Notas em Matemática Aplicada

ISSN 2175-3385
E-ISSN 2236-5915

Volume 93, 2022

Sociedade Brasileira de Matemática Aplicada e Computacional

2022

A Sociedade Brasileira de Matemática Aplicada e Computacional - SBMAC publica, desde as primeiras edições do evento, monografias dos cursos que são ministrados nos CNMAC.

Para a comemoração dos 25 anos da SBMAC, que ocorreu durante o XXVI CNMAC em 2003, foi criada a série **Notas em Matemática Aplicada** para publicar as monografias dos minicursos ministrados nos CNMAC, o que permaneceu até o XXXIII CNMAC em 2010.

A partir de 2011, a série passa a publicar, também, livros nas áreas de interesse da SBMAC. Os autores que submeterem textos à série Notas em Matemática Aplicada devem estar cientes de que poderão ser convidados a ministrarem minicursos nos eventos patrocinados pela SBMAC, em especial nos CNMAC, sobre assunto a que se refere o texto.

O livro deve ser preparado em **Latex (compatível com o Miktex versão 2.9), as figuras em eps** e deve ter entre **80 e 150 páginas**. O texto deve ser redigido de forma clara, acompanhado de uma excelente revisão bibliográfica e de **exercícios de verificação de aprendizagem** ao final de cada capítulo.

Veja todos os títulos publicados nesta série na página
https://proceedings.science/notas-sbmac

Sociedade Brasileira de Matemática Aplicada e Computacional

2022

MÉTODOS ITERATIVOS PARA PROBLEMAS DE QUADRADOS MÍNIMOS LINEARES

Rafael Aleixo de Carvalho
rafael.aleixo@ufsc.br

Felipe Vieira
f.vieira@ufsc.br

Departamento de Matemática
Campus Blumenau
Universidade Federal de Santa Catarina

Sociedade Brasileira de Matemática Aplicada e Computacional

São Carlos - SP, Brasil
2022

Coordenação Editorial: Mateus Bernardes

Coordenação Editorial da Série: Sandra M. C. Malta

Editora: SBMAC

Capa: Matheus Botossi Trindade

Patrocínio: SBMAC

Catalogação elaborada pela Biblioteca do IBILCE/UNESP
Bibliotecária: Maria Luiza Fernandes Jardim Froner

de Carvalho, Rafael Aleixo
Métodos Iterativos para Problemas de Quadrados Mínimos Lineares - São Carlos, SP : SBMAC, 2022, 200 p., 21,5 cm - (Notas em Matematica Aplicada; v. 93)

ISBN 978-65-86388-09-1 e-ISBN 978-65-86388-08-4

1. Quadrados Mínimos 2. Métodos iterativos 3. Álgebra linear numérica
I. de Carvalho, Rafael A. II. Vieira, Felipe III. Título. IV. Série

CDD - 51

À Daniela, Louise e à pequena Júlia.
Dedicamos

Conteúdo

Prefácio

A álgebra linear, em especial a teoria de matrizes, é amplamente aplicada na resolução de problemas nas ciências básicas e nas avançadas, como nas engenharias, na economia e na estatística. E dentre as diversas maneiras de se abordar problemas nessas áreas, uma das mais importantes é a resolução de sistemas lineares e problemas de quadrados mínimos.

O objetivo dessas notas é apresentar as técnicas clássicas e as mais recentes para a resolução iterativa de problemas de quadrados mínimos lineares. Embora a abordagem para a resolução numérica desses tipos de problemas possa ser através de métodos diretos ou iterativos, é este último que contemplamos neste material visto que eles constituem a forma mais adequada de se tratar problemas de grande porte.

Começamos resumindo aspectos muito importantes e cruciais de álgebra linear que serão necessários ao longo dos demais capítulos deste livro. Não teremos todas as demonstrações, afinal o objetivo é apenas relembrar conceitos, mas sempre indicaremos bibliografias com mais detalhes.

No Capítulo 2, o problema de quadrados mínimos lineares é apresentado e, a partir daí, deduzimos as equações normais através de certas interpretações geométricas. Ademais, uma curta discussão sobre a estabilidade dos problemas de quadrados mínimos é apresentada.

O Capítulo 3 é dedicado ao estudo dos chamados métodos iterativos clássicos, tais como o método de Jacobi, Gauss-Seidel e SOR. Além disso, apresentamos os métodos semi-iterativos, em particular o baseado nos polinômios de Chebyshev.

No Capítulo 4, apresentamos os métodos iterativos de Krylov e, para isso, inicialmente apresentamos os subespaços de Krylov. Posteriormente, estudamos os métodos de Arnoldi, de Lanczos, de gradientes conjugados e suas variações para abordar problemas de quadrados mínimos. Ademais, são apresentados os métodos SYMMLQ, MINRES, LSQR e o mais recente LSMR.

Por fim, o Capítulo 5 é dedicado ao estudo de precondicionamento de problemas de quadrados mínimos, com o objetivo de acelerar tais métodos iterativos. Apresentamos os principais aspectos dessa teoria, mas sem um grande aprofundamento.

O livro nasce com o objetivo de fornecer um texto, em português e de fácil acesso, para o estudo de métodos iterativos para problemas de qua-

drados mínimos lineares. Esperamos que seja de grande valor àqueles que trabalham com problemas inversos ou que apenas tenham que "resolver" um sistema linear retangular ou não simétrico. Também, para cada processo estudado, apresentamos um pseudocódigo que sumariza o conteúdo e permite ao leitor implementá-lo, seguindo esse roteiro, para realizar seus próprios testes numéricos. No total são apresentados 30 pseudocódigos.

O público alvo deste livro são os estudantes de final de graduação e início de pós graduação, além daqueles que se interessam pelo tema ou que queiram apenas "refrescar" a mente. Assume-se um conhecimento introdutório de álgebra linear, sobretudo as fatorações LU, Cholesky, QR e SVD, além de conceitos de projeção, projeção ortogonal e complemento ortogonal. Para uma boa revisão desses conceitos sugerimos [68, 102, 147].

Blumenau, 04 de fevereiro de 2022.

Rafael Aleixo de Carvalho e Felipe Vieira

Capítulo 1

Tópicos de Álgebra Linear

Nosso objetivo neste capítulo é apresentar os principais conceitos da Álgebra Linear, que serão necessários para o desenvolvimento da teoria que aborda a resolução de problemas de quadrados mínimos.

1.1 Produto Interno

Para começar, apresentamos um simples conceito que norteia todo o desenvolvimento da teoria: o produto interno. Note que os produtos internos estarão presentes ao longo de todo este capítulo. Sempre consideraremos o espaço vetorial $\mathbb{K}^n$ sobre $\mathbb{K}$, onde este é $\mathbb{R}$ ou $\mathbb{C}$. Iniciemos com a definição do complexo conjugado de um número em $\mathbb{K}$.

Definição 1.1. *Seja $z = a + bi \in \mathbb{C}$, com $a, b, \in \mathbb{R}$. O complexo conjugado de z, denotado por $\overline{z}$, é dado por $\overline{z} = a - bi$. Caso $z \in \mathbb{R}$, perceba que $z = \overline{z}$.*

A definição de produto interno se baseia em quatro propriedades a serem verificadas.

Definição 1.2. *Um produto interno sobre $\mathbb{K}^n$ é uma função $\langle\ ,\ \rangle : \mathbb{K}^n \times \mathbb{K}^n \to \mathbb{K}$ que satisfaz*

P1. $\langle v_1 + v_2, v\rangle = \langle v_1, v\rangle + \langle v_2, v\rangle$, $\forall\, v, v_1, v_2 \in \mathbb{K}^n$.

P2. $\langle \lambda v_1, v_2\rangle = \lambda \langle v_1, v_2\rangle$, $\forall\, \lambda \in \mathbb{K}$ *e* $\forall\, v_1, v_2 \in \mathbb{K}^n$.

P3. $\langle v_1, v_2\rangle = \overline{\langle v_2, v_1\rangle}$, $\forall\, v_1, v_2 \in \mathbb{K}^n$.

P4. $\langle v, v\rangle > 0$ *se* $0 \neq v \in \mathbb{K}^n$.

Dado $\langle\ ,\ \rangle$ um produto interno em $\mathbb{K}^n$ seguem, dessas quatro propriedades:

1. $\langle 0, v\rangle = \langle v, 0\rangle = 0$, para todo $v \in \mathbb{K}^n$.

2. $\langle v, v\rangle = 0$ se, e somente se, $v = 0$.

3. $\langle v, \lambda v_1 + v_2 \rangle = \overline{\lambda} \langle v, v_1 \rangle + \langle v, v_2 \rangle$.

4. De forma geral, recursivamente, obtém-se

$$\left\langle \sum_{i=1}^{n} \alpha_i l_i, \sum_{j=1}^{m} \beta_j m_j \right\rangle = \sum_{i=1}^{n} \sum_{j=1}^{m} \alpha_i \overline{\beta_j} \langle l_i, m_j \rangle , \tag{1.1.1}$$

para todos $l_i, m_j \in \mathbb{K}^n$ e $\alpha_i, \beta_j \in \mathbb{K}$, $i = 1, \ldots, n$ e $j = 1, \ldots, m$.

Propriedade 1.1. *Sejam $\mathbb{B} = \{e_1, \ldots, e_n\}$ uma base ordenada e $\langle\ ,\ \rangle$ um produto interno de $\mathbb{K}^n$. O produto interno $\langle\ ,\ \rangle$ é completamente determinado pelos valores da matriz $G \in \mathbb{K}^{n \times n}$, onde $g_{ij} = \langle e_i, e_j \rangle$.*

Demonstração. Sejam $x, y \in \mathbb{K}^n$, então $x = \sum_{i=1}^{n} \alpha_i e_i$ e $y = \sum_{j=1}^{n} \beta_j e_j$. Assim, pela equação (1.1.1),

$$\langle x, y \rangle = \left\langle \sum_{i=1}^{n} \alpha_i e_i, \sum_{j=1}^{n} \beta_j e_j \right\rangle = \sum_{i,j=1}^{n} \alpha_i \overline{\beta_j} \langle e_i, e_j \rangle = \sum_{i,j=1}^{n} \overline{\beta_j} g_{ij} \alpha_i.$$

Para continuarmos, lembre que se $x \in \mathbb{K}^n$, então x^* é a transposta conjugada de x. Ademais, $[x]_{\mathbb{B}}$ é o vetor com as coordenadas de x em relação à base $\mathbb{B}$. Assim, podemos escrever

$$\langle x, y \rangle = [y]_{\mathbb{B}}^* G [x]_{\mathbb{B}} = Y^* G X,$$

se definirmos $X, Y \in \mathbb{K}^{n \times 1}$ como, respectivamente, as matrizes coluna com as coordenadas de x e y em relação à base ordenada $\mathbb{B}$. $\square$

A matriz G é chamada de matriz do produto interno com relação à base $\mathbb{B}$.

Considere em $\mathbb{K}^n$, $l = (l_1, \ldots, l_n)$ e $m = (m_1, \ldots, m_n)$. Então,

$$\langle l, m \rangle = \sum_{i=1}^{n} l_i \overline{m_i},$$

é chamado de produto interno canônico de $\mathbb{K}^n$. Se $\mathbb{K} = \mathbb{R}$,

$$\langle l, m \rangle = \sum_{i=1}^{n} l_i m_i$$

é também conhecido por produto escalar. Por fim, dados $\alpha_i > 0$, com $i = 1, \ldots, n$ números reais, temos que

$$\langle l, m \rangle = \sum_{i=1}^{n} \alpha_i l_i \overline{m_i}$$

é um produto interno em $\mathbb{K}^n$.

Observação 1.1. *[26, pág. 162] Quando $\mathbb{K} = \mathbb{R}$ a propriedade $P3$ se torna $\langle l, m\rangle = \langle m, l\rangle$, $\forall\, l, m \in \mathbb{R}$. Assim, dizemos que no caso real, o produto interno é simétrico.*
Perceba que se também tivéssemos $\langle l, m\rangle = \langle m, l\rangle$ em $\mathbb{K} = \mathbb{C}$, então

$$\langle il, il\rangle = i\,\langle l, il\rangle = i\,\langle il, l\rangle = i^2\,\langle l, l\rangle = -\,\langle l, l\rangle\,,$$

o que contradiria um dos $\langle l, l\rangle > 0$ e $\langle il, il\rangle > 0$ de $P4$. □

Exemplo 1.1. *Vejamos alguns exemplos de produtos internos.*

1. *O produto interno canônico em $\mathbb{K}^{n\times n}$, o espaço vetorial das matrizes quadradas, é*

$$\langle A, B\rangle = \sum_{i,j=1}^{n} a_{ij}\overline{b_{ij}}.$$

   Introduzindo a transposta conjugada B^* da matriz B, onde $b^*_{ij} = \overline{b_{ji}}$, podemos expressar o produto interno acima como, $\langle A, B\rangle = \mathrm{tr}(AB^*) = \mathrm{tr}(B^*A)$.

2. *Seja $\mathbb{K}^{n\times 1}$, o espaço das matrizes colunas em $\mathbb{K}$. Se $Q \in \mathbb{K}^{n\times n}$ é uma matriz inversível, então para $X, Y \in \mathbb{K}^{n\times 1}$, $\langle X, Y\rangle = Y^*Q^*QX$ é um produto interno.*

Definição 1.3. *Um espaço que possui um produto interno sobre $\mathbb{R}$ é dito um espaço euclidiano. Se esse produto interno for sobre $\mathbb{C}$, dizemos que $\mathbb{K}^n$ é um espaço hermitiano.*

A proposição abaixo mostra como construir um produto interno em um espaço vetorial a partir de um produto interno definido em um outro espaço vetorial.

Propriedade 1.2. *[26, pág. 163] Seja $T : \mathbb{K}^n \to \mathbb{K}^m$ uma transformação linear injetora (não singular) e $\langle\ ,\ \rangle$ um produto interno em $\mathbb{K}^m$. Então,*

$$\langle x, y\rangle_T := \langle T(x), T(y)\rangle\,,\ \forall\, x, y \in \mathbb{K}^n$$

também é um produto interno em $\mathbb{K}^n$.

Demonstração. Demonstremos as quatro propriedades de produto interno. Sejam $x_1, x_2, x, y \in \mathbb{K}^n$ e $\lambda \in \mathbb{K}$:

P1.

$$\begin{aligned}\langle x_1 + x_2, y\rangle_T &= \langle T(x_1 + x_2), T(y)\rangle = \langle T(x_1) + T(x_2), T(y)\rangle\\ &= \langle T(x_1), T(y)\rangle + \langle T(x_2), T(y)\rangle\\ &= \langle x_1, y\rangle_T + \langle x_2, y\rangle_T\,.\end{aligned}$$

P2.

$$\begin{aligned}\langle \lambda x, y\rangle_T &= \langle T(\lambda x), T(y)\rangle = \langle \lambda T(x), T(y)\rangle \\ &= \lambda \langle T(x), T(y)\rangle \\ &= \lambda \langle x, y\rangle_T .\end{aligned}$$

P3. $\langle x, y\rangle_T = \langle T(x), T(y)\rangle = \overline{\langle T(y), T(x)\rangle} = \overline{\langle y, x\rangle}_T$.

P4. Suponha que $0 \neq x$. Como T é injetora $T(x) \neq 0$, logo

$$\langle x, x\rangle_T = \langle T(x), T(x)\rangle > 0.$$

□

Dado M um subespaço vetorial de $\mathbb{K}^n$, como $\iota : M \to \mathbb{K}^n$, $\iota(x) = x$, é uma transformação linear e injetora, chamada de inclusão natural, temos que qualquer $\langle\ ,\ \rangle$ de $\mathbb{K}^n$ restrito à M será um produto interno em M.

1.2 Normas de Vetores e Matrizes

Nessa seção faremos uma revisão dos conceitos de normas de vetor e de matriz, além de introduzir um conceito importante denominado o número de condição de uma matriz.

Definição 1.4. *Uma norma em $\mathbb{K}^n$ é uma função real $\|\cdot\|$ que satisfaz as seguintes propriedades:*

N1. $\|x\| \geqslant 0$ *para todo* $x \in \mathbb{K}^n$*, e* $\|x\| = 0$ *somente se* $x = 0$.

N2. $\|\lambda x\| = |\lambda| \cdot \|x\|$*, para todos* $x \in \mathbb{K}^n$ *e* $\lambda \in \mathbb{K}$.

N3. $\|x + y\| \leqslant \|x\| + \|y\|$*, para todos* $x, y \in \mathbb{K}^n$.

Um vetor $x \in \mathbb{K}^n$ é dito unitário com respeito à norma $\|\cdot\|$, se $\|x\| = 1$. Existe uma infinidade de funções reais que satisfazem os axiomas de norma e, provavelmente, a mais simples é a função módulo sobre $\mathbb{R}$. Para facilitar a diferenciação das normas mais úteis, utilizaremos um subscrito ao lado de suas notações. Uma classe importante delas na álgebra linear aplicada é a das p-normas, onde p é um inteiro positivo:

$$\|x\|_p = \left(|x_1|^p + \cdots + |x_n|^p\right)^{\frac{1}{p}}, \quad p \geqslant 1.$$

Dentre as p-normas, as mais importantes são as normas 1, 2 e, com um abuso de notação, ∞, que são dadas por

$$\begin{aligned}\|x\|_1 &= |x_1| + \cdots + |x_n|, \\ \|x\|_2 &= \left(|x_1|^2 + \cdots + |x_n|^2\right)^{\frac{1}{2}} = (x^* x)^{\frac{1}{2}}, \\ \|x\|_\infty &= \max_{1 \leqslant i \leqslant n} |x_i|.\end{aligned}$$

Outra importante classe de normas é a formada pelas chamadas normas elípticas, que são dadas por

$$\|x\| = (x^* B x)^{\frac{1}{2}},$$

onde $B \in \mathbb{K}^{n\times n}$ é uma matriz simétrica definida positiva.

Enunciaremos três importantes desigualdades que utilizam normas de vetores em espaços euclidianos. Suas demonstrações são deixadas a cargo do leitor e são facilmente encontradas em livros de álgebra linear ou análise funcional.

Teorema 1.1. *Dados $x, y \in \mathbb{R}^n$ e $\alpha, \beta \in \mathbb{R}$, valem as seguintes desigualdades.*

1. *Desigualdade de Young [101, pág. 2]:* $|\alpha\beta| \leqslant \dfrac{\epsilon}{2}|\alpha|^2 + \dfrac{1}{2\epsilon}|\beta|^2$, $\epsilon \in (0, \infty)$.

2. *Desigualdade de Hölder [58, pág. 69]:* $|x^T y| \leqslant \|x\|_p \|y\|_q$, $\dfrac{1}{p} + \dfrac{1}{q} = 1$.

3. *Desigualdade de Cauchy-Schwarz [58, pág. 69]:* $|x^T y| \leqslant \|x\|_2 \|y\|_2$.

Outro conceito referente a normas que é bastante útil é o de normas equivalentes.

Definição 1.5. *Duas normas $\|\cdot\|_\alpha$ e $\|\cdot\|_\beta$ definidas em um mesmo espaço vetorial são ditas equivalentes se existem c_1, c_2 escalares positivos tais que*

$$c_1 \|x\|_\alpha \leqslant \|x\|_\beta \leqslant c_2 \|x\|_\alpha,$$

para todos vetores x.

Em particular, para o espaço euclidiano $\mathbb{K}^n$, todas as normas são equivalentes [103, pág. 18]. Por exemplo,

$$\begin{aligned} \|x\|_2 &\leqslant \|x\|_1 \leqslant \sqrt{n}\,\|x\|_2, \\ \|x\|_\infty &\leqslant \|x\|_2 \leqslant \sqrt{n}\,\|x\|_\infty, \\ \|x\|_\infty &\leqslant \|x\|_1 \leqslant n\,\|x\|_\infty. \end{aligned}$$

Uma propriedade das 2-normas é que elas são preservadas sob transformações ortogonais, isto é, se $U \in \mathbb{K}^{n\times n}$ é ortogonal, então $\|Ux\|_2 = \|x\|_2$.

Teorema 1.2. *Seja $Q \in \mathbb{K}^{n\times n}$ uma matriz ortogonal e $x \in \mathbb{K}^n$. Então*

$$\|Qx\|_2^2 = \|x\|_2^2.$$

Demonstração. $\|Qx\|_2^2 = (Qx)^*(Qx) = x^* Q^* Q x = x^* x = \|x\|_2^2$. □

Uma maneira rápida de se obter normas é através dos produtos internos, como vemos na próxima definição.

Definição 1.6. *Seja L um $\mathbb{K}$-espaço vetorial com produto interno $\langle\ ,\ \rangle$. Se, para $l \in L$, definirmos*

$$\|l\| = \sqrt{\langle l, l\rangle},$$

então obtemos uma norma.

Embora já estejamos denominando essa $\|.\|$ de norma, precisamos demonstrar que essa definição satisfaz as três propriedades das normas. Note que a mera junção dessa definição com a Definição 1.2 implica que valem

N1. $\|l\| \geqslant 0$, $\forall\, l \in L$ e $\|l\| = 0 \Leftrightarrow l = 0$.

N2. $\|\alpha l\| = |\alpha|\, \|l\|$, $\forall\, \alpha \in \mathbb{K}$ e $\forall\, l \in L$.

Para demonstrar $N3$, que é chamada de desigualdade triangular, precisamos de alguns resultados auxiliares que vemos a seguir.

Observação 1.2. *Dado L um $\mathbb{K}$-espaço vetorial com produto interno $\langle\ ,\ \rangle$, a norma proveniente deste satisfaz as seguintes identidades.*

1. *Para todos $l_1, l_2 \in L$,*

$$\|l_1 \pm l_2\|^2 = \|l_1\|^2 \pm 2\,\mathrm{Re}\,\langle l_1, l_2\rangle + \|l_2\|^2 .$$

Com efeito,

$$\begin{aligned}
\|l_1 + l_2\|^2 &= \langle l_1 + l_2, l_1 + l_2\rangle \overset{P1)}{=} \langle l_1, l_1 + l_2\rangle + \langle l_2, l_1 + l_2\rangle \\
&\overset{P3)}{=} \overline{\langle l_1 + l_2, l_1\rangle} + \overline{\langle l_1 + l_2, l_2\rangle} \\
&\overset{P1)}{=} \overline{\langle l_1, l_1\rangle} + \overline{\langle l_2, l_1\rangle} + \overline{\langle l_1, l_2\rangle} + \overline{\langle l_2, l_2\rangle} \\
&= \|l_1\|^2 + \langle l_1, l_2\rangle + \overline{\langle l_1, l_2\rangle} + \|l_2\|^2 \\
&= \|l_1\|^2 + 2\,\mathrm{Re}\,\langle l_1, l_2\rangle + \|l_2\|^2 .
\end{aligned}$$

Analogamente, se demonstra a identidade para $\|l_1 - l_2\|^2$.

2. *Se $\mathbb{K} = \mathbb{R}$, temos*

$$\langle l_1, l_2\rangle = \frac{1}{4}\|l_1 + l_2\|^2 - \frac{1}{4}\|l_1 - l_2\|^2 .$$

De fato,

$$\begin{aligned}
&\|l_1 + l_2\|^2 - \|l_1 - l_2\|^2 \\
&\quad = \left(\|l_1\|^2 + 2\langle l_1, l_2\rangle + \|l_2\|^2\right) - \left(\|l_1\|^2 - 2\langle l_1, l_2\rangle + \|l_2\|^2\right) \\
&\quad = 4\langle l_1, l_2\rangle .
\end{aligned}$$

3. Se $\mathbb{K} = \mathbb{C}$, temos

$$\langle l_1, l_2 \rangle = \frac{1}{4} \|l_1 + l_2\|^2 - \frac{1}{4} \|l_1 - l_2\|^2 + \frac{i}{4} \|l_1 + il_2\|^2 - \frac{i}{4} \|l_1 - il_2\|^2 .$$

A demonstração desse fato é deixada como exercício.

□

As duas últimas identidades são chamadas de identidades de polarização. Ademais, em um espaço vetorial com produto interno vale a chamada desigualdade de Cauchy-Schwarz, que aparece em muitas aplicações.

Teorema 1.3. *(Desigualdade de Cauchy-Schwarz) Seja L um $\mathbb{K}$-espaço vetorial com produto interno. Então a norma obtida a partir deste produto interno satisfaz*

$$|\langle l_1, l_2 \rangle| \leqslant \|l_1\| \cdot \|l_2\| , \ \ \forall\, l_1, l_2 \in L.$$

A igualdade vale se, e somente se, o conjunto $\{l_1, l_2\}$ for linearmente dependente.

Demonstração. Se $l_1 = 0$, vale a desigualdade com $0 \leqslant 0$. Considere, então, $l_1 \neq 0$ e defina

$$m := l_2 - \frac{\overline{\langle l_1, l_2 \rangle}}{\|l_1\|^2} l_1.$$

Assim,

$$\langle l_1, m \rangle = \langle l_1, l_2 \rangle - \frac{\langle l_1, l_2 \rangle}{\|l_1\|^2} \langle l_1, l_1 \rangle = 0.$$

Portanto,

$$\begin{aligned}
0 \leqslant \|m\|^2 &= \left\langle l_2 - \frac{\overline{\langle l_1, l_2 \rangle}}{\|l_1\|^2} l_1, m \right\rangle \\
&= \langle l_2, m \rangle - \frac{\overline{\langle l_1, l_2 \rangle}}{\|l_1\|^2} \cancelto{0}{\langle l_1, m \rangle} \\
&= \left\langle l_2, l_2 - \frac{\overline{\langle l_1, l_2 \rangle}}{\|l_1\|^2} l_1 \right\rangle \\
&= \langle l_2, l_2 \rangle - \frac{\langle l_1, l_2 \rangle}{\|l_1\|^2} \overline{\langle l_1, l_2 \rangle} \\
&= \|l_2\|^2 - \frac{|\langle l_1, l_2 \rangle|^2}{\|l_1\|^2}.
\end{aligned}$$

Logo $|\langle l_1, l_2 \rangle|^2 \leqslant \|l_1\|^2 \cdot \|l_2\|^2$. Perceba que a demonstração garante que, para $l_1 \neq 0$, temos $|\langle l_1, l_2 \rangle| < \|l_1\| \cdot \|l_2\|$, a menos que $l_2 = \dfrac{\overline{\langle l_1, l_2 \rangle}}{\|l_1\|^2} l_1$. □

Exemplo 1.2. *[67, pág. 278] Considerando alguns produtos internos específicos, temos as seguintes aplicações da desigualdade de Cauchy-Schwarz.*

1. $|\mathrm{tr}(AB^*)| \leqslant \sqrt{\mathrm{tr}(AA^*) \cdot \mathrm{tr}(BB^*)}$.

2. $\left|\int_0^1 f(x)\overline{g(x)}\,dx\right| \leqslant \sqrt{\left(\int_0^1 |f(x)|^2\,dx\right) \cdot \left(\int_0^1 |g(x)|^2\,dx\right)}$.

Assim, finalmente podemos demonstrar que a norma originada de um produto interno satisfaz $N3$.

Corolário 1.1. *(Desigualdade Triangular) Seja L um $\mathbb{K}$-espaço vetorial com produto interno. Então a norma obtida deste satisfaz*

$$\|l_1 + l_2\| \leqslant \|l_1\| + \|l_2\|\,, \quad \forall\, l_1, l_2 \in L.$$

Demonstração. Note que, para todo $z \in \mathbb{C}$, $\mathrm{Re}\, z \leqslant |z|$. Portanto,

$$\begin{aligned}
\|l_1 + l_2\|^2 &= \|l_1\|^2 + 2\,\mathrm{Re}\,\langle l_1, l_2\rangle + \|l_2\|^2 \\
&\leqslant \|l_1\|^2 + 2|\,\langle l_1, l_2\rangle\,| + \|l_2\|^2 \\
&\leqslant \|l_1\|^2 + 2\,\|l_1\| \cdot \|l_2\| + \|l_2\|^2 \\
&= (\|l_1\| + \|l_2\|)^2\,.
\end{aligned}$$

□

Assim, a partir de agora, quando estivermos trabalhando com um espaço que possua um produto interno, sempre consideraremos a norma que é obtida a partir deste. Casos diferentes, serão explicitados.

Considere $\mathbb{R}^n$ sobre $\mathbb{R}$ com o produto interno canônico. Para todos $x.y \in \mathbb{R}^n$, com $x = (x_1, \ldots, x_n)$ e $y = (y_1, \ldots, y_n)$, dizemos que

$$\|x - y\| = \sqrt{\sum_{i=1}^{n} (x_i - y_i)^2}$$

indica a distância entre x e y. Isto é condizente com a geometria: basta construir um n-cubo em $\mathbb{R}^n$ com ambos os pontos em vértices opostos. Para calcular sua distância, basta fazer uma série de triângulos retângulos e aplicar o Teorema de Pitágoras, que obteremos a mesma expressão.

Em particular, se $y = 0$, então $\|x\|$ é a distância de x até a origem. Assim, fica explícito o motivo de norma de um vetor ser escrita, muitas vezes, como o "tamanho" do vetor em questão.

Exemplo 1.3. *Se em $\mathbb{R}^2$ sobre $\mathbb{R}$ consideramos o seguinte produto interno*

$$\langle (x_1, y_1), (x_2, y_2)\rangle = 3x_1x_2 + 16y_1y_2,$$

temos que $\|(1,0)\| = \sqrt{3}$ e $\|(0,1)\| = 4$.

Os métodos iterativos para problemas de quadrados mínimos procuram encontrar soluções aproximadas para os problemas propostos. Assim, é conveniente definirmos alguns conceitos úteis.

Definição 1.7. *Suponha que $\tilde{x} \in \mathbb{K}^n$ seja uma aproximação para $x \in \mathbb{K}^n$. Para uma dada norma de vetores $\|\cdot\|$ definimos o erro absoluto de $\tilde{x}$ por*

$$e_{abs} = \|\tilde{x} - x\| .$$

No caso em que $x \neq 0$, definimos o erro relativo de $\tilde{x}$ como

$$e_{rel} = \frac{\|\tilde{x} - x\|}{\|x\|}.$$

No caso da norma ∞, o erro relativo diz a quantidade de dígitos significativos da aproximação $\tilde{x}$. Ou seja, $e_{\text{rel}} \approx 10^{-p}$ significa que a maior componente de $\tilde{x}$ tem aproximadamente p dígitos significantes corretos.

Definição 1.8. *Dizemos que a sequência $\{x^{(k)}\}$ de vetores de $\mathbb{K}^n$ converge para $x \in \mathbb{K}^n$, se*

$$\lim_{k\to\infty} \left\| x^{(k)} - x \right\| = 0.$$

Note que, como em $\mathbb{K}^n$ todas as normas são equivalentes, então basta demonstrar que o limite acima é válido para alguma norma, que será válido para qualquer norma.

Utilizando as normas de vetores, podemos definir normas para matrizes, já que $\mathbb{K}^{m\times n}$ é isomorfo a $\mathbb{K}^{mn}$. Assim, como para vetores, utilizaremos a notação de dupla barra para a representação da norma de matrizes, isto é, denotaremos por $\|A\|$ a norma da matriz $A \in \mathbb{K}^{m\times n}$. Ademais, todas as normas de matrizes são equivalentes e, novamente, utilizaremos o subscrito para diferenciar normas distintas.

Uma norma de matriz importante e utilizada com muita frequência em álgebra linear aplicada é a chamada norma de Frobenius, dada por

$$\|A\|_F = \sqrt{\sum_{i=1}^{m}\sum_{j=1}^{n} |a_{ij}|^2} = \sqrt{\operatorname{tr}(A^T A)},$$

onde $A \in \mathbb{K}^{m\times n}$.

Também são importantes as normas 1 e ∞ de matrizes, dadas por

$$\|A\|_1 = \max_{1\leqslant j\leqslant n} \sum_{i=1}^{m} |a_{ij}| \quad \text{e} \quad \|A\|_\infty = \max_{1\leqslant i\leqslant m} \sum_{j=1}^{n} |a_{ij}|,$$

ou seja, a norma 1 de uma matriz A é dada pelo máximo das somas dos valores absolutos das colunas de A, enquanto a norma ∞ é dada pelo máximo das somas dos valores absolutos das linhas de A.

E vale destacar as p-normas, $1 < p < \infty$,

$$\|A\|_p = \sup_{x \neq 0} \frac{\|Ax\|_p}{\|x\|_p} = \max_{\|x\|_p = 1} \|Ax\|_p,$$

em que a segunda igualdade é válida, pois $\|\cdot\|$ é uma função contínua e $S = \{x \in \mathbb{R}^n \mid \|x\| = 1\}$ é um conjunto compacto e, portanto, pelo Teorema de Weierstrass $\|\cdot\|$ é limitada e atinge seus extremos [90].

Uma descrição mais detalhada dessas p-normas é bastante complicada, porém estimativas podem ser feitas utilizando uma generalização do método da potência [65]. Já, a 2-norma goza de uma propriedade que permite seu cálculo de forma razoavelmente fácil. A ideia aqui apresentada será a base para a determinação dos valores singulares na decomposição SVD.

Teorema 1.4. *[58, pág. 73] Se $A \in \mathbb{R}^{m \times n}$, então existe um vetor x, unitário na 2-norma, tal que $A^T Ax = \mu^2 x$, onde $\mu = \|A\|_2$.*

A demonstração é deixada a cargo do leitor e este teorema implica que $\|A\|_2^2 = \lambda_{max}(A^T A)$ [101], o maior autovalor de $A^T A$. Por isso, a 2-norma de uma matriz $A \in \mathbb{K}^{m \times n}$ também é conhecida como a norma espectral de A.

Como consequência temos que

$$\|A\|_2^2 \leqslant \|A\|_1 \|A\|_\infty .$$

Com efeito, para $x \neq 0$, temos $A^T Ax = \mu^2 x$, onde $\mu = \|A\|_2$ e, portanto,

$$\mu^2 \|x\|_1 = \left\|A^T Ax\right\|_1 \leqslant \left\|A^T\right\|_1 \|A\|_1 \|x\|_1 = \|A\|_\infty \|A\|_1 \|x\|_1 .$$

Logo, $\mu^2 \leqslant \|A\|_\infty \|A\|_1$.

Uma forma de se obter normas de matrizes é através de normas de vetores.

Definição 1.9. *Seja $A \in \mathbb{K}^{m \times n}$. Dizemos que a norma de matriz $\|\cdot\|$ é consistente com as normas de vetor $\|\cdot\|_\alpha$ do $\mathbb{K}^n$ e $\|\cdot\|_\beta$ do $\mathbb{K}^m$ se*

$$\|A\| = \sup_{x \neq 0} \frac{\|Ax\|_\beta}{\|x\|_\alpha} = \max_{\|x\|_\alpha = 1} \|Ax\|_\beta.$$

A segunda igualdade é válida, novamente, pelo Teorema de Weierstrass. As p-normas em $\mathbb{K}^{m \times n}$ são normas consistentes com as p-normas de vetores

$$\|A\|_p = \sup_{x \neq 0} \frac{\|Ax\|_p}{\|x\|_p} = \max_{\|x\|_p = 1} \|Ax\|_p.$$

Segue imediatamente da definição de norma de matriz consistente que, abandonando subscritos, as seguintes propriedades são válidas

1. $\|Ax\| \leqslant \|A\| \|x\|$, para todos $x \in \mathbb{K}^n$ e $A \in \mathbb{K}^{m \times n}$.

2. $\|AB\| \leqslant \|A\| \, \|B\|$, para todos $A \in \mathbb{K}^{m\times q}$ e $B \in \mathbb{K}^{q\times n}$.

Por fim, observe que nem toda norma de matriz é consistente. Por exemplo, considere $\|A\| = \max |a_{ij}|$ e

$$A = B = \begin{bmatrix} 1 & 1 \\ 1 & 1 \end{bmatrix}.$$

Nesse caso, $\|AB\| > \|A\| \, \|B\|$.

Da mesma forma que para normas de vetores, temos que a sequência $\{A^{(k)}\}$ de matrizes de $\mathbb{K}^{m\times n}$ converge para $K \in \mathbb{R}^{m\times n}$, se

$$\lim_{k\to\infty} \left\| A^{(k)} - A \right\| = 0.$$

Por conta da equivalência das normas de matrizes, a escolha da norma acima é irrelevante para garantir a convergência.

1.3 Bases Ortogonais e Ortonormais

Além de produzirem normas, os produtos internos nos ajudam a refinar o conceito de base de um espaço.

Definição 1.10. *Seja L um $\mathbb{K}$-espaço vetorial com produto interno $\langle\ ,\ \rangle$ e sejam $l, m \in L$. Dizemos que os vetores são ortogonais ou perpendiculares se $\langle l, m\rangle = 0$, e um subconjunto M de L é chamado de ortogonal se os seus elementos são dois a dois ortogonais. Ademais, definimos que M é um conjunto ortonormal se for um conjunto ortogonal e se $\|m\| = m$, $\forall\, m \in M$.*

Usaremos a notação $l \perp m$ para indicar que os vetores l e m são ortogonais. O vetor nulo é ortogonal a qualquer elemento $l \in L$, pois $\langle l, 0\rangle = \langle 0, l\rangle = 0$, $\forall\, l \in L$. Ademais, o vetor nulo é o único vetor com essa propriedade. Um conjunto ortonormal pode ser visto como um conjunto onde seus elementos são mutualmente perpendiculares, cada um com tamanho 1.

Exemplo 1.4. *1. As bases canônicas de $\mathbb{R}^n$, $\mathbb{C}^n$, $\mathbb{R}^{n\times n}$ e $\mathbb{C}^{n\times n}$ com os produtos internos canônicos são conjuntos ortonormais.*

2. Em $\mathbb{R}^2$, o vetor (a, b) é ortogonal a $(-b, a)$ com respeito ao produto interno canônico, pois

$$\langle (a, b), (-b, a)\rangle = -ab + ba = 0.$$

Porém, se considerarmos em $\mathbb{R}^2$ o produto interno

$$\langle x, y\rangle = x_1y_1 - x_2y_1 - x_1y_2 + 4x_2y_2,$$

os vetores (a, b) e $(-b, a)$ são ortogonais se, e somente se, vale a igualdade $b = \frac{1}{2}(-3 \pm \sqrt{13})a$.

3. *Sejam* $L = \mathbb{C}^{n\times n}$ *e* E^{pq} *a matriz cuja única entrada não nula é* 1 *e está na posição* (p,q) *da matriz. Então, o conjunto das matrizes* E^{pq} *com respeito ao produto interno* $\langle A, B\rangle = \operatorname{tr}(AB^*)$ *é ortonormal, pois*

$$\langle E^{pq}, E^{rs}\rangle = \operatorname{tr}(E^{pq}E^{sr}) = \delta_{qs}\operatorname{tr}(E^{pr}) = \delta_{qs}\delta_{pr},$$

onde $\delta_{pq} = 1$ *se* $p = q$ *e* $\delta_{pq} = 0$ *se* $p \neq q$.

4. *Seja* $L = \mathscr{C}([0, 2\pi], \mathbb{R})$*, o espaço das funções contínuas de* $[0, 2\pi]$ *em* $\mathbb{R}$*, com o produto interno canônico. Então,*

$$M = \{f_n \in L \mid f_n(t) = \cos nt,\ n \in \mathbb{N}\}$$

é um conjunto ortogonal pois, utilizando integração por partes,

$$\langle f_n, f_m\rangle = \int_0^{2\pi} \cos nt \cos mt\, dt = \begin{cases} 0, & \text{se } m \neq n, \\ \pi, & \text{se } m = n \neq 0, \\ 2\pi, & \text{se } m = n = 0. \end{cases}$$

Assim, $e_0(t) = \dfrac{1}{\sqrt{2\pi}}$ *e* $e_n(t) = \dfrac{\cos nt}{\sqrt{\pi}}$ *para* $n = 1, 2, \ldots$ *formam um subconjunto ortonormal infinito de* L.

Exemplo 1.5. *Vejamos como encontrar um vetor não nulo* w *que seja ortogonal a* $l = (1, 0, -1)$ *e* $m = (1, 2, 3)$ *em* $\mathbb{R}^3$ *com o produto interno canônico. Seja* $w = (x, y, z)$*. Como queremos que* w *seja ortogonal a* l *e* m *simultaneamente, devemos impor que* $\langle l, w\rangle = 0$ *e* $\langle m, w\rangle = 0$*. Essas condições geram o seguinte sistema linear,*

$$\begin{cases} x - z &= 0, \\ x + 2y + 3z &= 0. \end{cases}$$

Considere z *a variável livre do sistema. Tomando* $z = 1$*, obtemos* $x = 1$ *e* $y = -2$*. Portanto,* $w = (1, -2, 1)$ *é uma solução possível.*

Conjuntos ortogonais são interessantes pois gozam de certas propriedades úteis em álgebra linear. Primeiramente, definamos o conceito de subespaço gerado por um subconjunto de um espaço vetorial.

Definição 1.11. *Seja* L *um* $\mathbb{K}$*-espaço vetorial e* M *um subconjunto de* L*. O subespaço gerado por* M *e denotado por* $\operatorname{span} M$ *é o espaço das combinações lineares (finitas) dos elementos de* M*. Pode ser demonstrado que* $\operatorname{span} M$ *é um subespaço vetorial de* L.

Propriedade 1.3. *[26, pág. 174] Seja* L *um* $\mathbb{K}$*-espaço vetorial com produto interno* $\langle\ ,\ \rangle$ *e seja* M *um subconjunto ortogonal de* L *formado por vetores não nulos. Então:*

1. *Se* $l \in \operatorname{span} M$*, então*

$$l = \sum_{i=1}^{n} \frac{\langle l, l_i\rangle}{\|l_i\|^2} l_i, \quad \text{com } l_i \in M.$$

2. M é linearmente independente.

Demonstração. 1. Seja $l \in span\, M$. Então, existem $a_i \in \mathbb{K}$ e $l_i \in M$, $i = 1, \ldots, n$, tais que

$$\langle l, l_j \rangle = \left\langle \sum_{i=1}^{n} a_i l_i, l_j \right\rangle = \sum_{i=1}^{n} a_i \langle l_i, l_j \rangle = a_j \langle l_j, l_j \rangle .$$

Portanto, $a_j = \dfrac{\langle l, l_j \rangle}{\|l_j\|^2}$, $j = 1, \ldots, n$ o que nos leva a $l = \displaystyle\sum_{i=1}^{n} \frac{\langle l, l_i \rangle}{\|l_i\|^2} l_i$.

2. Utilizando o item anterior, se $l \in span\, M$, então $l = 0$ implica

$$a_i = \frac{\langle 0, l_i \rangle}{\|l_i\|^2} = 0.$$

Logo, M é linearmente independente.

□

Corolário 1.2. *Seja L um $\mathbb{K}$-espaço vetorial com produto interno $\langle\ ,\ \rangle$ e seja $\{l_1, \ldots, l_n\}$ uma base ortonormal de L. Então, para todo $l \in L$, temos*

$$l = \sum_{i=1}^{n} \langle l, l_i \rangle l_i.$$

Definição 1.12. *Sejam L um $\mathbb{K}$-espaço vetorial de dimensão finita com produto interno $\langle\ ,\ \rangle$, $\mathbb{B} = \{l_1, \ldots, l_n\}$ um subconjunto ortogonal de L formado por vetores não nulos e $l \in span\{l_1, \ldots, l_n\}$. Os escalares*

$$a_i = \frac{\langle l, l_i \rangle}{\|l_i\|^2}, \quad i = 1, \ldots, n$$

são denominados os coeficientes de Fourier do vetor l em relação à base $\mathbb{B}$ de $span\, \mathbb{B}$.

1.4 Processo de Gram-Schmidt

Nessa seção veremos o processo de ortogonalização de Gram-Schmidt que permite, a partir de um subconjunto finito e linearmente independente, construir um conjunto ortogonal de forma que o subespaço gerado por ambos seja o mesmo.

O processo de ortogonalização de Gram-Schmidt foi essencialmente descrito por Jørgen Pedersen Gram[1] [59] e Erhard Schmidt[2] [129, pág. 472] de forma independente. Porém, o processo descrito por Gram e Schmidt

[1] `https://mathshistory.st-andrews.ac.uk/Biographies/Gram/`
[2] `https://mathshistory.st-andrews.ac.uk/Biographies/Schmidt/`

já havia aparecido em trabalhos de Pierre-Simon Laplace. Nas definições de hoje, o método de Gram-Schmidt é o método desenvolvido por Schmidt, já o método desenvolvido por Gram convencionou-se chamar de método de Gram-Schmidt modificado. A primeira citação do "processo de ortogonalização de Gram-Schmidt" é feita em 1935 [159, pág. 57], e uma excelente referência que trata da história e das propriedades desse método é [88].

Seja L um $\mathbb{K}$-espaço vetorial e $\mathbb{B} = \{l_1, \ldots, l_n\} \subseteq L$ um conjunto linearmente independente. Vamos construir, indutivamente, $\mathbb{B}' = \{m_1, \ldots, m_n\} \subseteq L$ que seja ortogonal e tal que, $span\,\mathbb{B} = span\,\mathbb{B}'$.

Tome $m_1 = l_1$ e suponha que $m_1, \ldots, m_k$, $1 \leqslant k < n$ já tenham sido escolhidos. Assim para cada j, $\{m_1, \ldots, m_j\}$, $1 \leqslant j \leqslant k$ é um conjunto ortogonal. Definimos

$$m_{k+1} = l_{k+1} - \sum_{i=1}^{k} \frac{\langle l_{k+1}, m_i \rangle}{\|m_i\|^2} m_i. \tag{1.4.2}$$

Assim, para cada $1 \leqslant j \leqslant k$,

$$\begin{aligned} \langle m_{k+1}, m_j \rangle &= \langle l_{k+1}, m_j \rangle - \sum_{i=1}^{k} \frac{\langle l_{k+1}, m_i \rangle}{\|m_i\|^2} \langle m_i, m_j \rangle \\ &= \langle l_{k+1}, m_j \rangle - \langle l_{k+1}, m_j \rangle = 0. \end{aligned}$$

Logo, por construção, $\mathbb{B}'$ é um conjunto ortogonal com n elementos no subespaço vetorial $span\,\mathbb{B}$, portanto é uma base de $span\,\mathbb{B}$. Logo, $span\,\mathbb{B} = span\,\mathbb{B}'$.

Observação 1.3. *[67, pág. 281] O processo de ortogonalização de Gram-Schmidt pode ser usado, apesar de não ser muito prático, para testar se um conjunto dado em um espaço vetorial com produto interno é linearmente dependente. Com efeito, suponha que $\mathbb{B} = \{l_1, \ldots, l_n\}$ seja um conjunto linearmente dependente e assuma $l_1 \neq 0$, para excluir o caso trivial. Seja k o maior inteiro para o qual $l_1, \ldots, l_k$ seja linearmente independente, nem que precisemos reordenar os vetores l_j. Portanto $1 \leqslant k < n$ e considere $\{m_1, \ldots, m_k\}$ os vetores obtidos de $\{l_1, \ldots, l_k\}$ pelo processo de ortogonalização de Gram-Schmidt. Então,*

$$m_{k+1} = l_{k+1} - \sum_{i=1}^{k} \frac{\langle l_{k+1}, m_i \rangle}{\|m_i\|^2} m_i$$

é necessariamente 0, pois $m_{k+1} \in span\,\mathbb{B}$ e é ortogonal a cada um desses vetores. Por outro lado, se $m_1, \ldots, m_k$ são diferentes de 0 e $m_{k+1} = 0$, então $\{l_1, \ldots, l_{k+1}\}$ é linearmente dependente. □

Apenas para exemplificar o processo de ortogonalização de Gram-Schmidt, temos para $n = 5$,

- $m_1 = l_1$.
- $m_2 = l_2 - \dfrac{\langle l_2, m_1\rangle}{\|m_1\|^2} m_1$.
- $m_3 = l_3 - \dfrac{\langle l_3, m_1\rangle}{\|m_1\|^2} m_1 - \dfrac{\langle l_3, m_2\rangle}{\|m_2\|^2} m_2$.
- $m_4 = l_4 - \dfrac{\langle l_4, m_1\rangle}{\|m_1\|^2} m_1 - \dfrac{\langle l_4, m_2\rangle}{\|m_2\|^2} m_2 - \dfrac{\langle l_4, m_3\rangle}{\|m_3\|^2} m_3$.
- $m_5 = l_5 - \dfrac{\langle l_5, m_1\rangle}{\|m_1\|^2} m_1 - \dfrac{\langle l_5, m_2\rangle}{\|m_2\|^2} m_2 - \dfrac{\langle l_5, m_3\rangle}{\|m_3\|^2} m_3 - \dfrac{\langle l_5, m_4\rangle}{\|m_4\|^2} m_4$.

Vejamos uma representação geométrica do cálculo de m_2 e m_3.

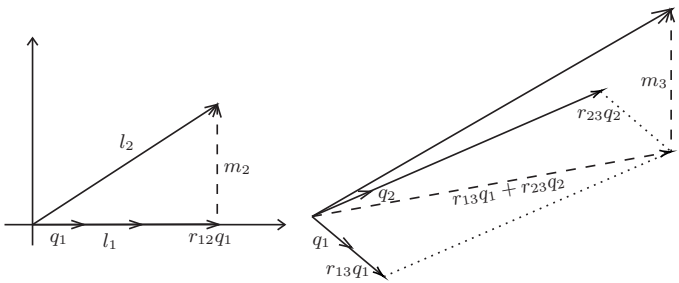

Figura 1.1: Representação geométrica do processo de ortogonalização de Gram-Schmidt.

A Figura 1.1 (esq.) apresenta o vetor l_1 e o vetor normalizado $q_1 = l_1/\|l_1\|_2$. Chamando $r_{12} = \langle l_2, q_1\rangle$, temos $m_2 = l_2 - r_{12}q_1$, ou seja, o vetor m_2 é obtido, subtraindo de l_2, a sua projeção ortogonal sobre o subespaço $span\{q_1\}$. Já a Figura 1.1 (dir.) mostra uma representação geométrica da obtenção do vetor m_3. Assumindo que q_1 e q_2 são conhecidos, projetamos ortogonalmente o vetor l_3 sobre o espaço $span\{q_1, q_2\}$, e posteriormente calculamos a diferença entre l_3 e sua projeção ortogonal.

Exemplo 1.6. *Considere $\mathbb{R}^4$ com o produto interno canônico e sejam $l_1 = (1,1,1,1)$, $l_2 = (1,2,4,5)$ e $l_3 = (1,-3,-4,-2)$. Vamos utilizar o processo de ortogonalização de Gram-Schmidt para encontrar uma base ortonormal do subespaço vetorial $M = span\,\{l_1, l_2, l_3\}$. Primeiramente, utilizaremos o processo de ortogonalização de Gram-Schmidt para encontrar uma base ortogonal de M e, posteriormente, normalizamos esses vetores. Com efeito,*

$$m_1 = l_1 = (1,1,1,1).$$

O vetor m_2 é calculado da seguinte forma:

$$m_2 = l_2 - \frac{\langle l_2, m_1\rangle}{\|m_1\|^2} m_1 = (1,2,4,5) - \frac{12}{4}(1,1,1,1) = (-2,-1,1,2).$$

Por fim,

$$\begin{aligned} m_3 &= l_3 - \frac{\langle l_3, m_1\rangle}{\|m_1\|^2} m_1 - \frac{\langle l_3, m_2\rangle}{\|m_2\|^2} m_2 \\ &= (1,-3,-4,-2) + \frac{8}{4}(1,1,1,1) + \frac{7}{10}(-2,-1,1,2) \\ &= \frac{1}{10}(16,-17,-13,14). \end{aligned}$$

Assim, os vetores m_1, m_2 e m_3 são mutuamente ortogonais. Para encontrar uma base ortonormal de M vamos normalizar os vetores m_1, m_2 e m_3:

$$q_1 = \frac{m_1}{\|m_1\|} = \frac{1}{2}(1,1,1,1),$$

$$q_2 = \frac{m_2}{\|m_2\|} = \frac{1}{\sqrt{10}}(-2,-1,1,2),$$

$$q_3 = \frac{m_3}{\|m_3\|} = \frac{1}{\sqrt{910}}(16,-17,-13,14)$$

formam uma base ortonormal de M.

Exemplo 1.7. *Considere o $\mathbb{C}$-espaço vetorial $\mathbb{C}^3$ com o produto interno canônico. Vamos encontrar uma base ortonormal do subespaço vetorial M gerado pelos vetores $l_1 = (1,i,0)$ e $l_2 = (1,2,1-i)$. Pelo processo de ortogonalização de Gram-Schmidt obtemos m_1 e m_2 dados por:*

$$\begin{aligned} m_1 &= l_1 = (1,i,0) \\ m_2 &= l_2 - \frac{\langle l_2, m_1\rangle}{\|m_1\|^2} m_1 \\ &= (1,2,1-i) - \frac{1-2i}{2}(1,2,1-i) \\ &= \frac{1}{2}(1+2i, 1-2i, 2-2i). \end{aligned}$$

Note que, $\|m_1\| = \sqrt{2}$ e $\|m_2\| = \sqrt{18}/2$. Portanto,

$$q_1 = \frac{m_1}{\|m_1\|} = \frac{1}{\sqrt{2}}(1,i,0),$$

$$q_2 = \frac{m_2}{\|m_2\|} = \frac{1}{\sqrt{18}}(1+2i, 1-2i, 2-2i)$$

formam uma base ortonormal de M.

Um possível algoritmo para o processo de ortogonalização de Gram-Schmidt é

Algoritmo 1 Processo de Gram-Schmidt

1: **function** Q=GSC(A)
2: $[m, n]$ =size(A); R =zeros(n);
3: **for** $j = 1 : n$ **do**
4: $q_j = a_j$;
5: **for** $i = 1 : j - 1$ **do**
6: $r_{ij} = \langle q_i, q_j \rangle$;
7: **end for**
8: **for** $i = 1 : j - 1$ **do**
9: $q_j = q_j - r_{ij} q_i$;
10: **end for**
11: $r_{jj} = \|q_j\|_2^2$;
12: **if** $r_{jj} = 0$ **then**
13: disp('Os vetores dados são LD')
14: $r_j = [\ ]$;
15: $r_{j:} = [\ ]$;
16: $q_j = [\ ]$;
17: **Stop**
18: **else**
19: $q_j = q_j / r_{jj}$;
20: **end if**
21: **end for**
22: **end function**

A entrada do algoritmo é uma matriz $A \in \mathbb{K}^{m \times n}$ cujas colunas são os vetores que queremos ortogonalizar. Primeiramente, observe que se $m < n$, então as colunas de A formam um conjunto linearmente dependente. Também, note que não precisávamos criar a matriz R cujas entradas são $r_{ij} = \langle q_i, q_j \rangle$. As linhas 12-17 têm por objetivo identificar se o conjunto de vetores dados é linearmente dependente, à luz da Observação 1.3. A linha 14 é baseada em um comando MATLAB que deleta a coluna r_j na matriz R, assim como na linha 16 deletamos a j-ésima coluna de Q. Já a notação da linha 15 diz que a linha j de R é deletada. A necessidade de deletar essas colunas é por conta da criação da coluna antes de verificar que os vetores dados são linearmente dependentes. Já a última linha de R é deletada pois é nula. Por fim, o algoritmo acima tem como saída um conjunto ortonormal, as colunas de Q. Alguns autores chamam esse método de Gram-Schmidt clássico.

Como consequência imediata do processo de ortogonalização de Gram-Schmidt, obtemos o seguinte resultado.

Teorema 1.5. *Todo espaço vetorial de dimensão finita com produto interno possui uma base ortonormal.*

Demonstração. Seja L um espaço vetorial de dimensão finita com produto interno $\langle\ ,\ \rangle$ e base $\mathbb{B} = \{l_1, \ldots, l_n\}$. Aplicando o processo de ortogona-

lização de Gram-Schmidt obtemos uma base $\mathbb{B}' = \{m_1, \ldots, m_n\}$ ortogonal. A base ortonormal de L é obtida trocando m_i por $\dfrac{m_i}{\|m_i\|}$, ou seja, $\mathbb{B}'' = \left\{\frac{m_1}{\|m_1\|}, \ldots, \frac{m_n}{\|m_n\|}\right\}$ é uma base ortonormal de L. □

Observação 1.4. *Uma vantagem das bases ortonormais sobre bases quaisquer de um espaço vetorial é que cálculos envolvendo coordenadas de vetores se tornam mais simples com as bases ortonormais. Para exemplificar, suponha que L é um espaço vetorial de dimensão finita com produto interno $\langle\ ,\ \rangle$ e considere a base ordenada $\mathbb{B} = \{l_1, \ldots, l_n\}$ de L. Sabemos que a matriz G, $g_{ij} = \langle l_j, l_i\rangle$, define completamente o produto interno em L. Se $\mathbb{B}$ é ortonormal, então G é a matriz identidade e, portanto, $\forall\, x, y \in L$,*

$$\langle x, y\rangle = \sum_{i,j=1}^{n} \overline{y}_i g_{ij} x_j = \sum_{i=1}^{n} x_i \overline{y}_i. \tag{1.4.3}$$

Portanto, em termos de uma base ortonormal, o produto interno em L parece o produto interno canônico em $\mathbb{K}^n$. □

O método de Gram-Schmidt clássico (GSC) é numericamente bastante sensível a erros de arredondamento que causam a perda da ortogonalidade dos vetores calculados. Uma solução é a aplicação do método de Gram-Schmidt modificado (GSM), cujo processo de ortogonalização é o mesmo, com uma alteração nos cálculos. Já o processo recursivo do GSC é baseado em (1.4.2), isto é,

$$q_j = l_j - \sum_{i=1}^{j-1} r_{ij} q_i,$$

com $r_{ij} = \langle l_j, q_i\rangle$. Não há a necessidade de se dividir pela norma de q_i, pois a cada passo normalizamos o vetor resultante. Assim, no GSC calcula-se todos coeficientes r_{ij} e, posteriormente, as somas são efetuadas. No método de Gram-Schmidt modificado cada novo vetor, digamos q_j, é computado de forma que seja ortogonal aos demais vetores já determinados e, posteriormente, a j-ésima coluna de R é determinada. Portanto, a principal diferença entre GSC e GSM é como a matriz R é calculada.

Watkins [156, pág. 229] resume bem o processo GSM. Para atualizar o vetor l_j e transformá-lo em q_j computa-se o coeficiente $r_{1j} = \langle l_j, q_1\rangle$ que é utilizado para a atualização,

$$l_j^{(1)} = l_j - r_{1j} q_1.$$

Assim, $l_j^{(1)}$ é ortogonal a q_1 (Exercício 42). O próximo coeficiente r_{2j} é computado, mas agora utilizando $l_j^{(1)}$ em vez de l_j, ou seja, $r_{2j} = \langle l_j^{(1)}, q_2\rangle$. Atualizando $l_j^{(1)}$, obtemos

$$l_j^{(2)} = l_j^{(1)} - r_{2j} q_2,$$

que é ortogonal a $\{q_1, q_2\}$. Após $j-1$ passos obtemos $l_j^{(j-1)} = q_j$ e ortogonal a $\{q_1, q_2, \ldots, q_{j-1}\}$. O GSM pode ser resumido no seguinte algoritmo.

Algoritmo 2 Processo de Gram-Schmidt Modificado

```
function Q=GSM(A)
    [m,n] =size(A); R =zeros(n);
    for j = 1 : n do
        q_j = a_j;
        for i = 1 : j − 1 do
            r_ij = ⟨q_i, q_j⟩;
            q_j = q_j − r_ij q_i;
        end for
        r_jj = ‖q_j‖_2;
        if r_jj = 0 then
            disp('Os vetores dados são LD')
            r_j = [ ];
            r_j: = [ ];
            q_j = [ ];
            Stop
        else
            q_j = q_j/r_jj;
        end if
    end for
end function
```

Um resumo comparativo entre GSC e GSM [88] é

$$r_{ij} = \begin{cases} \langle l_j, q_i \rangle & (GSC), \\ \left\langle l_j - \displaystyle\sum_{i=1}^{j-1} r_{ij} q_i,\, q_i \right\rangle & (GSM). \end{cases}$$

Defina $Q_{j-1} = [q_1\ q_2\ \cdots\ q_{j-1}]$. A k-ésima coluna de Q é computada da seguinte forma

$$\tilde{q}_j = \begin{cases} \left(I - Q_{j-1}Q_{j-1}^*\right) l_j & (GSC), \\ \left(I - q_{j-1}q_{j-1}^*\right) \cdots \left(I - q_1 q_1^*\right) l_j & (GSM). \end{cases} \tag{1.4.4}$$

Em ambos os casos, $q_j = \tilde{q}_j / \|\tilde{q}_j\|_2$. As iterações acima são para $j > 2$ e $1 < i < j$, já que as duas primeiras colunas de Q e R e a primeira linha de R são determinadas com a mesma aritmética para ambos os processos.

Como dito anteriormente as expressões em (1.4.4) são equivalentes em aritmética exata, porém em aritmética de precisão finita (caso numérico) são diferentes. Vejamos um exemplo para mostrar essa situação

Exemplo 1.8. *Seguindo a metodologia do exemplo proposto por [88, pág. 503] criamos uma matriz* 10×10 *que é escrita como*

$$A = U^T D V,$$

onde U e V são matrizes ortogonais e $D = \mathrm{diag}(1, 10^{-1}, \ldots, 10^{-9})$. Assim, $\kappa_2(A) = 10^9$. Björck [10] demonstrou que existe uma constante $c_1(m,n)$, tal que $c_1 \boldsymbol{u}\kappa(A) < 1$ de forma que a matriz Q calculada através do método de Gram-Schmidt modificado satisfaz a seguinte desigualdade

$$\|I - Q^*Q\| \leqslant \frac{c_1 \boldsymbol{u}\kappa_2(A)}{1 - c_1 \boldsymbol{u}\kappa_2(A)},$$

onde $\boldsymbol{u}$ é o epsilon da máquina. Smoktunowicz, Barlow e Langou [133, pág. 302] demonstraram que existe uma constante $c_2(m,n)$, tal que $c_2 \boldsymbol{u}\kappa(A)^2 < 1$ de forma que a matriz Q calculada através do método GSC ligeiramente alterado satisfaz a seguinte desigualdade

$$\|I - Q^*Q\| \leqslant c_2 \boldsymbol{u}\kappa(A)^2.$$

Porém, essa estimativa não vale em geral para o método GSC padrão. Sejam $A_j = [a_1 \ \ldots \ a_j]$ e Q_C e Q_M as matrizes com colunas ortonormais geradas pelo GSC e GSM, respectivamente. A Tabela 1.1 fornece uma comparação entre $\|I_k - Q_C^*Q_C\|_2$ e $\|I_k - Q_M^*Q_M\|_2$.

k	$\kappa(A_k)$	$\|I_k - Q_C^*Q_C\|_2$	$\|I_k - Q_M^*Q_M\|_2$
1	$1.0000e+000$	$1.1102e-016$	$1.1102e-016$
2	$8.2745e+000$	$8.8909e-016$	$8.8909e-016$
3	$1.4075e+002$	$5.6060e-014$	$5.7560e-015$
4	$2.6838e+003$	$1.5405e-012$	$9.1397e-015$
5	$2.7027e+004$	$4.6871e-011$	$1.0754e-012$
6	$2.0873e+006$	$3.2880e-007$	$8.1200e-011$
7	$3.7577e+006$	$8.5099e-006$	$1.4233e-010$
8	$1.5424e+007$	$4.4448e-004$	$2.8667e-010$
9	$1.0744e+008$	$6.8247e-003$	$4.2404e-009$
10	$1.0000e+009$	$9.8891e-001$	$4.4368e-009$

Tabela 1.1: Comparação da perda de ortogonalidade para GSC e GSM.

