

数学／数式処理システム MATLAB の使い方（1）

本学の汎用システム上で使える数学／数式処理システムのうち、ここでは MATLAB の使い方を簡単に説明する。（数式処理システムには他にも Mathematica などがあるが、それはまたいずれ。）

MATLAB は関数や行列の計算、2次元、3次元のグラフ表示、数値的な微分・積分、記号的な微分・積分をはじめとして様々な機能を持つが、今回は2変数関数のグラフを描いてみるまでを目標とする。

以下では、総合情報処理センターの汎用計算機システム (dream) 上で、X 端末から X-window 環境 (CDE 環境) で使用することを念頭におく。他にもセンターのパソコン上などでも（機能が限定された版が）使えるが、それらについては別途に案内する。

1 MATLAB の起動と終了

コマンド入力ウィンドウ (dtterm) から、以下のコマンドを入力する。

```
dream% matlab
```

すると画面ウィンドウが一瞬出て消えたあと、以下のようなメッセージとプロンプトが表示される。

```
< M A T L A B (R) >
(c) Copyright 1984-97 The MathWorks, Inc.
All Rights Reserved
Version 5.1.0.421
May 25 1997

To get started, type one of these commands: helpwin, helpdesk, or demo.
For information on all of the MathWorks products, type tour.

>>
```

プロンプトの “>>” が出たところで入力待ち状態になる。

- ここで “quit” と打てば MATLAB が終了する。
- “demo” と打てば様々なデモ機能が実行できる。適当なところをマウスでクリックして実行してみるとよい。
- “help”（または “helpwin”）とすれば、オンラインのヘルプが表示される。help の後に項目名を書くと、その項目の説明がテキストとして画面に表示される。あらかじめ

```
>> more on
```

と打っておけば、1画面分表示されたところで画面下に “--more--” と表示されて停止する。表示を続けるにはリターンを、そこでやめるには “q” を打てばよい。

helpwin はウィンドウ・メニュー方式で説明を見たい項目を選択する。

- 残念ながら、上のヘルプを含めて全てのメッセージは英語で表示される。

なお、MATLAB のマニュアルは実習室やセンター事務室に一揃いが用意してある。ただしこちらも英語である。日本語の参考書のいくつか出版されている。平賀の手元にあるのは：

- 小林一行：「MATLAB ハンドブック」、秀和システム (1998)、本体 2800 円

2 関数グラフの表示

次の2変数関数：

$$z = f(x, y) = x(x^2 - 1) - y^2 \quad (-2 \leq x \leq 2, -2 \leq y \leq 2)$$

のグラフを表示してみよう（まずこれがどういうグラフになるかを先に考えてみること）。

```
>> [x,y] = meshgrid([-2:0.2:2]);
>> z = x .* (x.^2 -1) - y.^2;
>> mesh(x,y,z);
```

と打てば、新たなウィンドウができてそこに3次元グラフが表示される（図1：ただし画面上ではカラー表示される）。次に：

```
>> view([0 0]);
>> view([90 0]);
```

と打てば、それぞれ x 軸方向、 y 軸方向で真横から見た図が表示される（図2,3）。

図1は単にメッシュグリッド、つまり針金状に示した輪郭が記されるだけだが、`mesh(x,y,z)` の代わりに `surf(x,y,z)` とすれば、面に陰影をつけて塗りつぶした表示がなされる（図4）。

```
>> contour(x,y,z,20);
```

とすれば、20段階に分けた等高線が表示される。この図はよく見ると縦横の縮尺が合っていない。そこで

```
>> axis('square');
```

とすれば、縦横同じ縮尺の正方形に画面が変換される（図5）。またこの状態で

```
>> [u,v] = gradient(z,0.2);
>> hold on;
>> quiver(x,y,u,v);
>> hold off;
```

とすれば、勾配ベクトルの矢線表示が重ね書きされる（図6）。また

```
>> contour3(x,y,z,20);
```

とすれば立体的な等高線図が表示される（図7）。以上が基本である。

各々の手順を説明しよう。

1. 上の x, y, z などの変数名である。これは C などの変数と同様、英数字列ならなんでもよい。大文字と小文字は区別されるので注意すること。変数への値の代入は、 C と同様、`'='` で行う（例：`x = 10;`）。ただし、 C の変数と違って、ベクトル（=配列）、行列（=2次元配列）なども自由に使用できる。

2. `[x,y] = meshgrid([-2:0.2:2]);`

これは最小値が -2 、最大値が 2 で 0.2 刻みの2次元配列を作り、その一方の軸に x 、もう一方の軸に y という名前を付けろ、という意味である。したがって 21×21 の行列が作られる。

2つの軸の幅や刻みの大きさを変えるには、

```
[x,y] = meshgrid([-2:0.2:2], [-1:0.1:1]);
```

などとすればよい（ y のほうが $-1 \leq y \leq 1$ で 0.1 刻みになる）。

3. `z = x .* (x.^2 -1) - y.^2;`

関数値 z を求める計算式を書く。数式は C と同様の書き方で $+$, $-$ が加算・減算の記号だが、乗算、除算、べき乗については、`.*`, `./`, `.^` のように、先頭に `'` (ピリオド) がつく点に注意 (一覧は `help arith`, `help slash` を参照)。

また関数記号として、`sin(x)`, `cos(x)`, `tan(x)`, `exp(x)`($= e^x$), `log(x)`, `sqrt(x)`($= \sqrt{x}$) など使える (一覧は `help elfun` を参照)。

例: $z = \sin(x^2 + y^2)$ を求めるなら、

```
z = sin(x.^2 + y.^2);
```

とする (図 8)。

4. `mesh`, `surf` などは上記のように、計算した x,y,z のグラフを表示するコマンドである。

また `contour`, `contour3` は 2 次元、3 次元の等高線表示コマンドで、第 4 引数として等高線を何本引くかを指定する。

5. `axis('square')` は表示を正方形にするためのおまじない、`hold on/off` は重ね書きをする (on) しない (off) の指定である。

