MA580H Matrix Computations

Lectures 5 & 6: System of Linear Equations-I

Rafikul Alam
Department of Mathematics
IIT Guwahati

Outline

- Solution of triangular system
- Gaussian elimination
- LU decomposition

Linear system

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

Problem: Solve $Ax = b$ for $x \in \mathbb{R}^n$.

Idea: For a nonsingular M, the solution of $MAx = Mb$ is given by

$$x = (MA)^{-1}Mb = A^{-1}M^{-1}Mb = A^{-1}b.$$

So, the strategy is to choose M so that the system

$$MAx = Mb$$

is easy to solve. Gaussian elimination provides such an M for which MA is upper triangular.

The MATLAB command

```
>> x= A\b
```

solves the system $Ax = b$ using Gaussian elimination.

Lower triangular linear system

Consider the lower triangular linear system of equations

$$\begin{bmatrix} \ell_{11} & & & \\ \ell_{21} & \ell_{22} & & \\ \vdots & \vdots & \ddots & \\ \ell_{n1} & \ell_{n2} & \cdots & \ell_{nn} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix} = \begin{bmatrix} b_1 \\ b_2 \\ \vdots \\ b_n \end{bmatrix}.$$

By forward substitution, we have

$$\begin{aligned} x_1 &= b_1/\ell_{11} \\ x_i &= \left(b_i - \textstyle\sum_{j=1}^{i-1} \ell_{ij} x_j\right)/\ell_{jj}, \quad i = 2:n. \end{aligned}$$

Cost: n^2 flops

Indeed, $\sum_{i=1}^{n} 2i = \int_0^n 2x dx +$ lower order terms $\simeq n^2$.

Column-oriented forward substitution

Writing $Lx = b$ as

$$\mathtt{L}(:,1)\mathtt{x}(1) + \cdots + \mathtt{L}(:,\mathtt{n})\mathtt{x}(\mathtt{n}) = \mathtt{b}$$

we obtain column-oriented forward substitution.

```
x = zeros(n,1);
for j=1:n-1
    x(j) =b(j)/L(j,j);
    b(j+1:n) = b(j+1:n)-L(j+1:n,j)*x(j);
end
x(n) = b(n)/L(n,n);
```

- Solving a lower triangular system costs n^2 flops.

Upper triangular linear system

Consider the upper triangular system

$$\begin{bmatrix} u_{11} & u_{12} & \cdots & u_{1n} \\ & u_{22} & \cdots & u_{2n} \\ & & \ddots & \vdots \\ & & & u_{nn} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix} = \begin{bmatrix} b_1 \\ b_2 \\ \vdots \\ b_n \end{bmatrix}.$$

If $u_{11}, \ldots, u_{nn}$ are nonzero, then by back substitution, we have a unique solution

$$\begin{aligned} x_n &= b_n / u_{nn} \\ x_i &= \left(b_i - \textstyle\sum_{j=i+1}^{n} u_{ij} x_j \right) / u_{jj}, \quad i = n-1, \ldots, 1. \end{aligned}$$

Cost: An upper triangular system is solved by back substitution and costs n^2 flops.

Gaussian elimination

Strategy: Transform a given linear system $Ax = b$ to an equivalent triangular linear system $\widehat{A}x = \widehat{b}$. Consider the system

$$\begin{array}{r} x - y - z = 2 \\ 3x - 3y + 2z = 16 \\ 2x - y + z = 9 \end{array} \iff \underbrace{\left[\begin{array}{rrr|r} 1 & -1 & -1 & 2 \\ 3 & -3 & 2 & 16 \\ 2 & -1 & 1 & 9 \end{array}\right]}_{\text{augmented matrix}}$$

Use first equation to eliminating x from 2nd and 3rd equation

$$\begin{array}{r} x - y - z = 2 \\ 5z = 10 \\ y + 3z = 5 \end{array} \iff \left[\begin{array}{rrr|r} 1 & -1 & -1 & 2 \\ 0 & 0 & 5 & 10 \\ 0 & 1 & 3 & 5 \end{array}\right].$$

Now interchange 2nd and 3rd equations

$$\begin{array}{r} x - y - z = 2 \\ y + 3z = 5 \\ 5z = 10 \end{array} \iff \left[\begin{array}{rrr|r} 1 & -1 & -1 & 2 \\ 0 & 1 & 3 & 5 \\ 0 & 0 & 5 & 10 \end{array}\right] \Rightarrow \begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} 3 \\ -1 \\ 2 \end{bmatrix}.$$