Observe que com o aumento das iterações o processo GSC apresenta uma severa perda de ortogonalidade, enquanto que o processo GSM também apresenta perda de ortogonalidade, mas em menor escala.

Uma forma de minimizar os efeitos da perda de ortogonalidade no método de Gram-Schmidt utiliza um processo de reortogonalização dos vetores em uma dada precisão finita, com o objetivo de ter vetores ortogonais ao nível do *epsilon* da máquina. Denotaremos tal método por GSMR.

De forma geral esse processo é baseado na seguinte ideia: analise se o vetor q_j calculado pelo processo de Gram-Schmidt é ortogonal a $\{q_1, \ldots, q_{j-1}\}$ de forma satisfatória. Caso não seja, ortogonalize novamente, com q_j sendo o vetor de entrada. Passemos aos detalhes formais desse processo, que foi

sugerido por Davis [33, pág. 353]. O primeiro a propor uma condição de verificação de ortogonalidade foi Rutishauser [122, pág. 111] e [123, pág. 290].

Considere $\tilde{q}_j = l_j - \sum_{i=1}^{j-1} r_{ij} q_i$, o j-ésimo vetor do processo de Gram-Schmidt. Se $\|\tilde{q}_j\|_2 \leqslant \frac{1}{K}\|l_j\|_2$ com $K = 10$, então ortogonalize $\tilde{q}_j$. Daniel et al. [30, pág. 774] propuseram $K = \sqrt{2}$, que é um valor bastante utilizado. Uma miríade de artigos foi publicada propondo outros valores de K ou definindo intervalos para K, usualmente $0,1 \leqslant 1/K \leqslant 1/\sqrt{2}$. Para uma revisão sobre esses fatos vide [88, p.506]. Outras referências que discutem o processo de perda de ortogonalidade e suas propriedades são [15, 51].

Note que se pode aplicar o processo de reortogonalização várias vezes, porém, em geral, dois processos de ortogonalização são suficientes para alcançar a ortogonalidade entre os vetores com uma precisão perto da máquina. Há autores que propõem a utilização da reortogonalização a cada passo, e nesse caso, o custo computacional duplica. Para fins didáticos, vamos apresentar um algoritmo de reortogonalização utilizando o processo de Gram-Schmidt modificado.

Algoritmo 3 Processo de Gram-Schmidt Modificado com Reortogonalização

```
function Q=GSMR(A)
    [m,n] =size(A); R =zeros(n);
    for j = 1 : n do
        q_j = a_j;
        for i = 1 : j − 1 do
            for k = 1 : 2 do
                aux = ⟨q_i, q_j⟩;
                q_j = q_j − aux q_i;
                r_ij = r_ij + aux;
            end for
        end for
        r_jj = ‖q_j‖_2;
        if r_jj = 0 then
            disp('Os vetores dados são LD')
            r_j = [ ];
            r_j: = [ ];
            q_j = [ ];
            Stop
        else
            q_j = q_j/r_jj;
        end if
    end for
end function
```

Para $j \geqslant 2$ aplicamos a reortogonalização da seguinte forma, para $i =$

$1, 2$:

$$\tilde{q}_j^{(i)} = \left(I - q_{j-1}q_{j-1}^*\right)\cdots\left(I - q_1 q_1^*\right)\tilde{q}_j^{(i-1)} = \tilde{q}_j^{(i-1)} - \sum_{i=1}^{j-1} q_i \left(q_i^* \tilde{q}_j^{(i-1)}\right),$$

onde $\tilde{q}_j^{(0)} = l_j$ e, após o passo j, obtemos $q_j = \tilde{q}_j^{(2)}/\|\tilde{q}_j^{(2)}\|_2$.

O algoritmo para o GSC com reortogonalização é trivialmente construído a partir deste último e não é apresentado. Mais a frente, num contexto mais sofisticado que utiliza o GSM, apresentamos o algoritmo que reortogonaliza apenas os casos necessários. Vejamos novamente o Exemplo 1.8, mas considerando o GSM com reortogonalização.

Exemplo 1.9. *Considerando a mesma metodologia apresentada no Exemplo 1.8, obtemos os seguintes resultados.*

k	$\kappa(A_k)$	$\|I_k - Q_C^* Q_C\|_2$	$\|I_k - Q_M^* Q_M\|_2$	$\|I_k - Q_{MR}^* Q_{MR}\|_2$
1	$1.0000e+000$	$2.2204e-016$	$2.2204e-016$	$2.2204e-016$
2	$1.7675e+001$	$5.5708e-016$	$5.5708e-016$	$2.2854e-016$
3	$2.3042e+002$	$6.3837e-015$	$2.6212e-015$	$2.8371e-016$
4	$1.9026e+003$	$3.6340e-013$	$4.8484e-014$	$6.7782e-016$
5	$1.0649e+005$	$7.0439e-010$	$4.5256e-012$	$1.4358e-014$
6	$2.3575e+005$	$2.7008e-008$	$1.0084e-011$	$2.4828e-014$
7	$1.1370e+006$	$2.1548e-006$	$1.3686e-010$	$7.4952e-013$
8	$1.1165e+007$	$2.5030e-004$	$1.1356e-009$	$6.4295e-012$
9	$1.0052e+008$	$6.7349e-002$	$3.8516e-009$	$1.2464e-010$
10	$1.0000e+009$	$1.0066e+000$	$2.3471e-008$	$2.6876e-010$

Tabela 1.2: Comparação da perda de ortogonalidade para GSC, GSM e GSMR.

Observe que o GSMR apresenta melhores resultados que o GSC e GSM. Porém, ainda há margem para melhorias no procedimento de reortogonalização.
Apresentamos o GSC e o GSM novamente, pois nossas matrizes ortogonais são aleatórias. Portanto, a matriz A em questão é diferente da matriz do Exemplo 1.8.

Observação 1.5. *É justificado que as colunas referentes ao GSC e ao GSM são novamente apresentadas devido à aleatoriedade das matrizes geradas. No entanto, fixando a semente da geração, seria possível reproduzir exatamente o mesmo experimento, de modo que a repetição não seria necessária.* □

O processo de reortogonalização pode, eventualmente, não apresentar bons resultados. Há certos problemas que necessitam uma robustez no

cálculo de vetores ortogonais. Uma possível melhoria é através da super-ortogonalização, termo definido por Rutishauser. Não pretendemos nos aprofundar nesse tema mas sugerimos [45], uma coleção póstuma dos trabalhos de Rutishauser. Outra opção para o problema da perda de ortogonalidade é a utilização das transformações de Householder.

1.5 Subespaços Ortogonais

Definição 1.13. *Sejam L um espaço vetorial com produto interno e S um subconjunto de L. Definimos o complemento ortogonal de S em L e denotamos por $S^\perp$ como o conjunto de todos os vetores de L que são ortogonais a todos os vetores de S, ou seja,*

$$S^\perp = \{l \in L \mid \langle l, m\rangle = 0, \forall\, m \in S\}.$$

Note que, se $S = \{0\}$, então $S^\perp = L$. Ademais, se uma base de L pertence a S, então $S^\perp = \{0\}$.

O complemento ortogonal de qualquer subconjunto S de L é um subespaço vetorial de L. Por essa razão, às vezes $S^\perp$ é denominado subespaço ortogonal a S.

Propriedade 1.4. *Sejam L um $\mathbb{K}$-espaço vetorial com produto interno e S um subconjunto de L. Então, o complemento ortogonal $S^\perp$ de S é um subespaço vetorial de L.*

Demonstração. Seja $m \in S$ um vetor arbitrário. Note que $0 \in S^\perp$, pois $\langle 0, m\rangle = 0$. Dados $l_1, l_2 \in S^\perp$ temos $\langle l_1, m\rangle = \langle l_2, m\rangle = 0$, que nos leva a

$$\langle l_1 + l_2, m\rangle = \langle l_1, m\rangle + \langle l_2, m\rangle = 0 + 0 = 0.$$

Portanto, $l_1 + l_2 \in S^\perp$. Por fim, se $\lambda \in \mathbb{K}$ e $l \in S^\perp$ temos

$$\langle \lambda l, m\rangle = \lambda \langle l, m\rangle = \lambda \cdot 0 = 0.$$

Portanto, $\lambda l \in S^\perp$. □

Até agora assumimos S como um mero subconjunto de vetores de L. Porém, se S é um subespaço vetorial de L, então $S^\perp$ apresenta algumas propriedades interessantes adicionais.

Propriedade 1.5. *[26, pág. 179] Seja L um $\mathbb{K}$-espaço vetorial com produto interno. Além disso, seja M um subespaço vetorial de L e $\mathbb{B} = \{m_1, \ldots, m_n\}$ um conjunto gerador de M. Então, $l \in M^\perp$ se, e somente se, $\langle l, m_i\rangle = 0$, $i = 1, \ldots, n$.*

Demonstração. Seja $m \in M$ e como $\mathbb{B}$ é um conjunto gerador de M, existem escalares $a_1, \ldots, a_n$, tais que

$$m = \sum_{i=1}^{n} a_i m_i.$$

Assim, para $l \in L$, tal que $\langle l, m_i \rangle = 0$, $i = 1, \ldots, n$, temos

$$\langle l, m \rangle = \left\langle l, \sum_{i=1}^{n} a_i m_i \right\rangle = \sum_{i=1}^{n} \overline{a}_i \langle l, m_i \rangle = 0.$$

Portanto, $l \in M^{\perp}$. Por outro lado, se $l \in M^{\perp}$, então $\langle l, m \rangle = 0$, $\forall\, m \in M$, em particular, para os vetores de $\mathbb{B}$. $\square$

Teorema 1.6. *[26, pág. 179] Sejam L um espaço vetorial de dimensão finita com produto interno e M um subespaço vetorial de L. Então,*

$$L = M \oplus M^{\perp}.$$

Demonstração. Seja $\mathbb{B} = \{m_1, \ldots, m_k\}$ uma base ortogonal de M (Propriedade 1.5) e seja $\mathscr{C} = \{m_1, \ldots, m_k, m_{k+1}, \ldots, m_n\}$ uma base ortogonal de L. Assim, pela Propriedade 1.3, para todo $l \in L$,

$$l = \sum_{i=1}^{n} \frac{\langle l, m_i \rangle}{\|m_i\|^2} m_i = \sum_{i=1}^{k} \frac{\langle l, m_i \rangle}{\|m_i\|^2} m_i + \sum_{i=k+1}^{n} \frac{\langle l, m_i \rangle}{\|m_i\|^2} m_i.$$

Claramente

$$\sum_{i=1}^{k} \frac{\langle l, m_i \rangle}{\|m_i\|^2} m_i \in M$$

e, ainda,

$$\left\langle m_j, \sum_{i=k+1}^{n} \frac{\langle l, m_i \rangle}{\|m_i\|^2} m_i \right\rangle = \sum_{i=k+1}^{n} \frac{\overline{\langle l, m_i \rangle}}{\|m_i\|^2} \langle m_j, m_i \rangle = 0, \;\; j = 1, \ldots, k.$$

Segue-se, portanto, da Proposição 1.5, que

$$\sum_{i=k+1}^{n} \frac{\langle l, m_i \rangle}{\|m_i\|^2} m_i \in M^{\perp}.$$

Logo, $L = M + M^{\perp}$. Por outro lado, seja $m \in M \cap M^{\perp}$. Assim, $\langle m, l \rangle = 0$, para todo $l \in M$, em particular, para $l = m$, ou seja, $\langle m, m \rangle = 0$, implicando em $m = 0$. $\square$

Uma consequência imediata do teorema acima é que

$$dim_{\mathbb{K}} L = dim_{\mathbb{K}} M + dim_{\mathbb{K}} M^{\perp}. \tag{1.5.5}$$

O complemento ortogonal de um subespaço vetorial M é utilizado para aproximar elementos de L por elementos de M e tem diversas aplicações em matemática, como soluções aproximadas de sistemas lineares incompatíveis, aproximação de funções por polinômios, entre outras.

Propriedade 1.6. *[26, pág. 182] Sejam L um $\mathbb{K}$-espaço vetorial com produto interno e M um subespaço vetorial de L. Então, para cada $l \in L$, existe um único $m \in M$, tal que $l - m \in M^{\perp}$.*

Demonstração. Seja $\mathbb{B} = \{m_1, \ldots, m_n\}$ uma base ortogonal do subespaço vetorial M. Para cada $l \in L$, considere

$$m := \frac{\langle l, m_1 \rangle}{\|m_1\|^2} m_1 + \cdots + \frac{\langle l, m_n \rangle}{\|m_n\|^2} m_n.$$

Assim, $m \in M$ e para cada $i = 1, \ldots, n$,

$$\begin{aligned} \langle l - m, m_i \rangle &= \langle l, m_i \rangle - \langle m, m_i \rangle \\ &= \langle l, m_i \rangle - \sum_{j=1}^{n} \frac{\langle l, m_j \rangle}{\|m_j\|^2} \langle m_i, m_j \rangle \\ &= 0. \end{aligned}$$

Portanto, pela Propriedade 1.5, $l - m \in M^{\perp}$.
Provemos, agora, a unicidade. Para isso, sejam $m, m' \in M$, tais que $l - m \in M^{\perp}$ e $l - m' \in M^{\perp}$. Então,

$$\begin{aligned} \langle m - m', m - m' \rangle &= \langle m - m', (m - m') + (l - l) \rangle \\ &= \langle m - m', m - l \rangle + \langle m - m', l - m' \rangle \\ &= 0 + 0 = 0, \end{aligned}$$

pois $m - m' \in M$, $m - l \in M^{\perp}$ e $l - m' \in M^{\perp}$. Logo, $m - m' = 0$, ou seja, $m = m'$. □

Definição 1.14. *Sejam L um $\mathbb{K}$-espaço vetorial com produto interno e M um subespaço vetorial de L. Então para cada $l \in L$, aquele $m \in M$ tal que $l - m \in M^{\perp}$ é denominado a projeção ortogonal de l sobre o subespaço vetorial M, e denotado por $m = \operatorname{proj}_M l$.*

Na próxima seção aprofundamos o estudo sobre as projeções ortogonais. A Propriedade 1.6 diz que a projeção ortogonal é única e

$$\operatorname{proj}_M l = \frac{\langle l, m_1 \rangle}{\|m_1\|^2} m_1 + \cdots + \frac{\langle l, m_n \rangle}{\|m_n\|^2} m_n. \tag{1.5.6}$$

Propriedade 1.7. *[26, pág. 183] Sejam L um $\mathbb{K}$-espaço vetorial com produto interno e M um subespaço vetorial de L. Para cada $l \in L$, as seguintes afirmações são equivalentes.*

1. *Existe $m_0 \in M$ tal que $l - m_0 \in M^{\perp}$.*

2. Existe $m_0 \in M$ tal que $\|l - m_0\| < \|l - m\|$, $\forall\, m \in M$ e $m \neq m_0$.

Demonstração. Seja $m_0 \in M$ tal que $l - m_0 \in M^{\perp}$. Para cada $m \in M$, temos $m - m_0 \in M$ e assim $l - m_0 \perp m - m_0$. Segue do Exercício 49 que

$$\|l - m\|^2 = \|l - m_0 + m_0 - m\|^2 = \|l - m_0\|^2 + \|m - m_0\|^2 .$$

Portanto, para $m \neq m_0$,

$$\|l - m_0\| < \|l - m\| .$$

Por outro lado, seja $m_0 \in M$, tal que

$$\|l - m_0\| < \|l - m\| , \ \ \forall\, m \in M, \ m \neq m_0. \tag{1.5.7}$$

Suponha, por absurdo, que $l - m_0 \notin M^{\perp}$, ou seja, existe $m_1 \in M$ tal que $\langle l - m_0, m_1 \rangle \neq 0$. Como $m_1 \neq 0$, considere o subespaço vetorial

$$M' = span\{m_0, m_1\}$$

que, nos diz que $M' \subseteq M$ e $dim_{\mathbb{K}} M' = 1$ ou 2. Pela Propriedade 1.6, existe $\operatorname{proj}_{M'} l := m_0'$. Assim, $l - m_0' \in M'^{\perp}$ e, portanto,

$$\left\|l - m_0'\right\| < \left\|l - m'\right\| , \ \ \forall\, m' \in M', \ m' \neq m_0'.$$

Note que, $m_0' \neq m_0$, pois $\langle l - m_0, m_1 \rangle \neq 0$. Daí,

$$\left\|l - m_0'\right\| < \|l - m_0\| . \tag{1.5.8}$$

De (1.5.7) e (1.5.8) temos

$$\|l - m_0\| < \left\|l - m_0'\right\| < \|l - m_0\| ,$$

uma contradição. □

A interpretação dessa proposição é que $\operatorname{proj}_M l$, quando existe, é a melhor aproximação de l por um vetor de M e vice-versa.

Quando a dimensão de M for finita, o problema de determinar a projeção ortogonal de um vetor $l \in L$ sobre M é equivalente a determinar um vetor $m \in M$ que melhor se aproxima de L. Na Seção 2.1 veremos o método dos quadrados mínimos para a resolução de sistemas lineares, que se baseia essencialmente nesse resultado.

Exemplo 1.10. *[26, pág. 184]*

1. *Sejam $\mathbb{R}^3$ com o produto interno canônico, $l = (3, 0, 2)$ e o subespaço vetorial $M = span\{(1, 0, -2), (1, 1, 1)\}$. Queremos calcular $\operatorname{proj}_M l$, mas note que, $m_1 = (1, 0, -2)$ e $m_2 = (1, 1, 1)$ não formam uma base ortogonal de*

M. Assim, inicialmente precisamos encontrar uma base ortogonal $\{l_1, l_2\}$ de M. Pelo processo de ortogonalização de Gram-Schmidt, obtemos

$$\begin{aligned} l_1 &= m_1 = (1,0,-2) \\ l_2 &= m_2 - \frac{\langle m_2, l_1\rangle}{\|l_1\|^2} l_1 \\ &= (1,1,1) + \frac{1}{5}(1,0,-2) \\ &= \frac{1}{5}(6,5,3). \end{aligned}$$

Portanto, por (1.5.6),

$$\operatorname{proj}_M l = \frac{\langle l, l_1\rangle}{\|l_1\|^2} l_1 + \frac{\langle l, l_2\rangle}{\|l_2\|^2} l_2 = \frac{1}{7}(13,12,10).$$

2. *Definimos $\mathbb{R}_k[x]$ o espaço dos polinômios de grau menor ou igual a k. Seja $L = \mathbb{R}_3[x]$ com o produto interno*

$$\langle p, q\rangle = \int_0^1 p(x)q(x)\,dx.$$

Vamos calcular o polinômio de grau 1 que melhor aproxima $p(x) = x^3$. Como $\mathbb{R}_1[x]$ tem dimensão finita, o problema é resolvido encontrando $\operatorname{proj}_{\mathbb{R}_1[x]} x^3$. Com efeito, seja $\mathbb{B} = \{1, x\}$ a base canônica de $\mathbb{R}_1[x]$. Como a base não é ortogonal, apliquemos o processo de ortogonalização de Gram-Schmidt,

$$\begin{aligned} p_1(x) &= 1 \\ p_2(x) &= x - \frac{\langle x, 1\rangle}{\|1\|^2} 1 = x - \frac{1}{2}. \end{aligned}$$

Logo,

$$\begin{aligned} \operatorname{proj}_{\mathbb{R}_1[x]} x^3 &= \frac{\langle x^3, 1\rangle}{\|1\|^2} 1 + \frac{\langle x^3, x - \frac{1}{2}\rangle}{\|x - \frac{1}{2}\|^2}\left(x - \frac{1}{2}\right) \\ &= \frac{1}{4} + \frac{9}{10}\left(x - \frac{1}{2}\right) = \frac{9}{10}x - \frac{1}{5}. \end{aligned}$$

1.6 Projeções e Projeções Ortogonais

Para entender as ideias básicas do método de quadrados mínimos precisaremos desenvolver alguns conceitos sobre projeções ortogonais. Seja $P \in \mathbb{K}^{n\times n}$

uma matriz quadrada. A matriz P é uma projeção se, e somente se, $P^2 = P$. Portanto, chamamos de matriz de projeção ou projetor, uma matriz quadrada P que satisfaz a propriedade $P^2 = P$.

A interpretação geométrica de uma projeção P é que, para todo vetor $x \in \mathbb{K}^n$, o vetor Px é a "sombra" de x sobre $\mathrm{Im}(P)$. Se $x \in \mathrm{Im}(P)$, então a "sombra" de x sobre $\mathrm{Im}(P)$ é o próprio x, ou seja, $Px = x$. A demonstração desse fato é simples: dado $x \in \mathrm{Im}(P)$, existe $y \in \mathbb{K}^n$ tal que $x = Py$. Logo,

$$Px = P(Py) = P^2y = Py = x.$$

Neste contexto, o Teorema do Núcleo e da Imagem pode ser reescrito da seguinte forma: Seja $P \in \mathbb{K}^{n\times n}$ uma matriz de projeção, então

$$\mathbb{K}^n = \mathrm{Ker}\,(P) \oplus \mathrm{Im}(P). \tag{1.6.9}$$

Embora este resultado valha para qualquer matriz $A \in \mathbb{K}^{m\times n}$, restringimos às matrizes quadradas e de projeção, pois é o que necessitamos.

Definição 1.15. *Seja $P \in \mathbb{K}^{n\times n}$ um projetor. Dizemos que P é uma matriz de projeção ortogonal ou um projetor ortogonal se os subespaços* $\mathrm{Ker}\,(P)$ *e* $\mathrm{Im}(P)$ *são ortogonais.*

Essa definição é clara do ponto de vista geométrico, porém não muito prática do ponto de vista algébrico. O próximo resultado permite utilizar matrizes de projeção ortogonal em contextos algébricos.

Propriedade 1.8. *[147, pág. 44] Seja $P \in \mathbb{K}^{n\times n}$ um projetor. Então, P é uma matriz de projeção ortogonal se, e somente se, $P = P^*$.*

Demonstração. ($\Rightarrow$) Seja P um projetor ortogonal. Note que, para todo $y \in \mathbb{K}^n$, $(y - Py) \in \mathrm{Ker}\,(P)$. Portanto, pela definição de projetor ortogonal, para todos $x, y \in \mathbb{K}^n$, $\langle Px, (y - Py)\rangle = 0$ ou, equivalentemente, $\langle Px, y\rangle - \langle Px, Py\rangle = 0$. Logo,

$$\langle x, Py\rangle = \langle Px, Py\rangle = \langle Px, y\rangle = \langle x, P^*y\rangle\,.$$

Assim, $P = P^*$.
($\Leftarrow$) Seja P um projetor auto-adjunto. Para todos $x, y \in \mathbb{K}^n$, $Px \in \mathrm{Im}(P)$ e $(y - Py) \in \mathrm{Ker}\,(P)$. Assim,

$$\begin{aligned}\langle Px, (y - Py)\rangle &= \langle x, P^*(y - Py)\rangle = \langle x, P(y - Py)\rangle \\ &= \left\langle x, (Py - P^2y)\right\rangle = 0.\end{aligned}$$

Portanto, P é uma matriz de projeção ortogonal. □

Vejamos como determinar a matriz ou operador projeção ortogonal sobre um subespaço vetorial M de $\mathbb{K}^m$. Mas antes provaremos um lema que será utilizado nessa construção.

Lema 1.1. *Uma matriz $A \in \mathbb{K}^{m\times n}$ tem posto completo se, e somente se, A^*A é não singular.*

Demonstração. Vamos demonstrar as contra positivas. Se A^*A é singular, então existe $x \neq 0$, tal que $A^*Ax = 0$, que implica $0 = \langle x, A^*Ax\rangle = \langle Ax, Ax\rangle$, ou seja, $Ax = 0$. Logo, A é singular e, portanto, tem posto deficiente. Reciprocamente, se A tem posto deficiente, então existe $x \neq 0$, tal que $Ax = 0$ e, portanto, $A^*Ax = 0$. Logo, A^*A é singular. □

Propriedade 1.9. *[147, pág. 46] Seja M um subespaço vetorial de $\mathbb{K}^m$ de dimensão $n \geqslant 1$. A matriz $P \in \mathbb{K}^{m\times n}$ de projeção ortogonal sobre M é dada por*

$$P = A(A^*A)^{-1}A^*,$$

onde as colunas da matriz A são os vetores da base de M.

Demonstração. Seja $\mathbb{B} = \{a_1, \ldots, a_n\}$ uma base de M. Defina a matriz $A \in \mathbb{K}^{m\times n}$, cujas colunas são os vetores da base $\mathbb{B}$. Claramente, $M = \mathrm{Im}(A)$ e, dado $z \in \mathbb{K}^m$, considere $y \in \mathrm{Im}(A)$ a projeção ortogonal de z sobre $\mathrm{Im}(A)$. Pela definição de projeção ortogonal $(y - z) \perp \mathrm{Im}(A)$ e, portanto, $\langle a_i, y - z\rangle = 0$, $i = 1, \ldots, n$, que matricialmente se torna, $A^*(y - z) = 0$. Como $y \in \mathrm{Im}(A)$, existe $x \in \mathbb{K}^m$, tal que $y = Ax$. Portanto,

$$A^*(Ax - z) = 0 \;\Leftrightarrow\; A^*Ax = A^*z \;\Leftrightarrow\; x = (A^*A)^{-1}A^*z.$$

Note que, pelo Lema 1.1, $(A^*A)^{-1}$ existe. Porém, queremos conhecer a projeção ortogonal y que, de fato, é

$$y = Ax = A(A^*A)^{-1}A^*z.$$

Chamando a matriz de projeção ortogonal de P e como z é um vetor arbitrário de $\mathbb{K}^m$, temos

$$P = A(A^*A)^{-1}A^*.$$

□

Observação 1.6. *A matriz P acima é, por construção, um projetor ortogonal. Porém, é um bom exercício demonstrar esse fato e deixamos a cargo do leitor (Exercício 79).* □

Os projetores são muito utilizados em aplicações de álgebra linear e, para uma descrição mais completa desse tema, sugerimos [136, 139].

1.7 Subespaços Fundamentais e a Alternativa de Fredholm

Dada uma matriz $A \in \mathbb{K}^{m\times n}$, seus quatro subespaços fundamentais em álgebra linear aplicada são o núcleo de A, o núcleo de A^*, a imagem de A e a imagem de A^*. Assim, temos a seguinte definição.

Definição 1.16. *Seja* $A \in \mathbb{K}^{m\times n}$.

1. *O conjunto* $\{x \in \mathbb{K}^n \mid Ax = 0\}$ *é chamado de núcleo de* A *e será denotado por* $\operatorname{Ker}(A)$.

2. *O conjunto* $\{x \in \mathbb{K}^m \mid A^*x = 0\}$ *é chamado de núcleo de* A^* *ou conúcleo de* A *e será denotado por* $\operatorname{Ker}(A^*)$ *ou* $\operatorname{Coker}(A)$.

3. *O conjunto* $\{y \in \mathbb{K}^m \mid \exists\, x \in \mathbb{K}^n,\ Ax = y\}$ *é chamado de imagem de* A *e será denotado por* $\operatorname{Im}(A)$.

4. *O conjunto* $\{y \in \mathbb{K}^n \mid \exists\, x \in \mathbb{K}^m,\ A^*x = y\}$ *é chamado de imagem de* A^* *ou coimagem de* A *e será denotado por* $\operatorname{Im}(A^*)$ *ou* $\operatorname{Coim}(A)$.

O próximo resultado afirma que $\operatorname{Ker}(A)$ e $\operatorname{Coim}(A)$ são complementos ortogonais um do outro, assim como, $\operatorname{Im}(A)$ tem como complemento ortogonal $\operatorname{Coker}(A)$.

Teorema 1.7. *Seja* $A \in \mathbb{K}^{m\times n}$ *uma matriz arbitrária. Então:*

1. $\operatorname{Im}(A)^\perp = \operatorname{Ker}(A^*)$.

2. $\operatorname{Ker}(A)^\perp = \operatorname{Im}(A^*)$.

Demonstração. Dado $x \in \operatorname{Im}(A)^\perp$ temos, $\forall\, y \in \mathbb{K}^n$,

$$\begin{aligned} x \in \operatorname{Im}(A)^\perp \iff \langle x, Ay\rangle = 0 \iff \langle A^*x, y\rangle = 0 \\ \iff A^*x = 0 \iff x \in \operatorname{Ker}(A^*). \end{aligned}$$

A penúltima equivalência segue imediatamente do Exercício 51.
Para 2, seja $x \in \operatorname{Ker}(A)$. Então, $\forall\, y \in \mathbb{K}^n$, temos

$$\begin{aligned} x \in \operatorname{Ker}(A) \iff \langle Ax, y\rangle = 0 \iff \langle x, A^*y\rangle = 0 \\ \iff x \in [\operatorname{Im}(A^*)]^\perp. \end{aligned}$$

Portanto, $\operatorname{Ker}(A) = \operatorname{Im}(A^*)^\perp$ ou, de forma equivalente, $\operatorname{Ker}(A)^\perp = \operatorname{Im}(A^*)$. □

O seguinte corolário é uma aplicação direta desse teorema, do Teorema do Núcleo e da Imagem e do fato que $\operatorname{rank}(A) = \operatorname{rank}(A^*)$.

Corolário 1.3. *Seja* $A \in \mathbb{K}^{m\times n}$ *uma matriz arbitrária de porto* r*, então*

1. $dim_{\mathbb{K}}[\operatorname{Im}(A)] = dim_{\mathbb{K}}[\operatorname{Im}(A^*)] = r$.

2. $dim_{\mathbb{K}}[\operatorname{Ker}(A)] = n - r$.

3. $dim_{\mathbb{K}}[\operatorname{Coker}(A)] = m - r$.

4. $\mathbb{K}^n = \operatorname{Ker}(A) \oplus \operatorname{Im}(A^*)$.

5. $\mathbb{K}^m = \operatorname{Ker}(A^*) \oplus \operatorname{Im}(A)$.

Esse resultado é conhecido como Teorema Fundamental da Álgebra Linear [102, pág. 114]. Como consequência do Teorema 1.7 e do Corolário 1.3 demonstramos um resultado chamado de Alternativa de Fredholm.[3]

Teorema 1.8. *[102, pág. 222]* ***(Alternativa de Fredholm)*** *O sistema linear $Ax = b$, com $A \in \mathbb{K}^{m\times n}$, tem solução se, e somente se, b é ortogonal a $\operatorname{Ker}(A^*)$.*

Demonstração. O sistema linear $Ax = b$ tem solução se, e somente se, $b \in \operatorname{Im}(A)$, e isso é equivalente a $b \in \operatorname{Ker}(A^*)^\perp$ pelo Teorema 1.7. □

Observação 1.7. *Em termos de conjuntos a Alternativa de Fredholm diz que "$b \in \operatorname{Im}(A) \Leftrightarrow b \in \operatorname{Ker}(A^*)^\perp$". Nessa forma a Alternativa de Fredholm é um simples corolário do Teorema 1.7.* □

Observação 1.8. *Uma forma equivalente de enunciar a Alternativa de Fredholm é a seguinte [128, pág. 155].*

(Alternativa de Fredholm) *Sejam $A \in \mathbb{K}^{m\times n}$ e $b \in \mathbb{K}^m$. Então, exatamente uma das afirmações é verdadeira:*

1. $Ax = b$ é consistente, isto é, tem solução.

2. Existe y tal que $A^*y = 0$ e $y^*b \neq 0$.

□

Exemplo 1.11. *Vejamos como utilizar a Alternativa de Fredholm com base no resultado da Observação 1.8. Esse exemplo foi proposto por Scheick [128, pág. 155]. Suponha que $A \in \mathbb{K}^{n\times n}$ seja auto-adjunta e tenha posto $n-1$. Suponha também que $Au = 0$ para $u = (1, \ldots 1)$. Vamos demonstrar que $Ax = b$ tem solução se e somente se $b_1 + \cdots + b_n = 0$.*

Com efeito, a Alternativa de Fredholm afirma que $b \in \operatorname{Im}(A)$ se, e somente se, $b \in \operatorname{Ker}(A^*)^\perp$. Portanto, para todo $x \in \operatorname{Ker}(A^*) = \operatorname{Ker}(A)$, $x^*b = 0$. Por outro lado, $\operatorname{Ker}(A)$ é um subespaço 1-dimensional, assim $x = \lambda u$, ou seja, u é a base de $\operatorname{Ker}(A)$. Então, $b \in \operatorname{Im}(A) \Leftrightarrow u^*b = 0$, ou seja, $b_1 + \cdots + b_n = 0$.

Exemplo 1.12. *Podemos também utilizar a eliminação gaussiana para determinar as condições que garantam que $Ax = b$ tenha solução. Vejamos*

[3] `https://mathshistory.st-andrews.ac.uk/Biographies/Fredholm/`

um exemplo. Considere a matriz

$$A = \begin{bmatrix} 1 & -1 & -1 \\ 1 & 1 & 1 \\ 0 & -2 & -2 \end{bmatrix}.$$

e o vetor $b = [b_1, b_2, b_3]^T$. *A condição para* $Ax = b$ *ser consistente é* $b_1 - b_2 - b_3 = 0$. *Com efeito,*

$$\left[\begin{array}{ccc|c} 1 & -1 & -1 & b_1 \\ 1 & 1 & 1 & b_2 \\ 0 & -2 & -2 & b_3 \end{array}\right] \sim \left[\begin{array}{ccc|c} 1 & -1 & -1 & b_1 \\ 0 & -2 & -2 & b_1 - b_2 \\ 0 & -2 & -2 & b_3 \end{array}\right]$$

$$\sim \left[\begin{array}{ccc|c} 1 & -1 & -1 & b_1 \\ 0 & -2 & -2 & b_1 - b_2 \\ 0 & 0 & 0 & b_1 - b_2 - b_3 \end{array}\right].$$

Portanto, a condição para a consistência do sistema linear $Ax = b$ *é* $b_1 - b_2 - b_3 = 0$. *Calculemos, agora, a condição para o sistema linear ser consistente utilizando a Alternativa de Fredholm. De fato, devemos encontrar as condições para que* $b = [b_1, b_2, b_3]^T$ *pertença à* $\operatorname{Ker}(A^*)^\perp = \operatorname{Im}(A)$. *Primeiramente, calculemos uma base para* $\operatorname{Ker}(A^*)$. *Com efeito,*

$$\begin{bmatrix} 1 & 1 & 0 \\ -1 & 1 & -2 \\ -1 & 1 & -2 \end{bmatrix} \sim \begin{bmatrix} 1 & 1 & 0 \\ 0 & 2 & -2 \\ 0 & 2 & -2 \end{bmatrix} \sim \begin{bmatrix} 1 & 1 & 0 \\ 0 & 2 & -2 \\ 0 & 0 & 0 \end{bmatrix}.$$

Logo, há um grau de liberdade. Assim, $\operatorname{Ker}(A^*) = span\{[1, -1, -1]^T\}$. *Mas queremos que* $b \in \operatorname{Ker}(A^*)^\perp$, *isto é, quando* $b^*[1, -1, -1]^T = 0$, *se e somente se,* $b_1 - b_2 - b_3 = 0$.

1.8 Exercícios

1. Expanda:

1. $\langle 2l_1 + 5l_2, 4m_1 + 2m_2 \rangle$.
2. $\langle 4il_1 + 3l_2, 3m_1 + 5m_2 + m_3 \rangle$.
3. $\langle 4il_1 + 3l_2, 3m_1 + 5m_2 + im_3 \rangle$.
4. $\|l + 2v\|^2$.

2. Considere o espaço vetorial $\mathbb{K}^n$, onde $\mathbb{K} = \mathbb{C}$ ou $\mathbb{R}$. Sejam $l = (l_1, \ldots, l_n)$ e $m = (m_1, \ldots, m_n)$ vetores de $\mathbb{K}^n$. Demonstre que são produtos internos em $\mathbb{K}^n$:

1. $\langle l, m \rangle = \displaystyle\sum_{i=1}^{n} l_i \overline{m_i}$.

2. $\langle l, m\rangle = \sum_{i=1}^{n} \alpha_i l_i \overline{m_i}$, com $\alpha_i \in \mathbb{R}_*^+$, $i = 1, \ldots, n$.

3. Seja $L = \mathbb{R}^2$ e $x = (x_1, x_2)$ e $y = (y_1, y_2)$ vetores de $\mathbb{R}^2$, demonstre que

$$\langle x, y\rangle = x_1y_1 - x_2y_1 - x_1y_2 + 4x_2y_2$$

é um produto interno.

4. Seja $\mathbb{B} = \{e_1, e_2\}$ a base canônica de $\mathbb{R}^2$. Encontre um produto interno em $\mathbb{R}^2$ tal que $\langle e_1, e_2\rangle = 3$.

5. Considere os seguintes vetores de $\mathbb{R}^3$, $x = (1, 2, 3)$, $y = (1, 0, 1)$ e $z = (3, 2, 1)$. Assuma que $\langle\ ,\ \rangle$ é o produto interno canônico em $\mathbb{R}^3$. Calcule

1. $\langle 2x + y, z\rangle$.
2. $\langle x, y\rangle$.
3. $\langle x, y + z\rangle$.
4. $\langle x, 2z\rangle$.
5. $\|x\|$.
6. $\|y\|$.

6. Considere $\mathbb{R}^2$ com o produto interno canônico. Sejam $x = (1, 3)$ e $y = (2, -1)$. Encontre o vetor $z \in \mathbb{R}^2$ tal que $\langle x, z\rangle = 1$ e $\langle y, z\rangle = -1$.

7. Sejam $L = \mathbb{C}^2$, $x = (i, 2) \in \mathbb{C}^2$ e $y = (0, 3) \in \mathbb{C}^2$. Determine $z \in \mathbb{C}^2$, tal que $\langle x, z\rangle = 2$ e $\langle y, z\rangle = 2i$.

8. Seja L um $\mathbb{R}$-espaço vetorial e sejam $x, y \in L$, tais que $\|x\| = \|y\| = 1$ e $\|x - y\| = 2$. Determine $\langle x, y\rangle$.

9. Considere $L = \mathbb{C}[x]$ e sejam $f(t) = t^2 - it$ e $g(t) = t^3 + it^2 - (2 + 5i)t + 3$. Considere em L o produto interno,

$$\langle f, g\rangle = \int_0^1 f(x)g(x)dx.$$

1. Calcule $\langle f, g\rangle$.
2. Calcule $\|f\|$.
3. Calcule $\|g\|$.
4. Normalize f e g.

10. Sabendo que $\|l\| = 3$ e $\|m\| = 5$, com l e m vetores de um espaço euclidiano, determine $\alpha \in \mathbb{R}$ tal que $\langle l + \alpha m, l - \alpha m\rangle = 0$.

11. Seja L um $\mathbb{K}$-espaço vetorial com produto interno $\langle\ ,\ \rangle$. Demonstre que:

a) $\langle 0, l\rangle = \langle l, 0\rangle = 0$, para todo $l \in L$.

b) Se $\langle l, m\rangle = 0$, para todo $m \in L$, então $l = 0$.

12. Sejam L um $\mathbb{C}$-espaço vetorial e $l_1, l_2 \in L$. Demonstre a identidade de polarização

$$\langle l_1, l_2 \rangle = \frac{1}{4}\|l_1 + l_2\|^2 - \frac{1}{4}\|l_1 - l_2\|^2 + \frac{i}{4}\|l_1 + il_2\|^2 - \frac{i}{4}\|l_1 - il_2\|^2$$

13. Seja L um $\mathbb{C}$-espaço vetorial com produto interno $\langle\ ,\ \rangle$. Sejam $x, y \in L$, mostre que vale a lei do paralelogramo:

$$\|u + v\|^2 + \|u - v\|^2 = 2(\|u\|^2 + \|v\|^2).$$

14. Seja L um $\mathbb{R}$-espaço vetorial com produto interno. Mostre que $\forall x, y \in L$ vale

$$\langle x, y \rangle \leqslant \frac{\|x\|^2 + \|y\|^2}{2}.$$

15. Prove que em qualquer espaço vetorial complexo com produto interno vale

$$|\|l\| - \|m\|| \leqslant \|l - m\|.$$

16. Seja S um conjunto de geradores do espaço vetorial com produto interno L. Se os vetores $l, m \in L$ são tais que $\langle l, n \rangle = \langle m, n \rangle$, para todo $n \in S$, então prove que $l = m$.

17. Sejam $A \in \mathbb{R}^{2\times 2}$ e $X, Y \in \mathbb{R}^{2\times 1}$, e considere a aplicação $f_A(X, Y) = Y^T AX$. Mostre que f_A é um produto interno em $\mathbb{R}^{2\times 1}$ se, e somente se, $A = A^T$ e $\det(A), a_{11}, a_{22}$ são todos positivos.

18. Seja L um espaço vetorial real com produto interno. Demonstre que, $\forall\, l, m \in L$, $\|l\| = \|m\|$ se, e somente se, $\langle l + m, l - m \rangle = 0$.

19. Sejam l e m vetores de um espaço vetorial euclidiano. Prove que $\langle l, m \rangle = 0$ se, e somente se, $\|l + \alpha m\| \geqslant \|l\|$, $\forall\, \alpha \in \mathbb{R}$.

20. Demonstre que $\|\cdot\|_1$, $\|\cdot\|_2$ e $\|\cdot\|_\infty$ são normas.

21. Seja $x = (x_1, x_2, x_3, x_4) \in \mathbb{R}^4$. Demonstre que

$$\|x\| = 2\,|x_1| + \sqrt{3\,|x_2|^2 + \max|x_3|\,, 2\,|x_4|^2}$$

define uma norma em $\mathbb{R}^4$.

22. Seja $\|\cdot\|$ uma norma em $\mathbb{K}^n$. Mostre que $|\,\|x\| - \|y\|\,| \leqslant \|x - y\|$, para todos $x, y \in \mathbb{K}^n$.

23. Mostre que se $x \in \mathbb{K}^n$, então $\lim\limits_{p\to\infty} \|x\|_p = \|x\|_\infty$.

24. Sejam $x, y \in \mathbb{R}^n$ e defina $\varphi : \mathbb{R} \to \mathbb{R}$ por $\varphi(\alpha) = \|x - \alpha y\|_2$. Demonstre que φ é minimizada por $\alpha = \dfrac{x^T y}{y^T y}$.

25. Mostre que em $\mathbb{K}^n$, $\lim_{k\to\infty} x^{(k)} = x$ se, e somente se, $\lim_{k\to\infty} x_j^{(k)} = x_j$, para cada $j = 1, \ldots, n$.

26. $\|I\| = 1$ para qualquer norma de matriz? Justifique.

27. Sejam $A \in \mathbb{K}^{n\times n}$ não singular e $\|\cdot\|$ uma norma submultiplicativa. Demonstre que $\|A\|^{-1} \leqslant \|A^{-1}\|$.

28. Sejam $A, B \in \mathbb{K}^{n\times n}$ com A inversível e $\|\cdot\|$ uma norma matricial submultiplicativa. Demonstre que, se $\left\|A^{-1}B\right\| < 1$, então

1. $\|I + A^{-1}B\| \leqslant (1 + \|A^{-1}B\|)$.
2. $\|(I + A^{-1}B)^{-1}\| \leqslant \dfrac{1}{1 - \|A^{-1}B\|}$.

29. Demonstre que $\|\cdot\|_F$ e $\|\cdot\|_p$ satisfazem os axiomas de norma em $\mathbb{K}^{m\times n}$.

30. Sejam $A \in \mathbb{R}^{n\times n}$ invertível e $B \in \mathbb{R}^{n\times n}$ tais que vale $\|B - A\| \leq \|A^{-1}\|^{-1}$. Prove que B é invertível.

31. Seja $A \in \mathbb{R}^{n\times n}$, tal que $\|A\| < 1$. Demonstre que $I - A$ é não singular.

32. Seja $\|\cdot\|$ uma norma arbitrária em $\mathbb{K}^n$ e $P \in \mathbb{K}^{n\times n}$ uma matriz não singular e defina a norma $\|x\|_\alpha = \|Px\|$. Demonstre que, $\|A\|_\alpha = \left\|PAP^{-1}\right\|$.

33. Sejam $A \in \mathbb{K}^{n\times n}$. Demonstre que $\|A^*A\|_F \leqslant \|A\|_F^2$.

34. Demonstre que se L é uma matriz triangular inferior com diagonal formada por elementos reais, então a desigualdade

$$\sqrt{2}\|L\|_F \leqslant \|L + L^*\|_F.$$

35. Seja $A \in \mathbb{K}^{m\times n}$ de posto p. Demonstre que,

1. $\|A\|_2 \leqslant \|A\|_F \leqslant \sqrt{p}\|A\|_2$.
2. $\|A\|_2 \leqslant \sqrt{\|A\|_1 \|A\|_\infty}$.

36. Sejam $A \in \mathbb{K}^{m\times r}$ e $B \in \mathbb{K}^{r\times n}$. Demonstre a validade da desigualdade $\|AB\|_F \leqslant \|A\|_2\|B\|_F$.

37. Mostre que se $0 \neq x \in \mathbb{K}^n$ e $A \in \mathbb{K}^{n\times n}$, então

$$\left\|A\left(I - \frac{xx^*}{x^*x}\right)\right\|_F = \|A\|_F - \frac{\|Ax\|_2^2}{x^*x}.$$

38. Suponha que $x \in \mathbb{K}^m$ e $y \in \mathbb{K}^n$. Mostre que se $E = xy^*$, então

1. $\|E\|_F = \|E\|_2 = \|x\|_2 \|y\|_2$.
2. $\|E\|_\infty \leqslant \|x\|_\infty \|y\|_1$.

39. Seja $L = \mathbb{C}^{2\times 2}$ com o produto interno $\langle A, B\rangle = \operatorname{tr}(AB^*)$. O subconjunto

$$S = \left\{ \begin{pmatrix} i & 0 \\ 0 & 0 \end{pmatrix}, \begin{pmatrix} 0 & i \\ 0 & 0 \end{pmatrix} \begin{pmatrix} 0 & 0 \\ i & 0 \end{pmatrix}, \begin{pmatrix} 0 & 0 \\ 0 & i \end{pmatrix} \right\}$$

é ortonormal?

40. Considere o espaço vetorial $\mathbb{R}^2$ com o produto interno $\langle l, m\rangle = x_1y_1 + 2x_2y_2$, para $l = (x_1, x_2)$ e $m = (y_1, y_2)$. Verifique se os vetores abaixo são ortogonais em relação a esse produto interno.

1. $l = (1, 1)$ e $m = (2, -1)$.
2. $l = (2, 1)$ e $m = (-1, 1)$.
3. $l = (3, 2)$ e $m = (2, -1)$.

41. Considere o espaço vetorial $\mathbb{R}^2$. Dê um exemplo de vetores linearmente independentes que não são ortogonais e um outro exemplo de vetores ortogonais que não são linearmente independentes.

42. Seja L um espaço vetorial com produto interno. Dados $l, m \in L$ com m não nulo e $\lambda = \dfrac{\langle l, m\rangle}{\|m\|^2}$, mostre que $l - \lambda m$ é ortogonal a m.

43. Seja $M = \{(x, y, z) \in \mathbb{R}^3 \mid 2x - 3y = 0\}$. Determine uma base ortonormal de M, considerando o produto interno canônico de $\mathbb{R}^3$.

44. Determine $a \in \mathbb{R}$ tal que os vetores $l = (1, a+1, a)$ e $m = (a-1, a, a+1)$ sejam ortogonais em $\mathbb{R}^3$, considerando o produto interno canônico de $\mathbb{R}^3$.

45. Sejam $x = (x_1, x_2, x_3)$ e $y = (y_1, y_2, y_3)$ vetores em $\mathbb{R}^3$. O produto vetorial de x por y é definido como o vetor

$$x \times y = (x_2y_3 - x_3y_2, x_3y_1 - x_1y_3, x_1y_2 - x_2y_1)\,.$$

Prove que valem as seguintes propriedades:

1. $x \times y = -y \times x$.
2. $x \times (y + y') = x \times y + x \times y'$.
3. $x \times (\alpha y) = \alpha(x \times y)$, $\alpha \in \mathbb{R}$.
4. $x \times y = 0$ se, e somente se, $\{x, y\}$ é linearmente dependente.
5. $x \times y$ é ortogonal a x e a y.

46. Seja L um espaço vetorial com produto interno de dimensão finita e base ortonormal $\{l_1, \ldots, l_n\}$. Mostre que, para quaisquer vetores $l, m \in L$,

$$\langle l, m\rangle = \sum_{i=1}^{n} \langle l, l_i\rangle \overline{\langle m, l_i\rangle}.$$

47. Seja $\mathcal{B} = \{l_1, \ldots, l_n\}$ um conjunto ortogonal de vetores não nulos em um espaço vetorial L com produto interno. Seja $l \in L$.

1. Demonstre a desigualdade de Bessel:

$$\sum_{i=1}^{n} \frac{|\langle l, l_i \rangle|^2}{\|l_i\|^2} \leqslant \|l\|^2 .$$

2. Mostre que a igualdade (chamada de identidade de Parseval) vale se, e somente se,

$$l = \sum_{i=1}^{n} \frac{\langle l, l_i \rangle}{\|l_i\|^2} l_i,$$

ou seja, $\mathcal{B}$ é uma base de L.

48. Seja L um espaço vetorial com produto interno. Para quaisquer vetores $l, m \in L$, prove que $\|l\|m + \|m\|l$ e $\|l\|m - \|m\|l$ são ortogonais.

49. Seja L um $\mathbb{K}$-espaço vetorial com produto interno e sejam $l, m \in L$.

1. Mostre que se $l \perp m$, então $\|l + m\|^2 = \|l\|^2 + \|m\|^2$.
2. Para $\mathbb{K} = \mathbb{R}$, mostre que se $\|l + m\|^2 = \|l\|^2 + \|m\|^2$, então $l \perp m$.
3. Mostre que o item anterior é falso para $\mathbb{K} = \mathbb{C}$.
4. Para $\mathbb{K} = \mathbb{C}$, mostre: $\|\alpha l + \beta m\|^2 = \|\alpha l\|^2 + \|\beta m\|^2$ para todos $\alpha, \beta \in \mathbb{C}$ se, e somente se, $l \perp m$.

50. Seja L um espaço com produto interno. A distância entre dois vetores $l, m \in L$ é definida por

$$d(l, m) \coloneqq \|l - m\| .$$

Mostre que,

1. $d(l, m) \geqslant 0$.
2. $d(l, m) = 0$ se, e somente se, $l = m$.
3. $d(l, m) = d(m, l)$.
4. $d(l, m) \leqslant d(l, n) + d(n, m)$, para todo $n \in L$.

51. Seja L um espaço com produto interno. Mostre que $l = m$ se, e somente se, $\langle l, n \rangle = \langle m, n \rangle$, para todo $n \in L$.

52. Seja L um $\mathbb{K}$-espaço vetorial com produto interno e $C \subseteq L$ um conjunto convexo e seja $l \in L$ fora de C, isto é, $l \notin C$. Se existirem $x_1, x_2 \in C$, tais que para todo $x \in C$,

$$\|l - x_1\| \leq \|l - x\| \text{ e } \|l - x_2\| \leq \|l - x\| ,$$

então $x_1 = x_2$.

53. Seja $\mathbb{B} = \{[1,2,1]^T, [1,0,1]^T, [1,2,0]^T\}$ uma base de $\mathbb{R}^3$. Use processo de ortogonalização de Gram-Schmidt clássico e modificado para encontrar um base ortonormal de $\mathbb{R}^3$ a partir de $\mathbb{B}$, considerando o produto interno canônico de $\mathbb{R}^3$.

54. Seja $\mathbb{B} = \{[1,0,1,-2]^T, [1,-1,1,2]^T, [1,2,-1,0]^T, [-1,1,0,1]^T\}$ uma base de $\mathbb{R}^4$. Use processo de ortogonalização de Gram-Schmidt clássico e modificado para encontrar um base ortonormal de $\mathbb{R}^4$ a partir de $\mathbb{B}$, considerando o produto interno canônico de $\mathbb{R}^4$.

55. Considere a base $\mathbb{B} = \{[1,i]^*, [i,1]^*\}$ de $\mathbb{C}^2$. Use processo de ortogonalização de Gram-Schmidt clássico e modificado para encontrar um base ortonormal de $\mathbb{C}^3$ a partir de $\mathbb{B}$, considerando o produto interno canônico de $\mathbb{C}^2$.

56. Seja $B = span\{[1+i, 1-3i, 2+i]^*, [1-i, 3+2i, 1-i]^*\} \subseteq \mathbb{C}^3$. Use processo de ortogonalização de Gram-Schmidt clássico e modificado para encontrar um base ortonormal do subespaço vetorial B, considerando o produto interno canônico de $\mathbb{C}^3$.

57. Ortonormalize o conjunto de vetores

$$\{[1,1,0,2]^T, [1,-1,1,2]^T, [2,0,1,4]^T\} \subseteq \mathbb{R}^4.$$

O resultado muda após uma permutação dos vetores?

58. Determine uma base ortonormal para o subespaço vetorial

$$M = \{(x,y,z,w) \in \mathbb{R}^4 \,|\, x+y=1\}.$$

59. Encontre uma base do subespaço vetorial $M^\perp$, onde M é o subespaço vetorial de $\mathbb{R}^4$ gerado pelo vetores $(1,0,1,1)$ e $(1,1,2,0)$. Ortonormalize essa base. Considere o produto interno canônico.

60. Encontre uma base ortonormal para $l^\perp$ em $\mathbb{C}^3$, onde $l = (1, i, 1+i)$. Considere o produto interno canônico.

61. Considere $\mathbb{R}^4$ com o produto interno canônico. Encontre uma base ortogonal para M e uma base para $M^\perp$, onde

$$M = \{(x,y,z,w) \in \mathbb{R}^4 \,|\, x+y+2z-w=0\}.$$

62. Considere $\mathbb{R}^3$ com o produto interno

$$\langle (x_1,y_1,z_1),(x_2,y_2,z_2)\rangle = x_1x_2 + 2y_1y_2 + z_1z_2.$$

Encontre uma base do subespaço ortogonal a

$$M = span\{(1,1,1),(0,-2,-3)\}.$$

63. Considere $\mathbb{R}_2[x]$ com o produto interno

$$\langle p, q\rangle = \int_0^1 p(x)q(x)\,dx.$$

Determine uma base ortonormal de $M^\perp = span\{1, x^2+3\}$.

64. Seja $L = \mathscr{C}([-1,1],\mathbb{R})$ com o produto interno

$$\langle f, g\rangle = \int_{-1}^1 f(x)g(x)\,dx.$$

Seja $M \in L$ o subespaço vetorial das funções ímpares. Determine $M^\perp$.

65. Seja $L = \mathbb{C}^{n\times n}$ com o produto interno $\langle A, B\rangle = \operatorname{tr}(AB^*)$. Encontre o complemento ortogonal do subespaço vetorial das matrizes diagonais.

66. Demonstre a identidade de Apolônio: Seja L um $\mathbb{K}$-espaço vetorial com produto interno então, para todos $l, m, n \in L$,

$$\|l-m\|^2 + \|l-n\|^2 = \frac{1}{2}\|m-n\|^2 + 2\left\|l - \frac{1}{2}(m+n)\right\|^2.$$

67. Sejam L um $\mathbb{K}$-espaço vetorial com produto interno e M, N subespaços vetoriais de L, tais que $M \subseteq N^\perp$ e $L = M + N$. Mostre que $M = N^\perp$.

68. Sejam L um espaço vetorial com produto interno de dimensão finita e $\mathbb{B} = \{l_1, \ldots, l_n\}$ uma base ortonormal de L. Se $T \in \mathcal{L}(L, L)$ e $A = [T]_{\mathbb{B}}$, mostre que $a_{ij} = \langle T(l_j), l_i\rangle$, $i, j = 1, \ldots, n$.

69. Sejam M_1, M_2 subconjuntos de um espaço vetorial L. Demonstre que se $M_1 \subseteq M_2$, então $M_2^\perp \subseteq M_1^\perp$.

70. Sejam L um $\mathbb{K}$-espaço vetorial com produto interno e S um subconjunto de L.

1. Mostre que $span\, S \subseteq \left(S^\perp\right)^\perp$.
2. Se L tem dimensão finita, mostre que $\left(S^\perp\right)^\perp = span\, S$.

71. Sejam M e N subespaços vetoriais do $\mathbb{K}$-espaço vetorial L de dimensão finita com produto interno. Demonstre que

1. $(M+N)^\perp = M^\perp \cap N^\perp$.
2. $(M \cap N)^\perp = M^\perp + N^\perp$.

72. Sejam L um $\mathbb{C}$-espaço vetorial com produto interno e M um subespaço vetorial de L. Suponha que $l \in L$ é um vetor que satisfaz $\langle l, m\rangle + \langle m, l\rangle \leqslant \langle m, m\rangle$, $\forall m \in M$. Prove que $l \in M^\perp$.

73. Sejam L um $\mathbb{K}$-espaço vetorial, tal que $L = M_1 \oplus M_2$. Sejam f_1 e f_2 os produtos internos de M_1 e M_2, respectivamente. Mostre que existe um único produto interno f em L, tal que

1. $M_2 = M_1^{\perp}$, e
2. $f(\alpha, \beta) = f_i(\alpha, \beta)$, com $\alpha, \beta \in M_i$, $i = 1, 2$.

74. Seja M um subespaço vetorial de um $\mathbb{K}$-espaço vetorial L com produto interno e de dimensão finita. Mostre que, cada classe de equivalência em L/M contém exatamente um vetor que é ortogonal à M.

75. Considere $\mathbb{R}_3[x]$ com o produto interno

$$\langle p, q \rangle = \sum_{k=-2}^{1} p(k)q(k).$$

Calcule $\text{proj}_{\mathbb{R}_1[x]}(x^2 - 1)$.

76. Considere $\mathscr{C}([0, 2\pi], \mathbb{R})$ com o produto interno

$$\langle f, g \rangle = \int_0^{2\pi} f(t)g(t)\, dt.$$

Determine a função de $M = span\{1, \operatorname{sen} t, \cos t\}$ que melhor aproxima $f(t) = t - 1$ em $[0, 2\pi]$.

77. Considere $\mathbb{R}^{2\times 2}$ com o produto interno canônico. Seja

$$M = span\left\{ \begin{pmatrix} -1 & 0 \\ 3 & -1 \end{pmatrix}, \begin{pmatrix} 7 & 0 \\ 0 & 4 \end{pmatrix} \right\}.$$

Determine o vetor de M que melhor aproxima

$$X = \begin{pmatrix} 0 & -1 \\ 0 & -1 \end{pmatrix}.$$

78. Sejam L um espaço euclidiano de dimensão finita e M um subespaço vetorial de L. Mostre que, $\forall\, l, m \in L$,

$$\langle \text{proj}_M\, l, m \rangle = \langle l, \text{proj}_M\, m \rangle .$$

79. Seja $A \in \mathbb{K}^{m\times n}$ uma matriz de posto completo. Demonstre que a matriz $P = A(A^*A)^{-1}A^*$ é um projetor ortogonal.

80. Sejam

$$A = \begin{bmatrix} 1 & 1 & 3 \\ 0 & 2 & -1 \\ -1 & -1 & 0 \\ 0 & 2 & 2 \end{bmatrix} \quad \text{e} \quad B = \begin{bmatrix} 1 & 1 & 3 \\ 0 & 2 & -1 \\ -1 & -1 & 0 \\ 0 & 2 & 3 \end{bmatrix}.$$

1. Encontre o projetor ortogonal P em $\mathrm{Im}(A)$ e calcule $P(-1,0,1,2)$.
2. Encontre o projetor ortogonal P em $\mathrm{Im}(B)$ e calcule $P(-1,0,1,2)$.

81. Seja P uma projeção não nula. Demonstre que $\|P\|_2 \geqslant 1$ e que $\|P\|_2 = 1$ se, e somente se, P é uma projeção ortogonal.

82. Seja $A \in \mathbb{R}^{m\times n}$, com $m \geqslant n$. Demonstre que $A^T A$ é não singular se, e somente se, A tem posto completo.

83. Sejam $P_1, P_2 \in \mathbb{R}^{n\times n}$ projeções sobre subespaços distintos. $P_1 + P_2$ é uma projeção? Justifique.

84. Seja $P \in \mathbb{R}^{n\times n}$ um projetor. Demonstre que $I + P$ é não singular e encontre $(I+P)^{-1}$.

85. Demonstre que se $P \in \mathbb{R}^{n\times n}$ é um projetor ortogonal, então $I - 2P$ é uma matriz ortogonal.

86. Demonstre que o conjunto dos valores singulares de uma projeção ortogonal são $\{0,1\}$.

87. Seja $P \in \mathbb{K}^{n\times n}$ um projetor ortogonal. Demonstre que $\|P\|_2 = 1$.

88. Mostre que $A \in \mathbb{R}^{n\times n}$ é simétrica definida positiva se, e somente se, sua decomposição SVD é da forma $A = V\Sigma V^T$, com Σ não singular.

89. Demonstre que se $A \in \mathbb{R}^{n\times n}$ é uma matriz simétrica definida positiva e $\alpha > -\sigma_n$, então $A + \alpha I$ é também simétrica definida positiva com valores singulares $\sigma_j + \alpha$, $j = 1, \ldots, n$.

90. Sejam $A \in \mathbb{R}^{n\times n}$ e $\alpha > 0$. Encontre uma expressão para os valores singulares de $(A^T A + \alpha I)^{-1} A^T$ em termos de α e dos valores singulares de A.

91. Considere a seguinte matriz real

$$A = \begin{pmatrix} 1 & 2 & -2 & 1 \\ 0 & 3 & 2 & 1 \\ -1 & -3 & 2 & 0 \\ 1 & -2 & 2 & -1 \end{pmatrix}.$$

1. Encontre uma base de $\mathrm{Ker}\,(A^*)$.
2. Encontre uma base de $\mathrm{Ker}\,(A^*)^\perp$.
3. Encontre uma base para o subespaço gerado pelas colunas de A.
4. Há alguma relação entre os itens 1, 2 e 3? Justifique.
5. Utilize a Alternativa de Fredholm para determinar as condições para $Ax = b$ ser consistente.

