**ĐẠI HỌC CÔNG NGHIỆP HÀ NỘI**

**TRƯỜNG CÔNG NGHỆ THÔNG TIN VÀ TRUYỀN THÔNG**

**======\*\*\*======**

****

**BÁO CÁO THỰC NGHIỆM**

**HỌC PHẦN IT6094: TRÍ TUỆ NHÂN TẠO**

**ĐỀ TÀI**

**ỨNG DỤNG NAÏVE BAYES ĐỂ DỰ ĐOÁN BỆNH CÚM TỪ CÁC TRIỆU CHỨNG BAN ĐẦU**

|  |  |
| --- | --- |
| **GVHD:** | **Ths. Mai Thanh Hồng** |
| **Nhóm – Lớp:** | **09 –** 20251IT6094002 |
| **Thành viên:** | **Nguyễn Đức Mạnh - 2023602491** |
|  | **Nguyễn Đình Nguyên -** 2023602530 |
|  | **Đào Thúy Nhi - 2023602754** |
|  | **Ngô Tiến Quang -** 2023602047 |
|  |

**Hà Nội - Năm 2025**

**ĐẠI HỌC CÔNG NGHIỆP HÀ NỘI**

**TRƯỜNG CÔNG NGHỆ THÔNG TIN VÀ TRUYỀN THÔNG**

**======\*\*\*======**

****

**BÁO CÁO THỰC NGHIỆM**

**HỌC PHẦN IT6094: TRÍ TUỆ NHÂN TẠO**

**ĐỀ TÀI**

**ỨNG DỤNG NAÏVE BAYES ĐỂ DỰ ĐOÁN BỆNH CÚM TỪ CÁC TRIỆU CHỨNG BAN ĐẦU**

|  |  |
| --- | --- |
| **GVHD:** | **Ths. Mai Thanh Hồng** |
| **Nhóm – Lớp:** | **09 –** 20251IT6094002 |
| **Thành viên:** | **Nguyễn Đức Mạnh - 2023602491** |
|  | **Nguyễn Đình Nguyên -** 2023602530 |
|  | **Đào Thúy Nhi - 2023602754** |
|  | **Ngô Tiến Quang -** 2023602047 |
|  |

**Hà Nội - Năm 2025**

# LỜI CẢM ƠN

Trước tiên, chúng em xin bày tỏ lòng biết ơn sâu sắc tới giảng viên Mai Thanh Hồng – người đã tận tình hướng dẫn, đồng hành và hỗ trợ chúng em trong suốt quá trình thực hiện bài tập lớn này. Cô không chỉ là người truyền đạt kiến thức một cách mạch lạc, dễ hiểu, mà còn là một người cô đầy tâm huyết, luôn sẵn sàng lắng nghe, chia sẻ và truyền cảm hứng để chúng em không ngừng nỗ lực học tập và hoàn thiện bản thân. Nhờ sự chỉ dẫn tận tụy của cô, chúng em đã tiếp thu được nhiều kiến thức quý báu và rèn luyện thêm tinh thần trách nhiệm trong công việc. Những góp ý chân thành từ cô đã giúp bài làm của nhóm chúng em hoàn thiện hơn rất nhiều. Trong quá trình nghiên cứu và làm bài tập lớn, do năng lực, kiến thức, trình độ bản thân chúng em còn hạn chế nên không tránh khỏi những thiếu sót, chúng em mong nhận được sự thông cảm và những góp ý từ cô.

Chúng em xin chân thành cảm ơn!

Nhóm sinh viên thực hiện

Nhóm 09

# MỤC LỤC

**[LỜI CẢM ƠN 1](#_Toc216523708)**

**[MỤC LỤC 2](#_Toc216523709)**

**[DANH MỤC HÌNH ẢNH 4](#_Toc216523710)**

**[DANH MỤC BẢNG BIỂU 5](#_Toc216523711)**

**[MỞ ĐẦU 6](#_Toc216523712)**

**[1. Lý do chọn đề tài 6](#_Toc216523713)**

**[2. Mục đích nghiên cứu 6](#_Toc216523714)**

**[3. Đối tượng và phạm vi nghiên cứu 6](#_Toc216523715)**

**[4. Phương pháp nghiên cứu 7](#_Toc216523716)**

**[5. Bố cục của báo cáo 7](#_Toc216523717)**

**[CHƯƠNG 1: CƠ SỞ LÝ THUYẾT 8](#_Toc216523718)**

**[1.1 Tổng quan về trí tuệ nhân tạo và học máy 8](#_Toc216523719)**

**[1.1.1. Khái niệm trí tuệ nhân tạo 8](#_Toc216523720)**

**[1.1.2. Tại sao cần trí tuệ nhân tạo 8](#_Toc216523721)**

**[1.1.3. Ứng dụng của trí tuệ nhân tạo 8](#_Toc216523722)**

**[1.1.4. Các lĩnh vực nghiên cứu chính của AI 9](#_Toc216523723)**

**[1.1.5. Lịch sử phát triển của trí tuệ nhân tạo 9](#_Toc216523724)**

**[1.2 Phân loại trong học máy 9](#_Toc216523725)**

**[1.2.1. Khái niệm học máy 9](#_Toc216523726)**

**[1.2.2. Ứng dụng của học máy 10](#_Toc216523727)**

**[1.2.3. Các bước cơ bản trong học máy 10](#_Toc216523728)**

**[1.2.4. Một số phương pháp học máy 13](#_Toc216523729)**

**[1.2.4.1 Học có giám sát (Supervised Learning) 13](#_Toc216523730)**

**[1.2.4.2 Học không giám sát (Unsupervised Learning) 15](#_Toc216523731)**

**[1.2.4.3 Học tăng cường (Reinforcement Learning) 15](#_Toc216523732)**

**[1.3 Tổng quan về thuật toán Naïve Bayes 16](#_Toc216523733)**

**[1.3.1 Cơ sở lý thuyết của định lý Bayes 16](#_Toc216523734)**

**[1.3.1.1 Một số khái niệm cơ bản về xác xuất 16](#_Toc216523735)**

**[1.3.1.2 Định lý Bayes 18](#_Toc216523736)**

**[1.3.2 Phân lớp Naïve Bayes 19](#_Toc216523737)**

**[1.3.3 Các loại mô hình Naïve Bayes 20](#_Toc216523738)**

**[1.3.3.1. Gaussian Naïve Bayes 20](#_Toc216523739)**

**[1.3.3.2. Multinomial Naïve Bayes 21](#_Toc216523740)**

**[1.3.3.3. Bernoulli Naïve Bayes 22](#_Toc216523741)**

**[1.4 Ưu điểm và hạn chế của thuật toán Naïve Bayes 23](#_Toc216523742)**

**[1.4.1 Ưu điểm 23](#_Toc216523743)**

**[1.4.2 Hạn chế 23](#_Toc216523744)**

**[1.5. Ứng dụng thực tế của Naïve Bayes trong lĩnh vực y tế 24](#_Toc216523745)**

**[CHƯƠNG 2: ỨNG DỤNG VÀ THỰC NGHIỆM NAÏVE BAYES TRONG CHẨN ĐOÁN BỆNH CÚM TỪ CÁC TRIỆU CHỨNG BAN ĐẦU 25](#_Toc216523746)**

**[2.1 Phát biểu bài toán 25](#_Toc216523747)**

**[2.1.1 Mô tả bài toán 25](#_Toc216523748)**

**[2.1.2 Xác định đầu vào đầu ra 25](#_Toc216523749)**

**[2.2 Cơ sở dữ liệu 26](#_Toc216523750)**

**[2.2.1 Thu thập dữ liệu 26](#_Toc216523751)**

**[2.2.2 Xử lý dữ liệu 27](#_Toc216523752)**

**[2.2.3 Cơ sở dữ liệu thu được 30](#_Toc216523753)**

**[2.3 Xây dựng mô hình Naïve Bayes 32](#_Toc216523754)**

**[2.3.1 Lựa chọn phương pháp 32](#_Toc216523755)**

**[2.3.2 Lựa chọn biến thể Naive bayes 32](#_Toc216523756)**

**[2.3.3 Cài đặt mô hình 33](#_Toc216523757)**

**[2.4 Kết quả thực nghiệm 37](#_Toc216523759)**

**[KẾT LUẬN VÀ ĐỊNH HƯỚNG PHÁT TRIỂN 40](#_Toc216523760)**

**[TÀI LIỆU THAM KHẢO 42](#_Toc216523761)**

# DANH MỤC HÌNH ẢNH

[Hình 1.1: Thu thập dữ liệu trong học máy 11](#_Toc216524093)

[Hình 1.2: Tiền xử lý dữ liệu 11](#_Toc216524094)

[Hình 1.3: Lựa chọn và huấn luyện mô hình 12](#_Toc216524095)

[Hình 1.4: Đánh giá mô hình 12](#_Toc216524096)

[Hình 1.5: Học có giám sát 13](#_Toc216524097)

[Hình 1.6: Ví dụ nhận dạng chữ viết tay 14](#_Toc216524098)

[Hình 1.7: Ví dụ dò tìm các khuôn mặt 14](#_Toc216524099)

[Hình 1.8: Dò không giám sát 15](#_Toc216524100)

[Hình 1.9: Học tăng cường 15](#_Toc216524101)

[Hình 1.10: Các ví dụ học tăng cường 16](#_Toc216524102)

[Hình 2.1: Bộ dữ liệu công khai “Disease Symptom and Patient Profile Dataset” 26](#_Toc216524103)

[Hình 2.2: Các thuộc tính cần thiết cho đề tài 27](#_Toc216524104)

[Hình 2.3: Đọc dữ liệu và kiếm tra các cột 33](#_Toc216524105)

[Hình 2.4: Chuyển các nhãn Yes/No sang dạng nhị phân 0/1 33](#_Toc216524106)

[Hình 2.5: Biến tuổi (age) được chuyển thành nhóm tuổi 34](#_Toc216524107)

[Hình 2.6: Tạo nhóm tuổi age\_group 34](#_Toc216524108)

[Hình 2.7: Xác định các cột đặc trưng 35](#_Toc216524109)

[Hình 2.8: Chia dữ liệu train/test 35](#_Toc216524110)

