陣列程式語言 J 概論

郭平欣

國立東華大學經濟學系副教授

pkuo@mail.ndhu.edu.tw

認識 J.....	1
學習 J.....	2
安裝與執行.....	3
第一次接觸.....	3
算數運算.....	3
數字格式.....	5
線性代數.....	5
邏輯比較.....	7
計算 e 值.....	7
平均數與三叉動詞.....	8
黃金率與雙鉤動詞.....	9
結語.....	10

認識 J

本文的目的在介紹電腦程式語言 J 的一些基本功能。J 是由 Jsoftware 公司生產的一種陣列語言，為 APL 的繼承者，與 Java、JavaScript 或 JScript 沒有關係。它的命名早過 Sun 的 Java，只是由於普遍性不如，名字被鳩佔鵲巢了。

若要處理一個以上的數字，不論是時間數列或者田野調查的資料，集合（陣列）是比較有效率的處理單位，而 J 就是被設計來處理陣列資料工作的。因此特別適合計量經濟、財務金融與保險業或任何其他需要統計與分析大量資料的使用者；在講授前述課程，或者線性代數及數值分析等類似課程，是配合學生實習很好的工具。

J 尤其擅長處理陣列資料的數學、統計與邏輯分析，讓許多複雜的工作，只要一個或數個字元的指令就可完成。

例如，定義一個 2x2 的矩陣 A，指令為 `2 2 $`，其中 `2` 代表指定，`$` 的前面為矩陣的形狀，後面接著為要輸入矩陣的資料，將會依序置入 A 的各列。矩陣只是 2 維陣列的一個例子，用類似的方法可以輕易地輸入任何維度與形狀的陣列資料。

計算反矩陣是線性代數一個重要的工作，用 J 更是易如反掌，指令只有「`%`」兩個字元，故在前例 A 為方陣的情況下，得到 A 的反矩陣，並將之置於 B。

至於矩陣的乘法，其指令為「`+/*`」，故會得到單位矩陣。另一個長得很類似的指令「`-/*`」被用來計算行列式，故會得到 A 的行列式。如果要計算 A 各列的平均數，用 `+/` 即可。

以上幾個簡單的例子，可以看出 J 語言的強大功能。J 提供了許多種類似的指令與函數來處理文字或者數字陣列的相關工作，對於為作研究需要寫程式者，將會很高興地發現用 J 寫程式是如此的精簡而迅速，複雜的矩陣操作工作就好像用計算器做算數運算一般。話說回來，由於 J 支援 Pocket PC，若將之灌入 PDA，那還真的是一台超級計算器呢！

J 並不是憑空生出來的新的程式語言，它是由 APL 衍生出來的一般目的高階程式語言。對程式語言發展熟悉者，可能會聽過由 Ken Iverson 發展出來的 APL 語言，由一堆稱為 Iverson 符號的奇怪符號組成，例如行政院主計處過去就曾在 IBM 主架電腦上用 APL 來做台灣的經濟預測。APL 最強的功能是那一堆處理陣列的符號。許多工作若以其他語言來寫，需要許多行的指令，APL 只要少數幾個符號就解決了。

世間事往往利弊相隨，APL 的特異功能多，但是相對就不易學習。不過對中國人來說，中國字就是一堆符號，幾千幾萬個字都在學了，不差這幾百個。但是 APL 所用的符號，大半在一般鍵盤上找不到，這就有點麻煩了。在過去，IBM 主架電腦必須用很貴的終端機，這就大大地不利於 APL 的普及。到了 PC 年代，總算有一些公司將 APL 轉過來，但是用特殊符號不利於跨平台作業。且對老外來說，怪怪的符號總是學習的障礙，不利於推廣，所以一直有人想要以

標準的 ASCII 來取代。

解鈴還需繫鈴人，Ken Iverson 又與 Roger Hui 共同研發設計 J 語言來取代 APL，但是這回只用標準的 ASCII 字元，將 APL 功能強大的特殊字元，完全以標準文數字拼字的方式來取代。

在 APL 數十年經驗的基礎上，J 自 1990 起又經過了十數年的演化，雖然目前仍在發展中，整套系統基本上已經相當穩定。它的 ASCII 符號表示方法設計得非常巧妙，使得即使是使用簡單的字母，也能夠建構出一套完整而有一致性的語言體系。作為一個完整的程式語言，J 也提供整合的程式發展環境，具有編寫應用程式的圖形介面與設計程式視窗的工具，以及與其他程式語言或應用程式溝通的介面，而且已經建有許多標準的程式庫、公用程式與套裝程式組，對學習 J 語言與開發研發應用程式提供很大的助益。

