One thing missing last time:

Catastrophic cancellation:

Substracting two nearly equal numbers cancel the most significant digits but the result can have large relative error

ex1. evaluate $\frac{1}{1-x} - 1$ for $|x| << 1$, $x \in \mathbb{F}$

Method 1: Direct evaluation

$|\delta_i| \leq \varepsilon_{mach}$

$$\text{Output}_1 = \left[\frac{1}{(1-x)(1+\delta_1)} (1+\delta_2) - 1 \right] (1+\delta_3)$$

$$= \frac{[1+\delta_2 - (1-x)(1+\delta_1)](1+\delta_3)}{(1-x)(1+\delta_1)}$$

$$= \frac{\delta_2 - \delta_1 + x(1+\delta_1)}{1+x} \frac{1+\delta_3}{1+\delta_1}$$

when $x \sim O(\delta_2 - \delta_1)$, relative error $\sim O\left(\frac{\delta_2-\delta_1}{x}\right) = O(1)$

Method 2: Rearrange calculation

from $\frac{1}{1-x} - 1 = \frac{x}{1-x}$

$$\text{Output}_2 = \frac{x(1+\delta_1)}{(1-x)(1+\delta_2)} (1+\delta_2)$$

relative error $\sim O(\delta)$ even when $x \sim O(\delta)$

ex 2. $\frac{e^x - 1}{x}$, $|x| << 1$

assume the exp and log function are both computed with a relative error not exceeding ε_{mach}

from Taylor expansion

$$\frac{e^x-1}{x} = \frac{1+x+\frac{1}{2}x^2+\cdots-1}{x} \approx 1+\frac{1}{2}x+O(x^2)$$

Method 1: Direct evaluation

$$\text{Output}_1 = \frac{\left[e^x(1+\delta_1)-1\right](1+\delta_2)}{x(1+\delta_3)}(1+\delta_4)$$

$$= \frac{\left(1+x+\frac{1}{2}x^2+\cdots\right)(1+\delta_1)-1}{x}\,\frac{1+\delta_2}{1+\delta_3}(1+\delta_4)$$

$$\approx \left(\frac{\delta_1}{x}+1+\frac{1}{2}x\right)\frac{1+\delta_2}{1+\delta_3}(1+\delta_4)$$

relative error $\sim O\left(\frac{\delta_1}{x}\right)$

Method 2: Rearrange calculation

First compute $\hat{y} = e^x(1+\delta_1)$

then $\text{Output}_2 = \dfrac{\hat{y}-1}{\log\hat{y}}(1+\delta_2)$

exercise: while the relative errors of numerator and denominator are $O(1)$ for $x\sim O(\varepsilon_{mach})$, Output_2 has $O(\varepsilon_{mach})$ relative error and is accurate

---

Last time: $y=f(x)$, $x\in\mathbb{R}^n$, $y\in\mathbb{R}^m$, approximated by $\hat{f}(x)$

Input		Output

$$\begin{tikzcd}
x \arrow[rr, "f"] \arrow[dd, leftrightarrow, "\text{backward error}"'] \arrow[ddrr, dashed, "\hat{f}"] & & y=f(x) \arrow[dd, leftrightarrow, "\text{forward error}"] \\
& & \\
x+\Delta x \arrow[rr] & & \hat{y}=\hat{f}(x)=f(x+\Delta x)
\end{tikzcd}$$

relative forward error $= \dfrac{\|\hat{f}(x) - f(x)\|}{\|f(x)\|}$

relative backward error $= \dfrac{\|\Delta x\|}{\|x\|}$ s.t. $f(x+\Delta x) = \hat{f}(x)$

Definition : An algorithm is

not unique. can often be chosen to be arbitrary large

- backward stable if $\exists\ \Delta x$, s.t. $\hat{f}(x) = f(x+\Delta x)$,

  $\dfrac{\|\Delta x\|}{\|x\|} = O(\varepsilon_{mach})$

- (numerically) stable if $\exists\ \Delta x, \Delta y$ s.t. $\hat{f}(x) + \Delta y = f(x+\Delta x)$

  $\|\Delta y\|/\|y\|, \ \|\Delta x\|/\|x\| = O(\varepsilon_{mach})$

- accurate (forward stable) if $\dfrac{\|\hat{f}(x) - f(x)\|}{\|f(x)\|}$ is small ($\sim O(\varepsilon_{mach})$)

ex 1. Inner product is backward stable

$|x| := (|x_i|)_{i=1}^{n}$

inner product using floating point numbers

$fl(x^T y) = (x+\Delta x)^T y$ with $|\Delta x| \le \gamma_n |x|$, $\gamma_n = O(n\varepsilon_{mach})$

$|x^T y - fl(x^T y)| = |\Delta x^T y| \le \gamma_n |x|^T |y|$ (*)

ex 2. Outer product is not backward stable

but satisfies $fl(xy^T) = xy^T + E$, $\|E\| \le \varepsilon_{mach} \|xy^T\|$

hence numerically stable

exercise

Remark : backward stability implies numerical stability.

