Resolução de Sistemas Lineares

INF1608 – Análise Numérica

Waldemar Celes
celes@inf.puc-rio.br

Departamento de Informática, PUC-Rio

Sistema de Equações Lineares

$$A_{n \times n}\mathbf{x}_n = \mathbf{b}_n$$

Sistema de Equações Lineares

$$A_{n \times n}\mathbf{x}_n = \mathbf{b}_n$$

Exemplo:

$$\begin{aligned} x + y &= 3 \\ 3x - 4y &= 2 \end{aligned}$$

Sistema de Equações Lineares

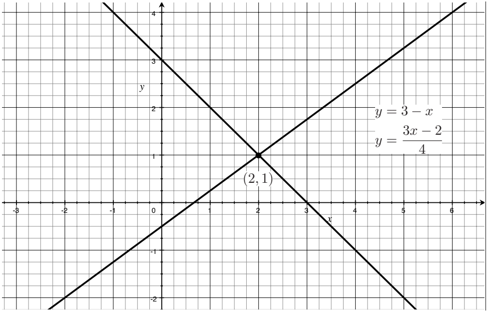
$$A_{n\times n}\mathbf{x}_n = \mathbf{b}_n$$

Exemplo:

$$x + y = 3$$
$$3x - 4y = 2$$

Ponto de vista geométrico

- Interseção de duas retas no plano xy



Sistema de Equações Lineares

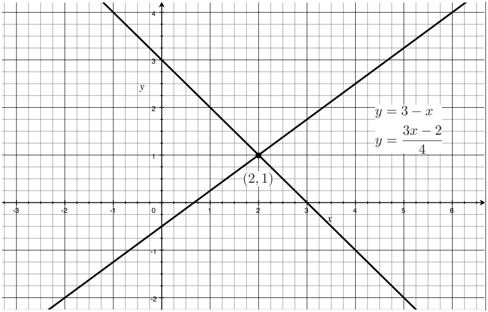
$$A_{n\times n}\mathbf{x}_n = \mathbf{b}_n$$

Exemplo:

$$x + y = 3$$
$$3x - 4y = 2$$

Ponto de vista geométrico

- Interseção de duas retas no plano xy



Como estender para n equações?

Eliminação de Gauss

Propriedade O sistema não se altera se:

- Trocarmos a ordem das equações
- Multiplicarmos uma equação por k tal que $k \neq 0$
- Adicionarmos (ou subtrairmos) uma equação por outra
  - A equação resultante substitui uma das duas

Eliminação de Gauss

Propriedade O sistema não se altera se:

- Trocarmos a ordem das equações
- Multiplicarmos uma equação por k tal que $k \neq 0$
- Adicionarmos (ou subtrairmos) uma equação por outra
  - A equação resultante substitui uma das duas

Exemplo:

$$\begin{aligned} x + 2y - z &= 3 \\ 2x + y - z &= 3 \\ -3x + y + z &= -6 \end{aligned}$$

Eliminação de Gauss

Propriedade O sistema não se altera se:

- Trocarmos a ordem das equações
- Multiplicarmos uma equação por k tal que $k \neq 0$
- Adicionarmos (ou subtrairmos) uma equação por outra
  - A equação resultante substitui uma das duas

Exemplo:

$$\begin{aligned} x + 2y - z &= 3 \\ 2x + y - z &= 3 \\ -3x + y + z &= -6 \end{aligned}$$

Forma matricial:

$$A \quad \mathbf{x} \quad = \quad \mathbf{b}$$

$$\begin{array}{cc} & \longrightarrow j \\ \begin{array}{c} \downarrow \\ i \\ \end{array} & \begin{bmatrix} 1 & 2 & -1 \\ 2 & 1 & -2 \\ -3 & 1 & 1 \end{bmatrix} \end{array} \begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} 3 \\ 3 \\ -6 \end{bmatrix}$$

Eliminação de Gauss

Objetivo

- Transformar a matriz A em uma matriz triangular superior:

$$a_{ij} = 0 \quad \forall \quad i > j$$

Eliminação de Gauss

Objetivo

- Transformar a matriz A em uma matriz triangular superior:

$$a_{ij} = 0 \quad \forall \quad i > j$$

- Eliminar os elementos da coluna j abaixo da diagonal ($i > j$)
  - Para eliminar aij, fazemos:

$$L_i \leftarrow L_i - \frac{a_{ij}}{a_{jj}} L_p$$

$$\text{onde } L_i = a_{ik}, k = 0, 1, 2, ...$$

- O valor $f_{ij} = \frac{a_{ij}}{a_{jj}}$ é chamado **fator** da eliminação

Eliminação de Gauss

Para a coluna $j = 0$:

