Numerical Matrix Analysis

Notes #13 — Conditioning and Stability: Stability of Back Substitution

Peter Blomgren
⟨blomgren@sdsu.edu⟩

Department of Mathematics and Statistics
Dynamical Systems Group
Computational Sciences Research Center
San Diego State University
San Diego, CA 92182-7720

http://terminus.sdsu.edu/

Spring 2024
(Revised: March 7, 2024)

Outline

Last Time: Stability of Householder Triangularization

- We discussed the stability properties of QR-factorization by Householder Triangularization (HT-QR).
  - Numerical "evidence" that HT-QR is backward stable.
  - Statement (proof by reference to Higham's *Accuracy and Stability of Numerical Algorithms*) that HT-QR is backward stable

- Showed that solving $A\vec{x} = \vec{b}$ using HT-QR and backward substitution is backward stable, assuming that

  (1) $QR = A$ by HT-QR is backward stable

  (2) $\tilde{w} = Q^*\vec{b}$ is backward stable

  (3) $R\vec{x} = \tilde{w}$ by back substitution is backward stable

- **Today:** Explicit proof of (3), and implicit proof of (2).

Backward Stability of Back Substitution

Back substitution is one of the **easiest non-trivial algorithms** we study in numerical linear algebra, and is therefore a good venue for a full backward stability proof.

The proof for backward stability of Householder triangularization follows the same pattern, but the details become more cumbersome.

Back-substitution applies to $R\vec{x} = \vec{b}$, where

$$\begin{bmatrix} r_{11} & r_{12} & \cdots & r_{1m} \\ & r_{22} & & r_{2m} \\ & & \ddots & \vdots \\ & & & r_{mm} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_m \end{bmatrix} = \begin{bmatrix} b_1 \\ b_2 \\ \vdots \\ b_m \end{bmatrix}$$

Upper (and lower) triangular matrices are generated by, *e.g.* the QR-factorization [NOTES#6–7], Gaussian elimination [NOTES#16–17], and the Cholesky factorization [NOTES#17].

Algorithm: Back-Substitution

Algorithm (Back-Substitution)

1: $x_m \leftarrow b_m / r_{mm}$
2: **for** $\ell \in \{(m-1), \dots, 1\}$ **do**
3: $\quad x_\ell \leftarrow \left(b_\ell - \sum_{k=\ell+1}^{m} x_k r_{\ell k} \right) / r_{\ell\ell}$
4: **end for**

Note that the algorithm breaks if $r_{\ell\ell} = 0$ for some ℓ.

For this discussion we make the assumption that $b_\ell - \sum(x_k r_{\ell k})$ is computed as $(m-\ell)$ subtractions performed in k-increasing order.

Simplification: In the theorem/proof, we use the convention that if the denominator in a statement like $\frac{|\delta r_{i\ell}|}{|r_{i\ell}|} \leq m\varepsilon_{\text{mach}}$ is zero, we implicitly assert that the numerator is also zero, as $\varepsilon_{\text{mach}} \to 0$. This can be fully formalized, but at this stage it unnecessarily complicates the discussion).

Reference: Key Floating Point Axioms

Floating Point Representation Axiom

$\forall x \in \mathbb{R}$, there exists ϵ with $|\epsilon| \leq \epsilon_{\text{mach}}$,
such that $\texttt{fl}(x) = x(1+\epsilon)$.

The Fundamental Axiom of Floating Point Arithmetic

For all $x, y \in \mathbb{F}_n$ (where $\mathbb{F}_n$ is the set of n-bit floating point numbers), there exists ϵ with $|\epsilon| \leq \epsilon_{\text{mach}}$, such that

$$\begin{aligned} x \oplus y &= (x+y)(1+\epsilon), & x \ominus y &= (x-y)(1+\epsilon), \\ x \otimes y &= (x*y)(1+\epsilon), & x \oslash y &= (x/y)(1+\epsilon) \end{aligned}$$

San Diego State University

Back-Substitution: Backward Stability Theorem

Theorem (Solving an Upper Triangular System $R\vec{x} = \vec{b}$ Using Back-Substitution is Backward Stable)

Let the back-substitution algorithm be applied to $R\vec{x} = \vec{b}$, where $R \in \mathbb{C}^{m \times m}$ is upper triangular; $\vec{b}, \vec{x} \in \mathbb{C}^m$; in a floating-point environment satisfying the floating point axioms. The algorithm is backward stable in the sense that the computed solution $\tilde{x} \in \mathbb{C}^m$ satisfies

$$(R + \delta R)\tilde{x} = \vec{b}$$

for some upper triangular $\delta R \in \mathbb{C}^{m \times m}$ with

$$\frac{\|\delta R\|}{\|R\|} = \mathcal{O}(\varepsilon_{mach}).$$

Specifically, for each i, ℓ

$$\frac{|\delta r_{i\ell}|}{|r_{i\ell}|} \leq m\varepsilon_{mach} + \mathcal{O}(\varepsilon_{mach}^2).$$

