Sistemas Lineares - Eliminação de Gauss

Andréa Maria Pedrosa Valli

Laboratório de Computação de Alto Desempenho (LCAD)
Departamento de Informática
Universidade Federal do Espírito Santo - UFES, Vitória, ES, Brasil

Eliminação de Gauss

Sistema linear $n \times n$:

$$\begin{array}{rcl}
a_{11}x_1 + a_{12}x_2 + a_{13}x_3 + \cdots + a_{1n}x_n & = & b_1 \\
a_{21}x_1 + a_{22}x_2 + a_{23}x_3 + \cdots + a_{2n}x_n & = & b_2 \\
\cdot \quad \cdot \quad \cdot & & \cdot \\
\cdot \quad \cdot \quad \cdot & & \cdot \\
a_{n1}x_1 + a_{n2}x_2 + a_{n3}x_3 + \cdots + a_{nn}x_n & = & b_n
\end{array}$$

a_{ij} = coeficientes, b_j = constantes, x_j = variáveis $(i, j = 1, \cdots, n)$
Na forma matricial $Ax = b$

$$\begin{bmatrix} a_{11} & a_{12} & a_{13} & \cdots & a_{1n} \\ a_{21} & a_{22} & a_{23} & \cdots & a_{2n} \\ \vdots & \vdots & \vdots & & \vdots \\ a_{n1} & a_{n2} & a_{n3} & \cdots & a_{nn} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix} = \begin{bmatrix} b_1 \\ b_2 \\ \vdots \\ b_n \end{bmatrix}$$

Sistema triangular superior $n \times n$:

$$\begin{bmatrix} a_{11} & a_{12} & a_{13} & \cdots & a_{1n} \\ 0 & a_{22} & a_{23} & \cdots & a_{2n} \\ 0 & 0 & a_{33} & \cdots & a_{3n} \\ \vdots & \vdots & \vdots & & \vdots \\ 0 & 0 & 0 & \cdots & a_{nn} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ \vdots \\ x_n \end{bmatrix} = \begin{bmatrix} b_1 \\ b_2 \\ b_3 \\ \vdots \\ b_n \end{bmatrix}$$

Assuma que o sistema tem solução única: $a_{ii} \neq 0, \ i = 1, \cdots, n$.

Solução:

$$\begin{aligned} a_{nn}\, x_n &= b_n \Rightarrow x_n = \frac{b_n}{a_{nn}} \\ a_{n-1,n-1}x_{n-1} + a_{n-1,n}x_n &= b_{n-1} \Rightarrow x_{n-1} = \frac{b_{n-1} - a_{n-1,n}x_n}{a_{n-1,n-1}} \\ \text{linha i} \ \Rightarrow \ x_i &= \frac{b_i - \sum_{j=i+1}^{n} a_{ij}x_j}{a_{ii}} \end{aligned}$$

Algoritmo para a substituição regressiva: $x_i = \dfrac{b_i - \sum_{j=i+1}^{n} a_{ij} x_j}{a_{ii}}$

Data: A,b,n

Result: x

```
for i=n,1,-1 do
    soma = b[i];
    for j=i+1,n,1 do
        soma = soma - a[i][j] * x[j];
    end
    x[i] = soma/a[i][i];
end
```

Esforço computacional (№de operações (+,-,x,/) ou flops):

divisão: n

subtração e multiplicação: $2\sum_{j=1}^{n-1} j = 2n(n-1)/2$

total $= n^2$

Idéia do método:

$$Ax = b \qquad \Longrightarrow \qquad \tilde{A}x = \tilde{b}$$

operações de linhas elementares

onde $\tilde{A}$ é uma matriz triangular superior.

Operações de linhas elementares:

- trocar a ordem de duas equações;
- multiplicar uma equação por uma constante não nula;
- somar uma equação à outra.

Observação: A eliminação deve ser feita de forma sistemática, ou seja, usando uma sequência de operações elementares de modo a transformar um sistema linear em um outro equivalente, onde a matriz é triangular superior.

