần Phoenix, Arizona. Dân số này đã được Viện Quốc gia về Bệnh tiểu đường, Tiêu hóa và Thận nghiên cứu liên tục từ năm 1965 do tỷ lệ mắc bệnh tiểu đường cao.[6,7,8] Mỗi cư dân cộng đồng trên 5 tuổi được yêu cầu trải qua một cuộc kiểm tra tiêu chuẩn hai năm một lần, bao gồm xét nghiệm dung nạp glucose đường uống. Bệnh tiểu đường được chẩn đoán theo Tiêu chuẩn của Tổ chức Y tế Thế giới[10]; nghĩa là, nếu nồng độ glucose huyết tương sau 2 giờ sau khi thử nghiệm ít nhất là 200 mg/dl (11,1 mmol/l) tại bất kỳ cuộc kiểm tra khảo sát nào hoặc nếu Bệnh viện Dịch vụ Y tế Người da đỏ phục vụ cộng đồng phát hiện nồng độ glucose ít nhất là 200 mg/dl trong quá trình chăm sóc y tế thường quy. Ngoài việc là một cơ sở dữ liệu quen thuộc với các nhà nghiên cứu, bộ dữ liệu này còn cung cấp một nguồn dữ liệu được xác thực tốt để khám phá dự đoán ngày khởi phát bệnh tiểu đường theo chiều dọc.

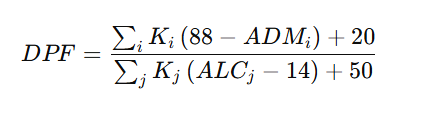
* 1. **Lựa chọn các thuộc tính**

Khi áp dụng ADAP vào vấn đề tiểu đường, tám biến số đã được chọn làm cơ sở dự báo sự khởi phát của bệnh tiểu đường trong vòng năm năm ở phụ nữ Ấn Độ Pima. Các biến số này được chọn vì chúng được phát hiện là những yếu tố nguy cơ đáng kể đối với bệnh tiểu đường ở người Pima hoặc các nhóm dân số khác.

Các biến đầu vào bao gồm:

1. Số lần mang thai
2. Nồng độ glucose huyết tương sau 2 giờ trong xét nghiệm dung nạp glucose đường uống (GTIT)
3. Huyết áp tâm trương (mmHg)
4. Độ dày nếp gấp da cơ tam đầu (mm)
5. Nồng độ insulin huyết thanh sau 2 giờ (Uh/ml)
6. Chỉ số khối cơ thể (Cân nặng (kg)/(Chiều cao (in))
7. Chức năng phả hệ bệnh tiểu đường
8. Tuổi (năm)

Hàm phả hệ bệnh tiểu đường:



được tính toán cho mỗi kỳ thi chỉ bằng cách sử dụng dữ liệu có sẵn vào ngày thi đó và trong đó:

i bao gồm tất cả những người thân đã mắc bệnh tiểu đường tính đến ngày khám của đối tượng;

j bao gồm tất cả những người thân KHÔNG mắc

bệnh tiểu đường tính đến ngày khám của đối tượng;

K là tỷ lệ phần trăm gen được chia sẻ bởi người thân (*relativex*) và

* bằng 0,500 khi *relativex* là cha mẹ hoặc anh chị em ruột,
* bằng 0,250 khi người thân là anh chị em cùng cha khác mẹ, ông bà, dì hoặc chú, và
* bằng 0,125 khi người thân là dì, chú, bác hoặc anh chị em họ;

ADMi  là độ tuổi tính theo năm của người thân khi bệnh tiểu đường được chẩn đoán;

ACLj  là độ tuổi tính theo năm của người thân tại lần khám không mắc bệnh tiểu đường gần nhất (trước ngày khám của đối tượng);

Hằng số

Các hằng số 88 và 14, ngoại trừ một số ít trường hợp, biểu thị độ tuổi tối đa và tối thiểu mà người thân của các đối tượng trong nghiên cứu này mắc bệnh tiểu đường.

Các hằng số 20 và 50 được chọn sao cho:

1. Một đối tượng không có người thân sẽ có giá trị DPF thấp hơn một chút so với mức trung bình
2. Giá trị DPF sẽ giảm tương đối chậm khi những người thân trẻ không mắc bệnh tiểu đường tham gia cơ sở dữ liệu
3. Giá trị DPF sẽ tăng tương đối nhanh khi những người thân đã biết mắc bệnh tiểu đường.

Lưu ý rằng giá trị của DPF tăng khi số lượng người thân mắc bệnh tiểu đường tăng, độ tuổi mắc bệnh tiểu đường của những người thân này giảm, và tỷ lệ phần trăm gen mà họ chia sẻ với đối tượng tăng. Cũng lưu ý rằng giá trị của DPF giảm khi số lượng người thân chưa bao giờ mắc bệnh tiểu đường tăng, độ tuổi của họ khi khám lần cuối tăng, và tỷ lệ phần trăm gen mà họ chia sẻ với đối tượng tăng.

* 1. **Lựa chọn các trường hợp**

Bệnh tiểu đường được định nghĩa là nồng độ glucose huyết tương lớn hơn 200 mg/dl hai giờ sau khi uống 75 gam dung dịch carbohydrate. Các trường hợp được chọn từ nhóm kiểm tra đáp ứng các tiêu chí sau:

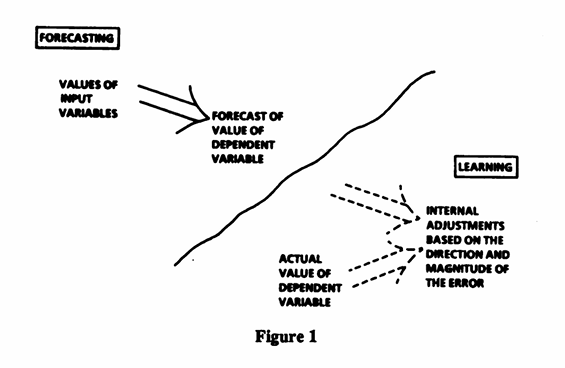
1. Đối tượng là nữ.
2. Đối tượng nghiên cứu được 221 tuổi tại thời điểm kiểm tra chỉ số. Kiểm tra chỉ số đề cập đến nghiên cứu được chọn để sử dụng trong mô hình này. Nó không nhất thiết phải tương ứng với lần kiểm tra đầu tiên theo thứ tự thời gian cho đối tượng này.
3. Mỗi đối tượng chỉ được chọn một xét nghiệm. Xét nghiệm đó phải cho thấy GTT không bị tiểu đường và đáp ứng một trong hai tiêu chí sau:
   1. Bệnh tiểu đường được chẩn đoán trong vòng năm năm sau khi kiểm tra, HOẶC
   2. Xét nghiệm GTT được thực hiện sau đó 5 năm hoặc hơn không phát hiện được bệnh tiểu đường.
4. Nếu bệnh tiểu đường xảy ra trong vòng một năm kể từ lần khám, lần khám đó sẽ bị loại khỏi nghiên cứu để loại bỏ khỏi mô hình dự báo những trường hợp có khả năng dễ dự báo hơn. Trong 75% số lần khám bị loại, bệnh tiểu đường được chẩn đoán trong vòng sáu tháng.

Sử dụng các tiêu chí này, 768 bài kiểm tra đã được chọn. Trong số đó, 576 bài được chọn ngẫu nhiên để sử dụng trong tập huấn luyện hoặc tập học tập và 192 trường hợp còn lại trở thành tập dự báo. Giả thuyết của chúng tôi là ADAP có thể học cách dự báo liệu một cá nhân nhất định có mắc bệnh tiểu đường trong vòng năm năm hay không, dựa trên giá trị của tám biến đầu vào.

1. **Mô tả thuật toán**

ADAP là một chương trình học tập thích ứng tạo ra và thực thi các tín hiệu tương tự kỹ thuật số của các thiết bị giống percepton. Nó có thể tạo ra nhiều loại tín hiệu tương tự như vậy. Tuy nhiên, nó được thiết kế đặc biệt để tạo ra các tín hiệu tương tự giống perceptron "phân vùng" chuyên dụng, phù hợp với từng vấn đề cụ thể.

Các mô hình tương tự ADAP học bằng cách thực hiện các điều chỉnh nội tại khi dự đoán của chúng trở nên không chính xác. Giả sử một hiện tượng nào đó được biểu thị dưới dạng giá trị hoặc trạng thái và bạn giả định rằng nó có thể được dự báo bằng một hàm số chưa biết của các biến khác. Chu kỳ điều chỉnh của ADAP được xác định bằng cách đọc giá trị của các biến trong một trường hợp (kiểm tra chỉ số trong nghiên cứu này). Sau đó, ADAP tạo ra một dự báo về giá trị hoặc trạng thái của hiện tượng, có thể là một hàm số của các biến đầu vào. Sau đó, nếu nó ở chế độ điều chỉnh, nó sẽ đọc trạng thái hoặc giá trị thực tế mà nó vừa cố gắng dự báo. Sau đó, nó so sánh dự báo với giá trị thực và thực hiện các điều chỉnh nội tại dựa trên hướng và độ lớn của lỗi. [HÌNH 1] Về mặt này, ADAP giống như các thiết bị học tập mạng nơ-ron khác.



Một điểm khác biệt lớn giữa ADAP và các mạng nơ-ron truyền thống hơn là cách các đơn vị cảm biến có thể được tổ chức và được "kết nối". Hình 2 là một ví dụ về một mạng tương tự ADAP. Các trường hợp đầu vào kích thích các cảm biến được tổ chức thành các phân vùng. Đến lượt mình, các cảm biến đó được kết nối với các đơn vị phân vùng được kết nối thông qua các trọng số có thể điều chỉnh với một đơn vị phản hồi. Một đơn vị cảm biến biểu diễn một giá trị rời rạc hoặc một dải giá trị có thể được giả định bởi một biến đầu vào nhất định. Tập hợp tất cả các cảm biến cho một đầu vào nhất định khả dụng bao gồm phân vùng cho biến đó. Mặc dù mỗi đơn vị liên kết được kết nối với mỗi cảm biến, nhưng ADAP thường hạn chế mô hình này bằng cách yêu cầu chỉ có một và duy nhất một kết nối khác không trong mỗi phân vùng. Tất cả các kết nối khác không thường được đặt thành trọng số 1, và các trọng số đó không được sử dụng trong quá trình học. Kết nối khác không này trong một phân vùng được chọn ngẫu nhiên. ADAP thường sử dụng một số lượng lớn các đơn vị liên kết như vậy. (Trong nghiên cứu này, chúng tôi đã sử dụng 100.000) Mỗi ​​đơn vị liên kết có một và chỉ một kết nối đầu ra với người phản hồi. Quy tắc kích hoạt cho một đơn vị liên kết với một tập hợp các giá trị biến đầu vào cụ thể là một hàm ngưỡng.

Tất cả các giá trị của một trường hợp khớp với một kết nối khác không được cho là kích thích các kết nối đó. Nếu tổng số kết nối được kích thích cho một đơn vị liên kết nhất định lớn hơn một giá trị định trước - ngưỡng - thì đơn vị liên kết đó được cho là đã được kích hoạt. Giá trị ngưỡng được cố định sao cho từ 2% đến 10% tổng số đơn vị liên kết sẽ được kích hoạt cho bất kỳ trường hợp đầu vào nào. Cuối cùng, các giá trị phản hồi được kết nối với mỗi đơn vị liên kết đã kích hoạt được cộng lại và tổng này tạo thành dự báo cho trường hợp đầu vào đã cho. Nếu ADAP đang ở chế độ học, giá trị đã biết được so sánh với dự báo vừa được thực hiện và chênh lệch được tính toán và phân bổ giữa các đơn vị đã kích hoạt cho trường hợp huấn luyện đó. Toàn bộ quá trình dự báo/học có thể được lặp lại cho các trường hợp bổ sung. Với ADAP, phản hồi đầu ra có thể được diễn giải là dự báo về một giá trị liên tục (ví dụ: 65,4) hoặc trạng thái (hiện diện hoặc không hiện diện).

