矩陣大檢閱

丁南僑
香港大學數學系

我們從小就懂得把單個的數字，作加減乘除等等的運算，以解決一些日常碰到的問題。例如四個人均分十二個水果，我們懂得把十二除以四，求得每人可分得三個水果。但是假如有關的問題涉及太多數字，而數字之間又有內在的關係，以單個數字的運算方法再難以把問題解決的時候，就可能需要把眾多的數字結合成一組一組，通過組與組之間的一些特別的運算方法，才容易把問題解決。

將數字作長方形排列，所成的組合，就叫矩陣。

讓我們看看以下的例子。相信大家都到過快餐店吃漢堡包和薯條，另一方面，營養專家又告訴我們，吃得太多這類食物對健康無益。現在就讓我們分析一下這些食物的營養成分。

食物給我們的養分，主要是碳水化合物、蛋白質、和脂肪。當然，除此之外，還有纖維質、維他命、礦物質等等，但為了簡單起見，我們暫且只考慮剛才提及的前三項。

根據資料，我們知道一個漢堡包的麵包，含有20克碳水化合物，2克蛋白質，2克脂肪。漢堡包裡頭的那塊牛肉連醬料，共含有5克碳水化合物，16克蛋白質，7克脂肪，至於薯條和汽水，我們亦有類似的資料。我們可以把這些數據，排列成以下的矩陣：

$$
\begin{array}{r c}
 & \begin{array}{cccc} \text{麵包(每個)} & \text{牛肉(每塊)} & \text{薯條(每十克)} & \text{汽水(每百毫升)} \end{array} \\
\begin{array}{r} \text{碳水化合物(克)} \\ \text{蛋白質(克)} \\ \text{脂肪(克)} \end{array} & \begin{bmatrix} 20 & 5 & 4 & 10 \\ 2 & 16 & 1 & 0 \\ 2 & 7 & 2 & 0 \end{bmatrix}
\end{array}
$$

這個矩陣有三個橫行，四個縱列，我們稱它為一個三乘四的矩陣。它的第一、二、三、四列，羅列了一個麵包、一塊牛肉、十克薯條、和一百毫升汽水所含的養分。

本文根據作者於一九九六年十月二十六日，在香港大學主辦的「數趣漫話」講座中的演講內容整理而成。另:作者在預備講座期間，得周偉文、蕭文強、歐陽亦鷖、朱進強等諸位先生提供寶貴意見，為此謹表謝意。

一份漢堡包餐有一個麵包和一塊牛肉、一包50克的細薯條，和一杯200毫升的細汽水。如果你嫌不夠飽，也可以選擇巨無霸餐。下面的矩陣，列出了漢堡包餐和巨無霸餐的配料：

$$
\begin{array}{r|cc|}
 & \text{漢堡包餐一份} & \text{巨無霸餐一份} \\
\text{麵包(個)} & 1 & 1.5 \\
\text{牛肉(塊)} & 1 & 2 \\
\text{薯條(10克)} & 5 & 8 \\
\text{汽水(100毫升)} & 2 & 4
\end{array}
$$

我們把這個矩陣叫作 B，把剛才的矩陣叫作 A，將 A 放在 B 的隔鄰，再作一些簡單運算，就可以求得一份漢堡包餐所含的碳水化合物有多少：

$$20 \times 1 + 5 \times 1 + 4 \times 5 + 10 \times 2 = 65\text{克}$$

$$
\begin{bmatrix} 20 & 5 & 4 & 10 \\ 2 & 16 & 1 & 0 \\ 2 & 7 & 2 & 0 \end{bmatrix}
\begin{bmatrix} 1 & 1.5 \\ 1 & 2 \\ 5 & 8 \\ 2 & 4 \end{bmatrix}
$$

至於巨無霸餐的蛋白質含量，則是：

$$2 \times 1.5 + 16 \times 2 + 1 \times 8 + 0 \times 4 = 43\text{克}$$

$$
\begin{bmatrix} 20 & 5 & 4 & 10 \\ 2 & 16 & 1 & 0 \\ 2 & 7 & 2 & 0 \end{bmatrix}
\begin{bmatrix} 1 & 1.5 \\ 1 & 2 \\ 5 & 8 \\ 2 & 4 \end{bmatrix}
$$