[Hình 2.9: Tạo pipeline và huấn luyện GausianNB 36](#_Toc216524111)

[Hình 2.10: Đánh giá và lưu mô hình 36](#_Toc216524112)

[Hình 2.11: Nhập dữ liệu từ người dùng và tạo DataFrame 37](#_Toc216524113)

[Hình 2.12: Dự đoán xác suất cho từng nhãn 37](#_Toc216524114)

[Hình 2.13: Dự đoán nhãn kết quả 37](#_Toc216524115)

[Hình 2.14: Giao diện 38](#_Toc216524116)

[Hình 2.15: Kết quả minh họa trên giao diện đã xây dựng 39](#_Toc216524117)

# DANH MỤC BẢNG BIỂU

[Bảng 2.1: Chuẩn hóa tên biến 28](#_Toc216524154)

[Bảng 2.2: Mã hóa dữ liệu biến nhị phân 28](#_Toc216524155)

[Bảng 2.3: Mã hóa dữ liệu biến Age 29](#_Toc216524156)

[Bảng 2.4: Một số mẫu kết quả dữ liệu thu được sau xử lý 31](#_Toc216524157)

# MỞ ĐẦU

## Lý do chọn đề tài

Trong quá trình học môn Trí tuệ nhân tạo, nhóm chúng tôi được yêu cầu vận dụng một thuật toán đã được tìm hiểu trong chương trình học để áp dụng vào một bài toán thực tế, từ đó hình thành đề tài cho báo cáo cuối kỳ. Sau khi xem xét nhiều hướng tiếp cận, nhóm nhận thấy thuật toán Naïve Bayes phù hợp cho các bài toán phân loại dựa trên dữ liệu rời rạc, đặc biệt là các triệu chứng y tế dạng nhị phân. Dữ liệu được chuẩn hóa và mã hóa số (0/1 cho các biến nhị phân, OneHotEncode cho các biến phân loại), giúp Naïve Bayes áp dụng hiệu quả. Đồng thời, bệnh cúm là một bệnh phổ biến với các dấu hiệu tương đối rõ ràng, rất thích hợp để xây dựng một mô hình dự đoán dựa vào xác suất. Do đó, nhóm lựa chọn đề tài “Ứng dụng Naïve Bayes để dự đoán bệnh cúm từ các triệu chứng ban đầu” nhằm vừa đáp ứng yêu cầu môn học, vừa mang tính thực tiễn cao.

## Mục đích nghiên cứu

Mục tiêu của đề tài là xây dựng một mô hình dự đoán khả năng mắc bệnh cúm dựa trên các triệu chứng cơ bản như sốt (fever), ho (cough), mệt mỏi (fatigue), khó thở (difficulty\_breathing), kết hợp với các thông tin cá nhân như độ tuổi và giới tính. Ngoài việc huấn luyện mô hình dự đoán, nhóm cũng hướng tới xây dựng một biểu mẫu nhập liệu (ứng dụng Streamlit) giúp người dùng điền các triệu chứng của mình và nhận kết quả phân loại Positive/Negative. Qua đó, đề tài giúp minh họa cách thuật toán Naïve Bayes hoạt động trên dữ liệu thực tế và cách áp dụng vào một hệ thống đơn giản hỗ trợ chẩn đoán ban đầu.

## Đối tượng và phạm vi nghiên cứu

Đối tượng nghiên cứu của đề tài là thuật toán Naïve Bayes và phương pháp phân lớp dựa trên xác suất có điều kiện. Biến nhị phân được mã hóa thành 0/1, biến phân loại được OneHotEncode, và tuổi được nhóm theo age\_group. Thuật toán Gaussian Naïve Bayes được sử dụng phù hợp với dữ liệu sau mã hóa. Phạm vi nghiên cứu giới hạn trong bài toán dự đoán bệnh cúm với hai nhãn Positive/Negative, dựa trên tập dữ liệu mô phỏng hoặc trích lọc từ “Disease Symptoms and Patient Profile”. Đề tài không mở rộng sang các bệnh truyền nhiễm khác, không so sánh với nhiều thuật toán học máy khác, và không đi sâu vào phân tích chuyên môn y khoa mà chỉ tập trung vào khía cạnh ứng dụng thuật toán trong học phần.

## Phương pháp nghiên cứu

Đề tài được thực hiện dựa trên kết hợp giữa nghiên cứu lý thuyết và thực nghiệm. Nhóm bắt đầu bằng việc tìm hiểu cơ sở lý thuyết về trí tuệ nhân tạo, phân loại trong học máy và thuật toán Naïve Bayes. Tiếp theo, nhóm tiến hành thu thập dữ liệu, mã hóa nhị phân, OneHotEncode các thuộc tính phân loại và nhóm tuổi theo age\_group, đồng thời chuẩn hóa các giá trị đầu vào. Bài toán được mô hình hóa dưới dạng phân loại nhị phân, áp dụng công thức xác suất của Naïve Bayes để tính toán khả năng mắc bệnh. Cuối cùng, nhóm tiến hành thử nghiệm mô hình trên dữ liệu và đánh giá mức độ chính xác của kết quả dự đoán, đồng thời xây dựng ứng dụng nhập liệu đơn giản bằng Streamlit để minh họa cách mô hình hoạt động.

## Bố cục của báo cáo

Báo cáo được chia thành các phần chính theo cấu trúc sau:

* Mở đầu: trình bày lý do chọn đề tài, mục đích nghiên cứu, đối tượng và phạm vi nghiên cứu, cùng phương pháp mà nhóm sử dụng để thực hiện đề tài.
* Chương 1.Cơ sở lý thuyết: cung cấp nền tảng liên quan đến trí tuệ nhân tạo, học máy, các phương pháp phân loại và cơ sở hoạt động của thuật toán Naïve Bayes.
* Chương 2.Ứng dụng và thực nghiệm Naïve Bayes trong chuẩn đoán bệnh cúm từ các triệu chứng ban đầu mô tả chi tiết bài toán chẩn đoán bệnh cúm, tập dữ liệu sử dụng, các bước tiền xử lý, quá trình mô hình hóa và kết quả thực nghiệm.
* Kết luận và định hướng phát triển: tổng hợp các kết quả đạt được, đồng thời đề xuất những hướng mở rộng trong tương lai.

# CHƯƠNG 1: CƠ SỞ LÝ THUYẾT

## Tổng quan về trí tuệ nhân tạo và học máy

### 1.1.1. Khái niệm trí tuệ nhân tạo

Trí tuệ nhân tạo (Artificial Intelligence – AI) là ngành khoa học nghiên cứu các phương pháp nhằm giúp máy tính thực hiện được những hành vi yêu cầu trí tuệ của con người, chẳng hạn như suy luận, học hỏi từ dữ liệu, ra quyết định hoặc thích ứng với môi trường. Mục tiêu của AI là xây dựng các hệ thống có khả năng tự động xử lý thông tin, giải quyết vấn đề và đưa ra kết luận dựa trên tri thức đã học được.

AI không chỉ mô phỏng khả năng tư duy mà còn hướng đến việc tạo ra các tác nhân thông minh có thể tự hoạt động hợp lý trong các tình huống khác nhau. Như vậy, AI đóng vai trò quan trọng trong việc phát triển các công nghệ tự động hóa và hỗ trợ con người trong nhiều lĩnh vực.

### 1.1.2. Tại sao cần trí tuệ nhân tạo

AI trở nên cần thiết vì lượng dữ liệu mà con người tạo ra ngày càng lớn, vượt quá khả năng xử lý thủ công. Các hệ thống truyền thống gặp khó khăn trong việc phân tích dữ liệu phức tạp, không có cấu trúc hoặc thay đổi liên tục. AI giúp giải quyết vấn đề này thông qua khả năng học tập, phát hiện quy luật và dự đoán kết quả.

Bên cạnh đó, AI giúp tăng hiệu suất công việc, giảm chi phí vận hành và tạo ra các giải pháp tối ưu hơn so với phương pháp truyền thống. Trong nhiều lĩnh vực quan trọng như y tế, tài chính, giáo dục hay giao thông, AI mang lại độ chính xác và tốc độ xử lý vượt trội, góp phần nâng cao chất lượng dịch vụ và quyết định.

### 1.1.3. Ứng dụng của trí tuệ nhân tạo

AI được ứng dụng rộng rãi trong đời sống và sản xuất. Trong thị giác máy tính, AI cho phép hệ thống nhận diện hình ảnh, theo dõi đối tượng hoặc phân tích video trong thời gian thực. Đây là nền tảng cho các công nghệ như giám sát thông minh và xe tự lái.

Trong xử lý ngôn ngữ tự nhiên, AI giúp máy tính hiểu và tạo ra ngôn ngữ, từ đó hỗ trợ dịch tự động, trợ lý ảo và chatbot. Ngoài ra, AI còn được ứng dụng trong hệ thống gợi ý, tài chính, y tế, robot công nghiệp và nhiều lĩnh vực khác, thể hiện vai trò quan trọng trong quá trình chuyển đổi số và tự động hóa.

### 1.1.4. Các lĩnh vực nghiên cứu chính của AI

Trí tuệ nhân tạo bao gồm nhiều lĩnh vực nghiên cứu, trong đó học máy (Machine Learning) là trọng tâm. Học máy tập trung vào việc phát triển các thuật toán cho phép hệ thống học từ dữ liệu và cải thiện hiệu suất theo thời gian. Các thuật toán tìm kiếm và tối ưu hóa được sử dụng để giải bài toán trong các không gian trạng thái lớn.

Biểu diễn tri thức và suy luận logic là nhánh nghiên cứu về cách lưu trữ và xử lý thông tin dưới dạng các cấu trúc có ý nghĩa để hệ thống có thể đưa ra kết luận hợp lý. Ngoài ra, các lĩnh vực như lập kế hoạch, heuristic, mạng nơ-ron nhân tạo, học sâu (deep learning) và lập trình di truyền cũng đóng góp quan trọng vào sự phát triển của AI hiện đại.

### 1.1.5. Lịch sử phát triển của trí tuệ nhân tạo

AI bắt đầu hình thành từ mô hình nơ-ron nhân tạo của McCulloch và Pitts năm 1943. Đến năm 1956, hội nghị Dartmouth chính thức đặt tên “Artificial Intelligence”, đánh dấu sự ra đời của lĩnh vực này. Từ thập niên 1960, AI phát triển nhanh nhưng cũng trải qua giai đoạn suy thoái do hạn chế về công nghệ.