92. Para cada matriz abaixo encontre as condições sobre b para que o sistema linear $Ax = b$ seja consistente.

1. $\begin{bmatrix} 1 & -1 & 0 \\ -1 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$ 2. $\begin{bmatrix} 1 & -1 & 0 \\ -1 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}$ 3. $\begin{bmatrix} 1 & -1 & 0 \\ -1 & -1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$.

93. Demonstre a equivalência entre o Teorema 1.8 e a Observação 1.8.

94. Seja $V \in \mathbb{K}^n$ um subespaço vetorial gerado pelos vetores $\{v_1, \ldots, v_n\}$. Demonstre que $x \in V^\perp$ se, e somente se, $x \in \operatorname{Ker}(A^*)$, onde $A = [v_1, \ldots, v_n]$.

95. Demonstre que:

1. $\operatorname{Ker}(A) = \operatorname{Ker}(A^*A)$
2. $\operatorname{Ker}(A^*) = \operatorname{Ker}(AA^*)$
3. $\operatorname{Im}(A) = \operatorname{Im}(AA^*)$
4. $\operatorname{Im}(A^*) = \operatorname{Im}(A^*A)$
5. $\operatorname{rank}(A) = \operatorname{rank}(AA^*)$
6. $\operatorname{rank}(A) = \operatorname{rank}(A^*A)$.

96. Seja $A \in \mathbb{K}^{m\times n}$ uma matriz auto-adjunta. Demonstre que o sistema $Ax = b$ é consistente se, e somente se, b é ortogonal a $\operatorname{Ker}(A)$.

97. Seja $A \in \mathbb{K}^{m\times n}$. Prove que se $\{x_1, \ldots, x_r\}$ formam uma base de $\operatorname{Im}(A^*)$, então $\{Ax_1, \ldots, Ax_r\}$ formam uma base de $\operatorname{Im}(A)$.

98. Demonstre que um sistema linear $Ax = b$ consistente tem uma única solução $y \in \operatorname{Coim}(A)$ satisfazendo $Ay = b$. A solução geral do sistema linear $Ax = b$ é $x = y+z$, com $z \in \operatorname{Ker}(A)$. Ademais, a solução particular w é a solução de menor norma euclidiana entre todas as soluções do sistema linear $Ax = b$.

Capítulo 2

Quadrados Mínimos

Neste capítulo introduziremos o conceito de quadrados mínimos, além de demonstrarmos algumas propriedades. A disputa sobre a "paternidade" da ideia de quadrados mínimos é um pouco controversa. Em 1805 Adrien-Marie Legendre publica um livro [87] onde descreve de forma inédita, clara e concisa o método dos quadrados mínimos, enquanto em 1808/1809 um matemático americano chamado Robert Adrian "descobre" tais ideias naquele livro.

Porém em 1809, Johann Carl Friedrich Gauss publica um trabalho sobre o cálculo de órbitas celestes, descrevendo o método em questão e declara ser o inventor do método de quadrados mínimos, pois o havia desenvolvido em 1795 [143]. Uma análise histórica sobre a controvérsia entre Legendre e Gauss pode ser encontrada em [113], já uma referência técnica sobre o desenvolvimento do método de quadrados mínimos é [100].

2.1 O Problema de Quadrados Mínimos Lineares

O método de quadrados mínimos é uma ferramenta indispensável nas ciências aplicadas. Esse método posto em linguagem de álgebra linear é o arcabouço para a determinação de uma "solução" de um sistema linear sobredeterminado $Ax = b$, ou seja, um sistema linear em que $A \in \mathbb{K}^{m\times n}$ tem mais linhas (equações) que colunas (incógnitas).

O estudo dos problemas de quadrados mínimos é muito recorrente e tem uma vasta produção bibliográfica associada, como [12, 58, 86, 100]. Nessa seção pretendemos dar uma breve introdução ao assunto e, portanto, trabalhamos seus aspectos mais básicos. Utilizaremos a expressão quadrados mínimos ao conceito de quadrados mínimos lineares, visto que quadrados mínimos não lineares não fazem parte do escopo desse livro.

A ideia que está por trás desse método é "resolver" um sistema linear $Ax = b$ minimizando o resíduo $r = b - Ax$ na chamada 2-norma, ou seja, a norma obtida do produto interno canônico em $\mathbb{R}^n$. Mais a frente veremos as ideias geométricas que fundamentam a escolha da solução de quadrados mínimos como sendo aquela que minimiza o resíduo.

Passemos, agora, à formalização do conceito de quadrados mínimos. Considere um sistema linear $Ax = b$ com $A \in \mathbb{R}^{m\times n}$ e $m > n$. Em geral esse problema não tem solução, pois uma solução para o problema existe somente se $b \in \mathrm{Im}(A)$. Mas, como $b \in \mathbb{R}^m$ e a imagem de A é um subespaço de no máximo dimensão n, essa escolha de b é bem difícil.

Na impossibilidade de sempre se ter $b \in \mathrm{Im}(A)$, chamamos de solução do sistema linear sobredeterminado o vetor $\hat{x}$ de tal forma que $b - A\hat{x}$ tenha a menor norma possível. Formalizando matematicamente, pede-se que o resíduo

$$r = b - Ax \in \mathbb{R}^m$$

tenha a menor norma possível. Assim, a solução de quadrados mínimos é o vetor $\hat{x} \in \mathbb{R}^n$ que minimiza

$$\|b - Ax\|_2 .$$

Isso indica que a solução de quadrados mínimos é o vetor $\hat{x}$ com $A\hat{x} \in \mathrm{Im}(A)$ sendo o vetor mais próximo de b com relação a essa norma. Caso $b \in \mathrm{Im}(A)$, então $\hat{x}$ é de fato a solução do sistema linear, pois nesse caso o mínimo da norma do resíduo é zero.

Vejamos um exemplo de como surgem problemas que culminam na resolução de um sistema linear sobredeterminado.

Exemplo 2.1. *[147, pág. 79] Suponha que sejam dados m pares ordenados*

$$(x_1, y_1), \ldots, (x_m, y_m) \in \mathbb{R}^2,$$

com $x_i \neq x_j$ para $i \neq j$. Esses pontos representam, por exemplo, medições físicas nos pontos x_i. Queremos encontrar um polinômio de grau $n-1$

$$p(x) = a_0 + a_1 x + a_2 x^2 + \cdots + a_{n-1} x^{n-1}, \;\; n < m,$$

que descreva, de maneira satisfatória, as medições coletadas. Para isso, vamos determinar os coeficientes do polinômio de forma que minimizem a soma dos quadrados da diferença entre o valor de $p(x_i)$ e y_i medido, ou seja, estamos interessados em

$$\min \sum_{i=1}^{m} |p(x_i) - y_i|^2. \tag{2.1.1}$$

O problema pode ser reescrito matricialmente como,

$$\begin{bmatrix} 1 & x_1 & \cdots & x_1^{n-1} \\ 1 & x_2 & \cdots & x_2^{n-1} \\ 1 & x_3 & \cdots & x_3^{n-1} \\ \vdots & \vdots & & \vdots \\ 1 & x_m & \cdots & x_m^{n-1} \end{bmatrix} \begin{bmatrix} a_0 \\ a_1 \\ \vdots \\ a_{n-1} \end{bmatrix} \approx \begin{bmatrix} y_1 \\ y_2 \\ y_3 \\ \vdots \\ y_m \end{bmatrix}.$$

Note que, a soma dos quadrados $|p(x_i) - y_i|^2$ em (2.1.1) é igual ao quadrado da 2-norma do resíduo, $\|r\|_2^2$, desse sistema linear, e a matriz do lado esquerdo é conhecida como matriz de Vandermonde.
Exemplificando, considere que tenhamos as seguintes 15 medições

1. $(1, 1.03781)$
2. $(2, 1.87969)$
3. $(3, 2.65018)$
4. $(4, 0.47452)$
5. $(5, 1.45103)$
6. $(6, 2.23287)$
7. $(7, 0.89296)$
8. $(8, 0.39126)$
9. $(9, 0.29256)$
10. $(10, 1.71118)$
11. $(11, 0.49072)$
12. $(12, 0.29978)$
13. $(13, 2.35501)$
14. $(14, 0.63270)$
15. $(15, 2.43428)$.

Suponha que queiramos encontrar um polinômio de grau 10 *que descreva no sentido de quadrados mínimos os pontos experimentais acima. Na Figura 2.1 mostramos os pontos e o polinômio de grau* 10.

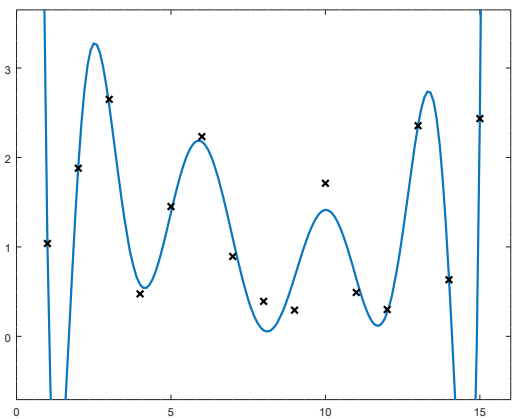

Figura 2.1: Polinômio de grau 10 que aproxima, no sentido quadrados mínimos, os pontos experimentais dados.

Nesse exemplo, apenas apresentamos a solução de quadrados mínimos do problema (Figura 2.1) e não dissemos como ela foi calculada. A chave para encontrar a solução de quadrados mínimos é a projeção ortogonal.

Primeiramente note que, em geral, os problemas de quadrados mínimos não cumprem a Alternativa de Fredholm (Teorema 1.8). Como já dito, o objetivo é encontrar um vetor $\hat{x}$ de forma que $A\hat{x} \in \mathrm{Im}(A)$ esteja o mais próximo possível de b, e nesse caso, a norma do resíduo $b - Ax$ é minimizada. Isso claramente acontece quando projeta-se b ortogonalmente na imagem da matriz A. A Figura 2.2 mostra, esquematicamente, a projeção ortogonal de b sobre $\mathrm{Im}(A)$.

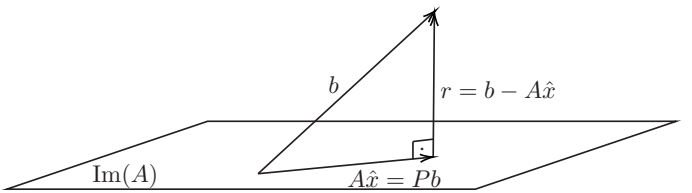

Figura 2.2: Ilustração para o problema de quadrados mínimos em termos de projeção ortogonal.

Chamando de P a matriz projeção ortogonal, então a solução de quadrados mínimos de $Ax = b$ é o vetor $\hat{x} \in \mathbb{R}^n$, tal que $A\hat{x} = Pb$. Para demonstrar esse fato, precisaremos dos seguintes resultados auxiliares: o primeiro é o Teorema 1.7 e, o segundo, o Lema 1.1, que enunciamos novamente para fins didáticos.

Lema 2.1. *Seja $A \in \mathbb{R}^{m\times n}$ uma matriz arbitrária. Então* $\mathrm{Im}(A)^\perp = \mathrm{Ker}\,(A^T)$ *e* $\mathrm{Ker}\,(A)^\perp = \mathrm{Im}(A^*)$.

Lema 2.2. *Uma matriz $A \in \mathbb{K}^{m\times n}$ tem posto completo se, e somente se, A^TA é não singular.*

Teorema 2.1. *[147, pág. 80] Sejam $A \in \mathbb{R}^{m\times n}$ uma matriz que tem posto completo ($m \geqslant n$) e $b \in \mathbb{R}^m$. Um vetor $x \in \mathbb{R}^n$ minimiza a norma do resíduo $\|r\|_2 = \|b - Ax\|_2$ se, e somente se, $r \perp \mathrm{Im}(A)$, ou seja,*

$$A^Tr = 0, \tag{2.1.2}$$

ou equivalentemente,

$$A^TAx = A^Tb \tag{2.1.3}$$

ou equivalentemente,

$$Pb = Ax \tag{2.1.4}$$

onde $P \in \mathbb{R}^{m\times m}$ é o projetor ortogonal sobre $\mathrm{Im}(A)$.

Demonstração. Como A, por hipótese, tem posto completo, então pelo Lema 2.2, A^TA admite inversa. Comecemos demonstrando as equivalências. A equivalência entre (2.1.2) e (2.1.3) segue da definição do resíduo r,

$$A^Tr = 0 \;\Leftrightarrow\; A^T(b - Ax) = 0 \;\Leftrightarrow\; A^TAx = A^Tb.$$

Já a equivalência entre (2.1.2) e (2.1.4) segue das propriedades de projeção ortogonal:

$$\begin{aligned} A^Tr = 0 \;&\Rightarrow\; A^Tb = A^TAx \;\Rightarrow\; A(A^TA)^{-1}A^Tb = A(A^TA)^{-1}A^TAx \\ &\Rightarrow\; Pb = Ax. \end{aligned}$$

Por outro lado,

$$Pb = Ax \;\Rightarrow\; Pb = A[(A^TA)^{-1}(A^TA)]x \;\Rightarrow\; Pb = PAx \;\Rightarrow\; Pr = 0$$

$$\Rightarrow\; 0 = A^TPr = A^T\left[A(A^TA)^{-1}A^T\right]r = A^Tr.$$

Agora, suponha que $r \perp \mathrm{Im}(A)$, pelo Lema 2.1, $r \in \mathrm{Ker}\,(A^T)$, ou seja, $A^Tr = 0$. Como $A^Tr = 0$ é equivalente a $Pb = Ax$, suponha $\hat{y} = Pb \in \mathrm{Im}(A)$ e tome $z \in \mathrm{Im}(A)$ que seja distinto de $\hat{y}$. Como $z - \hat{y}$ é ortogonal a $b - \hat{y}$,

$$\|b - z\|_2^2 = \|b - \hat{y}\|_2^2 + \|\hat{y} - z\|_2^2 > \|b - \hat{y}\|_2^2\,.$$

Portanto, $\hat{y} \in \mathrm{Im}(A)$ é o único vetor que minimiza $\|b - y\|_2$, para todo $y \in \mathrm{Im}(A)$. Por outro lado, se $x \in \mathbb{R}^n$ minimiza a norma do resíduo $\|r\|_2 = \|b - Ax\|_2$, então por propriedades da projeção ortogonal, $(b - Pb) \perp \mathrm{Im}(A)$, ou seja, $r \perp \mathrm{Im}(A)$. □

Observação 2.1. *1. O sistema linear $n \times n$ (2.1.3) é conhecido pelo termo "equações normais". Portanto, como demonstrado, o sistema linear composto das equações normais é não singular se, e somente se, A tem posto completo, ou seja, a solução do problema de quadrados mínimos tem solução única se, e somente se, A tem posto completo.*

2. Caso A não tenha posto completo convencionamos de considerar a solução de mínima norma. Ao longo desse livro, focaremos no caso de A ter posto completo, embora esporadicamente falemos das soluções de mínima norma.

3. Em geral, a solução de quadrados mínimos é encontrada resolvendo-se o sistema linear $A^TAx = A^Tb$, em vez de $x = (A^TA)^{-1}A^Tb$.

□

Exemplo 2.2. *Encontre a solução de quadrados mínimos do sistema linear*

$$\begin{cases} x + 2y &= 5 \\ 3x + 4y &= 11 \\ 5x + 6y &= 17 \end{cases}$$

O sistema linear acima pode ser reescrito na forma matricial $A\hat{x} = b$, onde

$$A = \begin{bmatrix} 1 & 2 \\ 3 & 4 \\ 5 & 6 \end{bmatrix} \quad e \quad b = \begin{bmatrix} 5 \\ 11 \\ 17 \end{bmatrix}.$$

Note que, as colunas de A são linearmente independentes e logo A tem posto completo. Assim, $A\hat{x} = b$ admite uma única solução de quadrados mínimos

que pode ser determinada através da resolução do sistema linear $A^T A\hat{x} = A^T b$,

$$\begin{bmatrix} 35 & 44 \\ 44 & 56 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} 123 \\ 156 \end{bmatrix}$$

cuja solução é $x = 1$ *e* $y = 2$.
Para um sistema 2×2 *encontrar a solução das equações normais é simples. Porém, quando a dimensão aumenta o trabalho se torna hercúleo. Como* $A^T A$ *é auto-adjunta (hermitiana) e definida positiva*[1] *podemos aplicar a decomposição de Cholesky, isto é,* $A^T A = R^T R$, *onde* R *é triangular superior. Mas, mesmo assim, há limitações para essa abordagem. Para* $n \gg 1$ *a decomposição de Cholesky se torna bastante cara computacionalmente. Nesse caso, o métodos iterativos são de grande valor.*

Exemplo 2.3. *Encontre a solução de quadrados mínimos do sistema linear*

$$\begin{cases} x + 2y &= 1 \\ x + 2y &= 2 \end{cases}$$

O sistema linear acima pode ser reescrito na forma matricial $A\hat{x} = b$, *onde*

$$A = \begin{bmatrix} 1 & 2 \\ 1 & 2 \end{bmatrix} \quad e \quad b = \begin{bmatrix} 1 \\ 2 \end{bmatrix}.$$

Observe que esse problema não satisfaz a hipótese de A *ter posto completo, pois* $posto(A) = 1$. *Vejamos o que ocorre quando a hipótese de ser posto completo é quebrada. Para determinar a solução vamos resolver o sistema linear* $A^T A\hat{x} = A^T b$,

$$\begin{bmatrix} 2 & 4 \\ 4 & 8 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} 3 \\ 6 \end{bmatrix}.$$

O sistema acima tem um grau de liberdade, cuja solução parametrizada é

$$\hat{x} = \begin{bmatrix} \dfrac{3 - 4\lambda}{2} & \lambda \end{bmatrix}^T,$$

com $\lambda \in \mathbb{R}$. *Portanto, o sistema linear dado apresenta infinitas soluções de quadrados mínimos, e nesse caso, tomamos a solução de menor norma. Observe que o problema original é incompatível.*

Exemplo 2.4. *Considere o sistema linear do Exemplo 2.2. Podemos formular o problema de quadrados mínimos utilizando a fatoração* QR *da seguinte forma. Suponha que* $A \in \mathbb{R}^{m\times n}$, $m \geqslant n$, *tenha posto completo e, isso nos diz que existe a fatoração* $A = QR$, *com* $Q \in \mathbb{R}^{m\times n}$ *uma matriz com colunas ortonormais e* $R \in \mathbb{R}^{n\times n}$ *não singular. Sob estas condições,* $A^T A = R^T R$ *e o sistema linear* $A^T Ax = A^T b$ *tem solução*

$$\hat{x} = (A^T A)^{-1} A^T b \;\Rightarrow\; \hat{x} = (R^T R)^{-1}(QR)^T b \;\Rightarrow\; \hat{x} = R^{-1} Q^T b.$$

[1] Como há alguma confusão entre autores, alertamos que em nossa definição exigimos que a matriz seja simétrica.

Portanto, o sistema linear $A^TA\hat{x} = A^Tb$ é convertido em um sistema linear triangular superior $R\hat{x} = Q^Tb$. Utilizando o mesmo problema do Exemplo 2.2,

$$\begin{cases} u_1 &= a_1 &= [1,3,5]^T, \\ \\ u_2 &= a_2 - \dfrac{\langle a_2, u_1\rangle}{\|u_1\|^2}u_1 = [2,4,6]^T - \dfrac{44}{35}[1,3,5]^T = \left[\dfrac{26}{35}, \dfrac{8}{35}, -\dfrac{2}{7}\right]^T. \end{cases}$$

Logo,

$$Q = \begin{bmatrix} 1/\sqrt{35} & 13/\sqrt{210} \\ 3/\sqrt{35} & 4/\sqrt{210} \\ 5/\sqrt{35} & -5/\sqrt{210} \end{bmatrix}$$

e como $R = Q^TA$,

$$R = \begin{bmatrix} \sqrt{35} & 44/\sqrt{35} \\ 0 & 12/\sqrt{210} \end{bmatrix}.$$

Assim, o sistema linear $A^TA\hat{x} = A^Tb$ é convertido em um sistema linear triangular superior $R\hat{x} = Q^Tb$,

$$\begin{bmatrix} \sqrt{35} & 44/\sqrt{35} \\ 0 & 12/\sqrt{210} \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} 123/\sqrt{35} \\ 24/\sqrt{210} \end{bmatrix} \Rightarrow \hat{x} = \begin{bmatrix} 1 \\ 2 \end{bmatrix}.$$

O resumo do desenvolvimento para problemas de quadrados mínimos para sistemas lineares sobredeterminados e de posto completo é: dados $A \in \mathbb{R}^{m\times n}$ posto completo ($m \geqslant n$) e $b \in \mathbb{R}^m$, a solução (única) de quadrados mínimos do sistema linear $Ax = b$ é $x = A^\dagger b$, onde $A^\dagger = (A^TA)^{-1}A^T$.

A resolução de sistemas lineares subdeterminados de posto completo segue uma metodologia análoga, porém agora busca-se uma solução de norma mínima, entre as infinitas soluções do sistema linear $Ax = b$, onde $A \in \mathbb{R}^{m\times n}$ tem posto completo ($m \leqslant n$) e, nesse caso, $x = A^\dagger b$, onde $A^\dagger = A^T(AA^T)^{-1}$. Os dois casos falados até agora são casos particulares de solução de um sistema linear $Ax = b$, cuja solução é $x = A^\dagger b$, onde $A^\dagger$ é chamada de inversa de Moore-Penrose ou pseudoinversa.

Observação 2.2. *Vejamos como determinar a pseudoinversa de uma matriz qualquer utilizando a decomposição em valores singulares. Com efeito, sejam $A \in \mathbb{R}^{m\times n}$ de posto r e $A = U\Sigma V^T$ a sua SVD. Primeiramente, observe que a notação $\Sigma = \operatorname{diag}(\sigma_1, \sigma_2, \ldots, \sigma_r)$ se refere à diagonal principal de $\Sigma \in \mathbb{R}^{m\times n}$. Dependendo da relação entre m e n, a matriz Σ pode ser uma matriz quadrada, uma matriz retangular deitada ou uma matriz retangular em pé. De forma genérica podemos representar a matriz Σ da seguinte forma*

$$\Sigma = \begin{bmatrix} \Sigma_r & 0 \\ 0 & 0 \end{bmatrix},$$

onde $\Sigma_r = \operatorname{diag}(\sigma_1, \sigma_2, \ldots, \sigma_r) \in \mathbb{R}^{r\times r}$ *é uma matriz quadrada invertível. Nessas condições, definimos*

$$\Sigma^\dagger := \begin{bmatrix} \Sigma_r^{-1} & 0 \\ 0 & 0 \end{bmatrix} \in \mathbb{R}^{n\times m}.$$

Note que, $\Sigma^\dagger$ *satisfaz as quatro condições de uma pseudoinversa*[2] *[111, pág. 406]. A pseudoinversa de* A *é*

$$A^\dagger = \sum_{i=i}^{r} \frac{1}{\sigma_i} v_i u_i^T = V_r \Sigma_r^{-1} U_r^T,$$

onde $U_r = [u_1|u_2|\cdots|u_r] \in \mathbb{R}^{m\times r}$ *e* $V_r = [v_1|v_2|\cdots|v_r] \in \mathbb{R}^{n\times r}$. □

Observação 2.3. *Com a definição de pseudoinversa podemos modificar a forma como vemos a projeção ortogonal sobre um subespaço vetorial* M*. Construímos uma matriz* A *cujas colunas são os vetores de uma base de* $M = \operatorname{Im}(A)$ *e, com essa abordagem, a matriz* A *trivialmente possui posto completo e, portanto,* $P = A(A^TA)^{-1}A^T = AA^\dagger$*. Resumidamente,* $P = AA^\dagger$*, onde* A *tem posto completo, é a matriz projeção ortogonal sobre* $\operatorname{Im}(A)$*. Se* A *não tivesse posto completo, então* A^*A *seria singular (Lema 2.2) e, portanto, para aplicar a metodologia acima teríamos que construir uma matriz* $\hat{A}$*, cujas colunas seriam as colunas de* A *que formam um conjunto linearmente independente maximal. Porém, por construção,* $P = AA^\dagger$ *é a projeção ortogonal sobre* $\operatorname{Im}(A)$*, mesmo para matrizes posto deficientes. Ou seja, as matrizes (ou operadores)* $P = AA^\dagger$ *e* $Q = A^\dagger A$ *são projetores ortogonais (Exercício 6) e,*

1. P *é um projetor ortogonal sobre* $\operatorname{Im}(A)$.
2. Q *é um projetor ortogonal sobre* $\operatorname{Im}(A^T)$.
3. $I - P$ *é um projetor ortogonal sobre* $\operatorname{Ker}(A^T)$.
4. $I - Q$ *é um projetor ortogonal sobre* $\operatorname{Ker}(A)$.

□

[2] $X = A^\dagger$ se, e somente se, satisfaz as seguintes propriedades

$$\begin{aligned} AXA &= A; \\ XAX &= X; \\ (AX)^* &= AX; \\ (AX)^* &= XA. \end{aligned}$$

2.2 Sensibilidade dos Problemas de Quadrados Mínimos

Nessa seção apresentamos uma análise de sensibilidade para soluções de problemas de quadrados mínimos. Vamos iniciar a análise entendendo como perturbações sobre $A \in \mathbb{R}^{m\times n}$ afetam a pseudoinversa. Vejamos um simples exemplo de como as perturbações $A + \delta A$ se tornam não limitadas, quando $\operatorname{rank}(A) \neq \operatorname{rank}(A + \delta A)$ [157, pág. 219]. Sejam $\sigma \neq 0$ e

$$A = \begin{bmatrix} \sigma & 0 \\ 0 & 0 \end{bmatrix} \;\Rightarrow\; A^\dagger = \begin{bmatrix} -1/\sigma & 0 \\ 0 & 0 \end{bmatrix}.$$

Para $\epsilon > 0$ defina a perturbação

$$\delta A = \begin{bmatrix} 0 & \epsilon \\ \epsilon & 0 \end{bmatrix}.$$

Não é difícil verificar que $\|\delta A\|_2 = \epsilon$. Assim,

$$A + \delta A = \begin{bmatrix} \sigma & \epsilon \\ \epsilon & 0 \end{bmatrix} \;\Rightarrow\; (A + \delta A)^\dagger = (A + \delta A)^{-1} = \begin{bmatrix} 0 & 1/\epsilon \\ 1/\epsilon & -\sigma/\epsilon^2 \end{bmatrix}.$$

Note que, $\operatorname{rank}(A) = 1$ e $\operatorname{rank}(A + \delta A) = 2$. Logo,

$$A^\dagger - (A + \delta A)^\dagger = \begin{bmatrix} -1/\sigma & 1/\epsilon \\ 1/\epsilon & -\sigma/\epsilon^2 \end{bmatrix}$$

e, portanto,

$$\left\| A^\dagger - (A + \delta A)^\dagger \right\|_2 \geqslant \frac{1}{\epsilon} = \frac{1}{\|\delta A\|_2}.$$

Por exemplo, para $\sigma = 2$ e $\epsilon = 0,01$ temos

$$\left\| A^\dagger - (A + \delta A)^\dagger \right\|_2 = \frac{\sqrt{1600080001}}{2}.$$

Definição 2.1. *Uma matriz $B \in \mathbb{R}^{m\times n}$ é dita uma perturbação aguda de A, se*

$$\operatorname{rank}(B) = \operatorname{rank}(A) = \operatorname{rank}(P_A B P_{A^T}),$$

onde P_A é a projeção ortogonal sobre $\operatorname{Im}(A)$.

Vamos demonstrar que se $A + \delta A$ não é uma perturbação aguda de A, então resultados como os do parágrafo anterior ocorrem.

Teorema 2.2. *[136, pág. 643] Se $B = A + \delta A$ não é uma perturbação aguda de A, então*

$$\|B^\dagger - A^\dagger\|_2 \geqslant \frac{1}{\|\delta A\|_2}.$$

Ademais, se $\operatorname{rank}(B) \geqslant \operatorname{rank}(A)$, *então*

$$\|B^\dagger\|_2 \geqslant \frac{1}{\|\delta A\|_2}.$$

Demonstração. Se $\operatorname{rank}(B) \geqslant \operatorname{rank}(A)$, então existe $y \in \operatorname{Im}(B)$, $\|y\|_2 = 1$, tal que $y \in \operatorname{Im}(A)^\perp$ (caso contrário trabalhe com A^T e B^T). Assim,

$$1 = y^T y = y^T P_B y = y^T B B^\dagger y = y^T (A+\delta A) B^\dagger y = y^T \delta A\, B^\dagger y \leqslant \|\delta A\|_2 \|B^\dagger y\|_2,$$

ou seja, $\|B^\dagger\|_2 \geqslant 1/\|\delta A\|_2$. Note que, $A^\dagger y = A^\dagger P_A y = 0$, pois $y \in \operatorname{Im}(A)^\perp$. Portanto,

$$\frac{1}{\|\delta A\|_2} \leqslant \|B^\dagger y\|_2 \leqslant \|(B^\dagger - A^\dagger)y\|_2 \leqslant \|B^\dagger - A^\dagger\|_2.$$

□

Sob certas condições é possível estimar a 2-norma de uma perturbação de uma matriz A.

Teorema 2.3. *[157, pág. 220] Sejam $A, \delta A \in \mathbb{R}^{m\times n}$. Se*

$$\operatorname{rank}(A) = \operatorname{rank}(A+\delta A) = r \quad e \quad \|\delta A\|_2 < \frac{1}{\|A^\dagger\|_2},$$

então

$$\|(A+\delta A)^\dagger\|_2 \leqslant \frac{\|A^\dagger\|_2}{1 - \|A^\dagger\|_2 \|\delta A\|_2}.$$

Demonstração. Considere os valores singulares de A, $\sigma_1(A) \geqslant \cdots \geqslant \sigma_r(A) > 0$. Portanto, pela Observação 2.2,

$$\|A^\dagger\|_2 = \frac{1}{\sigma_r(A)}.$$

Pelo princípio minmax [70, pág. 89],

$$\frac{1}{\|(A+\delta A)^\dagger\|_2} = \sigma_r(A+\delta A) \geqslant \sigma_r(A) - \|\delta A\|_2 = \frac{1}{\|A^\dagger\|_2} - \|\delta A\|_2.$$

□

Vejamos um fato conhecido sobre perturbações invertíveis de matrizes invertíveis. Note que

$$B^{-1}A - I = B^{-1}(B - \delta A) - I = B^{-1}B - B^{-1}\delta A - I = -B^{-1}\delta A,$$

o que nos diz que $B^{-1} - A^{-1} = -B^{-1}\delta A\, A^{-1}$, com $B = A + \delta A$. Portanto,

$$\|B^{-1} - A^{-1}\| \leqslant \|B^{-1}\|\, \|A^{-1}\|\, \|\delta A\|.$$

Uma desigualdade semelhante é válida para o caso das pseudoinversas com certas restrições sobre A e B.

Teorema 2.4. *Sejam $A, B, \delta A \in \mathbb{R}^{m\times n}$, tais que $B = A + \delta A$ e* $\operatorname{rank}(A) = \operatorname{rank}(B)$. *Então,*

$$\|B^\dagger - A^\dagger\| \leqslant \mu \|B^\dagger\|\, \|A^\dagger\|\, \|\delta A\|, \tag{2.2.5}$$

onde $\mu = 1$ para a norma de Frobenius e, para a 2-norma,

$$\mu = \begin{cases} \dfrac{1+\sqrt{5}}{2} & \text{se } \operatorname{rank}(A) < \min\{m,n\} \\[2ex] \sqrt{2} & \text{se } \operatorname{rank}(A) = \min\{m,n\}. \end{cases}$$

Demonstração. A demonstração desse fato é extremamente longa. O resultado para a 2-norma foi demonstrado por Wedin [157, pág. 221] e, o resultado para a norma de Frobenius, foi demonstrado por van der Sluis e Veltkamp [149, pág. 263]. □

Corolário 2.1. *[12, pág. 27]* $\lim\limits_{\delta A\to 0} (A+\delta A)^\dagger = A^\dagger$ *se, e somente se,* $\lim\limits_{\delta A\to 0} \operatorname{rank}(A+\delta A) = \operatorname{rank}(A)$.

Estimativas para $\|B^\dagger - A^\dagger\|$ podem ser determinadas com hipóteses mais fracas, mas para a norma de Frobenius. Para demonstrar esse fato, precisamos de um resultado auxiliar.

Lema 2.3. *[23, pág. 340] Sejam $U = (U_1, U_2) \in \mathbb{R}^{m\times m}$ e $V = (V_1, V_2) \in \mathbb{R}^{n\times n}$ matrizes unitárias, onde $U_1 \in \mathbb{R}^{m\times r}$ e $V_1 \in \mathbb{R}^{n\times s}$ com $r \leqslant m$ e $s \leqslant n$. Então, para qualquer matrix $\delta A \in \mathbb{R}^{m\times n}$, temos*

$$\|\delta A\|_F^2 = \|U_1^T \delta A\, V_1\|_F^2 + \|U_1^T \delta A\, V_2\|_F^2 + \|U_2^T \delta A\, V_1\|_F^2 + \|U_2^T \delta A\, V_2\|_F^2.$$

Demonstração. Como a norma de Frobenius é unitariamente invariante, ou seja, $\|\delta A\|_F = \|U^T \delta A\, V\|_F$, segue que

$$\|\delta A\|_F^2 = \|U^T \delta A\, V\|_F^2 = \left\| \begin{bmatrix} U_1^T \delta A\, V_1 & U_1^T \delta A\, V_2 \\ U_2^T \delta A\, V_1 & U_2^T \delta A\, V_2 \end{bmatrix} \right\|_F^2.$$

□

Teorema 2.5. *[96, pág. 958] Sejam $A, B, \delta A \in \mathbb{R}^{m\times n}$, tais que $B = A + \delta A$ e* $\operatorname{rank}(A) = r$. *Então,*

$$\|B^\dagger - A^\dagger\|_F \leqslant \max\left\{ \|A^\dagger\|_2^2, \|B^\dagger\|_2^2 \right\} \|\delta A\|_F.$$

Demonstração. Seja $B = A + \delta A \in \mathbb{R}^{m\times n}$, com $\operatorname{rank}(B) = s$. Considere as decomposições em valores singulares de A e B,

$$A = U \begin{bmatrix} \Sigma_r & 0 \\ 0 & 0 \end{bmatrix} V^T \quad \text{e} \quad B = \tilde{U} \begin{bmatrix} \tilde{\Sigma}_s & 0 \\ 0 & 0 \end{bmatrix} \tilde{V}^T,$$

onde $U = (U_1, U_2), \tilde{U} = (\tilde{U}_1, \tilde{U}_2) \in \mathbb{R}^{m\times m}$ e $V = (V_1, V_2), \tilde{V} = (\tilde{V}_1, \tilde{V}_2) \in \mathbb{R}^{n\times n}$ são matrizes unitárias. A perturbação δA é escrita da seguinte forma

$$\delta A = B - A = \tilde{U}_1 \tilde{\Sigma}_s \tilde{V}_1^T - U_1 \Sigma_r V_1. \tag{2.2.6}$$

Ademais,

$$\begin{cases} U^T U = I_m & \Rightarrow \quad U_1^T U_1 = I_r \text{ e } U_1^T U_2 = 0, \\ V^T V = I_n & \Rightarrow \quad V_1^T V_1 = I_r \text{ e } V_1^T V_2 = 0, \\ \tilde{U}^T \tilde{U} = I_m & \Rightarrow \quad \tilde{U}_1^T \tilde{U}_1 = I_s \text{ e } \tilde{U}_1^T \tilde{U}_2 = 0, \\ \tilde{V}^T \tilde{V} = I_n & \Rightarrow \quad \tilde{V}_1^T \tilde{V}_1 = I_s \text{ e } \tilde{V}_1^T \tilde{V}_2 = 0. \end{cases}$$

A partir dessas quatro identidades e utilizando (2.2.6), obtemos

$$\begin{cases} \tilde{\Sigma}_s \tilde{V}_1^T V_1 - \tilde{U}_1^T U_1 \Sigma_r = \tilde{U}_1^T \delta A\, V_1, \\ U_1^T \tilde{U}_1 \tilde{\Sigma}_s - \Sigma_r V_1^T \tilde{V}_1 = U_1^T \delta A\, \tilde{V}_1, \end{cases} \tag{2.2.7}$$

e

$$\begin{cases} U_2^T \tilde{U}_1 \tilde{\Sigma}_s = U_2^T \delta A\, \tilde{V}_1, \qquad \tilde{U}_2^T U_1 \Sigma_r = -\tilde{U}_2^T \delta A\, V_1, \\ \tilde{\Sigma}_s \tilde{V}_1^T V_2 = \tilde{U}_1^T \delta A\, V_2, \qquad \Sigma_r V_1^T \tilde{V}_2 = -U_1^T \delta A\, \tilde{V}_2. \end{cases} \tag{2.2.8}$$

Como Σ_r e $\tilde{\Sigma}_s$ são não singulares, então (2.2.7) podem ser reescritas da seguinte forma

$$\begin{cases} \tilde{V}_1^T V_1 \Sigma_r^{-1} - \tilde{\Sigma}_s^{-1} \tilde{U}_1^T U_1 = \tilde{\Sigma}_s^{-1} \tilde{U}_1^T \delta A\, V_1 \Sigma_r^{-1}, \\ \Sigma_r^{-1} U_1^T \tilde{U}_1 - V_1^T \tilde{V}_1 \tilde{\Sigma}_s^{-1} = \Sigma_r^{-1} U_1^T \delta A\, \tilde{V}_1 \tilde{\Sigma}_s^{-1}. \end{cases} \tag{2.2.9}$$

As pseudoinversas de A e B são dadas por $A^\dagger = V_1 \Sigma_r^{-1} U_1^T$ e $B^\dagger = \tilde{V}_1 \tilde{\Sigma}_s^{-1} \tilde{U}_1^T$ e, portanto,

$$\begin{bmatrix} \tilde{V}_1^T \\ \tilde{V}_2^T \end{bmatrix} (B^\dagger - A^\dagger)(U_1, U_2) = \begin{bmatrix} \tilde{\Sigma}_s^{-1} \tilde{U}_1^T U_1 - \tilde{V}_1^T V_1 \Sigma_r^{-1} & \tilde{\Sigma}_s^{-1} \tilde{U}_1^T U_2 \\ -\tilde{V}_2^T V_1 \Sigma_r^{-1} & 0 \end{bmatrix} \tag{2.2.10}$$

e

$$\begin{bmatrix} V_1^T \\ V_2^T \end{bmatrix} (B^\dagger - A^\dagger)(\tilde{U}_1, \tilde{U}_2) = \begin{bmatrix} V_1^T \tilde{V}_1 \tilde{\Sigma}_s^{-1} - \Sigma_r^{-1} U_1^T \tilde{U}_1 & -\Sigma_r^{-1} U_1^T \tilde{U}_2 \\ V_2^T \tilde{V}_1 \tilde{\Sigma}_s^{-1} & 0 \end{bmatrix}. \tag{2.2.11}$$

A partir de (2.2.10) e (2.2.11) e utilizando o fato da norma de Frobenius ser unitariamente invariante, obtemos

$$\begin{aligned} 2\|B^\dagger - A^\dagger\|_F^2 = {} & \|\tilde{\Sigma}_s^{-1} \tilde{U}_1^T U_1 - \tilde{V}_1^T V_1 \Sigma_r^{-1}\|_F^2 + \|\tilde{\Sigma}_s^{-1} \tilde{U}_1^T U_2\|_F^2 + \|\tilde{V}_2^T V_1 \Sigma_r^{-1}\|_F^2 \\ & + \|V_1^T \tilde{V}_1 \tilde{\Sigma}_s^{-1} - \Sigma_r^{-1} U_1^T \tilde{U}_1\|_F^2 + \|\Sigma_r^{-1} U_1^T \tilde{U}_2\|_F^2 + \|V_2^T \tilde{V}_1 \tilde{\Sigma}_s^{-1}\|_F^2. \end{aligned} \tag{2.2.12}$$

Utilizando as equações (2.2.8), (2.2.9), (2.2.12) e o Lema 2.3, obtemos

$$
\begin{aligned}
2\|B^\dagger - A^\dagger\|_F^2 &= \|\tilde{\Sigma}_s^{-1}\tilde{U}_1^T\delta A\, V_1\Sigma_r^{-1}\|_F^2 + \|\tilde{\Sigma}_s^{-2}\tilde{V}_1^T\delta A^T U_2\|_F^2 \\
&\quad + \|\tilde{V}_2^T\delta A^T U_1\Sigma_r^{-2}\|_F^2 + \|\Sigma_r^{-1}U_1^T\delta A\,\tilde{V}_1\tilde{\Sigma}_s^{-1}\|_F^2 \\
&\quad + \|\Sigma_s^{-2}V_1^T\delta A^T\tilde{U}_2\|_F^2 + \|V_2^T\delta A^T\tilde{U}_1\tilde{\Sigma}_r^{-2}\|_F^2 \\
&\leqslant \frac{1}{\sigma_r^2\tilde{\sigma}_s^2}\left[\|\tilde{U}_1^T\delta A\, V_1\|_F^2 + \|U_1^T\delta A\,\tilde{V}_1\|_F^2\right] \\
&\quad + \frac{1}{\tilde{\sigma}_s^4}\left[\|\tilde{V}_1^T\delta A^T U_2\|_F^2 + \|V_2^T\delta A^T\tilde{U}_1\|_F^2\right] \\
&\quad + \frac{1}{\sigma_r^4}\left[\|\tilde{V}_2^T\delta A^T U_1\|_F^2 + \|V_1^T\delta A^T\tilde{U}_2\|_F^2\right] \\
&\leqslant \max\left\{\frac{1}{\sigma_r^4}, \frac{1}{\tilde{\sigma}_s^4}\right\}\Big[\|\tilde{U}_1^T\delta A\, V_1\|_F^2 + \|\tilde{U}_1^T\delta A\, V_2\|_F^2 + \|\tilde{U}_2^T\delta A\, V_1\|_F^2 \\
&\quad + \|U_1^T\delta A\,\tilde{V}_1\|_F^2 + \|U_1^T\delta A\,\tilde{V}_2\|_F^2 + \|U_2^T\delta A\,\tilde{V}_1\|_F^2\Big] \\
&\leqslant 2\max\left\{\frac{1}{\sigma_r^4}, \frac{1}{\tilde{\sigma}_s^4}\right\}\|\delta A\|_F^2.
\end{aligned}
$$

Mas, $\|A^\dagger\| = 1/\sigma_r$ e $\|B^\dagger\| = 1/\sigma_s$. Logo,

$$\|B^\dagger - A^\dagger\|_F \leqslant \max\left\{\|A^\dagger\|_2^2, \|B^\dagger\|_2^2\right\}\|\delta A\|_F.$$

□

Observe que, $\kappa(A^TA) = \kappa(A)^2$, onde $\kappa(A) = \|A^{-1}\|\cdot\|A\|$ é o número de condição da matriz A. Com efeito, seja $B = A^TA$ e considere a SVD de A

$$A = U\begin{bmatrix}\Sigma_r & 0\\ 0 & 0\end{bmatrix}V^T.$$

Avaliando o produto A^TA, obtemos

$$B = V\begin{bmatrix}\Sigma_r^2 & 0\\ 0 & 0\end{bmatrix}V^T.$$

Portanto,

$$\kappa(B) = \frac{\sigma_1(B)}{\sigma_r(B)} = \frac{\sigma_1(A)^2}{\sigma_r(A)^2} = \kappa(A)^2.$$

Assim, se A é mal condicionada, então o problema de quadrados mínimos será muito pior condicionado. Uma regra prática é trabalhar com duas vezes

mais dígitos significativos do que os dados por A e b. Uma discussão mais aprofundada sobre a aritmética de ponto flutuante do sistema normal é feita por Björck [13, pág. 224].

Vamos considerar agora o efeito de perturbações nas equações normais $A^T Ax = A^T b$ sobre as soluções de quadrados mínimos $x = A^\dagger b$. Já havíamos definido P_A, a projeção ortogonal sobre $\mathrm{Im}(A)$, e agora, definamos $P_A^\perp$ como a projeção ortogonal sobre $\mathrm{Ker}\,(A^T)$. Primeiramente apresentamos uma discussão baseada em Wedin [157] e, posteriormente, enfraquecemos as hipóteses fazendo uma discussão baseada em Björck [13]. Antes de analisarmos as perturbações de um sistema normal, demonstremos um resultado chamado de Teorema da Decomposição.

Teorema 2.6 (Teorema da Decomposição)**.** *[157, pág. 218] Sejam $A, \delta A \in \mathbb{R}^{m\times n}$ e $B = A + \delta A$, então*

$$B^\dagger - A^\dagger = -B^\dagger \delta A\, A^\dagger + B^\dagger P_A^\perp - P_{B^T}^\perp A^\dagger, \tag{2.2.13}$$

$$P_{A^T} P_{B^T}^\perp = -A^\dagger \delta A\, P_{B^T}^\perp, \tag{2.2.14}$$

$$P_A^\perp P_B = P_A^\perp \delta A\, B^\dagger, \tag{2.2.15}$$

$$B^\dagger - A^\dagger = -B^\dagger \delta A\, A^\dagger + (B^T B)^\dagger \delta A^T P_A^\perp + P_{B^T}^\perp \delta A^T (AA^T)^\dagger. \tag{2.2.16}$$

Demonstração. Decomponha $B^\dagger - A^\dagger$ com respeito à $\mathrm{Im}(A)$ e $\mathrm{Ker}\,(A^T) = \mathrm{Im}(A)^\perp$ em $\mathbb{R}^m$ e com respeito à $\mathrm{Im}(B^T)$ e $\mathrm{Ker}\,(B) = \mathrm{Im}(B^T)^\perp$ em $\mathbb{R}^n$, obtendo

$$\begin{aligned} B^\dagger - A^\dagger &= (P_{B^T} + P_{B^T}^\perp)(B^\dagger - A^\dagger)(P_A + P_A^\perp) \\ &= P_{B^T}(B^\dagger - A^\dagger)P_A + B^\dagger P_A^\perp - P_{B^T}^\perp A^\dagger. \end{aligned}$$

Note que, pela Observação 2.3,

$$P_{B^T}(B^\dagger - A^\dagger)P_A = B^\dagger AA^\dagger - B^\dagger BA^\dagger = -B^\dagger \delta A\, A^\dagger, \tag{2.2.17}$$

demonstrando assim a identidade (2.2.13). A demonstração de (2.2.14) segue de

$$P_{A^T} P_{B^T}^\perp = A^\dagger A P_{B^T}^\perp = A^\dagger (B - \delta A) P_{B^T}^\perp = -A^\dagger \delta A\, P_{B^T}^\perp.$$

A identidade (2.2.15) é demonstrada analogamente e, utilizando o fato que as projeções ortogonais são auto-adjuntas, uma consequência de (2.2.14) e (2.2.15) é

$$\begin{cases} P_{B^T}^\perp A^\dagger = P_{B^T}^\perp P_{A^T} A^\dagger = -P_{B^T}^\perp \delta A^T A^{*\dagger} A^\dagger, \\ B^\dagger P_A^\perp = B^\dagger P_B P_A^\perp = B^\dagger B^{*\dagger} \delta A^T P_A^\perp. \end{cases} \tag{2.2.18}$$

A identidade (2.2.16) é obtida quando substituímos (2.2.18) em (2.2.13). $\square$

Vamos estudar o comportamento do sistema normal quando perturbamos a matriz de coeficientes $A \in \mathbb{R}^{m\times n}$ e o vetor de dados $b \in \mathbb{R}^m$. A discussão se aplica para sistemas sobredeterminados e subdeterminados, posto completo ou não. Para facilitar a notação vamos definir algumas quantidades. Sejam $A + \delta A$ a perturbação de A e $b + \delta b$ a perturbação de b. A perturbação da solução $x = A^\dagger b$ é $x + \delta x = (A + \delta A)^\dagger(b + \delta b)$, já a perturbação do resíduo $r = b - Ax$ é dada por $r + \delta r = (b + \delta b) + (A + \delta A)(b + \delta b)$. Denotemos por $\kappa = \|A\|_2\|A^\dagger\|_2$ o número de condição espectral (2-norma). Ademais, $\epsilon = \|\delta A\|_2/\|A\|_2$ e $y = A^{T\dagger}x = (AA^T)^\dagger b$. Note que $\kappa\epsilon = \|A^\dagger\|_2\,\|\delta A\|_2$.

Teorema 2.7. *[157, pág. 224] Seja $A \in \mathbb{R}^{m\times n}$. Assuma* $\operatorname{rank}(A) = \operatorname{rank}(A + \delta A)$ *e $\kappa\epsilon < 1$. Então*

$$\|\delta x\|_2 \leqslant \frac{\kappa}{(1-\kappa\epsilon\|A\|_2)}\,[\epsilon\|x\|_2\,\|A\|_2 + \|\delta b\|_2 + \kappa\epsilon\|r\|_2] + \epsilon\|y\|_2\,\|A\|_2 \tag{2.2.19}$$

e

$$\|\delta r\|_2 \leqslant \epsilon\|x\|_2\,\|A\|_2 + \|\delta b\|_2 + \kappa\epsilon\|r\|_2. \tag{2.2.20}$$

Demonstração. Pelo Teorema da Decomposição, temos

$$\begin{aligned}
\delta x &= (A+\delta A)^\dagger(b+\delta b) - A^\dagger b \\
&= \left[-(A+\delta A)^\dagger\delta A\,A^\dagger + (A+\delta A)^\dagger P_A^\perp - P^\perp_{(A+\delta A)^T}A^\dagger\right] b + (A+\delta A)^\dagger\delta b \\
&= \left[-(A+\delta A)^\dagger\delta A x + (A+\delta A)^\dagger r + (A+\delta A)^\dagger\delta b\right] - P^\perp_{(A+\delta A)^T}x.
\end{aligned}$$

Os termos entre colchetes pertencem à $\operatorname{Im}(A+\delta A)^T$, já o último pertence ao $\operatorname{Ker}(A+\delta A)$. Tome a 2-norma de δx, aplique a desigualdade triangular e façamos algumas majorações de cada termo envolvido no cômputo de $\|\delta x\|_2$. Pelo Teorema 2.3, temos

$$\|(A+\delta A)^\dagger\|_2 \leqslant \frac{\|A^\dagger\|_2}{1 - \|A^\dagger\|_2\,\|\delta A\|_2}.$$

Assim,

$$\begin{cases}
\|(A+\delta A)^\dagger\delta A x\|_2 \leqslant \dfrac{\kappa\epsilon}{1-\kappa\epsilon}\|x\|_2, \\[2ex]
\|(A+\delta A)^\dagger r\|_2 \leqslant \|(A+\delta A)^\dagger\|_2\,\|P_{(A+\delta A)}P_A^\perp\|_2\,\|r\|_2 \overset{(2.2.15)}{\leqslant} \dfrac{\kappa^2\epsilon}{1-\kappa\epsilon}\,\dfrac{\|r\|_2}{\|A\|_2}, \\[2ex]
\|(A+\delta A)^\dagger\delta b\|_2 \leqslant \dfrac{\kappa}{1-\kappa\epsilon}\,\dfrac{\|\delta b\|_2}{\|A\|_2}, \\[2ex]
\|P^\perp_{(A+\delta A)^T}x\|_2 = \|P^\perp_{(A+\delta A)^T}P_{A^T}x\|_2 \\
\qquad\qquad \overset{(2.2.14)}{=} \|P^\perp_{(A+\delta A)^T}\delta A^T A^{*\dagger}x\|_2 \leqslant \|\delta A\|_2\,\|y\|_2.
\end{cases}$$

Concluindo, assim, a demonstração de (2.2.19). Para demonstrar (2.2.20), observe que

$$\begin{aligned}\delta r &= (b+\delta b)-(A+\delta A)(x+\delta x)-(b-Ax)\\ &= P^{\perp}_{(A+\delta A)}(b+\delta b)-P^{\perp}_{A}b\\ &= P^{\perp}_{(A+\delta A)}\delta b+P^{\perp}_{(A+\delta)}P_A b-P_{(A+\delta A)}P^{\perp}_{A}r.\end{aligned}$$

Utilizando argumento similar à demonstração de (2.2.14), obtemos

$$P^{\perp}_{(A+\delta)}P_A = P^{\perp}_{(A+\delta)}AA^{\dagger} = P^{\perp}_{(A+\delta)}[(A+\delta A)-\delta A]A^{\dagger} = -P^{\perp}_{(A+\delta)}\delta AA^{\dagger}.$$

Da mesma forma que procedemos na demonstração de (2.2.19), vamos majorar $\|\delta r\|_2$ estimando cada componente independentemente. Com efeito,

$$\|P^{\perp}_{(A+\delta A)}\delta b\|_2 \leqslant \|\delta b\|_2,$$

$$\|P^{\perp}_{(A+\delta)}P_A b\|_2 = \|P^{\perp}_{(A+\delta)}\delta AA^{\dagger}b\|_2 = \|P^{\perp}_{(A+\delta)}\delta Ax\|_2 \leqslant \|\delta A\|_2\,\|x\|_2,$$

$$\|P_{(A+\delta A)}P^{\perp}_{A}r\|_2 \overset{(2.2.15)}{\leqslant} \|\delta A\|_2\,\|A^{\dagger}\|_2\,\|r\|_2,$$

demonstrando assim (2.2.20). □

Teorema 2.8. *[13, pág. 235] Considere o problema de quadrados mínimos dado por* $\min\|b-Ax\|_2$ *com* $A\in\mathbb{R}^{m\times n}$ *posto completo e* $b\in\mathbb{R}^m$. *Sejam* $A+\delta A$ *e* $b+\delta b$ *perturbações dos dados e assuma que a perturbação* $A+\delta A$ *é tal que* $\operatorname{rank}(A)=\operatorname{rank}(A+\delta A)=n$. *Se termos de segunda ordem são negligenciados, então*

$$\|\delta x\|_2 \leqslant \|A^{\dagger}\|_2\left[\|\delta b\|_2+\|\delta A\|_2\,\|x\|_2+\|A^{\dagger}\|_2\,\|\delta A\|_2\,\|r\|_2\right] \tag{2.2.21}$$

e

$$\|\delta r\|_2 \leqslant \|\delta b\|_2+\|\delta A\|_2\,\|x\|_2+\|A^{\dagger}\|_2\,\|\delta A\|_2\,\|r\|_2. \tag{2.2.22}$$

Demonstração. Considere o sistema aumentado associado ao problema de quadrados mínimos $A^TAx=A^Tb$,

$$\begin{bmatrix} I & A\\ A^T & 0\end{bmatrix}\begin{bmatrix} r\\ x\end{bmatrix}=\begin{bmatrix} b\\ 0\end{bmatrix}.$$

O sistema aumentado é não singular, pois A tem posto completo. Considere, agora, o sistema aumentado das perturbações,

$$\begin{bmatrix} I & A+\delta A\\ (A+\delta A)^T & 0\end{bmatrix}\begin{bmatrix} r+\delta r\\ x+\delta x\end{bmatrix}=\begin{bmatrix} b+\delta b\\ 0\end{bmatrix}.$$

Subtraindo o sistema não perturbado do sistema perturbado e não considerando os termos de segunda ordem, obtemos

$$\begin{bmatrix} I & A \\ A^T & 0 \end{bmatrix} \begin{bmatrix} \delta r \\ \delta x \end{bmatrix} = \begin{bmatrix} \delta b - \delta A x \\ -\delta A^T r \end{bmatrix}.$$

Pela fórmula da inversão de Banachiewicz[3] [13, pág. 20], temos

$$\begin{bmatrix} I & A \\ A^T & 0 \end{bmatrix}^{-1} = \begin{bmatrix} I - A(A^TA)^{-1}A^T & A(A^TA)^{-1} \\ (A^TA)^{-1}A^T & -(A^TA)^{-1} \end{bmatrix} = \begin{bmatrix} P_A^{\perp} & A^{*\dagger} \\ A^{\dagger} & -A^{\dagger}A^{*\dagger} \end{bmatrix}.$$

Portanto,

$$\delta x = A^{\dagger}(\delta b - \delta A x) + A^{\dagger}A^{*\dagger}\delta A^T r$$

e

$$\delta r = P_A^{\perp}(\delta b - \delta A x) - A^{*\dagger}\delta A^T r.$$

O resultado é demonstrado após o cálculo da 2-norma e a utilização da desigualdade triangular. □

Se $x \neq 0$ e tomando $\delta b = 0$, obtemos

$$\frac{\|\delta x\|_2}{\|x\|_2} \leqslant \kappa_{LS} \frac{\|\delta A\|_2}{\|A\|_2}, \quad \text{onde,} \quad \kappa_{LS} = \kappa(A)\left[1 + \frac{\|A^{\dagger}\|_2 \, \|r\|_2}{\|x\|_2}\right].$$

Malyshev [92, pág. 1188] mostra que as estimativas como as acima são superestimadores e apresenta uma estimativa mais precisa para κ_{LS}, dada por

$$\kappa_{LS} = \kappa(A)\sqrt{1 + \left[\frac{\|A^{\dagger}\|_2 \, \|r\|_2}{\|x\|_2}\right]^2}.$$

Já Grcar [60, pág. 2943] apresenta outra estimativa, tanto inferior quanto superior, para o número de condição κ_{LS}, além de fazer uma comparação entre as estimativas publicadas.

2.3 Exercícios

1. Encontre a solução de quadrados mínimos para os sistemas lineares a seguir.

1. $\begin{cases} 2y + 2z &= 2 \\ x - y + z &= 1 \\ x + y + 3z &= 1 \\ x + 3y + 2z &= 1 \end{cases}$ 2. $\begin{cases} x + 3y &= 1 \\ 2x + y &= 0 \\ 2x + 2y &= 0 \\ -2y &= 1 \end{cases}$

[3] MacTutor: `https://mathshistory.st-andrews.ac.uk/Biographies/Banachiewicz/`

2. Encontre o vetor pertencente à $\mathrm{Im}(A)$ mais próximo a b, onde

$$A = \begin{bmatrix} 1 & 4 \\ 1 & -1 \\ 0 & 2 \end{bmatrix} \quad \text{e} \quad b = \begin{bmatrix} 1 \\ 0 \\ 1 \end{bmatrix}.$$

3. É possível que um sistema de equações lineares com mais equações que incógnitas ter mais de uma solução de quadrados mínimos? Explique.

4. Utilize o método de quadrados mínimos para encontrar a reta e a quádrica que melhor ajustam os dados abaixo.

x	-6	-4	-2	0	2	4	6
y	6	1	-1	0	3	8	15

Determine qual dessa curvas melhor ajusta os dados computando o erro quadrático em cada caso.

5. Utilize o método de quadrados mínimos para determinar o ponto na reta $y = 2x$ mais próximo de $(1, 1)$.

6. Seja $A \in \mathbb{K}^{m\times n}$. Demonstre que $P = AA^\dagger$ e $Q = A^\dagger A$ são projetores ortogonais.

7. O problema de quadrados mínimos

$$\textit{minimize} \ \|b - Ax\|_2^2$$

pode ser reescrito na seguinte forma funcional

$$\textit{minimize} \ f(x_1, \ldots, x_n) = \sum_{i=1}^{m} (a_{i1}x_1 + \cdots + a_{in}x_n - b_i)^2.$$

Utilize derivação parcial para deduzir a equação normal $A^T Ax = A^T b$.

8. [147] Tome $m = 50$ e $n = 12$. Usando o comando de Matlab linspace defina t um vetor coluna m-dimensional correspondente a um grid espaçado linearmente entre 0 e 1. Usando os comandos de Matlab vander e fliplr, defina A uma matriz $m \times n$ associada ao problema de quadrados mínimos nesse grid por um polinômio de grau $n - 1$. Tome b como sendo a função $\cos(4t)$ avaliada no grid definido. Agora, calcule e imprima (com uma precisão de 16 casas decimais) o vetor de coeficientes de quadrados mínimos x por seis métodos:

1. Formação e solução das equações normais, usando o comando de Matlab \,
2. Fatoração QR computado por mgs (Gram-Schmidt modificado),

3. Fatoração QR computado por house (Triangularização de Householder),
4. Fatoração QR computado por qr (também triangularização de Householder),
5. $x = A\backslash b$ no Matlab (também baseado em fatoração QR),
6. SVD, usando o comando svd do Matlab,
7. Os cálculos acima irão produzir seis listas de doze coeficientes. Em cada lists, marque com caneta vermelha os dígitos que parecem errados (afetados por erros de arredondamento). Comente as diferenças observadas. As equações normais apresentam instabilidade? Você não precisa explicar suas observações.

9. Sob as hipóteses do Teorema 2.7, encontre $\|\delta x\|_2/\|x\|_2$ para o caso que $b \in \mathrm{Im}(A)$.

10. Sejam V e W subespaços vetoriais de $\mathbb{R}^n$. Ademais, P_X e P_Y são projeções ortogonais sobre X e Y, respectivamente. Demonstre que, σ é um valor singular de $P_X P_Y$ se, e somente se, σ^2 é um autovalor de $P_Y P_X$.

11. Mostre que $\mathrm{rank}(A) = \mathrm{rank}(A + \delta A)$ não é suficiente para $A + \delta A$ ser uma perturbação aguda de A.

12. Considere a estimativa para $\|\delta x\|_2$ dada por (2.2.21). Demonstre que

$$\lim_{\epsilon \to 0} \sup_{\|\delta A\|_2 \leqslant \epsilon} \left[\frac{\|\delta x\|_2}{\|x\|_2} \Big/ \frac{\|\delta A\|_2}{\|A\|_2} \right] \leqslant \kappa(A) \left[1 + \frac{\|A^\dagger\|_2 \, \|r\|_2}{\|x\|_2} \right].$$

13. [58] Assuma $A^T A x = A^T b$, $(A^T A + F)\hat{x} = A^T b$ e $2\|F\|_2 \leqslant \sigma_n(A)^2$. Mostre que se $r = b - Ax$ e $\hat{r} = b - A\hat{x}$, então $\hat{r} - r = A(A^T A + F)^{-1} F x$ e

$$\|\hat{r} - r\|_2 \leqslant 2\kappa_2(A) \frac{\|F\|_2}{\|A\|_2} \|x\|_2.$$

Capítulo 3

Métodos Iterativos Básicos

Os métodos iterativos para a determinação da solução de um sistema linear obtido pelo método de quadrados mínimos $A^T Ax = A^T b$, ou para o problema de norma mínima $y = Az$, $AA^T z = b$, também chamadas equações normais do segundo tipo, são essencialmente os mesmos métodos iterativos para a determinação da solução de sistemas lineares em geral. As diferenças surgem quando exploramos o fato de estarmos trabalhando com equações normais e, com isso, conseguimos "aprimorar" os métodos já conhecidos sem que tenhamos que calcular o produto $A^T A$ (ou AA^T) explicitamente.

