

# Bài 4

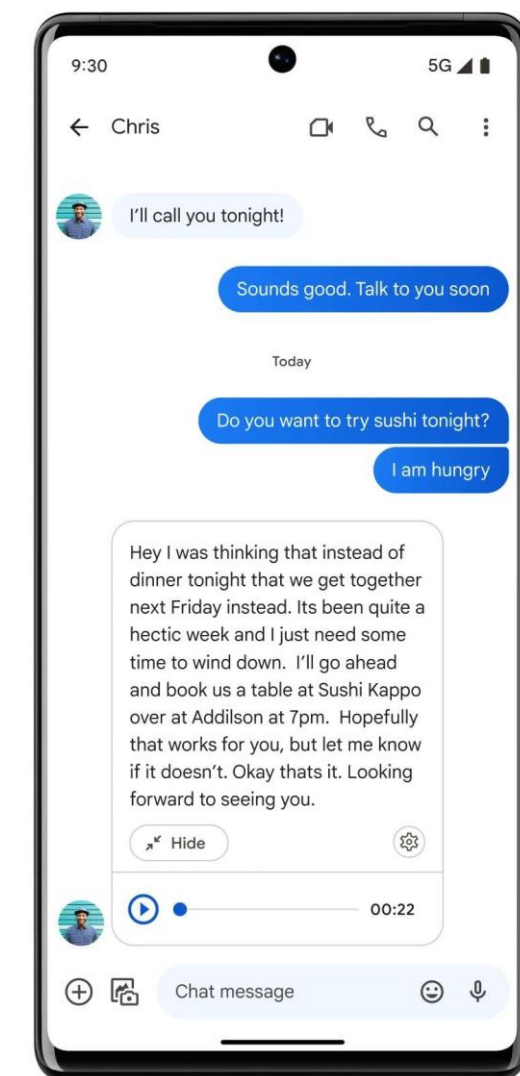
# Thuật toán Naive Bayes

Giảng viên: TS. Nguyễn Ngọc Giang

SĐT: 0862411011



Email: [giangnn.cntt@dainam.edu.vn](mailto:giangnn.cntt@dainam.edu.vn)

- Ý tưởng thuật toán: **Nhận một tin nhắn của một bạn chưa xác định trong nhóm và dự đoán xem tin nhắn đó của ai viết?**



- $P(A|B)$ : xác suất của A khi biết B
- $P(A)$ : xác suất xảy ra của A
- $P(B|A)$ : xác suất của B khi biết A
- $P(B)$ : xác suất xảy ra của B



$$P(A | B) = \frac{P(B | A)P(A)}{P(B)}$$


- $P(\text{lửa})$ : Xác suất có lửa;  $P(\text{khói})$ : Xác suất có khói
- $P(\text{lửa}|\text{khói})$ : Xác suất có lửa khi có khói
- $P(\text{khói}|\text{lửa})$ : Xác suất nhìn thấy khói khi có lửa
- XS đám cháy nguy hiểm là 1%; XS thấy khói là 10% và 90% đám cháy tạo ra khói

$$P(\text{lửa}|\text{khói}) = \frac{P(\text{lửa}) * P(\text{khói}|\text{lửa})}{P(\text{khói})} = \frac{1\% * 90\%}{10\%} = 9\%$$





### Biến ngẫu nhiên

- Trong lí thuyết xác suất, sự kiện được hiểu như là một sự việc, một hiện tượng nào đó của cuộc sống tự nhiên và xã hội.
- Phép thử ngẫu nhiên (hay còn gọi là phép thử) là một hành động hay thí nghiệm mà ta không đoán trước được kết quả của nó, tuy nhiên có thể xác định được tập hợp tất cả các kết quả có thể có của phép thử đó.

**Ví dụ.** Gieo một con xúc xắc đồng chất trên một mặt phẳng (phép thử). Phép thử này có 6 kết quả là: xuất hiện mặt 1 chấm, mặt 2 chấm, ..., mặt 6 chấm.

Mỗi kết quả này cùng với các kết quả phức tạp hơn như: xuất hiện mặt có số chấm là số nguyên tố, mặt có số chấm chẵn, mặt có số chấm là bội của 2, đều có thể coi là các sự kiện.

Sự kiện được gọi là tất yếu, nếu nó chắc chắn xảy ra, và được gọi là bất khả, nếu nó không thể xảy ra khi thực hiện phép thử.

Còn nếu sự kiện có thể hoặc không xảy ra sẽ được gọi là sự kiện ngẫu nhiên.

## Xác Suất

Giả sử ta tiến hành  $n$  phép thử với cùng một hệ điều kiện thấy có  $n_A$  lần xuất hiện sự kiện  $A$ . Số  $n_A$  được gọi là tần số xuất hiện sự kiện  $A$ . Khi đó xác suất của sự kiện  $A$ , kí hiệu là  $P(A)$ , được cho bởi:

$$P(A) = \frac{n_A}{n}$$

**Ví dụ.** Trong hộp đựng 20 viên bi gồm 14 viên màu đỏ và 6 viên màu trắng.

- Lấy ngẫu nhiên 1 viên bi. Tính xác suất để viên bi lấy được là viên màu trắng.
- Lấy ngẫu nhiên (không hoàn lại) 5 viên bi từ trong hộp. Tính xác suất để trong 5 viên bi lấy ra có 3 viên đỏ. Biết rằng các viên bi giống nhau.

$$P(3\text{đỏ}, 2\text{trắng}) = \frac{C_{14}^3 * C_6^2}{C_{20}^5}$$

$$C_n^k = \frac{n!}{k! * (n - k)!}$$

### Cộng xác suất

- **Định lý.** Nếu  $A, B$  là 2 biến cố xung khắc thì  $P(A+B)=P(A)+P(B)$ .
- **Chú ý:** Có thể mở rộng ra cho trường hợp  $n$  biến cố.
- **Hệ quả.** Nếu  $n$  biến cố  $A_1, A_2, \dots, A_n$  xung khắc và đầy đủ thì:  $\sum_{i=1}^n P(A_i)=1$ .  
Với mọi biến cố  $A$ , ta có:  $P(A^c)=1-P(A)$ .
- **Định lý.** Cho  $A, B$  là 2 biến cố bất kỳ thì ta có:  $P(A+B)=P(A)+P(B)-P(AB)$ .
- **Ví dụ.** Trong số 60 học viên của một lớp có 20 học viên học khá môn Toán, 30 học viên học khá môn Vật lý và 10 học viên học khá cả 2 môn học trên. Chọn ngẫu nhiên một học viên từ lớp trên. Tính xác suất để học viên được chọn học khá ít nhất 1 trong 2 môn học.

