

Octave の使い方

1 起動方法

- 1:まず、スタート すべてのプログラム Cygwin Bash Shell を起動させる。
 - 2:octave と入力すれば Octave が起動する。
- Octave 用のフォルダは、C:\cygwin\home\ユーザー名。このフォルダの中にスクリプトファイルを置いておけば実行できる。

2 各種コマンド

2.1 基本的な演算

Octave は基本的に行列の数値計算を扱うために開発されたので、変数や関数、演算等も行列を扱うことを前提に定義されている (実数、複素数はスカラー扱い)。

- ・行列 (ベクトル) の定義 : $M = [1, 2, 3; 4, 5, 6]$ とすれば、

$$M = \begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \end{bmatrix}$$

という行列ができる。また、行列 M の転置行列は M' と入力すればよい。

- ・成分表示と代入 : $M(1, 2)$ とすれば、行列 M の (1, 2) 成分を指す。また、 $M(1, 2) = 0$ と入力すれば、(1, 2) 成分に 0 を代入できる。因みに数字だけでなく文字も代入できるが、日本語は避けたほうが良い。文字の代入方法は $S = ' abcdef'$ のように、代入したい文字列を ' で囲めばよい。

また、 $M(2, :)$ とすれば、 M の 2 行目、 $M(:, 3)$ とすれば、 M の 3 列目が表示される。

- ・まとめて成分入力 :

$V = [a : b : c]$ とすれば、 a から増分 b で c までの値が V の各成分に入力される。

例 :

$$V = [0 : 1 : 4]$$

$$\rightarrow V = [0, 1, 2, 3, 4]$$

演算記号一覧

演算記号	例	概要
+	$A + B$	行列の足し算
-	$A - B$	行列の引き算
*	$A * B$	行列の掛け算 (特に、 B が定数なら行列の定数倍)
^	$A ^ b$	行列のべき乗 ($b = -1$ とすると、 A の逆行列になる)
/	A / B	左の行列と / の右の行列の逆行列の掛け算
\	$A \backslash B$	右の行列と \ の左の行列の逆行列の掛け算 ¹
.*	$C = A .* B$	$C_{ij} = A_{ij} \times B_{ij}$ 、行列の各成分毎の掛け算
./	$A ./ B$	左の行列と ./ の右の行列の逆行列の各成分毎の掛け算
.\	$A .\ B$	右の行列と .\ の左の行列の逆行列の各成分毎の掛け算

2.2 文字や数字の出力

計算結果を分かりやすく表示するためには、'sprintf', 'disp' という出力用関数を使えばよい。

例：

```
>> r = 1.2, a = 1, S='sample'
r = 1.2000
a = 1
S = sample
>> text='This is a %s, r=%f and a=%d'
text = This is a %s, r=%f and a=%d;
>> TEXT=sprintf(text,S,r,a)
TEXT = This is a sample, r=1.200000 and a=1
>> disp(TEXT)
This is a sample, r=1.200000 and a=1
```

このように表示される。'text' と 'TEXT' はどんな名前でも構わないが、既に Octave に定義してある関数名 (sin 等) は避けること。毎回 $a = 1$ 等が表示されるのが煩わしい場合は ; を各命令文の後に付けば、

```
>> r=1.2;a=1;S='sample';
>> text='This is a %s, r=%f and a=%d'
>> TEXT=(text,S,r,a);
>> disp(TEXT)
This is a sample, r=1.200000 and a=1
```

と ; を付けていない箇所のみ表示される。%s などの役割は下の表を参照。

コマンド	概要
%f	小数を表示するのに使う
%d	整数を表示するのに使う
%s	文字列を表示するのに使う
	改行したいときに使う。

¹\ は ¥ でも同じ。

2.3 関数

関数について述べる前に、Octave で予め定義されている特別な数について説明する。

特別な数

コマンド	概要
pi	円周率 π
e	自然対数 e
i	虚数単位 i (または、 $\sqrt{-1}$)

Remark 1. これらの数は予め Octave 内で定義されているが、例えば、 $pi = 3$ と入力することで、 pi の値が上書き (この場合は 3) されてしまう。特に、 i はかなり特殊な定義がされているので、上書きをしないように気をつけるべきである。