Từ cuối những năm 1970 đến 1980, các hệ chuyên gia như MYCIN và DENDRAL mang lại nhiều ứng dụng thực tế. Đến những năm 1990, các mô hình thống kê và thuật toán học máy hiện đại như mạng Bayes, SVM phát triển mạnh. Từ 2010 trở lại đây, sự xuất hiện của học sâu và các mô hình lớn đã đưa AI bước vào thời kỳ bùng nổ với nhiều thành tựu nổi bật như AlphaGo và các mô hình ngôn ngữ hiện đại.

## Phân loại trong học máy

### 1.2.1. Khái niệm học máy

Học máy (Machine Learning) là một lĩnh vực của khoa học máy tính cho phép hệ thống tự cải thiện hiệu suất thông qua kinh nghiệm thu được từ dữ liệu. Theo định nghĩa kinh điển của Tom Mitchell (1998), một chương trình máy tính được coi là học nếu hiệu suất thực hiện một nhiệm vụ cụ thể được nâng cao khi tích lũy thêm kinh nghiệm, được đo bằng một thước đo hiệu năng nhất định. Học máy là một nhánh quan trọng của trí tuệ nhân tạo, sử dụng các phương pháp thống kê để tìm ra quy luật tiềm ẩn trong dữ liệu. Trong đó, học sâu (Deep Learning) là một tập con của học máy, sử dụng mô hình mạng nơ-ron nhiều tầng để xử lý dữ liệu phức tạp như hình ảnh, giọng nói và ngôn ngữ tự nhiên. Sự ra đời của học máy giúp máy tính có khả năng tự động phân tích, suy luận và đưa ra quyết định trong nhiều bối cảnh mà trước đây chỉ con người mới có thể thực hiện.

### 1.2.2. Ứng dụng của học máy

Học máy đã trở thành công nghệ trọng tâm trong nhiều lĩnh vực hiện đại. Trong thương mại điện tử và mạng xã hội, ML được ứng dụng để xây dựng các hệ thống khuyến nghị, phân tích hành vi người dùng và đưa ra gợi ý nội dung cá nhân hóa.

Trong giao thông thông minh, mô hình ML thu thập dữ liệu từ camera, cảm biến, bản đồ để dự đoán lưu lượng xe theo từng thời điểm nhằm giúp giảm ùn tắc và nâng cao an toàn. Lĩnh vực tài chính sử dụng ML để phát hiện gian lận, phân tích rủi ro và dự báo thị trường. Trong y tế, ML hỗ trợ chuẩn đoán dựa trên hình ảnh và hồ sơ bệnh nhân. Ngoài ra, học máy còn được ứng dụng trong thị giác máy tính, xử lý tiếng nói, robot tự hành và tự động hóa doanh nghiệp, cho thấy vai trò ngày càng rộng lớn của nó trong cuộc sống hiện đại.

### 1.2.3. Các bước cơ bản trong học máy

**Bước 1: Xác định bài toán**

Việc xác định bài toán là bước đầu tiên và quan trọng nhất trong quy trình học máy. Ở giai đoạn này, người phân tích cần hiểu rõ vấn đề cần giải quyết, mục đích của mô hình và các khái niệm liên quan đến lĩnh vực dữ liệu. Quá trình xác định bài toán thường là quá trình lặp lại để làm rõ yêu cầu và chọn ra hướng tiếp cận phù hợp, đồng thời giúp xác định loại dữ liệu cần thu thập và nhóm kỹ thuật học máy sẽ được áp dụng.

**Bước 2: Thu thập dữ liệu**

Khi bài toán đã được xác định rõ, dữ liệu cần thiết sẽ được thu thập từ các nguồn phù hợp. Dữ liệu có thể lấy từ hệ thống nội bộ, kho lưu trữ, trình thu thập nhật ký, cảm biến, phương tiện ghi hình hoặc từ nguồn bên ngoài như web và API. Ngay cả khi dữ liệu đã có sẵn, người phân tích vẫn có thể cần thu thập thêm dữ liệu bổ sung để phục vụ mô hình học máy.



Hình 1.1: Thu thập dữ liệu trong học máy

**Bước 3: Tiền xử lý dữ liệu**

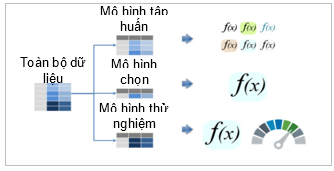


Hình 1.2: Tiền xử lý dữ liệu

Dữ liệu sau khi thu thập thường chưa thể sử dụng ngay mà cần phải được tiền xử lý. Đây là giai đoạn tốn nhiều thời gian nhất trong toàn bộ quy trình. Các bước tiền xử lý bao gồm làm sạch dữ liệu, chuẩn hóa, biến đổi dữ liệu, rời rạc hóa, lấy mẫu, và giảm chiều. Việc tiền xử lý tốt giúp dữ liệu trở nên phù hợp và làm tăng hiệu quả của mô hình học máy.

**Bước 4: Lựa chọn và huấn luyện mô hình**

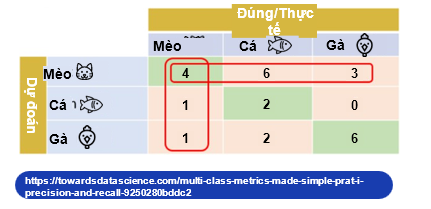
Sau khi dữ liệu được chuẩn hóa, bước tiếp theo là lựa chọn mô hình phù hợp và tiến hành huấn luyện. Trong học có giám sát, dữ liệu được chia thành tập huấn luyện và tập đánh giá. Mô hình sẽ học từ tập huấn luyện để xây dựng các quy tắc và mối quan hệ trong dữ liệu. Đối với học không giám sát, do dữ liệu không có nhãn, mô hình chủ yếu phân tích để tìm ra cấu trúc hoặc mẫu ẩn trong dữ liệu.



Hình 1.3: Lựa chọn và huấn luyện mô hình

**Bước 5: Đánh giá mô hình**

Mô hình sau khi huấn luyện cần được đánh giá bằng các bộ dữ liệu kiểm tra để kiểm tra mức độ chính xác và khả năng dự đoán. Trong học có giám sát, các chỉ số như độ chính xác và ma trận nhầm lẫn thường được sử dụng. Với học không giám sát, việc đánh giá dựa vào mức độ hợp lý và khả năng diễn giải của các cụm dữ liệu hoặc quy tắc được rút ra từ mô hình.



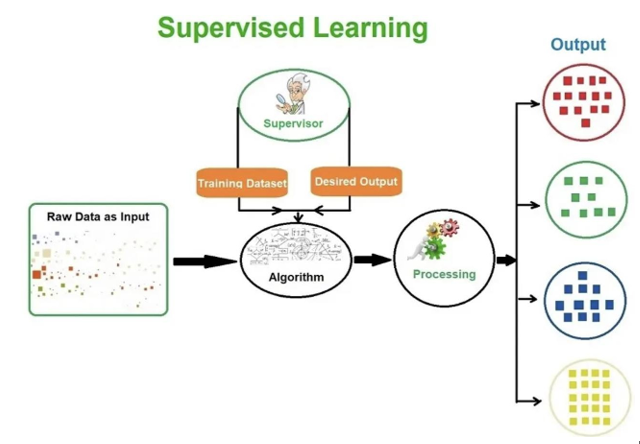
Hình 1.4: Đánh giá mô hình

**Bước 6: Cải thiện hoạt động và ứng dụng mô hình**

Mô hình thường không đạt hiệu quả tối ưu ngay từ lần thử đầu tiên, vì vậy cần điều chỉnh liên tục các tham số, thử các thuật toán khác hoặc thay đổi cách ước lượng để cải thiện hiệu suất. Khi mô hình đạt chất lượng mong muốn, nó có thể được triển khai vào thực tế để phục vụ doanh nghiệp hoặc hệ thống. Việc giám sát và cập nhật mô hình cũng cần được thực hiện thường xuyên nhằm duy trì hiệu quả lâu dài.

### 1.2.4. Một số phương pháp học máy

#### 1.2.4.1 Học có giám sát (Supervised Learning)



Hình 1.5: Học có giám sát

Học có giám sát là phương pháp trong đó dữ liệu đầu vào đi kèm với nhãn giúp mô hình học cách dự đoán giá trị đầu ra. Hai dạng chính của phương pháp này là phân loại và hồi quy. Phân loại được áp dụng khi nhãn được chia thành các nhóm hữu hạn, như phân loại email spam hay nhận dạng chữ số viết tay. Hồi quy được sử dụng khi nhãn là giá trị liên tục, như dự đoán giá nhà hoặc dự báo doanh thu. Phương pháp này là nền tảng của nhiều ứng dụng thực tế bởi khả năng tạo ra mô hình dự đoán mạnh và có kiểm soát.

**Ví dụ 1:** trong nhận dạng chữ viết tay, ta có ảnh của hàng nghìn ví dụ của mỗi chữ số được viết bởi nhiều người khác nhau. Chúng ta đưa các bức ảnh này vào trong một thuật toán và chỉ cho nó biết mỗi bức ảnh tương ứng với chữ số nào. Sau khi thuật toán tạo ra một mô hình, tức một hàm số mà đầu vào là một bức ảnh và đầu ra là một chữ số, khi nhận được một bức ảnh mới mà mô hình chưa nhìn thấy bao giờ, nó sẽ dự đoán bức ảnh đó chứa chữ



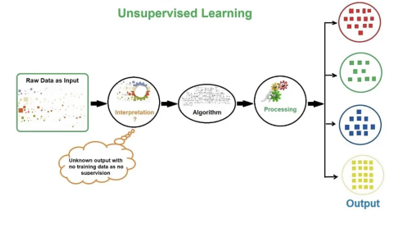
Hình 1.6: Ví dụ nhận dạng chữ viết tay

**Ví dụ 2:** Dò tìm các khuôn mặt trong 1 bức ảnh cũng là một thuật toán Supervised learning với training data (dữ liệu học) là hàng ngàn cặp (ảnh, mặt người) và (ảnh, không phải mặt người) được đưa vào.



