1."Xin chào, chào mừng các bạn đến với bài học về dự đoán bệnh tiểu đường dựa trên các chỉ số chẩn đoán. Hôm nay, chúng ta sẽ dự đoán bệnh tiểu đường dựa trên các chỉ số chẩn đoán của bệnh nhân. Bộ dữ liệu này ban đầu được lấy từ Viện Quốc gia về bệnh tiểu đường, bệnh tiêu hóa và bệnh thận (NIDDK). NIDDK tiến hành và hỗ trợ nghiên cứu về nhiều căn bệnh phổ biến, tốn kém và mãn tính nhằm cải thiện sức khỏe. Một trong sáu người mắc bệnh tiểu đường trên thế giới là người Ấn Độ. Con số này đưa Ấn Độ vào danh sách 10 quốc gia hàng đầu có số người mắc bệnh tiểu đường, và Ấn Độ đứng thứ hai với khoảng 77 triệu người mắc bệnh tiểu đường. Trung Quốc đứng đầu với hơn 160 triệu người mắc bệnh tiểu đường.

Mục tiêu của bộ dữ liệu là dự đoán liệu một bệnh nhân có mắc bệnh tiểu đường hay không. Chúng ta dự đoán dựa trên các chỉ số chẩn đoán có trong bộ dữ liệu. Một số tiêu chí đã được đặt ra để lựa chọn các mẫu từ một cơ sở dữ liệu lớn. Đặc biệt, tất cả bệnh nhân trong bộ dữ liệu này đều là nữ, ít nhất 21 tuổi và thuộc dân tộc Pima Ấn Độ. Bộ dữ liệu bao gồm nhiều biến số y tế dự đoán và một biến mục tiêu là kết quả. Các biến số dự đoán bao gồm số lần mang thai của bệnh nhân, chỉ số BMI, mức insulin, tuổi, huyết áp, và các chỉ số khác.

Bạn có thể xây dựng một mô hình học máy để dự đoán chính xác liệu bệnh nhân có mắc bệnh tiểu đường hay không? Có, bằng cách sử dụng các hồ sơ y tế này, chúng ta sẽ cố gắng xây dựng một mô hình học máy để dự đoán liệu một người có mắc bệnh tiểu đường hay không."

2. "Xin chào, chào mừng bạn đến với bài học về việc nhập thư viện và bộ dữ liệu. Hãy bắt đầu xây dựng mô hình học máy. Tôi sẽ cung cấp cho bạn liên kết đến thư mục này, bạn có thể tải thư mục này dưới dạng tệp zip. Bạn có thể sử dụng đường dẫn tệp này để nhập bộ dữ liệu và bắt đầu làm việc trên đó. Hãy mở Jupyter Notebook của chúng ta và bắt đầu giải quyết vấn đề học máy này.

Bộ dữ liệu này chứa nhiều biến số dự đoán y tế và một biến mục tiêu. Hãy xem mô tả. Mục tiêu là xây dựng một mô hình học máy để dự đoán liệu bệnh nhân có mắc bệnh tiểu đường hay không. Mô hình học máy là gì? Một mô hình trong học máy là kết quả của một thuật toán học máy chạy trên dữ liệu. Một mô hình đại diện cho những gì mà thuật toán học máy đã học được.

Đầu tiên, chúng ta có cột "pregnancies", là số lần một người (nữ) mang thai trong đời. Tiếp theo, chúng ta sẽ kiểm tra nồng độ glucose trong huyết tương, sau đó là huyết áp tâm trương, được tính bằng mm. Tiếp theo, chúng ta sẽ kiểm tra độ dày lớp da của cơ tam đầu, được tính bằng mm, rồi đến mức insulin trong huyết thanh sau hai giờ. Tiếp theo, chúng ta sẽ kiểm tra chỉ số khối cơ thể (BMI), được tính bằng kg/m². Sau đó, chúng ta sẽ kiểm tra chức năng phả hệ tiểu đường, và cuối cùng là tuổi của bệnh nhân tính theo năm.