SAN DIEGO STATE UNIVERSITY

Proof: $m = 1$

When $m = 1$, back substitution terminates in one step

$$\tilde{x}_1 = b_1 \oslash r_{11}$$

The error introduced in this step is captured by

$$\tilde{x}_1 = \frac{b_1}{r_{11}}(1 + \epsilon_1^{\oslash}), \quad |\epsilon_1^{\oslash}| \leq \varepsilon_{\text{mach}}.$$

Since we want the express the error in terms of **perturbations of R**, we write

$$\tilde{x}_1 = \frac{b_1}{r_{11}(1 + \epsilon_1')}, \quad |\epsilon_1'| \leq \varepsilon_{\text{mach}} + \mathcal{O}(\varepsilon_{\text{mach}}^2).$$

Hence,

$$(r_{11} + \delta r_{11})\tilde{x}_1 = b_1, \quad \frac{|\delta r_{11}|}{|r_{11}|} \leq \varepsilon_{\text{mach}} + \mathcal{O}(\varepsilon_{\text{mach}}^2) = \mathcal{O}(\varepsilon_{\text{mach}}).$$

San Diego State University

A Note on $(1+\epsilon)$ and $1/(1+\epsilon')$

In backward stability proofs we frequently need to move terms of the type $(1+\epsilon)$ from/to the numerator to/from the denominator.

We do this because we want to express all the floating point errors as perturbations to a specific part of the expression, *e.g.* the matrix R in the instance of backward substitution.

When ϵ is small, we can set

$$\epsilon' = \frac{-\epsilon}{1+\epsilon} \sim -\epsilon(1-\epsilon+\mathcal{O}(\epsilon^2)) = -\epsilon + \mathcal{O}(\epsilon^2)$$

and thus $\left(\textbf{discarding } \mathcal{O}\left(\epsilon^2\right) \textbf{-terms}\right)$

$$1+\epsilon' = \frac{1+\epsilon}{1+\epsilon} - \frac{\epsilon}{1+\epsilon} = \frac{1+\epsilon-\epsilon}{1+\epsilon} = \frac{1}{1+\epsilon} \quad \Rightarrow \quad \frac{\mathbf{1}}{\mathbf{1}+\epsilon'} = \mathbf{1}+\epsilon.$$

Bottom line: we can move $(1+\epsilon)$ terms (where $|\epsilon| \leq \varepsilon_{\text{mach}} \ll 1$) between the numerator and denominator, and only introduce errors of the order $\mathcal{O}(\varepsilon_{\text{mach}}^2)$, *i.e.* $|\epsilon'| \leq \varepsilon_{\text{mach}} + \mathcal{O}(\varepsilon_{\text{mach}}^2)$.

Proof: $m = 2$

Step one (which computes $\tilde{x}_2$) is exactly like the $m = 1$ case:

$$\tilde{x}_2 = \frac{b_2}{r_{22}(1+\epsilon_1^{\oslash})}, \quad |\epsilon_1| \leq \varepsilon_{\text{mach}} + \mathcal{O}(\varepsilon_{\text{mach}}^2).$$

The second step is defined by

$$\tilde{x}_1 = (b_1 \ominus (\tilde{x}_2 \otimes r_{12})) \oslash r_{11}.$$

We get

$$\begin{aligned}
\tilde{x}_1 &= (b_1 \ominus (\tilde{x}_2 r_{12}(1+\epsilon_2^{\otimes}))) \oslash r_{11} \\
&= (b_1 - \tilde{x}_2 r_{12}(1+\epsilon_2^{\otimes}))(1+\epsilon_3^{\ominus}) \oslash r_{11} \\
&= \frac{(b_1 - \tilde{x}_2 r_{12}(1+\epsilon_2^{\otimes}))(1+\epsilon_3^{\ominus})(1+\epsilon_4^{\oslash})}{r_{11}}
\end{aligned}$$

Proof: $m = 2$

As before, we can shift the $(1+\epsilon_3^{\ominus})$ and $(1+\epsilon_4^{\oslash})$ terms to the denominator

$$\tilde{x}_1 = \frac{b_1 - \tilde{x}_2 r_{12}(1+\epsilon_2^{\otimes})}{r_{11}(1+\epsilon_3'^{\ominus})(1+\epsilon_4'^{\oslash})} = \frac{b_1 - \tilde{x}_2 \mathbf{r_{12}(1+\epsilon_2^{\otimes})}}{\mathbf{r_{11}(1+2\epsilon_5^{\ominus,\oslash})}}$$

where $|\epsilon_{3,4}'|, |\epsilon_5| \leq \varepsilon_{\text{mach}} + \mathcal{O}(\varepsilon_{\text{mach}}^2)$.

