Linear Algebra - Computations

- The discussion in this course will focus on Computation. In terms of material it should be a review.

- Please use the lessons to enhance your competency in the subject

System of linear Equations (Motivation is known)

$$\begin{aligned} a_{11}x_1 + \dots &+ a_{1n}x_n = b_1 \\ \vdots & \qquad \vdots \\ a_{m1}x_1 \dots &+ a_{mn}x_n = b_m \end{aligned}$$

$a_{ij} \in \mathbb{R}$ $\quad 1 \le i \le n$

$b_j \in \mathbb{R}$ $\quad 1 \le j \le m$

x_i $\quad 1 \le i \le n$ - unknown

- In short we shall write

$$(*) \quad Ax = b$$

$\underbrace{a_{ij} \in \mathbb{R}, A_{m\times n} = [a_{ij}], b \in \mathbb{R}^m}_{\text{Given}}$ and $x \in \mathbb{R}^n$ unknown

Q:- Can we solve for $x \in \mathbb{R}^n$ in an efficient manner?

Primary objective

Review : — results / facts on Consistency of $Ax = b$

- Elementary Row operations / Echelon form.

- Role of rank(A) & rank([A : b])

- $b = x_1 A_{*1} + x_2 A_{*2} \cdots + x_n A_{*n} \Leftrightarrow Ax = b$

  $b \in R(A) \Leftrightarrow$ Consistency.

- linearly independence & non-singular $A_{n \times n}$

- dimension of $N(A)$ = Null space (A)

- det(A) & Eigen-values of A

will give some review problems this week.

Our starting point will be: • $m = n$ & $Ax = b$ is consistent

• Efficiently compute x.

- For Examples:-

Keep track
- # of computations.
- all necessary or not.
- Are there other ways?

Q - Can we program in R or write a pseudo-code?

Example 1:-

$n=3$ [Diagonal]

$b = \begin{bmatrix} 3 \\ 12 \\ 25 \end{bmatrix}$

$a_{11}=1, \quad a_{22}=3, \quad \text{and} \quad a_{33}=5$

$a_{ij}=0 \quad i \neq j$

$$\begin{bmatrix} 1 & 0 & 0 \\ 0 & 3 & 0 \\ 0 & 0 & 5 \end{bmatrix} x = \begin{bmatrix} 3 \\ 12 \\ 25 \end{bmatrix}$$

$$\Rightarrow \underbrace{\begin{aligned} 1x_1 &= 3 \\ 3x_2 &= 12 \\ 5x_3 &= 25 \end{aligned}} \quad \Rightarrow \underbrace{\begin{aligned} x_1 &= 3 \\ x_2 &= 4 \\ x_3 &= 5 \end{aligned}} \checkmark$$

$A_{n\times n}$ – diagonal matrix with $a_{ii} \neq 0$ $\quad a_{ij} = 0 \quad i \neq j$

– for i in 1 to n

$$x_i = \frac{b_i}{a_{ii}}$$

end.

# of computations in $Ax=b$ for solving x $= O(n)$
A – diagonal

Example 2 :

$$L = \begin{bmatrix} l_{11} & & 0 \\ \vdots & l_{22} & \\ l_{n1} & \cdots & l_{nn} \end{bmatrix}$$

lower triangular matrix

$L_{ii} \neq 0$

$L_{ij} = 0 \quad j > i$

Solve :

$$\begin{bmatrix} -2 & 0 & 0 \\ 1 & 3 & 0 \\ 2 & -2 & 4 \end{bmatrix} x = \begin{bmatrix} 9 \\ -1 \\ 8 \end{bmatrix}$$

$$\underbrace{\begin{aligned} -2x_1 &= 9 \\ x_1 + 3x_2 &= -1 \\ 2x_1 - 2x_2 + 4x_3 &= 8 \end{aligned}}$$

