Ôn tập một số kiến thức Toán cho Machine Learning

Thai Trung Hieu (hieutt42) - Fsoft QAI Quy Nhon

Ngày 22 tháng 4 năm 2022

Một số topic quan trọng

1. Đại số tuyến tính và hình học giải tích - Linear Algebra and Analytic Geometry (ma trận, vector, ...).
2. Giải tích - Calculus (đạo hàm, tích phân)
3. Xác suất - Probability (biến ngẫu nhiên, hàm phân phối, công thức Bayes, ...)

Vectors

Không gian thực $\mathbb{R}^n$ là tập hợp tất cả bộ số có dạng $[x_1, x_2, \ldots, x_n]$ với $x_1, x_2, \ldots, x_n$ là các số thực.
Mỗi phần tử của $\mathbb{R}^n$ được gọi là 1 vector $x = [x_1, x_2, \ldots, x_n]$.
Chú ý: nói riêng trong đại số tuyến tính, vector luôn được ngầm hiểu là viết theo hàng dọc: $\begin{bmatrix} 1 \\ 3 \\ 7 \end{bmatrix}$
Về sau, ta sẽ dùng vector để mô tả input của các model Machine Learning và Deep Learning.

One hot encoding

- Thuật toán Machine Learning chỉ làm việc trên các con số và các vector. Để nói với thuật toán về con mèo, ta cần biểu diễn con mèo bằng 1 con số hay 1 vector nào đó.
- Xét 1 ví dụ. Ta cần xây dựng 1 model nhận diện 3 chữ cái "A", "B" và "C". Một cách đơn giản, ta có thể đặt 1="A", 2="B" và 3="C". Tuy nhiên, chúng ta có đang muốn nói với model rằng sự khác biệt giữa "A" và "C" (3-1=2) lớn hơn sự khác biệt giữa "A" và "B" (2-1=1)?

One hot encoding

Phương pháp one hot encoding đề xuất cách làm như sau.

- Gán "A" với vector $(1,0,0)$.
- Gán "B" với vector $(0,1,0)$.
- Gán "C" với vector $(0,0,1)$.

Trong các slides sau chúng ta sẽ thấy khoảng cách giữa 3 vector trên bằng nhau.
One hot encoding là 1 ứng dụng đơn giản nhưng rất hiệu quả của vector trong Machine Learning.

Matrix

Ma trận (matrix) là 1 bảng chữ nhật gồm các số, kí hiệu hoặc biểu thức được sắp xếp theo hàng và cột. Ma trận vuông là ma trận có số dòng và số cột bằng nhau. Phần tử đứng ở dòng thứ i, cột thứ j được kí hiệu là a_{ij}.
Ví dụ: ma trận

$$\begin{bmatrix} 1 & 2 & 3 & 5 \\ 3 & 4 & 5 & 8 \\ 7 & 8 & 9 & 1 \end{bmatrix}$$

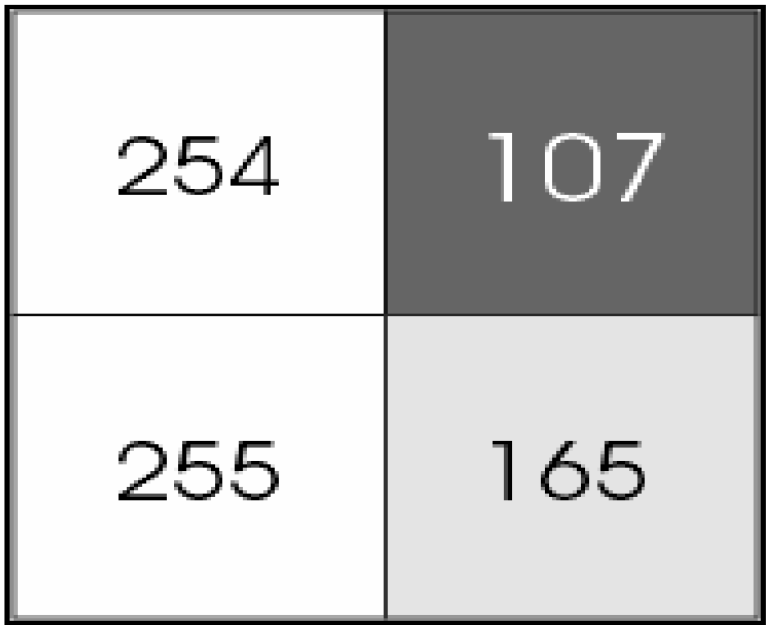
có 3 dòng, 4 cột. $a_{1,2} = 2, a_{3,1} = 7, a_{2,4} = 8$.

Matrices in Machine Learning

- Ma trận có thể được hiểu là 1 tập hợp nhiều vector. Trong Machine Learning, ta thường dùng ma trận để mô tảng data (dạng bảng, dạng ảnh, dạng text, ...) cũng như mô tả các model.
- Để thể hiện các vấn đề toán học trong model, ta cần các phép toán trên ma trận. Ta sẽ tìm hiểu vấn đề này trong các slide tiếp theo.
- Có thể nói ma trận là ngôn ngữ của Machine Learning.

Images as matrices

Mỗi tấm ảnh là 1 ma trận (đối với máy tính). Mỗi 1 điểm ảnh của 1 tấm ảnh đen trắng được mô tả bằng 1 con số từ 0 đến 255. Số càng lớn thì màu càng gần với màu trắng. Số càng nhỏ thì màu càng gần với màu đen.



Adding two matrices

Các phép toán trên ma trận giúp chúng ta mô tả các model Machine Learning một cách dễ dàng. Phép cộng 2 ma trận được thực hiện như sau:

$$\begin{bmatrix} 1 & 2 & 3 \\ 3 & 4 & 5 \\ 7 & 8 & 9 \end{bmatrix} + \begin{bmatrix} a & b & c \\ d & e & f \\ g & h & i \end{bmatrix} = \begin{bmatrix} 1+a & 2+b & 3+c \\ 3+d & 4+e & 5+f \\ 7+g & 8+h & 9+i \end{bmatrix}.$$

Ví dụ:

$$\begin{bmatrix} 1 & 2 & 3 \\ 3 & 4 & 5 \\ 7 & 8 & 9 \end{bmatrix} + \begin{bmatrix} 3 & 2 & 0 \\ -2 & -1 & 2 \\ 1 & 4 & -3 \end{bmatrix} = \begin{bmatrix} 4 & 4 & 3 \\ 1 & 3 & 7 \\ 8 & 12 & 6 \end{bmatrix}.$$

Multiplying a number with a matrix

Phép nhân 1 số k với 1 ma trận được thực hiện như sau:

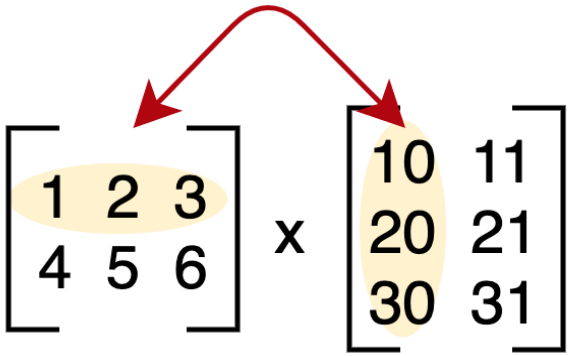
$$k \cdot \begin{bmatrix} a & b & c \\ d & e & f \\ g & h & i \end{bmatrix} = \begin{bmatrix} k \cdot a & k \cdot b & k \cdot c \\ k \cdot d & k \cdot e & k \cdot f \\ k \cdot g & k \cdot h & k \cdot i \end{bmatrix}.$$

