

マテマティカによる線形代数の演習

金子 晃

計算機に数値ではなく数式の計算をさせるソフトのことを数式処理ソフトと云います。市販のソフトの中にも老舗の REDUCE を始め、Mathematica, Maple など、パソコンで動くものがかなり出回っていますが、いずれも高価なのが欠点です。今日は市販ソフトの代表格で、情報処理センターがサイトライセンスを持っている Mathematica の使い方を学習します。

1 Mathematica の立ち上げ方

情報処理センターの6階演習室、情報の5階計算機室のいずれでも Mathematica を使うことができます。

Mathematica の主な特徴：

- 1) 関数の引数は () でなく [] で括る。
- 2) 関数名は頭文字が大文字になっている。
- 3) 一指令ごとに結果が出力されるが、指令の末尾に ; を付けることでそれが抑制される。最初はやたらに ; を使わず、途中結果を見ながら進む方が安全である。
- 4) 必ずしも自動的に一番簡単な形にまで持って行ってくれず、望ましい答に到達するには Simplify 指令が必要となることがある。
- 5) 計算結果は必ずしも人間がやったときの一番自然な表現とは一致しない。また計算の途中経過を見るのも非常に難しい。うまく利用するには頭も使う必要がある。
- 6) 数値は有理数でどこまでも正確に計算しようとするので、近似数値計算をするには N 指令を付ける必要がある。最後の三つはもう少し一般に数式処理ソフトにかなり共通した特徴です。このあたりをわきまえて、上手に使う必要があります。自分で計算した結果の検算に使うには非常に強力です。場合によっては、Mathematica の出した結果を見て証明方法を思い付くこともあります。いずれにせよ、Mathematica に習熟することと数学を勉強することとは、必ずしも同じではありません。 手計算の練習も怠らないようにしないと、試験で真っ青になります。

まずは、難しいことを考えずに使ってみましょう。ターミナルウィンドウの一つで

```
math
```

と入力して Mathematica を起動し、以下の命令を順に実行してみよ。(指令を打ち終って CR キー (ENTER キー) を押せば、結果かエラーメッセージのどちらかが出力されます。括弧が整合的でなく、式として完結していないような場合は、改行しても引続き入力待ちの状態となります。) ちなみに、X の描画ができる環境で

```
mathematica
```

を実行すると、ノートブックスタイルのものが起動し、Windows95 上の Mathematica を立ち上げたときとそっくりの環境になります。この場合は、シフトキーを押しながら CR キーを叩くと、計算式が評価され (計算が実行され) ます。

終るときは

```
Exit
```

と打ち込みます。グラフィカルな mathematica の場合はマウスで終了メニューが選べます。

2 線型代数

以下、In[n]:= が Mathematica の入力プロンプトを表し、その後続く部分がユーザーの入力すべきデータです。Out[n]= に続く部分がその入力に対する Mathematica の答です。n は通し番号で、%n によりその内容を後から引用できます。

行列の定義

- 1) In[1]:= A={{1,2,3},{4,5,6},{7,8,9}} (行列 A をリストとして定義。これが行列の内部構造.)
Out[1]= {{1, 2, 3}, {4, 5, 6}, {7, 8, 9}}
- 2) In[2]:= MatrixForm[A] (A を行列の形で出力させる函数)
Out[2]= MatrixForm=

1	2	3
4	5	6
7	8	9

3) In[3]:= A[[1]] (A の第 1 行)

Out[3]= {1, 2, 3}

4) In[4]:= A[[1,2]] (A の (1,2) 成分)

Out[4]= 2

今後は Mathematica の入力プロンプトを省略する. 出力結果も省略することが多い.

5) B=Table[f[i,j],{i,3},{j,4}] (行列 B の成分を $f[i,j]$, $1 \leq i \leq 3, 1 \leq j \leq 4$ とする)

例: van der Monde 行列:

6) B=Table[a^(i*j),{i,4},{j,4}]

7) Array[C,{m,n}] (不定文字 $C[i,j]$ を成分とする $m \times n$ 型行列を生成; 実行するときは m, n には数値を入れること)

8) G=DiagonalMatrix[{a,b,c}] (a, b, c を対角成分とする対角型行列を生成)

9) F=IdentityMatrix[3] (3 次の単位行列を生成)

練習問題 次の行列を定義せよ:

$$10) A = \begin{pmatrix} 1 & -1 & 1 \\ 1 & 0 & 3 \\ 1 & -1 & 2 \end{pmatrix} \quad 11) B = \begin{pmatrix} 1 & 2 & -1 \\ 1 & 3 & 2 \\ 1 & 2 & 0 \end{pmatrix} \quad 12) C = \begin{pmatrix} -21 & -6 & 23 \\ 10 & 3 & -11 \\ -4 & -1 & 4 \end{pmatrix}$$

☞ Mathematica では C は常微分方程式を解く関数 DSolve が使う任意定数, D は微分演算の記号として予約されているので使えない. 他の記号で代用してください.