用同樣的方法，我們可以求得每份 P 餐所含的養分 Q 的份量，這裡 P 是漢堡包餐或巨無霸餐，Q 是碳水化合物、蛋白質或脂肪。我們把得到的結果，寫成以下矩陣：

$$
\begin{bmatrix} 20 & 5 & 4 & 10 \\ 2 & 16 & 1 & 0 \\ 2 & 7 & 2 & 0 \end{bmatrix}
\begin{bmatrix} 1 & 1.5 \\ 1 & 2 \\ 5 & 8 \\ 2 & 4 \end{bmatrix}
=
\begin{array}{r|cc|}
 & \text{漢堡包餐一份} & \text{巨無霸餐一份} \\
\text{碳水化合物(克)} & 65 & 112 \\
\text{蛋白質(克)} & 23 & 43 \\
\text{脂肪(克)} & 19 & 33
\end{array}
$$

我們把這個三行、兩列的矩陣叫作 C，稱它爲矩陣 A 與矩陣 B 乘積，寫作 $AB=C$。其中矩陣 C 之內的位於第 i 行、第 j 列的數字，是矩陣 A 的第 i

行乘以矩陣 B 的第 j 列的結果。這就是矩陣的乘法。請留意矩陣 A 的列數剛好等如矩陣 B 的行數，否則 A 的行不能乘以 B 的列。所以，在這個例子中，AB 是符合定義的乘積，但 BA 則無解，因爲 B 的列數不等如 A 的行數，B 的行不能乘以 A 的列。由此可見，矩陣乘積 AB 與 BA 一般來說並不相等。

讓我們看看剛才的快餐所含的熱量。原來無論碳水化合物、蛋白質、脂肪都可以被人體消化吸收而產生能量。以下是有關的換算表：

$$\begin{array}{cc} & \begin{array}{ccc} \text{碳水化合物一克} & \text{蛋白質一克} & \text{脂肪一克} \end{array} \\ \text{熱量(卡路里)} & \begin{bmatrix} 4 & 4 & 9 \end{bmatrix} \end{array}$$

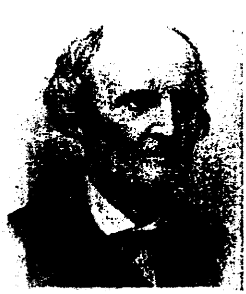
這是一個由三個數字組成的向量，但亦可以看成是一個一乘三的矩陣，讓我們稱它作 E 。將 E 乘以剛才得到的矩陣 C，結果是：

$$\begin{bmatrix} 4 & 4 & 9 \end{bmatrix} \begin{bmatrix} 65 & 112 \\ 23 & 43 \\ 19 & 33 \end{bmatrix} = \begin{array}{cc} & \begin{array}{cc} \text{漢堡包餐一份} & \text{巨無霸餐一份} \end{array} \\ \text{熱量(卡路里)} & \begin{bmatrix} 523 & 917 \end{bmatrix} \end{array}$$

恰恰是一份漢堡包餐或一份巨無霸餐所提供的熱量。

在以上的例子，我們通過三個矩陣 E、A、B 和它們的乘法，將漢堡包餐、巨無霸餐、麵包、牛肉、薯條、汽水、碳水化合物、蛋白質、脂肪、和熱量之間的多角關係清楚簡潔的表述及計算出來，可見矩陣及它的乘法確是非常有效的計算工具。

以上介紹的矩陣乘法，是由英國數學家 Arthur Cayley 在 1858 年的一篇論文內首先提出的。百多年前的英國，還沒有漢堡包連鎖店，Cayley 也不曾計算過漢堡包和薯條汽水的營養成分。究竟 Cayley 是受甚麼啓發而「發明」了矩陣的乘法呢？原來 Cayley 是在研究綫性變換理論的過程中，發現了矩陣乘法的規律。除了矩陣的乘法之外，Cayley 在他的論文內又爲矩陣的加法和乘上單個數字的乘法作出定義，Cayley 稱這套數學體系爲「矩陣代數」，從此也開展了後世對矩陣理論的研究。所以，Cayley 可算是矩陣理論之父。



