

Vector và ma trận

☑ Trắc nghiệm 6 ⬇ PDF bài

Mỗi học sinh trong bảng điểm được mô tả bằng ba con số: giờ tự học mỗi tuần, điểm học kỳ 1 và số buổi vắng. Bốn trăm học sinh là một bảng 400 dòng, 3 cột. Một model tuyến tính dự đoán điểm cuối kỳ cho cả 400 học sinh bằng một phép nhân, viết trong NumPy là $X @ w$; một lớp của mạng neural cũng là một phép nhân như thế. Bài 0.2 đã dùng $@$; bài này giải thích nó tính gì: vector là gì, tích vô hướng đo gì, và vì sao số cột của ma trận bên trái phải bằng số hàng của ma trận bên phải.

MỤC TIÊU

- Cộng vector, nhân vector với số, tính độ dài và khoảng cách.
- Tính tích vô hướng và cosine similarity.
- Viết dự đoán của model tuyến tính bằng $X @ w + b$.
- Đoán shape của phép nhân ma trận và phép chuyển vị.

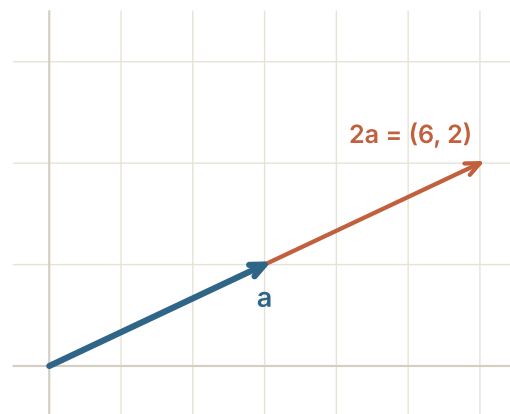
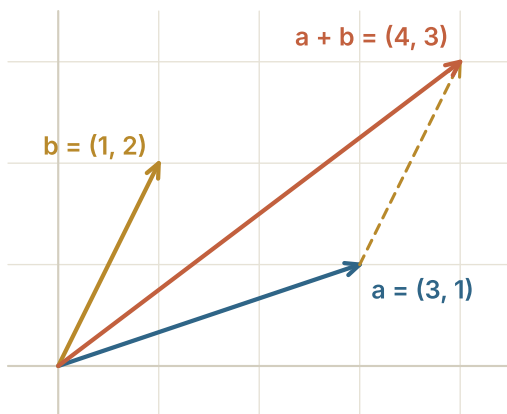
Vector: một dãy số có thứ tự

Một **vector** là một dãy số có thứ tự. Học sinh A tự học 6 giờ, điểm học kỳ 1 là 7.5, vắng 1 buổi, nên được mô tả bằng vector $\mathbf{a} = (6, 7.5, 1)$. Thứ tự quan trọng: $(7.5, 6, 1)$ là một học sinh khác. Số phần tử của vector gọi là **số chiều**. Mỗi con số mô tả một mẫu dữ liệu, như giờ tự học hay số buổi vắng, gọi là một **đặc trưng** (feature); mỗi đặc trưng là một chiều.

Vector hai chiều vẽ được thành một mũi tên từ gốc tọa độ tới điểm (x, y) . Hai phép tính cơ bản đều làm **từng phần tử**:

$$\mathbf{a} + \mathbf{b} = (a_1 + b_1, a_2 + b_2), \quad c \mathbf{a} = (ca_1, ca_2).$$

Cộng hai vector là nối mũi tên này vào đầu mũi tên kia. Nhân với số c là kéo dài mũi tên c lần, hoặc đảo chiều khi $c < 0$.



Cộng hai vector là nối đuôi chúng; nhân với 2 kéo dài mũi tên gấp đôi.

Python

```
import numpy as np

a = np.array([3.0, 1.0])
b = np.array([1.0, 2.0])
print(a + b, 2 * a, a - b)
```

KẾT QUẢ

```
[4. 3.] [6. 2.] [ 2. -1.]
```

Phép trừ $\mathbf{a} - \mathbf{b}$ cho mũi tên đi từ đầu $\mathbf{b}$ tới đầu $\mathbf{a}$. Mục sau dùng nó để đo khoảng cách.