- Para eliminar a_{10}:
  - Fator

$$f_{10} = \frac{a_{10}}{a_{00}} = \frac{2}{1} = 2$$

  - Alterando a L_1

$$L_1 = L_1 - f_{10} L_0$$

Representação matricial compacta:

$$\left[\begin{array}{rrr|r} 1 & 2 & -1 & 3 \\ 2 & 1 & -2 & 3 \\ -3 & 1 & 1 & -6 \end{array}\right] \longrightarrow \left[\begin{array}{rrr|r} 1 & 2 & -1 & 3 \\ \mathbf{0} & \mathbf{-3} & \mathbf{0} & \mathbf{-3} \\ -3 & 1 & 1 & -6 \end{array}\right]$$

Eliminação de Gauss

Para a coluna $j = 0$:

- Para eliminar a_{20}:
  - Fator

$$f_{20} = \frac{a_{20}}{a_{00}} = \frac{-3}{1} = -3$$

  - Alterando a L_2

$$L_2 = L_2 - f_{20} L_0$$

Representação matricial compacta:

$$\left[\begin{array}{ccc|c} 1 & 2 & -1 & 3 \\ 0 & -3 & 0 & -3 \\ -3 & 1 & 1 & -6 \end{array}\right] \longrightarrow \left[\begin{array}{ccc|c} 1 & 2 & -1 & 3 \\ 0 & -3 & 0 & -3 \\ \mathbf{0} & \mathbf{7} & \mathbf{-2} & \mathbf{3} \end{array}\right]$$

Eliminação de Gauss

Para a coluna $j = 1$:

- Para eliminar a_{21}:
  - Fator

$$f_{21} = \frac{a_{21}}{a_{11}} = \frac{7}{-3} = -\frac{7}{3}$$

  - Alterando a L_2

$$L_2 = L_2 - f_{21} L_1$$

Representação matricial compacta:

$$\left[\begin{array}{ccc|c} 1 & 2 & -1 & 3 \\ 0 & -3 & 0 & -3 \\ 0 & 7 & -2 & 3 \end{array}\right] \longrightarrow \left[\begin{array}{ccc|c} 1 & 2 & -1 & 3 \\ 0 & -3 & 0 & -3 \\ \mathbf{0} & \mathbf{0} & \mathbf{-2} & \mathbf{-4} \end{array}\right]$$

Eliminação de Gauss

Retro-substituição (*back substitution*)

- Determinar x_2, x_1, x_0, nesta ordem

$$a_{ii}\,x_i + \sum_{j=i+1}^{n-1} a_{ij} \underbrace{x_j}_{\text{já conhecidos, pois } j>i} = b_i$$

Eliminação de Gauss

Retro-substituição (*back substitution*)

- Determinar x_2, x_1, x_0, nesta ordem

$$a_{ii}\,x_i + \sum_{j=i+1}^{n-1} a_{ij} \underbrace{x_j}_{\text{já conhecidos, pois } j > i} = b_i$$

$$\therefore\; x_i = \frac{b_i - \sum_{j=i+1}^{n-1} a_{ij} x_j}{a_{ii}}$$

Eliminação de Gauss

Retro-substituição no exemplo

$$\begin{bmatrix} 1 & 2 & -1 \\ 0 & -3 & 0 \\ \mathbf{0} & \mathbf{0} & -\mathbf{2} \end{bmatrix} \begin{bmatrix} x_0 \\ x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} 3 \\ -3 \\ -\mathbf{4} \end{bmatrix}$$

$$x_2 = \frac{-4}{-2} = 2$$

Eliminação de Gauss

Retro-substituição no exemplo

$$\begin{bmatrix} 1 & 2 & -1 \\ 0 & -3 & 0 \\ \mathbf{0} & \mathbf{0} & \mathbf{-2} \end{bmatrix} \begin{bmatrix} x_0 \\ x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} 3 \\ -3 \\ \mathbf{-4} \end{bmatrix} \qquad x_2 = \frac{-4}{-2} = 2$$

$$\begin{bmatrix} 1 & 2 & -1 \\ \mathbf{0} & \mathbf{-3} & \mathbf{0} \\ 0 & 0 & -2 \end{bmatrix} \begin{bmatrix} x_0 \\ x_1 \\ 2 \end{bmatrix} = \begin{bmatrix} 3 \\ \mathbf{-3} \\ -4 \end{bmatrix} \qquad x_1 = \frac{-3-0}{-3} = 1$$

