Solving System of Linear Equations (I)

Mohammad Tavalla

Sixth Lecture

Mohammad.Tavalla@uoit.ca

1. Solving System of Linear Equations using Direct Methods
   - The Gaussian Elimination Method with Backward Substitution
     - Examples
   - Pivoting
   - Partial Pivoting (Maximal Column Pivoting) Method
   - Scaled Partial Pivoting (Scaled Column Pivoting) Method
   - LU-Factorization Method
   - Back Substitution with the LU-Factorization
   - Forward/Backward Substitution with the LU-Factorization
   - Gauss-Jordan Method
   - PA=LU-Factorization Method
2. Gaussian Elimination Without Row Interchanging
   - Diagonally Dominant Matrices
   - Strictly Diagonally Dominant Matrices
   - $LD\tilde{L}^t$-Factorization
   - Positive Definite Matrix
   - Symmetric Positive Definite Matrices and Gaussian Elimination

System of Linear Equations

$$
\begin{array}{lllllll}
\textbf{E1} & : a_{11}x_1 & +a_{12}x_2 & +\cdots & +a_{1n}x_n & = a_{1,n+1} \\
\textbf{E2} & : a_{21}x_1 & +a_{22}x_2 & +\cdots & +a_{2n}x_n & = a_{2,n+1} \\
\vdots & \vdots & \vdots & \ldots & \vdots & \vdots \\
\textbf{Em} & : a_{m1}x_1 & +a_{m2}x_2 & +\cdots & +a_{mn}x_n & = a_{m,n+1}
\end{array}
$$

The matrix form of this system is $Ax = b$ where

$$
A = \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & \ldots & \vdots \\ a_{m1} & a_{m2} & \cdots & a_{mn} \end{bmatrix}, \; x = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix}, \; b = \begin{bmatrix} b_1 \\ b_2 \\ \vdots \\ b_m \end{bmatrix}
$$

General $n \times n$ system of Linear Equations

The augmented matrix of a general $n \times n$ system of Linear Equations $\mathbf{AX} = \mathbf{b}$ is:

$$\left(\begin{array}{cccc|c} a_{11} & a_{12} & \cdots & a_{1n} & b_1 \\ a_{21} & a_{22} & \cdots & a_{2n} & b_2 \\ \vdots & \vdots & \cdots & \vdots & \vdots \\ a_{n1} & a_{n2} & \cdots & a_{nn} & b_n \end{array}\right)$$

The **Gaussian elimination method** turns all entries in the **j**th column under a_{jj} into zero. In fact, for every $i > j$ we replace the **i**th row with $R_i - \frac{a_{ij}}{a_{jj}} R_j$.

Note: a_{11} and all updated a_{jj}'s which are eventually divisors in Gaussian elimination are called **pivots** and *must to be non-zero*.

Gaussian Elimination Method

$$\begin{cases} 2x - 2y - z = -2 \\ 4x + y - 2z = 1 \\ -2x + y - z = -3 \end{cases} \rightarrow \begin{matrix} E_2 - 2E_1 \\ E_3 - (-E_1) \end{matrix} \rightarrow \begin{cases} 2x - 2y - z = -2 \\ 0 + 5y + 0 = 5 \\ 0 - y - 2z = -5 \end{cases}$$

$$E_3 - \left(-\frac{1}{5}E_2\right) \rightarrow \begin{cases} 2x - 2y - z = -2 \\ 0 + 5y + 0 = 5 \\ 0 + 0 - 2z = -4 \end{cases}$$

Then by **back substitute** or **back solving** we have:

$$\begin{cases} -2z = -4 \rightarrow \boxed{z=2} \\ 5y = 5 \rightarrow \boxed{y=1} \\ 2x - 2 - 2 - 2 \rightarrow \boxed{x=1} \end{cases}$$

Augmented Matrix

$$\left(\begin{array}{ccc|c} 2 & -2 & -1 & -2 \\ 4 & 1 & -2 & 1 \\ -2 & 1 & -1 & -3 \end{array}\right) \rightarrow \begin{array}{c} R_2 - 2R_1 \\ R_3 - (-R_1) \end{array} \rightarrow \left(\begin{array}{ccc|c} 2 & -2 & -1 & -2 \\ 0 & 5 & 0 & 5 \\ 0 & -1 & -2 & -4 \end{array}\right)$$

$$R_3 - \left(-\frac{1}{5}\right) R_2 \rightarrow \left(\begin{array}{ccc|c} 2 & -2 & -1 & -2 \\ 0 & 5 & 0 & 5 \\ 0 & 0 & -2 & -4 \end{array}\right)$$