Eliminação de Gauss ingênua [2]:

FIGURA 9.3
As duas fases da eliminação de Gauss: eliminação progressiva e substituição regressiva. As linhas indicam o número de vezes que os coeficientes e as constantes foram modificados.

Eliminação progressiva

$$\left[\begin{array}{ccc:c} a_{11} & a_{12} & a_{13} & b_1 \\ a_{21} & a_{22} & a_{23} & b_2 \\ a_{31} & a_{32} & a_{33} & b_3 \end{array}\right]$$

$$\Downarrow$$

$$\left[\begin{array}{ccc:c} a_{11} & a_{12} & a_{13} & b_1 \\ & a'_{22} & a'_{23} & b'_2 \\ & & a''_{33} & b''_3 \end{array}\right]$$

$$\Downarrow$$

Substituição regressiva

$$x_3 = b''_3/a''_{33}$$
$$x_2 = (b'_2 - a'_{23}x_3)/a'_{22}$$
$$x_1 = (b_1 - a_{12}x_2 - a_{13}x_3)/a_{11}$$

Exemplo: sistema 4×4

$$\left[\begin{array}{cccc|c} -3 & 8 & -2 & 3 & 6 \\ 7 & -1 & 2 & 3 & 11 \\ -2 & 3 & 1 & 6 & 8 \\ 1 & -2 & 6 & 2 & 7 \end{array}\right] \quad \text{solução exata: } \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \end{bmatrix}$$

Primeiro Passo: Eliminar os coeficientes da primeira coluna abaixo da diagonal: pivô: $a_{11} = -3$

multiplicadores: $m_{21} = -7/3$, $m_{31} = 2/3$, $m_{41} = -1/3$

$\Rightarrow L_2 \leftarrow L_2 - (-7/3)L_1$, $L_3 \leftarrow L_3 - (2/3)L_1$, $L_4 \leftarrow L_4 - (-1/3)L_1$

$$\left[\begin{array}{cccc|c} -3 & 8 & -2 & 3 & 6 \\ 7 & 17.667 & -2.667 & 10 & 25.000 \\ -2 & -2.333 & 2.333 & 4 & 4.000 \\ 1 & 0.667 & 5.333 & 3 & 9.000 \end{array}\right]$$

$$\left[\begin{array}{cccc|c} -3 & 8 & -2 & 3 & 6 \\ 7 & 17.667 & -2.667 & 10 & 25.000 \\ -2 & -2.333 & 2.333 & 4 & 4.000 \\ 1 & 0.667 & 5.333 & 3 & 9.000 \end{array}\right]$$

Segundo Passo: Eliminar os coeficientes da segunda coluna abaixo da diagonal

pivô: $a_{22} = 17.667$

multiplicadores: $m_{32} = -2.333/17.667$, $m_{42} = 0.667/17.667$

$\Rightarrow L_3 \leftarrow L_3 - (-2.333/17.667)L_2$, $L_4 \leftarrow L_4 - (0.667/17.667)L_2$

$$\left[\begin{array}{cccc|c} -3 & 8 & -2 & 3 & 6 \\ 7 & 17.667 & -2.667 & 10 & 25.000 \\ -2 & -2.333 & 1.981 & 5.321 & 7.301 \\ 1 & 0.667 & 5.434 & 2.623 & 8.056 \end{array}\right]$$

$$\left[\begin{array}{cccc|c} -3 & 8 & -2 & 3 & 6 \\ 7 & 17.667 & -2.667 & 10 & 25.000 \\ -2 & -2.333 & 1.981 & 5.321 & 7.301 \\ 1 & 0.667 & 5.434 & 2.623 & 8.056 \end{array}\right]$$

