MAP3122: Métodos numéricos e aplicações

Quadrimestral 2022

Antoine Laurain

Resolução de sistemas lineares

Eliminação Gaussiana

Neste capítulo veremos técnicas diretas para resolver o sistema linear

$$(SL): \begin{cases} E_1: & a_{11}x_1 + a_{12}x_2 + \cdots + a_{1n}x_n &= b_1 \\ E_2: & a_{21}x_1 + a_{22}x_2 + \cdots + a_{2n}x_n &= b_2 \\ \vdots & \vdots & \vdots \\ E_n: & a_{n1}x_1 + a_{n2}x_2 + \cdots + a_{nn}x_n &= b_n \end{cases}$$

(SL) = sistema linear. Para resolver (SL) usamos três operações

1. multiplicação de uma linha por $\lambda \neq 0$, isto é $(\lambda E_i) \to (E_i)$
2. multiplicar e adicionar linhas: $(E_i + \lambda E_j) \to (E_i)$
3. transpostar E_i e E_j: $(E_i) \leftrightarrow (E_j)$

Por meio destas operações, transformamos (SL) em um sistema linear mais fácil de resolver.

Eliminação Gaussiana

Exemplo: Repetimos estas operações até chegar na forma triangular do sistema.

Sistema inicial:

$$\begin{array}{rlrrrrrr}
E_1: & x_1 & +x_2 & & +3x_4 & = & 4 \\
E_2: & 2x_1 & +x_2 & -x_3 & +x_4 & = & 1 \\
E_3: & 3x_1 & -x_2 & -x_3 & +2x_4 & = & -3 \\
E_4: & -x_1 & +2x_2 & +3x_3 & -x_4 & = & 4
\end{array}$$

Primeira etapa:

$$\begin{array}{rlrrrrrr}
 & E_1: & x_1 & +x_2 & & +3x_4 & = & 4 \\
(E_2 - 2E_1) \to & E_2: & & -x_2 & -x_3 & -5x_4 & = & -7 \\
(E_3 - 3E_1) \to & E_3: & & -4x_2 & -x_3 & -7x_4 & = & -15 \\
(E_4 + E_1) \to & E_4: & & 3x_2 & +3x_3 & +2x_4 & = & 8
\end{array}$$

Segunda e última etapa:

$$\begin{array}{rlrrrrrr}
 & E_1: & x_1 & +x_2 & & +3x_4 & = & 4 \\
 & E_2: & & -x_2 & -x_3 & -5x_4 & = & -7 \\
(E_3 - 4E_2) \to & E_3: & & & 3x_3 & 13x_4 & = & 13 \\
(E_4 + 3E_2) \to & E_4: & & & & -13x_4 & = & -13
\end{array}$$

Obtivemos a *forma triangular* do sistema, que pode ser resolvida por um processo de substituição retroativa.

Eliminação Gaussiana

A substituição retroativa fornece a solução:

$$\begin{array}{lllll} E_4 \Rightarrow & x_4 & = & 1 \\ E_3 \Rightarrow & x_3 & = & \frac{1}{3}(13 - 13x_4) = 0 \\ E_2 \Rightarrow & x_2 & = & -(-7 + 5x_4 + x_3) = 2 \\ E_1 \Rightarrow & x_1 & = & 4 - 3x_4 - x_2 = -1 \end{array}$$

Esta técnica chama-se eliminação Gaussiana.

Não precisamos escrever as variáveis $\{x_i\}_{i=1}^n$, podemos fazer os cálculos usando notação matricial.

Definição: Uma matriz $n \times m$ é um arranjo retangular de elementos com n linhas e m colunas. Usamos as notações $A = (a_{ij}) \in \mathbb{R}^{n \times m}$ e também

$$A = \begin{pmatrix} a_{11} & \dots & a_{1m} \\ \vdots & \ddots & \vdots \\ a_{n1} & \dots & a_{nm} \end{pmatrix}.$$

A notação $A \in \mathbb{R}^{n \times m}$ significa que as entradas a_{ij} de A são reais, e que a matriz tem n linhas e m colunas. Se $n = 1$ ou $m = 1$, então A é um vetor.

Regras de cálculo para matrizes

Sejam duas matrizes $A \in \mathbb{R}^{n\times m}$ e $B \in \mathbb{R}^{m\times p}$. Denotando

$$A = (a_{ij})_{\substack{i=1,\dots,n\\ j=1,\dots,m}} \qquad B = (b_{ij})_{\substack{i=1,\dots,m\\ j=1,\dots,p}}$$

o produto AB é definido por:

$$AB = \left(\sum_{j=1}^{m} a_{ij}b_{jk}\right)_{\substack{i=1,\dots,n\\ k=1,\dots,p}}$$

Se $X \in \mathbb{R}^{m\times 1}$ for uma matriz com apenas uma coluna, dizemos que X é um vetor e temos que

$$AX = \left(\sum_{j=1}^{m} a_{ij}x_j\right)_{i=1,\dots,n}$$ é um vetor também.

Então o sistema linear (SL) pode ser escrito na forma matricial como:

$$Ax = b \Leftrightarrow (SL), \qquad \text{com } b = (b_i)_{i=1,\dots,m}$$

Também podemos representar (SL) usando a matriz expandida:

$$\left[\begin{array}{cccc|c} a_{11} & a_{12} & \dots & a_{1n} & b_1 \\ a_{21} & a_{22} & \dots & a_{2n} & b_2 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ a_{n1} & a_{n2} & \dots & a_{nn} & b_n \end{array}\right]$$

Regras de cálculo para matrizes

Exemplo: Voltamos ao exemplo anterior:

Sistema inicial:

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

Primeira etapa:

$$\left[\begin{array}{cccc|c} 1 & 1 & 0 & 3 & 4 \\ 0 & -1 & -1 & -5 & -7 \\ 0 & -4 & -1 & -7 & -15 \\ 0 & 3 & 3 & 2 & 8 \end{array}\right]$$

Segunda e última etapa (forma triangular do sistema):

$$\left[\begin{array}{cccc|c} 1 & 1 & 0 & 3 & 4 \\ 0 & -1 & -1 & -5 & -7 \\ 0 & 0 & 3 & 13 & 13 \\ 0 & 0 & 0 & -13 & -13 \end{array}\right]$$

Eliminação Gaussiana (caso geral)

$$[A|b] = \left[\begin{array}{cccc|c} a_{11} & a_{12} & \dots & a_{1n} & b_1 \\ a_{21} & a_{22} & \dots & a_{2n} & b_2 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ a_{n1} & a_{n2} & \dots & a_{nn} & b_n \end{array}\right]$$

Primeira etapa: $E_i - \frac{a_{i1}}{a_{11}} E_1 \to E_i$ para $i = 2, 3, \dots, n$ (se $a_{11} \neq 0$)

Resultado:

$$\left[\begin{array}{cccc|c} a_{11}^{(1)} & a_{12}^{(1)} & \dots & a_{1n}^{(1)} & b_1^{(1)} \\ 0 & a_{22}^{(1)} & \dots & a_{2n}^{(1)} & b_2^{(1)} \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & a_{n2}^{(1)} & \dots & a_{nn}^{(1)} & b_n^{(1)} \end{array}\right]$$

k-ésima etapa: $E_i - \frac{a_{ik}^{(k-1)}}{a_{kk}^{(k-1)}} E_k \to E_i$ para $i = k+1, k+2, \dots, n$ (se $a_{kk}^{(k-1)} \neq 0$)

Resultado na $(n-1)$-ésima etapa: obtemos uma matriz triangular superior

$$\left[\begin{array}{cccc|c} a_{11}^{(n-1)} & a_{12}^{(n-1)} & \dots & a_{1n}^{(n-1)} & b_1^{(n-1)} \\ 0 & a_{22}^{(n-1)} & \dots & a_{2n}^{(n-1)} & b_2^{(n-1)} \\ \vdots & \ddots & \ddots & \vdots & \vdots \\ 0 & \dots & 0 & a_{nn}^{(n-1)} & b_n^{(n-1)} \end{array}\right]$$

Eliminação Gaussiana (caso geral)

Depois fazemos uma substituição retroativa:

$$x_n = \frac{b_n^{(n-1)}}{a_{nn}^{(n-1)}}, \qquad x_i = \frac{b_i^{(n-1)} - \sum_{j=i+1}^{n} a_{ij}^{(n-1)} x_j}{a_{ii}^{(n-1)}} \qquad \text{para } i = n-1, n-2, \ldots, 1$$

Observação: Este procedimento funciona apenas se $a_{kk}^{(k-1)} \neq 0$ para todos k. Se acontecer $a_{kk}^{(k-1)} = 0$ para algum k, podemos transpostar duas linhas para contornar o problema.