Arthur Cayley
(1821-1895)

矩陣的乘法雖然只有一百多年的歷史，但人類在更早的時候已懂得用矩陣方法解決方程問題。

中國有一部很古老的數學書籍，叫《九章算術》，這部書大約在公元一世紀的東漢時代寫成，書內有二百四十六道數學應用題，按題目的類形分爲九個章節，所以書名叫作《九章算術》。其中第八章是「方程」，講述方程的解法。它的第一題是這樣的：

「今有上禾三秉、中禾二秉、下禾一秉，實三十九斗；上禾二秉、中禾三秉、下禾一秉，實三十四斗；上禾一秉、中禾二秉、下禾三秉，實二十六斗；問上、中、下禾實一秉各幾何？」

這裡說有上、中、下三級的扎成一捆一捆的禾稻。三捆上禾、兩捆中禾、一捆下禾合共脫得三十九斗的穀粒；兩捆上禾、三捆中禾、一捆下禾合共脫得三十四斗穀粒……問上、中、下各級的禾稻每捆可脫得穀粒多少？這是一組包含三條綫性方程、三個未知數的聯立方程式。如果以 x，y，z 表示上、中、下禾各一捆所脫得的穀粒的量，這方程組就是：$\begin{cases} 3x + 2y + z = 39 \\ 2x + 3y + z = 34 \\ x + 2y + 3z = 26 \end{cases}$。

《九章算術》中給出的解法如下：

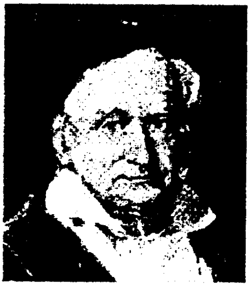
「置上禾三秉、中禾二秉、下禾一秉，實三十九斗於右方；中、左禾列如右方。以右行上禾遍乘中行而以直除，爲術之意，令少行減多行，反覆相減，則頭位必先盡…」

根據指示，我們把 3,2,1,39 寫成一個矩陣的右方的縱列，再以同樣方法寫下矩陣中間和左方的兩列，成爲一個四乘三的矩陣。然後把右列頭頂的 3 乘到整個中間的縱列去，再將結果逐次地減以右列，直至中列頭頂的數目變爲 0：

$$\begin{bmatrix} 1 & 2 & 3 \\ 2 & 3 & 2 \\ 3 & 1 & 1 \\ 26 & 34 & 39 \end{bmatrix} \rightarrow \begin{bmatrix} 1 & 6 & 3 \\ 2 & 9 & 2 \\ 3 & 3 & 1 \\ 26 & 102 & 39 \end{bmatrix} \rightarrow \begin{bmatrix} 1 & 3 & 3 \\ 2 & 7 & 2 \\ 3 & 2 & 1 \\ 26 & 63 & 39 \end{bmatrix} \rightarrow \begin{bmatrix} 1 & 0 & 3 \\ 2 & 5 & 2 \\ 3 & 1 & 1 \\ 26 & 24 & 39 \end{bmatrix}$$

用同樣方法處理最左列，矩陣變成：$\begin{bmatrix} 0 & 0 & 3 \\ 4 & 5 & 2 \\ 8 & 1 & 1 \\ 39 & 24 & 39 \end{bmatrix}$。這時把中列的第一個非零數字 5 遍乘左列，再將結果逐次減去中列，矩陣變成：$\begin{bmatrix} 0 & 0 & 3 \\ 0 & 5 & 2 \\ 36 & 1 & 1 \\ 99 & 24 & 39 \end{bmatrix}$。從結果的最左列，可以求得下禾一捆有 $\frac{99}{36} = 2\frac{3}{4}$ 斗穀粒，將這結果代入中間的縱列，可以求得中禾的答案，最後將下禾、中禾的答案代入右列，便可求得上禾的答案。

顯然地，這個方法除了解決這道三個未知數、三條方程的方程組以外，也可以爲一般的 n 個未知數、n 條綫性方程的方程組求解。這個按步驟的、系統化的方法，實在是中國古代數學的重大成就。在西方的典籍中，系統化的綫性方程組的算法記載，比較《九章算術》起碼要晚了 1500 年！



