淺談固有值問題$Ax = \lambda x$數值解法

林文偉

(一) 固有值問題

什麼叫做固有值問題呢? 任意給定一個複數方陣 $A(\in \mathbf{C}^{n\times n})$, 若常數 $\lambda(\in \mathbf{C})$ 及非零向量 $x(\in \mathbf{C}^n)$ 滿足 $Ax=\lambda x$, 則稱 λ 爲 A 的固有值 (eigenvalue), 而 x 稱爲對應於λ的固有向量 (eigen-vector corresponding to λ)。所以固有值問題, 即是求解 λ, 使得 $(A-\lambda I)x=0$。由於$x\neq 0$, 所以 $A-\lambda I$ 必爲奇異 (singular), 即 $\det(A-\lambda I)=0$

定義 A 的譜集合 (spectrum) $=\{\lambda \in \mathbf{C}|\det(A-\lambda I)=0\}$, 爲固有值所構成的集合。而將 $\det(A-\lambda I)$ 展開可得一個 λ 的一元 n 次多項式, 寫爲 $\det(A-\lambda I)=p(\lambda)=\prod_{i=1}^{s}(\lambda-\lambda_i)^{m_i}$其中 $\sum_{i=1}^{s} m_i = n$ $\lambda_i \neq \lambda j$, 這樣就可以把任何一個矩陣的固有值求出來。

事實上, 要如何解此多項式呢? 我們知道, 在低於五次時, 多項式才有公式解, 而高於五次時, 雖沒有公式解, 但我們可以用牛頓法來求其近似解, 但是這樣不一定行的通, 我們來看下面的一個特例。

例: 假設特徵多項式爲 $p(\lambda)=(\lambda-1)(\lambda-2)\cdots(\lambda-20)=\lambda^{20}-210\lambda^{19}+\cdots+243290\cdots0$, 多項式在存入電腦時, 會有一個捨入誤差, 所以我將此多項式寫成 $p(\lambda,\varepsilon)=p(\lambda)-210\varepsilon\cdot\lambda^{19}$, 我們只考慮在 λ^{19} 次方項的係數有些許誤差, 其他項的係數均無誤差, 我們想要探討在原先多項式中, 20 爲其中的一個固有值, 而當存入電腦後, 有了此項誤差之後, 會變成多少? 這個多項式, 其實爲 ε 的可微分方程式, 我們考慮相對誤差 $\frac{\lambda(\varepsilon)-\lambda(0)}{\lambda(0)}$, 其中 $\lambda(0)=20$, 這個相對誤差近似於 $\frac{1}{\lambda(0)}\varepsilon\frac{d\lambda(0)}{d\varepsilon}$, 因爲 $\frac{\lambda(\varepsilon)-\lambda(0)}{\lambda(0)}$ 約若等於 $\frac{d\lambda}{d\varepsilon}(0)$, 再對 ε 做微分, 得

$$\frac{dp}{d\varepsilon}=\left[\frac{dp}{d\lambda}\frac{d\lambda}{d\varepsilon}+\frac{dp}{d\varepsilon}\right]_{\lambda=\lambda(\varepsilon)}=0 \qquad (1)$$

將 (1) 式代入, 得到相對誤差

$$\begin{aligned} E &= \frac{1}{\lambda(0)}\varepsilon\cdot\left[-(\frac{dp}{d\lambda})^{-1}\frac{dp}{d\varepsilon}\right]_{\lambda=\lambda(0)} \\ &= \prod_{j=1}^{19}(20-j)^{-1}20^{18}\cdot 210\cdot\varepsilon \\ &\approx 4.5\times 10^{8}\times\varepsilon \end{aligned}$$

取 $\varepsilon=2^{-23}/210$, 推得相對誤差 $E\approx 0.26$, 原本特徵值爲 20, 但經過電腦運作, 因所用的方法不好, 結果反而得到 20.26, 相對誤差頗大, 所以牛頓法並非適用於所有情形。在這

樣的情形下, 我們將如何改善呢? 先看下面定理。

定理: (Schur Theorem)