Eliminação de Gauss

Retro-substituição no exemplo

$$\begin{bmatrix} 1 & 2 & -1 \\ 0 & -3 & 0 \\ \mathbf{0} & \mathbf{0} & -\mathbf{2} \end{bmatrix} \begin{bmatrix} x_0 \\ x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} 3 \\ -3 \\ -\mathbf{4} \end{bmatrix} \qquad x_2 = \frac{-4}{-2} = 2$$

$$\begin{bmatrix} 1 & 2 & -1 \\ \mathbf{0} & -\mathbf{3} & \mathbf{0} \\ 0 & 0 & -2 \end{bmatrix} \begin{bmatrix} x_0 \\ x_1 \\ 2 \end{bmatrix} = \begin{bmatrix} 3 \\ -\mathbf{3} \\ -4 \end{bmatrix} \qquad x_1 = \frac{-3-0}{-3} = 1$$

$$\begin{bmatrix} \mathbf{1} & \mathbf{2} & -\mathbf{1} \\ 0 & -3 & 0 \\ 0 & 0 & -2 \end{bmatrix} \begin{bmatrix} x_0 \\ 1 \\ 2 \end{bmatrix} = \begin{bmatrix} \mathbf{3} \\ -3 \\ -4 \end{bmatrix} \qquad x_0 = \frac{3-(2-2)}{1} = 3$$

Eliminação de Gauss

Algoritmo: **Eliminação**

for $j = 0$ **to** $n-2$
    elimina coluna j
    for $i = j+1$ **to** $n-1$
        elimina a_{ij}
        $f = \dfrac{a_{ij}}{a_{jj}}$
        for $k = j$ **to** $n-1$
            $a_{ik} = a_{ik} - a_{jk}\,f$
        $b_i = b_i - b_j\,f$

- Note que o **for** de k pode começar em $k = j+1$, pois sabemos que a_{ij} resulta em zero e que valor não será usado

Eliminação de Gauss

Algoritmo: **Retro-substituição**

$$
\begin{array}{l}
\textbf{for } i = n-1 \textbf{ to } 0 \textbf{ step } -1 \\
\quad s = 0 \\
\quad \textbf{for } j = i+1 \textbf{ to } n-1 \\
\quad\quad s = s + a_{ij}\, x_j \\
\quad x_i = \dfrac{b_i - s}{a_{ii}}
\end{array}
$$

Eliminação de Gauss

Algoritmo: **Retro-substituição**

$$
\begin{aligned}
&\textbf{for } i = n-1 \textbf{ to } 0 \textbf{ step } -1\\
&\qquad s = 0\\
&\qquad \textbf{for } j = i+1 \textbf{ to } n-1\\
&\qquad\qquad s = s + a_{ij}\, x_j\\
&\qquad x_i = \frac{b_i - s}{a_{ii}}
\end{aligned}
$$

Custo computacional:

- Eliminação:
- Retro-substituição:

Eliminação de Gauss

Algoritmo: **Retro-substituição**

$$
\begin{array}{l}
\textbf{for } i = n-1 \textbf{ to } 0 \textbf{ step } -1 \\
\quad s = 0 \\
\quad \textbf{for } j = i+1 \textbf{ to } n-1 \\
\quad\quad s = s + a_{ij}\, x_j \\
\quad x_i = \dfrac{b_i - s}{a_{ii}}
\end{array}
$$

Custo computacional:

- Eliminação: $O(n^3)$
- Retro-substituição: $O(n^2)$
- **Total**: $O(n^3)$

Medição de erros

Definições

- Norma máxima de um vetor

$$\|x\| = \max_i |x_i|$$

Medição de erros

Definições

- Norma máxima de um vetor

$$\|x\| = \max_i |x_i|$$

- Norma máxima de uma matriz

$$\|A\| = \max_i \left(\sum_j |a_{ij}| \right)$$

Medição de erros

Resolução de sistemas lineares

- Erro regressivo (avaliado na entrada)

Medição de erros

Resolução de sistemas lineares

- Erro regressivo (avaliado na entrada)

$$\underbrace{\mathbf{r}}_{\text{resíduo}} = \mathbf{b} - A\mathbf{x}_c, \text{ onde } \mathbf{x}_c \text{ é a solução computada}$$