Carl Friedrich Gauss
(1777-1855)

到了十九世紀初，德國大數學家 Gauss 在計算小行星 Pallas 軌跡的時候，從觀察小行星所得的數據資料，得到一組六個未知數、六條方程的綫性方程組。Gauss 自創了一套方法為這個方程組求解，這個方法，正正就是《九章算術》中的方法。

因著 Gauss 的大名，這個綫性方程的解法此後就被稱為 Gauss elimination method。直到現在，幾乎所有的綫性代數書本裡，都有關於 Gauss elimination 的內容。而這方法，到今天仍是解綫性方程組的最基本方法。那麼，Gauss elimination 是否就是解決綫性聯立方程組問題的最終答案呢？

今天的工程師和科技人員，面對愈來愈複雜龐大的數學問題，雖然有電腦的幫助，但也要用創新的計算方法，才可以把問題解決。例如，在量子化學、統計力學、流體動力學、氣象預測等等的研究裡面，科學家接觸的，很多時候是一些數萬乘數萬，或甚至是數十萬乘數十萬那麼大的係數矩陣。假使用 Gauss elimination 方法，一步一步地計算，就是用世界上最快的電腦，恐怕在我們的有生之年也解不到答案。此外，由於電腦的記憶體有限，不能任意逼近所有的實數，所以用電腦作數值運算，很難避免數值誤差的出現。這些難題，催生了一門新興的數學學科，就是數值綫性代數，也有人稱之為矩陣計算理論。它的主要課題，是研究怎樣可以更快捷、更準確地作矩陣的數值運算。

例如，很多工程問題的係數矩陣都有一些特別的結構，數學家便盡量利用這些結構的特性，設計一些既快且準的計算方法。此外，數學家和電子計算機專家也嘗試為一些用上成百個或更多的計算機處理器組成的超級電腦，設計矩陣的平行式的運算方法。矩陣計算理論這門學科，正是方興未艾，這是拜電子計算機科學的急速發展所賜。

現代社會的訊息傳遞，愈來愈倚賴圖像，而圖像也可以用矩陣的形式表示：一幅單色的圖像或照片，可以分割為 m 橫行、n 縱列，亦即分割為 mn 個小格子，每個小格子的亮度，用一個 0 至 255 之間的整數表示， 這裡 0 代表黑色，255 代表白色，接近 0 的數字代表較暗的灰色，接近 255 的數字代表較光亮的灰色，那麼，原來的那幅單色圖像，便是一個 m 乘 n 的矩陣，這個矩陣的每個元素，都是 0 至 255 之間的整數。由於任何 0 至 255 之間的整數，都可以表示成 8 位的二進制數字，例如 0000 0000 代表 0，1111 1111 代表 255，用電腦的術語來說，就是照片上的每個小格的亮度，可以用 8 bit、亦即一個 byte 表示。

至於彩色圖像，我們可以用紅綠藍三原色的光學原理和分色技術，把圖像分為紅、綠、藍三幅單色圖像，亦即把圖像分拆為三個矩陣。圖像中每個小格子的顏色，由三個分別代表紅、綠、藍亮度的數字表示。由於每個數字佔用 8 bit，所以每個小格子的顏色用了 8x3=24 bit 表示。套用電腦的術語，這是一幅 24 bit 的 true color 數碼圖像。

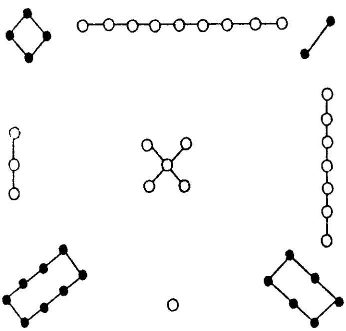
數碼圖像根本上就是矩陣，所以，當我們觀看數碼圖像的時候，事實上是看矩陣。我們這篇文章的題目，固然是「矩陣大檢閱」，大家有沒有想過，當你們用電腦觀看 video CD 電影光碟的時候，其實是在觀看一個一個的矩陣，也是在作「矩陣大檢閱」哩！