## ***Xác suất có điều kiện***

- **Định nghĩa.** Xác suất của biến cố A được xác định với điều kiện biến cố B đã xảy ra được gọi là xác suất có điều kiện của biến cố A, ký hiệu là  $P(A/B)$  hoặc  $P(A|B)$ .

**Ví dụ:** Trong hộp đựng 20 viên bi gồm 14 viên màu đỏ và 6 viên màu trắng.

- Lấy ngẫu nhiên 1 viên bi. Tính xác suất để viên bi lấy được là viên màu đỏ.
- Sau khi đã lấy được 1 viên bi màu đỏ. Lấy ngẫu nhiên tiếp 1 viên bi. Tính xác suất để viên bi lấy được là viên màu đỏ.

- **Hệ quả.** Hai biến cố A,B được gọi là độc lập nếu thỏa mãn một trong các trường hợp sau đây:

$$P(A|B) = P(A), \quad P(B|A) = P(B)$$



## ***Nhân xác suất***

- **Định lý.** Với A,B là hai biến cố bất kỳ:

$$P(AB) = P(A) * P(B|A) = P(B)P(A|B)$$

- **Hệ quả.** Với hai biến cố A,B là độc lập ta có

$$P(AB) = P(A) * P(B)$$

- **Ví dụ.** Một tiểu đội thông tin quản lý 3 máy vô tuyến, xác suất để trong một ngày làm việc các máy bị hỏng tương ứng là: 0,1; 0,2; 0,3. Tìm xác suất để trong một ngày làm việc có:
- Đúng 1 máy bị hỏng.
  - Ít nhất 1 máy bị hỏng.

## Công thức Bayes

$$P(A|B) = \frac{P(B|A) * P(A)}{P(B)}$$

$$posterior\_probability = \frac{likelihood * class\_prior\_probability}{predictor\_prior\_probability}$$

### Tổng quát:

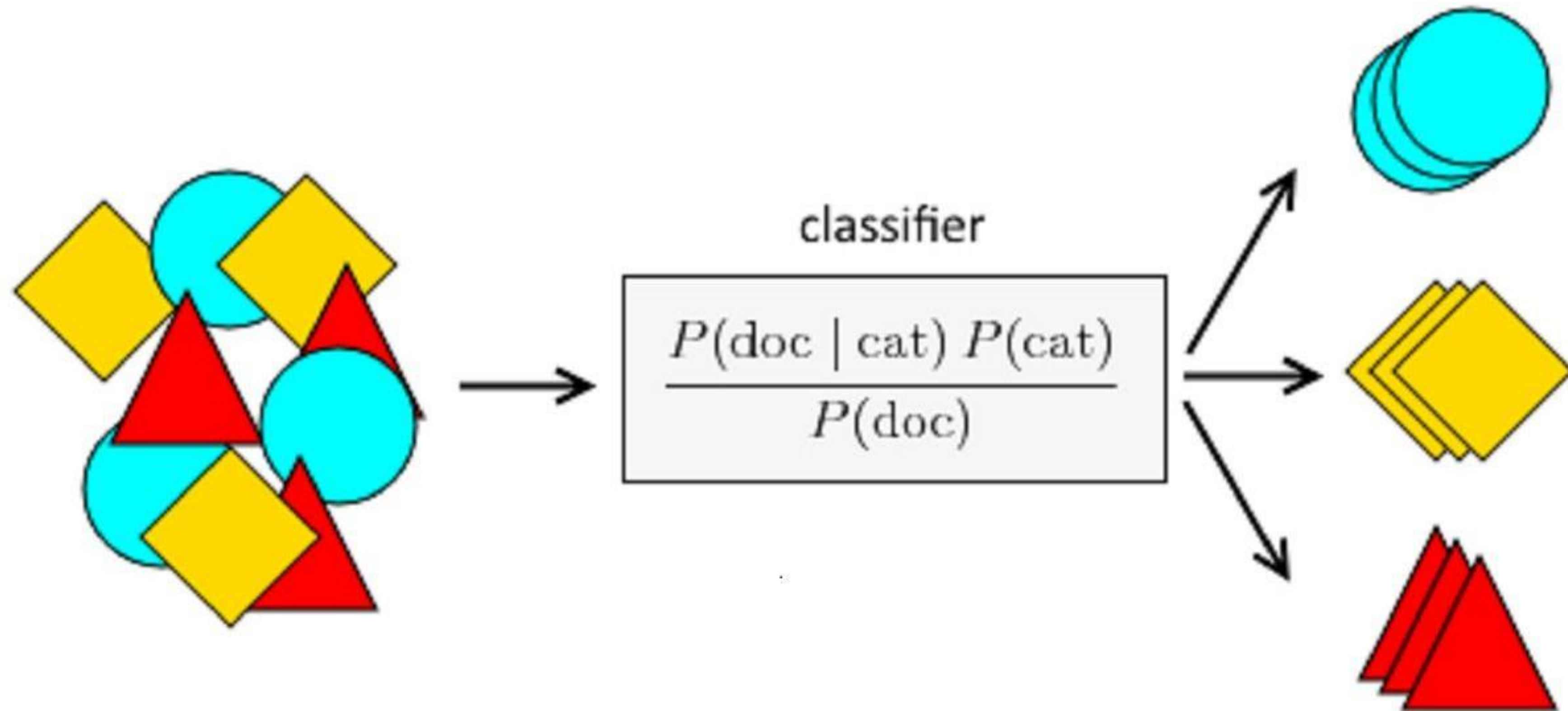
$$P(A_k|B) = \frac{P(A_k) * P(B|A_k)}{P(B)} = \frac{P(A_k) * P(B|A_k)}{\sum_{i=1}^n P(A_i) * P(B|A_i)}$$

**Ví dụ.** Dây chuyền lắp ráp máy vô tuyến điện gồm các linh kiện là sản phẩm từ 2 nhà máy sản xuất ra. Số linh kiện nhà máy 1 sản xuất chiếm 55%, số linh kiện nhà máy 2 sản xuất chiếm 45%; tỷ lệ sản phẩm đạt tiêu chuẩn của nhà máy 1 là 90%, nhà máy 2 là 87%. Lấy ngẫu nhiên ra 1 linh kiện từ dây chuyền lắp ráp đó ra kiểm tra thì được kết quả linh kiện đạt chuẩn. Tìm xác suất để linh kiện đó do nhà máy 1 sản xuất?