Hình 1.7: Ví dụ dò tìm các khuôn mặt

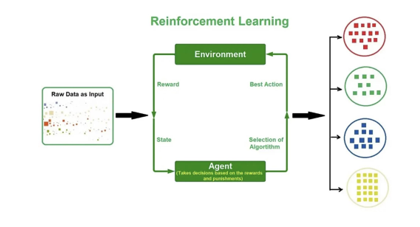
#### 1.2.4.2 Học không giám sát (Unsupervised Learning)



Hình 1.8: Dò không giám sát

Học không giám sát làm việc với dữ liệu không có nhãn, nhằm tìm ra cấu trúc ẩn hoặc mối quan hệ tự nhiên trong dữ liệu. Các thuật toán phổ biến gồm phân cụm và giảm chiều. Phân cụm giúp nhóm các đối tượng có đặc điểm tương đồng, thường được dùng trong phân khúc khách hàng. Giảm chiều cho phép thu gọn dữ liệu nhiều thuộc tính thành số thuộc tính ít hơn nhưng vẫn giữ được thông tin quan trọng, hỗ trợ trực quan hóa và giảm chi phí tính toán. Phương pháp này rất hữu ích khi xử lý dữ liệu lớn chưa có phân loại ban đầu.

#### 1.2.4.3 Học tăng cường (Reinforcement Learning)



Hình 1.9: Học tăng cường

Học tăng cường là phương pháp mô hình hóa quá trình ra quyết định của một tác nhân (agent) trong môi trường. Tác nhân thực hiện các hành động và nhận phần thưởng hoặc hình phạt, từ đó tự học chiến lược tối ưu để tối đa hóa tổng phần thưởng theo thời gian. Đây là phương pháp thường được sử dụng trong robot, các hệ thống điều khiển tự động và đặc biệt nổi tiếng với ứng dụng trong trò chơi chiến lược như AlphaGo, nơi mô hình có khả năng học cách đi từng nước để giành chiến thắng.



Hình 1.10: Các ví dụ học tăng cường

## 1.3 Tổng quan về thuật toán Naïve Bayes

Thuật toán Naïve Bayes là một trong những phương pháp phân loại đơn giản nhưng hiệu quả nhất trong lĩnh vực học máy. Thuật toán dựa trên định lý Bayes kết hợp với giả định độc lập có điều kiện giữa các thuộc tính, từ đó cho phép mô hình ước tính xác suất một dữ liệu mới thuộc về một lớp nào đó. Nhờ có nền tảng toán học rõ ràng, khả năng xử lý tốt dữ liệu lớn và tốc độ huấn luyện nhanh, Naïve Bayes được ứng dụng rộng rãi trong phân loại văn bản, lọc thư rác và đặc biệt hữu ích trong các bài toán dự đoán trong lĩnh vực y tế.

### 1.3.1 Cơ sở lý thuyết của định lý Bayes

Để hiểu rõ nguyên lý hoạt động của thuật toán Naïve Bayes, cần nắm được cơ sở toán học liên quan đến xác suất và định lý Bayes. Những kiến thức này giúp giải thích cách mô hình ước tính khả năng một mẫu dữ liệu thuộc về một lớp cụ thể dựa trên các thuộc tính quan sát được. Vì vậy, phần này trình bày một số khái niệm xác suất cơ bản và nội dung của định lý Bayes làm nền tảng cho các mục tiếp theo.

#### 1.3.1.1 Một số khái niệm cơ bản về xác xuất

* Hiện tượng tất yếu: là những hiện tượng nếu được thực hiện ở điều kiện giống nhau thì kết quả giống nhau. Chẳng hạn khi đun nước đến 1000C thì nước sôi. Hiện tượng tất yếu là đối tượng nghiên cứu của Vật lý, Hóa học.
* Hiện tượng ngẫu nhiên: là những hiện tượng dù đã được quan sát ở điều kiện giống nhau, nhưng kết quả có thể khác nhau. Ví dụ: tung đồng xu, và quan sát “Sấp” hay “Ngửa”. Hiện tượng ngẫu nhiên là đối tượng nghiên cứu của xác suất học.

Trong một hiện tượng ngẫu nhiên ta không thể biết được chắc chắn kết quả xảy ra như thế nào, nhưng có thể hình dung ra được các kết quả có thể xảy ra. Tập hợp các kết quả xảy ra được gọi là không gian mẫu, ký hiệu là Ω.

Ví dụ: Tung đồng xu, Ω = {Sấp, Ngửa}; tung con xúc sắc, Ω = {1, 2, 3, 4, 5, 6}

Biến cố: là một tập con của không gian mẫu, ký hiệu là: A, B, C . . .

Ví dụ: Tung con xúc sắc, gọi A là biến cố được số chẵn và B là biến cố được số lẻ, nên A = {2, 4, 6}, B = {1, 3, 5}.

Các biến cố là các tập con, nên ta dùng các phép tính trên tập hợp cho biến cố:

* Phần hội: A ∪ B (A hay B xảy ra).
* Phần giao: A ∩ B = AB (A và B xảy ra).
* Phần bù: A = Ω \ A (A không xảy ra).

Quan sát các hiện tượng ta thấy có những hiện tượng thường xảy ra, có những hiện tượng ít xảy ra. Xác suất là một con số đo lường mức độ xảy ra của một biến cố.

Xác suất của A, ký hiệu là P(A) là tỷ số giữa số phần tử của A và số phần tử của không gian mẫu Ω



1. **Tính chất cơ bản của xác suất:**

0 ≤ P(A) ≤ 1

P(True) = 1

P(False) = 0

P(A ∪ B) = P(A) + P(B) – P(A ∩ B)

1. **Xác suất có điều kiện:**

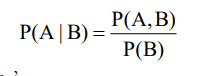
P(A | B) là phần của không gian mà trong đó A là đúng, với điều kiện (đã biết) là B đúng.

Ví dụ:

A: Tôi sẽ đi chơi vào ngày mai,

B: Trời sẽ không mưa vào ngày mai,

P(A | B): Xác suất của việc tôi sẽ đi chơi vào ngày mai nếu đã biết rằng trời sẽ không mưa vào ngày mai.

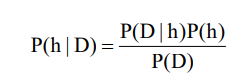


1. **Công thức xác suất toàn phần:**

Nếu B1 + B2 + …+ Bn = Ω và BiBj = ∅ ∀i ≠ j, khi đó với biến cố A liên quan được tính theo công thức:



#### 1.3.1.2 Định lý Bayes



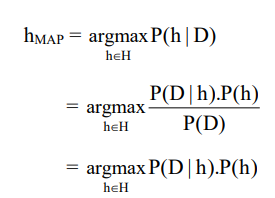
P(h): xác suất trước (tiên nghiệm) của giả thiết phân lớp h.

P(D): xác suất trước (tiên nghiệm) của việc quan sát được dữ liệu D.

P(D | h): xác suất có điều kiện của việc quan sát được dữ liệu D, nếu biết giả thiết phân lớp h là đúng.

P(h | D): xác suất có điều kiện của giả thiết phân lớp h là đúng, nếu quan sát được dữ liệu D.

Với H là một tập các giả thiết phân lớp, hệ thống học sẽ tìm giả thiết có thể xảy ra nhất để h ∈ H đối với các dữ liệu D quan sát được. Giả thiết h này được gọi là giả thiết có xác suất hậu nghiệm cực đại (Maximum a posteriori – MAP)



Giả thiết có thể nhất hMAP = h1 nếu P(h1 | D) ≥ P(h2 | D); ngược lại, hMAP = h2.

Do P(D) là như nhau đối với cả hai giả thiết h1 và h2, nên có thể bỏ qua đại lượng P(D). Vì vậy, chỉ cần tính hai biểu thức: P(D | h1).P(h1) và P(D | h2).P(h2), là ta có thể đưa ra quyết định phân lớp.

### 1.3.2 Phân lớp Naïve Bayes

Bài toán phân lớp (classification problem): một tập dữ liệu huấn luyện D\_train, trong đó mỗi mẫu x được biểu diễn là một vectơ n chiều (x1, x2, ..., xn) và một tập xác định các nhãn lớp C = {c1, c2,..., cm,}. Với mẫu mới z, thì z sẽ được phân vào lớp nào?

Mục tiêu: xác định phân lớp phù hợp nhất đối với z.



Giả sử trong phương pháp phân lớp Naïve Bayes, các thuộc tính là độc lập có điều kiện đối với các lớp:



Phân lớp Naïve Bayes tìm phân lớp có thể nhất đối với z



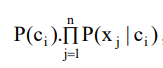
Giai đoạn huấn luyện (training phase):

* Đối với mỗi phân lớp có thể (mỗi nhãn lớp) ci ∈ C: tính giá trị xác suất trước P(ci)
* Đối với mỗi giá trị thuộc tính xj, tính giá trị xác suất xảy ra của giá trị thuộc tính đó đối với một phân lớp ci: P(xj | ci).

Giai đoạn phân lớp (classification phase): cần gán nhãn cho một mẫu mới,

thực hiện:

* Đối với mỗi phân lớp ci ∈C, tính giá trị của biểu thức:



* Xác định phân lớp của z là lớp có thể nhất c\* :



nghĩa là z được dự đoán thuộc lớp ci nếu và chỉ nếu

P(ci | z) > P(cj | z) với 1 ≤ j ≤ m, j ≠ i.

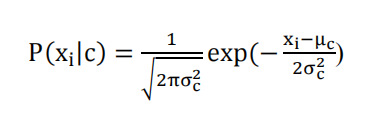
### 1.3.3 Các loại mô hình Naïve Bayes

#### 1.3.3.1. Gaussian Naïve Bayes

Gaussian Naive Bayes là một phiên bản của thuật toán Naive Bayes, được sử dụng

trong các bài toán phân loại khi các đặc trưng đầu vào là dữ liệu liên tục, chẳng hạn như chiều cao, cân nặng, nhiệt độ, hoặc các giá trị số khác. Thuật toán này kế thừa 10 nguyên lý cốt lõi của Naive Bayes, dựa trên định lý Bayes và giả định rằng các đặc trưng là độc lập với nhau (naive assumption). Điểm khác biệt chính của Gaussian Naive Bayes nằm ở việc nó giả định rằng xác suất của các đặc trưng trong mỗi lớp tuân theo phân phối chuẩn (Gaussian distribution).