既然數碼圖像就是矩陣，對數碼圖像作影像處理，實際上是對矩陣作變換。因此，矩陣理論在數碼圖像處理技術方面，扣當了很重要的角色。

例如，右面的上圖是一幅從書本上拍得的照片，由於一些技術問題，照片拍得模糊不清。但通過對這照片影像作一些矩陣的計算，可以把影像變爲如右面下圖般清晰，從而辨認出照片中的文字。

另一項很重要的影像處理技術是影像壓縮，用以節省儲存數碼圖像所需的空間，和節省在電腦網絡上傳遞影像檔案所需的時間。歐洲、美國和日本正在研究開發的高解像電視系統，能否成功，也要看今後的影像壓縮技術有否新的突破。現今最通行的影像壓縮方法，用在照片的是 JPEG，用在電影的是 MPEG，兩者都是用一種叫 Discrete Cosine Transform (DCT) 的矩陣變換作基礎。一幅高解像度的彩色圖像，經過DCT作壓縮，所佔的空間（檔案長度）可能只是原來的數十分之一，而圖像的質素跟原本的那幅只有非常小的分別，人眼難以察覺。

以上所述的圖像處理，可說是矩陣理論的一些新應用，現在讓我們回過頭來看一個古老的矩陣。根據中國神話傳說，在大禹治水的時候，從黃河裡跳出了一隻龍馬，從洛水中跳出一隻神龜，這兩隻神獸各背著一幅圖，即「河圖」和「洛書」，其中的一幅是這樣子的，它又稱作「九宮」：



把這幅圖上一條條橫直綫上的圓點數目記下來，便是一個三乘三的矩陣。這矩陣中的數字剛巧是從一到九的整數，而且每一行、每一列和兩條對角綫上的數字之和都是十五。這種用 1 到 n^2 作元素、所有橫行、直列、對角綫的數字之和都相等的 n 乘 n 方形矩陣，我們稱作幻方。

中國古代數學家對幻方研究作出了不少貢獻，可惜後來這些研究受到陰陽

五行學說的影響，它的數學內容變得愈來愈少，神秘主義和迷信的色彩愈來愈強。

大家可能會問：是否對每個正整數 n，都可以構作出一個 n 乘 n 的幻方？$n=1$ 的情況太顯淺，讓我們考慮 $n=2$ 的情形。若要把 1，2，3，4 填入一個二乘二的矩陣內，1 必定出現在某一位置，但爲了達到在含 1 的行和列都有相同的數字和，這行和這列的剩下未塡的數字必須相同，這便違背了數字只能用一次的原則。所以，二乘二的幻方是不存在的。

至於 $n=3$ 的情況，剛才提到的九宮就是一個三乘三的幻方。事實上，只要 n 大於二，就有 n 乘 n 的幻方。至於幻方的構作方法，有興趣的同學可以看蕭文強敎授所著的《1，2，3，… 以外——數學奇趣錄》。

除了幻方，另一種名爲拉丁方的矩陣也很有趣。所謂拉丁方，就是一個 n 乘 n 的矩陣，它的每一行、每一列的元素所成的集合，都是同一個包含 n 個不同元素的集合。以下就是兩個四乘四的拉丁方：

$$\begin{bmatrix} a & b & c & d \\ b & a & d & c \\ c & d & a & b \\ d & c & b & a \end{bmatrix} \qquad \begin{bmatrix} a & b & c & d \\ d & a & b & c \\ c & d & a & b \\ b & c & d & a \end{bmatrix}$$

這兩個矩陣的每一行每一列所含的元素，都是 a, b, c, d，其中的第二個拉丁方，也同時是一個環迴矩陣。所謂環迴矩陣，就是一個 n 乘 n 的方形矩陣，它的元素有以下的規律：假如某一橫行的元素依次序是 a_1，…，a_n，則它的對下一行的元素是 a_n，a_1，…，a_{n-1}，亦即把原本排在最後的 a_n 放到最前面。如果 a_1，…，a_n 都不相同的話，環迴矩陣也同時是拉丁方。由此可見，n 乘 n 的拉丁方一定存在。

