



# Scalaires, Vecteurs et Matrices

**MATLAB**



**MATrix LABoratory**

**1- Opérations arithmétiques**

**2- Manipulation des vecteurs et matrices**

**3- Manipulations des nombres complexes**

**4- Sauvegarde de données**

- Respect des priorités usuelles des opérations  $\{+, -, *, /\}$  (associativité, commutativité, ...)

➡ Existence d'une division à gauche :  $A \setminus B$ .  
B est divisé par A.

- Elévation à une puissance : symbole  $\wedge$

Exemple :

```
» 3^4
Ans =
      81
» 9^(12)
Ans =
2.8243e+11
```

## Saisie d'un vecteur

**Vecteur  
ligne**

```
» x = [1 2 3]  
x =  
    1 2 3
```

**Vecteur  
colonne**

```
» y = [4; 5; 6;]  
y =  
     4  
     5  
     6
```

## Construction d'un vecteur à partir d'un autre

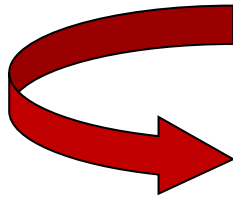
```
» w = [x 4 5]  
w =  
    1 2 3 4 5
```

```
» z = [3; y; 7;]  
z =  
     3  
     4  
     5  
     6  
     7
```

## Construction d'un vecteur

➡ On connaît la valeur initiale, la valeur finale et le pas

$$\mathbf{X} = [ \mathbf{V}_{\text{initiale}} : \mathbf{Pas} : \mathbf{V}_{\text{finale}} ]$$



```
»x = [0:2:8]
```

```
x =
```

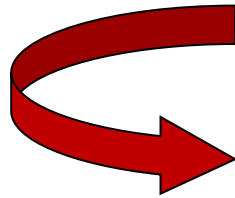
```
0 2 4 6 8
```

- On peut omettre le pas. Il est égal à 1 par défaut.
- On peut omettre les crochets.

## Construction d'un vecteur

➡ On connaît la valeur initiale, la valeur finale et le nombre de points

**X = linspace( V<sub>initiale</sub>, V<sub>finale</sub>, nbpoints)**



```
»x = linspace (0,8,5)
```

```
x =
```

```
0 2 4 6 8
```

- On peut omettre le nombre de points. Il est égal à 100 par défaut.
- **X = logspace( debut, fin, nbpoints) → Valeurs**  
entre  $10^{\text{debut}}$  et  $10^{\text{fin}}$

## Taille d'un vecteur

- `size(x)` donne les dimensions de la variable x.
- `length(x)` donne la longueur de la variable x.

```
»x = [0:2:8];
»size(x)
Ans =
    1 5
»length(x)
Ans =
    5
```

## Transposé d'un vecteur

```
»x = [0:2:4];
»y = x'
Ans =
    0
    2
    4
```

## Récupération d'une composante

Soit le vecteur  $x = [8 \ 0 \ 5]$

➡  $X(1) = 8$

➡  $X([1 \ 3]) = 8 \ 5$

**Les indices d'un tableau  
commencent à 1**

## Opérations sur les vecteurs

➔ Avec un scalaire : addition, soustraction, multiplication, division.

➔ L'addition et la soustraction de vecteurs de mêmes dimensions se font élément par élément.

```
»x = [1 2 3];  
»y = [0 2 2];
```



```
»x + y
```

```
Ans =
```

```
1 4 5
```



➔ Le produit d'un vecteur colonne de taille  $n$  par un vecteur ligne de taille  $m$  donne une matrice  $(n,m)$ .

```
»x = [1 2 3];  
»y = [0 2 2];
```



```
» x'*y
```

```
ans =
```

```
0    2    2
```

```
0    4    4
```

```
0    6    6
```

➔ En précédant **d'un point** les opérateurs, les calculs se font élément par élément

```
»x = [1 2 3];  
»y = [0 2 2];
```



```
» x.*y
```

```
ans =
```

```
0    4    6
```

## Quelques commandes particulières

`sum(x)` : Somme des composantes du vecteur

`prod(x)` : Produit des composantes du vecteur

`sqrt(x)` : Racine carrée des composantes du vecteur

`mean(x)` : Moyenne des composantes du vecteur

`log(x)` : Logarithme népérien des composantes du vecteur

`sin(x)` : Sinus des composantes du vecteur

`exp(x)` : Exponentielle des composantes du vecteur

## Saisie d'une matrice

```
»w = [0 1 2; 3 4 5]
```

```
w =
```

```
0 1 2
```

```
3 4 5
```

ou

```
»w = [0 1 2
```

```
3 4 5]
```

```
w =
```

```
0 1 2
```

```
3 4 5
```