對任何一個矩陣 A, 存在 unitary 矩陣 U, 即 $U^*U=I$, 使得 $U^*AU=R$, R 爲一上三角矩陣。

所以 $\det(A-\lambda I)=\det(URU^*-\lambda I)=\det[U(R-\lambda I)U^*]=\det(R-\lambda I)$, 因爲 R 爲上三角矩陣, 所以 A 的固有値即爲 R 矩陣的對角元素, 即 $\lambda_i=R_{ii}$, $i=1,2,\cdots,n$, 這種求固有値的方法, 理論上是很簡單, 但在證明這個定理時, 是用數學歸納法。意即在 Schur 定理中的上三角矩陣 R, 並非容易求得。但是這個定理相當重要, 現在我們來看一個例子。

例: 假設現在蓋一個農場, 養殖大小不同的豬隻。一年之後有 2/3 大型豬賣出, 1/3 留下。而一年之後, 有 3/4 中型豬變成大型豬, 3/5 小型豬變成中型豬, 而大型豬每年可生 2/3 的小型豬。現在令大型豬個數 x_1, 中型豬個數 x_2, 小型豬個數 x_3, 則整個問題可寫成:

$$\begin{bmatrix} \frac{1}{3} & \frac{3}{4} & 0 \\ 0 & \frac{1}{4} & \frac{3}{5} \\ \frac{2}{3} & 0 & \frac{2}{5} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} y_1 \\ y_2 \\ y_3 \end{bmatrix}$$

$$\text{令} \quad A=\begin{bmatrix} \frac{1}{3} & \frac{3}{4} & 0 \\ 0 & \frac{1}{4} & \frac{3}{5} \\ \frac{2}{3} & 0 & \frac{2}{5} \end{bmatrix}$$

但在農場上, 養殖大小不同的豬隻, 須有不同的設備, 所以我們當然希望每一年不要有太大的變化, 這樣才便於經營與管理。所以, 即希望得到 $Ax=\lambda x$ 其中 $\lambda=1$ 且 $x\neq 0$。這便是固有値問題, 然而如何確定矩陣 A 中有一個固有値爲 1 呢? 我們看 A 的轉置矩陣 A^T, 因 $A^T\cdot\begin{bmatrix}1\\1\\1\end{bmatrix}=1\cdot\begin{bmatrix}1\\1\\1\end{bmatrix}$ 所以可以確定 A^T 有固有値 1, 所以 A 也有固有値 1, 雖然他們分別對應的固有向量不同, 但他們共有固有値 1。而經過求固有値之後, 我們求得 A 的固有値爲 1 及 $-0.008\pm 0.577i$, 而對應於 1 的固有向量爲 $(0.575 \quad 0.511 \quad 0.639)^T$, 所以我們在養殖豬隻時, 不同體型的豬隻按照上述的比例, 則可以得到最佳的結果。

(二) 冪法 (Power Method)

接下來我們要介紹的方法, 利用乘冪來收斂至所欲求的固有値。

假設 A 爲可對角化 (diagonalizable) 矩陣, 即存在 n 個固有値, 設爲 $\lambda_1,\lambda_2,\cdots,\lambda_n$; 其相對的固有向量爲 $x_1,x_2,\ldots,x_m$, 所以 $Ax_i=\lambda_i x_i \quad i=1,2,\cdots,n$, 假設這 n 個固有値中, 有一個絕對値爲最大, 令 $|\lambda_1|>|\lambda_2|\geq\cdots\geq|\lambda_n|$, 任給一個不爲 0 的起始値 u_0, 我們可以將 u_0 寫成所有固有向量的線性組合, 即 $U_0=\sum_{i=1}^n\alpha_i x_i$, 將 A 乘以 u_0, 反覆地乘, 因爲 x_i 爲 A 的 固有向量, 所以我們得到 $A^sU_0=\sum_{i=1}^n\alpha_i\lambda_i^s x_i=\lambda_1^s\{\lambda_1x_1+\sum_{i=2}^n\alpha_i(\frac{\lambda_i}{\lambda_1})^s x_i\}$ 因此, $\lambda_1^{-s}A^sU_0=\alpha_1x_1+\sum_{i=2}^m\alpha_i(\frac{\lambda_i}{\lambda_1})^s x_i$, 又當 s 很大時, $\frac{\lambda_i}{\lambda_1}\to 0$。所以數列

$$\{\lambda_1^{-s}A^sU_0\}\to\alpha_1x_1 \qquad (2)$$

這個式子是一個方向, 指引我們如何去求固有值, 而 λ_1 是我們想求的, 但是上述式子, 並不能求得 λ_1 的值, 所以, 我們來看所謂的冪法, 是如何求取固有值呢?