Dado um sistema linear $Ax = b$, com $A \in \mathbb{R}^{m\times n}$ de posto completo e $m \geqslant n$, o problema de quadrados mínimos $A^T Ax = A^T b$ tem solução única. Para resolver as equações normais em busca de tal solução, precisamos calcular $A^T A$ com um custo de $\mathcal{O}(n^2 m)$ flops, enquanto uma resolução dessas equações por algum método direto tem um custo, em geral, de $\mathcal{O}(n^3)$ flops. O problema é que, com o avanço da ciência e da era dos dados, esse custo computacional torna esses métodos de resolução impraticáveis, pois n cresce muito rapidamente.

Assim, chegamos nos métodos iterativos para a resolução desses sistemas lineares, e eles têm, em geral, um custo de $\mathcal{O}(n^2)$ flops ou menos. No caso de quadrados mínimos, geralmente não precisamos calcular o produto $A^T A$ explicitamente, o que torna a utilização de métodos iterativos para problemas de quadrados mínimos bastante atrativa.

Nesse capítulo trabalharemos com matrizes reais e parâmetros reais. As ideias aqui discutidas se estendem naturalmente ao corpo dos complexos com pequenas adaptações, e a bibliografia básica que seguimos para o desenvolvimento do capítulo é o livro de Björck [12].

3.1 Métodos Iterativos Estacionários

A classe mais simples de métodos iterativos para resolver as equações normais é a classe dos métodos iterativos estacionários. Considere a decomposição da matriz das equações normais dada por $A^T A = M - N$, onde M é não singular.

Sob essas condições os métodos iterativos estacionários têm a seguinte forma

$$Mx^{(k+1)} = Nx^{(k)} + A^T b, \qquad k = 0, 1, 2, \ldots \tag{3.1.1}$$

com $x^{(0)}$, a aproximação inicial. Ademais, a matriz M deve ser escolhida de forma que, para cada iteração $k = 0, 1, 2, \ldots$, o sistema linear $Mx^{(k+1)} = Nx^{(k)} + b$ seja de fácil resolução. Para garantir rápida convergência do método iterativo estacionário, pede-se que $M \approx A^T A$ e $N \approx 0$. Assim, procuramos uma matriz M não singular, que de alguma forma aproxime A e que garanta fácil resolução para os sistemas lineares que são gerados a cada iteração do método [156, pág. 545].

Antes de apresentarmos alguns métodos iterativos estacionários, analisemos as condições de convergência do método iterativo estacionário geral (3.1.1). Para esse objetivo, defina

$$G = M^{-1}N = I - M^{-1}A^T A \qquad \text{e} \qquad c = M^{-1}A^T b, \tag{3.1.2}$$

onde G é chamada de matriz de iteração. Assim, a equação (3.1.1) pode ser reescrita como

$$x^{(k+1)} = Gx^{(k)} + c, \qquad k = 0, 1, 2, \ldots \tag{3.1.3}$$

Definição 3.1. *Chamamos o método iterativo* (3.1.3) *de convergente se a sequência de iterações* $\{x^{(k)}\}$ *convergir, para qualquer valor de* $x^{(0)}$ *dado.*

Não é difícil ver que se x é o limite da sequência $\{x^{(k)}\}$, então é um ponto fixo de $f(x) = Gx + c$, ou seja, $x = Gx + c$. Para demonstrarmos o teorema de convergência do método (3.1.3) precisamos de uma definição.

Definição 3.2. *Seja* $G \in \mathbb{R}^{n\times n}$. *O raio espectral de* G, $\rho(G)$, *é dado por*

$$\rho(G) = \max_{1\leq i\leq n} |\lambda_i(G)|,$$

onde λ_i *é o* i*-ésimo autovalor de* G.

Enunciaremos três resultados que são a base do estudo de convergência dos métodos iterativos estacionários.

Teorema 3.1. *[144, pág. 442] Sejam* $G \in \mathbb{R}^{n\times n}$ *e* $\epsilon > 0$. *Então existe uma norma de vetores tal que,*

$$\rho(G) \leqslant \|G\| \leqslant \rho(G) + \epsilon. \tag{3.1.4}$$

Demonstração. Seja x um autovetor associado a um autovalor λ de G, assim $Gx = \lambda\, x$. Portanto,

$$|\lambda|\ \|x\| = \|\lambda x\| = \|Gx\| \leqslant \|G\|\ \|x\|\,.$$

Como x é um vetor não nulo, então $|\lambda| \leqslant \|G\|$. Portanto, $\rho(G) \leqslant \|G\|$.
Agora, considere a forma de Jordan de G, isto é, $J = P^{-1}GP$. Defina,

$D_\epsilon = \operatorname{diag}(1, \epsilon, \epsilon^2, \ldots, \epsilon^{n-1})$. Note que, $J_\epsilon = D_\epsilon^{-1} J D_\epsilon$ tem a mesma forma de J, exceto pelo fato que os 1's foram trocados por ϵ's. Portanto,

$$\|J_\epsilon\|_\infty = \left\|D_\epsilon^{-1} P^{-1} G P D_\epsilon\right\|_\infty \leqslant \rho(G) + \epsilon.$$

Seja $S = PD_\epsilon$ e defina $\|x\| = \left\|S^{-1}x\right\|_\infty$ que é uma norma. Logo,

$$\begin{aligned}
\|G\| &= \max_{\|x\|=1} \|Gx\| = \max_{\|S^{-1}x\|_\infty=1} \left\|S^{-1}Gx\right\|_\infty \\
&= \max_{\|y\|_\infty=1} \left\|S^{-1}GSy\right\|_\infty = \max_{\|y\|_\infty=1} \|J_\epsilon y\|_\infty \\
&= \|J_\epsilon\|_\infty \leqslant \rho(G) + \epsilon.
\end{aligned}$$

□

Observe da demonstração que a primeira desigualdade de (3.1.4) é válida para qualquer norma.

Teorema 3.2. *[103, pág. 25] Seja $G \in \mathbb{R}^{n\times n}$. Então, $\lim_{k\to\infty} G^k = 0$ se, e somente se, $\rho(G) < 1$.*

Demonstração. Se $\rho(G) < 1$, então, por (3.1.4), $\|G\| < 1$ para alguma norma de $\mathbb{R}^n$. Mas,

$$\lim_{k\to\infty} \|G^k\| \leqslant \lim_{k\to\infty} \|G\|^k = 0.$$

Reciprocamente, suponha $\rho(G) \geqslant 1$. Portanto, existe autovalor $|\lambda| \geqslant 1$, e seja x o autovetor associado a λ. Portanto, para cada inteiro positivo k,

$$\|G^k x\| = \|\lambda^k x\| = |\lambda|^k \|x\| \geqslant \|x\| \;\Rightarrow\; \|G^k\| \geqslant 1.$$

Logo, $\lim_{k\to\infty} G^k \neq 0$. □

Essa proposição nos diz que o estudo de convergência de um método iterativo estacionário é equivalente ao estudo do espectro da matriz de iteração. Os dois teoremas acima podem ser resumidos da seguinte forma.

Teorema 3.3. *Seja $G \in \mathbb{R}^{n\times n}$. As seguintes condições são equivalentes:*

1. $\lim_{k\to\infty} G^k = 0$,
2. $\lim_{k\to\infty} G^k z = 0$, $\forall z \in \mathbb{C}^n$,
3. $\rho(G) < 1$,
4. $\|G\| < 1$, *para pelo menos uma norma de matriz.*

Passemos, agora, ao teorema da convergência dos métodos iterativos estacionários (3.1.3).

Teorema 3.4. *[12, pág. 275] O método iterativo estacionário* $x^{(k+1)} = Gx^{(k)} + c$ *é convergente para qualquer vetor inicial* $x^{(0)}$ *se, e somente se,* $\rho(G) < 1$.

Demonstração. Subtraindo $x = Gx + c$ de (3.1.3), segue que

$$x^{(k)} - x = G(x^{(k-1)} - x) = \cdots = G^k(x^{(0)} - x). \tag{3.1.5}$$

Portanto, $\lim_{k\to\infty} x^{(k)} = x$ se, e somente se, $\lim_{k\to\infty} G^k = 0$. Do Teorema 3.3 demonstra-se o resultado. □

Da equação (3.1.5) e para qualquer par de normas consistentes, obtemos

$$\|x^{(k)} - x\| \leq \|G^k\|\|x^{(0)} - x\|.$$

Assim, $\|G^k\|$ é o principal elemento para medirmos o comportamento do erro local, $e^{(k)} = x^{(k)} - x$, a cada iteração k do método iterativo. A partir desse fato, definimos os conceitos de taxa média de convergência e de taxa assintótica de convergente.

Definição 3.3. *Considere o método iterativo*

$$x^{(k+1)} = Gx^{(k)} + c, \qquad k = 0, 1, 2, \ldots.$$

Definimos,

$$R_k(G) = -\frac{1}{k}\log_{10}\|G^k\| \quad e \quad R_\infty(G) = -\log_{10}\rho(G)$$

como a taxa média e a taxa assintótica de convergência, respectivamente.

Para o desenvolvimento de métodos iterativos estacionários para problemas de quadrados mínimos, temos a seguinte definição.

Definição 3.4. *O método iterativo estacionário*

$$Mx^{(k+1)} = Nx^{(k)} + A^T b, \qquad k = 0, 1, 2, \ldots$$

é dito simetrizável se a matriz $I - G$ *é similar a uma matriz simétrica definida positiva, onde* G *é definida em* (3.1.2).

Essa definição diz que o método iterativo estacionário é simetrizável se existe W não singular, tal que

$$W(I - G)W^{-1} = WM^{-1}A^T AW^{-1}$$

seja definida positiva[1]. Especificamente, para o caso de métodos iterativos para problemas de quadrados temos o seguinte resultado.

[1] Não esqueça que aqui, consideramos que as matrizes definidas positivas são, também, simétricas.

Teorema 3.5. *[12, pág. 275] Um método iterativo estacionário para as equações normais é simetrizável se a matriz M, da decomposição $A^TA = M - N$, for definida positiva.*

Demonstração. Seja R o fator de Cholesky de A^TA. Assim,

$$RM^{-1}(A^TA)R^{-1} = RM^{-1}(R^TR)R^{-1} = RM^{-1}R^T$$

que é definida positiva, pois M é definida positiva, por hipótese. □

Os conceitos de decomposição do tipo $A = M - N$ podem ser estendidos para matrizes retangulares e os trabalhos iniciais sobre o assunto podem ser encontrados em [7, 114, 146].

3.2 Métodos Iterativos Clássicos

Nessa seção estudamos os chamamos métodos iterativos clássicos, a saber, métodos de Landweber, Jacobi, Gauss-Seidel e SOR. Como estamos trabalhando com as equações normais associadas a um sistema linear $Ax = b$, apresentamos algumas adaptações ao contexto em questão para os métodos supracitados. A discussão é baseada em [12] e, a menos que dito o contrário, $A \in \mathbb{R}^{m\times n}$ com $m \geqslant n$ e $\operatorname{rank}(A) = n$.

3.2.1 Método de Landweber

Seja $\alpha > 0$ e considere a seguinte decomposição

$$M = \frac{1}{\alpha}I \qquad \text{e} \qquad N = \frac{1}{\alpha}I - A^TA.$$

De (3.1.2),

$$G_L = M^{-1}N = I - \alpha A^TA.$$

Este método iterativo é chamado método de Landweber ou método de Richardson[2] [14, 82, 119] . Note que, como $I - G_L = \alpha A^TA$ é simétrica definida positiva, então o método iterativo é simetrizável. Nesse caso,

$$x^{(k+1)} = G_Lx^{(k)} + M^{-1}A^Tb = x^{(k)} + \alpha A^T(b - Ax^{(k)}), \tag{3.2.6}$$

que não requer a geração das equações normais. Ademais, os autovalores de G_L são

$$\lambda_j(G_L) = 1 - \alpha\sigma_j^2, \qquad j = 1, \ldots, n,$$

onde σ_k são os valores singulares de A. Pelo Teorema 3.4, a convergência do método para a solução de quadrados mínimos é garantida para

$$0 < \alpha < \frac{2}{\sigma_1^2} \qquad \text{e} \qquad x^{(0)} \in \mathcal{R}(A^T).$$

[2]MacTutor: `https://mathshistory.st-andrews.ac.uk/Biographies/Richardson/`

3.2.2 Método de Jacobi

Seja $A = [a_1|a_2|\cdots|a_n] \in \mathbb{R}^{m\times n}$. Defina $D_A = \text{diag}(d_1, \ldots, d_n)$, com $d_j = a_j^T a_j = \|a_j\|^2$ a matriz formada pela diagonal de $A^T A$ e L_A a matriz triangular inferior com diagonal nula formada pelos elementos abaixo da diagonal principal de $A^T A$. O método de Jacobi[3] é obtido com a fatoração $A^T A = L_A + D_A + L_A^T = D_A + (A^T A - D_A)$, ou seja, pela equação (3.1.1), $M = D_A$ e $N = D_A - A^T A$. Assim, por (3.1.2), temos

$$G_J = I - D_A^{-1} A^T A \quad \text{e} \quad c_J = D_A^{-1} A^T b. \tag{3.2.7}$$

Portanto, a iteração do método de Jacobi é dada por

$$\begin{aligned} x^{(k+1)} &= x^{(k)} - D_A^{-1}(A^T A)x^{(k)} + D_A^{-1} A^T b \\ &= x^{(k)} + D_A^{-1} A^T (b - Ax^{(k)}). \end{aligned} \tag{3.2.8}$$

Observe que a implementação do método de Jacobi não envolve o cálculo de $A^T A$. Componente a componente, o método de Jacobi é representado por

$$x_i^{(k+1)} = \frac{1}{d_i}\left(A^T b_i - \sum_{\substack{j=1,\\ i\neq j}}^{n} [A^T A]_{ij}\, x_j^{(k)} \right),$$

para $i = 1, \ldots, n$.

Note que, o método de Jacobi é simetrizável pois, como a matriz de iteração do método de Jacobi dada por (3.2.7) é $G_J = I - D_A^{-1} A^T A$, temos

$$D_A^{1/2}(I - G_J)D_A^{-1/2} = D_A^{1/2} A^T A D_A^{-1/2}.$$

Antes de demonstrar dois resultados de convergência para o método de Jacobi, definamos o conceito de matriz estritamente diagonal dominante por linhas.

Definição 3.5. *Uma matriz $A \in \mathbb{R}^{n\times n}$ é chamada estritamente diagonal dominante por linhas se*

$$|a_{ii}| > \sum_{\substack{j=1,\\ i\neq j}}^{n} |a_{ij}|, \quad i = 1, \ldots, n.$$

Uma matriz $A \in \mathbb{R}^{n\times n}$ é chamada estritamente diagonal dominante por colunas se

$$|a_{ii}| > \sum_{\substack{i=1,\\ j\neq i}}^{n} |a_{ij}|, \quad j = 1, \ldots, n.$$

Uma matriz $A \in \mathbb{R}^{n\times n}$ é chamada estritamente diagonal dominante se é estritamente diagonal dominante por linhas e por colunas.

[3] MacTutor: `https://mathshistory.st-andrews.ac.uk/Biographies/Jacobi/`

Teorema 3.6. *O método de Jacobi converge para qualquer $x^{(0)}$ inicial se $A^T A$ for estritamente diagonal dominante.*

Demonstração. Vamos demonstrar que $\|G_J\|_\infty < 1$. Com efeito,

$$\|G_J\|_\infty = \max_{1\leqslant i\leqslant n} \sum_{j=1}^{n} |g_{ij}| \overset{(3.2.7)}{=} \max_{1\leqslant i\leqslant n} \sum_{j=1}^{n} \frac{|a_{ij}|}{|a_{ii}|} < 1.$$

Portanto, pelos teoremas 3.3 e 3.4, o método de Jacobi converge para qualquer condição inicial. □

Se assumirmos que $A^T A$ é estritamente diagonal dominante por colunas, então o método de Jacobi converge. A demonstração é simples, basta considerar a 1-norma de A. Nem toda matriz definida positiva é estritamente diagonal dominante por linhas e/ou colunas e, portanto, essa proposição não garante a convergência do método de Jacobi.

Podemos demonstrar outro resultado de convergência para o método de Jacobi, que é corolário do Teorema de Householder[4]-John[5] [76, 104].

Teorema 3.7 (Householder-John)**.** *Sejam $A, B \in \mathbb{R}^{n\times n}$ duas matrizes tais que A e $A - B - B^T$ são simétricas definidas positivas. Então $\rho(G) < 1$, onde $G = -(A - B)^{-1}B$.*

Demonstração. Sejam λ um autovalor de G e $x \neq 0$ o autovetor associado, ou seja, $Gx = \lambda\, x$. Pela definição de G,

$$Gx = \lambda\, x \;\Leftrightarrow\; -Bx = \lambda(A - B)x.$$

Note que, $\lambda \neq 1$, pois caso contrário A seria singular, contradizendo a hipótese. Portanto,

$$\begin{aligned} -x^*Bx = \lambda\, x^*(A - B)x \;&\Leftrightarrow\; \lambda\, x^*Bx - x^*Bx = \lambda x^*Ax \\ &\Leftrightarrow\; x^*Bx = \frac{\lambda}{\lambda - 1} x^*Ax. \end{aligned} \tag{3.2.9}$$

Observe que, λ e x podem eventualmente tomar valores no corpo dos complexos. Ademais, escrevendo $x = x_1 + i\, x_2$, então

$$x^*Ax = (x_1 + ix_2)^*A(x_1 + ix_2) = x_1^T Ax_1 + x_2^T Ax_2 > 0,$$

pois A é definida positiva. Pelo mesmo argumento, $x^*(A - B - B^T)x > 0$. Portanto,

$$\begin{aligned} 0 \;&<\; x^*Ax - x^*Bx - x^*B^T x \overset{(3.2.9)}{=} \left(1 - \frac{\lambda}{\lambda - 1} - \frac{\overline{\lambda}}{\overline{\lambda} - 1}\right) x^*Ax \\ &=\; \frac{1 - |\lambda|^2}{|\lambda - 1|} x^*Ax. \end{aligned}$$

Como $\lambda \neq 1$, então $|\lambda - 1| > 0$. Assim, $1 - |\lambda|^2 > 0$, ou seja, $|\lambda| < 1$. □

[4]MacTutor: `https://mathshistory.st-andrews.ac.uk/Biographies/Householder/`
[5]MacTutor: `https://mathshistory.st-andrews.ac.uk/Biographies/John/`

Corolário 3.1. *Considere o sistema normal $A^TAx = A^Tb$. Se $2D_A - A^TA$ é simétrica definida positiva, então o método de Jacobi aplicado ao sistema normal é convergente.*

Demonstração. No Teorema 3.7 coloque $A = A^TA$ e $B = A^TA - D_A$. Com essa configuração, obtemos o método de Jacobi para o sistema normal. De fato,

$$\begin{aligned} -(A-B)^{-1}B &= -\left(A^TA - A^TA + D_A\right)^{-1}\left(A^TA - D_A\right) \\ &= D_A^{-1}\left(D_A - A^TA\right) \\ &\overset{(3.2.7)}{=} G_J. \end{aligned}$$

Mas, A^TA é definida positiva e

$$A^TA - \left(A^TA - D_A\right) - \left(A^TA - D_A\right)^T = 2D_A - A^TA,$$

que é definida positiva, por hipótese. Portanto, pelo Teorema 3.7, o método de Jacobi é convergente. □

O método de Jacobi pode ser utilizado para a resolução iterativa das equações normais de segunda ordem, obtendo

$$y^{(k+1)} = y^{(k)} + AD_A^{-1}(c - A^Ty^{(k)}).$$

3.2.3 Métodos de Redução Residual

Os métodos de redução residual foram introduzidos por de la Garza [46] para matrizes quadradas e, posteriormente, estudados com mais detalhes por Householder e Bauer [69].

Seja $p_j \notin \operatorname{Ker}(A)$, $j = 1, \ldots, n$ uma sequência de n vetores não nulos. Os métodos de redução residual são computados através das seguintes aproximações

$$\begin{cases} x^{(j+1)} = x^{(j)} + \alpha_j p_j, \\ \alpha_j = \dfrac{p_j^TA^T(b - Ax^{(j)})}{\|Ap_j\|_2^2}. \end{cases} \tag{3.2.10}$$

Seja $r^{(j)} = b - Ax^{(j)}$, o j-ésimo resíduo. Então $r^{(j+1)} \perp Ap_j$:

$$\begin{aligned}
\left\langle r^{(j+1)}, Ap_j \right\rangle &= \left\langle b - Ax^{(j+1)}, Ap_j \right\rangle \\
&= \left\langle b - A(x^{(j)} + \alpha_j p_j), Ap_j \right\rangle \\
&= \left\langle b - Ax^{(j)}, Ap_j \right\rangle - \langle \alpha_j Ap_j, Ap_j \rangle \\
&= \left\langle b - Ax^{(j)}, Ap_j \right\rangle - \left\langle \frac{p_j^T A^T (b - Ax^{(j)}) Ap_j}{\|Ap_j\|_2^2}, Ap_j \right\rangle \\
&= 0.
\end{aligned}$$

Assim, $\left\|r^{(j+1)}\right\|_2^2 = \left\|r^{(j)}\right\|_2^2 - |\alpha_j|^2 \|Ap_j\|_2^2 \leqslant \left\|r^{(j)}\right\|_2^2$. Por isso, a classe de métodos (3.2.10) é dita de redução residual.

3.2.4 Método de Gauss-Seidel

Sejam A e D_A como definidos para o método de Jacobi. O método de Gauss[6]-Seidel[7] é obtido através da fatoração da matriz das equações normais, $A^T A = L_A + D_A + L_A^T$, onde L_A é a matriz estritamente triangular inferior extraída de $A^T A$.

Tome $M = L_A + D_A$ e $N = -L_A^T$, substituindo em (3.1.1), temos

$$x^{(k+1)} = x^{(k)} + D_A^{-1} \left[A^T b - L_A x^{(k+1)} - \left(D_A + L_A^T \right) x^{(k)} \right]. \tag{3.2.11}$$

Componente a componente, o método de Gauss-Seidel é representado por

$$x_i^{(k+1)} = \frac{1}{d_i} \left(A^T b_i - \sum_{j=1}^{i-1} [A^T A]_{ij}\, x_j^{(k+1)} - \sum_{j=i+1}^{n} [A^T A]_{ij}\, x_j^{(k)} \right), \tag{3.2.12}$$

para $i = 1, \ldots, n$. A matriz de iteração do método de Gauss-Seidel é

$$G_{GS} = -(L_A + D_A)^{-1} L_A^T. \tag{3.2.13}$$

Observe que, por possuir uma coluna nula, a matriz G_{GS} é sempre singular, e que na equação (3.2.12) há a necessidade de se calcular o produto $A^T A$ sendo, portanto, um problema numérico. Mesmo assim, o método de Gauss-Seidel pode ser visto como um método de redução residual, com construção completamente distinta da que temos feito até agora. Björck e Elfving [14] apresentam essa discussão.

[6] `https://mathshistory.st-andrews.ac.uk/Biographies/Gauss/`
[7] `https://mathshistory.st-andrews.ac.uk/Biographies/Seidel/`

Considere que cada atualização de soluções aproximadas para o sistema linear $A^TAx = A^Tb$, onde $A = [a_1|a_2|\cdots|a_n] \in \mathbb{R}^{m\times n}$, é feita através de n passos intermediários da forma: $z^{(1)} = x^{(k)}$ e $x^{(k+1)} = z^{(n+1)}$, com

$$z^{(j+1)} = z^{(j)} + \delta_j e_j.$$

Assumimos que A tem posto completo e que $e_j \in \mathbb{R}^n$ são vetores unitários em ordem cíclica. O escalar δ_j é escolhido de forma que a j-ésima componente do resíduo seja nula, ou seja, tomando $r^{(j)} = b - Az^{(j)}$,

$$\begin{aligned} 0 &= \left\langle A^T r^{(j+1)}, e_j \right\rangle = \left\langle A^T b - A^T A(z^{(j)} + \delta_j e_j), e_j \right\rangle \\ &= \left\langle A^T (r^{(j)} - \delta_j A e_j), e_j \right\rangle. \end{aligned}$$

Portanto,

$$\delta_j = \frac{a_j^T r^{(j)}}{d_j}.$$

Assim, o método de Gauss-Seidel tem a forma

$$\begin{cases} z^{(1)} = x^{(k)}, \\ \delta_j = \dfrac{a_j^T r^{(j)}}{d_j}, \\ z^{(j+1)} = z^{(j)} + \delta_j e_j, \quad j = 1, \ldots, n, \\ r^{(j+1)} = b - Az^{(j+1)}, \\ x^{(k+1)} = z^{(n+1)}. \end{cases} \tag{3.2.14}$$

Uma vantagem dessa abordagem é que não há a necessidade de determinar A^TA. Ademais, no j-ésimo passo intermediário, somente a componente j de $z^{(j)}$ é alterada e, portanto, podemos atualizar o resíduo sem a necessidade de calcular $Az^{(j)}$ a cada iteração. Portanto, podemos reformular esse método:

Algoritmo 4 Sweep do método de Gauss-Seidel

```
function GS-Sweep(x^(k), r^(k), n)
    r := r^(k);
    z := x^(k);
    for j = 1 : n do
        δ_j = a_j^T r / d_j;
        z := z + δ_j e_j;
        r := r − δ_j a_j;
    end for
    r^(k+1) := r and x^(k+1) := z;
end function
```

Um fator importante nessa implementação é que o custo computacional depende apenas da esparsidade de A e/ou A^TA [125]. O método de Gauss-Seidel não é simetrizável, ao contrário dos métodos de Richardson de primeira-ordem e o de Jacobi, e a ordem das colunas de A influenciam na convergência do método [12]. Nesse caso, pode-se deduzir o método de Gauss-Seidel para resolver as equações normais de segunda-ordem. Para o leitor interessado nessa dedução, vide [12, 125]. Por fim, ambos os métodos de Jacobi e Gauss-Seidel podem ser generalizados para matrizes em bloco.

O método de Gauss-Seidel para problemas de quadrados mínimos é convergente para qualquer $x^{(0)}$, fato esse que é um corolário do teorema de convergência do método SOR, que veremos na próxima seção, para matrizes simétricas e definidas positivas. Também, pode ser demonstrado como um corolário do Teorema de Householder-John.

Teorema 3.8. *Considere o sistema normal $A^TAx = A^Tb$, com A posto completo, então o método de Gauss-Seidel aplicado a esse tipo de sistema linear é convergente.*

Demonstração. No Teorema 3.7 coloque $A = A^TA$ e $B = L_A^T$. Com essa configuração, obtemos o método de Gauss-Seidel para o sistema normal. Com efeito,

$$\begin{aligned} -(A-B)^{-1}B &= -\left(A^TA - L_A^T\right)^{-1} L_A^T \\ &= -\left(D_A + L_A\right)^{-1} L_A^T \\ &\overset{(3.2.13)}{=} G_{GS}. \end{aligned}$$

Mas, A^TA é definida positiva e

$$A^TA - L_A^T - \left(L_A^T\right)^T = D_A.$$

que é definida positiva, pois A^TA é definida positiva. Portanto, pelo Teorema 3.7, o método de Gauss-Seidel é convergente. □

O método que hoje chamamos de método Gauss-Seidel tem diversos nomes, por exemplo, na literatura russa ele é conhecido como método de Nékrasov [99]. Liebmann [89], nos estudos de soluções discretizadas para a equação de Poisson, desenvolveu o método que hoje chamamos de Gauss-Seidel e, por isso, quando é aplicado para determinar soluções numéricas de equações diferenciais parciais, é chamado de método de Liebmann [126].

3.2.5 Métodos SOR e SSOR

Uma forma de acelerar a convergência de um método iterativo é através de uma técnica chamada de relaxação sucessiva e pode ser aplicada, a princípio, em qualquer método iterativo de convergência lenta. Utilizaremos essa

técnica para acelerar a convergência do método de Gauss-Seidel e o chamaremos de SOR (*successive over-relaxation*). A ideia dos métodos SOR surge com David Young [160, 161] e Stanley Frankel [42], que chamou de método de Liebmann acelerado. Nos métodos SOR, em geral, a iteração $k+1$ é obtida como uma média da iteração k e do valor $x^{(k+1)}$ obtido por algum método iterativo, em nosso caso, Gauss-Seidel [29]. Assim,

$$x_{SOR}^{(k+1)} = (1-\omega)\, x_{SOR}^{(k)} + \omega\, x_{GS}^{(k+1)}.$$

O parâmetro ω é denominado parâmetro de relaxação e observe que, para $\omega = 1$, o método de Gauss-Seidel é recuperado. Portanto, a partir de (3.2.12), obtemos

$$x_i^{(k+1)} = (1-\omega)\, x_i^{(k)} + \frac{\omega}{d_i}\left(A^T b_i - \sum_{j=1}^{i-1} [A^T A]_{ij}\, x_j^{(k+1)} - \sum_{j=i+1}^{n} [A^T A]_{ij}\, x_j^{(k)} \right),$$

para $i = 1, \ldots, n$. Esse formato do método SOR é útil para demonstrar que o método é convergente se, somente se, $0 < \omega < 2$, já que $A^T A$ é definida positiva.

Sejam L_A e D_A como definidos para os métodos de Jacobi e Gauss-Seidel. O método SOR é obtido através da decomposição da matriz das equações normais, $A^T A = L_A + D_A + L_A^T$, onde L_A é a matriz estritamente triangular inferior extraída de $A^T A$. Tome

$$M = L_A + \frac{1}{\omega} D_A \quad \text{e} \quad N = \left(\frac{1}{\omega} - 1\right) D_A - L_A^T, \tag{3.2.15}$$

substituindo em (3.1.1), temos

$$x^{(k+1)} = (D_A + \omega L_A)^{-1} \left[(1-\omega) D_A - \omega L_A^T\right] x^{(k)} + (D_A + \omega L_A)^{-1} A^T b. \tag{3.2.16}$$

Para entender a questão da convergência do método SOR precisamos analisar a matriz $G = M^{-1} N$, que nesse caso é dada por

$$\begin{aligned} G_{SOR(\omega)} &= (D_A + \omega L_A)^{-1} \left[(1-\omega) D_A - \omega L_A^T\right] \\ &= \left(I + \omega D_A^{-1} L_A\right)^{-1} \left[(1-\omega) I - \omega D_A^{-1} L_A^T\right]. \end{aligned} \tag{3.2.17}$$

Como afirmamos anteriormente, o método é convergente se, somente se, $0 < \omega < 2$. Primeiramente, demonstraremos um teorema que não tem como hipótese que a matriz seja definida positiva. Porém, para manter a consistência de notação, continuaremos trabalhando com $A^T A$.

Teorema 3.9. *[61, pág. 150] Para qualquer $\omega \in \mathbb{R}$, temos*

$$\left|\rho\left(G_{SOR(\omega)}\right)\right| \geqslant |1-\omega|.$$

Demonstração. De (3.2.17), temos

$$\begin{aligned}
\det\left(G_{SOR(\omega)}\right) &= \det\left\{\left(I+\omega D_A^{-1}L_A\right)^{-1}\left[(1-\omega)I-\omega D_A^{-1}L_A^T\right]\right\} \\
&= \det\left[\left(I+\omega D_A^{-1}L_A\right)^{-1}\right]\cdot\det\left[(1-\omega)I-\omega D_A^{-1}L_A^T\right] \\
&= (1-\omega)^n\,.
\end{aligned}$$

A última igualdade segue do fato de $D_A^{-1}L_A$ e $D_A^{-1}L_A^T$ serem matrizes triangulares com diagonal principal formada por zeros. Por outro lado, o determinante de $G_{SOR(\omega)}$ é o produto de seus autovalores. Portanto, pelo menos um autovalor de $G_{SOR(\omega)}$ deve ser, em módulo, maior ou igual a $|1-\omega|$. $\square$

Queremos garantir a convergência do método SOR e, para isso, basta utilizar o Teorema 3.4:

$$|1-\omega| \leqslant \left|\rho\left(G_{SOR(\omega)}\right)\right| < 1 \quad\Leftrightarrow\quad \omega\in(0,2).$$

Esse resultado é uma condição necessária para a convergência do método SOR e é devido a William Kahan [72].

Porém, A^TA é definida positiva. Utilizaremos esse fato para demonstrar que todos os autovalores de $G_{SOR(\omega)}$ são em módulo menores que 1, ou seja, o método SOR para sistemas lineares normais é convergente se, somente se, $0<\omega<2$ [34, pág. 290]. Com efeito,

$$M = L_A + \frac{1}{\omega}D_A = \omega^{-1}(D_A+\omega L_A)$$

e, para aliviar a notação, chamemos

$$B = A^TA,\;\; L = L_A,\;\; D = D_A \;\text{ e }\; Q = B^{-1}(2M-B).$$

Note que $\mathrm{Re}(\lambda_i)(Q)>0$ para todo $i=1,\ldots,n$, pois dado $x\in\mathbb{C}^n$,

$$Qx=\lambda x \quad\Leftrightarrow\quad (2M-B)x=\lambda Bx \quad\Leftrightarrow\quad x^*(2M-B)x=\lambda\left(x^*Bx\right).$$

Somando a última identidade acima à sua transposta conjugada, obtemos

$$\begin{aligned}
x^*(M+M^T-B)x=\mathrm{Re}(\lambda)\left(x^*Bx\right) \quad&\Leftrightarrow\quad \mathrm{Re}(\lambda)=\frac{x^*(M+M^T-B)x}{x^*Bx} \\
&\Leftrightarrow\quad \mathrm{Re}(\lambda)=\left(\frac{2}{\omega}-1\right)\frac{x^*Dx}{x^*Bx}.
\end{aligned}$$

Como B e D são definidas positivas e $\frac{2}{\omega}-1>0$, então $\mathrm{Re}(\lambda_i)(Q)>0$, para todo $i=1,\ldots,n$. Note que,

$$(Q-I)(Q+I)^{-1} = 2B^{-1}(M-B)\frac{1}{2}M^{-1}B = I-M^{-1}B \overset{(3.1.2)}{=} G_{SOR(\omega)}.$$

Considere a série de Laurent [165] de

$$f(z) = \frac{z-1}{z+1} = \sum_{k=-\infty}^{\infty} a_k\,(z+1)^k.$$

Seja $Q = UTU^T$ a decomposição de Schur de Q. Portanto,

$$\begin{aligned} f(Q) &= (Q-I)(Q+I)^{-1} = \sum_{k=-\infty}^{\infty} a_k\,(Q+I)^k \\ &= U\left[\sum_{k=-\infty}^{\infty} a_k\,(T+I)^k\right]U^T = U f(T) U^T. \end{aligned}$$

Assim, $f(Q)$ e $f(T)$ têm os mesmos autovalores e, como a diagonal de T é formada pelos autovalores $\lambda_i(Q)$, temos que $f(\lambda_i(Q))$ são os autovalores de $f(Q)$. Logo,

$$\left|\lambda_i(G_{SOR(\omega)})\right| = |\lambda_i(f(Q))| = |f(\lambda_i(Q))| = \left|\frac{\lambda_i(Q)-1}{\lambda_i(Q)+1}\right| < 1.$$

Tal resultado é conhecido como Teorema de Ostrowski-Reich [105, 117]. A extensão do teorema para matrizes não simétricas (hermitianas) foi realizada por Ortega e Plemmons [104] e, posteriormente, Yuan [163] estendeu esse resultado para matrizes singulares.

Vejamos agora como $\rho(G_J)$, $\rho(G_{GS})$ e $\rho(G_{SOR(\omega)})$ estão relacionados, onde as matrizes de iteração dos métodos de Jacobi, Gaus-Seidel e SOR são dadas por (3.2.7), (3.2.13), (3.2.17), respectivamente.

Definição 3.6. *A matriz B possui a "propriedade A" se existe uma matriz de permutação P, tal que*

$$PBP^T = \begin{bmatrix} D_1 & U_1 \\ L_1 & D_2 \end{bmatrix},$$

onde D_1 e D_2 são matrizes diagonais.

Essa definição pode ser pensada em termos de grafos, vide [34]. Se uma matriz satisfaz a propriedade A, então

$$\begin{aligned} PBP^T &= \begin{bmatrix} D_1 & U_1 \\ L_1 & D_2 \end{bmatrix} \\ &= \begin{bmatrix} D_1 & 0 \\ 0 & D_2 \end{bmatrix} - \begin{bmatrix} 0 & 0 \\ -L_1 & 0 \end{bmatrix} - \begin{bmatrix} 0 & -U_1 \\ 0 & 0 \end{bmatrix} \qquad (3.2.18) \\ &= D - \tilde{L} - \tilde{U}. \end{aligned}$$

Definição 3.7. *Seja B uma matriz com decomposição $B = D(I - L - U)$, onde D é uma matriz diagonal não singular, L é uma matriz estritamente triangular inferior, e U é uma matriz estritamente triangular superior. Assim, B é dita consistentemente ordenada se os autovalores de*

$$G(\alpha) = \alpha\, L + \alpha^{-1}\, U, \qquad \alpha \neq 0,$$

são independentes de α.

Teorema 3.10. *Uma matriz B que satisfaz a propriedade A é consistentemente ordenada.*

Demonstração. Suponha que B satisfaça a propriedade A. Então, pela igualdade (3.2.18),

$$\begin{aligned}
B &= P^T \Big(D - (\tilde{L} + \tilde{U}) \Big) P \\
&= \hat{D} - (\hat{L} + \hat{U}) \\
&= \hat{D} - \hat{L} - \hat{U} \\
&= \hat{D}(I - \hat{D}^{-1}\hat{L} - \hat{D}^{-1}\hat{U}) \\
&= \hat{D}(I - L - U) \\
&= \begin{bmatrix} \hat{D}_1 & 0 \\ 0 & \hat{D}_2 \end{bmatrix} \left(I - \begin{bmatrix} 0 & -U_1 \\ -L_1 & 0 \end{bmatrix} \right).
\end{aligned}$$

Da primeira para a segunda linha cabe uma explicação. Note que, a conjugação de uma matriz diagonal por uma matriz de permutação, ainda é uma matriz diagonal. Já a conjugação de uma matriz com diagonal nula, nesse caso, $\tilde{L} + \tilde{U}$, ainda é uma matriz com diagonal nula que, posteriormente, pode ser decomposta na soma de uma matriz estritamente triangular inferior e uma matriz estritamente triangular superior, nesse caso $\hat{L} + \hat{U}$. Considere, para $\alpha \neq 0$,

$$G(\alpha) = \alpha\, L + \alpha^{-1}\, U = \begin{bmatrix} 0 & -\alpha^{-1}\, U_1 \\ -\alpha\, L_1 & 0 \end{bmatrix}.$$

Como

$$\begin{bmatrix} I & 0 \\ 0 & \alpha\, I \end{bmatrix}^{-1} G(\alpha) \begin{bmatrix} I & 0 \\ 0 & \alpha\, I \end{bmatrix} = \begin{bmatrix} 0 & -U_1 \\ -L_1 & 0 \end{bmatrix} = G(1),$$

$G(\alpha)$ e $G(1)$ são similares e, portanto, possuem os mesmos autovalores. Mas, os autovalores de $G(1)$ independem de α. Assim, B é consistentemente ordenada. $\square$

Definamos

$$G_J(\alpha) = \alpha\, D_A^{-1} L_A + \alpha^{-1}\, D_A^{-1} L_A^T = \alpha\, L + \alpha^{-1}\, U.$$

Observe que, por (3.2.7), $G_J(1) = G_J$. Portanto, de forma análoga à demonstração do Teorema 3.10 verifica-se que a matriz de iteração de Jacobi, G_J, é consistentemente ordenada.

Teorema 3.11. *[34, pág. 293] Se $A^T A$ é uma matriz consistentemente ordenada e $\omega \neq 0$, então valem as seguintes afirmações.*

1. *Os autovalores de G_J aparecem em pares, um positivo e o outro negativo, de mesma multiplicidade.*

2. *Se μ é um autovalor de G_J e $(\lambda+\omega-1)^2 = \lambda\omega^2\mu^2$, então λ é um autovalor de $G_{SOR(\omega)}$.*

3. *Reciprocamente, se $\lambda \neq 0$ é um autovalor de $G_{SOR(\omega)}$ e $(\lambda+\omega-1)^2 = \lambda\omega^2\mu^2$, então μ é um autovalor de G_J.*

Demonstração. 1. A matriz de iteração G_J é consistentemente ordenada, portanto os autovalores de $G_J(\alpha)$ são independentes do parâmetro α, mais especificamente, os autovalores de $G_J(\alpha)$ são os autovalores de G_J. Note que, $G_J = G_J(1)$ e $G_J(-1) = -G_J(1)$ possuem os mesmos autovalores e em pares $\pm$.

2. Assuma que μ é um autovalor de G_J e que $(\lambda+\omega-1)^2 = \lambda\omega^2\mu^2$. Para $\lambda = 0$, obtemos $\omega = 1$, ou seja, 0 é autovalor de $G_{SOR(1)} = G_{GS}$, pois G_{GS} é singular. Para $\lambda \neq 0$,

$$\begin{aligned}
0 &= \det\left(\lambda\, I - G_{SOR(\omega)}\right) \\
&= \det\left[\left(I + \omega\, D_A^{-1} L_A\right)\left(\lambda\, I - G_{SOR(\omega)}\right)\right] \\
&= \det\left[(\lambda+\omega-1)I + \omega\lambda\, D_A^{-1} L_A + \omega\, D_A^{-1} L_A^T\right] \\
&= \det\left\{\omega\sqrt{\lambda}\left[\left(\frac{\lambda+\omega-1}{\omega\sqrt{\lambda}}\right) I + \sqrt{\lambda}\, D_A^{-1} L_A + \frac{1}{\sqrt{\lambda}}\, D_A^{-1} L_A^T\right]\right\} \\
&= \left(\omega\sqrt{\lambda}\right)^n \det\left[\left(\frac{\lambda+\omega-1}{\omega\sqrt{\lambda}}\right) I + D_A^{-1} L_A + D_A^{-1} L_A^T\right] \\
&= \left(\omega\sqrt{\lambda}\right)^n \det\left[\left(\frac{\lambda+\omega-1}{\omega\sqrt{\lambda}}\right) I - G_J\right] \\
&= \left(\omega\sqrt{\lambda}\right)^n \det\left(\mu I - G_J\right).
\end{aligned}$$

A antepenúltima igualdade segue do fato que a matriz de iteração G_J é consistentemente ordenada. Assim,

$$\mu = \frac{\lambda + \omega - 1}{\omega\sqrt{\lambda}} \quad \Leftrightarrow \quad (\lambda + \omega - 1)^2 = \lambda\omega^2\mu^2.$$

3. Se $\lambda \neq 0$, as igualdades da demonstração do item 2 podem ser feitas de baixo para cima.

□

O método de Gauss-Seidel é obtido atribuindo $\omega = 1$ ao método SOR e, a partir desse teorema que acabamos de demonstrar, verifica-se que $\lambda^2 = \lambda\mu^2$. Portanto, se $\lambda \neq 0$, então $\lambda = \mu^2$ e $\rho(G_{GS}) = [\rho(G_J)]^2$, demonstrando assim, que a convergência do método de Gauss-Seidel é mais rápida que a convergência do método de Jacobi, na verdade duas vezes mais rápida (Definição 3.3). Ademais, se A^TA satisfaz algumas condições, então é possível determinar uma fórmula para o parâmetro de relaxação ótimo, ω_{opt}, para o método SOR.

Teorema 3.12. *[36, 61] Suponha que A^TA seja consistentemente ordenada, G_J tenha autovalores reais e $\alpha = \rho(G_J) < 1$. Então,*

$$\begin{cases} \omega_{opt} = \dfrac{2}{1 + \sqrt{1 - \alpha^2}}, \\[2ex] \rho\left(G_{SOR(\omega)}\right) = \begin{cases} \dfrac{1}{4}\left(\alpha\omega + \sqrt{\alpha^2\omega^2 - 4(\omega - 1)}\right), & 0 < \omega \leqslant \omega_{opt}, \\[2ex] \omega - 1, & \omega_{opt} \leqslant \omega < 2, \end{cases} \\[2ex] \rho\left(G_{SOR(\omega_{opt})}\right) = \omega_{opt} - 1 = \dfrac{\mu^2}{\left(1 + \sqrt{1 - \mu^2}\right)^2}, \\[2ex] \rho\left(G_{SOR(\omega_{opt})}\right) < \rho\left(G_{SOR(\omega)}\right), \qquad \omega \in (0, 2) \setminus \{\omega_{opt}\}. \end{cases}$$

Demonstração. As raízes de $(\lambda + \omega - 1)^2 = \lambda\omega^2\mu^2$ são

$$\lambda = \left(\frac{\mu\omega \pm \sqrt{\mu^2\omega^2 - 4\omega + 4}}{2}\right)^2. \tag{3.2.19}$$

Pelo Teorema 3.11, se μ é autovalor de G_J, então ambos os valores para λ expressos em (3.2.19) são autovalores de $G_{SOR(\omega)}$. Para que as raízes acima sejam reais, precisamos que $\mu^2\omega^2 - 4\omega + 4 \geqslant 0$. Porém,

$$\lambda = \mu^2\omega^2 - 4\omega + 4 = 0 \quad \Leftrightarrow \quad \omega = \frac{2 \pm 2\sqrt{1 - \mu^2}}{\mu^2}.$$

Portanto, as raízes (3.2.19) são reais e positivas quando

$$0 < \omega \leqslant \tilde{\omega} = \frac{2 - 2\sqrt{1-\mu^2}}{\mu^2} = \frac{2}{1+\sqrt{1-\mu^2}}.$$

Nesse caso, ambas as raízes em (3.2.19) são positivas e a maior raiz é

$$\frac{1}{4}\left(\mu\omega + \sqrt{\mu^2\omega^2 - 4(\omega-1)}\right). \tag{3.2.20}$$

Essa expressão, para algum $\omega \in (0, \tilde{\omega}]$ é estritamente crescente como uma função de μ. Já para $\tilde{\omega} < \omega < 2$, as raízes (3.2.19) são complexas de módulo $\omega - 1$, pois

$$|\lambda| = \frac{1}{4}\left[\mu^2\omega^2 - \mu^2\omega^2 + 4(\omega-1)\right] = \omega - 1.$$

Portanto, por continuidade (ou simples verificação), $\lambda(\tilde{\omega}) = \tilde{\omega} - 1$. Agora, observe que $\tilde{\omega}$, vista como uma função de μ, é estritamente crescente e, portanto,

$$\tilde{\omega} \leqslant \frac{2 - 2\sqrt{1-\alpha^2}}{\mu^2} = \frac{2}{1+\sqrt{1-\alpha^2}} = \omega_{opt}.$$

Fixando $\mu = \alpha = \rho(G_J)$, em (3.2.20), observa-se que a função em ω é estritamente decrescente no intervalo $(0, \omega_{opt}]$ e, portanto, $\rho\left(G_{SOR(\omega_{opt})}\right) < \rho\left(G_{SOR(\omega)}\right)$, para $\omega \in (0,2) \setminus \{\omega_{opt}\}$. □

Para visualizar esses fatos apresentamos na Figura 3.1 um gráfico do raio espectral da matriz de iteração $G_{SOR(\omega)}$ em função do parâmetro de relaxação ω para quatro diferentes $\alpha = \rho(G_J)$. Pode-se observar claramente que $\rho\left(G_{SOR(\omega_{opt})}\right) < \rho\left(G_{SOR(\omega)}\right)$, para $\omega \in (0,2) \setminus \{\omega_{opt}\}$. Após atingir ω_{opt}, os autovalores de $G_{SOR(\omega)}$ se tornam complexos, porém com módulo $\omega - 1$.

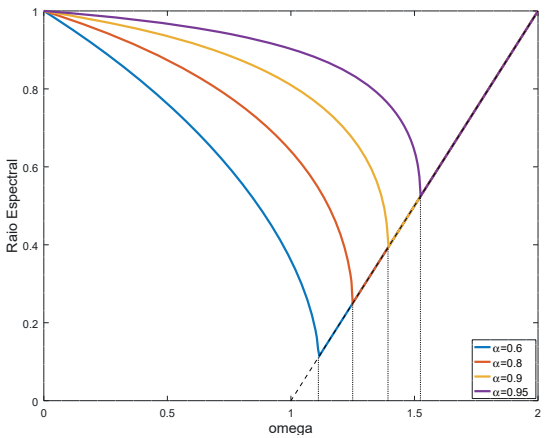

Figura 3.1: Raio espectral da matriz de iteração $G_{SOR(\omega)}$ para quatro diferentes $\alpha = \rho(G_J)$.

I. S. Katz [75] fornece outras estimativas para a taxa de convergência do método SOR já que, segundo o próprio autor, outros autores ofereceram "estimativas bastante rudimentares da taxa de convergência" do método SOR.

Todo esse trabalho foi feito para estudar a convergência do método SOR. Porém, a implementação do método como descrito em (3.2.16) necessita da formação de $A^T A$. Seguindo a ideia da implementação do método de Gauss-Seidel podemos implementar o método SOR como um método de redução residual equivalente, como a seguir.

Algoritmo 5 Sweep do método SOR

function SOR-SWEEP($x^{(k)}, r^{(k)}, n$)
  $r := r^{(k)}$;
  $z := x^{(k)}$;
  for $j = 1 : n$ **do**
    $\delta_j = \omega \dfrac{a_j^T r}{d_j}$;
    $z := z + \delta_j e_j$;
    $r := r - \delta_j a_j$;
  end for
  $r^{(k+1)} := r$ and $x^{(k+1)} := z$;
end function

Para a determinação do método SOR, que chamaremos de avançado, as M e N foram determinadas em (3.2.15). Porém, poderíamos ter escolhido as matrizes M e N como

$$M = L_A^T + \frac{1}{\omega} D_A \quad \text{e} \quad N = \left(\frac{1}{\omega} - 1\right) D_A - L_A,$$

dando origem ao método SOR atrasado. Portanto, o método SOR atualiza de cima para baixo, enquanto o atrasado, de baixo para cima. Nesse caso, a implementação de um passo do método SOR atrasado é dado por

Algoritmo 6 Sweep do método SOR atrasado

function SOR-SWEEP($x^{(k)}, r^{(k)}, n$)
  $r := r^{(k)}$;
  $z := x^{(k)}$;
  for $j = n : 1$ **do**
    $\delta_j = \omega \dfrac{a_j^T r}{d_j}$;
    $z := z + \delta_j e_j$;
    $r := r - \delta_j a_j$;
  end for
  $r^{(k+1)} := r$ and $x^{(k+1)} := z$;
end function

Como $A^T A$ é simétrica, podemos combinar os métodos SOR avançado e

atrasado da seguinte forma,

$$\begin{cases} (D_A + \omega L_A)\, y^{(k+1)} &= \left[(1-\omega)D_A - \omega L_A^T\right] x^{(k)} + A^T b, \\ \left(D_A + \omega L_A^T\right) x^{(k+1)} &= \left[(1-\omega)D_A - \omega L_A\right] y^{(k+1)} + A^T b. \end{cases}$$

Esse método iterativo é chamado de método SOR simétrico (SSOR), e foi introduzido por John W. Sheldon [131] no estudo de soluções numéricas para a equação de Laplace. Uma análise sobre aspectos da convergência é encontrada em [162]. Observe que, se definirmos M e N como em (3.2.15), então uma iteração do método SSOR é dada por,

$$\begin{cases} My^{(k+1)} &= Nx^{(k)} + A^T b, \\ M^T x^{(k+1)} &= N^T y^{(k+1)} + A^T b. \end{cases} \tag{3.2.21}$$

Assim, substituindo $y^{(k+1)}$ da primeira identidade na segunda, obtemos

$$\begin{aligned} x^{(k+1)} &= M^{-T} N^T M^{-1} N\, x^{(k)} + M^{-T}\left(M + N^T\right) M^{-1} A^T b \\ &= G_{SSOR}\, x^{(k)} + M_{SSOR}^{-1} A^T b. \end{aligned}$$

Portanto, $G_{SSOR} = M^{-T} N^T M^{-1} N$ e

$$\begin{aligned} M_{SSOR} &= M(M + N^T)^{-1} M^T \\ &= M\left[L_A + \frac{1}{\omega} D_A + \left(\frac{1}{\omega} - 1\right) D_A - L_A\right]^{-1} M^T \\ &= M\left[\frac{2-\omega}{\omega} D_A\right]^{-1} M^T \\ &= \frac{\omega}{2-\omega}\left(L_A + \frac{1}{\omega} D_A\right) D_A^{-1}\left(L_A^T + \frac{1}{\omega} D_A\right). \end{aligned}$$

Como $A^T A$ tem diagonal positiva, então M_{SSOR} é definida positiva. Ademais, M_{SSOR} está definida se $\omega \in (0,2)$ e, portanto, podemos concluir que o método SSOR é sempre convergente para $\omega \in (0,2)$.

Teorema 3.13. *[58, pág. 620] Considere o problema de quadrados mínimos $A^T A = A^T b$, com $A \in \mathbb{R}^{m\times n}$, $\omega \in (0,2)$,*

$$M = \frac{1}{\omega} D_A + L_A \quad e \quad N = \left(\frac{1}{\omega} - 1\right) D_A - L_A^T.$$

Então, a matriz de iteração do método SSOR, $G_{SSOR} = M^{-T} N^T M^{-1} N$, tem autovalores reais e $\rho\left(G_{SSOR}\right) < 1$.

Demonstração. Como $D_A = \operatorname{diag}(d_1, \ldots, d_n)$, a diagonal de $A^T A$, tem entradas positivas por ser definida positiva ($d_j = a_j^T a_j = \|a_j\|^2$). Então, podemos definir $D_A^{1/2} = \operatorname{diag}(\sqrt{d_1}, \ldots, \sqrt{d_n})$, ou seja, $D_A = D_A^{1/2} D_A^{1/2}$. Seja $G_1 = D_A^{1/2} G_{SSOR} D_A^{-1/2}$ e $L = D_A^{-1/2} L_A D_A^{-1/2}$. Para determinar G_1 vamos analisar por partes,

$$\begin{cases} D_A^{1/2}\left(D_A + \omega L_A^T\right)^{-1} = \left(I + \omega L^T\right)^{-1} D_A^{-1/2}, \\ [(1-\omega)D_A - \omega L_A] = D_A^{1/2}\left[(1-\omega)D_A^{1/2} - \omega D_A^{-1/2} L_A\right], \\ (D_A + \omega L_A)^{-1} = D^{-1/2}\left(D^{1/2} + \omega L_A D^{-1/2}\right)^{-1}, \\ \left[(1-\omega)D_A - \omega L_A^T\right] D_A^{-1/2} = D^{1/2}\left[(1-\omega)I - \omega L^T\right]. \end{cases} \tag{3.2.22}$$

A matriz G_1 é encontrada a partir do produto das quatro identidades (3.2.22). Portanto,

$$G_1 = \left(I + \omega L^T\right)^{-1} [(1-\omega)I - \omega L] (I + \omega L)^{-1} \left[(1-\omega)I - \omega L^T\right].$$

Porém,

$$\begin{aligned} [(1-\omega)I - \omega L] (I + \omega L)^{-1} &= I - \omega (I + \omega L)^{-1} - 2\omega L (I + \omega L)^{-1} \\ &= (I + \omega L)^{-1}\left[(I + \omega L) - \omega I - 2\omega (I + \omega L) L (I + \omega L)^{-1}\right] \\ &= (I + \omega L)^{-1}\left[(I + \omega L) - \omega I - 2\omega L (I + \omega L) (I + \omega L)^{-1}\right] \\ &= (I + \omega L)^{-1} [(1-\omega)I - \omega L]. \end{aligned}$$

Logo,

$$G_1 = \left(I + \omega L^T\right)^{-1} (I + \omega L)^{-1} [(1-\omega)I - \omega L] \left[(1-\omega)I - \omega L^T\right].$$

Afirmamos que, se $\lambda \in \sigma(G_1)$, então $\lambda \in [0, 1)$. Com efeito, seja $v \in \mathbb{R}^n$ um autovetor de G_1 associado a λ, isto é, $G_1 v = \lambda v$. Logo,

$$[(1-\omega)I - \omega L] \left[(1-\omega)I - \omega L^T\right] v = \lambda (I + \omega L) \left(I + \omega L^T\right) v.$$

Esse problema é um problema de valor singular generalizado ([58, pág. 501]). Portanto, λ é real e $\lambda \geqslant 0$. Assuma, sem perda de generalidade, que $\|v\|_2 = 1$.

Assim, temos

$$\begin{aligned}
\lambda &= \frac{\left\|(1-\omega)v - \omega L^T v\right\|_2^2}{\|v + \omega L^T v\|_2^2} \\
&= \frac{(1-\omega)^2 - 2\omega(1-\omega)v^T L^T v + \omega^2 \left\|L^T v\right\|_2^2}{1 + 2\omega v^T L^T v + \omega^2 \|L^T v\|_2^2} \\
&= 1 - \omega(2-\omega)\frac{1 + 2v^T L^T v}{\|v + \omega L^T v\|_2^2}.
\end{aligned}$$

Porém, como $\omega \in (0,2)$, então $\omega(2-\omega) > 0$. Ademais,

$$\begin{aligned}
v^T D_A^{-1/2}(A^T A)D_A^{-1/2} v &= v^T D_A^{-1/2}\left(D_A + L_A + L_A^T\right) D_A^{-1/2} v \\
&= 1 + 2v^T L^T v.
\end{aligned}$$

Como $A^T A$ é definida positiva, então $1+2v^T L^T v > 0$ e, portanto, $\lambda < 1$. $\square$

Uma iteração do método SOR simétrico é dada em (3.2.21). Assim, baseando-se na implementação dos métodos SOR avançado e atrasado podemos implementar o método SSOR da seguinte maneira,

Algoritmo 7 Sweep do método SSOR

```
function SOR-Sweep(x^(k), r^(k), n)
    r := r^(k);
    z := x^(k);
    for j = 1 : n do
        δ_j = ω (a_j^T r)/d_j;
        z := z + δ_j e_j;
        r := r − δ_j a_j;
    end for
    for j = n : 1 do
        δ_j = ω (a_j^T r)/d_j;
        z := z + δ_j e_j;
        r := r − δ_j a_j;
    end for
    r^(k+1) := r and x^(k+1) := z;
end function
```

Em geral, o método SSOR não apresenta vantagens quando comparado com o SOR, porém é utilizado como precondicionador de métodos não estacionários.

Vejamos uma simples comparação entre a *performance* dos métodos de Gauss-Seidel e SOR. Primeiramente, comparamos os dois métodos quando aplicados a uma matriz A gerada pelo comando MATLAB "sprand(20, 15, .6)"

que cria uma matriz esparsa 20×15 com 60% de densidade, e com o vetor b que foi criado com o comando "rand(20, 1)" que cria um vetor coluna com entradas aleatórias. Para esse teste utilizamos $\omega = 1.1$ no método SOR. Obtivemos $\rho(G_{GS}) = 0.95981$ e $\rho(G_{SOR}) = 0.94639$. Como o raio espectral de G_{SOR} é menor que o raio espectral de G_{GS}, temos que o método SOR converge em menos iterações. De fato, com o critério de parada $\|x^{(k+1)} - x^{(k)}\|_\infty < 10^{-5}$, o método de Gauss-Seidel convergiu em 279 iterações, enquanto o SOR em 216 iterações.

Como já discutido, nem sempre o método SOR é mais eficiente que o método Gauss-Seidel. Para exemplificar esse fato, na Figura 3.2 apresentamos um gráfico de ω por número de iterações. Para a geração do gráfico utilizamos os mesmos A, b e critério de parada supracitados. Para esse exemplo, a melhor escolha de ω é 1.22 que converge em 145 iterações.

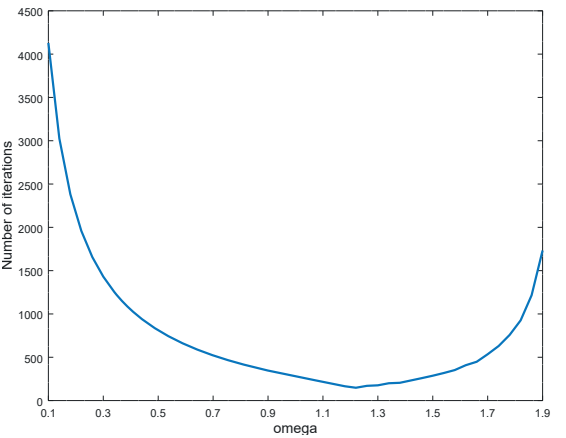

Figura 3.2: Comparação do número de iterações do método SOR para diferentes valores de ω.

Para um segundo experimento utilizamos a matriz "well1850" do *Matrix Market*[8] que é 1850×712 com 8758 entradas não nulas. Nesse caso, temos $\rho(G_{GS}) = 0.99948$ e $\rho(G_{SOR}) = 0.99863$, além disso, usamos uma tolerância de 10^{-3} e $\omega = 1.45$. O método de Gauss-Seidel converge em 1856 iterações e o método SOR em 1546 iterações. O método SOR é, nesse exemplo, por volta 17% mais eficiente. Pode parecer pouco, mas quando temos esses algoritmos como *solvers* dentro de um laço, 17% é um grande ganho em tempo de máquina.

Novamente, o estudo do método SOR para as equações normais de segundo tipo é bastante similar ao que apresentamos e, para um aprofundamento, recomendamos o livro de Björk [12]. Muitas propriedades (maioria

[8] `https://math.nist.gov/MatrixMarket/`

não coberta neste texto) do método SOR e métodos relacionados são discutidas por Hadjidimos [62].

3.3 Métodos Semi-Iterativos

Nesta seção apresentamos o conceito de métodos semi-iterativos focando no de Chebyshev. Para um bom embasamento, iniciamos com uma pequena discussão acerca dos polinômios de Chebyshev.

3.3.1 Polinômios de Chebyshev

As chamadas funções especiais surgem como soluções de certas equações diferenciais ordinárias de segunda ordem e, uma de suas principais aplicações, aparece em cálculos de física teórica. Os polinômios de Chebyshev são um caso particular da chamada função hipergeométrica ou de Gauss, e sua principal aplicação é em análise numérica. Em nosso caso, os utilizamos no estudo de aceleração nos chamados métodos semi-iterativos para a resolução de problemas de quadrados mínimos.

Eles também aparecem, por exemplo, no estudo de estimativas de autovalores de matrizes esparsas via processo de Lanczos. Apresentaremos apenas alguns fatos sobre polinômios de Chebyshev sem muito rigor matemático e para uma leitura mais aprofundada sobre o assunto sugerimos [93, 120]. Para uma leitura completa sobre funções especiais sugerimos [1, 116]. Por fim, a discussão a seguir é baseada em [2, 153].