## Taille d'une matrice

```
»size(w)
```

```
Ans =
```

```
2 3
```

On peut aussi écrire :

**$[m,n] = \text{size}(w)$**

m : nombre de ligne

n : nombre de colonne

## Extraction d'un élément de matrice

»w = [0 1 2; 3 4 5]

w =

0 1 2

3 4 5



Élément de la 2<sup>ème</sup> ligne et  
de la 1<sup>ère</sup> colonne

$$W(2,1) = 3$$

## Extraction d'une partie de matrice (colonne ou ligne)

Première ligne :

$$L1 = W(1,:) \quad \swarrow$$

Seconde colonne :

$$C2 = W(:,2) \quad \nwarrow$$

Indique que l'on  
prend toutes les  
colonnes

## Opérations arithmétiques

➡ L'addition et la soustraction de matrices de mêmes dimensions se font élément par élément.

➡ La multiplication

$$A * B = \begin{vmatrix} a_1 & a_2 \\ a_3 & a_4 \end{vmatrix} * \begin{vmatrix} b_1 & b_2 \\ b_3 & b_4 \end{vmatrix} = \begin{vmatrix} a_1 b_1 + a_2 b_3 & a_1 b_2 + a_2 b_4 \\ a_3 b_1 + a_4 b_3 & a_3 b_2 + a_4 b_4 \end{vmatrix}$$

$$A . * B = \begin{vmatrix} a_1 b_1 & a_2 b_2 \\ a_3 b_3 & a_4 b_4 \end{vmatrix}$$

➡ La division

$A/B$  correspond au calcul  $A * B^{-1}$

## Passage d'une matrice à un vecteur

»W = [0 1 2; 3 4 5]

W =

0 1 2

3 4 5



Vecteur colonne

$$v = W(:,);$$

## Matrices particulières



Matrice identité

$$M = \text{eye}(3);$$

»M =

1	0	0
0	1	0
0	0	1



Matrice nulle

$$M = \text{zeros}(2,3);$$

»M =

0	0	0
0	0	0

## Matrices particulières



Matrice unité

$M = \text{ones}(2,3) ;$

»M =  
1 1 1  
1 1 1



Matrice aléatoire

$M = \text{rand}(2,2) ;$

»M =  
0.45 0.82  
0.01 0.44

## Commandes particulières

$\text{sum}(M)$  ou  $\text{sum}(M,1)$  : Somme des éléments des colonnes de la matrice

$\text{sum}(M,2)$  : Somme des éléments des lignes de la matrice

Idem pour les fonctions  $\text{prod}(M)$  et  $\text{mean}(M)$ , produit et moyenne.

## Quelques commandes particulières

$\text{sqrt}(M)$  : racine carrée des éléments de la matrice

$\text{det}(M)$  : déterminant de la matrice

$\text{inv}(M)$  : inverse de la matrice



➔ L'imaginaire pur est noté **i** ou **j**.

## Ecriture d'un nombre complexe

$$\gg z = 4 + 3j$$

$z =$

$$4.0000 + 3.0000i$$

## Conjugué d'un nombre complexe

$$\gg x = \text{conj}(z)$$

$x =$

$$4.0000 - 3.0000i$$

## Parties réelle et imaginaire d'un nombre complexe

$$\gg a = \text{real}(z)$$

$a =$

4

$$\gg b = \text{imag}(z)$$

$b =$

3

## Module d'un nombre complexe

$$r = \text{abs}(z) ;$$

»  $r = \text{abs}(z)$   
 $r =$   
5

## Argument d'un nombre complexe

$$\text{theta} = \text{angle}(z) ;$$

»  $\text{theta} = \text{angle}(z)$   
 $\text{theta} =$   
0.6435



Le module est donné en radians.

## Sauvegarde de fichiers de données ou fichiers MAT

➡ commande **save**

### Exemple

- On définit une variable t
- On veut la sauvegarder dans un fichier var.t.mat

```
» t = [1 2 3]
t =
      1  2  3
```

*save nom-fichier variable*

```
» save var.t t
```

- On peut effacer toutes les variables de la mémoire

**clear all**

```
» t
??? Undefined function or variable t
```

## Chargement de fichiers de données ou fichiers MAT

➡ commande **load**

- Si on désire rappeler la variable t

```
load nom-fichier
```

```
» load vart
```

Conséquence : la variable est, de nouveau, connue.

```
» t  
t =  
1 2 3
```

- S'il existe plusieurs variables

```
clear variables
```

```
» t = [1 2 3]; x = 1; y = 2*x;  
» clear t x
```