Now

$$(R + \delta R)\tilde{x} = \vec{b}$$

since $\mathbf{r_{11}}$ is perturbed by the factor $\mathbf{(1+2\epsilon_5^{\ominus,\oslash})}$, $\mathbf{r_{12}}$ by the factor $\mathbf{(1+\epsilon_2^{\otimes})}$, and r_{22} by the factor $(1+\epsilon_1^{\oslash})$. The entries satisfy

$$\begin{bmatrix} |\delta r_{11}|/|r_{11}| & |\delta r_{12}|/|r_{12}| \\ & |\delta r_{22}|/|r_{22}| \end{bmatrix} = \begin{bmatrix} 2|\epsilon_5^{\ominus,\oslash}| & |\epsilon_2^{\otimes}| \\ & |\epsilon_1^{\oslash}| \end{bmatrix} \leq \begin{bmatrix} 2 & 1 \\ & 1 \end{bmatrix} \varepsilon_{\text{mach}} + \mathcal{O}(\varepsilon_{\text{mach}}^2)$$

Thus $\|\delta R\|/\|R\| = \mathcal{O}(\varepsilon_{\text{mach}})$.

San Diego State University

Proof: $m = 3$

The first two steps are as before, and we get

$$\begin{cases} \tilde{x}_3 = b_3 \oslash r_{33} & = \dfrac{b_3}{r_{33}(1+\epsilon_1^{\oslash})} \\[2ex] \tilde{x}_2 = (b_2 \ominus (\tilde{x}_3 \otimes r_{23})) \oslash r_{22} & = \dfrac{b_2 - \tilde{x}_3 r_{23}(1+\epsilon_2^{\otimes})}{r_{22}(1+2\epsilon_3^{\oslash,\ominus})} \end{cases}$$

where superscipts on ϵs indicate the source operation; now

$$\begin{bmatrix} 2|\epsilon_3| & |\epsilon_2| \\ & |\epsilon_1| \end{bmatrix} \leq \begin{bmatrix} 2 & 1 \\ & 1 \end{bmatrix} \varepsilon_{\text{mach}} + \mathcal{O}(\varepsilon_{\text{mach}}^2)$$

We take a deep breath, and write down the third step

$$\tilde{x}_1 = [(b_1 \ominus (\tilde{x}_2 \otimes r_{12})) \ominus (\tilde{x}_3 \otimes r_{13})] \oslash r_{11}$$

Proof: $m = 3$

We expand the two $\otimes$ operations, and write

$$\tilde{x}_1 = \left[(b_1 \ominus \tilde{x}_2 r_{12}(1+\epsilon_4^{\otimes})) \ominus \tilde{x}_3 r_{13}(1+\epsilon_5^{\otimes}) \right] \oslash r_{11}$$

We introduce error bounds for the $\ominus$ operations

$$\tilde{x}_1 = \left[(b_1 - \tilde{x}_2 r_{12}(1+\epsilon_4^{\otimes}))(1+\epsilon_6^{\ominus}) - \tilde{x}_3 r_{13}(1+\epsilon_5^{\otimes}) \right] (1+\epsilon_7^{\ominus}) \oslash r_{11}$$

Finally, we convert $\oslash$ to a mathematical division with a perturbation ϵ_8; and move both the $(1+\epsilon_{7,8})$ expressions to the denominator

$$\tilde{x}_1 = \frac{(\mathbf{b_1} - \tilde{x}_2 r_{12}(1+\epsilon_4^{\otimes}))\mathbf{(1+\epsilon_6^{\ominus})} - \tilde{x}_3 r_{13}(1+\epsilon_5^{\otimes})}{r_{11}(1+\epsilon_7'^{\ominus})(1+\epsilon_8'^{\oslash})}$$

As it stands, we have introduced a perturbation in b_1. This was not our intention, so we ship $(1+\epsilon_6^{\ominus})$ to the denominator as well...

Proof: $m = 3$

We now have an expression with perturbations in only $r_{1\ell}$:

$$\tilde{x}_1 = \frac{b_1 - \tilde{x}_2 r_{12}(1+\epsilon_4^{\otimes}) - \tilde{x}_3 r_{13}(1+\epsilon_5^{\otimes})\mathbf{(1+\epsilon_6'^{\ominus})}}{r_{11}\mathbf{(1+\epsilon_6'^{\ominus})}(1+\epsilon_7'^{\ominus})(1+\epsilon_8'^{\oslash})}$$

where $|\epsilon_{4,5}| \leq \varepsilon_{\text{mach}}$, and $|\epsilon'_{6,7,8}| \leq \varepsilon_{\text{mach}} + \mathcal{O}(\varepsilon^2_{\text{mach}})$.

