

## 1. 到達目標

- マトリクスとベクトルの掛け算を計算するプログラムを作成できる。
- マトリクスとマトリクスの掛け算を計算するプログラムを作成できる。

## 2. ベクトル・マトリクス演算の必要性

- 線形代数学は、構造解析やデータ分析の分野では欠かせないものである。
- 2年生で学ぶ不静定力学Ⅱでは、マトリクス計算ができることが必須になる。すなわち、建築構造設計の分野では、マトリクス演算の知識が必ず必要になる。
- マトリクス計算（行列計算）は、コンピュータを利用した計算に非常にマッチしている。

## 3. 線形代数の基礎（復習）

### (1) 行列の表記

$m$  個の行と  $n$  個の列をもつ行列は「 $m \times n$  行列」と呼ばれる。 $m \times n$  行列は下記のように書ける。

行列の要素は、2重添字記号で表される。第1の添字は行の番号を、第2の添字は列の番号を示す。たとえば、 $a_{56}$  は、5行目6列目の要素を示している。

複数行・複数列で構成される行列はマトリクスとも呼ばれる。1行だけからなる行列は行ベクトル、1列だけからなる行列は列ベクトルと呼ばれる。

### (2) 行列の相等

二つの行列  $A$ ,  $B$  が等しいのは、両者の型  $m \times n$  が同じで、しかも対応するすべての要素が等しい場合である。

### (3) 行列の和

行列の加法は同型の行列の間にだけ定義される。

行列  $A$  + 行列  $B$  の和は、両者の対応する要素を加えることによって得られる。

$$A = \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix} \quad B = \begin{bmatrix} b_{11} & b_{12} \\ b_{21} & b_{22} \end{bmatrix}$$

$$A + B = \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix} + \begin{bmatrix} b_{11} & b_{12} \\ b_{21} & b_{22} \end{bmatrix} = \begin{bmatrix} a_{11} + b_{11} & a_{12} + b_{12} \\ a_{21} + b_{21} & a_{22} + b_{22} \end{bmatrix}$$

たとえば、

## (4) 行列の積

( $A$  の列数 ) = ( $B$  の行数) のときに限って行列の乗法が定義される。

$$A \times B = C$$

$$A = \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix} \quad B = \begin{bmatrix} b_{11} & b_{12} & b_{13} \\ b_{21} & b_{22} & b_{23} \end{bmatrix}$$

$$\begin{aligned} A \times B &= \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix} \times \begin{bmatrix} b_{11} & b_{12} & b_{13} \\ b_{21} & b_{22} & b_{23} \end{bmatrix} \\ &= \begin{bmatrix} a_{11} \times b_{11} + a_{12} \times b_{21} & a_{11} \times b_{12} + a_{12} \times b_{22} & a_{11} \times b_{13} + a_{12} \times b_{23} \\ a_{21} \times b_{11} + a_{22} \times b_{21} & a_{21} \times b_{12} + a_{22} \times b_{22} & a_{21} \times b_{13} + a_{22} \times b_{23} \end{bmatrix} = C \end{aligned}$$

行列  $A$  の型は  $m \times u$ , 行列  $B$  の型は  $u \times p$  のとき, 行列  $C$  の型は  $m \times p$  になる。

$C$  行列の要素は次のようになる。

行列  $C$  の各要素を求めるには,  $A$  の  $j$  行目の各要素に  $B$  の  $k$  列目の対応する要素を掛け, これらの  $n$  個の積を加え合わせればよい。

このことを簡単に「行を列に掛ける」という。

たとえば,

$$\begin{aligned} & \begin{bmatrix} B= \\ \end{bmatrix} \quad \begin{bmatrix} \\ \end{bmatrix} \\ C = A \times B &= \begin{bmatrix} \times \\ \end{bmatrix} = \begin{bmatrix} 1 \times 1 + 2 \times 4 & 1 \times 2 + 2 \times 5 & 1 \times 3 + 2 \times 6 \\ 3 \times 1 + 4 \times 4 & 3 \times 2 + 4 \times 5 & 3 \times 3 + 4 \times 6 \end{bmatrix} = \begin{bmatrix} \\ \end{bmatrix} \end{aligned}$$

## 4. 例題1 (マトリクスとベクトルの掛け算)

(1)下記のマトリクス  $A$  とベクトル  $B$  の積を求めるプログラムを作成せよ。

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

The screenshot shows an Excel spreadsheet with the following data:

	1	2	3	4	5	6	7	8
1		1	2	5		17		
2		3	4	6		39		
3		A		B		C		

The Visual Basic editor shows the following code for '例題1()':

```

Sub 例題1()
    Dim a(2, 2), b(2), c(2) 'マトリクス・ベクトルの宣言
    For i = 1 To 2
        For j = 1 To 2
            a(i, j) = Cells(i, 1 + j) 'マトリクスaを読み込む
        Next j
    Next i

    For i = 1 To 2
        b(i) = Cells(i, 4) 'ベクトルbを読み込む
    Next i

    For i = 1 To 2
        s = 0
        For j = 1 To 2
            s = s + a(i, j) * b(j) 'a×b
        Next j
        c(i) = s
    Next i

    For i = 1 To 2
        Cells(i, 6) = c(i) '結果を出力
    Next i
End Sub

