Métodos Iterativos: Gradientes Conjugados

INF1608 – Análise Numérica

Departamento de Informática, PUC-Rio

Reformulando o Problema

Sistema linear

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

Exemplo:

$$\begin{bmatrix} 3 & 2 \\ 2 & 6 \end{bmatrix} \mathbf{b} = \begin{bmatrix} 2 \\ -8 \end{bmatrix}$$

- Solução:

$$\mathbf{x} = \begin{bmatrix} 2 \\ -2 \end{bmatrix}$$

- Interpretação geométrica
  - Interseção dos hiperplanos

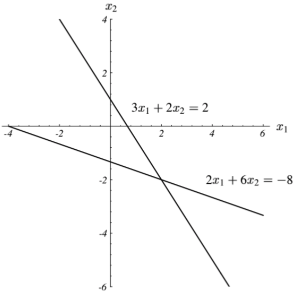


Figura extraída de "An Introduction to the Conjugate Gradient Method Without the Agonizing Pain", Shewchuk, 1994

Reformulando o Problema

Forma Quadrática

- Equação escalar na forma:

$$f(\mathbf{x}) = \frac{1}{2}\mathbf{x}^T A\mathbf{x} - \mathbf{b}^T\mathbf{x} + c$$

  - onde c é um valor escalar (p.e. $c = 0$)

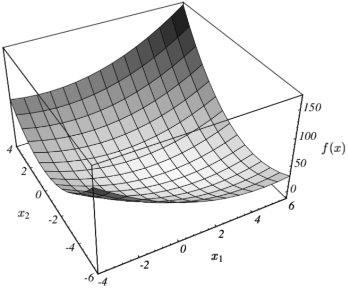


Figura extraída de "An Introduction to the Conjugate Gradient Method Without the Agonizing Pain", Shewchuk, 1994

Reformulando o Problema

Gradiente da forma quadrática (sendo A simétrica)

$$f'(\mathbf{x}) = A\mathbf{x} - b$$

- Ponto mínimo de $f(x)$:

$$f'(\mathbf{x}) = 0 \rightarrow A\mathbf{x} = \mathbf{b}$$

- Sendo A positiva definida, $f'(\mathbf{x}) = 0$ será sempre um **ponto de mínimo**, pois:

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

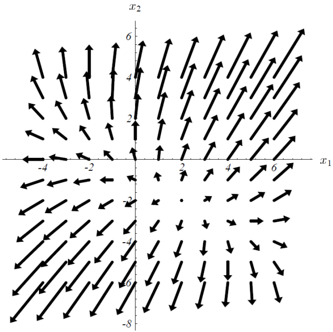
- Dessa forma, determinar solução de $A\mathbf{x} = \mathbf{b}$ é equivalente a encontrar o ponto **mínimo** de $f(\mathbf{x})$

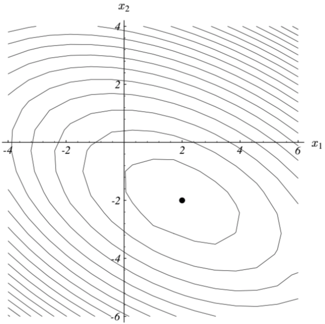
Reformulando o Problema

Gradiente da forma quadrática

$$f'(\mathbf{x}) = A\mathbf{x} - b$$

Podemos achar o mínimo através do gradiente.



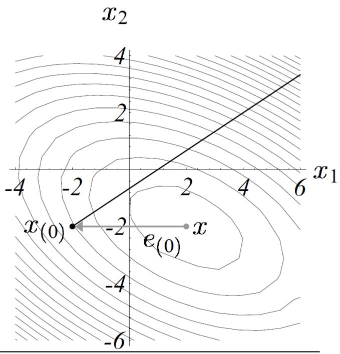


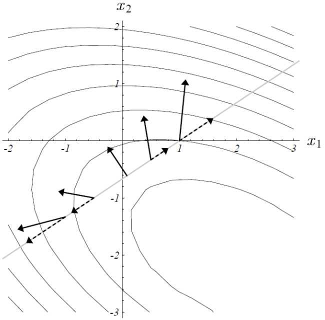
Figuras extraídas de "An Introduction to the Conjugate Gradient Method Without the Agonizing Pain", Shewchuk, 1994

