 **TRƯỜNG ĐẠI HỌC SƯ PHẠM KỸ THUẬT TP. HỒ CHÍ MINH**

**KHOA ĐÀO TẠO CHẤT LƯỢNG CAO**

**NGÀNH CÔNG NGHỆ THÔNG TIN**

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_



**BÁO CÁO ĐỒ ÁN 3**

**Đề tài: Tìm hiểu giải thuật Naive Bayes**

**GVHD:** *Ths***.***Từ Tuyết Hồng*

**SVTH:**

*Phan Xuân Tuấn Anh 17110096*

*Hoàng Tiến Thành 17110224*

**TP. Hồ Chí Minh, tháng 12 năm 2020**

**NHẬN XÉT CỦA GIẢNG VIÊN**

**………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………**

**………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………**

**Giảng viên**

**Lời cam đoan**

Trong quá trình thực hiện đồ án một, nhóm thực hiện đã được nhận nhiều giúp đỡ, đóng góp ý kiến và chỉ bảo nhiệt tình của cô và bạn bè. Nhóm chúng em xin gửi lời cảm ơn chân thành đến cô Từ Tuyết Hồng, giảng viên phụ trách hướng dẫn người đã tận tình hướng dẫn, chỉ bảo nhóm em trong suốt quá trình làm đồ án. Nhóm đã cố gắng thực hiện đồ án làm sao để được tốt nhất nhưng vì kiến thức còn hạn hẹp và thời gian có hạn nên nhóm chưa thể tối ưu hết các chức năng trong đồ án cũng như không thể tránh khỏi các thiếu sót. Vì vậy rất mong nhận được sự góp ý của cô để đồ án của nhóm có thể hoàn thiện hơn. Một lần nữa, nhóm thực hiện chân thành cảm ơn cô Từ Tuyết Hồng và các bạn.

Nhóm của em xin cam đoan tất cả đều là quá trình học tập, nghiên cứu các giải thuật đã yêu cầu thực hiện cho các giải thuật. Nhóm không đạo văn và đạo code, nếu có nhóm xin chịu trách nhiệm hoàn toàn về các vấn đề xảy ra về sau.

Mục Lục

[**CHƯƠNG 1: TỔNG QUAN VỀ NAIVE BAYES** 2](#_Toc58461754)

[1. Định nghĩa: 2](#_Toc58461755)

[2. Các mô hình thuật toán Naive Bayes: 3](#_Toc58461756)

[2.1. Multinomial Naive Bayes: 3](#_Toc58461757)

[2.2. Bernoulli Naive Bayes: 4](#_Toc58461758)

[2.3. Gaussian Naive Bayes: 4](#_Toc58461759)

[3. Kết luận: 5](#_Toc58461760)

[3.1. Ưu điểm: 5](#_Toc58461761)

[3.2. Nhược điểm: 5](#_Toc58461762)

[**CHƯƠNG 2: DEMO GAUSSIAN NAIVE BAYES** 6](#_Toc58461763)

[1. Thư viện: 6](#_Toc58461764)

[2. Chương trình : 7](#_Toc58461765)

[3.Tài liệu tham khảo: 13](#_Toc58461766)

# **CHƯƠNG 1: TỔNG QUAN VỀ NAIVE BAYES**

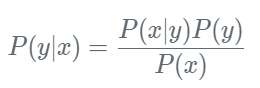
## 1. Định nghĩa:

Định lý Bayes (Bayes’ Theorem) là một định lý toán học để tính xác suất xảy ra của một sự kiện ngẫu nhiên A khi biết sự kiện liên quan B đã xảy ra.

Định lý này đặt theo tên nhà toán học Thomas Bayes, người Anh sống ở thế kỷ 18. Đây là một trong những công cụ vô cùng hữu ích, người bạn thân của các Data Scientist, những người làm trong ngành khoa học dữ liệu.

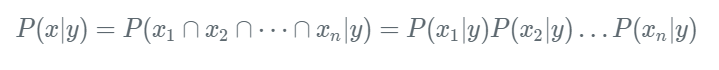
Naive Bayes Classification (NBC) là một thuật toán phân loại dựa trên tính toán xác suất áp dụng định lý Bayes, thuật toán này thuộc nhóm Supervised Learning.

Theo định lý Bayes, ta có công thức tính xác suất ngẫu nhiên của sự kiện y khi biết x như sau:

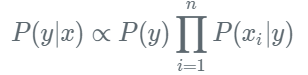
(1)

Giả sử ta phân chia 1 sự kiện x thành n thành phần khác nhau .