Medição de erros

Resolução de sistemas lineares

- Erro regressivo (avaliado na entrada)

$$\underbrace{\mathbf{r}}_{\text{resíduo}} = \mathbf{b} - A\mathbf{x}_c, \text{ onde } \mathbf{x}_c \text{ é a solução computada}$$

$$e_{reg} = \|\mathbf{b} - A\mathbf{x}\|$$

- Erro progressivo (avaliado na saída)

$$e_{prog} = \|\mathbf{x}_e - \mathbf{x}_c\|$$

se conhecêssemos $\mathbf{x}_e$

Medição de erros

Erros relativos

$$e_{reg-rel} = \frac{\|\mathbf{b} - A\mathbf{x}_c\|}{\|\mathbf{b}\|} = \frac{\|\mathbf{r}\|}{\|\mathbf{b}\|}$$

$$e_{prog-rel} = \frac{\|\mathbf{x}_e - \mathbf{x}_c\|}{\|\mathbf{x}_e\|}$$

Medição de erros

Fontes de erros

- Divisão por zero (pivô igual a zero ou muito pequeno)
  - Pode ser evitado
- Matriz mal condicionada
  - Não pode ser evitado

Medição de erros

Fontes de erros

- Divisão por zero (pivô igual a zero ou muito pequeno)
  - Pode ser evitado
- Matriz mal condicionada
  - Não pode ser evitado

Fator de ampliação do erro

$$F_{AE} = \frac{e_{prog-rel}}{e_{reg-rel}}$$

Medição de erros

Definição: Número de Condicionamento

- Representa o máximo fator de ampliação de erro na solução de $A\mathbf{x} = \mathbf{b}$, para qualquer $\mathbf{b}$

$$cond(A) = \|A\| \|A^{-1}\|$$

Medição de erros

Definição: Número de Condicionamento

- Representa o máximo fator de ampliação de erro na solução de $A\mathbf{x} = \mathbf{b}$, para qualquer $\mathbf{b}$

$$cond(A) = \|A\| \|A^{-1}\|$$

Exemplo:

$$A = \begin{bmatrix} 1 & 1 \\ 1.0001 & 1 \end{bmatrix}$$

Medição de erros

Definição: Número de Condicionamento

- Representa o máximo fator de ampliação de erro na solução de $A\mathbf{x} = \mathbf{b}$, para qualquer $\mathbf{b}$

$$cond(A) = \|A\| \|A^{-1}\|$$

Exemplo:

$$A = \begin{bmatrix} 1 & 1 \\ 1.0001 & 1 \end{bmatrix} \qquad A^{-1} = \begin{bmatrix} -10000 & 10000 \\ 10001 & -10000 \end{bmatrix}$$

Medição de erros

Definição: Número de Condicionamento

- Representa o máximo fator de ampliação de erro na solução de $A\mathbf{x} = \mathbf{b}$, para qualquer $\mathbf{b}$

$$cond(A) = \|A\| \|A^{-1}\|$$

Exemplo:

$$A = \begin{bmatrix} 1 & 1 \\ 1.0001 & 1 \end{bmatrix} \qquad A^{-1} = \begin{bmatrix} -10000 & 10000 \\ 10001 & -10000 \end{bmatrix}$$

Logo:

$$cond(A) = 2.0001\ 20001 \approx 40004$$

- A matriz A é mal condicionada
  - Dificilmente teremos controle do erro da solução

Eliminação de Gauss

Tratamento de divisão por zero e afins

- Troca de linhas durante Eliminação de Gauss
  - Evitar divisão por zero (ou por um número pequeno)

Eliminação de Gauss

Tratamento de divisão por zero e afins

- Troca de linhas durante Eliminação de Gauss
  - Evitar divisão por zero (ou por um número pequeno)

Exemplo sem troca de linhas: $f = 10^{20}$

$$\left[\begin{array}{cc|c} 10^{-20} & 1 & 1 \\ 1 & 2 & 4 \end{array}\right] \rightarrow \left[\begin{array}{cc|c} 10^{-20} & 1 & 1 \\ 0 & 2 - 10^{20} & 4 - 10^{20} \end{array}\right] \therefore x = \left[\begin{array}{c} 0 \\ 1 \end{array}\right]$$

Eliminação de Gauss

Tratamento de divisão por zero e afins

- Troca de linhas durante Eliminação de Gauss
  - Evitar divisão por zero (ou por um número pequeno)

