BITS Pilani
Pilani Campus

Mathematical Foundations for Data Science

MFDS Team

innovate achieve lead

BITS Pilani
Pilani Campus

DSECL ZC416, MFDS

Lecture No.1

Agenda

- Solution of linear systems – an overview

- Gauss elimination methods
  - sensitivity to changes in A
  - pivoting and
  - operations count

- LU decomposition methods
  - Doolittle's method
  - Crout's method

Linear System

Matrix Form of the Linear System (1).

From the definition of matrix multiplication we see that the m equations of (1) may be written as a single vector equation

(2) $$\mathbf{A}\mathbf{x} = \mathbf{b}$$

where the **coefficient matrix A** = $[a_{jk}]$ is the $m \times n$ matrix

$$\mathbf{A} = \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \cdot & \cdot & \cdots & \cdot \\ a_{m1} & a_{m2} & \cdots & a_{mn} \end{bmatrix}, \quad \text{and} \quad \mathbf{x} = \begin{bmatrix} x_1 \\ \cdot \\ \cdot \\ \cdot \\ x_n \end{bmatrix} \quad \text{and} \quad \mathbf{b} = \begin{bmatrix} b_1 \\ \vdots \\ b_m \end{bmatrix}$$

are column vectors.

Matrix Form of Linear System

Matrix Form of the Linear System (1). (continued)

We assume that the coefficients a_{jk} are not all zero, so that **A** is not a zero matrix. Note that **x** has n components, whereas **b** has m components. The matrix

$$\tilde{\mathbf{A}} = \left[\begin{array}{ccc:c} a_{11} & \cdots & a_{1n} & b_1 \\ \cdot & \cdots & \cdot & \cdot \\ \cdot & \cdots & \cdot & \cdot \\ a_{m1} & \cdots & a_{mn} & b_m \end{array}\right]$$

is called the **augmented matrix** of the system (1). The dashed vertical line could be omitted, as we shall do later. It is merely a reminder that the last column of $\tilde{\mathbf{A}}$ did not come from matrix **A** but came from vector **b**. Thus, we *augmented* the matrix **A**.

Gauss Elimination and Back Substitution

Triangular form:

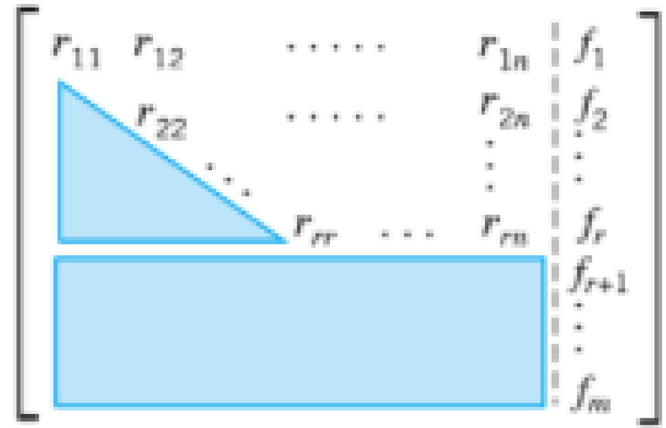
Triangular means that all the nonzero entries of the corresponding coefficient matrix lie above the diagonal and form an upside-down 90° triangle. Then we can solve the system by **back substitution.**

Since a linear system is completely determined by its augmented matrix, ***Gauss elimination can be done by merely considering the matrices***.

(We do this again in the next example, emphasizing the matrices by writing them first and the equations behind them, just as a help in order not to lose track.)

Gauss Elimination

At the end of the Gauss elimination (before the back substitution), the row echelon form of the augmented matrix will be



1. in upper triangular form

2. have the first r (r is rank) rows non-zero

3. Exactly m - r rows would be zero rows

4. If consistent , then last m- r rows will be zero rows

5. If any one of the last m-r rows is non-zero, it would imply inconsistency

6. facilitates the back substitution

Solution

The number of nonzero rows, r, in the row-reduced coefficient matrix **R** is called the **rank of R** and also the **rank of A**. Here is the method for determining whether $\mathbf{Ax} = \mathbf{b}$ has solutions and what they are:

(a) No solution. If r is less than m (meaning that **R** actually has at least one row of all 0s) *and* at least one of the numbers $f_{r+1}, f_{r+2}, \ldots, f_m$ is not zero, then the system $\mathbf{Rx} = \mathbf{f}$ is inconsistent: No solution is possible. Therefore the system $\mathbf{Ax} = \mathbf{b}$ is inconsistent as well.

Solution

If the system is consistent (either $r = m$, or $r < m$ *and* all the numbers $f_{r+1}, f_{r+2}, \dots, f_m$ are zero), then there are solutions.

