

数値解析第12回 逆行列と計算量

1. 部分ピボット選択法

右図は、行列の階段化が第 $k-1$ 段まで完成した状態を示す。第 k 段を作るときのピボット行番号を p ($k \leq p \leq n$) とすると、 k 行は p 行と交換され、消去に進む。部分ピボット選択法は、その第 k 成分の絶対値が最大の行をピボット行に選ぶ。

$$\begin{pmatrix} * & \cdots & \cdots & \cdots & \cdots & * \\ 0 & \ddots & & & & \vdots \\ \vdots & \ddots & * & \cdots & \cdots & * \\ 0 & \cdots & 0 & a_{kk} & \cdots & a_{kn} \\ \vdots & & \vdots & \vdots & & \vdots \\ 0 & \cdots & 0 & a_{ik} & \cdots & a_{in} \\ \vdots & & \vdots & \vdots & & \vdots \\ 0 & \cdots & 0 & a_{nk} & \cdots & a_{nn} \end{pmatrix}$$

行交換前は $|a_{pk}| = \max_{k \leq i \leq n} |a_{ik}|$ 。 k 行は p 行との行交換後では $|a_{kk}| \geq |a_{ik}|$ ($k+1 \leq i \leq n$) が成立する。よって、消去においてピボット行の $l_{ik} = a_{ik} / a_{kk}$ 倍が第 i 行が引かれるが、 $|l_{ik}| \leq 1$ ゆえ、誤差の伝播が小さい。この方法の効果は、次の2点である。

- (1) 零でないピボット要素 a_{kk} を与える。
- (2) $|a_{kk}|$ を大きくし、誤差の伝播を押さえる。

2. 計算量の考え方

数値解析において、「 $c = x^T A y$ を計算せよ」とは、メモリ上に行列 A 、ベクトル x, y のデータがあるとき、スカラー量 c を計算してメモリに格納せよという問題である。

$c = x^T (Ay)$ と積の順序を決めると、中間結果を $z = Ay$ とし、

$$(1) \text{ 線形変換: } z = Ay \quad : n^2 \text{ flops}$$

$$(2) \text{ 内 積: } c = x^T z \quad : n \text{ flops}$$

の順に計算が進行する。これが c を計算するアルゴリズムで、既存のアルゴリズム「線形変換」と「内積」を用いている。計算量は合計 $n^2 + n$ flops である。

実際の計算ではベクトル z とスカラー c を格納するメモリ領域の確保を行うが、自明なので明示しない。

3. 逆行列の計算

$n \times n$ の逆行列 $B = A^{-1}$ を計算する。 $B = (b_1, b_2, \dots, b_n)$ とすると、 $AB = I$ より、

$$AB = A(b_1, b_2, \dots, b_n) = (Ab_1, Ab_2, \dots, Ab_n) = (e_1, e_2, \dots, e_n) = I.$$

$$\therefore Ab_i = e_i = (0, \dots, 0, \overset{i}{1}, 0, \dots, 0)^T \quad (1 \leq i \leq n). \quad (1)$$

式(1)は b_i に関する n 本の方程式で、係数行列 A が共通。ゆえに既存のアルゴリズム「LU分解」と「LU求解」を用いて、次の計算法が得られる。

<逆行列計算法> 計算量 $n^3 + O(n^2)$ flops.

$$(1) \text{ LU分解: } A = P^{-1}LU \quad : \frac{1}{3}n^3 + O(n^2) \text{ flops}$$

$$(2) \text{ LU求解: } b_i = A^{-1}e_i \quad (1 \leq i \leq n) \quad : \frac{2}{3}n^3 + O(n^2) \text{ flops (注)}$$

(注) n 本の方程式を全て解く計算量。素朴には $n \times (n^2 + O(n)) = n^3 + O(n^2)$ flops だが、式(1)右辺の e_i の要素のほとんどが0であることを利用し、計算量を減らしている。具体的には、LU求解の前進代入を第1行からではなく、第 i 行から始める。

4. 使うべからず「逆行列法」

方程式 $Ax = b$ の解は $x = A^{-1}b$ ゆえ、アルゴリズム「逆行列」と「線形変換」により、次の解法が考えられる。

<逆行列法> 計算量 $n^3 + O(n^2)$ flops .