A equação hipergeométrica é uma equação diferencial ordinária de segunda ordem dada por

$$z(1-z)\,\frac{d^2y}{dz^2} + [\gamma - (\alpha+\beta+1)z]\,\frac{dy}{dz} - \alpha\beta\, y = 0, \tag{3.3.23}$$

onde $\alpha, \beta, \gamma \in \mathbb{R}$. Observe que os pontos 0, 1 e ∞ são pontos singulares regulares dessa equação. Parece estranho que ∞ seja um ponto singular, mas aqui estamos considerando o plano complexo estendido ($\mathbb{C} \cup \{\infty\}$) e, portanto, é possível encontrar soluções em série para (3.3.23) através do método de Frobenius [153]. Pelo menos uma solução da equação hipergeométrica em torno de $z_0 = 0$ é analítica (há uma em torno de $z_0 = 1$ também), que é dada por

$${}_2F_1\,(\alpha\,,\,\beta\,;\,\gamma\,;\,z) = \sum_{n=0}^{\infty} \frac{(\alpha)_n(\beta)_n}{(\gamma)_n}\frac{z^n}{n!}. \tag{3.3.24}$$

Essa série é chamada de série hipergeométrica e $(\gamma)_n = \gamma(\gamma+1)\cdots(\gamma+n-1)$ é chamado de símbolo de Pochhammer e, por definição, $(\gamma)_0 = 1$. A notação ${}_2F_1$ é motivada pela existência de dois símbolos de Pochhammer no numerador e um no denominador da série. Existe, também, a ${}_1F_1$ que é a função hipergeométrica confluente, mas não estudaremos aqui. Precisamos

restringir os possíveis valores de γ para evitar divisão por zero. Nesse caso $\gamma \neq 0, -1, -2, \ldots$, pois para $\gamma = -k$, com k inteiro positivo, $(\gamma)_{k+1} = 0$.

Para os pontos onde a série hipergeométrica converge definimos a função hipergeométrica, e é fácil ver que para $|z| < 1$ essa série converge absolutamente, enquanto para $|z| > 1$, diverge. Já para $|z| = 1$ essa série converge se $\gamma - \alpha - \beta > 0$.

Há ainda a possibilidade de representar a função hipergeométrica, definida para $|z| < 1$, como uma integral no plano complexo retirando o segmento $[0, \infty]$. Sob essa condições,

$$ {}_2F_1\left(\alpha\,,\,\beta\,;\,\gamma\,;\,z\right) = \frac{1}{B(\beta, \gamma-\beta)} \int_0^1 t^{\beta-1}(1-t)^{\gamma-\beta-1}(1-tz)^{-\alpha} dt, $$

onde B é a função beta (outra função especial). Para definir a função hipergeométrica analisamos apenas uma das 24 soluções básicas da equação hipergeométrica, pois as outras 23 não são necessárias para a definição dos polinômios de Chebyshev.

A partir da série hipergeométrica (e da hipergeométrica confluente) podemos definir uma série de polinômios ortogonais que possuem inúmeras aplicações, entre eles os polinômios de Chebyshev. Existem dois tipos deles, o de tipo I e o de tipo II. Nós precisamos apenas dos de tipo I, que chamaremos apenas de polinômios de Chebyshev. O n-ésimo polinômio de Chebyshev $T_n(x)$ é definido da seguinte forma,

$$ T_n(x) = {}_2F_1\left(-n\,,\,n\,;\,\frac{1}{2}\,;\,\frac{1-x}{2}\right). $$

Uma propriedade importante dos polinômios de Chebyshev é que eles satisfazem a relação de recorrência

$$ T_{n+1}(x) - 2xT_n(x) + T_{n-1}(x) = 0. \tag{3.3.25} $$

Assim, se soubermos T_0 e T_1, então todos os outros podem ser deduzidos. Por exemplo,

$$ T_0(x) = {}_2F_1\left(0\,,\,0\,;\,\frac{1}{2}\,;\,\frac{1-x}{2}\right) \overset{(3.3.24)}{=} \sum_{n=0}^{\infty} \frac{(0)_n(0)_n}{(1/2)_n} \frac{(1-x)^n}{2^n n!} = 1. $$

Já o polinômio T_1 é dado por

$$ \begin{aligned} T_1(x) &= {}_2F_1\left(-1\,,\,1\,;\,\frac{1}{2}\,;\,\frac{1-x}{2}\right) \\ &= \sum_{n=0}^{\infty} \frac{(-1)_n(1)_n}{(1/2)_n} \frac{(1-x)^n}{2^n n!} \\ &= \sum_{n=0}^{\infty} \frac{(-1)_n\, n!}{2^{-n}(2n-1)!!} \frac{(1-x)^n}{2^n n!} \\ &= 1 - (1-x) = x. \end{aligned} $$

$T_0(x) = 1$
$T_1(x) = x$
$T_2(x) = 2x^2 - 1$
$T_3(x) = 4x^3 - 3x$
$T_4(x) = 8x^4 - 8x^2 + 1$
$T_5(x) = 16x^5 - 20x^3 + 5x$
$T_6(x) = 32x^6 - 48x^4 + 18x^2 - 1$

Tabela 3.1: Polinômios de Chebyshev.

Observe que $T_0(1) = 1$ e $T_1(1) = 1$ e, a partir da relação de recorrência para os polinômios de Chebyshev, pode-se demonstrar por indução que $T_n(1) = 1$, para todo n inteiro não negativo. Esse é dos fatos que justificam a utilização desses polinômios na aceleração de métodos iterativos. Os outros polinômios de Chebyshev são determinados através da relação de recorrência (3.3.25), conforme a Tabela 3.1.

Na Figura 3.3 apresentamos o comportamento de alguns polinômios de Chebyshev. Os gráficos foram feito no intervalo $[-1, 1]$, região onde os polinômios de Chebyshev são convergentes. Vale lembrar que a função hipergeométrica é absolutamente convergente em $|z| \in (-1, 1)$ e é convergente para $|z| = 1$ se $\gamma - \alpha - \beta > 0$. No caso dos polinômios de Chebyshev $\alpha = -n$, $\beta = n$ e $\gamma = 1/2$, a desigualdade é sempre satisfeita.

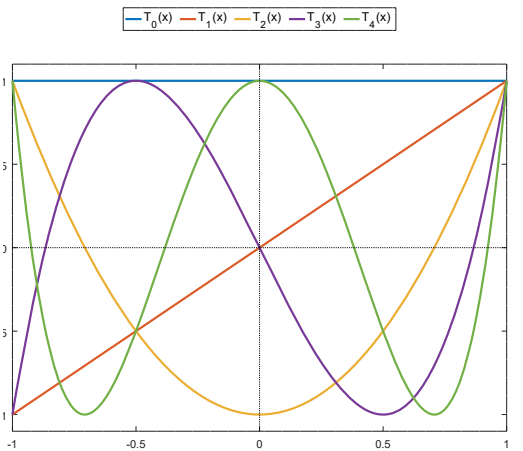

Figura 3.3: Polinômios de Chebyshev.

Uma propriedade que torna os polinômios de Chebyshev interessantes é que eles são ortogonais em $\mathcal{L}^2_\rho$, o espaço das funções quadrado integráveis com peso $\rho(x) > 1$, nesse caso $\rho(x) = 1/\sqrt{1-x^2}$. Assim

$$\int_{-1}^{1} T_m(x)T_n(x)\,\frac{1}{\sqrt{1-x^2}}\,dx = \begin{cases} 0 & \text{para } m \neq n, \\ \pi & \text{para } m = n = 0, \\ \dfrac{\pi}{2} & \text{para } m = n \neq 0. \end{cases}$$

Chebyshev foi provavelmente o primeiro matemático a reconhecer o conceito geral de polinômios ortogonais. Em seu trabalho de 1854, [22], onde os polinômios que recebem seu nome aparecem pela primeira vez, Chebyshev introduz os fundamentos da escola russa de teoria da aproximação.

3.3.2 O Método Semi-Iterativo de Chebyshev

Considere um método iterativo estacionário para determinar uma solução para $A^T Ax = A^T b$ dado por

$$Mx^{(k+1)} = Nx^{(k)} + A^T b. \tag{3.3.26}$$

Nessa seção queremos apresentar o conceito de aceleração polinomial de métodos iterativos. Para tanto, seguiremos a abordagem de [58] e assumiremos que $G = M^{-1}N = I - M^{-1}A^T A$ é simétrica e $\rho(G) < 1$. Caso o método iterativo estacionário seja simetrizável, basta seguir [12]. Uma análise de aceleração polinomial utilizando polinômios de Chebyshev é desenvolvida em [53, 54] e, por fim, uma boa revisão bibliográfica para métodos semi-iterativos sem a hipótese de G ser simétrica pode ser encontrada em [58, pág. 624].

Para acelerar a convergência do método iterativo consideramos a combinação linear das primeiras k aproximações, ou seja,

$$y^{(k)} = \sum_{i=1}^{k} v_{ik}\, x^{(i)},$$

onde $y^{(0)} = x^{(0)}$ e

$$\sum_{i=1}^{k} v_{ik} = 1,$$

que chamamos de método semi-iterativo com respeito ao método iterativo (3.3.26) ou método de aceleração polinomial. O objetivo é que $y^{(k)}$ represente uma melhor aproximação que $x^{(k)}$ para a solução do problema de quadrados mínimos. Para isso, considere o polinômio

$$p_k(t) = \sum_{i=1}^{k} v_{ik}\, t^i$$

com $p_k(1) = 1$. A expressão do erro para o método proposto é

$$y^{(k)} - x = \sum_{i=1}^{k} v_{ik}(x^{(i)} - x) = \sum_{i=1}^{k} v_{ik} G^i e^{(0)} = p_k(G)\, e^{(0)}. \tag{3.3.27}$$

Como G é simétrica, é ortogonalmente diagonalizável e, portanto, existe S ortogonal tal que $G = S^T DS$. Por outro lado, como $\rho(G) < 1$, existem reais α e β conhecidos de forma que $-1 < \alpha \leqslant \lambda_i \leqslant \beta < 1$. Por simplicidade, assuma $\alpha = -\beta$. Logo,

$$\|p_k(G)\|_2 = \|p_k(D)\|_2 = \max_{\lambda \in \sigma(D)} |p_k(\lambda)| \leqslant \max_{-\beta \leqslant \lambda \leqslant \beta} |p_k(\lambda)|\,.$$

Portanto, uma estimativa para a taxa de convergência depois de k passos é

$$\rho(p_k(G)) \leqslant \max_{-\beta \leqslant \lambda \leqslant \beta} |p_k(\lambda)|\,.$$

Queremos minimizar essa quantidade, ou seja, queremos encontrar p_k que satisfaça

$$\inf_{p_k \in \mathbb{R}_k^1[t]} \max_{\lambda \in [-\beta,\beta]} |p_k(\lambda)|,$$

onde $\mathbb{R}_k^1[t]$ representa o conjunto dos polinômios com coeficientes reais de grau k, tais que $p_k(1) = 1$. Uma discussão sobre a solução desse problema e que, de fato, essa escolha acelera a convergência pode ser encontrada em [152, pág. 149–156]. A solução é

$$p_k(z) = \frac{T_k(F(z))}{T_k(\mu)},$$

em que T_k é o k-ésimo polinômio de Chebyshev, F é o homeomorfismo entre $[-\beta, \beta]$ e $[-1, 1]$ dado por

$$F(z) = \frac{z+\beta}{\beta} - 1 = \frac{z}{\beta}.$$

e $\mu = F(1)$. Chamamos esse método de método semi-iterativo de Chebyshev com relação ao método iterativo (3.3.26). Note que, $p_k(1) = 1$ e p_k é limitado por $1/|T_k(\mu)|$ em $[-\beta, \beta]$. Da equação (3.3.27), obtemos

$$\|y^{(k)} - x\| \leqslant \|p_k(G)\| \ \|e^{(0)}\| \ \Rightarrow \ \|y^{(k)} - x\| \leqslant \frac{1}{|T_k(\mu)|} \|e^{(0)}\|.$$

A implementação desse método se torna problemática conforme k aumenta muito, e uma abordagem para contornar essa dificuldade é apresentada em [93]. Para calcular $y^{(k)}$ de forma mais eficiente utilizamos a seguinte relação de recorrência que os polinômios de Chebyshev satisfazem:

$$T_{k+1}(x) - 2xT_k(x) + T_{k-1}(x) = 0.$$

A partir dela e definindo,

$$\Phi = F(G) = \frac{1}{\beta} G,$$

obtemos

$$\begin{cases} T_{k+1}(\mu) &= 2xT_k(\mu) - T_{k-1}(\mu), \\ \\ T_{k+1}(\Phi) &= 2xT_k(\Phi) - T_{k-1}(\Phi). \end{cases}$$

Por outro lado,

$$\begin{aligned} y^{(k+1)} - y^{(k-1)} &= (y^{(k+1)} - x) - (y^{(k-1)} - x) \\ &= [p_{k+1}(G) - p_{k-1}(G)]e^{(0)} \\ &= \left(\frac{T_{k+1}(\Phi)}{T_{k+1}(\mu)} - \frac{T_{k-1}(\Phi)}{T_{k-1}(\mu)}\right) e^{(0)}, \end{aligned}$$

e

$$y^{(k)} - y^{(k-1)} = \left(\frac{T_k(\Phi)}{T_k(\mu)} - \frac{T_{k-1}(\Phi)}{T_{k-1}(\mu)} \right) e^{(0)}.$$

Defina,

$$\omega_{k+1} = 1 + \frac{T_{k-1}(\mu)}{T_{k+1}(\mu)} = 2\mu \, \frac{T_k(\mu)}{T_{k+1}(\mu)}.$$

Note que

$$(y^{(k+1)} - y^{(k-1)}) - \omega_{k+1}(y^{(k)} - y^{(k-1)}) =$$

$$= \left\{ \frac{1}{T_{k+1}(\mu)} \left[T_{k+1}(\Phi) - 2\mu T_k(\Phi) \right] - \frac{T_{k-1}(\Phi)}{T_{k-1}(\mu)} \left[1 - 2\mu \frac{T_k(\mu)}{T_{k+1}(\mu)} \right] \right\} e^{(0)},$$

e como

$$-\frac{T_{k-1}(\Phi)}{T_{k-1}(\mu)} \left[1 - 2\mu \frac{T_k(\mu)}{T_{k+1}(\mu)} \right] = \frac{T_{k-1}(\Phi)}{T_{k-1}(\mu)} \, \frac{T_{k-1}(\mu)}{T_{k+1}(\mu)} = \frac{T_{k-1}(\Phi)}{T_{k+1}(\mu)},$$

temos

$$\begin{aligned}
(y^{(k+1)} - y^{(k-1)}) - \omega_{k+1}(y^{(k)} - y^{(k-1)}) = & \\
= \quad & \frac{1}{T_{k+1}(\mu)} \left[T_{k+1}(\Phi) - 2\mu T_k(\Phi) + T_{k-1}(\Phi) \right] e^{(0)} \\
= \quad & \frac{2T_k(\Phi)}{T_{k+1}(\mu)} (\Phi - \mu I) e^{(0)} \\
= \quad & \frac{2T_k(\mu)}{\beta T_{k+1}(\mu)} (G - I) \frac{T_k(\Phi)}{T_k(\mu)} \, e^{(0)} \\
\overset{(3.3.27)}{=} \quad & \omega_{k+1}(G - I)(y^{(k)} - x) \\
= \quad & \omega_{k+1} \, z^{(k)}.
\end{aligned}$$

Assim,

$$y^{(k+1)} = y^{(k-1)} + \omega_{k+1}(y^{(k)} + z^{(k)} - y^{(k-1)}).$$

Ademais, $z^{(k)}$ satisfaz o seguinte sistema linear,

$$M z^{(k)} = M(G - I)(y^{(k)} - x) = A^T A(x - y^{(k)}) = A^T r^{(k)}.$$

Com essa implementação o método semi-iterativo de Chebyshev não precisa acessar os últimos $x^{(0)}, \ldots, x^{(k)}$ e realizar a combinação linear para determinar $y^{(k)}$. Seu algoritmo é dado por

Algoritmo 8 Método semi-iterativo de Chebyshev

```
function CHEBYSHEV(β, x^(0), A, b)
    c_0 = 1; c_1 = 1/β; y^(0) = x^(0);
    My^(1) = Ny^(0) + A^T b;
    r^(1) = b − Ay^(1); k = 1;
    while ‖r^(k)‖ ⩾ tol do
        c_{k+1} = (2/β)c_k − c_{k−1};
        ω_{k+1} = 1 + (c_{k−1}/c_{k+1});
        Mz^(k) = A^T r^(k);
        y^(k+1) = y^(k−1) + ω_{k+1}(y^(k) + z^(k) − y^(k−1));
        k = k + 1;
        r^(k) = b − Ay^(k);
    end while
end function
```

3.4 Exercícios

1. Mostre que se $x_k \to x^*$, para $k \to \infty$, então $\limsup_{k\to\infty} \|x_k - x^*\|^{1/k}$ é independente da norma.

2. Seja

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

Mostre que os métodos de Jacobi, Gauss-Seidel e SOR convergem quando aplicados a sistemas lineares da forma $Ax = b$ com tal matriz.

3. Mostre que se $A = M - N$ é singular, então não podemos ter $\rho(M^{-1}N) < 1$ mesmo que M seja não singular.

4. [5, pág. 164] Sejam $A^TA = M - N \in \mathbb{R}^{n\times n}$ com M não singular, $Mx^{(k+1)} = Nx^{(k)} + A^Tb$ um método iterativo estacionário e $G = M^{-1}N$. O fator $R_k = \|B^k\|^{1/k}$ é chamado de fator de convergência médio após k passos do método iterativo. Demonstre que:

1. Existem constantes positivas $c, C \in \mathbb{R}$ tais que

$$cm^{s-1}\rho(G)^k \leqslant \|B^k\| \leqslant Cm^{s-1}\rho(G)^k, \quad k = 1, 2, 3, \ldots$$

onde s é a ordem do maior bloco de Jordan associado a um autovalor λ, com $|\lambda| = \rho(G)$.

2. $R_k \to \rho(G)$, $k \to \infty$.

5. [58, pág. 623] Considere a iteração

$$y^{(k+1)} = \omega(By^{(k)} + d - y^{(k-1)}) + y^{(k-1)},$$

onde B tem a decomposição de Schur $Q^T BQ = \operatorname{diag}(\lambda_1, \ldots, \lambda_n)$ como $\lambda_1 \geqslant \cdots \geqslant \lambda_n$. Assuma que $x = Bx + d$.

1. Deduza uma equação para $e_k = y^{(k)} - x$.
2. Assuma $y^{(1)} = By^{(0)} + d$. Mostre que $e_k = p_k(B)e_0$, onde p_k é um polinômio par/ímpar se k é par/ímpar.
3. Escreva $f^{(k)} = Q^T e_k$. Deduza uma equação de diferenças para $f_j^{(k)}$, para $j = 1, \ldots, n$. Tente especificar a solução geral para $f_j^{(1)}$ e $f_j^{(0)}$ dados.
4. Mostre como determinar um ω ótimo.

6. [5, pág. 194] Seja $A^T A = M - N$, com M não singular. Considere $M(\alpha) = (1+\alpha)M$ e $N(\alpha) = M(\alpha) - A^T A = N + \alpha M$, e sejam $\lambda_1 \leqslant \lambda_2 \leqslant \cdots \lambda_n < 1$ autovalores de $M^{-1}N$. Demonstre que o método iterativo $M(\alpha)x^{(k+1)} = N(\alpha)x^{(k)} + A^T b$, $k = 1, 2, 3, \ldots$ converge para qualquer α, $\alpha > -(1 + \lambda_1)/2$ e $\min_\alpha \rho(M^{-1}(\alpha)N(\alpha)) = \rho(M^{-1}(\alpha^*)N(\alpha^*))$, onde $\alpha^* = -(\lambda_1 + \lambda_n)/2$.

7. Demonstre que a matriz de iteração do método de Gauss-Seidel é sempre singular.

8. O método de Gauss-Seidel é dado classicamente como (3.2.12). Deduza o método de Gauss-Seidel (3.2.14) como um método de redução residual.

9. [25, pág. 184] Para resolver o problema $Ax = b$ definimos uma sequência de decomposições regulares $A = M_k - N_k$, com $k = 1, 2, 3, \ldots$ e M_k não singular. Tal decomposição está associada ao método iterativo $x^{(k+1)} = M_k^{-1} N_k x^{(k)} + M_k^{-1} b$.

1. Demonstre que a condição $\rho(M_k^{-1} N_k) < 1$ para $k = 1, 2, 3, \ldots$ não é suficiente, em geral, para garantir convergência.
2. Suponha que exista $\delta > 0$ tal que $\rho(M_k^{-1} N_k) \leqslant 1 - \delta$ para todo k. O método iterativo converge?

10. Implemente os algoritmos apresentados no capítulo e utilize o modelo well1850 encontrado no *Matrix Market*.[9] Compare a performance de cada algoritmo.

11. [58, pág. 623] Suponha que queiramos resolver o problema de quadrados mínimos linear $\min \|Ax - b\|_2$, com $A \in \mathbb{R}^{m \times n}$, $\operatorname{rank}(A) = r \leqslant n$ e $b \in \mathbb{R}^m$. Considere o esquema iterativo

$$Mx^{(k+1)} = Nx^{(k)} + A^T b,$$

[9] `https://math.nist.gov/MatrixMarket/`

onde $M = (A^T A + \lambda W)$, $N = \lambda W$, $\lambda > 0$ e $W \in \mathbb{R}^{n \times n}$ é simétrica definida positiva.

1. Mostre que $M^{-1}N$ é diagonalizável e que $\rho(M^{-1}N) < 1$ se $\operatorname{rank}(A) = n$.
2. Suponha $x_0 = 0$ e que $\|v\|_W = (v^T W v)^{1/2}$, a W-norma. Mostre que independente do posto de A, a iterada $x^{(k)}$ converge para a solução mínima na W-norma.
3. Mostre que se $\operatorname{rank}(A) = n$, então $\|x_{\mathrm{LS}} - x^{(k+1)}\|_W \leqslant \|x_{\mathrm{LS}} - x^{(k)}\|_W$.
4. Mostre como implementar o esquema iterativo dada a fatoração QR de
$$M = \begin{bmatrix} A \\ \sqrt{\lambda} F \end{bmatrix}$$
onde $W = FF^T$ é a fatoração de Cholesky de W.

Capítulo 4

Métodos Iterativos em Subespaços Krylov

A resolução de problemas de quadrados mínimos lineares (sistemas lineares) pode ser efetuada de duas formas. A primeira forma é através de métodos diretos que, em grande parte, se baseiam em fatorações matriciais. A vantagem dessa abordagem é a determinação da solução do problema em um número finito de passos, enquanto a desvantagem é que eles podem destruir a esparsidade do problema e amplificar erros de arredondamento, entre outros.

A segunda forma é através de métodos iterativos, como os que temos estudado. Essa abordagem é baseada na construção de uma sequência que, sob certas hipóteses, converge para a solução. Dessa forma, o número de operações para a determinação não pode ser medido pois em teoria é infinito, porém essa abordagem, em geral, preserva a esparsidade do problema, é menos sensível a erros de arredondamento e consome menos memória.

Os métodos iterativos em subespaços de Krylov[1], também chamados de métodos semi-diretos, agregam algumas vantagens dos métodos diretos e dos iterativos. Sua principal vantagem é que a solução do problema, a menos de erros de arredondamento, é alcançada em um número finito de passos. Ademais, eles preservam a esparsidade, não são muito sensíveis a erros de arredondamento e não consomem muita memória.

Alexei Krylov em 1931 [79] desenvolveu o método para a resolução de uma equação que determina a frequência de vibração de um sistema mecânico. A metodologia é trabalhada através do estudo de problemas de autovalores e, para determinar numericamente os autovalores de uma matriz quadrada A, Krylov usa sequências da forma $\{b, Ab, A^2b, \ldots\}$ na determinação do polinômio característico de A. Surge, então, o conceito de subespaços de Krylov. Para uma leitura sobre subespaços e métodos de Krylov, além de outros aspectos mais avançados que os apresentados neste texto, sugerimos [17, 18].

[1] `https://mathshistory.st-andrews.ac.uk/Biographies/Krylov_Aleksei/`

4.1 Subespaços de Krylov

Definição 4.1. *Sejam $A \in \mathbb{R}^{n\times n}$ e $b \in \mathbb{R}^n$ não nulo. Definimos os seguintes conceitos:*

1. $\{b,\, Ab,\, A^2b,\, \ldots\}$ *é chamado de sequência de Krylov.*
2. $\mathcal{K}(A,b) = span\{b,\, Ab,\, A^2b,\, \ldots\}$ *é chamado de espaço de Krylov associado a A e b.*
3. $\mathcal{K}_k(A,b) = span\{b,\, Ab,\, A^2b,\, \ldots,\, A^{k-1}b\}$ *é chamado de subespaço de Krylov de ordem k associado a A e b.*
4. $K_k(A,b) = [b \,|\, Ab \,|\, \cdots \,|\, A^{k-1}b] \in \mathbb{R}^{n\times k}$ *é chamada de matriz de Krylov.*

Sempre que estiver claro o contexto, chamaremos o espaço de Krylov $\mathcal{K}(A,b)$ de $\mathcal{K}$ e, da mesma forma, chamaremos $\mathcal{K}_k(A,b)$ de $\mathcal{K}_k$. Observe que $\mathcal{K}_k \subseteq \mathbb{R}^n$ e, por convenção, $\mathcal{K}_0 = \{0\}$. Podemos definir o subespaço de Krylov $\mathcal{K}_k$ em termos da matriz de Krylov por $\mathcal{K}_k = \mathrm{Im}(K_k)$.

Os subespaços de Krylov satisfazem algumas propriedades.

Teorema 4.1. *[138, pág. 267] Sejam $A \in \mathbb{R}^{n\times n}$ e $b \in \mathbb{R}^n$ não nulo. Então,*

1. $\mathcal{K}_k(A,b) \subseteq \mathcal{K}_{k+1}(A,b)$;
2. $A\mathcal{K}_k(A,b) \subseteq \mathcal{K}_{k+1}(A,b)$;
3. $\mathcal{K}_k(A,b) = \mathcal{K}_k(\sigma A, \xi b)$, *para $\sigma, \xi \in \mathbb{R}$ não nulos;*
4. $\mathcal{K}_k(A,b) = \mathcal{K}_k(A - \kappa I, b)$, *para $\kappa \in \mathbb{R}$;*
5. $\mathcal{K}_k(P^{-1}AP, P^{-1}b) = P^{-1}\mathcal{K}_k(A,b)$, *para $P \in \mathbb{R}^{n\times n}$ não singular.*

A demonstração desses fatos segue imediatamente da definição de $\mathcal{K}_k(A,b)$ e é deixada a cargo do leitor.

Teorema 4.2. *[102, pág. 537] Suponha que $\mathcal{K}_{k-1} \neq \mathcal{K}_k$. Se $w \in \mathcal{K}_k \setminus \mathcal{K}_{k-1}$, então $Aw \in \mathcal{K}_{k+1}$ e $\mathcal{K}_{k+1} = span\,(\{Aw\} \cup \mathscr{B})$, com $\mathscr{B}$ uma base de $\mathcal{K}_k$. Ademais, se $Aw \in \mathcal{K}_k$, então $\mathcal{K}_{k+1} = \mathcal{K}_k$, ou seja, os subespaços de Krylov se estabilizam na ordem k.*

Demonstração. Como $w \in \mathcal{K}_k$, então

$$w = c_1 b + c_2 Ab + c_3 A^2 b + \cdots + c_k A^{k-1} b,$$

com $c_i \in \mathbb{R}$ e c_k não nulo. Portanto,

$$Aw = c_1 Ab + c_2 A^2 b + c_3 A^3 b + \cdots + c_k A^k b \in \mathcal{K}_{k+1}.$$

Se $Aw \in \mathcal{K}_k$, então

$$\sum_{i=1}^{k} \hat{c}_i A^{i-1} b = \sum_{j=1}^{k} c_j A^j b \implies c_k A^k b = \sum_{i=1}^{k} (\hat{c}_i - c_{i-1}) A^{i-1} b,$$

com $c_0 = 0$. Como $\hat{c}_k$ e c_k são não nulos, então $A^k b$ é não nulo e $\mathcal{K}_k = \mathcal{K}_{k+1}$. Se $Aw \notin \mathcal{K}_k$, então

$$A^k b = \frac{1}{c_k} Aw - \sum_{i=1}^{k-1} \frac{c_i}{c_k} A^i b.$$

Como $A^k b \in \mathcal{K}_{k+1}$, mas é escrito como a combinação linear de Aw e elementos da base de $\mathcal{K}_k$, então $\mathcal{K}_{k+1}$ é gerado pela união de $\{Aw\}$ e uma base de $\mathcal{K}_k$. □

Observe que a base $\mathscr{B} = \{b, Ab, \ldots, A^{k-1}b\}$ de $\mathcal{K}_k$ não é uma base apropriada para os subespaços de Krylov do ponto de vista numérico. Com efeito, pelo método das potências, conforme k aumenta o vetor $A^k b$ se aproxima da direção do autovetor dominante, consequentemente esses vetores se tornam linearmente dependentes na precisão da aritmética de ponto flutuante. Para evitar essa situação poderíamos ortogonalizar a base $\mathscr{B}$, porém esse problema é mal condicionado.

Vamos descrever como obter uma base para os espaços de Krylov que seja mais apropriada do ponto de vista numérico. Começaremos com a ideia geral, chamada de método de Arnoldi [3] e, depois, passaremos ao método de Lanczos[2] [80], que supõe que a matriz A seja definida positiva, como nos problemas de quadrados mínimos. Para a construção da base em questão vamos seguir a metodologia empregada em [138, 151].

Uma base para $\mathcal{K}_k(A, b)$ é $\mathscr{B} = \{b, Ab, \ldots, A^{k-1}b\}$ e, para uma notação mais agradável, chamemos $u_j = A^{j-1}b$ e denotemos a matriz de Krylov por $U_k \in \mathbb{R}^{n\times k}$, ou seja, as colunas de U_k são os vetores u_j, $j = 1, \ldots, k$. Vejamos a conexão entre a matriz A e a matriz de Krylov U_k.

Teorema 4.3. *Sejam $A \in \mathbb{R}^{n\times n}$, $b \in \mathbb{R}^n$, $U_k \in \mathbb{R}^{n\times k}$ a k-ésima matriz de Krylov e $W_k \in \mathbb{R}^{k\times k}$ uma matriz de zeros, exceto $w_{j+1\,j} = 1$, $j = 1, \ldots, k-1$. Então,*

$$AU_k = U_k W_k + u_{k+1} e_k^T, \tag{4.1.1}$$

onde e_k é o k-ésimo vetor da base canônica de $\mathbb{R}^k$.

Demonstração. A demonstração é feita por indução. Para $k = 1$, temos que $U_1 = u_1 = b$ e $W_1 = 0$, o que implica que $AU_1 = Au_1$, $U_1 W_1 = 0$ e $u_2 = A^2 b = Au_1$. Como $e_1 = 1$, então $u_2 e_1^T = u_2$ e, portanto, $AU_1 = U_1 W_1 + u_2 e_1^T$.

Agora, suponha que $AU_k = U_k W_k + u_{k+1} e_k^T$. Como $u_{k+1} = A^k b$, então

[2] `https://mathshistory.st-andrews.ac.uk/Biographies/Lanczos/`

$Au_{k+1} = u_{k+2}$. Logo,

$$\begin{aligned} AU_{k+1} &= [AU_k \,|\, Au_{k+1}] = [AU_k \,|\, u_{k+2}] = \left[U_k W_k + u_{k+1} e_k^T \,|\, u_{k+2}\right] \\ &= \left[U_k W_k + u_{k+1} e_k^T \,|\, 0\right] + u_{k+2} e_{k+1}^T \\ &= [u_1 \,|\, u_2 \,|\, \cdots \,|\, u_{k+1} \,|\, 0] + u_{k+2} e_{k+1}^T \\ &= U_{k+1} W_{k+1} + u_{k+2} e_{k+1}^T. \end{aligned}$$

□

Recursivamente, o resultado acima pode ser reescrito como

$$AU_k = \sum_{j=1}^{k} u_{j+1} e_j^T,$$

onde e_j é o j-ésimo vetor da base canônica de $\mathbb{R}^j$.

Seja $U_k = Q_k R_k$ a fatoração QR de U_k. Substituindo em (4.1.1), obtemos $AQ_k R_k = Q_k R_k W_k + u_{k+1} e_k^T$ e, portanto,

$$\begin{aligned} AQ_k &= Q_k R_k W_k R_k^{-1} + u_{k+1} e_k^T R_k^{-1} \\ &= Q_k \tilde{H}_k + u_{k+1} e_k^T R_k^{-1} \qquad (4.1.2) \\ &= Q_k \tilde{H}_k + \frac{1}{r_{kk}} u_{k+1} e_k^T. \end{aligned}$$

Teorema 4.4. *Sejam W_k e R_k como acabamos de definir. Então $\tilde{H}_k = R_k W_k R_k^{-1}$ é uma matriz de Hessenberg superior.*

Demonstração. Procedamos por indução. Primeiramente, note que $\tilde{H}_1 = R_1 W_1 R_1^{-1}$ é trivialmente Hessenberg superior. Assuma que $\tilde{H}_k = R_k W_k R_k^{-1}$, portanto

$$\begin{aligned} \tilde{H}_{k+1} &= R_{k+1} W_{k+1} R_{k+1}^{-1} \\ &= \begin{bmatrix} R_k & f \\ 0 & a \end{bmatrix} \begin{bmatrix} W_k & 0 \\ e_k^T & 0 \end{bmatrix} \begin{bmatrix} R_k^{-1} & g \\ 0 & 1/a \end{bmatrix} \\ &= \begin{bmatrix} R_k W_k R_k^{-1} + f e_k^T R_k^{-1} & R_k W_k g + f e_k^T g \\ a e_k^T R_k^{-1} & a e_k^T g \end{bmatrix} \\ &= \begin{bmatrix} \tilde{H}_k + f e_k^T R_k^{-1} & R_k W_k g + f e_k^T g \\ a e_k^T R_k^{-1} & a e_k^T g \end{bmatrix}, \end{aligned}$$

que, claramente, é uma matriz de Hessenberg superior. □

Considere a fatoração QR de $U_{k+1} = Q_{k+1}R_{k+1}$. Note que,

$$R_{k+1} = \begin{bmatrix} R_k & \tilde{r} \\ 0 & r_{k+1\,k+1} \end{bmatrix}.$$

Portanto,

$$u_{k+1} = Q_k\tilde{r} + r_{k+1\,k+1}q_{k+1},$$

onde q_{k+1} é a última coluna de Q_{k+1}. Substituindo em (4.1.2), obtemos

$$\begin{aligned} AQ_k &= Q_k\left(\tilde{H}_k + \frac{1}{r_{kk}}\,\tilde{r}e_k^T\right) + \frac{r_{k+1\,k+1}}{r_{kk}}\,q_{k+1}e_k^T \\ &= Q_kH_k + \alpha\,q_{k+1}e_k^T. \end{aligned} \tag{4.1.3}$$

Daí,

$$Q_k^TAQ_k = H_k + \alpha\,Q_k^Tq_{k+1}e_k^T.$$

Como as colunas de Q_{k+1} são ortogonais, então $\alpha\,Q_k^Tq_{k+1}e_k^T = 0$ e, portanto,

$$Q_k^TAQ_k = H_k.$$

Ademais, por (4.1.3) e utilizando a ortogonalidade das colunas de Q_{k+1}, obtemos

$$q_{k+1}^TAQ_k = \cancelto{0}{q_{k+1}^TQ_kH_k} + \alpha\,q_{k+1}^Tq_{k+1}e_k^T = \alpha\,e_k^T.$$

Por outro lado,

$$q_{k+1}^TAQ_kQ_k^Tq_k = \alpha\,e_k^TQ_k^Tq_k \implies \alpha = q_{k+1}^TAq_k = h_{k+1\,k},$$

pois $Q_{k+1}^TAQ_{k+1} = H_{k+1}$. Para melhor compreender os resultados que demonstramos considere o seguinte teorema.

Teorema 4.5. *[58, pág. 381] Seja $A \in \mathbb{R}^{n\times n}$. Suponha que as matrizes $Q, V \in \mathbb{R}^{n\times n}$ sejam ortogonais com a propriedade que $Q^TAQ = H$ e $V^TAV = G$, com H e G matrizes de Hessenberg superiores. Seja k o menor inteiro positivo, tal que $h_{k+1\,k} = 0$, com a convenção que $k = n$ para H não reduzida (elementos da subdiagonal não nulos). Se $q_1 = v_1$, então $q_i = \pm v_i$ e $|h_{i\,i-1}| = |g_{i\,i-1}|$ para $i = 2, \ldots, n$. Ademais, se $k < n$, então $g_{k+1\,k} = 0$.*

Demonstração. Defina $W = V^TQ$, que é ortogonal, e note que $GW = WH$. Comparando as colunas dessa igualdade, obtemos

$$h_{i\,i-1}w_i = Gw_{i-1} - \sum_{j=1}^{i-1} h_{j\,i-1}w_j, \quad i = 2, \ldots, k.$$

Como, por hipótese, $w_1 = e_1$, segue que $[w_1 \mid \cdots \mid w_k]$ é triangular superior e, para $i = 2, \ldots, k$, temos $w_i = \pm e_i$. Por outro lado, $w_i = V^T q_i$ e $h_{i\,i-1} = w_i^T G w_{i-1}$, de onde temos que

$$v_i = \pm q_i \quad \text{e} \quad |h_{i\,i-1}| = |g_{i\,i-1}|, \quad i = 2, \ldots, k.$$

Se $k < n$, então

$$\begin{aligned} g_{k+1\,k} &= e_{k+1}^T G e_k = \pm e_{k+1}^T G W e_k = \pm e_{k+1}^T W H e_k \\ &= \pm e_{k+1}^T \sum_{i=1}^{k} h_{ik} W e_i = \pm \sum_{i=1}^{k} h_{ik} e_{k+1}^T e_i = 0. \end{aligned}$$

□

Essa proposição afirma que a redução de A a uma matriz de Hessenberg H é unicamente determinada por q_1 a menos de sinais. Porém, $q_1 = b/\|b\|_2$ que é um vetor dado pelas condições do problema. Vale a pena ressaltar que a discussão até agora foi feita inteiramente em aritmética exata.

O método de Arnoldi ou iteração de Arnoldi é baseado nessa construção de Q_k e H_k. O processo é iniciado com a determinação de $q_1 = b/\|b\|_2$ e, para computar q_2, devemos calcular Aq_1 e ortonormalizar o conjunto $\{q_1, Aq_1\}$. O vetor q_3 é obtido da ortonormalização de $\{q_1, q_2, Aq_2\}$ e, em geral, se $\{q_1, q_2, \ldots, q_k\}$ é uma base de $\mathcal{K}_k(A, b)$, então tome $t = Aq_k$ e ortonormalize o conjunto $\{q_1, q_2, \ldots, q_k, Aq_k\}$ para obter q_{k+1}. O algoritmo a seguir resume o método de Arnoldi com Gram-Schmidt modificado (GSM).

Algoritmo 9 Método de Arnoldi com GSM

```
function [H,Q]=ARNOLDI(A,b)
    [m,n] = size(A);
    q_1 = b/ ||b||_2;
    for j = 1 : n-1 do
        t = A q_j;
        for i = 1 : j do
            h_ij = <q_i, t>;
            t = t - h_ij q_i;
        end for
        h_{j+1 j} = ||t||_2;
        if h_{j+1 j} == 0 then stop
        else
            q_{j+1} = t/h_{j+1 j};
        end if
    end for
end function
```

Como resultado, obtemos uma base ortogonal para o subespaço $\mathcal{K}_k(A, b)$. Observe que t pode ser nulo e, nesse caso, o processo é abortado. Isto acontece quando t é uma combinação linear dos vetores que são utilizados na ortogonalização de t. Seja $Q_k \in \mathbb{R}^{n \times k}$ a matriz cujas colunas são os

vetores $q_1, \ldots, q_k$. Como Q_k é obtida pela mesma metodologia que gerou (4.1.3), então

$$AQ_{k-1} = Q_k H_{k,k-1}, \tag{4.1.4}$$

onde $H_{k,k-1} \in \mathbb{R}^{k \times k-1}$ é uma matriz de Hessenberg superior. Essa decomposição é conhecida como decomposição de Arnoldi de ordem k, e ela pode ser obtida de diversos modos. Empregamos o processo de Gram-Schmidt modificado para minimizar a perda de ortogonalidade dos vetores por causa de erros de arredondamento. Uma discussão sobre a raiz do problema e como recuperar a ortogonalidade em precisão finita pode ser encontrada em [88]. Um possível refinamento desse algoritmo baseado em um processo de reortogonalização foi proposto por [30, pág. 774].

Algoritmo 10 Método de Arnoldi com GSM e reortogonalização

```
function [H, Q]=ARNOLDIREORT(A, b, κ)
    [m, n] = size(A);
    q_1 = b/ ||b||_2;
    for j = 1 : n − 1 do
        t = Aq_j;
        τ = ||t||_2;
        for i = 1 : j do
            h_ij = ⟨q_i, t⟩;
            t = t − h_ij q_i;
        end for
        if ||t||_2/τ ⩽ κ then
            for i = 1 : j do
                ρ = ⟨q_i, t⟩;
                t = t − ρq_i;
                h_ij = h_ij + ρ;
            end for
        end if
        h_{j+1 j} = ||t||_2;
        if h_{j+1 j} == 0 then stop
        else
            q_{j+1} = t/h_{j+1 j};
        end if
    end for
end function
```

A ideia é garantir que o conjunto de vetores seja ortogonal na precisão que estamos trabalhando. Depois da ortogonalização verifica-se se o novo vetor não normalizado tem norma significativamente menor do que o novo vetor no começo do passo de ortogonalização, digamos por um fator $\kappa < 1$. No artigo original, Daniel et al. propuseram $\kappa = 1/\sqrt{2}$. Nesse caso há problemas de cancelamento e o processo de Gram-Schmidt modificado é aplicado novamente. Ao final garantimos que o conjunto de vetores obtido tem uma perda mútua de ortogonalidade de um fator limitado a $1/\kappa$, de forma relativa [151, pág. 31].

Vejamos a seguir como que Walker [154] propõe a utilização de transformações de Householder para a construção da base ortogonal. O método

de Arnoldi resumidamente é dado por (4.1.4), isto é,

$$AQ_{k-1} = Q_k H_{k,k-1}.$$

Por construção, a primeira coluna de Q_k é $q_1 = b/\|b\|_2$ e, portanto

$$[b \,|\, AQ_{k-1}] = Q_k R_k,$$

onde

$$R_k = \begin{bmatrix} \|b\|_2 & h_{11} & h_{12} & \cdots & h_{1\,k-1} \\ 0 & h_{21} & h_{22} & \cdots & h_{2\,k-1} \\ 0 & 0 & h_{31} & \cdots & h_{3\,k-1} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & h_{k\,k-1} \end{bmatrix}.$$

Assim, o espaço coluna de $[b \,|\, AQ_{k-1}]$ é $\mathcal{K}_k(A, b)$ e, além disso, $Q_k R_k$ é a fatoração QR de $[b \,|\, AQ_{k-1}]$. O método de Arnoldi utilizando transformações de Householder é baseado nesse fato. A ideia a ser utilizada determina a fatoração QR de $[b \,|\, AQ_{k-1}]$ utilizando as transformações de Householder P_j de modo que $P_k P_{k-1} \cdots P_2 P_1 [b \,|\, AQ_{k-1}] = R_k$. Imediatamente, segue que

$$Q_k = P_1 P_2 \cdots P_{k-1} P_k.$$

Por construção P_j mantém as primeiras $j-1$ linhas inalteradas, assim $q_1 = P_1 e_1$, $q_2 = P_1 P_2 e_2$, $\ldots$, $q_k = P_k \cdots P_2 P_1 e_k$. Os vetores e_i são a i-ésima coluna da matriz $I \in \mathbb{R}^{m \times n}$ e, um algoritmo para abordar essa construção tem a seguinte forma.

Algoritmo 11 Método de Arnoldi com transformações de Householder

```
function [H, Q, R]=ARNOLDIH(A, b)
    [m, n] = size(A);
    z_1 = b;  I = eye(m, n);
    Q_a = I;  Q_r = I;
    for j = 1 : n do
        [~, w_a, ~] =HOUSEHOLDER(z_{j:m,j});
        w_a = [zeros(j − 1, 1); w_a];
        β = 2/‖w_a‖_2^2;
        P = I − β(w_a w_a^T);
        r_j = P z_j;
        Q_a = Q_a P;
        Q_r = P Q_r;
        q_j = Q_a I_j; % I_j = e_j
        z_{j+1} = Q_r A q_j;
    end for
    H = Q^T A Q_{:,1:n−1};
end function
```

Um algoritmo para a fatoração QR via transformações de Householder é dado a seguir.

Algoritmo 12 Fatoração QR - Transformações de Householder

```
function [Q, R]=HOUSEHOLDER(A)
    [m, n] =size(A);
    Q = I;  R = A;
    for j = 1 : min(m, n) do
        r = ||R_{j:end,j}||_2;
        s = −sign(R_jj);
        u = R_jj − sr;
        v = R_{j:end,j}/u;
        v(1) = 1;
        β = −su/r;
        R_{j:end,:} = R_{j:end,:} − βv(v^T R_{j:end,:});
        Q_{:,j:end} = Q_{:,j:end} − β(Q_{:,j:end} v)v^T;
    end for
end function
```

Teorema 4.6. *[154, pág. 156] O conjunto $\{q_1, \ldots, q_k\}$ criado pelo Algoritmo 11 é uma base ortonormal de $\mathcal{K}_k(A, b)$.*

Demonstração. Vamos demonstrar por indução sobre os índices k. Para $k = 1$ o conjunto $\{q_1\}$ é trivialmente ortogonal. Suponha que $\{q_1, \ldots, q_k\}$ é conjunto ortonormal para $1 < j \leqslant k$. Pelo fato de $Pq_1 = e_1$ e

$$P_k P_{k-1} \cdots P_2 P_1 [b \,|\, AQ_{k-1}]$$

ser triangular superior, segue que o vetor de Householder para $1 < j \leqslant k$ tem as primeiras $j - 1$ entradas nulas. Então $P_k \cdots P_2 P_1 e_k = q_k$ para $1 \leqslant j \leqslant k$, isto é, $P_1 \cdots P_j = [q_1 \,|\, \cdots \,|\, q_j]$. Em particular vale para $j = k$, ou seja, $\{q_1, \ldots, q_k\}$ é um conjunto ortonormal de vetores.

Mostremos, agora, que $\mathcal{K}_k = span\{q_1, \ldots, q_k\}$. O resultado segue trivialmente para $k = 1$. Assuma que,

$$\mathcal{K}_j = span\{q_1,\, Aq_1,\, \ldots,\, A^{j-1}q_1\} = span\{q_1, \ldots, q_j\},\;\; j = 1, \ldots, k.$$

Como $1 \leqslant j < k$, $[q_1 \,|\, Aq_1 \,|\, \cdots \,|\, Aq_j]$ tem posto $j + 1$ e, portanto,

$$P_{j+1} \cdots P_2 P_1 [q_1 \,|\, Aq_1 \,|\, \cdots \,|\, Aq_j]$$

é triangular superior de posto $j + 1$, ou seja, $span\{q_1,\, Aq_1,\, \ldots,\, Aq_j\}$ é o subespaço gerado pelas primeiras $j + 1$ colunas de $P_1 \cdots P_{j+1}$. Portanto, $\mathcal{K}_{j+1} = span\{q_1, \ldots, q_{j+1}\}$. □

Outra forma de demonstrar a ortogonalidade dos vetores q_j é considerando $1 \leqslant i < j \leqslant \text{rank}(A)$ e verificando que

$$\begin{aligned} \langle v_i, v_j \rangle &= (P_1 P_2 \cdots P_i e_i)^T (P_1 P_2 \cdots P_j e_j) \;=\; e_i^T P_{i+1} \cdots P_j e_j \\ &= (P_j \cdots P_{i+1} e_i)^T e_j \;=\; e_i^T e_j \;=\; 0. \end{aligned}$$

Se A é simétrica, então $H_{k,k} = Q_k^T A Q_k$ é uma matriz de Hessenberg superior simétrica, portanto tridiagonal. A metodologia aplicada na determinação da base ortonormal de $\mathcal{K}_k(A, b)$ é chamada de método de Lanczos, ou processo de Lanczos. Obviamente que a aplicação do método de Arnoldi a uma matriz simétrica fornece a base ortogonal procurada mas, em vez de aplicarmos diretamente o método de Arnoldi, podemos modificá-lo de forma a necessitar menor processamento de máquina e armazenamento de dados. Portanto, o método de Lanczos é obtido após essas pequenas modificações na iteração de Arnoldi.

Efetivamente, para obter o método de Lanczos defina

$$\alpha_k = h_{ij} \quad \text{e} \quad \beta_k = h_{k-1\,k}.$$

A matriz de Hessenberg (tridiagonal) resultante tem a seguinte forma

$$H_{k+1\,k} = \begin{bmatrix} \alpha_1 & \beta_2 & & & \\ \beta_2 & \alpha_2 & \beta_3 & & \\ & \beta_3 & \alpha_3 & \ddots & \\ & & \ddots & \ddots & \beta_k \\ & & & \beta_k & \alpha_k \\ & & & & \beta_{k+1} \end{bmatrix} = \begin{bmatrix} H_k \\ \beta_{k+1} e_k^T \end{bmatrix}.$$

A matriz de Hessenberg construída pelo método de Lanczos possui uma linha extra e, por isso, definamos a matriz de Hessenberg procurada como a submatriz principal de ordem k, isto é, $H_k \in \mathbb{R}^{k\times k}$. A partir de (4.1.3) obtemos para a primeira coluna

$$Aq_1 = \alpha_1 q_1 + \beta_1 q_2.$$

Da ortogonalidade de q_1 e q_2, temos

$$\alpha_1 = q_1^T A q_1 \quad \text{e} \quad \beta_1 = \|Aq_1 - \alpha_1 q_1\|_2$$

e, para a k-ésima coluna, obtemos

$$Aq_k = \beta_{k+1} q_{k+1} + \alpha_k q_k + \beta_k q_{k-1},$$

onde

$$\alpha_k = q_k^T A q_k \quad \text{e} \quad \beta_{k+1} = \|Aq_k - \alpha_k q_k - \beta_k q_{k-1}\|_2.$$

Denominando $t_k = Aq_k$, calculamos a $(k+1)$-ésima coluna da seguinte forma

$$q_{k+1} = \frac{t_k - \alpha_k q_k - \beta_k q_{k-1}}{\beta_{k+1}}.$$

Dessa forma, um possível algoritmo para o método de Lanczos é

Algoritmo 13 Método de Lanczos

```
function Q=LANCZOS(A, b)
    [m, n] = size(A);
    β1 = ||b||2;  q0 = 0;  q1 = b/β1;
    for j = 1 : m − 1 do
        t = Aqj;
        α = ⟨qj, t⟩;
        t = t − αqj − βj qj−1;
        βj+1 = ||t||2;
        if βj+1 == 0 then stop
        else
            qj+1 = t/βj+1;
        end if
    end for
end function
```

Os vetores q_j que formam a matriz Q são chamados de vetores de Lanczos. Podemos também aplicar as transformações de Householder para o método de Lanczos, como feito em [57].

4.2 O Método dos Gradientes Conjugados e Variações

Originalmente, o método dos gradientes conjugados foi deduzido por Hestenes e Stiefel [64] no começo da década de 50. Uma de suas vantagens é que, em aritmética exata, ele converge em um número finito de passos. Mas, trabalhamos com aritmética finita e o método perde essa propriedade quando erros de arredondamento estão presentes.

Stiefel [140] publicou o primeiro artigo que cita como utilizar o método de gradientes conjugados para resolver as equações normais oriundas de problemas de quadrados mínimos. O primeiro a discutir e aplicar o método dos gradientes conjugados (CG) precondicionado foi Läuchli [83] em 1959. Outros dois pesquisadores que discutiram o CG foram Lawson [85] e Chen [24].

Ainda na década de 70, o método de gradientes conjugados começou a ser mais estudado como um método iterativo [118] e, nos dias de hoje, esse método é uma importante ferramenta no estudo de problemas de quadrados mínimos de grande porte. Para fatos históricos referentes ao desenvolvimento do método dos gradientes conjugados sugerimos [4, 56].

Os métodos iterativos que apresentam boa performance são os métodos SOR e semi-iterativo – em nosso caso, Chebyshev. Esses métodos, em geral, apresentam um problema: a difícil escolha dos parâmetros. No caso do método SOR é a escolha de ω e no caso do método semi-iterativo de Chebyshev, a escolha do maior e menor autovalores da matriz de iteração. Os métodos baseados em subespaços de Krylov não apresentam essa problemática. Como estamos interessados no estudo de quadrados mínimos, vamos apresentar o método de gradientes conjugados (CG) e suas variantes.

Há outros métodos baseados em espaços de Krylov, tais como GMRES [127], Bi-CG [38, 44, 81], CGS [134] e Bi-CGSTAB [150], que não são cobertos neste texto. Nesta seção vamos iniciar com a dedução do CG e, posteriormente, focaremos na teoria de quadrados mínimos.

A primeira diferença entre os métodos iterativos estacionários e os métodos de Krylov é a matriz de iteração. Os métodos de Krylov não possuem matrizes de iteração, o que, do ponto de vista computacional, é ótimo pois utiliza-se menos memória para o cálculo dos passos iterativos. A hipótese básica do CG é que o método é válido para matrizes (simétricas) definidas positivas que surgem naturalmente do estudo de soluções numéricas de equações diferenciais parciais elípticas [16]. O CG pode ser visto como uma variação do método de Lanczos (Algoritmo 13) estudado na seção 4.1. A dedução do método dos gradientes conjugados aqui apresentada é baseada no método de máxima descida, nos subespaços de Krylov e no método de Lanczos. Nos basearemos nas discussões de Golub e Van Loan [58], Shewchuck [132], e Stoer e Bulirsch [144].

4.2.1 Sistema Linear $\times$ Forma Quadrática

Antes de deduzirmos o método dos gradientes conjugados demonstramos que encontrar, no caso de uma matriz definida positiva, a solução de um sistema linear $Ax = b$ é equivalente a minimizar uma forma quadrática.

Definição 4.2. *Sejam $A \in \mathbb{R}^{n\times n}$ uma matriz definida positiva e $b \in \mathbb{R}^n$. Uma forma quadrática é uma função $f : \mathbb{R}^n \to \mathbb{R}$, definida por*

$$f(x) = \frac{1}{2}x^T Ax - x^T b + c. \tag{4.2.5}$$

Por exemplo, sejam

$$A = \begin{bmatrix} 3 & 1 \\ 1 & 6 \end{bmatrix} \quad \text{e} \quad b = \begin{bmatrix} 1 \\ 2 \end{bmatrix}.$$

O gráfico da forma quadrática associada a A, b e $c = 0$ está apresentada na Figura 4.1 e, na Figura 4.2, apresentamos suas curvas de nível. O minimizador de f é $x = [1.8, -2.1]^T$ que, por sua vez, também é a solução de $Ax = b$, como vemos a seguir.

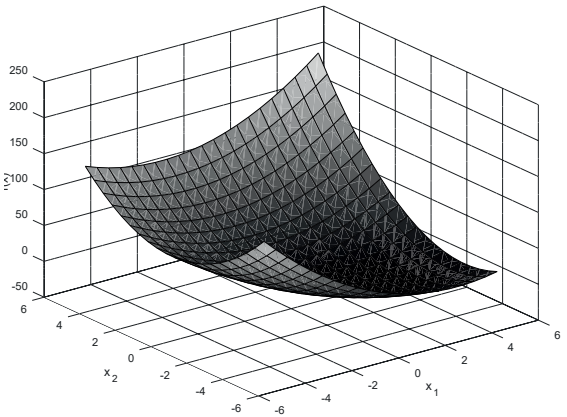

Figura 4.1: Forma quadrática.

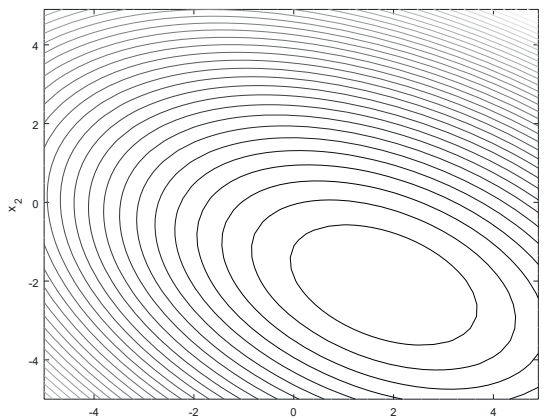

Figura 4.2: Curvas de nível da forma quadrática, ponto crítico de f é $x = [1.8, -2.1]^T$.

Afirmamos que minimizar (4.2.5) é equivalente a resolver o sistema linear $Ax = b$. Com efeito,

$$\nabla f(x) = \frac{1}{2}A^T x + \frac{1}{2}Ax - b.$$

Como, por definição, A é simétrica, então

$$\nabla f(x) = Ax - b. \tag{4.2.6}$$

Portanto, o ponto crítico x_* de $f(x)$ é o vetor que soluciona o sistema linear $Ax = b$. Assim, transformamos o problema de encontrar a solução de $Ax = b$ em

$$\min_{x \in \mathbb{R}^n} \left(\frac{1}{2} x^T A x - x^T b + c \right).$$

Ademais, como A satisfaz a condição de positividade, então o ponto crítico x_* é um minimizador global de f. De fato, seja e um termo de erro, então

$$\begin{aligned} f(x_* + e) &= \frac{1}{2}(x_* + e)^T A (x_* + e) - (x_* + e)^T b + c \\ &= \frac{1}{2} x_*^T A x_* + e^T A x_* + \frac{1}{2} e^T A e - b^T x_* - b^T e + c \\ &= \left(\frac{1}{2} x_*^T A x_* - b^T x_* + c \right) + e^T b + \frac{1}{2} e^T A e - b^T e \\ &= f(x_*) + \frac{1}{2} e^T A e. \end{aligned} \tag{4.2.7}$$

Como A é definida positiva, então $e^T A e > 0$ e, portanto, x_* é um minimizador global de f. Apesar de estarmos interessados em resolver sistemas lineares, trocamos a abordagem para um problema de minimização, pois o método de máxima descida e o de gradientes conjugados são relativamente baratos computacionalmente e de forma indireta fornecem a procurada solução do sistema linear em questão.

4.2.2 O Método de Máxima Descida

O método de máxima descida ou método do gradiente é um método de minimização em que, a cada iteração, escolhe-se a direção em que f decresce mais rapidamente. A discussão aqui apresentada se baseia em Shewchuck [132].

Então, esse método se inicia com um "chute" inicial x_0 e, a cada iteração, busca-se um refinamento da solução na direção $-\nabla f(x_i) = b - Ax_i$, $i = 1, 2, \ldots$ ((4.2.6)). Assim, criamos uma sequência de aproximações que a cada passo "descem" no paraboloide (forma quadrática) até a aproximação x_n estar perto o suficiente do ponto x_*.

O vetor erro ou, simplesmente, erro $e_i = x_i - x_*$, é a medida de quão longe a solução aproximada x_i está da solução exata e, o resíduo $r_i = b - Ax_i$ nos diz quão precisa é a aproximação Ax_i de b. Ademais, $r_i = -Ae_i$ pois

$$Ae_i = A(x_i - x_*) = Ax_i - Ax_* = Ax_i - b = -r_i.$$

Suponha que $x_0 = [0, 0]^T$ que, por sua vez, cai em alguma curva de nível da forma quadrática da Figura 4.2. O próximo passo é calculado de maneira iterativa, isto é

$$x_1 = x_0 + \alpha r_0. \tag{4.2.8}$$

Note que $r_0 = -\nabla f(x_0)$, ou seja, é a direção de máxima descida de f, e o escalar α é o fator de alongamento ou encolhimento do vetor gradiente. O próximo passo é determinar o parâmetro α e, para isso, utilizaremos o procedimento de pesquisa linear ou *line search* que determina o parâmetro α de maneira que a forma quadrática f seja minimizada ao longo da direção de máxima descida.

A determinação de α para passar da iteração 0 para a iteração 1 é dada calculando-se a derivada direcional e, posteriormente, igualando a zero, isto é, quando

$$\frac{d}{d\alpha} f(x_1) = 0.$$

Pela regra da cadeia, obtemos

$$\frac{d}{d\alpha} f(x_1) = \nabla^T f(x_1) \frac{d}{d\alpha} x_1 \overset{(4.2.8)}{=} \nabla^T f(x_1) r_0,$$

ou seja, quando a direção $\nabla f(x_1)$ é ortogonal ao resíduo r_0. Como $r_i = -\nabla f(x_i)$, $i = 1, 2, \ldots$, a minimização ocorre quando $r_1^T r_0 = 0$. Mas,

$$r_1^T r_0 = 0 \Rightarrow (b - Ax_1)^T r_0 = 0 \overset{(4.2.8)}{\Rightarrow} [b - A(x_0 + \alpha r_0)]^T r_0 = 0,$$

que nos leva a

$$\alpha = \frac{r_0^T r_0}{r_0^T A r_0}.$$

Portanto, o método de máxima descida é descrito por

$$r_i = b - Ax_i, \quad \alpha_i = \frac{r_i^T r_i}{r_i^T A r_i}, \quad x_{i+1} = x_i + \alpha_i r_i.$$

Esse método requer duas multiplicações matriz-vetor por iteração, que acabam por dominar seu custo computacional. Felizmente, um desses produtos pode ser eliminado multiplicando $x_{i+1} = x_i + \alpha_i r_i$ por $-A$ em ambos os lados e adicionando b, obtendo

$$r_{i+1} = r_i - \alpha_i A r_i. \tag{4.2.9}$$

Observe que o produto Ar_i é necessário para atualizar α_i. Além disso, um ponto negativo dessa abordagem é que o resíduo pode ficar contaminado por erros de arredondamento cujo resultado é a convergência para um valor diferente de x_*. Esse problema pode ser evitado através de um uso periódico de $r_i = b - Ax_i$ para atualizar o resíduo. Um possível algoritmo para a determinação da solução de um sistema linear via método de máxima descida é descrito a seguir.

Algoritmo 14 Método de Máxima Descida

```
function [x, i]=MMD(A, b, x, i_max, ε)
    i = 0;
    r = b − Ax;
    δ = r^T r;
    δ_0 = δ;
    while i < imax e δ > ε^2 δ_0 do
        q = Ar;
        α = δ / (r^T q);
        x = x + αr;
        if i é divisível por 50 then
            r = b − Ax;
        else
            r = r − αq;   %equação (4.2.9)
        end if
        δ = r^T r;
        i = i + 1;
    end while
end function
```

A entrada do algoritmo são os parâmetros do sistema linear $Ax = b$, o chute inicial x_0, o número máximo de iterações i_{max} e a tolerância desejada ϵ. O algoritmo termina quando o número máximo de iterações é excedido ou quando $\|r_i\| \leqslant \epsilon\|r_0\|$. A fórmula recursiva rápida[3] para a determinação do resíduo é usada, mas a cada 50 iterações, o resíduo exato é recalculado para remover eventuais erros causados pela aritmética de ponto flutuante. O número 50 é arbitrário e, para n grande, $\sqrt{n}$ pode ser mais apropriado. Isso previne que o procedimento termine cedo e errado por causa de erros de arredondamento.