Ví dụ

$$2 \cdot \begin{bmatrix} 1 & 2 & 0 \\ 3 & -2 & 4 \\ 5 & -3 & 7 \end{bmatrix} = \begin{bmatrix} 2 & 4 & 0 \\ 6 & -4 & 8 \\ 10 & -6 & 14 \end{bmatrix}.$$

Multiplication of 2 matrices

Phép nhân 2 ma trận được thực hiện như sau:



$$\begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \end{bmatrix} \times \begin{bmatrix} 10 & 11 \\ 20 & 21 \\ 30 & 31 \end{bmatrix}$$

$$= \begin{bmatrix} 1x10 + 2x20 + 3x30 & 1x11 + 2x21 + 3x31 \\ 4x10 + 5x20 + 6x30 & 4x11 + 5x21 + 6x31 \end{bmatrix}$$

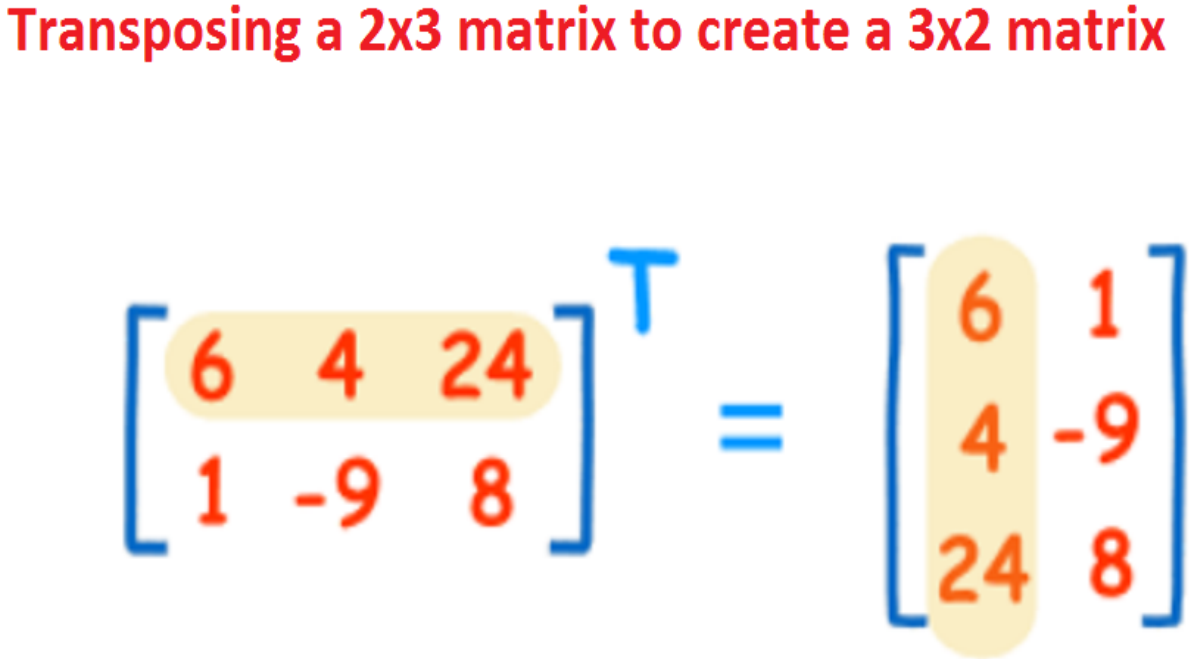
$$= \begin{bmatrix} 10+40+90 & 11+42+93 \\ 40+100+180 & 44+105+186 \end{bmatrix} = \begin{bmatrix} 140 & 146 \\ 320 & 335 \end{bmatrix}$$

Multiplication of 2 matrices

- Phép nhân 2 ma trận không giao hoán. Nghĩa là $A \times B$ có thể khác $B \times A$.
- Không phải lúc nào cũng có thể nhân 2 ma trận với nhau.
- Để thực hiện phép toán $A \times B$, số cột của ma trận A phải bằng số dòng của ma trận B.

Transpose of a matrix

Chuyển vị (transpose) 1 ma trận là chuyển dòng thành cột (hoặc cột thành dòng). Ma trận chuyển vị giúp việc trình bày 1 số công thức toán trở nên đơn giản. Ta sẽ thấy 1 ví dụ ở slide về Scalar product.



The main diagonal of a matrix

Đường chéo chính của 1 ma trận vuông là tập hợp các điểm có chỉ số hàng và cột bằng nhau (các phần tử $a_{i,i}$) (đi từ góc trên bên trái xuống góc dưới bên phải).

$$\begin{bmatrix} a_{11} & 0 & \dots & 0 \\ 0 & a_{22} & \dots & 0 \\ \dots & \dots & \dots & \dots \\ 0 & 0 & \dots & a_{nn} \end{bmatrix}$$

Identity matrix

Ma trận đơn vị I_n là ma trận vuông kích thước $n \times n$ có tất cả các phần tử trên đường chéo chính bằng 1, các phần tử còn lại đều bằng 0.

$$\mathrm{I}_2 = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, \mathrm{I}_3 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}, \mathrm{I}_4 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$A \times \mathrm{I}_n = \mathrm{I}_n \times A = A$ với mọi ma trận vuông A kích thước $n \times n$. Ma trận đơn vị đối với phép nhân ma trận cũng giống như số 1 đối với phép nhân số thông thường.

Inverse of a matrix

Ma trận nghịch đảo của một ma trận vuông A là một ma trận được kí hiệu là A^{-1} thoả mãn

$$A \times A^{-1} = A^{-1} \times A = \mathrm{I}$$

trong đó I là ma trận đơn vị. Hiểu nôm na, ma trận nghịch đảo có vai trò giống như nghịch đảo của 1 số khác 0. Ví dụ: 2 có nghịch đảo là $\frac{1}{2}$ vì $2 \times \frac{1}{2} = 1$.

Scalar product (dot product)

Tích vô hướng của 2 vector $A = (A_x, A_y, A_z)$ và $B = (B_x, B_y, B_z)$ được tính như sau:

$$\begin{bmatrix} A_x & A_y & A_z \end{bmatrix} \begin{bmatrix} B_x \\ B_y \\ B_z \end{bmatrix} = A_x B_x + A_y B_y + A_z B_z = \vec{A} \cdot \vec{B}$$

Chú ý: tích vô hướng của 2 vector A và B có thể viết như sau

$$A \cdot B = A^T B$$

trong đó vế bên phải là phép nhân ma trận hàng A^T với ma trận cột B.

