Math 381 - Fall 2022

Jay Newby

University of Alberta

Week 10

Last Time

1. Example: forward and backward error for Householder Triangularization
2. Backward stability for QR decomposition by Householder Triangularization
3. Backward stability for nonsingular linear systems using QR

Today

1. Continue backward stability analysis for nonsingular linear systems using QR
2. Conditioning of the least squares problem

Review from last time

Algorithm: Solving $Ax = b$ using QR Factorization

Let $A \in \mathbb{R}^{n \times n}$ be nonsingular and $b \in \mathbb{R}^n$.

1. $QR = A$ with Householder Triangularization
2. $y = Q^T b$ (using the algorithm in Monday slides)
3. $x = R^{-1} A$ solve the upper triangular linear system using Backward Substitution

Backward Stability of Householder $y = Q^T b$

Let $y = Q^T b$. The output of the Householder $y = Q^T b$ algorithm generates a $\tilde{y}$ that exactly solves

$$(Q + \delta Q)\tilde{y} = b,$$

for some $\delta Q \in \mathbb{R}^{n\times n}$ such that $\|\delta Q\| = O(\epsilon)$.

Backward Stability of Backward Substitution

Let $R \in \mathbb{R}^{n\times n}$ be upper triangular and nonsingular, and let $x = R^{-1}y$ for some $y \in \mathbb{R}^n$. The output of Backward Substitution generates a $\tilde{x}$ that exactly solves

$$(R + \delta R)\tilde{x} = y,$$

for some $\delta R \in \mathbb{R}^{n\times n}$ such that $\frac{\|\delta R\|}{\|R\|} = O(\epsilon)$.

If each step is backward stable, is the entire algorithm also backward stable?

Theorem:

The algorithm on Slide 4 is backward stable; it generates a solution $\tilde{x}$ that exactly solves

$$(A + \Delta A)\tilde{x} = b,$$

for some $\Delta A \in \mathbb{R}^{n\times n}$ such that $\frac{\|\Delta A\|}{\|A\|} = O(\epsilon)$.

Proof (sketch):

Backward stability of QR: $\tilde{Q}\tilde{R} = A + \delta A$ s.t. $\frac{\|\delta A\|}{\|A\|} = O(\epsilon)$

$$[\underbrace{\tilde{Q}}_{\text{step 1}} + \underbrace{\delta Q}_{\text{step 3}}][\underbrace{\tilde{R}}_{\text{step 1}} + \underbrace{\delta R}_{\text{step 2}}]\tilde{x} = b$$

expand

$$[\tilde{Q}\tilde{R} + \tilde{Q}\delta R + \delta Q\tilde{R} + \delta Q\delta R]\tilde{x} = b$$

$$\Rightarrow [A + \underbrace{\delta A + \tilde{Q}\delta R + \delta Q\tilde{R} + \delta Q\delta R}_{\Delta A}]\tilde{x} = b$$

want to show $\frac{\|\Delta A\|}{\|A\|} = O(\epsilon)$, where

$\Delta A = \delta A + \tilde{Q}\delta R + \delta Q\tilde{R} + \delta Q\delta R$

rest is an exercise

What does backward stability tell us about accuracy?

Theorem:

The solution $\tilde{x}$ computed by the algorithm on Slide 4 satisfies

$$\frac{\|x - \tilde{x}\|}{\|x\|} = O(\kappa(A)\epsilon).$$

Backward stability $\Rightarrow$ $[A + \Delta A]\tilde{x} = b = Ax$

$\Rightarrow A(\tilde{x} - x) = -\Delta A \tilde{x}$

$\Rightarrow \tilde{x} - x = -A^{-1}\Delta A \tilde{x}$

$\Rightarrow \|\tilde{x} - x\| \leq \|A^{-1}\| \, \|\Delta A\| \, \|\tilde{x}\|$

$\Rightarrow \dfrac{\|\tilde{x} - x\|}{\|x\|} \leq \|A^{-1}\| \|A\| \dfrac{\|\tilde{x}\|}{\|x\|} \dfrac{\|\Delta A\|}{\|A\|} \leq \kappa(A) \dfrac{\|\Delta A\|}{\|A\|} \leq \kappa(A)\varepsilon$

Conditioning determines the relationship between backward stability and accuracy

Theorem:

The solution $\tilde{x}$ computed by a backward stable algorithm satisfies

$$\frac{\|x - \tilde{x}\|}{\|x\|} = O(\kappa(p)\epsilon),$$

where $\kappa(p)$ is the condition number for the problem with parameters p.

Algorithms will give inaccurate answers to badly conditioned problems (even if they are backward stable).

What about the conditioning of the least squares problem?

It is oftain said that the Cholesky algorithm applied to the normal equations is an "unstable" algorithm for solving the least squares problem. Is this true?

Suppose we have two algorithms with the same backward stability result

For example:

$$\|(A + \delta A)\tilde{x} - b\|_2^2 = \min,$$

for some $\delta A \in \mathbb{R}^{m \times n}$ such that

$$\frac{\|\delta A\|}{\|A\|} = O(\epsilon).$$

Can one algorithm be more resilient than another to badly conditioned matrices (i.e., $\kappa(A) \gg 1$)?

Suppose we have two algorithms with different backward stability results

- If two backward stable algorithms yield exact solutions to different perturbed problems, the condition number of the problem will be different also (in general).
- If the problem parameters are perturbed in two different ways, the resulting condition numbers characterizing the response of the error to those perturbations will be different.