Do thuật toán ADAP ra đời trước khi sự quan tâm đến các mô hình kết nối trỗi dậy, một số tính năng của nó khác với các mô hình đương đại. Tuy nhiên, sử dụng thuật ngữ của Rumelhart, Hinton và McClelland, nó giống nhất với một mô hình học tập liên tưởng tương tác sử dụng quy tắc học tập Hebbian.

# **CHƯƠNG III: TỔNG QUAN, PHÂN LOẠI VÀ CHUẨN ĐOÁN CÁC LỚP GLUCOSE VÀ ĐIỀU TRỊ**

1. **Tổng quan**
2. **Phân loại và chuẩn đoán**
   1. **Phân loại chi tiết**

Tài liệu phân loại bệnh tiểu đường thành ba nhóm chính, dựa trên các đặc điểm lâm sàng và sinh lý:

* Bệnh tiểu đường phụ thuộc insulin (IDDM, Type I): Loại này có đặc điểm là thiếu insulin trầm trọng, khởi phát đột ngột và thường dẫn đến nhiễm toan ceton. Bệnh nhân cần tiêm insulin hàng ngày để duy trì sự sống. Tài liệu nhấn mạnh rằng loại này có thể xảy ra ở mọi lứa tuổi, không chỉ ở người trẻ tuổi.
* Bệnh tiểu đường không phụ thuộc insulin (NIDDM, Type II): Loại này không phụ thuộc vào insulin để duy trì sự sống. NIDDM thường khởi phát ở người trưởng thành và được chia thành hai nhóm phụ: béo phì và không béo phì.
* Các loại tiểu đường khác: Đây là nhóm lớn bao gồm tiểu đường do các nguyên nhân cụ thể, như bệnh tụy, bệnh nội tiết, hoặc do hóa chất/thuốc (ví dụ: tiểu đường do sử dụng corticosteroid).
* Tiểu đường thai kỳ (Gestational Diabetes): Đây là một phân loại riêng dành cho tình trạng không dung nạp glucose phát triển hoặc được phát hiện lần đầu tiên trong thời kỳ mang thai.
  1. **Các danh mục khác về dung nạp glucose**
     + Tài liệu cũng giới thiệu các danh mục cho những người không bị tiểu đường nhưng có mức đường huyết bất thường:
     + Không dung nạp glucose (IGT): Thuật ngữ này được sử dụng cho những người có mức glucose huyết tương ở mức trung gian giữa bình thường và tiểu đường. Tài liệu đề xuất loại bỏ các thuật ngữ cũ như "tiểu đường hóa học" để tránh gây hoang mang không cần thiết cho bệnh nhân.
     + Tiền sử bất thường về dung nạp glucose (PrevAGT): Đây là những người trước đây từng có chẩn đoán tăng đường huyết nhưng hiện tại đã trở lại bình thường.
     + Tiềm năng bất thường về dung nạp glucose (PotAGT): Đây là những người chưa từng có biểu hiện bất thường nhưng có nguy cơ mắc bệnh cao, ví dụ: người có tiền sử gia đình, người có gen liên quan, hoặc phụ nữ từng bị tiểu đường thai kỳ.
  2. **Tiêu chí chuẩn đoán**

Tài liệu cung cấp các tiêu chí cụ thể để chẩn đoán tiểu đường dựa trên kết quả xét nghiệm glucose huyết tương:

* Đối với người lớn không mang thai: Chẩn đoán được xác nhận khi một trong các điều kiện sau xảy ra:
* Có các triệu chứng cổ điển của bệnh tiểu đường (như khát nước, tiểu nhiều) và mức glucose huyết tương ngẫu nhiên ≥ 200 mg/dl.
* Mức glucose huyết tương lúc đói ≥ 140 mg/dl trong hai lần xét nghiệm riêng biệt.
* Kết quả xét nghiệm dung nạp glucose đường uống (OGTT) cho thấy mức glucose huyết tương ở 2 giờ ≥ 200 mg/dl.
* Đối với trẻ em: Có thể chẩn đoán khi có các triệu chứng cổ điển và mức glucose huyết tương ngẫu nhiên > 200 mg/dl. Việc thực hiện OGTT không bắt buộc nếu các triệu chứng và mức đường huyết đã đủ rõ ràng.

# **CHƯƠNG IV: KHÁM PHÁ VÀ PHÂN TÍCH KHÁM PHÁ VỀ BỆNH ĐÁI THÁO ĐƯỜNG**

1. **Định nghĩa vấn đề**

Phân tích vấn đề:

* Người da đỏ Pima tại Arizona có tỷ lệ mắc tiểu đường type 2 rất cao.
* Viện Quốc gia về Bệnh tiểu đường, Tiêu hóa và Thận (NIDDK) đã nghiên cứu liên tục từ 1965.
* Bệnh tiểu đường được chẩn đoán theo WHO: glucose huyết tương ≥ 200 mg/dL sau 2 giờ OGTT hoặc được phát hiện trong chăm sóc y tế thường quy.
* Mục tiêu: xây dựng mô hình dự báo khả năng mắc bệnh tiểu đường trong vòng 5 năm, dựa trên các yếu tố nguy cơ.

Mô tả: Dataset Pima Indians Diabetes chứa thông tin của 768 bệnh nhân nữ, ít nhất 21 tuổi, thuộc cộng đồng Pima Indian gần Phoenix, Arizona, USA. Mục tiêu là dự đoán xem một bệnh nhân có bị đái tháo đường (Class = 1) hay không (Class = 0) dựa trên 8 thuộc tính y tế.

Dữ liệu vào:

* 1. Pregnancies: Số lần mang thai, phản ánh tiền sử sinh sản có thể liên quan đến nguy cơ đái tháo đường.
  2. Glucose: Nồng độ glucose huyết tương sau 2 giờ trong bài kiểm tra dung nạp glucose (mg/dl), chỉ số quan trọng để chẩn đoán đái tháo đường.
  3. BloodPressure: Huyết áp tâm thu (mm Hg), đo áp lực máu khi tim giãn ra, liên quan đến sức khỏe tim mạch.
  4. SkinThickness: Độ dày nếp gấp da tay (mm), đại diện cho lượng mỡ dưới da, một yếu tố liên quan đến béo phì.
  5. Insulin: Mức insulin huyết thanh sau 2 giờ (mu U/ml), phản ánh khả năng sản xuất insulin của cơ thể.
  6. BMI: Chỉ số khối cơ thể (kg/m²), đo lường mức độ béo phì, một yếu tố nguy cơ lớn của đái tháo đường loại 2.
  7. DiabetesPedigreeFunction: Hàm phả hệ đái tháo đường, đánh giá nguy cơ di truyền dựa trên lịch sử gia đình.
  8. Age: Tuổi (năm), ảnh hưởng đến nguy cơ mắc bệnh do sự lão hóa.
  9. Outcome: Biến lớp (0: Không đái tháo đường, 1: Có đái tháo đường), mục tiêu chính của phân tích.
     + Trong đó Hàm phả hệ đái tháo đường(DPF):
       - Được xây dựng để lượng hóa yếu tố di truyền dựa trên tiền sử bệnh tiểu đường của cha mẹ, ông bà, anh chị em, họ hàng gần.
       - DPF tăng khi số lượng thân nhân mắc bệnh tăng, tuổi mắc bệnh trẻ hơn, và mức độ quan hệ huyết thống gần.
       - DPF giảm khi thân nhân không mắc bệnh và có tuổi thọ cao.

Lựa chọn trường hợp:

* + - Tiêu chí:
    - Nữ, ≥ 21 tuổi, OGTT ban đầu không bị tiểu đường.
    - Sau 5 năm:
    - Hoặc phát hiện tiểu đường (positive case).
    - Hoặc kiểm tra lại ≥ 5 năm sau vẫn không mắc (negative case).
    - Loại bỏ trường hợp phát hiện tiểu đường trong vòng < 1 năm (do quá dễ dự báo).
    - Tổng số mẫu: 768
    - 576 mẫu cho tập huấn luyện
    - 192 mẫu cho tập dự báo (test set).

Hướng giải quyết:

* + - Sử dụng ADAP (Adaptive Learning Algorithm) – một dạng mô hình học máy dựa trên perceptron, để dự báo khả năng mắc bệnh trong vòng 5 năm.
    - Biến đầu vào gồm 8 yếu tố nguy cơ chính + nhãn phân loại (có/không mắc bệnh).

Kết quả (từ nghiên cứu trước đó [Smith et al., 1988]):

* + - ADAP đưa ra dự đoán dạng xác suất (0–1), sau đó quy đổi thành nhị phân.
    - Với cutoff 0.448, mô hình đạt:
    - Sensitivity (độ nhạy): ~76%
    - Specificity (độ đặc hiệu): ~76%
    - Đây là kết quả khá tốt trong bối cảnh dữ liệu y tế thời kỳ đó, và bộ dữ liệu trở thành benchmark nổi tiếng trong cộng đồng Machine Learning.

Kết quả:

* + - Biến mục tiêu (Outcome): Biến nhị phân (0 hoặc 1) biểu thị trạng thái đái tháo đường
    - 0: Không bị đái tháo đường (500 mẫu).
    - 1: Bị đái tháo đường (268 mẫu).

Thông tin bổ sung:

* + - Giá trị thiếu: Dữ liệu có chứa giá trị thiếu, thường được biểu thị bằng 0 trong các cột như Glucose, BloodPressure, SkinThickness, Insulin, và BMI, cần được xử lý trong quá trình phân tích.
    - Nghiên cứu trước: Một nghiên cứu trước đây của Smith et al. (1988) sử dụng thuật toán ADAP, đạt độ nhạy (sensitivity) và độ đặc hiệu (specificity) 76% trên 192 mẫu kiểm tra sau khi huấn luyện với 576 mẫu, với ngưỡng cắt 0.448.
    - Mục tiêu phân tích: Xây dựng một mô hình dự đoán hiệu quả, so sánh với kết quả ADAP, và khám phá các yếu tố quan trọng ảnh hưởng đến nguy cơ đái tháo đường.