- $P(\text{lửa})$ : Xác suất có lửa;  $P(\text{khói})$ : Xác suất có khói
- $P(\text{lửa}|\text{khói})$ : Xác suất có lửa khi có khói
- $P(\text{khói}|\text{lửa})$ : Xác suất nhìn thấy khói khi có lửa
- **XS đám cháy nguy hiểm là 1%; XS thấy khói là 10% và 90% đám cháy tạo ra khói**

$$P(\text{lửa}|\text{khói}) = \frac{P(\text{lửa}) * P(\text{khói}|\text{lửa})}{P(\text{khói})} = \frac{1\% * 90\%}{10\%} = 9\%$$





<https://www.linkedin.com/pulse/naive-bayes-theorem-machine-learning-rohit-bele>



**D: TrainSet;  $C(C_1, C_2, \dots, C_m)$ : Tập các lớp**

**$X(x_1, x_2, \dots, x_n)$ : Dữ liệu cần phân lớp**

**1. Phương pháp tìm lớp  $C_k$  tập  $X$  được dự báo thuộc:**

$$P(C_k/X) = \text{Max}(P(C_i/X)) \text{ với } i=1,2,\dots,m$$

**2. Phương pháp tìm xác suất lớn nhất:**

$$\text{Max}(P(X|C_i) * P(C_i))$$

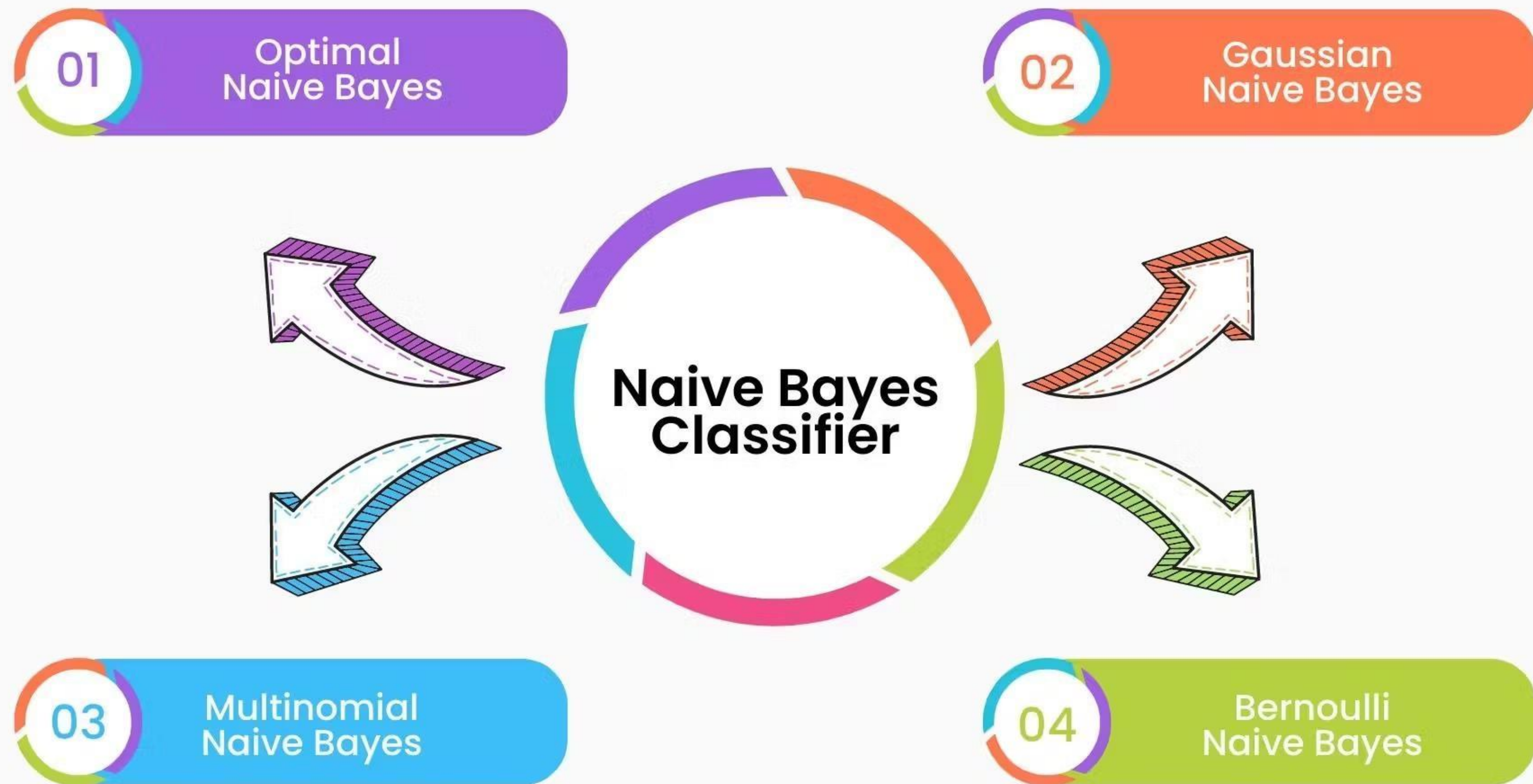
**3. Khi số thuộc tính nhiều có thể tính bằng cách:**

$$P(X/C_i) = P(x_1/C_i) * P(x_2/C_i) * \dots * P(x_n/C_i)$$

The Naive Bayes Algorithm is one of the crucial algorithms in machine learning that helps with classification problems. It is derived from Bayes' probability theory and is used for text classification, where you train high-dimensional datasets. Some best examples of the Naive Bayes Algorithm are sentimental analysis, classifying new articles, and spam filtration.