Octave に定義されている関数の表を次項に載せておく。これらの関数も簡単に書き換わってしまうので、注意すること。

関数一覧

コマンド	例	概要
<i>cos</i>	$C = \cos(A)$	$C_{ij} = \cos(A_{ij})$ 。他の三角関数 (<i>sin</i> , <i>tan</i>) も同様。
<i>loga</i>	$C = \log_a(A)$	$C_{ij} = \log_a(A_{ij})$ 。a を省略すると、底は自然対数 <i>e</i> になる。
<i>exp</i>	$C = \exp(A)$	$C_{ij} = e^{A_{ij}}$ 。
<i>abs</i>	$C = \text{abs}(A)$	$C_{ij} = A_{ij} $ 。
<i>sqrt</i>	$C = \text{sqrt}(A)$	$C_{ij} = \sqrt{A_{ij}}$ 。
<i>ones</i>	$\text{ones}(m, n)$	全ての成分が 1 の $m \times n$ 行列。(n.n) は (n) としても同じ。
<i>zeros</i>	$\text{zeros}(m, n)$	全ての成分が 0 の $m \times n$ 行列。
<i>eye</i>	$\text{eye}(n)$	n 次の単位行列。
<i>tril</i>	$\text{tril}(A)$	A から作られる下三角行列。
<i>triu</i>	$\text{triu}(A)$	A から作られる上三角行列。
(条件文)	$C = (A > 0)$	$A > 0 \Rightarrow C_{ij} = 1$ 、それ以外は $C_{ij} = 0$ を代入。 ($A > 0$) は他の条件にすることも可能。
<i>size</i>	$\text{size}(A)$	A の行数と列数を表示する。
<i>size</i>	$\text{size}(A, 1)$	A の行数を表示する。
<i>size</i>	$\text{size}(A, 2)$	A の列数を表示する。
<i>diag</i>	$\text{diag}(A)$	A の対角成分を表示する。 A が $1 \times n$ ベクトルのときは下記を参照。
<i>diag</i>	$C = \text{diag}(A)$	$C_{ii} = A_i$ 、それ以外は $C_{ij} = 0$ を代入。
<i>sum</i>	$C = \text{sum}(A)$	A がベクトルのときは $C = \sum_{i=1}^n A_i$ という値、 行列のときは、 $C_i = \sum_{j=1}^n A_{ji}$ というベクトル。 ²
<i>rand</i>	$\text{rand}(m, n)$	各成分が (0, 1) の一様分布に従う $m \times n$ 行列。
<i>randn</i>	$\text{randn}(m, n)$	各成分が期待値 0, 分散 1 の正規分布に従う $m \times n$ 行列。

²従って、 $\text{sum}(\text{sum}(A))$ は A の全成分の和になる。

関数一覧

コマンド	概要
<i>skewness</i> (<i>A</i>)	<i>A</i> の歪度を返す。
<i>kurtosis</i> (<i>A</i>)	<i>A</i> の尖度を返す。
<i>corroef</i> (<i>U</i> , <i>V</i>)	もし <i>U</i> , <i>V</i> の各行が観測値で、各列が変数ならば <i>corroef</i> (<i>U</i> , <i>V</i>) の (<i>i</i> , <i>j</i>) 番目の項は <i>U</i> の <i>i</i> 番目の変数と <i>V</i> の <i>j</i> 番目 <i>p</i> の変数との間の相関。
<i>cov</i> (<i>U</i> , <i>V</i>)	<i>cov</i> (<i>U</i> , <i>V</i>) の (<i>i</i> , <i>j</i>) 番目の項は、 <i>U</i> の <i>i</i> 番目の変数と <i>V</i> の <i>j</i> 番目の <i>p</i> 変数との間の共分散。
<i>sort</i> (<i>A</i>)	各列について要素を昇順で並び返す。
<i>max</i> (<i>A</i>)	各列ごとの最大値を返す。
<i>range</i> (<i>A</i>)	各列について最大値と最小値を返す。
<i>find</i> (<i>A</i>)	行列のゼロでない要素のインデックスのベクトルを返す。
<i>[i, j] = find</i> (<i>A</i>)	<i>find</i> は、行列のゼロでない要素の 行と列のインデックスを返す。
<i>det</i> (<i>A</i>)	行列式の値を返す。
<i>rank</i> (<i>A</i>)	行列の rank を求める。
<i>trace</i> (<i>A</i>)	行列の対角成分の和を計算する。
<i>null</i> (<i>A</i>)	<i>A</i> の零空間の直行基底を返す。
<i>eig</i> (<i>A</i>)	固有値を求める。
<i>S = svd</i> (<i>A</i>)	$A = U W V^t$ となるように <i>A</i> を特異値分解したときの対角行列 <i>W</i> の係数をベクトルとして返す。
<i>norm</i> (<i>A</i>)	<i>A</i> のノルムを計算する。
<i>cond</i> (<i>A</i>)	行列の 2 ノルム条件数を計算する。
<i>inv</i> (<i>A</i>)	逆行列を計算する。
<i>logm</i> (<i>A</i>)	行列の対数計算する。
<i>sqrtn</i> (<i>A</i>)	行列平方根を計算する。
<i>chol</i> (<i>A</i>)	正定値である対象行列 <i>A</i> を cholesky 分解する。
<i>Prod</i> (<i>x</i> , <i>dim</i>)	次元 <i>dim</i> 方向の積を計算する。
<i>Sumsq</i> (<i>x</i> , <i>dim</i>)	次元 <i>dim</i> 方向の積和を計算する。

