Lecture 22 - *Stability of Gaussian Elimination*

OBJECTIVE:
Gaussian elimination with partial pivoting is theoretically unstable!
But it is stable in practice!
This paradox has a statistical explanation.

◇ STABILITY AND THE NORM OF $\mathbf{L}$ AND $\mathbf{U}$

When we deal with orthogonal transformations, stability analysis is usually pretty straightforward.

Gaussian elimination with partial pivoting does <u>not</u> involve orthogonal transformations
$\rightarrow$ its stability analysis is quite complicated!

Recall from Lecture 20,

$$\mathbf{A} = \begin{bmatrix} 10^{-20} & 1 \\ 1 & 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 10^{20} & 1 \end{bmatrix} \begin{bmatrix} 10^{-20} & 1 \\ 0 & 1 - 10^{20} \end{bmatrix} = \mathbf{LU}$$

Note 1. $\mathbf{L}$ *has an entry of* 10^{20}
$\rightarrow$ *solving a linear system based on such an* $\mathbf{L}$ *has rounding errors magnified to* $10^{20}\epsilon_{\text{machine}}$, *leading to an inaccurate answer!*

It turns out this example is somewhat generic in the sense that instability in Gaussian elimination (with or without pivoting) arise if $\|\mathbf{L}\|$ and/or $\|\mathbf{U}\|$ is large relative to $\|\mathbf{A}\|$.

$\rightarrow$ the idea behind pivoting is to make sure $\|\mathbf{L}\|, \|\mathbf{U}\|$ are not too large!

Theorem 1. *Let* $\mathbf{A} \in \mathbb{R}^{m\times m}$ *be nonsingular.*
Compute $\mathbf{A} = \mathbf{LU}$ *by Gaussian elimination.*
Then the computed matrices $\tilde{\mathbf{L}}, \tilde{\mathbf{U}}$ *satisfy*

$$\tilde{\mathbf{L}}\tilde{\mathbf{U}} = \mathbf{A} + \delta\mathbf{A}, \qquad \frac{\|\delta\mathbf{A}\|}{\|\mathbf{L}\|\|\mathbf{U}\|} = \mathcal{O}(\epsilon_{\text{machine}})$$

for some $\delta\mathbf{A} \in \mathbb{R}^{m\times m}$.

Note 2. *To include partial pivoting, we can replace* $\mathbf{A}$ *with* $\mathbf{PA}$.

Note 3. *As usual, we make no claims about the smallness of* $\tilde{\mathbf{L}} - \mathbf{L}$ *or* $\tilde{\mathbf{U}} - \mathbf{U}$*; only about* $\tilde{\mathbf{L}}\tilde{\mathbf{U}} - \mathbf{L}\mathbf{U}$.

Note 4. *The key difference with this estimate is that* $\|\delta\mathbf{A}\|$ *is small compared to* $\|\mathbf{L}\|\|\mathbf{U}\|$, *not* $\|\mathbf{A}\|$.
If $\|\mathbf{L}\|\|\mathbf{U}\| = \mathcal{O}(\|\mathbf{A}\|)$, *then Gaussian elimination is backward stable.*
Otherwise, we must expect backward instability!

It is easy to construct $\mathbf{A}$ for which $\|\mathbf{L}\|, \|\mathbf{U}\|$ are unboundedly large when computed by pure Gaussian elimination.

$\rightarrow$ This algorithm is unstable; we will treat only Gaussian elimination with partial pivoting.

$\diamond$ GROWTH FACTORS

For Gaussian elimination with partial pivoting, the pivot element is chosen from maximization over a column
$\rightarrow$ the elements of $\mathbf{L}$ all satisfy $|l_{ij}| \leq 1$
$\rightarrow$ $\|\mathbf{L}\| = \mathcal{O}(1)$

$\therefore$ The condition $\frac{\|\delta \mathbf{A}\|}{\|\mathbf{L}\|\|\mathbf{U}\|} = \mathcal{O}(\epsilon_{\text{machine}})$ for backward stability of Gaussian elimination with partial pivoting reduces to $\frac{\|\delta \mathbf{A}\|}{\|\mathbf{U}\|} = \mathcal{O}(\epsilon_{\text{machine}})$.
So the algorithm is backward stable provided $\|\mathbf{U}\| = \mathcal{O}(\|\mathbf{A}\|)$.