Nguyên lý hoạt động: Trong Gaussian Naive Bayes, ta giả định rằng các đặc trưng xi (với i=1,2,...,n) theo phân phối chuẩn với giá trị trung bình μ và độ lệch chuẩn σ. Xác suất có điều kiện P(xi∣c) được tính theo công thức của phân phối chuẩn:



Trong đó:

* μc là giá trị trung bình của đặc trưng xi trong lớp c
* σc là độ lệch chuẩn tương ứng

#### 1.3.3.2. Multinomial Naïve Bayes

Mô hình này chủ yếu được sử dụng trong phân loại văn bản mà feature vectors

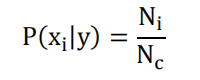
được tính bằng Bags of Words. Lúc này, mỗi văn bản được biểu diễn bởi một vector

có độ dài d chính là số từ trong từ điển. Giá trị của thành phần thứ i trong mỗi vector

chính là số lần từ thứ i xuất hiện trong văn bản đó.

Khi đó, p(xi|y) tỉ lệ với tần suất từ thứ i (hay feature thứ i cho trường hợp tổng quát)

xuất hiện trong các văn bản của class y. Giá trị này có thể được tính bằng cách:

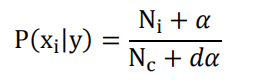


Trong đó:

* Ni là tổng số lần từ xi xuất hiện trong văn bản
* Nc là tổng số lần từ của tất cả các từ x1,x2,…,xn xuất hiện trong văn bản

Công thức trên có hạn chế là khi từ xi không xuất hiện lần nào trong văn bản, ta sẽ có Ni=0. Điều này làm cho P(xi|y)=0

Để khắc phục vấn đề này, người ta sử dụng kỹ thuật gọi là Laplace Smoothiing bằng cách cộng thêm vào cả tử và mẫu để giá trị luôn khác 0



Trong đó:

* α thường là số dương, bằng 1
* dα được cộng vào mẫu để đảm bảo ∑i=1dP(xi∣y)=1

#### 1.3.3.3. Bernoulli Naïve Bayes

Bernoulli Naïve Bayes là một biến thể của thuật toán Naïve Bayes, được sử dụng trong các bài toán phân loại mà dữ liệu đầu vào là các đặc trưng nhị phân (binary features).

Mỗi đặc trưng xi trong vector đặc trưng X=(x1,x2,...,xn)chỉ nhận hai giá trị:

* xi=1: đặc trưng xuất hiện.
* xi=0: đặc trưng không xuất hiện.

Khác với Multinomial Naïve Bayes (làm việc với tần suất từ), Bernoulli Naïve Bayes chỉ quan tâm xem một đặc trưng có xuất hiện hay không trong mẫu dữ liệu, nên rất phù hợp cho các tác vụ như lọc spam, xác định sự xuất hiện từ khóa, hoặc nhận diện chủ đề văn bản với vector hóa dữ liệu dạng Bag-of-Words nhị phân.

Nguyên lý hoạt động :Mô hình này được áp dụng cho các loại dữ liệu mà mỗi thành phần là một giá trị binary - bằng 0 hoặc 1. Ví dụ: cũng với loại văn bản nhưng thay vì đếm tổng số lần xuất hiện của 1 từ trong văn bản, ta chỉ cần quan tâm từ đó có xuất hiện hay không.

Khi đó, P(xi|c) được tính bằng:



Với p(i|c) có thể được hiểu là xác suất từ thứ I xuất hiện trong các văn bản của class c

## Ưu điểm và hạn chế của thuật toán Naïve Bayes

### 1.4.1 Ưu điểm

* Đơn giản, dễ hiểu và dễ triển khai: Thuật toán có cơ sở lý thuyết rõ ràng, các bước tính toán dựa trên công thức xác suất Bayes nên rất dễ cài đặt.
* Hiệu quả với dữ liệu lớn: Naïve Bayes hoạt động tốt ngay cả khi kích thước dữ liệu lớn, do việc tính toán xác suất tương đối nhanh và không đòi hỏi nhiều tài nguyên.
* Không yêu cầu nhiều dữ liệu huấn luyện: Mô hình đạt hiệu quả tốt ngay cả khi tập huấn luyện nhỏ, vì chỉ cần ước lượng các xác suất đơn giản.
* Xử lý tốt dữ liệu có nhiễu hoặc thiếu: Thuật toán có khả năng “chịu lỗi” khá tốt, do việc tính xác suất trung bình giúp giảm ảnh hưởng của dữ liệu thiếu.
* Hiệu quả cao trong bài toán phân loại văn bản: Đặc biệt phù hợp với các ứng dụng như lọc thư rác (spam filter), phân tích cảm xúc, và gán nhãn văn bản.
* Dễ mở rộng và kết hợp: Có thể kết hợp với các mô hình khác (như Decision Tree, SVM) để tạo ra các hệ thống lai (hybrid systems) nhằm cải thiện độ chính xác.

### 1.4.2 Hạn chế

* Giả định độc lập có điều kiện không thực tế: Naïve Bayes giả định rằng các đặc trưng độc lập nhau khi biết lớp — điều này hiếm khi đúng trong thực tế, vì các đặc trưng thường có mối liên hệ (ví dụ: nhiệt độ và độ ẩm trong dự đoán thời tiết).
* Khó biểu diễn mối quan hệ giữa các đặc trưng: Nếu hai đặc trưng có quan hệ mạnh với nhau, mô hình có thể đánh giá sai xác suất.
* Xác suất bằng 0 (Zero-frequency problem): Khi một giá trị đặc trưng chưa từng xuất hiện trong dữ liệu huấn luyện, xác suất có thể bằng 0. Cần kỹ thuật “Laplacian Smoothing” để khắc phục.
* Giới hạn với dữ liệu liên tục không chuẩn: Nếu dữ liệu liên tục không tuân theo phân phối Gaussian, Gaussian NB có thể cho kết quả sai lệch.
* Độ chính xác thấp hơn các mô hình phức tạp hơn: So với các thuật toán như Random Forest hoặc Neural Network, Naïve Bayes thường cho độ chính xác thấp hơn trong các bài toán phức tạp.

## 1.5. Ứng dụng thực tế của Naïve Bayes trong lĩnh vực y tế

Thuật toán Naïve Bayes được ứng dụng rộng rãi trong y học nhờ khả năng phân loại nhanh, đơn giản và giải thích được kết quả. Một số ứng dụng cụ thể gồm:

* Chẩn đoán bệnh dựa trên triệu chứng ban đầu: Mô hình phân tích các triệu chứng của bệnh nhân (sốt, ho, đau đầu, khó thở, v.v.) để dự đoán khả năng mắc bệnh như cúm, sốt xuất huyết, COVID-19,…
* Phân tích kết quả xét nghiệm: Naïve Bayes có thể dự đoán khả năng dương tính hoặc âm tính của một bệnh (như tiểu đường, ung thư) dựa trên dữ liệu xét nghiệm máu, nước tiểu hoặc hình ảnh y khoa.
* Phân loại hình ảnh y tế: Dùng trong xử lý ảnh chẩn đoán (CT, MRI, X-quang) để xác định khối u, tổn thương hoặc vùng bất thường trong cơ thể.
* Dự đoán nguy cơ mắc bệnh mãn tính: Sử dụng dữ liệu sức khỏe (tuổi, chỉ số BMI, huyết áp, đường huyết, thói quen sinh hoạt) để dự đoán nguy cơ mắc các bệnh như tim mạch, tiểu đường, béo phì.
* Hỗ trợ ra quyết định lâm sàng (Clinical Decision Support Systems - CDSS): Naïve Bayes giúp bác sĩ đưa ra chẩn đoán sơ bộ dựa trên dữ liệu bệnh án điện tử (EHR), tiết kiệm thời gian và nâng cao độ chính xác.
* Phân tích gen và sinh học phân tử: Trong nghiên cứu di truyền học, thuật toán được dùng để phân loại mẫu gen, dự đoán chức năng gen hoặc nhận dạng các mẫu ADN bất thường.

# CHƯƠNG 2: ỨNG DỤNG VÀ THỰC NGHIỆM NAÏVE BAYES TRONG CHẨN ĐOÁN BỆNH CÚM TỪ CÁC TRIỆU CHỨNG BAN ĐẦU

## 2.1 Phát biểu bài toán

### 2.1.1 Mô tả bài toán

Trong bối cảnh trợ giúp chẩn đoán sớm các bệnh truyền nhiễm, việc xây dựng một hệ thống hỗ trợ phát hiện bệnh cúm từ các triệu chứng ban đầu có ý nghĩa thiết thực. Bài toán được đặt ra như sau: Cho một bản ghi chứa các thông tin triệu chứng và hồ sơ cá nhân (ví dụ: sốt, ho, mệt mỏi, khó thở, tuổi, giới tính), nhiệm vụ là dự đoán **Outcome** là **Positive** (mắc cúm) hoặc **Negative** (không mắc cúm).

### 2.1.2 Xác định đầu vào đầu ra

1. **Đầu vào**

Bộ dữ liệu đầu vào gồm các thuộc tính sau (đã được chuẩn hóa tên cột và xử lý kiểu dữ liệu):

* fever (Sốt) – Dạng Yes/No, mã hóa thành 1/0
* cough (Ho) – Yes/No, mã hóa thành 1/0
* fatigue (Mệt mỏi) – Yes/No, mã hóa thành 1/0
* difficulty\_breathing (Khó thở) – Yes/No, mã hóa thành 1/0
* age (Tuổi) – Dạng số nguyên, được chuyển thành nhóm tuổi age\_group theo 4 nhóm. Nhóm tuổi này sẽ được OneHotEncode trong pipeline khi huấn luyện mô hình.
* gender (Giới tính) – Male/Female, được OneHotEncode thành các cột nhị phân.

Toàn bộ dữ liệu trên được dùng làm đầu vào cho mô hình Naïve Bayes trong quá trình huấn luyện và dự đoán.