Exemplo: Eliminação de Gauss com pivotamento

Sistema inicial:

$$[A|b] = \left[\begin{array}{cccc|c} 1 & -1 & 2 & -1 & -8 \\ 2 & -2 & 3 & -3 & -20 \\ 1 & 1 & 1 & 0 & -2 \\ 1 & -1 & 4 & 3 & 4 \end{array}\right]$$

Primeira etapa:

$$\begin{array}{rcl} E_2 - E_1 & \rightarrow & E_2 \\ E_3 - E_1 & \rightarrow & E_3 \\ E_4 - E_1 & \rightarrow & E_4 \end{array} \left[\begin{array}{cccc|c} 1 & -1 & 2 & -1 & -8 \\ 0 & 0 & -1 & -1 & -4 \\ 0 & 2 & -1 & 1 & 6 \\ 0 & 0 & 2 & 4 & 12 \end{array}\right]$$

Aqui $a_{22}^{(1)} = 0$ é o elemente pivô. Como $a_{22}^{(1)} = 0$, não podemos continuar a eliminação de Gauss. Podemos usar a transposição $(E_2) \leftrightarrow (E_3)$.

Eliminação Gaussiana (caso geral)

Segunda etapa:

$$\begin{array}{ccc} E_3 & \rightarrow & E_2 \\ E_2 & \rightarrow & E_3 \end{array} \left[\begin{array}{cccc|c} 1 & -1 & 2 & -1 & -8 \\ 0 & 2 & -1 & 1 & 6 \\ 0 & 0 & -1 & -1 & -4 \\ 0 & 0 & 2 & 4 & 12 \end{array}\right]$$

Terceira etapa:

$$E_4 + 2E_3 \rightarrow E_4 \left[\begin{array}{cccc|c} 1 & -1 & 2 & -1 & -8 \\ 0 & 2 & -1 & 1 & 6 \\ 0 & 0 & -1 & -1 & -4 \\ 0 & 0 & 0 & 2 & 4 \end{array}\right]$$

Solução usando substituição retroativa: $x_4 = 2$, $x_3 = 2$, $x_2 = 3$, $x_1 = -7$

Contar o número de operações básicas na eliminação de Gauss

- Na k-ésima etapa da eliminação de Gauss (têm $n-1$ etapas), fazemos (o exponente $(k-1)$ representa o número da etapa)

$$E_i - m_{ik}E_k \to E_i \quad \text{com } m_{ik} = \frac{a_{ik}^{(k-1)}}{a_{kk}^{(k-1)}} \quad \text{para } i = k+1, \ldots, n$$

- Cálculo de m_{ik}: são $(n-k)$ divisões (pois $i = k+1, \ldots, n$).
- Cálculo de $m_{ik}E_k$: são $\underbrace{(n-k)}_{n^o \text{ de linhas}} \times \underbrace{(n-k+1)}_{\substack{n^o \text{ de elementos } \neq 0 \\ \text{na linha } k}}$ multiplicações.
- Cálculo de $E_i - m_{ik}E_k$: são $\underbrace{(n-k)}_{n^o \text{ de linhas}} \times \underbrace{(n-k+1)}_{\substack{n^o \text{ de elementos } \neq 0 \\ \text{na linha } k}}$ subtrações.
- Total de operações básicas:

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

pois há no máximo $(n-1)$ etapas na eliminação de Gauss.

Contar o número de operações básicas na eliminação de Gauss

- Definindo $\ell = n - k$ obtemos

$$\begin{aligned}\sum_{k=1}^{n-1} 2(n-k)(n-k+1) + (n-k) &= \sum_{\ell=1}^{n-1} 2\ell(\ell+1) + \ell = 2\sum_{\ell=1}^{n-1} \ell^2 + 3\sum_{\ell=1}^{n-1} \ell \\ &= 2\frac{(n-1)n(2n-1)}{6} + \frac{3n(n-1)}{2} \\ &= \frac{2n^3}{3} + \frac{n^2}{2} - \frac{7n}{6}\end{aligned}$$

- Então a quantidade de operações básicas para a eliminação de Gauss é $O(\frac{2}{3}n^3)$ (isto é, da ordem $\frac{2}{3}n^3$).
- Número de operações para a substituição retroativa (para resolver (SL)):
  - $\frac{n^2-n}{2}$ multiplicações / divisões
  - $\frac{n^2-n}{2}$ adições / subtrações
- Número total de operações:

$$\frac{2n^3}{3} + \frac{n^2}{2} - \frac{7n}{6} + 2\left(\frac{n^2-n}{2}\right) = \frac{2n^3}{3} + \frac{3n^2}{2} - \frac{13n}{6}$$

- Então a quantidade de operações básicas para resolver o (SL) usando eliminação de Gauss é também $O(\frac{2}{3}n^3)$.

Fatoração LU

- A eliminação de Gauss fornece a fatoração $A = LU$

$$L = \text{ matriz triangular inferior } (L = \text{“lower”})$$
$$U = \text{ matriz triangular superior } (U = \text{“upper”})$$

- Podemos usar a fatoração LU para resolver o (SL) $Ax = b$. Usando $Ux = y$ temos

$$Ax = b \Leftrightarrow LUx = b \Leftrightarrow Ly = b.$$

- Primeiro, resolver $Ly = b$ usando substituição para frente (complexidade $O(n^2)$).
- Segundo, resolver $Ux = y$ usando substituição retroativa (complexidade $O(n^2)$).
- Se a fatoração $A = LU$ for dada, calcular x é $O(n^2)$; comparar com a complexidade $O(n^3)$ da eliminação de Gauss.
- A fatoração $A = LU$ tem complexidade $O(\frac{n^3}{3})$.
- Então para resolver um sistema $Ax = b$, a fatoração LU tem complexidade semelhante à eliminação de Gauss.
- Se houver vários sistemas $Ax_k = b_k$, $k = 1, \ldots, K$, com a mesma matriz A, podemos reusar a fatoração LU e o cálculo dos x_k é apenas $O(n^2)$.

Relação entre eliminação de Gauss e fatoração LU

Para simplificar, vamos supor que não precisamos de intercâmbio de linhas na eliminação de Gauss.

Primeira etapa: $E_i - m_{i1}E_1 \to E_i, \forall i = 2, 3, \ldots, n$, onde $m_{i1} = \frac{a_{i1}}{a_{11}}$. Esse é equivalente a multiplicar A a esquerda por M_1, a **matriz de primeira transformação de Gauss**:

$$M_1 = \begin{pmatrix} 1 & 0 & 0 & \cdots & 0 \\ -m_{21} & 1 & 0 & \cdots & 0 \\ -m_{31} & 0 & 1 & \ddots & \vdots \\ \vdots & \vdots & \ddots & \ddots & 0 \\ -m_{n1} & 0 & \cdots & 0 & 1 \end{pmatrix}$$

$\longrightarrow$ obtemos $M_1 AX = M_1 b$, definimos $A_1 = A$, $A_2 = M_1 A$

Relação entre eliminação de Gauss e fatoração LU

k-ésima etapa: $E_i - m_{ik}E_1 \to E_i, \forall i = k+1, k+2, \ldots, n$, com $m_{ik} = \frac{a_{ik}^{(k-1)}}{a_{kk}^{(k-1)}}$, onde $a_{ik}^{(k-1)}$ e $a_{kk}^{(k-1)}$ são coeficientes de A_k, e com

$$M_k = \begin{pmatrix} 1 & 0 & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & 0 \\ 0 & \ddots & \ddots & & & & & \cdots & 0 \\ \vdots & \ddots & \ddots & \ddots & & & & \cdots & 0 \\ \vdots & & \ddots & \ddots & \ddots & & & \cdots & 0 \\ \vdots & & & 0 & 1 & 0 & & \cdots & 0 \\ \vdots & & & \vdots & -m_{k+1,k} & 1 & 0 & \cdots & 0 \\ \vdots & & & \vdots & \vdots & 0 & 1 & \ddots & \vdots \\ \vdots & & & \vdots & \vdots & \vdots & \ddots & \ddots & 0 \\ 0 & 0 & \cdots & 0 & -m_{n,k} & 0 & \cdots & 0 & 1 \end{pmatrix}$$

Os coeficientes $-m_{k+1,k}, \ldots, -m_{n,k}$ ficam na k-ésima coluna.