判定式

コマンド	概要
<i>ischar</i> (<i>V</i>)	<i>V</i> が文字列ならば 1, そうでなければ 0 を返す。
<i>isreal</i> (<i>V</i>)	<i>V</i> が実数値ならば 1, そうでなければ 0 を返す。
<i>isfinite</i> (<i>V</i>)	<i>V</i> が有限値ならば 1, そうでなければ 0 を返す。
<i>isinf</i> (<i>V</i>)	<i>V</i> 内の無限大要素については 1 を、それ以外には 0 を返す。
<i>isnan</i> (<i>V</i>)	<i>V</i> 内の要素が <i>nan</i> であれば 1 を、それ以外には 0 を返す。
<i>any</i> (<i>V</i>)	<i>V</i> 内に 0 でない要素があるときに 1 を、それ以外には 0 を返す。
<i>all</i> (<i>V</i>)	<i>V</i> 内の全ての要素が 0 でないときに 1 を、それ以外には 0 を返す。

3 プログラムを組む

3.1 構文を書く

条件分岐 : if

ある条件で場合分けをして処理を行いたい場合は if 文を使う。

書き方

```
if(条件文)
  処理 1
else
  処理 2
end if
```

繰り返し文 : for

ある処理を繰り返し行いたい場合は for 文を使う。

書き方

```
for 変数 = 変数が動く範囲指定
  繰り返し処理
endfor
```

条件文の書き方

記号	意味
$a == b$	a と b は等しい
$a \neq b$	a と b は等しくない
$a \geq b$	a が b より大きい
a	a が 0 以外
条件文 1 && 条件文 2	条件 1 かつ条件 2
条件文 条件文	条件 1 または条件 2
!(条件文)	条件文でないとき

例：if 文,for 文を用いて 10 行 10 列の単位行列 A を設定する

```
>> for n = [1 : 10]
>> for m = [1 : 10]
>> if(n == m)
>> A(n, m) = 1;
>> else A(n, m) = 0;
>> end
>> end
>> end
```

3.2 関数を自分で定義する

例をいくつか挙げる。

例 0：定義する関数の名前 (受け取る変数)=関数が返す値

以下、Octave に入力

```
>> function 適当な変数の名前 = 定義する関数の名前 (受け取る変数)
適当な変数の名前 = 関数が返す値; end
```

例 1：somme(a, b) = a + b

```
>> function resultat = somme(a, b)
resultat = a + b;
end
```

例 2：me(V) = [V の平均, V の分散]

```
>> function [m, e] = me(V)
m = mean(V);
if(nargout == 2)
e = std(V); end
return
```

$$\text{例 3 : } fact(n) = \begin{cases} 1 & n=1 \\ n! & \text{それ以外} \end{cases}$$

```
>> function fn = fact(n);
if(n <= 1), fn = 1;
else fn = n * fact(n - 1);
end
```

$$\text{例 4 : } trisin(a, b, c) = \sin(a) \cdot \sin(b) \cdot \sin(c)$$

```
>> trisin = inline('sin(a) * sin(b) * sin(c)');
```

