

Lexique Matlab : Manipulation des matrices

Création de vecteur et de Matrices

La manipulation de matrices est particulièrement importante sous Matlab. Voici quelques exemples de commandes pour créer des matrices :

- **a=[1 2 3;4 5 6]** : crée une matrice de 2 lignes et 3 colonnes contenant les éléments 1 2 3 sur la première ligne et 4 5 6 sur la deuxième.
- **b=[1 ;2 ;3]** : Crée un vecteur colonne valant 1 2 3.
- **a=[1,2,3]** (identique **a =[1 2 3]**) : crée le vecteur ligne 1 2 3 .
- **b=[a;1 2 3]** : Crée une matrice dont la première ligne vaut a et la deuxième ligne vaut 1 2 3.
- **a=1:5** : Crée un vecteur valant 1 2 3 4 5.
- **a=1:2:10** : Crée un vecteur valant 1 3 5 7 9.
- **a=[1 3:5]** : Crée un vecteur valant 1 3 4 5.
- **a=(1:3)'** : Crée un vecteur colonne valant 1 2 3 (' permet de faire la transposée).
- **a=zeros(3)** : Crée une matrice de 0 de taille 3x3.
- **a=zeros(2,3,4)** : Crée une matrice de 0 de taille 2x3x4.
- **a=ones(2,4)** : Crée une matrice de 1 de taille 2x4.
- **a=eye(5)** : Crée la matrice identité de taille 5x5
- **a=linspace(1,10,5)** Crée un vecteur de 5 éléments linéairement répartis entre 1 et 10.
- **a=logspace(1,10,5)** Crée un vecteur de 5 éléments logarithmiquement répartis entre 1 et 10.

Petit exemple : Créer avec la ligne de commande la plus courte possible la matrice A valant : 1 2 3 4 5 6 7 8 9

Solution : 2 4 6 8 0 1 2 3 4
A=[1:9;2:2:8 0:4;2*ones(1,9)] 2 2 2 2 2 2 2 2

Accès aux éléments et sous-matrices

Soit la matrice **a=[1 2 3;4 5 6]** et le vecteur **b=[1 2 3]**.

IMPORTANT : les indices des matrices Matlab commencent à 1 (et non 0 comme en C).

- **a(1,2)=4** : Affecte 4 à l'élément de la première ligne et deuxième colonne de la matrice a.
- **b(3)=4** : Affecte 4 au troisième élément du vecteur b.
- **a(1,:)** : Sous matrice correspondant à la première ligne.
- **a(:,2)** : Sous matrice correspondant à la deuxième colonne.
- **a(2,2:end)** : Sous matrice correspondant à la deuxième ligne, colonne 2 à fin(soit 3).
- **a(2,[1 end-1])** : Sous matrice correspondant à la deuxième ligne, colonne 1 et fin-1(soit 2).
- **a([2 3],[1 2])=[1 2;3 4]** : Affecte [1 2;3 4] à la sous matrice formé par les lignes 2 et 3, colonne 1 et 2.
- **a(:)** : Vecteur colonne formé par tous les éléments de a.
- **find(g==5)** : Donne les positions des éléments de g qui sont égaux à 5 (**find** est une fonction qui renvoie les positions des éléments non nuls)

Logical indexing :

Il est possible de construire une sous matrice en appliquant un masque à la matrice principale. Ex : **v=[1 2 3 4];**
v(logical([1 0 1 0])) retourne [1 3] et **v(v>2)** retourne [3 4] (Equivalent à **v(find(v>2))** mais plus rapide)

Opérations sur les matrices

Prenons **A=[1+i 2+i 0;1 2 3;1 i 0]** une matrice complexe et **B=[1 2 3;4 5 6;7 8 9]**.

