Introduction à Scilab

Fabien Navarro

1 Introduction

Scilab est un logiciel libre de calcul numérique développé par par l'inria (institut national de recherche en informatique et automatique). Il est disponible à `http://www.scilab.org/`.

Commencer par lancer Scilab en tapant dans un console la commande suivante : `:~$ scilab`

Le logiciel se lance et vous obtenez une fenêtre avec une invite de commande (le prompt) du type suivant : `-->`

Cependant, assez rapidement, on est amené à écrire des scripts (suite d'instructions *Scilab*), puis des fonctions et il est préfèrable de travailler avec un éditeur de texte. *Scilab* dispose de son propre editeur (accessible via le bouton ou en tapant `edit`). Il est possible d'utiliser un éditeur standard comme *kate*.

2 La syntaxe et les types de données

Les majuscules et minuscules sont différenciées. Les lignes commençant par `//` ne sont pas interprétées, ce sont des commentaires. Chaque ligne doit se terminer par un point-virgule. Les chaines de caractères sont entre guillemets.

Les constantes mathématiques sont précédées du caractère `%`. On a $i =$`%i`, $\pi =$`%pi`, $e =$`%e`, . . .

des problèmes d'encodage peuvent arriver d'un système à l'autre : éviter d'écrire les noms de fichiers avec des caractères accentués et éviter les accents dans vos scripts.

3 Utilisation de l'aide

Les commandes principales pour lancer l'aide de *Scilab* sont les commandes `help nom_fonction` et `apropos nom_fonction`. On peut également disposer d'une interface graphique en tapant simplement `help` dans l'invite de commande.

4 Matrices

Un des types de base de *Scilab* est constitué par les matrices de nombres réels ou complexes.

La façon la plus simple de définir une matrice (ou un vecteur, ou un scalaire qui ne sont que des matrices particulières) est d'entrer au clavier la liste de ses éléments, en adoptant les conventions suivantes :

— les éléments d'une même ligne sont séparés par des espaces ou des virgules ;
— la liste des éléments doit être entouré de crochets [] ;
— chaque ligne, sauf la dernière, doit se terminer par un point-virgule.

Par exemple, la commande :

```
A=[1 1 1;2 4 8;3 9 2]
```

produit la sortie :

```
 A =
! 1. 1. 1. !
! 2. 4. 8. !
! 3. 9. 2. !
```

En fait si vous terminez l'instruction par un point virgule, le résultat n'apparaît pas à l'écran. Essayer par exemple

```
-->b=[2 10 44 190];
```

pour voir le contenu du vecteur ligne b, on tape simplement :

```
-->b
```

Une instruction très longue peut être écrite sur plusieurs lignes en écrivant trois points à la fin de chaque ligne à poursuivre :

```
-->T = [ 1 0 0 0 0 0 ;...
--> 1 2 0 0 0 0 ;...
--> 1 2 3 0 0 0 ;...
--> 1 2 3 0 0 0 ;...
--> 1 2 3 4 0 0 ;...
--> 1 2 3 4 5 0 ;...
--> 1 2 3 4 5 6 ]
```

Il existe des fonctions pour construire et manipuler des matrices :

`eye(n,m)`	matrice identité de taille $n \times m$.
`ones(n,m)`	matrice de taille $n \times m$ ne contenant que des 1.
`zeros(n,m)`	matrice de taille $n \times m$ ne contenant que des 0.
`triu(M)`	partie triangulaire supérieure de la matrice M.
`tril(M)`	partie triangulaire inférieure de la matrice M.
`rand(n,m)`	matrice aléatoire de taille $n \times m$ à coefficients réels dans $[0, 1[$ suivant la loi uniforme.
`A'`	la transposée de la matrice A.
`size(A)`	la taille de la matrice A.
`length(A)`	le nombre d'éléments de la matrice A.

Les vecteurs sont simplement des matrices dont l'une des tailles vaut 1. On parle donc de vecteur ligne ou colonne selon les cas. Voici quelques fonctions pour créer et manipuler des vecteurs :

`linspace(a,b,n)`	vecteur x à n composantes régulièrement réparties entre x_1 et x_n, c.a.d que l'on a $n-1$ intervalles et les coordonnées de x vérifient :$x_{i+1} - x_i = \frac{x_n - x_1}{n-1}$
`y=a:p:b`	vecteur y dont on a spécifié l'incrément : on part de a puis on. ajoute p à chaque étape sans dépasser b.
`i=1:5`	lorsque l'incrḿent vaut 1 on peut se permettre de l'omettre.