Relação entre eliminação de Gauss e fatoração LU

$(n-1)$-ésima (e última) etapa: Obtemos

$$\underbrace{M_{n-1}M_{n-2}\ldots M_2M_1A}_{=A_n}X = M_{n-1}M_{n-2}\ldots M_2M_1b$$

Como fizemos a eliminação de Gauss, temos que A_n é triangular superior, então definimos $U = A_n = M_{n-1}M_{n-2}\ldots M_2M_1A$. Multiplicando por M_{n-1}^{-1}, depois M_{n-2}^{-1}, etc ... do lado esquerdo de U obtemos

$$\underbrace{M_1^{-1}M_2^{-1}\ldots M_{n-2}^{-1}M_{n-1}^{-1}}_{=L}U = A$$

É fácil verificar que $M_k^{-1} = 2I - M_k$ (onde I é a matriz identidade) e

$$M_1^{-1}M_2^{-1}\ldots M_{n-2}^{-1}M_{n-1}^{-1} = \begin{pmatrix} 1 & 0 & 0 & \cdots & 0 \\ m_{21} & 1 & 0 & \cdots & 0 \\ m_{31} & \ddots & 1 & \ddots & \vdots \\ \vdots & \vdots & \ddots & \ddots & 0 \\ m_{n1} & \cdots & \cdots & m_{n,n-1} & 1 \end{pmatrix}$$

(basta escrever $M_k^{-1} = I + E_k$ e $M_j^{-1} = I + E_j$ obtemos $M_k^{-1}M_j^{-1} = I + E_k + E_j + \underbrace{E_kE_j}_{=0}$)

Relação entre eliminação de Gauss e fatoração LU

Teorema: Se a eliminação de Gauss pode ser executada no sistema linear $Ax = b$ sem intercâmbio de linhas, então a matriz A pode ser fatorada no produto $A = LU$ de uma matriz triangular inferior unitária L e uma matriz triangular superior U, com

$$L = \begin{pmatrix} 1 & 0 & 0 & \cdots & 0 \\ m_{21} & 1 & 0 & \cdots & 0 \\ m_{31} & \ddots & 1 & \ddots & \vdots \\ \vdots & \vdots & \ddots & \ddots & 0 \\ m_{n1} & \cdots & \cdots & m_{n,n-1} & 1 \end{pmatrix} \qquad U = \begin{pmatrix} a_{11}^{(n-1)} & a_{12}^{(n-1)} & \cdots & a_{1n}^{(n-1)} \\ 0 & a_{22}^{(n-1)} & \cdots & a_{2n}^{(n-1)} \\ \vdots & \ddots & \ddots & \vdots \\ 0 & \cdots & 0 & a_{nn}^{(n-1)} \end{pmatrix}$$

Observação: L unitária significa que os coeficientes diagonais de L são todos iguais a 1.

Fatoração LU

Exemplo: Voltamos ao exemplo anterior:

$$A = \begin{pmatrix} 1 & 1 & 0 & 3 \\ 2 & 1 & -1 & 1 \\ 3 & -1 & -1 & 2 \\ -1 & 2 & 3 & -1 \end{pmatrix} = \underbrace{\begin{pmatrix} 1 & 0 & 0 & 0 \\ 2 & 1 & 0 & 0 \\ 3 & 4 & 1 & 0 \\ -1 & -3 & 0 & 1 \end{pmatrix}}_{=L} \underbrace{\begin{pmatrix} 1 & 1 & 0 & 3 \\ 0 & -1 & -1 & -5 \\ 0 & 0 & 3 & 13 \\ 0 & 0 & 0 & -13 \end{pmatrix}}_{=U}$$

- Os coeficientes de U já foram calculados no exemplo anterior usando eliminação de Gauss.
- Para calcular os coeficientes de L usamos $m_{ik} = \frac{a_{ik}^{(k-1)}}{a_{kk}^{(k-1)}}$, então precisamos escrever todas as etapas da eliminação de Gauss. Por exemplo

$$m_{21} = \frac{a_{21}^{(1)}}{a_{11}^{(1)}} = 2, \quad m_{32} = \frac{a_{32}^{(2)}}{a_{22}^{(2)}} = 2.$$

Fatoração LU

- Depois, para resolver $Ax = LUx = b = \begin{pmatrix} 8 & 7 & 14 & -7 \end{pmatrix}^\top$ fazemos

$$y = Ux \quad \text{e} \quad Ly = b \Rightarrow \begin{pmatrix} 1 & 0 & 0 & 0 \\ 2 & 1 & 0 & 0 \\ 3 & 4 & 1 & 0 \\ -1 & -3 & 0 & 1 \end{pmatrix} \begin{pmatrix} y_1 \\ y_2 \\ y_3 \\ y_4 \end{pmatrix} = \begin{pmatrix} 8 \\ 7 \\ 14 \\ -7 \end{pmatrix}$$

  e usando substituição, começando com y_1, obtemos a solução

$$\begin{aligned} y_1 &= 8 \\ y_2 &= 7 - 2y_1 = -9 \\ y_3 &= 14 - 3y_1 - 4y_3 = 26 \\ y_4 &= -7 + y_1 + 3y_2 = -26 \end{aligned}$$

- Depois resolvemos $Ux = y$ para x:

$$\begin{pmatrix} 1 & 1 & 0 & 3 \\ 0 & -1 & -1 & -5 \\ 0 & 0 & 3 & 13 \\ 0 & 0 & 0 & -13 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{pmatrix} = \begin{pmatrix} 8 \\ -9 \\ 26 \\ -26 \end{pmatrix}$$

  e usando substituição retroativa, começando com x_4, obtemos a solução de $Ax = b$:

$$x_4 = 2, \quad x_3 = 0, \quad x_2 = -1, \quad x_1 = 3.$$

Cálculo direto dos elementos de L e U

- As entradas de $A = LU$ são $a_{ij} = \sum_{k=1}^{n} \ell_{ik} u_{kj}$, $1 \leq i, j \leq n$.
- Como L e U são triangular inferior e superior, o alcance de k estende-se apenas até $\min\{i, j\}$. Isto produz

$$a_{ij} = \sum_{k=1}^{j} \ell_{ik} u_{kj}, \qquad \text{para } 1 \leq j \leq i \leq n$$

$$a_{ij} = \sum_{k=1}^{i} \ell_{ik} u_{kj}, \qquad \text{para } 1 \leq i \leq j \leq n$$

- Rearranjando estas equações e usando $\ell_{ii} = 1$ achamos

$$(L_{ij}) \qquad \ell_{ij} = \frac{1}{u_{jj}} \left(a_{ij} - \sum_{k=1}^{j-1} \ell_{ik} u_{kj} \right) \qquad \text{para } i = 2, \ldots, n, \quad j = 1, \ldots, i-1$$

$$(U_{ij}) \qquad u_{ij} = a_{ij} - \sum_{k=1}^{i-1} \ell_{ik} u_{kj} \qquad \text{para } i = 1, \ldots, n, \quad j = i, \ldots, n.$$

- Os primeiros termos são

$$(U_{1j}) \qquad u_{1j} = a_{1j}, \quad j = 1, 2, \ldots, n$$

$$(L_{i1}) \qquad \ell_{11} = 1, \quad \ell_{i1} = \frac{a_{i1}}{u_{11}}, \quad i = 2, \ldots, n$$

Relação entre eliminação de Gauss e fatoração LU

Algoritmo para fatoração LU

Inicialização:
calcule u_{1j} usando (U_{1j}) para $j = 1, \ldots, n$,
calcule ℓ_{i1} usando (L_{i1}) para $i = 1, \ldots, n$.

Para $i = 2, \ldots, n$:
    Para $j = 1, \ldots, i-1$:
        calcule ℓ_{ij} usando (L_{ij}) e $\ell_{ii} = 1$.
    Para $j = i, \ldots, n$:
        calcule u_{ij} usando (U_{ij}).

Observação: Se durante o processo ocorrer $u_{jj} = 0$, então não podemos calcular ℓ_{ij}. Neste caso precisamos modificar o algoritmo para fazer intercâmbio de linhas.

Fatoração de Cholesky

- $A \in \mathbb{R}^{n\times n}$ é definida positiva $\Leftrightarrow \langle Ax, x\rangle > 0$ para todos $x \neq 0$, $x \in \mathbb{R}^n$.
- Seja $A \in \mathbb{R}^{n\times n}$ uma **matriz simétrica e definida positiva**.
- A fatoração de Cholesky é do tipo $A = LL^\top$, onde L é uma matriz triangular inferior com $\ell_{ii} > 0$, onde $L = (\ell_{ij})_{i,j=1}^n$.
- Essa fatoração sempre existe e é única, quando A é definida positiva.
- A complexidade da fatoração de Cholesky é $O(\frac{1}{3}n^3)$, então o custo é a metade da fatoração LU que é da ordem $O(\frac{2}{3}n^3)$.
- Para calcular os coeficientes de L:

$$
\begin{aligned}
A = LL^\top &\Leftrightarrow a_{ij} = \sum_{k=1}^{n} \ell_{ik}\tilde{\ell}_{kj}, \qquad \text{onde } L^\top = (\tilde{\ell}_{ij})_{i,j=1}^n \\
&\Leftrightarrow a_{ij} = \sum_{k=1}^{n} \ell_{ik}\ell_{jk}, \\
&\Leftrightarrow a_{ij} = \sum_{k=1}^{\min\{i,j\}} \ell_{ik}\ell_{jk}, \qquad \text{pois } L \text{ é triangular inferior.}
\end{aligned}
$$