Gaussian elimination

```
function x=gauss(A,b)
% x=gauss(A,b) solves the linear system Ax=b using Gaussian
% elimination with partial pivoting on [A, b].
n=length(b);
norma=norm(A,1);
A=[A,b]; % augmented matrix
for i=1:n
[maxpv,kmax]=max(abs(A(i:n,i))); % look for Pivot A(kmax,i)
kmax=kmax+i-1;
if maxpv < 1e-14*norma; % only small pivots
error('matrix is singular')
end
if i ~= kmax % interchange rows
A([i, kmax],:) = A([kmax, i], :);
end
A(i+1:n,i)=A(i+1:n,i)/A(i,i); % elimination step
A(i+1:n,i+1:n+1)=A(i+1:n,i+1:n+1)-A(i+1:n,i)*A(i,i+1:n+1);
end
x=backsubs(A,A(:,n+1));
```

Gaussian elimination (GE)

Gaussian elimination can be rewritten as a method that factorizes a matrix. We consider three variants of GE. These variants yield three matrix factorizations, namely,

- LU factorization: $\mathrm{A} = \mathrm{LU}$
- Row permuted LU factorization: $\mathrm{PA} = \mathrm{LU}$
- Row and column permuted LU factorization: $\mathrm{PAQ} = \mathrm{LU}$

Here P and Q are permutation matrices. An $n \times n$ permutation matrix is obtained by permuting rows of the identity matrix I_n.

The matrix L is unit lower triangular and U is upper triangular. A lower triangular matrix L is called unit lower triangular if the diagonal entries of L are 1, that is, $\ell_{jj} = 1$ for $j = 1 : n$.

LU Decomposition

Definition: An LU decomposition of a matrix $A \in \mathbb{R}^{n\times n}$ is a factorization of the form $A = LU$, where L is unit lower triangular and U is upper triangular. Thus

$$A = \begin{bmatrix} a_{11} & \cdots & a_{1n} \\ \vdots & \ddots & \vdots \\ a_{n1} & \cdots & a_{nn} \end{bmatrix} = \begin{bmatrix} 1 & & \\ \vdots & \ddots & \\ \ell_{n1} & \cdots & 1 \end{bmatrix} \begin{bmatrix} u_{11} & \cdots & u_{1n} \\ & \ddots & \vdots \\ & & u_{nn} \end{bmatrix} = LU.$$

We wish to construct a nonsingular M such that MA is upper triangular. We expect M to have the following properties:

- $M = L_{n-1}^{-1} L_{n-2}^{-1} \cdots L_1^{-1}$
- Each L_j is unit lower triangular
- The product $L := L_1 L_2 \cdots L_{n-1}$ requires NO computation

Then

$$MA = U \Longrightarrow L_{n-1}^{-1} L_{n-2}^{-1} \cdots L_1^{-1} A = U \Longrightarrow A = LU,$$

where L is unit lower-triangular and U is upper-triangular.

LU factorization

Suppose A is 4×4 matrix. Then schematically

$$\underbrace{\begin{bmatrix} \times & \times & \times & \times \\ \times & \times & \times & \times \\ \times & \times & \times & \times \\ \times & \times & \times & \times \end{bmatrix}}_{A} \longrightarrow \underbrace{\begin{bmatrix} \times & \times & \times & \times \\ & \times & \times & \times \\ & \times & \times & \times \\ & \times & \times & \times \end{bmatrix}}_{L_1^{-1}A}$$

$$\underbrace{\begin{bmatrix} \times & \times & \times & \times \\ & \times & \times & \times \\ & \times & \times & \times \\ & \times & \times & \times \end{bmatrix}}_{L_1^{-1}A} \longrightarrow \underbrace{\begin{bmatrix} \times & \times & \times & \times \\ & \times & \times & \times \\ & & \times & \times \\ & & \times & \times \end{bmatrix}}_{L_2^{-1}L_1^{-1}A}$$

$$\underbrace{\begin{bmatrix} \times & \times & \times & \times \\ & \times & \times & \times \\ & & \times & \times \\ & & \times & \times \end{bmatrix}}_{L_2^{-1}L_1^{-1}A} \longrightarrow \underbrace{\begin{bmatrix} \times & \times & \times & \times \\ & \times & \times & \times \\ & & \times & \times \\ & & & \times \end{bmatrix}}_{L_3^{-1}L_2^{-1}L_1^{-1}A}$$