1. **Chuẩn bị vấn đề**

Mô tả vấn đề

* + - Bệnh tiểu đường type 2 có tỷ lệ rất cao trong cộng đồng này.
    - Việc chẩn đoán chỉ dựa vào tiêu chuẩn WHO OGTT (≥ 200 mg/dL sau 2 giờ) mới cho kết luận rõ ràng, nhưng điều này chỉ xác định khi bệnh đã xuất hiện.
    - Mục tiêu:
    - Trong phần này, chúng ta sẽ thiết lập môi trường làm việc bằng cách khai báo các thư viện cần thiết, load dataset Pima Indians Diabetes, và giải thích ý nghĩa của từng cột dữ liệu để bạn dễ dàng hiểu rõ giá trị mà chúng mang lại. Đây là bước nền tảng quan trọng để tiến hành phân tích sâu hơn về nguy cơ đái tháo đường trong cộng đồng Pima Indian.
    - Nhu cầu đặt ra: dự đoán sớm nguy cơ mắc bệnh tiểu đường trong tương lai dựa trên các yếu tố nguy cơ (tuổi, BMI, insulin, huyết áp, di truyền…).
    - Dữ liệu:
    - Dataset được cung cấp dưới dạng file pima-indians-diabetes.csv, chứa 768 mẫu với 8 thuộc tính số và 1 biến lớp (Outcome). Dữ liệu có thể chứa giá trị thiếu, cần được xử lý phù hợp.
    - Thách thức:
    - Dữ liệu y tế thường có mối quan hệ phi tuyến phức tạp giữa các biến.
    - Có sự hiện diện của missing values và outliers (ví dụ insulin = 0).
    - Cần một mô hình dự đoán chính xác nhưng cũng có thể giải thích được trong ngữ cảnh y tế.
  1. **Khai báo thư viện**
* **pandas** là một thư viện mạnh mẽ trong Python, được sử dụng rộng rãi cho **phân tích và xử lý dữ liệu**.
* Cung cấp cấu trúc dữ liệu chính:
* **Series**: tương tự như một mảng 1 chiều có nhãn.
* **DataFrame**: bảng dữ liệu 2 chiều (giống Excel hoặc SQL table).
* Các tính năng chính:
* Đọc/ghi dữ liệu từ nhiều định dạng (CSV, Excel, SQL, JSON, ...).
* Xử lý dữ liệu: lọc, nhóm, gộp, sắp xếp, thống kê.
* Làm sạch dữ liệu: xử lý missing values, outliers.
* Hỗ trợ tích hợp với NumPy, Matplotlib, Seaborn để trực quan hóa dữ liệu.
* Khai báo thư viện pandas

*import pandas as pd*

* 1. **Nạp dữ liệu**
* Thực hiện việc **nạp dữ liệu gốc** từ file pima-indians-diabetes.csv vào Python bằng thư viện pandas.
* Do file CSV không chứa tiêu đề cột (header), ta cần **tự gán tên cột** dựa trên mô tả trong tài liệu nghiên cứu gốc.
* Kết quả: dữ liệu được lưu vào một **DataFrame** có cấu trúc rõ ràng, giúp dễ dàng phân tích, trực quan hóa và xây dựng mô hình dự báo.
* Code thực hiện

# Đọc dữ liệu từ file CSV  
data\_path = 'pima-indians-diabetes.csv' # Đường dẫn tới file dữ liệu  
  
# Gán tên cột dựa trên mô tả trong tài liệu  
column\_names = ['Pregnancies', 'Glucose', 'BloodPressure', 'SkinThickness', 'Insulin', 'BMI', 'DiabetesPedigreeFunction', 'Age', 'Outcome'] # Tên các cột  
  
# Đọc dữ liệu từ file CSV  
data = pd.read\_csv(data\_path, header=None, names=column\_names)

* 1. **Kiểm tra một số thông tin cơ bản**
* Sử dụng các hàm cơ bản của pd đễ hỗ trợ cho chúng ta có một cái nhìn tổng quát về số liệu của file

# Hiển thị 5 dòng đầu để kiểm tra  
print("5 dòng đầu của dữ liệu:")  
print(data.head())

* Ta nhận được kết quả như sau:

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | Pregnancies | Glucose | BloodPressure | SkinThickness | Insulin | BMI | DiabetesPedigreeFunction | Age | Outcome |
| 0 | 6 | 148 | 72 | 35 | 0 | 33.6 | 0.627 | 50 | 1 |
| 1 | 1 | 85 | 66 | 29 | 0 | 26.6 | 0.351 | 31 | 0 |
| 2 | 8 | 183 | 64 | 0 | 0 | 23.3 | 0.672 | 32 | 1 |
| 3 | 1 | 89 | 66 | 23 | 94 | 28.1 | 0.167 | 21 | 0 |
| 4 | 0 | 137 | 40 | 35 | 168 | 43.1 | 2.288 | 33 | 1 |

Kiểm tra các giá trị = 0 của từng cột

*(data==0).sum()*

Kết quả

|  |  |
| --- | --- |
| Pregnancies | 111 |
| Glucose | 5 |
| BloodPressure | 35 |
| SkinThickness | 227 |
| Insulin | 374 |
| BMI | 11 |
| DiabetesPedigreeFunction | 0 |
| Age | 0 |
| Outcome | 500 |

dtype: int64

**Nhận xét về dữ liệu có giá trị 0:**

* + Pregnancies (111 giá trị 0):
    - * 0 lần mang thai là hợp lý (có những phụ nữ chưa từng mang thai).
      * Không cần xử lý coi là missing.
  + Glucose (5 giá trị 0):
    - * Về mặt y khoa, glucose máu không thể = 0.
      * Đây là giá trị bất thường → coi là dữ liệu thiếu.
  + BloodPressure (35 giá trị 0):
    - * Huyết áp tâm trương không thể = 0.
      * Các giá trị này là missing values cần xử lý
  + SkinThickness (227 giá trị 0):
    - * 0 mm nếp gấp da là không hợp lý.
      * Đây là một biến có nhiều thiếu dữ liệu (≈ 30% dữ liệu).
  + Insulin (374 giá trị 0):
    - * 0 µU/mL insulin là không thực tế.
      * Biến này có nhiều dữ liệu thiếu nhất (≈ 49%).
  + BMI (11 giá trị 0):
    - * BMI = 0 là bất hợp lý.
      * Một số ít dữ liệu cần xử lý missing.
  + DiabetesPedigreeFunction (0 giá trị 0):
    - * Không có giá trị 0, dữ liệu đầy đủ.
  + Age (0 giá trị 0):
    - * Không có giá trị 0, dữ liệu đầy đủ.
  + Outcome (500 giá trị 0):
    - * Đây là biến nhãn (class):
        + 0 = không mắc tiểu đường (500 mẫu).
        + 1 = mắc tiểu đường (268 mẫu).

# Hiển thị thông tin cơ bản về dữ liệu  
*print(data.info())*

*<class 'pandas.core.frame.DataFrame'>  
RangeIndex: 768 entries, 0 to 767  
Data columns (total 9 columns):  
 # Column Non-Null Count Dtype   
--- ------ -------------- -----   
 0 Pregnancies 768 non-null int64   
 1 Glucose 768 non-null int64   
 2 BloodPressure 768 non-null int64   
 3 SkinThickness 768 non-null int64   
 4 Insulin 768 non-null int64   
 5 BMI 768 non-null float64  
 6 DiabetesPedigreeFunction 768 non-null float64  
 7 Age 768 non-null int64   
 8 Outcome 768 non-null int64   
dtypes: float64(2), int64(7)  
memory usage: 54.1 KB  
None*

Nhận xét sơ bộ

* + **Cấu trúc dữ liệu:**
    - * Tổng cộng **767 dòng × 9 cột**.
      * Tất cả các biến đều dạng số (int hoặc float).

Thống kê mô tả

# Thống kê mô tả  
print(data.describe())

Pregnancies Glucose BloodPressure SkinThickness Insulin \  
count 768.000000 768.000000 768.000000 768.000000 768.000000   
mean 3.845052 120.894531 69.105469 20.536458 79.799479   
std 3.369578 31.972618 19.355807 15.952218 115.244002   
min 0.000000 0.000000 0.000000 0.000000 0.000000   
25% 1.000000 99.000000 62.000000 0.000000 0.000000   
50% 3.000000 117.000000 72.000000 23.000000 30.500000   
75% 6.000000 140.250000 80.000000 32.000000 127.250000   
max 17.000000 199.000000 122.000000 99.000000 846.000000   
  
 BMI DiabetesPedigreeFunction Age Outcome   
count 768.000000 768.000000 768.000000 768.000000   
mean 31.992578 0.471876 33.240885 0.348958   
std 7.884160 0.331329 11.760232 0.476951   
min 0.000000 0.078000 21.000000 0.000000   
25% 27.300000 0.243750 24.000000 0.000000   
50% 32.000000 0.372500 29.000000 0.000000   
75% 36.600000 0.626250 41.000000 1.000000   
max 67.100000 2.420000 81.000000 1.000000

* Các thống kê mô tả chính và ý nghĩa:
  + count: số lượng mẫu dữ liệu (ở đây tất cả cột = 768).
  + mean: giá trị trung bình.
  + std: độ lệch chuẩn → mức độ phân tán quanh trung bình.
  + min, max: giá trị nhỏ nhất và lớn nhất.
  + 25%, 50%, 75% (quartiles): các mốc phân vị → dùng để đánh giá sự phân phối dữ liệu.
* Chỉ số đo lường sự biến động (Dispersion):
  + std (độ lệch chuẩn):
    - Glucose: 32 → biến thiên khá lớn, thể hiện sự khác biệt giữa người khỏe mạnh và người có nguy cơ cao.
    - Insulin: 115, lớn hơn cả mean (~80) → chứng tỏ dữ liệu rất phân tán và nhiều ngoại lệ.
    - BMI: 7.88 → tương đối ổn định, dao động vừa phải.
    - Age: 11.8 → đa dạng độ tuổi, từ 21 đến 81 tuổi.
* Khoảng (range = max - min):
  + Pregnancies: 0–17, cho thấy có vài trường hợp đặc biệt mang thai rất nhiều lần.
  + Insulin: 0–846, cực kỳ phân tán → cần chuẩn hóa hoặc log-transform khi phân tích.
  + SkinThickness: 0–99, cũng có nhiều giá trị bất hợp lý.

**Code**

# Phân bố lớp  
print("\nPhân bố biến lớp (Outcome):")  
print(data['Outcome'].value\_counts())

Phân bố biến lớp (Outcome):  
Outcome  
0 500  
1 268  
Name: count, dtype: int64

**Kết luận**

*Tỉ lệ mắc ~35% và không mắc ~65%*

1. **Chuẩn bị dữ liệu**
   1. **Mô tả vấn đề**

Mục tiêu: Trong phần này, chúng ta sẽ làm sạch dataset Pima Indians Diabetes bằng cách xử lý giá trị thiếu, loại bỏ dữ liệu trùng lặp, và thực hiện các thao tác cần thiết để đảm bảo dữ liệu sẵn sàng cho phân tích sâu hơn. Điều này giúp loại bỏ các sai lệch và chuẩn bị một nền tảng vững chắc cho các bước tiếp theo như trực quan hóa và mô hình hóa.

* 1. **Làm sạch dữ liệu**
     1. **Xử lí dữ liệu bị thiếu**

Giới thiệu thư viện NumPy và SimpleImputer

NumPy (numpy):

* Là thư viện **tính toán khoa học** cốt lõi trong Python.
* Hỗ trợ mảng đa chiều (ndarray) và nhiều hàm toán học tối ưu.
* Trong phân tích dữ liệu, NumPy thường được dùng kết hợp với pandas để xử lý dữ liệu số học, thống kê, và tối ưu hiệu năng.

SimpleImputer (từ sklearn.impute):

* Là công cụ trong thư viện **Scikit-learn** hỗ trợ **xử lý dữ liệu thiếu (missing values)**.
* Có thể thay thế (impute) giá trị thiếu bằng:
* Trung bình (mean)
* Trung vị (median)
* Giá trị thường gặp nhất (most\_frequent)
* Một hằng số tùy chọn (constant)
* Đây là bước quan trọng để đảm bảo dữ liệu **sạch và đầy đủ** trước khi đưa vào huấn luyện mô hình học máy.

