Pivoting and the LU factorization

MATH 6610 Lecture 16

October 14, 2020

Trefethen & Bau: Lecture 21

LU and Gaussian elimination

Let $A \in \mathbb{C}^{n \times n}$ be an invertible matrix. Recall that, if Gaussian elimination succeeds, then

$$A = LU,$$

where L and U are lower- and upper-triangular, respectively.

LU and Gaussian elimination

Let $A \in \mathbb{C}^{n\times n}$ be an invertible matrix. Recall that, if Gaussian elimination succeeds, then

$$A = LU,$$

where L and U are lower- and upper-triangular, respectively.

"Standard" Gaussian elimination fails in some cases, e.g., with

$$A = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}.$$

Pivoting, I

The standard approach to "fixing" this problem is pivoting, which interchanges rows and/or columns.

(For previous example: pivot rows)

$$\begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \xrightarrow{\text{pivot}} \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$$

Pivoting, I

The standard approach to "fixing" this problem is pivoting, which interchanges rows and/or columns.

We know pivoting by another name: permutations.

$P = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$ (is a permutation)

$$P \underbrace{\begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}}_{A} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$$

Pivoting, I

The standard approach to "fixing" this problem is pivoting, which interchanges rows and/or columns.

We know pivoting by another name: permutations.

General pivoting strategy: permute rows so that diagonal elements during elimination are non-zero.
(For stability, pivot so that diagonal elements have maximum magnitude.)

Preliminaries: $P(j,k) \in \mathbb{C}^{n \times n}$ is a permutation matrix corresponding to the permutation σ defined as $\sigma(\ell) = \begin{cases} k & \text{if } \ell = j \\ j & \text{if } \ell = k \\ \ell & \text{otherwise} \end{cases}$

I.e., $P(j,k)$ interchanges elements j & k.

$$A = \begin{pmatrix} a_{1,1} & a_{2,1} & \cdots & a_{n,1} \\ a_{1,2} & \vdots & & \vdots \\ \vdots & \vdots & & \vdots \\ a_{1,n} & a_{2,n} & \cdots & a_{n,n} \end{pmatrix}$$

Even if $a_{1,1} \neq 0$, perform pivoting.

Define $j_1 := \underset{k \in \{1,\dots,n\}}{\operatorname{argmax}} |a_{1,k}|$ (row index of first column with max magnitude)

(Note: $\cancel{j_1 \neq 0}$ if A is invertible)

$a_{1,j_1} \neq 0$

Then:

$$P(1,j_1)\,A = \begin{array}{r} \\ \\ \\ \\ \\ \text{row } j_1 \rightarrow \\ \\ \\ \\ \end{array} \begin{pmatrix} a_{1,j_1} & \cdots & a_{n,j_1} \\ a_{1,2} & & \vdots \\ a_{1,3} & & \vdots \\ \vdots & & \vdots \\ a_{1,j_1-1} & & a_{n,j_1-1} \\ a_{1,1} & & a_{n,1} \\ a_{1,j_1+1} & & \vdots \\ \vdots & & \vdots \\ a_{1,n} & \cdots & a_{n,n} \end{pmatrix}$$

if $a_{1,j_1} \neq 0$: perform elimination in column 1

$\exists\, L_1^{-1}$, lower-triangular matrix s.t.

$$A_2 = L_1^{-1} P(1, j_1) A = \begin{pmatrix} X & | & | & & | \\ 0 & & & & \\ 0 & a_2^{(2)} & a_3^{(2)} & \cdots & a_n^{(2)} \\ \vdots & & & & \\ 0 & | & | & & | \end{pmatrix}$$

$$\Longrightarrow A = \underbrace{P(1, j_1)}_{P_1} L_1 A_2 =: P_1 L_1 A_2$$

Repeat elimination for A_2: $j_2 := \underset{k \in \{2, \ldots n\}}{\operatorname{argmax}} |a_{2,k}^{(2)}|$

Pivot row j_2 up to row 2, and perform elimination.

$$A_3 = L_2^{-1} P(2, j_2) A_2 = \begin{pmatrix} X & X & | & & | \\ 0 & X & & & \\ \vdots & 0 & a_3^{(3)} & \cdots & a_n^{(3)} \\ \vdots & \vdots & & & \\ 0 & 0 & | & & | \end{pmatrix}$$

$$\Longrightarrow A_2 = P_2 L_2 A_2 \,, \quad P_2 := P(2, j_2)$$

At step $m \le n-1$, pivot based on column m:

$$j_m := \underset{k \in \{m, \dots, n\}}{\operatorname{argmax}} |a_{m,k}^{(m)}|$$

Eliminate: $A_{m+1} = L_m^{-1} P(m, j_m) A_m$

$$\Longrightarrow A_m = P_m L_m A_{m+1}, \quad P_m := P(m, j_m)$$

$$\Longrightarrow A = P_1 L_1 A_2 = P_1 L_1 (P_2 L_2 A_3)$$

$$= P_1 L_1 P_2 L_2 (P_3 L_3 A_4)$$

$$\vdots$$

$$= (P_1 L_1) \cdots (P_{n-1} L_{n-1}) A_n$$

upper-triangular.

Pivoting, I

The standard approach to “fixing” this problem is pivoting, which interchanges rows and/or columns.

We know pivoting by another name: permutations.

General pivoting strategy: permute rows so that diagonal elements during elimination are non-zero.
(For stability, pivot so that diagonal elements have maximum magnitude.)