Vamos começar o estudo da convergência do método de máxima descida por um caso particular. Suponha que o erro e_i é um autovetor de A associado ao autovalor λ_{e_i}. Note que, $r_i = -Ae_i = -\lambda_{e_i} e_i$. Subtraindo x^* de ambos os lados de $x_{i+1} = x_i + \alpha_i r_i$, obtemos

$$e_{i+1} = e_i + \frac{r_i^T r_i}{r_i^T A r_i} r_i = e_i - \lambda_{e_i} \frac{r_i^T r_i}{\lambda_{e_i}\left(r_i^T r_i\right)} e_i = 0$$

e a convergência ocorre em um passo, afinal o ponto x_i está em um dos eixos principais do elipsoide (Figura 4.2). Assim, o resíduo aponta diretamente para o centro do elipsoide, que é a solução do problema. Resumidamente, se $\alpha_i = \lambda_{e_i}^{-1}$, a convergência é instantânea.

Para uma análise mais geral, expressamos e_i como uma combinação linear dos autovetores de A. Vamos, primeiramente, supor que os autovetores formem uma base ortonormal de $\mathbb{R}^n$ — matrizes simétricas e normais satisfazem esse critério pelo Teorema Espectral. O vetor erro, então, tem a

[3] $r = r - \alpha q$ é rápida porque não requer o cálculo de um produto matriz-vetor.

seguinte forma

$$e_i = \sum_{i=1}^{n} \xi_i v_i.$$

Observe que não necessariamente todos os autovalores devem ser distintos. Logo,

$$r_i = -Ae_i = -\sum_{i=1}^{n} \xi_i \lambda_i v_i. \tag{4.2.10}$$

Por outro lado,

$$\|e_i\|^2 = e_i^T e_i = \sum_{i=1}^{n} \xi_i^2, \tag{4.2.11}$$

$$e_i^T A e_i = \left(\sum_{i=1}^{n} \xi_i v_i^T\right)\left(\sum_{j=1}^{n} \xi_j \lambda_j v_j\right) = \sum_{i=1}^{n} \xi_i^2 \lambda_i, \tag{4.2.12}$$

e

$$\|r_i\|^2 = r_i^T r_i = \sum_{i=1}^{n} \xi_i^2 \lambda_i^2, \tag{4.2.13}$$

$$r_i^T A r_i = \left(\sum_{i=1}^{n} \xi_i \lambda_i v_i^T\right)\left(\sum_{j=1}^{n} \xi_j \lambda_j^2 v_j\right) = \sum_{i=1}^{n} \xi_j^2 \lambda_j^3. \tag{4.2.14}$$

Sob essas hipóteses,

$$e_{i+1} = e_i + \frac{r_i^T r_i}{r_i^T A r_i} r_i = e_i + \frac{\sum_{i=1}^{n} \xi_i^2 \lambda_i^2}{\sum_{i=1}^{n} \xi_i^2 \lambda_i^3} r_i. \tag{4.2.15}$$

Se e_i é arbitrário, mas todos os autovetores de A estão associados a um único autovalor λ, (4.2.15) se torna

$$e_{i+1} = e_i - \frac{\lambda^2 \sum_{i=1}^{n} \xi_i^2}{\lambda^3 \sum_{i=1}^{n} \xi_i^2}(\lambda e_i) = 0.$$

Como todos autovalores são idênticos o elipsoide é esférico, o que faz qualquer ponto apontar para o centro da família de esferas, isto é, a convergência é instantânea. Porém, as hipóteses que fizemos até agora são bastante restritivas.

A demonstração do caso geral está baseada na existência de uma base de autovetores de A, isto é, A é diagonalizável, que não é um problema já que A é definida positiva. Claramente, todos os autovalores de A são não nulos pois estamos trabalhando com matrizes definidas positivas e, portanto, tem autovalores positivos.

Para facilitar a notação utilizaremos a A-norma de vetores, que é definida por $\|x\|_A^2 = x^T Ax$. A seguir apresentamos uma relação entre $\|e_{i+1}\|_A^2$ e $\|e_i\|_A^2$:,

$$\begin{aligned}
\|e_{i+1}\|_A^2 &= e_{i+1}^T A e_{i+1} = (e_i^T + \alpha_i r_i^T) A (e_i + \alpha_i r_i) \\
&= e_i^T A e_i + 2\alpha_i r_i^T A e_i + \alpha_i^2 r_i^T A r_i \text{ (simetria de } A) \\
&= \|e_i\|_A^2 - 2\frac{r_i^T r_i}{r_i^T A r_i}(r_i^T r_i) + \left(\frac{r_i^T r_i}{r_i^T A r_i}\right)^2 r_i^T A r_i = \|e_i\|_A^2 - \frac{\left(r_i^T r_i\right)^2}{r_i^T A r_i} \\
&= \|e_i\|_A^2 \left(1 - \frac{\left(r_i^T r_i\right)^2}{\left(r_i^T A r_i\right)\left(e_i^T A e_i\right)}\right) \\
&= \|e_i\|_A^2 \left(1 - \frac{\left(\sum_{i=1}^n \xi_i^2 \lambda_i^2\right)^2}{\left(\sum_{i=1}^n \xi_i^2 \lambda_i^3\right)\left(\sum_{i=1}^n \xi_i^2 \lambda_i\right)}\right) \\
&= \|e_i\|_A^2 \omega^2,
\end{aligned}$$

onde a penúltima igualdade segue de (4.2.12), (4.2.13), (4.2.14) e de

$$\omega^2 := 1 - \frac{\left(\sum_{i=1}^n \xi_i^2 \lambda_i^2\right)^2}{\left(\sum_{i=1}^n \xi_i^2 \lambda_i^3\right)\left(\sum_{i=1}^n \xi_i^2 \lambda_i\right)}.$$

A análise de convergência do método de máxima descida é finalizada ao determinarmos um limitante superior para ω^2. Por simplicidade de notação trabalhamos, inicialmente, com $n = 2$. Assuma $\lambda_1 \geqslant \lambda_2$ e defina o número de condição espectral $\kappa := \kappa_2(A) = \lambda_1/\lambda_2 \geqslant 1$. A inclinação de e_i em relação ao sistema de coordenadas definido pelos autovetores de A, que depende do ponto inicial, é denotado por $\mu = \xi_2/\xi_1$. Sob essas condições,

$$\omega^2 = 1 - \frac{\left(\xi_1^2\lambda_1^2 + \xi_2^2\lambda_2^2\right)^2}{\left(\xi_1^2\lambda_1^3 + \xi_2^2\lambda_2^3\right)\left(\xi_1^2\lambda_1 + \xi_2^2\lambda_2\right)} = 1 - \frac{(\kappa^2 + \mu^2)^2}{(\kappa + \mu^2)(\kappa^3 + \mu^2)}. \quad (4.2.16)$$

Na Figura 4.3 apresentamos o gráfico da convergência do método de máxima descida. Observe que para valores pequenos de κ e/ou μ a taxa é menor e, portanto, a convergência é mais rápida. Por outro lado, se o sistema linear é mal condicionado, ou seja, $\kappa_2(A)$ é grande, então a taxa se aproxima de 1 definindo uma convergência lenta.

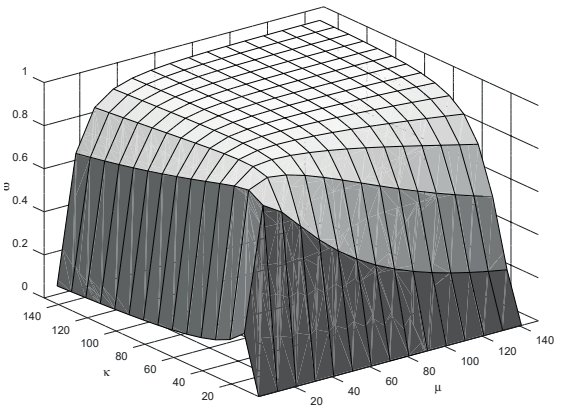

Figura 4.3: Gráfico da convergência do método de máxima descida.

Fixe o número de condição κ, pois a matriz A é dada. Com um pouco de Cálculo I, encontramos que a função $\omega(\mu)$ é maximizada quando $\mu^2 = \kappa^2$. Utilizando esse fato podemos determinar um limitante superior para a taxa de convergência ω:

$$\omega^2 \leqslant 1 - \frac{4\kappa^4}{\kappa^5 + 2\kappa^4 + \kappa^3} = \frac{\kappa^5 - 2\kappa^4 + \kappa^3}{\kappa^5 + 2\kappa^4 + \kappa^3} = \frac{(\kappa - 1)^2}{(\kappa + 1)^2}.$$

Portanto,

$$\omega \leqslant \frac{\kappa - 1}{\kappa + 1}. \tag{4.2.17}$$

A Figura 4.4 mostra o comportamento da taxa de convergência do método de máxima descida em função do numero de condição $\kappa_2(A)$. Observe que há uma estrita faixa onde ω é baixo e, portanto, a convergência é rápida. Rapidamente ω atinge valores ao redor de $0,95$ caracterizando uma convergência lenta.

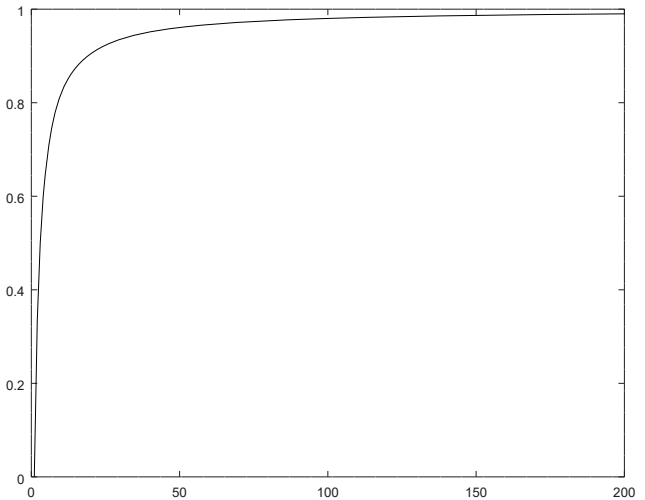

Figura 4.4: Gráfico da convergência do método de máxima descida.

Como já dito anteriormente, (4.2.17) é válida para $n > 2$ [91] e, nesse caso,

$$\kappa_2(A) = \frac{\lambda_{\max}}{\lambda_{\min}}$$

onde A é uma matriz definida positiva. Ademais,

$$\|e_i\|_A \leqslant \|e_0\|_A \left(\frac{\kappa_2(A)-1}{\kappa_2(A)+1}\right)^i \tag{4.2.18}$$

e

$$\frac{f(x_i)-f(x_*)}{f(x_0-f(x_*))} \overset{(4.2.7)}{=} \frac{\frac{1}{2}e_i^T A e_i}{\frac{1}{2}e_0^T A e_0} \leqslant \left(\frac{\kappa_2(A)-1}{\kappa_2(A)+1}\right)^{2i}.$$

Daí,

$$\lim_{k\to\infty} \|e_k\|_A = \|e_0\|_A \lim_{k\to\infty} \left(\frac{\kappa_2(A)-1}{\kappa_2(A)+1}\right)^k = 0$$

e

$$\lim_{k\to\infty} [f(x_i)-f(x_*)] = [f(x_0)-f(x_*)] \lim_{k\to\infty} \left(\frac{\kappa_2(A)-1}{\kappa_2(A)+1}\right)^{2k} = 0.$$

Para demonstrar o caso geral vamos precisar de um lema conhecido como desigualdade de Kantorovich [74].

Lema 4.1. *[158, pág. 186] Seja $A \in \mathbb{R}^{n\times n}$ uma matriz definida positiva com λ_1 o menor autovalor de A e λ_n o maior autovalor de A. Então,*

$$\frac{x^T A x}{x^T x}\,\frac{x^T A^{-1} x}{x^T x} \leqslant \frac{(\lambda_1+\lambda_n)^2}{4\lambda_1\lambda_n}.$$

Demonstração. Sem perda de generalidade podemos assumir que $x \in \mathbb{R}^n$ seja unitário. Então

$$x = \sum_{j=1}^{n} x_j w_j,$$

com $\{w_1, \ldots, w_n\}$ uma base ortonormal de $\mathbb{R}^n$ formada por autovetores de A. Assim,

$$\begin{cases} x^T A x = \displaystyle\sum_{j=1}^{n} \lambda_j x_j^2, \\ \\ x^T A^{-1} x = \displaystyle\sum_{j=1}^{n} \lambda_j^{-1} x_j^2, \end{cases}$$

onde $x_j^2 \in [0, 1]$. Definamos

$$\lambda := \sum_{j=1}^{n} \lambda_j x_j^2 = x^T A x.$$

Assim, λ é a combinação convexa de autovalores de A e, portanto, $\lambda \in [\lambda_1, \lambda_n]$. Definamos pontos sobre o gráfico de $g(t) = 1/t$,

$$P_j := \left(\lambda_j, \frac{1}{\lambda_j}\right)$$

e

$$P := \left(\lambda, \sum_{j=1}^{n} \lambda_j^{-1} x_j^2\right).$$

A função g é convexa, então os pontos P_j estão abaixo do segmento de reta que liga P_1 a P_n, e observe que o ponto P pode ser visto como

$$P = \sum_{j=1}^{n} x_j^2 P_j,$$

ou seja, P é uma combinação convexa dos pontos P_j e, portanto, está contido na envoltória convexa de $P_1, \ldots, P_n$. Assim, P não pode estar acima do segmento de reta que liga P_1 a P_n, que é dado por

$$h(t) = \frac{\lambda_1 + \lambda_n + t}{\lambda_1 \lambda_n}.$$

Assim,

$$x^T A^{-1} x = \sum_{j=1}^{n} \lambda_j^{-1} x_j^2 \leqslant h(\lambda) = \frac{\lambda_1 + \lambda_n + \lambda}{\lambda_1 \lambda_n}.$$

Logo,

$$(x^T A x)(x^T A^{-1} x) \leqslant \lambda \frac{\lambda_1 + \lambda_n + \lambda}{\lambda_1 \lambda_n}.$$

A expressão do lado direito é uma função quadrática cujo máximo é

$$\lambda = \frac{\lambda_1 + \lambda_n}{2},$$

que nos leva ao resultado desejado. □

Teorema 4.7. *[158, pág. 187] Sejam $A \in \mathbb{R}^{n\times n}$ uma matriz definida positiva e $b \in \mathbb{R}^n$. O método de máxima descida converge, para qualquer $x_0 \in \mathbb{R}^n$, para a solução do sistema linear $Ax = b$, que denotamos x_*. Ademais, se as iteradas do método são denotadas por $\{x_j\}$, então*

$$\|x_* - x_i\|_A \leqslant \|x_* - x_0\|_A \left(\frac{\kappa_2(A) - 1}{\kappa_2(A) + 1}\right)^i. \tag{4.2.19}$$

Demonstração. Comecemos com duas relações envolvendo o erro e o resíduo em cada iteração,

$$\begin{cases} e_{i+1} = x_* - x_i - \alpha_i r_i = e_i - \alpha_i r_i, \\ \\ r_{i+1} = b - Ax_{i+1} = A(x_* - x_{i+1}) = Ae_{i+1}. \end{cases}$$

Logo,

$$\|e_{i+1}\|_A^2 = e_{i+1}^T A(e_i - \alpha_i r_i) = e_{i+1}^T Ae_i - \alpha_i e_{i+1}^T Ar_i.$$

Da ortogonalidade de r_i e r_{i+1} (Verifique!) e, como $r_{i+1} = Ae_{i+1}$, mostramos que

$$e_{i+1}^T Ar_i = (Ae_{i+1})^T r_i = r_{i+1}^T r_i = 0.$$

Portanto,

$$\begin{aligned} \|e_{i+1}\|_A^2 &= e_i Ae_{i+1} = r_i^T e_{i+1} = r_i^T(e_i - \alpha_i r_i) \\ &= r_i^T e_i - \alpha_i r_i^T r_i = (Ae_i)^T e_i - \alpha_i r_i^T r_i \\ &= (Ae_i)^T e_i \left(1 - \alpha_i \frac{r_i^T r_i}{(Ae_i)^T e_i}\right) \\ &= \|e_i\|_A^2 \left(1 - \frac{r_i^T r_i}{r_i^T Ar_i} \frac{r_i^T r_i}{r_i^T A^{-1} r_i}\right). \end{aligned}$$

Pela desigualdade de Kantorovich, obtemos

$$\begin{aligned} \|e_{i+1}\|_A^2 &\leqslant \|e_i\|_A^2 \left(1 - \frac{4\lambda_1\lambda_n}{(\lambda_1 + \lambda_n)^2}\right) \\ &= \|e_i\|_A^2 \frac{(\lambda_n - \lambda_1)^2}{(\lambda_n + \lambda_1)^2}. \end{aligned}$$

Dessa forma,

$$\|e_{i+1}\|_A^2 \leqslant \frac{\kappa_2(A) - 1}{\kappa_2(A) + 1} \|e_i\|_A^2$$

e para obter o resultado, basta iterar a desigualdade. □

Como corolário desse teorema podemos estimar a quantidade de iterações do método de máxima descida de forma a reduzir $\|e_0\|_A$ a $\|e_{k_*}\|_A < \epsilon\|e_0\|_A$. Essa estimativa é válida apenas para aritmética exata.

Corolário 4.1. *[158, pág. 187] Seja $A \in \mathbb{R}^{n\times n}$ uma matriz definida positiva. Se as iteradas do método são denotadas por $\{x_j\}$, então para um dado $\epsilon > 0$, $\|e_k\|_A < \epsilon\|e_0\|_A$ quando*

$$k \geqslant \frac{1}{2}\left[\kappa_2(A)\ln\left(\frac{1}{\epsilon}\right)\right]. \tag{4.2.20}$$

Demonstração. Para facilitar a notação defina $\kappa = \kappa_2(A)$. De (4.2.19) concluímos que para o erro relativo ser menor que ϵ devemos ter

$$\left(\frac{\kappa+1}{\kappa-1}\right)^k \geqslant \frac{1}{\epsilon}.$$

Aplicando o logaritmo natural na desigualdade acima, obtemos

$$k \geqslant \frac{\ln\left(\dfrac{1}{\epsilon}\right)}{\ln\left(\dfrac{\kappa+1}{\kappa-1}\right)}. \tag{4.2.21}$$

Fazendo a expansão em série, já que $\kappa \geqslant 1$,

$$\ln\left(\frac{\kappa+1}{\kappa-1}\right) = 2\sum_{i=1}^{\infty}\frac{1}{2i+1}\kappa^{-2i-1} \geqslant \frac{2}{\kappa}. \tag{4.2.22}$$

Substituindo (4.2.22) em (4.2.21), obtemos (4.2.20). □

Passemos, agora, à discussão sobre a complexidade do método de máxima descida (Algoritmo 14). A cada iteração, a multiplicação matriz-vetor é o fator com mais operações, em geral da ordem $\mathcal{O}(n\cdot m)$, onde m é a quantidade de elementos não nulos da matriz A dada. Por outro lado, o número de iterações para garantir a convergência para uma dada tolerância ϵ é definida por (4.2.20). Assim, a complexidade do método de máxima descida é da ordem $\mathcal{O}(nm\kappa)$, onde $\kappa := \kappa_2(A)$.

Apesar do método de máxima descida ter garantia de convergência, sua taxa de convergência é muito baixa, embora haja variações desse método que convergem mais rapidamente. Ao leitor interessado recomendamos [6, 9, 37, 39, 106, 130]. Outras técnicas igualmente convergentes e com taxa de convergência mais alta foram estudadas, a mais conhecida é o método dos gradientes conjugados [64].

4.2.3 Subespaços Interessantes

Para o trabalho de determinar o minimizador da forma quadrática f poderíamos utilizar o método de máxima descida, porém sua convergência é bastante lenta (Figura 4.4). Uma forma de melhorar a performance desse método é aumentando a dimensão do espaço de pesquisa (*search space*). Lembre que a k-ésima iteração desse método trabalha sobre o espaço afim $x_k + span\{\nabla f(x_k)\}$.

Suponha que queiramos resolver $Ax = b$ e que haja uma cadeia de subespaços $S_1 \subseteq S_2 \subseteq \cdots \subseteq S_n = \mathbb{R}^n$, com $dim(S_k) = k$. Ademais, dado o ponto inicial x_0, suponha que no k-ésimo passo de iteração encontramos

$$x_k = \min_{x_0+S_k} f(x). \tag{4.2.23}$$

Com essas hipóteses, temos $f(x_1) \geqslant f(x_2) \geqslant \cdots \geqslant f(x_n) = f(x_*)$ e como $S_n = \mathbb{R}^n$, concluiremos que $x_* = A^{-1}b$. Nosso objetivo é produzir uma cadeia de subespaços S_k com as propriedades supracitadas. Porém, essa abordagem tem um ponto fraco para n grande e, para evitar esse tipo de problema precisamos que essa convergência seja rápida de forma a terminar na iteração $k \ll n$.

Como já dito, a direção de maior decrescimento da forma quadrática f em x_k é a direção do gradiente,

$$g_k = \nabla f(x_k) \overset{(4.2.6)}{=} Ax_k - b.$$

Faz sentido que S_k inclua x_k e g_k pois, dessa forma, garantimos que a iteração do método de gradientes conjugados será pelo menos tão boa quanto uma iteração do método de máxima descida, ou seja,

$$\min_{x \in x_0+S_{k+1}} f(x_k) = f(x_{k+1}) \leqslant \min_{\alpha \in \mathbb{R}} f(x_k + \alpha r_k). \tag{4.2.24}$$

Seja x_0 o ponto inicial e $g_0 \coloneqq Ax_0 - b$. Como $\nabla f(x_k) \in span\{x_k, Ax_k\}$, então a única opção é

$$S_k = \mathcal{K}_k(A, g_0).$$

Uma base conveniente para S_k é determinada pelo algoritmo de Lanczos (Algoritmo 13, pág. 105). Antes de passarmos à descrição do método dos gradientes conjugados, vejamos alguns resultados que vão ao encontro dessa discussão e que nos ajudam a entender melhor as propriedades de minimização do método.

O primeiro resultado afirma que minimizar a forma quadrática f sobre um subespaço de $\mathbb{R}^n$ é o mesmo que minimizar $\|x - x_*\|_A$ nesse subespaço.

Teorema 4.8. *[77, pág. 12] Seja $S \subset \mathbb{R}^n$. Se x_k minimiza f sobre S, então x_k também minimiza $\|x_* - x\|_A = \|r\|_{A^{-1}}$ sobre S, onde $r = b - Ax$.*

Demonstração. Note que,

$$\|x_* - x\|_A^2 = (x_* - x)^T A(x_* - x) = x^T Ax - x^T Ax_* - x_*^T Ax + x_*^T Ax_*.$$

Como A é simétrica e $Ax_* = b$,

$$-x^T Ax_* - x_*^T Ax = -2x^T Ax_* = -2x^T b$$

e, portanto,

$$\|x_* - x\|_A^2 = 2f(x) + x_*^T Ax_* - 2c.$$

Como f é independente de $x_*^T Ax_*$, então minimizar f é equivalente a minimizar $\|x_* - x\|_A$. Seja $e = x - x_*$, então

$$\|e\|_A^2 = e^T Ae = (Ae)^T A^{-1}(Ae) = [A(x - x_*)]^T A^{-1}[A(x - x_*)] = \|r\|_{A^{-1}}^2.$$

□

A consequência imediata dessa proposição é que como x_k minimiza f sobre $x_0 + \mathcal{K}_k(A, g_0)$, então

$$\|x_* - x_k\|_A \leqslant \|x_* - w\|_A \tag{4.2.25}$$

para todo $w \in x_0 + \mathcal{K}_k(A, g_0)$. Como $w \in x_0 + \mathcal{K}_k(A, g_0)$, então

$$w = x_0 + \sum_{j=0}^{k-1} \gamma_j A^j r_0 \;\Rightarrow\; x_* - w = x_* - x_0 - \sum_{j=0}^{k-1} \gamma_j A^j r_0.$$

Mas, $r_0 = b - Ax_0 = A(x_* - x_0)$ e, portanto,

$$x_* - w = x_* - x_0 - \sum_{j=0}^{k-1} \gamma_j A^{j+1}(x_* - x_0) = p(A)(x_* - x_0),$$

onde

$$p(z) = 1 - \sum_{j=0}^{k-1} \gamma_j z^{j+1}$$

é um polinômio de grau k e tal que $p(0) = 1$. Logo,

$$\|x_* - x_k\|_A = \inf_{p \in \mathcal{P}_k,\, p(0)=1} \|p(A)(x_* - x_0)\|_A, \tag{4.2.26}$$

onde $\mathcal{P}_k$ é o espaço vetorial dos polinômios de grau k. Como A é definida positiva, pelo Teorema Espectral, $A = UDU^T$, com U uma matriz ortogonal e D uma matriz diagonal cujas entradas são os autovalores de A que, por sua vez, são positivos. Assim,

$$p(A) = Up(D)U^T.$$

Defina $A^{1/2} := UD^{1/2}U^T$, onde $[D^{1/2}]_{ii} = \sqrt{d_{ii}}$.

Lema 4.2. *[77, pág. 16] Seja $A \in \mathbb{R}^{n\times n}$ uma matriz definida positiva com autovalores positivos $\lambda_1 \geqslant \lambda_2 \geqslant \cdots \geqslant \lambda_n$. Então, para todo $z \in \mathbb{R}^n$,*

$$\|A^{1/2}z\|_2 = \|z\|_A \tag{4.2.27}$$

e

$$\lambda_n^{1/2}\|z\|_A \leqslant \|Az\|_2 \leqslant \lambda_1^{1/2}\|z\|_A. \tag{4.2.28}$$

Demonstração. A identidade (4.2.27) segue de

$$\|z\|_A^2 = z^T Az = \left(A^{1/2}z\right)^T \left(A^{1/2}z\right) = \|A^{1/2}z\|_2^2.$$

Para demonstrar (4.2.28) tome u_i um autovalor unitário associado a λ_i. Pelo Teorema Espectral, $A = UDU^T$ e, portanto,

$$Az = \sum_{i=1}^{n} \lambda_i (u_i^T z) u_i.$$

Assim,

$$\begin{aligned}
\lambda_n \|z\|_A^2 &= \lambda_n \|A^{1/2}z\|_2^2 = \lambda_n \sum_{i=1}^{n} \lambda_i (u_i^T z)^2 \\
&\leqslant \sum_{i=1}^{n} \lambda_i^2 (u_i^T z)^2 \leqslant \lambda_1 \sum_{i=1}^{n} \lambda_i (u_i^T z)^2 \\
&= \lambda_1 \|A^{1/2}z\|_2^2 = \lambda_1 \|z\|_A^2.
\end{aligned}$$

Como

$$\|Az\|_2^2 = \sum_{i=1}^{n} \lambda_i^2 (u_i^T z)^2,$$

vale a desigualdade (4.2.28). □

Pelo Lema 4.2, para todo $x \in \mathbb{R}^n$,

$$\|p(A)z\|_A = \|A^{1/2}p(A)z\|_2 \leqslant \|p(A)\|_2 \|A^{1/2}z\|_2 \leqslant \|p(A)\|_2 \|z\|_A.$$

De (4.2.26) combinado com esse resultado, temos

$$\|x_* - x_k\|_A \leqslant \|x_* - x_0\|_A \inf_{p\in\mathcal{P}_k,\, p(0)=1} \max_{z\in\sigma(A)} |p(z)|, \tag{4.2.29}$$

onde $\sigma(A)$ é o espectro de A e, portanto,

$$\frac{\|x_* - x_k\|_A}{\|x_* - x_0\|_A} \leqslant \max_{z\in\sigma(A)} |\overline{p}_k(z)|.$$

O polinômio $\overline{p}_k$ é chamado de polinômio residual [141].

A partir desse ponto seguiremos a proposta de Golub e Van Loan [58, pág. 628] para apresentar três versões do método de gradientes conjugados.

4.2.4 Método dos Gradientes Conjugados: Versão 1

A primeira versão do método dos gradientes (CG) é uma versão "bruta", não lapidada, por assim dizer, mas do ponto de vista teórico fornece os rudimentos do CG.

O algoritmo de Lanczos (Algoritmo 13, pág. 105) gera uma base para subespaços de Krylov e, em sua k-ésima iteração, construímos

$$Q = [q_1 \mid \ldots \mid q_k] \in \mathbb{R}^{n\times k}$$

com as colunas formando um conjunto ortonormal, a matriz tridiagonal

$$T_k = \begin{bmatrix} \alpha_1 & \beta_1 & 0 & \cdots & 0 \\ \beta_1 & \alpha_2 & \ddots & & \vdots \\ 0 & \ddots & \ddots & \ddots & 0 \\ \vdots & & \ddots & \ddots & \beta_{k-1} \\ 0 & \cdots & 0 & \beta_{k-1} & \alpha_k \end{bmatrix}$$

e o vetor $r_k \in \mathrm{Im}(Q)^{\perp}$ de forma que

$$AQ_k = Q_k T_k + r_k e_k^T,$$

onde $r_k = (A - \alpha_k I)q_k - \beta_{k-1}q_{k-1}$ e e_k é a k-ésima coluna da matriz identidade de $\mathbb{R}^{n\times n}$. A matriz T_k é definida positiva, pois $T_k = Q_k^T A Q_k$. Para facilitar a notação chamamos a matriz de Hessenberg $H_{k,k}$ simétrica (matriz tridiagonal) de T_k. A solução para o problema (4.2.23) é encontrada definindo $q_1 = r_0/\beta_0$, onde o resíduo $r_0 = -g_0 = b - Ax_0$ e $\beta_0 = \|r_0\|_2$. Como Q_k é uma base para $S_k = \mathcal{K}_k(A, g_0)$, minimizar f sobre $x_0 + S_k$ é equivalente à

$$\min_{y\in\mathbb{R}^k} f(x_0 + Q_k y). \tag{4.2.30}$$

Mas,

$$\begin{aligned} f(x_0 + Q_k y) &= \frac{1}{2}(x_0 + Q_k y)^T A(x_0 + Q_k y) - (x_0 + Q_k y)^T b + c \\ &= f(x_0) + \frac{1}{2} y^T (Q_k^T A Q_k) y - y^T (Q_k^T r_0) \end{aligned} \tag{4.2.31}$$

e $\beta_0 Q_k(:,1) = r_0$. O ponto crítico de (4.2.31) é determinado por

$$\nabla f(x_0 + Q_k y) = (Q_k^T A Q_k) y - Q_k^T r_0 = 0$$

se, e somente se,

$$T_k y_k = Q_k^T r_0 = \beta_0 e_1,$$

onde y_k é o minimizador do problema (4.2.30). Um algoritmo para o método dos gradientes conjugados (versão 1) é

Algoritmo 15 Método dos Gradientes Conjugados: Versão 1

```
function x_*=CGV1(A, b, x_0)
    k = 0;  r_0 = b − Ax_0;  β_0 = ‖r_0‖_2;  q_0 = 0;
    while β_k ≠ 0 do
        q_{k+1} = r_k/β_k;
        k = k + 1;
        α_k = q_k^T A q_k;
        T_k y_k = β_0 e_1;
        x_k = Q_k y_k;
        r_k = (A − α_k I) q_k − β_{k−1} q_{k−1};
        β_k = ‖r_k‖_2;
    end while
    x_* = x_k;
end function
```

Esse algoritmo sofre de dois problemas. O primeiro é que x_k é determinado através de um produto matriz-vetor, além de haver um sistema linear a ser resolvido, fazendo-o não muito eficaz para problemas de grande porte. De fato, esse algoritmo tem complexidade $\mathcal{O}(n^3)$ no pior cenário. Obviamente, que aqui estamos pensando em resolução por fatoração LU ou Cholesky, porém existem outros algoritmos que envolvem produtos matriz-matriz como o algoritmo de Coppersmith-Winograd que assintoticamente tem complexidade da $\mathcal{O}(n^{2,375})$, mas não faz parte do escopo desse livro. O segundo é que o algoritmo é instável em aritmética finita. Por isso, vamos aprimorá-lo na próxima seção.

Mas antes, vejamos três resultados equivalentes que tratam sobre a convergência finita do método do gradientes conjugados.

Teorema 4.9. *[58, pág. 630] Se k_* é a dimensão do menor espaço invariante que contém r_0, então o Algoritmo 15 do método de gradientes conjugados termina com $x_{k_*} = x_*$.*

Teorema 4.10. *[77, pág. 15] Seja $A \in \mathbb{R}^{n\times n}$ uma matriz definida positiva com autovetores $u_1, \ldots, u_n$. Se b é a combinação linear de k_* autovetores de A, isto é,*

$$b = \sum_{j=1}^{k_*} \gamma_j u_{i_j},$$

então o Algoritmo 15 do método de gradientes conjugados aplicado a $Ax = b$ com $x_0 = 0$ termina com $x_{k_*} = x_*$.

Demonstração. Sejam $\lambda_{i_1}, \ldots, \lambda_{i_{k_*}}$ os autovalores associados aos autovetores $u_{i_1}, \ldots, ui_{k_*}$. Portanto,

$$x_* = A^{-1}b = \sum_{j=1}^{k_*} \frac{\gamma_j}{\lambda_{i_j}} u_{i_j}.$$

Considere o seguinte polinômio residual,

$$\overline{p}(z) = \prod_{j=1}^{k_*} \frac{(\lambda_{i_j} - z)}{\lambda_{i_j}}$$

em que, claramente, $\overline{p}(0) = 1$ e $\overline{p} \in \mathcal{P}_k$. Ademais, $\overline{p}(\lambda_{i_j}) = 0$, para $1 \leqslant j \leqslant k_*$. Assim,

$$\overline{p}(A)x_* = \sum_{j=1}^{k_*} \overline{p}(\lambda_{i_j}) \frac{\gamma_j}{\lambda_{i_j}} u_{i_j} = 0.$$

Por (4.2.26) e já que $x_0 = 0$, obtemos

$$\|x_{k_*} - x_*\|_A \leqslant \|\overline{p}(A)x_*\|_A = 0,$$

como queríamos demonstrar. □

Teorema 4.11. *[77, pág. 15] Seja $A \in \mathbb{R}^{n\times n}$ uma matriz definida positiva. Assuma que existem exatamente $k_* \leqslant n$ autovalores distintos de A. Então, o Algoritmo 15 do método de gradientes conjugados termina com $x_{k_*} = x_*$.*

A demonstração desse teorema é similar à demonstração do Teorema 4.10 e deixada a cargo do leitor. Como o maior subespaço invariante de $\mathbb{R}^n$ é o próprio $\mathbb{R}^n$, então o método dos gradientes conjugados converge no máximo em n iterações, em aritmética exata.

4.2.5 Método dos Gradientes Conjugados: Versão 2

A versão 2 do método dos gradientes conjugados é um aprimoramento da versão 1. O objetivo dessa versão é trabalhar em cima do sistema linear $T_k y_k = \beta_0 e_1$ e do produto matriz vetor $x_k = Q_k y_k$, evitando a utilização de vetores de Lanczos para as atualizações supracitadas.

Como $T_k = Q_k^T A Q_k$ é definida positiva possui fatoração $T_k = L_k D_k L_k^T$, onde

$$L_k = \begin{bmatrix} 1 & 0 & \cdots & 0 \\ l_1 & 1 & \cdots & 0 \\ \vdots & \ddots & \ddots & \vdots \\ 0 & \cdots & l_k & 1 \end{bmatrix} \quad \text{e} \quad D_k = \begin{bmatrix} d_1 & 0 & \cdots & 0 \\ 0 & d_2 & \cdots & 0 \\ \vdots & \ddots & \ddots & \vdots \\ 0 & \cdots & 0 & d_k \end{bmatrix}.$$

Realizando o produto $L_k D_k L_k^T$ e comparando com as entradas de T_k, obtemos

$$\begin{cases} d_1 = \alpha_1, \\ \\ l_{i-1} = \beta_{i-1}/d_{i-1} \text{ para } i = 2, \ldots, n, \\ \\ d_i = \alpha_i - l_{l-1}\beta_{i-1} \text{ para } i = 2, \ldots, n. \end{cases}$$

Agora, vamos transformar o sistema linear $T_k = L_k D_k L_k^T y_k = \beta_0 e_1$ em dois sistemas lineares, a saber,

$$\begin{cases} L_k D_k v_k = \beta_0 e_1, \\ \\ L_k^T y_k = v_k. \end{cases}$$

Por outro lado, se $C_k \in \mathbb{R}^{n\times k}$ satisfaz

$$C_k L_k^T = Q_k, \tag{4.2.32}$$

então

$$x_k = x_0 + Q_k y_k = x_0 + C_k L_k^T y_k = x_0 + C_k v_k. \tag{4.2.33}$$

Ainda assim, (4.2.33) não está boa, pois mesmo trocando $x_k = x_0 + Q_k y_k$ por $x_k = x_0 + C_k v_k$, C_k continua sendo uma matriz densa que envolve vetores de Lanczos. A diferença entre Q_k e C_k é que C_k pode ser determinada iterativamente. Ademais, também existe uma relação recursiva para v_k: considere a bidiagonal inferior do sistema linear $L_k D_k v_k = \beta_0 e_1$. Concluímos que

$$\left[\begin{array}{cccc|c} d_1 & 0 & \cdots & 0 & 0 \\ d_1 l_1 & d_2 & 0 & \cdots & 0 \\ 0 & d_1 l_1 & d_3 & \cdots & 0 \\ \vdots & & \ddots & \ddots & \vdots \\ \hline 0 & \cdots & 0 & d_{k-1} l_{k-1} & d_k \end{array}\right] \left[\begin{array}{c} \nu_1 \\ \nu_2 \\ \nu_3 \\ \vdots \\ \hline \nu_k \end{array}\right] = \left[\begin{array}{c} \beta_0 \\ 0 \\ 0 \\ \vdots \\ \hline 0 \end{array}\right].$$

Assim,

$$v_k = \left[\begin{array}{c} \nu_1 \\ \vdots \\ \nu_{k-1} \\ \hline \nu_k \end{array}\right] = \left[\begin{array}{c} v_{k-1} \\ \hline \nu_k \end{array}\right], \tag{4.2.34}$$

onde

$$\nu_k = \begin{cases} \beta_0 / d_1 & \text{para } k = 1, \\ \\ -d_{k-1} l_{k-1} \nu_{k-1} / d_k & \text{para } k > 1. \end{cases} \tag{4.2.35}$$

Agora considere a equação (4.2.32)

$$[c_1 \,|\, c_2 \,|\, c_3 \,|\, \cdots \,|\, c_k] \begin{bmatrix} 1 & l_1 & 0 & \cdots & 0 \\ 0 & 1 & l_2 & \cdots & 0 \\ \vdots & & \ddots & \ddots & \vdots \\ 0 & \cdots & 0 & 1 & l_{k-1} \\ 0 & \cdots & 0 & 0 & 1 \end{bmatrix} = [q_1 \,|\, q_2 \,|\, q_3 \,|\, \cdots \,|\, q_k].$$

Concluímos, então, que

$$C_k = [C_{k-1} \,|\, c_k], \tag{4.2.36}$$

onde

$$c_k = \begin{cases} q_1 & \text{para } k = 1, \\ q_k - l_{k-1}c_{k-1} & \text{para } k > 1. \end{cases} \tag{4.2.37}$$

De (4.2.34) e (4.2.36), obtemos

$$x_k = x_0 + C_k v_k = x_0 + C_{k-1}v_{k-1} + v_k c_k = x_{k-1} + \nu_k c_k.$$

Dessa forma, o vetor x_k pode ser determinado recursivamente e não via uma resolução de sistema linear com um produto matriz-vetor. Utilizando (4.2.35) e (4.2.37) no Algoritmo 15, obtemos a segunda versão do método de gradientes conjugados. Cada iteração, agora, tem apenas um produto matriz-vetor e as outras operações que somam $13n$ flops.

Algoritmo 16 Método dos Gradientes Conjugados: Versão 2

```
function x_*=CGV2(A, b, x_0)
    k = 0;  r_0 = b − Ax_0;  β_0 = ‖r_0‖_2;  q_0 = 0;  c_0 = 0
    while β_k ≠ 0 do
        q_{k+1} = r_k/β_k;
        k = k + 1;
        α_k = q_k^T A q_k;
        if k=1 then
            d_1 = α_1;  ν_1 = β_0/d_1;  c_1 = q_1;
        else
            l_{k−1} = β_{k−1}/d_{k−1};  d_k = α_k − β_{k−1} l_{k−1};
            ν_k = −β_{k−1} ν_{k−1}/d_k;  c_k = q_k − l_{k−1} c_{k−1};
        end if
        x_k = x_{k−1} + ν_k c_k;
        r_k = (A − α_k I) q_k − β_{k−1} q_{k−1};
        β_k = ‖r_k‖_2;
    end while
    x_* = x_k;
end function
```

Como dito anteriormente, o método de gradientes conjugados, em aritmética exata, converge em no máximo n iterações para uma matriz $\mathbb{R}^{n\times n}$ definida positiva. Para outras discussões teóricas sobre esse algoritmo recomendamos [58, pág. 628].

Vejamos agora o resultado que justifica o nome método dos gradientes conjugados, ou seja, $\nabla f(x_i)^T \nabla f(x_j) = 0$ para $i \neq j$.

Teorema 4.12. *[58, pág. 633] Se $x_1, \ldots, x_k$ são gerados pelo Algoritmo 16 (pág. 125), então $g_i^T g_j = 0$ para todo $i \neq j$ e $1 \leqslant i, j \leqslant k$. Ademais $g_k = \nu_k r_k$, onde ν_k e r_k são definidos pelo Algoritmo 16.*

Demonstração. Observe que da tridiagonalização parcial

$$g_k = Ax_k - b = A(x_0 + Q_k y_k) - b = -r_0 + (Q_k T_k + r_k e_k^T) y_k.$$

Como $Q_k T_k y_k = \beta_0 Q_k e_1 = r_0$, então $g_k = (e_k^T y_k) r_k$. Mas r_k é múltiplo de q_{k+1} e, portanto, os vetores g_k são mutuamente ortogonais. A segunda afirmação é trivial. □

O próximo resultado demonstra que as direções de busca $c_1, \ldots, c_k$ são A-conjugadas (A-ortogonais) no produto interno induzido por A, isto é, $\langle c_i, c_j \rangle_A = c_i^T A c_j = 0$ para $i \neq j$.

Teorema 4.13. *[58, pág. 633] Se $c_1, \ldots, c_k$ são gerados pelo Algoritmo 16, então*

$$c_i^T A c_j = \begin{cases} 0 & \text{para } i \neq j, \\ d_j & \text{para } i = j, \end{cases}$$

onde $1 \leqslant i, j \leqslant k$.

Demonstração. Sabe-se que $Q_k = C_k L_k^T$ e $T_k = Q_k^T A Q_k$ e, portanto,

$$T_k = L_k (C_k^T A C_k) L_k^T$$

Pela unicidade da fatoração LDL^T, temos

$$D_k = C_k^T A C_k \;\Rightarrow\; c_i^T A c_j = [D_k]_{ij}.$$

□

4.2.6 Método dos Gradientes Conjugados: Versão Hestenes-Stiefel

A formulação de Hestenes e Stiefel [64] é obtida quando reescrevemos o Algoritmo 16 de forma a evitar os vetores de Lanczos e remover os fatores explícitos referentes à fatoração LDL^T. Uma vantagem dessa alteração é que o critério de parada pode ser formulado em termos do resíduo $r_k = b - Ax_k$ em vez dos vetores de Lanczos $r_k = (A - \alpha_k I) q_k - \beta_{k-1} q_{k-1}$, cuja interpretação geométrica é de mais difícil entendimento.

Considere a direção de busca no Algoritmo 16,

$$c_k = q_k - l_{k-1} c_{k-1}.$$

Como q_k é múltiplo de $g_k = \nabla f(x_k)$, a direção de pesquisa pode ser reescalonada, digamos para p_k, de forma que

$$p_k = g_{k-1} + \tau_{k-1} p_{k-1}$$

e, consequentemente,

$$A p_k = A g_{k-1} + \tau_{k-1} A p_{k-1}.$$

Pela A-ortogonalidade das direções de busca c_k (Teorema 4.13), $p_i^T A p_j = 0$ para $i \neq j$. Portanto,

$$\tau_{k-1} = -\frac{p_{k-1}^T A g_{k-1}}{p_{k-1}^T A p_{k-1}} \tag{4.2.38}$$

e

$$p_k^T A g_{k-1} = p_k^T A p_k. \tag{4.2.39}$$

Como p_k é um múltiplo de c_k, a fórmula de atualização $x_k = x_{k-1} + \nu_k c_k$ no Algoritmo 16 toma a forma

$$x_k = x_{k-1} + \mu_k p_k$$

e, portanto, $Ax_k = Ax_{k-1} + \mu_k Ap_k$. Subtraindo b de ambos os lados, obtemos

$$g_k = g_{k-1} + \mu_k Ap_k$$

e, pela A-ortogonalidade dos gradientes (Teorema 4.12) e de (4.2.39), temos

$$\mu_k = \frac{g_{k-1}^T g_{k-1}}{g_{k-1}^T Ap_k} = \frac{g_{k-1}^T g_{k-1}}{p_k^T Ap_k}.$$

Das identidades $g_{k-1} = g_{k-2} + \mu_{k-1} Ap_{k-1}$ e $g_{k-1}^T g_{k-2} = 0$ concluímos que

$$\begin{cases} g_{k-1}^T g_{k-1} = -\mu_{k-1} g_{k-1}^T Ap_{k-1}, \\ \\ g_{k-2}^T g_{k-2} = \mu_{k-1} g_{k-2}^T Ap_{k-1} = \mu_{k-1} p_{k-1}^T Ap_{k-1}. \end{cases}$$

Substituindo as identidades acima em (4.2.38), temos

$$\tau_{k-1} = \frac{g_{k-1}^T g_{k-1}}{g_{k-2}^T g_{k-2}}.$$

Por fim, o resíduo r_k é calculado através de $r_k = -g_k = b - Ax_k$. Reescrevendo o Algoritmo 16, obtemos o algoritmo para o método dos gradientes conjugados.

Algoritmo 17 Método dos Gradientes Conjugados: Versão de Hestenes e Stiefel

```
function x*=CGVHS(A, b, x0)
    k = 0;  r0 = b − Ax0;
    while ||rk||2 > 0 do
        k = k + 1;
        if k=1 then
            pk = r0;
        else
            τk−1 = (r_{k−1}^T r_{k−1})/(r_{k−2}^T r_{k−2});
            pk = r_{k−1} + τk−1 p_{k−1};
        end if
        qk = Apk;
        μk = (r_{k−1}^T r_{k−1})/(pk^T qk);
        xk = x_{k−1} + μk pk;
        rk = r_{k−1} − μk qk;
    end while
    x* = xk;
end function
```

Assim como na fatoração QR, o Algoritmo 17 pode apresentar perda de ortogonalidade entre os resíduos por conta da aritmética finita. Uma apresentação detalhada do assunto pode ser encontrada em [97].

O Lema 4.2 (pág. 120) nos permite demonstrar uma proposição que ajuda a analisar o término das iterações e convergência dos resíduos do método dos gradiente conjugados.

Teorema 4.14. *[77, pág. 16] Se $A \in \mathbb{R}^{n\times n}$ é uma matriz definida positiva, então*

$$\|r_k\|_2 \leqslant \sqrt{\kappa_2(A)}\|r_0\|_2 \frac{\|x_k - x_*\|_A}{\|x_0 - x_*\|_A}.$$

Demonstração. Como A é definida positiva, então pelo Teorema Espectral, $A = UDU^T$, com U uma matriz ortogonal e D uma matriz diagonal cujos elementos são os autovalores de A ($d_{ii} = \lambda_i$). Ademais, $\lambda_1 \geqslant \lambda_2 \geqslant \cdots \geqslant \lambda_n > 0$. Sabemos que $\|A\|_2 = \lambda_1$, $\|A^{-1}\|_2 = 1/\lambda_n$ e $\kappa_2(A) = \lambda_1/\lambda_n$ e, assim,

$$\frac{\|b - Ax_k\|_2}{\|b - Ax_0\|_2} = \frac{\|A(x_* - x_k)\|_2}{\|A(x_* - x_0)\|_2} \leqslant \sqrt{\frac{\lambda_1}{\lambda_n}} \frac{\|x_k - x_*\|_A}{\|x_0 - x_*\|_A},$$

demonstrando o resultado. □

Segue imediatamente dessa proposição que $\|r_{k_*}\|_2 = \|b - Ax_{k_*}\|_2 = 0$, garantindo que tanto o erro, quanto o resíduo, convergem em no máximo n iterações.

Uma característica do CG é que o erro relativo é monotonicamente decrescente entre iterações, conforme o teorema a seguir.

Teorema 4.15. *[158, pág. 196] Seja $A \in \mathbb{R}^{n\times n}$ uma matriz definida positiva. A sequência de iterações do método dos gradientes conjugados é monotonicamente decrescente na A-norma, ou seja, para todo k*

$$\|x_* - x_{k+1}\|_A \leqslant \|x_* - x_k\|_A.$$

Demonstração. De (4.2.25), x_k minimiza f sobre $x_0 + \mathcal{K}_k(A, g_0) \subseteq x_0 + \mathcal{K}_{k+1}(A, g_0)$. Portanto, segue imediatamente que $\|x_* - x_{k+1}\|_A \leqslant \|x_* - x_k\|_A$, pois x_{k+1} minimiza f sobre $x_0 + \mathcal{K}_{k+1}(A, g_0)$. □

Exemplo 4.1. *Para ver esse efeito, considere uma matriz aleatória A, 50×50, com entradas no intervalo $[0, 100]$ e calcule $A^T A$, obtendo uma matriz definida positiva. A Figura 4.5 mostra o gráfico da A-norma do erro ($\|x_* - x_k\|_A$) e a solução "exata" foi determinada por fatoração de Cholesky.*

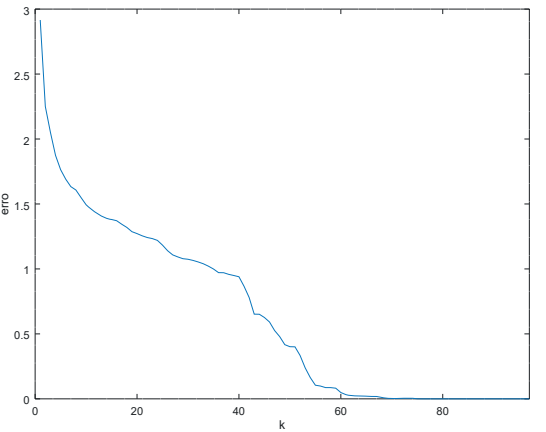

Figura 4.5: Gráfico da A-norma do erro de CG para uma matriz aleatória definida positiva.

O próximo resultado refere-se a uma estimativa para o erro relativo do método dos gradientes conjugados, e foi demonstrado em [27, 32, 73, 95].

Teorema 4.16. *[147, pág. 299] Se x_* é a solução do sistema linear positivo definido $Ax = b$ e é calculada pelo Algoritmo 17 (pág. 127), então*

$$\|x_{k+1} - x_*\|_A \leqslant 2\left(\frac{\sqrt{\kappa_2(A)} - 1}{\sqrt{\kappa_2(A)} + 1}\right)^k \|x_0 - x_*\|_A. \tag{4.2.40}$$

Demonstração. Para facilitar a notação, denote $\kappa := \kappa_2(A)$. Por (4.2.29), basta encontrar um polinômio de forma que seu máximo sobre o espectro de A seja a expressão do meio na desigualdade abaixo,

$$\frac{\|e_k\|_A}{\|e_0\|_A} \leqslant \frac{2}{\left[\left(\frac{\sqrt{\kappa}+1}{\sqrt{\kappa}-1}\right)^k + \left(\frac{\sqrt{\kappa}+1}{\sqrt{\kappa}-1}\right)^{-k}\right]} \leqslant 2\left(\frac{\sqrt{\kappa}-1}{\sqrt{\kappa}+1}\right)^k.$$

O polinômio que satisfaz essa condição é um polinômio de Chebyshev escalado e transladado,

$$\overline{p}_k(x) = \frac{T_k\left(\dfrac{\gamma - 2x}{\lambda_{\max} - \lambda_{\min}}\right)}{T_k(\gamma)},$$

onde T_k é o polinômio de Chebyshev de grau k e

$$\gamma = \frac{\lambda_{\max} + \lambda_{\min}}{\lambda_{\max} - \lambda_{\min}} = \frac{\kappa + 1}{\kappa - 1}.$$

Note que, para $\lambda_{\min} \leqslant x \leqslant \lambda_{\max}$ o argumento do polinômio de Chebyshev no numerador de $\overline{p}_k$ é, em módulo, limitado por 1, isto é, o numerador de $\overline{p}_k$ é limitado superiormente por 1. Portanto, basta demonstrar que

$$T_k(\gamma) = T_k\left(\frac{\kappa+1}{\kappa-1}\right) = \frac{1}{2}\left[\left(\frac{\sqrt{\kappa}+1}{\sqrt{\kappa}-1}\right)^k + \left(\frac{\sqrt{\kappa}+1}{\sqrt{\kappa}-1}\right)^{-k}\right].$$

Considere a seguinte mudança de variáveis

$$x = \frac{1}{2}\left(z + \frac{1}{z}\right).$$

Assim,

$$T_k(x) = \frac{1}{2}\left(z^k + \frac{1}{z^k}\right).$$

Se $x = \gamma$, devemos ter

$$\frac{\kappa+1}{\kappa-1} = z + \frac{1}{z}$$

que implica em uma equação quadrática, a saber,

$$\frac{1}{2}z^2 - \left(\frac{\kappa+1}{\kappa-1}\right)z + \frac{1}{2} = 0,$$

cuja solução é

$$z = \frac{\sqrt{\kappa}+1}{\sqrt{\kappa}-1}.$$

Logo

$$T_k(\gamma) = \frac{1}{2}\left(z^k + \frac{1}{z^k}\right) = \frac{1}{2}\left[\left(\frac{\sqrt{\kappa}+1}{\sqrt{\kappa}-1}\right)^k + \left(\frac{\sqrt{\kappa}+1}{\sqrt{\kappa}-1}\right)^{-k}\right],$$

como queríamos demonstrar. □

Vejamos um exemplo com duas formas de estimar o número mínimo de iterações, em aritmética exata, para garantir a convergência do CG para uma dada tolerância.

Exemplo 4.2. *Esse exemplo foi sugerido por [77, pág. 17]. Assuma* $x_0 = 0$ *e que os autovalores de* A *estão contidos no intervalo* $(9, 11)$. *Tomemos os seguinte polinômio residual*

$$\overline{p}_k(z) = \frac{(10-z)^k}{10^k}.$$

Aplicando (4.2.29), *obtemos*

$$\|x_k - x_*\|_A \leqslant \|x_*\|_A \max_{9 \leqslant z \leqslant 11} |\overline{p}_k(z)| = \|x_*\|_A 10^{-k}.$$

Assim, o tamanho da A-norma do erro relativo será reduzido por um fator de, digamos, 10^{-3} *quando,*

$$10^{-k} \leqslant 10^{-3} \quad \Rightarrow \quad k \geqslant 3.$$

Do Teorema 4.14, obtemos

$$\frac{\|b - Ax_k\|_2}{\|b\|_2} \leqslant \sqrt{\frac{\lambda_1}{\lambda_n}} \frac{\|x_k - x_*\|_A}{\|x_*\|_A}$$

e, substituindo os valores,

$$\frac{\|b - Ax_k\|_2}{\|b\|_2} \leqslant \frac{\sqrt{11} \times 10^{-k}}{3}.$$

Assim, o tamanho da A-norma do resíduo relativo será reduzido por um fator de 10^{-3} *quando,*

$$10^{-k} \leqslant \frac{3 \cdot 10^{-3}}{\sqrt{11}} \quad \Rightarrow \quad k \geqslant 4.$$

Veja que as estimativas que fizemos foram para um polinômio $\overline{p}_k$ *específico. Vejamos estimativas mais precisas que as feitas acima. Pelo Teorema 4.16,*

$$\frac{\|x_{k+1} - x_*\|_A}{\|x_*\|_A} \leqslant 2 \left(\frac{\sqrt{\kappa_2(A)} - 1}{\sqrt{\kappa_2(A)} + 1} \right)^k \tag{4.2.41}$$

e como

$$\frac{\sqrt{\kappa_2(A)} - 1}{\sqrt{\kappa_2(A)} + 1} \leqslant \frac{\sqrt{11} - 3}{\sqrt{11} + 3} \approx 0{,}05,$$

podemos predizer quantas iterações são necessárias para que a A-norma do erro relativo seja reduzido por um fator de 10^{-3}*:*

$$2 \cdot 0{,}05^k < 10^{-3} \quad \Rightarrow \quad k > -\log_{10}(2000)/\log_{10}(0{,}05) \approx 3{,}3/1{,}3 \approx 2{,}6.$$

Parece que a estimativa do número de iterações utilizando (4.2.41) *é o melhor caminho comparada à estimativa determinando um polinômio residual. Mas,* (4.2.41) *aplicada a um problema que tem os autovalores organizados em* clusters *pode produzir um apocalipse numérico. Com efeito, seja* $x_0 = 0$ *e suponha que os autovalores de A estejam organizados em dois* clusters *nos intervalos* $(1; 1{,}5)$ *e* $(399, 400)$*. Assim, sabemos que* $\kappa_2(A) \leqslant 400$*. Aplicando,* (4.2.41)*, obtemos*

$$\frac{\|x_{k+1} - x_*\|_A}{\|x_*\|_A} \leqslant 2(19/21)^k \approx 2(0{,}91)^k.$$

Para reduzir o erro relativo a um fator de 10^{-3}*, temos que*

$$2(0{,}91)^k < 10^{-3} \quad \Rightarrow \quad k > -\log_{10}(2000)/\log_{10}(0{,}91) \approx 80{,}6.$$

Agora, usemos o seguinte polinômio residual

$$\overline{p}_{3k}(z) = \frac{(1{,}25 - z)^k (400 - z)^{2k}}{1{,}25^k 400^{2k}}.$$

Nesse caso,

$$\max_{z \in \sigma(A)} |\overline{p}_{3k}(z)| \leqslant (0{,}25/1{,}25)^k = 0{,}2^k.$$

Para haver uma redução do erro relativo por um fator de 10^{-3} *precisamos que*

$$0{,}2^k < 10^{-3} \quad \Rightarrow \quad k > -3/\log_{10}(0{,}2) \approx 4{,}3.$$

Enquanto a metodologia utilizando (4.2.41) *prevê* 81 *iterações, a metodologia utilizando um determinado polinômio residual prevê* 15 *iterações, que é o menor múltiplo de* 3 *maior que* $3 \times 4{,}3 = 12{,}9$.

Analisando as estimativas de convergência, observamos que quando o número de condição satisfaz $\kappa_2(A) \approx 1$, a convergência é rápida. Se $\kappa_2(A) \gg 1$ a convergência é lenta. A transformação de um problema em outro com os autovalores clusterizados ao redor de 1 é chamado de precondicionamento, que veremos mais a frente.

Como corolário do Teorema 4.16, podemos estimar a quantidade de iterações do método dos gradiente conjugados de forma a reduzir $\|e_0\|_A$ à $\|e_{k_*}\|_A < \epsilon \|e_0\|_A$. Essa estimativa é válida apenas para aritmética exata.

Teorema 4.17. *[158, pág. 212] Seja* $A \in \mathbb{R}^{n \times n}$ *uma matriz definida positiva. Se as iteradas do método dos gradientes conjugados são denotadas por* $\{x_j\}$*, então para um dado* $\epsilon > 0$*,* $\|e_k\|_A < \epsilon \|e_0\|_A$ *quando*

$$k \geqslant \frac{1}{2} \left[\sqrt{\kappa_2(A)} \ln \left(\frac{1}{2\epsilon} \right) \right]. \tag{4.2.42}$$

Demonstração. Para facilitar a notação, defina $\kappa = \kappa_2(A)$. De (4.2.40) concluímos que para o erro relativo ser menor que ϵ devemos ter

$$2 \left(\frac{\sqrt{\kappa} + 1}{\sqrt{\kappa} - 1} \right)^k > \frac{1}{\epsilon}.$$

Aplicando o logaritmo natural na desigualdade acima, obtemos

$$k \geqslant \frac{\ln \left(\dfrac{1}{2\epsilon} \right)}{\ln \left(\dfrac{\sqrt{\kappa} + 1}{\sqrt{\kappa} - 1} \right)}. \tag{4.2.43}$$

Fazendo a expansão em série, já que $\kappa \geqslant 1$,

$$\ln \left(\frac{\sqrt{\kappa} + 1}{\sqrt{\kappa} - 1} \right) = 2 \sum_{i=1}^{\infty} \frac{1}{2i+1} (\sqrt{\kappa})^{-2i-1} \geqslant \frac{2}{\sqrt{\kappa}}. \tag{4.2.44}$$

Substituindo (4.2.43) em (4.2.44), obtemos (4.2.42). $\square$

A título de exemplo, geramos uma matriz aleatória com $\kappa_2(A) = 53,42$ e tomamos $\epsilon = 10^{-4}$. Assim, (4.2.42) forneceu uma estimativa de convergência, em aritmética exata, de 32 iterações e, em aritmética finita, o método convergiu em 57 iterações. Apenas para comparação, o método de máxima descida convergiu em 319 iterações.

Passemos, agora, à discussão mais aprofundada sobre a complexidade do método dos gradientes conjugados (Algoritmo 17). A cada iteração a multiplicação matriz-vetor é o fator com mais operações ($\mathcal{O}(n \cdot m)$, onde m é a quantidade de elementos não nulos da matriz A dada). Por outro lado, o número de iterações do CG para garantir a convergência para uma dada tolerância ϵ é definido por (4.2.42), o que faz a complexidade do método dos gradientes conjugados ser $\mathcal{O}(nm\sqrt{\kappa})$, onde $\kappa := \kappa_2(A)$.

Existem várias técnicas para a dedução do método dos gradientes conjugados, e escolhemos essa abordagem por ser bastante intuitiva. Para uma abordagem via métodos de direções conjugadas sugerimos [132, pág. 21], e para uma abordagem via propriedade de minimização recomendamos [77, pág. 11]. Na verdade, todas essas abordagens são equivalentes.

Muito se publicou e se publica sobre CG, e para o leitor interessado sugerimos alguns artigos relevantes como [31, 40, 48, 63, 148]. Por fim, o método de gradientes conjugados é de extrema importância para problemas aplicados, como podemos ver em [112, 121, 145, 164].

