314653007 吳隆傑

1. Introduction and Problem Statement

探討的問題

如何在實數或複數點集上，高效、靈活且穩健地計算有理函數逼近，於給定實數或複數點集上的函數 $f(z)$，使其結果為一個有理函數 $r(z)$

問題的重要性

1. 提供 Compact Representations of Function：當函數 on or near the domain of approximation，或者在 unbounded domains 有極點或其他奇異點時，有理函數比多項式更有效率。
2. Extrapolation：提取函數在已知範圍之外的極點、數值或其他性質，例如序列和級數收斂加速的標準方法（eta and epsilon algorithms）就是基於有理函數逼近，有理逼近是解決解析延拓（analytic continuation）問題最有效的方法。

AAA 演算法的重要性

在有理逼近領域提供了高效、靈活性和穩健性，是作者在其他演算法中沒見過的，包含：

1. 性能優勢：在圓盤或區間上，AAA 演算法也可能贏過現有其他方法。
2. 領域靈活性：AAA 演算法的特點是不受特定逼近域（interval, a circle, a disk, or a point）的限制，能有效處理可能包含非連通區域、形狀不規則或 possibly unbounded 的離散點集。
3. 極點處理能力：被逼近的函數即使極點位於採樣點集之中，AAA 演算法也能有效運作。

2. Method Explanation

重心有理表示法 (Barycentric Rational Representation)

AAA 演算法的基礎是重心公式，其形式如下：

$$r(z) = \frac{n(z)}{d(z)} = \sum_{j=1}^{m} \frac{w_j f_j}{z - z_j} \Bigg/ \sum_{j=1}^{m} \frac{w_j}{z - z_j},$$

m是逼近的階數， z_j是支援點 (support points)，f_j是函數值(data values)，w_j是權重(weights)，如果權重w_j不為零，則$r(z)$是一個 $(m-1,m-1)$的有理插值函數，滿足$r(z_j)=f_j$的插值條件。

重心表示法的關鍵在於數值品質（numerical quality），這種表示法由$1/(z-z_j)$項的商組成，選擇良好的z_j可以使表示法具有足夠的獨立性，使得條件數良好（well-conditioned），在許多情況下比$p(z)/q(z)$的多項式商表示法，或向量擬合（vector fitting）使用的部分分式表示法條件數更優。

貪婪選擇支援點（Adaptive Greedy Selection）

演算法以迭代方式逐步逼近，在每一步都貪婪地選擇最佳支援點，避免指數級的不穩定性

1. 初始化：演算法從一個有限的採樣點集Z開始。
2. 貪婪選擇支援點 (Greedy Selection)：在第m步，新的支援點z_m會從未被選為支援點的剩餘採樣點集$Z^{(m-1)}$中選出，依據該點能使上一步的非線性殘差$|f(z)-n(z)/d(z)|$達到最大絕對值。
3. 建立線性化最小平方法問題 (Linearized Least-Squares)：

   原始目標是最小化$f(z)\approx\frac{n(z)}{d(z)}$在採樣點集上的誤差，將此非線性問題線性化為$f(z)d(z)\approx n(z)$，目標是求解權重$w=(w_1,\dots w_m)^T$，使其在排除支援點$z_j$的剩餘點集$Z^{(m)}$上，最小化$|fd-n|_{Z^{(m)}}$（殘差的離散二範數），同時滿足權重向量的正規化條件$\|w\|_m=1$
4. 矩陣形式（Loewner 矩陣）：上述最小化問題被轉化為一個線性代數問題：$\min\left\|A^{(m)}w\right\|_{M-m}$。其中$A^{(m)}$是一個Loewner 矩陣，元素結構為$\frac{F_i^{(m)}-f_j}{Z_i^{(m)}-z_j}$，而且可以用 Cauchy 矩陣$C$構建，表達為$A^{(m)}=S_FC-CS_f$
5. 奇異值分解求解 (SVD Solution)：通過計算$A^{(m)}$矩陣的奇異值分解 (SVD)，權重向量w被設定為與最小奇異值對應的最終右奇異向量。
6. 迭代與終止：演算法持續進行，在非線性殘差降至預設容忍度（10^{-13}）以下終止，線性化殘差的範數$\sigma_{min}(A^{(m)})$ 是不遞增的函數。

