

DÉFINITION DE VECTEURS ET MATRICES

[]	Définition d'un vecteur	<pre>--> x=[1,2,3] x = ! 1. 2. 3. !</pre>
a :h :b	Définition d'un espace constant de pas h entre a et b	<pre>--> 0:.5:1 ans = ! 0. 0.5 1. !</pre>
zeros(n,p) ones(n,p) rand(n,p) eye(n,p)	Définition des matrices de taille $n \times p$: (0), (1), aléatoire et identité	<pre>--> B=eye(2,3) B = ! 1. 0. 0. ! ! 0. 1. 0. !</pre>
A(x,y)	Extrait la sous matrice de A définie par les indices (x_i, y_j)	<pre>--> B(1,1:2) ans = ! 1. 0. !</pre>

MANIPULATION DE VECTEURS ET MATRICES

size(A)	Renvoie le format de la matrice A	<pre>--> size(B) ans = ! 2. 3. !</pre>
diag(A)	Renvoie le vecteur contenant les éléments diagonaux de la matrice A	<pre>--> diag(B) ans = ! 1. ! ! 1. !</pre>
diag(x)	Renvoie la matrice diagonale ayant pour éléments diagonaux les composantes du vecteur x	<pre>--> diag(x) ans = ! 1. 0. 0. ! ! 0. 2. 0. ! ! 0. 0. 3. !</pre>
.#	Opération # élément par élément ($\# \in \{*, /, \wedge\}$)	<pre>--> x.^2 ans = ! 1. 4. 9. !</pre>
A'	Transposée (ou adjointe) de A	<pre>--> b=x' b = ! 1. ! ! 2. ! ! 3. !</pre>
\	Résolution d'un système linéaire (ici $Ay=b$)	<pre>--> A=[0,1,1;1,0,1;1,1,0] --> y=A\b y = ! 2. ! ! 1. ! ! 0. !</pre>

ALGÈBRE LINÉAIRE

norm(A,p) norm(x,p)	Norme p de la matrice A ou du vecteur x . $p \in \mathbb{N}^*$ ou $p='inf', 'fro'$	<pre>--> norm(B,'fro') ans = 1.4142136</pre>
det(A) rank(A) trace(A)	Déterminant, rang et trace de la matrice A	<pre>--> det(A) ans = 2.</pre>
spec(A)	Renvoie la liste des valeurs propres de A	<pre>--> spec(A) ans = ! -1. ! ! -1. ! ! 2. !</pre>
inv(A)	Inverse de la matrice A	<pre>--> inv(A) ans ! -0.5 0.5 0.5 ! ! 0.5 -0.5 0.5 ! ! 0.5 0.5 -0.5 !</pre>

GRAPHIQUES

<pre>plot2d(Mx,My,style, strf,leg,rect,nax)</pre>	Mx et My sont deux matrices de même taille $n \times p$. On dessine les p courbes $My(:,i)$ en fonction de $Mx(:,i)$. Les options sont facultatives (help plot2d pour leur description)
---	---

PROGRAMMATION

EN-TÊTE DE FONCTION function RES=f(DON) ... endfunction	BOUCLE for for I=A:H:B, ... end	BOOLENS & et ou ~ non
STRUCTURE if if cond_1 then ... elseif cond_2 then ... else ... end	BOUCLE while while condition, ... end	= < > <= >= ~= <>