Norm of a vector

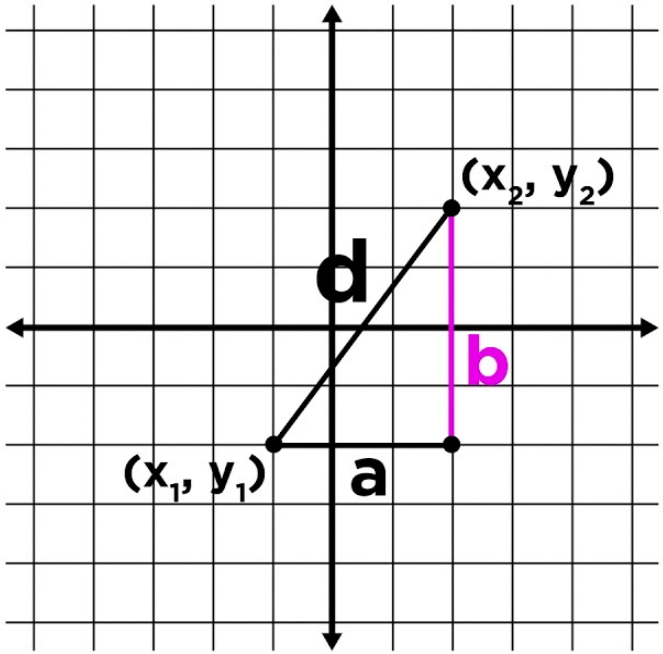
Xét vector $x = (x_1, x_2, \ldots, x_n)$. Khi đó chuẩn hoặc độ dài (norm-length) của vector x được tính như sau:

$$\|x\| = (x^T x)^{1/2} = \sqrt{x_1^2 + x_2^2 + \ldots + x_n^2}$$

Chú ý: $x^T x$ là phép nhân ma trận hàng x^T với ma trận cột x. Khái niệm chuẩn của vector có vai trò quan trọng trong việc tính khoảng cách, 1 yếu tố then chốt trong việc xây dựng các model Machine Learning.

Distance between two points

Deriving and Using the Distance Formula



solve for d

$$a = x_2 - x_1$$

$$b = y_2 - y_1$$

$$a^2 + b^2 = d^2$$

Pythagorean Theorem

$$d^2 = (x_2 - x_1)^2 + (y_2 - y_1)^2$$

Distance between two points

Khoảng cách giữa 2 điểm $A(a_1, a_2, \ldots, a_n)$ và $B(b_1, b_2, \ldots, b_n)$ là

$$\sqrt{(a_1 - b_1)^2 + (a_2 - b_2)^2 + \ldots + (a_n - b_n)^2}.$$

Chuẩn của vector $x = (x_1, x_2, \ldots, x_n)$ chính là khoảng cách từ điểm $X(x_1, x_2, \ldots, x_n)$ đến gốc toạ độ O.
Công thức trên thường được gọi là khoảng cách Euclid. Đây là cách tính khoảng cách thường được sử dụng nhất. Ngoài ra, có nhiều cách tính khoảng cách đặc biệt phục vụ cho một số bài toán nhất định.

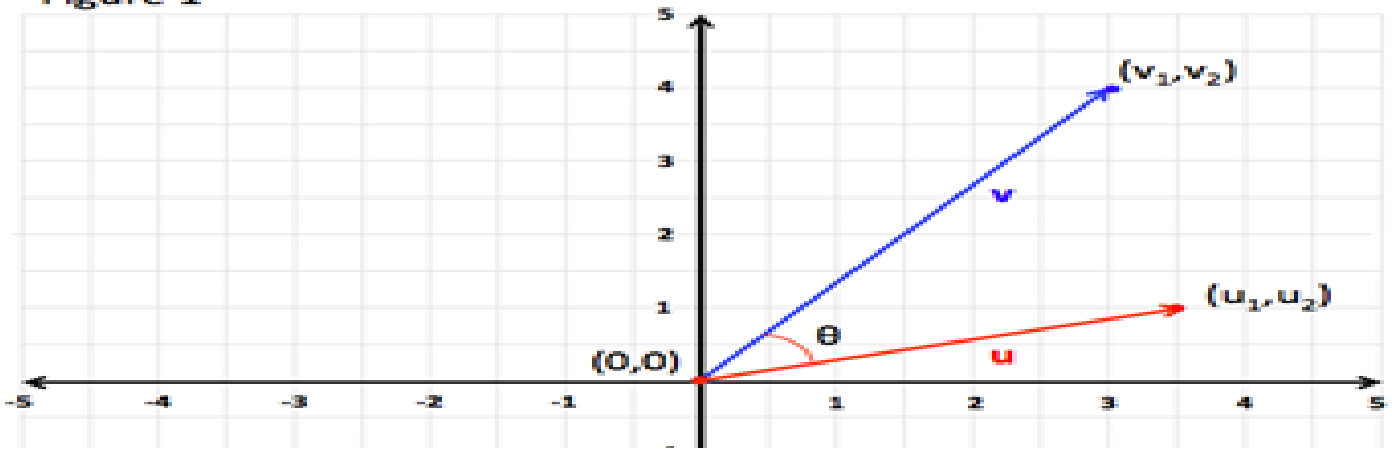
Distance in Machine Learning

Việc tính khoảng cách có vai trò rất quan trọng trong việc đánh giá sai số của model regression và trong bài toán phân cụm trong Machine Learning.

Angle between two vectors

Gọi θ là góc giữa 2 vector $v = (v_1, v_2)$ và $u = (u_1, u_2)$.

Figure 1



Angle between two vectors

Khi đó, giá trị cosine (hàm số lượng giác sin,cos, tan, cotan) của góc θ được tính như sau:

$$cos(\theta) = \frac{u \cdot v}{\|u\| \|v\|}$$

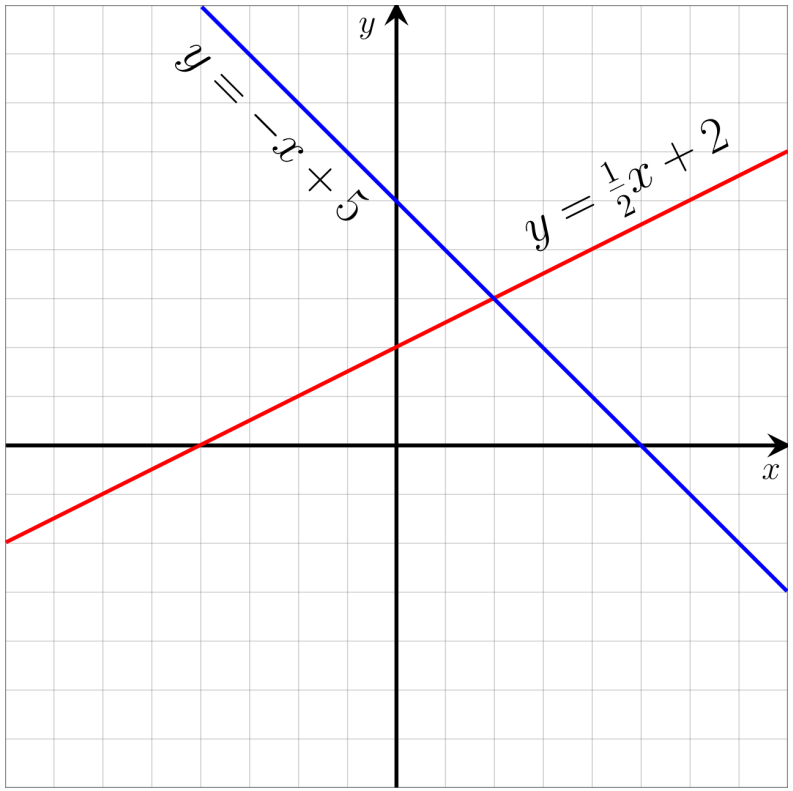
trong đó $u \cdot v = v_1 u_1 + v_2 u_2$ là tích vô hướng của u và v, $\|u\|$ và $\|v\|$ lần lượt là độ dài (norm) của 2 vector u và v.