Cuối cùng, chúng ta sẽ kiểm tra biến kết quả, cho biết liệu người đó có mắc bệnh tiểu đường hay không. Trong biến kết quả, chúng ta có hai giá trị là 0 và 1, với 1 nghĩa là "có" và 0 nghĩa là "không". Có 268 người trong tổng số 768 người có mắc bệnh tiểu đường (kết quả là 1), và các trường hợp còn lại là 0. Chúng ta không thể phân loại sai các mục. Giả sử nếu một người mắc bệnh tiểu đường mà bạn dự đoán rằng người đó không mắc, thì người đó sẽ không uống thuốc và sẽ gặp nguy hiểm. Vì vậy, chúng ta không thể mạo hiểm, vì đây là một dự án thuộc lĩnh vực y tế.

Do là một dự án trong lĩnh vực y tế, chúng ta cần một bộ dữ liệu lớn. Tuy nhiên, ở đây bộ dữ liệu khá nhỏ, nên chúng ta không thể có kết quả chính xác hoàn toàn, nhưng chúng ta sẽ cố gắng đạt được kết quả dự đoán tốt trong mô hình này.

Bước tiếp theo là nhập tất cả các thư viện cần thiết. Hãy nhập tất cả các thư viện cần thiết. Chúng ta nhập `numpy` dưới dạng `np`, `pandas` dưới dạng `pd`. Chúng ta sử dụng các gói `numpy` và `pandas` cho việc phân tích dữ liệu và làm sạch dữ liệu. Chúng ta nhập `matplotlib.pyplot` dưới dạng `plt`, `seaborn` dưới dạng `sns`. Chúng ta sử dụng `matplotlib` và `seaborn` cho việc trực quan hóa dữ liệu. Chúng ta nhập hàm `plot\_decision\_regions`, đây là hàm dùng để vẽ các vùng quyết định của các bộ phân loại trong một hoặc hai chiều. Chúng ta nhập `missingno` dưới dạng `msno` để trực quan hóa các giá trị thiếu và giá trị `NaN` bằng thư viện `missingno`.

Chúng ta nhập nhiều thư viện khác từ `sklearn`. `Scikit-learn` là một gói học máy. Chúng ta cũng nhập `GridSearchCV`, đây là công cụ dùng để tìm các siêu tham số tối ưu của một mô hình nhằm cho kết quả dự đoán chính xác nhất. Chúng ta nhập `classification\_report`, `confusion\_matrix`, và `KNeighborsClassifier`. `KNeighborsClassifier` là một thuật toán phân loại đơn giản. Chúng ta sử dụng báo cáo phân loại (`classification\_report`) để so sánh các mô hình phân loại, và ma trận nhầm lẫn (`confusion\_matrix`) để đánh giá hiệu suất của một mô hình.

Hãy chạy lại ô mã này để nhập tất cả các thư viện cần thiết. Tôi nghĩ bạn đã hoàn thành khóa học về học máy nên bạn có thể đã biết tất cả các thuật ngữ được sử dụng ở đây. Hãy chạy ô mã này. Bây giờ, chúng ta sẽ nhập bộ dữ liệu. Tôi sẽ chỉ ra đường dẫn của tệp trên máy tính. Đây, bạn có thể thấy vị trí của tệp CSV của bộ dữ liệu. Hãy sao chép đường dẫn này và dán vào Jupyter Notebook.

Bây giờ chúng ta cần nhập bộ dữ liệu. Chúng ta sử dụng phương thức `read\_csv` để tải bộ dữ liệu. Sau đó, chúng ta sẽ chạy ô mã. Chúng ta đã nhập bộ dữ liệu và lưu nó trong một khung dữ liệu (`DataFrame`) của `pandas` tên là `diabetes\_df`. Hãy kiểm tra 5 dòng đầu tiên của khung dữ liệu này. Chúng ta đã lưu bộ dữ liệu trong `diabetes\_df`. Dùng phương thức `.head()` để kiểm tra 5 mẫu đầu tiên từ khung dữ liệu.

Ở đây, chúng ta có các cột: "pregnancies", "glucose", "blood pressure", "skin thickness", "insulin", "BMI", "diabetes pedigree function", "age", và cuối cùng là "outcome". Ở đây, giá trị "1" có nghĩa là "có" và giá trị "0" có nghĩa là "không"."

