%%%%% 矩陣的處理與運算

%%% 矩陣的索引或下標

% Example 1 **（列，行）**

A = [4 10 1 6 2; 8 2 9 4 7; 7 5 7 1 5; 0 3 4 5 4; 23 13 13 0 3];

A(2,3) - A(12) A(12)**為由上往下開始數的第12個**

**一維的索引值**

% Example 2

A = [4 10 1 6 2; 8 2 9 4 7; 7 5 7 1 5; 0 3 4 5 4; 23 13 13 0 3];

B = A(4:5, 2:3)

B = A([9 14; 10 15]) **同上**

% Example 3

A = [4 10 1 6 2; 8 2 9 4 7; 7 5 7 1 5; 0 3 4 5 4; 23 13 13 0 3];

B = A(1:5, 5) % B = A(1:end, end)

B = A(:, 5) % B = A(:, end)

**End代表最後一行或最後一列**

% Example 4

A = [4 10 1 6 2; 8 2 9 4 7; 7 5 7 1 5; 0 3 4 5 4; 23 13 13 0 3];

A(2, : ) = [] % 刪除 A 矩陣的第二列

% Example 5

A = [4 10 1 6 2; 7 5 7 1 5; 0 3 4 5 4; 23 13 13 0 3];

A(:, [2 4 5]) = [] % 刪除 A 矩陣的第二、四、五直行

**裡面記得是一個向量**

% Example 6

A = [4 1; 7 7; 0 4; 23 13];

B = [A 1./A] % 1./A 是矩陣 A 每個元素的倒數

**1除以A裡每個元素的值，矩陣前要加 .**

一張含有 桌 的圖片

自動產生的描述要上下並起來，中間要加分號 ; → B = [A ; 1./A]

**Example.6**

一張含有 傢俱, 白色 的圖片

自動產生的描述% Example 7

**Example.7**

B = [0 1 2 3; 2 3 4 5; 5 6 7 8; 7 8 9 0];

d = diag(B) % 取出矩陣 B 的對角線元素

**reshape**

一張含有 文字, 時鐘, 手錶 的圖片

自動產生的描述

% Example 8

B = [0 1 2 3; 2 3 4 5; 5 6 7 8; 7 8 9 0];

C = reshape(B, 2, 8 ) % 將矩陣 B 排成 2×8 的新矩陣 C

**(矩陣, 列, 行)**

%%% 特殊用途矩陣

% Example 9

**產生矩陣？**

H=hilb(4)

% Example 10

**直接幫你產生Magic矩陣**

M = magic(5)

fprintf('直行總和：'); sum(M) % M 的每一個直行總和

fprintf('橫列總和：'); sum(M, 2) % M 的每一個橫列總和

fprintf('對角線總和：'); sum(diag(M)) % M 的對角線總和

fprintf('反對角線總和：'); sum(diag(fliplr(M))) % M 的反對角線總和

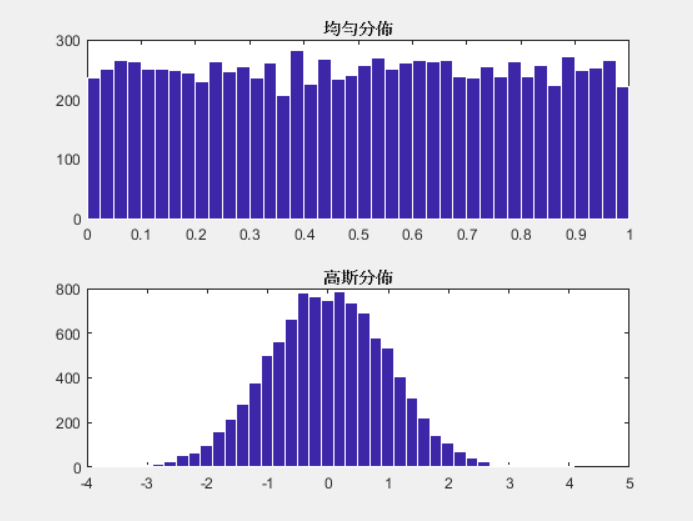
% Example 11

x1 = rand(10000, 1);

x2 = randn(10000, 1);

subplot(2,1,1); hist(x1, 40); title('均勻分佈');

subplot(2,1,2); hist(x2, 40); title('高斯分佈');

****set(findobj(gcf, 'type', 'patch'), 'EdgeColor', 'w'); % 改邊線為白色

**有點像下雨的長條圖，台北和高雄的**

**Example.11**

%%% 矩陣的數學運算

% Example 12

A = [12 34 56 20];

B = [1 3 2 4];

C = A + B

**Example.13**

% Example 13

**一張含有 文字, 時鐘 的圖片

自動產生的描述**

A = [1; 2]; → **前面的直行 = 後面的橫列**

B = [3, 4, 5];

C = A\*B ★ C = B \* A 會跑出Error

% Example 14

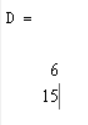
A = magic(3);

B = A^2 **次方的運算用於方陣，因為自己的行 = 自己的列**

% Example 15

A = [12; 45];

**Example.15**

**一張含有 文字 的圖片

自動產生的描述**B = [2; 3]; **= 元素與元素做運算 = [12 \* 2 ; 45 \* 3]**

一張含有 文字 的圖片

自動產生的描述 **用A \* B 會Error**

C = A.\*B % 注意「\*」前面的句點

D = A./B % 注意「/」前面的句點

E = A.^2 % 注意「^」前面的句點

% Example 16

i = sqrt(-1); % 單位虛數

z = [1+i, 2; 3, 1+2i];

w = z' % 共軛轉置（注意 z 後面的單引號）

% Example 17

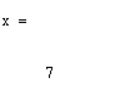
i = sqrt(-1); % 單位虛數

z = [1+i, 2; 3, 1+2i];

w = z.' % 單純轉置（注意 z 後面的句點及單引號）

% Example 18

a = [3 4]; **向量的norm（範數）？Ex： x + y 開根號**

x = norm(a, 1)

y = norm(a, 2) **→**Two Norm

z = norm(a, inf) 可以理解成一個Vector的長度

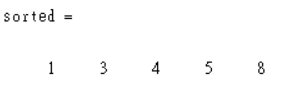
**Example.18**

% Example 19

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

norm(A, 2)

**Example.20**



% Example 20

x = [3 5 8 1 4];

[sorted, index] = sort(x) % 對矩陣 x 的元素進行排序

**Example.21**

% Example 21



x = magic(5);

[colMax, colMaxIndex] = max(x)

**橫列中最大值的Index**

% Example 22

x = magic(5);

[colMax, colMaxIndex] = max(x); **第5個橫列**

[maxValue, maxIndex] = max(colMax); **第3個值**

fprintf('Max value = x(%d, %d) = %d\n', ...

colMaxIndex(maxIndex), maxIndex, maxValue);

**Example.22**

%%% 矩陣的內部資料型態

% Example 23

clear all % 清除所有工作空間的變數

x\_double = magic(10);

一張含有 桌 的圖片

自動產生的描述x\_single = single(x\_double);

**Example.23 （沒講）**

x32 = uint32(x\_double);

x16 = uint16(x\_double);

x8 = uint8(x\_double);

whos

% Example 24

fprintf('uint8(300) ='); disp(uint8(300)); % uint8 的最大值為 255

fprintf('int8(-500) ='); disp(int8(-500)); % int8 的最小值為 -128