Cosine similarity

- Giá trị cosine được tính ở slide trước cũng được sử dụng để đo mức độ "giống nhau" giữa 2 vector. Giá trị này càng lớn thì 2 vector càng "giống nhau".
- Kĩ thuật này được sử dụng nhiều trong xử lí ngôn ngữ tự nhiên Natural Language Processing.

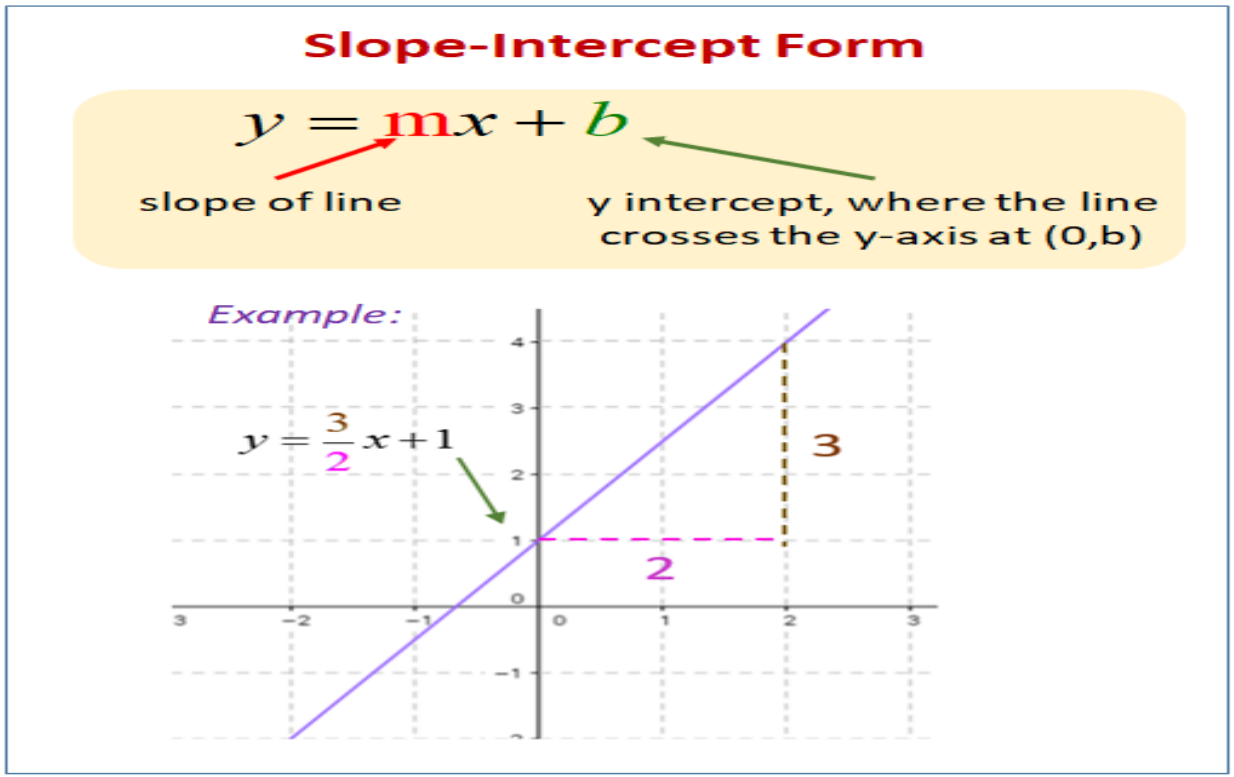
Equation of a hyperplane

Phương trình của 1 đường thẳng thường được viết dưới 2 dạng: $y = ax + b$ hoặc $ax + by + c = 0$ với a, b và c là các số thực. Đây là cơ sở cho model Linear Regression trong Machine Learning.



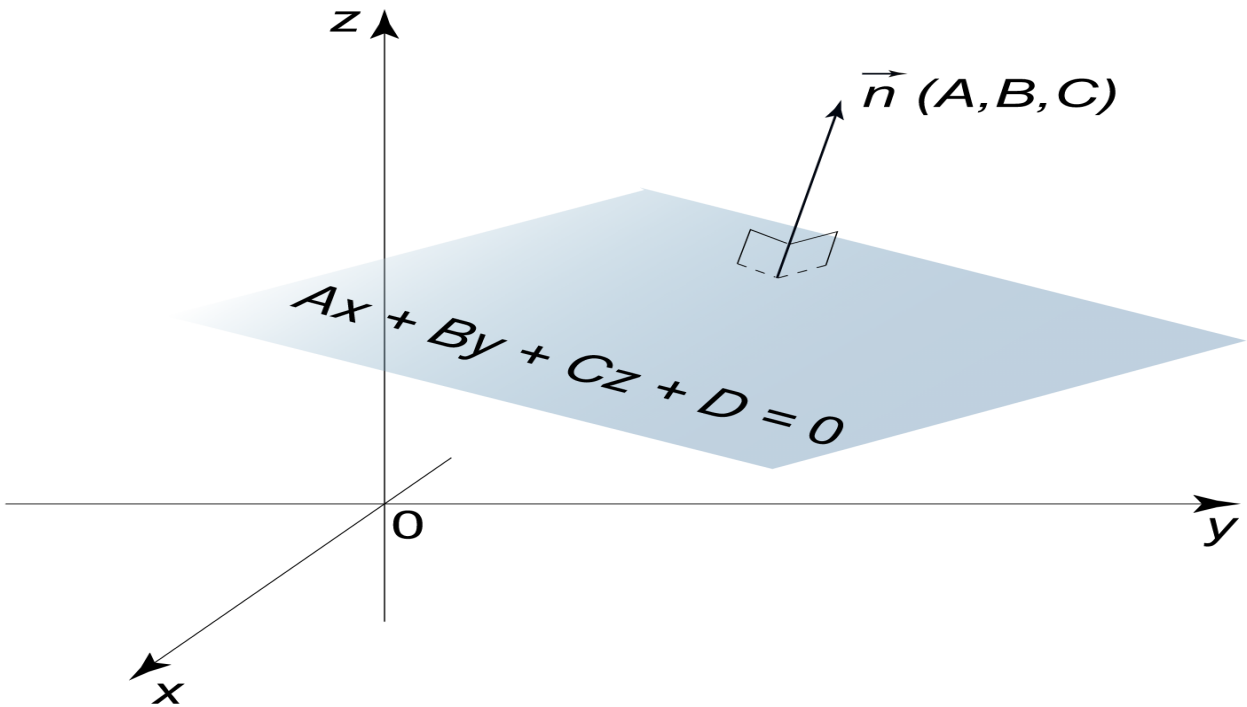
Slope

Nếu slope (độ dốc) lớn hơn 0, đường thẳng đi lên. Nếu slope bé hơn 0, đường thẳng đi xuống. Nếu slope =0, đường thẳng đi ngang.



