# Lê Văn Thông

**3123410362**

# Định nghĩa thuật toán phân loại naive bayes.

Phân loại Naive Bayes là gì?

Naive Bayes Classifier thuộc họ ‘[thuật toán học tạo sinh](https://www.analyticsvidhya.com/blog/2023/04/what-is-generative-ai/)**’** , nhằm mục đích mô hình hóa sự phân bố đầu vào trong một lớp hoặc danh mục cụ thể. Không giống như các bộ phân loại phân biệt như ‘[hồi quy logistic](https://www.analyticsvidhya.com/blog/2021/08/conceptual-understanding-of-logistic-regression-for-data-science-beginners/)**’** , Naive Bayes Classifier không học được đặc điểm nào là quan trọng nhất để phân biệt giữa các lớp. Thuật toán này được sử dụng rộng rãi trong phân loại văn bản, lọc thư rác và hệ thống đề xuất.

# 1 Thuật toán naive bayes hoạt động như thế nào?

Thuật toán hoạt động bằng cách tính toán xác suất một điểm dữ liệu thuộc về một lớp cụ thể. Nó sử dụng định lý Bayes để xác định xác suất hậu nghiệm của một lớp, cho một tập hợp các đặc trưng quan sát được. Giả định độc lập "ngây thơ" đơn giản hóa đáng kể phép tính này. Thay vì xem xét các mối quan hệ phức tạp giữa các đặc trưng, mô hình coi đóng góp của mỗi đặc trưng vào kết quả là hoàn toàn riêng biệt.

Ví dụ: khi phân loại một email là thư rác hay không phải thư rác, bộ phân loại Naive Bayes giả định rằng sự hiện diện của từ "sale" là độc lập với sự hiện diện của từ "free". Giả định này hiếm khi đúng, nhưng nó cho phép mô hình học và đưa ra dự đoán rất nhanh mà không cần một lượng lớn [**dữ liệu huấn luyện**](https://www.ultralytics.com/vi/glossary/training-data). Điều quan trọng là phải phân biệt Naive Bayes với [**Mạng Bayesian**](https://www.ultralytics.com/vi/glossary/bayesian-network); mặc dù cả hai đều sử dụng các nguyên tắc Bayesian, nhưng Mạng Bayesian là một mô hình tổng quát hơn có thể thể hiện các phụ thuộc phức tạp, trong khi Naive Bayes là một bộ phân loại cụ thể với một giả định độc lập cứng nhắc.

## Hãy giải thích định lý Bayes và giả định "ngây thơ" trong thuật toán này?

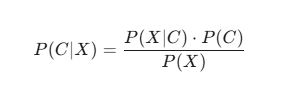
Khái niệm:

Định lý Bayes là một công thức toán học dùng để xác định xác suất có điều kiện. Xác suất có điều kiện là khả năng một kết quả xảy ra dựa trên một kết quả trước đó trong những hoàn cảnh tương tự. Do đó, Định lý Bayes cung cấp một cách để sửa đổi hoặc cập nhật một dự đoán hoặc lý thuyết hiện có khi có bằng chứng mới.

Trong tài chính, Định lý Bayes có thể được sử dụng để đánh giá hoặc đánh giá lại [rủi ro](https://www.investopedia.com/terms/r/risk.asp) khi cho người vay tiềm năng vay tiền.

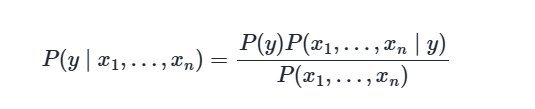
Định lý Bayes được đặt theo tên nhà toán học người Anh thế kỷ 18 Thomas Bayes. Nó còn được gọi là Quy tắc Bayes hoặc Luật Bayes và là nền tảng của lĩnh vực thống kê Bayes.

Công thức Định lý Bayes:

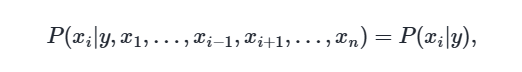


Giả định "Ngây thơ" (Naive Assumption):

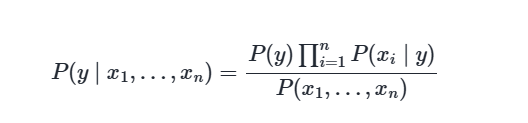
Phương pháp Bayes ngây thơ là một tập hợp các thuật toán học có giám sát dựa trên việc áp dụng định lý Bayes với giả định "ngây thơ" về tính độc lập có điều kiện giữa mọi cặp đặc trưng với giá trị của biến lớp cho trước. Định lý Bayes phát biểu mối quan hệ sau, với biến lớp cho trước:và vector đặc trưng phụ thuộc x1 bởi vì xn, :



