

Eliminação de Gauss

Antonio Elias Fabris

Instituto de Matemática e Estatística
Universidade de São Paulo

Map 2210 – Aplicações de Álgebra Linear

Fatoração LU : Idéia

- Como na triangularização de Householder, eliminação de Gauss transforma uma matriz A numa triangular superior através de multiplicações à esquerda de matrizes triangulares inferiores por A
 - ▶ Mas na eliminação de Gauss as matrizes triangulares **não** são unitárias
- Seja $A \in \mathbf{C}^{m \times m}$. Novamente, como na triangularização de Householder, introduzimos sucessivamente zeros abaixo da diagonal principal
 - ▶ Para cada linha, subtraindo-se múltiplos desta linha das linhas subsequentes
 - ▶ Conforme comentado acima, este processo é equivalente às multiplicações à esquerda de matrizes triangulares inferiores L_k :

$$\underbrace{L_{m-1} \cdots L_2 L_1}_{L^{-1}} A = U$$

- Podemos então obter uma fatoração LU de A , isto é:

$$A = LU$$

L triangular inferior (diagonais iguais 1) e U triangular superior.

Fatoração LU : Triangularização por Triangulares

$$\begin{bmatrix} \times & \times & \times & \times \\ \times & \times & \times & \times \\ \times & \times & \times & \times \\ \times & \times & \times & \times \end{bmatrix} \xrightarrow{L_1} \begin{bmatrix} \times & \times & \times & \times \\ 0 & \times & \times & \times \\ 0 & \times & \times & \times \\ 0 & \times & \times & \times \end{bmatrix} \xrightarrow{L_2} \begin{bmatrix} \times & \times & \times & \times \\ & \times & \times & \times \\ 0 & \times & \times & \times \\ 0 & \times & \times & \times \end{bmatrix} \xrightarrow{L_3} \begin{bmatrix} \times & \times & \times & \times \\ & \times & \times & \times \\ & & \times & \times \\ & & 0 & \times \end{bmatrix}$$

$A \qquad L_1 A \qquad L_2 L_1 A \qquad L_3 L_2 L_1 A$

- A k -ésima transformação L_k introduz zeros abaixo da diagonal na coluna k subtraindo múltiplos da linha k das linhas $k+1, \dots, m$
- Como as $k-1$ entradas da linha k já são nulas, esta operação não destrói os zeros previamente introduzidos
- Nos próximos slides: L_k é triangular inferior com diagonal unitária
- Taxonomia dos algoritmos para a fatoração de uma matriz:
 - ▶ Gram-Schmidt: $A = QR$ ortogonalização por triangulares
 - ▶ Householder: $A = QR$ triangularização por ortogonais
 - ▶ Eliminação de Gauss: $A = LU$ triangularização por triangulares

Fatoração $A = LU$: Exemplo

$$A = \begin{bmatrix} 2 & 1 & 1 & 0 \\ 4 & 3 & 3 & 1 \\ 8 & 7 & 9 & 5 \\ 6 & 7 & 9 & 8 \end{bmatrix}$$

$$L_1 A = \begin{bmatrix} 1 & & & \\ -2 & 1 & & \\ -4 & & 1 & \\ -3 & & & 1 \end{bmatrix} \begin{bmatrix} 2 & 1 & 1 & 0 \\ 4 & 3 & 3 & 1 \\ 8 & 7 & 9 & 5 \\ 6 & 7 & 9 & 8 \end{bmatrix} = \begin{bmatrix} 2 & 1 & 1 & 0 \\ & 1 & 1 & 1 \\ & 3 & 5 & 5 \\ & 4 & 6 & 8 \end{bmatrix}$$

$$L_2 L_1 A = \begin{bmatrix} 1 & & & \\ & 1 & & \\ -3 & & 1 & \\ -4 & & & 1 \end{bmatrix} \begin{bmatrix} 2 & 1 & 1 & 0 \\ & 1 & 1 & 1 \\ & 3 & 5 & 5 \\ & 4 & 6 & 8 \end{bmatrix} = \begin{bmatrix} 2 & 1 & 1 & 0 \\ & 1 & 1 & 1 \\ & & 2 & 2 \\ & & 2 & 4 \end{bmatrix}$$

$$L_3 L_2 L_1 A = \begin{bmatrix} 1 & & & \\ & 1 & & \\ & & 1 & \\ & & -1 & 1 \end{bmatrix} \begin{bmatrix} 2 & 1 & 1 & 0 \\ & 1 & 1 & 1 \\ & & 2 & 2 \\ & & 2 & 4 \end{bmatrix} = \begin{bmatrix} 2 & 1 & 1 & 0 \\ & 1 & 1 & 1 \\ & & 2 & 2 \\ & & & 2 \end{bmatrix} = U$$

