LU Factorization

Square Linear Systems

$$A\vec{x} = \vec{b}$$

where $A \in \mathbb{R}^{n\times n}$, $\vec{b} \in \mathbb{R}^n$ are given.

- Polynomial interpolation

Last time

- "Simple" systems (triangular)

$$\begin{cases} U\vec{x} = \vec{y} & : \text{Backward subs.} \\ L\vec{x} = \vec{y} & : \text{Forward elim.} \end{cases}$$

Today

- General matrix A.

Contents

Gaussian Elimination

General Method: Gaussian Elimination

- *Gaussian elimination* is an algorithm for solving a general system of linear equations that involves a sequence of row operations performed on the associated matrix of coefficients. (preserve the solution of given system)
- This is also known as the method of row reduction.
- There are three variations to this method:
  - G.E. without pivoting
  - G.E. with partial pivoting (that is, row pivoting)
  - G.E. with full pivoting (that is, row and column pivoting)

To solve $A\vec{x} = \vec{b}$:

$$S = \underbrace{[\,A \mid \vec{b}\,]}_{\substack{n\times(n+1) \\ \text{augmented matrix}}} \xrightarrow{\text{row ops.}} \underbrace{[\,\underbrace{U}_{\text{upper-}\Delta} \mid \tilde{\beta}\,]}_{\text{echelon form}} \longrightarrow \underbrace{[\,I \mid \tilde{\beta}\,]}_{\text{reduced-echelon form}}$$

the soln.

What are the allowed row operations in general G.E.?

---

* ① Row interchange (or swap) : $R_i \leftrightarrow R_j$

② Row scaling : $R_i \to cR_i$

* ③ Row replacement : $R_i \to R_i + cR_j$

G.E. Without Pivoting: Example

Key Example

Solve the following system of equations.

$$\begin{cases} 2x_1 + 2x_2 + x_3 = 6 \\ -4x_1 + 6x_2 + x_3 = -8 \\ 5x_1 - 5x_2 + 3x_3 = 4 \end{cases} \xrightarrow{\text{matrix equation}} \underbrace{\begin{bmatrix} 2 & 2 & 1 \\ -4 & 6 & 1 \\ 5 & -5 & 3 \end{bmatrix}}_{A} \underbrace{\begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix}}_{\mathbf{x}} = \underbrace{\begin{bmatrix} 6 \\ -8 \\ 4 \end{bmatrix}}_{\mathbf{b}}$$

Want: $-4 + \left(\frac{4}{2}\right) 2 = 0$

Step 1: Write the corresponding *augmented matrix* and row-reduce to an echelon form.

$R_2 \to R_2 + \left(\frac{4}{2}\right) R_1$

$$\left[\begin{array}{ccc|c} 2 & 2 & 1 & 6 \\ -4 & 6 & 1 & -8 \\ 5 & -5 & 3 & 4 \end{array}\right] \to \left[\begin{array}{ccc|c} 2 & 2 & 1 & 6 \\ 0 & 10 & 3 & 4 \\ 0 & 10 & -0.5 & 11 \end{array}\right] \to \left[\begin{array}{ccc|c} 2 & 2 & 1 & 6 \\ 0 & 10 & 3 & 4 \\ 0 & 0 & 3.5 & -7 \end{array}\right].$$

$R_3 \to R_3 + \left(\frac{-5}{2}\right) R_1$

U $\qquad \vec{\beta}$

Step 2: Solve for x_3, then x_2, and then x_1 via *backward substitution.*

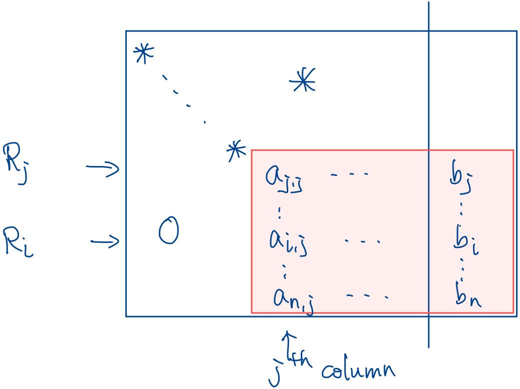
$$\mathbf{x} = (3, 1, -2)^{\mathrm{T}}.$$