$\Longrightarrow \quad x_1 = -\frac{9}{2}$

$$x_2 = \frac{-1 - x_1}{3} = \frac{-1 + \frac{9}{2}}{3} = \frac{7}{6}$$

$$x_3 = \frac{8 - 2x_1 + 2x_2}{4} = 2 + \frac{9}{4} + \frac{7}{12} = \frac{58}{12}$$

$$x_1 = \frac{b_1}{L_{11}}$$

$$\boxed{x_j = \frac{b_j - \sum_{k=1}^{j-1} L_{kj} x_k}{L_{jj}}} \leftarrow (*)$$

Solve: $Lx = b$ $\quad L_{ii} \neq 0$ and $l_{ij} = 0,\ j > i$

Forward substitution :- [Technique]

$$x_1 = \frac{b_1}{L_{11}}.$$

for $j = 2, \dots n$
  $\alpha = b_j$
  for $k = 1 \dots j-1$
    $\alpha = \alpha - L_{kj}\, x_k$
  end
  $x_j = \frac{\alpha}{L_{jj}}$
end

Example:- $U_{n\times n}$ – upper triangular

$$U = \begin{bmatrix} u_{11} & u_{12} & \cdots & u_{1n} \\ 0 & \ddots & & \vdots \\ 0 & & \ddots & \vdots \\ 0 & \cdots & 0 & u_{nn} \end{bmatrix}$$

$u_{ii} \neq 0$ $\quad U_{ij} = 0$ $\quad i < j$

Solve: $Ux = b$ $\quad U_{ii} \neq 0$ and $U_{ij} = 0,\ i < j$

Backward substitution :-

$$\boxed{x_n = \frac{b_n}{u_{nn}} \qquad x_j = \frac{b_j - \sum_{k=j+1}^{n} u_{kj} x_k}{u_{jj}}}$$

- $x_n = \frac{b_n}{u_{nn}}$

for $j = n-1, \dots, 1$
  $\alpha = b_j$
  for $k = j+1 \dots n$
    $\alpha = \alpha - U_{kj}\, x_k$
  end
  $x_j = \frac{\alpha}{u_{jj}}$
end

Question:- # of computations in backward/forward substitutions is $O(n^{\alpha})$ $\alpha = ?$

General $A_{n\times n}$: Solve $Ax = b$

Gaussian Elimination : Use linear Algebra to reduce $Ax=b \rightarrow Ux = b'$

- (via elementary row operations)

where U - upper triangular matrix

Review

So that solutions to both system are the same.

Example :-

$$\begin{bmatrix} -3 & 2 & -1 \\ 6 & -6 & 7 \\ 3 & -4 & 4 \end{bmatrix} x = \begin{bmatrix} -1 \\ -7 \\ -6 \end{bmatrix}$$

Consider the augmented matrix

$$[A \vdots b] = \left[\begin{array}{ccc|c} -3 & 2 & -1 & -1 \\ 6 & -6 & 7 & -7 \\ 3 & -4 & 4 & -6 \end{array}\right]$$

Reduce $[A:b]$ to Echelon form via Elementary row operations

$R_2 \rightarrow R_2 + 2R_1$

$$\left[\begin{array}{ccc|c} -3 & 2 & -1 & -1 \\ 0 & -2 & 5 & -9 \\ 3 & -4 & 4 & -6 \end{array}\right]$$

$R_3 \rightarrow R_1 + R_3$

$$\left[\begin{array}{ccc|c} 3 & 2 & -1 & -1 \\ 0 & -2 & 5 & -9 \\ 0 & -2 & 3 & -7 \end{array}\right]$$

$R_3 \to R_3 - R_2$

$$\left[\begin{array}{ccc|c} 3 & 2 & -1 & -1 \\ 0 & -2 & 5 & -9 \\ 0 & 0 & -2 & 2 \end{array}\right]$$

System reduced to $U x = b'$

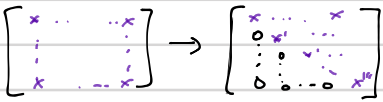
with $U = \begin{bmatrix} 3 & 2 & -1 \\ 0 & -2 & 5 \\ 0 & 0 & -2 \end{bmatrix}$ & $b' = \begin{bmatrix} -1 \\ -9 \\ 2 \end{bmatrix}$