Define the *growth factor* for $\mathbf{A}$ as

$$\rho = \frac{\max_{ij} |u_{ij}|}{\max_{i,j} |a_{ij}|}$$

If $\rho = \mathcal{O}(1)$, not much growth has happened while computing $\mathbf{U}$ from $\mathbf{A}$
$\rightarrow$ The elimination process is stable in this case.

But, if ρ is larger than this, we must expect some instability.

The definition of ρ implies $\|\mathbf{U}\| = \mathcal{O}(\rho\|\mathbf{A}\|)$ and leads to the following result.

Theorem 2. *Suppose we compute* $\mathbf{PA} = \mathbf{LU}$ *for a given* $\mathbf{A} \in \mathbb{R}^{m \times m}$.
Then the computed matrices $\tilde{\mathbf{P}}, \tilde{\mathbf{L}}$, *and* $\tilde{\mathbf{U}}$ *satisfy*

$$\tilde{\mathbf{L}}\tilde{\mathbf{U}} = \tilde{\mathbf{P}}\mathbf{A} + \delta\mathbf{A}$$

where $\frac{\|\delta\mathbf{A}\|}{\|\mathbf{A}\|} = \mathcal{O}(\rho\epsilon_{\text{machine}})$ *for some* $\delta\mathbf{A} \in \mathbb{R}^{m \times m}$.

If $|l_{ij}| < 1$ for each $i > j$ ($\leftrightarrow$ there are no ties in choosing the exact pivot), then $\tilde{\mathbf{P}} = \mathbf{P}$ for all sufficiently small $\epsilon_{\text{machine}}$.

In summary, Gaussian elimination with partial pivoting is backward stable if $\rho = \mathcal{O}(1)$ *uniformly* for all matrices of a given dimension m; otherwise it is not.

But, this statement needs qualification!

$\diamond$ WORST-CASE INSTABILITY

It is possible to construct matrices with large ρ — despite the beneficial effects of pivoting!

e.g., Consider

$$
\mathbf{A} = \begin{bmatrix} 1 & & & & 1 \\ -1 & 1 & & & 1 \\ -1 & -1 & 1 & & 1 \\ -1 & -1 & -1 & 1 & 1 \\ -1 & -1 & -1 & -1 & 1 \end{bmatrix}
$$

No pivoting actually takes place in reducing $\mathbf{A}$ to $\mathbf{U}$.

However, at each step, the entries in the last column (except for the first one of course) all double!

The end result is

$$
\begin{bmatrix} 1 & & & & 1 \\ & 1 & & & 2 \\ & & 1 & & 4 \\ & & & 1 & 8 \\ & & & & 16 \end{bmatrix} \qquad \textit{(verify!)}
$$

Thus for the 5×5 matrix, $\rho = 16$.

In general, for an $m \times m$ matrix of the same form, $\rho = 2^{m-1}$.

$\rightarrow$ It turns out this is the largest growth factor possible.

$\rho = 2^m \quad \rightarrow \quad$ m bits of precision are lost in computation

$\quad\quad\quad\quad \rightarrow \quad$ this is a disaster!

Typical computers represent floating-point numbers with only 64 bits.

So, if this worst-case growth factor was often met in practice, we would not be able to solve large systems of equations!

Here is where we have to refine our discussion of stability.

First we recap our results in terms of the classical concepts of stability.

Then we consider a more practical interpretation.

Theorem 3. *According to the discussion in Lecture 14, Gaussian elimination with partial pivoting is backward stable.*

This statement is true because we can find stability bound that applies to all matrices *for each fixed dimension* m.

