

MATLAB 的數學運算

last modified September 18, 2011

解決數學問題常需直接面對數學公式，基礎的訓練是以紙筆解題。本章做為以電腦解題的開端，了解 MATLAB 語言處理數學公式計算的原理，並藉此進一步熟練 MATLAB 的矩陣表示法及其計算方式與技巧，提供學生學習在面對數學公式時，如何利用電腦語言有效率的去計算公式的值。試著從對電腦語言的探索，去發覺解決問題的方法及其差異性。

本章將學到關於程式設計

變數的運用、MATLAB 計算公式的技巧與變化。

〈本章關於 MATLAB 的指令與語法〉

操作元 (operators): .*

指令: sum, mean, var, std, length, zscore, scatter, corr, corrcoef, load, axis, xlim, ylim, xlsread, importdata

語法: 資料檔的讀取: 檔案格式 MAT, TXT, XLS

1 練習

假設量測自變數 X 與應變數 Y 所得的資料為

X	6	3	5	8	7
Y	2	4	3	7	6

資料準備：依前一個練習的方式，在 MATLAB 的命令視窗裡將這些數值建立成 x, y 矩陣 (向量)，如 ¹

$$x = [6 \ 3 \ 5 \ 8 \ 7]$$

範例 1. 依序利用下列的 *MATLAB* 指令，計算變數 X 的 *sample mean*:

$$\mu_x = \frac{\sum_{k=1}^N x_k}{N}, \text{ where } N = 5$$

1. $\mu = (x(1) + x(2) + x(3) + x(4) + x(5))/5$

2. $\mu = \text{sum}(x)/5$

3. $\mu = \text{mean}(x)$

其中 *sum*, *mean* 為 MATLAB 提供的指令，使用前請先利用 *help* 瞭解其使用方式。請注意分母的 5 代表樣本數 N ，在程式設計的技巧上，會先設定一變數，譬如 $N = 5$ ，代表樣本數，之後的指令便以 N 取代固定的常數 5。當樣本數很大時，則可以利用 MATLAB 指令直接計算向量的長度，其指令如下

¹請注意程式設計中變數的命名不可輕忽，最好能賦予有意義且相關的名稱，有利程式的可讀性，在未來除錯或維護上都會很方便，無形中節省許多不必要的時間浪費。在此以小寫的 x 代表變數 X 的樣本資料，使用小寫是確知這是個向量不是矩陣，來符合以小寫字母代表向量或常數，大寫代表矩陣的慣用語法。當然習慣隨人，只要保持一致即可。

```
N=length(x)
```

範例 2. 依序利用下列的 *MATLAB* 指令, 計算變數 X 的 *sample variance*:

$$S_x^2 = \frac{\sum_{k=1}^N (x_k - \mu_x)^2}{N - 1}$$

1. 模仿範例 1 之 1 的作法, 非常繁瑣, 但請耐心的完成, 其中 μ_x 可以直接利用範例 1 的結果, 即以變數 mu 代表 μ_x 的值。

2. 模仿範例 1 之 2 的作法, 利用向量的運算並加入 *sum* 指令以減少繁雜程度, 即

$$\sum_{k=1}^N x_k^2 = \text{sum}(x .* x)$$

3. 利用矩陣運算的技巧, 將平方與累加合併計算, 進一步精簡指令, 即

$$\sum_{k=1}^N x_k^2 = x * x'$$

4. 利用 *MATLAB* 內建函式 *var*。

上述不同的過程應該得到相同的結果, 只是繁瑣與精簡之差別。程式設計當然力求精簡, 除便利可讀性外, 往往效率也會比較高。本練習的目的在讓初學者體驗 *MATLAB* 矩陣運算的效率與技巧, 當數學式本身非常繁雜時會需要這些技巧。初學者應具耐心 細心的輸入與比較才能漸得 *MATLAB* 程式設計之精髓。

範例 3. 計算每一個 x 值的 *z-score*: $z = \frac{x - \mu_x}{S_x}$

參考指令: *zscore*

範例 4. 計算 $S_{xy} = \sum_{k=1}^N (x_k - \mu_x)(y_k - \mu_y)$, 你可以用多少種方法來計算這個值呢? 哪一種方法最乾淨俐落?