1. **Đầu ra**

Đầu ra của bài toán là biến mục tiêu outcome, biểu thị kết quả chẩn đoán tình trạng bệnh cúm của bệnh nhân. Thuộc tính này nhận một trong hai giá trị:

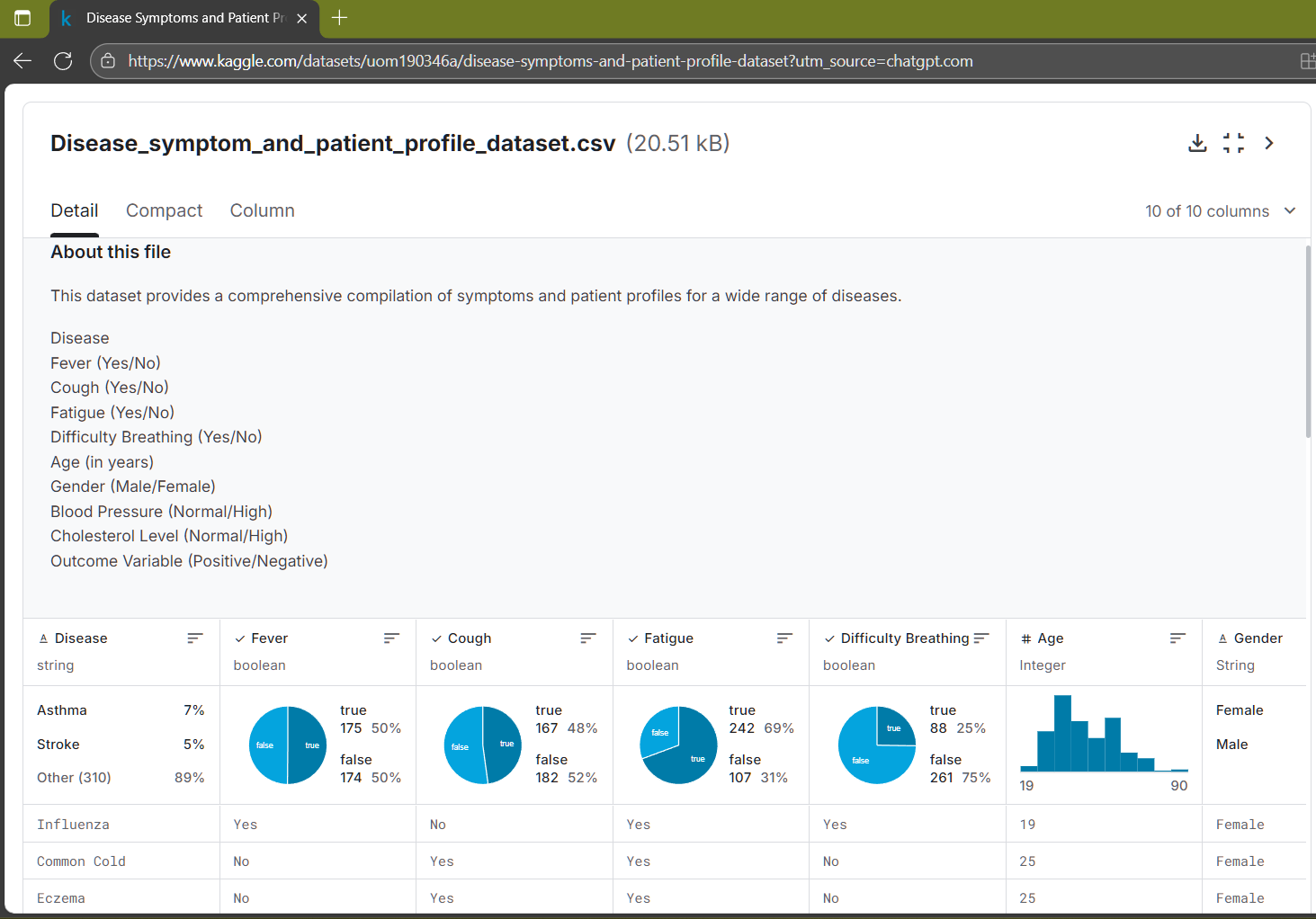
* Positive – bệnh nhân được dự đoán mắc cúm
* Negative – bệnh nhân được dự đoán không mắc cúm

Biến outcome đóng vai trò là nhãn phân loại mà mô hình Naïve Bayes cần học và dự đoán dựa trên các triệu chứng và thông tin cá nhân đầu vào. Trong quá trình huấn luyện, mô hình sử dụng biến này làm căn cứ để tính toán xác suất và xây dựng quy tắc phân loại.

## 2.2 Cơ sở dữ liệu

### 2.2.1 Thu thập dữ liệu

Trong đề tài này, nhóm sử dụng bộ dữ liệu công khai “Disease Symptom and Patient Profile Dataset” được thu thập từ nền tảng Kaggle. Bộ dữ liệu bao gồm các thông tin về triệu chứng của bệnh nhân và một số yếu tố sức khỏe liên quan.

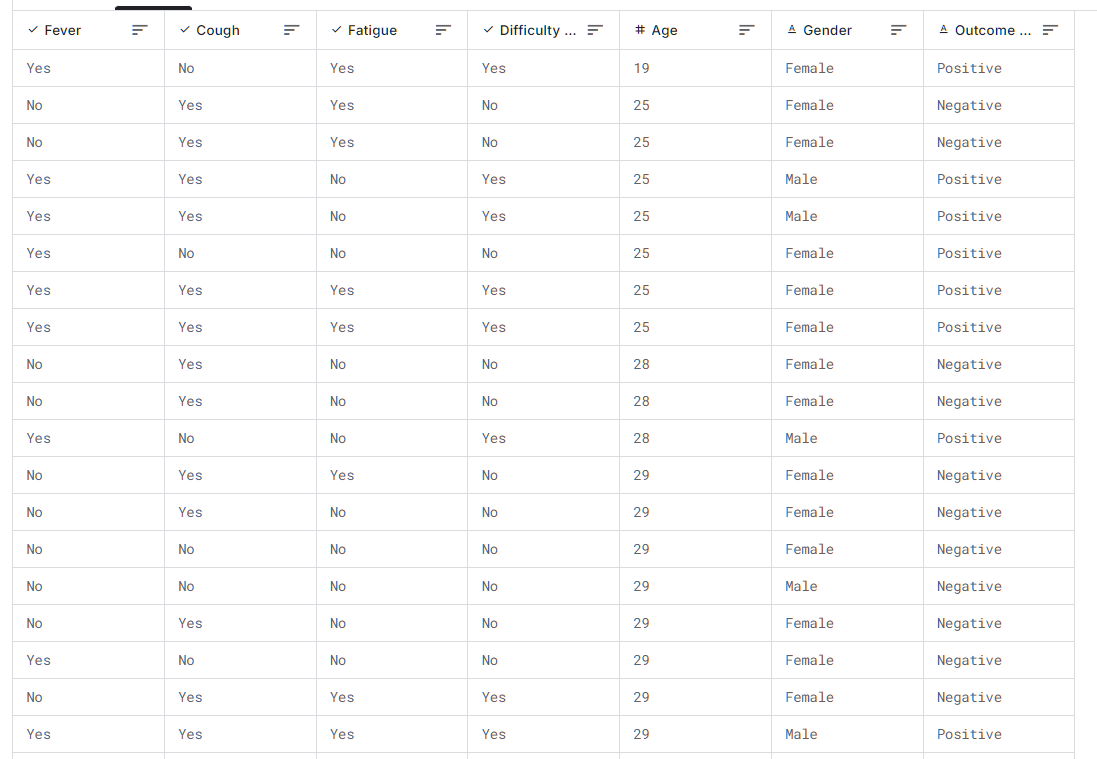


Hình 2.1: Bộ dữ liệu công khai “Disease Symptom and Patient Profile Dataset”

Dữ liệu gốc có tổng cộng 10 thuộc tính. Tuy nhiên, để phù hợp với mục tiêu xây dựng mô hình dự đoán nguy cơ mắc bệnh cúm dựa trên các triệu chứng ban đầu, nhóm tiến hành xem xét mức độ liên quan của các trường và chỉ lựa chọn 7 thuộc tính cần thiết nhất. Đây là những thuộc tính phản ánh trực tiếp tình trạng sức khỏe hiện tại của bệnh nhân.

7 thuộc tính được chọn gồm:

* Fever (Sốt – Yes/No)
* Cough (Ho – Yes/No)
* Fatigue (Mệt mỏi – Yes/No)
* Difficulty Breathing (Khó thở – Yes/No)
* Age (Tuổi – số nguyên)
* Gender (Giới tính – Male/Female)
* outcome\_variable(Positive/Negative)



Hình 2.2: Các thuộc tính cần thiết cho đề tài

Việc chọn lọc này giúp mô hình tập trung vào các biến có ý nghĩa nhất, tránh nhiễu dữ liệu và tăng hiệu quả huấn luyện.

### 2.2.2 Xử lý dữ liệu

Để chuẩn bị dữ liệu cho Naïve Bayes, nhóm tiến hành các bước xử lý chi tiết:

**Bước 1: Chuẩn hóa tên biến**

Dữ liệu gốc có tên cột dài, chứa dấu cách, khó lập trình. Nhóm thực hiện:

* Loại bỏ dấu cách, ký tự đặc biệt.
* Chuyển tên cột về chữ thường, dễ nhớ.

Giúp tránh lỗi khi lập trình, dễ truy cập dữ liệu và thống nhất trong toàn bộ báo cáo, mô hình.