Exemplo sem troca de linhas: $f = 10^{20}$

$$\left[\begin{array}{cc|c} 10^{-20} & 1 & 1 \\ 1 & 2 & 4 \end{array}\right] \rightarrow \left[\begin{array}{cc|c} 10^{-20} & 1 & 1 \\ 0 & 2-10^{20} & 4-10^{20} \end{array}\right] \therefore x = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$$

Exemplo com troca de linhas: $f = 10^{-20}$

$$\left[\begin{array}{cc|c} 1 & 2 & 4 \\ 10^{-20} & 1 & 1 \end{array}\right] \rightarrow \left[\begin{array}{cc|c} 1 & 2 & 4 \\ 0 & 1-2\times 10^{-20} & 1-4\times 10^{-20} \end{array}\right] \therefore x = \begin{bmatrix} 2 \\ 1 \end{bmatrix}$$

- Solução ainda não exata, mas próxima

Pivotamento

Objetivo: Manter $|f| \leq 1$

- Antes da eliminação em cada coluna, identifica-se a linha com elemento na coluna de maior valor absoluto e trocam-se as linhas, mantendo o maior valor absoluto como pivô.

Logo, o pivô será:

$$|a_{pj}| \geq |a_{ij}| \;\; \forall \;\; j \leq i \leq n-1$$

Pivotamento

Algortimo

$\ldots$

antes da eliminação da coluna j

$p = j$

for $k = j + 1$ **to** $n - 1$

  if $|a_{kj}| > |a_{pj}|$

    $p = k$

troca linhas j e p

for $k = j$ **to** $n - 1$

  $a_{jk} \leftrightarrow a_{pk}$

$b_j \leftrightarrow b_p$

$\ldots$

Fatoração LU

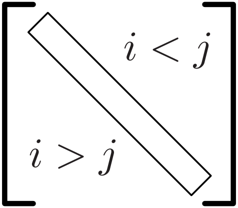
Representação matricial da Eliminação de Gauss

$$\text{Matriz triangular} \begin{cases} \text{inferior } (L) \\ \text{superior } (U) \end{cases}$$

Fatoração LU

Representação matricial da Eliminação de Gauss

$$\text{Matriz triangular} \begin{cases} \text{inferior } (L) \\ \text{superior } (U) \end{cases}$$



$$l_{ij} = 0 \;\; \forall \;\; i < j$$
$$u_{ij} = 0 \;\; \forall \;\; i > j$$

Fatoração LU

Sistema

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

Fatoração da matriz

$$A = L\,U$$

Fatoração LU

Sistema

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

Fatoração da matriz

$$A = L\,U$$

- Eliminação de Gauss $\longrightarrow$ produz U
- Como determinar L?

Fatoração LU

Sistema

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

Fatoração da matriz

$$A = L\,U$$

- Eliminação de Gauss $\longrightarrow$ produz U
- Como determinar L?

$$l_{ij} = \begin{cases} 1, \text{ se } i = j \text{ (diagonal)} \\ f_{ij} = \frac{a_{ij}}{a_{jj}}, \text{ se } i > j \end{cases}$$

- Elementos inferiores são os fatores da Eliminação de Gauss

Fatoração LU

Exemplo

$$\begin{bmatrix} 2 & 2 \\ 3 & -4 \end{bmatrix}$$

Fatoração LU

Exemplo

$$\begin{bmatrix} 2 & 2 \\ 3 & -4 \end{bmatrix}$$

- Eliminação: $f_{10} = \frac{3}{2}$

$$U = \begin{bmatrix} 2 & 2 \\ 3 - 2\frac{3}{2} & -4 - 2\frac{3}{2} \end{bmatrix} = \begin{bmatrix} 2 & 2 \\ 0 & -7 \end{bmatrix}$$

Fatoração LU

Exemplo

$$\begin{bmatrix} 2 & 2 \\ 3 & -4 \end{bmatrix}$$

- Eliminação: $f_{10} = \frac{3}{2}$

$$U = \begin{bmatrix} 2 & 2 \\ 3 - 2\frac{3}{2} & -4 - 2\frac{3}{2} \end{bmatrix} = \begin{bmatrix} 2 & 2 \\ 0 & -7 \end{bmatrix}$$

- Triangular inferior

$$L = \begin{bmatrix} 1 & 0 \\ \frac{3}{2} & 1 \end{bmatrix}$$

Fatoração LU

Exemplo

$$\begin{bmatrix} 2 & 2 \\ 3 & -4 \end{bmatrix}$$

- Eliminação: $f_{10} = \frac{3}{2}$

$$U = \begin{bmatrix} 2 & 2 \\ 3 - 2\frac{3}{2} & -4 - 2\frac{3}{2} \end{bmatrix} = \begin{bmatrix} 2 & 2 \\ 0 & -7 \end{bmatrix}$$