Determinação do Mínimo da Função

Método *Steepest Descent* (Gradiente descendente)

- Tomar a direção inversa do gradiente
- Para cada passo: $x_1 = x_0 + \alpha r_0$ (até um certo $r_i < tol$)
  - Lembrando que $r_i = b - Ax_i$, logo $f'(x_i) = -r_i$
  - Qual o valor de α? (ponto em $r_1^T r_0 = 0$)





Figuras extraídas de "An Introduction to the Conjugate Gradient Method Without the Agonizing Pain", Shewchuk, 1994

Método *Steepest Descent* (Gradiente descendente)

Algoritmo:

$$
\begin{aligned}
r_k =& b - Ax_k \\
\alpha_k =& \frac{r_k^T r_k}{r_k^T A r_k} \\
x_{k+1} =& x_i + \alpha_k r_k
\end{aligned}
$$

Pré-multiplicando x_{k+1} por $-A$ e somando $+b$:

$$
\begin{aligned}
-Ax_{k+1} + b =& - Ax_i - \alpha_k A r_k + b \\
b - Ax_{k+1} =& b - Ax_i - \alpha_k A r_k \\
r_{k+1} =& r_k - \alpha_k A r_k
\end{aligned}
$$

(calcula Ar_k apenas 1 vez para α_k e r_{k+1})

Método *Steepest Descent* (Gradiente descendente)

Converge, mas resulta em muitas iterações Passos em direções repetidas

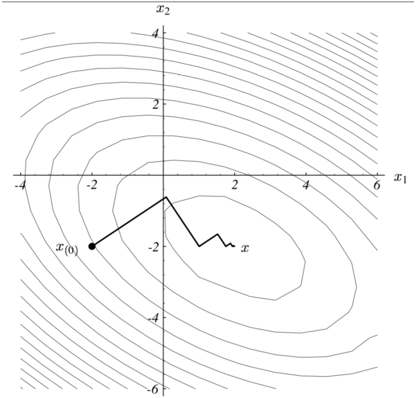


Figura extraída de "An Introduction to the Conjugate Gradient Method Without the Agonizing Pain", Shewchuk, 1994

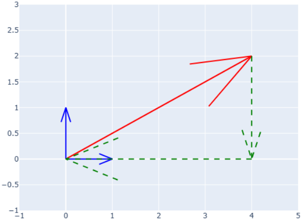
Determinando mínimo em n passos

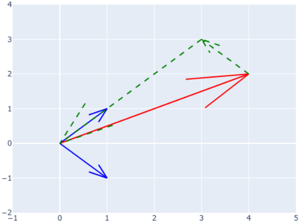
E se soubermos o valor do erro?

- Desconsiderando o óbvio $x_s = x_k - e_k$

Podemos zerar cada uma das n componentes utilizando um conjunto de vetores ortogonais ($u_i^T u_j = 0$)

- Remove todo o erro em uma direção
- Cobre todo o espaço de dimensão n
- Erro zerado após n iterações





Determinando mínimo em n passos

Buscar as direções ortogonais $\mathbf{d}_0, \mathbf{d}_1, ..., \mathbf{d}_{n-1}$

- Erro expresso como combinação linear de $\mathbf{d}_k's$
- Um passo em cada direção
  - Elimina cada componente do erro na direção $\mathbf{d}_k$