像這些幻方、拉丁方的研究，都是關於矩陣的元素的位置分佈和組合，這都屬於組合矩陣理論的範圍。組合矩陣理論，其中也包括了用矩陣的方法去解決組合學的問題。這門學科，可說是矩陣理論和組合學的結合成果。

組合矩陣理論之中的一個至今尙未解決的問題，是關於 Hadamard 矩陣的。所謂 Hadamard 矩陣就是一個正方形的矩陣，其中的元素只可以是 1 和 -1，此外它的任何兩行的對應位置上的元素，都有一半的元素有相同的正負號，另一半有相反的正負號。下面就是一個二乘二、一個四乘四的 Hadamard 矩陣：

$$\begin{bmatrix} 1 & 1 \\ -1 & 1 \end{bmatrix} \qquad \begin{bmatrix} 1 & 1 & 1 & 1 \\ -1 & 1 & -1 & 1 \\ -1 & -1 & 1 & 1 \\ 1 & -1 & -1 & 1 \end{bmatrix}$$

假如 H 是一個 m 乘 m 的 Hadamard 矩陣，那麼 $\begin{bmatrix} H & H \\ -H & H \end{bmatrix}$ 就是一個 $2m$ 乘 $2m$ 的 Hadamard 矩陣。用這個方法，由二乘二的 Hadamard 矩陣開始，可以構作出 2^k乘2^k 的 Hadamard 矩陣，其中 k 是任意的正整數 1，2，…。

另一方面，如果 H 是一個 n 乘 n 的 Hadamard 矩陣，而 n 是大於 2 的話，那麼，我們可以假設 H 的第一行的所有元素都是+1，否則可以把有 -1 作爲第一個元素的直列遍乘以 -1，結果並不影響 H 作爲 Hadamard 矩陣的特性。現在把第一、二行作排比，可知第二行的一半元素是 1 另一半是 -1，我們可以不失一般性地假設第二行的 1 都位於左邊，-1 都位於右邊。設第三行的左半元素有 a 個是 1，b 個是 -1，右半元素有 c 個是 1，d 個是 -1，由於左右兩半的元素數目相同，所以 $a+b=c+d$。將第一、三行作排比，可得 $a+c=b+d$ ；將第二、三行作排比，可得 $a+d=b+c$。由此求得 $a=b=c=d$，因此 n 必須是四的倍數。

1 1 1 1	
1 1 1	-1 -1 . . . -1
a 個 1, b 個-1	c 個 1, d 個 -1

我們很自然地反問：如果 n 是四的倍數，是否一定可以構作出 n 乘 n 的 Hadamard 矩陣？這個問題，數學家至今仍然未知答案。不過，對於小於 268 的所有四的倍數 n，我們知道 n 乘 n 的 Hadamard 矩陣是存在的。

組合矩陣理論有很多實際的應用，其中一項就是實驗設計。Hadamard 矩陣除了應用於實驗設計之外，也可以用來構作一種很有效的Reed-Muller自動改錯通訊電碼，美國太空總署就曾經用這種電碼傳送太空探測船飛近火星拍得的照片。

有關矩陣應用的其他例子，實在是太多了。例如，在統計學、物理學、經濟學、系統理論、控制論、代數學、編碼理論、馬可夫過程、圖論、網絡分析等等方面，矩陣可說是担當了重要的角色。但因爲篇幅有限，我們不在這裡詳述了。

一直以來，大學的數學課程，大多只是在綫性代數的科目內附帶地教一些矩陣的內容，但由於矩陣理論在各門應用科學的重要性愈來愈顯現，已經有一些外國的大學考慮開授有關矩陣理論的專門課程。香港大學也不例外。事實上，我們早在兩年前已經開始教授「矩陣理論及應用」的科目，這是香港首間大專院校在本科課程（undergraduate syllabus）開辦這方面的科目，除了數學系的同學修讀之外，也有統計系和電子計算機系的同學修讀，還有工程系的碩士研究生旁聽這門課程。

此外，我們也準備在明年爲三年級的同學開辦一門「矩陣計算理論」，希望能夠豐富同學們在矩陣方面的知識。