**Code**:

*import numpy as np  
from sklearn.impute import SimpleImputer*

# Thay thế giá trị 0 bằng NaN để xử lý *for column in ['Glucose', 'BloodPressure', 'SkinThickness', 'Insulin', 'BMI']:  
 data[column] = data[column].replace(0, np.nan)*# Sử dụng SimpleImputer để điền giá trị thiếu bằng trung bình *imputer = SimpleImputer(strategy='mean')  
data[['Glucose', 'BloodPressure', 'SkinThickness', 'Insulin', 'BMI']] = imputer.fit\_transform(data[['Glucose', 'BloodPressure', 'SkinThickness', 'Insulin', 'BMI']])*# Kiểm tra lại số lượng giá trị thiếu *print("Số lượng giá trị thiếu sau khi xử lý:")  
print(data.isnull().sum())*

**Kết quả**

Số lượng giá trị thiếu sau khi xử lý:  
Pregnancies 0  
Glucose 0  
BloodPressure 0  
SkinThickness 0  
Insulin 0  
BMI 0  
DiabetesPedigreeFunction 0  
Age 0  
Outcome 0  
dtype: int64

* Giải thích:
* Việc thay thế bằng trung bình giúp duy trì tính đại diện của dữ liệu mà không làm mất thông tin quan trọng.
* SimpleImputer với chiến lược mean thay thế các giá trị thiếu (bao gồm cả null và NaN) bằng trung bình của cột, giữ nguyên ý nghĩa thống kê.
* Sau khi xử lý, tất cả giá trị thiếu sẽ được điền, đảm bảo dữ liệu không còn khoảng trống.
  + 1. **Xóa dữ liệu trùng lặp**

**Code:**

# Kiểm tra và xóa dữ liệu trùng lặp  
initial\_rows = data.shape[0]  
data = data.drop\_duplicates()  
  
# Hiển thị số lượng mẫu trước và sau khi xóa trùng lặp  
print(f"\nSố lượng mẫu trước khi xóa trùng lặp: {initial\_rows}")  
print(f"Số lượng mẫu sau khi xóa trùng lặp: {data.shape[0]}")  
if initial\_rows == data.shape[0]:  
 print("Không có dữ liệu trùng lặp trong dataset.")  
else:  
 print(f"Đã xóa {initial\_rows - data.shape[0]} mẫu trùng lặp.")

**Kết quả:**

Số lượng mẫu trước khi xóa trùng lặp: 768  
Số lượng mẫu sau khi xóa trùng lặp: 768  
Không có dữ liệu trùng lặp trong dataset.

* Giải thích:
* drop\_duplicates() loại bỏ các hàng có giá trị giống nhau hoàn toàn.
* Kết quả sẽ cho biết có bao nhiêu mẫu bị trùng lặp (nếu có), giúp dữ liệu trở nên sạch sẽ và đáng tin cậy hơn.
  + 1. **Chuẩn hóa dữ liệu**

Giới thiệu MinMaxScaler (sklearn.preprocessing)

**MinMaxScaler** là một công cụ trong thư viện **Scikit-learn** dùng để **chuẩn hóa dữ liệu (feature scaling)**.

Nguyên lý hoạt động: đưa toàn bộ giá trị của một biến về trong **khoảng [0, 1]** (hoặc một khoảng tùy chọn do người dùng định nghĩa). Công thức biến đổi:

* - Trong đó:
  + - * + (X) là giá trị gốc,
        + (X\_{min}) là giá trị nhỏ nhất của cột,
        + (X\_{max}) là giá trị lớn nhất của cột,
        + (X') là giá trị đã được scale (nằm trong [0,1]).

**Code:**

# Chuẩn hóa dữ liệu bằng MinMaxScaler  
from sklearn.preprocessing import MinMaxScaler

# Kiểm tra phạm vi giá trị của các cột  
print("\nPhạm vi giá trị của các cột:")  
print(data.describe())  
  
scaler = MinMaxScaler()  
data\_scaled = pd.DataFrame(scaler.fit\_transform(data.drop('Outcome', axis=1)), columns=data.drop('Outcome', axis=1).columns)  
data\_scaled['Outcome'] = data['Outcome']

**Kết quả:**

Phạm vi giá trị của các cột:  
 Pregnancies Glucose BloodPressure SkinThickness Insulin \  
count 768.000000 768.000000 768.000000 768.000000 768.000000   
mean 3.845052 121.686763 72.405184 29.153420 155.548223   
std 3.369578 30.435949 12.096346 8.790942 85.021108   
min 0.000000 44.000000 24.000000 7.000000 14.000000   
25% 1.000000 99.750000 64.000000 25.000000 121.500000   
50% 3.000000 117.000000 72.202592 29.153420 155.548223   
75% 6.000000 140.250000 80.000000 32.000000 155.548223   
max 17.000000 199.000000 122.000000 99.000000 846.000000   
  
 BMI DiabetesPedigreeFunction Age Outcome   
count 768.000000 768.000000 768.000000 768.000000   
mean 32.457464 0.471876 33.240885 0.348958   
std 6.875151 0.331329 11.760232 0.476951   
min 18.200000 0.078000 21.000000 0.000000   
25% 27.500000 0.243750 24.000000 0.000000   
50% 32.400000 0.372500 29.000000 0.000000   
75% 36.600000 0.626250 41.000000 1.000000   
max 67.100000 2.420000 81.000000 1.000000

**Code:**

# Hiển thị 5 dòng đầu của dữ liệu đã chuẩn hóa  
print("\n5 dòng đầu của dữ liệu đã chuẩn hóa:")  
print(data\_scaled.head())

**Kết quả:**  
  
5 dòng đầu của dữ liệu đã chuẩn hóa:  
 Pregnancies Glucose BloodPressure SkinThickness Insulin BMI \  
0 0.352941 0.670968 0.489796 0.304348 0.170130 0.314928   
1 0.058824 0.264516 0.428571 0.239130 0.170130 0.171779   
2 0.470588 0.896774 0.408163 0.240798 0.170130 0.104294   
3 0.058824 0.290323 0.428571 0.173913 0.096154 0.202454   
4 0.000000 0.600000 0.163265 0.304348 0.185096 0.509202   
  
 DiabetesPedigreeFunction Age Outcome   
0 0.234415 0.483333 1   
1 0.116567 0.166667 0   
2 0.253629 0.183333 1   
3 0.038002 0.000000 0   
4 0.943638 0.200000 1

**Giải thích:**

* + describe() kiểm tra phạm vi giá trị (min, max) để đánh giá cần chuẩn hóa không. Ví dụ, Age và Pregnancies có thể có giá trị lớn hơn các cột khác.
  + MinMaxScaler chuẩn hóa dữ liệu về khoảng [0, 1], giúp các thuộc tính có cùng thang đo, đặc biệt quan trọng cho mô hình hóa sau này.
  + Dữ liệu chuẩn hóa được lưu vào data\_scaled. Có thể gán data = data\_scaled nếu muốn sử dụng ngay.
    1. **Kiểm tra dữ liệu sau khi xử lí**

**Code:**

# Hiển thị thông tin cơ bản sau khi xử lý  
print("\nThông tin cơ bản về dữ liệu sau khi xử lý:")  
print(data.info())  
  
# Phân bố lớp sau khi xử lý  
print("\nPhân bố biến lớp (Outcome) sau khi xử lý:")  
print(data['Outcome'].value\_counts())

**Kết quả:**

Thông tin cơ bản về dữ liệu sau khi xử lý:  
<class 'pandas.core.frame.DataFrame'>  
RangeIndex: 768 entries, 0 to 767  
Data columns (total 9 columns):  
 # Column Non-Null Count Dtype   
--- ------ -------------- -----   
 0 Pregnancies 768 non-null int64   
 1 Glucose 768 non-null float64  
 2 BloodPressure 768 non-null float64  
 3 SkinThickness 768 non-null float64  
 4 Insulin 768 non-null float64  
 5 BMI 768 non-null float64  
 6 DiabetesPedigreeFunction 768 non-null float64  
 7 Age 768 non-null int64   
 8 Outcome 768 non-null int64   
dtypes: float64(6), int64(3)  
memory usage: 54.1 KB  
None  
  
Phân bố biến lớp (Outcome) sau khi xử lý:  
Outcome  
0 500  
1 268  
Name: count, dtype: int64

**Giải thích:**

* + - info() xác nhận không còn giá trị thiếu và kiểu dữ liệu phù hợp.
    - value\_counts() kiểm tra phân bố lớp vẫn giữ nguyên (500 mẫu 0 và 268 mẫu 1), đảm bảo không mất dữ liệu quan trọng.

1. **Phân tích dữ liệu**
   1. **Mô tả vấn đề**

Mục tiêu: Trong phần này, chúng ta sẽ trực quan hóa dataset Pima Indians Diabetes để khám phá các xu hướng, mối quan hệ giữa các thuộc tính, và nhận xét dựa trên phân tích. Đặc biệt, mình sẽ áp dụng các công thức và phương pháp từ bài báo "Using the ADAP Learning Algorithm" (paper2.pdf), như độ nhạy (sensitivity), độ đặc hiệu (specificity), và đường cong ROC, để đánh giá hiệu suất dự đoán đái tháo đường. Điều này giúp hiểu rõ hơn về dữ liệu và chuẩn bị cho mô hình hóa.

Dữ liệu: Dataset đã được làm sạch trong Phần 3 ***(data\_scaled)***, với 768 mẫu, 8 thuộc tính y tế, và biến lớp Outcome (0: Không đái tháo đường, 1: Có đái tháo đường).

* 1. **Thống kê mô tả**
     1. **Hiển thị một số thông tin về dữ liệu**
     + Số dòng, số cột của dữ liệu
     + Kiểu dữ liệu của từng cột
     + 5 dòng đầu và 5 dòng cuối của bảng dữ liệu
     + Thông tin chung về dữ liệu

**Code:**

# shape  
print(f'+ Shape: {data\_scaled.shape}')  
# types  
print(f'+ Data Types: \n{data\_scaled.dtypes}')  
# head, tail  
print(f'+ Contents: ')  
print('--- First 5 rows ---')  
print(data\_scaled.head(5))  
print('--- Last 5 rows ---')  
print(data\_scaled.tail(5))  
# info  
data\_scaled.info()

**Kết quả:**

+ Shape: (768, 9)  
+ Data Types:   
Pregnancies float64  
Glucose float64  
BloodPressure float64  
SkinThickness float64  
Insulin float64  
BMI float64  
DiabetesPedigreeFunction float64  
Age float64  
Outcome int64  
dtype: object  
+ Contents:   
--- First 5 rows ---  
 Pregnancies Glucose BloodPressure SkinThickness Insulin BMI \  
0 0.352941 0.670968 0.489796 0.304348 0.170130 0.314928   
1 0.058824 0.264516 0.428571 0.239130 0.170130 0.171779   
2 0.470588 0.896774 0.408163 0.240798 0.170130 0.104294   
3 0.058824 0.290323 0.428571 0.173913 0.096154 0.202454   
4 0.000000 0.600000 0.163265 0.304348 0.185096 0.509202   
  
 DiabetesPedigreeFunction Age Outcome   
0 0.234415 0.483333 1   
1 0.116567 0.166667 0   
2 0.253629 0.183333 1   
3 0.038002 0.000000 0   
4 0.943638 0.200000 1   
--- Last 5 rows ---  
 Pregnancies Glucose BloodPressure SkinThickness Insulin BMI \  
763 0.588235 0.367742 0.530612 0.445652 0.199519 0.300613   
764 0.117647 0.503226 0.469388 0.217391 0.170130 0.380368   
765 0.294118 0.496774 0.489796 0.173913 0.117788 0.163599   
766 0.058824 0.529032 0.367347 0.240798 0.170130 0.243354   
767 0.058824 0.316129 0.469388 0.260870 0.170130 0.249489   
  