Pour obtenir le coefficient i, j d'une matrice A, il suffit d'utiliser la syntaxe $A(i, j)$. Dans le cas d'un vecteur v en ligne ou en colonne, la syntaxe est $v(i)$.

Il est également possible de concaténer des vecteurs ou des matrices. Il suffit d'utiliser la même syntaxe que pour créer un vecteur ou une matrice en remplaçant les valeurs par une matrice ou un vecteur. Par exemple :

```
--> a=[1 2 ; 3 4]; b=[7 8]; d=[a ; b]
```

Il faut tout de même que la taille des objets soit compatible.

5 Exercices

Dans l'exo 1, nous allons utiliser *Scilab* comme une calculette. Ensuite il sera préférable d'utiliser un éditeur.

Exercice 1. Tester les commandes suivantes :

1) Calculer i^2, $\cos(\pi/3)$, e^1, $(1+2i)^3$, $\frac{1+5i}{6i+2}$.
2) Construire deux variables A et B contenant respectivement les valeurs `Vrai` et `Faux`.
3) Affecter l'entier 5 une variable a.
4) Affectation d'un vecteur :`-->u=[3,4,5,6]`.
5) Que donnent les calculs suivants ?
   `-->size(u)`,
   `-->length(u)`,
   `-->2+u`,
   `-->3*u`,
   `-->u/10`,
   `-->exp(u)`,
   `-->u*u`,
   `-->u.*u`,
   `-->[u 10]`,
   `-->u'`,
   `-->u(3)`,
   `-->u(5)=10`
6) Affectation d'une matrice :
   `-->M=[3,4;5,6]`.
7) Concaténation :
   `-->N=[M,M]`,
   `-->P=[M;M]`,
   `-->Q=[M,[1;2]]`,
   `-->[M;[1,2]]` ,
   `-->P(5,:)=[7,8]`
8) Extraction :
   `-->M(2,2)`,
   `-->N(:,1)`,
   `-->N(2,:)`,
   `-->N(1,[2,3])`,
   `-->M(1,$)`,
   `--> N($-1,$-1)`
   ($ représente le plus grand indice possible)
9) Comparer :
   `-->M*M`,
   `-->M.*M`,

```
-->M^3,
-->M.^3.
```

Exercice 2.

1) Créer un nouveau dossier sur le disque dur, dans votre répertoire personnel (sur le bureau, dans vos documents ou sur une clé USB) où seront enregistrés vos programmes le temps de la séance (placer vous dans le dossier, faire un clic droit et choisisser ouvrir un terminal dans le dossier puis taper `scilab4`).
2) Ouvrir l'éditeur de *Scilab* (ou celui de votre choix) et créer un fichier nommé `essai1.sce` contenant les instructions suivantes : `A = rand(4,4); B = inv(A); C = A*B`
(Par convention, les extensions `.sce` sont utilisées pour les fichiers de commandes et `.sci` pour les fichiers de fonctions)
3) Exécuter le script précédent en allant dans la fenêtre *Scilab* et en utilisant l'option Fichier/Exécuter dans la barre de menu (ou dans la console
`-->exec('chemin_fichier/nomfichier.sce')` attention au chemin, taper `pwd` dans le prompt de *Scilab*).

Exercice 3. Soit

$$A = \begin{pmatrix} 1 & 0 & 1 \\ 0 & 1 & 2 \\ 0 & 0 & 4 \end{pmatrix}$$

1) Rechercher de l'aide sur le déterminant et calculer le déterminant de A.
2) Calculer le rang puis l'inverse de A et calculer $A^{-1}A$.
3) Rechercher de l'aide sur les valeurs propres d'une matrice (vous pouvez utiliser la commande `apropos`) et calculer le spectre (l'ensemble des valeurs propres) de A.
4) On suppose maintenant que $A_{1,2} = 6$. Refaire les calculs précédents.

Exercice 4.

1) Rechercher de l'aide sur la fonction `poly`.
2) Définir le polynôme $P = (X - 1)(X + 1)(X + 2)$.
3) Définir le polynôme $Q = 1 + 2X + 3X^2 + 4X^4$.
4) Rechercher de l'aide sur les racines d'un polyôme.
5) Calculer les racines de P et de Q.
6) Calculer le produit PQ.

Exercice 5. On considère la matrice A définie dans l'exercice 3.

1) Construire la matrice $M = XI - A$.
2) Calculer $R = \det(M)$.
3) Calculer les racines de R et comparer les aux valeurs propres de A. En déduire la nature de R.

