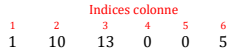


# TP MATLAB N=°2

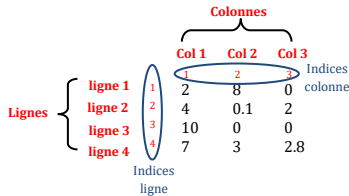
## VECTEURS ET MATRICES

### 1. DÉFINITION

— **Vecteur** : Collection de données de même type (un ensemble d'éléments, ex : des nombres), constitué d'une seule ligne et plusieurs colonnes.

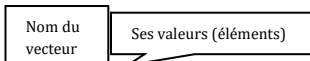


— **Matrice** : Collection de données de même type constituée de plusieurs lignes et plusieurs colonnes.



### 2. CRÉATION (VECTEUR/MATRICE)

On définit un vecteur en donnant la liste de ses éléments entre **crochets** [ ].



```
>> v=[1.1 2.2 3.3]
```

```
v =  
1.1000 2.2000 3.3000
```

Les éléments sont séparés **au choix** par des **espaces** ou par des **virgules**.

```
>> v=[1.1, 2.2, 3.3]
```

```
v =  
1.1000 2.2000 3.3000
```

```
>> v2 = [2 6 3]
```

```
v2 =  
2 6 3
```

Un vecteur ligne de taille 3 (il contient 3 éléments).

```
>> vcol = [6 ; 8 ; 5]
```

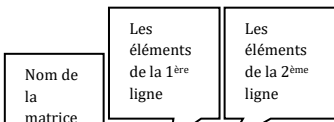
```
vcol =  
6  
8  
5
```

Un vecteur colonne de taille 3.

```
>> vcol = [6 8 5]'
```

```
vcol =  
6  
8  
5
```

Un vecteur colonne également (utilisation de la transposée).



```
>> m=[1 1 1 ; 2 2 2]
```

```
m =  
1 1 1  
2 2 2
```

Une matrice de dimension 2 x 3 (elle contient 2 lignes et 3 colonnes).

```
>> A=[1 2;3 4] % une matrice de dimension 2x2
```

```
A =  
1 2  
3 4
```

```
>> mat=[1 2 0 ; 3 0 5 ; 10 0 8 ; 4 4 78]
```

```
mat =  
1 2 0  
3 0 5  
10 0 8  
4 4 78
```

Une matrice de dimension 4x3

### 3. FONCTIONS SUR LES VECTEURS/MATRICES

- En plus des fonctions mathématiques usuelles (cos, sin, log, exp, etc), il y a d'autres fonctions spécifiques qui peuvent s'appliquer sur les vecteurs et/ou les matrices dont **certaines** sont mentionnées dans le Tableau 1.
- Soit le vecteur v=[1 0 0 78 100 0.1 0.2 8]
- Et soit la matrice m=[1 2 0 0;3 0 5 7; 10 0 8 13 ; 4 4 78 2]

>> min(v)	ans = 0	La fonction min retourne la plus petite valeur (minimum).
>> max(v)	ans = 100	La fonction max retourne la plus grande valeur (maximum).
>> sum(v)	ans = 187.3000	La somme des éléments de v: 1+ 0+ 0+ 78+ 100+ 0.1+ 0.2+ 8
>>mean(v)	ans = 23.4125	La moyenne des éléments de v: (1+ 0+ 0+ 78+ 100+ 0.1+ 0.2+ 8)/8
>>prod(v)	ans = 0	Le produit : 1* 0* 0* 78* 100* 0.1* 0.2* 8
>>length(v)	ans = 8	La taille du vecteur.
>> size(v)	ans = 1 8	La dimension du vecteur: nombre de lignes (toujours 1) et nombre de colonnes.
>> min(m)	ans = 1 0 0 0	Le minimum de chaque colonne
>> max(m)	ans = 10 4 78 13	Le maximum de chaque colonne
>> sum(m)	ans = 18 6 91 22	La somme de chaque colonne
>>mean(m)	ans = 4.5000 1.5000 22.7500 5.5000	
>>prod(m)	ans = 120 0 0 0	
>> size(m)	ans = 4 4	La dimension de la matrice: nombre de lignes et de colonnes de m: 4 lignes et 4 colonnes.
>>det(mat)	ans = -4768	renvoie le déterminant de la matrice carrée mat.