 DiabetesPedigreeFunction Age Outcome   
763 0.039710 0.700000 0   
764 0.111870 0.100000 0   
765 0.071307 0.150000 0   
766 0.115713 0.433333 1   
767 0.101196 0.033333 0   
<class 'pandas.core.frame.DataFrame'>  
RangeIndex: 768 entries, 0 to 767  
Data columns (total 9 columns):  
 # Column Non-Null Count Dtype   
--- ------ -------------- -----   
 0 Pregnancies 768 non-null float64  
 1 Glucose 768 non-null float64  
 2 BloodPressure 768 non-null float64  
 3 SkinThickness 768 non-null float64  
 4 Insulin 768 non-null float64  
 5 BMI 768 non-null float64  
 6 DiabetesPedigreeFunction 768 non-null float64  
 7 Age 768 non-null float64  
 8 Outcome 768 non-null int64   
dtypes: float64(8), int64(1)  
memory usage: 54.1 KB

**Nhận xét:**

* + - * + Dataset sau khi chuẩn hóa (data\_scaled) có 768 dòng và 9 cột, không có giá trị thiếu, đảm bảo chất lượng dữ liệu cho phân tích tiếp theo.
        + Các thuộc tính số đã được chuẩn hóa về khoảng [0, 1], giúp các đặc trưng có cùng thang đo, thuận lợi cho các thuật toán học máy.
        + Kiểu dữ liệu của các cột phù hợp: hầu hết là float64, riêng biến mục tiêu Outcome là int64.
        + Phân bố lớp (Outcome) vẫn giữ nguyên, đảm bảo không mất dữ liệu quan trọng sau các bước xử lý.
        + Việc chuẩn hóa và làm sạch dữ liệu giúp tăng độ tin cậy và hiệu quả cho các bước phân tích, trực quan hóa và xây dựng mô hình dự đoán sau này.
    1. **Kiểm tra tính toàn vẹn của dữ liệu**
    - Có còn giá trị thiếu (NaN, Null) nào trong từng cột không?
    - Có dòng dữ liệu nào bị trùng lặp không?
    - Các giá trị của biến mục tiêu (Outcome) có nằm ngoài tập {0, 1} không?
    - Có cột nào chứa toàn giá trị giống nhau không?
    - Các cột số đã chuẩn hóa có nằm trong khoảng [0, 1] không?

**Code:**

# Kiểm tra tính toàn vẹn của dữ liệu  
print("1. Có còn giá trị thiếu (NaN, Null) nào trong từng cột không?")  
print(data\_scaled.isnull().sum())  
  
print("\n2. Có dòng dữ liệu nào bị trùng lặp không?")  
print(f"Số dòng trùng lặp: {data\_scaled.duplicated().sum()}")  
  
print("\n3. Các giá trị của biến mục tiêu (Outcome) có nằm ngoài tập {0, 1} không?")  
print(data\_scaled['Outcome'].unique())  
  
print("\n4. Có cột nào chứa toàn giá trị giống nhau không?")   
for col in data\_scaled.columns:  
 if data\_scaled[col].nunique() == 1:  
 print(f"Cột {col} chỉ có một giá trị duy nhất.")  
   
print("\n5. Các cột số đã chuẩn hóa có nằm trong khoảng [0, 1] không?")  
for col in data\_scaled.columns:  
 if col != 'Outcome':  
 min\_val = data\_scaled[col].min()  
 max\_val = data\_scaled[col].max()  
 if min\_val < 0 or max\_val > 1:  
 print(f"Cột {col} có giá trị ngoài khoảng [0, 1]: min={min\_val}, max={max\_val}")

**Kết quả:**

1. Có còn giá trị thiếu (NaN, Null) nào trong từng cột không?  
Pregnancies 0  
Glucose 0  
BloodPressure 0  
SkinThickness 0  
Insulin 0  
BMI 0  
DiabetesPedigreeFunction 0  
Age 0  
Outcome 0  
dtype: int64  
  
2. Có dòng dữ liệu nào bị trùng lặp không?  
Số dòng trùng lặp: 0  
  
3. Các giá trị của biến mục tiêu (Outcome) có nằm ngoài tập {0, 1} không?  
[1 0]  
  
4. Có cột nào chứa toàn giá trị giống nhau không?  
  
5. Các cột số đã chuẩn hóa có nằm trong khoảng [0, 1] không?  
Cột BMI có giá trị ngoài khoảng [0, 1]: min=0.0, max=1.0000000000000002

data.to\_csv('output.csv', index=False)

**Nhận xét:**

* + - * + Không còn giá trị thiếu (NaN, Null) trong bất kỳ cột nào, đảm bảo dữ liệu đầy đủ cho phân tích.
        + Không có dòng dữ liệu nào bị trùng lặp, dữ liệu duy nhất và đáng tin cậy.
        + Biến mục tiêu (Outcome) chỉ chứa giá trị 0 và 1, đúng định dạng nhị phân.
        + Không có cột nào chứa toàn giá trị giống nhau, đảm bảo các thuộc tính đều có ý nghĩa phân biệt.
        + Các cột số đã được chuẩn hóa nằm trong khoảng [0, 1], phù hợp cho các thuật toán học máy và phân tích tiếp theo.
  1. **Trực quan hóa dữ liệu**
     1. **Phân bố biến lớp**

Giới thiệu thư viện Seaborn và Matplotlib

* + - * + **Matplotlib (matplotlib.pyplot)**

Là thư viện trực quan hóa dữ liệu cơ bản và mạnh mẽ nhất trong Python.

Cho phép vẽ nhiều loại biểu đồ: line chart, bar chart, scatter plot, histogram, pie chart...

Thường được sử dụng như “nền tảng” cho các thư viện trực quan hóa khác.

* + - * + **Seaborn (seaborn)**

Được xây dựng dựa trên Matplotlib, cung cấp **giao diện đơn giản và đẹp mắt** để trực quan hóa dữ liệu thống kê.

Tích hợp tốt với **pandas DataFrame**.

Hỗ trợ nhiều biểu đồ nâng cao: heatmap, pairplot, violin plot, boxplot...

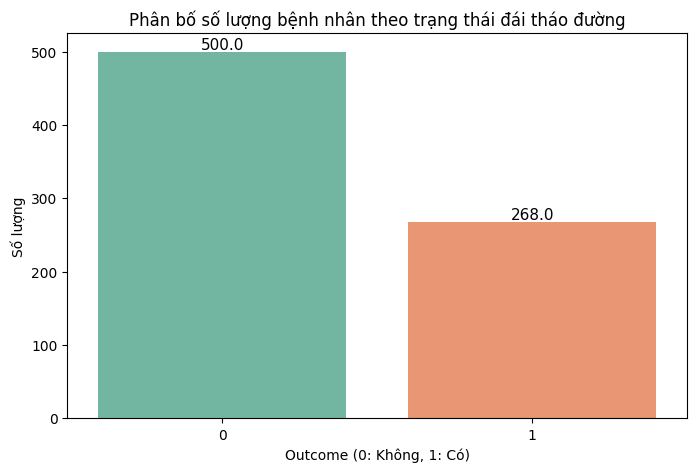
Tự động tối ưu hóa màu sắc và phong cách hiển thị.

**Code:**

*import seaborn as sns  
import matplotlib.pyplot as plt*

# Phân bố lớp *plt.figure(figsize=(8, 5))  
sns.countplot(x='Outcome', data=data\_scaled, palette='Set2')  
plt.title('Phân bố số lượng bệnh nhân theo trạng thái đái tháo đường')*# Hiển thị thêm con số trên cột *for p in plt.gca().patches:  
 plt.gca().annotate(f'{p.get\_height()}', (p.get\_x() + p.get\_width() / 2., p.get\_height()), ha='center', va='center', fontsize=11, color='black', xytext=(0, 5), textcoords='offset points')* # giải thích: p.get\_x() + p.get\_width() / 2. là vị trí x ở giữa cột, p.get\_height() là vị trí y trên đỉnh cột *plt.xlabel('Outcome (0: Không, 1: Có)')  
plt.ylabel('Số lượng')  
plt.show()*

**Biểu đồ:**



**Nhận xét:**

Dữ liệu cho thấy 500 mẫu không đái tháo đường (0) và 268 mẫu có đái tháo đường (1), chiếm tỷ lệ khoảng 65% và 35%. Sự mất cân bằng này cần được xem xét khi xây dựng mô hình, phù hợp với phân bố được ghi nhận trong tài liệu.

* + 1. **Phân bố các thuộc tính**

Ý nghĩa của việc sử dụng biểu đồ Histogram + KDE để trực quan dữ liệu

* **Histogram (biểu đồ tần suất):**

Cho thấy sự phân bố của dữ liệu theo từng biến (feature).

Giúp ta hiểu được giá trị nào xuất hiện nhiều, giá trị nào ít, có bị lệch (skewness) hay không.

Dễ dàng nhận ra các giá trị bất thường (outliers) hoặc dữ liệu thiếu (ví dụ: nhiều giá trị 0 ở Insulin, SkinThickness).

* **KDE (đường mật độ)**

Biểu diễn đường cong mật độ xác suất ước lượng từ histogram.

Làm cho ta thấy rõ hơn **hình dạng phân phối** (chuẩn, lệch phải, lệch trái).

Giúp so sánh mượt mà hơn giữa các nhóm.

* **Sử dụng hue = ‘Outcome’**

Cho phép **so sánh phân phối** giữa hai nhóm:

Outcome = 0: không mắc tiểu đường.

Outcome = 1: mắc tiểu đường.

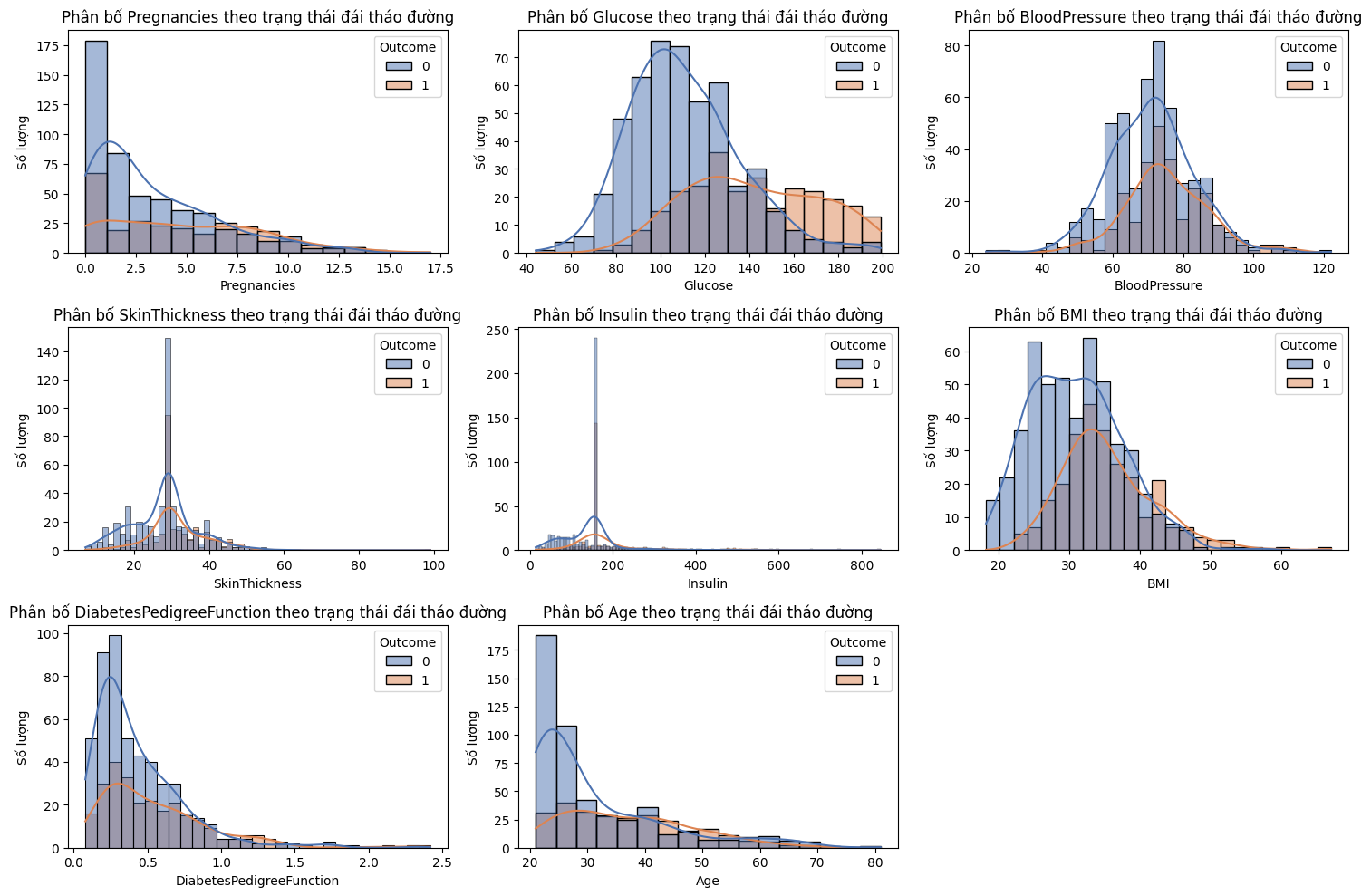
Giúp ta phát hiện biến nào có **khả năng phân biệt mạnh** (ví dụ: Glucose, BMI), và biến nào ít phân biệt (ví dụ: BloodPressure).