Hyperplane equation

Phương trình mặt phẳng trong không gian 3 chiều có dạng $Ax + By + Cy + D = 0$ với A, B, C và D là các số thực.



Công thức này và dạng tổng quát của nó giúp mô tả model Linear Regression nhiều chiều.

Hyperplane equation by matrix notation

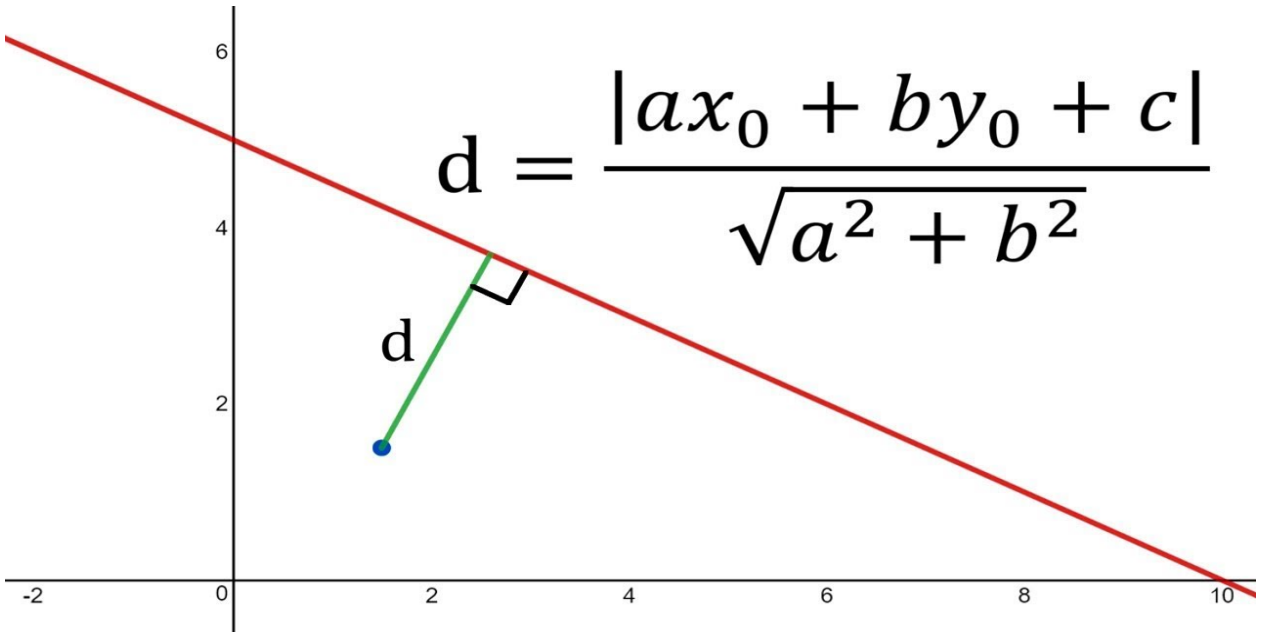
Trong Machine Learning, biểu thức dạng $w_1x_1 + w_2x_2 + w_3x_3$ thường được viết bằng ngôn ngữ ma trận như sau.

$$w_1x_1 + w_2x_2 + w_3x_3 = w^Tx$$

trong đó $w^T = (w_1, w_2, w_3)$ là 1 ma trận hàng và $x = (x_1, x_2, x_3)$ luôn được hiểu là ma trận cột (viết theo chiều dọc).

Distance from a point to a hyperplane

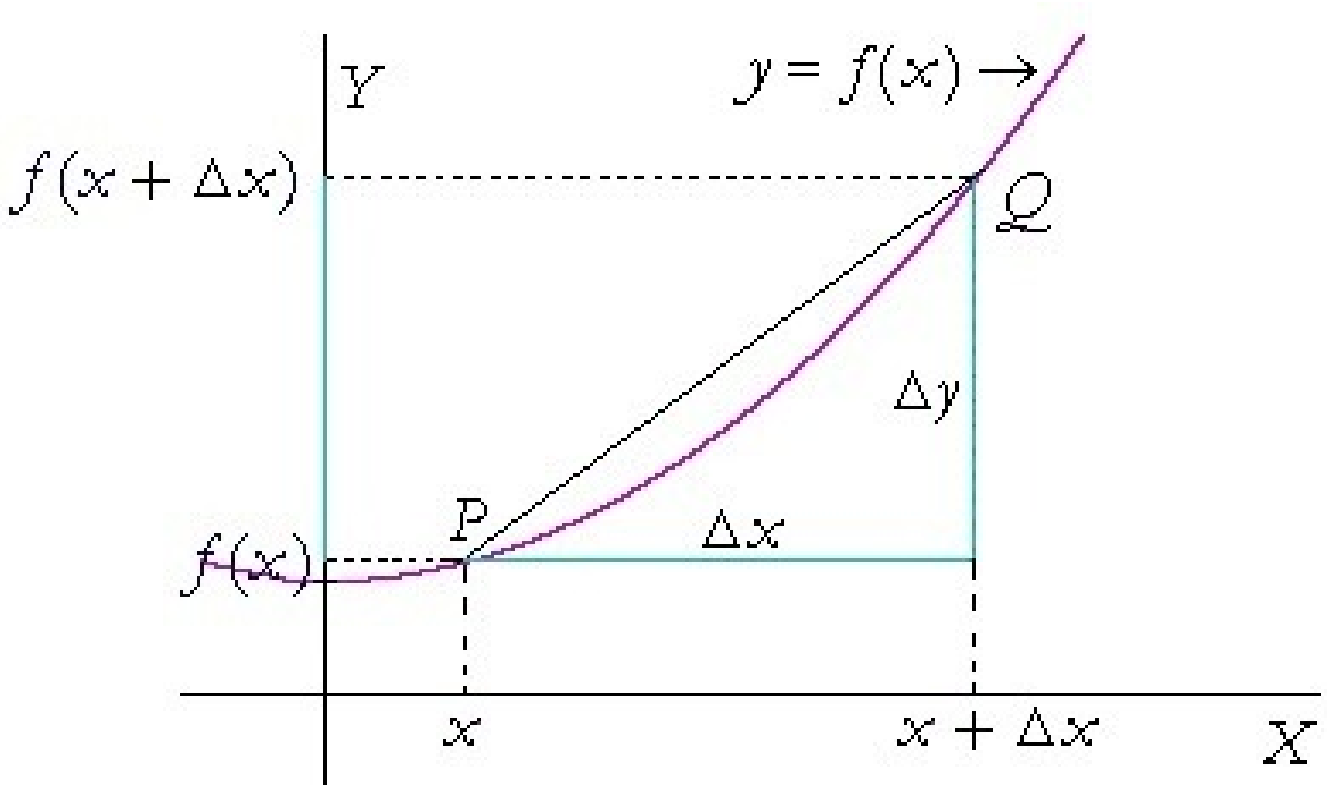
The distance from the point $A(x_0, y_0)$ to the line $ax + by + c = 0$ is



Việc tính khoảng cách này đóng vai trò rất quan trọng trong Machine Learning. Nó trực tiếp giúp xây dựng các model Linear Regression, SVM, ...