Terceiro Passo: Eliminar os coeficientes da terceira coluna abaixo da diagonal

pivô: $a_{33} = 1.981$

multiplicadores: $m_{43} = 5.434/1.981$

Operações: $L_4 \leftarrow L_4 - (5.434/1.981)L_3$

$$\left[\begin{array}{cccc|c} -3 & 8 & -2 & 3 & 6 \\ 7 & 17.667 & -2.667 & 10 & 25.000 \\ -2 & -2.333 & 1.981 & 5.321 & 7.301 \\ 1 & 0.667 & 5.434 & -11.971 & -11.971 \end{array}\right]$$

Substituição Regressiva:

$$\begin{bmatrix} -3 & 8 & -2 & 3 \\ 7 & 17.667 & -2.667 & 10 \\ -2 & -2.333 & 1.981 & 5.321 \\ 1 & 0.667 & 5.434 & -11.971 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} = \begin{bmatrix} 6 \\ 25.000 \\ 7.301 \\ -11.971 \end{bmatrix}$$

$$\begin{aligned} -11.971\, x_4 = -11.971 \Rightarrow x_4 &= 1.000 \\ 1.981 x_3 + 5.321 x_4 = 7.301 \Rightarrow x_3 &= 0.999 \\ 17.667 x_2 - 2.667 x_3 + 10 x_4 = 25.000 \Rightarrow x_2 &= 1.000 \\ -3x_1 + 8x_2 - 2x_3 + 3x_4 = 6 \Rightarrow x_1 &= 1.001 \end{aligned}$$

Cálculo do Resíduo: $R = b - Ax$

$$R = \begin{bmatrix} 6 \\ 11 \\ 8 \\ 7 \end{bmatrix} - \begin{bmatrix} -3 & 8 & -2 & 3 \\ 7 & -1 & 2 & 3 \\ -2 & 3 & 1 & 6 \\ 1 & -2 & 6 & 2 \end{bmatrix} \begin{bmatrix} 1.000 \\ 0.999 \\ 1.000 \\ 1.001 \end{bmatrix} = \begin{bmatrix} 0.001 \\ -0.005 \\ 0.003 \\ 0.005 \end{bmatrix}$$

Observação: A solução é exata a menos dos erros de ponto flutuante. Sendo assim, o resíduo tem que ser bem pequeno, em torno do número de casas decimais utilizadas para os cálculos.

Algoritmo para a Eliminação Progressiva:

Passo k: Eliminar os coeficientes da k-ésima coluna abaixo da diagonal $(1 \leq k \leq n-1)$

Operação sobre a Linha i:

$$L_i \leftarrow L_i - m_{ik} L_k \text{ onde } m_{ik} = \frac{a_{ik}}{a_{kk}}, \quad k+1 \leq i \leq n$$

$$\Rightarrow a_{ij} \leftarrow a_{ij} - \frac{a_{ik}}{a_{kk}} a_{kj}, \quad k+1 \leq j \leq n$$

$$\Rightarrow b_i \leftarrow b_i - \frac{a_{ik}}{a_{kk}} b_k$$

```
Data: A,b,n
Result: x
for k=1,n-1 do
    for i=k+1,n do
        fator = a[i][k] / a[k][k];
        for j=k+1,n do
            a[i][j] = a[i][j] - fator * a[k][j];
        end
        b[i] = b[i] - fator * b[k]
    end
end
```

Esforço computacional:

adição e subtração: $n^3/3 + O(n)$

multiplicação e divisão: $n^3/3 + O(n^2)$

total $= 2n^3/3 + O(n^2)$

Obs: $O(m^n)$ significa “termos de ordem m^n e menores“.

Esporço Computacional:
Eliminação Progressiva: $2n^3/3 + O(n^2)$
Substituição Regressiva: n^2

n	Elim.	Subst.	Flops	$2n^3/3$	% Elim.
10	705	100	805	667	87.58%
100	671550	10000	681550	666667	98.53%
1000	6.67×10^8	1×10^6	6.68×10^8	6.67×10^8	99.85%

- O tempo de computação cresce bastante à medida que o sistema fica maior. A quantidade de flops cresce quase três ordens de grandeza para cada aumento na ordem de grandeza da dimensão;
- A maior parte do esforço vem da parte da eliminação. Esforços para melhorar o algoritmo devem se concentrar neste passo.

