

Calcul Numérique - Cheat Sheet Matlab

Aubin SIONVILLE

Télécom St Etienne 2023-2024

Variables

`a = 1` : `a` prend la valeur 1
`x = [1,2,3]` : `x` est le vecteur ligne [1,2,3]
`x = [1;2;3]` : `x` est le vecteur colonne [1;2;3]
`A = [1,2,3;4,5,6]` : `A` est une matrice 2×3
`x(2) = 4` : `x` devient [1,4,3]
`A(1,2) = 0` : `A` devient [1,0,3;4,5,6]

Opérations sur les variables

`1+2`, `3-4`, `5*6`, `7/8` : addition, soustraction, multiplication, division
`1^2` : puissance
`sqrt(4)` : racine carrée
`exp(1)` : exponentielle
`expm1(1)` : exponentielle moins 1
`log(2)` : logarithme népérien
`log1p(2)` : logarithme népérien plus 1
`log10(10)` : logarithme décimal
`sin(pi)`, `cos(2*pi)`, `tan(pi/6)` : sinus, cosinus, tangente
`abs(-1)` : valeur absolue
`floor(1.5)`, `ceil(1.5)` : partie entière par défaut, par excès

Création de matrices et vecteurs

`zeros(n,m)` : matrice nulle $n \times m$
`ones(n,m)` : matrice de 1 $n \times m$
`eye(n)` : I_n `eye(n,m)` : matrice $n \times m$ avec des 1 sur la diagonale
`rand(n,m)` : matrice $n \times m$ de nombres aléatoires entre 0 et 1
`linspace(a,b,n)` : vecteur ligne de n valeurs équiréparties entre a et b inclus

Opérations sur les matrices et vecteurs

`A+B`, `A-B`, `x+k` : addition, soustraction, addition d'un scalaire
`A*B`, `A*x` : multiplication matricielle, multiplication matrice-vecteur
`A.*B` : multiplication terme à terme
`k*A`, `k*x` : multiplication par un scalaire
`A^n` : puissance
`A.^n` : puissance terme à terme
`A\B` : résolution de $AX = B$
`A'` : transposée de A
`inv(A)` : inverse de A
`det(A)` : déterminant de A
`x(a:b)` : vecteur ligne des éléments de `x` d'indice `a` à `b`
`A(n,:)` : n -ième vecteur ligne de A
`A(:,n)` : n -ième vecteur colonne de A
`A(5, 1:3)` : éléments de la 5ème ligne de A d'indice 1 à 3

Graphiques

`plot(x,y)` : affiche le graphe de y en fonction de x
`plot(x1,y1,x2,y2)` : affiche les graphes de y_1 et y_2 en fonction de x_1 et x_2
`semilogx(x,y)` : affiche le graphe de y en fonction de $\log(x)$
`semilogy(x,y)` : affiche le graphe de $\log(y)$ en fonction de x
`loglog(x,y)` : affiche le graphe de $\log(y)$ en fonction de $\log(x)$
`plot(x,y,'o')` : affiche le graphe de y en fonction de x avec des points
`title('titre')` : ajoute un titre au graphe
`xlabel('titre')` : ajoute un titre à l'axe des abscisses
`ylabel('titre')` : ajoute un titre à l'axe des ordonnées
`legend('courbe1','courbe2')` : ajoute une légende au graphe
`grid on` : ajoute une grille au graphe
`axis([xmin,xmax,ymin,ymax])` : définit les bornes des axes
`hold on` : continue de dessiner sur le même graphe
`hold off` : arrête de dessiner sur le même graphe
`figure` : crée une nouvelle fenêtre de graphe

Fonctions

```
function output = nom_fonction(input)
    output = ...
end
```

Fonctions anonymes

`f = @(x) x^2` : définit la "fonction anonyme" $f(x) = x^2$

Logique

`a == b` : a est égal à b ?
`a ~= b` : a est différent de b ?
`a > b` , `a < b` : a est supérieur, inférieur à b ?
`a >= b` , `a <= b` : a est supérieur ou égal, inférieur ou égal à b ?
`cond1 && cond2` : cond1 et cond2 sont vraies ?
`cond1 || cond2` : cond1 ou cond2 est vraie ?

Conditions et boucles

```
if condition1
    ...
elseif condition2
    ...
else
    ...
end

for i = 1:n          (for i = 1:k:n pour un pas de k)
    ...
end

while condition
    ...
end
```

Misc

`[ligne de code];` : exécute la ligne de code sans afficher le résultat