(The bound involves the constant 2^{m-1}.)

The problem is that this bound is so large for practical values of m that if it was met on a regular basis, Theorem 22.2 would be useless in practice.

Bottom Line:	It is theoretically possible for Gaussian elimination with partial pivoting to be explosively unstable on certain "cooked -up" matrices. But, if we consider performance in practice, it is stable.

$\diamond$ STABILITY IN PRACTICE

Despite the theoretical existence of matrices that give Gaussian elimination with partial pivoting problems, the algorithm is stable in practice.

i.e., the funny matrices that produce the large factors $\mathbf{U}$ never seem to appear in real applications
$\rightarrow$ none have been seen in over 50 years of computing.

How can this be explained?

The answer seems to be that these special instability-causing matrices "never" arise in practice for statistical reasons.

We now give an idea as to how this could be. A full account is not even possible because the matter is not yet fully understood!

The description is given in terms of random matrices, with the argument being that if we observe something about the properties of random matrices, these observations can be transferred to real matrices.

Suppose we create a number of random matrices of dimension m with random entries from $\mathbf{N}(0,1)$ and normalized by $\sqrt{m}$.

In Matlab,

```
A = randn(m,m)/sqrt(m);
```

If you then factor and measure the growth rate ρ for these matrices, you will find a <u>very</u> small number of matrices for which $\rho > \sqrt{m}$.

$\rightarrow$ The probability that a matrix has a large ρ seems to decrease exponentially.

e.g., In Figure 22.2 (page 168) of the text, 3 million matrices were factored.
Even though the theoretical maximum value of ρ was $2^{32}-1$, the actual maximum recorded was 11.99.

If you pick a billion matrices at random, you will almost certainly not find one for which Gaussian elimination with partial pivoting is unstable.

$\diamond$ A PARTIAL EXPLANATION

If $\mathbf{PA} = \mathbf{LU}$, then $\mathbf{U} = \mathbf{L}^{-1}\mathbf{PA}$

$$\therefore \quad \rho \text{ large } \longleftrightarrow \mathbf{L}^{-1} \text{ large}$$

Now random $\mathbf{L}$ will have large $\mathbf{L}^{-1}$ (in fact, exponentially large with increasing m).

However, the $\mathbf{L}$'s produced from Gaussian elimination are far from random!

In fact, the entries $(\mathbf{L}^{-1})_{ij}$ are usually less than one in absolute value.

In Figure 22.3, we can see the results of the following experiment.

Consider a random matrix $\mathbf{A}$.

Compute its LU decomposition, then plot the entries of $\mathbf{L}^{-1}$, marking the ones that satisfy $|(\mathbf{L}^{-1})_{ij}| \geq 1$:

```
(spy(abs(inv(L)) >=1)
```

For a 128×128 matrix, we notice

$$\max_{ij} |(\mathbf{L}^{-1})_{ij}| = 2.67.$$

If we now randomize only the signs of off-diagonal entries, and repeat the experiment, we find

$$\max_{ij} |(\tilde{\mathbf{L}}^{-1})_{ij}| = 2.27 \times 10^4.$$

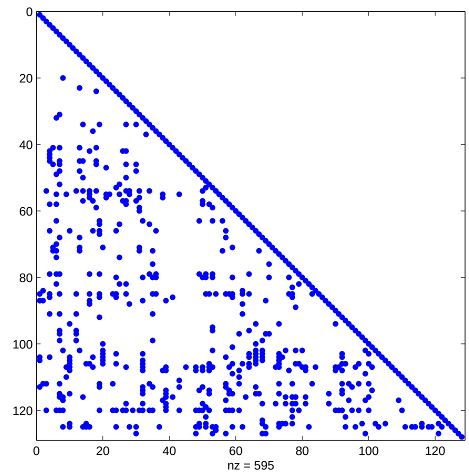

Figure 1: random $\mathbf{A}$; $\max_{i,j} |(\mathbf{L}^{-1})_{ij}| = 2.8623$.