由於使用標準 ASCII，J 支援許多種工作平台。例如微軟視窗家族的 Windows XP、Pocket PC 等，Apple Macintosh family、Palm PC、Handheld PC、Linux、RS6000/AIX、SunOS、NetBSD 以及 FreeBSD 等 Unix 系統。

J 語言由 jsoftware 公司在銷售，過去一套七、八百美元跑不掉。對一個還不確定能用來做啥的新學習者來說，這個價錢實在不便宜。不過在千禧年之後，大概為了要打破曲高和寡的魔咒，該公司居然大方送，個人使用者或初次使用者都可以免費註冊，包括除錯與保密等所有完整功能。這一套功能強大的跨平台語言，對於需要處理大量研究數據，或者作數值分析的使用者，絕對值得花一點時間學習，必然是終生受用。有興趣想進一步瞭解或者使用者，可以到該公司之網頁看看，其網址為 www.jsoftware.com。

學習 J

想要完全精通 J 並不是那麼容易，它不像一般程式語言，只要懂語法就可以了。若沒有陣列的概念，學習 J 永遠不會懂反矩陣指令到底在作啥；不懂統計，就不可能欣賞 J 求迴歸係數的精簡迅速。因此當在學 J 的同時，也必須學習其相關的知識。如果已經有相關知識的背景，學起來才會是事半功倍。

雖然完全精通不易，但是 J 語言的功能繁多，一些不需要特殊知識背景的功能，例如一般數學指令、字串處理功能以及編程指令等，要上手並不難。只是有許多複雜的符號以及語法，卻也可能讓想要登堂者望之卻步。本文的目的是讓新手能夠很快的上手，略覓堂奧。一旦入門，學習過程的喜悅自然會讓想要深入研究者上癮，欲罷不能。

學習一個新語言的最佳途徑就是看範例。看看別人怎麼作，依樣畫葫蘆即可，故本文用一些範例用來展示 J 語言的基本指令。所選的範例只是 J 所有功能的一小部分，主要是數學或統計方面的問題，因為比較容易瞭解。各範例顯示互動式 J 的輸入與輸出，並附上數行的說明。

除了看範例，依樣畫葫蘆也是很重，所以讀者不妨進入 J 系統，邊看邊作，作了以後，看過的知識才會變成自己的。

安裝與執行

先到 www.jsoftware.com 取得自己電腦作業平台的最新版本，依照其說明將 J 系統安裝完成。以下的學習將以 PC 作業平台視窗版本為準。安裝好以後，程式集裡就多了一個 J 的檔案夾，按 J 的圖案即可進入 J 預設的第一個工作區 1.ijx。

工作區為 J 的執行視窗，是互動作業的區域，任何指令輸入後立即執行，完成後將結果立即送回該視窗。工作區的副檔名為 ijx，使用者可以由 J 系統的選單 File->New IJX 開啟新的工作區，且同時可以開啟多個工作區。

使用者的輸入列與 J 回應的輸出列都在相同的工作區顯示，差異在於輸入列內縮四個空白，而 J 的回應只有一個空白縮頭。安裝完成以後，測試是否安裝成功的最簡單測試為，在工作區鍵入大寫之 CR，按 enter 鍵輸入後若回應空行，如下所示，表示安裝都沒問題，可以開始使用 J 系統了。

CR	NB. 輸入指令
	NB. 回應空白

陰影的段落為 J 系統的指令與回應，NB. 之後為註釋。

第一次接觸

算數運算

在進一步介紹之前，先來看一些算數運算的範例。在工作區直接敲進下列句子，按下 enter 後立即得到結果的回應。

```
1^2+3-4*5%6
1
```

算數運算元加減乘除分別以 +-*% 代表，而 ^ 為乘冪。讀者一定會覺得奇怪，前式如此複雜的混合運算，結果怎麼會是 1 呢？

傳統數學規定，算數運算的優先順序(precedence)為乘冪最優先，再過來先乘除後加減。而 J 語言的規則更簡單，一切指令都是由右向左執行。這是因為 J 的運算元非常之多，非但不易訂出優先順序，即使勉強為之，也沒幾個使用者能夠搞得清楚，因此乾脆以簡單法則規定，一律由右向左。