Classification algorithms are used for categorizing new observations into predefined classes for the uninitiated data. The Naive Bayes Algorithm is known for its simplicity and effectiveness. It is faster to build models and make predictions with this algorithm. While creating any ML model, it is better to apply the Bayes theorem. Application of Naive Bayes Algorithms requires the involvement of [expert ML developers](#).

[\*https://www.turing.com/kb/an-introduction-to-naive-bayes-algorithm-for-beginners\*](https://www.turing.com/kb/an-introduction-to-naive-bayes-algorithm-for-beginners)



**D: TrainSet;  $C(C_1, C_2, \dots, C_m)$ : Tập các lớp**

**$X(x_1, x_2, \dots, x_n)$ : Dữ liệu cần phân lớp**

**1. Phương pháp tìm lớp  $C_k$  tập  $X$  được dự báo thuộc:**

$$P(C_k/X) = \text{Max}(P(C_i/X)) \text{ với } i=1,2,\dots,m$$

**2. Phương pháp tìm xác suất lớn nhất:**

$$\text{Max}(P(X|C_i) * P(C_i))$$

**3. Khi số thuộc tính nhiều có thể tính bằng cách:**

$$P(X/C_i) = P(x_1/C_i) * P(x_2/C_i) * \dots * P(x_n/C_i)$$

