數學軟體專題

奇值分解式

平 斯

方陣如果以秩來區分，最簡單的要算單秩張量，代數的說法是兩個向量的張量乘積$M=uv^T$，在力學裡管它叫Dyad。至於其他的方陣就可能非常複雜，但總不離這些單秩張量的線性組合。例如隨意一組向量的正交基底$\{u_1,u_2,\cdots,u_n\}$可以用來描述所有的n階方陣

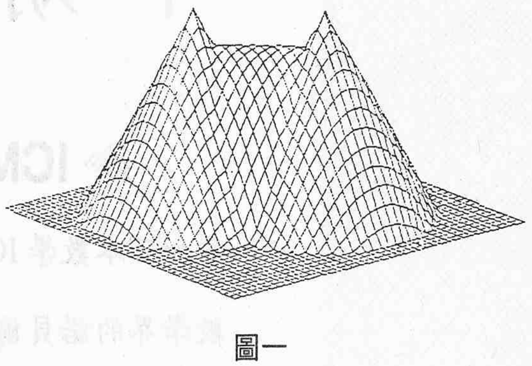
$$M=\sum a_{ij}u_iu_j^T$$

例如單位方陣可寫成$I=u_1u_1^T+\cdots+u_nu_n^T$。

這樣精簡的式子，並不是單位方陣獨有的特性，每個對稱的方陣，也可以找適當的基底把它寫成$M=\sum\lambda_iu_iu_i^T$。至於其他的任意方陣，也有個約分至最簡單的形式。只是必須找兩組，而不是一組，正交的基底$\{u_1,\cdots,u_n\}$和$\{v_1,\cdots,v_n\}$使得方陣寫成

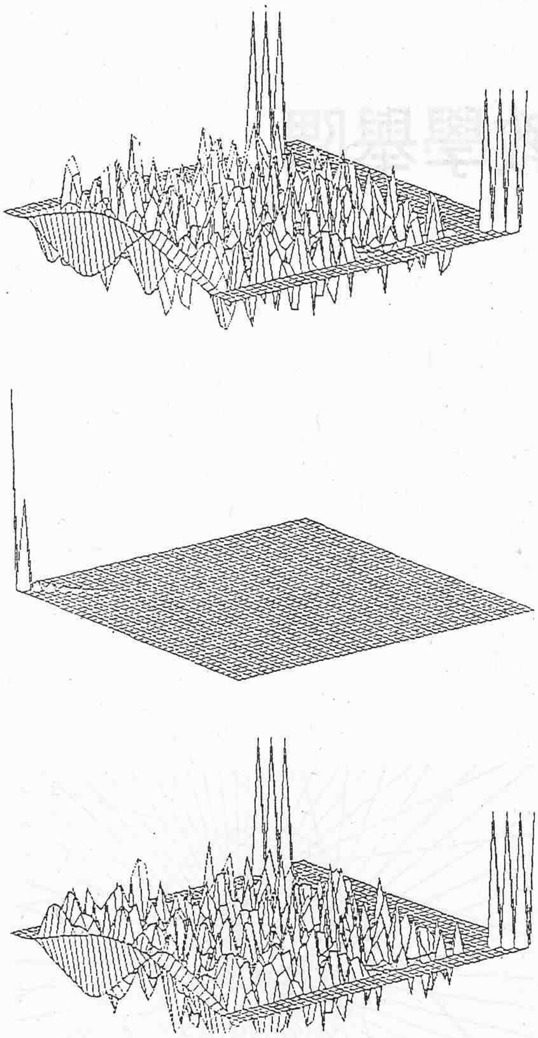
$$(*)\qquad M=\lambda_1u_1v_1^T+\cdots+\lambda_nu_nv_n^T$$

這就是一般所稱的奇值分解式 (Singular Value Decomposition，簡稱SVD)。這套教科書裡學來的理論，正如所有其他漂亮的理論一樣，都有很實際的應用。這裡要提供一個簡單的例子：



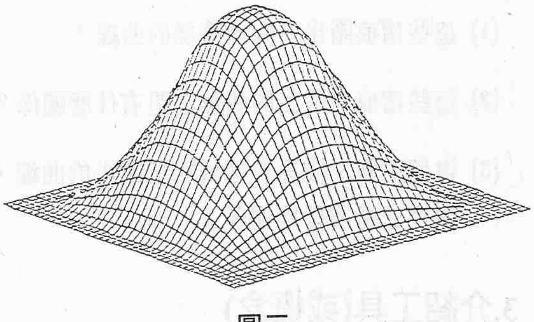
圖一

圖一是個40×40方陣的立體示意圖，在坐標(i,j)上的高度就是a_{ij}，它可分解成$M=U\wedge V^T$，這個式子其實和$(*)$是一致的，只是兩組正交基底的行向量集合起來，構成正交方陣$U=(u_1,\cdots,u_n$和$V=(v_1,\cdots,v_n)$而$\Lambda=diag(\lambda_1,\cdots,\lambda_n)$是個對角線方陣，這些值就稱爲奇值。下面依序列出這三個方陣。



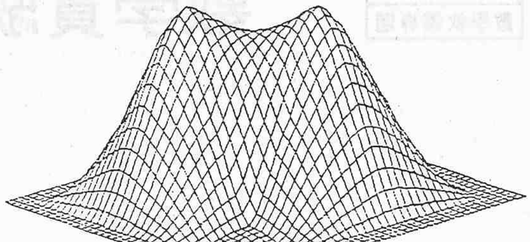
圖二

如果只取(*) 的頭一個單秩張量，圖三驗證了 Perron 定理：正項矩陣的第一個奇值和所對應的固守向量全都是正的。



圖三

再取下一個單秩張量做和，圖四顯示結果和原來的方陣已經相差無幾。然而這個方陣只用了不足200 個數據，比較起原來的40 × 40 ，只佔八分之一。



圖四

這個例子的教訓是當我們希望用較低秩的來逼近一個方陣時，SVD 能提供令人滿意的選擇。

一般數據化影像是由排列整齊的像素 (PIXEL) 所構成的。每個像素的明暗度以數據表示，構成一個方陣。由於解析度不同，這個方陣的尺碼可能是256 × 256 或1024 × 1024 不等。這樣複雜的方陣佔用了龐大的儲存空間，操作起來當然也相當耗時。上面的例子説明可以用較低秩的方陣來取代而仍不失眞，結果節省了相當的空間和時間。因此SVD 成了濃縮資料的一個有力手段。

本文中的圖全由操作**MATLAB** 獲得，這套風行一時的數學軟體，全名是矩陣實驗室，由一夥數值計算的人聯手發展出來。其前身 **Linpack** 和**Eispack** 是線性代數與固守值問題解法的副程式集。類似這種爲特殊目標而設計的還有**Quadpack** ，**Itpack** ，**Minpack** 和**Ellpack** ，分別是積分、疊代法、最佳化和橢圓偏微方程等。有一部分收集在另一個大型**IMSL** 的集子裡。然而使用**IMSL** 和**MATLAB** 兩者感覺完全不同。打個比方，前者好像是手排的陽春車，而後者像舒坦的豪華車，方便太多了。

後記：自暑假以來，**IMSL** 也推出了用於工作站的豪華版本，很快也會有個人電腦的版本出現。