$U \vec{x} = \vec{\beta}$

Reduction to Echelon Form

- Only allowed row operation : row replacement ($R_i \to R_i + cR_j$)

- Let $j \in \mathbb{N}[1, n-1]$. Suppose first $(j-1)$ columns have been "worked out".



$$R_i \longrightarrow R_i + \boxed{\left(-\frac{a_{i,j}}{a_{j,j}}\right)} R_j$$

Want: $0 = a_{i,j} + \left(\frac{-a_{i,j}}{a_{j,j}}\right) a_{j,j}$

multiplier

G.E. without Pivoting: General Procedure

As shown in the example, G.E. without pivoting involves two steps:

1. **Row reduction:** Transform $A\mathbf{x} = \mathbf{b}$ to $U\mathbf{x} = \boldsymbol{\beta}$ where

$$U = \begin{bmatrix} u_{11} & u_{12} & \cdots & u_{1n} \\ & u_{22} & \cdots & u_{2n} \\ & & \ddots & \vdots \\ \mathbf{0} & & & u_{nn} \end{bmatrix} \quad \text{and} \quad \boldsymbol{\beta} = \begin{bmatrix} \beta_1 \\ \beta_2 \\ \vdots \\ \beta_n \end{bmatrix}.$$

2. **Backward substitution:** Solve $U\mathbf{x} = \boldsymbol{\beta}$ for $\mathbf{x}$ by

$$\begin{cases} x_n = \dfrac{\beta_n}{u_{nn}} \quad \text{and} \\ x_i = \dfrac{1}{u_{ii}} \left(\beta_i - \displaystyle\sum_{j=i+1}^{n} u_{ij} x_j \right), \quad \text{for } i = n-1, n-2, \ldots, 1\,. \end{cases}$$

G.E. without Pivoting: MATLAB Implementation

Convention

i : row index

j : column index

```
1  function x = GEnp(A, b)
2    % Step 1: Row reduction to upper tri. system
3    S = [A, b];              % augmented matrix
4    n = size(A, 1);          % or n = length(A) or n = length(b)
5    for j = 1:n-1
6      for i = j+1:n;
7        mult = -S(i,j)/S(j,j);
8        S(i,:) = S(i,:) + mult*S(j,:);
9      end
10   end
11   % Step 2: Backward substitution
12   U = S(:,1:end-1);
13   beta = S(:,end);
14   x = backsub(U, beta);
15 end
```

% $R_i \to R_i + \left(-\dfrac{a_{i,j}}{a_{j,j}}\right) R_j$

$[A \mid \vec{b}\,] \longrightarrow [U \mid \vec{\beta}\,]$

Code from last lecture

Exercise. Rewrite Lines 6–9 without using a loop. (Think *vectorized*!)

G.E. with Partial Pivoting: Procedure

row interchange

Allowed ops: $\begin{cases} R_i \leftrightarrow R_j \\ R_i \to R_i + cR_j \end{cases}$

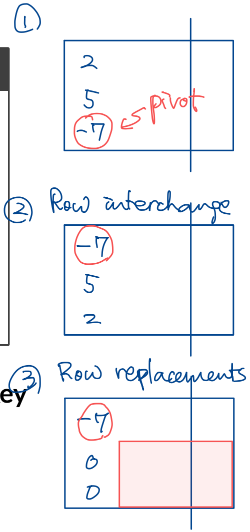
In this variation of G.E., reduction to echelon form is done slightly differently.

