

Informatique Matlab : révision

1. Généralités

help nom_fonction : affiche l'aide de la fonction nom_fonction.

whos : affiche la liste des variables actuellement en mémoire.

print -dpdf toto.pdf : imprime la figure courante dans le fichier toto.pdf.

sortie='toto.pdf' ; print('-dpdf',sortie) : idem.

load toto : lit le fichier dont le nom est toto.mat

Il y'a différentes façons d'utiliser matlab :

- En mode interactif : cela consiste à taper les commandes directement dans la console matlab. La touche 'tabulation' permet de compléter automatiquement le nom de la commande qu'on est entrain de taper. Les flèches haut/bas permettent de rappeler les commandes tapées précédemment. Plusieurs commandes peuvent être tapées sur la même ligne si on les sépare par un ';'.
- Avec des scripts : les scripts sont des fichiers .m dont les lignes sont exécutées une par une.
- avec des fonctions.

2. Création de vecteurs/matrices

V=[1 2 3 4 ; 5 6 7 8] : crée une matrice $V = \begin{pmatrix} 1 & 2 & 3 & 4 \\ 5 & 6 & 7 & 8 \end{pmatrix}$

V=[1 :3 :12] : $V=[1 \ 3 \ 7 \ 10]$;

V=zeros(2,5) : $V = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{pmatrix}$

V=ones(2,5) : $V = \begin{pmatrix} 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \end{pmatrix}$

3. Accéder aux éléments d'une matrice

On suppose qu'on a une matrice $A = \begin{pmatrix} 1 & 2 & 3 & 4 \\ 5 & 6 & 7 & 8 \end{pmatrix}$

On peut accéder aux éléments de A en spécifiant 2 coordonnées :

A(1,1)=1 ;

A(2,1)=5 ;

A(2,3)=7 ;

A(2,1 :end-1)=[5 6 7] (2ème ligne tous les éléments sauf le dernier) ;

A(:,2)=[2 6] (2ème colonne) ;

Ou en ne spécifiant qu'une seule coordonnée. Dans ce cas A est comme un vecteur contenant chaque ligne de la matrice :

A(:)=[1 ; 5 ; 2 ; 6 ; 3 ; 7 ; 4 ; 8] : vecteur colonne contenant tous les éléments de A

A(2 :4)=[5 ; 2 ; 6 ; 3] : vecteur colonne contenant les 2 à 4 de A.

A([5 3 6])=[3 ; 2 ; 7] : vecteur colonne contenant le 5ème, 2ème et 7ème élément de A.

A([1 :3 :end]) : ne garde qu'un élément sur 3 de A cad [1 6 4].

A(2 :end-2)=[5 ; 2 ; 6 ; 3 ; 7] ;

4. Manipulations simples de matrices

On suppose une matrice $A = \begin{pmatrix} 1 & 2 & 3 & 4 \\ 5 & 6 & 7 & 8 \end{pmatrix}$

A'=[1 5 ; 2 6 ; 3 7 ; 4 8]= $\begin{pmatrix} 1 & 5 \\ 2 & 6 \\ 3 & 7 \\ 4 & 8 \end{pmatrix}$ = transposé de A

flipr(A)= $\begin{pmatrix} 4 & 3 & 2 & 1 \\ 8 & 7 & 6 & 5 \end{pmatrix}$: inverse l'ordre des colonnes ('retourne' la matrice horizontalement)

flipud(A)= $\begin{pmatrix} 5 & 6 & 7 & 8 \\ 1 & 2 & 3 & 4 \end{pmatrix}$: inverse l'ordre des lignes ('retourne la matrice verticalement)

repmat(A,[2 1]) = $\begin{pmatrix} 5 & 6 & 7 & 8 \\ 1 & 2 & 3 & 4 \\ 5 & 6 & 7 & 8 \\ 1 & 2 & 3 & 4 \end{pmatrix}$: réplique la matrice A 2 fois verticalement.

repmat(A,[1 2]) = $\begin{pmatrix} 5 & 6 & 7 & 8 & 5 & 6 & 7 & 8 \\ 1 & 2 & 3 & 4 & 1 & 2 & 3 & 4 \end{pmatrix}$ réplique la matrice A 2 fois horizontalement.

size(A)=[2 4] : vecteur indiquant le nombre de ligne et de colonne de A.

numel(A)=8 : nombre d'éléments (de nombre) contenus dans la matrice A.