3. Simple Implementation / Experiment

實驗設定與步驟

假設要逼近的函數是$f(z) = 1/(z+2)$，在$z = -2$有極點的有理函數。

1. 採樣點和函數值的定義(M, Z, F)

   在複數平面上定義$M = 10$個採樣點，假設在$[-1,1]$區間的實數上取 10 個點$z_1 \sim z_{10}$。
2. 選擇支援點(m, z, f)

   選擇$m = 3$個點作為支援點z_j，對應的函數值f_j，建立一個為$(m-1, m-1) = (2,2)$的有理逼近式。
3. 定義剩餘點集$(Z^{(m)}, F^{(m)}, J)$

   AAA 演算法通過在非支援點集$Z^{(m)}$上最小化線性殘差來求解權重w。

   $Z^{(m)}$：從Z中移除z後剩餘$M - m = 7$的個點

   $F^{(m)}$：函數在$Z^{(m)}$上的值

   J：用於索引$Z^{(m)}$點的索引集
4. 構建用於 SVD 求解的 Loewner 矩陣$A^{(m)}$

   AAA 演算法的目標是最小化線性化殘差$\min\|fd - n\|_{Z^{(m)}}$

   轉化為矩陣問題：$\min\left\|A^{(m)}w\right\|_{M-m}$

   i.構建 Cauchy 矩陣C：一個$(M-m) * m$矩陣，元素定義了 AAA 逼近使用的部分分式基底$1/(z - z_j)$。

   $$C_{i,j} = \frac{1}{Z_i^{(m)} - z_j}$$

   ii.縮放矩陣S_F和S_f：

   S_F是$(M-m) * (M-m)$對角矩陣，對角線元素為$F^{(m)}$的值。

   S_f是$m * m$對角矩陣，對角線元素為f(支援點的函數值) 的值。

   iii.構建 Loewner 矩陣$A^{(m)}$：$A^{(m)} = S_F C - C S_f$

   $$A_{i,j}^{(m)} = \frac{F_i^{(m)} - f_j}{Z_i^{(m)} - z_j}$$
5. 求權重向量w

   AAA 演算法通過對$A^{(m)}$進行奇異值分解(SVD)來求解權重w

$$A^{(m)} = U\Sigma V^*$$

權重向量w為對應最小奇異值的最終右奇異向量$V(:,m)$，同時滿足$\|w\|_m = 1$的正規化條件。

6. 計算有理函數$r(z)$的分子$n(z)$和分母$d(z)$，當權重w被確定，可以在非支援點$Z^{(m)}$上計算分子和分母的值N和D：$N = C \cdot (w \odot f), D = C \cdot w$
$(w \odot f)$為元素級的乘積，有理逼近結果R在這些點上就是$R\left(Z^{(m)}\right) = N/D$

4. Summary or conclusion

AAA 演算法的核心優勢在於速度、靈活性和穩健性，通過結合兩種關鍵思想，重心表示法和貪婪支援點選擇來實現。

核心優勢 (AAA Strengths)

1. 數值穩健性與良好條件數：採用重心表示法（barycentric form）更優良的條件數（well-conditioned)，避開浮點數運算中常見的指數不穩定性。
2. 抗虛假極點能力（Froissart Doublets）：演算法的設計讓結果通常極少或沒有數值虛假極點，如果出現虛假極點，也可以通過最小平方法步驟（cleanup）移除。
3. 普適性與靈活性：AAA 演算法的特性在它不受特定逼近域的限制 (interval, a circle, a disk, or a point）的限制，能有效處理可能包含非連通區域、形狀不規則或 possibly unbounded 的離散點集。
4. 處理奇異點的能力：演算法能夠有效運作，即使被逼近的函數的極點位於採樣點集之中。
5. 高效且實用：演算法速度快且具備靈活性，不用使用者輸入參數，核心複雜度為$O(Mm^3)$次浮點運算，在大多數應用中因為逼近次數m較小，計算量是適中的。
6. 收斂特性：演算法有內在的穩定性，因為線性化殘差範數$\sigma_{min}(A^{(m)})$ 是一個不遞增的函數。