- On the augmented matrix $[A \mid \mathbf{b}]$,

Key Process (partial pivoting)

1. Find the entry in the first column with the largest absolute value. This entry is called the *pivot*.
2. Perform a row interchange, if necessary, so that the pivot is on the first diagonal position.
3. Use elementary row operations to reduce the remaining entries in the first column to zero.

- Once done, ignore the first row and first column and repeat the **Key Process** on the remaining submatrix.
- Continue this until the matrix is in a row-echelon form.



G.E. with Partial Pivoting: Example

Let's solve the example on p. 5 again, now using G.E. with partial pivoting.

1st column:

$$\left[\begin{array}{rrr|r} 2 & 2 & 1 & 6 \\ -4 & 6 & 1 & -8 \\ 5 & -5 & 3 & 4 \end{array}\right] \xrightarrow{\text{pivot}} \left[\begin{array}{rrr|r} 5 & -5 & 3 & 4 \\ -4 & 6 & 1 & -8 \\ 2 & 2 & 1 & 6 \end{array}\right] \xrightarrow{\text{zero}} \left[\begin{array}{rrr|r} 5 & -5 & 3 & 4 \\ 0 & 2 & 3.4 & -4.8 \\ 0 & 4 & -0.2 & 4.4 \end{array}\right]$$

2nd column:

$$\left[\begin{array}{rrr|r} 5 & -5 & 3 & 4 \\ 0 & 2 & 3.4 & -4.8 \\ 0 & 4 & -0.2 & 4.4 \end{array}\right] \xrightarrow{\text{pivot}} \left[\begin{array}{rrr|r} 5 & -5 & 3 & 4 \\ 0 & 4 & -0.2 & 4.4 \\ 0 & 2 & 3.4 & -4.8 \end{array}\right] \xrightarrow{\text{zero}} \left[\begin{array}{rrr|r} 5 & -5 & 3 & 4 \\ 0 & 4 & -0.2 & 4.4 \\ 0 & 0 & 3.5 & -7 \end{array}\right]$$

Now that the last matrix is upper triangular, we work up from the third equation to the second to the first and obtain the same solution as before.

G.E. with Partial Pivoting: MATLAB Implementation

HW

Exercise

Write a MATLAB function `GEpp.m` which carries out G.E. with partial pivoting.

- Modify `GEnp.m` on p. 7 to incorporate partial pivoting.
- The only part that needs to be changed is the for-loop starting at Line 5.
  - Right after `for j = 1:n-1`, find the index of the pivot element of the jth column of A below the diagonal.

```
[~, iM] = max(abs(A(j:end,j)));
iM = iM + j - 1;
```

  - If the pivot element is not on the diagonal, swap rows so that it is on the diagonal.

```
if j ~= iM
    S([j iM], :) = S([iM j], :)
end
```

Why Is Pivoting Necessary?

Example

Given $\underbrace{\epsilon \ll 1}$, solve the system

ϵ is a tiny positive #.

$$\begin{bmatrix} -\epsilon & 1 \\ 1 & -1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} 1-\epsilon \\ 0 \end{bmatrix}$$

Ans. $\begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \end{bmatrix}$

using Gaussian elimination with and <u>without partial pivoting</u>.

Without pivoting: By $R_2 \to R_2 + (1/\epsilon)R_1$, we have

$$\begin{bmatrix} -\epsilon & 1 \\ 0 & -1+1/\epsilon \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} 1-\epsilon \\ 1/\epsilon - 1 \end{bmatrix} \xrightarrow{\text{back subs.}} \begin{cases} x_2 = 1, \\ x_1 = \dfrac{(1-\epsilon)-1}{-\epsilon}. \end{cases} = \frac{-\epsilon}{-\epsilon} = 1$$

- In exact arithmetic, this yields the correct solution.
- In floating-point arithmetic, calculation of x_1 suffers from catastrophic cancellation.

Since ϵ is small, $1-\epsilon \approx 1$.