HỎI NHANH

$\mathbf{a} = (2, -1)$ và $\mathbf{b} = (0, 3)$. $2\mathbf{a} + \mathbf{b}$ bằng bao nhiêu?

A $(2, 2)$

B $(4, 1)$

C $(4, 5)$

D $(4, -3)$

Độ dài và khoảng cách

Độ dài của vector (3, 4) là cạnh huyền của tam giác vuông có hai cạnh 3 và 4, bằng 5 theo định lý Pythagoras. Tổng quát cho d chiều, độ dài, hay **chuẩn L2** (L2 norm), là

$$\|\mathbf{a}\| = \sqrt{a_1^2 + a_2^2 + \dots + a_d^2}.$$

Khoảng cách giữa hai điểm là độ dài của hiệu: $\|\mathbf{a} - \mathbf{b}\|$. Đây là khoảng cách Euclid, cách đo mà kNN ở bài 2.4 dùng để tìm hàng xóm gần nhất.

Còn một cách đo khác: cộng trị tuyệt đối từng thành phần, $|a_1| + |a_2| + \dots$, gọi là **chuẩn L1**. Trên bàn cờ, xe đi theo hàng và cột, nên quãng đường của nó là khoảng cách L1, còn quãng đường chim bay là khoảng cách L2.

Python

```
p = np.array([1.0, 2.0])
q = np.array([4.0, 6.0])
d = q - p
print(d, np.linalg.norm(d), np.sqrt((d ** 2).sum()), np.abs(d).sum())
```

KẾT QUẢ

```
[3.  4.]  5.0  5.0  7.0
```

Từ p tới q phải đi 3 theo trục ngang và 4 theo trục dọc. Đường thẳng dài 5, đường đi theo ô dài 7. `np.linalg.norm` mặc định tính chuẩn L2, đúng bằng công thức viết tay.

HỎI NHANH

Vector (1, 2, 2) có độ dài L2 bằng bao nhiêu?

A 5

B 3

C 9

D $\sqrt{5}$

Tích vô hướng

Tích vô hướng (dot product) của hai vector cùng số chiều là tổng các tích từng cặp:

$$\mathbf{a} \cdot \mathbf{b} = a_1b_1 + a_2b_2 + \dots + a_db_d.$$

Kết quả là **một số**, không phải vector. Ý nghĩa đầu tiên là **tổng có trọng số**. Ở bài 0.4, hàm bậc nhất $y = wx + b$ chỉ có một đầu vào. Với ba đặc trưng, model tuyến tính tính

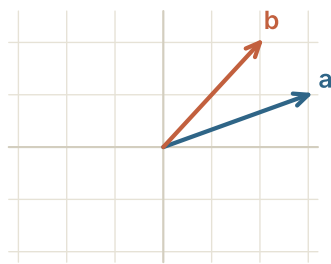
$$\hat{y} = w_1x_1 + w_2x_2 + w_3x_3 + b = \mathbf{w} \cdot \mathbf{x} + b.$$

Mỗi trọng số w_j nói đặc trưng thứ j đóng góp bao nhiêu. Trọng số âm cho số buổi vắng nghĩa là vắng càng nhiều thì điểm dự đoán càng thấp.

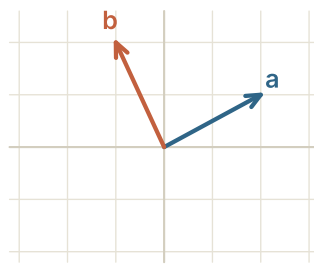
Ý nghĩa thứ hai là hình học. Gọi θ là góc giữa hai mũi tên, khi đó

$$\mathbf{a} \cdot \mathbf{b} = \|\mathbf{a}\| \|\mathbf{b}\| \cos \theta.$$

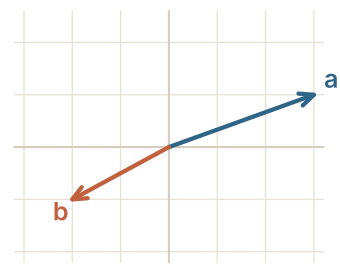
Không cần nhớ lượng giác để đọc công thức này. $\cos \theta$ bằng 1 khi hai mũi tên cùng hướng, bằng 0 khi chúng vuông góc, bằng -1 khi ngược hướng. Vì vậy dấu của tích vô hướng cho biết hai vector “cùng phía” hay “ngược phía”.



cùng hướng
 $a \cdot b = 8$



vuông góc
 $a \cdot b = 0$



ngược hướng
 $a \cdot b = -7$

$a \cdot b$ dương khi cùng hướng, bằng 0 khi vuông góc, âm khi ngược hướng.