Bảng 2.1: Chuẩn hóa tên biến

| **Tên gốc** | **Tên chuẩn hóa** | **Ghi chú** |
| --- | --- | --- |
| Fever | fever | Triệu chứng sốt |
| Cough | cough | Triệu chứng ho |
| Fatigue | fatigue | Triệu chứng mệt mỏi |
| Difficulty Breathing | difficulty\_breathing | Triệu chứng khó thở |
| Age | age | Tuổi bệnh nhân |
| Gender | gender | Giới tính bệnh nhân |
| Outcome Variable | outcome\_variable | Kết quả chẩn đoán bệnh |

**Bước 2: Mã hóa dữ liệu (Encoding)**

Naïve Bayes yêu cầu dữ liệu rời rạc hoặc numeric để tính xác suất. Nhóm tiến hành:

1. **Biến nhị phân**

Bảng 2.2: Mã hóa dữ liệu biến nhị phân

| **Biến** | **Giá trị gốc** | **Giá trị sau mã hóa** | **Giải thích** |
| --- | --- | --- | --- |
| fever | Yes / No | 1 / 0 | 1 = bệnh nhân sốt, 0 = không sốt |
| cough | Yes / No | 1 / 0 | 1 = bệnh nhân ho, 0 = không ho |
| fatigue | Yes / No | 1 / 0 | 1 = bệnh nhân mệt mỏi, 0 = không mệt |
| difficulty\_breathing | Yes / No | 1 / 0 | 1 = khó thở, 0 = bình thường |
| gender | Male / Female | OneHotEncoder (handle\_unknown='ignore') | Mỗi giá trị giới tính tạo một cột nhị phân (1 = đúng giá trị, 0 = không). Giá trị mới (ví dụ: other)không có trong tập huấn luyện sẽ bỏ qua mà không gây lỗi |
| outcome\_variable | Positive/Negative | Positive/Negative | Positive: dự đoán mắc bệnh cúm Negative: dự đoán không mắc bệnh cúm |

**b) Biến numeric (Age)**

Do tuổi là dữ liệu liên tục, nhưng mức độ ảnh hưởng đến bệnh cảm cúm chủ yếu mang tính phân nhóm(trẻ em, thanh niên, trung niên, người già). Vì vậy, dữ liệu tuổi được phân thành 4 nhóm tuổi để mô hình dễ nhận biết rủi ro theo nhóm tuổi. Mặc dù GaussianNB có thể xử lý dữ liệu liên tục, việc nhóm tuổi giúp mô hình dễ học các đặc trưng quan trọng và đồng nhất với các biến nhị phân/OneHotEncode khác trong pipeline.

Bảng 2.3: Mã hóa dữ liệu biến Age

|  |  |  |
| --- | --- | --- |
| **Nhóm** | **Điều kiện** | **Mã lưu trong dữ liệu** |
| Trẻ em | < 18 | Tre\_em |
| Thanh niên | 18-35 | Thanh\_nien |
| Trung niên | 36-60 | Trung\_nien |
| Người già | > 60 | Nguoi\_gia |

**Bước 3: Kiểm tra dữ liệu**

Nhóm tiến hành kiểm tra dữ liệu để phát hiện các vấn đề tiềm ẩn và đảm bảo chất lượng trước khi đưa vào mô hình Naïve Bayes. Dữ liệu đầy đủ, không có giá trị trống, nên không cần xử lý các giá trị thiếu. Tất cả các giá trị đều nằm trong phạm vi hợp lý, ví dụ tuổi từ 1–90 và các triệu chứng sốt, ho, mệt mỏi ở dạng nhị phân 0/1, không có giá trị bất thường.

Phân bố dữ liệu tương đối cân bằng, giúp mô hình không bị lệch và tránh việc học quá mức vào nhãn chiếm ưu thế:

* Fever: 48% Yes, 52% No → cân bằng.
* Cough: 50% Yes, 50% No → cân bằng.
* Fatigue: 46% Yes, 54% No → tương đối cân bằng.
* Age\_Group: Trẻ em 18%, Thanh niên 35%, Trung niên 30%, Người già 17% → phân bố hợp lý.
* Outcome: Có bệnh 45%, Không bệnh 55% → cân bằng.

**Bước 4: Chuẩn bị dữ liệu cho mô hình**

Sau khi kiểm tra, nhóm chuẩn bị dữ liệu để đưa vào mô hình. Tất cả các biến đã ở dạng số hoặc nhị phân, nên có thể trực tiếp tính xác suất mà không cần chuẩn hóa thêm, vì Naïve Bayes dựa trên xác suất rời rạc, không dựa trên khoảng cách hay tỉ lệ. Dữ liệu được chia thành hai phần: 80% dùng để huấn luyện mô hình và 20% dùng để kiểm tra hiệu suất dự đoán. Việc chia như vậy giúp đánh giá được khả năng dự đoán của mô hình trên dữ liệu mới, đảm bảo kết quả khách quan và chính xác.

### 2.2.3 Cơ sở dữ liệu thu được

Sau bước xử lý:

* Số thuộc tính: 7 (fever, cough, fatigue, difficultly\_breathing, age\_group, gender, outcome\_variable)
* Dữ liệu sạch 100%: không null, không trùng lặp
* Giá trị đã chuẩn hóa và mã hóa đồng nhất
* Tất cả biến numeric/boolean, sẵn sàng huấn luyện mô hình Naïve Bayes
* Phân bố dữ liệu cân bằng, giúp mô hình học tốt và dự đoán chính xác
* Thu được tất cả 349 mẫu kết quả với các thuộc tính : fever, cough, fatigue, difficulty\_breathing, age, gender, outcome .
* Một số mẫu kết quả thu được :

Bảng 2.4: Một số mẫu kết quả dữ liệu thu được sau xử lý

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **fever** | **cough** | **fatigue** | **difficulty\_breathing** | **age** | **gender** | **outcome\_variable** |
| 1 | 0 | 1 | 1 | 19 | Female | Positive |
| 0 | 1 | 1 | 0 | 25 | Female | Negative |
| 1 | 1 | 0 | 1 | 25 | Male | Positive |
| 1 | 0 | 0 | 0 | 30 | Female | Negative |
| 0 | 1 | 1 | 1 | 28 | Male | Positive |
| 1 | 1 | 1 | 0 | 35 | Female | Positive |
| 0 | 0 | 0 | 0 | 40 | Male | Negative |
| 1 | 0 | 1 | 0 | 22 | Female | Positive |
| 0 | 1 | 0 | 1 | 27 | Male | Negative |
| 1 | 1 | 1 | 1 | 33 | Female | Positive |
| 0 | 0 | 1 | 0 | 45 | Male | Negative |
| 1 | 1 | 0 | 0 | 50 | Female | Positive |
| 0 | 1 | 1 | 1 | 18 | Male | Positive |
| 1 | 0 | 0 | 1 | 60 | Female | Positive |
| 0 | 0 | 0 | 0 | 65 | Male | Negative |
| 1 | 1 | 1 | 0 | 38 | Female | Positive |

## 2.3 Xây dựng mô hình Naïve Bayes

### 2.3.1 Lựa chọn phương pháp

Thuật toán Naïve Bayes được lựa chọn vì phù hợp với đặc thù dữ liệu triệu chứng y khoa – vốn chủ yếu ở dạng nhị phân hoặc phân loại như ho, sốt, đau đầu, mệt mỏi. Naïve Bayes xử lý rất tốt loại dữ liệu này nhờ giả định độc lập có điều kiện, giúp mô hình đơn giản nhưng vẫn hiệu quả.

Bên cạnh đó, Naïve Bayes có tốc độ huấn luyện nhanh, không yêu cầu lượng dữ liệu lớn và dễ triển khai trong môi trường thực tế. Điều này đặc biệt quan trọng trong bối cảnh chẩn đoán sớm, nơi hệ thống cần đưa ra kết quả nhanh và ổn định dù nguồn dữ liệu hạn chế.

Ngoài ra, mô hình Naïve Bayes dựa trên xác suất nên có khả năng giải thích rõ ràng lý do dự đoán, hỗ trợ nhân viên y tế hiểu được triệu chứng nào ảnh hưởng nhiều đến kết quả. Mô hình cũng hoạt động tốt với dữ liệu nhiễu hoặc thiếu, phù hợp với tình huống bệnh nhân không mô tả đầy đủ triệu chứng

### 2.3.2 Lựa chọn biến thể Naive bayes

Trong số các biến thể của thuật toán Naïve Bayes, nhóm lựa chọn Gaussian Naïve Bayes (GaussianNB) cho bài toán này. Tập thuộc tính đầu vào bao gồm:

* Các biến nhị phân: fever, cough, fatigue, difficulty\_breathing (mã hóa 1/0).
* Các biến phân loại: gender, age\_group (sau OneHotEncode).

Các thuộc tính đầu vào đã được mã hóa dạng số, giúp GaussianNB có thể mô hình hóa hiệu quả nhờ giả định phân phối Gaussian trên các đặc trưng numeric. Dữ liệu y tế thường mang tính hỗn hợp, bao gồm triệu chứng nhị phân và các thuộc tính định danh, nên GaussianNB vẫn phù hợp và hoạt động tốt.

So với các biến thể khác:

* BernoulliNB: chỉ phù hợp với dữ liệu nhị phân thuần túy.
* CategoricalNB: thích hợp với dữ liệu rời rạc chưa được mã hóa.
* GaussianNB: linh hoạt khi dữ liệu có yếu tố numeric hoặc hỗn hợp, như dữ liệu của bài toán này. Ngoài ra, GaussianNB có độ ổn định cao khi triển khai ứng dụng Web, cho phép xử lý dự đoán nhanh và mượt mà.

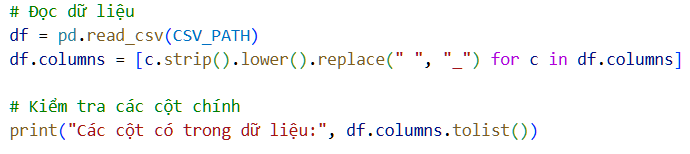
Với đặc điểm dữ liệu và yêu cầu mô hình hóa đơn giản nhưng hiệu quả, GaussianNB được lựa chọn làm biến thể chính cho mô hình Naïve Bayes trong đề tài.

### 2.3.3 Cài đặt mô hình

### Bước 1: Đọc dữ liệu và tiền xử lý

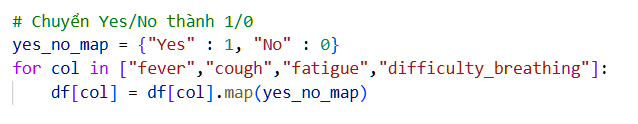
Trước tiên, dữ liệu triệu chứng được đọc từ file CSV và các cột được chuẩn hóa tên. Các nhãn Yes/No của triệu chứng như sốt, ho, mệt mỏi, khó thở được chuyển sang dạng nhị phân 0/1, phù hợp với mô hình Gaussian Naïve Bayes. Biến tuổi (age) được chuyển thành nhóm tuổi (age\_group) để biến liên tục thành phân loại, giúp mô hình dễ học hơn.