Why Is Pivoting Necessary? (cont')

Example

Given $\epsilon \ll 1$, solve the system

$$\begin{bmatrix} -\epsilon & 1 \\ 1 & -1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} 1-\epsilon \\ 0 \end{bmatrix}$$

using Gaussian elimination with and without partial pivoting.

With partial pivoting: First, swap the rows $R_1 \leftrightarrow R_2$, and then do $R_2 \to R_2 + \epsilon R_1$ to obtain

$$\begin{bmatrix} 1 & -1 \\ 0 & 1-\epsilon \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} 0 \\ 1-\epsilon \end{bmatrix} \implies \begin{cases} x_2 = 1, \\ x_1 = \dfrac{0-(-1)}{1}. \end{cases}$$

- Each of the arithmetic steps (to compute x_1, x_2) is well-conditioned.
- The solution is computed stably.

LU Factorization

Emulation of Gaussian Elimination

In this section, we emulate row operations steps required in Gaussian elimination by matrix multiplications. **Two major operations.**

- Row interchange $R_i \leftrightarrow R_j$:

the matrix obtained by interchanging i^{th} and j^{th} row of I.

$$P(i,j)A, \quad \text{where } P(i,j) \text{ is an elementary permutation matrix.}$$

- Row replacement $R_i \to R_i + cR_j$:

$$\underbrace{(I + c\mathbf{e}_i\mathbf{e}_j^{\mathrm{T}})}_{\text{Gaussian transformation matrix.}}A$$

See *Notes on Row and Column Operations* for more details.

Key Example Revisited

Let's work out the key example from last time once again, now in matrix form $A\mathbf{x} = \mathbf{b}$.

$$\begin{bmatrix} 2 & 2 & 1 \\ -4 & 6 & 1 \\ 5 & -5 & 3 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} 6 \\ -8 \\ 4 \end{bmatrix}.$$

[Pivot] Switch R_1 and R_3 using $P(1,3)$:

$$\underbrace{\begin{bmatrix} 0 & 0 & 1 \\ 0 & 1 & 0 \\ 1 & 0 & 0 \end{bmatrix}}_{P(1,3)} \left[\begin{bmatrix} 2 & 2 & 1 \\ -4 & 6 & 1 \\ 5 & -5 & 3 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} 6 \\ -8 \\ 4 \end{bmatrix} \right] \longrightarrow \begin{bmatrix} 5 & -5 & 3 \\ -4 & 6 & 1 \\ 2 & 2 & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} 4 \\ -8 \\ 6 \end{bmatrix}$$

[Zero] Do row operations $R_2 \to R_2 + (4/5)R_1$ and $R_3 \to R_3 - (2/5)R_1$:

$$\underbrace{\begin{bmatrix} 1 & 0 & 0 \\ 4/5 & 1 & 0 \\ -2/5 & 0 & 1 \end{bmatrix}}_{G_1} \left[\begin{bmatrix} 5 & -5 & 3 \\ -4 & 6 & 1 \\ 2 & 2 & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} 4 \\ -8 \\ 6 \end{bmatrix} \right] \longrightarrow \begin{bmatrix} 5 & -5 & 3 \\ 0 & 2 & 3.4 \\ 0 & 4 & -0.2 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} 4 \\ -4.8 \\ 4.4 \end{bmatrix}$$

Key Example Revisited (cont')

[Pivot] Switch R_2 and R_3 using $P(2,3)$:

$$\underbrace{\begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & 0 \end{bmatrix}}_{P(2,3)} \left[\begin{bmatrix} 5 & -5 & 3 \\ 0 & 2 & 3.4 \\ 0 & 4 & -0.2 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} 4 \\ -4.8 \\ 4.4 \end{bmatrix} \right] \longrightarrow \begin{bmatrix} 5 & -5 & 3 \\ 0 & 4 & -0.2 \\ 0 & 2 & 3.4 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} 4 \\ 4.4 \\ -4.8 \end{bmatrix}$$