Procedimento

$$\mathbf{x}_{k+1} = \mathbf{x}_k + \alpha_k \mathbf{d}_k$$

- Determinação de α_k
  - $\mathbf{e}_{k+1}$ ortogonal a $\mathbf{d}_k$

$$\mathbf{d}_k^T \mathbf{e}_{k+1} = 0$$

$$\mathbf{d}_k^T (\mathbf{e}_k + \alpha_k \mathbf{d}_k) = 0$$

$$\alpha_k = -\frac{\mathbf{d}_k^T \mathbf{e}_k}{\mathbf{d}_k^T \mathbf{d}_k}$$

  - Mas não conhecemos $\mathbf{e}_k$

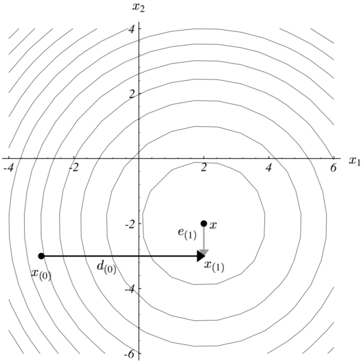


Figura extraída de "An Introduction to the Conjugate Gradient Method Without the Agonizing Pain", Shewchuk, 1994

Solução

Procurar vetores A-ortogonais, ou conjugados ($d_i^T A d_j = 0$)

- d_k passa a ser A-ortogonal a e_{k+1}, e erro composto por d_k's
  - i.e. $e_0 = \sum_{j=0}^{n-1} \delta_j d_j$ e $e_n = 0$
- Cada passo $\delta_j d_j$ remove uma das componentes d_j
  - Linearmente independentes para uma matriz SPD
  - Abrangem todo o espaço de dimensão n

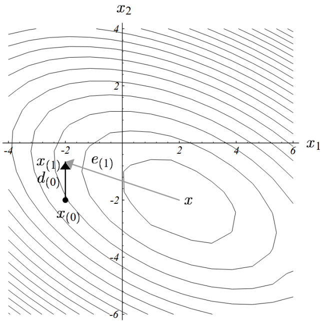


Figura extraída de "An Introduction to the Conjugate Gradient Method Without the Agonizing Pain", Shewchuk, 1994

Exemplo [1]

$$\begin{bmatrix} 2 & 1 \\ 1 & 2 \end{bmatrix} \begin{bmatrix} x_0 \\ x_1 \end{bmatrix} = \begin{bmatrix} 10 \\ 8 \end{bmatrix}, \; x_0 = \begin{bmatrix} 0 \\ 0 \end{bmatrix}, \; x_s = \begin{bmatrix} 4 \\ 2 \end{bmatrix}, \; Ae_0 = \begin{bmatrix} -10 \\ -8 \end{bmatrix}$$

A partir das direções $\begin{bmatrix} 1 \\ 1 \end{bmatrix}$ e $\begin{bmatrix} 1 \\ -1 \end{bmatrix}$: $\begin{bmatrix} 1 \\ 1 \end{bmatrix}^T A \begin{bmatrix} 1 \\ -1 \end{bmatrix} = 0$

$$e_0 = -3 \begin{bmatrix} 1 \\ 1 \end{bmatrix} - 1 \begin{bmatrix} 1 \\ -1 \end{bmatrix}, \; Ae_0 = -3A \begin{bmatrix} 1 \\ 1 \end{bmatrix} - 1A \begin{bmatrix} 1 \\ -1 \end{bmatrix}$$

$$x_s = \begin{bmatrix} 0 \\ 0 \end{bmatrix} + 3 \begin{bmatrix} 1 \\ 1 \end{bmatrix} + 1 \begin{bmatrix} 1 \\ -1 \end{bmatrix}$$

(Achar x_s é análogo a zerar o erro, mas qual o valor do erro?)