Then by **back substitute** or **back solving** we have:

$$\begin{cases} -2z = -4 \rightarrow \boxed{\text{z=2}} \\ 5y = 5 \rightarrow \boxed{\text{y=1}} \\ 2x - 2 - 2 - 2 \rightarrow \boxed{\text{x=1}} \end{cases}$$

MATLAB and System of Linear Equations

$$\begin{cases} 4x & - & 2y & = & 15 \\ -3x & - & y & = & 8 \end{cases}$$

- `MATLAB solve` command
  $[x, y] = \texttt{solve}('\ 4*x-2*y=15\ '\ ,\ '\ -3*x-y=8\ ')$

Operation Count

- **Elimination step:** The elimination step for a system of n equations in n unknowns can be completed in $\frac{2}{3}n^3 + \frac{1}{2}n^2 - \frac{7}{6}n$ operations.
- When n is large, we write elimination step takes on the order of n^3 or in short $O(n^3)$.
- **Back-substitution step:** The back-substitution step for a triangular system of n equations in n unknowns can be completed in n^2 operations.
- Back-substitution takes on the order of n^2 or in short $O(n^2)$.

Pivoting

Suppose ϵ is a sufficiently small number and $B,\ C,\ D,\ E$, and F are arbitrary real numbers. Then

$$\begin{cases} \epsilon x & + & By & = & C \\ Dx & + & Ey & = & F \end{cases}$$

because of round-off errors can have an approximate solution as $y \approx \frac{C}{B}$ and $x \approx 0$!!
One method to fix this problem is called **partial pivoting**.

Partial Pivoting (Maximal Column Pivoting) Method

The problem with small pivots can sometimes be resolved with **partial pivoting method.** That is by choosing the entry with largest magnitude in each column as the pivot and apply a row interchange. That is

1. If $|a_{pk}^{(k)}| = \max_{k\leq i\leq n}|a_{ik}^{(k)}|$ then
2. $(E_k) \longleftrightarrow (E_p)$

Check this:

$$\begin{cases} 0.002x + 0.11y = 9.2 \\ 1.32x + 665y = 17 \end{cases} \rightarrow \begin{cases} 1.32x + 665y = 17 \\ 0.002x + 0.11y = 9.2 \end{cases}$$

Although this strategy is sufficient for many linear systems, situations do arise when it is inadequate.

Scaled Partial Pivoting (Scaled Column Pivoting) Method

It places the element in the pivot position that is largest relative to the entries in its row. That is

1. If $s_i = \max_{1 \le j \le n} |a_{ij}|$ and $s_i \neq 0$ then the appropriate row interchange is the least $p \ge i$ such that:

$$\frac{|a_{pi}|}{s_p} = \max_{i \le k \le n} \frac{|a_{ki}|}{s_k}$$

2. $(E_i) \longleftrightarrow (E_p)$ if $i \neq p$.

Check this:

$$\begin{cases} 0.002x + 0.11y = 9.2 \\ 1.32x + 665y = 17 \end{cases}$$

LU-Factorization Method

Lower and Upper Triangular Matrices

Definition: An $m \times n$ matrix L is **lower triangular** if its entries satisfy $l_{ij} = 0$ for $i < j$.

Definition: An $m \times n$ matrix U is **upper triangular** if its entries satisfy $u_{ij} = 0$ for $i > j$.

The following properties can be verified:

- Let $L_{ks}(-c) = (l_{ij})_{n \times n}$ be such that

$$\begin{cases} l_{ij} = -c & , \quad i = k, j = s \\ l_{ij} = 1 & , \quad i = j \\ l_{ij} = 0 & , \quad \text{otherwise} \end{cases}$$

Then $A \to L_{ks}(-c)A$ represents the row operation "subtracting c times sth row from the kth row". That is $R_k - cR_s \to R_k$.