行列の演算

13) A+B (A と B の行列和) A-B (A と B の行列差)

A.B (A と B の行列積) c*A (A の c によるスカラー積)

☞ 二つの行列 A, B に対して $A*B$ と書いても, Mathematica は成分毎の積を計算してしまうので十分注意しましょう.

練習問題 10)–12) で定義した行列 A, B, C に対し, 次の計算をせよ:

14) $A+B$ 15) $A-C$ 16) AB 17) ABC

行列の関数

18) Det[A] (正方行列 A の行列式, 戻り値はスカラー)

Inverse[A] (正則な正方行列 A の逆行列, 戻り値は A と同サイズの行列)

Transpose[A] (行列 A の転置, 戻り値は A と同サイズの行列)

Eigenvalues[A] (正方行列 A の固有値のリストを与える)

Eigenvalues[N[A]] (同上の数値解)

Eigenvectors[A] (正方行列 A の固有ベクトルのリストを与える)

Eigenvectors[N[A]] (同上の数値解)

MatrixPower[A,n] (正方行列 A の n 乗, 戻り値は A と同サイズの行列)

MatrixExp[A] (正方行列 A の指数関数, 戻り値は A と同サイズの行列)

固有多項式には特別な関数はない. 例えば $\text{Det}[x*\text{IdentityMatrix}[n]-A]$ で計算できる. ここに n は A のサイズ. また A の第 i 列ベクトルは $\text{Transpose}[A][[i]]$ で計算できる.

練習問題 10) の行列 A について, これらの出力をすべて確かめよ. ただし冪乗の計算では $n=10$ とせよ.

練習問題 次の行列式を計算せよ. 答はなるべく因数分解した形で与えよ. (因数分解は Factor[] という関数を使う.)

$$19) \begin{vmatrix} 1 & a & a^3 \\ 1 & b & b^3 \\ 1 & c & c^3 \end{vmatrix} \quad 20) \begin{vmatrix} b+c & a & a \\ b & c+a & b \\ c & c & a+b \end{vmatrix} \quad 21) \begin{vmatrix} 0 & a^2 & b^2 & 1 \\ a^2 & 0 & c^2 & 1 \\ b^2 & c^2 & 0 & 1 \\ 1 & 1 & 1 & 0 \end{vmatrix}$$

☞ 上の練習問題で三番目の行列を C の代わりに c に代入していると, ここでも c にそれが入っているので, とんでもない結果になる. そういう場合は Clear[c] を実行して変数 c を初期化してからやればよい.

ベクトルの演算 ベクトルは 1 行だけの行列と解釈されますが, 積演算の右側に来る場合は転置行列をとらなくても列ベクトルとみなされ, 行列のベクトルへの作用やベクトルの内積も演算 A.B でカバーされます.

- 22) $V=\{2,3,-1\}$ ベクトル $V=(2,3,-1)^T$ の定義
 23) $A \cdot V$ 行列 A とベクトル V の積
 24) $W=\{1,0,1\}$; $V \cdot W$ ベクトル V と W の内積

連立一次方程式の解

- 25) `Solve[{a11*x+a12*y, a21*x+a22*y]=={b1,b2}, {x,y}]` あるいは
`Solve[{a11*x+a12*y==b1, a21*x+a22*y==b2}, {x,y}]`
 26) `Reduce[{a11*x+a12*y, a21*x+a22*y]=={b1,b2}, {x,y}]` を使うと “分母が 0 にならないとき” という条件を付けてくれる。

練習問題 次の連立一次方程式を解け：

$$27) \begin{cases} x + y + z = 0, \\ 2x + 3y + 5z = -1, \\ 5x - 4y + 6z = 1 \end{cases} \quad 28) \begin{cases} -2x + y - 2z - u = 3, \\ 3x - y + 3z + 2u = -7, \\ 5x + 3y + 2z + 3u = -10, \\ -4x + y + z + u = 4 \end{cases} \quad 29) \begin{cases} 2x + ay - z = 3a, \\ x - y + 3z = a, \\ (a+2)x + y + 2az = -1 \end{cases}$$