Fatoração $A = LU$: Dois Fatos

$$\textcircled{1} \quad L_1^{-1} = \begin{bmatrix} 1 & & & \\ -2 & 1 & & \\ -4 & & 1 & \\ -3 & & & 1 \end{bmatrix}^{-1} = \begin{bmatrix} 1 & & & \\ 2 & 1 & & \\ 4 & & 1 & \\ 3 & & & 1 \end{bmatrix}$$

$$\textcircled{2} \quad \underbrace{\begin{bmatrix} 1 & & & \\ 2 & 1 & & \\ 4 & & 1 & \\ 3 & & & 1 \end{bmatrix}}_{L_1^{-1}} \underbrace{\begin{bmatrix} 1 & & & \\ & 1 & & \\ & 3 & 1 & \\ & 4 & & 1 \end{bmatrix}}_{L_2^{-1}} \underbrace{\begin{bmatrix} 1 & & & \\ & 1 & & \\ & & 1 & \\ & & 1 & 1 \end{bmatrix}}_{L_3^{-1}} = \begin{bmatrix} 1 & & & \\ 2 & 1 & & \\ 4 & 3 & 1 & \\ 3 & 4 & 1 & 1 \end{bmatrix}$$

$$L_3 L_2 L_1 A = U \quad \Rightarrow \quad A = L_1^{-1} L_2^{-1} L_3^{-1} U$$

$$\underbrace{\begin{bmatrix} 2 & 1 & 1 & 0 \\ 4 & 3 & 3 & 1 \\ 8 & 7 & 9 & 5 \\ 6 & 7 & 9 & 8 \end{bmatrix}}_A = \underbrace{\begin{bmatrix} 1 & & & \\ 2 & 1 & & \\ 4 & 3 & 1 & \\ 3 & 4 & 1 & 1 \end{bmatrix}}_L \underbrace{\begin{bmatrix} 2 & 1 & 1 & 0 \\ & 1 & 1 & 1 \\ & & 2 & 2 \\ & & & 2 \end{bmatrix}}_U$$

Fórmulas Gerais - 1

$$x_k = \begin{bmatrix} x_{1k} \\ \vdots \\ x_{kk} \\ x_{k+1,k} \\ \vdots \\ x_{mk} \end{bmatrix} \xrightarrow{L_k=?} L_k x_k = \begin{bmatrix} x_{1k} \\ \vdots \\ x_{kk} \\ 0 \\ \vdots \\ 0 \end{bmatrix} \quad L_k = \begin{bmatrix} 1 & & & & \\ & \ddots & & & \\ & & 1 & & \\ & & -\frac{x_{k+1,k}}{x_{kk}} & 1 & \\ & & \vdots & & \ddots \\ & & -\frac{x_{mk}}{x_{kk}} & & & 1 \end{bmatrix}$$

- $l_k \stackrel{\text{def}}{=} \left[0 \dots 0 \frac{x_{k+1,k}}{x_{kk}} \dots \frac{x_{mk}}{x_{kk}} \right]^* \implies L_k = I - l_k e_k^*$

- $(I - l_k e_k^*)(I + l_k e_k^*) = I - l_k \underbrace{e_k^* l_k}_0 e_k^* = I \implies L_k^{-1} = I + l_k e_k^*$

Fórmulas Gerais - 2

- **Exercício 1:** Sendo $L_k^{-1} =$

$$\begin{bmatrix} 1 & & & & \\ & \ddots & & & \\ & & 1 & & \\ & & \frac{x_{k+1,k}}{x_{kk}} & 1 & \\ & & \vdots & & \ddots \\ & & \frac{x_{mk}}{x_{kk}} & & & 1 \end{bmatrix}$$
prove que

$$L = L_1^{-1} L_2^{-1} \dots L_{m-1}^{-1} = \begin{bmatrix} 1 & & & & \\ \frac{x_{21}}{x_{11}} & 1 & & & \\ \frac{x_{31}}{x_{11}} & \frac{x_{32}}{x_{22}} & 1 & & \\ \vdots & \vdots & \ddots & \ddots & \\ \frac{x_{m1}}{x_{11}} & \frac{x_{m2}}{x_{22}} & \dots & \frac{x_{m,m-1}}{x_{m-1,m-1}} & 1 \end{bmatrix}$$

- Sugestão: $L_k^{-1} L_{k+1}^{-1} = (I + l_k e_k^*)(I + l_{k+1} e_{k+1}^*) = I + l_k e_k^* + l_{k+1} e_{k+1}^*$

- Na prática, as matrizes L_k não são construídas nem multiplicadas explicitamente: os multiplicadores l_{jk} são calculados e armazenados em L e as transformações L_k são aplicadas implicitamente