Example

Let $A := \begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 0 \end{bmatrix}$. Consider $L_1 := \begin{bmatrix} 1 & 0 & 0 \\ 4 & 1 & 0 \\ 7 & 0 & 1 \end{bmatrix}$. Then

$$L_1^{-1} = \begin{bmatrix} 1 & 0 & 0 \\ -4 & 1 & 0 \\ -7 & 0 & 1 \end{bmatrix} \text{ and } L_1^{-1}A = \begin{bmatrix} 1 & 2 & 3 \\ 0 & -3 & -6 \\ 0 & -6 & -21 \end{bmatrix}.$$

Now consider $L_2 := \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 2 & 1 \end{bmatrix}$. Then $L_2^{-1} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & -2 & 1 \end{bmatrix}$,

$$U := L_2^{-1}L_1^{-1}A = \begin{bmatrix} 1 & 2 & 3 \\ 0 & -3 & -6 \\ 0 & 0 & -9 \end{bmatrix} \text{ and } L := L_1L_2 = \begin{bmatrix} 1 & 0 & 0 \\ 4 & 1 & 0 \\ 7 & 2 & 1 \end{bmatrix}.$$

Thus we obtain $A = LU$.

Elimination matrix

Define $\ell_k := \begin{bmatrix} 0 \\ \vdots \\ 0 \\ \ell_{k+1,k} \\ \vdots \\ \ell_{nk} \end{bmatrix}$ and $L_k := I + \ell_k e_k^\top$ for $k = 1 : (n-1)$.

Then

$$\begin{aligned} L_k &= I + \begin{bmatrix} 0 & \cdots & 0 & \ell_k & 0 & \cdots & 0 \end{bmatrix} \\ &= \begin{bmatrix} 1 & & & & & \\ & \ddots & & & & \\ & & 1 & & & \\ & & \ell_{k+1,k} & 1 & & \\ & & \vdots & & \ddots & \\ & & \ell_{nk} & & & 1 \end{bmatrix} \end{aligned}$$

is unit lower triangular.

Product of elimination matrices

Consider $L_k = I + \ell_k e_k^\top$. By construction $e_k^\top \ell_k = 0$. Consequently

$$\underbrace{(I + \ell_k e_k^\top)}_{L_k}(I - \ell_k e_k^\top) = I + \ell_k e_k^\top - \ell_k e_k^\top - \ell_k e_k^\top \ell_k e_k^\top = I.$$

This shows that $L_k^{-1} = I - \ell_k e_k^\top$. Next observe that

$$L_k L_{k+1} = (I + \ell_k e_k^\top)(I + \ell_{k+1} e_{k+1}^\top) = I + \ell_k e_k^\top + \ell_{k+1} e_{k+1}^\top.$$

Consequently

$$\begin{aligned} L &= L_1 L_2 \cdots L_{n-1} = I + \ell_1 e_1^\top + \ell_2 e_2^\top + \cdots + \ell_{n-1} e_{n-1}^\top \\ &= I + \begin{bmatrix} \ell_1 & \ell_2 & \cdots & \ell_{n-1} & 0 \end{bmatrix} = \begin{bmatrix} 1 & & & & \\ \ell_{21} & 1 & & & \\ \ell_{31} & \ell_{32} & 1 & & \\ \vdots & \vdots & \ddots & \ddots & \\ \ell_{n1} & \ell_{n2} & \cdots & \ell_{n,n-1} & 1 \end{bmatrix}. \end{aligned}$$

Creating zeros via elimination matrix

Applying L_k^{-1} to the k-th column of an $n \times n$ matrix A, we have

$$L_k^{-1}\begin{bmatrix} a_{1k} \\ \vdots \\ a_{kk} \\ a_{k+1,k} \\ \vdots \\ a_{nk} \end{bmatrix} = (I - \ell_k e_k^\top)\begin{bmatrix} a_{1k} \\ \vdots \\ a_{kk} \\ a_{k+1,k} \\ \vdots \\ a_{nk} \end{bmatrix} = \begin{bmatrix} a_{1k} \\ \vdots \\ a_{kk} \\ a_{k+1,k} \\ \vdots \\ a_{nk} \end{bmatrix} - \begin{bmatrix} 0 \\ \vdots \\ 0 \\ \ell_{k+1,k} \\ \vdots \\ \ell_{nk} \end{bmatrix} a_{kk}$$

$$= \begin{bmatrix} a_{k1} \\ \vdots \\ a_{kk} \\ 0 \\ \vdots \\ 0 \end{bmatrix} \text{ when } \ell_{ik} = a_{ik}/a_{kk},\ i = k+1:n.$$

This shows that if $a_{kk} \neq 0$ then L_k can be used to create zeros in the k-th column of A below a_{kk}.