Chia tích vô hướng cho hai độ dài, ta còn lại đúng $\cos \theta$, gọi là **cosine similarity**. Nó chỉ đo hướng, không đo độ dài, nên nằm trong khoảng từ -1 đến 1 . Các chương sau biểu diễn một từ hay một ảnh bằng vector, gọi là embedding, rồi dùng cosine similarity để tìm từ hay ảnh giống nhau.

Python

```
w = np.array([0.1, 0.8, -0.3])      # giờ tự học, điểm HK1, số buổi vắng
x = np.array([6.0, 7.5, 1.0])      # học sinh A
print(w @ x, (w * x).sum(), w @ x + 1.0)

def cosine(u, v):
    return u @ v / (np.linalg.norm(u) * np.linalg.norm(v))

meo = np.array([0.9, 0.8, 0.1])
cho = np.array([0.8, 0.9, 0.2])
oto = np.array([0.1, 0.2, 0.9])
print(round(cosine(meo, cho), 3), round(cosine(meo, oto), 3))
```

KẾT QUẢ

```
6.3 6.3 7.3
0.99 0.303
```

Dòng đầu: tích vô hướng bằng tổng các tích từng cặp, $0.6 + 6.0 - 0.3 = 6.3$; cộng thêm $b = 1$ được điểm dự đoán 7.3. Dòng hai dùng ba vector đồ chơi cho ba từ: “mèo” và “chó” gần cùng hướng nên cosine gần 1, “mèo” và “ô tô” lệch hướng nhiều nên cosine chỉ 0.3.

HỎI NHANH

$\mathbf{a} = (2, 1)$ và $\mathbf{b} = (-1, 2)$. Tích vô hướng $\mathbf{a} \cdot \mathbf{b}$ bằng bao nhiêu, và hai vector nằm thế nào với nhau?

A $(-2, 2)$; cùng hướng

B 4; cùng hướng

C -4 ; ngược hướng

D 0; vuông góc

Ma trận: một bảng số

Xếp vector của bốn học sinh thành bốn hàng, ta được một **ma trận** 4 hàng, 3 cột, viết là ma trận 4×3 hay shape $(4, 3)$. Quy ước của học máy: **mỗi hàng là một mẫu, mỗi cột là một đặc trưng**. Ma trận dữ liệu thường ký hiệu X , shape (n, d) với n mẫu và d đặc trưng.

Nhân ma trận X với vector trọng số $\mathbf{w}$ là làm tích vô hướng của **từng hàng** với $\mathbf{w}$:

$$X\mathbf{w} = \begin{pmatrix} \mathbf{x}_1 \cdot \mathbf{w} \\ \mathbf{x}_2 \cdot \mathbf{w} \\ \vdots \\ \mathbf{x}_n \cdot \mathbf{w} \end{pmatrix}.$$

Một phép nhân cho dự đoán của cả n học sinh. Mỗi hàng có 3 số và $\mathbf{w}$ có 3 số, nên số cột của X phải bằng độ dài của $\mathbf{w}$. Kết quả có n số, một số cho mỗi hàng.

Python

```
X = np.array([[6.0, 7.5, 1.0],
              [2.0, 6.0, 4.0],
              [10.0, 8.5, 0.0],
              [4.0, 5.5, 2.0]])

b = 1.0
print(X.shape, w.shape, (X @ w).shape)
```

```
print(X @ w + b)
print([round(float(row @ w + b), 1) for row in X])
```

KẾT QUẢ

```
(4, 3) (3,) (4,)
[7.3 4.8 8.8 5.2]
[7.3, 4.8, 8.8, 5.2]
```

Dòng cuối tính lại bằng vòng lặp qua từng hàng và ra đúng bốn số. Phép $X @ w$ làm cùng việc ấy, chỉ nhanh hơn nhiều khi bảng có hàng triệu dòng. Số b được cộng vào cả bốn dự đoán nhờ broadcasting của bài 0.2.

