Open Platform of Transparent Analysis Tools for fNIRS

付録：MATLAB の操作

国立研究開発法人 産業技術総合研究所

POTATo では基本的な解析はグラフィカル・ユーザ・インタフェィス(GUI)を用いて行うことができます。ただ外部との I/O、データの選択と表示などの MATLA を理解することで、操作の幅が広がります。

ここでは、これらの MATLAB の操作として、以下の目次に従い説明します。

1. サンプルデータ

本書で用いるサンプルデータとして次のようなデータを考えます。

1クラスに3人の生徒がいて、全員が教科 1, 2, 3 の試験を受け、それらの点数の結果が以下のようになったとします。

クラス 1

	1	2	3
1	29	96	87
2	99	91	78
3	1	98	62

クラス 2

	1	2	3
1	95	64	25
2	97	40	81
3	85	24	43

クラス 3

	1	2	3
1	57	85	19
2	53	23	26
3	64	2	59

（各表の行は出席番号、列は教科番号）

このデータを 3x3x3 の多次元配列 score に保存し、MATLAB のサンプル操作を行います。この時、各次元は出席番号、教科番号、クラス番号とします。

2. ワークスペース

MATLAB ではデータは変数に保存します。変数はワークスペースとよばれるメモリ領域に保存されます。

ワークスペース内の変数はワークスペースブラウザで表示・変更できます。ブラウザが開いていない場合は、コマンド workspace を実行することにより開きます。

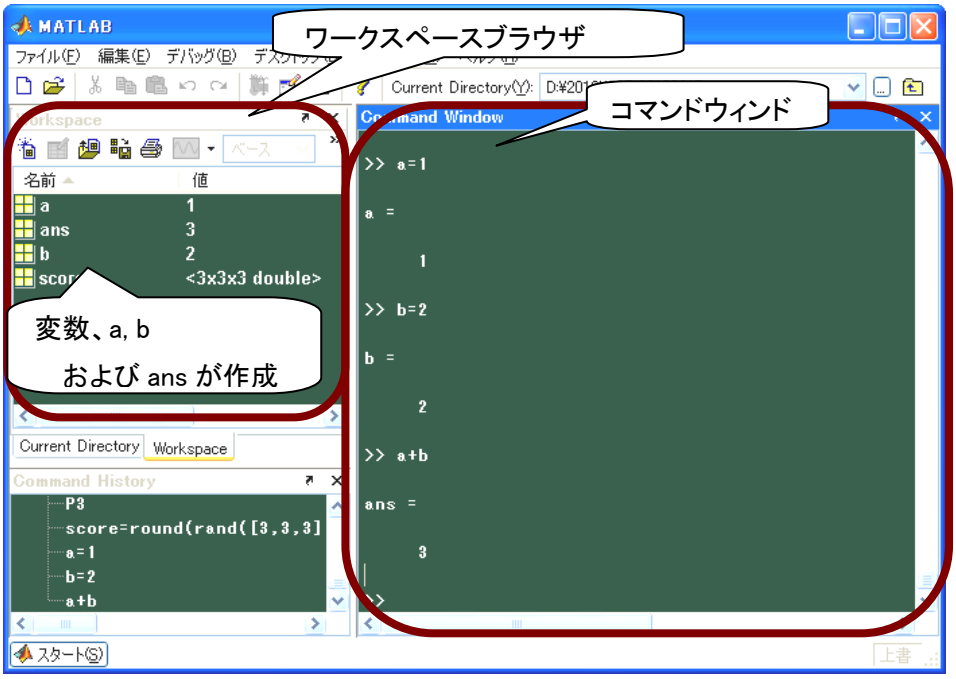
最初にコマンドウィンドウ上で、次のようにタイプします。

```
>> a=1
>> b=2
>> a+b
```

その結果、ワークスペースに a,b, という変数が保存さます。また、この変数 a, b は同じワークスペース上で利用することができます。例えば、以下のように計算で利用可能です。

```
>> a+b
```

この結果、a+b の結果が変数 ans に代入されます。



ここでサンプルデータを作成します。

行列を入力・作成する場合、全体を[と] で括ります。各要素はスペースで区切りますが、行を変える時は;により区切ります。

サンプルデータの各クラスに対応する行列の入力は以下のように行います。

```
class1=[ 29   96   87; 99   91   78;  1   98   62]
class2=[ 95   64   25; 97   40   81; 85   24   43]
class3=[ 57   85   19; 53   23   26; 64    2   59]
```

クラス 1

	1	2	3
1	29	96	87
2	99	91	78
3	1	98	62

クラス 2

	1	2	3
1	95	64	25
2	97	40	81
3	85	24	43

クラス 3

	1	2	3
1	57	85	19
2	53	23	26
3	64	2	59

この時、各クラスは 3 次元正方行列、つまり 3x3 の2次元配列です。これを 3x3x3 の3次元配列に変更します。

ヒント:

特に触れない限り、MATLAB では変数を行列ではなく配列と考えます。

そのため次元は配列の次元を指し、

行列等の次数は(各配列の)サイズという表現で使われます。

class1 は 3x3 の行列ですが、これは 3x3 の配列もしくは 3x3x1 の配列と同等に考えることが可能です。そこで、3x3x1 の 3 つの配列を 3 次元目で連結します。

配列の連結には関数 cat を用い以下のように実行します。

```
score = cat(3,class1, class2, class3)
```

以上で、サンプルデータ用の 3x3x3 の3次元配列 score がワークスペースに作成できました。

サンプルにある様に3次元配列は単に複数の行列を連結したものととらえることも可能です。ただ、例えば”3次元配列にすることで教科1のデータのみ取得する”といった別の視点にたってデータを取り出すことが簡単できます。

そこで次に、作成した多次元配列から一部のデータを取り出す方法について説明します。

3. データの選択

3.1. データの選択

このとき、出席番号 a、教科 b、クラス c の点数を取り出す際、以下のように指定します。

```
score(a,b,c);
```

例えば、a=1, b=2, c=3 の場合、クラス3の 1 行2列目のデータである 85 が取得できます。ここで最後の次元のサイズが 1 の時、その次元は削除されます。

クラス 3 の点数

	1	2	3
1	57	85	19
2	53	23	26
3	64	2	59

3.2. ベクトルによる選択

a, b c はベクトルで指定することができます。またベクトルは離散的な値も取り得ます。

```
score(1,[1 3],2);
```

この結果、クラス 2 の出席番号1の人の教科 1 と3の点数[95 25]が返ってきます。

クラス 2 の点数

	1	2	3
1	95	64	25
2	97	40	81
3	85	24	43

ベクトルは【開始値:増分:終了値】という形式で指定することができます。たとえば、1 から2つおきで 7 までと指定したい場合、"1:2:7"という表記になり、結果[1 3 5 7]が得られます。なお、増分を省略した場合は1です。

3.3. 特殊な選択

また配列の値選択時に":"を指定するとその次元の全データを意味し、"end"と指定するとその次元のサイズを示します。このことを利用して以下のようにデータを取り出せます。

```
score(:,2:end,1);    % score(1:3, 2:3, 1) と同じ
```

この結果、[96 87; 91 78; 98 62]が返ってきます。

クラス 1 の点数

	1	2	3
1	29	96	87
2	99	91	78
3	1	98	62

3.4. 配列の整形

同様に上記例で、教科 3 の全員の値を取得したい場合、以下のように指定します。

```
score(:,3 ,:);
```

この時、結果は 3x1x3 の多次元配列になります。

クラス 1

	1	2	3
1	29	96	87
2	99	91	78
3	1	98	62

クラス 2

	1	2	3
1	95	64	25
2	97	40	81
3	85	24	43

クラス 3

	1	2	3
1	57	85	19
2	53	23	26
3	64	2	59

データが見難いと感じた場合、データを整形します。

```
score3 = squeeze(score(:,3,:));
```

squeeze はサイズ1の次元を削除します。

この結果得られる score1 は 3x3 の教科 3 の点数を示す配列になります。この時、各次元は出席番号、クラス番号とします。

教科 3 の得点

	1	2	3
1	87	25	19
2	78	81	26
3	62	43	59

なお、squeeze は他の次元のサイズが1の可能性がある場合、その次元が削除されることにより予期せぬデータの次元の解釈を行う危険性があります。

このような場合、reshape 関数により作成する配列のサイズを指定します。

```
s = size(score);
score3 = reshape(score(:,3,:),s([1 3]));
```

ここで s = [3 3 3]で、score(:,1,:)で得られた 9 個のデータを3×3 (s([1 3]) = [3 3]) の配列に変換するという命令になります。

4. 簡単な演算

MATLAB の変数は配列や行列を用います。

また、MATLAB の演算子には加算、減算、乗算、除算がありますが、これらの演算は変数の型により動作が変わります。すなわち行列と行列の乗算は行列の乗算になります。

ここで、一般の概念とは異なる演算子の動作について説明します。

4.1. スカラーと行列の和算

スカラーと配列の和は各要素にスカラーを足します。

```
score3 = score3+10;
```

教科 3 の得点

	1	2	3
1	87	25	19
2	78	81	26
3	62	43	59

＋10 =

	1	2	3
1	97	35	29
2	88	91	36
3	72	53	69

4.2. 行列の除算

行列の除算は逆行列の積算として定義されています。

例えば、score3 * x = [1;1;1] となるようなベクトルを求めたいとき、x=score3^{-1}*[1;1;1]で計算でき、次のように計算します。

```
x= score3¥ones([3 1]);
```