```

例題1(1)の実行順序

マトリクス $a$ , ベクトル $b$ のサイズを宣言	⑦ $i=2$	⑭ $i=1$	⑳ $j=1$	㉑ $j=1$	㉒ $i=1$
マトリクス $a$ を読み込む	⑧ $j=1$	⑮ $b(1)=Cells(1,4)$	㉓ $s=0+a(1,1)*b(1)$	㉔ $s=0+a(2,1)*b(1)$	㉕ $Cells(1,6)=c(1)$
① $i=1$	⑨ $a(2,1)=Cells(2,2)$	⑯ $i=2$	㉖ $j=2$	㉗ $j=2$	㉘ $i=2$

② $j=1$	⑩ $j=2$	⑰ $b(2)=Cells(2,4)$	⑳ $s=s+a(1,2)*b(2)$	㉓ $s=s+a(2,2)*b(2)$	㉖ $Cells(2,6)=c(2)$
③ $a(1,1)=Cells(1,2)$	⑪ $a(2,2)=Cells(2,3)$	⑱ $i=3 (i>2)$	㉔ $j=3 (j>2)$	㉗ $j=3 (j>2)$	㉙ $i=3 (i>2)$
④ $j=2$	⑫ $j=3 (j>2)$	㉕ $a \times b$	㉘ $c(1)=s$	㉚ $c(2)=s$	—終了—
⑤ $a(1,2)=Cells(1,3)$	⑬ $i=3 (i>2)$	㉖ $i=1$	㉙ $i=2$	㉛ $i=3 (i>2)$	
⑥ $j=3 (j>2)$	ベクトル $b$ を読み込む	㉚ $s=0$	㉜ $s=0$	結果を出力	

(2)下記のマトリクス A とベクトル B の積を求めるプログラムを作成せよ。

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

The screenshot shows an Excel spreadsheet with the following data:

	1	2	3	4	5	6	7	8
1		1	2	3	10		68	
2		4	5	6	11		167	
3		7	8	9	12		266	

The VBA code window shows the following code for '例題2':

```

Sub 例題2()
    Dim a(3, 3), b(3), c(3) 'マトリクス・ベクトルの宣言
    For i = 1 To 3
        For j = 1 To 3
            a(i, j) = Cells(i, 1 + j) 'マトリクスaを読み込む
        Next j
    Next i

    For i = 1 To 3
        b(i) = Cells(i, 5) 'ベクトルbを読み込む
    Next i

    For i = 1 To 3
        s = 0
        For j = 1 To 3
            s = s + a(i, j) * b(j) 'a×b
        Next j
        c(i) = s
    Next i

    For i = 1 To 3
        Cells(i, 7) = c(i) '結果を出力
    Next i
End Sub

```

## 5. 演習問題 1

$n$  行  $n$  列のマトリクス  $A$  と  $n$  行のベクトル  $B$  の積を求めるプログラムを作成し、それを使って上記の例題 1 の(1)と(2)を計算せよ。

$$B = \begin{pmatrix} b_1 \\ \vdots \\ b_n \end{pmatrix}$$

マトリクスのサイズ

	1	2	3	4	5	6	7	8
1	2	[ 1	2 ]	[ 5 ]		[ 17 ]		
2		[ 3	4 ]	[ 6 ]		[ 39 ]		
3			A	B		C		

```

Sub 例題3()
End Sub

```

**6. 演習問題1の解答**

$n$ 行 $n$ 列のマトリクス $A$ と $n$ 行のベクトル $B$ の積を求めるプログラムを作成し、それを使って上記の例題1の(1)と(2)を計算せよ。

$$B = \begin{pmatrix} b_1 \\ \dots \\ b_n \end{pmatrix}$$



	1	2	3	4	5	6	7	8
1	2	[ 1	2 ]	[ 5 ]		[ 17		
2		[ 3	4 ]	[ 6 ]		[ 39		
3		A		B		C		

  

Microsoft Visual Basic - 例題3.xlsm - [Module1 (コード)]

ファイル(E) 編集(E) 表示(V) 挿入(I) 書式(O) デバッグ(D) 実行(R) ツール(T) アドイン(A)  
 ウィンドウ(W) ヘルプ(H)

プロジェクト - VBAPrX (General) 例題3

```

Sub 例題3()
    'マトリクス・ベクトルの最大サイズを10までと宣言
    Dim a(10, 10), b(10), c(10)

    n = Cells(1, 1) 'マトリクスサイズの読み込み

    For i = 1 To n
        For j = 1 To n
            a(i, j) = Cells(i, 1 + j) 'マトリクスaを読み込む
        Next j
    Next i

    For i = 1 To n
        b(i) = Cells(i, 1 + n + 1) 'ベクトルbを読み込む
    Next i

    For i = 1 To n
        s = 0
        For j = 1 To n
            s = s + a(i, j) * b(j) 'a×b
        Next j
        c(i) = s
    Next i

    For i = 1 To n
        Cells(i, 1 + n + 1 + 2) = c(i) '結果を出力
    Next i

End Sub
    
```

プロパティ - Module1X  
 Module1 Module  
 全体 項目別  
 オブジェクト Module1