3. "Chào mừng đến với bài học về khám phá tập dữ liệu. Hãy tiếp tục. Bây giờ chúng ta sẽ kiểm tra tất cả tên cột trong tập dữ liệu. Để kiểm tra tên các cột của khung dữ liệu `diabetes\_DF`, chúng ta sử dụng phương thức `columns`. Bây giờ, khung dữ liệu `diabetes\_DF.columns`. Chạy ô lệnh này. Chúng ta có thể thấy danh sách tất cả tên cột của khung dữ liệu. Tập dữ liệu có tổng cộng chín cột.

Tiếp theo, chúng ta sẽ kiểm tra thêm thông tin về khung dữ liệu bằng cách sử dụng phương thức `info`. Chúng ta sử dụng phương thức `info` để kiểm tra chỉ mục, kiểu dữ liệu, các cột, giá trị không rỗng và dung lượng bộ nhớ. Chúng ta sẽ sử dụng phương thức `info`. Có tổng cộng 768 mục từ 0 đến 767. Ở đây chúng ta có thể thấy rằng các cột đều ở dạng số nguyên và số thực. Dung lượng bộ nhớ của dữ liệu là 54 kilobytes. Chúng ta không có bất kỳ giá trị chuỗi nào.

Hãy kiểm tra giá trị trung bình, trung vị và trung bình cho tất cả các cột này. Chúng ta sử dụng phương thức `describe` của pandas để xem một số chi tiết thống kê cơ bản như phần trăm, trung bình, độ lệch chuẩn, v.v. Chạy ô lệnh này. Chúng ta có cột `pregnancies`, số lần một người mang thai tối đa là 17 lần, có nghĩa là một phụ nữ mang thai 17 lần và là bệnh nhân tiểu đường. Chúng ta thấy số lần tối thiểu là 0. Sau đó, có 75% phụ nữ mang thai khoảng 6 lần. Tiếp theo, chúng ta thấy giá trị glucose, rồi đến huyết áp. Huyết áp tối đa là 122. Ở đây chúng ta thấy huyết áp tối thiểu là 0. Độ dày da tối đa là 99 và tối thiểu là 0. Giá trị insulin tối đa là 846 và tối thiểu là 0. Ở đây chúng ta có độ tuổi, với tuổi cao nhất là 81 và thấp nhất là 21.

Tôi hy vọng mọi người đều hiểu được tập dữ liệu. Hãy kiểm tra dạng chuyển vị của dữ liệu này. Theo cách này, chúng ta không thể thấy chi tiết về tất cả các cột, chẳng hạn như các đặc trưng `outcome` và `age`. Chúng ta sử dụng phương thức `transpose` để đổi hàng thành cột và cột thành hàng. Chúng ta có thể kiểm tra tất cả các cột này cùng một lúc bằng cách sử dụng phương thức này. Tiếp tục nào.

Bây giờ, hãy kiểm tra xem chúng ta có giá trị null nào không. Chúng ta không thấy các giá trị null hoặc bị thiếu. `False` chỉ ra không có giá trị bị thiếu, `True` chỉ ra có giá trị bị thiếu. Chúng ta không thấy bất kỳ giá trị null nào. Hãy kiểm tra tổng số giá trị null. Ở đây chúng ta không có bất kỳ giá trị null nào, vì vậy tất cả các cột đều hiển thị giá trị 0.

Bây giờ, hãy kiểm tra xem chúng ta có giá trị `NaN` nào không. Nếu có các giá trị 0 trong dữ liệu, chúng ta sẽ không thấy những giá trị đó là các giá trị bị thiếu. Trong trường hợp này, chúng ta thay thế các giá trị 0 bằng các giá trị `NaN`. Hãy kiểm tra xem chúng ta có giá trị `NaN` nào không. Chạy ô lệnh này. Ở đây, chúng ta có thể thấy rằng có 374 giá trị `NaN` trong cột `insulin`, 227 giá trị `NaN` trong cột `skin thickness`, 35 giá trị `NaN` trong cột `blood pressure`, và 11 giá trị `NaN` trong cột `BMI`.”