HỎI NHANH

Bảng dữ liệu có 500 học sinh, mỗi học sinh 8 đặc trưng. Vector trọng số w phải có shape gì, và $X @ w$ có shape gì?

- A w là (500,), kết quả là (8,)
- B w là (8,), kết quả là (8,)
- C w là (8,), kết quả là (500,)
- D w là (500, 8), kết quả là (500,)

Nhân hai ma trận

Giờ muốn hai dự đoán cho mỗi học sinh, chẳng hạn điểm Toán và điểm Văn. Mỗi môn có một vector trọng số riêng. Xếp hai vector ấy thành **hai cột** của ma trận W shape (3, 2). Khi đó XW có shape (4, 2): ô ở hàng i , cột j là tích vô hướng của **hàng i của X** với **cột j của W** .

Quy tắc shape:

$$(m, k) @ (k, n) \rightarrow (m, n).$$

Hai số k ở giữa phải bằng nhau, vì đó là độ dài của hai vector đem nhân vô hướng. Hai số ở ngoài cho shape kết quả. Một lớp của mạng neural làm đúng phép này: X là (n, d) , W là (d, h) , kết quả (n, h) là h đặc trưng mới cho mỗi mẫu. Mỗi cột của W là một nơ-ron.

X (4, 3)				W (3, 2)			H (4, 2)		
6	7.5	1	@	0.1	-0.2	=	6.3	2.95	$2 \cdot (-0.2) + 6 \cdot 0.5$ $\leftarrow + 4 \cdot 0.4$ $= 4.2$
2	6	4		0.8	0.5		3.8	4.2	
10	8.5	0		-0.3	0.4		7.8	2.25	
4	5.5	2					4.2	2.75	

$(4, 3) @ (3, 2) \rightarrow (4, 2)$: ô tô đậm là tích của hàng ô tô đậm với cột ô tô đậm.

Chuyển vị X^T (trong NumPy là $X.T$) đổi hàng thành cột, nên shape (m, n) thành (n, m) . Thứ tự nhân quan trọng: AB và BA nói chung khác nhau, và nhiều khi một trong hai còn không nhân được vì shape không khớp. **Ma trận đơn vị** I có 1 trên đường chéo, 0 ở mọi ô khác; nhân với I không đổi gì, như nhân một số với 1.

Python

```
W = np.array([[0.1, -0.2],
              [0.8, 0.5],
              [-0.3, 0.4]])

H = X @ W
print(H.shape)
print(np.round(H, 2))
print(X.T.shape, W.T.shape)

try:
    W @ X
except ValueError as e:
    print("W @ X lỗi:", e)

I = np.eye(3)
print(np.allclose(X @ I, X))
```

KẾT QUẢ

```
(4, 2)
[[6.3  2.95]
 [3.8  4.2 ]
 [7.8  2.25]
 [4.2  2.75]]
```

```
(3, 4) (2, 3)
```

```
W @ X lỗi: matmul: Input operand 1 has a mismatch in its core dimension 0, with gufunc signature (n?,k),(k,m?)->(n?,m?) (size 4 is different from 2)
```

```
True
```

Cột đầu của H là $X @ w$ ở mục trước, chưa cộng b , vì cột đầu của W chính là w . $W @ X$ là $(3, 2) @ (4, 3)$: số 2 và số 4 ở giữa không bằng nhau, nên NumPy báo lỗi. Thông báo lỗi dài nhưng đọc được ý chính ở cuối: kích thước 4 khác 2.

HỎI NHANH

Nhân ma trận 3×2 với ma trận 2×4 cho kết quả gì?