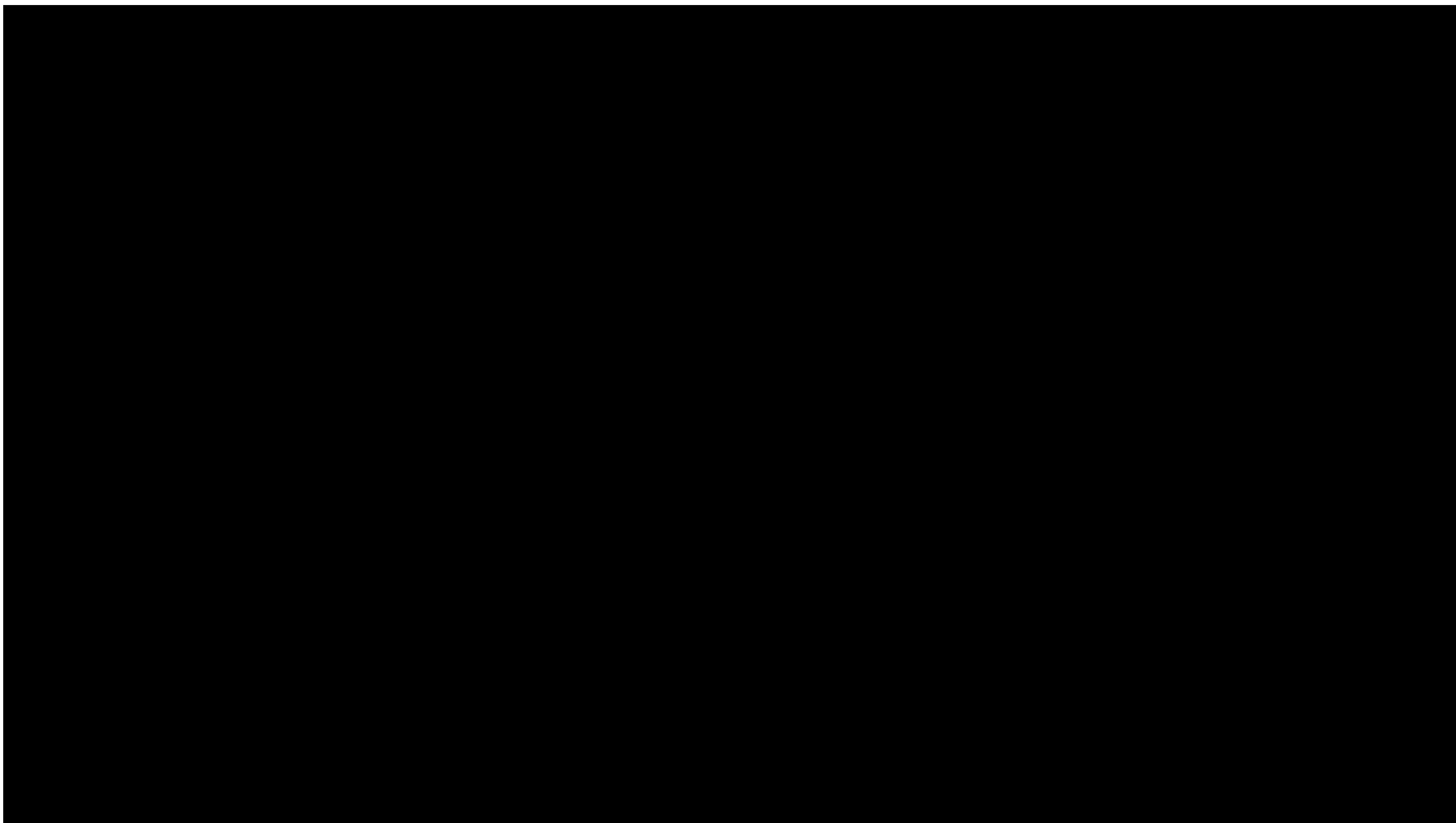
Cách tính  
 $P(x_j/C_i)$



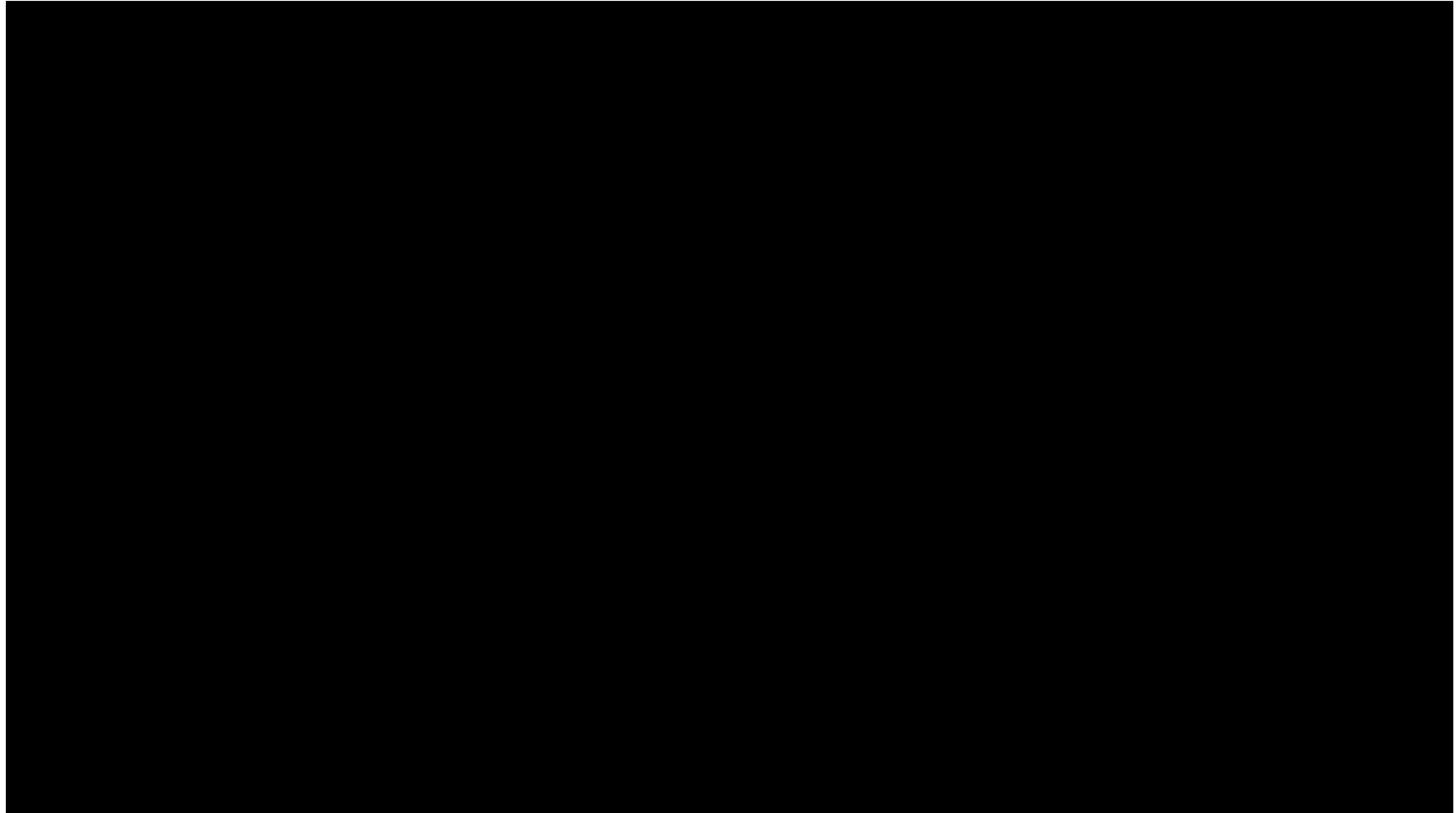
- **Optimal Naive Bayes**, ý tưởng là cải thiện hiệu quả của Naive Bayes bằng cách điều chỉnh các tham số hoặc cấu trúc của mô hình để đạt được kết quả tốt nhất có thể.
- **Gaussian Naive Bayes** là một biến thể của thuật toán Naive Bayes, được sử dụng đặc biệt trong các tình huống mà các đặc trưng của dữ liệu được giả định tuân theo phân phối Gaussian (hay còn gọi là phân phối chuẩn).

- **Multinomial Naive Bayes** là một biến thể của thuật toán Naive Bayes, đặc biệt thích hợp cho các bài toán phân loại mà đặc trưng của dữ liệu là số lượng các sự kiện (counts) hoặc tần suất xuất hiện của các từ trong văn bản.
- **Bernoulli Naive Bayes** là một biến thể của thuật toán Naive Bayes, được thiết kế đặc biệt để làm việc với dữ liệu nhị phân hoặc dữ liệu mà các đặc trưng là các biến nhị phân (binary features).