Exemplo [2]

$$\begin{bmatrix} 3 & 2 \\ 2 & 6 \end{bmatrix} \begin{bmatrix} x_0 \\ x_1 \end{bmatrix} = \begin{bmatrix} 2 \\ -8 \end{bmatrix}, \; x_s = \begin{bmatrix} 2 \\ -2 \end{bmatrix}, \; Ae_0 = \begin{bmatrix} -2 \\ +8 \end{bmatrix}$$

A partir das direções $\begin{bmatrix} 1 \\ 2 \end{bmatrix}$ e $\begin{bmatrix} -2 \\ 1 \end{bmatrix}$: $\begin{bmatrix} 1 \\ 2 \end{bmatrix}^T A \begin{bmatrix} -2 \\ 1 \end{bmatrix} = 0$

$$e_0 = +0.4 \begin{bmatrix} 1 \\ 2 \end{bmatrix} + 1.2 \begin{bmatrix} -2 \\ 1 \end{bmatrix}, \; Ae_0 = +0.4A \begin{bmatrix} 1 \\ 2 \end{bmatrix} + 1.2A \begin{bmatrix} -2 \\ 1 \end{bmatrix}$$

$$x_s = \begin{bmatrix} 0 \\ 0 \end{bmatrix} - 0.4 \begin{bmatrix} 1 \\ 2 \end{bmatrix} - 1.2 \begin{bmatrix} -2 \\ 1 \end{bmatrix}$$

(Achar x_s é análogo a zerar o erro, mas qual o valor do erro?)

Erro transformado

Erro da solução

$$\mathbf{e}_k = \mathbf{x}_k - \mathbf{x}_s, \quad \mathbf{x}_s = \text{solução}$$

Resíduo

$$\mathbf{r}_k = \mathbf{b} - A\mathbf{x}_k$$

Substituindo $\mathbf{x}_k$ no resíduo...

$$\mathbf{r}_k = \mathbf{b} - A(\mathbf{e}_k + \mathbf{x}_s), \quad (\mathbf{b} - A\mathbf{x}_s) = 0$$
$$\mathbf{r}_k = -A\mathbf{e}_k = -f'(\mathbf{x}), \quad \text{sabendo que } f'(\mathbf{x}) = A\mathbf{x} - b$$

- O resíduo é o erro transformado por A no mesmo espaço de **b**
- Podemos zerar as componentes de r por direções A-ortogonais

Método Gradiente Conjugado

Aplicado a uma matriz simétrica positiva definida

Dada uma estimativa inicial $\mathbf{x}_0$...

- Elimina, um a um, as componentes do erro

Método direto *versus* iterativo:

- **Direto**: Solução exata após n iterações
- **Iterativo**: Parar antes de n iterações via tolerância numérica

Atualiza 3 vetores a cada iteração:

1. Vetor solução x_k
2. Vetor resíduo r_k
   - Sendo que $Ax_{k+1} + r_{k+1} = Ax_k + r_k = b$
3. Vetor direção d_k, onde $d_{k+1}Ad_k = 0$
   - Atualiza x_k para uma melhor solução x_{k+1}
   - Atualiza resíduo r_k para r_{k+1}

Método Gradiente Conjugado: Premissas/Propriedades

Direções de busca conjugadas em pares: $\mathbf{d}_{k+1} A \mathbf{d}_k = 0$

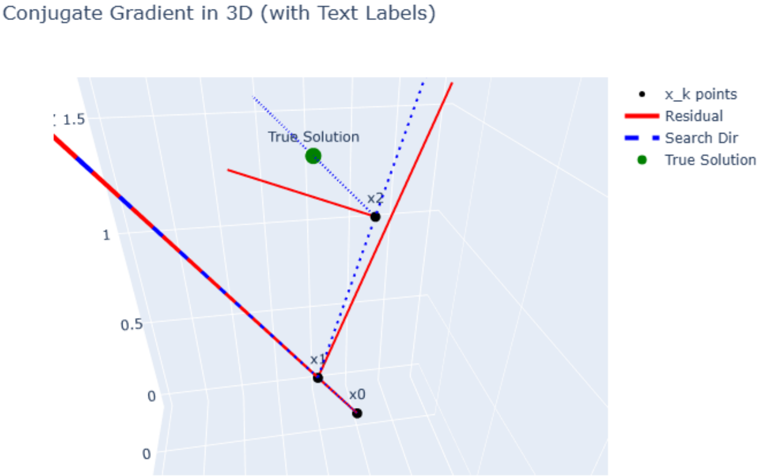
Direção de busca ortogonal ao resíduo

- $\mathbf{d}_k$ é A-ortogonal a $\mathbf{e}_{k+1}$: $\mathbf{d}_k \perp A\mathbf{e}_{k+1}$
- $\mathbf{d}_k$ é ortogonal a $\mathbf{r}_{k+1} = -A\mathbf{e}_{k+1}$: $\mathbf{d}_k \perp \mathbf{r}_{k+1}$
- $\mathbf{d}_k$ zera a componente de $\mathbf{r}_k$, na direção $\mathbf{d}_k$