- Usando esta fórmula podemos calcular ℓ_{ii} (observe que a raiz está sempre bem definida pois A é positiva definida)

$$
a_{ii} = \sum_{k=1}^{i} (\ell_{ik})^2 = (\ell_{ii})^2 + \sum_{k=1}^{i-1} (\ell_{ik})^2 \quad \Rightarrow \ell_{ii} = \sqrt{a_{ii} - \sum_{k=1}^{i-1} (\ell_{ik})^2}
$$

Fatoração de Cholesky

- Suponhamos $j < i$, então (observe que $\ell_{jj} > 0$ quando A positiva definida)

$$a_{ij} = \sum_{k=1}^{j} \ell_{ik}\ell_{jk} = \ell_{ij}\ell_{jj} + \sum_{k=1}^{j-1} \ell_{ik}\ell_{jk} \quad \Rightarrow \ell_{ij} = \frac{1}{\ell_{jj}} \left(a_{ij} - \sum_{k=1}^{j-1} \ell_{ik}\ell_{jk} \right)$$

- **Algoritmo para fatoração de Cholesky-Banachiewicz**

  Inicialização: $\ell_{11} = \sqrt{a_{11}}$

  Para $i = 2, \ldots, n$:

  Para $j = 1, \ldots, i-1$:

  calcule $\ell_{ij} = \frac{1}{\ell_{jj}} \left(a_{ij} - \sum_{k=1}^{j-1} \ell_{ik}\ell_{jk} \right)$

  calcule $\ell_{ii} = \sqrt{a_{ii} - \sum_{k=1}^{i-1} (\ell_{ik})^2}$

- **Exemplo de cálculo dos primeiros termos no algorítmo:**

$$\ell_{11} = \sqrt{a_{11}} \rightarrow \ell_{21} = \frac{a_{21}}{\ell_{11}} \rightarrow \ell_{22} = \sqrt{a_{22} - (\ell_{21})^2}$$

$$\rightarrow \ell_{31} = \frac{a_{31}}{\ell_{11}} \rightarrow \ell_{32} = \frac{a_{32}}{\ell_{22}} - \ell_{31}\ell_{21} \rightarrow \ell_{33} = \sqrt{a_{33} - (\ell_{31})^2 - (\ell_{32})^2}$$

Observamos como o cálculo de ℓ_{ij} precisa dos valores de coeficientes calculados em passos anteriores, por isso a ordem de cálculo dos coeficientes ℓ_{ij} é importante.

Sistemas mal condicionados

- Alguns sistemas são muito sensíveis a pequenas alterações nos dados, isto é, uma pequena alteração nos dados pode gerar uma grande alteração na solução.

$$(SL): \begin{cases} x_1 + 0,98x_2 &= 4.95 \\ x_1 + x_2 &= 5.0 \end{cases}$$

tem solução exata $(2,5; 2,5)$, enquanto

$$(SL): \begin{cases} x_1 + 0,99x_2 &= 4.95 \\ x_1 + x_2 &= 5.0 \end{cases}$$

tem solução exata $(0; 5, 0)$. Uma alteração de 1% nos dados $(0,98 \rightarrow 0,99)$ gera uma alteração de 100% na solução $((2,5; 2,5) \rightarrow (0; 5, 0))$. Este tipo de sistema é chamado mal condicionado.

- Quando resolvemos $Ax = b$, os erros de arredondamento fazem com que a solução obtida $\tilde{x}$ possa ser encarada como a solução de um outro sistema $\tilde{A}\tilde{x} = \tilde{b}$.
- Se A for mal condicionada, a solução $\tilde{x}$ pode ser muito diferente da solução exata x.
- Por causa deste fenômeno, métodos diretos como a fatoração podem fornecer uma solução errada se A for mal condicionada.

Sistemas mal condicionados

Definição: O número de condicionamento de uma matriz não-singular A é

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

onde

$$\|A\| = \sup_{x \in \mathbb{R}^n, x \neq 0} \frac{\|Ax\|}{\|x\|}.$$

Se $\kappa(A) \gg 1$, a matriz é mal-condicionada. O número de condicionamento depende da norma escolhida.

Teorema: $A \in \mathbb{R}^{n \times n}$ não-singular, $b \in \mathbb{R}^n_*$, $Ax = b$ e $A(x + \delta x) = b + \delta b$ com $\delta x, \delta b \in \mathbb{R}^n$. Então $x \in \mathbb{R}^n_*$ e

$$\frac{\|\delta x\|}{\|x\|} \leq \kappa(A) \frac{\|\delta b\|}{\|b\|}.$$

Interpretação do teorema: Por causa do arredondamento, a solução de $Ax = b$ não é exata e pode ser escrita como $x + \delta x$, de modo que $A(x + \delta x) = b + \delta b$, com $\|\delta b\|$ pequena. Se $\kappa(A)$ for grande, então $\|\delta x\|$ pode ser grande mesmo se $\|\delta b\|$ for pequeno. Por isso queremos $\kappa(A)$ o menor possível.

Condensação pivotal

Os erros de arredondamento que ocorrem nas operações usando um computador podem comprometer seriamente a solução obtida.

Exemplo: Consideramos o (SL) seguinte:

$$[A|b] = \left[\begin{array}{ccc|c} 1 & 4 & 52 & 57 \\ 27 & 110 & -3 & 134 \\ 22 & 2 & 14 & 38 \end{array}\right]$$

A solução exata é $x = (1; 1; 1)$. Trabalhamos com três algarismos significativos. O método da eliminação de Gauss fornece a forma triangular seguinte do sistema (usando truncamento nos cálculos):

$$\left[\begin{array}{ccc|c} 1 & 4 & 52 & 57 \\ 0 & 2 & -14000 & -1410 \\ 0 & 0 & -61300 & -61800 \end{array}\right]$$

A solução deste sistema é $\tilde{x} = (1,01; 0,0; 4,5)$, que é bastante diferente da solução exata $x = (1; 1; 1)$.

Condensação pivotal

A condensação pivotal consiste em fazer intercâmbio de linhas, tal que o pivô no início de cada etapa $i = 1, \ldots, n-1$, seja o número de maior valor absoluto dentre os elementos da i-ésima coluna abaixo da diagonal (sem ser na diagonal).

Exemplo: Trabalhamos com três algarismos significativos. Consideramos o (SL) seguinte, cuja solução exata é $x = (1; 1; 1)$:

$$[A|b] = \left[\begin{array}{ccc|c} 1 & 4 & 52 & 57 \\ 27 & 110 & -3 & 134 \\ 22 & 2 & 14 & 38 \end{array}\right]$$

Permutamos a primeira e a segunda linha pois 27 tem o maior valor absoluto abaixo da diagonal na primeira coluna:

$$\left[\begin{array}{ccc|c} 27 & 110 & -3 & 134 \\ 1 & 4 & 52 & 57 \\ 22 & 2 & 14 & 38 \end{array}\right]$$

Condensação pivotal

Depois fazemos uma etapa da eliminação de Gauss:

$$\left[\begin{array}{ccc|c} 27 & 110 & -3 & 134 \\ 0 & -0,0741 & 52,1 & 52 \\ 0 & -87,7 & 16,5 & -71 \end{array}\right]$$

Fazemos mais uma condensação pivotal, permutando a segunda e terceira linha:

$$\left[\begin{array}{ccc|c} 27 & 110 & -3 & 134 \\ 0 & -87,7 & 16,5 & -71 \\ 0 & -0,0741 & 52,1 & 52 \end{array}\right]$$

Depois fazemos uma etapa da eliminação de Gauss:

$$\left[\begin{array}{ccc|c} 27 & 110 & -3 & 134 \\ 0 & -87,7 & 16,5 & -71 \\ 0 & 0 & 52,1 & 52,1 \end{array}\right]$$

A solução deste sistema é $\tilde{x} = (1,0; 0,998; 1,0)$, que agora é próximo da solução exata $x = (1; 1; 1)$.

Métodos iterativos

- A fatoração LU é um método direto que fornece a solução exata de um sistema linear $Ax = b$, mas na prática tem erros de arredondamento, e se o sistema linear for mal condicionado, o erro pode ser grande.
- Por causa destes erros de arredondamento, não precisamos de um método que calcule a solução exata, só precisamos de um método que minimize o erro.
- Métodos iterativos calculam uma sequência de aproximações $x^{(1)}, x^{(2)}, \ldots, x^{(k)} \to x^*$ solução de $Ax = b$. Em geral estes métodos não fornecem a solução exata x^* em um número finito de iterações, mas isto pode acontecer.

Vantagens dos métodos iterativos:

1. Métodos iterativos às vezes precisam de menos cálculos que métodos diretos (já vimos nos capítulos anteriores que a complexidade da fatoração LU é $O(n^3)$).
2. Métodos iterativos podem ajudar a diminuir o erro de métodos diretos.
3. Quando A for esparsa (isto significa que a maioria das entradas de A são zeros), a fatoração $A = LU$ é ineficiente pois L e U não são esparsas em geral, o que deixa os cálculos mais pesados que o necessário.