A Ma trận 2×2

B Ma trận 3×2

C Ma trận 3×4

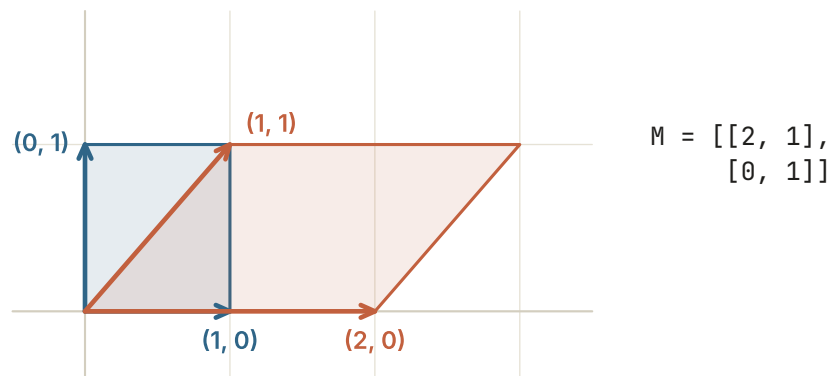
D Không nhân được vì hai ma trận khác shape

ĐỀ THI NOAI 2026 Singapore · Vòng sơ loại · Câu 12

Ma trận như một phép biến đổi

Một ma trận 2×2 nhân với mọi điểm của mặt phẳng là một cách **biến đổi** mặt phẳng.

$\begin{pmatrix} 2 & 0 \\ 0 & 1 \end{pmatrix}$ kéo giãn trục ngang gấp đôi. $\begin{pmatrix} 0 & -1 \\ 1 & 0 \end{pmatrix}$ xoay mọi điểm 90 độ ngược chiều kim đồng hồ. Cột đầu của ma trận là nơi điểm $(1, 0)$ đi tới, cột sau là nơi điểm $(0, 1)$ đi tới.



Cột của M là nơi $(1, 0)$ và $(0, 1)$ đi tới.

Một lớp của mạng neural cũng làm việc này: nó biến đổi dữ liệu sang một hệ tọa độ mới, nơi các lớp dễ tách nhau hơn.

Trong toán, điểm được viết thành cột và nhân bên phải ma trận: $M\mathbf{v}$. Trong code, mỗi điểm là một hàng của bảng X , nên cùng phép biến đổi ấy viết là $X @ M.T$.

Python

```
square = np.array([[0, 0], [1, 0], [1, 1], [0, 1]], dtype=float) # mỗi hàng một đỉnh
M = np.array([[2.0, 1.0],
              [0.0, 1.0]])
print(square @ M.T)
```

KẾT QUẢ

```
[[0. 0.]
 [2. 0.]
 [3. 1.]
 [1. 1.]]
```

Đỉnh $(1, 0)$ đi tới $(2, 0)$, đúng cột đầu của M ; đỉnh $(0, 1)$ đi tới $(1, 1)$, đúng cột sau. Hình vuông thành hình bình hành.

HỎI NHANH

Ma trận $M = \begin{pmatrix} 0 & -1 \\ 1 & 0 \end{pmatrix}$ xoay điểm 90 độ. Điểm $(1, 0)$ đi tới đâu?

A $(0, 1)$

B $(0, -1)$

C $(-1, 0)$

D $(1, 0)$, vì điểm nằm trên trục

Khi đi thi

- Ghi shape cạnh mỗi biến trước khi nhân. Nhân không được thì xem hai số ở giữa.
- @ là nhân ma trận, * là nhân từng phần tử.
- Kiểm tra @ w bằng một hàng: tính $X[0] @ w$ rồi so với phần tử đầu.

Tóm tắt

- Vector là dãy số có thứ tự; cộng và nhân với số làm từng phần tử.
- Độ dài L2 $\sqrt{\sum a_i^2}$, L1 $\sum |a_i|$; khoảng cách là độ dài của hiệu.
- Tích vô hướng $\sum a_i b_i$ là một số: tổng có trọng số, và $\|\mathbf{a}\| \|\mathbf{b}\| \cos \theta$; cosine similarity chỉ đo hướng.
- Ma trận dữ liệu (n, d) : hàng là mẫu, cột là đặc trưng; $X\mathbf{w}$ là tích vô hướng từng hàng với $\mathbf{w}$.
- $(m, k) @ (k, n) \rightarrow (m, n)$; chuyển vị đổi (m, n) thành (n, m) ; $AB \neq BA$ nói chung; nhân với I không đổi gì.
- Ma trận 2×2 biến đổi mặt phẳng; cột của nó là ảnh của hai vector đơn vị.

Kiểm tra cuối bài

Mười câu dưới đây bao quát cả bài. Làm hết rồi xem điểm; câu nào sai thì đọc lại mục tương ứng.