Lower and Upper Triangular Matrices

- $L_{ks}(-c)^{-1} = L_{ks}(c)$
- The following matrix product holds:

$$\begin{pmatrix} 1 & & \\ c_1 & 1 & \\ & & 1 \end{pmatrix} \begin{pmatrix} 1 & & \\ & 1 & \\ c_2 & & 1 \end{pmatrix} \begin{pmatrix} 1 & & \\ & 1 & \\ & c_3 & 1 \end{pmatrix} = \begin{pmatrix} 1 & & \\ c_1 & 1 & \\ c_2 & c_3 & 1 \end{pmatrix}$$

Therefore, the **LU-factorization** is based on a lower triangular with diagonal entries equal to 1 made up of factors for eliminations and the resulting upper triangular in Gaussian elimination by turning entries below the main diagonal into zero.

Note

$det(A) = det(L)det(U) = det(U)$.

Examples

1. $$\begin{pmatrix} 3 & 1 & 2 \\ 6 & 3 & 4 \\ 3 & 1 & 5 \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 \\ -2 & 1 & 0 \\ -1 & -1 & 1 \end{pmatrix} \begin{pmatrix} 3 & 1 & 2 \\ 0 & 1 & 0 \\ 0 & 0 & 3 \end{pmatrix}$$

2. $$\begin{pmatrix} 4 & 2 & 0 \\ 4 & 4 & 2 \\ 2 & 2 & 3 \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 \\ -1 & 1 & 0 \\ -\frac{1}{2} & -1 & 1 \end{pmatrix} \begin{pmatrix} 4 & 2 & 0 \\ 0 & 2 & 2 \\ 0 & 0 & 1 \end{pmatrix}$$

LU-Factorization Theorem

Theorem (LU-Factorization)

If Gaussian elimination can be performed on the linear system $Ax = b$ without row interchanges, then the matrix A can be factored into the product of a lower-triangular matrix L and an upper-triangular matrix U, that is, $A = LU$, where $m_{ji} = a_{ji}^{(i)}/a_{ii}^{(i)}$,

$$L = \begin{bmatrix} 1 & 0 & \cdots & 0 \\ m_{21} & 1 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ m_{n1} & m_{n2} & \cdots & 1 \end{bmatrix}, \; U = \begin{bmatrix} a_{11}^{(1)} & a_{12}^{(1)} & \cdots & a_{1n}^{(1)} \\ 0 & a_{22}^{(2)} & \cdots & a_{2n}^{(2)} \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & a_{nn}^{(n)} \end{bmatrix}$$

Forward/Backward Substitution with the LU-Factorization

Suppose $Ax = b$ and $A = LU$ is the LU-factorization. Then the system $Ax = b$ can be solved as follows: ($O(n^2)$ Operations each)

1. Let $Ux = Y$ and solve $Ly = b$ by ***forward substitution method***.
2. Solve $Ux = Y$ by ***backward substitution method***.

Remark

1. *Since the right-hand-side* ***b*** *doesn't enter the calculations until elimination is finished; one can solve a set of k systems*

$$Ax = b^{(1)}, \cdots, \; Ax = b^{(k)}$$

   with only one elimination. Thus the ***LU-factorization*** *reduces the number of operations in this situation from $\frac{2kn^3}{3}$ to $\frac{2n^3}{3} + 2kn^2$.*
2. *When n and k are large, this is a significant difference.*

Solving a System by LU-Factorization

$$\left\{\begin{array}{rcrcrcrcr} x_1 & + & x_2 & & & + & 3x_4 & = & 8 \\ 2x_1 & + & x_2 & - & x_3 & + & 4x_3 & = & 7 \\ 3x_1 & - & x_2 & - & x_3 & + & 2x_4 & = & 14 \\ -x_1 & + & 2x_2 & + & 3x_3 & - & x_4 & = & -7 \end{array}\right.$$

$$\begin{bmatrix} 1 & 1 & 0 & 3 \\ 2 & 1 & -1 & 1 \\ 3 & -1 & -1 & 2 \\ -1 & 2 & 3 & -1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 2 & 1 & 0 & 0 \\ 3 & 4 & 1 & 0 \\ -1 & -3 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 1 & 0 & 3 \\ 0 & -1 & -1 & -5 \\ 0 & 0 & 3 & 13 \\ 0 & 0 & 0 & -13 \end{bmatrix}$$
$$= LU$$