Naive Bayes theo đúng như tên gọi dựa vào một giả thiết ngây thơ rằng là các thành phần độc lập với nhau. Từ đó ta có thể tính được:

(2)

Do đó ta có:

 (3)

Trên thực tế thì ít khi tìm được dữ liệu mà các thành phần là hoàn toàn độc lập với nhau. Tuy nhiên giả thiết này giúp cách tính toán trở nên đơn giản, training data nhanh, đem lại hiệu quả bất ngờ với các lớp bài toán nhất định.

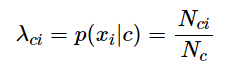
Cách xác định các thành phần (class) của dữ liệu dựa trên giả thiết này có tên là Naive Bayes Classifier.

## 2. Các mô hình thuật toán Naive Bayes:

### 2.1. 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. 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:

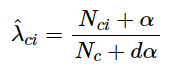
(4)

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 , từ đó .

Cách tính này có một hạn chế là nếu có một từ mới chưa bao giờ xuất hiện trong class *c* thì biểu thức (4) sẽ bằng 0, điều này dẫn đến vế phải của (3) bằng 0 bất kể các giá trị còn lại có lớn thế nào. Việc này sẽ dẫn đến kết quả không chính xác.

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

(5)

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 .

Như vậy, mỗi class *c* sẽ được mô tả bởi bộ các số dương có tổng bằng 1: .

### 2.2. Bernoulli Naive Bayes:

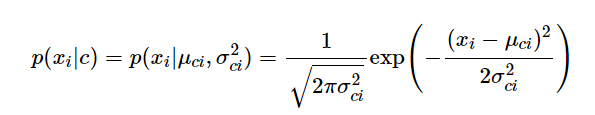
Mô hình này được sử dụng khi các đặc trưng đầu vào chỉ nhận giá trị nhị phân 0 hoặc 1 (phân bố Bernoulli).

(6)

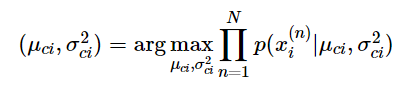
Với p(i|c) có thể được hiểu là xác suất từ thứ *i* xuất hiện trong các văn bản của class *c.*

### 2.3. Gaussian Naive Bayes:

Khi các đặc trưng nhận giá trị liên tục, ta giả sử các đặc trưng đó có phân phối Gaussian.

(7)

Trong đó:

(8)

## 3. Kết luận:

### 3.1. Ưu điểm:

- Giả định độc lập: hoạt động tốt cho nhiều bài toán/miền sữ liệu và ứng dụng. Đơn giản nhưng đủ tốt để giải quyết nhiều bài toán như phân lớp văn bản, lọc spam,..

- Cho phép kết hợp tri thức tiền nghiệm (prior knowledge) và dữ liệu quan sát được (obserwed data). Tốt khi có sự chệnh lệch số lượng giữa các lớp phân loại.

- Huấn luyện mô hình (ước lượng tham số) dễ và nhanh.

### 3.2. Nhược điểm:

- Giả định độc lập (ưu điểm cũng chính là nhược điểm) hầu hết các trường hợp thực tế trong đó có các thuộc tính trong các đối tượng thường phụ thuộc lẫn nhau.

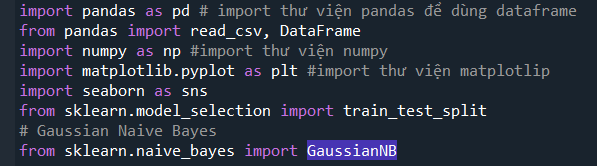
- Vấn đề zero

- Mô hình không được huẩn luyện bằng phượng pháp tối ưu mạnh và chặt chẽ. Tham số mủa mô hình là các ước lượng xác suất điều kiện đơn lẻ. Không tính đến sự tương tác giữa các ước lượng này. nhau.

# **CHƯƠNG 2: DEMO GAUSSIAN NAIVE BAYES**

## 1. Thư viện:

Dự đoán liệu thu nhập có vượt quá 50 nghìn đô la / năm hay không dựa trên dữ liệu điều tra dân số.



Thư viện pandas trong python là một thư viện mã nguồn mở, hỗ trợ đắc lực trong thao tác dữ liệu. Đây cũng là bộ công cụ phân tích và xử lý dữ liệu mạnh mẽ của ngôn ngữ lập trình python. Thư viện này được sử dụng rộng rãi trong cả nghiên cứu lẫn phát triển các ứng dụng về khoa học dữ liệu. Thư viện này sử dụng một cấu trúc dữ liệu riêng là Dataframe. Pandas cung cấp rất nhiều chức năng xử lý và làm việc trên cấu trúc dữ liệu này.