<https://www.youtube.com/watch?v=O2L2Uv9pdDA>



<https://www.youtube.com/watch?v=H3EjCKtIVog>





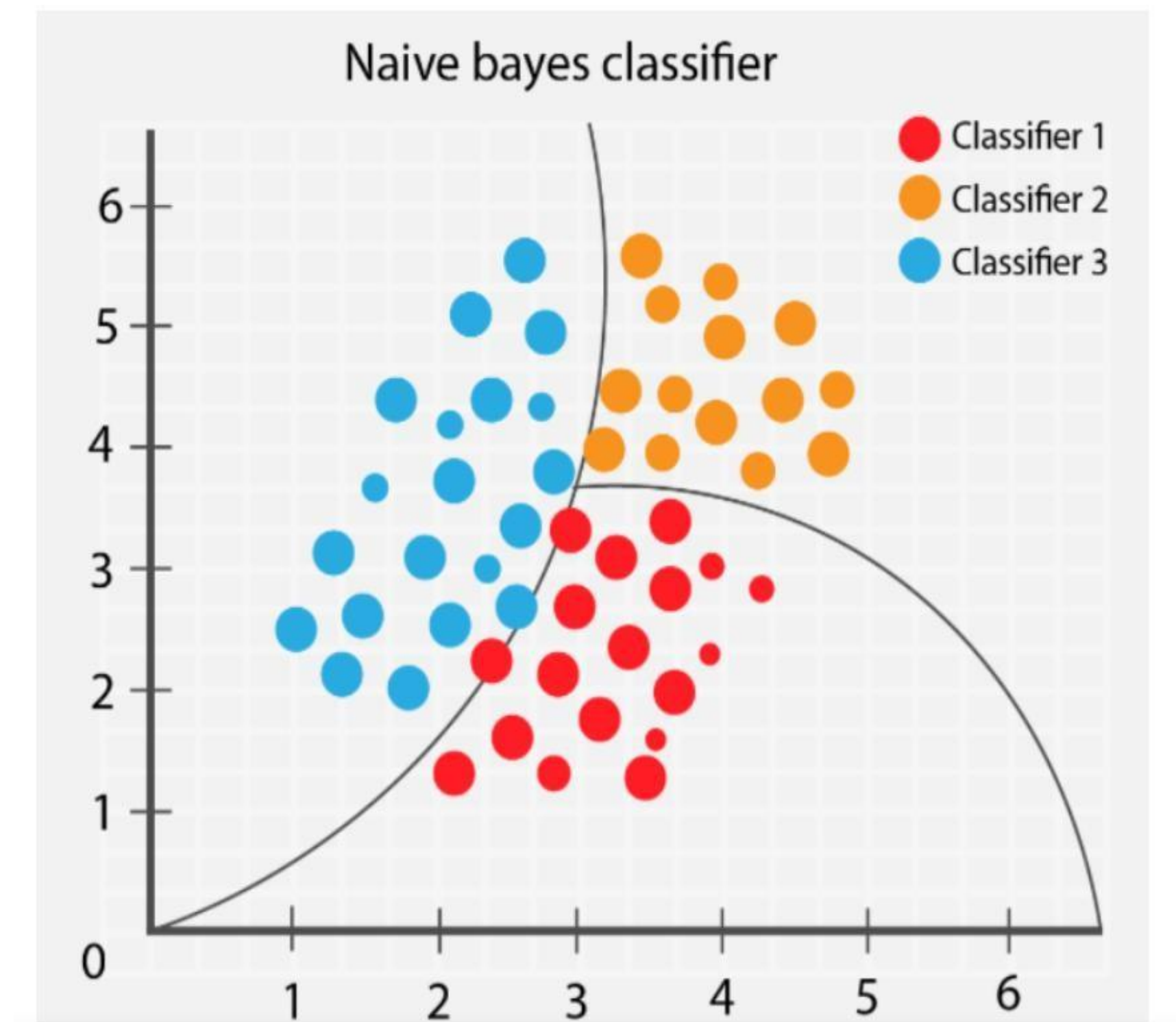
- **Ưu điểm**

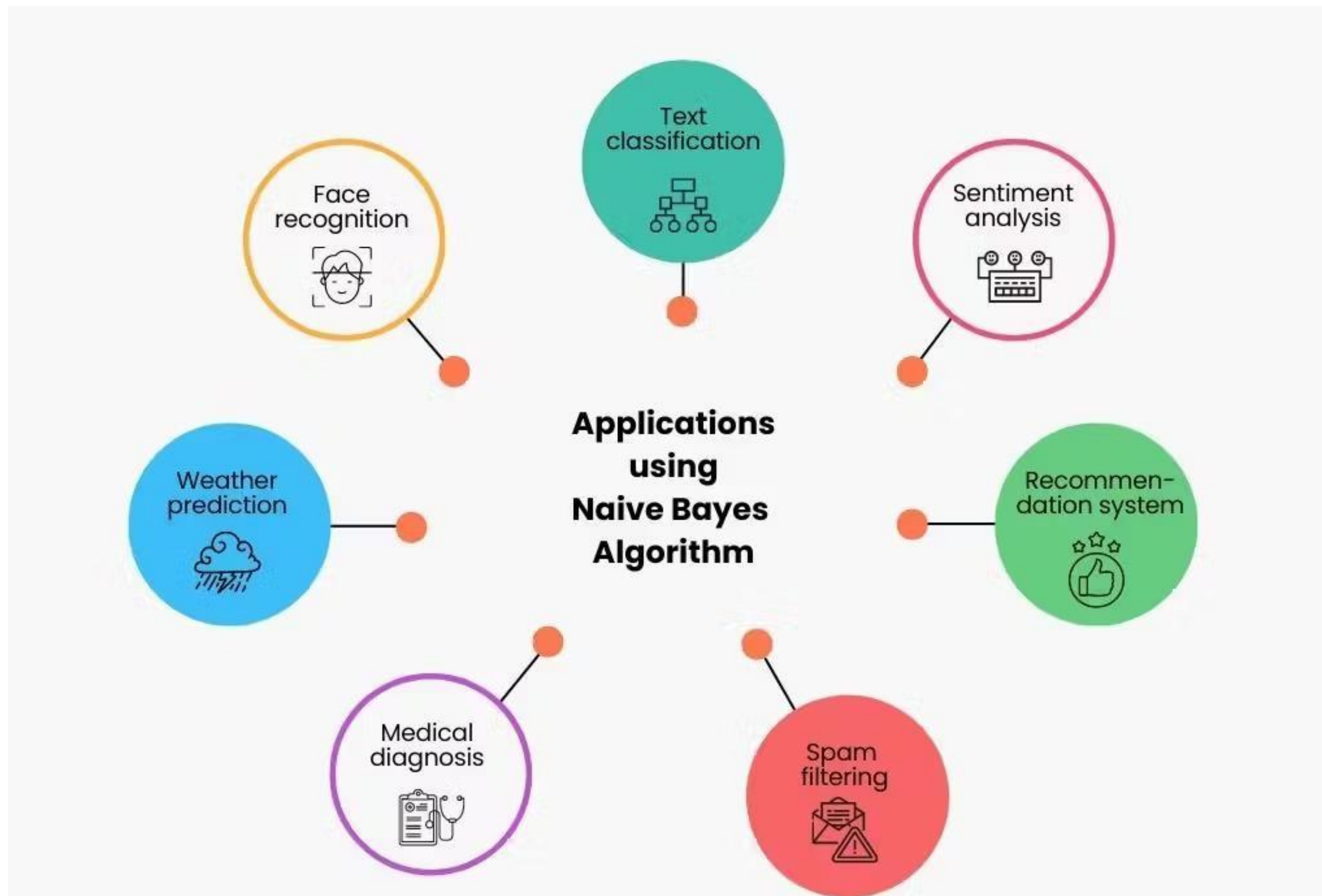
- Không yêu cầu quá nhiều dữ liệu huấn luyện
- Cài đặt đơn giản
- Có thể xử lý được cả dữ liệu số và dữ liệu nhãn
- Không quá nhạy cảm với dữ liệu ngoại lệ
- Tốc độ dự đoán nhanh

- **Nhược điểm**

- Vấn đề “tần suất bằng 0”  $\Rightarrow$  sử dụng phương pháp smoothing
- Có giả định tất cả các thuộc tính là độc lập với nhau

