

第 4 章

MATLAB の基礎 3 — 行列演算

4.1 行列の定義と演算

行列の定義もベクトル同様リストを用いる。要素はカンマで区切り，一行ごとにセミコロンを入れ，一つのリストとして定義する。例えば実正方行列 $A, B \in \mathbb{R}^{3 \times 3}$ を

$$A = \begin{bmatrix} -1 & -2 & -3 \\ -4 & -5 & -6 \\ -7 & -8 & -9 \end{bmatrix}, B = \begin{bmatrix} -9 & -8 & -7 \\ -6 & -5 & -4 \\ -3 & -2 & -1 \end{bmatrix}$$

とし，これをそれぞれ `mat_a`, `mat_b` という変数に代入するには

```
>> mat_a = [-1, -2, -3; -4, -5, -6; -7, -8, -9]
```

```
mat_a =
```

```

-1    -2    -3
-4    -5    -6
-7    -8    -9
```

```
>> mat_b = [-9, -8, -7; -6, -5, -4; -3, -2, -1]
```

```
mat_b =
```

```

-9    -8    -7
-6    -5    -4
-3    -2    -1
```

となる。

行列の演算は，ベクトル同様の演算子を使用できる。例えば $A + B$, $A - B$, AB は次のように計算する。

```
mat_a + mat_b
```

```
mat_a - mat_b
```

```
mat_a * mat_b
```

ベクトルと行列の掛け算も同様に

```
mat_a * vec_v
mat_b * vec_w
```

とすればよい。

■単位行列 単位行列 I は `eye(行数, 列数)` という関数で求めることができる。 n 次正方行列の単位行列 I_n の場合は `eye(n)` でも良い。

```
>> eye(2, 2)
```

```
ans =
```

```
    1    0
    0    1
```

```
>> eye(2)
```

```
ans =
```

```
    1    0
    0    1
```

```
>> eye(3, 3)
```

```
ans =
```

```
    1    0    0
    0    1    0
    0    0    1
```

■零行列 零行列 O は `zeros(行数, 列数)` という関数で求めることができる。 n 次正方行列の場合は `zeros(n)` でも良い。

```
>> zeros(2, 2)
```

```
ans =
```

```
    0    0
    0    0
```

```
>> zeros(2)
```

```
ans =
```

```
    0    0
```

```
0 0
```

```
>> zeros(3)
```

```
ans =
```

```
0 0 0
0 0 0
0 0 0
```

■その他の行列 全ての成分が1となる行列は `ones(行数, 列数)` という関数で求めることができる。 n 次正方形行列の場合は `zeros(n)` でも良い。

```
>> ones(3, 1)
```

```
ans =
```

```
1
1
1
```

```
>> ones(3)
```

```
ans =
```

```
1 1 1
1 1 1
1 1 1
```

成分がゼロでない適当な行列を作りたい時には、乱数行列 `rand(行数, 列数)`, `rand(正方形行列の次数)` で求めることができる。この場合、実行するごとに異なる値が得られる。

```
>> rand(3,1), rand(3, 1)
```

```
ans =
```

```
0.2785
0.5469
0.9575
```

```
ans =
```

```
0.9572
```

```
0.4854
0.8003
```

```
>> rand(3)
```

```
ans =
```

```
0.1419    0.7922    0.0357
0.4218    0.9595    0.8491
0.9157    0.6557    0.9340
```

問題 4.1

$A, B \in \mathbb{R}^{4 \times 4}$ が

$$A = \begin{bmatrix} 3 & 1 & 2 & 9 \\ -2 & 3 & -2 & 7 \\ 0 & -4 & 5 & -2 \\ -2 & 3 & -4 & 0 \end{bmatrix}, B = \begin{bmatrix} 4 & 2 & -7 & 3 \\ 2 & 8 & -2 & -9 \\ 0 & 4 & -2 & 2 \\ -1 & 3 & 2 & 0 \end{bmatrix}$$