Least squares problem (full rank)

Linear least squares problem

Let $A \in \mathbb{R}^{m \times n}$ be full rank with $m > n$, and let $b \in \mathbb{R}^m$. We wish to find the vector $x \in \mathbb{R}^n$ which solves the minimization problem,

$$\min_{x \in \mathbb{R}^n} \|b - Ax\|_2^2.$$

We can generalize the condition number of a matrix to cover all full rank matrices

Our previous definition applied only to square non singular matrices

Definition: condition number of full rank matrices

Let $A \in \mathbb{R}^{n \times m}$ be full rank with $m > n$, and let $A^{\dagger}$ be the appropriately defined pseudo inverse of A. The condition number is defined as

$$\kappa(A) = \|A\| \|A^{\dagger}\|.$$

Setting the scene with two definitions

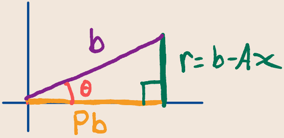
These will show up in the condition number for the least squares problem

Definition: angle of closeness of fit

Let $y = Pb$, where P is an orthogonal projector onto the range of A.

$$\theta = \cos^{-1}\left(\frac{\|y\|}{\|b\|}\right),$$

$$0 \leq \theta \leq \pi/2.$$



Definition:

$$\eta = \frac{\|A\|\|x\|}{\|Ax\|},$$

$$1 \leq \eta \leq \kappa(A).$$

$\|Ax\| \leq \|A\|\|x\|$

Theorem: Conditioning of solving the least squares problem to perturbations in A

Given the perturbation

$$\delta A = \begin{bmatrix} \delta A_1 \\ \delta A_2 \end{bmatrix} \qquad \begin{array}{r|c|} & n \\ \hline n & \delta A_1 \\ \hline m-n & \delta A_2 \\ \hline \end{array}$$

where $\delta A_1 \in \mathbb{R}^{n\times n}$ and $\delta A_2 \in \mathbb{R}^{(m-n)\times n}$, we have the following 2-norm bounds on accuracy of the least squares solution

$$\frac{\|\tilde{x} - x\|}{\|x\|} \leq \kappa(A)\frac{\|\delta A_1\|}{\|A\|} + \frac{\kappa(A)^2 \tan(\theta)}{\eta}\frac{\|\delta A_2\|}{\|A\|}$$

$$1 \leq \eta \leq \kappa(A)$$

Backward stability: QR (Gram-Schmidt or Householder)

Theorem: Least squares with the QR decomposition

Solving the least squares problem using the QR decomposition with Householder Triangularization or Gram-Schmidt orthogonalization is backward stable. That is, the approximate solution $\tilde{x}$ satisfies

$$\|(A+\delta A)\tilde{x} - b\|_2^2 = \min,$$

for some $\delta A \in \mathbb{R}^{m\times n}$ such that

$$\frac{\|\delta A\|}{\|A\|} = O(\epsilon).$$

Backward stability: SVD

Theorem: Least squares with the SVD decomposition

Solving the least squares problem using the SVD decomposition is backward stable. That is, the approximate solution $\tilde{x}$ satisfies

$$\|(A + \delta A)\tilde{x} - b\|_2^2 = \min,$$

for some $\delta A \in \mathbb{R}^{m \times n}$ such that

$$\frac{\|\delta A\|}{\|A\|} = O(\epsilon).$$

Backward stability: Cholesky

$M = A^T A$

Theorem: Cholesky factorization to solve $Mx = y$

Solving a linear system $Mx = y$ with $M \in \mathbb{R}^{m \times m}$ invertible and symmetric is backward stable. That is, the approximate solution $\tilde{x}$ satisfies

$$(M + \delta M)\tilde{x} = y,$$

for some $\delta M \in \mathbb{R}^{m \times m}$ such that

$$\frac{\|\delta M\|}{\|M\|} = O(\epsilon).$$

This is the definition of backward stable! So why do we say that Cholesky + normal equations algorithm is unstable?

Conditioning of solving normal equations

Theorem: Conditioning of solving the normal equations to perturbations in A

$Mx = b$

$\kappa(M) = \kappa(A^T A) = \kappa(A)^2$

Given a solution $\tilde{x}$ that satisfies

$$(A^T A + \delta M)\tilde{x} = A^T b$$

for $\frac{\|\delta M\|}{\|A^T A\|} = O(\epsilon)$, the relative error in the exact solution x is bounded by

$$\frac{\|\tilde{x} - x\|}{\|x\|} = O(\kappa(A)^2 \epsilon)$$

The condition numbers for solving least squares are different because the backward stability perturbation for normal equations is different from QR and SVD

Normal Equation

QR or SVD

$$\frac{\|\tilde{x}-x\|}{\|x\|} = O(\kappa(A)^2\epsilon), \qquad \frac{\|\tilde{x}-x\|}{\|x\|} = O\left(\left(\kappa(A) + \frac{\tan(\theta)}{\eta}\kappa(A)^2\right)\epsilon\right)$$

Better when $0 < \theta \ll 1$

or $\eta \gg 1$

The condition numbers for solving least squares are different because the backward stability perturbation for normal equations is different from QR and SVD

Solving the least squares problem using the normal equations is “unstable” (i.e., more sensitive to badly conditioned matrix A than QR and SVD) unless the problem is restricted to $\kappa(A)$ small or $\tan(\theta)/\eta$ large. In other words, the normal equations are unstable for ill-conditioned matrices involving close fits.

By the same token, the QR and SVD algorithms are “unstable” (compared to the normal equations) if $\kappa(A)$ and $\tan(\theta)/\eta$ are large.

Rank deficient least squares

The idea is to minimize $\|b - Ax\|_2^2$ and $\|x\|_2$. SVD is backward stable for solving the problem. The general SVD is

$$A = \begin{bmatrix} \hat{U} & U_0 \end{bmatrix} \begin{bmatrix} \hat{\Sigma} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \hat{V}^T \\ V_0^T \end{bmatrix}.$$

The above can be use to derive a general solution to the rank deficient least squares problem.