LU-Factorization Theorem

Definition (kth Leading Principal Submatrix)

For a square matrix $A_n = (a_{ij})$, and $1 \leq k < n$, the kth leading principal submatrix $A^{(k)} = (a'_{ij})$ is a square matrix of order k defined as

$$a'_{ij} = a_{ij} \ , \quad 1 \leq i,j \leq k$$

Theorem (The Existence of LU-Factorization)

The square matrix $A_{n\times n}$ has an $LU-$factorization if det$\Big(A^{(k)}\Big) \neq 0$ for all $k = 1, \cdots, n-1$. If the $LU-$factorization exists and A is non-singular, then the $LU-$factorization is unique and det$A = u_{11}\cdots u_{nn}$ the product of entries on the main diagonal of U.

Gauss-Jordan Method

This method is described as follows. Use the ith equation to eliminate not only x_i from the equations $E_{i+1}, E_{i+2}, \cdots, E_n$, as was done in the Gaussian elimination method, but also from $E_1, E_2, \cdots, E_{i-1}$. Upon reducing $[A, b]$ to:

$$A = \begin{bmatrix} a_{11}^{(1)} & 0 & & \cdots & 0 \\ 0 & a_{22}^{(2)} & 0 & \cdots & 0 \\ \vdots & \vdots & & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & a_{nn}^{(n)} \end{bmatrix}$$

where $a_{kk}^{(k)}$ is the updated pivot on the kth row.

PA=LU Factorization Method

Recall

- A *permutation* matrix P is the identity matrix with its rows re-ordered.
- One efficient way to store a permutation matrix P is with an integer $n-$vector p whose components correspond to row numbers including 1 in each column.
- If P is a permutation matrix and A is a matrix, then PA is a row permuted version of A and AP a column permuted.
- Permutation matrix P is orthogonal, that is $p^{-1} = P^t$

A More General Case

When one has to make some row interchanging in order to determine the LU-factorization of the matrix A, one has to multiply A by a permutation matrix P then write the LU-factorization. Note that since $P^{(-1)} = P^t$ we can write:

$$Ax = b \rightarrow PAx = \tilde{L}\tilde{U}x = Pb \rightarrow P^t\tilde{L}\tilde{U}x = b$$

Note

The matrix $\tilde{U}$ is still upper triangular but $P^t\tilde{L}$ is no longer lower triangular unless $P = I$.

Check this: $A = \begin{bmatrix} 3 & 17 & 10 \\ 2 & 4 & -2 \\ 6 & 18 & -12 \end{bmatrix}$, $b = \begin{bmatrix} -1 \\ 0 \\ 1 \end{bmatrix}$

LU-Factorization in MATLAB

$>>$ `[L,U,P]=lu(A)`
returns the LU-factorization along with a permutation matrix P of a given square matrix A.

The Avoidance of Pivoting

Diagonally Dominant Matrices

Definition (Diagonally Dominant Matrix)

A square matrix A of order n is said to be ***diagonally dominant*** *when*

$$|a_{ii}| \geq \sum_{i \neq j=1}^{n} |a_{ij}| \ , \ \forall i = 1, 2, \cdots, n$$

It is ***strictly diagonally dominant*** *when the inequality is strict.*

$$\begin{pmatrix} -9 & -2 & -1 \\ 4 & 11 & -2 \\ -2 & 1 & -5 \end{pmatrix}$$

Strictly Diagonally Dominant Matrices-Theorem

Theorem (SDD Matrices and GE Method)

A strictly diagonally dominant matrix A is non-singular. Moreover, in this case, Gaussian elimination can be performed on any linear system of the form $Ax = b$ to obtain its unique solution without row or column interchanges, and the computations will be ***stable*** *with respect to the growth of round-off errors.*

$LD\tilde{L}^t$-Factorization

Theorem ($LD\tilde{L}^t$-Factorization)

If all the leading principal submatrices of A_n are non-singular, then there exist unique unit lower triangular matrices L and $\tilde{L}$ and a unique diagonal matrix $D =$diag$(d_1, \cdots, d_n)$ such that $A = LD\tilde{L}^t$.