であるとき、次の計算を行え。

1. $3A + 2B$
2. A^2
3. $-3A^2 + 2B$

4.2 行列要素の取り出し

行列 A の特定の行列要素 a_{ij} を一つだけ取り出す時には、 $A(i, j)$ と指定する。

```
>> A = [1, 2, 3; 4, 5, 6; 7, 8, 9]
```

```
A =
```

```
1     2     3
4     5     6
7     8     9
```

```
>> A(1, 3)
```

```
ans =
```

```
3
```

```
>> A(3, 1)
```

```
ans =
```

7

行列の特定の行成分, 列成分のみを取り出す時には, $A(\text{行番号}, :)$, $A(:, \text{列番号})$ のようにコロン (:) 演算子を指定する。

```
>> A(1, :)
```

```
ans =
```

```
1     2     3
```

```
>> A(:, 3)
```

```
ans =
```

```
3
```

```
6
```

```
9
```

特定の範囲のブロック行列 (小行列) を一ずつずらしながら取り出すには, インデックス部分を開始番号:終了番号と指定する。例えば以下の例は

$$A = \begin{bmatrix} * & 2 & 3 \\ * & 5 & 6 \\ * & * & * \end{bmatrix}$$

の部分を取り出している。

```
>> A(1:2, 2:3)
```

```
ans =
```

```
2     3
```

```
5     6
```

行列の行や列を逆順にする時には

```
>> A(3:-1:1, :)
```

```
ans =
```

```
7     8     9
```

```
4     5     6
```

```
1     2     3
```

```
>> A(:, 3:-1:1)
```

```
ans =
```

```

3     2     1
6     5     4
9     8     7
```

とする。このようにコロンの演算子を使ったインデックスを使うと、かなり細かい指定が可能となる。

ベクトルを並べて行列を作ることもしできる。例えば

$$\mathbf{v}_1 = \begin{bmatrix} 1 \\ 2 \\ 3 \end{bmatrix}, \mathbf{v}_2 = \begin{bmatrix} 3 \\ 4 \\ 5 \end{bmatrix}, \mathbf{v}_3 = \begin{bmatrix} 6 \\ 7 \\ 8 \end{bmatrix} \Rightarrow P = [\mathbf{v}_1 \ \mathbf{v}_2 \ \mathbf{v}_3]$$

は次のように実行できる。

```
>> v1 = [1; 2; 3], v2 = [4; 5; 6], v3 = [7; 8; 9]
```

```
v1 =
```

```

1
2
3
```

```
v2 =
```

```

4
5
6
```

```
v3 =
```

```

7
8
9
```

```
>> P = [v1, v2, v3]
```

```
P =
```

```

1     4     7
2     5     8
```


行の定数倍と加算 $R(i, j; c)$ i 行目に j 行目の c 倍 ($c \neq 0$) を加える :

$$R(i, j; c) = \begin{matrix} & & & & & j \text{ 列目} \\ & & & & & \vdots \\ & & & & & \vdots \\ & & & & & \vdots \\ i \text{ 行目} & \cdots & \cdots & 1 & \cdots & c \\ & & & & \ddots & \vdots \\ & & & & & 1 \\ & & & & & & \ddots \\ & & & & & & & 1 \end{matrix}$$

定数 c が非ゼロである時, 3つの基本変形行列 $P(i, j)$, $Q(i; c)$, $R(i, j; c)$ は正則行列となる。

定理 4.1 (基本変形行列の逆行列)

$c \neq 0$ の時,

$$P(i, j)^{-1} = P(j, i) \quad (4.1)$$

$$Q(i; c)^{-1} = Q\left(i; \frac{1}{c}\right) \quad (4.2)$$

$$R(i, j; c)^{-1} = R(i, j; -c) \quad (4.3)$$

となる。

行の基本変形を行うことで, 全ての要素が零にならない行数が決まる。これが行列の階数 (**rank**) であり, 次のように書く。

$$\text{rank}\left(\begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix}\right) = 1, \quad \text{rank}\left(\begin{bmatrix} -5 & 0 & 5 \\ -5 & 0 & 5 \\ 1 & 2 & 0 \end{bmatrix}\right) = 2$$