Multiplying a matrix by an elimination matrix

Let $A \in \mathbb{R}^{n\times n}$ and $L_k := I + \ell_k e_k^\top \in \mathbb{R}^{n\times n}$. Then

$$L_k^{-1} A = (I - \ell_k e_k^\top) A = A - \ell_k e_k^\top A = A - \ell_k \begin{bmatrix} a_{k1} & \cdots & a_{kn} \end{bmatrix}$$

$$= \left[\begin{array}{ccccc} a_{11} & \cdots & a_{kk} & \cdots & a_{1n} \\ \vdots & \ddots & \vdots & \cdots & \vdots \\ a_{k1} & \cdots & a_{kk} & \cdots & a_{kn} \\ \hline a_{k+1,n} & \cdots & a_{k+1,k} & \cdots & a_{k+1,n} \\ \vdots & \cdots & \vdots & \ddots & \vdots \\ a_{n1} & \cdots & a_{nk} & \cdots & a_{nn} \end{array}\right] - \begin{bmatrix} 0 \\ \vdots \\ 0 \\ \ell_{k+1,k} \\ \vdots \\ \ell_{nk} \end{bmatrix} \begin{bmatrix} a_{k1} & \cdots & a_{kn} \end{bmatrix}.$$

The outer product shows that the first k rows of A remain unchanged when L_k^{-1} is multiplied to the left of A. Let $B := L_k^{-1} A$. In MATLAB, B can be written compactly as a rank-1 update (outer product from)

$$\mathtt{B} = \mathtt{A}(\mathtt{k}+1:\mathtt{n}, :) - \ell(\mathtt{k}+1:\mathtt{n}) * \mathtt{A}(\mathtt{k}, :)$$

Gaussian elimination = LU decomposition

For $L_1 := I + \ell_1 e_1^\top$, with $\ell_{i1} := a_{i1}/a_{11}$, $i = 2 : n$, we have

$$L_1^{-1}A = \left[\begin{array}{c|ccc} a_{11} & a_{12} & \cdots & a_{1n} \\ \hline 0 & a_{22}^{(1)} & \cdots & a_{2n}^{(1)} \\ \vdots & \vdots & & \vdots \\ 0 & a_{n2}^{(1)} & \cdots & a_{nn}^{(1)} \end{array}\right],$$

where $a_{ij}^{(1)} = a_{ij} - \ell_{i1} a_{1j}$.
Cost: $2(n-1)^2$ flops.

The elimination is possible only when $a_{11} \neq 0$.

The vector ℓ_1 can be stored in the first column of $L_1^{-1}A$ in place of zeros. This will reduce the storage requirement.

Thus overwriting A, in MATLAB notation, we have

```
A(2:n,1) = A(2:n,1)/A(1,1);   % multipliers
A(2:n,2:n) = A(2:n,2:n) - A(2:n,1) * A(1,2:n);
```

Gaussian elimination = LU decomposition

For $L_2 := I + \ell_2 e_2^\top$ with $\ell_{i2} := a_{i2}^{(1)}/a_{22}^{(1)}$, $i = 3 : n$, we have

$$L_2^{-1} L_1^{-1} A = \left[\begin{array}{c|c|cccc} a_{11} & a_{12} & a_{13} & \cdots & a_{1n} \\ \hline 0 & a_{22}^{(1)} & a_{23}^{(1)} & \cdots & a_{2n}^{(1)} \\ \hline 0 & 0 & a_{33}^{(2)} & \cdots & a_{3n}^{(2)} \\ \vdots & \vdots & \vdots & & \vdots \\ 0 & 0 & a_{n3}^{(2)} & \cdots & a_{nn}^{(2)} \end{array}\right],$$

where $a_{ij}^{(2)} = a_{ij}^{(1)} - \ell_{i2} a_{2j}^{(1)}$.
Cost: $2(n-2)^2$ flops.

The elimination is possible only when $a_{22}^{(1)} \neq 0$. The vector ℓ_2 can be stored in the second column of $L_2^{-1} L_1^{-1} A$ in place of zeros.

Again, overwriting A, in MATLAB notation, we have

$$\mathtt{A(3:n,2) = A(3:n,2)/A(2,2);} \quad \text{\% multipliers}$$
$$\mathtt{A(3:n,3:n) = A(3:n,3:n) - A(3:n,2) * A(2,3:n);}$$

Hence we have $L_{n-1}^{-1} \cdots L_1^{-1} A = U \Rightarrow A = L_1 L_2 \cdots L_{n-1} U = LU$.
Cost: $2(n-1)^2 + 2(n-2)^2 + \cdots + 2 \simeq 2n^3/3$ flops.