- Triangular inferior

$$L = \begin{bmatrix} 1 & 0 \\ \frac{3}{2} & 1 \end{bmatrix}$$

Comprovação

$$LU = \begin{bmatrix} 1 & 0 \\ \frac{3}{2} & 1 \end{bmatrix} \begin{bmatrix} 2 & 2 \\ 0 & -7 \end{bmatrix} = \begin{bmatrix} 2 & 2 \\ 3 & -4 \end{bmatrix} = A$$

Fatoração LU

Dado que $A = LU$, como resolver o sistema?

Fatoração LU

Dado que $A = LU$, como resolver o sistema?

$$
\begin{aligned}
A\mathbf{x} &= \mathbf{b} \\
L\,U\mathbf{x} &= \mathbf{b}
\end{aligned}
$$

$$
\begin{cases}
L\mathbf{y} & = \mathbf{b} \\
U\mathbf{x} & = \mathbf{y}
\end{cases}
$$

- Acha-se **y** por **substituição progressiva**
- Acha-se **x** por **substituição regressiva** (retro-substituição)

Fatoração LU

Dado que $A = LU$, como resolver o sistema?

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

$$L\,U\mathbf{x} = \mathbf{b}$$

$$\begin{cases} L\mathbf{y} & = \mathbf{b} \\ U\mathbf{x} & = \mathbf{y} \end{cases}$$

- Acha-se **y** por **substituição progressiva**
- Acha-se **x** por **substituição regressiva** (retro-substituição)

Note que a fatoração de A não altera b

- Solução para diferentes b pode ser obtida em tempo $O(n^2)$
  - Mas o tempo de fatoração é $O(n^3)$

Fatoração $PA = LU$

Como combinar fatoração e pivotamento?

Fatoração $PA = LU$

Como combinar fatoração e pivotamento?

- É necessário registrar as permutações
  - Garantir correta permutação dos elementos de $\mathbf{b}$
  - Garantir correta permutação dos elementos de L (fatores)
    - U não é afetada por permutações pois elementos são nulos

Fatoração $PA = LU$

Como combinar fatoração e pivotamento?

- É necessário registrar as permutações
  - Garantir correta permutação dos elementos de $\mathbf{b}$
  - Garantir correta permutação dos elementos de L (fatores)
    - U não é afetada por permutações pois elementos são nulos

Matriz de permutação: P

- Registra permutação de linhas
- Obtida trocando-se linhas da matriz identidade
  - Exemplo: troca da segunda com terceira linha

$$P = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & 0 \end{bmatrix}$$

Fatoração $PA = LU$

Como combinar fatoração e pivotamento?

- É necessário registrar as permutações
  - Garantir correta permutação dos elementos de $\mathbf{b}$
  - Garantir correta permutação dos elementos de L (fatores)
    - U não é afetada por permutações pois elementos são nulos

Matriz de permutação: P

- Registra permutação de linhas
- Obtida trocando-se linhas da matriz identidade
  - Exemplo: troca da segunda com terceira linha

$$P = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & 0 \end{bmatrix}$$

Note que:

- $PA \rightarrow$ troca linhas de A
- $P\mathbf{b} \rightarrow$ troca linhas (elementos) de $\mathbf{b}$

Fatoração $PA = LU$

Como armazenar LU de forma otimizada?

Fatoração $PA = LU$

Como armazenar LU de forma otimizada?

- Transforma A em LU, *in place*
- Diagonal de L implícita
- Elimina necessidade de pós-permutar L

Fatoração $PA = LU$

Como armazenar LU de forma otimizada?