Tableau 1 : Fonctions sur les vecteurs et/ou matrice

#### Remarques :

- Souvent, si l'argument d'entrée des fonctions citées ci-dessus est un **vecteur**, le résultat est un **scalaire** et lorsque ces fonctions sont appliquées à une **matrice**, elles agissent **colonne par colonne** et retournent un **vecteur ligne**.

### 4. ACCÈS

- **VECTEURS : V(indice\_élément)**  

```
>> V = [5 7 9 6 3] ;  
>> V(2) %donne le 2ème élément du vecteur  
ans =  
7  
>> V(3:5) %donne les éléments du 3ème jusqu'au 5ème  
ans =
```

```

9 6 3
>> V(:) %retourne le vecteur sous forme de vecteur
colonne
ans =
5
7
9
6
3
>> V(end) %retourne le dernier élément du vecteur
ans =
3
>> V([2,3:5])
ans =
7 9 6 3
>> V([2,3:5,1])
ans =
7 9 6 3 5
>> V([2,3:5,1,2])
ans =
7 9 6 3 5 7

```

— **MATRICES : M(indice\_ligne, indice\_colonne)**

```

>> M = [5 7 6 ; 4 9 6 ; 7 8 3]
M =
5 7 6
4 9 6
7 8 3
>> M(3,2) %retourne l'élément de l'intersection de la
ligne 3 et de la colonne 2 qui est 8
ans =
8
>> M(2:end,1:2) %retourne les éléments sur la 2ème et
3ème ligne et la 1ère et 2ème colonne
ans =
4 9
7 8
>> M(1,:) %retourne toute la première ligne
ans =
5 7 6
>> M(:,3) %retourne toute la dernière colonne
ans =
6
6
3
>> A = [1 2 0 4 ; 12 13 14 15]
A =
1 2 0 4
12 13 14 15
>> A(2,2)
ans =
13
>> A(2,:)
ans =
12 13 14 15
>> A(:,4)
ans =
4
15
>> A([1 3; 4 8])
ans =
1 2
13 15
>> A([1 3 5 7])
ans =
1 2 0 4
>> A([1 1 1])
ans =
1 2 0 4

```

Dans ce genre d'accès les indices s'inversent, ils deviennent colonne par colonne.

A

1	2	0	4
12	13	14	15

Indices

1 1 1

**5. OPÉRATIONS AVEC LES MATRICES**

— **Addition et soustraction**

Les deux opérateurs sont les mêmes que pour les scalaires, à savoir + et -. À partir du moment où les deux tableaux (vecteurs ou matrices) concernés ont la même taille, le tableau résultant est obtenu en ajoutant ou soustrayant les termes de chaque tableau.

```

>> v1 = [1 0 0 9];
>> v2 = [0 3 2 -1];
>> v1+v2
ans =
1 3 2 8
>> v2-v1
ans =
-1 3 2 -10
>> v1+10
ans =
11 10 10 19
>> v1-3
ans =
-2 -3 -3 6
>> v3 = [1 0];
>> v1+v3

```

??? Error using ==> plus  
Matrix dimensions must agree.

v1 et v3 n'ont pas la même taille

```

>> A = [1 2 3; 0 0 1; 1 0 0]
A =
1 2 3
0 0 1
1 0 0
>> B = [-1 0 3; 1 2 0; 1 1 -2]
B =
-1 0 3
1 2 0
1 1 -2
>> A+B
ans =
0 2 6
1 2 1
2 1 -2
>> A-B
ans =
2 2 0
-1 -2 1
0 -1 2
>> A+5
ans =
6 7 8
5 5 6
6 5 5
>> B-2
ans =
-3 -2 1
-1 0 -2
-1 -1 -4