`gradient`, `quiver` については省略する。

3 ファイルからの読み込み

同じことを何度もやったり、一部だけ変えて実行する場合、上のようにいちいちすべてを打つのは面倒だし、打ち間違いをすると悲惨である。そこで打ち込む内容をあらかじめファイルに入れておけば、それを読み出して実行することができる。そうすれば、個別の変数値などを指定するだけで、同じプログラムが簡単に再実行できる。

まず `xxx.m` というファイルに次の内容を打ち込む (`xxx` のところはなんでもよい)。

```
[x,y] = meshgrid([-2:0.2:2]);  
z = x .* (x.^2 -1) - y.^2;  
mesh(x,y,z);
```

すると MATLAB の中で

```
>> xxx
```

(つまりファイル名の `“.m”` を取り除いたもの) を打てば、その内容が実行される。

少し変更するには、ファイルを別のファイルにコピーして、必要箇所を修正すればよい。あるいは変更箇所を変数で指定しておいて、実行時に必要な変数値を設定するなどしてもよい。

4 グラフの出力・印刷

作成したグラフをファイルに書き出したり、紙に印刷したりもできる。

現在表示中のグラフをファイルに書き出すには、

```
>> print -dps xxx.ps  
ないしは  
>> print -deps xxx.eps
```

とする。xxx はファイル名で、なんでもよい。ps と eps の違いについては省略する。

このファイルをプリンタに送れば、グラフが表示される。例えば `open` に出力するには:

```
dream% lp -d open -Tpostscript xxx.ps
```

とすればよい (`“-Tpostscript”` は不用のはず)。

图 1

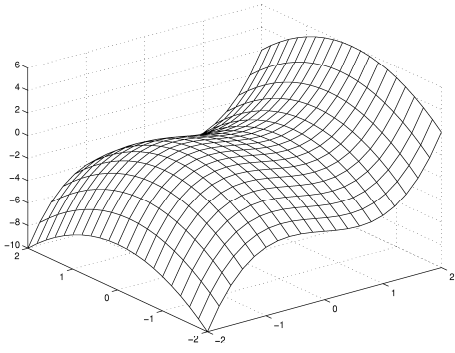


图 2

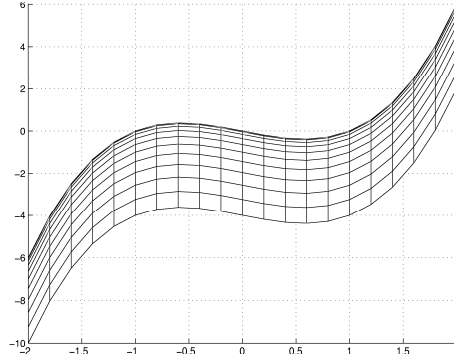


图 3

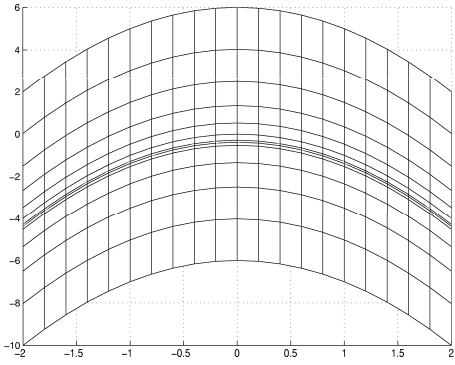


图 4

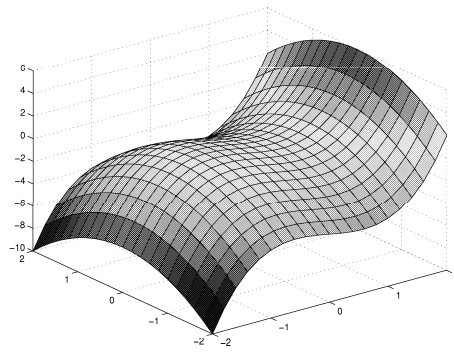


图 5

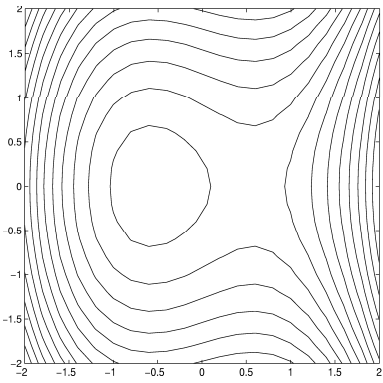


图 6

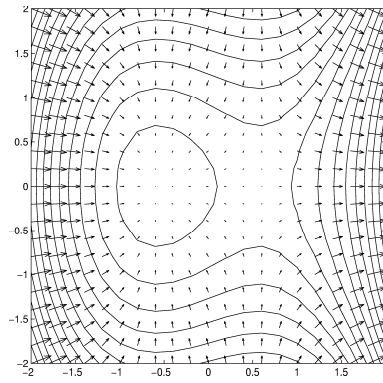


图 7

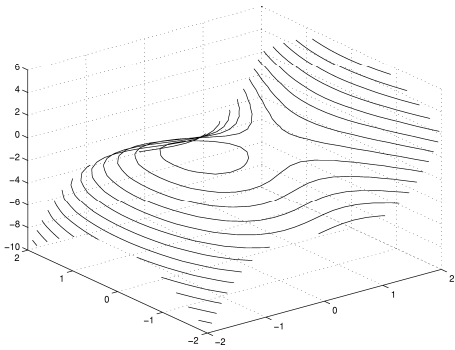


图 8

