Numerical singularity (vs. exact singularity) 23 Oct '23

During numerical factorization in a floating-point system,

we usually cannot expect <u>exact</u> zeros.

But a matrix is "near singular" if all elements below the diagonal in column k (and $\hat{a}_{kk}$)

at the k^{th} stage have magnitude $\lesssim$ eps $\max\limits_{1\le j<k} |u_{jj}|$.

We say the matrix has <u>numerical singularity</u>. commonly used heuristic

Complexity of Gaussian Elimination

We will count multiplication/addition pairs; i.e., $mx + b$.
$\quad\quad\quad\quad\quad\quad\quad\quad\quad\quad\quad\quad\quad\quad\quad\quad\quad\quad\quad\quad\quad$ ↑mult ↑add

Each is a <u>flop</u> (floating-point operation).

Comparisons and divisions are also there, but they are relatively cheap ($O(n)$ per stage); so we just count flops.

Computing the LU-factorization

<u>1^{st} stage</u>

$$\begin{bmatrix} \square & \overbrace{* \quad * \quad \cdots \quad *}^{n-1} \\ 0 & * \quad * \quad \cdots \quad * \\ 0 & * \quad * \quad \cdots \quad * \\ \vdots & \vdots\; n-1 \\ 0 & * \quad * \quad \cdots \quad * \end{bmatrix}$$

$*$s are elements inducing flops

Adding a multiple of row ① to rows ②, ..., ⓝ costs $(n-1)^2$ flops.

<u>2^{nd} stage</u> $(n-2)^2$ flops

$\vdots$

<u>$(n-1)$ stage</u> 1 flop

$$\text{Total: } (n-1)^2 + (n-2)^2 + \ldots + 1 = \sum_{i=1}^{n-1} (n-i)^2$$
$$= \frac{(n-1)n(2(n-1)+1)}{6}$$
$$= \frac{(n-1)n(2n-1)}{6}$$
$$= \frac{n^3}{3} + O(n^2) \text{ flops.}$$

↑ this constant matters in practice

Computing forward solve ($Ld = b$)

Structure:

$$\begin{bmatrix} 1 & & & 0 \\ & 1 & & \\ & & \ddots & \\ & & & 1 \end{bmatrix} \begin{bmatrix} d_1 \\ \vdots \\ d_n \end{bmatrix} = \begin{bmatrix} b_1 \\ \vdots \\ b_n \end{bmatrix} \Rightarrow \begin{cases} d_1 = b_1 & 0 \text{ flops} \\ d_2 = b_2 - l_{21} d_1 & 1 \text{ flop} \\ d_3 = b_3 - l_{32} d_2 - l_{31} d_1 & 2 \text{ flops} \\ \vdots & \\ d_n = b_n - l_{n(n-1)} d_{n-1} - \ldots - l_{n1} d_1 & n-1 \text{ flops} \end{cases}$$

$$\text{Total: } 0 + 1 + 2 + \cdots + (n-1) = \sum_{i=0}^{n-1} i = \frac{(n-1)n}{2}$$
$$= \frac{n^2}{2} + O(n) \text{ flops.}$$

Computing backward solve ($Ux = d$)

Similar: $\frac{n^2}{2} + O(n)$ flops.

<u>Total forward/backward:</u> $n^2 + O(n)$.

<u>Compare to LU-factorization:</u> $\frac{n^3}{3} + O(n^2)$.

$\frac{n}{3}$ times more expensive (in flops)!

This is why we factorize <u>once</u> if there are several systems to solve, only differing in b.

If we applied Gaussian Elim for each, it would take $O(n^3)$ <u>each time</u>.

Roundoff error analysis of GE

Factorization: $PA = LU$

Due to rounding error (initial and propagated), actually get $\hat{L}, \hat{U}$ and $\hat{P}$ st

$$\hat{P}(A+E) = \hat{L}\hat{U}.$$

Hopefully, $\|E\|$ is small compared to $\|A\|$ (if we pivot during the factorization).

($\hat{P}$ is potentially a diff^t pivoting strategy from P, but it is still just I_n with rows permuted.)

Actually, solving (forward and backward) can introduce more roundoff error,

but it can all still be represented as

$$(A+\tilde{E})\hat{x} = b$$

where $\tilde{E}$ is slightly diff^t from E (but is essentially E).

Dropping the distinction: $(A+E)\hat{x} = b$.

Equivalently, let $E\hat{x} = r$; then

$$(A+E)\hat{x} = b \iff \underset{\substack{\uparrow \\ \text{residual} \\ \text{(dust left over)}}}{r} = b - A\hat{x}.$$

Is $\|E\|$ small compared to $\|A\|$?

Result: If we use row partial pivoting during the factorization,

can show that

(*) $\|E\| \lesssim k \cdot eps \|A\|$,

where k is not too large, grows with n, depends on pivoting.

Similarly, for computed sol^n $\hat{x}$ and $r = b - A\hat{x}$,

$$\|r\| \lesssim k \cdot eps \|b\|$$

(*) $$\iff \underbrace{\frac{\|r\|}{\|b\|}}_{\text{relative residual}} \lesssim k \cdot eps.$$

Can prove this, but very technical.

Does this mean that $\dfrac{\|\hat{x} - \underset{\uparrow \text{ true sol}^n}{x}\|}{\|x\|}$ (relative error) is small?

Remember: We don't have x (true sol^n), so we can't directly calculate relative error.

e.g.,

$$\begin{bmatrix} .780 & .563 \\ .913 & .659 \end{bmatrix} x = \begin{bmatrix} .217 \\ .254 \end{bmatrix}. \quad (\text{True sol}^n\text{: } x = \begin{bmatrix} 1 \\ -1 \end{bmatrix};\ \text{in general not known})$$

Consider the computed sol^{n}s

$$\hat{x}_\alpha = \begin{bmatrix} .999 \\ -1.001 \end{bmatrix}, \quad \hat{x}_\beta = \begin{bmatrix} .341 \\ -.087 \end{bmatrix}.$$

Residuals:

$$r_\alpha = b - A\hat{x}_\alpha = \begin{bmatrix} -.001243 \\ -.001572 \end{bmatrix}, \qquad r_\beta = b - A\hat{x}_\beta = \underbrace{\begin{bmatrix} -.000001 \\ 0 \end{bmatrix}}_{\text{smaller residual}}.$$

So $\dfrac{\|r_\beta\|}{\|b\|} \ll \dfrac{\|r_\alpha\|}{\|b\|}$.

Then why is $\dfrac{\|\hat{x}_\alpha - x\|}{\|x\|}$ so much smaller than $\dfrac{\|\hat{x}_\beta - x\|}{\|x\|}$??

Need the relationship b/w relative error and relative residual.

In particular, when does a small relative residual (which is guaranteed if we use row partial pivoting) guarantee a small relative error — how do we check sol^n accuracy?

$$\underbrace{\frac{\|r\|}{\|b\|}}_{\text{relative residual}} \xleftrightarrow{\text{relationship?}} \underbrace{\frac{\|\hat{x} - x\|}{\|x\|}}_{\text{relative error}}$$

...