```

— **Multiplication, division et puissance**

- **Terme à terme** (A.\*B, A./B, .^)
- **Au sens matriciel** (A\*B, A/B, ^)
- Soit les vecteurs v1 et v2 suivants :  
 >>v1=[1 0 0 9];  
 >>v2=[0 3 2 -1];

- Et soit les matrices A et B suivants :  
 >> A=[1 2 3; 0 0 1; 1 0 0]  
 >> B=[-1 0 3; 1 2 0; 1 1 -2]

		Vecteur	Matrice
Matriciel	*	>> v1*v2 ??? Error using ==> mtimes Inner matrix dimensions must agree. <b><u>(Ce calcul n'a pas de sens)</u></b>  >> v1*3 ans = 3 0 0 27	>> A*B ans = 4 7 -3 1 1 -2 -1 0 3  >> A*2 ans = 2 4 6 0 0 2 2 0 0
	/	>> v1/v2 ans = <b>-0.6429</b>  >> v2/2 ans = 0 1.5 1.0 -0.5	>> A/B ans = -3 4 -6 -1 1 -2 -4 3 -6  >> A/2 ans = 0.50 1.00 1.50 0 0 0.50 0.50 0 0
	^	>> v1^v2 ??? Error using ==> mpower At least one operand must be scalar.	>> B^A ??? Error using ==> mpower At least one operand must be scalar.  >> B^2 ans = 4 3 -9 1 4 3 -2 0 7
Terme à terme	.*	>> v1.*v2 ans = 0 0 0 -9	>> A.*B ans = -1 0 9 0 0 0 1 0 0
	./	>> v1./v2 ans = Inf 0 0 -9	>> A./B ans = -1 Inf 1 0 0 Inf 1 0 0
	.^	>> v1.^v2 ans = 1.0 0 0 0.11	>> A.^B ans = 1 1 27 0 0 1 1 0 Inf

## 6. MODIFICATION

Soit le vecteur v suivant :

```
>> v=[4 7 8 10 0 0]
```

```
v =  
4 7 8 10 0 0
```

— Modifier le 3<sup>ème</sup> élément à 100

```
>> v(3)=100
```

```
v =  
4 7 100 10 0 0
```

— Modifier le 3<sup>ème</sup> et le 1<sup>er</sup> élément à 0

```
>> v([3,1])=[0,0]
```

```
v =  
0 7 0 10 0 0
```

— Modifier le 1<sup>er</sup> et le 3<sup>ème</sup> à 1000, et le 4<sup>ème</sup> et le 5<sup>ème</sup> à 9

```
>> v([1,3:5])=[1000,1000,9,9]
```

```
v =  
1000 7 1000 9 9 0
```

```
>> v(2:6)=1
```

```
v =  
1000 1 1 1 1 1
```

Soit la matrice m suivante :

```
>> m=[1 0 0; -1 0 5; 0 0 0; 2 -2 1]
```

```
m =  
1 0 0  
-1 0 5  
0 0 0  
2 -2 1
```

```
>> m(4,2)=100
```

```
m =  
1 0 0  
-1 0 5  
0 0 0  
2 100 1
```

```
> m(2:end,1:2)=0
```

```
m =  
1 0 0  
0 0 5  
0 0 0  
0 0 1
```

## Exercice

1. Créer les vecteurs v1 et v2 suivants :

```
2 10 5 0 1 0 0
```

```
0 8 1 1
```

2. Calculer la somme de v1 et v2.

3. Donner les résultats des commandes suivantes : v1(2), v1(3:5), v2(end), v2(:)

4. Quelle est la commande MATLAB qui nous donne la taille du vecteur v1 ?

5. Modifier le deuxième élément du v2 (donner n'importe quelle valeur numérique).