(b) Unique solution. If the system is consistent and $r = n$, there is exactly one solution, which can be found by back substitution.

(c) Infinitely many solutions. To obtain any of these solutions, choose values of $x_{r+1}, \dots, x_n$ arbitrarily. Then solve the rth equation for x_r (in terms of those arbitrary values), then the $(r - 1)$st equation for x_{r-1}, and so on up the line.

Gauss Elimination

Solve $Ax = b$

Consists of two phases:

Forward elimination

Back substitution

Forward Elimination reduces $Ax = b$ to an upper triangular system $Tx = b'$

Back substitution can then solve $Tx = b'$ for x

$$\begin{bmatrix} a_{11} & a_{12} & a_{13} & | & b_1 \\ a_{21} & a_{22} & a_{23} & | & b_2 \\ a_{31} & a_{32} & a_{33} & | & b_3 \end{bmatrix}$$

$$\Downarrow$$

$$\begin{bmatrix} a_{11} & a_{12} & a_{13} & | & b_1 \\ 0 & a'_{22} & a'_{23} & | & b'_2 \\ 0 & 0 & a''_{33} & | & b''_3 \end{bmatrix}$$

Forward Elimination

$$\Downarrow$$

$$x_3 = \frac{b''_3}{a''_{33}} \qquad x_2 = \frac{b'_2 - a'_{23}x_3}{a'_{22}}$$

$$x_1 = \frac{b_1 - a_{13}x_3 - a_{12}x_2}{a_{11}}$$

Back Substitution

Pitfalls of Gauss Elimination

innovate achieve lead

Division by zero

It is possible that during both elimination and back-substitution phases a division by zero can occur.

<u>For example:</u>

$$
\begin{aligned}
2x_2 + 3x_3 &= 8 \\
4x_1 + 6x_2 + 7x_3 &= -3 \\
2x_1 + x_2 + 6x_3 &= 5
\end{aligned}
\qquad
A = \begin{matrix} 0 & 2 & 3 \\ 4 & 6 & 7 \\ 2 & 1 & 6 \end{matrix}
$$

$a_{11} = 0$
(the pivot element)

It is possible that during both elimination and back-substitution phases
a division by zero can occur.

<u>Solution</u>: **Pivoting**

Pitfalls of Gauss Elimination

Round-off errors

Because computers carry only a limited number of significant figures, round-off errors will occur and they will *propagate* from one iteration to the next.

This problem is especially important when **large** numbers of equations (100 or more) are to be solved.

Always use **double-precision** numbers/arithmetic. It is slow but needed for correctness!

It is also a good idea to substitute your results back into the original equations and check whether a substantial error has occurred.

Ill conditioned systems

- Systems where small changes in coefficients result in large change in solution

$$x_1 + 2x_2 = 10$$
$$1.1x_1 + 2x_2 = 10.4$$

$\rightarrow x_1 = 4.0 \;\; \& \;\; x_2 = 3.0$

$$x_1 + 2x_2 = 10$$
$$1.05x_1 + 2x_2 = 10.4$$

$\rightarrow x_1 = 8.0 \;\; \& \;\; x_2 = 1.0$

Condition Number

Condition number of a non singular matrix A is defined by $\kappa(\mathbf{A}) = \|\mathbf{A}\| \, \|\mathbf{A}^{-1}\|$.

By convention, cond (A) = ∞ if A is singular

Example: $A = \begin{bmatrix} 2 & -1 & 1 \\ 1 & 0 & 1 \\ 3 & -1 & 4 \end{bmatrix}$ $\|A\|_1 = 6$ $\|A\|_\infty = 8$

$$A^{-1} = \begin{bmatrix} 0.5 & 1.5 & -0.5 \\ -0.5 & 2.5 & -0.5 \\ -0.5 & -0.5 & 0.5 \end{bmatrix} \quad \|A^{-1}\|_1 = 4.5 \quad \|A^{-1}\|_\infty = 3.5$$

$\text{cond}_1 (A) = 6 \times 4.5 = 27$
$\text{cond}_\infty (A) = 8 \times 3.5 = 28$

1. There is no sharp dividing line between "well-conditioned" and "ill-conditioned," but generally the situation will get worse as we go from systems with small $\kappa(\mathbf{A})$ to systems with larger $\kappa(\mathbf{A})$. Now always $\kappa(\mathbf{A}) \geqq 1$, so that values of 10 or 20 or so give no reason for concern, whereas $\kappa(\mathbf{A}) = 100$, say, calls for caution, and systems such as those in Examples 1 and 2 are extremely ill-conditioned.