Gauss-Jordan algoritmo: $[A|b] \implies [I|x]$,

onde I é a matriz identidade e x é a solução do sistema. Neste método o esforço computacional é $O(n^3)$, ou seja, aproximadamente 50% mais operações que a eliminação de Gauss ingênua.

Esforço Computacional:

- Regra de Cramer: $O(n!)$
- Gauss-Jordan: $O(n^3)$
- Eliminação de Gauss ingênua: $O(2n^3/3)$

Obs: a regra de Cramer é inviável computacionalmente quando n é grande. Observe que a regra de Cramer envolve o cálculo de determinantes.

Problemas com a Eliminação de Gauss ingênua

1. Divisão por zero
   Exemplo: solução exata $(1, 1, 1)^T$

$$\begin{aligned} 2x_2 + 3x_3 &= 5 \\ x_1 - 3x_2 + x_3 &= -1 \\ 2x_1 + x_3 &= 3 \end{aligned}$$

2. Erros de arredondamento
   Exemplo: solução exata $(1/3, 2/3)^T$

$$\begin{aligned} 0.0003x_1 + 3x_2 &= 2.0001 \\ x_1 + x_2 &= 1 \end{aligned}$$

$$\begin{bmatrix} 0.0003 & 3 & | & 2.0001 \\ 1 & 1 & | & 1 \end{bmatrix} L_2 \leftarrow L_2 - \frac{1}{0.0003} L_1$$

$$\Rightarrow \begin{bmatrix} 0.0003 & 3 & | & 2.0001 \\ 0 & -9999 & | & -6666 \end{bmatrix}$$

$$\begin{aligned} x_2 &= 0.6666 \approx 2/3 \\ x_1 &= \frac{2.0001 - 3(x_2)}{0.0003} \end{aligned}$$

Tabela: Resultado muito sensível à precisão.

№de Dígitos	x_2	x_1	% Error relativo x_1
3	0.667	-3.33	1099
4	0.6667	0.0000	100
5	0.66667	0.30000	10
6	0.666667	0.330000	1
7	0.6666667	0.3330000	0.1

Técnicas para melhorar a solução:

- Usar mais dígitos significativos, ou seja, aumentar a precisão.
- Usar a estratégia de pivoteamento parcial.

Pivoteamento Parcial:

1. no início de cada etapa k, $1 \leq k \leq n-1$, escolher para pivô o elemento de maior módulo entre os coeficientes a_{ik}, $k \leq i \leq n$,
2. trocar as linhas k e i, se for necessário.

Exemplo: solução exata $(1/3, 2/3)^T$

$$\begin{aligned} 0.0003x_1 + 3x_2 &= 2.0001 \\ x_1 + x_2 &= 1 \end{aligned}$$

$$L_1 \longleftrightarrow L_2 \Rightarrow \left[\begin{array}{cc|c} 1 & 1 & 1 \\ 0.0003 & 3 & 2.0001 \end{array}\right] L_2 \leftarrow L_2 - \frac{0.0003}{1} L_1$$

$$\Rightarrow \left[\begin{array}{cc|c} 1 & 1 & 1 \\ 0 & 2.9997 & 1.9998 \end{array}\right]$$

$$\begin{aligned} x_2 &= 0.6666 \approx 2/3 \\ x_1 &= 1 - x_2 \end{aligned}$$

Tabela: Resultado usando pivoteamento parcial.