Eliminação de Gauss sem Pivotamento

Entrada: $A = [a_{ij}] \in \mathbb{C}^{m \times m}$

Saída: $L = [l_{ij}] \in \mathbb{C}^{m \times m}$, $U = [u_{ij}] \in \mathbb{C}^{n \times m}$

```

1:  $U=A$ ,  $L=I$ 
2: for  $k = 1$  to  $m - 1$  do
3:   for  $j = k + 1$  to  $m$  do
4:      $l_{jk} = u_{jk} / u_{kk}$ 
5:      $u(j, k : m) = u(j, k : m) - l_{jk} u(k, k : m)$ 
6:   end for
7: end for

```

- Custo dominado pelo laço implícito na linha 5:

$$\sum_{k=1}^{m-1} \sum_{j=k+1}^m 2(m-k+1) \sim$$

$$\sum_{k=1}^m \sum_{j=k+1}^m 2(m-k) =$$

$$\sum_{k=1}^m 2(m-k)^2 \sim \frac{2}{3} m^3$$

- As três matrizes A, L, U não são todas necessárias no algoritmo: para minimizar a alocação de memória, ambas L e U podem ser escritas na mesma matriz A (Exercício 3, slide 11)

Instabilidade da Eliminação de Gauss sem Pivotamento

- **Exemplo 0:** $A = \begin{bmatrix} 0 & 1 \\ 1 & 1 \end{bmatrix}$ tem posto máximo e é bem condicionada. Entretanto, o método falha no primeiro passo
- **Exemplo 1:** Com uma pequena perturbação $\varepsilon > 0$, $A = \begin{bmatrix} \varepsilon & 1 \\ 1 & 1 \end{bmatrix}$, o método agora não falha:

$$L = \begin{bmatrix} 1 & 0 \\ 1/\varepsilon & 1 \end{bmatrix} \quad U = \begin{bmatrix} \varepsilon & 1 \\ 0 & 1 - 1/\varepsilon \end{bmatrix}$$

- **Exemplo 2:** Suponha agora que $\varepsilon > 0$ seja suficientemente pequeno tal que $\text{float}(1 - 1/\varepsilon) = -1/\varepsilon$. Temos a aproximação

$$\tilde{U} = \begin{bmatrix} \varepsilon & 1 \\ 0 & -1/\varepsilon \end{bmatrix}$$

que é uma matriz bastante próxima de U relativa a $\|U\|$.

Instabilidade da Eliminação de Gauss sem Pivotamento

- **Exemplo 2** (continuação): Entretanto,

$$\tilde{L}\tilde{U} = \begin{bmatrix} 1 & 0 \\ 1/\varepsilon & 1/\varepsilon \end{bmatrix} \begin{bmatrix} \varepsilon & 1 \\ 0 & 1/\varepsilon \end{bmatrix} = \begin{bmatrix} \varepsilon & 1 \\ 1 & 0 \end{bmatrix} =: \tilde{A}$$

que tem erro $1 \gg \varepsilon$:

$$A - \tilde{A} = \begin{bmatrix} 0 & 0 \\ 0 & 1 \end{bmatrix}$$

- Para matrizes maiores a situação tende a piorar mais ainda ...
- **Uma Solução:** permutar linhas (e possivelmente colunas) tal que os respectivos multiplicadores sejam relativamente pequenos, isto é:

► ELIMINAÇÃO DE GAUSS COM PIVOTAMENTO

Exercícios

- **Exercício 2:** Seja $A \in \mathbb{C}^{m \times m}$ matriz não-singular ($\det A \neq 0$). Mostre que A tem uma fatoração LU se e somente se para cada k com $1 \leq k \leq m$, o bloco $k \times k$ superior esquerdo $A(1:k, 1:k)$ é não-singular (Sugestão: as operações entre linhas da eliminação de Gauss não altera $\det A(1:k, 1:k)$). Prove que esta fatoração LU é única.
- **Exercício 3[Desafio]:** Como na maioria dos algoritmos do curso, eliminação de Gauss envolve laços triplos encaixados. No algoritmo da eliminação de Gauss, existem dois laços explícitos, e um terceiro laço implícito nos vetores $u(j, k:m)$ e $u(k, k:m)$. Reescreva este algoritmo com somente um laço explícito indexado por k . Dentro deste laço, U será atualizada em cada passo por uma alguma matriz produto-exterior de posto 1. Esta forma de eliminação de Gauss via "produto-exterior" pode ser uma ótimo ponto de início de modificação do algoritmo de eliminação de Gauss apresentado se o objetivo é otimizar a performance (velocidade e alocação de memória) computacional.