2 觀察

1. 觀察常數在矩陣運算中的角色與結果。
2. 觀察累加 Σ 與矩陣的關係，並且想像上述的問題如果資料量大的時候，計算上，表達上有何困擾？這是為什麼 MATLAB 強調避免直接使用加法，而以矩陣的運算取代。
3. 變數的命名必須非常謹慎小心，不可與 MATLAB 內建的指令相同，如此將混淆該指令的功能，使之失效。譬如範例 2 的 sample variance，一般直接會取用 var 當作變數名稱，但卻與 MATLAB 內建的指令相同，會使隨後使用的 var 功能失效。

3 作業

兩變數間相關性的觀察與相關係數的計算 (注意繪圖時 x,y 軸的 scale)

1. 寫下上述範例 4 的各種計算方式，每一種方式都盡量用一條指令完成。
2. 根據網頁上提供的資料 (<http://web.ntpu.edu.tw/~ccw/statmath/book.htm>):
資料 1,
 - 先畫出 x, y 的散佈圖並以目視約略判斷 x, y 的相關性。
 - 計算變數 x, y 的相關係數 (the coefficient of correlation):

$$r = \frac{S_{xy}}{\sqrt{S_{xx}S_{yy}}}$$

其中 S_{xx} 及 S_{yy} 分別是變數 x 與 y 的變異，與之前練習所計算的變異數關係為 $S_{xx} = (N - 1)S_x^2$, $S_{yy} = (N - 1)S_y^2$ 。觀察這個 r 值與散佈圖呈現的樣子是否吻合？

- MATLAB 也提供了計算相關係數的指令，例如 corr、corrcoef，請自行了解其使用方式，最後並驗證自己的答案。

- 利用指令 `text` 將 r 值寫在散佈圖上 (如圖 1), 並寫下計算 r 值的所有指令。
 - MATLAB 也提供簡便的散佈圖指令; `scatter`, 讀者不妨參考線上手冊的說明, 自行找出使用的方式。
3. 同 2, 但使用資料 2。觀察這個 r 值與散佈圖呈現的樣子是否吻合?
 4. 同 2, 但使用資料 3。觀察這個 r 值與散佈圖呈現的樣子是否吻合?

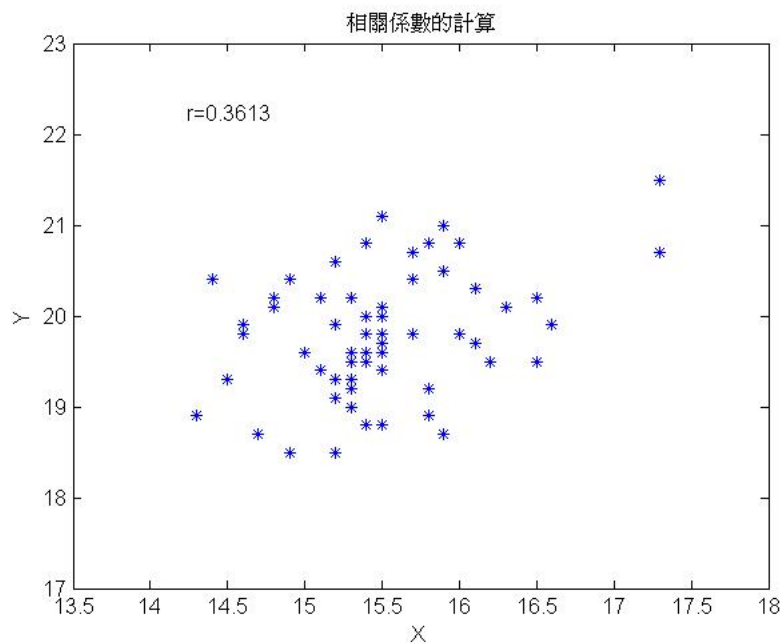


圖 1: 散佈圖與相關係數的計算

補充1:

作業所採用的資料來自檔案, 如 `data1.txt`。在 MATLAB 中讀取檔案的方式為:

```
load data1.txt
```

從 MATLAB 讀取外部資料檔時, 必須注意兩件事:

1. 資料檔案的位置: 當開啓 MATLAB 軟體時, MATLAB 主動設定其工作區域在預設的目錄 (隨版本與安裝的目錄相關), 執行上述指令時, 資料檔案所在的位置必須與工作目錄相同。如果不同, 作法有二: (1) 將檔案轉移至預設的工作目錄 (2) 更改工作目錄至檔案所在位置。第二種方式比較常見, 更改目錄的作法如圖 2 所示。

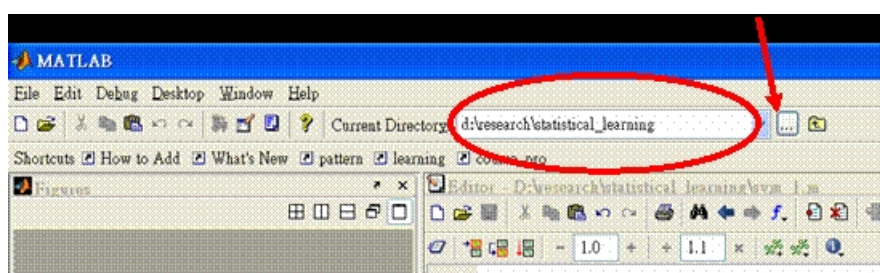


圖 2: 更改工作區域目錄

2. 檔案的格式: MATLAB 支援幾種資料檔案格式, 常見的有文字檔 (*.txt) 及 EXCEL 檔, 另一種為 MATLAB 專用的 binary 檔 (*.mat), 本章暫不討論。其中文字檔需以類似 EXCEL 的行列方式編輯, 讀入後將被視為矩陣的行列。

當使用 whos 指令察看命令視窗的變數時, 將發現新的變數 data1, 代表下載的資料檔案 data1.txt 的內容, 其大小為 $N \times 2$, 其中第一行代表變數 X , 第二列代表變數 Y 。通常資料讀入後, 會依實際情況將內容指定給特定變數, 方便後續的運算, 譬如

```
x=data1(:,1):  
y=data1(:,2):
```

檔名直接轉為變數名稱是 MATLAB 內定的作法, 如果不喜歡或嫌麻煩, 當然可以換, 譬如, $Y=data1$ 。不過也可以一氣呵成, 如,

```
Y=load('data1.txt');
```

讀取 EXCEL 檔的指令與 TXT 檔不同, 如 (先下載資料檔 data1.xls)

```
Y=xlsread('data1');
```

另外, 指令 `importdata` 可以讀取多種不同型態的檔案, 譬如, TXT, MAT, XLS, WAV, AU, JPG...。做法如下

```
Y=importdata('data1.xls');  
x=Y.data(:,1);  
y=Y.data(:,2);
```

對大部分檔案格式而言, MATLAB 會以一個結構式的變數存放資料及附加的訊息。結構式的變數是一種多層結構的變數, 第一層是變數名稱, 第二層才開始存放資料或其他相關訊息。取用的方式如上第二行, 以 `'.'` 的方式連結兩層變數名稱。使用前可以先在命令視窗將變數叫出來看看。詳細的用法無法在此一一說明, 有興趣的讀者不妨到線上手冊察看, 然後試讀不同的檔案, 相信很快便能上手。

補充2:

作業中要求將計算出來的 r 值寫在散佈圖上, 這需要用到指令 `text` 或 `gtext`, 其差別在 `text` 必須指定 (x,y) 座標, 而 `gtext` 則可以使用滑鼠選定適當的位置, 詳細的使用方式請自行參考使用手冊。勤用 `help` 是必要的, 畢竟每個指令都有不同的參數給定方式, 簡單的 `help` 一下, 立刻便能查知, 無須記憶。

補充3:

散佈圖的觀察會隨著 X, Y 軸的座標範圍而在目視上有所不同, 適當的 `zoom-in` 或 `zoom-out` 可以讓圖形的表現更貼切, 如圖1並非畫出來就長這樣。畫完圖之後座標軸可以利用指令 `axis` 加以改變, 如

```
axis([13.5 18 17 23])
```

這四個數字的意義不難從圖中窺知, 但詳細的介紹與使用方式請利用線上使用手冊或輸入 `help axis` 指令。如果只想改變 X 或 Y 軸的範圍, 則可以使用指令 `xlim` 或 `ylim`, 使用前請參考手冊上的說明。