Método de Gauss-Seidel

Vamos começar com um método iterativo clássico: o método de Gauss-Seidel (GS). Seja $Ax = b$ um sistema linear de ordem n. Suponhamos $a_{ii} \neq 0$ para todos $i = 1, 2, \ldots, n$. Primeiro vamos rearranjar o sistema linear:

$$Ax = b \Leftrightarrow \begin{cases} x_1 &= \frac{1}{a_{11}} \left[b_1 - a_{12}x_2 - a_{13}x_3 - \cdots - a_{1n}x_n \right] \\ x_2 &= \frac{1}{a_{22}} \left[b_2 - a_{21}x_1 - a_{23}x_3 - \cdots - a_{2n}x_n \right] \\ \vdots & \vdots \\ x_n &= \frac{1}{a_{nn}} \left[b_n - a_{n,1}x_1 - a_{n,2}x_2 - \cdots - a_{n,n-1}x_{n-1} \right] \end{cases}$$

Método de Gauss-Seidel (GS): Escolhe um vetor inicial $x^{(0)} = (x_1^{(0)}, x_2^{(0)}, \cdots, x_n^{(0)})$. Supondo o valor do vetor $x^{(k)}$ conhecido ($x^{(k)}$ é o valor na k-ésima iteração), calculamos a próxima iteração com:

$$\begin{aligned} x_1^{(k+1)} &= \frac{1}{a_{11}} \left[b_1 - a_{12}x_2^{(k)} - a_{13}x_3^{(k)} - \cdots - a_{1n}x_n^{(k)} \right] \\ x_2^{(k+1)} &= \frac{1}{a_{22}} \left[b_2 - a_{21}x_1^{(k+1)} - a_{23}x_3^{(k)} - \cdots - a_{2n}x_n^{(k)} \right] \\ \vdots & \quad \vdots \\ x_{n-1}^{(k+1)} &= \frac{1}{a_{n-1,n-1}} \left[b_{n-1} - a_{n-1,1}x_1^{(k+1)} - \cdots - a_{n-1,n-2}x_{n-2}^{(k+1)} - a_{n-1,n}x_n^{(k)} \right] \\ x_n^{(k+1)} &= \frac{1}{a_{nn}} \left[b_n - a_{n,1}x_1^{(k+1)} - a_{n,2}x_2^{(k+1)} - \cdots - a_{n,n-1}x_{n-1}^{(k+1)} \right] \end{aligned}$$

Método de Gauss-Seidel

- No método de Gauss-Seidel, é importante não errar nas posições dos números de iteração $(k+1)$ ou (k): os termos $x_i^{(k)}$ aparecem "acima da diagonal" enquanto os termos $x_i^{(k+1)}$ aparecem "abaixo da diagonal".
- As entradas na iteração $(k+1)$ são calculadas nesta ordem:

$$x_1^{(k+1)} \to x_2^{(k+1)} \to x_1^{(k+1)} \cdots \to x_n^{(k+1)}$$

- **Por que o método GS converge para a solução de $Ax = b$?**

  $\longrightarrow$ a convergência de $x^{(k)}$ não é garantida em geral, mas se $x^{(k)}$ convergir para algum x^* (isto é $x^{(k)} \to x^*$), então temos também $x^{(k+1)} \to x^*$, e passando no limite na iteração de GS (isto é, substituir $x^{(k)}$ e $x^{(k+1)}$ por x^*), obtemos de fato o sistema $Ax^* = b$, então x^* é solução de $Ax = b$.
- Nos próximos slides vamos ver quais são as condições que garantem a convergência $x^{(k)} \to x^*$.

Convergência do método de Gauss-Seidel

- Dizemos que o processo iterativo converge (CV), se, para a sequência de aproximações gerada, dado $\varepsilon > 0$, existir $\overline{k}$, tal que $\forall k > \overline{k}$, temos $|x_i^{(k)} - x_i^*| \leq \varepsilon$ para todos $i = 1, 2, \ldots, n$.
- Na prática, este critério de convergência não é usável, pois não conhecemos x^*. Por isso precisamos de algum critério de parada. Definimos então:

$$\text{Var}^{(k)} = \max\{ V_1^{(k)}, V_2^{(k)}, \ldots, V_n^{(k)} \}$$

$$\text{onde } V_i^{(k)} = \begin{cases} \frac{|x_i^{(k)} - x_i^{(k-1)}|}{|x_i^{(k)}|} & \text{se} \quad x_i^{(k)} \neq 0 \\ 0 & \text{se} \quad x_i^{(k)} = 0 = x_i^{(k-1)} \\ 1 & \text{se} \quad x_i^{(k)} = 0 \text{ e } x_i^{(k-1)} \neq 0 \end{cases}$$

- Dizemos que o processo converge quando $\text{Var}^{(k)} \leq \varepsilon$ para algum k.
- Precisamos também estipular um número máximo de iterações IT_{max} para garantir que o algoritmo pare em tempo finito.

Método de Gauss-Seidel: exemplo

Exemplo: Consideramos o sistema linear seguinte:

$$(SL): \begin{cases} 4x_1 + x_2 + x_3 &= 5 \\ -2x_1 + 5x_2 + x_3 &= 0 \\ 3x_1 + x_2 + 6x_3 &= -6,5 \end{cases}$$

Trabalhamos com 3 algarismos significativos, com a inicialização $x^{(0)} = (0, 0, 0)$ e os parâmetros para o critério de parada $\varepsilon = 0,01$ e $IT_{max} = 5$.

As iterações do método de GS têm a forma seguinte:

$$\begin{cases} x_1^{(k+1)} &= \frac{1}{4}\left[5 - x_2^{(k)} - x_3^{(k)}\right] \\ x_2^{(k+1)} &= \frac{1}{5}\left[0 + 2x_1^{(k+1)} - x_3^{(k)}\right] \\ x_3^{(k+1)} &= \frac{1}{6}\left[-6,5 - 3x_1^{(k+1)} - x_2^{(k+1)}\right] \end{cases}$$

Lembramos que as entradas na iteração $(k+1)$ são calculadas nesta ordem:

$$x_1^{(k+1)} \to x_2^{(k+1)} \to x_3^{(k+1)}$$

Método de Gauss-Seidel: exemplo

Tabela de convergência: Lembramos que fazemos todos os cálculos com 3 algarismos significativos (quer dizer que estamos fazendo vários arredondamentos nos cálculos):

iteração k	$x_1^{(k)}$	$x_2^{(k)}$	$x_3^{(k)}$	$\text{Var}^{(k)}$
0	0	0	0	$-$
1	$1,25$	$0,5$	$-1,8$	1
2	$1,58$	$0,992$	$-2,03$	$0,496$
3	$1,51$	$1,01$	$-2,0$	$0,0464$
4	$1,5$	$1,0$	$-2,0$	$0,01 \leq \varepsilon$

De acordo com nosso critério de parada, como $\text{Var}^{(4)} \leq \varepsilon$, vamos considerar $x^{(4)} = (1,5; 1,0; -2,0)$ como solução aproximada do sistema. De fato, é fácil verificar que $x^{(4)}$ é a solução exata do (SL).

Apesar de ter acontecido neste pequeno exemplo, em geral é raro achar a solução exata de um sistema linear usando GS. Mas sempre podemos achar uma solução aproximada com precisão arbitrariamente alta, se calcularmos iterações suficientes.

Método de Gauss-Seidel: exemplo

Exemplo: Consideramos o sistema linear seguinte:

$$(SL): \begin{cases} 5x_1 + 3x_2 &= 15 \\ -4x_1 + 10x_2 &= 19 \end{cases}$$

Trabalhamos com 3 algarismos significativos, com a inicialização $x^{(0)} = (0; 0)$ e os parâmetros para o critério de parada $\varepsilon = 0,005$ e $IT_{max} = 10$.