知道 J 的指令由右向左執行以後，上例的結果就很明白了，乘冪最後做，不論 2+ 右方的結果為何，1 的任何乘冪永遠為 1。

雖然這樣的規定可以將指令的優先性化繁為簡，可是算數運算不合一般的習慣，是必須付出的代價。新的習慣不難養成，真受不了，也可以用小刮號的方式來解決優先順序的問題。下式即會先算刮號內的乘冪，再加上右方運算之結果。

```
(1^2)+3-4*5%6
0.666667
```

本例的結果為一個無限循環小數，但顯示的結果只有六位數的精確度，這只是預設的列印精確度，J 在電腦內儲存的精確度遠高於此。要更改列印精確度

的方式如下：

```
pps =. 9!:11
pps 9
(1^2)+3-4*5%6
0.666666667
```

其中=.為指定指令，用來指定 9!:11 給 pps，這相當於將 pps 指定為列印精確度的指令，而用 pps 9 來指定列印精確度為 9，列印精確度的上限為 20 位數。

任何有理數都可以表示成分數，要將計算結果以分數表示，在最前方加上 x:，以前例

```
x:(1^2)+3-4*5%6
2r3
```

得到精確的結果。

若運算的引數不只有一個數字，合法的運算必須兩邊維數相同，或一邊為純量。若兩邊維數相同，運算元作用於每一個對應的元素；若一邊為純量，則作用於另一邊的每一個元素。例如

```
1 2 3+4 5 6      NB. 向量相加
5 7 9
1+4 5 6          NB. 純量加上向量
5 6 7
```

數字格式

除了一般的實數外，J 語言也定義了許多其他數字表達形式。複數以 j 連結實數與虛數而成，如 3j4 代表 3+4i；分數由 r 連結兩個整數，如 3r4 代表 3/4；負號以 _ 來與動詞減去符號-作區別，動作-3 的結果為名詞_3。

```
%:_9          NB. 開根號
0j3
2r3*3r4       NB. 分數相乘
1r2
```

式子中 NB.的後面為註釋。前例%:為開跟號指令，根號負九得到虛數三；而由分數相乘的例子可以看出，計算的結果會自動化簡。

其他類似的數字表達式還有科學計數，例如 表作

```
2e3          NB. 2*10^3
2000
```

既然可以用 e 代表以 10 為底，當然也可以用別的常數為底來表達數字。以圓周率為底的科學計數， $2\pi^3$ 表作

```
2p3          NB. 2*Pi^1
62.0126
```

以尤拉數 e 為底的科學計數， $2e^3$ 表作

```
2x3          NB. 2*Pi^1
40.1711
```

相當於 $2 * (2.71828)^3$ 。

如果要以非十進位轉換成十進位，可用數字 +b 為底，例如

```
16b1a
26
```

其中 16b 指定數字 1a 為十六進位，輸入後立即轉換成十進位的結果 $1 * 16^1 + 10 * 16^0$ 。

線性代數

J 的強項在於陣列的處理，因此線性代數的應用是最合適不過的。首先用 J 隨機產生一個 3x3 的矩陣 A，

```
]A=:?3 3$100
66 99 87
47 87 30
68 46 9
```

其中 3 3\$100 產生一個內容全為 100 的 3x3 方陣，?指令則在方陣的每個元素都以均等分配的方式產生一個介於 0-99 的整數，亦即每個元素都隨機取自前 100 個指標(非負整數)。A 左方的「]」指令，功能是将新指定變數 A 的內容顯現出來。

矩陣相乘的指令為「+/.*」，「+/'與「.*」之間務必要有一個空白。以前例的 A 自乘

```
A+/.*A
14925 19149 9495
9231 13602 6969
7262 11148 7377
```

而「-/.*」為行列式的指令，「-/'與「.*」也要有一個空白。前例 A 的行列式為

```
-/.*x
_205917
```

求解線性聯立方程組 $Ax=b$ ，得到 $x=A^{-1}b$ 。照舊讓 J 先隨機產生向量 b

```
]b=:.? 3$100
6 82 38
```

A 的反矩陣用單邊動詞「%.」，故「%.A」得到 A^{-1} 。記得要先用小刮號將求 A 反矩陣的指令包起來，否則會得到 Ab 的反矩陣。

```
]x=(%.A)+/.*b
_0.372762 1.68301 _1.5634
```