• Q: When is a backward/numerically stable algorithm accurate?

$$\underbrace{\frac{\|\hat{f}(x)-f(x)\|}{\|f(x)\|}}_{\text{forward error}} = \frac{\|f(x+\Delta x)+\Delta y - f(x)\|}{\|f(x)\|}$$

$$\overset{\text{triangle ineq}}{\leq} \frac{\|f(x+\Delta x)-f(x)\|}{\|f(x)\|} + \underbrace{\frac{\|\Delta y\|}{\|y\|}}_{O(\varepsilon_{mach})}$$

$$= \frac{\|f(x+\Delta x)-f(x)\| / \|f(x)\|}{\|\Delta x\| / \|x\|} \underbrace{\frac{\|\Delta x\|}{\|x\|}}_{\text{backward error}} + O(\varepsilon_{mach})$$

(Relative) condition number

$$\kappa(x) := \sup_{\|\Delta x\| \leq \varepsilon_{mach}\|x\|} \frac{\|f(x+\Delta x)-f(x)\| / \|f(x)\|}{\|\Delta x\| / \|x\|}$$

Thm: For a numerically stable algorithm.

$$\frac{\|\hat{f}(x)-f(x)\|}{\|f(x)\|} = O(\kappa(x)\,\varepsilon_{mach})$$

Remark: Forward error $\lesssim$ condition number $\times$ backward error

The condition # κ measures the sensitivity of f to perturbed inputs, which is independent of the algorithm used.

Detour: Vector and matrix norm

To quantify errors for vectors/matrices, we use norms

$\|\cdot\| : \mathbb{C}^n$ (or $\mathbb{C}^{m\times n}$) $\longrightarrow \mathbb{R}$

satisfying 1) $\|x\| \geq 0$, "=" iff $x=0$
2) $\|\alpha x\| = |\alpha| \, \|x\|$, $\forall \alpha \in \mathbb{C}$, $x \in \mathbb{C}^n$
3) $\|x+y\| \leq \|x\| + \|y\|$

example:

Vector norm:

x^* conjugate transpose

1) $\|x\|_p = \left(\sum_{i=1}^{n} |x_i|^p\right)^{1/p}$, $1 \leq p < +\infty$. p-norm

2) $\|x\|_\infty = \max_{1\leq i\leq n} |x_i|$ ∞-norm

3) $p=2 \Rightarrow \|x\|_2 = (x^* x)^{1/2}$ Euclidean norm

Matrix norm:

1) $\|A\|_F = \left(\sum_{ij} |a_{ij}|^2\right)^{1/2} = [\mathrm{tr}(A^* A)]^{1/2}$ Frobenius norm

2) $|A|_\infty = \max_{ij} |a_{ij}|$ max norm

3) $\|A\|_{\alpha,\beta} = \max_{x\neq 0} \dfrac{\|Ax\|_\beta}{\|x\|_\alpha}$ subordinate norm

The subordinate matrix norm measures the size of the output relative to the size of the input.

• example of subordinate norm:

1) $\|x\|_1 = \sum_{i=1}^{n} |x_i|$ is 1-norm

$$Ax = \begin{bmatrix} | & & | \\ a_1 & \cdots & a_n \\ | & & | \end{bmatrix} \begin{bmatrix} x_1 \\ \vdots \\ x_n \end{bmatrix} = x_1 \begin{bmatrix} | \\ a_1 \\ | \end{bmatrix} + \cdots + x_n \begin{bmatrix} | \\ a_n \\ | \end{bmatrix}$$

$$\|Ax\|_1 = \left\| x_1 \begin{bmatrix} | \\ a_1 \\ | \end{bmatrix} + \cdots + x_n \begin{bmatrix} | \\ a_n \\ | \end{bmatrix} \right\|_1$$

$$\leq \sum_{i=1}^{n} |x_i| \, \|a_i\|_1$$

$$\leq \underbrace{\left[\max_{1\leq i\leq n} \|a_i\|_1\right]}_{= \|A\|_1} \|x\|_1$$

("=" holds for $x = e_i = (0, \dots, 0, 1, 0, \dots 0)^T$ picks out max $\|\cdot\|_1$ column of A)

2) $\|A\|_2 = \max_{x\neq 0} \frac{\|Ax\|_2}{\|x\|_2} = \max_{\|x\|_2=1} \sqrt{x^* A^* A x}$

$= \max_{\|x\|_2=1} \sum_{i=1}^{n} |x^* u_i|^2 \lambda_i$

$= \lambda_{max}(A^*A)$ ("=" take $x = u_{i_{max}}$)