* Đọc dữ liệu và kiểm tra các cột



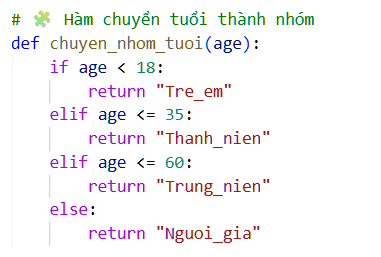
Hình 2.3: Đọc dữ liệu và kiếm tra các cột

* Các nhãn Yes/No được chuyển sang dạng nhị phân 0/1, giúp mô hình GaussianNB có thể xử lý dữ liệu numeric

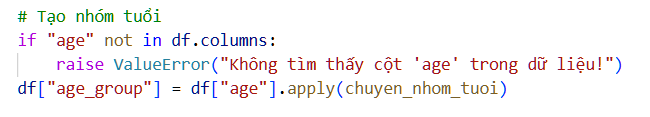


Hình 2.4: Chuyển các nhãn Yes/No sang dạng nhị phân 0/1

* Biến tuổi (age) được chuyển thành nhóm tuổi(age\_group) để biến liên tục thành biến phân loại.



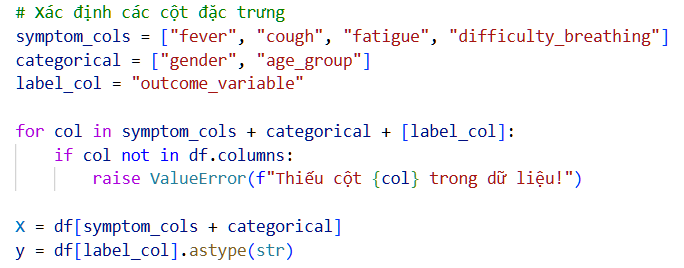
Hình 2.5: Biến tuổi (age) được chuyển thành nhóm tuổi



Hình 2.6: Tạo nhóm tuổi age\_group

**Bước 2: Xác định đặc trưng và nhãn**

Các cột triệu chứng và biến phân loại (gender, age\_group) được chọn làm đặc trưng (X), trong khi cột kết quả (outcome\_variable) là nhãn (y). Việc tách biệt rõ ràng giúp mô hình biết đâu là thông tin đầu vào và đâu là kết quả cần dự đoán.



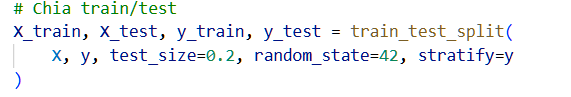
Hình 2.7: Xác định các cột đặc trưng

Trong đó:

* x là ma trận đặc trưng gồm các triệu chứng và biến phân loại.
* y là nhãn cần dự đoán (Positive/Negative).

**Bước 3: Chia dữ liệu train/test**

Dữ liệu được chia thành 80% cho huấn luyện và 20% cho kiểm tra, đồng thời duy trì tỷ lệ lớp nhãn giống nhau (stratify). Việc này giúp đánh giá chính xác khả năng dự đoán của mô hình trên dữ liệu chưa từng thấy và tránh hiện tượng overfitting.



Hình 2.8: Chia dữ liệu train/test

**Bước 4: Tạo pipeline và huấn luyện Gaussian Naïve Bayes**

Pipeline bao gồm OneHotEncoder cho biến phân loại và GaussianNB.



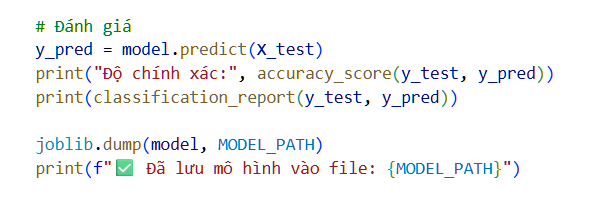
Hình 2.9: Tạo pipeline và huấn luyện GausianNB

Trong đó:

* GaussianNB tính xác suất tiên nghiệm P(Y) và xác suất điều kiện P(Xi|Y).
* OneHotEncoder biểu diễn biến phân loại dưới dạng numeric, phù hợp với GaussianNB.

**Bước 5: Dự đoán, đánh giá và lưu mô hình**

Sau khi huấn luyện, mô hình được đánh giá và lưu để sử dụng trong ứng dụng Streamlit.



Hình 2.10: Đánh giá và lưu mô hình

Ứng dụng với Streamlit:

* Nhập dữ liệu từ người dùng và tạo DataFrame



Hình 2.11: Nhập dữ liệu từ người dùng và tạo DataFrame

* Dự đoán xác suất cho từng nhãn



Hình 2.12: Dự đoán xác suất cho từng nhãn

* Dự đoán nhã kết quả



Hình 2.13: Dự đoán nhãn kết quả

## 2.4 Kết quả thực nghiệm

Dưới đây, là hình ảnh minh họa giao diện mà nhóm đã xây dựng được:

****

Hình 2.14: Giao diện

Sau khi xây dựng mô hình Naïve Bayes và triển khai ứng dụng trên giao diện Streamlit, chúng tôi tiến hành nhập các triệu chứng của người dùng để kiểm tra khả năng dự đoán bệnh cúm. Giao diện cho phép người dùng chọn các triệu chứng, giới tính và nhóm tuổi, sau đó nhấn nút “Chẩn đoán” để nhận kết quả.



Hình 2.15: Kết quả minh họa trên giao diện đã xây dựng

Ví dụ minh họa trong hình cho thấy trường hợp một người dùng nữ, thuộc nhóm tuổi thanh niên (18–35), có các triệu chứng: sốt, mệt mỏi và khó thở, nhưng không ho. Mô hình dự đoán Positive (có khả năng mắc cúm) với xác suất:

* Positive: 0.79
* Negative: 0.21

Kết quả này trùng khớp với mẫu 1 trong dataset đã sử dụng để huấn luyện và kiểm tra mô hình, chứng tỏ rằng mô hình Naïve Bayes đã học được mối quan hệ giữa các triệu chứng và kết quả bệnh một cách chính xác. Việc cung cấp cả dự đoán và xác suất giúp người dùng nắm rõ mức độ nguy cơ và tin cậy của dự đoán.

# KẾT LUẬN VÀ ĐỊNH HƯỚNG PHÁT TRIỂN

**a) Kết luận**

Trong quá trình thực hiện đề tài, nhóm đã xây dựng một mô hình Naïve Bayes để dự đoán khả năng mắc bệnh cúm dựa trên các triệu chứng ban đầu như sốt, ho, mệt mỏi, khó thở, kết hợp với thông tin cá nhân như độ tuổi và giới tính. Nhóm đã hoàn tất các bước tiền xử lý dữ liệu, bao gồm mã hóa các biến nhị phân, phân loại tuổi theo nhóm và chuẩn hóa dữ liệu, từ đó đưa dữ liệu vào mô hình phân loại nhị phân. Bên cạnh đó, nhóm cũng thiết kế một biểu mẫu nhập liệu đơn giản giúp người dùng có thể điền triệu chứng của mình và nhận kết quả dự đoán trực tiếp. Kết quả thực nghiệm cho thấy mô hình vận hành ổn định, cung cấp dự đoán có độ tin cậy tương đối, phù hợp với yêu cầu học phần và minh họa rõ cách thuật toán Naïve Bayes hoạt động trên dữ liệu thực tế.

Tuy nhiên, đề tài vẫn tồn tại một số hạn chế. Dữ liệu sử dụng chủ yếu là mô phỏng, trích lọc từ bộ dữ liệu gốc, chưa phản ánh đầy đủ tính đa dạng của bệnh nhân thực tế. Mô hình hiện chỉ tập trung vào bệnh cúm và không mở rộng cho các bệnh truyền nhiễm khác. Trong quá trình triển khai, nhóm gặp khó khăn trong việc thu thập nguồn dữ liệu, lựa chọn biến đầu vào phù hợp và xử lý dữ liệu để tương thích với Naïve Bayes, đồng thời hiệu suất dự đoán có thể bị ảnh hưởng khi dữ liệu phức tạp hoặc mất cân bằng.

**b) Định hướng phát triển**

Trong tương lai, mô hình có thể được mở rộng để dự đoán nhiều bệnh truyền nhiễm phổ biến cùng lúc, chẳng hạn như viêm họng, cảm lạnh hay COVID-19, thông qua các mô hình đa nhãn hoặc nhiều mô hình Naïve Bayes riêng biệt. Việc thu thập thêm dữ liệu thực tế và bổ sung các biến số liên quan sẽ giúp nâng cao độ chính xác của dự đoán. Ngoài ra, có thể thử nghiệm các thuật toán học máy khác như Decision Tree, Random Forest hoặc Logistic Regression để so sánh hiệu quả và tối ưu hóa mô hình. Về ứng dụng, mô hình có thể phát triển thành hệ thống web hoặc mobile, hỗ trợ người dùng nhập dữ liệu nhiều triệu chứng, nhận dự đoán nhiều bệnh, lưu trữ lịch sử và tạo báo cáo thống kê, đồng thời tích hợp vào các cơ sở y tế để hỗ trợ chẩn đoán ban đầu và theo dõi dịch tễ học, góp phần nâng cao hiệu quả công tác y tế và phòng chống bệnh truyền nhiễm.

# TÀI LIỆU THAM KHẢO

|  |  |
| --- | --- |
| [1] | Nguyễn Phương Nga, Giáo trình Trí tuệ nhân tạo, Thống kê, 2021. |
| [2] | Bùi Việt Hà, Python cơ bản, ĐHQGHN, 2021. |
| [3] | Hoàng Xuân Huấn, Giáo trình học máy, ĐHQGHN, 2015 |
| [4] | Laksika Tharmalingam, Disease Symptoms and Patient Profile Dataset, Kaggle, 2023. (https://www.kaggle.com/datasets/uom190346a/disease-symptoms-and-patient-profile-dataset?) |
| [5] | World Health Organization (WHO), Influenza (Seasonal) Fact Sheet, WHO, 2024. (https://www.who.int/news-room/fact-sheets/detail/influenza-(seasonal)) |