假設 ℓ 是一個 linear functional, 則冪法可寫爲

$$\text{(Power Method)}\begin{cases} u_0: \text{起始值} \\ v_{s+1}=Au_s \\ k_{s+1}=\ell(v_{s+1}) \\ v_{s+1}=v_{s+1}/k_{s+1} \end{cases}$$

上述的遞迴式, 即爲求取固有值的冪法。但爲何這樣即可求到固有值呢?

定理: 假設 $\alpha_1\neq 0$, $\ell(x_1)\neq 0$, $\ell(v_s)\neq 0$, $s=1,2,\cdots,n$,

$$\text{則}\quad \lim_{s\to\infty}k_s=\lambda_1$$
$$\text{且}\quad \lim_{s\to\infty}U_s=\frac{x_1}{\ell(x_1)}$$

證明: 因爲

$$\begin{aligned} u_S &= A^sU_0/\ell(A^su_0) \\ k_S &= \ell(A^su_0)/\ell(A^{s-1}u_0) \end{aligned}$$

由第2式 $\Longrightarrow$

$$\lambda_1^{-s}\ell(A^su_0)\longrightarrow \alpha_1\ell(x_1)$$
$$\lambda_1^{-s+1}\ell(A^{s-1}u_0)\longrightarrow \alpha_1\ell(x_1)$$

所以 $\lambda_1^{-1}k_s\longrightarrow 1\Longrightarrow \lim_{s\to\infty}k_s=\lambda_1$

$$\begin{aligned} \text{又}\quad u_s &= A^su_0/\ell(A^su_0) \\ &= \alpha_1x_1+\sum_{i=2}^n\alpha_i(\frac{\lambda_i}{\lambda_1})^s x_i/\ell(\alpha_1x_1 \\ &\quad +\sum_{i=2}^n\alpha_i(\frac{\lambda_i}{\lambda_1})^s x_i) \\ &\longrightarrow \frac{\alpha_1x_1}{\alpha_1\ell(x_1)} \end{aligned}$$

由上述的證明, 我們得到冪法 (PM) 的收斂速度和 λ_2 與 λ_1 的比值有很大的關係。如果這個比值爲 0.9, $(0.9)^{20}=0.12$, 而 $(0.9)^{100}=2.6\times 10^{-5}$, 收斂的速度很差, 但若比值爲 0.5, 則 $(0.5)^{20}=9.5\times 10^{-1}$ 收斂速度較快。所以這個比值對PM 很有關係, 一但這個比值很接近 1, 收斂速度則相當慢, 此時, 該如何解決呢?

(三) 逆冪法 (The Inverse Power Method)

當 λ_2 和 λ_1 的比值很接近 1 時, 收斂速度很慢, 此時我們需要用逆冪法。此方法陳述如下, 原先固有值問題爲 $Ax=\lambda x$, 給定任一非固有值的常數 α, 則方程式改爲 $(A-\alpha I)x=(\lambda-\alpha)x$, 因此 $(A-\alpha I)^{-1}x=\frac{1}{\lambda-\alpha}x$, 則 $(A-\alpha I)^{-1}$ 的固有值爲 $\frac{1}{\lambda_1-\alpha},\frac{1}{\lambda_2-\alpha},\cdots,\frac{1}{\lambda_n-\alpha}$, 假設 $\alpha=0.99$, 而 $\lambda_1=1,\lambda_2=0.9$, 則此時 $(A-\alpha I)^{-1}$ 的固有值爲 $\frac{1}{1-0.99},\frac{1}{0.9-0.99}\cdots$, 原先 $|\frac{\lambda_2}{\lambda_1}|=0.9$, 而現在 $(A-\alpha I)^{-1}$ 的前二個固有值比爲 $\left|\frac{\lambda_1-\alpha}{\lambda_2-\alpha}\right|=\frac{1}{9}$, 可能 1,2 步就收斂了。事實上, 逆冪法非常簡單, 只是將 A 換成 $(A-\alpha I)^{-1}$ 就有意想不到的結果。

我們將逆冪法的關係式寫成:

$$\text{(IPM)}\begin{cases} u_0: \text{起始值} \\ v_{s+1}=(A-\alpha I)^{-1}u_s \\ k_{s+1}=\ell(v_{s+1}) \\ u_{s+1}=v_{s+1}/k_{s+1} \end{cases}$$