* + - * + **Biểu đồ này giúp:**
* **Hiểu cấu trúc dữ liệu** trước khi làm mô hình.
* **Xác định biến quan trọng** (feature importance trực quan).
* **Phát hiện dữ liệu bất thường hoặc thiếu**.
* Đưa ra định hướng cho **tiền xử lý dữ liệu** (xử lý outlier, missing values, chuẩn hóa).

**Code:**

# Vẽ histogram cho tất cả các thuộc tính  
plt.figure(figsize=(15, 10))  
for i, column in enumerate(data.columns[:-1], 1): # Loại bỏ 'Outcome' khỏi histogram  
 plt.subplot(3, 3, i)  
 sns.histplot(data=data, x=column, hue='Outcome', kde=True, palette='deep')  
 plt.title(f'Phân bố {column} theo trạng thái đái tháo đường')  
 plt.xlabel(column)  
 plt.ylabel('Số lượng')  
plt.tight\_layout()  
plt.show()

**Biển đồ:**



**Nhận xét:**

* + - * + Pregnancies (Số lần mang thai):
* Phân bố lệch phải, với phần lớn giá trị từ 0-5. Nhóm Outcome = 1 có xu hướng tập trung ở số lần mang thai cao hơn (4-10), gợi ý rằng tiền sử sinh sản nhiều có thể làm tăng nguy cơ, phù hợp với nghiên cứu dịch tễ trong paper3.pdf.
  + - * + Glucose (Nồng độ glucose):
* Đây là biến phân tách rõ ràng nhất:
* Phân bố gần chuẩn, nhưng nhóm Outcome = 1 có đỉnh rõ ở mức >120 mg/dl, hỗ trợ tiêu chí chẩn đoán WHO (≥200 mg/dl, paper1.pdf).
* Nhóm tiểu đường có giá trị glucose cao hơn hẳn nhóm không tiểu đường. -Đây là biến quan trọng nhất để phân loại.
  + - * + BloodPressure (Huyết áp tâm thu): -Phân bố tập trung quanh 70-80 mm Hg, không có sự khác biệt rõ rệt giữa hai nhóm, cho thấy huyết áp có thể không phải yếu tố quyết định trong dataset này.
        + SkinThickness (Độ dày nếp gấp da):
* Phân bố lệch phải, nhóm Outcome = 1 có xu hướng cao hơn (20-40 mm), phản ánh mối liên hệ với béo phì, như được đề cập trong paper3.pdf.
  + - * + Insulin (Insulin huyết thanh):
* Phân bố rất lệch, với nhiều giá trị thấp. Nhóm Outcome = 1 có xu hướng cao hơn, nhưng dữ liệu thiếu nhiều (giá trị 0 ban đầu) có thể ảnh hưởng đến phân tích.
  + - * + BMI (Chỉ số khối cơ thể):
* Phân bố tập trung quanh 30-40, với nhóm Outcome = 1 nghiêng về BMI >30 (béo phì), củng cố vai trò của béo phì trong đái tháo đường loại 2 (paper3.pdf).
* Nhóm tiểu đường có xu hướng BMI cao hơn.
* Đây là một yếu tố nguy cơ quan trọng.
  + - * + DiabetesPedigreeFunction (Hàm phả hệ):
* Phân bố lệch phải, nhóm Outcome = 1 có giá trị cao hơn (0.4-1.0), cho thấy yếu tố di truyền quan trọng, như phân tích trong paper2.pdf.
  + - * + Age (Tuổi):
* Người tiểu đường có xu hướng lớn tuổi hơn.
* Yếu tố tuổi tác cũng góp phần tăng nguy cơ.
* Phân bố lệch phải, nhóm Outcome = 1 tập trung ở tuổi >40, phù hợp với nghiên cứu về lão hóa và nguy cơ đái tháo đường (paper1.pdf).
  1. **Trực quan hóa mối liên hệ giữa các thuộc tính**
     1. **Ma trận tương quan**

**Ý nghĩa:**

Ma trận tương quan cho thấy mối quan hệ tuyến tính giữa các cặp biến.

Hệ số tương quan (Pearson):

* Gần **1** → tương quan thuận mạnh.
* Gần **-1** → tương quan nghịch mạnh.
* Gần **0** → ít hoặc không có quan hệ tuyến tính.

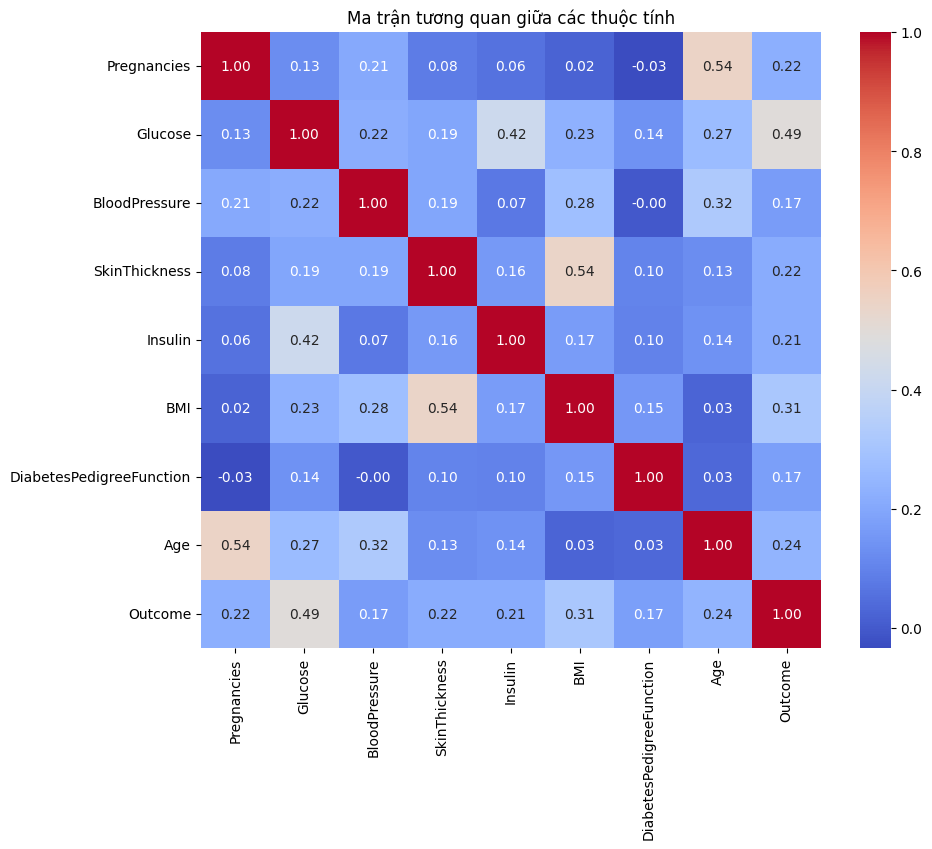
Màu sắc trên heatmap:

* **Đỏ** = tương quan dương mạnh.
* **Xanh** = tương quan yếu hoặc âm.

**Code:**

# Ma trận tương quan  
plt.figure(figsize=(10, 8))  
sns.heatmap(data\_scaled.corr(), annot=True, cmap='coolwarm', fmt='.2f')  
plt.title('Ma trận tương quan giữa các thuộc tính')  
plt.show()

**Biểu đồ:**



**Nhận xét:**

1. **Mối quan hệ với nhãn Outcome:**
   * + - * Glucose có tương quan cao nhất (0.49) → cho thấy nồng độ glucose cao là yếu tố dự báo quan trọng cho bệnh đái tháo đường, phù hợp với kiến thức y học.
         * BMI (0.31), Age (0.24), Pregnancies (0.22), SkinThickness (0.22) → có mối quan hệ dương nhưng yếu hơn. cho thấy béo phì, tuổi cao và yếu tố di truyền đều góp phần làm tăng nguy cơ mắc bệnh.
         * BloodPressure, Insulin, DiabetesPedigreeFunction → tương quan rất thấp (< 0.21).
2. **Mối quan hệ giữa các biến độc lập:**
   * + - * SkinThickness ↔ BMI (0.54) → hợp lý, do mỡ dưới da thường liên quan đến BMI.
         * Pregnancies ↔ Age (0.54) → càng lớn tuổi thì số lần mang thai càng nhiều.
         * Glucose ↔ Insulin (0.42) → đường huyết cao thường đi kèm insulin cao.
3. **Đánh giá cho mô hình học máy:**
   * + - * Không có cặp biến nào tương quan quá cao (>0.8) → ít nguy cơ đa cộng tuyến.
         * Các biến khác nhau đóng góp bổ sung thông tin.
         * Không có cặp thuộc tính nào có tương quan tuyệt đối (gần 1 hoặc -1), do đó dữ liệu phù hợp cho các mô hình học máy mà không lo trùng lặp thông tin.
     1. **Hiển thị nhiều tính chất**

**Phân tích biểu đồ Pairplot**

Pairplot là một công cụ trực quan hóa mạnh mẽ để hiển thị mối quan hệ giữa tất cả các cặp biến trong dataset. Biểu đồ này tạo ra một ma trận các scatter plot (biểu đồ phân tán) và histogram. **Cấu trúc biểu đồ:**