- **A'** : Adjointe de A (Conjugué de la transposée).
- **A.'** : Transposée de A.
- **C=B^-1** : Inverse de B.
- **C*B** : Produit matriciel.
- **C.*B** : Produit terme à terme.
- **C+B** : Somme.
- **C^2** : C*C (puissance matricielle).
- **C.^B** : Puissance terme à terme.
- **C./B** : Division terme à terme.
- **B\C** : Equivalent à **B^-1*C** mais plus rapide car ne calcule pas l'inverse de B (... et donc moins d'erreur).
- **B/C** : Equivalent à **(C\B)'**.

Note : On peut remplacer un des opérandes par un scalaire. Par exemple **B+2** ajoute 2 à chacun des termes de B.

Lexique Matlab : Manipulation des matrices

Fonctions usuelles sur les matrices

Prenons $B=[1\ 2\ 3;4\ 5\ 6;7\ 8\ 9]$, $v=[1\ 0\ 1]$

- **size(B)** : Vecteur correspondant à la taille de chacune des dimensions de B (ici [3 3]).
- **size(B,1)** : Vecteur correspondant à la taille de la première dimension de B.
- **length(v)** : Nombre d'éléments dans v (Souvent utilisé quand v peut être un vecteur ligne ou colonne).
- **length(B)** : Nombre total d'éléments dans B (ici 9).
- **f=diag(B)** : Vecteur colonne contenant la diagonale de B.
- **diag(f)** : Matrice diagonale dont la diagonale correspond à f.
- **det(B)** : Déterminant.
- **norm(B)** : Norme de B (voir les options pour choisir la norme).
- **rank(B)** : Rang.
- **trace(B)** : Trace.

Statistiques : Chacune de ses fonctions peut être appliquée sur un vecteur ligne ou colonne. Si elles sont appliquées sur des matrices chacune des colonnes est traitée indépendamment formant ainsi un vecteur ligne de résultats.

- **sum(B)** : Somme.
- **prod(B)** : Produit.
- **mean(B)** : Moyenne.
- **std(B)** : Ecart type.
- **var(B)** : Variance corrigée (sans biais).
- **var(B,1)** : Variance.
- **max(B)** : maximum.
- **min(B)** : minimum.
- **cumsum(v)** : Somme cumulée (ici [1 1 2]).
- **cumprod(v)** : Produit cumulée (ici [1 0 0]).
- **rand(3,4)** : Renvoie une matrice 3x4 dont chaque terme suit une loi uniforme sur [0 ;1]
- **randn(3,4)** : Renvoie une matrice 3x4 dont chaque terme suit une loi gaussienne.
- **null(h)** : Matrice orthogonale. ($h*\text{null}(h)=0$)
- **pinv(h)** : Pseudo inverse.
- **sort(R)** : Trie les éléments de chacune des colonnes indépendamment.
- **sortrows(R,3)** : Trie les lignes par rapport aux éléments de la troisième colonne.

Fonctions avancées

Fonctions d'analyse numérique

- **lu** : Décomposition LU.
- **chol** : Cholesky.
- **qr** : Décomposition QR.
- **qzhess** : Hessenberg.
- **schur** : Schur.
- **housh** : Householder.
- **krylov** : Krylov.
- **cond(C)** : calcule le conditionnement de la matrice C (« norme 2 » par défaut)
- **[P,D]=eig(C)** : Diagonalisation de C : $P*D*P^{-1}$
- **compan(p)** : matrice compagnon de p

Fonctions de transformation de matrices :

- **reshape([1 2 3 4 5 6],2,3)** : renvoie [1 2 3;4 5 6] (Recrée une matrice à partir d'un vecteur).
- **fliplr([1 2 3;4 5 6])** : renvoie [3 2 1;6 5 4] (replace les éléments de droite à gauche).
- **flipud([1 2 3;4 5 6])** : renvoie [4 5 6;1 2 3] (replace les éléments de bas en haut).
- **flipdim** : Replace les éléments dans l'ordre inversé suivant la dimension spécifié.
- **tril(B)** : matrice triangulaire inférieure de B
- **triu(B)** : matrice triangulaire supérieure de B