This results in the decomposition, $P_j L_j$: elimination on step j.

$$A = P_1 L_1 \, P_2 L_2 \, \cdots \, P_{n-1} L_{n-1} U, \quad (\star).$$

where P_j is a permutation matrix that permutes row j with row k for some $k \geqslant j$.

Note: P_j is not (generally) lower-triangular.

Pivoting, II

$$A = P_1 L_1 \; P_2 L_2 \; \cdots \; P_{n-1} L_{n-1} U,$$

where P_j is a permutation matrix that permutes row j with row k for some $k \geqslant j$.

Pivoting, II

$$A = P_1 L_1 \; P_2 L_2 \; \cdots \; P_{n-1} L_{n-1} U,$$

where P_j is a permutation matrix that permutes row j with row k for some $k \geqslant j$.

$\star$

One can show that $\left(L_j P_k = P_k \widetilde{L}_j \right)$ if $j < k$ for some other lower-triangular matrix $\widetilde{L}_j$, so that

$$A = \left(\prod_{j=1}^{n-1} P_j \right) \left(\prod_{j=1}^{n-1} \widetilde{L}_j \right) U.$$

E.g. $P_1 L_1 P_2 L_2 = P_1 P_2 \widetilde{L}_1 L_2$

How to show $\star$? Direct computation, using structure of L_j & P_k.

Recall: (i) $\prod_{j=1}^{n-1} \tilde{L}_j \leftarrow$ lower triangular

(ii) $\prod_{j=1}^{n-1} P_j \leftarrow$ a permutation matrix.

$\Rightarrow \left(\prod_{j=1}^{n-1} P_j\right)^{-1}$ is a permutation matrix.

Pivoted LU

In fact, we can show that this *row pivoting* strategy always works.

if A is invertible.

Theorem

If $A \in \mathbb{C}^{n\times n}$ is invertible, then there exists

- *a permutation matrix P,*
- *a lower-triangular matrix L,*
- *an upper-triangular matrix U,*

such that

$$\underline{PA = LU}$$

This row pivoting strategy is called "partial" pivoting.

[This is actually how linear systems are solved.]

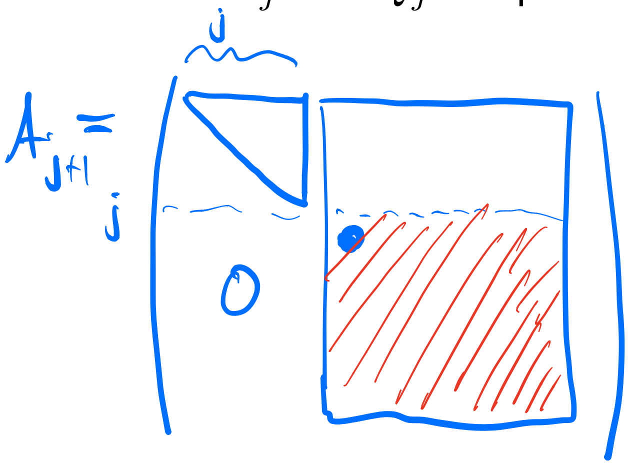
More pivoting

Row pivoting is not the only option.

For example, *full* pivoting permutes *both* lower rows and rightmost columns in search of a maximum-magnitude pivot.

$$A = P_1 L_1 \; P_2 L_2 \; \cdots \; P_{n-1} L_{n-1} \; U \; Q_{n-1} Q_{n-2} \cdots Q_1,$$

where both P_j and Q_j are permutation matrices.



•: pivot location

///: elements of A_{j+1} that are analyzed to find maximum magnitude element.

matrices Q_j: column permutation matrices

More pivoting

Row pivoting is not the only option.

For example, *full* pivoting permutes *both* lower rows and rightmost columns in search of a maximum-magnitude pivot.

$$A = P_1 L_1 \, P_2 L_2 \, \cdots \, P_{n-1} L_{n-1} \, U \, Q_{n-1} Q_{n-2} \cdots Q_1,$$

where both P_j and Q_j are permutation matrices.

This achieves the full-pivoted LU decomposition,

$$PAQ = LU. \quad (\star)$$

Generally, full-pivoting LU is more stable than partial pivoting. But this is also more expensive.

More pivoting

Row pivoting is not the only option.

For example, *full* pivoting permutes *both* lower rows and rightmost columns in search of a maximum-magnitude pivot.

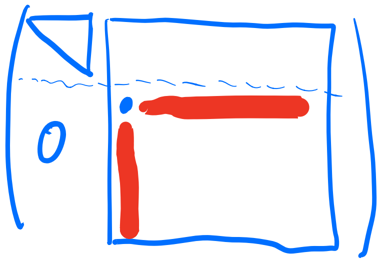
$$A = P_1 L_1 \, P_2 L_2 \, \cdots \, P_{n-1} L_{n-1} \, U \, Q_{n-1} Q_{n-2} \cdots Q_1,$$

where both P_j and Q_j are permutation matrices.

This achieves the full-pivoted LU decomposition,

$$PAQ = LU.$$

An alternative is *rook pivoting*, which performs a permutation similar to the above, except that at elimination step j, the maximum is sought *only* over row j and column j.

$A_{k+1} =$ 

• pivot location

•: locations over which to search for maximum magnitude element.

So: rook pivoting is a compromise between full and partial pivoting.

"Theorem": LU decompositions with partial pivoting are <u>not</u> stable.

But: this instability doesn't happen in practice.