CÂU 1

Vector $\mathbf{v} = (3, 4)$ có độ dài Euclid (chuẩn L2) bằng bao nhiêu?

A 7

B 5

C 25

D 12

ĐỀ THI NOAI 2026 Singapore · Vòng sơ loại · Câu 13

CÂU 2

A và B là hai ma trận 3×3 . Phép nào **không** phải phép toán ma trận thông thường?

A $A + B$

B AB

C $A \div B$

D $A - B$

ĐỀ THI NOAI 2026 Singapore · Vòng sơ loại · Câu 14

CÂU 3

Ma trận đơn vị I có đặc điểm gì?

- A Mọi ô bằng 1
- B Mọi ô bằng 0
- C Đường chéo bằng 0, mọi ô khác bằng 1
- D Đường chéo bằng 1, mọi ô khác bằng 0

PHÒNG THEO NOAI 2026 Singapore · Vòng sơ loại · Câu 15

CÂU 4

Ma trận A có shape 2×3 . A^T có shape gì?

- A 3×2
- B 2×3
- C 3×3
- D 2×2

ĐỀ THI NOAI 2026 Singapore · Vòng sơ loại · Câu 16

CÂU 5

Trong phép nhân ma trận AB , kích thước nào phải bằng nhau?

A Số hàng của A và số hàng của B

B Số hàng của A và số cột của B

C Mọi kích thước phải bằng nhau

D Số cột của A và số hàng của B

ĐỀ THI NOAI 2026 Singapore · Vòng sơ loại · Câu 20

CÂU 6

Ma trận nào dưới đây đối xứng, tức là bằng chuyển vị của chính nó?

A $\begin{pmatrix} 1 & 2 \\ 3 & 4 \end{pmatrix}$

B $\begin{pmatrix} 1 & 0 \\ 1 & 0 \end{pmatrix}$

C $\begin{pmatrix} 1 & 2 \\ 2 & 3 \end{pmatrix}$

D $\begin{pmatrix} 0 & 1 \\ 0 & 0 \end{pmatrix}$

ĐỀ THI NOAI 2026 Singapore · Vòng sơ loại · Câu 18

CÂU 7

Một lớp của mạng neural nhận batch X shape $(32, 784)$ và cho ra $(32, 128)$. Ma trận trọng số W trong $X @ W$ có shape gì?

A $(32, 128)$

B $(128, 784)$

C $(784, 128)$

D $(32, 784)$

CÂU 8

Hai embedding $\mathbf{u}$ và $\mathbf{v}$ có cosine similarity bằng -0.9 . Nhân đôi $\mathbf{u}$ thì cosine similarity bằng bao nhiêu?

A Vẫn -0.9

B -1.8

C -0.45

D 0.9

CÂU 9

Bảng có hai cột: tuổi (từ 10 đến 18) và thu nhập gia đình (từ 5 triệu đến 80 triệu đồng mỗi tháng). Dùng khoảng cách Euclid trên dữ liệu thô để tìm học sinh giống nhau thì chuyện gì xảy ra?

- A Hai cột đóng góp ngang nhau vào khoảng cách
- B Cột thu nhập gần như quyết định cả khoảng cách
- C Cột tuổi quyết định vì nó ít biến động hơn
- D Không tính được vì hai cột khác đơn vị đo

CÂU 10

A shape (5, 3) và B shape (5, 3). Lệnh nào chạy được?

- A A @ B
- B A @ B.T.T
- C B @ A
- D A.T @ B

Đọc thêm

- 3Blue1Brown — [Essence of linear algebra](#), tập 1–4 và 9. Vector, tổ hợp tuyến tính, ma trận như phép biến đổi, tích vô hướng, bằng hình động.
- Khan Academy — [Vectors and spaces](#). Bài giảng và bài tập ngắn về vector, độ dài, tích vô hướng.
- Deisenroth, Faisal, Ong — [Mathematics for Machine Learning](#), chương 2 và 3. Sách miễn phí; đọc phần ma trận, chuẩn và tích vô hướng khi muốn chặt chẽ hơn.
- NumPy — [Linear algebra \(numpy.linalg\)](#). Danh sách hàm norm, matmul, eye và các phép khác dùng trong bài.