$\underbrace{\qquad\qquad\qquad\qquad}_{\text{Upper Dar}}$

$$\left.\begin{aligned} x_3 &= \frac{2}{-2} = -1 \\ x_2 &= \frac{-9 - 5x_3}{-2} = 2 \\ x_1 &= \frac{-1 - 2x_2 + x_3}{-3} = 2 \end{aligned}\right\} \text{Back substitution}$$

Naive GE

Algorithm :-



```
C = [A:b]
for i in 1 to n-1
    for k in i+1 to n
        for j in i to n+1
            C_kj = C_kj - (C_ki / C_ii) C_ij
        end
    end
end
```

- Apply Back substitution Algorithm.

- The above algorithm may not work as is

  — if $c_{ii} = 0$ or c_{ii} is less than machine precision

- $\begin{cases} x_1 + \delta x_2 = \delta \\ x_1 + x_2 = 2 \end{cases}$ [order of the element] δ – large

Naive GE

Above Algorithm:

$R_2 \to R_2 - R_1$

$$\left.\begin{aligned} x_1 + \delta x_2 &= \delta \\ (1-\delta)x_2 &= 2-\delta \end{aligned}\right\} \text{Upper Dar}$$

$$\Rightarrow \quad x_2 = \frac{2-\delta}{1-\delta} \approx 1$$

$$x_1 = \delta - \delta x_2 \approx 0$$

$\boxed{\text{ERROR}}$

Altered Algorithm:

$R_2 \leftrightarrow R_1$

$$x_1 + x_2 = 2$$
$$x_1 + \delta x_2 = \delta$$

$R_2 \to R_2 - R_1$

$$x_1 + x_2 = 2$$
$$+(\delta - 1)x_2 = \delta - 2 \checkmark$$

$$\Rightarrow \quad x_2 = \frac{\delta - 2}{\delta - 1} \approx 1, \quad x_1 = 2 - x_2 \approx 1$$

$\boxed{\text{Closer to true Solution}}$

- Overall order of coefficients matter. (Comparitively)

Pivoting :-

- one would then have to interchange rows and proceed. This is called pivoting.

Question :- Which Row to interchange with ?

[effect of $c_{ii} = 0$]

Pivoting 1 :- find (Smallest) i_p : $|c_{i_p i}| = \max\{|c_{ki}|\ k \geq i\}$
and Swap Row $\underbrace{i_p \text{ and Row } i}$

Pv1 Algorithm :-

$C = [A : b]$
for i in 1 to $n-1$
  find i_p : $|c_{i_p i}| = \max\{|c_{ki}| : k \geq i\}$
  Row $i \Leftrightarrow$ Row i_p.

  for k in $i+1$ to n
    for j in i to $n+1$
      $c_{kj} = c_{kj} - \frac{c_{ki}}{c_{ii}} c_{ij}$
    end
  end
end

- Apply Back substitution Algorithm.

Question :- # of computations in Gaussian Elimination + back - Substitutions is $O(n^{\beta})$ $\beta = ?$

Pivoting2: $S_i = \max_{1 \le j \le n} |C_{ij}|$ for $1 \le i \le n$.

find i_p (Smallest) $\frac{|C_{i_p i}|}{S_{i_p}} = \max\left\{ \frac{|C_{ki}|}{S_k} : k \ge i \right\}$

Swap Row i_p and R_i

PV2 Algorithm:-

$C = [A : b]$

Compute $S_i = \max_{1 \le j \le n} |C_{ij}|$ $1 \le i \le n$.

for i in 1 to $n-1$

find i_p : $\frac{|C_{i_p i}|}{S_{i_p}} = \max\left\{ \frac{|C_{ki}|}{S_k} : k \ge i \right\}$
Smallest

Row i $\Leftrightarrow$ Row i_p.

for k in $i+1$ to n

for j in i to $n+1$

$C_{kj} = C_{kj} - \frac{C_{ki}}{C_{ii}} C_{ij}$

end

end

end

- Apply Back substitution Algorithm

Question:- How to handle Row swap in a program?