Proof: Use $LU-$factorization and let $D =$*diag*$(d_1, \cdots, d_n)$ with $d_i = u_{ii}$, $1 \leq i \leq n$ and $\tilde{L} = D^{-1}U$.■

Symmetric Positive Definite Matrix

Definition (Symmetric Positive Definite Matrix)

A symmetric matrix A of order n is said to be ***symmetric positive definite*** *(SPD) if $x^t Ax > 0$ for every n-dimensional vector $x \neq 0$.*

Example

$$A = \begin{pmatrix} 2 & 1 & 0 \\ 1 & 3 & 0 \\ 0 & 0 & 4 \end{pmatrix}, \; X = \begin{pmatrix} a \\ b \\ c \end{pmatrix} \neq 0$$

Symmetric Positive Definite Matrices and Gaussian Elimination

Theorem (NS condition for SPD matrices)

A symmetric matrix A is positive definite if and only if each of its leading principal sub-matrices has a positive determinant.

Example

$$A = \begin{pmatrix} 2 & 1 & 0 \\ 1 & 3 & 0 \\ 0 & 0 & 4 \end{pmatrix}, \ 2 > 0, \ \begin{vmatrix} 2 & 1 \\ 1 & 3 \end{vmatrix} = 5 > 0, \ \mathit{det}\ A = 20 > 0$$

Symmetric Positive Definite Matrices and Gaussian Elimination

Theorem (SPD Matrix and GE Method)

The symmetric matrix A is positive definite if and only if Gaussian elimination without row interchanges can be performed on the linear system $Ax = b$ with all pivot elements positive. Moreover, in this case, the computations are stable with respect to the growth of the round-off errors.

Corollary

The matrix A is symmetric positive definite if and only if A can be factored in the form LDL^t, where L is a unit lower triangular and D is a diagonal matrix with positive diagonal entries.

Proof: Note that since A is symmetric, in $A = LD\tilde{L}^t$, we have $L = \tilde{L}$.■

$$A = \begin{pmatrix} 2 & 1 & 0 \\ 1 & 3 & 0 \\ 0 & 0 & 4 \end{pmatrix}, \; L = \begin{pmatrix} 1 & 0 & 0 \\ 1/2 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix},$$

$$D = \begin{pmatrix} 2 & 0 & 0 \\ 0 & 5/2 & 0 \\ 0 & 0 & 4 \end{pmatrix}, \; U = \begin{pmatrix} 1 & 1/2 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} = L^t$$

Corollary (**Cholesky Factorization**)

The matrix A is symmetric positive definite if and only if A can be factored in the form GG^t , where G is a real lower triangular with non-zero main diagonal entries.

Proof: In $A = LDL^t$ let $G = L\mathit{diag}(\sqrt{d_1}, \cdots, \sqrt{d_n})$.■

In the previous example we can write;

$$D = \begin{pmatrix} \sqrt{2} & 0 & 0 \\ 0 & \sqrt{5/2} & 0 \\ 0 & 0 & 2 \end{pmatrix} \begin{pmatrix} \sqrt{2} & 0 & 0 \\ 0 & \sqrt{5/2} & 0 \\ 0 & 0 & 2 \end{pmatrix}$$

$$G = \begin{pmatrix} 1 & 0 & 0 \\ 1/2 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} \sqrt{2} & 0 & 0 \\ 0 & \sqrt{5/2} & 0 \\ 0 & 0 & 2 \end{pmatrix}$$

Summary of calculation in Cholesky factorization:

$$g_{11}^2 = a_{11},$$

$$g_{21}g_{11} = a_{21}, \ \& \ g_{21}^2 + g_{22}^2 = a_{22} \rightarrow g_{21} = \frac{a_{21}}{g_{11}}, \ g_{22} = \sqrt{a_{22} - \left(\frac{a_{21}}{g_{11}}\right)^2}$$

In general,

$$g_{ii}^2 = \frac{a_{ij} - \sum_{k=1}^{j-1} g_{ik}g_{jk}}{g_{jj}}, \ j = 1, \cdots, i-1$$

$$g_{ii} = \left(a_{ii} - \sum_{k=1}^{j-1} g_{ik}^2\right)^{1/2}, \ i = 1, \cdots, n$$