As iterações do método de GS têm a forma seguinte:

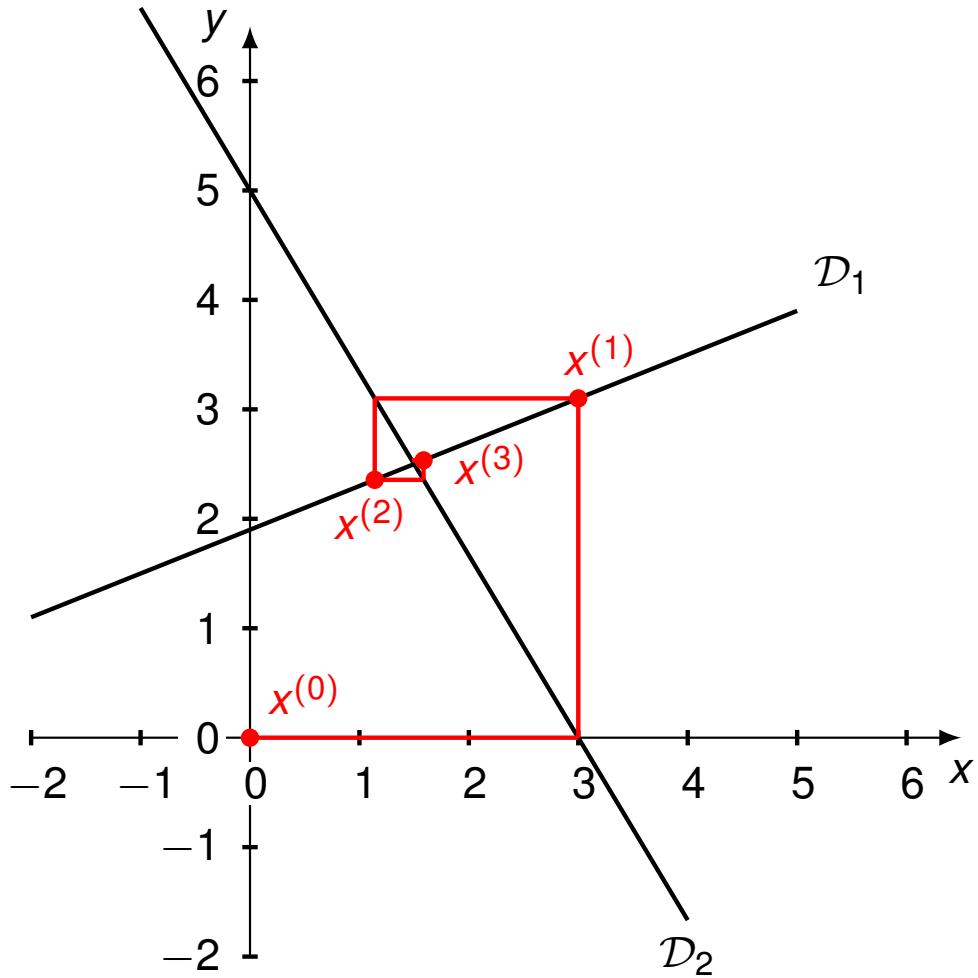
$$\begin{cases} x_1^{(k+1)} &= \frac{1}{5}\left[15 - 3x_2^{(k)}\right] \\ x_2^{(k+1)} &= \frac{1}{10}\left[19 + 4x_1^{(k+1)}\right] \end{cases}$$

Tabela de convergência: O algoritmo para na iteração 6 pois $\text{Var}^{(6)} = 0,004003 \leq \varepsilon$. Então a solução aproximada do (SL) é $x^{(6)} = (1,499; 2,5)$.

iteração k	$x_1^{(k)}$	$x_2^{(k)}$	$\text{Var}^{(k)}$
0	0	0	–
1	3,0	3,1	1
2	1,14	2,356	1,632
3	1,586	2,534	0,2812
4	1,480	2,492	0,07162
5	1,505	2,502	0,01661
6	1,499	2,5	$0,004003 \leq \varepsilon$

Método de Gauss-Seidel: exemplo

Interpretação geométrica do método GS: A solução x^* de (SL) é a interseção das retas $\mathcal{D}_1$ e $\mathcal{D}_2$, onde $\mathcal{D}_1$ tem a equação $-4x_1 + 10x_2 = 19$ e $\mathcal{D}_2$ tem a equação $5x_1 + 3x_2 = 15$. Observamos que as iterações $x^{(k)}$ seguem uma trajetoria espiral que converge para a solução exata x^*. Assim o GS pode ser interpretado como um *método de ponto fixo*, onde x^* seria o ponto fixo (ver o tópico "raízes de equações").



Convergência do método de Gauss-Seidel (dimensão 2)

Quando usamos Gauss-Seidel, é importante saber se a sequência $x^{(k)}$ produzida vai convergir. Existem condições sobre A que garantem a convergência do método GS.

Proposição: Dado o sistema linear seguinte, com $a_{11} \neq 0$ e $a_{22} \neq 0$:

$$(SL): \begin{cases} a_{11}x_1 + a_{12}x_2 &= b_1 \\ a_{21}x_1 + a_{22}x_2 &= b_2 \end{cases}$$

O processo iterativo de Gauss-Seidel, definido para $k \geq 1$ como

$$\begin{cases} x_1^{(k+1)} &= \frac{1}{a_{11}}\left[b_1 - a_{12}x_2^{(k)}\right] \\ x_2^{(k+1)} &= \frac{1}{a_{22}}\left[b_2 - a_{21}x_1^{(k+1)}\right] \end{cases}$$

converge se e somente se

$$\frac{|a_{12}a_{21}|}{|a_{11}a_{22}|} < 1.$$

Demonstração: Seja $x^* = (x_1^*, x_2^*)$ a solução exata do (SL) e $x^{(0)} = (x_1^{(0)}, x_2^{(0)})$ a inicialização. Definimos o erro na k-ésima iteração $\Delta x_i^{(k)} = x_i^* - x_i^{(k)}$, $i = 1, 2$. Dizemos que GS converge se e somente se

$$\lim_{k\to\infty} |\Delta x_i^{(k)}| = 0 \qquad \text{para } i = 1, 2.$$

Convergência do método de Gauss-Seidel (dimensão 2)

Temos

$$\begin{aligned}
\Delta x_1^{(k+1)} &= x_1^* - x_1^{(k+1)} = \frac{1}{a_{11}}\left[b_1 - a_{12}x_2^*\right] - \frac{1}{a_{11}}\left[b_1 - a_{12}x_2^{(k)}\right] \\
&= -\frac{a_{12}}{a_{11}}\left[x_2^* - x_2^{(k)}\right] = -\frac{a_{12}}{a_{11}}\Delta x_2^{(k)} \\
\Delta x_2^{(k)} &= x_2^* - x_2^{(k)} = \frac{1}{a_{22}}\left[b_2 - a_{21}x_1^*\right] - \frac{1}{a_{22}}\left[b_2 - a_{21}x_1^{(k)}\right] \\
&= -\frac{a_{21}}{a_{22}}\left[x_1^* - x_1^{(k)}\right] = -\frac{a_{21}}{a_{22}}\Delta x_1^{(k)}.
\end{aligned}$$

Assim, usando repetidas vezes essas igualdades, obtemos

$$\Delta x_1^{(k+1)} = -\frac{a_{12}}{a_{11}}\Delta x_2^{(k)} = \frac{a_{12}}{a_{11}}\frac{a_{21}}{a_{22}}\Delta x_1^{(k)} = \cdots = \left[\frac{a_{12}a_{21}}{a_{11}a_{22}}\right]^k \Delta x_1^{(1)}$$

e da mesma maneira

$$\Delta x_2^{(k+1)} = \left[\frac{a_{12}a_{21}}{a_{11}a_{22}}\right]^k \Delta x_2^{(1)}.$$

Observamos que os erros $\Delta x_i^{(1)}$, $i = 1, 2$ são independentes de k, eles dependem apenas do ponto incial $x^{(0)}$.

Convergência do método de Gauss-Seidel (dimensão 2)

Assim

$$\lim_{k\to\infty} |\Delta x_i^{(k)}| = \Delta x_i^{(1)} \lim_{k\to\infty} \left| \frac{a_{12}a_{21}}{a_{11}a_{22}} \right|^{k-1} \qquad \text{para } i = 1, 2.$$

Logo

$$\lim_{k\to\infty} |\Delta x_i^{(k)}| = 0 \text{ para } i = 1, 2 \quad \Leftrightarrow \quad \left| \frac{a_{12}a_{21}}{a_{11}a_{22}} \right| < 1$$

Convergência do método de Gauss-Seidel (dimensão 2)

Exemplo: Consideramos o sistema linear seguinte:

$$(SL): \begin{cases} -4x_1 + 10x_2 &= 19 \\ 5x_1 + 3x_2 &= 15 \end{cases}$$

Temos

$$\left|\frac{a_{12}a_{21}}{a_{11}a_{22}}\right| = \left|\frac{10 \times 5}{-4 \times 3}\right| = \frac{50}{12} > 1,$$

então GS não converge nessa configuração. Por outro lado, se trocarmos as linhas obtemos

$$\left|\frac{\tilde{a}_{12}\tilde{a}_{21}}{\tilde{a}_{11}\tilde{a}_{22}}\right| = \left|\frac{-4 \times 3}{10 \times 5}\right| = \frac{12}{50} < 1,$$

e GS converge nessa configuração. Então para todos sistemas de dimensão $n = 2$, sempre podemos ordenar as linhas de tal maneira que GS converge se

$$\left|\frac{a_{12}a_{21}}{a_{11}a_{22}}\right| \neq 1.$$

Se $\left|\frac{a_{12}a_{21}}{a_{11}a_{22}}\right| = 1$, pode acontecer que GS não converge em dimensão $n = 2$.