4.2.7 Método dos Gradientes Conjugados para Equações Normais

Sejam $A \in \mathbb{R}^{m\times n}$ com $m \geqslant n$ e $b \in \mathbb{R}^m$. O sistema linear sobredeterminado $Ax = b$ pode ser resolvido no sentido de quadrados mínimos através das equações normais $A^TAx = A^Tb$. Como A^TA é definida positiva podemos aplicar o método dos gradientes conjugados e, dessa forma, a iteração produz um vetor x_k tal que,

$$x_k = \min_{x\in x_0+\mathcal{K}_k(A^TA,A^Tr_0)} f_{A^TA}(x),$$

onde f_{A^TA} é a forma quadrática associada a A^TA e A^Tb, isto é,

$$f_{A^TA}(x) = \frac{1}{2}x^TA^TAx - x^TA^Tb + c = \frac{1}{2}\|Ax-b\|_2^2 - \frac{1}{2}b^Tb + c.$$

Essa abordagem é conhecida como *CGLS*, *Conjugate Gradient Least Squares* [110] ou *CGNR*, *Conjugate Gradient Normal Equation Residual* [44, 98]. O segundo nome é dado, pois o resíduo é minimizado a cada iteração. Com efeito,

$$\begin{aligned}\|x_* - x_k\|_{A^TA}^2 &= (x_* - x_k)A^TA(x_* - x_k) = (Ax_* - Ax_k)^T(Ax_* - Ax_k)\\ &= (b - Ax_k)^T(b - Ax_k) = \|r_k\|_2^2\end{aligned}$$

é minimizado sobre $x_0+\mathcal{K}_k(A^TA, A^Tb)$ em cada iteração. O grande problema desse método é a sua ineficiência, pois $\kappa_2(A^TA) = \kappa_2^2(A)$. Esse fato torna o CGNR muito lento, além da solução aproximada ser bastante sensível a erros de arredondamento.

Abaixo apresentamos um algoritmo do CGLS tomando o cuidado de não calcular A^TA por duas razões: primeiro pelo impacto na complexidade espacial, já que há a necessidade de se guardar a matriz A^TA na memória além da complexidade temporal da $\mathcal{O}(n^3)$ para o cálculo de A^TA. Segundo, a esparsidade da matriz A é reduzida quando passamos para A^TA.

Algoritmo 18 Método dos Gradientes Conjugados: CGLS

function x_*=CGLS(A, b, x_0)
  $k = 0$; $r_0 = b - Ax_0$; $p_0 = s_0 = A^Tr_0$; $\gamma_0 = \|s_0\|_2^2$;
  while $\|r_k\|_2 > 0$ **do**
    $q_k = Ap_k$;
    $\mu_k = \gamma_k/\|q_k\|_2^2$;
    $x_{k+1} = x_k + \mu_k p_k$;
    $r_{k+1} = r_k - \mu_k q_k$;
    $s_{k+1} = A^T r_{k+1}$;
    $\gamma_{k+1} = \|s_{k+1}\|_2^2$;
    $\tau_k = \gamma_{k+1}/\gamma_k$;
    $p_{k+1} = s_{k+1} + \tau_k p_k$;
    $k = k + 1$;
  end while
  $x_* = x_k$;
end function

Esse algoritmo é uma reorganização do Algoritmo 17, mais as adequações necessárias para resolver equações normais. A justificativa dessa alteração é deixada ao leitor.

Observação 4.1. *1. Podemos aplicar ideias similares na resolução de sistemas subdeterminados $Ax = b$, e esse método é chamado de CGNE:* Conjugate Gradient Normal Equation Error *[44], pois o erro é minimizado a cada iteração.*

2. Caso $A \in \mathbb{R}^{n\times n}$ seja não simétrica não podemos aplicar CG ao sistema linear $Ax = b$. Porém, podemos transformar o problema em um equivalente, a saber, $AA^Ty = b$, $x = A^Ty$ e aplicar CG a esse sistema. Essa abordagem sofre dos mesmos problemas que o CGLS.

□

Na literatura encontra-se uma variante desse algoritmo, onde a recorrência é sobre o resíduo das equações normais $s = A^T(b-Ax)$ e não sobre $r = b-Ax$. A seguir, apresentamos esse algoritmo.

Algoritmo 19 Método dos Gradientes Conjugados: CGLS (instável)

```
function x_*=CGLS(A, b, x_0)
    k = 0;  r_0 = b − Ax_0;  p_0 = s_0 = A^T r_0;  γ_0 = ||s_0||_2^2;
    while γ_k > 0 do
        q_k = Ap_k;
        μ_k = γ_k/||q_k||_2^2;
        x_{k+1} = x_k + μ_k p_k;
        s_{k+1} = s_k − μ_k(A^T q_k);
        γ_{k+1} = ||s_{k+1}||_2^2;
        τ_k = γ_{k+1}/γ_k;
        p_{k+1} = s_{k+1} + τ_k p_k;
        k = k + 1;
    end while
    x_* = x_k;
end function
```

Björck demonstra que o Algoritmo 19 é instável. Assuma $x_0 = 0$, observe que não há referência sobre b no algoritmo, exceto nas condições iniciais. Para $A^T b$ temos[4],

$$|\operatorname{fl}(A^T b) - A^T b| \leqslant \gamma_m |A^T||b|, \quad \gamma_m = \frac{m\mathbf{u}}{1 - m\mathbf{u}},$$

que é uma boa aproximação, como esperado. A solução perturbada correspondente a $c = \operatorname{fl}(A^T b)$ satisfaz

$$A^T A(x + \delta x) = A^T b + \delta c, \quad |\delta c| \leqslant \gamma_m |A^T||b|.$$

Portanto, $\delta x = (A^T A)^{-1}\delta c$, obtendo uma limitante componente a componente. Em termos de normas, temos

$$\frac{\|\delta x\|}{\|x\|} \leqslant \gamma_m \|(A^T A)^{-1}\| \frac{\|A\|\|b\|}{\|x\|} = \gamma_m \kappa^2(A) \left(\frac{\|b\|}{\|A\|\|x\|} \right).$$

Sem a referência a b na fase da recorrência os erros de arredondamento não são compensados.

A melhoria da performance do CGLS pode ser feita através de precondicionadores ou através de alguma outra técnica de abordagem do problema, tal como LSQR.

Exemplos envolvendo o CGLS são mostrados mais a frente quando comparado a outros métodos numéricos, tais como LSQR, LSMR e PCGLS.

4.3 SYMMLQ e MINRES

Como acabamos de ver, o método dos gradientes conjugados é uma excelente opção para sistemas lineares definidos positivos, mas para a resolução das

[4] Usamos a notação de [66], onde $\operatorname{fl}(x)$ é a representação de x em ponto flutuante e $\mathbf{u}$ é a precisão da máquina.

equações normais apresenta alguns problemas. Assim, temos de encontrar alternativas para esse caso.

Há outros métodos iterativos sobre espaços de Krylov que têm, como hipótese, a simetria da matriz em questão, e eles podem ser aplicados em sistemas lineares definidos positivos. Entre esses métodos dois se destacam, ambos desenvolvidos por Paige e Saunders [108] e chamados de SYMMLQ e MINRES. Esses métodos buscam ser mais eficientes que o CGLS e são baseados nos vetores de Lanczos.

Considere o sistema linear

$$r + Ax = b, \quad Ar = 0 \tag{4.3.45}$$

onde $A \in \mathbb{R}^{n\times n}$ é simétrica e pode ser indefinida ou singular. A ideia dos métodos SYMMLQ e MINRES é determinar uma sequência $x = V_k y$. Nos interessa soluções $x_k = V_k y_k$ que fornecem os valores estacionários para

$$f_k(y) = (AV_k y - b)^T B(AV_k y - b), \tag{4.3.46}$$

onde B é alguma matriz simétrica. Note que se B for definida positiva, $f_k(y)$ é a norma-B ao quadrado do resíduo e a matriz B foi definida para fins teóricos, não é utilizada explicitamente. O ponto estacionário de f_k é a solução do sistema linear

$$\nabla f_k(y) = 0 \Rightarrow 2V_k^T AB(AV_k y_k - b) = 0 \Rightarrow V_k^T ABAV_k y_k = V_k^T ABb, \tag{4.3.47}$$

ou seja, quando

$$V_k^T ABr_k = 0, \;\; r_k = b - Ax_k.$$

Os métodos SYMMLQ e MINRES tentam resolver (4.3.47). Como a segunda derivada é $2V_k^T ABAV_k$, segue que se ABA é definida positiva, então há um único y que minimiza f_k. Se ABA é semidefinida positiva, então y minimiza f_k, mas não é único. Se ABA é indefinida, y é apenas um ponto estacionário. Duas possíveis escolhas para B são $B = A^{-1}$ ou $B = I$.

Caso 1. Tomando $B = A^{-1}$, vamos considerar na verdade a pseudoinversa de Moore-Penrose, ou seja, $B = A^\dagger$. De (4.3.45), obtemos

$$AA^\dagger b = AA^\dagger r + AA^\dagger Ax = Ax,$$

e de (4.3.47)

$$V_k^T AV_k y_k = V_k^T Ax_k = V_k^T b - V_k^T r_k,$$

que não pode resolvida diretamente, a menos que saibamos o valor de $V_k^T r_k$. Vamos trabalhar apenas com os casos $r_k = 0$ ou $v_i \in \mathrm{Im}(A)$, $i = 1, \ldots, k$ e, portanto,

$$V_k^T AV_k y_k = V_k^T b, \quad x_k = V_k y_k \tag{4.3.48}$$

fornece o ponto estacionário de (4.3.46). Se V_k gera $\mathrm{Im}(A)$, então temos a solução de quadrados mínimos de menor norma.

Caso 2. Tomando $B = I$ podemos minimizar $\|r_k\|_2$ resolvendo

$$V_k^T A^2 V_k u_k = V_k^T Ab, \quad x_k = V_k^T u_k. \tag{4.3.49}$$

Se V_k gera $\mathrm{Im}(A)$, então x_k é a solução de quadrados mínimos de mínima norma de (4.3.45). O método baseado em (4.3.49) é chamado de MINRES (*minimum residual*).

Por outro lado, o processo de Lanczos [80] dado pelo Algoritmo 13 (pág. 105), resulta em uma matriz de Hessenberg simétrica, ou seja, uma matriz tridiagonal, que chamaremos de T_k. Uma efetiva variante computacional do processo de Lanczos [107] é

$$\beta_{j+1} v_{j+1} = Av_j - \alpha_j v_j - \beta_j v_{j-1}, \quad \alpha_j = v_j^T A v_j,$$

com $\beta_{j+1} \geqslant 0$ escolhido de forma que v_{j+1} seja unitário. O processo de Lanczos em sua k-ésima iteração pode ser resumido da seguinte forma

$$\begin{cases} v_1 = b/\beta_1, \quad \beta_1 := \|b\|_2, \\ AV_k = V_k T_k + \beta_{k+1} v_{k+1} e_k^T, \\ V_k^T V_k = I, \quad V_k^T v_{k+1} = 0, \end{cases} \tag{4.3.50}$$

onde

$$T_k = \begin{bmatrix} \alpha_1 & \beta_2 & 0 & \cdots & 0 \\ \beta_2 & \alpha_2 & \beta_3 & \cdots & 0 \\ 0 & \ddots & \ddots & \ddots & \vdots \\ \vdots & \vdots & \beta_{k-1} & \alpha_{k-1} & \beta_k \\ 0 & 0 & \cdots & \beta_k & \alpha_k \end{bmatrix}.$$

O processo termina quando $\beta_{k_*} = 0$. Portanto, podemos assumir que todos β_k são não nulos e, de (4.3.50), obtemos

$$V_k^T A V_k = T_k, \quad V_k^T b = \beta_1 e_1.$$

Com esses vetores no Caso 1, a equação (4.3.48) se torna

$$T_k y_k = \beta_1 e_1, \quad x_k = V_k y_k. \tag{4.3.51}$$

A partir dessa equação, podemos determinar x_k através do método de gradientes conjugados. Porém, estamos interessados em trabalhar com matrizes simétricas não necessariamente definidas positivas.

Considere a fatoração LQ da matriz T_k,

$$T_k = \overline{L}_k Q_k, \quad Q_k^T Q_k = I, \tag{4.3.52}$$

com $\overline{L}_k$ uma matriz triangular inferior. As quantidades com barra podem ser determinadas recursivamente e, como em (4.3.51), y_k não precisa ser computado. Para isso, defina

$$\overline{W}_k = [w_1 \,|\, w_2 \,|\, \cdots \,|\, w_{k-1} \,|\, \overline{w}_k] = V_k Q_k^T \tag{4.3.53}$$

e

$$\overline{z}_k = (\zeta_1, \ldots, \zeta_{k-1}, \overline{\zeta}_k)^T = Q_k y_k. \tag{4.3.54}$$

Então de (4.3.51) temos

$$\overline{L}_k \overline{z}_k = \beta_1 e_1, \quad x_k = \overline{W}_k \overline{z}_k. \tag{4.3.55}$$

Matematicamente, essa solução é a mesma que a de gradientes conjugados, porém sua fatoração é mais estável para T_k's indefinidos, ao contrário do método dos gradientes conjugados que utilizam a fatoração LDL^T.

A fatoração LQ em (4.3.52) é determinada através de rotações de Givens. Assim,

$$T_k Q_{1,2} \cdots Q_{k-1,k} = T_k Q_k^T = \overline{L}_k = \begin{bmatrix} \gamma_1 & & & & \\ \delta_2 & \gamma_2 & & & \\ \epsilon_3 & \delta_3 & \gamma_3 & & \\ & \ddots & \ddots & \ddots & \\ & & \epsilon_k & \delta_k & \overline{\gamma}_k \end{bmatrix}, \tag{4.3.56}$$

com $Q_{k,k+1}$ sendo a matriz que difere da matriz identidade apenas nos elementos

$$\begin{aligned} q_{k,k} &= -q_{k+1,k+1} = c_k = \cos\theta_k, \\ q_{k,k+1} &= q_{k+1,k} = s_k = \operatorname{sen}\theta_k. \end{aligned}$$

Ademais,

$$\gamma_k = (\overline{\gamma}_k^2 + \beta_{k+1}^2)^{1/2}, \quad c_k = \overline{\gamma}_k/\gamma_k, \quad s_k = \beta_{k+1}/\gamma_k. \tag{4.3.57}$$

Na discussão a seguir, por sugestão de Paige e Saunders [108, pág. 622], vamos utilizar L_k para denotar $\overline{L}_k$ e $\overline{\gamma}_k$ sendo substituído por γ_k. Da mesma forma, $z_k = (\zeta_1, \ldots, \zeta_{k-1}, \zeta_k)^T$ e $W_k = [w_1 \,|\, w_2 \,|\, \cdots \,|\, w_{k-1} \,|\, w_k]$ e, portanto

$$L_k z_k = \beta_1 e_1.$$

De (4.3.55) e (4.3.57)

$$\zeta_k = \overline{\gamma}_k \overline{\zeta}_k = c_k \overline{\zeta}_k.$$

Os autores afirmam que essa mudança de variáveis fornece melhores resultados numéricos. De (4.3.53) e de $Q_{k,k+1}$,

$$\begin{cases} w_k = c_k \overline{w}_k + s_k v_{k+1}, \\ \\ \overline{w}_{k+1} = s_k \overline{w}_k - c_k v_{k+1}, \end{cases}$$

com $\overline{w}_1 := v_1$. Combinando as equações (4.3.52) e (4.3.53) obtemos

$$[\overline{L}_k^T, \overline{W}_k^T] = Q_k[T_k, V_k^T].$$

Essa relação mostra que $\overline{L}_k$ e $\overline{W}_k$ são obtidos transformando T_k e V_k^T à forma Hessenberg superior.

O processo apresentado nas equações (4.3.52) a (4.3.55) não deve ser implementado diretamente, pois geraria o cálculo de x_k a cada passo. Como a cada iteração a parte líder de x_k permanece a mesma, basta atualizá-lo iterativamente. Assim,

$$x_k^L = W_k z_k = x_{k-1}^L + \zeta_k w_k. \tag{4.3.58}$$

De (4.3.55) e (4.3.58)

$$x_{k+1} = x_k^L + \overline{\zeta}_{k+1}\overline{w}_{k+1}.$$

Esse método é chamado de SYMMLQ.

Para definir um critério de parada vamos monitorar o resíduo $r_k = b - Ax_k$, porém determinar $\|r_k\|_2$ a cada iteração é desnecessário. De (4.3.50) e (4.3.51), obtemos

$$\begin{aligned} r_k &= b - Ax_k = \beta_1 v_1 - AV_k y_k \\ &= \beta_1 v_1 - V_k T_k y_k - \beta_{k+1} v_{k+1} e_k^T y_k \\ &= -\beta_{k+1}\eta_{kk} v_{k+1}, \end{aligned}$$

onde η_{kk} é o k-ésimo elemento de y_k. O vetor y_k não está disponível nos cálculos computacionais, mas há como determiná-lo a partir dos parâmetros do algoritmo. Com efeito, como $T_k = T_k^T = Q_k^T \overline{L}_k^T$, a equação (4.3.51) fornece

$$\overline{L}_k^T y_k = \beta_1 Q_k e_1.$$

Dos últimos elementos dessa igualdadede,

$$\overline{\gamma_k}\eta_{kk} = \beta_1 s_1 s_2 \cdots s_{k-1}$$

e, portanto, de (4.3.57)

$$r_k = -(\beta_1 s_1 s_2 \cdots s_k / c_k) v_{k-1}.$$

Logo,

$$\|r_k\|_2 = |\beta_1 s_1 s_2 \cdots s_k / c_k|.$$

Paige e Saunders [108, pág. 626] reportam que o SYMMLQ fornece, essencialmente, os mesmos resultados que o método de gradientes conjugados, porém recomendam o uso de CG, para matrizes definidas positivas, por ser mais eficiente.

Passemos, agora, para a dedução do método MINRES. De (4.3.50) temos

$$V_k^T A^2 V_k = T_k^2 + \beta_{k+1}^2 e_k e_k^T, \tag{4.3.59}$$

$$V_k Ab = \beta_1 V_k^T Ab_1 = \beta_1 T_k e_k, \tag{4.3.60}$$

e note que a matriz em (4.3.59) é pentadiagonal e ao menos semidefinida positiva. Portanto podemos utilizar a decomposição de Cholesky,

$$T_k^2 + \beta_{k+1} e_k e_k^T = \overline{L}_k \overline{L}_k^T + \beta_{k+1} e_k e_k^T = L_k L_k^T,$$

ou seja, o fator de Cholesky vem diretamente de T_k. De (4.3.49) temos

$$L_k L_k^T u_k = \beta_1 \overline{L}_k Q_k e_1 \tag{4.3.61}$$

e, de (4.3.56) e (4.3.57),

$$\overline{L}_k = L_k D_k, \;\; D_k = diag(1, 1 \ldots, 1, c_k).$$

Logo, L_k é não singular. A equação (4.3.61) fornece

$$L_k^T u_k = \beta_1 D_k Q_k e_1 = (\tau_1, \ldots, \tau_k)^T := t_k, \tag{4.3.62}$$

$$\tau_1 := \beta_1 c_1, \;\; \tau_i = \beta_1 s_1 s_2 \cdots s_{i-1} c_i, \; i = 2, \ldots, k \tag{4.3.63}$$

e, novamente, não precisamos determinar u_k pois, denotando

$$M_k := V_k L_k^{-T},$$

temos

$$x_k = V_k u_k = V_k L_k^{-T} L_k^T u_k = M_k t_k.$$

O método MINRES é recomendado para matrizes simétricas indefinidas de grande porte. Existem outras alternativas mais eficientes que o MINRES, uma delas é o método LSMR que veremos mais à frente.

4.4 Bidiagonalização de Golub-Kahan

Seja $A \in \mathbb{R}^{m \times n}$ com $m \geqslant n$, que não é uma restrição, pois, caso contrário, aplicamos o processo sobre A^T. A ideia da bidiagonalização de Golub-Kahan [52] é utilizar, em seu formato original, matrizes de Householder para zerar os correspondentes termos, por linhas e por colunas, a cada iteração para obter-se uma matriz bidiagonal ao final do processo. Assim, vamos construir uma sequência de matrizes $A = A^{(1)}, A^{(2)}, \ldots, A^{(n-1)}$ de forma que

$$A^{(k+1)} = Q_k A^{(k)} P_k,$$

onde P_k e Q_k são matrizes de Householder. Como exemplo, após o primeiro passo da bidiagonalização, temos

$$A^{(2)} = Q_1 A P_1 \begin{bmatrix} q_1 & e_2 & 0 & \cdots & 0 \\ 0 & a_{22}^{(1)} & a_{23}^{(1)} & \cdots & a_{2n}^{(1)} \\ 0 & a_{32}^{(1)} & a_{33}^{(1)} & \cdots & a_{3n}^{(1)} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & a_{m2}^{(1)} & a_{m3}^{(1)} & \cdots & a_{mn}^{(1)} \end{bmatrix}.$$

A matriz Q_1 foi determinada para zerar os $n-1$ elementos da primeira coluna e a matriz P_1 para zerar os $n-2$ elementos da primeira linha de A. Ao final, após $n-1$ iterações, temos uma fatoração da forma

$$A = U \begin{bmatrix} B \\ 0 \end{bmatrix} V^T,$$

onde $U = Q_n \cdots Q_1$, $V^T = P_1 \cdots P_{n-2}$ e

$$B = \begin{bmatrix} q_1 & e_2 & 0 & \cdots & 0 \\ 0 & q_2 & e_3 & \cdots & 0 \\ 0 & 0 & \ddots & \ddots & \vdots \\ \vdots & \vdots & \ddots & q_{n-1} & e_n \\ 0 & 0 & \cdots & 0 & q_n \end{bmatrix}.$$

O custo da redução à forma bidiagonal é $\mathcal{O}(n^2 m)$ flops. Às vezes trabalhamos com matrizes triangulares que necessitam ser transformadas em matrizes bidiagonais. Assim, uma implementação da bidiagonalização de Golub-Kahan é através de rotações de Givens. Uma extensa análise sobre o assunto pode ser encontrada em [19, 20].

Nosso objetivo, na próxima seção, é estudar o método LSQR de Paige e Saunders que, por sua vez, está baseado em um processo de bidiagonalização. Nessa seção seguiremos a abordagem de [12, pág. 303] e [11, pág. 661]. O processo de bidiagonalização que vamos detalhar é o processo de bidiagonalização de Golub-Kahan que foi aplicado no cálculo de valores singulares e seus respectivos vetores singulares.

Seja $A \in \mathbb{R}^{m \times n}$ com $m \geqslant n$, que admite a seguinte fatoração

$$A = U \begin{bmatrix} B \\ 0 \end{bmatrix} V^T, \tag{4.4.64}$$

onde $U^T U = I_m$, $V^T V = I_n$ e

$$B = \begin{bmatrix} \alpha_1 & \beta_1 & 0 & \cdots & 0 \\ 0 & \alpha_2 & \beta_2 & \cdots & 0 \\ 0 & 0 & \ddots & \ddots & \vdots \\ \vdots & \vdots & \ddots & \alpha_{n-1} & \beta_{n-1} \\ 0 & 0 & \cdots & 0 & \alpha_n \end{bmatrix}.$$

As matrizes U e V são determinadas por reflexões de Householder ou rotações de Givens, e suas colunas são geradas sequencialmente, assim como no processo de Lanczos. Suponha que $U_1 = [u_1 \mid \cdots \mid u_n]$. De (4.4.64), temos

$$\left\{ \begin{array}{rcl} AV & = & U_1 B, \\ \\ A^T U_1 & = & V B^T. \end{array} \right. \tag{4.4.65}$$

Igualando as colunas de ambas as identidades, obtemos

$$Av_j = \alpha_j u_j + \beta_{j-1} u_{j-1}, \tag{4.4.66}$$

$$A^T u_j = \alpha_j v_j + \beta_j v_{j+1}, \tag{4.4.67}$$

onde $j = 1, \ldots, n$ e $\beta_0 u_0 = \beta_n v_{n+1} = 0$. Resolvendo esse sistema linear para u_j e v_{j+1}, lembrando que $\|u_j\|_2 = \|v_j\|_2 = 1$, obtemos

$$r_j = Av_j - \beta_{j-1} u_{j-1}, \;\; \alpha_j = \|r_j\|_2, \;\; u_j = r_j/\alpha_j, \tag{4.4.68}$$

$$p_j = A^T u_j - \alpha_j v_j, \;\; \beta_j = \|p_j\|_2, \;\; v_{j+1} = p_j/\beta_j, \tag{4.4.69}$$

onde $j = 1, \ldots, n$. A seguir, apresentamos um algoritmo para a bidiagonalização de Golub-Kahan (superior).

Algoritmo 20 Bidiagonalização de Golub-Kahan (superior)

```
function [U, B, S]=BDLS(A)
    [m, n] = size(A);
    v_1 = e_1; %e_1 primeira coluna da matriz identidade
    β_0 u_0 = β_n v_{n+1} = 0;
    for j=1:n do
        r_j = A v_j − β_{j−1} u_{j−1};
        α_j = ‖r_j‖_2;
        u_j = r_j/α_j;
        p_j = A^T u_j − α_j v_j;
        β_j = ‖p_j‖_2;
        v_{j+1} = p_j/β_j;
    end for
end function
```

Apesar de apresentarmos o algoritmo com um *loop*, o algoritmo acima termina quando $\alpha_j = 0$ ou $\beta_j = 0$ (por quê?)

Das relações de recorrência (4.4.68) e (4.4.69) vemos que, para todo $j = 1, \ldots, n$, vale

$$\left\{ \begin{array}{l} v_j \in \mathcal{K}_j(A^T A, v_1), \\ \\ u_j \in \mathcal{K}_j(AA^T, Av_1), \end{array} \right.$$

Podemos também desenvolver um processo de bidiagonalização, de forma que o resultado seja uma matriz bidiagonal inferior, a saber,

$$B = \begin{bmatrix} \alpha_1 & 0 & 0 & \cdots & 0 \\ \beta_2 & \alpha_2 & 0 & \cdots & 0 \\ 0 & \beta_3 & \ddots & \ddots & \vdots \\ \vdots & \vdots & \ddots & \alpha_{n-1} & 0 \\ 0 & 0 & \cdots & \beta_n & \alpha_n \\ 0 & 0 & \cdots & 0 & \beta_{n+1} \end{bmatrix}.$$

Novamente, igualando as colunas em (4.4.65), obtemos

$$A^T u_j = \beta_j v_{j-1} + \alpha_j v_j, \tag{4.4.70}$$

$$A v_j = \alpha_j u_j + \beta_{j+1} u_{j+1}, \tag{4.4.71}$$

e, isolando v_j e u_{j+1}, obtemos

$$r_j = A^T u_j - \beta_j v_{j-1}, \;\; \alpha_j = \|r_j\|_2, \;\; v_j = r_j/\alpha_j, \tag{4.4.72}$$

$$p_j = A v_j - \alpha_j u_j, \;\; \beta_{j+1} = \|p_j\|_2, \;\; u_{j+1} = p_j/\beta_{j+1}. \tag{4.4.73}$$

Um possível algoritmo é dado a seguir.

Algoritmo 21 Bidiagonalização de Golub-Kahan (inferior)

```
function [U, B, S]=BDLI(A)
    [m, n] = size(A);
    v_1 = e_1; %e_1 primeira coluna da matriz identidade
    β_1 v_0 = α_{n+1} u_{n+1} = 0;
    for j=1:n do
        r_j = A^T u_j − β_j v_{j−1};
        α_j = ||r_j||_2;
        v_j = r_j/α_j;
        p_j = A v_j − α_j u_j;
        β_j = ||p_j||_2;
        u_{j+1} = p_j/β_{j+1};
    end for
end function
```

Das relações de recorrência (4.4.72) e (4.4.73) vemos que, para todo $j = 1, \ldots, n$, vale

$$\begin{cases} u_j \in \mathcal{K}_j(AA^T, u_1), \\ \\ u_j \in \mathcal{K}_j(A^T A, A^T v_1). \end{cases}$$

Björck discute de forma objetiva a relação entre a bidiagonalização de Golub-Kahan e o processo de Lanczos aplicado às matrizes AA^T e $A^T A$. Para uma discussão mais aprofundada recomendamos [28]. Para uma versão da bidiagonalização de Golub-Kahan por blocos recomendamos [55].

4.5 LSQR

Nesta seção vamos apresentar uma alternativa para o método dos gradientes conjugados, chamado de método LSQR. Esse método foi desenvolvido por Paige e Saunders [110], e a discussão a seguir é baseada em [12, 21].

Gostaríamos de encontrar uma sequência de aproximações para o problema de quadrados mínimos dado,

$$\min_{x\in\mathbb{R}^n} \|Ax-b\|_2, \;\; A\in\mathbb{R}^{m\times n}, \;\; m \geqslant n.$$

Para isso utilizamos as recorrências (4.4.72) e (4.4.73). Começando com b, tome

$$\beta_1 u_1 = b, \quad \alpha_1 v_1 = A^T u_1 \tag{4.5.74}$$

e para $j = 1, 2, \ldots$

$$\beta_{j+1} u_{j+1} = A v_j - \alpha_j u_j, \tag{4.5.75}$$

$$\alpha_{j+1} v_{j+1} = A^T u_{j+1} - \beta_{j+1} v_j, \tag{4.5.76}$$

onde $\alpha_{j+1}, \beta_{j+1} \geqslant 0$ são determinados de forma que $\|u_{j+1}\|_2 = \|v_{j+1}\|_2 = 1$. Após k passos temos computado V_k cujas colunas são os vetores v_j, $j = 1, \ldots, k$, U_{k+1}, cujas colunas são os vetores u_j, $j = 1, \ldots, k+1$, e

$$B_k = \begin{bmatrix} \alpha_1 & 0 & 0 & \cdots & 0 \\ \beta_2 & \alpha_2 & 0 & \cdots & 0 \\ 0 & \beta_3 & \ddots & \ddots & \vdots \\ \vdots & \vdots & \ddots & \alpha_{n-1} & 0 \\ 0 & 0 & \cdots & \beta_k & \alpha_k \\ 0 & 0 & \cdots & 0 & \beta_{k+1} \end{bmatrix} \in \mathbb{R}^{(k+1)\times k}. \tag{4.5.77}$$

As relações de recorrência de (4.5.74) até (4.5.77) podem ser reescritas matricialmente da seguinte forma

$$\beta_1 U_{k+1} e_1 = b, \tag{4.5.78}$$

$$A V_k = U_{k+1} B_k, \tag{4.5.79}$$

$$A^T U_{k+1} = V_k B_k^T + \alpha_{k+1} v_{k+1} e_{k+1}^T. \tag{4.5.80}$$

A solução que procuramos é $x_k \in \mathcal{K}(A^T A, A^T b) = span(V_{k+1})$, assim

$$x_k = V_k y_k. \tag{4.5.81}$$

Multiplicando (4.5.79) por y_k, temos

$$A x_k = A V_k y_k = U_{k+1} B_k y_k.$$

Por (4.5.78)

$$r_k = b - Ax_k = U_{k+1}t_{k+1}, \tag{4.5.82}$$

$$t_{k+1} = \beta_1 e_1 - B_k y_k.$$

Veja que $\|b - Ax_k\|_2$ é minimizado sobre $span(V_k)$ se y_k é a solução do problema de quadrados mínimos

$$\min_{y_k} \|B_k y_k - \beta_1 e_1\|_2. \tag{4.5.83}$$

Essa é a base do método LSQR: transformar um problema de minimização em outro mais simples. Vejamos agora como encontrar a solução do problema (4.5.83), e vale mencionar que o método LSQR é matematicamente equivalente ao CGLS [12, pág. 307].

O problema (4.5.83) é resolvido pela fatoração QR de B_k,

$$Q_k B_k = \begin{bmatrix} R_k \\ 0 \end{bmatrix}, \quad Q_k(\beta_1 e_1) = \begin{bmatrix} f_k \\ \overline{\phi}_{k+1} \end{bmatrix}, \tag{4.5.84}$$

onde R_k é uma matriz bidiagonal superior,

$$R_k = \begin{bmatrix} \rho_1 & \theta_1 & 0 & \cdots & 0 \\ 0 & \rho_2 & \theta_2 & \cdots & 0 \\ 0 & 0 & \ddots & \ddots & \vdots \\ \vdots & \vdots & \ddots & \rho_{n-1} & \theta_{n-1} \\ 0 & 0 & \cdots & 0 & \rho_n \end{bmatrix} \quad \text{e} \quad f_k = \begin{bmatrix} \phi_1 \\ \phi_2 \\ \vdots \\ \phi_{k-1} \\ \phi_k \end{bmatrix}.$$

A matriz Q_k é o produto de rotações de Givens $Q_k = G_{k,k+1}G_{k-1,k}\cdots, G_{1,2}$ de forma a eliminar a subdiagonal $\beta_2, \ldots, \beta_{k+1}$ de B_k. Para obter a solução y_k e o resíduo t_{k+1} resolvemos os sistemas lineares

$$R_k y_k = f_k, \tag{4.5.85}$$

$$t_{k+1} = Q_k^T \begin{bmatrix} 0 \\ \overline{\phi}_{k+1} \end{bmatrix}. \tag{4.5.86}$$

Observe que não podemos calcular a fatoração QR (4.5.84) a cada iteração, pois o custo computacional do algoritmo LSQR seria altíssimo, inviabilizando sua utilização e, por isso, empregamos uma relação de recorrência para realizar o trabalho. Assuma que tenhamos a fatoração de B_{k-1}. Assim, no passo k, a k-ésima coluna é adicionada, $Q_k = G_{k,k+1}Q_{k-1}$ e

$$G_{k,k+1}G_{k-1,k} \begin{bmatrix} 0 \\ \alpha_k \\ \beta_{k+1} \end{bmatrix} = \begin{bmatrix} \theta_k \\ \rho_k \\ 0 \end{bmatrix} \quad \text{e} \quad G_{k,k+1} \begin{bmatrix} \overline{\phi}_k \\ 0 \end{bmatrix} = \begin{bmatrix} \phi_k \\ \overline{\phi}_{k+1} \end{bmatrix}.$$

Note que as rotações $G_{k-2,k-1}, \ldots, G_{1,2}$ não são utilizadas, pois não afetam a k-ésima coluna.

Como $x_k = V_k y_k$, precisaríamos salvar os vetores $v_1, \ldots, v_k$, mas isso pode ser evitado. Combinando (4.5.81) com (4.5.85), obtemos

$$x_k = (V_k R_k^{-1}) f_k := Z_k f_k,$$

e a determinação de Z_k é feita através do sistema triangular inferior $R_k^T Z_k^T = V_k^T$. Assim, as colunas de Z_k são determinadas por substituição direta,

$$z_k = \frac{1}{\rho_k}(v_k - \theta_k z_{k-1}) \quad \text{e} \quad x_k = x_{k-1} + \phi_k z_k. \tag{4.5.87}$$

A seguir, apresentamos o algoritmo do método LSQR. Matematicamente, LSQR gera a mesma sequência de aproximações x_k que CGLS [12, p. 307], entretanto, LSQR é mais confiável em situações extremas como quando muitas iterações devem ser feitas até atingir a convergência, ou quando A é mal condicionada [110, pág. 70].

Algoritmo 22 Método LSQR

function x=LSQR(A, b)
  $x_0 = 0$;
  $\overline{u}_1 = b$; $\beta_1 = \|\overline{u}_1\|_2$; $u_1 = \overline{u}_1/\beta_1$;
  $\overline{v}_1 = A^T u_1$; $\alpha_1 = \|\overline{v}_1\|_2$; $v_1 = \overline{v}_1/\alpha_1$;
  $w_1 = v_1$;
  $\overline{\phi}_1 = \beta_1$;
  $\overline{\rho}_1 = \alpha_1$;
  while o critério de parada não é satisfeito (discussão abaixo) **do**
    $\overline{u}_{i+1} = Av_i - \alpha_i u_i$; $\beta_{i+1} = \|\overline{u}_{i+1}\|_2$; $u_{i+1} = \overline{u}_{i+1}/\beta_{i+1}$;
    $\overline{v}_{i+1} = A^T u_{i+1} - \beta_{i+1} v_i$; $\alpha_{i+1} = \|\overline{v}_{i+1}\|_2$; $v_{i+1} = \overline{v}_{i+1}/\alpha_{i+1}$;
    $\rho_i = (\overline{\rho}_i^2 + \beta_{i+1}^2)^{1/2}$;
    $c_i = \overline{\rho}_i/\rho_i$; $s_i = \beta_{i+1}/\rho_i$;
    $\theta_{i+1} = s_i \alpha_{i+1}$; $\overline{\rho}_{i+1} = c_i \alpha_{i+1}$;
    $\phi_i = c_i \overline{\phi}_i$; $\overline{\phi}_{i+1} = -s_i \overline{\phi}_i$;
    $x_i = x_{i-1} + (\phi_i/\rho_i) w_i$;
    $w_{i+1} = v_{i+1} - (\theta_{i+1}/\rho_i) w_i$;
    $i = i + 1$;
  end while
  $x = x_{i-1}$;
end function

A variável $w = \rho_k z_k$ foi introduzida para poupar um pouco de esforço computacional em (4.5.87). A implementação do método LSQR em Fortran é dada em [109].

Björck [12, pág. 309] analisa o custo computacional de LSQR que requer $3m + 5n$ multiplicações e o armazenamento de dois vetores m-dimensionais (u, Av) e três vetores n-dimensionais (x, v, w). Já o CGLS requer $2m + 3n$ multiplicações, dois vetores m-dimensionais e dois vetores n-dimensionais.

Antes de discutirmos os critério de parada para o método LSQR, vamos apresentar estimativas para $\|r_k\|_2$, $\|A^T r_k\|_2$, $\|x_k\|_2$, $\|A\|_F$ e $\kappa_F(A)$ que dependam das quantidades que já calculamos no Algoritmo 22, ou seja, com

um custo computacional menor do que calcular as quantidades utilizando suas definições formais.

Estimativa de $\|r_k\|_2$: De (4.5.82) e (4.5.86), obtemos

$$r_k = \overline{\phi}_{k+1} U_{k+1} Q_k^T e_{k+1} \tag{4.5.88}$$

e, como U e Q têm normas unitárias, temos

$$\|r_k\|_2 = \beta_1 s_k s_{k-1} \cdots s_1.$$

O método LSQR é não usual por não estimar $\|r_k\|_2$ durante a execução do código, porém sua estimativa é possível e praticamente de graça. Ademais o produto dos senos decai monotonicamente. Por fim, se $Ax = b$ for compatível, $\|r_k\|_2 \to 0$.

Estimativa de $\|A^T r_k\|_2$: Para problemas de quadrados mínimos a estimativa de $\|A^T r_k\|_2$ é importante, pois é o resíduo do sistema linear das equações normais. De (4.5.88), (4.5.80) e (4.5.84), obtemos

$$\begin{aligned} A^T r_k &= \overline{\phi}_{k+1}(V_k B_k^T + \alpha_{k+1} v_{k+1} e_{k+1}^T) Q_k^T e_{k+1} \\ &= \overline{\phi}_{k+1} V_k [R_k^T \ 0] e_{k+1} + \overline{\phi}_{k+1} \alpha_{k+1} (e_{k+1}^T Q_k^T e_{k+1}) v_k. \end{aligned}$$

O primeiro termo é nulo. Já o $k+1$-ésimo elemento da diagonal de Q_k é $-c_k$. Portanto,

$$A^T r_k = -(\overline{\phi}_{k+1} \alpha_{k+1} c_k) v_{k+1}$$

e, consequentemente,

$$\|A^T r_k\|_2 = \overline{\phi}_{k+1} \alpha_{k+1} |c_k|$$

lembrando que $V_k^T V_k = I$.

Estimativa de $\|x_k\|_2$: A matriz bidiagonal R_k pode ser transformada em uma matriz bidiagonal inferior pela fatoração ortogonal LQ,

$$R_k \overline{Q}_k^T = \overline{L_k} \tag{4.5.89}$$

onde Q_k é um produto de rotações de Givens. Definindo $\overline{z}_k$ a solução de

$$\overline{L_k} \overline{z_k} = f_k, \tag{4.5.90}$$

segue que,

$$x_k = (V_k R_k^{-1}) f_k = (V_k \overline{Q}_k^T) \overline{z}_k = \overline{W}_k \overline{z}_k$$

e, portanto,

$$\|x_k\|_2 = \|\overline{z}_k\|_2. \tag{4.5.91}$$

Pode parecer um custo alto a determinação de $\|x_k\|_2$, porém as partes líderes de $\overline{L}_k$, $\overline{Q}$, $\overline{W}_k$ e $\overline{z}_k$ não mudam com o avanço das iterações. A determinação

de $\|x_k\|_2$ custa 13 multiplicações por iteração que é negligenciável para n grande.

Estimativa de $\|A\|_F$ e $\kappa_F(A)$: De (4.5.76) $v_k \in \mathrm{Im}(A^T) = \mathrm{Ker}\,(A)^\perp = \mathrm{Ker}\,(A^TA)^\perp$ e, de (4.5.79), obtemos

$$B_k^T B_k = V_k^T A^T A V_k. \tag{4.5.92}$$

Pelo Teorema Min-Max de Courant[5]-Fischer,[6] os autovalores de $B_k^T B_k$ são limitados pelo menor e maior autovalor de A^TA. O mesmo vale para os valores singulares de B_k, que são limitados pelos valores singulares de A. Assim,

$$\|B_k\|_p \leqslant \|A\|_p, \quad p = 2, F.$$

Vamos utilizar $\|B_k\|_F$ como uma estimativa para $\|A\|_F$. Por outro lado, pelos mesmos argumentos apresentados acima e de (4.5.84),

$$B_k^T B_k = R_k^T R_k$$

e, portanto,

$$\|R_k^{-1}\|_p = \|B_k^\dagger\|_p \leqslant \|A^\dagger\|_p, \quad p = 2, F.$$

Como $Z_k = V_k R_k^{-1}$, então

$$1 \leqslant \|B_k\|_p \|Z_k\|_p \leqslant \|A\|_p \|A^\dagger\|_p = \kappa_p(A), \quad p = 2, F.$$

Vamos utilizar $\|B_k\|_F \|Z_k\|_F$ como estimativa de $\kappa_F(A)$. Note que, o cálculo das normas $\|B_k\|_F$ e $\|Z_k\|_F$ pode ser efetuado iterativamente via

$$\|B_k\|_F^2 = \|B_{k-1}\|_F^2 + \alpha_k^2 + \beta_{k+1}^2 \quad \text{e} \quad \|Z_k\|_F^2 = \|Z_{k-1}\|_F^2 + \|z_k\|_2^2, \tag{4.5.93}$$

cuja demonstração é deixada como exercício.

Para determinar o critério de parada para o método LSQR, Paige e Saunders [110] propõem três regras de parada, a saber, quando

S1. $\|r_k\|_2 \leqslant \mathrm{BTOL}\|b\|_2 + \mathrm{ATOL}\|A\|_F\|x_k\|_2$,

S2. $\dfrac{\|A^T r_k\|_2}{\|A\|_F \|r_k\|_2} \leqslant \mathrm{ATOL}$,

S3. $\kappa_F(A) \geqslant \mathrm{COLIM}$.

Os parâmetros ATOL, BTOL e COLIM são fornecidos pelo usuário. Ademais, a regra S1 é válida para sistemas compatíveis, S2 para sistemas incompatíveis e S3 para ambos. Os parâmetros ATOL e BTOL podem ser definidos em termo da acurácia do dado, isto é, se (A, b) é o dado fornecido e $(\tilde{A}, \tilde{b})$ são os dados reais (desconhecidos), então

$$ATOL = \frac{\|A - \tilde{A}\|_F}{\|A\|_F}.$$

[5] `https://mathshistory.st-andrews.ac.uk/Biographies/Courant/`
[6] `https://mathshistory.st-andrews.ac.uk/Biographies/Fischer/`

O mesmo vale para BTOL. A justificativa para os critérios de parada é encontrada em [110, 135].

Exemplo 4.3. *Para esse exemplo considere a matriz A como sendo o modelo WELL1850 do* Matrix Market[7] *e b um vetor de 1's.*

	CGLS	LSQR
# de iterações	438	440
$\|x_{k_*} - x_*\|_2$	$1.0349e-006$	$9.4378e-007$
$\|A(x_{k_*} - x_*)\|_2$	$1.2666e-007$	$1.1463e-007$

Tabela 4.1: Comparação entre CGLS e LSQR.

Utilizamos o critério de parada S1 com $ATOL = BTOL = 10^{-10}$. *Observe que tanto CGLS e LSQR apresentam o mesmo nível de aproximação, com aproximadamente o mesmo número de iterações.*

Em resumo, o método *LSQR* é equivalente ao *CGLS*, ou seja, quando aplicamos o método dos gradientes conjugados às equações normais. Uma característica importante do *LSQR* é que $\|r_k\|_2$ decresce monotonicamente.

4.6 LSMR

O método LSMR foi desenvolvido por Fong e Saunders [41] e é equivalente ao método MINRES aplicado às equações normais, tendo como característica principal o decrescimento monotônico de $\|A^T r_k\|_2$, embora também seja observado um decaimento monotônico de $\|r_k\|_2$. Assim, o método LSMR é indicado para problemas de grande porte cujo *solver* precisa ser encerrado antecipadamente.

Novamente, comecemos com o processo de bidiagonalização de Golub-Kahan. As identidades (4.5.79) e (4.5.80) no k-ésimo passo do processo de bidiagonalização são

$$AV_k = U_{k+1}B_k \quad \text{e} \quad A^T U_{k+1} = V_{k+1}L_{k+1}^T,$$

onde

$$B_k = \begin{bmatrix} \alpha_1 & 0 & 0 & \cdots & 0 \\ \beta_2 & \alpha_2 & 0 & \cdots & 0 \\ 0 & \beta_3 & \ddots & \ddots & \vdots \\ \vdots & \vdots & \ddots & \alpha_{n-1} & 0 \\ 0 & 0 & \cdots & \beta_k & \alpha_k \\ 0 & 0 & \cdots & 0 & \beta_{k+1} \end{bmatrix} \quad \text{e} \quad L_{k+1} = [B_k \ \alpha_{k+1}e_{k+1}].$$

[7] `https://math.nist.gov/MatrixMarket/data/Harwell-Boeing/lsq/well1033.html`

Como estamos interessados em aplicar essas ideias às equações normais, considere

$$\begin{aligned} A^T A V_k &= A^T U_{k+1} B_k = V_{k+1} L_{k+1} B_k = V_{k+1} \begin{bmatrix} B_k^T \\ \alpha_{k+1} e_{k+1}^T \end{bmatrix} B_k \\ &= V_{k+1} \begin{bmatrix} B_k^T B_k \\ \alpha_{k+1}\beta_{k+1} e_k^T \end{bmatrix} \end{aligned} \tag{4.6.94}$$

que é equivalente ao processo de Lanczos aplicado a $A^T A$ e $A^T b$.

As soluções de quadrados mínimos minimizam o resíduo $\|r\|_2$, com $r = b - Ax$. Por essa razão, em LSQR, a escolha de y_k em (4.5.81) foi a de minimizar o resíduo a cada passo de iteração. Note que,

$$r_k = b - AV_k yk = \beta_1 u_1 - U_{k+1} B_k y_k = U_{k+1}(\beta_1 e_1 - B_k y_k).$$

Assim, transformamos o problema $\min\limits_{x_k} \|Ax_k - b\|_2$ em $\min\limits_{y_k} \|\beta_1 e_1 - B_k y_k\|_2$.

Como já apresentado, o objetivo do LSMR é minimizar $\|A^T r_k\|_2$. Para tanto, defina $\overline{\beta}_k = \alpha_k \beta_k$. Logo, de (4.6.94),

$$\begin{aligned} A^T r_k &= A^T b - A^T A x_k = \beta_1 \alpha_1 v_1 - A^T A V_k y_k \\ &= \overline{\beta}_1 v_1 - V_{k+1} \begin{bmatrix} B_k^T B_k \\ \alpha_{k+1}\beta_{k+1} e_k^T \end{bmatrix} \\ &= V_{k+1} \left\{ \overline{\beta}_1 e_1 - \begin{bmatrix} B_k^T B_k \\ \overline{\beta}_{k+1} e_k^T \end{bmatrix} \right\}. \end{aligned}$$

Portanto, no método LSMR o problema $\min\limits_{x_k} \|Ax_k - b\|_2$ é resolvido através do subproblema

$$\min_{y_k} \|A^T r_k\|_2 = \min_{y_k} \left\| \overline{\beta}_1 e_1 - \begin{bmatrix} B_k^T B_k \\ \overline{\beta}_{k+1} e_k^T \end{bmatrix} y_k \right\|_2. \tag{4.6.95}$$

Diferentemente do método LSQR, o método LSMR é deduzido utilizando-se duas fatorações QR. A primeira é praticamente idêntica à fatoração aplicada em LSQR

$$Q_{k+1} B_k = \begin{bmatrix} R_k \\ 0 \end{bmatrix}, \quad \text{com} \quad R_k = \begin{bmatrix} \rho_1 & \theta_2 & & \\ & \rho_2 & \ddots & \\ & & \ddots & \theta_k \\ & & & \rho_k \end{bmatrix}. \tag{4.6.96}$$

Defina $t_k = R_k y_k$. Agora, resolvamos $R_k^T q_k = \overline{\beta}_{k+1} e_k$, cuja solução é

$$q_k = (\overline{\beta}_{k+1}/\rho_k) e_k := \varphi_k e_k,$$

onde $\rho_k = [R_k]_{kk}$. Diferentemente de LSQR, aplicamos uma segunda fatoração QR

$$\overline{Q}_{k+1} \begin{bmatrix} R_k^T & \overline{\beta}_1 e_1 \\ \varphi_k e_k & 0 \end{bmatrix} = \begin{bmatrix} \overline{R}_k & z_k \\ 0 & \overline{\zeta}_{k+1} \end{bmatrix}, \tag{4.6.97}$$

com

$$R_k = \begin{bmatrix} \overline{\rho_1} & \overline{\theta_2} & & \\ & \overline{\rho_2} & \ddots & \\ & & \ddots & \overline{\theta_k} \\ & & & \overline{\rho_k} \end{bmatrix}.$$

Combinando o resultado das duas fatorações com (4.6.95), obtemos

$$\begin{aligned} \min_{y_k} \|A^T r_k\|_2 &= \min_{y_k} \left\| \overline{\beta}_1 e_1 - \begin{bmatrix} R_k^T R_k \\ q_k^T R_k \end{bmatrix} y_k \right\|_2 \\ &= \min_{t_k} \left\| \overline{\beta}_1 e_1 - \begin{bmatrix} R_k^T \\ \varphi_k e_k^T \end{bmatrix} t_k \right\|_2 \\ &= \min_{t_k} \left\| \begin{bmatrix} z_k \\ \overline{\zeta}_{k+1} \end{bmatrix} - \begin{bmatrix} \overline{R}_k^T \\ 0 \end{bmatrix} t_k \right\|_2 . \end{aligned} \tag{4.6.98}$$

Portanto, o subproblema (4.6.95) é resolvido encontrando a solução de $\overline{R}_k t_k = z_k$, e não há a necessidade de determinarmos t_k, tampouco y_k. Podemos determinar x_k recursivamente, a partir de $x_k = V_k y_k$, $R_k y_k = t_k$ e $\overline{R}_k t_k = z_k$. Tome $x_0 = 0$ e considere $R_k^T W_k^T = V_k^T$ e $\overline{R}_k^T \overline{W}_k^T = W_k^T$. Assim,

$$x_k = W_k R_k y_k = W_k t_k = \overline{W}_k \overline{R}_k t_k = \overline{W}_k z_k = x_{k-1} + \zeta_k \overline{w}_k.$$

Continuando com a descrição do método LSMR vamos determinar relações de recorrência para W_k e $\overline{W}_k$. Para tanto, escrevamos

$$V_k = [v_1 \,|\, v_2 \,|\, \ldots \,|\, v_k], \qquad W_k = [w_1 \,|\, w_2 \,|\, \ldots \,|\, w_k],$$

$$\overline{W}_k = [\overline{w}_1 \,|\, \overline{w}_2 \,|\, \ldots \,|\, \overline{w}_k], \qquad z_k = (\zeta_1, \zeta_2, \ldots, \zeta_n)^T.$$

Assim como acontecia com o método LSQR, as partes líderes dessas quantidades permanecem constantes, e apenas incluímos termos na última entrada quando aumentamos a dimensão dos subespaços de Krylov. Ambas as fatorações QR propostas são feitas através de rotações de Givens, que são matrizes identidade exceto nas posições que queremos rotacionar. Em nosso caso, seja P_l a rotação de Givens que opera nas linhas e colunas l e $l+1$.

Dessa forma, $Q_{k+1} = P_k \cdots P_2 P_1$ e obtemos

$$Q_{k+1}B_{k+1} = Q_{k+1}\begin{bmatrix} B_k & \alpha_{k+1}e_{k+1} \\ 0 & \beta_{k+2} \end{bmatrix} = Q_{k+1}\begin{bmatrix} R_k & \theta_{k+1}e_k \\ 0 & \overline{\alpha}_{k+1} \\ 0 & \beta_{k+2} \end{bmatrix}$$

$$Q_{k+2}B_{k+1} = P_{k+1}\begin{bmatrix} R_k & \theta_{k+1}e_k \\ 0 & \overline{\alpha}_{k+1} \\ 0 & \beta_{k+2} \end{bmatrix} = \begin{bmatrix} R_k & \theta_{k+1}e_k \\ 0 & \rho_{k+1} \\ 0 & 0 \end{bmatrix}$$

e, portanto, $\theta_{k+1} = s_k\alpha_{k+1} = (\beta_{k+1}/\rho_k)\alpha_{k+1} = \overline{\beta}_{k+1}/\rho_k = \varphi_k$, ou seja, podemos utilizar θ_{k+1} em vez de φ_k.

Já da segunda fatoração QR

$$\overline{Q}_{k+1}\begin{bmatrix} R_k^T \\ \theta_{k+1}e_k^T \end{bmatrix} = \begin{bmatrix} \overline{R}_k \\ 0 \end{bmatrix},$$

$$\overline{Q}_{k+2}\begin{bmatrix} R_{k+1}^T \\ \theta_{k+2}e_{k+1}^T \end{bmatrix} = \overline{P}_{k+1}\begin{bmatrix} \overline{R}_k & \overline{\theta}_{k+1}e_k^T \\ 0 & \overline{c}_k\rho_{k+1} \\ 0 & \theta_{k+2} \end{bmatrix} = \begin{bmatrix} \overline{R}_k & \overline{\theta}_{k+1}e_k^T \\ 0 & \overline{\rho}_{k+1} \\ 0 & 0 \end{bmatrix}. \tag{4.6.99}$$

Considerando a última linha de $R_{k+1}^T W_{k+1}^T = V_{k+1}^T$ e $\overline{R}_{k+1}^T \overline{W}_{k+1}^T = W_{k+1}^T$, obtemos as relações de equivalência para w_{k+1} e $\overline{w}_{k+1}$

$$\begin{cases} \theta_{k+1}w_k^T + \rho_{k+1}w_{k+1}^T = v_{k+1}^T, \\ \\ \overline{\theta}_{k+1}\overline{w}_k^T + \overline{\rho}_{k+1}\overline{w}_{k+1}^T = w_{k+1}^T. \end{cases}$$

De forma a facilitar a compreensão de certos parâmetros que aparecem no algoritmo de LSMR temos

$$\begin{bmatrix} c_k & s_k \\ -s_k & c_k \end{bmatrix}\begin{bmatrix} \overline{\alpha}_k & 0 \\ \beta_{k+1} & \alpha_{k+1} \end{bmatrix} = \begin{bmatrix} \rho_k & \theta_{k+1} \\ 0 & \overline{\alpha}_{k+1} \end{bmatrix},$$

$$\begin{bmatrix} \overline{c}_k & \overline{s}_k \\ -\overline{s}_k & \overline{c}_k \end{bmatrix}\begin{bmatrix} \overline{c}_{k-1}\rho_k & 0 & \overline{\zeta}_k \\ \theta_{k+1} & \rho_{k+1} & 0 \end{bmatrix} = \begin{bmatrix} \overline{\rho}_k & \overline{\theta}_{k+1} & \overline{\zeta}_k \\ 0 & \overline{c}_k\rho_{k+1} & \overline{\zeta}_{k+1} \end{bmatrix}.$$

Por fim, o método corre mais eficientemente com a mudança de variáveis $h_k = \rho_k w_k$ e $\overline{h}_k = \rho_k\overline{\rho}_k\overline{w}_k$, e lembre que uma mudança de variáveis similar foi feita no método LSQR. A seguir, apresentamos um algoritmo para o LSMR e, assim como no LSQR, ainda vamos discutir os critérios de parada e a estimativa de alguns parâmetros.

Algoritmo 23 Método LSMR

function x=LSMR(A, b)
  $x_0 = 0;\ \ \overline{h}_0 = 0;$
  $\overline{u}_1 = b;\ \ \beta_1 = \|\overline{u}_1\|_2;\ \ u_1 = \overline{u}_1/\beta_1;$
  $\overline{v}_1 = A^T u_1;\ \ \alpha_1 = \|\overline{v}_1\|_2;\ \ v_1 = \overline{v}_1/\alpha_1;$
  $\overline{\alpha}_1 = \alpha_1;\ \ \overline{\zeta}_1 = \alpha_1\beta_1;$
  $\rho_0 = 1;\ \ \overline{\rho}_0 = \alpha_1;$
  $\overline{c}_0 = 1;\ \ \overline{s}_0 = 0;$
  $h_1 = v_1$
  while o critério de parada ser satisfeito (discussão abaixo) **do**
    $\overline{u}_{i+1} = Av_i - \alpha_i u_i;\ \ \beta_{i+1} = \|\overline{u}_{i+1}\|_2;\ \ u_{i+1} = \overline{u}_{i+1}/\beta_{i+1};$
    $\overline{v}_{i+1} = A^T u_{i+1} - \beta_{i+1} v_i;\ \ \alpha_{i+1} = \|\overline{v}_{i+1}\|_2;\ \ v_{i+1} = \overline{v}_{i+1}/\alpha_{i+1};$
    $\rho_i = (\overline{\alpha}_i^2 + \beta_{i+1}^2)^{1/2};$
    $c_i = \overline{\alpha}_i/\rho_i;\ \ s_i = \beta_{i+1}/\rho_i;$
    $\theta_{i+1} = s_i\alpha_{i+1};\ \ \overline{\alpha}_{i+1} = c_i\alpha_{i+1};$
    $\overline{\theta}_i = \overline{s}_{i-1}\rho_i;\ \ \overline{\rho}_i = [(\overline{c}_{i-1}\rho_i)^2 + \theta_{i+1}^2]^{1/2};$
    $\overline{c}_i = \overline{c}_{i-1}\rho_i/\overline{\rho}_i;\ \ \overline{s}_i = \theta_{i+1}/\overline{\rho}_i;$
    $\zeta_i = \overline{c}_i\overline{\zeta}_i;\ \ \overline{\zeta}_{i+1} = -\overline{s}_i\overline{\zeta}_i;$
    $\overline{h}_i = h_i - \left[\overline{\theta}_i\rho_i/(\rho_{i-1}\overline{\rho}_{i-1})\right]\overline{h}_{i-1};$
    $x_i = x_{i-1} + [\zeta_i/(\rho_i\overline{\rho}_i)]\overline{h}_i;$
    $h_{i+1} = v_{i+1} - (\theta_{i+1}/\rho_i)h_i;$
    $i = i + 1;$
  end while
  $x = x_{i-1};$
end function

Antes de discutirmos os critérios de parada para o método LSMR, vamos apresentar estimativas para $\|r_k\|_2$, $\|A^T r_k\|_2$, $\|x_k\|_2$, $\|A\|_F$ e $\kappa_F(A)$ que dependam das quantidades que já calculamos no Algoritmo 23, ou seja, com um custo computacional menor.

Estimativa de $\|r_k\|_2$: Para a estimativa de $\|r_k\|_2$ precisamos aplicar uma nova fatoração QR. A ideia é simples, transformamos $\overline{R}_k^T$ à forma bidiagonal superior via fatoração QR, a saber, $\tilde{R}_k = \tilde{Q}_k\overline{R}_k^T$, onde $\tilde{Q}_k = \tilde{P}_{k+1}\cdots\tilde{P}_1$. Essa fatoração QR extra tem custo baixo, pois requer apenas uma rotação a cada iteração. Sejam

$$\tilde{t}_k = \tilde{Q}_k t_k \quad \text{e} \quad \tilde{b}_k = \beta_1 \begin{bmatrix} \tilde{Q}_k & 0 \\ 0 & 1 \end{bmatrix} Q_{k+1} e_1.$$

Então, $r_k = b - Ax_k = \beta_1 u_1 - AV_k y_k = U_{k+1} e_1 \beta_1 - U_{k+1} B_k y_k$ fornece

$$\begin{aligned}
r_k &= U_{k+1}\left(e_1\beta_1 - Q_{k+1}^T \begin{bmatrix} R_k \\ 0 \end{bmatrix} y_k\right) \\
&= U_{k+1}\left(e_1\beta_1 - Q_{k+1}^T \begin{bmatrix} t_k \\ 0 \end{bmatrix}\right) \\
&= U_{k+1}\left(Q_{k+1}^T \begin{bmatrix} \tilde{Q}_k^T & 0 \\ 0 & 1 \end{bmatrix} \tilde{b}_k - \begin{bmatrix} \tilde{Q}_k^T \tilde{t}_k \\ 0 \end{bmatrix}\right) \\
&= U_{k+1} Q_{k+1}^T \begin{bmatrix} \tilde{Q}_k^T & 0 \\ 0 & 1 \end{bmatrix} \left(\tilde{b}_k - \begin{bmatrix} \tilde{t}_k \\ 0 \end{bmatrix}\right).
\end{aligned}$$

Por argumentos de ortogonalidade,

$$\|r_k\|_2 = \left\| \tilde{b}_k - \begin{bmatrix} \tilde{t}_k \\ 0 \end{bmatrix} \right\|_2 .$$

Os vetores $\tilde{b}_k$ e $\tilde{t}_k$ podem ser escritos da seguinte forma

$$\tilde{b}_k = \left(\tilde{\beta}_1, \cdots, \tilde{\beta}_{k-1}, \dot{\beta}_k, \ddot{\beta}_{k+1}\right)^T \quad \text{e} \quad \tilde{t}_k = (\tilde{\tau}_1, \ldots, \tilde{\tau}_{k-1}, \dot{\tau}_k)^T.$$

O vetor $\tilde{t}_k$ é solução do sistema linear triangular $\tilde{R}_k^T \tilde{t}_k = z_k$. No apêndice A de [41] é demonstrado que $\tilde{\beta}_i = \tilde{\tau}_i$, $i = 1, \ldots, k-1$, ou seja, uma vez determinado o vetor $\tilde{t}_k$, os primeiros $k-1$ elementos de $\tilde{b}_k$ são determinados faltando apenas determinar $\dot{\beta}_k$ e $\ddot{\beta}_{k+1}$. A seguir, apresentamos um algoritmo para a determinação de $\|r_k\|_2$.

