

2011.2.15

数理計画法、授業資料

せつかくなので、計算機を使って演習をしましょう

Scilab (サイラボ) を使います。

Scilab, A Free Scientific Software Package

Scilab は工学・理学での応用に役立つ強力でオープンな計算環境を提供する数値計算のためのソフトウェアパッケージです。Scilab には何百もの数学関数が含まれ、加えて、様々なプログラミング言語(C, C++, Fortran...)で作成されたプログラムを必要に応じて取り込む機能を備えています。洗練されたデータ構造(リスト、多項式、有理関数、線形システム...)を扱うことができる仕組みと、インタプリタ、高度なプログラミング言語を備えています。

Scilab ホームページ (<http://www.scilab.org>)から抜粋・直訳

よく似たソフトウェアに MATLAB や octave などがあります。scilab や octave には、商用で非常に広く使われている MATLAB に対するフリー・オープンな代替品としての側面があります。ただし、スクリプト言語の仕様は完全に互換ではありません。また、scilab は積極的に独自の機能・仕様を採用し、MATLAB からは徐々に離れつつあると言われています。その一方で octave はユーザインタフェースを含めてそっくりになるモードを備える等、MATLAB 互換路線を継続するようです。興味があれば octave も試してみてください。

今日の予定

1. scilab 練習
2. 演習課題

1 scilab 練習

起動と終了

```
% scilab[Enter]
```

```
quit[Enter]
```

基本操作

コンソールに直接コマンドを入力すれば、複素数や行列を扱うことのできる計算機として動作します。

入力履歴の表示や編集も可能です。

計算

```
1+2[Enter]
```

```
3*3[Enter]
```

```
2 ^ 0.5[Enter]
```

変数

```
X=4.2[Enter]
```

※変数名はアルファベットで始まるアルファベット/数字/アンダースコア

定数

```
%pi[Enter]
```

```
%e[Enter]
```

複素数

```
1+%i*2[Enter]
```

```
Z=2+%i*3[Enter]
```

```
Z*Z[Enter]
```

```
%i ^ 0.5[Enter]
```

行列

```
[ 1 2 ; 3 4 ][Enter]
```

```
%eye(2,2)[Enter]
```

```
%zeros(2,2)[Enter]
%ones(2,2)[Enter]
A=[1 2 ; 3 4][Enter]
A'[Enter]
A*A[Enter]
%eye/A
```

※四則演算、割り算を使った連立方程式の解法・逆行列の計算等ができる。

行列の要素どうしの演算

```
A=[1 2; 3 4][Enter]
B=A'[Enter]
A.*B[Enter]
A./B[Enter]
A+%i*B[Enter]
(A+%i*B)'[Enter]
```

※基本的に全ての数値は複素数を要素とする行列

便利な機能

継続行

... を行末に置く

```
A=[1 2 3;...
4 5 6][Enter]
```

等間隔ベクトル

初期値:増加量:終値

```
A=1:0.5:3[Enter]
```

zeros(), ones(), eye()

引数が複数の整数: $n_1 \times n_2 \times \dots$ 行列、

引数が行列: 引数と同じ大きさの行列、

```
A=zeros(2,4)[Enter]
```

```
B=ones([1:4])[Enter]
```

```
C=eye(A' B')[Enter]
```

デリミタ「;」

行末にセミコロンを置くと、結果出力が抑制されます。

プログラム中では計算結果を表示する必要がある場合以外は「;」を忘れずに。

```
A=zeros(2,4); [Enter]
```

※セミコロンを着けなかった場合と比較してください。

関数定義

```
function [x,y]=myfct(a,b)
```

```
x=a+b
```

```
y=a-b
```

```
endfunction
```

```
myfct(1,2)[Enter]
```

```
myfct([1 2; 3 4],[1 2; 3 4])[Enter]
```

※別ファイルで定義し、getf()で読み込むこともできる。

```
getf('functiondefinition.sci')[Enter]
```

グラフィックス

```
t=[0:0.05:1][Enter]
```

```
ct=cos(2*%pi*t)[Enter]
```

```
plot2d(t,ct)[Enter]
```

```
square(-0.5,-1.5,1.5,1.5)[Enter]
```

```
tht=2*%pi*t[Enter]
```

```
plot2d(tht,ct)[Enter]
```

```
square(-%pi/2,-1.5,2.5*%pi,1.5)[Enter]
```

グラフィックス 2

```
x1 = [-1:0.1:1];
```

```
x2 = [-1:0.1:1];
```

```
[xx,yy] = meshgrid(x,y);
```

```
plot3d(x,y,xx.^3+yy.^3-3*xx.*yy);
```