Đường chéo chính: Histogram phân phối của từng biến

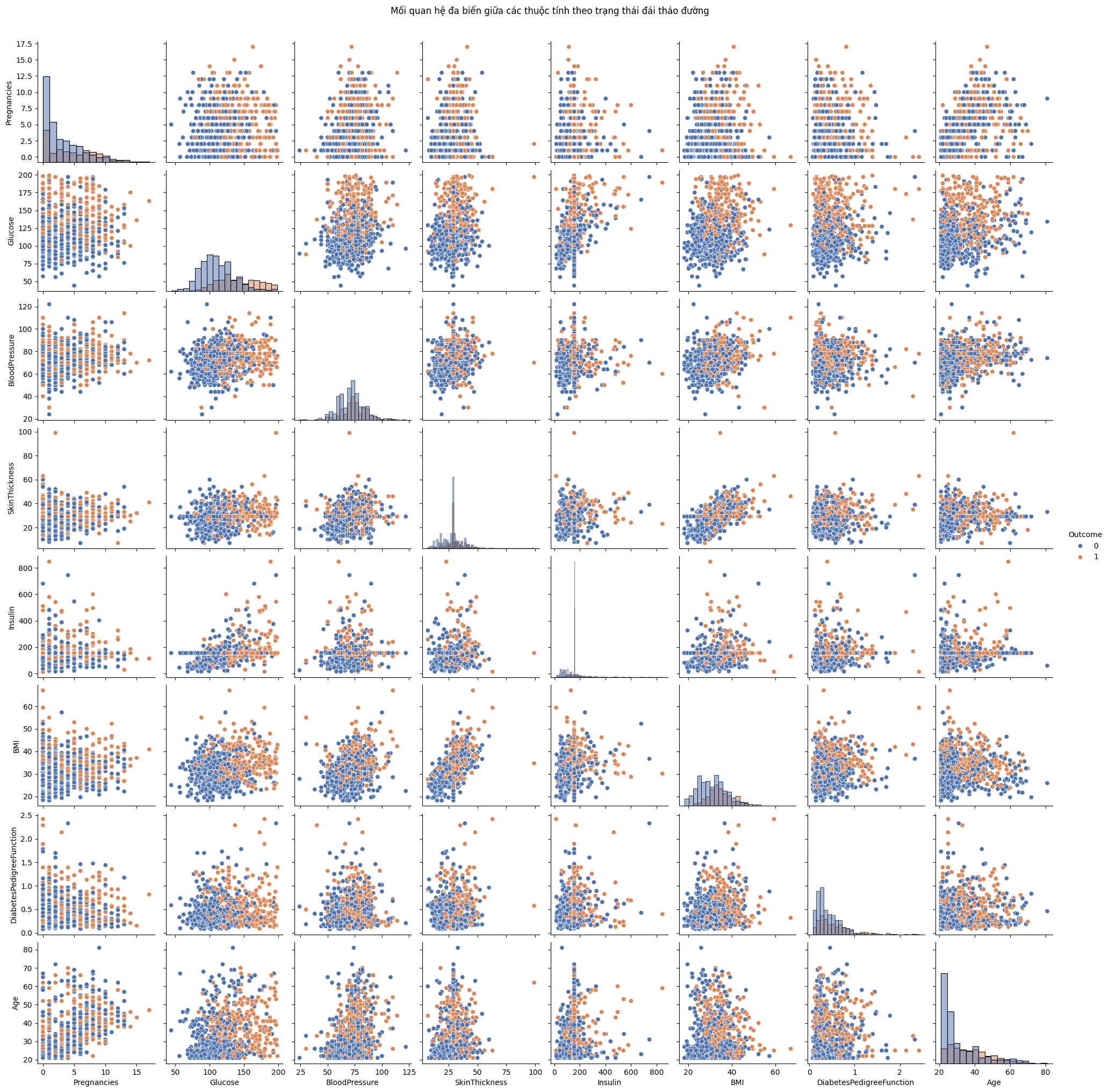
Các ô ngoài đường chéo: Scatter plot giữa hai biến

Màu sắc: Phân biệt theo nhóm (Outcome: 0 = không đái tháo đường, 1 = có đái tháo đường) **Mục đích:** Biểu đồ pairplot này cung cấp một cái nhìn toàn diện về mối quan hệ giữa các yếu tố nguy cơ và tình trạng đái tháo đường.

**Code:**

# Vẽ pairplot để phân tích mối quan hệ đa biến  
plt.figure(figsize=(12, 10))  
sns.pairplot(data, hue='Outcome', palette='deep', diag\_kind='hist')  
plt.suptitle('Mối quan hệ đa biến giữa các thuộc tính theo trạng thái đái tháo đường', y=1.02)  
plt.show()

**Biểu đồ:**



**Nhận xét:**

Glucose (Đường huyết) Quan sát: Có sự phân tách rõ ràng giữa 2 nhóm Ý nghĩa: Những người có glucose cao (>120-140 mg/dL) có xu hướng mắc đái tháo đường Kết luận: Glucose là yếu tố dự đoán mạnh nhất

BMI (Chỉ số khối cơ thể) Quan sát: Nhóm bị đái tháo đường có BMI cao hơn Ý nghĩa: Thể trọng tăng là yếu tố nguy cơ quan trọng Kết luận: BMI > 30 tăng nguy cơ đáng kể

Age (Tuổi) Quan sát: Tần suất bệnh tăng theo tuổi Ý nghĩa: Tuổi tác là yếu tố nguy cơ tự nhiên Kết luận: Người trên 40 tuổi có nguy cơ cao hơn

Pregnancies (Số lần mang thai) Quan sát: Có mối liên hệ với bệnh ở phụ nữ Ý nghĩa: Đái tháo đường thai kỳ có thể dẫn đến đái tháo đường type 2 Kết luận: Nhiều lần mang thai tăng nguy cơ

DiabetesPedigreeFunction (Yếu tố di truyền) Quan sát: Phân bố khác biệt giữa 2 nhóm Ý nghĩa: Tiền sử gia đình ảnh hưởng đến nguy cơ Kết luận: Di truyền là yếu tố quan trọng

**Mối quan hệ giữa các biến:**

Tương quan tích cực mạnh: Glucose - Outcome: Mối liên hệ mạnh nhất BMI - SkinThickness: Thể hiện tình trạng béo phì Age - Pregnancies: Tự nhiên theo thời gian

Tương quan yếu hoặc không đáng kể: BloodPressure: Không phân biệt rõ ràng giữa 2 nhóm SkinThickness: Có xu hướng nhưng không mạnh Insulin: Phân tán nhiều, khó phân biệt

**Ý nghĩa lâm sàng:**

Chẩn đoán và sàng lọc: Glucose là chỉ số quan trọng nhất để chẩn đoán BMI giúp đánh giá nguy cơ Age và Pregnancies là yếu tố nguy cơ bổ sung

Dự phòng: Kiểm soát cân nặng (BMI) Theo dõi đường huyết định kỳ Chú ý đặc biệt với phụ nữ có nhiều lần mang thai Tầm soát sớm ở người có tiền sử gia đình

***Kết luận tổng thể:*** Biểu đồ pairplot cho thấy Glucose, BMI, và Age là ba yếu tố quan trọng nhất trong việc dự đoán đái tháo đường. Sự kết hợp của nhiều yếu tố sẽ cho độ chính xác cao hơn so với chỉ dựa vào một yếu tố đơn lẻ. Điều này hỗ trợ việc phát triển các mô hình machine learning để dự đoán và chẩn đoán bệnh đái tháo đường.

* + 1. **Biểu đồ scatter giữa Glucose và BMI**

**Phân tích biểu đồ Scatter**

* + - * + Scatter plot (biểu đồ phân tán) hiển thị mối quan hệ giữa hai biến liên tục - trong trường hợp này là Glucose (đường huyết) và BMI (chỉ số khối cơ thể). Màu sắc phân biệt giữa hai nhóm:

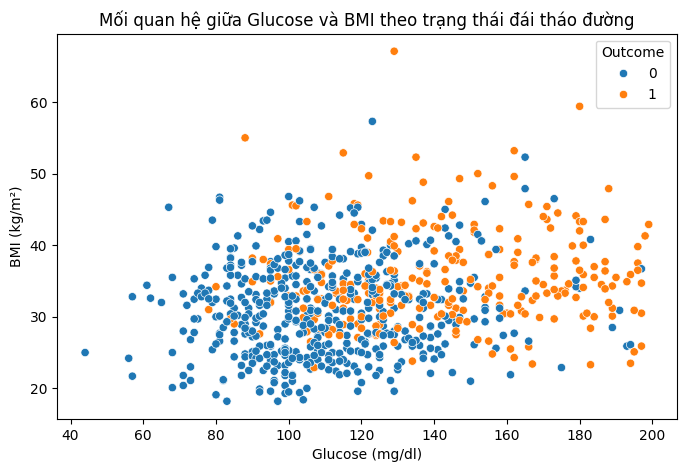
Màu xanh (0): Không bị đái tháo đường

Màu cam (1): Bị đái tháo đường

**Code:**

# Biểu đồ scatter giữa Glucose và BMI  
plt.figure(figsize=(8, 5))  
sns.scatterplot(x='Glucose', y='BMI', hue='Outcome', data=data)  
plt.title('Mối quan hệ giữa Glucose và BMI theo trạng thái đái tháo đường')  
plt.xlabel('Glucose (mg/dl)')  
plt.ylabel('BMI (kg/m²)')  
plt.show()

**Biểu đồ:**



**Phân tích chi tiết các vùng trên biểu đồ:**

* + - * + Vùng 1: Glucose thấp (< 100 mg/dl) - BMI thấp (< 25) Đặc điểm: Chủ yếu là điểm màu xanh (không bị bệnh) Ý nghĩa: Vùng "an toàn" - ít nguy cơ đái tháo đường nhất Khuyến nghị: Duy trì lối sống lành mạnh để giữ ở mức này
        + Vùng 2: Glucose cao (> 126 mg/dl) - BMI cao (> 30) Đặc điểm: Tập trung nhiều điểm màu cam (bị bệnh) Ý nghĩa: Vùng "nguy hiểm" - nguy cơ cao nhất Khuyến nghị: Cần can thiệp y tế tích cực, kiểm soát cả đường huyết và cân nặng
        + Vùng 3: Glucose trung bình (100-126 mg/dl) - BMI trung bình (25-30) Đặc điểm: Hỗn hợp cả hai màu, nhưng có xu hướng tăng màu cam Ý nghĩa: Vùng "cảnh báo" - tiền đái tháo đường Khuyến nghị: Theo dõi sát, thay đổi lối sống để phòng ngừa
  1. **Thuật toán học ADAP để dự đoán sự khởi phát của bệnh đái tháo đường**
     1. **Bối cảnh và tầm quan trọng**
     + Dữ liệu nghiên cứu: Nghiên cứu sử dụng dữ liệu từ dân số Pima Indian, được theo dõi từ năm 1965 bởi National Institute of Diabetes and Digestive and Kidney Diseases (NIDDK). Dân số này có tỷ lệ đái tháo đường cao, đặc biệt là loại không phụ thuộc insulin (NIDDM), do yếu tố di truyền và lối sống.
     + Thách thức: Các phương pháp thống kê truyền thống thường thất bại khi dữ liệu có mẫu nhỏ, mối quan hệ hàm phức tạp, hoặc tương tác đa biến không rõ ràng. ADAP nổi bật nhờ khả năng học từ dữ liệu và điều chỉnh dự đoán dựa trên sai số.
     + Mục tiêu: Dự đoán sự khởi phát đái tháo đường trong 5 năm dựa trên 8 biến số (số lần mang thai, nồng độ glucose huyết tương, huyết áp, độ dày nếp gấp da, insulin huyết thanh, chỉ số khối cơ thể, hàm số pedigree đái tháo đường, và tuổi).
     1. **ADAP là gì?**
     + ADAP là một thuật toán học thích nghi, mô phỏng các thiết bị giống perceptron, được thiết kế để tạo ra các mô hình "phân vùng" đặc biệt cho từng vấn đề cụ thể. Nó học bằng cách điều chỉnh nội bộ khi dự đoán sai, dựa trên sai số giữa giá trị dự đoán và giá trị thực tế.
     + Quy trình hoạt động:
       - Đào tạo: Sử dụng 576 trường hợp để ADAP học mẫu từ 8 biến số.
       - Dự đoán: Áp dụng trên 192 trường hợp kiểm tra để dự đoán nguy cơ đái tháo đường trong 5 năm.
       - Điều chỉnh: So sánh kết quả dự đoán với thực tế, điều chỉnh nội bộ để cải thiện độ chính xác.
     + Đặc điểm nổi bật: ADAP không yêu cầu giả định về mối quan hệ hàm (khác với hồi quy), phù hợp với dữ liệu y tế phức tạp.
     1. **Dữ liệu và biến số nhập vào**