Example

Consider $A := \begin{bmatrix} 2 & 4 & -2 \\ 4 & 9 & -3 \\ -2 & -3 & 7 \end{bmatrix}$. Then

$$\begin{aligned} L_1 &= I + \begin{bmatrix} 0 \\ 2 \\ -1 \end{bmatrix} e_1^\top, \quad L_1^{-1}A = \begin{bmatrix} 2 & 4 & -2 \\ 0 & 1 & 1 \\ 0 & 1 & 5 \end{bmatrix}, \\ L_2 &= I + \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} e_2^\top, \quad L_2^{-1}L_1^{-1}A = \begin{bmatrix} 2 & 4 & -2 \\ 0 & 1 & 1 \\ 0 & 0 & 4 \end{bmatrix}. \end{aligned}$$

This gives

$$A = L_1L_2 \begin{bmatrix} 2 & 4 & -2 \\ 0 & 1 & 1 \\ 0 & 0 & 4 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 2 & 1 & 0 \\ -1 & 1 & 1 \end{bmatrix} \begin{bmatrix} 2 & 4 & -2 \\ 0 & 1 & 1 \\ 0 & 0 & 4 \end{bmatrix}.$$

Gaussian Elimination with No Pivoting (GENP)

```
function [L, U] = GENP(A);
% [L U] = GENP(A) produces a unit
% lower triangular matrix L and an upper
% triangular matrix U so that A= LU.

[n, n] = size(A);
for k = 1:n-1
    % compute multipliers for k-th step
    A(k+1:n,k) = A(k+1:n,k)/A(k,k);
    % update A(k+1:n,k+1:n)
    j = k+1:n;
    A(j,j) = A(j,j)-A(j,k)*A(k,j);
end
% strict lower triangle of A, plus I
L = eye(n,n)+ tril(A,-1);
U = triu(A); % upper triangle of A
```

Solution of $Ax = b$ by LU factorization

An $n \times n$ linear system $Ax = b$ can be solved in three steps:

- Compute LU factorization $A = LU$. Cost: $\frac{2n^3}{3}$ flops.
- Solve $Ly = b$ for y. Cost: n^2 flops.
- Solve $Ux = y$ for x. Cost: n^2 flops.

Thus the cost for solving system $Ax = b$ is $2n^3/3$ flops.

Question: What will be the complexity if the system $Ax = b$ is solved as $\mathtt{x} = \mathtt{A}^{-1} * \mathtt{b}$? Answer: $2n^3/3 + 2n^3$ flops.

Question: Does LU decomposition of A exist when A is nonsingular?

Answer: Not always. LU decomposition of $A = \begin{bmatrix} 0 & 1 \\ 1 & 1 \end{bmatrix}$ does not exist.

Theorem: Let A be nonsingular. Then A admits a unique LU factorization $\Leftrightarrow$ all leading principal submatrices of A are nonsingular, that is, $A(1:j, 1:j)$ is nonsingular for $j = 1:n$.

Existence of LU factorization

Proof: Suppose that $A = LU$ exists and unique. Then writing

$$\begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix} = \begin{bmatrix} L_{11} & 0 \\ L_{21} & L_{22} \end{bmatrix} \begin{bmatrix} U_{11} & U_{12} \\ 0 & U_{22} \end{bmatrix},$$

we have $\det(A_{11}) = \det(L_{11})\det(U_{11}) = \det(U_{11}) \neq 0$. (Why?)

Conversely, suppose that all leading principal submatrices of A are nonsingular. We prove the result by induction. Suppose the result is true for $n-1$.

Let $\hat{A} = A(1:n-1, 1:n-1)$ and $\hat{A} = \hat{L}\hat{U}$ be unique LU factorization. Then writing

$$A = \begin{bmatrix} \hat{A} & b \\ c & a_{nn} \end{bmatrix} = \begin{bmatrix} \hat{L} & 0 \\ \ell & 1 \end{bmatrix} \begin{bmatrix} \hat{U} & u \\ 0 & d \end{bmatrix} = \begin{bmatrix} \hat{L}\hat{U} & \hat{L}u \\ \ell\hat{U} & \ell u + d \end{bmatrix},$$

we have $\hat{L}u = b$, $\ell\hat{U} = c$ and $d = a_{nn} - \ell u$ which give unique ℓ, u, d.

Finally, $0 \neq \det(A) = \det(\hat{U})d \Rightarrow d \neq 0$. This completes the proof. ■