Ortogonalidade entre resíduos $\mathbf{r}_{k+1} \perp \mathbf{r}_k$

- Passos são gerados para garantir essa propriedade
- Resíduo resultante $\mathbf{r}_{k+1}$ não possui projeção ao longo de $\mathbf{r}_k$

Método Gradiente Conjugado: Exemplo de matriz 3x3



Método Gradiente Conjugado

Atualização do vetor solução

$$\mathbf{x}_{k+1} = \mathbf{x}_k + \alpha_k \mathbf{d}_k$$

Atualização do vetor resíduo

- Sabe-se que:

$$A\mathbf{x}_{k+1} + \mathbf{r}_{k+1} = A\mathbf{x}_k + \mathbf{r}_k = b$$

  - r_{k+1} menor que r_k, e x_{k+1} mais próximo da solução x
- Substituindo x_{k+1}:

$$\begin{aligned} A(\mathbf{x}_k + \alpha_k \mathbf{d}_k) + \mathbf{r}_{k+1} &= A\mathbf{x}_k + \mathbf{r}_k \\ A\mathbf{x}_k + \alpha_k A\mathbf{d}_k + \mathbf{r}_{k+1} &= A\mathbf{x}_k + \mathbf{r}_k \\ \mathbf{r}_{k+1} &= \mathbf{r}_k - \alpha_k A\mathbf{d}_k \end{aligned}$$

Método Gradiente Conjugado

Confirmando, vimos que o resíduo é o erro transformado por A:

$$\mathbf{r}_k = -A\mathbf{e}_k$$

E o erro é atualizado a partir de:

$$\mathbf{e}_{k+1} = \mathbf{e}_k + \alpha_k \mathbf{d}_k$$

Pré-multiplicando por $-A$:

$$\begin{aligned} -A\mathbf{e}_{k+1} &= -A\mathbf{e}_k - \alpha_k A\mathbf{d}_k \\ \mathbf{r}_{k+1} &= \mathbf{r}_k - \alpha_k A\mathbf{d}_k \end{aligned}$$

Método Gradiente Conjugado

Escolha de α_k

- Como $\mathbf{d}_k$ deve ser ortogonal a $\mathbf{r}_{k+1}$

$$\mathbf{d}_k^T \mathbf{r}_{k+1} = 0$$

- Como $\mathbf{r}_{k+1} = \mathbf{r}_k - \alpha_k A\mathbf{d}_k$:

$$\begin{aligned} \mathbf{d}_k^T(\mathbf{r}_k - \alpha_k A\mathbf{d}_k) &= 0 \\ \mathbf{d}_k^T \mathbf{r}_k &= \alpha_k \mathbf{d}_k^T A\mathbf{d}_k \\ \therefore \alpha_k &= \frac{\mathbf{d}_k^T \mathbf{r}_k}{\mathbf{d}_k^T A\mathbf{d}_k} \end{aligned}$$

Método Gradiente Conjugado

Escolha da nova direção de busca

- Resíduo é expresso por uma combinação linear de $\mathbf{d}_k$'s

$$\mathbf{d}_{k+1} = \mathbf{r}_{k+1} + \beta_k \mathbf{d}_k$$

- Partindo do requisito das direções serem conjugadas entre si

$$\mathbf{d}_{k+1} A \mathbf{d}_k = 0$$

- Expandindo d_{k+1}:

$$(\mathbf{r}_{k+1}^T + \beta_k \mathbf{d}_k)^T (A\mathbf{d}_k) = \mathbf{r}_{k+1}^T A\mathbf{d}_k + \beta_k \mathbf{d}_k^T A \mathbf{d}_k = 0$$

$$\therefore \beta_k = -\frac{\mathbf{r}_{k+1}^T A \mathbf{d}_k}{\mathbf{d}_k^T A \mathbf{d}_k}$$