また、x * score3 = [1 1 1]となるようなベクトルを求めたいとき、x=[1 1 1]* score3^{-1}で計算でき、次のように計算します。

```
x= ones([1 3])/score3;
```

4.3. 配列の計算

配列の各要素の積算や除算を行うには演算子の前に”.”を着けます。

```
score3./score3;
```

結果は全要素が 1 の 3x3 の行列になります。

4.4. その他の演算子

その他の演算子として、べき乗”^”や行列の転置” ‘”があります。

5. 関数を用いた演算

次に簡単な関数を用いた演算例を示します。この時、一部のデータが欠落しているものとします。例のデータで一部の点数が取得できず、取得出来なかった箇所を NaN で示すことにします。

クラス 1

	1	2	3
1	29	96	87
2	99	91	78
3	NaN	98	62

クラス 2

	1	2	3
1	95	64	25
2	97	40	81
3	85	24	43

クラス 3

	1	2	3
1	57	85	19
2	53	23	26
3	64	NaN	59

（各表の行は出席番号、列は教科番号）

ここで、各クラス、各教科の平均値を取得したいとします。

最初に、各クラス、各教科の総和を取得するため sum 関数を用います。

```
tot= sum(score,1);
```

この結果、次のような結果を得ます。

クラス 1

	1	2	3
1	NaN	285	227

クラス 2

	1	2	3
1	277	128	148

クラス 3

	1	2	3
1	174	NaN	104

sum 関数や平均値を計算する mean 関数など、多くの関数は NaN を含むデータの計算結果は NaN を返します。

そこで、NaN データを除いて総和を取得します。

```
S0=score;
S0(isnan(score))=0;
Tot = sum(S0,1);
```

ここで、score を S0 にコピーし、S0 の要素が NaN の場所を isnan(score)で計算します。

isnan(score)は以下を返します。

クラス 1

	1	2	3
1	0	0	0
2	0	0	0
3	1	0	0

クラス 2

	1	2	3
1	0	0	0
2	0	0	0
3	0	0	0

クラス 3

	1	2	3
	0	0	0
	0	0	0
	0	1	0

なお、isnan(score0)は logical 型で、0:false, 1: true を示しています。

次に、S0 の isnan(score)==ture の箇所に0を設定します。その結果、S0 は以下の様な値になります。

クラス 1

	1	2	3
1	29	96	87
2	99	91	78
3	0	98	62

クラス 2

	1	2	3
1	95	64	25
2	97	40	81
3	85	24	43

クラス 3

	1	2	3
1	57	85	19
2	53	23	26
3	64	0	59

最後に、総和を再計算することによって NaN 部分を除いた総和が計算できます。

次に、平均値を取るため、足し算をした数を求め、その数で割ります。

```
n   = sum(~isnan(score),1);
M   = Tot/ (n + n==0);
M(n==0) = NaN;
```

なお、n+n==0 は n が 0 の場所に n を足しています。このことによりゼロ割を回避しています。

また、n==0 の結果には NaN を設定しています。

以上により各クラス、各教科の平均値が取得できました。

なお、これらの計算は POTATo が提供する関数 nan_fcn を用いて以下のように計算できます。

```
M = nan_fcn( 'mean' , score, 1)
```

また、MATLAB version 7.0.1 以降の Statistics Toolbox をお持ちの場合は関数 nanmean が利用できます。

6. ファイル I/O

MATLAB で出力した変数をファイルに保存したり、読み込んだりする方法を示します。

MATLAB の変数をそのまま保存、読込するには save/load 命令を用います。

```
save scoer.mat score
load score.mat
```

また、MATLAB 外部のツールにデータを渡すため、CSV ファイルなどにデータを出力する場合、dlmwrite 関数が便利です。

追加書込み'-append'オプションにより、データを分けて保存可能です。

```
>> dlmwrite('score.csv',score(:,:,1))
>> dlmwrite('score.csv',' ','-append');
>> dlmwrite('score.csv',score(:,:,2),'-append')
>> dlmwrite('score.csv',' ','-append');
>> dlmwrite('score.csv',score(:,:,3),'-append')
```

書き込み結果は以下のようになります。

	A	B	C
1	29	96	87
2	99	91	78
3	NaN	98	62
4			
5	95	64	25
6	97	40	81
7	85	24	43
8			
9	57	85	19
10	53	23	26
11	64	NaN	59

なお、MATLAB にはこの他にも様々な書込み関数が用意されています。例えば Excel ファイルへの出力は xlswrite 関数が利用できます。