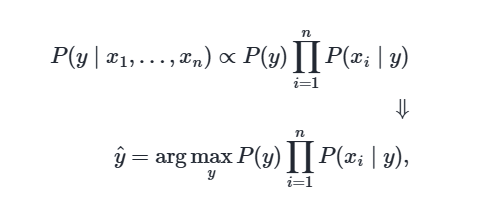
Sử dụng giả định độc lập có điều kiện ngây thơ rằng:



cho tất cả i, mối quan hệ này được đơn giản hóa thành:



Từ P(x1,….,xn) là hằng số với đầu vào cho trước, chúng ta có thể sử dụng quy tắc phân loại sau:



và chúng ta có thể sử dụng ước lượng A Posteriori Tối đa (MAP) để ước lượng P(y) Và P(xi,y) ; cái trước là tần số tương đối của lớp y trong tập huấn luyện.

Các bộ phân loại Bayes ngây thơ khác nhau chủ yếu khác nhau ở các giả định mà chúng đưa ra liên quan đến sự phân phối của P(xi,y).

Bất chấp những giả định có vẻ quá đơn giản, các bộ phân loại Bayes ngây thơ đã hoạt động khá hiệu quả trong nhiều tình huống thực tế, nổi tiếng là phân loại tài liệu và lọc thư rác. Chúng chỉ cần một lượng nhỏ dữ liệu huấn luyện để ước tính các tham số cần thiết. (Để biết lý do lý thuyết tại sao Bayes ngây thơ hoạt động tốt và loại dữ liệu nào phù hợp, hãy xem các tài liệu tham khảo bên dưới.)

Các bộ học và phân loại Naive Bayes có thể cực kỳ nhanh so với các phương pháp phức tạp hơn. Việc tách rời các phân phối đặc trưng có điều kiện của lớp đồng nghĩa với việc mỗi phân phối có thể được ước tính độc lập như một phân phối một chiều. Điều này giúp giảm bớt các vấn đề phát sinh từ "lời nguyền của chiều".

Mặt khác, mặc dù Bayes ngây thơ được biết đến là một bộ phân loại tốt, nhưng nó lại là một bộ ước lượng kém, do đó, kết quả xác suất predict\_probakhông được coi trọng quá mức.

Nguồn tham khảo: <https://scikit-learn.org/stable/modules/naive_bayes.html>

# 2Các loại mô hình Naive Bayes (Gaussian, Multinomial, Bernoulli) khác nhau ra sao?

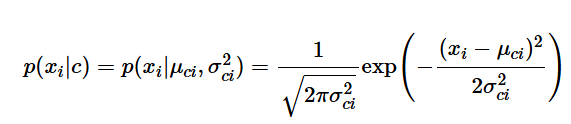
* **Gaussian Naive Bayes (GaussianNB):** Đây là một biến thể của bộ phân loại Naive Bayes, được sử dụng với phân phối chuẩn Gauss—tức là phân phối chuẩn—và các biến liên tục. Mô hình này được hiệu chỉnh bằng cách tìm giá trị trung bình và độ lệch chuẩn của mỗi lớp.
* **Multinomial Naive Bayes (MultinomialNB):** Loại phân loại Naive Bayes này giả định rằng các đặc trưng đến từ phân phối đa thức. Biến thể này hữu ích khi sử dụng dữ liệu rời rạc, chẳng hạn như số lần xuất hiện, và thường được áp dụng trong các trường hợp sử dụng xử lý ngôn ngữ tự nhiên, chẳng hạn như phân loại thư rác.
* **Bernoulli Naive Bayes (BernoulliNB):** Đây là một biến thể khác của bộ phân loại Naïve Bayes, được sử dụng với các biến Boolean—tức là các biến có hai giá trị, chẳng hạn như True và False hoặc 1 và 0.

## Khi nào nên sử dụng từng loại?

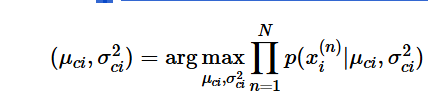
### Gaussian Naive Bayes

Mô hình này được sử dụng chủ yếu trong loại dữ liệu mà các thành phần là các biến liên tục.