Método Gradiente Conjugado: Algoritmo [1]

$\mathbf{x}_0 =$ estimativa inicial

$\mathbf{d}_0 = \mathbf{r}_0 = \mathbf{b} - A\mathbf{x}$

for $k = 0, 1, \cdots, n-1$ **do**

  if $||\mathbf{r}_k||_2 < tol$ **then**

    stop

$$\alpha_k = \frac{\mathbf{d}_k^T \mathbf{r}_k}{\mathbf{d}_k^T A \mathbf{d}_k}$$

$$\mathbf{x}_{k+1} = \mathbf{x}_k + \alpha_k \mathbf{d}_k$$

$$\mathbf{r}_{k+1} = \mathbf{r}_k - \alpha_k A \mathbf{d}_k$$

$$\beta_k = -\frac{\mathbf{r}_{k+1}^T A \mathbf{d}_k}{\mathbf{d}_k^T A \mathbf{d}_k}$$

$$\mathbf{d}_{k+1} = \mathbf{r}_{k+1} + \beta_k \mathbf{d}_k$$

Método Gradiente Conjugado

Forma alternativa de determinar $\alpha_k = \frac{\mathbf{d}_k^T \mathbf{r}_k}{\mathbf{d}_k^T A \mathbf{d}_k}$

- A partir da equação de $\mathbf{d}_k$:

$$\mathbf{d}_k = \mathbf{r}_k + \beta_{k-1}\mathbf{d}_{k-1}$$

- E $\mathbf{d}_{k-1} \perp \mathbf{r}_k$, podemos reescrever $\mathbf{d}_k^T \mathbf{r}_k$:

$$\mathbf{d}_k^T \mathbf{r}_k = (\mathbf{r}_k + \beta_{k-1}\mathbf{d}_{k-1})^T \mathbf{r}_k = \mathbf{r}_k^T \mathbf{r}_k + \beta_{k-1}\mathbf{d}_{k-1}^T \mathbf{r}_k = \mathbf{r}_k^T \mathbf{r}_k$$

- Assim, α_k pode ser obtido por:

$$\boxed{\alpha_k = \frac{\mathbf{r}_k^T \mathbf{r}_k}{\mathbf{d}_k^T A \mathbf{d}_k}}$$

Método Gradiente Conjugado

Forma alternativa de determinar $\beta_k = -\frac{\mathbf{r}_{k+1}^T A\mathbf{d}_k}{\mathbf{d}_k^T A\mathbf{d}_k}$

- A partir de:

$$\mathbf{r}_{k+1} = \mathbf{r}_k - \alpha_k A\mathbf{d}_k \rightarrow \boxed{A\mathbf{d}_k = \frac{\mathbf{r}_k - \mathbf{r}_{k+1}}{\alpha_k}}$$

$$\alpha_k = \frac{\mathbf{r}_k^T \mathbf{r}_k}{\mathbf{d}_k^T A\mathbf{d}_k} \rightarrow \boxed{\mathbf{d}_k^T A\mathbf{d}_k = \frac{\mathbf{r}_k^T \mathbf{r}_k}{\alpha_k}}$$

- Agora, substituindo em β_k e lembrando que $\mathbf{r}_{k+1} \perp \mathbf{r}_k$:

$$\beta_k = -\frac{-\mathbf{r}_{k+1}^T \frac{\mathbf{r}_k - \mathbf{r}_{k+1}}{\alpha_k}}{\frac{\mathbf{r}_k^T \mathbf{r}_k}{\alpha_k}} = -\frac{-\frac{\mathbf{r}_{k+1}^T \mathbf{r}_{k+1}}{\alpha_k}}{\frac{\mathbf{r}_k^T \mathbf{r}_k}{\alpha_k}} \rightarrow \boxed{\beta_k = \frac{\mathbf{r}_{k+1}^T \mathbf{r}_{k+1}}{\mathbf{r}_k^T \mathbf{r}_k}}$$