- Transforma A em LU, *in place*
- Diagonal de L implícita
- Elimina necessidade de pós-permutar L

$$\begin{bmatrix} a_{00} & a_{01} & a_{02} \\ a_{10} & a_{11} & a_{12} \\ a_{20} & a_{21} & a_{22} \end{bmatrix} \longrightarrow \begin{bmatrix} u_{00} & u_{01} & u_{02} \\ l_{10} & u_{11} & u_{12} \\ l_{20} & l_{21} & u_{22} \end{bmatrix}$$

Fatoração $PA = LU$

Resolvendo o sistema

$$\begin{aligned} A\mathbf{x} &= P\mathbf{b} \\ L\,U\mathbf{x} &= P\mathbf{b} \\ &\begin{cases} L\mathbf{y} & = P\mathbf{b} \\ U\mathbf{x} & = \mathbf{y} \end{cases} \end{aligned}$$

Fatoração $PA = LU$

Resolvendo o sistema

$$
\begin{aligned}
A\mathbf{x} &= P\mathbf{b} \\
LU\mathbf{x} &= P\mathbf{b}
\end{aligned}
$$

$$
\begin{cases}
L\mathbf{y} & = P\mathbf{b} \\
U\mathbf{x} & = \mathbf{y}
\end{cases}
$$

Armazenamento das permutações de forma otimizada

Fatoração $PA = LU$

Resolvendo o sistema

$$
\begin{aligned}
A\mathbf{x} &= P\mathbf{b} \\
LU\mathbf{x} &= P\mathbf{b} \\
&\begin{cases} L\mathbf{y} &= P\mathbf{b} \\ U\mathbf{x} &= \mathbf{y} \end{cases}
\end{aligned}
$$

Armazenamento das permutações de forma otimizada

- Registro das permutações em um vetor $\mathbf{p}$
  - Inicialização: $\mathbf{p}_i = i$
  - Troca de i com j: $\mathbf{p}_i \leftrightarrow \mathbf{p}_j$
  - Permutação em $\mathbf{b}$: $\mathbf{b}_{\mathbf{p}_i}$

Fatoração

Podemos diminuir o espaço de memória da fatoração?

- Se A for simétrica, é possível usar metade do espaço?

Fatoração

Podemos diminuir o espaço de memória da fatoração?

- Se A for simétrica, é possível usar metade do espaço?

- Sim, com **Fatoração de Cholesky**,
  se matriz for **simétrica positiva definida**

Sistemas de Equações Lineares

Matriz Simétrica Positiva Definida

$$\mathbf{x}^T A\,\mathbf{x} > 0, \quad \forall \quad \mathbf{x} = \mathbf{0}$$

Propriedades:

1. Elementos da diagonal principal são positivos
2. Autovalores são todos positivos
3. Qualquer submatriz principal é também positiva definida

Fatoração de Cholesky

Dada a matriz A simétrica positiva definida

- Objetivo: encontrar matriz triangular superior R tal que:

$$A = R^T R$$

Fatoração de Cholesky

Dada a matriz A simétrica positiva definida

- Objetivo: encontrar matriz triangular superior R tal que:

$$A = R^T R$$

Caso $n = 2$:

$$A = \begin{bmatrix} a & b \\ b & c \end{bmatrix}$$

Fatoração de Cholesky

Dada a matriz A simétrica positiva definida

- Objetivo: encontrar matriz triangular superior R tal que:

$$A = R^T R$$

Caso $n = 2$:

$$A = \begin{bmatrix} a & b \\ b & c \end{bmatrix}$$

$$\begin{bmatrix} a & b \\ b & c \end{bmatrix} = \begin{bmatrix} \sqrt{a} & 0 \\ u & v \end{bmatrix} \begin{bmatrix} \sqrt{a} & u \\ 0 & v \end{bmatrix} = \begin{bmatrix} a & u\sqrt{a} \\ u\sqrt{a} & u^2 + v^2 \end{bmatrix}$$

Fatoração de Cholesky

Dada a matriz A simétrica positiva definida

- Objetivo: encontrar matriz triangular superior R tal que:

$$A = R^T R$$

Caso $n = 2$:

$$A = \begin{bmatrix} a & b \\ b & c \end{bmatrix}$$

$$\begin{bmatrix} a & b \\ b & c \end{bmatrix} = \begin{bmatrix} \sqrt{a} & 0 \\ u & v \end{bmatrix} \begin{bmatrix} \sqrt{a} & u \\ 0 & v \end{bmatrix} = \begin{bmatrix} a & u\sqrt{a} \\ u\sqrt{a} & u^2 + v^2 \end{bmatrix}$$

$$b = u\sqrt{a} \quad \therefore \quad u = \frac{b}{\sqrt{a}}$$

$$c = \frac{b^2}{a} + v^2 \quad \therefore \quad v = \sqrt{c - \frac{b^2}{a}}$$

Fatoração de Cholesky

Caso $n \times n$:

$$A = \left[\begin{array}{c|c} a & \mathbf{b}^T \\ \hline \mathbf{b} & C \end{array}\right]$$