[Zero] Do a row operation $R_3 \to R_3 - (1/2)R_2$:

$$\underbrace{\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & -1/2 & 1 \end{bmatrix}}_{G_2} \left[\begin{bmatrix} 5 & -5 & 3 \\ 0 & 4 & -0.2 \\ 0 & 2 & 3.4 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} 4 \\ 4.4 \\ -4.8 \end{bmatrix} \right] \longrightarrow \underbrace{\begin{bmatrix} 5 & -5 & 3 \\ 0 & 4 & -0.2 \\ 0 & 0 & 3.5 \end{bmatrix}}_{U} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} 4 \\ 4.4 \\ -7 \end{bmatrix}$$

Analysis of Example

- The previous calculations can be summarized as

$$G_2P(2,3)G_1P(1,3)A = U\,. \tag{$\star$}$$

- Using the noted properties of permutation matrices and GTMs, ($\star$) can be written as

$$G_2P(2,3)G_1\underbrace{P(2,3)P(2,3)}_{=I}P(1,3)A = U$$
$$\longrightarrow G_2\underbrace{P(2,3)G_1P(2,3)}_{=:\widetilde{G}_1}\underbrace{P(2,3)P(1,3)}_{=:P}A = U\,.$$

- The above can be summarized as $PA = LU$ where $L = (G_2\widetilde{G}_1)^{-1}$ is a lower triangular matrix.

When $n=4$

$$G_3 P_3 G_2 P_2 G_1 P_1 A = U$$

Generalization – PLU Factorization

For an arbitrary matrix $A \in \mathbb{R}^{n\times n}$, the partial pivoting and row operations are intermixed as

$$G_{n-1}P(n-1, r_{n-1})\cdots G_2P(2,r_2)G_1P(1,r_1)A = U\,.$$

Going through the same calculations as above, it can always be written as

$$\left(\widetilde{G}_{n-1}\cdots\widetilde{G}_2\widetilde{G}_1\right)P(n-1,r_{n-1})\cdots P(2,r_2)P(1,r_1)A = U\,,$$

which again leads to $PA = LU$:

$$\underbrace{P(n-1,r_{n-1})\cdots P(2,r_2)P(1,r_1)}_{=:P}A = \underbrace{\left(\widetilde{G}_{n-1}\cdots\widetilde{G}_2\widetilde{G}_1\right)^{-1}}_{=:L}U\,.$$

This is called the **PLU factorization** of matrix A.

LU and PLU Factorization

If no pivoting is required, the previous procedure simplifies to

$$G_{n-1} \cdots G_2 G_1 A = U \,.$$

which leads to $A = LU$:

$$A = \underbrace{(G_{n-1} \cdots G_2 G_1)^{-1}}_{=:L} U \,.$$

This is called the **LU factorization** of matrix A.

Implementation of LU Factorization

```
function [L,U] = mylu(A)
% MYLU   LU factorization (demo only--not stable!).
% Input:
%   A    square matrix
% Output:
%   L,U  unit lower triangular and upper triangular such that
    LU=A
  n = length(A);
  L = eye(n);    % ones on diagonal
  % Gaussian elimination
  for j = 1:n-1
    for i = j+1:n
      L(i,j) = A(i,j) / A(j,j);    % row multiplier
      A(i,j:n) = A(i,j:n) - L(i,j)*A(j,j:n);
    end
  end
  U = triu(A);
end
```

Implementation of LU Factorization

Exercise. Write a MATLAB function `myplu` for PLU factorization by modifying the previous function `mylu.m`.

```
function [L,U,P] = myplu(A)
% MYPLU   PLU factorization (demo only--not stable!).
% Input:
%   A     square matrix
% Output:
%   P,L,U  permutation, unit lower triangular, and upper
    triangular such that LU=PA

% Your code here.




end
```