Stability of QR factorization

Recall: QR factorization is a series of steps of the form

$$\begin{bmatrix} I & \\ & H(\underset{\sim}{u_k}) \end{bmatrix} \underbrace{\boxed{\begin{bmatrix} * & * & * & * & * \\ 0 & * & * & * & * \\ 0 & 0 & * & * & * \\ 0 & 0 & * & * & * \\ 0 & 0 & * & * & * \end{bmatrix}}}_{R_{k-1}} = \underbrace{\begin{bmatrix} * & * & * & * & * \\ 0 & * & * & * & * \\ 0 & 0 & * & * & * \\ 0 & 0 & 0 & * & * \\ 0 & 0 & 0 & * & * \end{bmatrix}}_{R_k}$$

Each of these steps, alone, is backward stable: the computed $\widetilde{R_k}, \widetilde{u_k}$ is the exact result that we'd get if we started from $R_{k-1} + \underline{\Delta R_{k-1}}$, with

$$\|\Delta R_{k-1}\| = O(\mathbf{u})\|R_{k-1}\|.$$

(The $O(\mathbf{u})$ notation here may 'hide' polynomial factors in n).

Backward stability combined

At each step, we can pretend that all operations we have made up to there are the exact result obtained if we'd start from

$$A + \Delta_1 A + \Delta_2 A + \cdots + \Delta_{k-1} A.$$

and for each i we have $\|\Delta_i A\| \leq O(\mathbf{u})\|R_{k-1}\| \approx O(\mathbf{u})\|A\|$.

Important detail: $\|R_{k-1}\| \approx \|A\|$ because we get one from the other through orthogonal transformations.

QR factorization is backward stable: the computed Q, R are the exact result of $\mathtt{qr}(A + \Delta A)$, $\|\Delta A\| \leq O(\mathbf{u})\|A\|$.

(Some tedious algebra required to formalize all this — not our goal in this course.)

Backward stability of least squares algorithms

```
function x = solve_ls_QR(A, b)
[Q1, R1] = qr(A, 0); %backward stable
c = Q1'*b; %backward stable
x = R1 \ c; %backward stable
```

Backward stable: all errors here of size $\boxed{O(\mathbf{u})\|A\|.}$

```
function x = solve_ls_SVD(A, b)
[U, S, V] = svd(A, 0); %backward stable
c = U'*b; %backward stable
d = c ./ diag(S); %backward stable
x = V*d; %backward stable
```

$$d = \begin{bmatrix} c_1/\sigma_1 \\ c_2/\sigma_2 \\ \vdots \\ c_n/\sigma_n \end{bmatrix}$$

Backward stable: all errors here of size $O(\mathbf{u})\|A\|$.

(SVD is backward stable too — we still haven't seen how to compute it, but it's all orthogonal transformations, too.)

The problem with normal equations

```
function x = solve_ls_NE(A, b)
C = A' * A;
d = A' * b;
x = C \ d;
```

$\leftarrow (A^T A)^{-1} (A^T b)$

How is this not backward stable? Same kind of operations...

However,

- We use inputs multiple times: so for instance $\tilde{C} = (A + \Delta_1 A)^T (A + \Delta_2 A)$: not necessarily the same perturbation.

  $\|A^T A\| = \|A\|^2$

- In the last line, errors have size $O(\mathbf{u})(\|C\|) = O(\mathbf{u})(\|A\|^2)$.

So, long story short (we didn't *prove* everything)

The QR and SVD methods are backward stable, but normal equations always give errors of size $\underbrace{\kappa(A)^2}_{\|A\|}$, never $\kappa(A)$.

$$\left(\|A\|\cdot\|A^{-1}\|\right)^2 = \left(\|A^TA\|\cdot\|(A^TA)^{-1}\|\right)$$

$$\kappa(A)^2 = \kappa(A^TA)$$

---

original example:

$$A = \begin{bmatrix} 1 & 2 & 3 \\ 3 & 1 & 4 \\ \vdots & & \vdots \\ 1 & 2 & 3+10^{-8} \end{bmatrix}$$

at distance $\leq 10^{-8}$ from a non-full-col.-rank matrix

$$\sigma_n \leq 10^{-8} \qquad \kappa(A) = \frac{\|A\|}{\sigma_n} \geq 10^8$$

$b \in \operatorname{Im} A \qquad \cos\theta = 1 \qquad \tan\theta = 0$

condition number $\kappa(A)$ relative to both b, A

We expect QR and SVD to deliver results within $\kappa(A)\cdot \underline{u} \approx 10^{8}\cdot 10^{-16} = 10^{-8}$

$$\|x-\tilde{x}\| \lesssim 10^{-8}\cdot\|x\|$$

$$\begin{bmatrix} 3 \\ 4 \\ 5 \end{bmatrix}$$

Normal equations: result has an error of relative magnitude $\kappa(A)^2 u \approx (10^{8})^2 u = 10^{16}\cdot 10^{-16}$

≈ 1

Comparison of least squares algorithms

	Normal eqns	QR	SVD
$m \approx n$	$\frac{4}{3}n^3$	$\frac{4}{3}n^3$	$\approx 13n^3$
$m \gg n$	mn^2	$2mn^2$	$2mn^2$
	Unstable when $cond \approx \kappa(A)$	$\boxed{\text{Backward stable}}$	$\boxed{\text{Backward stable; reveals info on sensitivity, allows regularization}}$

Know when to use each one. QR is a good 'generic' choice.

A \ b.

Exercises

1. Show that $\|A^T A\| = \|A\|^2$. Hint: recall how $\|A\| = \|U\Sigma V^T\| = \|\Sigma\| = \sigma_1$: can we do something similar for $A^T A$?
2. Show that $\kappa(A^T A) = \kappa(A)^2$.