№de Dígitos	x_2	x_1	% Error relativo x_1
3	0.667	0.333	0.1
4	0.6667	0.3333	0.01
5	0.66667	0.33333	0.001
6	0.666667	0.333333	0.0001
7	0.6666667	0.3333333	0.00001

Pseudocódigo para implementar o pivoteamento parcial [2]:

```
p = k
maior = |a_{k,k}|
DOFOR ii = k+1, n
  dummy = |a_{ii,k}|
  IF (dummy > maior)
    maior = dummy
    p = ii
  END IF
END DO
IF (p ≠ k)
  DOFOR jj = k, n
    dummy = a_{p,jj}
    a_{p,jj} = a_{k,jj}
    a_{k,jj} = dummy
  END DO
  dummy = b_p
  b_p = b_k
  b_k = dummy
END IF
```

FIGURA 9.5

Exemplo: sistema 4×4

$$\left[\begin{array}{cccc|c} -3 & 8 & -2 & 3 & 6 \\ 7 & -1 & 2 & 3 & 11 \\ -2 & 3 & 1 & 6 & 8 \\ 1 & -2 & 6 & 2 & 7 \end{array}\right] \qquad \text{solução exata: } \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \end{bmatrix}$$

Primeiro Passo: Escolher o pivô (a_{11}), trocar linhas e eliminar os coeficientes da primeira coluna abaixo da diagonal

$$L_1 \longleftrightarrow L_2 \Rightarrow \left[\begin{array}{cccc|c} 7 & -1 & 2 & 3 & 11 \\ -3 & 8 & -2 & 3 & 6 \\ -2 & 3 & 1 & 6 & 8 \\ 1 & -2 & 6 & 2 & 7 \end{array}\right] \begin{array}{l} \Rightarrow L_2 \leftarrow L_2 - (-3/7)L_1 \\ \quad L_3 \leftarrow L_3 - (-2/7)L_1 \\ \quad L_4 \leftarrow L_4 - (1/7)L_1 \end{array}$$

$$\Rightarrow \left[\begin{array}{cccc|c} 7 & -1 & 2 & 3 & 11 \\ -3 & 7.571 & -1.143 & 4.286 & 10.714 \\ -2 & 2.714 & 1.571 & 6.857 & 11.143 \\ 1 & -1.857 & 5.714 & 1.571 & 5.429 \end{array}\right]$$

Segundo Passo: Escolher o pivô (a_{22}), trocar linhas e eliminar os coeficientes da segunda coluna abaixo da diagonal

$$\left[\begin{array}{cccc|c} 7 & -1 & 2 & 3 & 11 \\ -3 & 7.571 & -1.143 & 4.286 & 10.714 \\ -2 & 2.714 & 1.571 & 6.857 & 11.143 \\ 1 & -1.857 & 5.714 & 1.571 & 5.429 \end{array}\right]$$

$$\Rightarrow L_3 \leftarrow L_3 - (2.714/7.571)L_2$$
$$L_4 \leftarrow L_4 - (-1.857/7.571)L_2$$

$$\Rightarrow \left[\begin{array}{cccc|c} 7 & -1 & 2 & 3 & 11 \\ -3 & 7.571 & -1.143 & 4.286 & 10.714 \\ -2 & 2.714 & 1.981 & 5.321 & 7.302 \\ 1 & -1.875 & 5.434 & 2.623 & 8.057 \end{array}\right]$$

Terceiro Passo: Escolher o pivô (a_{33}), trocar linhas e eliminar os coeficientes da terceira coluna abaixo da diagonal

$$L_3 \longleftrightarrow L_4 \Rightarrow \left[\begin{array}{cccc|c} 7 & -1 & 2 & 3 & 11 \\ -3 & 7.571 & -1.143 & 4.286 & 10.714 \\ -2 & 2.714 & 5.434 & 2.623 & 8.057 \\ 1 & -1.857 & 1.981 & 5.321 & 7.302 \end{array}\right]$$

$$\Rightarrow L_4 \leftarrow L_4 - (1.981/5.434)L_3$$

$$\Rightarrow \left[\begin{array}{cccc|c} 7 & -1 & 2 & 3 & 11 \\ -3 & 7.571 & -1.143 & 4.286 & 10.714 \\ -2 & 2.714 & 5.434 & 2.623 & 8.057 \\ 1 & -1.857 & 1.981 & 4.365 & 4.364 \end{array}\right]$$