なお, MATLAB では階数を求める **rank** 関数があり, 次のように用いる。

```
>> rank([1, 0; 0, 0])
```

```
ans =
```

```
1
```

```
>> rank([-5, 0, 5; -5, 0, 5; 1, 2, 0])
```

```
ans =
```

```
2
```

問題 4.2

1. 行列 A が

$$A = \begin{bmatrix} 3 & 2 & 1 \\ 2 & 1 & 0 \\ 1 & 0 & -1 \end{bmatrix}$$

と与えられている時, 次の計算を行え。

- (a) $P(2,3)A$
- (b) $AP(2,3)$
- (c) $R(1,2;-1)A$

2. 次の行列 B の階数を求めるために必要となる基本変形行列を答えよ。また B の階数も答えよ。

$$B = \begin{bmatrix} 3 & 1 \\ -9 & -3 \end{bmatrix}$$

3. 次の行列の階数を求めよ。

- (a) $A_1 = \begin{bmatrix} 3 & 4 \\ 4 & 1 \end{bmatrix}$
- (b) $A_2 = \begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 3 \\ 6 & 7 & 3 \end{bmatrix}$

4.4 逆行列の計算

正方行列が正則である時、その逆行列を求めることができる。正則でない時にはその旨、エラーの表示が出る。これを確認してみよう。 A が正則行列、 B が非正則行列となる

$$A = \begin{bmatrix} 1 & 2 \\ 2 & 1 \end{bmatrix}, B = \begin{bmatrix} -2 & 6 \\ 1 & -3 \end{bmatrix}$$

という設定で、MATLAB で逆行列 A^{-1} (存在する)、 B^{-1} (存在しない) を求める。`inv` 関数でも、逆数⁽⁻¹⁾ の演算でも同じ結果になる。

```
>> A = [1, 2; 2, 1]
```

```
A =
```

```
1    2
2    1
```

```
>> B = [-2, 6; 1, -3]
```

```
B =
```

```
-2    6
1    -3
```

```
>> inv(A)
```

```
ans =
```

```
-0.3333    0.6667
0.6667   -0.3333
```

```
>> inv(B)
```

警告：行列が特異なため、正確に処理できません。

```
ans =
```

```
    Inf    Inf
    Inf    Inf
```

```
>> A^(-1)
```

```
ans =
```

```
-0.3333    0.6667
 0.6667   -0.3333
```

非正則行列 B に対しては、「行列が特異なため、正確に処理できません」というエラーメッセージが出ていることがわかる。なお、丸め誤差のため、行列によっては正則性の確認が正しくできないこともあることに留意すること。

■**検算** $AA^{-1} = A^{-1}A = I$ という計算結果になるかどうかを確認して検算を行えば良い。

```
>> A * inv(A)
```

```
ans =
```

```
    1    0
    0    1
```

```
>> inv(A) * A
```

```
ans =
```

```
    1    0
    0    1
```

問題 4.3

次の行列 C , D の逆行列を求めよ。また検算も行え。

$$C = \begin{bmatrix} 1 & 2 & 3 \\ 2 & 2 & 3 \\ 3 & 3 & 3 \end{bmatrix}, D = \begin{bmatrix} 1 & 1/2 & 1/3 & 1/4 \\ 1/2 & 1/3 & 1/4 & 1/5 \\ 1/3 & 1/4 & 1/5 & 1/6 \\ 1/4 & 1/5 & 1/6 & 1/7 \end{bmatrix}$$

4.5 連立一次方程式の解の計算

係数行列として n 次正方行列 A , 定数ベクトルとして n 次元のベクトル \mathbf{b} が与えられている時

$$A\mathbf{x} = \mathbf{b}$$

を満足する n 次元の未知ベクトル \mathbf{x} を求めてみよう。例として

$$\begin{bmatrix} 1 & 2 \\ 2 & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} 4 \\ 5 \end{bmatrix}$$