Với mỗi chiều dữ liệu i và một class c, xi tuân theo một phân phối chuẩn có kỳ vọng μci và phương sai σ2ci:



Trong đó, bộ tham số θ={μci,σ2ci} được xác định bằng Maximum Likelihood:

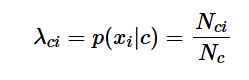


Đây là cách tính của thư viện sklearn. Chúng ta cũng có thể đánh giá các tham số bằng MAP nếu biết trước priors của μci và σ2ci

### Multinomial Naive Bayes

Mô hình này chủ yếu được sử dụng trong phân loại văn bản mà feature vectors được tính bằng [Bags of Words](https://machinelearningcoban.com/general/2017/02/06/featureengineering/#bag-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|c) 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 c. Giá trị này có thể được tính bằng cách:

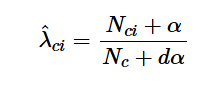


Trong đó:

* Nci là tổng số lần từ thứ i xuất hiện trong các văn bản của class c, nó được tính là tổng của tất cả các thành phần thứ i của các feature vectors ứng với class c.
* Nc là tổng số từ (kể cả lặp) xuất hiện trong class c. Nói cách khác, nó bằng tổng độ dài của toàn bộ các văn bản thuộc vào class c. Có thể suy ra rằng Nc=∑di=1Nci, từ đó ∑di=1λci=1.

Cách tính này có một hạn chế là nếu có một từ mới chưa bao giờ xuất hiện trong class c thì biểu thức (10) sẽ bằng 0, điều này dẫn đến vế phải của (7) bằng 0 bất kể các giá trị còn lại có lớn thế nào. Việc này sẽ dẫn đến kết quả không chính xác (xem thêm ví dụ ở mục sau).

Để giải quyết việc này, một kỹ thuật được gọi là Laplace smoothing được áp dụng:



Với α là một số dương, thường bằng 1, để tránh trường hợp tử số bằng 0. Mẫu số được cộng với dα để đảm bảo tổng xác suất ∑di=1^λci=1.

Như vậy, mỗi class c sẽ được mô tả bởi bộ các số dương có tổng bằng 1: ^λc={^λc1,…,^λcd}.

### Bernoulli Naive Bayes

Mô hình này được áp dụng cho các loại dữ liệu mà mỗi thành phần là một giá trị binary - bẳng 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.

Nguồn tham khảo các loại mô hình Naïve Bayes:

https://www.ibm.com/think/topics/naive-bayes

Nguồn tham khảo khi nào nên dùng từng loại:

<https://machinelearningcoban.com/2017/08/08/nbc/>

# 3 Tại sao Naive Bayes được gọi là "ngây thơ"?

-4Thuật toán này được coi là ngây thơ và vô tội chỉ vì **giả định mà nó dựa vào hầu như không bao giờ xảy ra trong thực tế.**

**Ví dụ:** Ứng dụng NLP: Phân loại email rác.

Bộ phân loại Naive Bayes đơn giản thường tính toán xác suất của các từ/cụm từ nhất định trong dữ liệu và đặt tên cho nhóm quan sát dựa trên xác suất cao nhất.

Giả sử nó phân loại email có từ khóa "Nhấp vào đây" thành hai nhóm bằng cách tính xác suất theo công thức sau:

P = **P(Click | N)** x **P(This | N)** x P(N), bằng với **P(This | N)** x **P(Click | N)** x P(N)

Hay nói cách khác, **P (Click This) = P (This Click)** . Bạn có thấy điều gì sai ở đây không?

Vì nó cho rằng “ **Click”** và “ **This”** là độc lập nên thứ tự các từ bị bỏ qua, điều này vi phạm đáng kể ngôn ngữ nói thực sự của con người.

Thứ hai, các từ trong phát hiện thư rác chắc chắn có trọng số khác nhau đối với biến mục tiêu. Ví dụ, " **Tiền** " có thể được tập trung hơn " **Cái** ".

Nguồn: <https://machinelearningcoban.com/2017/08/08/nbc/>

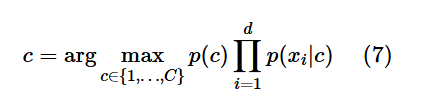
## Giả định về tính độc lập của các đặc trưng ảnh hưởng như thế nào đến hiệu suất của mô hình?

Giả thiết các chiều của dữ liệu độc lập với nhau, nếu biết c, là quá chặt và ít khi tìm được dữ liệu mà các thành phần hoàn toàn độc lập với nhau. Tuy nhiên, giả thiết *ngây ngô* này lại mang lại những kết quả tốt bất ngờ. Giả thiết về sự độc lập của các chiều dữ liệu này được gọi là *Naive Bayes* (xin không dịch). Cách xác định class của dữ liệu dựa trên giả thiết này có tên là *Naive Bayes Classifier (NBC)*.

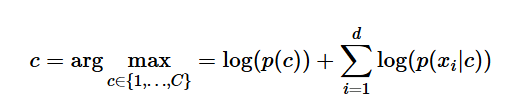
NBC, nhờ vào tính đơn giản một cách *ngây thơ*, có tốc độ training và test rất nhanh. Việc này giúp nó mang lại hiệu quả cao trong các bài toán large-scale.