Substituição Regressiva:

$$\left[\begin{array}{cccc|c} 7 & -1 & 2 & 3 & 11 \\ -3 & 7.571 & -1.143 & 4.286 & 10.714 \\ -2 & 2.714 & 5.434 & 2.623 & 8.057 \\ 1 & -1.857 & 1.981 & 4.365 & 4.364 \end{array}\right] \Rightarrow$$

$$\begin{aligned} 4.365x_4 = 4.364 \Rightarrow x_4 &= 1.000 \\ 5.434x_3 + 2.623x_4 = 8.057 \Rightarrow x_3 &= 1.000 \\ 7.571x_2 - 1.143x_3 + 4.286x_4 = 10.714 \Rightarrow x_2 &= 1.000 \\ 7x_1 - x_2 + 2x_3 + 3x_4 = 11 \Rightarrow x_1 &= 1.000 \end{aligned}$$

Cálculo do Resíduo: $R = b - Ax$

$$R = \begin{bmatrix} 6 \\ 11 \\ 8 \\ 7 \end{bmatrix} - \begin{bmatrix} -3 & 8 & -2 & 3 \\ 7 & -1 & 2 & 3 \\ -2 & 3 & 1 & 6 \\ 1 & -2 & 6 & 2 \end{bmatrix} \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

Observação: Na eliminação de Gauss todos os multiplicadores são em módulo menores ou iguais a 1.

Pseudocódigo para a implementação da Eliminação de Gauss [2]:

```
SUB Gauss (a, b, n, x, tol, er)
  DIMENSION s(n)
  er = 0
  DOFOR i = 1, n
    s_i = ABS(a_i,1)
    DOFOR j = 2, n
      IF ABS(a_i,j)>s_i THEN s_i = ABS(a_i,j)
    END DO
  END DO
  CALL Eliminate(a, s, n, b, tol, er)
  IF er ≠ −1 THEN
     CALL Substitute(a, n, b, x)
  END IF
END Gauss

SUB Eliminate (a, s, n, b, tol, er)
  DOFOR k = 1, n − 1
    CALL Pivot (a, b, s, n, k)
    IF ABS (a_k,k/s_k) < tol THEN
      er = −1
      EXIT DO
    END IF
    DOFOR i = k + 1, n
      fator = a_i,k/a_k,k
      DOFOR j = k + 1, n
        a_i,j = a_i,j − fator*a_k,j
      END DO
      b_i = b_i − fator * b_k
    END DO
  END DO
  IF ABS(a_k,k/s_k) < tol THEN er = −1
END Eliminate

SUB Pivot (a, b, s, n, k)
  p = k
  maior = ABS(a_k,k/s_k)
  DOFOR ii = k + 1, n
    dummy = ABS(a_ii,k/s_ii)
    IF dummy > big THEN
       maior = dummy
       p = ii
    END IF
  END DO
  IF p ≠ k THEN
     DOFOR jj = k, n
       dummy = a_p,jj
       a_p,jj = a_k,jj
       a_k,jj = dummy
     END DO
     dummy = b_p
     b_p = b_k
     b_k = dummy
     dummy = s_p
     s_p = s_k
     s_k = dummy
  END IF
END pivot

SUB Substitute (a, n, b, x)
  x_n = b_n/a_n,n
  DOFOR i = n − 1, 1, −1
    soma = 0
    DOFOR j = i + 1, n
      soma = soma + a_i,j * x_j
    END DO
    x_i = (b_i − soma) / a_i,i
  END DO
END Substitute
```

FIGURA 9.6

Bibliografia Básica

[1] Algoritmos Numéricos, Frederico F. Campos, Filho - 2ª Ed., Rio de Janeiro, LTC, 2007.

[2] Métodos Numéricos para Engenharia, Steven C. Chapa e Raymond P. Canale, Ed. McGraw-Hill, 5ª Ed., 2008.

[3] Cálculo Numérico - Aspectos Teóricos e Computacionais, Márcia A. G. Ruggiero e Vera Lúcia da Rocha Lopes, Ed. Pearson Education, 2ª Ed., 1996.