If we collect the limits on the relative sizes of the perturbations $|\delta r_{i\ell}|/|r_{i\ell}|$ we get the following 6 relations

$$\begin{bmatrix} |\delta r_{11}|/|r_{11}| & |\delta r_{12}|/|r_{12}| & |\delta r_{13}|/|r_{13}| \\ & |\delta r_{22}|/|r_{22}| & |\delta r_{23}|/|r_{23}| \\ & & |\delta r_{33}|/|r_{33}| \end{bmatrix} \leq \begin{bmatrix} 3 & 1 & 2 \\ & 2 & 1 \\ & & 1 \end{bmatrix} \varepsilon_{\text{mach}} + \mathcal{O}(\varepsilon^2_{\text{mach}})$$

We are now ready to identify the pattern for general values of m...

SAN DIEGO STATE UNIVERSITY

Proof: General m

The division by r_{ii} induces perturbations δr_{ii} only, since we always immediately shift that $(1+\epsilon_*)$-term to the denominator $1/(1+\epsilon'_*)$, hence the perturbation pattern is of the form

$$\oslash \quad \rightsquigarrow \quad I_{n\times n}\varepsilon_{\text{mach}} + \mathcal{O}(\varepsilon_{\text{mach}}^2)$$

The multiplications $\tilde{x}_i r_{\ell i}$ induces perturbations $\delta r_{\ell i}$ of relative size $\leq \varepsilon_{\text{mach}}$, the perturbation pattern is of the form

$$\otimes \quad \rightsquigarrow \quad \begin{bmatrix} 0 & 1 & 1 & \dots & 1 \\ & 0 & 1 & \dots & 1 \\ & & \ddots & \ddots & \vdots \\ & & & 0 & 1 \\ & & & & 0 \end{bmatrix} \varepsilon_{\text{mach}}$$

San Diego State University

Proof: General m

The most complicated contribution comes from the subtractions (and this is where the order of evaluation has an effect on the answer) — in computing $\tilde{x}_k$

$r_{k,k}$	is perturbed by	$(1+\epsilon'_*)^{m-k}$
$r_{k,k+1}$	is perturbed by	0
$r_{k,k+2}$	is perturbed by	$(1+\epsilon'_*)$
$r_{k,k+3}$	is perturbed by	$(1+\epsilon'_*)^2$
	$\vdots$	
$r_{k,m}$	is perturbed by	$(1+\epsilon'_*)^{m-k-1}$

See next slide for the pattern.

Proof: General m

$$\ominus \;\rightsquigarrow\; \begin{bmatrix} (m-1) & 0 & 1 & 2 & 3 & \dots & (m-2) \\ & \ddots & \ddots & \ddots & \ddots & \ddots & \vdots \\ & & 4 & 0 & 1 & 2 & 3 \\ & & & 3 & 0 & 1 & 2 \\ & & & & 2 & 0 & 1 \\ & & & & & 1 & 0 \\ & & & & & & 0 \end{bmatrix} \varepsilon_{\text{mach}} + \mathcal{O}(\varepsilon_{\text{mach}}^2)$$

Putting all this together gives...

Proof: General m — Collecting It All

$$\frac{|\delta R|}{|R|} \leq \begin{bmatrix} m & 1 & 2 & 3 & 4 & \dots & (m-1) \\ & \ddots & \ddots & \ddots & \ddots & \ddots & \vdots \\ & & 5 & 1 & 2 & 3 & 4 \\ & & & 4 & 1 & 2 & 3 \\ & & & & 3 & 1 & 2 \\ & & & & & 2 & 1 \\ & & & & & & 1 \end{bmatrix} \varepsilon_{\mathsf{mach}} + \mathcal{O}(\varepsilon_{\mathsf{mach}}^2)$$

Which completes the proof. $\square$

Comments

This is the standard approach for a backward stability analysis.

Errors introduced by the floating point operations $\oplus$, $\ominus$, $\otimes$, and $\oslash$ (in accordance with the axiom) are **reinterpreted** as errors in the initial data / or "problem."

Where appropriate, errors $\sim \mathcal{O}(\varepsilon_{\text{mach}})$ are freely moved between numerators and denominators.

Perturbations of order $\mathcal{O}(\varepsilon_{\text{mach}})$ are accumulated additively, *e.g.*

$$(1+\epsilon_1)(1+\epsilon_2) = (1+2\epsilon_3) + \mathcal{O}(\varepsilon_{\text{mach}}^2)$$

where $|\epsilon_{1,2,3}| \leq \varepsilon_{\text{mach}}$.

Least Squares Problems

Next, we turn our attention back to least squares problems.

- We take a detailed look at the **conditioning** of least squares problems; it is a subtle topic and has nontrivial implications for the **stability** (and ultimately, the **accuracy**) of least squares algorithms.
- Further, this will serve as our main example on detailed conditioning analysis (as Back-substitution served as the main example on detailed backward stability analysis).