Définition 1. (rayon spectral)

Le rayon spectral d'une matrice $A \in \mathcal{M}_n(\mathbb{R})$, noté $\rho(A)$, est défini de la manière suivante :

$$\rho(A) = \max_{1 \leq i \leq n} |\lambda_i(A)|, \tag{1}$$

où $\lambda_1(A), \ldots, \lambda_n(A)$ sont les valeurs propres de A.

Proposition 1. (normes matricielles)

Soit $A \in \mathcal{M}_n(\mathbb{R})$ une matrice quelconque. Alors on a :

$$\|A\|_1 = \max_{1\leq j\leq n} \sum_{i=1}^{n} |a_{ij}|,$$
$$\|A\|_2 = \sqrt{\rho(A^T A)},$$
$$\|A\|_\infty = \max_{1\leq i\leq n} \sum_{i=1}^{n} |a_{ij}|.$$

Définition 2. (conditionnement)

— *Le conditionnement d'une matrice inversible A relativement à une norme subordonnée, notée $\|\cdot\|_p$ est le nombre défini par :*

$$\mathrm{cond}_p(A) = \|A\|_p \|A^{-1}\|_p. \tag{2}$$

Si on considère un système linéaire du type $Ax = b$, ce nombre mesure la sensibilité de la solution x du système linéaire vis-à-vis des variations sur les données A et b (Le conditionnement dépend de la norme choisie et plus il est grand, plus la solution est sensible aux perturbations des données. Par défault la norme 2 est utilisée).

— *Soit A une matrice symétrique définie positive, alors on a*

$$\mathrm{cond}_2(A) = \frac{\lambda_{\max}(A)}{\lambda_{\min}(A)} = \rho(A)\rho(A^{-1}). \tag{3}$$

où $\lambda_{\max}(A)$ et $\lambda_{\min}(A)$ désigne la plus grande et la plus petite valeur propre de A.

Exercice 6. Soit $A \in \mathcal{M}_n(\mathbb{R})$

$$A = \begin{pmatrix} 2 & -1 & 0 & \dots & 0 \\ -1 & \ddots & \ddots & \ddots & \vdots \\ 0 & \ddots & \ddots & \ddots & 0 \\ \vdots & \ddots & \ddots & \ddots & -1 \\ 0 & \dots & 0 & -1 & 2 \end{pmatrix}$$

1) Rechercher de l'aide sur la fonction `diag`.
2) Tester la commande `diag(diag(ones(n,n)))` (avec $n = 10$ par exemple) puis tester `diag(diag(ones(n,n)),1)`.
3) Construire la matrice A (de taille $n \times n$).
4) Rechercher de l'aide sur la fonction `toeplitz`.
5) Construire la matrice A en utilisant cette fonction et la fonction `zeros`.
6) Pour $n = 10$, trouver les normes matricielles 1,2, et ∞ de la matrice A (vous pouvez utiliser l'aide sur la fonction `norm` ou alors utiliser la Proposition 1.) et comparer $\|A\|_1$ avec $\|A\|_\infty$.
7) Calculer le rayon spectral de A en utilisant la définition (1) et comparer $\|A\|_2$ avec $\rho(A)$.
8) Calculer la matrice transposée $A^\intercal$ de A. Que peut-on en déduire sur la matrice A ?
9) Calculer le spectre de A. Que peut-on en déduire sur la matrice A ?
10) Calculer le conditionnement cond_p de la matrice A pour $p = 1, 2, \infty$ (vous pouvez utiliser les équations (2) et (3) ou encore la fonction `cond` quand cela est possible).

11) On suppose maintenant que $A_{1,2} = -2000$. Refaire les questions 6,. . .,10.

Exercice 7. On considère la matrice de Hilbert B de taille $n \times n$ de terme général $b_{i,j} = \frac{1}{i+j-1}$

$$B_n = \begin{pmatrix} 1 & \frac{1}{2} & \cdots & \frac{1}{n} \\ \frac{1}{2} & \frac{1}{3} & \cdots & \frac{1}{n+1} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{1}{n} & \frac{1}{n+1} & \cdots & \frac{1}{2n-1} \end{pmatrix}$$

1) Écrire une fonction `matriceB` qui prend en argument la taille de la matrice B_n et qui renvoie B_n.
2) Pour $n = 5$, calculer la matrice transposée $A^{\intercal}$ de B_5. Que peut-on en déduire sur la matrice B_5 ?
3) Calculer le spectre de B_5. Que peut-on en déduire sur la matrice B_5 ?
4) Calculer maintenant $\text{cond}_2(B_n)$ pour $n = 2, \dots, 7$.