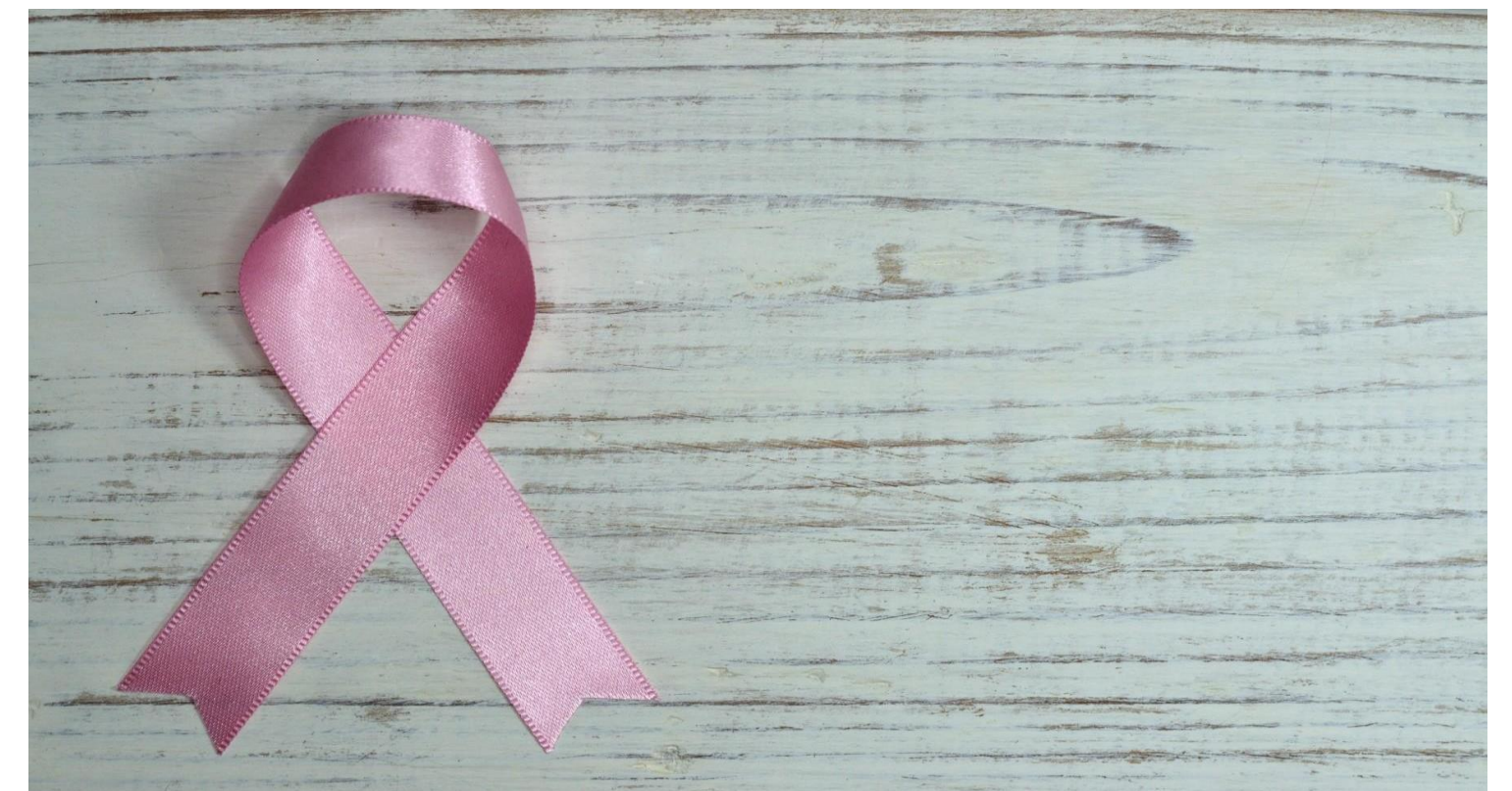
- Dự đoán với thời gian thực
- Phân loại Text/Lọc thư rác
- Hệ thống Recommendation







- Phân các bệnh nhân thành 2 lớp **ung thư** và **không ung thư**. Giả sử xác suất để một người bị ung thư là 0.008 tức là  $P(\text{cancer}) = 0.008$ ; và  $P(\text{nocancer}) = 0.992$ . Xác suất để bệnh nhân ung thư có kết quả xét nghiệm dương tính là 0.98 và xác suất để bệnh nhân không ung thư có kết quả dương tính là 0.03 tức là  $P(+|\text{cancer}) = 0.98$ ,  $P(+|\text{nocancer}) = 0.03$ . **Giả sử một bệnh nhân có kết quả xét nghiệm dương tính, xác định xem có mắc bệnh ung thư không?**



**X = (age = youth, income = medium, student = yes, credit\_rating = fair)**

ID	Age	Income	Student	credit_rating	Buy laptop
1	youth	high	no	fair	no
2	youth	high	no	excellent	no
3	middle	high	no	fair	yes
4	senior	medium	no	fair	yes
5	senior	low	yes	fair	yes
6	senior	low	yes	excellent	no
7	middle	low	yes	excellent	yes
8	youth	medium	no	fair	yes
9	youth	low	yes	fair	yes
10	senior	medium	yes	fair	yes
11	youth	medium	yes	excellent	yes
12	middle	medium	no	excellent	yes
13	middle	high	yes	fair	yes
14	senior	medium	no	excellent	no

**X = (Overcast, Mild, High)**

Outlook	Temperature	Humidity	Windy	Play
0	Rainy	Hot	High	FALSE
1	Rainy	Hot	High	TRUE
2	Overcast	Hot	High	FALSE
3	Sunny	Mild	High	FALSE
4	Sunny	Cool	Normal	FALSE
5	Sunny	Cool	Normal	TRUE
6	Overcast	Cool	Normal	TRUE
7	Rainy	Mild	High	FALSE
8	Rainy	Cool	Normal	FALSE
9	Sunny	Mild	Normal	FALSE
10	Rainy	Mild	Normal	TRUE
11	Overcast	Mild	High	TRUE
12	Overcast	Hot	Normal	FALSE
13	Sunny	Mild	High	TRUE