4 変数の定義と破壊

ここでは、変数の定義方法、変数の破壊の方法を紹介する。

演算記号一覧

演算記号	例	概要
input	input('n=')	文字の入力
save	save('ファイル名') A;	変数のバックアップ
clear	clear A	変数 A の破壊 (clear のみだと全ての変数の破壊)
load	load 'ファイル名'	修復
who	who	定義した変数の確認
fopen	a=fopen('file.txt','w+')	file.txt を w+ として開く。
fclose	fclose(a)	ファイルを閉じる
fprintf	fprintf(txt ファイル,%f, 行列)	行列の出力変換を txt ファイルに出力

5 ライブラリ関数

ここでは、確率密度、分布関数、逆分布関数、乱数発生、ユーティリティ関数に関するコマンドを紹介する。

5.1 確率密度

演算記号	概要
dbeta	ベータ分布
dbinom	二項分布
dchisq	カイ二乗分布
df	F 分布
dgamma	ガンマ分布
dhypg	超幾何分布
dnorm	正規分布
dt	t 分布

5.2 分布関数

演算記号	概要
pbeta	ベータ分布
pbinom	二項分布
pchisq	カイ二乗分布
pf	F分布
pgamma	ガンマ関数
phypg	超幾何分布
pks	コルモゴロフ・スミルノフ検定
pnorm	正規分布
pt	t分布

5.3 逆分布関数

演算記号	概要
qbeta	ベータ分布
qbinom	二項分布
qchisq	カイ二乗分布
qf	F分布
qgamma	ガンマ分布
qhypg	超幾何分布
qnorm	正規分布
qt	t分布

5.4 疑似乱数

演算記号	概要
rbeta	ベータ乱数
rbinom	二項乱数
rchisq	カイ二乗乱数
rexpweib	ワイブル乱数
rf	F分布
rgamma	ガンマ乱数
rgeom	幾何乱数
rhypg	超幾何乱数
rnorm	正規乱数
rpoiss	ポアソン乱数
rt	t乱数
rjbinom	負の二項分布の乱数
rjgamma	逆ガンマ乱数
rjpoiss	逆ポアソン乱数

5.5 ユーティリティ関数

演算記号	概要
bincoef	二項係数
quantile	ヒストグラムの4分位点の計算
histo	ヒストグラム表示

6 グラフとヒストグラム

ここでは、グラフに関するコマンドとグラフ・ヒストグラムの具体的な表示方法を紹介する。

演算記号	概要
clf	グラフをクリアする
hold on	重ね書きを許可する
subplot	グラフの位置の指定
title	グラフのタイトル
plot	グラフに結果を表示
xlabel	横軸の名称の表示
ylabel	縦軸の名称を表示
legend	グラフの名称を表示
hist	ヒストグラムを表示

グラフ描写の例 1 :

```
clf; hold on
subplot(3,2,1)
plot(cumsum(rand(1,100))./[1:100], 'r-')
title( 'plot' )
```

```
subplot(3,2,2)
plot(rand(1,10),rand(1,10), 'o')
title( 'plot' )
```

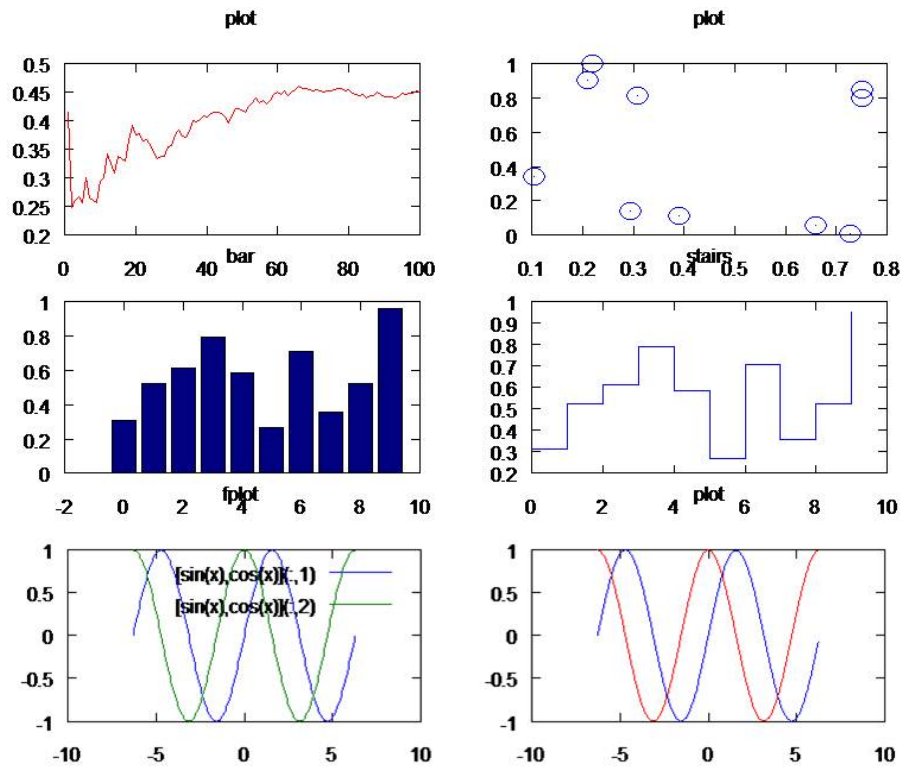
```
subplot(3,2,3)
V = rand(1,10);
bar([0:9],V)
title( 'bar' )
subplot(3,2,4)
stairs([0:9],V)
title( 'stairs' )
```

