

Calcul matriciel

Nous avons déjà rencontré les matrices considérées comme des tableaux de données. Toutes les fonctions de création ou de manipulation seront reprises.

Créations de vecteurs et de matrices

`eye(n,p)` : matrice identité de taille $n \times p$

`matrix(A,n,p)` : restructuration du tableau A selon les dimensions n et p

`ones(n,p)` matrice de taille $n \times p$ initialisées avec des 1

`rand(n,p)` matrice de taille $n \times p$ initialisées avec des valeurs aléatoires entre 0 et 1

`zeros(n,p)` matrice de taille $n \times p$ initialisée avec des 0

Opérations sur les matrices

Mais ces matrices sont des objets algébriques avec lesquels les règles de calcul vues en mathématiques vont être utilisées. Si A et B sont deux matrices de taille appropriée pour chaque opération, Scilab connaît les opérations suivantes :

$A + B$: addition matricielle. $A - B$: addition avec l'opposée

$A * B$: produit matriciel noté en mathématiques $A.B$

$A.*B$: produit élément par élément qui est propre à Scilab et qui n'est pas une opération du chapitre mathématique d'Algèbre Linéaire

A/B : division matricielle (qui n'existe pas en tant que telle en mathématiques !!!) $A/B = A * B^{-1}$

De même $A \setminus B$ qui correspond à $A^{-1} * B$

$A./B$ qui est la division élément par élément. De même pour $A.\setminus B$

A' est la transposée de la conjuguée et $A.'$ est la transposée

exemple :

Si $A = \begin{pmatrix} 1 & 1 \\ -2 & -2 \end{pmatrix}$ et $B = \begin{pmatrix} 1 & 1 \\ -1 & -1 \end{pmatrix}$

--> $A * B$, $A.*B$

ans =

! 0. 0. !

! 0. 0. !

ans =

! 1. 1. !

! 2. 2. !

Notons que le produit terme à terme peut très bien s'effectuer grâce à deux boucles imbriquées :

--> $C = \text{zeros}(2,2)$

$C =$

! 0. 0. !

! 0. 0. !

--> for $i=1:2$ do

--> for $j=1:2$ do

--> $C(i,j) = A(i,j) * B(i,j);$

--> end

--> end

--> disp(C)

! 1. 1. !

! 2. 2. !

Premiers exercices

1) Trouver une expression simple qui construit la matrice suivante :

! 1. 0. 0. 0. 0. 0. 1. !

! 0. 1. 0. 0. 0. 0. 2. !

! 0. 0. 1. 0. 0. 0. 3. !

! 0. 0. 0. 1. 0. 0. 4. !

! 0. 0. 0. 0. 1. 0. 5. !

! 0. 0. 0. 0. 0. 1. 6. !

! 1. 2. 3. 4. 5. 6. 7. !

2) Ecrire une fonction calculant le produit scalaire de deux vecteurs réels \vec{u} et \vec{v} :

$$\vec{u} \cdot \vec{v} = \sum_{i=1}^n u_i v_i$$

```
function exo1
A=zeros(7,7);
for i=1:7 do
A(i,i)=1;
A(i,7)=i;
A(7,i)=i;
end
disp(A)
```

```
function [x]=ps1(A,B)
x=sum(A.*B)
```

```
function [x]=ps2(A,B)
x=0;
for i=1:length(A) do //length(A) est le nombre d'élément de A
x=x+A(i)*B(i)
end
```

```
function exo2
A=input('entrez le premier vecteur');
B=input('Entrez la deuxième vecteur');
disp(ps1(A,B))
disp(ps2(A,B))
```