有效條件與假設

1. 目標函數特性：AAA 在需要 Compact Representations 或 Extrapolation 的應用中表現出色，像當函數具有極點或其他奇異點時，對於單位圓盤上的解析函數，有理逼近比多項式逼近更有效。

2. 數據充足：演算法需要一個有限的採樣點集Z，並假設函數$f(z)$在Z上有定義，每次迭代中，用於最小平方法的剩餘點集$Z^{(m)}$必須至少包含m個點，即$m \leq M/2$。
3. 終止條件：演算法預設在相對誤差至10^{-13}時終止。

限制與潛在失敗模式

儘管 AAA 演算法具有高穩健性，但還是存在一些限制和可能導致失敗或計算成本提高的情況：

1. 數學與數值限制 :

   i.非最優性：AAA 演算法不尋求在任何特定範數（如L^2或L^∞均勻範數）下實現最優逼近，對於需要達到L^∞最優（Minimax）的逼近問題，需要使用更專業的方法或 AAA 的變體。

   ii.收斂過度：如果收斂容忍度設置得過低，演算法會產生並積累大量的數值虛假極點。

   iii.大型問題的計算成本：演算法的複雜度為$O(Mm^3)$，對於採樣點數M很大，或者逼近階數m必須很高才能收斂的問題計算可能變慢。

2. 數據與函數依賴限制:

   i.對稱性破壞：核心 AAA 演算法通常會破壞函數的固有對稱性，可能會導致在實數區間逼近實函數時產生不希望出現的極點。

   ii.域內外一致性問題：對於有極點的函數，即使在閉合曲線上實現了高精度逼近，也不能保證結果在曲線包圍的整個區域內精確，除非逼近式具備足夠的極點數量來捕捉原函數的極點。

可能的改進或擴展建議

作者提出了多種 AAA 演算法的變體或擴展，以解決上述限制並擴大應用範圍

1. 解決數值和收斂問題

   i. L^∞最優逼近（Minimax）：建議使用迭代重加權（iterative reweighting*的非插值（non-interpolatory）AAA 變體，這是 Lawson 演算法的基礎，能夠將最大誤差降至更接近 Minimax 值。

   ii.對稱性強制：對於偶函數或奇函數的逼近，可以考慮將變數從z轉換為$z^{1/2}$，對於一般的對稱性，可以修改演算法，成對選擇支援點來保持對稱性。

   iii.處理大小數據對稱性：為了在 Riemann 球面上處理$f(z)$值非常大或為∞

的點，可以修改線性最小平方法問題，在$|f(z)| > 1$的點處，將矩陣A對應的行除以$f(z)$，以保持對大值和小值的對稱性。

2. 擴展逼近類型和範疇

i.非$(m-1, m-1)$類型逼近：考慮類型為(μ, ν)且$\mu \neq \nu$的有理逼近，需要限制權重向量$\{w_j\}$位於特定子空間以滿足分子和分母的次數限制。

ii.非插值逼近：引入獨立的參數$\{\alpha_j\}$和$\{\beta_j\}$來控制分子和分母，使有理逼近式不必在支援點處插值數據，這在與迭代重加權結合以實現最優逼近時特別有用。

iii.連續採樣點集：將 AAA 演算法直接應用於連續集合Z，此時 Loewner 和 Cauchy 矩陣變成準矩陣（quasimatrices），需要在每一步解決連續最佳化問題。

iv.附帶導數條件 (Hermite-AAA)：透過修改 Cauchy 矩陣C的列並增加 Loewner 矩陣$A^{(m)}$的行，可以將逼近擴展到匹配導數數據。

v.向量和矩陣逼近：將此方法擴展到處理向量或矩陣函數f和變數w的情境