Derivative (gradient, rate of change)

Rate of change (tốc độ thay đổi) R của 1 hàm số $f(x)$ được tính bằng $R = \frac{f(x+\Delta x)-f(x)}{\Delta x}$.



Derivative

Đạo hàm (derivative) của hàm số f tại điểm x được tính như sau:

$$\lim_{h \to 0} \frac{f(x+h) - f(x)}{h}.$$

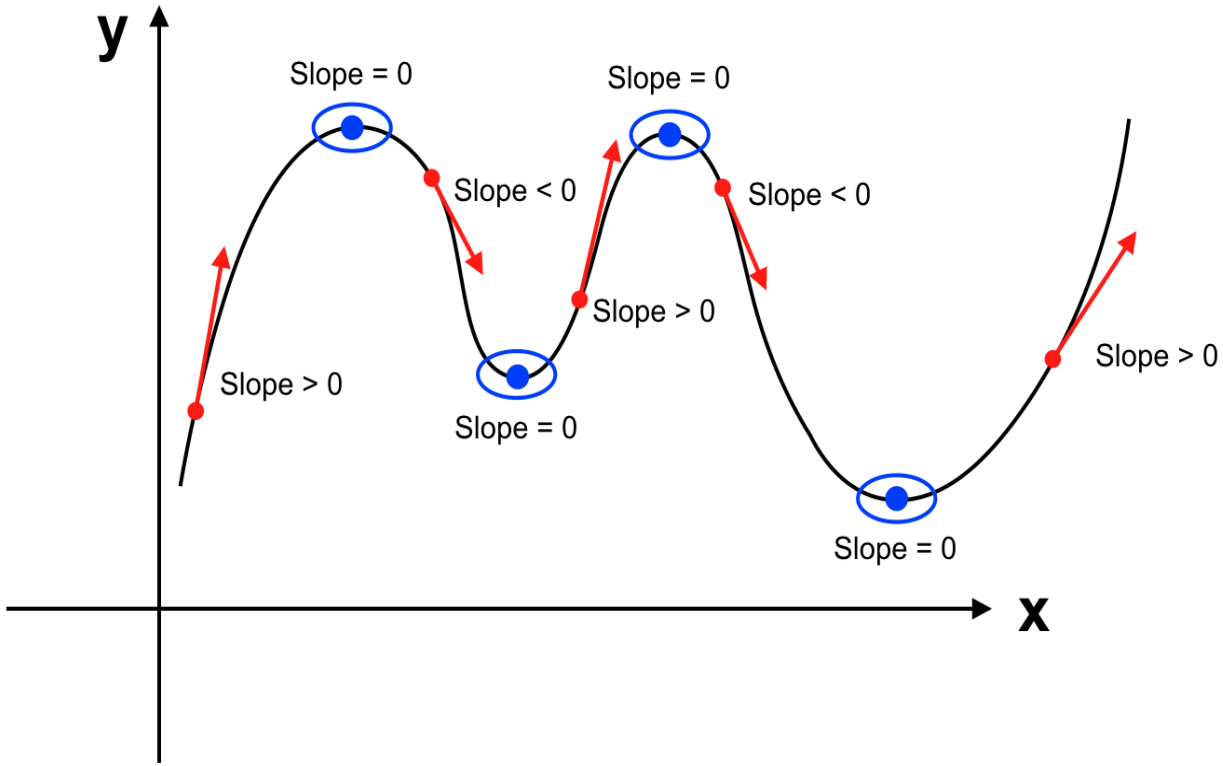
Nói chung, ta cứ hiểu công thức trên một cách đơn giản là tỉ lệ

$$\frac{f(x+h) - f(x)}{h}$$

với h rất nhỏ (0,001 chẳng hạn). Đạo hàm của $f(x)$ thường được kí hiệu là $f'(x)$ hoặc $\frac{df}{dx}$.

Derivative = Slope

Đạo hàm (hàm chỉ đường) của hàm số $f(x)$ tại điểm x_0 là hệ số góc (slope) của tiếp tuyến tại điểm $(x_0, f(x_0))$.