Fatoração de Cholesky

Caso $n \times n$:

$$A = \left[\begin{array}{c|c} a & \mathbf{b}^T \\ \hline \mathbf{b} & C \end{array}\right]$$

$$R^T R = \left[\begin{array}{c|ccc} \sqrt{a} & 0 & \dots & 0 \\ \hline \mathbf{u} & & V^T & \end{array}\right] \left[\begin{array}{c|c} \sqrt{a} & \mathbf{u}^T \\ \hline 0 & \\ \vdots & V \\ 0 & \end{array}\right] =$$

$$= \left[\begin{array}{c|c} a & \sqrt{a}\mathbf{u}^T \\ \hline \sqrt{a}\mathbf{u} & \mathbf{u}\,\mathbf{u}^T + V^T V \end{array}\right] \begin{cases} \mathbf{u} = \frac{\mathbf{b}}{\sqrt{a}} \\ V^T V = C - \mathbf{u}\mathbf{u}^T = A_1 \end{cases}$$

Fatoração de Cholesky

Procedimento

- Primeira linha

$$r_{00} = \sqrt{a_{00}}$$

$$\mathbf{u}^T = \frac{\mathbf{b}^T}{r_{00}}$$

$$A_1 = C - \mathbf{u}\mathbf{u}^T$$

- onde:
  - A_1 é uma matriz $(n-1) \times (n-1)$
  - Por indução, $A_1 = V^T V$, onde V é triangular superior

Fatoração de Cholesky

Algoritmo (fatoração *in place*)

- Entrada: $A_{n\times n}$, triangular
- Saída: R^T, triangular

$$
\begin{array}{ll}
\textbf{for } k = 0 \textbf{ to } n-1 & \\
\quad A_{kk} = \sqrt{A_{kk}} & = r_{kk} \\
\quad\quad \textbf{for } i = k+1 \textbf{ to } n-1 & \\
\quad\quad\quad A_{ik} = A_{ik}/A_{kk} & = \mathbf{u} \ \textit{(vetor coluna)} \\
\quad\quad \textbf{for } i = k+1 \textbf{ to } n-1 & \\
\quad\quad\quad \textbf{for } j = k+1 \textbf{ to } i & \\
\quad\quad\quad\quad A_{ij} = A_{ij} - A_{ik}A_{kj} & = A_{ij} - \mathbf{u}\mathbf{u}^T
\end{array}
$$

Fatoração de Cholesky

Resolvendo o sistema

$$
\begin{aligned}
A\mathbf{x} &= \mathbf{b} \\
R^T R\mathbf{x} &= \mathbf{b}
\end{aligned}
$$

$$
\begin{cases}
R^T\mathbf{y} & = \mathbf{b} \\
R\mathbf{x} & = \mathbf{y}
\end{cases}
$$

- Acha-se $\mathbf{y}$ por **substituição progressiva**
- Acha-se $\mathbf{x}$ por **substituição regressiva** (retro-substituição)

Fatoração de Cholesky

Resolvendo o sistema

$$
\begin{aligned}
A\mathbf{x} &= \mathbf{b} \\
R^T R\mathbf{x} &= \mathbf{b}
\end{aligned}
$$

$$
\begin{cases}
R^T\mathbf{y} & = \mathbf{b} \\
R\mathbf{x} & = \mathbf{y}
\end{cases}
$$

- Acha-se $\mathbf{y}$ por **substituição progressiva**
- Acha-se $\mathbf{x}$ por **substituição regressiva** (retro-substituição)

Observação

- Na fatoração de Cholesky,
  não é necessário uso de pivotamento

Exercícios propostos

1. Considerando a resolução de sistemas lineares na forma $A\mathbf{x} = \mathbf{b}$, baseado no método de Eliminação de Gauss, a fatoração LU pode resultar numa única matriz F que representa, de forma compacta, os elementos de L e de U. Para uma matriz 2×2, temos:

$$F = \begin{bmatrix} u_{00} & u_{01} \\ l_{10} & u_{11} \end{bmatrix}$$

   Considere a matriz A dada por:

$$A = \begin{bmatrix} 1 & 2 \\ -2 & 1 \end{bmatrix}$$

   Indique os valores da matriz F resultante da fatoração LU de A considerando os casos de:

   1.1 Não usar pivotamento
   1.2 Usar pivotamento

Exercícios propostos

2. Ache a fatoração de Cholesky $A = R^T R$ para:

$$A = \begin{bmatrix} 1 & 2 \\ 2 & 8 \end{bmatrix}$$