I=find(A(:) > 5 & A(:) <= 7) : renvoie les indices des éléments de A strictement supérieur à 5 et inférieur ou égale à 7.

A(I)=[7 8] = éléments de A respectant les conditions du **find**.

I=find(u<1 & v >6) : I= indice des éléments tel que u(I) < 1 et v(I) >6, u et v étant deux vecteurs de la même taille.

Soient u=[1 2 3] et v=[4 5 6] deux vecteurs

[u v]=[1 2 3 4 5 6] : concatène deux matrices/vecteurs u et v en les mettant côte à côte. Ils doivent avoir le même nombre de ligne.

[u; v] : [1 2 3; 4 5 6] : concatène deux matrices en les mettant l'une au dessus de l'autre.

u.*v=[4 10 18] : multiplication terme à terme de u et v.

u*v' = 32 : produit scalaire du vecteur ligne u et du vecteur colonne v'.

u'*v = $\begin{pmatrix} 4 & 5 & 6 \\ 8 & 10 & 12 \\ 12 & 15 & 18 \end{pmatrix}$: multiplication usuelle du vecteur colonne u' et du vecteur ligne v.

Les chaînes de caractères sont en fait des matrices contenant des lettres. Elles se manipulent comme des matrices usuelles :

titre='toto' : la variable titre contient toto

[titre,'-',num2str(4),'pdf'] : toto-4.pdf

5. Tracé de fonction 2D

Soient u=[1 2 3] et v=[4 5 6]

plot(u,v,'-k','linewidth',2) : dessine le vecteur v en fonction de u, en noir ('k') avec un trait d'épaisseur 2.

plot(u,v,'o') : dessine v en fonction de u, en mettant des ronds en chaque point.

plot(u,v,'o','MarkerSize',14,'MarkerFaceColor','r','MarkerEdgeColor','k') : idem mais avec des ronds ('o') rouges ('r') avec des bords noirs ('k'), de taille 14.

xlabel('temps','fontSize',16) : écrit 'temps' en taille 16 sous l'axe des x.

ylabel('v','fontSize',16) : idem pour l'axe des y.

titre('v en fonction de u','fontSize',18) : met un titre en taille 18.

legend(v(u)) : ajoute une légende.

xlim([1 3]) : indique que l'on souhaite que l'axe des x aille de 1 à 3.

ylim([4 6]) : idem pour l'axe des y.

imagesc(A) : Affiche la matrice A à l'écran

colorbar : Affiche la colorbar

colormap(jet) : choisit la colormap 'jet' ;

colormap(hot(4)) : choisit la colormap 'hot' en précisant qu'on ne souhaite que 4 couleurs.

caxis([0 4]) : règle l'échelle des couleurs entre 0 et 4.

pcolor(A) : affiche la matrice A avec pcolor qui est plus flexible de imagesc.

pcolor(X,Y,A) : affiche la matrice A, les coordonnées de chaque point étant spécifiées par les matrices X et Y. X,Y,A ont donc la même taille.

shading('interp') : permet d'afficher les couleurs suite à pcolor.

view([0 90]) : on regarde la matrice en vu 'du dessus' (choix par défaut).

view([45 80]) : on regarde la matrice de biais (3D).

6. Tracé de fonction 3D

surf(A) : dessine la matrice A en 3D.

surf(X,Y,A) : dessine la matrice A en 3D, les coordonnées de chaque point étant spécifiées par les matrices X et Y. X,Y,A ont donc la même taille.

view(45,80) : spécifie l'angle selon lequel on regarde l'image.

light : rajoute une source de lumière.

lighting gouraud : plus joli.

material dull, material shiny, material metal : spécifie comment la lumière est réfléchi par la surface.

plot3(x,y,z) : idem que plot mais en 3D : dessine une courbe 3D, les points définissant la courbe étant de coordonnées x,y,z

7. Boucles for

for ix=1 :3

...

end

for ix=[1 :2 :10]

...

end

for ix=[1 5 10 2]

...

end

for ix=find(...)

...

end