Ở bước **training**, các phân phối p(c) và p(xi|c),i=1,…,d sẽ được xác định dựa vào training data. Việc xác định các giá trị này có thể dựa vào [Maximum Likelihood Estimation hoặc Maximum A Posteriori](https://machinelearningcoban.com/2017/07/17/mlemap/).

Ở bước **test**, với một điểm dữ liệu mới x, class của nó sẽ được xác đinh bởi:



Khi d lớn và các xác suất nhỏ, biểu thức ở vế phải của (7) sẽ là một số rất nhỏ, khi tính toán có thể gặp sai số. Để giải quyết việc này, (7) thường được viết lại dưới dạng tương đương bằng cách lấy log của vế phải:



Việc này không ảnh hưởng tới kết quả vì log là một hàm đồng biến trên tập các số dương.

Mặc dù giả thiết mà Naive Bayes Classifiers sử dụng là quá phi thực tế, chúng vẫn hoạt động khá hiệu quả trong nhiều bài toán thực tế, đặc biệt là trong các bài toán phân loại văn bản, ví dụ như lọc tin nhắn rác hay lọc email spam. Trong phần sau của bài viết.

Cả việc training và test của NBC là cực kỳ nhanh khi so với các phương pháp classification phức tạp khác. Việc giả sử các thành phần trong dữ liệu là độc lập với nhau, nếu biết class, khiến cho việc tính toán mỗi phân phối p(xi|c) trở nên cực kỳ nhanh.

Mỗi giá trị p(c),c=1,2,…,C có thể được xác định như là tần suất xuất hiện của class c trong training data.

Việc tính toán p(xi|c) phụ thuộc vào loại dữ liệu. [Có ba loại được sử dụng phổ biến](http://scikit-learn.org/dev/modules/classes.html#module-sklearn.naive_bayes) là: Gaussian Naive Bayes, Multinomial Naive Bayes, và Bernoulli Naive .

Nguồn: <https://medium.com/@kirudang/why-naive-bayes-is-called-naive-and-what-are-the-benefits-of-being-naive-180757155b69>

# 4Ưu điểm và hạn chế của Naive Bayes so với các thuật toán phân loại khác như SVM hoặc Random Forest là gì?

## Phân loại Bayes ngây thơ (NB):

Bộ [phân loại Naive Bayes (NB)](https://www.geeksforgeeks.org/machine-learning/naive-bayes-classifiers/) là một mô hình học máy xác suất được sử dụng rộng rãi cho các tác vụ phân loại văn bản. Mặc dù có tên gọi đơn giản, nhưng hiệu quả của nó bắt nguồn từ nền tảng lý thuyết vững chắc và khả năng xử lý hiệu quả dữ liệu văn bản đa chiều. Nó đặc biệt hiệu quả với dữ liệu đa chiều và có thể xử lý các tập dữ liệu lớn một cách hiệu quả. Tính đơn giản, tốc độ và khả năng hoạt động tốt với dữ liệu hạn chế của thuật toán khiến nó trở thành một lựa chọn phổ biến, đặc biệt khi cần cân nhắc đến tài nguyên tính toán trong các ứng dụng thực tế. Về **Nền tảng Xác suất,** NB tận dụng [định lý Bayes](https://www.geeksforgeeks.org/maths/bayes-theorem/) , tính toán xác suất một văn bản thuộc một [lớp](https://www.geeksforgeeks.org/python/python-oops-concepts/) cụ thể dựa trên xác suất riêng lẻ của các từ cấu thành xuất hiện trong lớp đó.

* **Giả định ngây thơ:**  Khía cạnh "ngây thơ" nằm ở giả định rằng các từ xuất hiện trong một lớp là độc lập với nhau. Mặc dù giả định này hiếm khi hoàn toàn đúng, nhưng đáng ngạc nhiên là nó lại mang lại hiệu suất mạnh mẽ đáng kinh ngạc trong nhiều tình huống thực tế.
* **Tính linh hoạt:**  NB hoạt động tốt với cả biểu diễn từ đa thức và Bernoulli, thích ứng với các đặc điểm văn bản khác nhau. Đa thức ghi lại tần suất xuất hiện của từ trong một tài liệu, trong khi Bernoulli chỉ xem xét sự hiện diện hoặc vắng mặt của từ.
* NB yêu cầu thời gian đào tạo và [thiết kế tính năng](https://www.geeksforgeeks.org/machine-learning/what-is-feature-engineering/) tối thiểu , lý tưởng cho các ứng dụng yêu cầu dự đoán nhanh và thích ứng nhanh với dữ liệu mới.