以下是求解 x，並驗證答案 $Ax=b$ 。

```
]x=(%.A)+/ .*b
  A+/ .*x
6 82 38
```

J已經將以上求解過程簡化，同樣用動詞「%.」，但是兩邊各放一個變數，稱之為雙邊動詞。故可以寫成

```
b%.A
_0.372762 1.68301 _1.5634
```

同樣的指令可以用來計算線性迴歸模型的迴歸係數。當線性迴歸模型為 $y=Xb$ ， y 為應變數向量，亦即被解釋變數的觀察值， X 為自變數矩陣， X 的列代表觀察樣本，行代表解釋變數，觀察樣本的個數不得低於解釋變數的個數。估計的迴歸係數為

矩陣運算是最耗費電腦運算資源的，J 提供許多衡量系統的指令，稱之為外務詞，其中一個稱之為計時外務詞指令(6!:2)，可以用來衡量指令的執行時間。以下為計時指令的範例，

```
A=:?100 100$10000
  500 (6!:2) '%.A'
0.0130713
```

上例先產生一個 100x100 的矩陣 A，每個元素都隨機取自前 10000 個指標數(非負整數)，再詢問計算反矩陣的時間。計時指令的左引數為重複執行的次數，右引數以單引號將執行指令刮起來，結果得到重複 500 次後，計算 A 的反矩陣平均每次需要 0.0130713 秒。

邏輯比較

如同許多其他程式語言，J 將真值訂為 1，而偽值訂為 0。邏輯運算元的「且」表作*，「或」表作+，而「非」表作-。或與且的真值表可作如下

```
d (+./ ; *./) d=: 0 1
+---+---+
|0 1|0 0|
|1 1|0 1|
+---+---+
```

由右方開始， $d=:0 1$ 定義 d 為二個元素的向量，而 $d(...)$ 表示將左方的 d 與右方的 d 做運算。運算元在刮號內，以分號區分不同的運算元，本範例有兩個運算元，+./與*./，其中/代表插入，故將或(+)及且(*)插入左方 d 與右方 d 的所有元素間，得到邏輯運算表。相同的元素分別做兩種運算，結果分別置於不同的格子內，如前面的結果所顯示。

用來比較大小的二元關係有小於<、小於或等於<:、等於=、大於或等於>:、大於>等。

函數=| 用來測試數字是否等於其絕對值，故為非負檢定；函數=<.用來測試

數字是否等於其底整數，故為整數測試。而(=|)+.(=<.)測試數字為非負或整數：

```
(test=: (=|) +. (=<.) ) _2 _2.4 3 3.5
1 0 1 1
```

由於電腦內數字儲藏方式為二位元，因此精確度有其限制，數字關係的比較無法做到完全正確，因此存在容許誤差(tolerance)，為比較數字中最大數字的 2^{44} 。

計算 e 值

雖然 J 可以用指令 ^1 來計算尤拉常數 e，以下將以其泰勒展式的計算，來展示 J 的級數計算功能。由泰勒展式我們知道

$$e = \sum_{n=0}^{\infty} \frac{1}{n!}$$

一般而言，取前十項已經足夠接近，我們可以 0 到 9 的倒數數列相加來估算自然對數。以 J 來做，指令如下所示：

```
+/ % ! i.10
2.71828
```

其中「i.10」產生 0 到 9 的數列，稱為取前 10 個指標。接著以階層動詞「!»計算這個指標數列每一元素的階層，並以倒數動詞%分別計算倒數，最後以加總動詞「+/'得到結果 2.71828。

以數學的眼光看，由右而左的運算是有些奇怪，但是若以英語語法來看，倒是很自然，"the sum of the reciprocals of the factorials of the first 10 indices"，當然如果要以中文來念也是可以，只是必須從右方念起，「前 10 個指標取階層後倒數的加總」。

平均數與三叉動詞

平均數是 J 語法三叉(fork)應用的最佳範例。三個動詞串在一塊稱之為三叉，三叉動詞若為雙邊，其意義如下：

$x (f g h) y$ 代表 $(x f y) g (x h y)$ ，

若為單邊，其意義如下：

$(f g h) y$ 代表 $(f y) g (h y)$ 。

```
Sum =. +/
Average =. Sum % #
c =. 1 2 3 4
Average c
2.5
```