Convergência do método GS (critério de Sassenfeld)

- Vamos estudar condições suficientes para garantir a convergência do método GS para um sistema de ordem n.
- Seja $Ax = b$ de ordem n, com $a_{ii} \neq 0, \forall i = 1, 2, \ldots, n$.

Critério de Sassenfeld: Definimos

$$\beta_1 = \frac{1}{|a_{11}|} \sum_{j=2}^{n} |a_{1j}|, \qquad \beta_i = \frac{1}{|a_{ii}|} \left[\sum_{j=1}^{i-1} |a_{ij}|\beta_j + \sum_{j=i+1}^{n} |a_{ij}| \right], \quad \text{para } i = 2, 3, \ldots, n$$

Exemplo: Consideramos o (SL) seguinte:

$$\begin{pmatrix} 2 & 1 & -0,2 & 0,2 \\ 0,6 & 3 & -0,6 & -0,3 \\ -0,1 & -0,2 & 1 & 0,2 \\ 0,4 & 1,2 & 0,8 & 4 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{pmatrix} = \begin{pmatrix} 0,4 \\ -7,8 \\ 1,0 \\ -10,0 \end{pmatrix}$$

Calculamos:

$$\beta_1 = \frac{1}{2}(1 + 0,2 + 0,2) = 0,7$$

$$\beta_2 = \frac{1}{3}(0,6 \times 0,7 + 0,6 + 0,3) = 0,44$$

$$\beta_3 = \frac{1}{1}(0,1 \times 0,7 + 0,2 \times 0,44 + 0,2) = 0,358$$

$$\beta_4 = \frac{1}{4}(0,4 \times 0,7 + 1,2 \times 0,44 + 0,8 \times 0,358) = 0,2736$$

Convergência do método GS (critério de Sassenfeld)

Lema 1: Dado um sistema $Ax = b$, definimos o erro $\Delta x_i^{(k)} = x_i^* - x_i^{(k)}$, para todo $k \geq 1$, onde x^* é a solução exata. Então

$$|\Delta x_i^{(k+1)}| \leq \beta_i \max_{1 \leq j \leq n} |\Delta x_j^{(k)}|, k \geq 1$$

A demonstração é feita por indução, ver Humes / Melo / Yoshida / Martins, *Noções de Cálculo Numérico*, McGraw-Hill do Brasil, 1984 (ed 32), p. 74.

Proposição (critério de Sassenfeld): Seja $M = \max_{1 \leq i \leq n} \beta_i$. Se $M < 1$, então a iteração $x^{(k)}$ do método de Gauss-Seidel converge para a solução de $Ax = b$ quando $k \to \infty$.

Demonstração: Vamos mostrar que $\lim_{k \to \infty} |\Delta x_i^{(k)}| = 0$ para todos $i = 1, 2, \ldots, n$, onde $\Delta x_i^{(k)} = x_i^* - x_i^{(k)}$ é o erro e x^* é a solução exata de $Ax = b$. Usando o Lema 1, obtemos

$$|\Delta x_i^{(k+1)}| \leq M \max_{1 \leq j \leq n} |\Delta x_j^{(k)}|$$

$$\Rightarrow \max_{1 \leq i \leq n} |\Delta x_i^{(k+1)}| \leq M \max_{1 \leq j \leq n} |\Delta x_j^{(k)}| \leq M^2 \max_{1 \leq j \leq n} |\Delta x_j^{(k-1)}| \leq \cdots \leq M^k \max_{1 \leq j \leq n} |\Delta x_j^{(1)}|$$

$$\Rightarrow \lim_{k \to \infty} \max_{1 \leq i \leq n} |\Delta x_i^{(k+1)}| \leq M^k \max_{1 \leq j \leq n} |\Delta x_j^{(1)}| \to 0 \quad \text{pois } M < 1.$$

Então $\lim_{k \to \infty} |\Delta x_i^{(k)}| = 0$ para todo $1 \leq i \leq n$.

Convergência do método GS (critério de Sassenfeld)

Exemplo: Consideramos o (SL) seguinte:

$$\begin{pmatrix} 2 & 1 & -0,2 & 0,2 \\ 0,6 & 3 & -0,6 & -0,3 \\ -0,1 & -0,2 & 1 & 0,2 \\ 0,4 & 1,2 & 0,8 & 4 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{pmatrix} = \begin{pmatrix} 0,4 \\ -7,8 \\ 1,0 \\ -10,0 \end{pmatrix}$$

Calculamos $M = \max_{1 \leq i \leq n} \beta_i = 0,7$, então aplicando o critério de Sassenfeld obtemos que a sequência $x^{(k)}$ de GS converge para a solução de $Ax = b$. Essa convergência é independente da inicialização.

Observação: O critério de Sassenfeld é apenas uma *condição suficiente* de convergência. Isso quer dizer que a sequência de GS pode convergir, mesmo que o critério de Sassenfeld não esteja satisfeito. Por exemplo considere

$$\begin{aligned} 2x_1 + 4x_2 &= 14, \\ x_1 + 5x_2 &= 11. \end{aligned}$$

Este sistema não satisfaz o critério de Sassenfeld pois $\beta_1 = 2 > 1$, mas a iteração de GS converge pois (ver a proposição no caso da dimensão 2):

$$\left| \frac{a_{12}a_{21}}{a_{11}a_{22}} \right| = \frac{4}{10} < 1.$$

Convergência do método GS (critério de Sassenfeld)

Proposição: Seja $Ax = b$ um sistema linear dado, e suponhamos que o critério de Sassenfeld esteja satisfeito, isto é $M = \max_{1 \leq i \leq n} \beta_i < 1$. Dado $\varepsilon > 0$, se

$$\max_{1 \leq j \leq n} |x_j^{(k+1)} - x_j^{(k)}| \leq \frac{1 - M}{M} \varepsilon \quad \text{para algum } k \geq 1,$$

então $|\Delta x_i^{(k+1)}| \leq \varepsilon$ para todo $i = 1, 2, \ldots, n$, onde $\Delta x_i^{(k+1)} = x_i^* - x_i^{(k+1)}$ e x^* é a solução exata de $Ax = b$.

Demonstração: Ver Humes / Melo / Yoshida / Martins, *Noções de Cálculo Numérico*, McGraw-Hill do Brasil, 1984 (ed 32), proposição 8.3, p. 77.

Observação: Esta proposição pode ser usada como critério de parada no algoritmo de GS, pois precisamos apenas calcular M e a diferença entre duas iterações consecutivas $x^{(k)}$ e $x^{(k+1)}$. Assim podemos, em princípio, nos aproximar da solução exata x^* com uma precisão arbitrária ε.

Convergência do método GS (critério das linhas)

Proposição: Se os coeficientes de A satisfazem

$$(CL): \sum_{j=1, j\neq i}^{n} |a_{ij}| < |a_{ii}|, \quad i = 1, 2, \ldots, n, \quad \text{com } a_{ii} \neq 0,$$

então a iteração $x^{(k)}$ de GS converge para x^* solução de $Ax = b$.

Demonstração: Vamos mostrar que (CL) implica que o critério de Sassenfeld esteja satisfeito, e isso implica $x^{(k)} \to x^*$. Para isso, vamos mostrar $\beta_i < 1, i = 1, 2, \ldots, n$, por indução sobre i.

- Base da indução:

$$\beta_1 = \frac{1}{|a_{11}|} \sum_{j=2}^{n} |a_{1j}| < \frac{1}{|a_{11}|} |a_{11}| = 1$$

- Hipótese de indução: suponhamos que $\beta_j < 1$ para todos $j = 1, 2, \ldots, i-1$.
- Indução: per definição temos

$$\beta_i = \frac{1}{|a_{ii}|} \left[\sum_{j=1}^{i-1} |a_{ij}| \beta_j + \sum_{j=i+1}^{n} |a_{ij}| \right] \underbrace{\leq}_{\substack{\text{hipótese} \\ \text{de indução}}} \frac{1}{|a_{ii}|} \left[\sum_{j=1}^{i-1} |a_{ij}| + \sum_{j=i+1}^{n} |a_{ij}| \right] \underbrace{\leq}_{\substack{\text{critério} \\ \text{das linhas}}} \frac{|a_{ii}|}{|a_{ii}|} = 1$$

$$\underbrace{\Rightarrow}_{\text{indução}} \beta_i < 1 \text{ para todo } 1 \leq i \leq n \Rightarrow M = \max_{1 \leq i \leq n} \beta_i < 1 \Rightarrow \text{ critério de Sassenfeld satisfeito}$$

Observação: Existem sistemas que satisfazem Sassenfeld sem satisfazer (CL).