## Máy vectơ hỗ trợ (SVM):

[Máy vectơ hỗ trợ](https://www.geeksforgeeks.org/machine-learning/support-vector-machine-algorithm/) (SVM) là một thuật toán [học có giám sát](https://www.geeksforgeeks.org/machine-learning/supervised-machine-learning/) mạnh mẽ , nổi trội trong việc phân biệt các loại văn bản khác nhau, rất hữu ích cho các tác vụ như phân tích cảm xúc, gán nhãn chủ đề và phát hiện thư rác. Về bản chất, SVM hướng đến việc tìm ra siêu phẳng tối ưu - một ranh giới quyết định trong không gian đa chiều, phân tách rõ ràng các lớp văn bản khác nhau. Hãy tưởng tượng việc vẽ mỗi tài liệu văn bản thành một điểm dựa trên các đặc điểm được trích xuất của nó (ví dụ: sự hiện diện của từ, tần suất). SVM tìm kiếm siêu phẳng tối đa hóa khoảng cách giữa các lớp này, đảm bảo sự phân biệt rõ ràng ngay cả với dữ liệu chưa được biết đến.

Mô hình SVM được huấn luyện trên dữ liệu đã được gán nhãn, trong đó mỗi tài liệu thuộc một danh mục cụ thể. Mô hình học siêu phẳng tối ưu phân tách tốt nhất các danh mục này trong không gian đặc trưng. Để xác thực, dựa trên các vectơ đặc trưng của chúng, mô hình dự đoán lớp mà chúng thuộc về bằng cách đặt chúng vào phía tương ứng của siêu phẳng.

hình ảnh

Mặc dù SVM hoạt động mặc định với siêu phẳng tuyến tính, nhưng "thủ thuật kernel" cho phép chúng xử lý các mối quan hệ phi tuyến tính giữa các đặc điểm. Điều này rất quan trọng đối với văn bản, nơi tồn tại các mối quan hệ ngữ nghĩa phức tạp giữa các từ.  
SVM thường thể hiện độ chính xác cao trong các tác vụ phân loại văn bản, cho các tập dữ liệu nhỏ hơn. Chúng có thể xử lý hiệu quả dữ liệu thưa thớt vốn có trong văn bản, nơi nhiều đặc điểm có thể bị thiếu trong từng tài liệu riêng lẻ.

## Bảng so sánh:

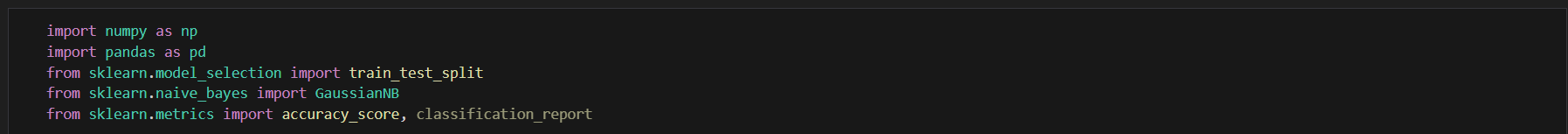
|  |  |  |
| --- | --- | --- |
| **Tiêu chuẩn** | **Bayes ngây thơ** | **SVM** |
| **Thuận lợi** | * Đơn giản và dễ thực hiện. * Hiệu quả về mặt tính toán. * Hoạt động tốt với các tập dữ liệu nhỏ. | * Có hiệu quả trong không gian có nhiều chiều. * Chống lại hiện tượng quá khớp. * Tính linh hoạt trong việc lựa chọn hàm hạt nhân. * Có thể nắm bắt được các mối quan hệ phức tạp. |
| **Hiệu quả** | * Đào tạo và dự đoán nhanh chóng. | * Việc đào tạo có thể tốn kém về mặt tính toán. * Đào tạo chậm hơn nhưng dự đoán nhanh hơn. |
| **Hiệu suất** | * Phù hợp cho các nhiệm vụ phân loại đơn giản. * Có thể xử lý tốt dữ liệu nhiễu. | * Hiệu suất tốt hơn trong các tác vụ phức tạp. * Nhạy cảm với dữ liệu nhiễu, đặc biệt nếu nó ảnh hưởng đến vị trí của ranh giới quyết định. |
| **Khả năng mở rộng** | * Có khả năng mở rộng tốt với các tập dữ liệu và tính năng lớn. | * Ít có khả năng mở rộng với các tập dữ liệu lớn. * Tốn nhiều bộ nhớ cho không gian tính năng lớn. |
| **Khả năng diễn giải** | * Cung cấp khả năng diễn giải trực tiếp. * Tính toán trực tiếp xác suất của lớp. | * Cung cấp ít khả năng diễn giải hơn. * Ranh giới quyết định khó diễn giải hơn. * Cung cấp ít thông tin chi tiết về tầm quan trọng của tính năng. |
| **Sự mạnh mẽ** | * Nhạy cảm với sự phân bố tính năng. * Có thể nhạy cảm với các hành vi vi phạm giả định về tính độc lập. | * Chống lại các yếu tố ngoại lai và nhiễu tốt hơn. |
| **Hạn chế** | * Sự phụ thuộc vào tính năng thách thức tính hợp lệ, tác động đến hiệu suất của Naive Bayes. * Sự đơn giản của Naive Bayes làm giảm độ chính xác của các mối quan hệ phức tạp. * Độ lệch phân phối tính năng làm giảm các giả định về hiệu suất của Naive Bayes. | * Đào tạo SVM đòi hỏi nhiều tài nguyên tính toán cho các tập dữ liệu lớn. * Sự thành công của SVM phụ thuộc vào việc điều chỉnh chính xác hạt nhân và quy tắc hóa. * SVM thiếu khả năng diễn giải, đặc biệt là trong phân loại văn bản có nhiều tính năng. |