とする。

■逆行列を用いる方法 既に見てきたように、係数行列 A の逆行列 A^{-1} の計算が可能であれば、

$$\mathbf{x} = A^{-1}\mathbf{b}$$

として計算すればよい。

```
>> A = [1, 2; 2, 1]
```

```
A =
```

```
1    2
2    1
```

```
>> b = [4; 5]
```

```
b =
```

```
4
5
```

```
>> x = inv(A) * b
```

```
x =
```

```
2.0000
1.0000
```

よって $\mathbf{x} = [2 \ 1]^T$ であることが分かった。

■直接解く方法 演算子 \ (バックスラッシュ, 日本語環境では円マーク ¥ に相当) を使用することで、逆行列を求めることなく、直接解を得ることもできる。

```
>> A = [1, 2; 2, 1]
```

```
A =
```

```
    1    2  
    2    1
```

```
>> b = [4; 5]
```

```
b =
```

```
    4  
    5
```

```
>> x = A \ b
```

```
x =
```

```
    2  
    1
```

■**検算** 求められた解が確かに連立一次方程式を満足しているかどうかは、 Ax の計算を行い、 \mathbf{b} と等しくなっているかを確認すればよい。

```
>> A * x
```

```
ans =
```

```
    4  
    5
```

問題 4.4

次の連立一次方程式の解を求めよ。また、検算も行え。

$$\begin{bmatrix} 3 & -1 & 0 \\ -1 & 3 & -1 \\ 0 & -1 & 3 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} -8 \\ 2 \\ 5 \end{bmatrix}$$

4.6 行列式

行列式 (determinant) が定義でき、一番小さい 2 次行列の行列式を土台として、次のようにして計算できる。

定義 4.2 (正方行列の行列式)

$$\begin{vmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{vmatrix} = a_{11}a_{22} - a_{12}a_{21}$$

これを土台とし、第 (i, j) 余因子 Δ_{ij} (i 行目と j 列目を除いた $n-1$ 次正方行列) の記号を用いて、 n 次正方行列の行列式は次のように再帰的に計算する。

$$\begin{aligned} & \begin{vmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & & \vdots \\ a_{n1} & a_{n2} & \cdots & a_{nn} \end{vmatrix} \\ &= \sum_{j=1}^n (-1)^{i+j} a_{ij} |\Delta_{ij}| \quad (j = 1, 2, \dots, n) \\ &= \sum_{j=1}^n (-1)^{i+j} a_{ij} |\Delta_{ij}| \quad (i = 1, 2, \dots, n) \end{aligned} \tag{4.4}$$

例えば 3 次正方行列の場合、例えば $i = 1$ とした時には次のように計算できる。

$$\begin{aligned} & \begin{vmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{vmatrix} = \sum_{j=1}^3 (-1)^{1+j} a_{1j} |\Delta_{1j}| \\ &= a_{11} \begin{vmatrix} a_{22} & a_{23} \\ a_{32} & a_{33} \end{vmatrix} - a_{12} \begin{vmatrix} a_{21} & a_{23} \\ a_{31} & a_{33} \end{vmatrix} + a_{13} \begin{vmatrix} a_{21} & a_{22} \\ a_{31} & a_{32} \end{vmatrix} \\ &= a_{11}a_{22}a_{33} + a_{12}a_{23}a_{31} + a_{13}a_{21}a_{32} - a_{13}a_{22}a_{31} - a_{12}a_{21}a_{33} - a_{11}a_{23}a_{32} \end{aligned}$$

実際には行列積における次の性質を用いて、基本変形を経由して計算量を減らしておいてから行列式の計算を行う。

定理 4.2 (行列積の行列式)