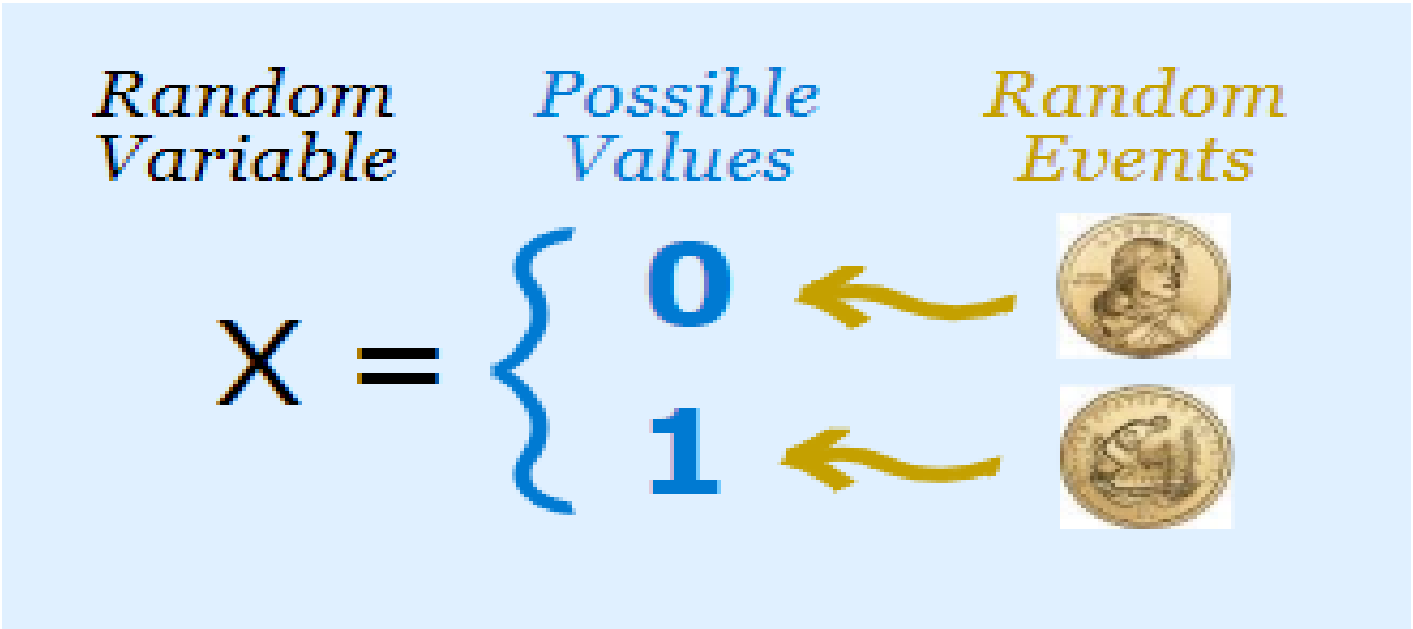


Application of Derivative in ML

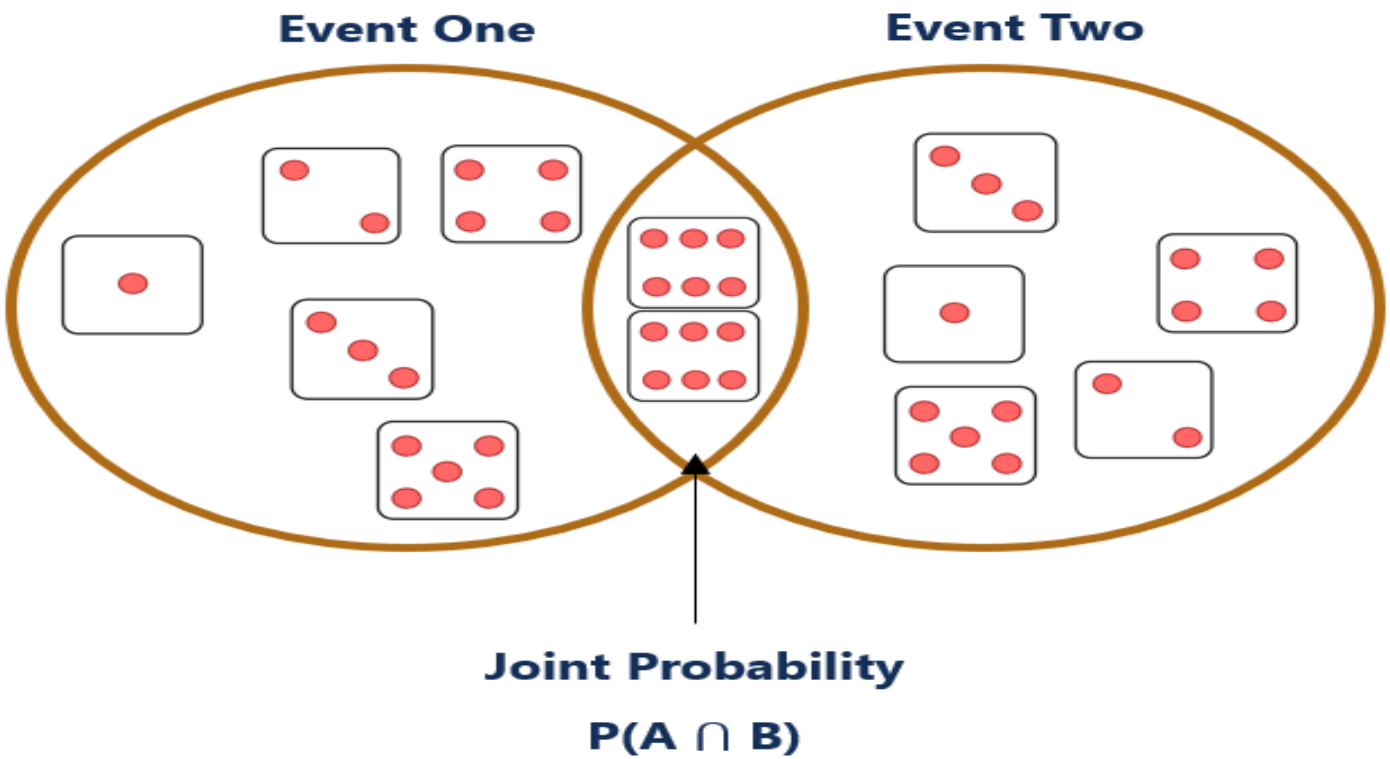
Đạo hàm là khái niệm toán học nền tảng của thuật toán gradient descend, là thuật toán tối ưu quan trọng nhất trong Machine Learning.

Random variables

Biến ngẫu nhiên X (random variables) dùng để kí hiệu kết quả của một thí nghiệm. Ví dụ, xét thí nghiệm tung 1 đồng xu. Ta có thể gán $X = 1$ nếu đồng xu lên mặt ngửa và $X = 0$ nếu đồng xu lên mặt xấp.