2. If $\kappa(\mathbf{A})$ is large (or small) in one norm, it will be large (or small, respectively) in any other norm. See Example 5.

Techniques for Improving the solution

Use of more significant figures – double precision arithmetic

Pivoting

If a pivot element is zero, normalization step leads to division by zero. The same problem may arise, when the pivot element is close to zero. Problem can be avoided:

Partial pivoting

- Switching the rows below so that the largest element is the pivot element.

Complete pivoting

- Searching for the largest element in all rows and columns then switching.
- This is rarely used because switching columns changes the order of x's and adds significant complexity and overhead ➔ costly

Scaling - used to reduce the round-off errors and improve accuracy

Partial Pivoting – Example

Pivoting Example

Example 14: Solve the following system using Gauss Elimination with pivoting.

$$
\begin{aligned}
2x_2 + \quad\quad + x_4 &= 0 \\
2x_1 + 2x_2 + 3x_3 + 2x_4 &= -2 \\
4x_1 - 3x_2 \quad\quad + x_4 &= -7 \\
6x_1 + x_2 - 6x_3 - 5x_4 &= 6
\end{aligned}
$$

Step 0: Form the augmented matrix

$$
\left[\begin{array}{cccc|c}
0 & 2 & 0 & 1 & 0 \\
2 & 2 & 3 & 2 & -2 \\
4 & -3 & 0 & 1 & -7 \\
6 & 1 & -6 & -5 & 6
\end{array}\right]
$$

Step 1: Forward Elimination

(1.1) Eliminate x_1. But the pivot element is 0. We have to interchange the 1st row with one of the rows below it. Interchange it with the 4th row because 6 is the largest possible pivot.

Partial Pivoting – Example

(1.1) Eliminate x_1 . But the pivot element is 0. We have to interchange the 1st row with one of the rows below it. Interchange it with the 4th row because 6 is the largest possible pivot.

$$\left[\begin{array}{cccc|c} 6 & 1 & -6 & -5 & 6 \\ 2 & 2 & 3 & 2 & -2 \\ 4 & -3 & 0 & 1 & -7 \\ 0 & 2 & 0 & 1 & 0 \end{array}\right]$$

Now eliminate x_1

$$\left[\begin{array}{cccc|c} 6 & 1 & -6 & -5 & 6 \\ 0 & 1.6667 & 5 & 3.6667 & -4 \\ 0 & -3.6667 & 4 & 4.3333 & -11 \\ 0 & 2 & 0 & 1 & 0 \end{array}\right]$$

(1.2) Eliminate x_2 .from the 3rd and 4th eqns. Pivot element is 1.6667. There is no division by zero problem. Still we will perform pivoting to reduce round-off errors. Interchange the 2nd and 3rd rows. Note that complete pivoting would interchange 2nd and 3rd columns.

$$\left[\begin{array}{cccc|c} 6 & 1 & -6 & -5 & 6 \\ 0 & -3.6667 & 4 & 4.3333 & -11 \\ 0 & 1.6667 & 5 & 3.6667 & -4 \\ 0 & 2 & 0 & 1 & 0 \end{array}\right]$$

Eliminate x_2

$$\left[\begin{array}{cccc|c} 6 & 1 & -6 & -5 & 6 \\ 0 & -3.6667 & 4 & 4.3333 & -11 \\ 0 & 0 & 6.8182 & 5.6364 & -9.0001 \\ 0 & 0 & 2.1818 & 3.3636 & -5.9999 \end{array}\right]$$

(1.3) Eliminate x_3 . $6.8182 > 2.1818$, therefore no pivoting is necessary.

$$\left[\begin{array}{cccc|c} 6 & 1 & -6 & -5 & 6 \\ 0 & -3.6667 & 4 & 4.3333 & -11 \\ 0 & 0 & 6.8182 & 5.6364 & -9.0001 \\ 0 & 0 & 0 & 1.5600 & -3.1199 \end{array}\right]$$

Partial Pivoting – Example

Step 2: Back substitution

x_4 = -3.1199 / 1.5600 = **-1.9999**

x_3 = [-9.0001 – 5.6364*(-1.9999)] / 6.8182 = **0.33325**

x_2 = [-11 – 4.3333*(-1.9999) – 4*0.33325] / -3.6667 = **1.0000**

x_1 = [6 – (-5)*(-1.9999) – (-6)*0.33325 – 1*1.0000] / 6 = **-0.50000**

Exact solution is x = $[-2 \quad 1/3 \quad 1 \quad -0.5]^T$. Use more than 5 sig. figs. to reduce round-off errors.