Convergência do método GS (critério das linhas)

Exemplo: Consideramos o (SL) seguinte:

$$\begin{pmatrix} 2 & 1 & -0,2 & 0,2 \\ 0,6 & 3 & -0,6 & -0,3 \\ -0,1 & -0,2 & 1 & 0,2 \\ 0,4 & 1,2 & 0,8 & 4 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{pmatrix} = \begin{pmatrix} 0,4 \\ -7,8 \\ 1,0 \\ -10,0 \end{pmatrix}$$

Vamos verificar se o critério das linhas está satisfeito:

$$(CL): \sum_{j=1, j\neq i}^{n} |a_{ij}| < |a_{ii}|, \quad i = 1, 2, \ldots, n, \quad \text{com } a_{ii} \neq 0.$$

Calculamos:

$$\begin{aligned} |1| + |-0,2| + |0,2| = 1,4 < |2| \quad & \text{1a linha} \\ |0,6| + |-0,6| + |-0,3| = 1,5 < |3| \quad & \text{2a linha} \\ |-0,1| + |-0,2| + |0,2| = 0,5 < |1| \quad & \text{3a linha} \\ |0,4| + |1,2| + |0,8| = 2,4 < |4| \quad & \text{4a linha} \end{aligned}$$

Então o critério das linhas está satisfeito, e consequentemente a iteração $x^{(k)}$ de GS converge para x^* solução de $Ax = b$.

Método SOR

- SOR = Successive Over-Relaxation (método de sobre-relaxamento)
- O método SOR é uma generalização e aperfeiçoamento do método GS.
- O objetivo é aproximar a solução de um sistema $Ax = b$ com $A \in \mathbb{R}^{n\times n}$ e $b \in \mathbb{R}^n$.
- Seja $x^{(k)}$ a sequência produzida usando GS. Definimos o resíduo

$$r^{(k),i} = b - A\hat{x}^{(k),i}, \text{ onde } \hat{x}^{(k),i} = (x_1^{(k)}, \ldots, x_{i-1}^{(k)}, x_i^{(k-1)}, \ldots, x_n^{(k-1)}).$$

- A iteração de GS satisfaz

$$x_i^{(k)} = \frac{1}{a_{ii}} \left(b_i - \sum_{j=1}^{i-1} a_{ij} x_j^{(k)} - \sum_{j=i+1}^{n} a_{ij} x_j^{(k-1)} \right)$$

Exercício 1: mostre que

$$x_i^{(k)} = x_i^{(k-1)} + \frac{r_i^{(k),i}}{a_{ii}}$$

Exercício 2: mostre que $r_{i+1}^{(k),i} = 0$ no método GS.

O objetivo dos métodos iterativos é que o resíduo $r^{(k),i}$ convirja para zero quando $k \to \infty$. Porém, a propriedade $r_{i+1}^{(k),i} = 0$ do método GS não é necessariamente a maneira mais eficiente de atingir este objetivo.

Método SOR

Então vamos considerar o método SOR seguinte, que é uma generalização de GS:

$$x_i^{(k)} = x_i^{(k-1)} + \omega \frac{r_i^{(k),i}}{a_{ii}} \text{ com } \omega > 0.$$

- Caso $0 < \omega < 1$: *sub-relaxamento*, para obter convergência quando GS não converge
- Caso $1 < \omega$: *sobre-relaxamento*, para acelerar a convergência quando GS converge
- Caso $\omega = 1$: este caso particular é GS.

O método SOR para aproximar a solução de um sistema $Ax = b$ de ordem n pode ser escrito desta forma:

$$x_i^{(k)} = (1-\omega)x_i^{(k-1)} + \frac{\omega}{a_{ii}} \left(b_i - \sum_{j=1}^{i-1} a_{ij} x_j^{(k)} - \sum_{j=i+1}^{n} a_{ij} x_j^{(k-1)} \right) \text{ para } i = 1, 2, \ldots, n.$$

Precisa calcular nesta ordem: $x_1^{(k)} \to x_2^{(k)} \to \cdots \to x_n^{(k)}$

Método SOR

Podemos também usar uma forma matricial para escrever o método SOR. Primeiro, rearranjamos a iteração para $x_i^{(k)}$ na forma seguinte:

$$x_i^{(k)} a_{ii} + \omega \sum_{j=1}^{i-1} a_{ij} x_j^{(k)} = (1-\omega) a_{ii} x_i^{(k-1)} - \omega \sum_{j=i+1}^{n} a_{ij} x_j^{(k-1)} + \omega b_i.$$

Em forma matricial, escreve-se:

$$(D - \omega L) x^{(k)} = [(1-\omega) D + \omega U] x^{(k-1)} + \omega b$$

onde $D = \text{diag}(a_{11}, a_{22}, \ldots, a_{nn})$,

$$L = \begin{bmatrix} 0 & \ldots & \ldots & 0 \\ -a_{21} & 0 & & \vdots \\ \vdots & \ddots & \ddots & \vdots \\ -a_{n1} & \ldots & -a_{n,n-1} & 0 \end{bmatrix} \qquad U = \begin{bmatrix} 0 & -a_{12} & \ldots & -a_{1n} \\ \vdots & 0 & \ddots & \vdots \\ \vdots & & \ddots & -a_{n-1,n} \\ 0 & \ldots & \ldots & 0 \end{bmatrix}$$

Método SOR

- Assim, a iteração do método SOR pode ser escrita na forma:

$$x^{(k)} = T_\omega x^{(k-1)} + c_\omega$$

onde

$$T_\omega = (D - \omega L)^{-1}[(1-\omega)D + \omega U], \qquad c_\omega = \omega(D - \omega L)^{-1} b$$

- Essa forma matricial da iteração do método SOR é útil para mostrar alguns resultados teóricos de convergência, estudando propriedades da matriz T_ω.
- Uma questão importante é: *como escolher o valor apropriado de ω para acelerar a convergência de GS?* Não existe resposta completa no caso geral, mas existem resultados para casos particulares.

Método SOR

Teorema (Kahan): Se $a_{ii} \neq 0$ para todos $i = 1, 2, \ldots, n$, então o método SOR pode convergir apenas se $0 < \omega < 2$.

Teorema (Ostrowski-Reich): Se A for definida positiva, e se $0 < \omega < 2$, então o método SOR converge para todo vetor inicial $x^{(0)}$ ($A \in \mathbb{R}^{n \times n}$ é definida positiva $\Leftrightarrow \langle Ax, x \rangle > 0$ para todos $x \neq 0$, $x \in \mathbb{R}^n$).

Definição (raio espectral): $A \in \mathbb{R}^{n \times n}$, então $\rho(A) = \max\{|\lambda_1|, \ldots, |\lambda_n|\}$ é o raio espectral de A, onde $\{\lambda_1, \ldots, \lambda_n\}$ são os autovalores de A.

Teorema: Se $A \in \mathbb{R}^{n \times n}$ for definida positiva e tridiagonal, então $\rho(T_g) = [\rho(T_j)]^2 < 1$, onde $\rho(T_g)$ é o raio espectral da matriz do método de GS, e $\rho(T_j)$ o raio espectral da matriz do método de Jacobi (isto é, com as iterações $x^{(k)} = T_g x^{(k-1)} + c_g$ e $x^{(k)} = T_j x^{(k-1)} + c_j$, com $T_g = (D - L)^{-1} U$, $c_g = (D - L)^{-1} b$ e $T_j = D^{-1}(L + U)$, $c_j = D^{-1} b$). A escolha ótima de ω para o método SOR é neste caso:

$$\omega = \frac{2}{1 + \sqrt{1 - [\rho(T_j)]^2}}, \qquad \text{e temos } \rho(T_\omega) = \omega - 1 < \rho(T_g) = [\rho(T_j)]^2$$

Observação: Este teorema mostra que a convergência de SOR é mais rápida que GS para essa escolha de ω.

Método SOR

Exemplo: Consideramos a matriz tridiagonal

$$A = \begin{bmatrix} 4 & 3 & 0 \\ 3 & 4 & -1 \\ 0 & -1 & 4 \end{bmatrix}$$

Podemos mostrar que A é definida positiva (exercício - mostre que as submatrizes principais de A têm determinantes positivos). Depois calculamos

$$T_j = D^{-1}(L+U) = \begin{bmatrix} 1/4 & 0 & 0 \\ 0 & 1/4 & 0 \\ 0 & 0 & 1/4 \end{bmatrix} \begin{bmatrix} 0 & -3 & 0 \\ -3 & 0 & 1 \\ 0 & 1 & 0 \end{bmatrix} = \begin{bmatrix} 0 & -0,75 & 0 \\ -0,75 & 0 & 0,25 \\ 0 & 0,25 & 0 \end{bmatrix}$$

Temos $\det(T_j - \lambda I) = -\lambda(\lambda^2 - 0,625)$, então $\rho(T_j) = \sqrt{0,625}$ e

$$\omega = \frac{2}{1 + \sqrt{1 - [\rho(T_j)]^2}} = \frac{2}{1 + \sqrt{1 - 0,625}} \approx 1,24.$$

Então o teorema indica que a escolha $\omega \approx 1,24$ no método SOR fornece uma convergência mais rápida que GS para a resolução de sistemas lineares $Ax = b$ com a matriz A dada acima.