任意の n 次正方行列 A, B に対しては

$$|AB| = |A||B|$$

である。

MATLAB で行列式を計算するためには、`det` 関数を用いる。例えば

$$A = \begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix}, B = \begin{bmatrix} 5 & 6 \\ 7 & 8 \end{bmatrix}$$

とする時、 $|A|$ と $|B|$ は次のように計算できる。

```
>> A = [1, 2; 3, 4]
```

```
A =
```

```
1    2
3    4
```

```
>> B = [5, 6; 7, 8]
```

```
B =
```

```
5    6
7    8
```

```
>> det(A), det(B)
```

```
ans =
```

```
    -2
```

```
ans =
```

```
 -2.0000
```

では $|A||B| = |AB|$ かどうかを確認してみよう。

```
>> det(A) * det(B)
```

```
ans =
```

```
  4.0000
```

```
>> A * B
```

```
ans =
```

```
    19    22
```

```
    43    50
```

```
>> det(A * B)
```

```
ans =
```

```
  4.0000
```

よって、 $|AB| = |A||B|$ が確認できた。

4.7 行列のノルム

行列のノルムはベクトルのノルムをベースにして定義される p ノルムと、行列を長いベクトルとみなして求める Frobenius ノルムがある。

定義 4.3 (行列の p ノルム)

任意の $A \in \mathbb{C}^{n \times n}$ に対し,

$$\|A\|_p = \max_{\mathbf{x} \neq 0} \frac{\|A\mathbf{x}\|_p}{\|\mathbf{x}\|_p}$$

と定義される $\|\cdot\|_p$ を行列の p ノルムと呼ぶ。

これによって、行列の 1 ノルム、ユークリッドノルム、無限大ノルムも同様に定義されることになる。

$$\begin{aligned} 1 \text{ ノルム } \|A\|_1 &= \max_{\mathbf{x} \neq 0} \frac{\|A\mathbf{x}\|_1}{\|\mathbf{x}\|_1} = \max_j \sum_{i=1}^n |a_{ij}| \\ \text{ユークリッドノルム } \|A\|_2 &= \max_{\mathbf{x} \neq 0} \frac{\|A\mathbf{x}\|_2}{\|\mathbf{x}\|_2} = \sqrt{\max_i \lambda_i(\overline{A^T}A)} \quad (\text{ここで } \lambda_i(A) \text{ は } A \text{ の } i \text{ 番目の固有値}) \\ \text{無限大ノルム } \|A\|_\infty &= \max_{\mathbf{x} \neq 0} \frac{\|A\mathbf{x}\|_\infty}{\|\mathbf{x}\|_\infty} = \max_i \sum_{j=1}^n |a_{ij}| \end{aligned}$$

なお、行列の p ノルムとベクトルの p ノルムとの間には、

$$\|A\mathbf{x}\|_p \leq \|A\|_p \cdot \|\mathbf{x}\|_p$$

という関係がある。よって、行列 A, B に対しても

$$\|AB\|_p \leq \|A\|_p \cdot \|B\|_p$$

が成り立つ。

Frobenius ノルム $\|\cdot\|_F$ は p ノルムとは異なり、行列成分を並べてベクトルとして見立てたユークリッドノルムである。 $A \in \mathbb{C}^{m \times n}$ に対して、 $\|A\|_F$ は次のように計算する。

$$\|A\|_F = \sqrt{\sum_{i=1}^m \sum_{j=1}^n |a_{ij}|^2} \quad (4.5)$$

例えば $A \in \mathbb{R}^{2 \times 2}$ が

$$A = \begin{bmatrix} -1 & -3 \\ -5 & -3 \end{bmatrix}$$

であるとき、

$$\begin{aligned} \|A\|_1 &= \max(|-1| + |-5|, |-3| + |-3|) = \max(6, 6) = 6 \\ \|A\|_\infty &= \max(|-1| + |-3|, |-5| + |-3|) = \max(4, 8) = 8 \end{aligned}$$