Nguồn: <https://www.geeksforgeeks.org/machine-learning/naive-bayes-vs-svm-for-text-classification/>

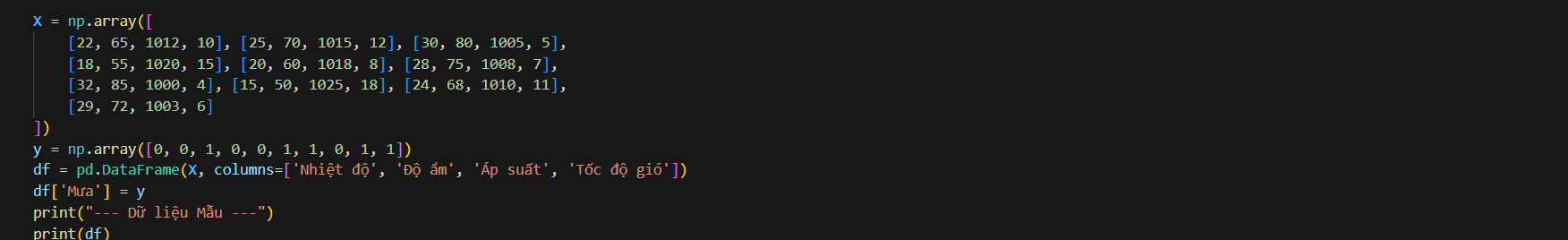
# 4Viết đoạn code mẫu bằng Python (sử dụng Scikit-learn) để xây dựng một mô hình Naive Bayes (ví dụ:Gaussian Naive Bayes) không? Hãy mô tả các bước thực hiện

Ta thử code mẫu trên một đoạn dữ liệu về thời tiết.

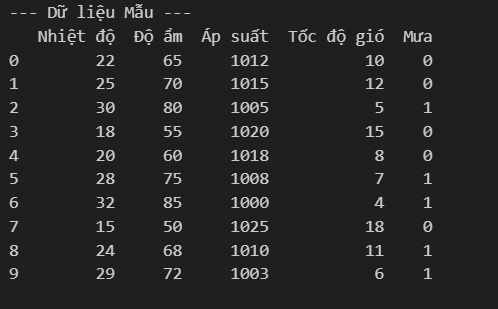
Bước 1 import thư viện.



Bước 2:chuẩn bị và khám phá dữ liệu.



Ta có được kết quả dữ liệu mẫu:



Nhận xét:

Dữ liệu giả định gồm 10 mẫu và 4 đặc trưng liên tục.

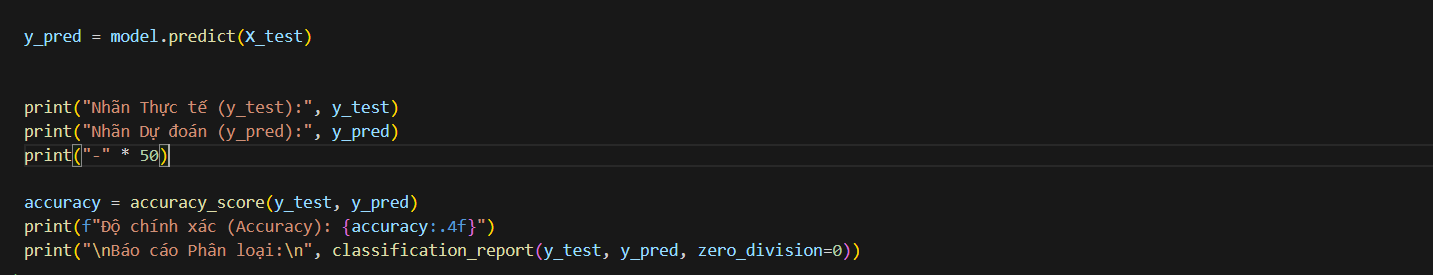
Số lượng mẫu lớp 0 ('Không mưa') là 5, số lượng mẫu lớp 1 ('Mưa') là 5.

