MA580H Matrix Computations

Lecture 10: Stability Analysis of Gaussian Elimination

Rafikul Alam
Department of Mathematics
IIT Guwahati

Outline

- Stability analysis of GEPP/GECP
- Accuracy of computed solutions

Backward stability

An algorithm is a function $\mathrm{ALG} : (X, \|\cdot\|) \longrightarrow (Y, \|\cdot\|)$ such that

- computation of $\mathrm{ALG}(\text{input})$ involves only a finite number of steps
- and each step performs a finite number of elementary arithmetic operations.

Let $S(d)$ be a solution of a problem with given data d and $\mathrm{ALG}(d)$ be the computed solution. Then the accuracy of the computed solution $\mathrm{ALG}(d)$ is measured by the (relative) error

$$\text{Error} \;=\; \frac{\|\mathrm{ALG}(d) - S(d)\|}{\|S(d)\|}$$

Definition: An algorithm ALG is said to be backward stable (stable) if

- $\mathrm{ALG}(d) = S(d + \Delta d)$ for some $\Delta d \in X$ such that $\dfrac{\|\Delta d\|}{\|d\|} = \mathcal{O}(\mathbf{u})$.

The quantity $\dfrac{\|\Delta d\|}{\|d\|}$ is called the backward error.

Examples

Example 1: Consider $Ax = b$. Then $x = S(A, b) = A^{-1}b$. Let $\hat{x} = \mathrm{ALG}(A, b)$.Then

ALG stable $\Longrightarrow \hat{x} = \mathrm{ALG}(A, b) = S(A + \Delta A, b + \Delta b)$, that is, $(A + \Delta A)\hat{x} = b + \Delta b$ such that $\dfrac{\|\Delta A\|}{\|A\|} = \mathcal{O}(\mathbf{u})$ and $\dfrac{\|\Delta b\|}{\|b\|} = \mathcal{O}(\mathbf{u})$.

Example 2: Consider the LU decomposition $A = LU$. Let $[L, U] = \mathrm{ALG}(A)$. Then ALG stable $\Longrightarrow A + \Delta A = LU$ for some ΔA such that $\|\Delta A\|/\|A\| = \mathcal{O}(\mathbf{u})$.

Example 3: Suppose $\mathrm{ALG}(d)$ computes $f(d) = e^d$ for $d \in \mathbb{R}$. Then ALG is stable if $\mathrm{ALG}(d) = f(d + \Delta d) = e^{d+\Delta d}$ and $|\Delta d|/|d| = \mathcal{O}(\mathbf{u})$.

Accuracy

Backward stability of ALG guarantees

$$\mathrm{ALG}(d) = S(d + \Delta d) \text{ and } \|\Delta d\|/\|d\| = \mathcal{O}(\mathbf{u}).$$

What can be said about the error in the solution?

$$\begin{aligned} \text{Error} &= \frac{\|\mathrm{ALG}(d) - S(d)\|}{\|S(d)\|} = \frac{\|S(d + \Delta d) - S(d)\|}{\|S(d)\|} \\ &\leq \kappa_S(d)\frac{\|\Delta d\|}{\|d\|}. \end{aligned}$$

- The quantity $\kappa_S(d)$ is called the condition number of S at d and measures the sensitivity of S at d.
- The algorithm ALG has no control on $\kappa_S(d)$.

Ill-conditioning

- For small relative changes in d we have

$$\frac{\|S(d+\Delta d) - S(d)\|}{\|S(d)\|} \lesssim \kappa_S(d)\frac{\|\Delta d\|}{\|d\|}$$

$$\begin{pmatrix} \text{Error in} \\ \text{solution} \end{pmatrix} \lesssim \text{cond.} \times \begin{pmatrix} \text{Error in} \\ \text{data} \end{pmatrix}$$

- Thus $S(d)$ is ill-conditioned if $\kappa_S(d) \gg 1.$ Otherwise, the problem is well-conditioned.

- How large $\kappa_S(d)$ is large enough? The answer depends on how choosy you are!

- If $\kappa_S(d) = 10^s$ then s digits may be lost in the solution computed by a stable algorithm.

Estimating the condition number

If S is differentiable at d then

$$\kappa_S(d) \simeq \frac{\|J_S(d)\| \, \|d\|}{\|S(d)\|},$$

where $J_S(d) = \left[\frac{\partial S_i}{\partial x_j}(d)\right]$ is the Jacobian of S at d.

Example: Consider $S(d) = \sqrt{d}$. Then $J_S(d) = S'(d) = 1/(2\sqrt{d})$, for $d \neq 0$ and $\mathrm{cond}_S(d) = 1/2$. ■

Example: Consider $S(d_1, d_2) = d_1 - d_2$. Then $J_S(d) = [1, -1]$ and

$$\kappa_S(d) = \frac{2\|d\|_\infty}{|d_1 - d_2|}.$$

For $d_1 := 1$, and $d_2 := 1 - 10^{-5}$, $\kappa_S(d) = 2 \times 10^5$. ■

Wilkinson's result (1961)

Theorem: Suppose we solve $Ax = b$ using GEPP in floating point arithmetic with unit roundoff $\mathbf{u}$. Let $\hat{x}$ be the computed solution. Then

$$(A + \Delta A)\hat{x} = b \text{ and } \frac{\|\Delta A\|_\infty}{\|A\|_\infty} \leq 2n^3 g_{\mathrm{pp}}(A)\mathbf{u}$$

where $g_{\mathrm{pp}}(A)$ is the pivot growth given by

$$g_{\mathrm{pp}}(A) := \frac{\max_{ij} |U(i,j)|}{\max_{ij} |A(i,j)|} = \frac{\|U\|_{\max}}{\|A\|_{\max}}$$

Thus, $\|x - \hat{x}\|_\infty / \|x\|_\infty \lesssim 2n^3 g_{\mathrm{pp}}(A)\mathrm{cond}_\infty(A)\mathbf{u}$.