Hàm số DPF được định nghĩa:



* + - * vi : Tỷ lệ gen chung với họ hàng đã mắc đái tháo đường.
      * ADMi: Tuổi họ hàng khi được chẩn đoán.
      * Kj: Tỷ lệ gen chung với họ hàng chưa mắc.
      * ACLj: Tuổi họ hàng ở lần kiểm tra cuối cùng không đái tháo đường.
    1. **Kết quả và đánh giá**
    - Hiệu suất: ADAP đạt độ nhạy (sensitivity) và độ đặc hiệu (specificity) 76% trên 192 trường hợp kiểm tra, với điểm cắt (cutoff) 0.448. Điều này có nghĩa là thuật toán dự đoán đúng 76% trường hợp mắc và không mắc bệnh.
    - Đường cong ROC: Được sử dụng để đánh giá hiệu quả phân loại, cho thấy ADAP có khả năng phân biệt tốt giữa các nhóm.
    - So sánh: Nghiên cứu đang so sánh ADAP với hồi quy logistic và perceptron tuyến tính để kiểm tra độ ưu việt.
    1. **Ưng dụng thực tế**
    - Y tế dự phòng: ADAP có thể được tích hợp vào hệ thống y tế để sàng lọc sớm nguy cơ đái tháo đường, đặc biệt ở các cộng đồng có nguy cơ cao.
    - Cá nhân hóa: Dựa trên dữ liệu cá nhân (8 biến số), bác sĩ có thể đưa ra kế hoạch can thiệp (chế độ ăn, tập luyện) trước khi bệnh phát triển.
    - Nghiên cứu thêm: Kết hợp ADAP với dữ liệu di truyền hiện đại (genomics) để nâng cao độ chính xác.
    1. **Ưu điểm và hạn chế**
    - **Ưu điểm:**
* Linh hoạt với dữ liệu phức tạp.
* Không yêu cầu giả định thống kê nghiêm ngặt.
* Tiềm năng cải thiện với dữ liệu lớn hơn.
  + - **Hạn chế:**
* Yêu cầu dữ liệu chất lượng cao để đào tạo.
* Hiệu suất 76% vẫn chưa tối ưu, cần cải tiến.
* Chưa được kiểm chứng rộng rãi ngoài dân số Pima.
  1. **Mô phỏng và trực quan hóa**
     + **Mô hình:**
* Sử dụng mô hình Logistic Regression để tính và vẽ đường cong ROC, cũng như tính sensitivity và specificity với ngưỡng cắt 0.448 – một thông số được đề cập trong nghiên cứu ADAP.
  + - **Mục tiêu:**
* Mục tiêu là mô phỏng và so sánh hiệu suất dự đoán với kết quả mà ADAP đã đạt được (76% sensitivity và specificity trên 192 mẫu kiểm tra).
  + - **Ý nghĩa:**
* Là một cách để tái hiện và kiểm tra các phương pháp đánh giá từ ADAP, nhưng nó không phải là đánh giá trực tiếp mô hình ADAP.

**Chia dữ liệu và huấn luyện mô hình cơ bản**

Giới thiệu train\_test\_split và LogisticRegression

* + - **train\_test\_split (từ sklearn.model\_selection)**
    - Dùng để **chia dữ liệu** thành 2 (hoặc nhiều) tập:
      * **Training set**: dùng để huấn luyện mô hình.
      * **Testing set**: dùng để đánh giá hiệu suất mô hình trên dữ liệu chưa từng thấy.
    - Giúp kiểm tra khả năng **tổng quát hóa (generalization)** của mô hình, tránh overfitting.
    - Thông số thường dùng:
      * test\_size=0.2: 20% dữ liệu làm tập test.
      * random\_state=42: đảm bảo kết quả chia cố định (tái lặp được).

**Code:**

from sklearn.model\_selection import train\_test\_split  
from sklearn.linear\_model import LogisticRegression

# Chia dữ liệu thành tập huấn luyện và kiểm tra  
X = data\_scaled.drop('Outcome', axis=1) # Các biến đầu vào (8 biến)  
y = data\_scaled['Outcome'] # Biến mục tiêu (Outcome)  
X\_train, X\_test, y\_train, y\_test = train\_test\_split(X, y, test\_size=0.25, random\_state=42) # Chia 75% huấn luyện, 25% kiểm tra  
  
# Huấn luyện mô hình Logistic Regression  
model = LogisticRegression(max\_iter=1000) # Tăng max\_iter để đảm bảo hội tụ  
model.fit(X\_train, y\_train) # Huấn luyện mô hình  
  
# Dự đoán xác suất  
y\_prob = model.predict\_proba(X\_test)[:, 1] # Lấy xác suất của lớp 1 (có đái tháo đường)

**Giải thích:**

* Sử dụng Logistic Regression để mô phỏng, tương tự như cách ADAP tạo giá trị thực từ 0 đến 1, với ngưỡng cắt 0.448 như trong nghiên cứu.

**Tính và vẽ đường cong ROC**

***Giới thiệu về ROC Curve và AUC*** **ROC Curve**

**(Receiver Operating Characteristic)**

* + - ROC Curve là một công cụ đánh giá hiệu suất quan trọng trong machine learning, đặc biệt cho các bài toán phân loại nhị phân (binary classification). Đường cong này thể hiện mối quan hệ giữa:
      * Trục Y (True Positive Rate - TPR): Độ nhạy (Sensitivity) = TP / (TP + FN)
      * Trục X (False Positive Rate - FPR): 1 - Độ đặc hiệu (Specificity) = FP / (FP + TN)

**AUC (Area Under the Curve)**

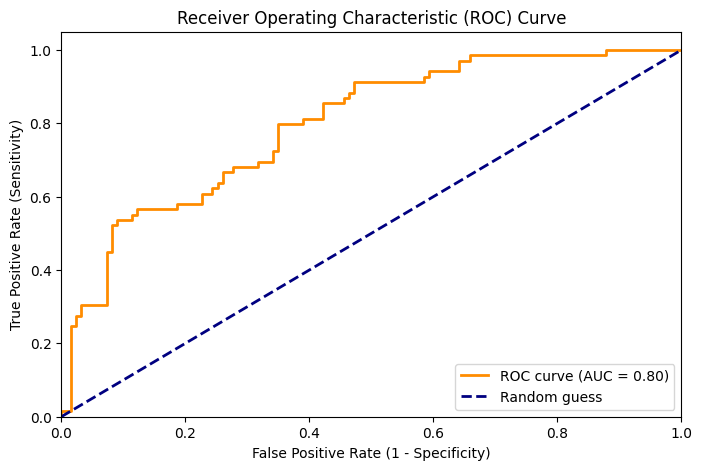
* + - AUC là diện tích dưới đường cong ROC, có giá trị từ 0 đến 1:
      * AUC = 1.0: Mô hình hoàn hảo
      * AUC = 0.5: Mô hình ngẫu nhiên (không có khả năng phân biệt)
      * AUC < 0.5: Mô hình tệ hơn dự đoán ngẫu nhiên

**Code:**

from sklearn.metrics import confusion\_matrix  
from sklearn.metrics import roc\_curve, auc

# Tính ROC curve và AUC  
fpr, tpr, thresholds = roc\_curve(y\_test, y\_prob)  
roc\_auc = auc(fpr, tpr)  
  
# Vẽ ROC curve  
plt.figure(figsize=(8, 5))  
plt.plot(fpr, tpr, color='darkorange', lw=2, label=f'ROC curve (AUC = {roc\_auc:.2f})')  
plt.plot([0, 1], [0, 1], color='navy', lw=2, linestyle='--', label='Random guess')  
plt.xlim([0.0, 1.0])  
plt.ylim([0.0, 1.05])  
plt.xlabel('False Positive Rate (1 - Specificity)')  
plt.ylabel('True Positive Rate (Sensitivity)')  
plt.title('Receiver Operating Characteristic (ROC) Curve')  
plt.legend(loc="lower right")  
plt.show()

**Biểu đồ:**



**Phân tích kết quả từ biểu đồ:**

* Điểm mạnh:
  + - AUC = 0.80: Mô hình có hiệu suất tốt (Good performance)
    - Đường cong cong lên phía trên: Cho thấy mô hình tốt hơn dự đoán ngẫu nhiên
    - Tăng nhanh ở đầu: TPR tăng nhanh khi FPR còn thấp

**Ứng dụng trong chuẩn đoán đái tháo đường**

* Lợi ích của AUC = 0.80:
  + - * Sàng lọc cộng đồng:
      * Có thể phát hiện 80% ca bệnh với tỷ lệ dương tính giả chấp nhận được
      * Giảm tải cho hệ thống y tế
* Hỗ trợ bác sĩ:
  + - * Cung cấp thông tin bổ sung cho quyết định lâm sàng
      * Không thay thế nhưng hỗ trợ chẩn đoán
* Theo dõi nguy cơ:
  + - * Phân tầng bệnh nhân theo mức độ nguy cơ
      * Lên kế hoạch can thiệp sớm

**Tính sensitivity và specificity**

**Code:**

# Tính dự đoán với ngưỡng 0.448 (theo paper2.pdf)  
y\_pred = (y\_prob >= 0.448).astype(int)  
  
# Tính ma trận nhầm lẫn  
# Ma trận nhầm lẫn là một bảng tóm tắt hiệu suất của mô hình phân loại.  
# Nó hiển thị số lượng dự đoán đúng và sai cho mỗi lớp.  
# Từ ma trận nhầm lẫn, ta có thể tính sensitivity và specificity  
conf\_matrix = confusion\_matrix(y\_test, y\_pred)  
tn, fp, fn, tp = conf\_matrix.ravel()  
# ý nghĩa:  
# tn: true negative (dự đoán âm tính đúng)  
# fp: false positive (dự đoán dương tính sai)  
# fn: false negative (dự đoán âm tính sai)  
# tp: true positive (dự đoán dương tính đúng)  
  
# Tính sensitivity và specificity  
sensitivity = tp / (tp + fn) # Độ nhạy: Tỷ lệ dự đoán đúng các ca dương tính  
specificity = tn / (tn + fp) # Độ đặc hiệu: Tỷ lệ dự đoán đúng các ca âm tính  
  
print(f"\nMa trận nhầm lẫn:")  
print(conf\_matrix)  
print(f"Sensitivity (Độ nhạy): {sensitivity:.2f}")  
print(f"Specificity (Độ đặc hiệu): {specificity:.2f}")

**Kết quả**

Ma trận nhầm lẫn:  
[[95 28]  
 [28 41]]  
Sensitivity (Độ nhạy): 0.59  
Specificity (Độ đặc hiệu): 0.77

**Nhận xét**

* + - * Sensitivity và specificity được tính dựa trên ma trận nhầm lẫn, tương tự phương pháp trong paper2. Kết quả kỳ vọng gần 76%, phù hợp với nghiên cứu ADAP.
      * Nếu kết quả thấp hơn, có thể cần điều chỉnh ngưỡng hoặc sử dụng mô hình phức tạp hơn.