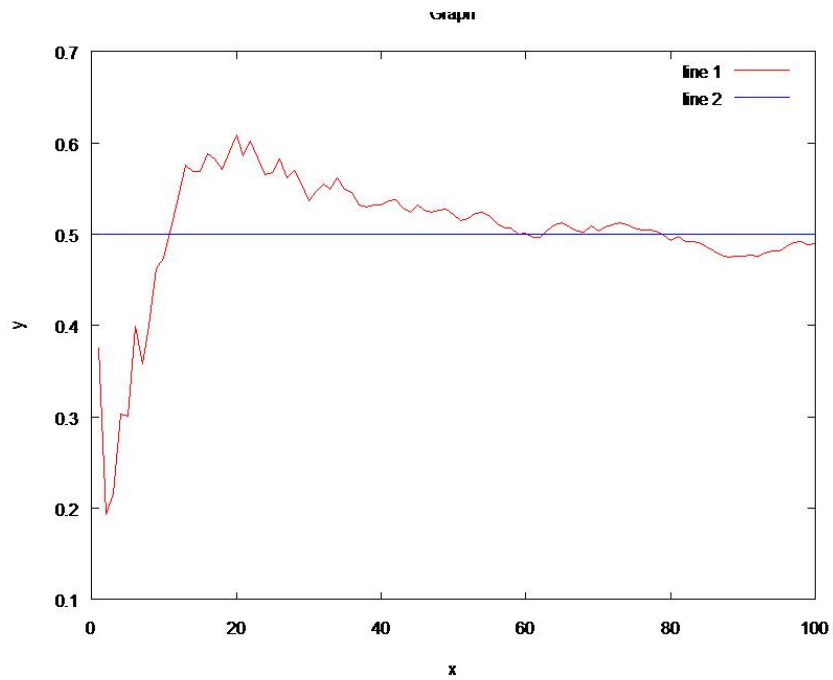
```
subplot(3,2,5)
fplot('[sin(x),cos(x)]',[-2*pi,2*pi])
title( 'fplot' )
```

```
subplot(3,2,6)
```

```
V = [-2*pi:1:2*pi];
plot(V,sin(V),'b-',V,cos(V),'r-')
title('plot')
```

このプログラムを実行すると、以下のグラフが描かれる。





例 2 :

```

clf; hold on;
plot(cumsum(rand(1,100))./[1:100]','r-')
plot(ones(1,100)/2,'b-')
title( 'Graph' )
xlabel( 'x' )
ylabel( 'y' )
legend( 'line 1','line2' )
legend( 'line 1','line 2' )

```

すると、

ヒストグラムの例 ;

```
clf ;  
V = rand(1,100);  
subplot(2,2,1)  
hist(V)  
title( 'hist(V)' )  
subplot(2,2,2)  
histo(V)  
title( 'histo(V)' )  
subplot(2,2,3)  
histo(V,10,1,1)  
title( 'histo(V,10,1,1)' )  
subplot(2,2,4)  
histo(V,10,0,1)  
title( 'histo(V,10,0,1)' )
```

すると、