# Lập trình với kho dữ liệu tự sinh





```
# Importing the required libraries
import pandas as pd
from sklearn.datasets import load_iris
from sklearn.model_selection import train_test_split
from sklearn.naive_bayes import GaussianNB
from sklearn.naive_bayes import MultinomialNB
from sklearn.naive_bayes import BernoulliNB
from sklearn import metrics

# Loading the Iris dataset
iris = load_iris()
X = iris.data # Features
y = iris.target # Target variable

# Splitting the dataset into training and testing sets
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.3, random_state=42)

# Creating an instance of the Naive Bayes classifier
model = GaussianNB()

# Training the model using the training data
model.fit(X_train, y_train)

# Making predictions on the test data
y_pred = model.predict(X_test)

# Evaluating the model's performance
accuracy = metrics.accuracy_score(y_test, y_pred)
print("Accuracy:", accuracy)
```

# Lập trình với kho dữ liệu Adults

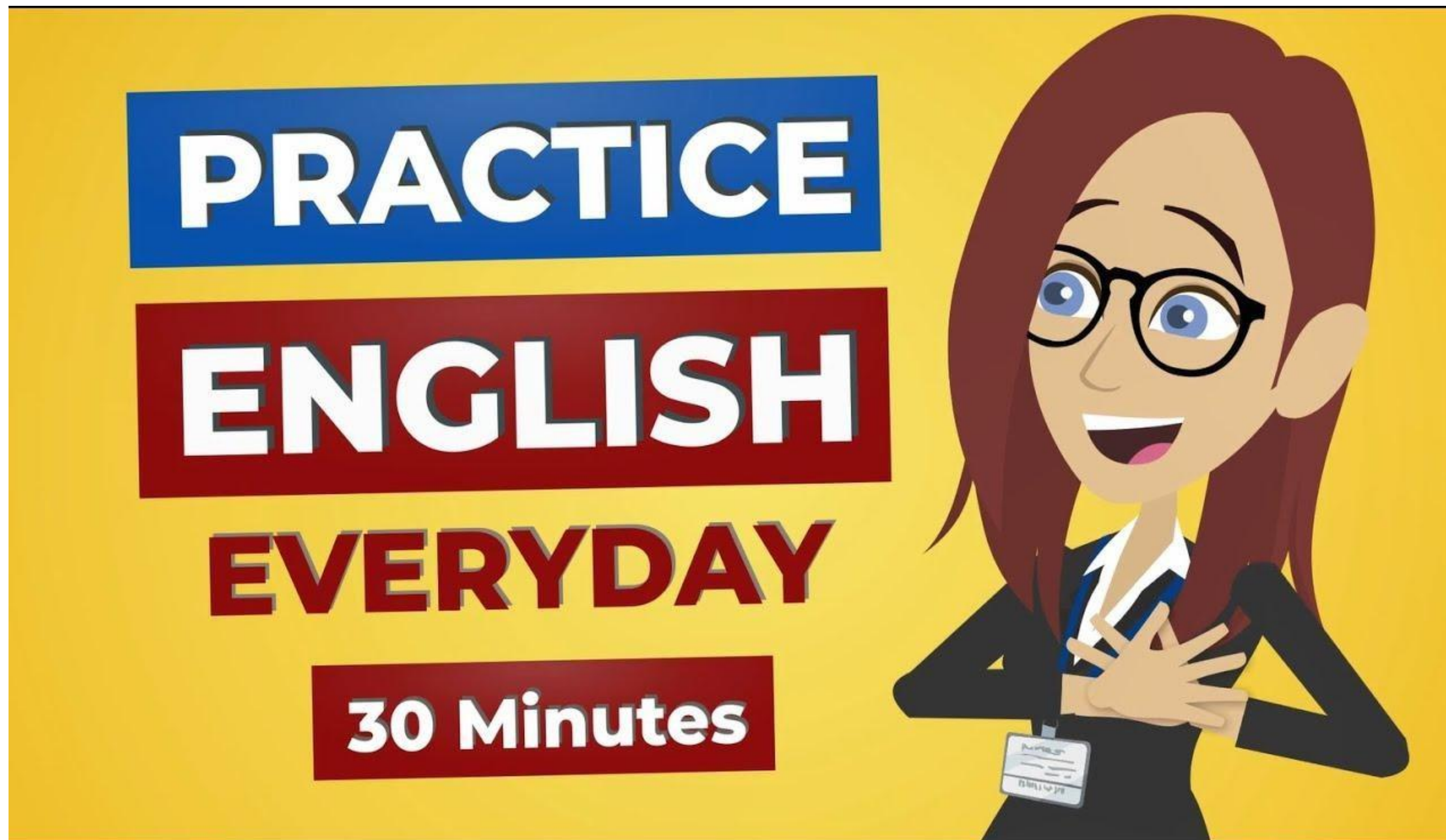
The type column distinguishes between a continuous (c) or discrete (d) attribute.

# feature	original feature	type (c/d)	# processed feat.
1	age	c	[1,5]
2	workclass	d	[6,13]
3	final weight	c	[14,18]
4	education	d	[19,34]
5	ed_num	c	[35,39]
6	marital_status	d	[40,46]
7	occupation	d	[47,60]
8	relationship	d	[61,66]
9	race	d	[67,71]
10	sex	d	[72,73]
11	capital_gain	c	[74,75]
12	capital_loss	c	[76,77]
13	hours × week	c	[78,82]
14	country	d	[83,123]



- Ý tưởng và hoạt động của thuật toán
- Luyện tập lập trình với Naive Bayes







## Naive Bayes Classifiers

Naive Bayes classifiers are a family of classifiers that are quite similar to the linear models discussed in the previous section. However, they tend to be even faster in training. The price paid for this efficiency is that naive Bayes models often provide generalization performance that is slightly worse than that of linear classifiers like `LogisticRegression` and `LinearSVC`.

The reason that naive Bayes models are so efficient is that they learn parameters by looking at each feature individually and collect simple per-class statistics from each feature. There are three kinds of naive Bayes classifiers implemented in `scikit-`

---

`learn`: `GaussianNB`, `BernoulliNB`, and `MultinomialNB`. `GaussianNB` can be applied to any continuous data, while `BernoulliNB` assumes binary data and `MultinomialNB` assumes count data (that is, that each feature represents an integer count of something, like how often a word appears in a sentence). `BernoulliNB` and `MultinomialNB` are mostly used in text data classification.

- Thuật toán Decision Tree
- Bài luyện tập tiếng Anh







*Thank You!*