Numpy là một thư viện lõi phục vụ cho khoa học máy tính của Python, hỗ trợ cho việc tính toán các mảng nhiều chiều, có kích thước lớn với các hàm đã được tối ưu áp dụng lên các mảng nhiều chiều đó. Numpy đặc biệt hữu ích khi thực hiện các hàm liên quan tới Đại Số Tuyến Tính.

Để thực hiện các suy luận thống kê cần thiết, cần phải trực quan hóa dữ liệu của bạn và Matplotlib là một trong những giải pháp như vậy cho người dùng Python. Nó là một thư viện vẽ đồ thị rất mạnh mẽ hữu ích cho những người làm việc với Python và NumPy.

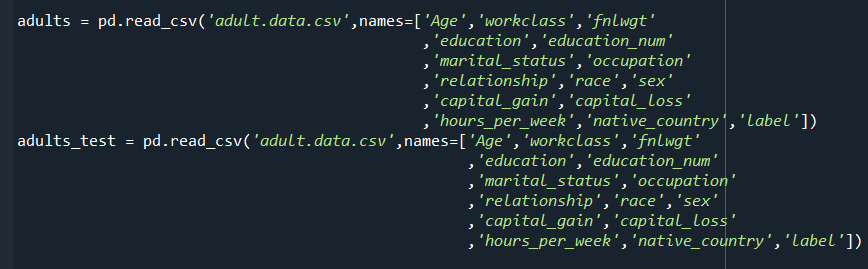
Seaborn là một trong những thư viện Python được đánh giá cao nhất thế giới được xây dựng nhằm mục đích tạo ra các hình ảnh trực quan đẹp mắt. Nó có thể được coi là một phần mở rộng của một thư viện khác có tên là Matplotlib vì nó được xây dựng trên đó.

sklearn.model\_selection.train\_test\_split tạo 4 phần dữ liệu sẽ được sử dụng để điều chỉnh và dự đoán các giá trị.

GaussianNB: được sử dụng để phân loại và giả định rằng, tất cả các thuộc tính đều tuân theo phân phối chuẩn.

## 2. Chương trình :

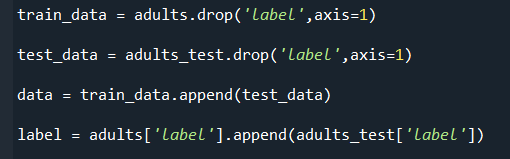
Đọc file adult.data.csv bằng thư viện Pandas.ReadCSV và đặt tên từng cột dữ liệu tương ứng.



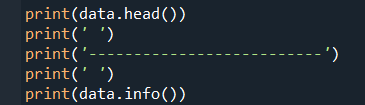
Xóa cột ‘label’ bằng cách sử dụng .drop() với axis=1.

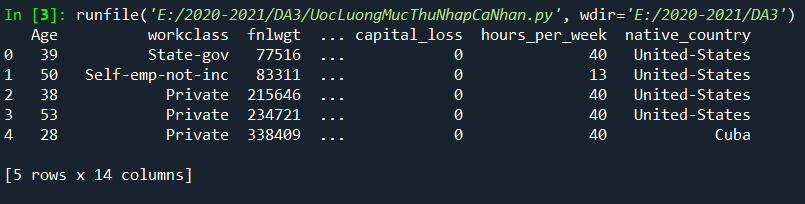
Gộp 2 list train\_data và test\_data thành 1 list data bằng cách sử dụng append() trong Python, đây là phương thức dùng để thêm một phần tử vào vị trí cuối cùng của List hiện tại, nó không trả về một List mới mà thay vào đó sẽ cập nhật List hiện tại.

Tương chúng ta cũng gộp cột ‘label’ trong adults và adults\_test thành 1 list label

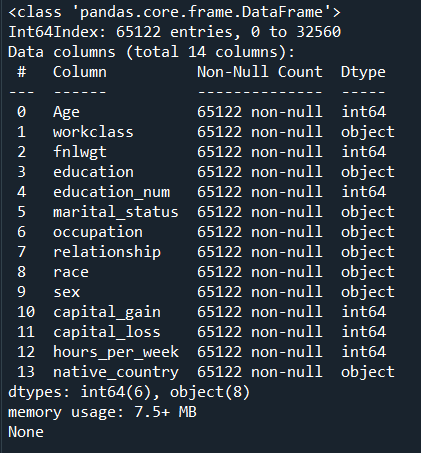


Sau đó hiển thị những dòng dữ liệu đầu để xem kết quả, ta được bảng dữ liệu:

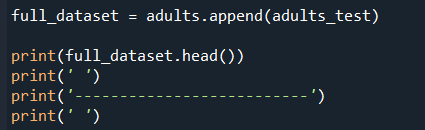


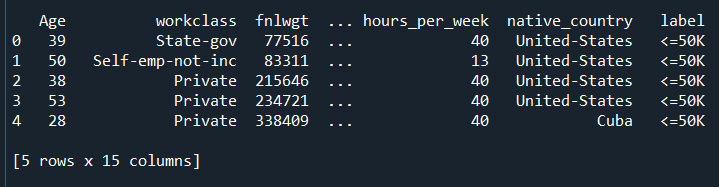


Sau đó dùng data.info() để hiện được thông tin của dataframe:



Gộp 2 list adults và adults\_test thành 1 list full\_dataset. Sau đó hiển thị những dòng dữ liệu đầu để xem kết quả

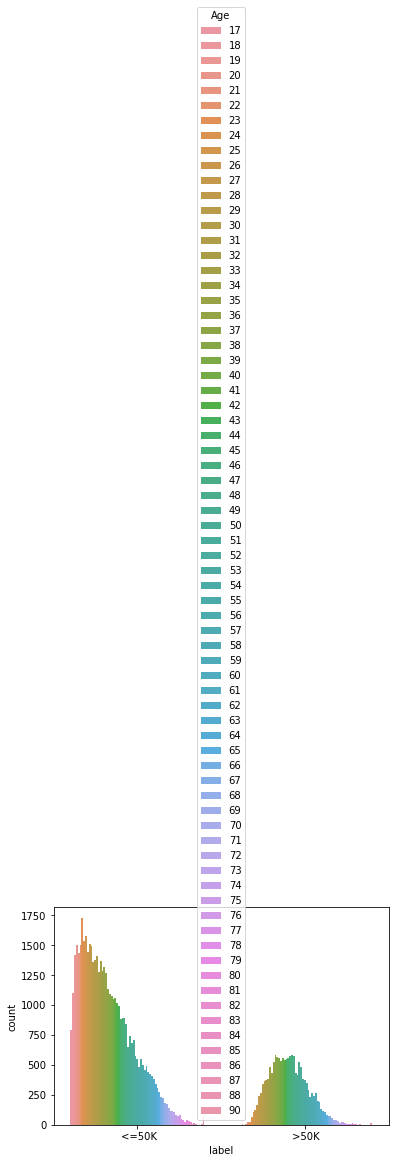




Dùng thư viện SNS để thực hiện vẽ các cột dữ liệu so sánh từng độ tuổi ứng với mức thu nhập tương ứng với hàm countplot của thư viện Seaborn

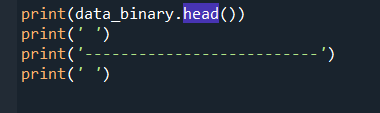


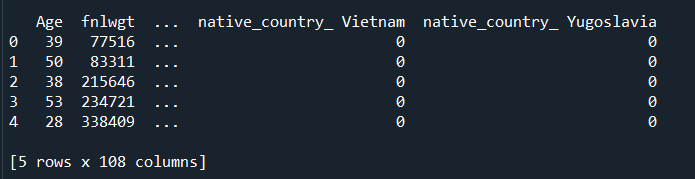
Kết quả:



Chuyển đổi biến phân loại thành chỉ số. Sau đó hiển thị những dòng dữ liệu đầu để xem kết quả

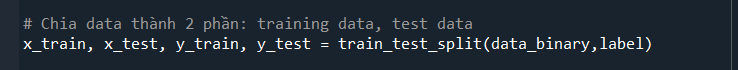






Sử dụng sklearn.model\_selection.train\_test\_split tạo 4 phần dữ liệu sẽ được sử dụng để điều chỉnh và dự đoán các giá trị.

Chia data thành 2 phần: training data, test data từ 2 list data\_binary và label



Khai báo và tạo một bộ phân loại Gaussian



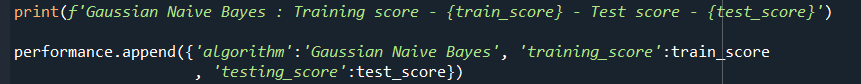
Cung cấp dữ liệu đào tạo vào mô hình



Trả lại độ chính xác trung bình trên các nhãn và dữ liệu thử nghiệm đã cho.



In ra độ chính xác của mô hình Gaussian Naive Bayes gồm độ chính xác training và độ chính xác testing





## 3.Tài liệu tham khảo:

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

<https://viblo.asia/p/thuat-toan-phan-lop-naive-bayes-924lJWPm5PM>

<https://bitly.com.vn/jcdjal>