Método Gradiente Conjugado: Algoritmo [2]

$\mathbf{x}_0 =$ estimativa inicial

$\mathbf{d}_0 = \mathbf{r}_0 = \mathbf{b} - A\mathbf{x}$

for $k = 0, 1, \cdots, n-1$ **do**

  if $||\mathbf{r}_k||_2 < tol$ **then**

    stop

$$\alpha_k = \frac{\mathbf{r}_k^T \mathbf{r}_k}{\mathbf{d}_k^T A \mathbf{d}_k}$$

$$\mathbf{x}_{k+1} = \mathbf{x}_k + \alpha_k \mathbf{d}_k$$

$$\mathbf{r}_{k+1} = \mathbf{r}_k - \alpha_k A \mathbf{d}_k$$

$$\beta_k = \frac{\mathbf{r}_{k+1}^T \mathbf{r}_{k+1}}{\mathbf{r}_k^T \mathbf{r}_k}$$

$$\mathbf{d}_{k+1} = \mathbf{r}_{k+1} + \beta_k \mathbf{d}_k$$

Método Gradiente Conjugado

Uso de precondicionadores

- Diminuir o número de condicionamento de A
- Acelerar convergência

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

$$M^{-1}A\mathbf{x} = M^{-1}\mathbf{b}$$

  - onde M é o precondicionador

Escolha do precondicionador

- O mais próximo de A possível
  - Para $M^{-1}A$ resultar numa matriz bem condicionada
- Fácil de inverter

Escolhas extremas

- $M = A$, resultaria na identidade
- $M = I$, teria inversa trivial

Método Gradiente Conjugado: Algoritmo [3]

$x_0 =$ estimativa inicial

$r_0 = b - Ax$

$d_0 = z_0 = M^{-1} r_0$

for $k = 0, 1, \cdots, n-1$ **do**

  if $||r_k||_2 < tol$ **then**

    stop

  $$\alpha_k = \frac{r_k^T z_k}{d_k^T A d_k}$$

  $x_{k+1} = x_k + \alpha_k d_k$

  $r_{k+1} = r_k - \alpha_k A d_k$

  $z_{k+1} = M^{-1} r_{k+1}$

  $\beta_k = r_{k+1}^T z_{k+1} / r_k^T z_k$

  $d_{k+1} = z_{k+1} + \beta_k d_k$

Método Gradiente Conjugado

Precondicionador de Jacobi

$$M = D$$

- Bom quando a matriz é estritamente diagonal dominante

Precondicionador simétrico sucessivo com sobrerelaxação (*Symmetrix Successive Over Relaxation* – SSOR)

$$M = (D + wL)D^{-1}(D + wU)$$

- com $w \in [0, 2]$
  - Se $w = 0$, tem-se Jacobi
  - Se $w = 1$, tem-se Gauss-Seidel

Resolução de $M\mathbf{z}_{k+1} = \mathbf{r}_{k+1}$ (ao invés de $z_{k+1} = M^{-1}r_{k+1}$)

- Precondiconador SSOR é um produto de matrizes triangulares

$$(I + wLD^{-1})(D + wU)$$

Exercícios propostos

1. No Método Gradiente Conjugado, qual a importância do uso de pré-condicionadores? O pré-condicionador de Jacobi é dado por $M = D$; em que situações ele é adequado?
2. Sabe-se que o Método Gradiente Conjugado garantidamente converge após n iterações. Como cada iteração tem complexidade $O(n^2)$, devido às operações de multiplicação de matriz por vetor, garante-se que o método chega a solução do sistema em $O(n^3)$. O método de Eliminação de Gauss também tem complexidade $O(n^3)$. Descreva um cenário onde o método Gradiente Conjugado apresenta ordem de complexidade linear ($O(n)$).