前兩行定義代動詞，第三行定義代名詞，第四行以代動詞「Average」處理代名詞「c」。Average c 可以直接寫作

```
(+/ % #) 1 2 3 4
```

其中「(+/%#)」為加總「+/'、除以「%」、計數「#」三個動詞的串連，因此

(+ / % #) 1 2 3 4

相當於

(+ / 1 2 3 4) % (# 1 2 3 4) ,

亦即對於給定的數列 1 2 3 4 , 加總後再除以其項數。

三叉(LT OR EQ)的意義可以由下述敘述間的相關性來理解：

x (LT OR EQ) y

(x LT y) OR (x EQ y) 。

黃金率與雙鉤動詞

黃金率是西方傳統認為最適宜人眼的圖畫寬與高的比例，一般以 τ 表之。黃金率減 1 的倒數恰好為其本身，略微計算得知 $\tau = (1 + \sqrt{5})/2$ 。以 J 計算，

- : 1 + % : 5

1.6180339887498949

其中「- :」為對半，而「% :」為根號，因此上式可以想成「對半 1 + 根號 5」。黃金率也可以表成

$$\tau = 1 + \frac{1}{1 + \frac{1}{1 + \dots}}$$

式中 τ 減 1 的倒數仍為 τ 。上述定義為無限多項倒數相加，若以有限項來求近似值，例如 4 項，則可寫成

$$\tau = 1 + \frac{1}{1 + \frac{1}{1 + \frac{1}{1}}}$$

為((((1 的倒數)加 1)的倒數)加 1)的倒數)加 1。以 J 計算的式子為

(+%) / 1 1 1 1

1.6666666666666665

動詞串「(+%)」為「加倒」，兩動詞串在一塊稱為雙鉤(hook)動詞，「(+%) /」將加倒雙鉤動詞由右到左依序加諸於陣列之各個元素，因此對 1 構成的數列連續加倒，就成為黃金率的近似值，數列愈長近似值愈接近真值。長度為 m 元素為 n 的數列以 J 表示為 m\$ n，因此前式可表為(+%) / 4\$ 1。當數列長度為 39 時，

(+%) / 39\$ 1

1.6180339887498949

計算出的近似值與在精確位數 20 之下的真值相同。

雙鉤動詞必須以小刮號圍繞，因為若不用刮號，動詞將由右方先執行，意義完全不同。例如

+% / 1 1 1 1

1

先執行連續除「%/ 1 1 1 1」得到 1，再由「+ 1」取 1 的共軛數，結果仍然為 1。

除了以刮號界定雙鉤動詞外，也可以代動詞定義雙鉤動詞。例如

```
ar=.+%  
ar/1 1 1 1  
1.6666666666666665
```

結語

讀到這裡，相信讀者已經可以領略J的強大功能，也會有動機進一步學習。J系統已經有相當不錯的學習規劃，進入J系統後，由選單 Help->Help 呼叫J的輔助系統(J Help System)，為超文體(HTML)寫成，有鍊結到一些相當有用的參考文章。初學者可以先閱讀 Eric Iverson 的 J Primer，或者 Roger Stokes 寫的 Learning J，很快就可以應用自如；進階者可以由 J Phrases 或者 J 系統提供的文稿檔學得許多技巧。J Dictionary 有所有系統單字的定義，若有人願意發揮國中時期背英文字典的精神，是一個迅速入門的途徑。若要查詢某單字的用法，也有依字母順序排列的索引可以檢索。加入 [J Forum](#) 也是個不錯的考慮，一些高手常在上面替人解惑，也可由其中瞭解J的最新發展。

有這麼多資源，學習J絕對容易，而其強大的功能絕對值得花時間去學習，當然，邊用邊學用還是成功學習的不二法門。

參考文獻

- Dickey, Leroy J. (1992) "What is J? An introduction," University of Waterloo.
- Hui, Roger K.W. and Kenneth E. Iverson, (2001) J Dictionary, Jsoftware Inc, 5/3.
- Iverson Eric, J Primer, (2001) Jsoftware Inc, 3/29.
- Burke, Chris, Roger K. W. Hui, Kenneth E. Iverson, Eugene E. McDonnell and Donald B. McIntyre, (2001) J Phrases, Jsoftware Inc, 3/29.