$$(1) \text{ 逆行列} \quad : B = A^{-1} \quad : n^3 + O(n^2) \text{ flops}$$

$$(2) \text{ 線形変換} \quad : x = Bb \quad : n^2 \text{ flops}$$

この逆行列法は、LU分解法と比べて次の3つの点で劣っている。

- (1) 計算量が多い(3倍) … 遅い
- (2) 使用メモリ量が多い … 高い
- (3) 精度が悪い … まずい

(2)は、行列の非零要素の配置と関係する。例えば、 $n \times n$ 行列のLU分解と逆行列

$$A = \begin{pmatrix} 1 & 1 & \ddots & O \\ 1 & 2 & \ddots & \\ & \ddots & \ddots & 1 \\ O & & 1 & 2 \end{pmatrix} = LU = \begin{pmatrix} 1 & & & O \\ 1 & 1 & & \\ & \ddots & \ddots & \\ O & & 1 & 1 \end{pmatrix} \begin{pmatrix} 1 & 1 & \ddots & O \\ & 1 & \ddots & \\ & & \ddots & 1 \\ O & & & 1 \end{pmatrix}, \quad A^{-1} = \begin{pmatrix} n & n-1 & \cdots & 1 \\ n-1 & n-1 & \cdots & 1 \\ \vdots & \vdots & \ddots & \vdots \\ 1 & 1 & \cdots & 1 \end{pmatrix}$$

を見る。 A, L, U は、非零要素のみ記憶すればよい。 L の対角要素は1に固定だから記憶しない。すると、メモリ量は A, L, U でそれぞれ $3n-2, n-1, 2n-1$ である。しかし A^{-1} は全ての要素が非零なので、それを記憶するには n^2 のメモリ量を必要とする。

(3)は(1)に起因する。計算量が多いと丸め誤差の蓄積が大きい。

5. 係数行列の構造と計算量

[例1] $A, B, C \in \mathbb{R}^{n \times n}, a, b \in \mathbb{R}^n$ とし、方程式 $\begin{pmatrix} A & O \\ C & B \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} a \\ b \end{pmatrix}$ を考える。

係数行列のサイズは $2n \times 2n$ ゆえ、構造を無視して解くと計算量は約 $\frac{1}{3}(2n)^3 = \frac{8}{3}n^3$ flops。ブロック成分ご

とに $Ax = a, Cx + By = b$ とすると、 $By = b' = b - Cx$ ゆえ、次の算法を得る。

$$(1) A, B \text{ をLU分解} \quad : 2 \times \frac{1}{3}n^3 + O(n^2) = \frac{2}{3}n^3 + O(n^2) \text{ flops}$$

$$(2) x = A^{-1}a \text{ をLU求解} \quad : n^2 \text{ flops}$$

$$(3) b' = b - Cx \text{ の計算} \quad : n^2 \text{ flops}$$

$$(4) y = B^{-1}b' \text{ をLU求解} \quad : n^2 \text{ flops}$$

合計 $\frac{2}{3}n^3 + O(n^2)$ flops で、素朴な計算法の計算量の約 $\frac{1}{4}$ である。階段は1段ずつ。 //

練習問題

$A, B \in \mathbb{R}^{n \times n}, a, b \in \mathbb{R}^n$ のとき、方程式 $\begin{pmatrix} A & B \\ O & A^2 \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} a \\ b \end{pmatrix}$ の効率的な解法を考えよ。

第12回練習問題

$A, B \in \mathbb{R}^{n \times n}, \mathbf{a}, \mathbf{b} \in \mathbb{R}^n$ のとき, 方程式 $\begin{pmatrix} A & B \\ O & A^2 \end{pmatrix} \begin{pmatrix} \mathbf{x} \\ \mathbf{y} \end{pmatrix} = \begin{pmatrix} \mathbf{a} \\ \mathbf{b} \end{pmatrix}$ の効率的な解法を考えよ.

練習問題解答

ブロック成分ごとに $A\mathbf{x} + B\mathbf{y} = \mathbf{a}, A^2\mathbf{y} = \mathbf{b}$. これより, $\mathbf{y} = A^{-2}\mathbf{b} = A^{-1}(A^{-1}\mathbf{b}), A\mathbf{x} = \mathbf{a}' = \mathbf{a} - B\mathbf{y}$ であるから, 次のアルゴリズムを得る.

(1) A をLU分解: $\frac{1}{3}n^3 + O(n^2)$ flops

(2) $\mathbf{z} = A^{-1}\mathbf{b}$ をLU求解: n^2 flops

(3) $\mathbf{y} = A^{-1}\mathbf{z}$ をLU求解: n^2 flops

(4) $\mathbf{a}' = \mathbf{a} - B\mathbf{y}$ を計算: n^2 flops

(5) $\mathbf{x} = A^{-1}\mathbf{a}'$ をLU求解: n^2 flops

計算量は合計 $\frac{1}{3}n^3 + O(n^2)$ flops. 行列の構造を考えない計算法の計算量の $\frac{1}{8}$. 必ず計算量を書く.