

produit de matrices

En NumPy, il existe deux types principaux de produits :

1. Produit élément par élément (Hadamard)

➤ Opérateur :

➤ Fonction équivalente : `np.multiply(A, B)`

Chaque élément de `A` est multiplié avec l'élément correspondant de `B`.

✓ Exemple :

```
import numpy as np

A = np.array([[1, 2],
              [3, 4]])

B = np.array([[5, 6],
              [7, 8]])

# Produit Hadamard (élément par élément)
C = A * B
print(C)
```

 Résultat :

```
[[ 5 12]
 [21 32]]
```

Ici :

- $C_{11}=1\times 5$
- $C_{12}=2\times 6$

- $C_{21}=3\times 7$
- $C_{22}=4\times 8$

2. Produit matriciel traditionnel

➤ Opérateur :  (depuis Python 3.5)

➤ Fonction équivalente : `np.dot(A, B)` ou `np.matmul(A, B)`

C'est le **produit linéaire** entre matrices (somme des produits ligne \times colonne).

✓ Exemple :

```
D = A @ B
print(D)
```

 Résultat :

```
[[19 22]
 [43 50]]
```

Calculs :

- $D_{11}=1\times 5+2\times 7=19$
- $D_{12}=1\times 6+2\times 8=22$
- $D_{21}=3\times 5+4\times 7=43$
- $D_{22}=3\times 6+4\times 8=50$

Résumé visuel

Type de produit	Symbole / fonction	Type d'opération	Exemple résultat
Hadamard (élément par élément)	<code>A * B</code> ou <code>np.multiply(A, B)</code>	Multiplication élément par élément	<code>[[5, 12], [21, 32]]</code>

Type de produit	Symbole / fonction	Type d'opération	Exemple résultat
Matriciel (linéaire)	<code>A @ B</code> , <code>np.dot(A,B)</code>	Somme ligne × colonne	<code>[[19, 22], [43, 50]]</code>