Đây là một tập dữ liệu cân bằng.

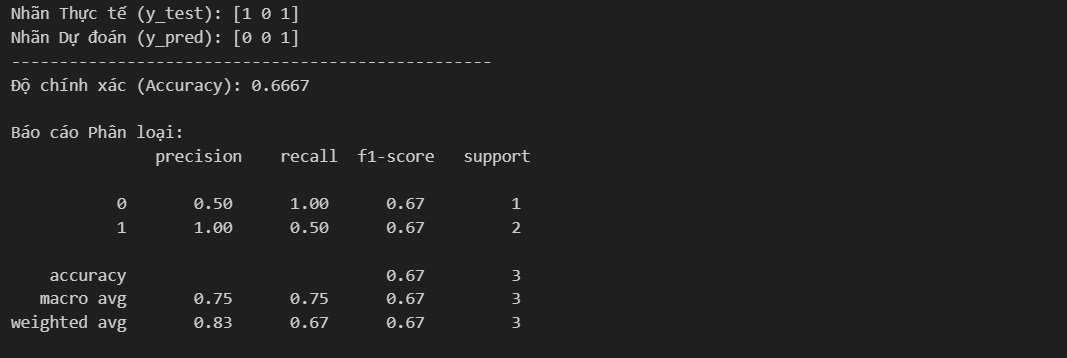
Bước 3:khởi tạo và huấn luyện mô hình.



Bước 4:dự đoán và đánh giá



Kết quả đạt được:



Độ chính xác (Accuracy): Cho biết tỷ lệ các mẫu được dự đoán đúng khoản

0.6667 = 2/3 mẫu đúng.

Báo cáo Phân loại (Classification Report):

- Precision (Độ chính xác): Tỷ lệ dự đoán 'Đúng' thực sự là đúng.

- Recall (Độ thu hồi): Tỷ lệ các trường hợp thực tế được dự đoán đúng.

- F1-score: Điểm cân bằng giữa Precision và Recall.

=> Trong ví dụ này, Naive Bayes có thể đã gặp khó khăn do tập dữ liệu huấn luyện quá nhỏ (chỉ 7 mẫu), làm giảm độ tin cậy của việc ước tính phân phối chuẩn (mean và variance).

# 5.Làm thế nào để xử lý dữ liệu phân loại (categorical data) trước khi áp dụng Multinomial Naive Bayes trong Python?

Để xử lý dữ liệu phân loại ta có phương pháp mã hoá đếm(Count Encoding / Count Vectorization) trước khi áp dụng Multinomial Naive Bayes.

# 6.Naive Bayes thường được sử dụng trong phân loại văn bản (text classification). Bạn có thể giải thích cách triển khai Naive Bayes cho bài toán này không?\

## Các bước triển khai cho văn bản:

### Bước 1: tiền xử lý văn bản.

Đây là bước chuẩn bị dữ liệu thô (tài liệu) để biến chúng thành định dạng có thể phân tích được:

**-**Tách từ (Tokenization): Chia văn bản thành các đơn vị nhỏ hơn (thường là từ).

**-**Chuyển đổi chữ thường (Lowercasing): Chuyển tất cả từ về chữ thường để tránh việc coi "The" và "the" là hai từ khác nhau.

**-**Loại bỏ Stopwords: Xóa các từ phổ biến nhưng không mang nhiều ý nghĩa phân loại (ví dụ: "thì", "là", "và", "của").

**-**Rút gọn từ (Stemming/Lemmatization): Đưa các từ về dạng gốc (ví dụ: "chạy", "đã chạy", "đang chạy" $\to$ "chạy").

### Bước 2:Trích xuất đặt trưng.

Mô hình Naive Bayes không thể làm việc trực tiếp với văn bản. Ta cần chuyển đổi văn bản đã được tiền xử lý thành các vectơ số bằng mô hình Túi Từ (BoW):

**-**Xây dựng Từ điển (Vocabulary): Tập hợp tất cả các từ vựng duy nhất từ toàn bộ dữ liệu huấn luyện.

**-**Mã hóa Đếm (Count Vectorization): Mỗi tài liệu (document) được biểu diễn thành một vectơ tần suất (count vector) có chiều dài bằng kích thước từ điển.