Gauss Elimination with Rounding

$$0.0004x_1 + 1.402x_2 = 1.406$$
$$0.4003x_1 - 1.502x_2 = 2.501$$

Original solution of the system is $x_1 = 10$, $x_2 = 1$

Picking the first of given equation as pivot equation , we have to multiply this equation by $m = 0.4003/0.0004 = 1001$ and subtract result from the second equation , obtaining

$$-1405x_2 = -1404 \quad \Rightarrow \quad x_2 = 0.9993$$

From first equation we get $x_1 = 12.5$

The failure occurs because $|a_{11}|$ is small compared to $|a_{12}|$ so that a small round off error in x_2 led to a large error in x_1

Gauss Elimination with Rounding

$$0.0004x_1 + 1.402x_2 = 1.406$$
$$0.4003x_1 - 1.502x_2 = 2.501$$

Picking the second of the given equations as the pivot equation, we have to multiply this equation by $0.0004/0.4003 = 0.0009993$ and subtract the result from the first equation obtaining

$$1.404x_2 = 1.404$$

$x_2 = 1$ and $x_1 = 10$

Note $|a_{21}|$ is not very small compared to $|a_{22}|$ so that a small round off error in x_2 would not lead to a large error in

x_1

Operation Count – Gauss Elimination

Important factors in judging the quality of a numerical method are

- Amount of storage
- Amount of time (= number of operations)

Consider Augmented Matrix of Ax = b, where $a_{in+1} = b_i$

$$\begin{bmatrix} a_{11} & a_{12} & \dots & a_{1n+1} \\ a_{21} & a_{22} & \dots & a_{2n+1} \\ \vdots & & \ddots & \vdots \\ a_{n1} & a_{n2} & \dots & a_{nn+1} \end{bmatrix}$$

Operation Count – Gauss Elimination

In elimination procedure to get the rank we will make all elements below main diagonal zero.

Total number of multiplications and additions required to determine the rank by elimination procedure are

$$2.\sum_{k=1}^{n-1}(n-k)(n-k+1)=O(n^3)$$

Total number of divisions is

$$\sum_{k=1}^{n-1}(n-k)=O(n^2)$$

Operation Count – Gauss Elimination

In back substitution total number of additions, multiplications and divisions required are

$$\left(2.\sum_{k=1}^{n}(n-k)\right)+n=O(n^2)$$

If an operation takes 10^{-9} sec, then

Algorithm	$n = 1000$	$n = 10000$
Elimination	0.7 sec	11 min
Back substitution	0.001 sec	0.1 sec

LU Factorization

We write square matrix A as

$$A = LU$$

Doolittle's Method : L is lower triangular matrix diag(L) = 1, l_{ii} = 1 and U is upper triangular matrix

Crout's Method : U is upper triangular matrix with diag(U) = 1, u_{ii} = 1 and L is lower triangular matrix

Cholesky's Method: $U = L^T$

Benefits of LU Decomposition

$A = LU$, Thus, the system $Ax = B$, is

$LUx = B$

Let $Ux = y$, then

$Ly = B$

Algorithm :-

Step-I Solve $Ly = B$, to find y.

Step-II Then solve $Ux = y$ to find x

•Methods of LU Factorization

Doolittle Method: The Factors L, U are defined as

$$L = \begin{bmatrix} 1 & 0 & 0 & 0 \\ l_{21} & 1 & 0 & 0 \\ l_{31} & l_{32} & 1 & 0 \\ l_{41} & l_{42} & l_{43} & 1 \end{bmatrix} \qquad U = \begin{bmatrix} u_{11} & u_{12} & u_{13} & u_{14} \\ 0 & u_{22} & u_{23} & u_{24} \\ 0 & 0 & u_{33} & u_{34} \\ 0 & 0 & 0 & u_{44} \end{bmatrix}$$

$$l_{ij} = 1, \quad \text{for } i = j$$
$$l_{ij} = 0, \quad \text{for } i < j$$
$$u_{ij} = 0, \quad \text{for } i > j$$

Crout's Method: The Factors L, U are defined as

$$L = \begin{bmatrix} l_{11} & 0 & 0 & 0 \\ l_{21} & l_{22} & 0 & 0 \\ l_{31} & l_{32} & l_{33} & 0 \\ l_{41} & l_{42} & l_{43} & l_{44} \end{bmatrix} \qquad U = \begin{bmatrix} 1 & u_{12} & u_{13} & u_{14} \\ 0 & 1 & u_{23} & u_{24} \\ 0 & 0 & 1 & u_{34} \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$l_{ij} = 0, \quad \text{for } i < j$$
$$u_{ij} = 1, \quad \text{for } i = j$$
$$u_{ij} = 0, \quad \text{for } i > j$$