$\boxed{\begin{array}{l}(u_i, \lambda_i), i=1,\dots,n \\ \text{eigenvector, eignvalue of matrix } A^*A. \\ \lambda_i \in \mathbb{R}\end{array}}$

Some properties:

1) $\|A\|_{\alpha,\beta} = \max_{x\neq 0} \|A\frac{x}{\|x\|_\alpha}\|_\beta = \max_{\|x\|_\alpha=1} \|Ax\|_\beta$

2) Any subordinate norm is consistent with the vector norm that induce it : $\|Ax\|_\beta \leq \|A\|_{\alpha,\beta} \|x\|_\alpha$

Any subordinate norm is submultiplicative: $\|AB\|_{\alpha,\gamma} \leq \|A\|_{\beta,\gamma} \cdot \|B\|_{\alpha,\beta}$

Pf: $\|ABx\|_\gamma \leq \|A\|_{\beta,\gamma} \|Bx\|_\beta \leq \|A\|_{\beta,\gamma} \|B\|_{\alpha,\beta} \|x\|_\alpha$

Divide both sides by $\|x\|$ and take supreme $x\neq 0$ ▨

3) The Frobenius norm is consistent with the Euclidean norm

$\|Ax\|_2 \leq \|A\|_F \|x\|_2$, and submultiplicatve. (exercise)

max norm is not submultiplicative: $|AB|_\infty \leq n |A|_\infty |B|_\infty$ (exercise)

4) (Equivalence of norms)

For any two vector/matrix norm, $\|\cdot\|_\alpha$, $\|\cdot\|_\beta$.

We have $r\|A\|_\alpha \leq \|A\|_\beta \leq s\|A\|_\alpha$

for some $r, s > 0$, for all $A \in \mathbb{C}^{m\times n}$

(r, s only depend on how the norm $\|\cdot\|_\alpha$, $\|\cdot\|_\beta$ are defined and the dimension m, n)

ex. $\frac{1}{\sqrt{n}}\|x\|_2 \leq \|x\|_1 \leq \sqrt{n}\|x\|_2$, $\frac{1}{\sqrt{n}}\|A\|_2 \leq \|A\|_1 \leq \sqrt{n}\|A\|_2$

Now we are ready to handle condition #'s

If $f(x) = (f_1(x), \cdots, f_m(x)) : \mathbb{R}^n \to \mathbb{R}^m$ is differentiable then,

$$f_j(x+\Delta x) = f_j(x) + \sum_{i=1}^{n} \frac{\partial f_j}{\partial x_i}(x)\, \Delta x_i + O(\|\Delta x\|^2)$$

Jacobian $Df(x) = \left(\frac{\partial f_j}{\partial x_i}(x) \right)_{\substack{1\le i\le n \\ 1\le j\le m}}$

Then $f(x+\Delta x) = f(x) + Df(x)\,\Delta x + O(\|\Delta x\|^2)$

Recall the definition.

$$K(x) := \sup_{\frac{\|\Delta x\|}{\|x\|} \le \varepsilon_{mach}} \frac{\|f(x+\Delta x) - f(x)\| / \|f(x)\|}{\|\Delta x\| / \|x\|}$$

$$= \sup_{\Delta x} \frac{\|Df(x)\,\Delta x + O(\|\Delta x\|^2)\|}{\|\Delta x\|} \frac{\|x\|}{\|f(x)\|}$$

$$\overset{\text{consistency of matrix norm}}{\le} \underbrace{\frac{\|Df(x)\|\,\|x\|}{\|f(x)\|}}_{\text{condition number for differentiable system}} + \underbrace{O(\varepsilon_{mach}\|x\|^2/\|f(x)\|)}_{\text{usually negligible or comparable to the previous term.}}$$

example: Summation function

$f(x) = \sum_{i=1}^{n} x_i$ (a special case of inner product with $y = \mathbb{1}$. hence backward stable)

$Df(x) = [1, \cdots, 1]$

Take $\|\cdot\|_1$ in the following

$$\|Df(x)\|_1 = 1$$

$$K(x) = \frac{\|Df(x)\|_1 \, \|x\|_1}{|f(x)|} = \frac{\sum_{i=1}^{n} |x_i|}{\left|\sum_{i=1}^{n} x_i\right|}$$

The forward error

$$\frac{|\hat{f}(x) - f(x)|}{|f(x)|} = O\left(\frac{\sum_{i=1}^{n} |x_i|}{\left|\sum_{i=1}^{n} x_i\right|} \varepsilon_{mach} \right)$$

Remarks:

1) Estimating the backward error $\frac{\|\Delta x\|}{\|x\|}$ is call backword error analysis. Combining backward error (of an algorithm) and condition # (of a problem) yields forward error.

2) Forward error bound can also be obtained directly here by using the error bound (*) (on pp. 3).