`Solve` は一般の方程式を対象としたソルバーである。不定・不能の場合も含めて連立一次方程式を解くには、専用の函数を使うほうが良い：

30) $A=\{\{1,2,-1\},\{2,4,-2\}\}$; $b=\{3,6\}$; $x_0=\text{LinearSolve}[A,b]$ $\begin{cases} x + 2y - z = 3, \\ 2x + 4y - 2z = 6 \end{cases}$ を解く

これで解の一つ (特殊解) が得られる。不能の場合は特殊解が空リスト `{}` となる。一般解はこれに $\text{Ker } A$ を加えて置けばよい：

- 31) `x1=NullSpace[A]` $\text{Ker } A$ の基底
 32) `Length[x1]` $\text{Ker } A$ の次元 (基底リストの長さとして求めている)
 33) `x=x0;For[i=1,i<=Length[x1],i++,x=x+x1[[i]]*t[i]];` `x t[1],...,t[n]` を一次結合の係数として一般解を表現
 34) `x[[1]]+2*x[[2]]-x[[3]]` (第一の方程式の検算)
 35) `Simplify[%]` (これをやらないと結果が必ずしも 3 にならない)
 \% は直前の出力結果を表す。

練習問題 次の連立一次方程式を解け。

$$36) \begin{cases} 2x - 8y + 3z = 1, \\ -6x + 24y - 9z = -3 \end{cases} \quad 37) \begin{cases} x + 2y + 3z = 0, \\ 4x + 5y + 6z = 0, \\ 7x + 8y + 9z = 0 \end{cases} \quad 38) \begin{cases} x + 2y + az = 1, \\ 2x + 4y + bz = 2, \\ 3x + 6y + cz = 3 \end{cases}$$

行列の階数 階数を求める函数は用意されていないので、自分でプログラムを書かねばならない。ここでは `NullSpace` を利用して次元の不変性に関する基本公式から導く：

- 39) `Dimension[A]` 行列 A のサイズ。{行数, 列数} というリストで返される。
 40) `Rank[A]:=Dimension[A][[2]]-Length[NullSpace[A]]` 行列 A の階数を与える函数の定義。

練習問題 次の行列の階数を求める。

$$41) \begin{pmatrix} b & a & a \\ a & b & a \\ a & a & b \end{pmatrix} \quad 42) \begin{pmatrix} 6 & 2 & 3 & -2 & -7 \\ 5 & 3 & 7 & -6 & -4 \\ 8 & 0 & -5 & 6 & 13 \\ 4 & -2 & -7 & -5 & -7 \end{pmatrix} \quad 43) \begin{pmatrix} 1 & 1 & 1 & 1 \\ 1 & 2 & 3 & 4 \\ a & b & c & d \end{pmatrix}$$

線型変換 一般の平面写像の描画と同様である。

行列 $\begin{pmatrix} a & b \\ c & d \end{pmatrix}$ が定める平面の線型変換の挙動を図示する：

- 44) `LinearTrans[a_,b_]:=Module[{original,mapped}, (* 局所変数の宣言 *)
 original=Graphics[Table[{Line[{{0,t},{1,t}],
 Line[{{t,0},{t,1}]}],{t,0,1,0.1}];
 mapped=Graphics[Table[{
 Line[{{a[[1]]*0+b[[1]]*t,a[[2]]*0+b[[2]]*t},
 {a[[1]]*1+b[[1]]*t,a[[2]]*1+b[[2]]*t}],
 Line[{{a[[1]]*t+b[[1]]*0,a[[2]]*t+b[[2]]*0},
 {a[[1]]*t+b[[1]]*1,a[[2]]*t+b[[2]]*1}}],
 {t,0,1,0.1}];`

```
Show[{original,mapped},AspectRatio->Automatic]
]
```

