数値解析第12回 逆行列と計算量

1．部分ピボット選択法

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

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 $p\,(k \le p \le n)$ 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．

行交換前は $|a_{pk}| = \max_{k\le i\le n} |a_{ik}|$． k 行は p 行との行交換後では $|a_{kk}| \ge |a_{ik}|\ (k+1 \le i \le n)$ 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， $|l_{ik}| \le 1$ 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，「 $c = \boldsymbol{x}^T \overset{n\times n}{A}\, \boldsymbol{y}$ を計算せよ」とは，メモリ上に行列 A ，ベクトル $\boldsymbol{x}$, $\boldsymbol{y}$ のデータがあるとき，スカラ量 c を計算してメモリに格納せよという問題である．

$c = \boldsymbol{x}^T(A\boldsymbol{y})$ と積の順序を決めると，中間結果を $\boldsymbol{z} = A\boldsymbol{y}$ として，

(1) 線形変換： $\boldsymbol{z} = A\boldsymbol{y}$ ： n^2 flops

(2) 内　　積： $c = \boldsymbol{x}^T \boldsymbol{z}$ ： n 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 $\boldsymbol{z}$ とスカラ 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

$\overset{n\times n}{A}$ の逆行列 $B = A^{-1}$ を計算する． $B = (\boldsymbol{b}_1, \boldsymbol{b}_2, \cdots, \boldsymbol{b}_n)$ とすると， $AB = I$ より，

$$AB = A(\boldsymbol{b}_1, \boldsymbol{b}_2, \cdots, \boldsymbol{b}_n) = (A\boldsymbol{b}_1, A\boldsymbol{b}_2, \cdots, A\boldsymbol{b}_n) = (\boldsymbol{e}_1, \boldsymbol{e}_2, \cdots, \boldsymbol{e}_n) = I\ .$$

$$\therefore\ A\boldsymbol{b}_i = \boldsymbol{e}_i = (0, \cdots, 0, \overset{i}{\check{1}}, 0, \cdots, 0)^T \quad (1 \le i \le n)\ . \tag{1}$$

式(1)は $\boldsymbol{b}_i$ 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) LU分解： $A = P^{-1}LU$ ： $\frac{1}{3}n^3 + O(n^2)$ flops

(2) LU求解： $\boldsymbol{b}_i = A^{-1}\boldsymbol{e}_i\ \ (1 \le i \le n)$ ： $\frac{2}{3}n^3 + O(n^2)$ 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 $\boldsymbol{e}_i$ 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 $A\boldsymbol{x}=\boldsymbol{b}$ の解は $\boldsymbol{x}=A^{-1}\boldsymbol{b}$ 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) 逆行列　：$B=A^{-1}$　：$n^3+O(n^2)$ flops

(2) 線形変換　：$\boldsymbol{x}=B\boldsymbol{b}$　：n^2 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

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

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］ $A,B,C\in\mathbb{R}^{n\times n}, \boldsymbol{a},\boldsymbol{b}\in\mathbb{R}^n$ とし，方程式 $\begin{pmatrix}A & O\\ C & B\end{pmatrix}\begin{pmatrix}\boldsymbol{x}\\ \boldsymbol{y}\end{pmatrix}=\begin{pmatrix}\boldsymbol{a}\\ \boldsymbol{b}\end{pmatrix}$ 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 $A\boldsymbol{x}=\boldsymbol{a}$, $C\boldsymbol{x}+B\boldsymbol{y}=\boldsymbol{b}$ とすると，$B\boldsymbol{y}=\boldsymbol{b}'=\boldsymbol{b}-C\boldsymbol{x}$ ゆえ，次の算法を得る．

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

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

(3) $\boldsymbol{b}'=\boldsymbol{b}-C\boldsymbol{x}$ の計算　：n^2 flops

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

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