Joint probability



Conditional probability

Conditional Probability Formula

$$P(A \mid B) = \frac{P(A \cap B)}{P(B)}$$

$P(A \mid B)$: Probability of A given B

$P(A \cap B)$: Probability of A and B

$P(B)$: Probability of B

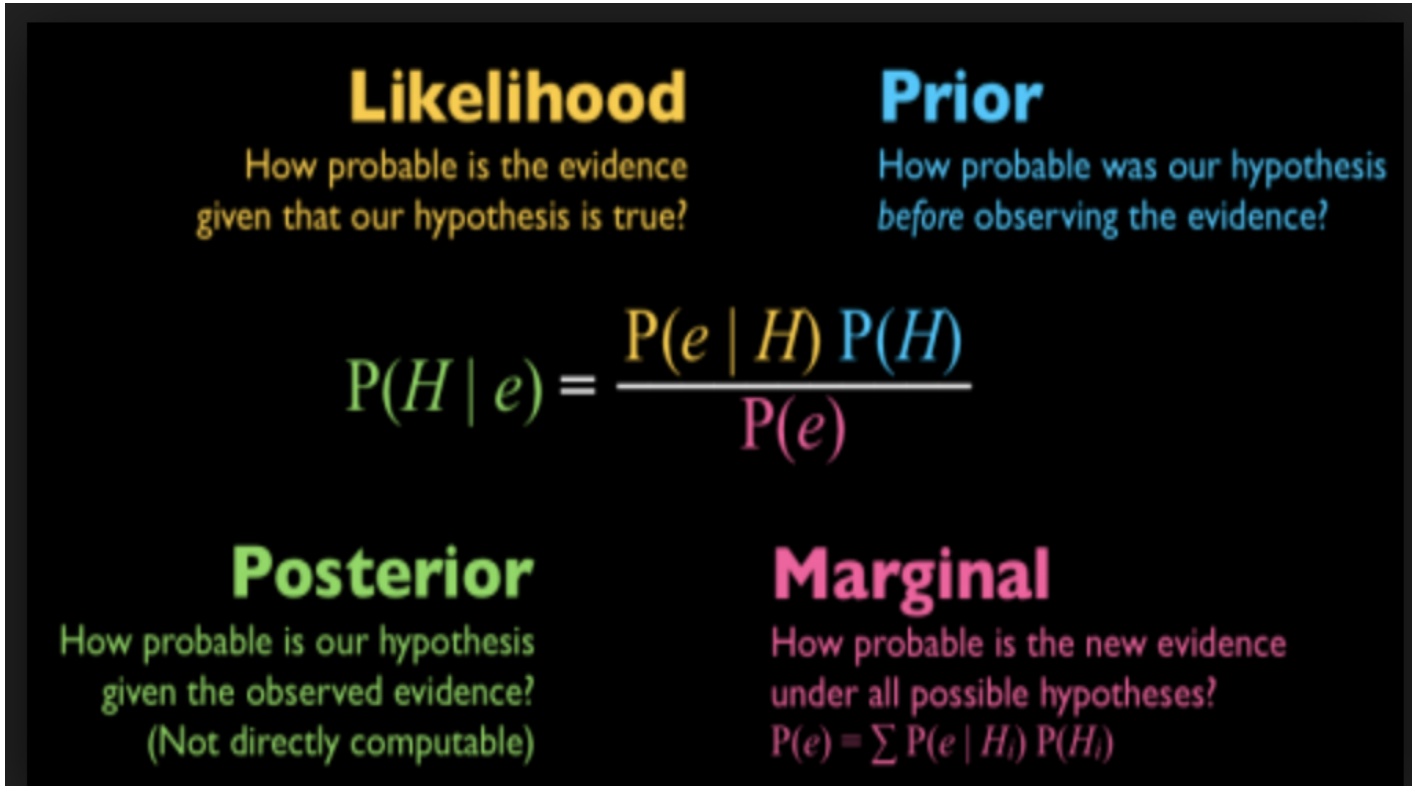
Bayesian formula

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

trong đó

- A và B là 2 sự kiện.
- $P(A|B)$ là xác suất xảy ra A biết rằng B đã xảy ra.
- $P(B|A)$ là xác suất xảy ra B biết rằng A đã xảy ra.
- $P(A)$ là xác suất xảy ra A.
- $P(B)$ là xác suất xảy ra B.

Bayesian formula



Probability density distribution (pdf)

Hàm mật độ xác suất mô tả "độ dày đặc" của dữ liệu.

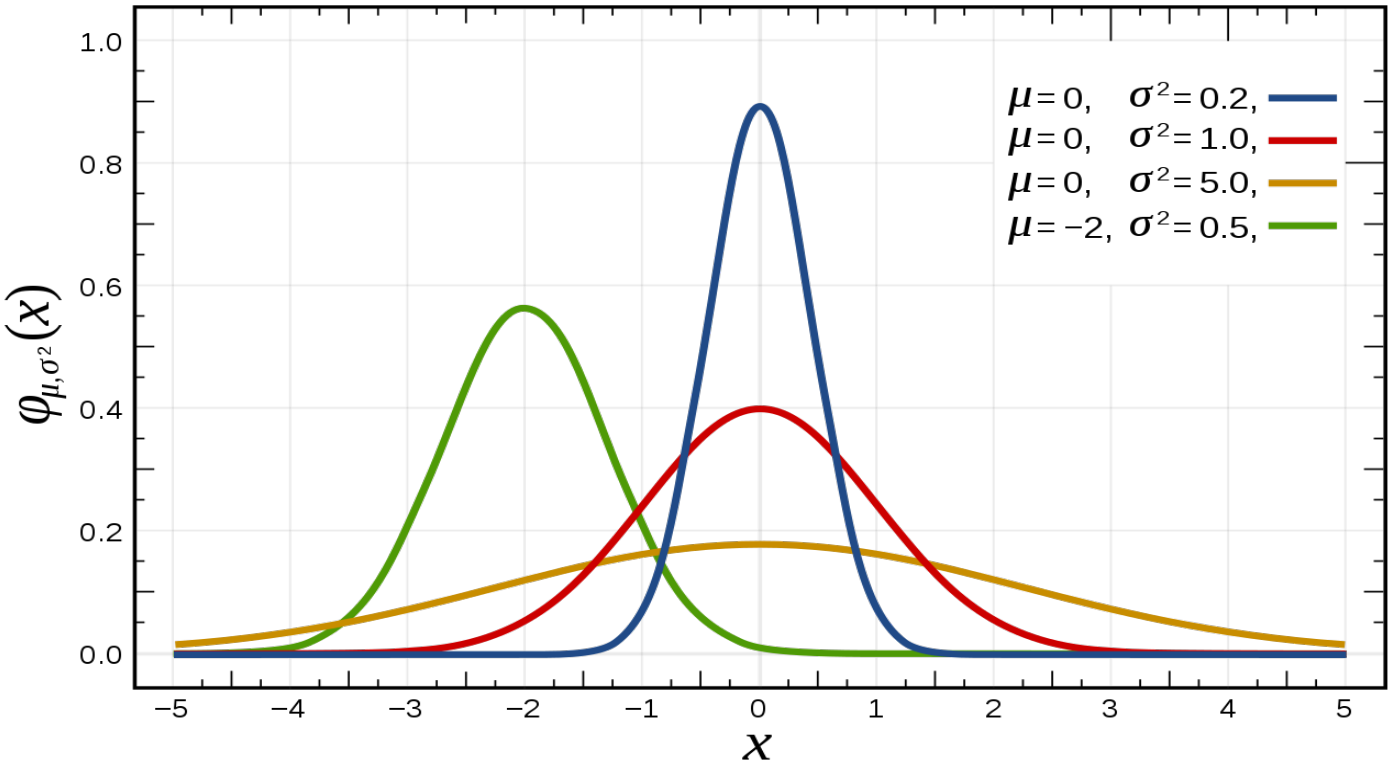
Gauss distribution - Normal distribution

Phân phối Gauss hay phân phối chuẩn là phân phối có hàm mật độ xác suất như sau:

$$f(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{1}{2}(\frac{x-\mu}{\sigma})^2}$$

trong đó μ là giá trị trung bình của phân phối, σ là độ lệch chuẩn.

Gauss distribution - Normal distribution



Maximum likelihood estimation

Giả sử có các điểm dữ liệu $x_1, x_2, \ldots, x_N$. Giả sử ta biết rằng các điểm dữ liệu này tuân theo một phân phối nào đó được mô tả bởi bộ tham số θ.
Maximum likelihood estimation là **tìm bộ tham số** θ sao cho xác suất sau đây đạt giá trị lớn nhất:

$$p(x_1, \ldots, x_N|\theta).$$

Trong tối ưu, ta thường diễn đạt việc này bằng kí hiệu sau

$$\theta = \arg\max_k p(x_1, \ldots, x_N|\theta).$$

Maximum likelihood estimation

Chú ý: Xác suất có điều kiện $P(A|B)$ là xác suất xảy ra sự kiện A biết rằng B đã xả ra.
Giá trị **xác suất có điều kiện** $p(x_1|\theta)$ chính là xác suất xảy ra sự kiện x_1 biết rằng phân phối được mô tả bởi bộ tham số θ. Tương tự,

$$p(x_1, \ldots, x_N|\theta)$$

là xác suất để toàn bộ sự kiện $x_1, \ldots, x_N$ đồng thời xảy ra biết rằng phân phối được mô tả bởi bộ tham số θ.

Maximum likelihood estimation

- Tóm lại, cho trước dữ liệu (biết trước kết qủa), ta đi tìm nguyên nhân (mô hình xác suất mô tả dữ liệu) sao cho xác suất xảy ra kết quả đã biết cao nhất có thể.
- Cách tiếp cận này được sử dụng rất nhiều trong các bài toán Machine Learning mà data đã được dán nhãn (biết trước kết quả). Bài toán kiểu này thường được gọi là Supervised Learning (việc học được hướng dẫn bởi label biết trước).

Xin cảm ơn sự chú ý theo dõi!