Giá trị tại mỗi vị trí trong vectơ là số lần từ đó xuất hiện trong tài liệu. Đây chính là định dạng đếm (count) mà Multinomial Naive Bayes yêu cầu.

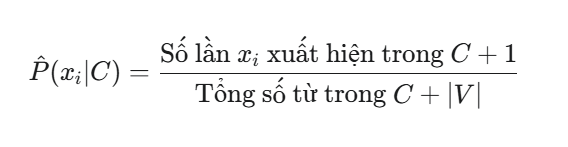
### Bước 3: Huấn luyện mô hình.

Sử dụng thuật toán Multinomial Naive Bayes (MNB) để học các xác suất từ dữ liệu đếm:

**-**Tính Xác suất Tiên nghiệm (P(C)): Xác suất một tài liệu thuộc về lớp $C$ (ví dụ: "Thể thao", "Kinh tế").

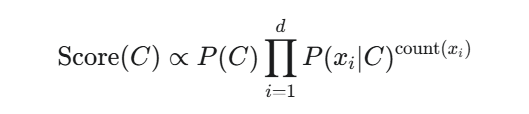
**-**Tính Khả năng Đặc trưng (P(x\_i|C)): Xác suất từ $x\_i$ xuất hiện trong một tài liệu thuộc lớp $C$.

MNB sử dụng công thức làm mịn Laplace (Laplace Smoothing) để tránh xác suất bằng không cho những từ chưa xuất hiện trong lớp đó:

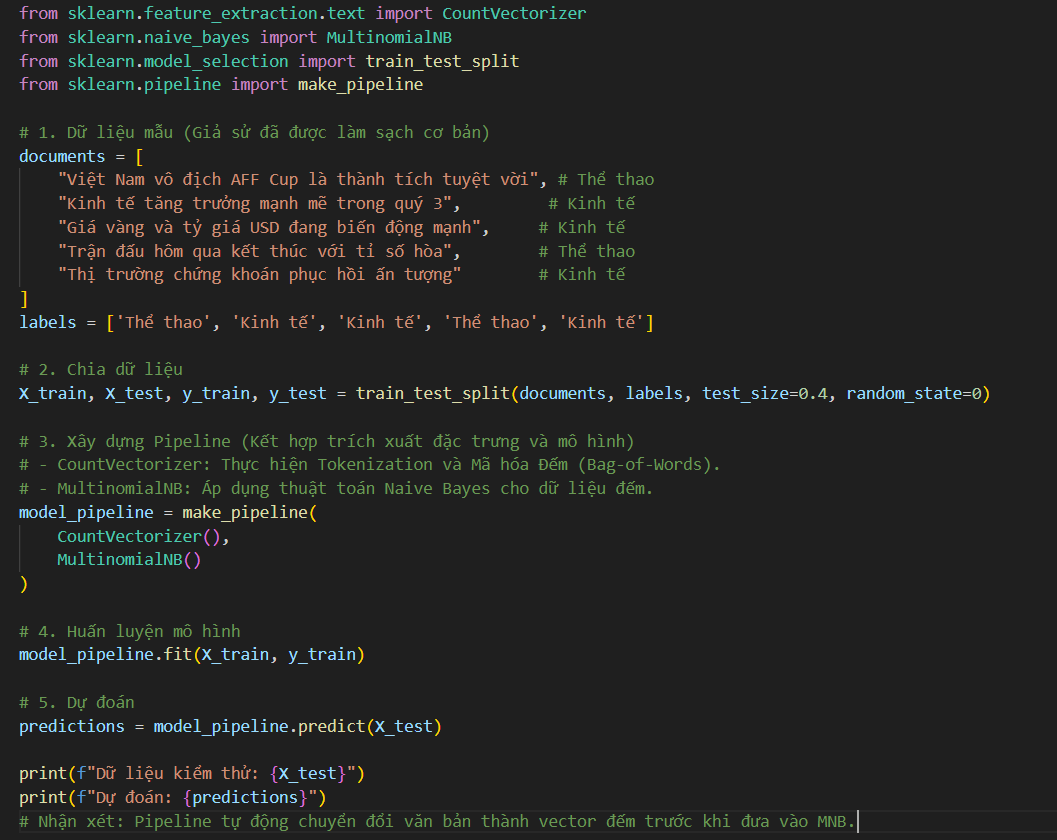


### Bước 4: Phân loại.

Khi có tài liệu mới X(mới), mô hình tính toán điểm số cho từng lớp C:



Sau đó, nó gán lớp có điểm số cao nhất cho tài liệu mới đó.



**Kết thúc.**