となる。また $\lambda_1(AA^T) = 22 - 2\sqrt{85}$, $\lambda_2(AA^T) = 22 + 2\sqrt{85}$ であるから

$$\|A\|_2 = \sqrt{\max(|\lambda_1(AA^T)|, |\lambda_2(AA^T)|)} = \sqrt{22 + 2\sqrt{85}}$$

となる。また、フロベニウスノルム $\|A\|_F$ は

$$\|A\|_F = \sqrt{|-1|^2 + |-3|^2 + |-5|^2 + |-3|^2} = \sqrt{44}$$

である。

MATLAB で行列ノルムを求めるには、ベクトルと同様に `norm` 関数を使って次のように計算できる。それぞれ、ユークリッドノルム、1 ノルム、無限大ノルム、フロベニウスノルムは

```
norm(mat_a)
norm(mat_a, 1)
norm(mat_a, 'inf')
norm(mat_a, 'fro')
```

として計算する。上記の例を計算してみると

```
>> A = [-1, -3; -5, -3]
```

```
A =
```

```
   -1   -3
   -5   -3
```

```
>> norm(A), norm(A, 1), norm(A, 'inf'), norm(A, 'fro')
```

```
ans =
```

```
   6.3592
```

```
ans =
```

```
   6
```

```
ans =
```

```
   8
```

```
ans =
```

```
   6.6332
```

となり、正しくノルムの計算ができていることが分かる。

問題 4.5

1. 行列 A が

$$A = \begin{bmatrix} -1 & -2 \\ -2 & -1 \end{bmatrix}$$

で与えられる時、 $\|A\|_1, \|A\|_\infty, \|A\|_F$ をそれぞれ求めよ。

2. 行列 B が

$$B = \begin{bmatrix} i & -i \\ 2+i & 1-3i \end{bmatrix}$$

で与えられる時, $\|B\|_1, \|B\|_\infty, \|B\|_F$ をそれぞれ求めよ。

4.8 行列の誤差

行列の誤差についてもベクトル同様に, 成分ごとの誤差とベクトル誤差を考えることができる。

例えば真の行列 $A \in \mathbb{R}^{n \times n}$ を

$$A = \begin{bmatrix} \sqrt{2} & \pi \\ \sin \pi/3 & -\sqrt{3} \end{bmatrix}$$

とする。これを有効桁数を 10 進 10 桁に丸めた近似行列 \tilde{A} は

$$\tilde{A} = \begin{bmatrix} 1.414213562 & 3.141592654 \\ 0.8660254038 & -1.732050808 \end{bmatrix}$$

となる。

成分ごとの絶対誤差, 相対誤差, およびノルム絶対誤差 $aE_2(\tilde{A})$, ノルム相対誤差 $rE_2(\tilde{A})$ は

```
>> abs(A - short_A)
```

```
ans =
```

```
1.0e-09 *
```

```
0.3731    0.4102
```

```
0.0156    0.4311
```

```
>> abs((A - short_A) ./ short_A)
```

```
ans =
```

```
1.0e-09 *
```

```
0.2638    0.1306
```

```
0.0180    0.2489
```

```
>> norm(A - short_A)
```

```
ans =
```

```
6.6277e-10
```

```
>> norm(A - short_A) / norm(A)
```

ans =

1.7930e-10

として計算できる。

問題 4.6

次の行列 H について、次の問いに答えよ。

$$H = \begin{bmatrix} 1 & 1/2 & 1/3 \\ 1/2 & 1/3 & 1/4 \\ 1/3 & 1/4 & 1/5 \end{bmatrix}$$

1. H^{-1} を求め、これを $\widetilde{H^{-1}}$ とせよ。
2. H^{-1} の真の値は

$$H^{-1} = \begin{bmatrix} 9 & -36 & 30 \\ -36 & 192 & -180 \\ 30 & -180 & 180 \end{bmatrix}$$

である。これを用いて、上記で求めた $\widetilde{H^{-1}}$ の成分ごとの相対誤差と $rE_2(\widetilde{H^{-1}})$ をそれぞれ求めよ。