仿效冪法 (PM), 我們也可證以下的結果。

定理: 若 $|\lambda_1-\alpha|<|\lambda_i-\alpha| \quad i\neq 1$, 且 $\alpha_1\neq 0$, $\ell(x_1)\neq 0$ 且 $\ell(v_s)\neq 0$ $s=1,\cdots,n$, 則 $\lim\limits_{s\to\infty} k_s=\frac{1}{\lambda_1-\alpha}$ 且 $\lim\limits_{s\to\infty} u_s=\frac{x}{l(x_1)}$。

我們另外還有一個重要的結果, 即是逆冪法其實是二次收斂的 (quadratic convergence), 我們也可以證明逆冪法其實就是牛頓法, 而牛頓法是二次收斂, 因此可證得逆冪法亦是二次收斂。如何證明逆冪法就是牛頓法呢?

先將逆冪法做一變型, 改為

$$(\text{IPM})\begin{cases} u_0: \text{起始值} \\ v_{s+1}=(A-\alpha_s I)^{-1}u_s \\ k_{s+1}=\ell(v_{s+1}) \\ u_{s+1}=v_{s+1}/k_{s+1}, \\ \alpha_{s+1}=\alpha_s+\frac{1}{k_{s+1}} \end{cases}$$

為什麼要做這樣的變型呢? 因為如果能將 α_s 取的愈靠近固有值的話, 則收斂速度較快。

現在考慮 $Au=\lambda u$ 為一非線性方程式, 且 $\ell^T u=1$, 則整個系統可寫為:

$$\begin{cases} Au-\lambda u=0 \\ l^T u=1 \end{cases}$$

此為一非線性系統有 $n+1$ 個未知數, 而這個系統可用牛頓法求解, 將 F 寫成:

$$F=\begin{pmatrix} u \\ \lambda \end{pmatrix}=\begin{pmatrix} Au-\lambda u \\ \ell^T u-1 \end{pmatrix}=0$$

F 的 Jacobian matrix 為

$$F'\begin{pmatrix} u \\ \lambda \end{pmatrix}=\begin{pmatrix} A-\lambda I & -u \\ \ell^T & 0 \end{pmatrix}$$

所以牛頓法寫為:

$$\begin{pmatrix} u_{s+1} \\ \lambda_{s+1} \end{pmatrix}=\begin{pmatrix} u_s \\ \lambda_s \end{pmatrix}-(F')^{-1}\left(F\begin{pmatrix} u_s \\ \lambda_s \end{pmatrix}\right)$$

乘開即為

$$\begin{aligned} &\begin{pmatrix} A-\lambda_s I & -u_s \\ \ell^T & 0 \end{pmatrix}\begin{pmatrix} u_{s+1} \\ \lambda_{s+1} \end{pmatrix} \\ =&\begin{pmatrix} A-\lambda_s I & -u^s \\ \ell^T & 0 \end{pmatrix}\begin{pmatrix} u_s \\ \lambda_s \end{pmatrix} \\ &-\begin{pmatrix} Au_s-\lambda_s u_s \\ \ell^T u_s-1 \end{pmatrix} \end{aligned}$$

所以 $(A-\lambda_s I)u_{s+1}=(\lambda_{s+1}-\lambda_s)u_s$,
令 $\lambda_{s+1}-\lambda_s=\frac{1}{k_{s+1}}$, 則 $\ell^T u_{s+1}=1$, 且 $(A-\lambda_s I)v_{s+1}=u_s$, $v_{s+1}=k_{s+1}u_{s+1}$。
即為上述的 IPM (Variant), 所以, IPM (Variant) 為局部二次收斂。

(四) QR法

求 A 的固有值的終極目標, 即是 Schur Form, 即 $U^*AU=R$, 又即 $AU=UR$, 如果用計算方法來迭代, 其迭代式為:

$$\begin{cases} U_0=I \\ AU_i=U_{i+1}R_{i+1} \end{cases} \tag{3}$$

其中 U_i: unitary 矩陣
R_i: 上三角矩陣

若 U_i 收斂至 U, 則 $R_{i+1}=U_{i+1}^*AU_i$ 收斂至 U^*AU, 而迭代式的左邊為二已知矩陣相乘, 等式的右邊即為大家所熟悉的的 Gram-Schmidt Process, 這是迭代式最基本的動機, 那現在需要做一些變化, 因為上式需要太多的機體運算, 所以我們做以下的修正:

$$Q_i=U_{i-1}^*U_i \qquad A_{i+1}=U_i^*AU_i$$

觀察:

$$\begin{aligned} A_i &= U_{i-1}^*AU_{i-1}=U_{i-1}^*U_iR_i \\ &= Q_iR_i \end{aligned}$$