☞ (* *) で囲まれた部分は Mathematica プログラミングにおける注釈を表します。

45) LinearTrans[{1,2},{3,4}] 上で定義した関数(プロシージャ)の使用法(行列の列ベクトルを与える) 裏返るかどうかも見るには、もとの図形が向きを持ったものの方がよい。いつものキャラクタにご登場願おう:

```
46) LinearTrans[a_,b_]:=Module[{original,mapped,transf},
  original={0.24,0.11},{0.18,0.11},{0.13,0.13},{0.08,0.17},{0.05,0.23},
    {0.03,0.34},{0.04,0.40},{0.05,0.44},{0.08,0.50},{0.11,0.53},{0.12,0.54},
    {0.08,0.56},{0.05,0.61},{0.05,0.66},{0.07,0.69},{0.09,0.70},{0.14,0.71},
    {0.18,0.69},{0.20,0.64},{0.20,0.60},{0.18,0.56},{0.21,0.57},{0.27,0.57},
    {0.25,0.61},{0.25,0.64},{0.27,0.69},{0.31,0.71},{0.35,0.71},{0.36,0.70},
    {0.38,0.69},{0.39,0.67},{0.40,0.65},{0.40,0.60},{0.38,0.56},{0.35,0.55},
    {0.36,0.53},{0.40,0.48},{0.42,0.42},{0.43,0.34},{0.43,0.31},{0.43,0.28},
    {0.42,0.25},{0.40,0.20},{0.34,0.13},{0.28,0.11},{0.24,0.11},{0.19,0.13},
    {0.16,0.15},{0.16,0.20},{0.18,0.24},{0.22,0.26},{0.28,0.24},{0.30,0.21},
    {0.31,0.19},{0.32,0.17},{0.33,0.14},{0.34,0.11},{0.37,0.10},{0.41,0.10},
    {0.46,0.12},{0.51,0.19}};
  ListPlot[original,PlotJoined->True,AspectRatio->Automatic,
    AxesOrigin->{0,0},Axes->Automatic];
  transf[p_]:=a[[1]]*p[[1]]+b[[1]]*p[[2]],a[[2]]*p[[1]]+b[[2]]*p[[2]];
  mapped=Map[transf,original]; (* 写像 transf で元データを変換 *)
  ListPlot[mapped,PlotJoined->True,AspectRatio->Automatic,
    AxesOrigin->{0,0},Axes->Automatic];
]
```

LinearTrans[{1,2},{0,1}] (* 使用例 (行列の列ベクトルを引数として与える) *)

47) 線型変換 $\begin{pmatrix} 1 & a \\ 1 & 1 \end{pmatrix}$ により平面がどのように写されるかを a の値により分類して示せ。

2次曲線と2次曲面の描画 線形代数で出て来る2次曲線は、いわゆる陰函数のグラフとなっています。一般に、陰函数のグラフを自分でプログラミングして描くときは、一つの変数について部分的に解き、一変数函数のグラフを重ね描きしてゆくか、適当なパラメータ表示に直して描くのがもっと早いですが、Mathematica で陰函数のグラフを描くには、専用のパッケージを利用するのが便利です。

48) 陰函数 $x^2 + y^2 = 1$ のグラフを描く。
Needs["Graphics`ImplicitPlot`"];
ImplicitPlot[x^2+y^2==1,{x,-2,2}];

ImplicitPlot を囲んでいるのはバッククォーテーションである(日本式キーボードでは のシフト)。

次の2次曲線の概形を描け。

- 49) $x^2 - 2xy + y^2 - x + 2y - 1 = 0$ 50) $2x^2 + 2xy + y^2 - 3x - 2y - 1 = 0$
51) $x^2 - xy - 3x + y + 4 = 0$ 52) $x^2 + xy - 2y^2 + x + y - 2 = 0$
53) $x^2 + 2xy - 3y^2 + 2x - 6y + 5 = 0$ 54) $x^2 - 8xy + 7y^2 + 6x - 6y + 9 = 0$

曲面のグラフは、最も手軽には2変数函数のグラフを繋いで描く。放物面の場合はこの方法で一発で描ける。

55) $z = x^2 + y^2$ のグラフ

```
Plot3D[x^2+y^2,{x,-2,2},{y,-2,2}]
```

デフォルトの設定では必ずしも満足する図が得られるとは限らない。微調整の例を示す。

```
g=Plot3D[x^2+y^2,{x,-2,2},{y,-2,2}]
Show[g,
  Boxed->False, Mesh->False,
  Lighting->True, LightSources->{{1,1,1},GrayLevel[1]},
  ClipFill->{GrayLevel[0], GrayLevel[1]},
  PlotPoints->{20,20},
  PlotRange->{0,4},
  BoxRatios->{1,1.33,1.2}
]
```

次の2次曲面の概形を描け。

- 56) $x^2 + y^2 + 5z^2 + 4xz + 2yz - 5z = 0$ 57) $x^2 + y^2 + 3z^2 + 4xy + 4xz + 2yz + 6y + 4z - 3 = 0$
58) $2xy + x + y + z = 0$ 59) $xy + yz + zx = 1$
60) $3x^2 + 4y^2 + 5z^2 + 4xy - 4yz + 10x - 12y + 13 = 0$

参考文献

- [1] 小池慎一 “Mathematica 数式処理入門”, 技術評論社
[2] S. Wolfram “Mathematica”, アジソン・ウェスレイ
[3] T. Wickham-Jones “Mathematica Graphics”, Springer, 1994.