- Elegant way of accounting for rounding errors. Bounds backward error rather than the error.
- Draws attention to pivot growth factor g_{pp}.
- Both $g_{\mathrm{pp}}(A)$ and $\mathrm{cond}_\infty(A)$ are easy to compute after getting L and U, costing just an extra $\mathcal{O}(n^2)$ flops.

Growth factor for GEPP

What do we know about $g_{\mathrm{pp}}(A)$?

Wilkinson (1954) proved that $g_{\mathrm{pp}}(A) \leq 2^{n-1}$. Usually $g_{\mathrm{pp}}(A) \simeq 1$ in practice. But examples exists for which $g_{\mathrm{pp}}(A) = 2^{n-1}$.

Wilkinson's matrix: 5×5 Wilkinson's matrix W is given by

$$\begin{bmatrix} 1 & 0 & 0 & 0 & 1 \\ -1 & 1 & 0 & 0 & 1 \\ -1 & -1 & 1 & 0 & 1 \\ -1 & -1 & -1 & 1 & 1 \\ -1 & -1 & -1 & -1 & 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ -1 & 1 & 0 & 0 & 0 \\ -1 & -1 & 1 & 0 & 0 \\ -1 & -1 & -1 & 1 & 0 \\ -1 & -1 & -1 & -1 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 & 2 \\ 0 & 0 & 1 & 0 & 2^2 \\ 0 & 0 & 0 & 1 & 2^3 \\ 0 & 0 & 0 & 0 & 2^4 \end{bmatrix}.$$

Note that $g_{\mathrm{pp}}(W) = 2^4$.

For an $n \times n$ Wilkinson matrix W, we have $W = LU$ with $U(n,n) = 2^{n-1}$. Hence $g_{\mathrm{pp}}(W) = 2^{n-1}$. The matrix W can be generated in MATLAB as follows

```
W = tril( 2*eye(n)-ones(n) ); W(:, n) = ones(n,1);
```

Growth factor for GEPP

An $n \times n$ matrix A is said to be diagonally dominant if $|a_{ii}| \geq \sum_{j=1, j\neq i}^{n} |a_{ij}|$ for $i = 1 : n$.

An $n \times n$ matrix A is said to be banded with bandwidth ℓ if $a_{ij} = 0$ for all $|i - j| > \ell$. For example, if $\ell = 1$ then A is tridiagonal and if $\ell = 2$ then A is pentadiagonal.

An $n \times n$ matrix A is said to be Hessenberg (i.e., upper Hessenberg form) if $a_{ij} = 0$ for $i > j + 1$.

Special matrices:

Matrix	$g_{\rm pp}(A)$
diag. dom	2
tridiagonal	2
banded (bandwidth p)	$2^{2p-1} - (p-1)2^{p-2}$
Hessenberg	n
SPD	1

Growth factor for GECP

- Wilkinson (1961) proved

$$g_{\mathrm{cp}}(A) \leq n^{1/2}(2.3^{1/2} \cdots n^{1/2})^{1/2} \sim c n^{1/2} n^{\frac{1}{4}\log n}.$$

- Usually, in practice, $g_{\mathrm{cp}}(A) \sim 1$. Determining the largest possible value of $g_{\mathrm{cp}}(A)$ is still an open problem.

Remark: There is no correlation between pivot growth of A and the condition number of A, that is, no correlation between $\mathrm{PG}(A)$ and $\mathrm{cond}(A)$. This is illustrated by Golub matrix.

```
function A = golub(n)
s = 10;
L = tril(round(s*randn(n)),-1)+eye(n);
U = triu(round(s*randn(n)),1)+eye(n);
A = L*U;
```

Golub matrix

`A = golub(10)` gives

$$\begin{bmatrix} 1 & -21 & 29 & -4 & 0 & 5 & -3 & -13 & -14 & -2 \\ 18 & -377 & 530 & -80 & -3 & 90 & -62 & -257 & -247 & -38 \\ -23 & 490 & -610 & 20 & -39 & -115 & 2 & 124 & 349 & 29 \\ 9 & -190 & 269 & -283 & -288 & 37 & -170 & -315 & -262 & -64 \\ 3 & -56 & 148 & -177 & -23 & 257 & -828 & -353 & 46 & -34 \\ -13 & 271 & -383 & -78 & -216 & -176 & 298 & 122 & 60 & 8 \\ -4 & 83 & -117 & -85 & -134 & -72 & -39 & -63 & -117 & -62 \\ 3 & -48 & 204 & -92 & 39 & 143 & -189 & -314 & 247 & -89 \\ 36 & -742 & 1159 & -290 & -127 & 176 & 267 & -747 & -358 & -291 \\ 28 & -574 & 916 & -113 & 164 & 397 & -289 & -552 & -333 & 414 \end{bmatrix}$$

For $n = 10$, we have $g_{pp}(A) = 1$ and $\mathrm{cond}_\infty(A) = 2.9219 \times 10^{18}$. For Wilkinson matrix with $n = 50$, we have $g_{pp}(A) = 2^{49} = 5.6295 \times 10^{14}$ and $\mathrm{cond}(A) = 22.306$.

Remark: Pivot growth for Cholesky factorization is 1. Hence the algorithm is backward stable.