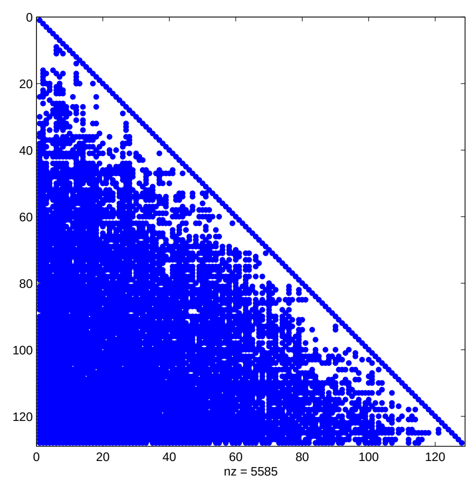

Figure 2: random $\tilde{\mathbf{L}}$; $\max_{i,j} |(\tilde{\mathbf{L}}^{-1})_{ij}| = 1.2189 \times 10^3$.

So why do the matrices $\mathbf{L}$ from Gaussian elimination almost never have large inverses?

We can see the answer by looking at column spaces.

Because $\mathbf{PA} = \mathbf{LU}$ and $\mathbf{U}$ is upper-triangular, the column spaces of $\mathbf{PA}$ and $\mathbf{L}$ are the same. *(verify!)*

i.e., the first column of $\mathbf{PA}$ spans the same space as the first column of $\mathbf{L}$, the first 2 columns of $\mathbf{PA}$ span the same space as the first 2 columns of $\mathbf{L}$, etc.

If $\mathbf{A}$ is random, so is the orientation of its column spaces
$\rightarrow$ the same must be true of $\mathbf{P}^{-1}\mathbf{L}$.

But, it can then be shown that if $\|\mathbf{P}^{-1}\mathbf{L}\|$ is large, then the column spaces of $\mathbf{P}^{-1}\mathbf{L}$ (or equivalently the column spaces of $\mathbf{L}$ since $\mathbf{P}^{-1}$ is just a permutation) must be skewed in a fashion that is far from normal.

i.e., $\mathbf{L}^{-1}$ will <u>not</u> have random orientation of column space if $\|\mathbf{P}^{-1}\mathbf{L}\|$ is large
$\leftrightarrow$ $\|\mathbf{P}^{-1}\mathbf{L}\|$ will not be large unless $\mathbf{A}$ has a special skewed orientation of its column spaces

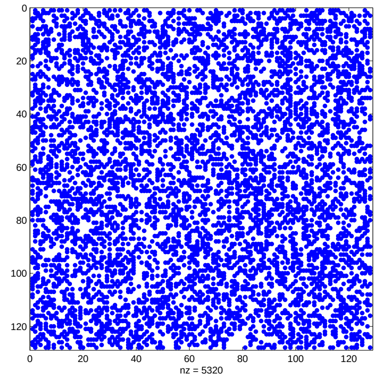

Figure 3: Q portrait of random $\mathbf{A}$.

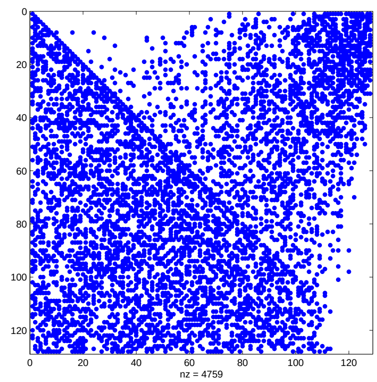

Figure 4: Q portrait of random $\tilde{\mathbf{L}}$.

In summary, Gaussian elimination with partial pivoting is highly unstable for certain special matrices.

For this instability to occur, the column space of $\mathbf{A}$ must be skewed in a very special way. Fortunately, such matrices seem to be extremely rare in applications.