另外, 又

$$\begin{aligned} R_iQ_i &= R_iU_{i-1}^*U_i = U^*AU_i \\ &= A_{i+1} \quad (\text{by } (3)) \end{aligned}$$

所以, 整個 (3) 可等價於:

$$\begin{cases} A_0 = A \\ A_i = Q_iR_i \\ A_{i+1} = R_iQ_i \end{cases} \tag{4}$$

(4) 式比 (3) 式節省許多計算量, 因爲在 (4) 式中, 不需做矩陣相乘, 而且還有許多技巧。原因是將 A 分解成 QR 後, 做完結果反過來相乘, 再予以分解, 重覆此步驟, 最後 A_{i+1} 就會收斂到一上三角矩陣, 因爲如果 U_i 收斂至 U, 則 Q_i 收斂至 I, 即 A_{i+1} 收斂至 R, 此爲基本 QR 演算法 (Basic QR Algorithm), 至於爲何 U_i 會收斂, 我們看下述定理。

定理: $A \in C^{n\times n}$ 且 $|\lambda_1| > |\lambda_2| > \cdots > |\lambda_n| > 0$, 則 QR 演算法收斂。意即 A_{i+1} 收斂至上三角矩陣。

證明: 令 $P_i = Q_1\cdots Q_i$

$$\begin{aligned} A_{i+1} &= R_iQ_i = Q_i^*A_iQ_i \\ &= Q_i^*\cdots Q_i^*AQ_1\cdots Q_i = P_i^*AP_i \end{aligned}$$

(Claim) $u_s = P_s\cdot e_1$, 則 $\{u_s\}$ 爲 PM(Variant) 所 生成的數列

$$\text{PM}\begin{cases} u_0 = e_1 \\ v_{s+1} = Au_s \\ k_{s+1} = \| v_{s+1} \|_2 \\ u_{s+1} = v_{s+1}/k_{s+1} \end{cases}$$

由定義得: $P_{s+1} = P_s\cdot Q_{s+1}R_{s+1}R_{s+1}^{-1} = P_{s+1}A_{s+1}R_{s+1}^{-1} = AP_sR_{s+1}^{-1} = AP_sR_{s+1}^{-1}$

所以 $u_{s+1} = P_{s+1}\cdot e_1 = Au_sr_{s+1}$, 其中 r_{s+1} 爲 R_{s+1}^{-1} 的 $(1,1)$ 元素

令 $u_s\cdot r_{s+1} = v_{s+1}$

則 $1 = u_{s+1}^*u_{s+1} = r_{s+1}^2\cdot v_{s+1}^*v_{s+1}$

$$r_{s+1} = (v_{s+1}^*v_{s+1})^{-\frac{1}{2}} = k_{s+1}$$

所以做 QR 演算法即爲做 PM(Variant), 而前面已證 PM 會收斂, 故 QR 亦會收斂。

最後再看 $A_{s+1} = P_s^*AP_s$, 推到 $A_{s+1}\cdot e_1 = P_s^*AP_Se_1 = P_sAu_s$ 收斂至 Ax_1, 即 $(\lambda_1, 0\cdots 0)^T$, 所以, A_{s+1} 收斂至矩陣爲此型態

$$\begin{pmatrix} \lambda_1 & * & \cdots\cdots & * \\ 0 & * & \cdots\cdots & * \\ \vdots & \vdots & & \vdots \\ \vdots & \vdots & & \vdots \\ 0 & * & \cdots\cdots & * \end{pmatrix}$$

但這收斂太慢了, 須改爲:

$$\begin{cases} A_0 = A \\ A_i - \alpha_iI = Q_iR_i \\ A_{i+1} = R_iQ_i + \alpha_iI \end{cases}$$

此過程稱爲位移式的 QR 演算法, 完全仿效前述逆冪法, 所以位移式的 QR 演算法是局部二次收斂的數値方法。

—本文作者任教於清華大學應用數學研究所—