Algoritmo 24 Método LSMR: estimativa $\|r_k\|_2$

```
function $\|r_k\|_2$=LSMRrk($\overline{\theta}_k, \overline{\rho}_k, \zeta_k, \zeta_{k-1}, c_k, s_k, \beta_1$)
    $\ddot{\beta}_{=}\beta_1$;  $\dot{\beta}_0 = 0$;  $\dot{\rho}_0 = 1$;  $\tilde{\tau}_{-1} = 0$;  $\tilde{\theta}_0 = 0$;  $\zeta_0 = 0$;
    $\hat{\beta}_k = c_k \ddot{\beta}_k$;  $\ddot{\beta}_{k+1} s_k \ddot{\beta}_k$;
    %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
    %%%%%% Para a $k$-ésima iteração repita os passos abaixo %%%%
    %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
    $\tilde{\rho}_{k-1} = (\dot{\rho}_{k-1}^2 + \overline{\theta}_k^2)^{1/2}$;  $\tilde{c}_{k-1} = \dot{\rho}_{k-1}/\tilde{\rho}_{k-1}$;  $\tilde{s}_{k-1} = \overline{\theta}_k/\tilde{\rho}_{k-1}$;
    $\tilde{\theta}_k = \tilde{s}_{k-1}\overline{\rho}_k$;  $\dot{\rho}_k = \tilde{c}_{k-1}\overline{\rho}_k$;
    $\tilde{\beta}_{k-1} = \tilde{c}_{k-1}\dot{\beta}_{k-1} + \tilde{s}_{k-1}\hat{\beta}_k$;  $\dot{\beta}_k = -\tilde{s}_{k-1}\dot{\beta}_{k-1} + \tilde{c}_{k-1}\hat{\beta}_k$;
    $\tilde{\tau}_{k-1} = (\zeta_{k-1} - \tilde{\theta}_{k-1}\tilde{\tau}_{k-2})/\tilde{\rho}_{k-1}$;  $\dot{\tau}_k = (\zeta - \tilde{\theta}_k\tilde{\tau}_{k-1})/\dot{\rho}_k$;
    $\gamma = (\dot{\beta}_k - \dot{\tau}_k)^2 + \ddot{\beta}_{k+1}^2$;  $\|r_k\|_2 = \sqrt{\gamma}$;
end function
```

Estimativa de $\|A^T r_k\|_2$: De (4.6.98) concluímos que $\|A^T r_k\|_2 = |\overline{\zeta}_{k+1}|$. Por outro lado $\overline{\zeta}_{k+1} = -\overline{s}_k \overline{\zeta}_k$ demonstra que $\|A^T r_k\|_2$ decresce monotonicamente.

Estimativa de $\|x_k\|_2$: Note que, como $x_k = V_k y_k$, $R_k y_k = t_k$ e $\overline{R}_k t_k = z_k$, então $x_k = V_k R_k^{-1} \overline{R}_k^{-1} z_k$. Da terceira fatoração QR, $\tilde{Q}_k \overline{R}_k^T = \tilde{R}_k$ e,

depois da aplicação de uma quarta fatoração QR, obtemos $\hat{Q}_k(\tilde{Q}_k R_k)^T = \hat{R}_k$. Portanto,

$$x_k = V_k R_k^{-1}\overline{R}_k^{-1} z_k = V_k R_k^{-1}\overline{R}_k^{-1}\overline{R}_k \tilde{Q}_k^T \tilde{z}_k = V_k R_k^{-1}\tilde{Q}_k^T \tilde{Q}_k R_k \hat{Q}_k^T \hat{z}_k = V_k \hat{Q}_k^T \hat{z}_k.$$

Os sistemas lineares, resolvidos por substituição direta, $\tilde{R}_k^T \tilde{z}_k = z_k$ e $\hat{R}_k^T \hat{z}_k = \tilde{z}_k$ fornecem $\hat{z}_k$. Logo, por argumentos de ortogonalidade, $\|x_k\|_2 = \|\hat{z}_k\|_2$.

Estimativa de $\|A\|_F$ e $\kappa_F(A)$: Do estudo de LSQR sabemos que os valores singulares de A e B_k estão entrelaçados e os limitantes são, respectivamente, o menor e o maior valor singular de A. Assim, a estimativa para $\|A\|_F$ é $\|B_k\|_F$. Já a estimativa de $\kappa_F(A)$ é $\kappa_F(B_k)$, com argumentos similares aos trabalhados no método LSQR. Sabemos que a matriz B_k satisfaz a seguinte relação de recorrência

$$\|B_{k+1}\|_F^2 = \|B_k\|_F^2 + \alpha_k^2 + \beta_{k+1}^2.$$

Considere (4.6.96), (4.6.97) e (4.6.98), e então vale a seguinte fatoração QLP [137]

$$Q_{k+1} B_k \overline{Q}_k^T = \begin{pmatrix} \overline{R}_{k-1}^T & 0 \\ \overline{\theta}_k e_{k-1}^T & \overline{c}_{k-1}\rho_k \end{pmatrix}.$$

Stewart demonstra que os valores singulares de B_k são aproximados pelos elementos da matriz bidiagonal inferior [137]. Como os elementos das diagonais são positivos, então

$$\kappa_F(B_k) = \frac{\max \mathscr{B}}{\min \mathscr{B}},$$

onde $\mathscr{B} = \{\overline{\rho}_1, \ldots, \overline{\rho}_{k-1}, \overline{c}_{k-1}\rho_k\}$.

Os critérios de parada para LSMR são os mesmos que foram desenvolvidos para o LSQR, a saber,

S1. $\|r_k\|_2 \leqslant \mathrm{BTOL}\|b\|_2 + \mathrm{ATOL}\|A\|_F\|x_k\|_2$,

S2. $\dfrac{\|A^T r_k\|_2}{\|A\|_F\|r_k\|_2} \leqslant \mathrm{ATOL}$,

S3. $\kappa_F(A) \geqslant \mathrm{COLIM}$,

onde ATOL, BTOL e COLIM são definidos pelo usuário e dependem do modelo utilizado.

Exemplo 4.4. *Vamos comparar os métodos CGLS, LSQR e LSMR na resolução do problema de mínimos quadrados*

$$\min_x \|Ax - b\|_2,$$

com $A \in \mathbb{R}^{m\times n}$ retirada do SuiteSparse Matrix Collection[8] *e $b \in \mathbb{R}^m$ um vetor formado por 1's. Vamos analisar os três principais parâmetros para*

[8] `https://sparse.tamu.edu/`, antigo *Florida University Sparse Matrix Collection.*

aferir a qualidade da solução encontrada. Utilizamos três modelos, a saber, well1850, lp_pilot e lp_woodw.

Modelo	Linhas	Colunas	Elementos não nulos	$\kappa_2(A)$
well 1850	1850	712	8755	$1.113129e+02$
lp_pilot	4860	1441	44375	$2.661950e+03$
lp_woodw	8418	1098	37847	$4.701471e+04$

Tabela 4.2: Informações básicas dos modelos utilizados.

Na Tabela 4.3 apresentamos o modelo well1850 com seus dados de convergência para CGLS, LSQR e LSMR.

well1850	CGLS	LSQR	LSMR
# de iterações	438	440	443
$\|e_k\|_2$	1.0349e-006	$9.4378e-007$	$1.1719e-006$
$\|r_k\|_2$	$1.2666e-007$	$1.1463e-007$	$1.1287e-007$
$\|A^T r_k\|_2$	$5.4132e-008$	$4.6619e-008$	$1.4145e-008$

Tabela 4.3: Comparação entre CGLS, LSQR e LSMR para well1850.

Na Figura 4.6 a seguir, apresentamos os gráficos do $e_i = x - x_k$, $r_i = Ae_i$ e $A^T r_i$. Na subfigura a) vemos a taxa de convergência de cada um dos métodos iterativos. Observe que os três têm taxas de convergência similares, vide a subfigura b), que é um zoom *da subfigura a). Na subfigura c) mostramos apenas os resíduos e note seu rápido decaimento. Vemos que ambos os métodos têm um decaimento monotônico. Na subfigura d) vemos que o decaimento monotônico de LSMR, como esperado, enquanto o decaimento de LSQR não é monotônico.*

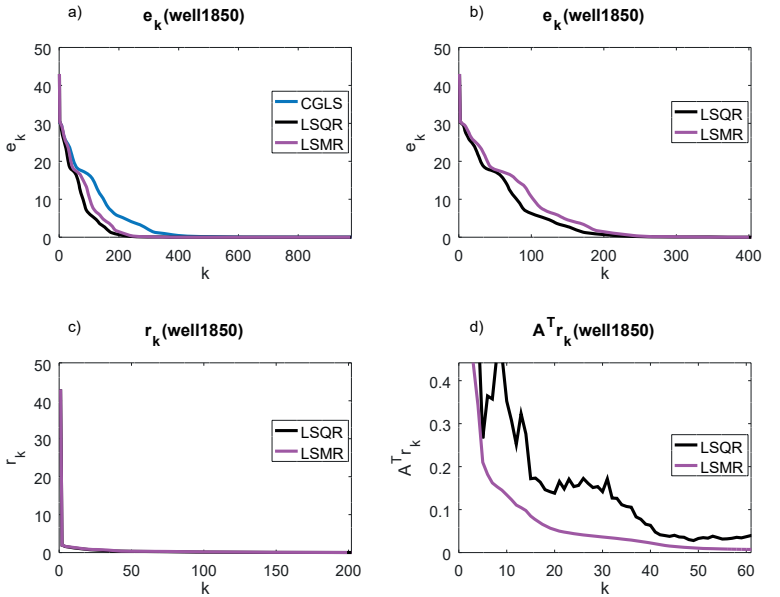

Figura 4.6: Resultados de $e_i = x - x_k$, $r_i = Ae_i$ e $A^T r_i$ para o modelo well1850.

Na Tabela 4.4 apresentamos o modelo lp_pilot com seus dados de convergência para CGLS, LSQR e LSMR.

lp_pilot	CGLS	LSQR	LSMR
# de iterações	5000	5000	5000
$\|e_k\|_2$	$1.6281e-009$	$1.4756e-012$	$2.1785e-011$
$\|r_k\|_2$	$1.0448e-009$	$1.6653e-012$	$3.6366e-012$
$\|A^T r_k\|_2$	$1.0442e-008$	$1.7679e-010$	$1.5916e-010$

Tabela 4.4: Comparação entre CGLS, LSQR e LSMR para lp_pilot.

Na Figura 4.7 a seguir, apresentamos os gráficos do $e_i = x - x_k$, $r_i = Ae_i$ e $A^T r_i$. Deixamos o solver *correr até um número máximo de iterações de 5000 para CGLS, LSQR e LSMR. Na subfigura a) vemos a taxa de convergência de cada um dos métodos iterativos, e observe que os três métodos têm taxas de convergência similares, vide a subfigura b). Comparando os gráficos das figuras 4.6 e 4.7, vemos que com o aumento do número de condição de A, a discrepância entre as convergências de LSQR-LSMR e CGLS se torna mais evidente. O subfigura a) confirma que LSQR e LSMR são alternativas mais robustas para problemas de quadrados mínimos. Na subfigura c) vemos que os métodos têm um decaimento monotônico e, na subfigura d), vemos que*

o decaimento de LSMR é monotônico (pela construção do método), mas o decaimento de LSQR não é monotônico, porém é convergente.

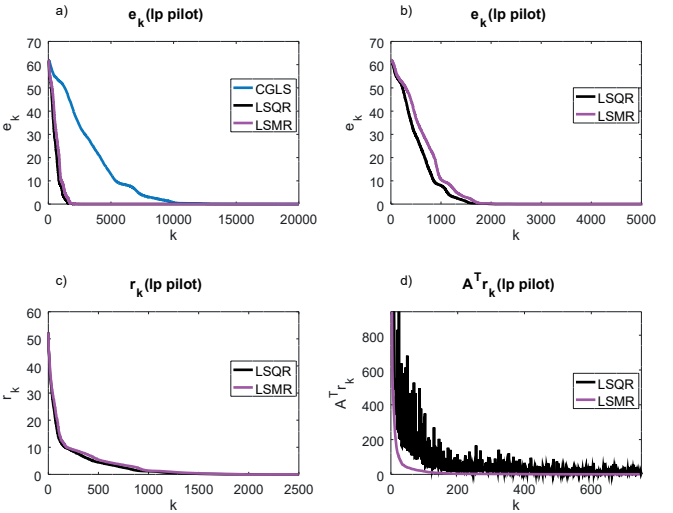

Figura 4.7: Resultados de $e_i = x - x_k$, $r_i = Ae_i$ e $A^T r_i$ para o modelo lp_pilot.

Na Tabela 4.5 apresentamos o modelo lp_woodw com seus dados de convergência para CGLS, LSQR e LSMR. Deixamos o solver *correr até um número máxima de iterações de 1500 para os três métodos. LSQR apresenta um melhor comportamento final em relação ao erro e ao resíduo porém, quando analisamos $A^T r_k$ ou o resíduo das equações normais, observamos que LSMR apresenta um melhor resultado. Isso se deve a dois fatos: primeiro, o decaimento do resíduo das equações normais para LSMR é monotônico enquanto para LSQR não, veja subfigura 4.8 d). Segundo, 1500 iterações é pouco para observar que LSQR também converge para $A^T r_k$.*

lp_woodw	CGLS	LSQR	LSMR
# de iterações	1500	1500	1500
$\|e_k\|_2$	$4.5094e-009$	$9.2429e-012$	$9.7958e-012$
$\|r_k\|_2$	$8.2994e-009$	$7.4612e-011$	$7.5144e-011$
$\|A^T r_k\|_2$	$1.8340e-005$	$9.8288e-007$	$9.8166e-007$

Tabela 4.5: Comparação entre CGLS, LSQR e LSMR para lp_woodw.

Na Figura 4.8, apresentamos os gráficos do $e_i = x - x_k$, $r_i = Ae_i$ e $A^T r_i$. Na subfigura a) vemos a taxa de convergência de cada um dos métodos iterativos.

Observe que apesar de LSQR parecer ter uma curva de decaimento mais inclinada que LSMR, a convergência de ambos se dá para um número similar de iterações, vide a subfigura b). Na subfigura c) mostramos que os três métodos têm um decaimento monotônico esperado e, na subfigura d), vemos o decaimento monotônico de LSMR e o decaimento oscilatório de CGLS e LSQR. A oscilação do CGLS é mais acentuada e, em geral, LSQR e LSMR são mais precisos que CGLS.

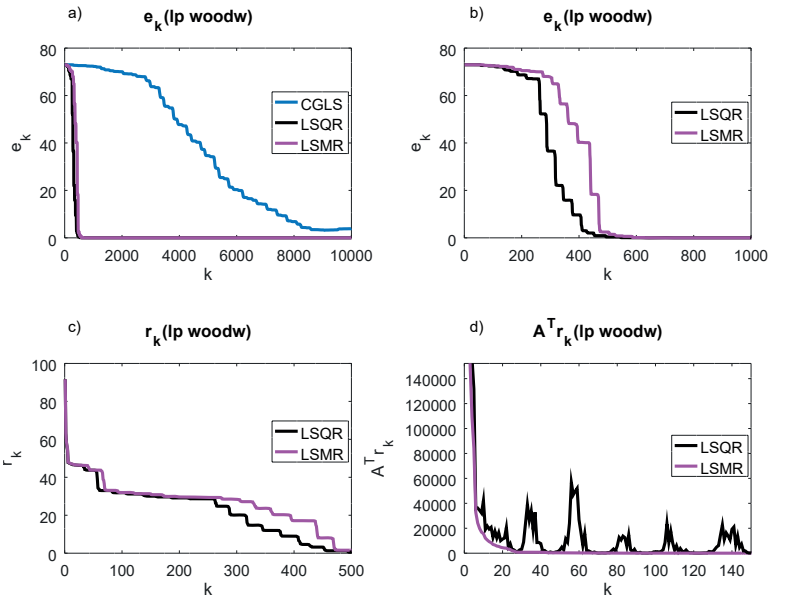

Figura 4.8: Resultados de $e_i = x - x_k$, $r_i = Ae_i$ e $A^T r_i$ para o modelo lp_woodw.

4.7 Exercícios

1. Demonstre o Teorema 4.1.

2. [77] Demonstre que se $A \in \mathbb{R}^{n\times n}$ tem exatamente $k \leqslant n$ autovalores distintos, então o método dos gradientes conjugados termina em no máximo k iterações.

3. [77] Seja $\{x_k\}$ a sequência das iteradas do método de gradientes conjugados. Demonstre que o resíduo $r_i \in \mathcal{K}_k(A, r_0)$, $i < k$.

4. [144] Seja $A \in \mathbb{R}^{n\times n}$ uma matriz definida positiva e $b \in \mathbb{R}^n$. Então, para cada vetor inicial $x_0 \in \mathbb{R}^n$ existe um menor inteiro não negativo $m \leqslant n$, tal que $p_m = 0$. Demonstre que os vetores x_k, p_k e r_k gerados pelo Algoritmo 17 tem as seguintes propriedades:

   1. $Ax_m = b$.
   2. $r_j^T p_i = 0$ para $0 \leqslant i < j \leqslant m$.
   3. $r_i^T p_i = r_i^T r_i$ para $i \leqslant m$.
   4. $p_i^T A p_j = 0$ para $0 \leqslant i < j \leqslant m$ e $p_j^T A p_j > 0$ para $j < m$.
   5. $r_i^T r_j = 0$ para $0 \leqslant i < j \leqslant m$ e $r_j^T r_j > 0$ para $j < m$.
   6. $r_i = b - Ax_i$ para $i \leqslant m$.

5. Demonstre, no contexto do método de máxima descida que r_i e r_{i+1} são ortogonais, para $i = 0, \ldots, n-1$.

6. Demonstre o Teorema 4.11 (pág. 123).

7. Sejam $A \in \mathbb{R}^{n\times n}$ uma matriz definida positiva e $p_0, \ldots, p_{n-1} \in \mathbb{R}^n \setminus \{0\}$ vetores dois a dois A-conjugados. Dado x_0, considere $r_0 = b - Ax_0$. Para $k = 0, \ldots, n-1$,

$$\begin{cases} \alpha_k = \dfrac{p_k^T r_k}{p_k^T A p_k}, \\[2ex] x_{k+1} = x_k + \alpha_k p_k, \\[2ex] r_{k+1} = r_k - \alpha_k A p_k. \end{cases}$$

   Demonstre que:

   1. $r_k = b - Ax_k$.
   2. $x_{k+1} = \min\limits_{\alpha\in\mathbb{R}} f(x_k + \alpha_k p_k)$, onde $f(x) = \dfrac{1}{2}x^T A x - x^T b + c$.
   3. $x_n = A^{-1}b$.
   4. $x_k = \min\limits_{x\in x_0+S_k} f(x)$, onde $S_k = span\{p_0, \ldots, p_{k-1}\}$.

8. Justifique teoricamente o Algoritmo 18.

9. Demonstre que o Algoritmo 20 termina quando $\alpha_j = 0$ ou $\beta_j = 0$.

10. Implemente o processo de bidiagonalização proposto por Golub e Kahan [52] utilizando reflexões de Householder. Faça o mesmo com rotações de Givens.

11. Demonstre que o Algoritmo 20 termina com $\alpha_j = 0$, $j < n$, se tivermos $\text{rank}(A) < n$.

12. Demonstre o resultado (4.5.93).

13. [41] Demonstre que tanto LSQR quanto LSMR fornecem soluções de norma mínima, quando A é posto deficiente.

Capítulo 5

Precondicionamento

Muitos métodos iterativos utilizados na resolução de problemas de quadrados mínimos têm, como ponto fraco, a lenta convergência quando o número de condição do modelo que está sendo trabalhado é elevado, vide CGLS. Por outro lado, os mesmos algoritmos convergem mais rapidamente quando o número de condição do modelo "está perto" do número de condição da identidade. A noção de precondicionamento surge com o objetivo de transformar um problema com alto número de condição em um matematicamente equivalente com número de condição baixo, proporcionando assim, uma rápida convergência.

Nesse capítulo apresentamos o conceito de precondicionamento e como aplicá-lo aos métodos iterativos que estudamos. Não temos como objetivo uma discussão aprofundada sobre o assunto, que por sua vez, é bastante vasto. Queremos apenas que o leitor saiba as noções básicas e como utilizá-las rapidamente para impor um ganho computacional aos métodos numéricos estudados nesse texto. As principais bibliografias sobre esse assunto, nas quais nos apoiamos, são os livros de Bertaccini e Durastante [8], Björck [12, 13] e Golub e Van Loan [58].

5.1 Noções Básicas

Considere o sistema linear $Ax = b$, com $b \in \mathbb{R}^n$ e $A \in \mathbb{R}^{n\times n}$, geralmente de grande porte e esparsa. Gostaríamos de acelerar a convergência de um método numérico aplicado ao sistema linear em questão. Para isso, precisamos encontrar um sistema linear equivalente $\tilde{A}x = \tilde{b}$ que possua propriedades espectrais mais interessantes e, portanto, que tenha um número de condição baixo.

Definição 5.1. *Seja $S \in \mathbb{R}^{n\times n}$ uma matriz invertível, chamada de precondicionador. Podemos efetuar três tipos de precondicionamento.*

1. Precondicionamento à esquerda:

$$S^{-1}Ax = S^{-1}b, \quad \tilde{A} = S^{-1}A.$$

2. *Precondicionamento à direita:*

$$AS^{-1}y = b, \quad \tilde{A} = AS^{-1} \ \ e \ \ x = S^{-1}y.$$

3. *Precondicionamento misto: para esse precondicionamento precisamos que S possa ser fatorada na forma $S = S_1S_2$ e, assim*

$$S_1^{-1}AS_2^{-1}y = S_1^{-1}b, \quad \tilde{A} = S_1^{-1}AS_2^{-1} \ \ e \ \ x = S_2^{-1}y.$$

Uma primeira abordagem que viria a nossa mente é $S = A$ e nesse caso $S^{-1} = A^{-1}$ e, portanto, $\tilde{A} = I$. Assim, o método convergiria em zero passos! Porém, essa abordagem não é possível por duas razões bem conhecidas. A primeira é que o cálculo de A^{-1} é muito mais caro computacionalmente que resolver o sistema. A segunda é que, em geral, a inversa de uma matriz esparsa é densa [49].

Para problemas de quadrados mínimos tomamos $S \in \mathbb{R}^{n\times n}$ uma matriz invertível e transformamos o problema

$$\min_x \|Ax - b\|_2$$

em um problema equivalente

$$\min_y \|AS^{-1}y - b\|_2, \quad Sx = y.$$

Em geral, utilizam-se precondicionadores à direita ou mistos pois, do ponto de vista computacional, AS^{-1} não deve ser formado por razões já explicadas. Porém, nos métodos numéricos precondicionados os produtos $AS^{-1}y$ e $S^{-T}A^Tr$ estarão presentes na dedução teórica dos mesmos e, portanto, um custo extra do precondicionamento será adicionado a um método iterativo para resolver $Sx = y$ e $S^Tq = s$ como alternativa à formação de matrizes inversas. A ideia é que o custo adicional de resolver sistemas lineares seja diluído no ganho computacional advindo do precondicionamento. Ademais, gostaríamos que os sistemas lineares fossem de fácil resolução.

Björck [12, pág. 283] resume o que se espera de um bom precondicionador, note que algumas características são parcialmente contraditórias:

1. AS^{-1} deve ser melhor condicionada que A e/ou possuir poucos valores singulares distintos;

2. S deve ter mais ou menos o mesmo número de elementos não nulos que A;

3. deve ser computacionalmente barato resolver sistemas lineares com S e S^T.

5.2 CGLS Precondicionado

Assuma que o precondicionador S seja dado — formas de obter precondicionadores é o tema das próximas seções. Vamos iniciar essa seção entendendo o precondicionamento do método dos gradientes conjugados e, posteriormente, passamos ao CGLS.

Sejam $A \in \mathbb{R}^{n\times n}$ uma matriz simétrica definida positiva e $b \in \mathbb{R}^n$, e considere o sistema linear $Ax = b$. O principal fato a ser notado é que a busca pelo precondicionador S deve ser feita de forma que o sistema $\tilde{A}x = \tilde{b}$ seja definido positivo. Uma óbvia opção é S simétrica definida positiva.

Observe que $S^{-1}A$ é simétrica para os A-produto interno e S-produto interno:

$$\left\langle S^{-1}Ax, y\right\rangle_S = \langle Ax, y\rangle = \langle x, Ay\rangle = \left\langle x, SS^{-1}Ay\right\rangle = \left\langle x, S^{-1}Ay\right\rangle_S,$$

e

$$\left\langle S^{-1}Ax, y\right\rangle_A = \left\langle AS^{-1}Ax, y\right\rangle = \left\langle x, AS^{-1}Ay\right\rangle = \left\langle x, S^{-1}Ay\right\rangle_A.$$

Queremos determinar um método iterativo para solucionar o sistema linear $\tilde{A}x = \tilde{b}$ e, para isso, vejamos o resíduo $\tilde{r}_k$,

$$\tilde{r}_k = \tilde{b} - \tilde{A}x_k = S^{-1}(b - Ax_k) = S^{-1}r_k := z_k.$$

O algoritmo do método dos gradientes conjugados é alterado substituindo-se o produto interno canônico de $\mathbb{R}^n$ pelo S-produto interno. Essa substituição não necessita ser explícita, pois

$$\begin{cases} \langle z_k, z_k\rangle_S = \left\langle S^{-1}r_k, S^{-1}r_k\right\rangle_S = \langle r_k, z_k\rangle, \\ \left\langle S^{-1}Ap_k, p_k\right\rangle_S = \langle Ap_k, p_k\rangle. \end{cases}$$

Para finalizar a construção do método iterativo note que,

$$\tilde{p}_k = \tilde{r}_k + \mu_k\tilde{p}_k = S^{-1}r_k + \mu_k\tilde{p}_k = z_k + \mu_k\tilde{p}_k.$$

De forma análoga à dedução do CG precondicionado podemos deduzir o CGLS precondicionado, que é deixado como exercício. A seguir, apresentamos um algoritmo para a resolução de sistemas lineares utilizando o método dos gradientes conjugados precondicionado.

Algoritmo 25 Método dos Gradientes Conjugados precondicionado

```
function x_*=PCG(S, A, b, x_0)
    k = 0;  r_0 = b − Ax_0;  Sz_0 = r_0;
    while ||r_k||_2 > 0 do
        k = k + 1;
        if k=1 then
            p_k = z_0;
        else
            τ_{k−1} = (r_{k−1}^T z_{k−1})/(r_{k−2}^T z_{k−2});
            p_k = z_{k−1} + τ_{k−1} p_{k−1};
        end if
        q_k = Ap_k;
        μ_k = (r_{k−1}^T z_{k−1})/(p_k^T q_k);
        x_k = x_{k−1} + μ_k p_k;
        r_k = r_{k−1} − μ_k q_k;
        Sz_k = r_k;
    end while
    x_* = x_k;
end function
```

Observe que um precondicionador à direita, ou seja, AS^{-1}, não é simétrico no produto interno canônico, tampouco no S-produto interno. Matematicamente, a cada iteração estamos realizando operações da forma

$$x \mapsto S^{-1}Ax.$$

Sejam $A \in \mathbb{R}^{m\times n}$ e $b \in \mathbb{R}^m$ e considere o sistema linear $Ax = b$. A solução de quadrados mínimos do sistema linear é obtida resolvendo o problema de minimização

$$\min_x \|Ax - b\|_2.$$

Gostaríamos de acelerar o método CGLS e, para isso, vamos resolver o seguinte problema (equivalente ao original)

$$\min_y \|AS^{-1}y - b\|_2, \quad Sx = y.$$

As equações normais para esse problema são

$$S^{-T}A^T(AS^{-1}y - b) = S^{-T}A^T(Ax - b) = 0.$$

Por esse fato, o CGLS precondicionado ainda minimiza o funcional $\|r_* - r_k\|_2$, onde $r_k = b - Ax_k$, mas essa minimização ocorre em outro subespaço de Krylov. Da Teorema 4.16, obtemos

$$\|r_* - r_k\|_2 < 2\left(\frac{\kappa(AS^{-1}) - 1}{\kappa(AS^{-1}) + 1}\right)\|r_* - r_0\|_2.$$

A seguir, apresentamos um algoritmo para o CGLS precondicionado (PCGLS).

Algoritmo 26 Método dos Gradientes Conjugados: CGLS precondicionado

```
function x_*=PCGLS(S, A, b, x_0)
    k = 0;  r_0 = b − Ax_0;  S^T p_0 = A^T r_0;  s_0 = p_0  γ_0 = ||s_0||_2^2;
    while γ_k > 0 do
        St_k = p_k
        q_k = At_k;
        μ_k = γ_k/||q_k||_2^2;
        x_{k+1} = x_k + μ_k t_k;
        r_{k+1} = r_k − μ_k q_k;
        S^T s_{k+1} = A^T r_{k+1};
        γ_{k+1} = ||s_{k+1}||_2^2;
        τ_k = γ_{k+1}/γ_k;
        p_{k+1} = s_{k+1} + τ_k p_k;
        k = k + 1;
    end while
    x_* = x_k;
end function
```

A próxima seção é dedicada à determinação de precondicionadores para os método CG e CGLS.

5.3 Precondicionadores de Fatorações Incompletas

Nessa seção estudaremos as fatorações incompletas e como utilizá-las para precondicionar um método iterativo, apresentando algumas técnicas básicas. O objetivo dos precondicionadores de fatorações incompletas é obter uma fatoração aproximada que preserve a esparsidade de A em seus fatores triangulares. Esses fatores triangulares são utilizados como precondicionadores, como veremos a seguir.

Definição 5.2. *Precondicionadores obtidos de uma aproximação para uma matriz A são chamados de precondicionadores implícitos.*

Nas próximas seções apresentamos a fatoração de Cholesky incompleta, as fatorações ortogonais incompletas e como determinar precondicionadores através da fatoração LU.

5.3.1 Fatoração de Cholesky Incompleta

Sejam $A \in \mathbb{R}^{n\times n}$ e $b \in \mathbb{R}^n$. Considere A definida positiva e, portanto, A admite fatoração de Cholesky $A = R^T R$, com $R \in \mathbb{R}^{n\times n}$ triangular superior. Sendo A esparsa, sua fatoração de Cholesky pode produzir R mais densa que A ou mesmo cheia. Seguindo a ideia de [8, 78, 94], sob condições que ainda veremos, a fatoração incompleta de Cholesky (IC) é

$$A = \tilde{R}^T \tilde{R} - C$$

onde $\tilde{R}$ é uma matriz triangular superior e C é uma matriz contendo os elementos que são descartados ao longo da decomposição para manter $\tilde{R}$

com o mesmo padrão de esparsidade da matriz A dada. Ademais, queremos que $\|C\|_F$ seja "pequeno".

Existem várias estratégias de como descartar elementos durante a fatoração e acumulá-los em C. Vamos apresentar uma dessas estratégias, a chamada fatoração de Cholesky incompleta com preenchimento de zeros ou, como é conhecida, IC(0). A ideia é durante o processo da fatoração de Cholesky descartar os elementos com posição (i, j) iguais a zero na parte triangular superior de A.

Sejamos um pouco mais precisos matematicamente. Defina

$$P = \{(i, j) \in \{1, \ldots, n\} \,:\, A = [a_{ij}],\ a_{ij} \neq 0\},$$

ou seja, P é o conjunto das posições das entradas de A não nulas, que definem o padrão de esparsidade de A. Durante o processo de fatoração de Cholesky, se $(i, j) \in P$, então calcula-se normalmente o elemento e se $(i, j) \notin P$, então o elemento $r_{ij} = 0$.

A seguir, apresentamos um algoritmo para o cálculo da fatoração de Cholesky incompleta, e note que não estamos construindo C, pois o fator relevante é R.

Algoritmo 27 Fatoração de Cholesky Incompleta

```
function G=IncChol(A)
    [m, n] =size(A);
    a_11 = √a_11;
    for j = 2 : n do
        a_1j = a_1j/a_11;
    end for
    for i = 2 : n − 1 do
        for k = 1 : i − 1 do
            a_ii = a_ii − a_ki^2;
        end for
        a_ii = √a_ii;
        for j = i + 1 : n do
            if (i, j) ∉ P then
                a_ij = 0;
            else
                for k = 1 : i − 1 do
                    a_ij = a_ij − a_ki a_kj;
                end for
                a_ij = a_ij/a_ii;
            end if
        end for
    end for
    for k = 1 : n − 1 do
        a_nn = a_nn − a_kn^2;
    end for
    a_nn = √a_nn;
    R = A−tril(A, −1);  % extrai a diagonal e as entradas acima da diagonal
end function
```

A hipótese de A ser definida positiva não é suficiente para garantir esse

algoritmo, isto é, a existência da fatoração de Cholesky incompleta. A existência é garantida através do conceito de M-matriz.

Definição 5.3. *Uma matriz $A = [a_{ij}] \in \mathbb{R}^{n\times n}$ é uma M-matriz se*

1. $a_{ij} \leqslant 0$ *para* $i \neq j$,
2. $a_{ii} > 0$,
3. $\det(A) \neq 0$,
4. $A^{-1} \geqslant 0$.

Observação 5.1. *Podem ser encontradas na literatura outras definições de M-matriz, embora Bertaccini e Durastante [8, pág. 95] demonstram a equivalência entre as definições. Para outras propriedades das M-matrizes, vide [115].* □

O próximo resultado, demonstrado por Stieljes [142] em 1887, fornece uma relação entre M-matrizes e matrizes definidas positivas.

Teorema 5.1. *(Stieljes) Seja $A \in \mathbb{R}^{n\times n}$, com $a_{ij} \leqslant 0$, para $i \neq j$. Então, A é uma M-matriz se A for definida positiva.*

Meijerink e van der Vorst [94, pág. 151] demonstram a existência e estabilidade da fatoração de Cholesky.

Teorema 5.2. *[12, 94] Se A é uma M-matriz simétrica, então existe para cada conjunto simétrico P, tal que $(i,j) \in P$ para $i = j$, uma única matriz triangular superior com $r_{ij} = 0$ se $(i,j) \notin P$, tal que $A = R^T R - C$, $(R^T R)^{-1} \geqslant 0$, $C \geqslant 0$.*

A demonstração envolve a equivalência entre fatoração de Cholesky e fatoração LU e é feita através da fatoração LU incompleta que foge do escopo do livro.

5.3.2 Fatorações Ortogonais Incompletas

Nessa seção vamos explorar o algoritmo de Gram-Schmidt modificado para deduzir uma fatoração incompleta, que também é chamada de IMGS ou Gram-Schmidt modificado incompleto, via fatoração QR. Com essa metodologia o método PCGLS converge mais rápido que IC.

Jennings e Ajiz [71] apresentam a ideia do método de Gram-Schmidt incompleto. Assim como para a fatoração de Cholesky incompleta, vamos utilizar uma tolerância para determinar os elementos a serem descartados. A ideia é comparar os elementos de R que não pertençam à diagonal e verificar se são menores que a tolerância multiplicada pela norma da i-ésima coluna de A, ou seja, $|r_{ij}| < \tau\|a_i\|_2$. A cada passo de IMGS determinamos uma matriz $\tilde{R}$ que é triangular superior com elementos da diagonal positivos e uma matriz $\tilde{Q}$, tal que $A = \tilde{Q}\tilde{R}$ e $span\{a_i, \ldots, a_k\} = span\{\tilde{q}_1, \ldots, \tilde{q}_k\} \quad k = 1, \ldots, n$. Importante dizer que se A tem posto completo então o seguinte algoritmo fornece a solução esperada.

Algoritmo 28 Processo de Gram-Schmidt Modificado Incompleto

```
function Q=IMGS(A)
    [m, n] =size(A); R =zeros(n);
    for j = 1 : n do
        q_j = a_j;
        for i = 1 : j − 1 do
            r_ij = ⟨q_i, q_j⟩;
            if |r_ij| < τ‖a_i‖_2 then
                r_ij = 0
            end if
            q_j = q_j − r_ij q_i;
        end for
        r_jj = ‖q_j‖_2;
        q_j = q_j/r_jj;
    end for
end function
```

Outras alternativas de implementação do IMGS são dadas por Saad [124] e Zlatev e Nilsen [166]. Wang, Gallivan e Bramley [155] propuseram o CIMGS ou método compacto de Gram-Schmidt incompleto, desenvolvendo um algoritmo em que não há a necessidade de armazenar a matriz Q. Pode ser mostrado que em aritmética exata o algoritmo CIMGS produz exatamente o mesmo resultado que o algoritmo IMGS. A seguir, apresentamos um algoritmo para CIMGS.

Algoritmo 29 Processo de Gram-Schmidt Modificado Incompleto: CIMGS

```
function [Q, R]=IMGS(A)
    for i = 1 : n do
        r_ii = √a_ii;
        for j = i + 1 : n do
            a_ij = a_ij/r_ii;
            if (i, j) ∉ P then r_ij = 0;
            else
                r_ij = a_ij;
            end if
        end for
        for j = i + 1 : n do
            for k = i + 1 : n do
                if (i, j) ∈ P || (i, k) ∈ P then a_kj = a_kj − a_ik a_ij;
                end if
            end for
        end for
    end for
end function
```

Exemplo 5.1. *Vejamos o desempenho de PCGLS comparado a CGLS, LSQR e LSMR. Para isso, utilizamos quatro matrizes* $m \times n$ *do* SuiteSparse Matrix Collection*, a saber, ash958, well1033, well1850 e lp_woodw. O vetor* $b \in \mathbb{R}^m$ *é formado por 1's e a solução exata foi calculada via decomposição LU.*

Modelo	Linhas	Colunas	Elementos não nulos	$\kappa_2(A)$
well 1033	1033	320	4732	$1.661333e+02$
well 1850	1850	712	8755	$1.113129e+02$
lp_woodw	8418	1098	37847	$4.701471e+04$
ash958	958	292	1916	$3.201358e+00$

Tabela 5.1: Informações básicas dos modelos utilizados.

Para os três primeiros modelos procuramos analisar a convergência e apresentamos os resultados na Tabela 5.2. Observe que PCGLS converge mais rapidamente que CGLS, LSQR e LSMR. Já no quarto teste tomamos um modelo mais robusto e temos como objetivo analisar o erro cometido por cada método. Para isso corremos os solvers *até um número máximo de iterações, 5000 neste caso, e observamos os erros. Note que, novamente, PCGLS apresenta um desempenho superior, ou seja, erros menores, quando comparado com CGLS, LSQR e LSMR.*

ash958	CGLS	LSQR	LSMR	PCGLS
# de iterações	21	23	23	18
$\|e_k\|_2$	$2.5308e-006$	$1.2218e-006$	$1.5509e-006$	$2.9212e-006$
$\|r_k\|_2$	$4.4229e-006$	$2.1477e-006$	$2.3900e-006$	$6.7951e-006$
$\|A^T r_k\|_2$	$9.7189e-006$	$4.8337e-006$	$4.2110e-006$	$1.7751e-005$
well1033	CGLS	LSQR	LSMR	PCGLS
# de iterações	154	157	161	140
$\|e_k\|_2$	$3.0791e-005$	$1.5813e-005$	$1.4918e-004$	$1.9634e-005$
$\|r_k\|_2$	$7.6860e-006$	$4.2473e-006$	$1.9492e-006$	$1.4276e-005$
$\|A^T r_k\|_2$	$5.3717e-006$	$1.6972e-006$	$2.5648e-007$	$1.8393e-005$
well1850	CGLS	LSQR	LSMR	PCGLS
# de iterações	395	397	403	312
$\|e_k\|_2$	$1.5920e-004$	$1.4774e-004$	$2.0242e-004$	$2.1690e-004$
$\|r_k\|_2$	$1.2251e-005$	$1.1552e-005$	$1.1611e-005$	$1.2205e-005$
$\|A^T r_k\|_2$	$3.4020e-006$	$3.4199e-006$	$8.0873e-007$	$3.1401e-006$
lp_woodw	CGLS	LSQR	LSMR	PCGLS
# de iterações	5000	5000	5000	5000
$\|e_k\|_2$	$3.7123e-012$	$1.0428e-011$	$1.0226e-011$	$2.8796e-013$
$\|r_k\|_2$	$4.4692e-012$	$7.4607e-011$	$7.4593e-011$	$1.9063e-012$
$\|A^T r_k\|_2$	$6.1527e-008$	$9.8313e-007$	$9.8318e-007$	$2.2444e-008$

Tabela 5.2: Comparação entre CGLS, LSQR e LSMR para ash958, well1033, well1850 e lp_woodw.

5.4 Precondicionadores Baseados na Fatoração LU

Nesta seção veremos um método que determina o precondicionador a partir de um sub-bloco de A mais fatoração LU, e o primeiro a sugerir esse fato foi Läuchli [84].

Assuma $A \in \mathbb{R}^{m\times n}$, $m \geqslant n$, uma matriz posto completo cujas linhas foram permutadas de forma que $A = \begin{bmatrix} A_1 \\ A_2 \end{bmatrix}$, com $A_1 \in \mathbb{R}^{n\times n}$, seja não singular. Usamos A_1 como precondicionador à direita e, desta forma, considere o problema equivalente de quadrados mínimos

$$\min_y \|AA_1^{-1}y - b\|_2, \;\; y = A_1 x. \tag{5.4.1}$$

Läuchli computa as matrizes A_1^{-1} e $C = A_2A_1^{-1}$ utilizando eliminação gaussiana com pivoteamento completo e aplica os método dos gradientes conjugados sobre as equações normais desse problema de minimização. Läuchli não levou em conta esparsidade. A forma mais eficiente de computar a fatoração LU é utilizando um algoritmo de fatoração LU para matrizes esparsas [47, 50].

Particionando os vetores b e r conforme a partição feita em A, o problema de quadrados mínimos (5.4.1) se torna

$$\min_y \left\| \begin{bmatrix} I_n \\ C \end{bmatrix} y - \begin{bmatrix} b_1 \\ b_2 \end{bmatrix} \right\|_2$$

com $C = A_2A_1^{-1}$. A forma aumentada das equações normais é

$$\begin{bmatrix} I_n & 0 & I_n \\ 0 & I_{m-n} & C \\ I_n & C^T & 0 \end{bmatrix} \begin{bmatrix} r_1 \\ r_2 \\ y \end{bmatrix} = \begin{bmatrix} b_1 \\ b_2 \\ 0 \end{bmatrix}$$

e, eliminado y, obtemos

$$\begin{bmatrix} I_n & C^T \\ C & I_{m-n} \end{bmatrix} \begin{bmatrix} r_1 \\ r_2 \end{bmatrix} = \begin{bmatrix} 0 \\ b_2 - Cb_1 \end{bmatrix}.$$

Podemos resolver o sistema utilizando o método dos gradientes conjugados [35] ou continuar com a eliminação e eliminar r_2. Portanto, o sistema linear acima se torna, $(I+C^TC)r_1 = C^T(b_2-Cb_1)$, que é um sistema linear definido positivo. Adaptando o Algoritmo 26 obtemos

Algoritmo 30 Método dos Gradientes Conjugados: CGLS precondicionado (LU)

function x_*=PCGLSLU(C, A, b, x_0)
  $k = 0$; $r_0 = b - Ax_0$; $p_0 = r_{1_0} + C^T r_{2_0}$; $s_0 = p_0$; $\gamma_0 = \|s_0\|_2^2$;
  while $\gamma_k > 0$ **do**
    $q_k = Cp_k$
    $\mu_k = \gamma_k/(\|p_k\|_2^2 + \|q_k\|_2^2)$;
    $r_{1_{k+1}} = r_{1_k} - \mu_k p_k$;
    $r_{2_{k+1}} = r_{2_k} - \mu_k q_k$;
    $s_{k+1} = r_{1_{k+1}} + C^T r_{2_{k+1}}$;
    $\gamma_{k+1} = \|s_{k+1}\|_2^2$;
    $\tau_k = \gamma_{k+1}/\gamma_k$;
    $p_{k+1} = s_{k+1} + \tau_k p_k$;
    $k = k + 1$;
  end while
  $A_1 x_* = b_1 - r_1$;
end function

A convergência desse algoritmo foi discutida em [12, 43], e os autovalores de $I_n - C^T C$ são $\lambda_i = 1 + \sigma_i^2(C)$, $i = 1, \ldots, n$. Assim,

$$\kappa(AA_1^{-1}) = \frac{\sqrt{1 + \sigma_1^2(C)}}{\sqrt{1 + \sigma_i^2(C)}} \leqslant \sqrt{1 + \alpha^2},$$

onde $\alpha = \sigma_1^2(C) = \|C\|_2 = \|A_2 A_1^{-1}\|_2$. Portanto, a taxa de convergência é

$$\|r - r_k\|_2 \leqslant 2 \left(\frac{\alpha}{1 + \sqrt{1 + \alpha^2}} \right)^{2k} \|r - r_0\|_2.$$

5.5 Exercícios

1. À luz da dedução do método dos gradientes conjugados precondicionado deduza o CGLS precondicionado de modo a refletir o Algoritmo 25.

2. Implemente os algoritmos apresentados no capítulo e utilize o modelo well1033 encontrado no *Matrix Market*.[1] Compare a performance de cada algoritmo.

[1] `https://math.nist.gov/MatrixMarket/`

Bibliografia

[1] ANDREWS, G. E.; ASKEY, R.; ROY, R. *Special Functions*. New York: Cambridge University Press.

[2] ARFKEN, G. B.; WEBER, H. J. *Mathematical Methods for Physicists*. 6. ed. New York: Elsevier Academic Press, 2005.

[3] ARNOLDI, W. E. The Principle of Minimized Iterations in the Solution of the Matrix Eigenvalue Problem. *Quarterly of Applied Mathematics*, v. 9, n. 1, p. 17–29, 1951.

[4] ASHBY, S. F.; MANTEUFFEL, T. A.; SAYLOR, P. E. A Taxonomy for Conjugate Gradient Methods. *SIAM Journal on Numerical Analysis*, v. 27, n. 6, p. 1542–1568, 1990.

[5] AXELSSON, O. *Iterative Solution Methods*. New York: Cambridge University Press, 1996.

[6] BARZILAI, J.; BORWEIN, J. M. Two-point step size gradient methods. *IMA Journal of Numerical Analysis*, v. 8, n. 1, p. 141–148, 1988.

[7] BERMAN, A.; PLEMMONS, R. J. Cones and Iterative Methods for Best Least Squares Solutions of Linear Systems. *SIAM Journal on Numerical Analysis*, v. 11, n. 1, p. 145–154, 1974.

[8] BERTACCINI, D.; DURASTANTE, F. *Iterative Methods and Preconditioning for Large and Sparse Linear Systems with Applications*. New York: Chapman and Hall/CRC, 2018.

[9] BIRGIN, E.; CHAMBOULEYRON, I.; MARTÍNEZ, J. M. Estimation of the optical constants and the thickness of thin films using unconstrained optimization. *Computational Physics*, n. 151, p. 862–880, 1999.

[10] BJÖRCK, A. Solving Least Squares Problems by Gram-Schmidt Orthogonalization. *BIT*, v. 7, p. 1–21, 1967.

[11] BJÖRCK, A. A bidiagonalization Algorithm for Solving Large and Sparse Ill-Posed Systems of Linear Equations. *BIT Numerical Mathematics*, v. 28, p. 659–670, 1988.

[12] BJÖRCK, A. *Numerical Methods for Least Squares Problems*. Philadelphia: Society for Industrial and Applied Mathematics, 1996.

[13] BJÖRCK, A. *Numerical Methods in Matrix Computations*. New York: Springer, 2015. (Texts in Applied Mathematics 59).

[14] BJÖRCK, A.; ELFVING, T. Accelerated Projection Methods for Computing Pseudoinverse Solutions of Systems of Linear Equations. *BIT Numerical Mathematics*, v. 19, p. 145–163, 1979.

[15] BJÖRCK, A.; PAIGE, C. C. Loss and Recapture of Orthogonality in the Modified Gram-Schmidt Algorithm. *SIAM Journal on Matrix Analysis and Applications*, v. 13, n. 1, p. 176–190, 1992.

[16] BRADIE, B. *A Friendly Introduction to Numerical Analysis*. Upper Saddle River: Pearson Prentice Hall, 2006.

[17] CARVALHO, L. M.; GRATTON, S. *Avanços em Métodos de Krylov para Solução de Sistemas Lineares de Grande Porte*. 2. ed. São Carlos: Sociedade Brasileira de Matemática Aplicada e Computacional, 2012. (Notas em Matemática Aplicada).

[18] CARVALHO, L. M. et al. *Álgebra Linear Numérica e Computacional - Métodos de Krylov para a Solução de Sistemas Lineares*. Rio de Janeiro: Editora Ciência Moderna, 2010.

[19] CHAN, T. F. Algorithm 581: An Improved Algorithm for Computing the Singular Value Decomposition. *ACM Trans. Math. Softw.*, Association for Computing Machinery, New York, NY, USA, v. 8, n. 1, p. 84–88, 1982.

[20] CHAN, T. F. An Improved Algorithm for Computing the Singular Value Decomposition. *ACM Trans. Math. Softw.*, Association for Computing Machinery, New York, NY, USA, v. 8, n. 1, p. 72–83, 1982.

[21] CHANG, X.-W.; PAIGE, C. C.; TITLEY-PELOQUIN, D. Stopping Criteria for the Iterative Solution of Linear Least Squares Problems. *SIAM Journal on Matrix Analysis and Applications*, v. 31, n. 2, p. 831–852, 2009.

[22] CHEBYSHEV, P. L. Théorie des Mécanismes Connus sous le Nom de Parallélogrammes. *Mémoires Présentés a l'Académie Impériale des Sciences de St-Pétersbourg par Divers Savants, volume VII*, p. 537–568, 1854.

[23] CHEN, X.; LI, W. A Note on the Perturbation Bounds of Eigenspaces for Hermitian Matrices. *Journal of Computational and Applied Mathematics*, v. 196, n. 1, p. 338 – 346, 2006.

[24] CHEN, Y. *Iterative Methods for Linear Least Squares Problems*. Waterloo: Technical Report CS-75-04, University of Waterloo, Canada (1975), 1975.

[25] CIARLET, P. G. *Introduction to Numerical Linear Algebra and Optimisation*. Cambridge: Cambridge University Press, 1989.

[26] COELHO, F. U.; LOURENÇO, M. L. *Um Curso de Álgebra Linear*. 1. ed. São Paulo: Edusp, 2001.

[27] CONCUS, P.; GOLUB, G. H.; O'LEARY, D. P. A Generalized Conjugate Gradient Method for the Numerical Solution of Elliptic Partial Differential Equations. In: BUNCH, J. R.; ROSE, D. J. (Ed.). *Sparse Matrix Computations*. New York: Academic Press, 1976. p. 309–332.

[28] CULLUM, J.; WILLOUGHBY, R. A.; LAKE, M. A Lanczos Algorithm for Computing Singular Values and Vectors of Large Matrices. *SIAM Journal on Scientific and Statistical Computing*, v. 4, n. 2, p. 197–215, 1983.

[29] CUNHA, M. *Métodos Numéricos*. 2. ed. Campinas: Editora da Unicamp, 2000.

[30] DANIEL, J. et al. Reorthogonalization and Stable Algorithms for Updating the Gram-Schmidt QR Factorization. *Mathemathics of Computation*, v. 30, p. 772–795, 1976.

[31] DANIEL, J. W. Convergence of the Conjugate Gradient Method with Computationally Convenient Modifications. *Numerische Mathematik*, v. 10, p. 125–131, 1967.

[32] DANIEL, J. W. The Conjugate Gradient Method for Linear and Nonlinear Operator Equations. *SIAM Journal on Numerical Analysis*, v. 4, n. 1, p. 10–26, 1967.

[33] DAVIS, P. J. Orthonormalizing Codes in Numerical Analysis. In: *Survey of Numerical Analysis*. New York: McGraw-Hill Book Co., 1962. p. 558–584.

[34] DEMMEL, J. W. *Applied Numerical Linear Algebra*. Philadelfia: Society for Industrial and Applied Mathematics, 1997.

[35] EVANS, D. J.; LI, C. Numerical Aspects of the Generalized CG-method Applied to Least Squares Problems. *Computing*, v. 41, p. 171–178, 1989.

[36] FADDEEV, D.; FADDEEVA, V. *Computational Methods of Linear Algebra*. San Francisco: W.H.Freeman, 1963.

[37] FIGUEIREDO, M. A.; NOWAK, R.; WRIGHT, S. Projection for sparse reconstruction: application to compressed sensing and other inverse problems. *IEEE J. Sel. Top. Signal Process*, n. 1, p. 586–597, 2007.

[38] FLETCHER, R. Conjugate Gradient Methods for Indefinite Systems. In: WATSON, G. (Ed.). *Numerical analysis*. New York: Springer, 1976, (Lecture Notes in Mathematics, v. 506). p. 73–89.

[39] FLETCHER, R.; POWELL, M. J. D. A Rapidly Convergent Descent Method for Minimization. *The Computer Journal*, v. 6, n. 2, p. 163–168, 1963.

[40] FLETCHER, R.; REEVES, C. M. Function Minimization by Conjugate Gradients. *The Computer Journal*, Oxford University Press, v. 7, n. 2, p. 149–154, 1964.

[41] FONG, D. C.-L.; SAUNDERS, M. LSMR: An Iterative Algorithm for Sparse Least-Squares Problems. *SIAM Journal on Scientific Computing*, v. 33, n. 5, p. 2950–2971, 2011.

[42] FRANKEL, S. P. Convergence Rates of Iterative Treatments of Partial Differential Equations. *Mathematical Tables and Other Aids to Computation*, American Mathematical Society, v. 4, n. 30, p. 65–75, 1950.

[43] FREUND, R. A Note on Two Block-SOR Methods for Sparse Least Squares Problems. *Linear Algebra and its Applications*, v. 88-89, p. 211–221, 1987.

[44] FREUND, R. W.; GOLUB, G. H.; NACHTIGAL, N. M. Iterative Solution of Linear Systems. *Acta Numerica*, Cambridge University Press, v. 1, p. 57–100, 1992.

[45] GANDER, W.; MOLINARI, L.; ŠVECOVÁ, H. *Numerische Prozeduren aus Nachlass und Lehre von Heinz Rutishauser*. Basel-Stuttgart: Birkhauser, 1977.

[46] GARZA, A. de la. An Iterative Method for Solving Systems of Linear Equations. Report K-731. Oak Ridge Gaseous Diffusion Plant, Oak Ridge, 1951.

[47] GEORGE, A.; NG, E. An Implementation of Gaussian Elimination with Partial Pivoting for Sparse Systems. *SIAM Journal on Scientific and Statistical Computing*, v. 6, n. 2, p. 390–409, 1985.

[48] GILBERT, J. C.; NOCEDAL, J. Global Convergence Properties of Conjugate Gradient Methods for Optimization. *SIAM Journal on Optimization*, v. 2, n. 1, p. 21–42, 1992.

[49] GILBERT, J. R. Predicting Structure in Sparse Matrix Computations. *SIAM Journal on Matrix Analysis and Applications*, v. 15, n. 1, p. 62–79, 1994.

[50] GILBERT, J. R.; MOLER, C.; SCHREIBER, R. Sparse Matrices in Matlab: Design and Implementation. *SIAM J. Matrix Anal. Appl.*, Society for Industrial and Applied Mathematics, USA, v. 13, n. 1, p. 333–356, 1992.

[51] GIRAUD, L.; LANGOU, J.; ROZLOZNIK, M. The Loss of Orthogonality in the Gram-Schmidt Orthogonalization Process. *Computers & Mathematics with Applications*, v. 50, n. 7, p. 1069 – 1075, 2005.

[52] GOLUB, G.; KAHAN, W. Calculating the Singular Values and Pseudo-Inverse of a Matrix. *Journal of the Society for Industrial and Applied Mathematics: Series B, Numerical Analysis*, Society for Industrial and Applied Mathematics, v. 2, n. 2, p. 205–224, 1965.

[53] GOLUB, G.; VARGA, R. Chebyshev Semi-Iterative Methods, Successive Overrelaxation Iterative Methods, and Second Order Richardson Iterative Methods - Part I. *Numerische Mathematik*, v. 3, p. 147–156, 1961.

[54] GOLUB, G.; VARGA, R. Chebyshev Semi-Iterative Methods, Successive Overrelaxation Iterative Methods, and Second Order Richardson Iterative Methods - Part II. *Numerische Mathematik*, v. 3, p. 157–168, 1961.

[55] GOLUB, G. H.; LUK, F. T.; OVERTON, M. L. A Block Lanczos Method for Computing the Singular Values and Corresponding Singular Vectors of a Matrix. *ACM Trans. Math. Softw.*, Association for Computing Machinery, New York, NY, USA, v. 7, n. 2, p. 149–169, 1981.

[56] GOLUB, G. H.; O'LEARY, D. P. Some History of the Conjugate Gradient and Lanczos Algorithms: 1948-1976. *SIAM Review*, v. 31, n. 1, p. 50–102, 1989.

[57] GOLUB, G. H.; UNDERWOOD, R. R.; WILKINSON, J. H. *The Lanczos Algorithm for the Symmetric $Ax = \lambda Bx$ Problem*. Stanford, CA, USA, 1972.

[58] GOLUB, G. H.; VAN LOAN, C. F. *Matrix Computations*. 4. ed. Baltimore: Johns Hopkins University Press, 2013. (Johns Hopkins Studies in the Mathematical Sciences).

[59] GRAM, J. Über die Entwickelung reeller Functionen in Reihen mittelst der Methode der kleinsten Quadrate. *Journal für die Reine und Angewandte Mathematik*, v. 94, p. 41–73, 1883.

[60] GRCAR, J. F. Spectral Condition Numbers of Orthogonal Projections and Full Rank Linear Least Squares Residuals. *SIAM Journal on Matrix Analysis and Applications*, v. 31, n. 5, p. 2934–2949, 2010.

[61] GREENBAUM, A. *Iterative Methods for Solving Linear Systems*. Philadelphia: Society for Industrial and Applied Mathematics, 1997.

[62] HADJIDIMOS, A. Successive Overrelaxation (SOR) and Related Methods. *Journal of Computational and Applied Mathematics*, v. 123, n. 1, p. 177–199, 2000. Numerical Analysis 2000. Vol. III: Linear Algebra.

[63] HESTENES, M. Iterative Methods for Solving Linear Equations. *Journal of Optimization Theory and Applications*, v. 11, p. 323 – 334, 1973.

[64] HESTENES, M. R.; STIEFEL, E. Methods of Conjugate Gradients for Solving Linear Systems. *J Res NIST*, v. 49, n. 6, p. 409–436, 1952.

[65] HIGHAM, N. J. Estimating the Matrix p-Norm. *Numerische Mathematik*, v. 62, p. 539–555, 1992.

[66] HIGHAM, N. J. *Accuracy and Stability of Numerical Algorithms*. 2. ed. Philadelfia: Society for Industrial Mathematics, 2002.

[67] HOFFMAN, K.; KUNZE, R. *Álgebra Linear*. 1. ed. São Paulo: Editora Polígono, 1971.

[68] HORN, R. A.; JOHNSON, C. R. *Matrix Analysis*. 2. ed. New York: Cambridge University Press, 2013.

[69] HOUSEHOLDER, A.; BAUER, F. On Certain Iterative Methods for Solving Linear Systems. *Numerische Mathematik*, v. 2, p. 55–59, 1960.

[70] HOUSEHOLDER, A. S. *The Theory of Matrices in Numerical Analysis*. New York: Blaisdell Pub. Co, 1964.

[71] JENNINGS, A.; AJIZ, M. A. Incomplete Methods for Solving $A^T Ax = b$. *SIAM J. Sci. Stat. Comput.*, Society for Industrial and Applied Mathematics, USA, v. 5, n. 4, p. 978–987, 1984.

[72] KAHAN, W. V. *Gauss-Seidel Methods of Solving Large Systems of Linear Equations*. Tese (Doutorado) — University of Toronto, Canada, 1958.

[73] KANIEL, S. Estimates for Some Computational Techniques in Linear Algebra. *Mathematics of Computation*, v. 20, p. 369–378, 1966.

[74] KANTOROVICH, L. V. Functional Analysis and Applied Mathematics (em Russo). *Uspekhi Mat. Nauk*, v. 3, p. 89–185, 1948.

[75] KATS, I. The Convergence Rate of the Method of Successive Over Relaxation. *USSR Computational Mathematics and Mathematical Physics*, v. 9, n. 5, p. 15 – 26, 1969.

[76] KELLER, H. B. On the Solution of Singular and Semidefinite Linear Systems by Iteration. *SIAM J. Numerical Analysis*, SIAM, v. 2, n. 2, p. 281–290, 1965.

[77] KELLEY, C. T. *Iterative Methods for Linear and Nonlinear Equations*. Philadelfia: Society for Industrial and Applied Mathematics, 1995.

[78] KERSHAW, D. S. The Incomplete Cholesky-Conjugate Gradient Method for the Iterative Solution of Systems of Linear Equations. *Journal of Computational Physics*, v. 26, n. 1, p. 43–65, 1978.

[79] KRYLOV, A. On the Numerical Solution of the Equation by Which in Technical Questions Frequencies of Small Oscillations of Material Systems Are Determined (em Russo). *Izvestija AN SSSR (News of Academy of Sciences of the USSR), Otdel. mat. i estest. nauk*, v. 7, n. 4, p. 491–539, 1931.

[80] LANCZOS, C. An Iteration Method for the Solution of the Eigenvalue Problem of Linear Differential and Integral Operators. *J. Res. Natl. Bur. Stand. B*, v. 45, p. 255–282, 1950.

[81] LANCZOS, C. Solution of Systems of Linear Equations by Minimized Iterations. *J. Res. Nat. Bur. Standards*, v. 49, n. 1, p. 33–53, 1952.

[82] LANDWEBER, L. An Iteration Formula for Fredholm Integral Equations of the First Kind. *American Journal of Mathematics*, v. 73, n. 3, p. 615–624, 1951.

[83] LÄUCHLI, P. Iterative Lösung und Fehlerabschätzung in der Ausgleichsrechnung. *Zeitschrift für Angewandte Mathematik und Physik (ZAMP)*, Springer, v. 10, p. 245–280, 1959.

[84] LÄUCHLI, P. Jordan-Elimination und Ausgleichung nach kleinsten Quadraten. *Numerische Mathematik*, v. 3, p. 226–240, 1961.

[85] LAWSON, C. L. *Sparse Matrix Methods Based on Orthogonality and Conjugacy*. Pasadena: Technical Memorandum 33-627, Jet Propulsion Laboratory, California Institute of Technology, 1973.

[86] LAWSON, C. L.; HANSON, R. J. *Solving Least Squares Problems*. Philadelfia: Society for Industrial and Applied Mathematics, 1995.

[87] LEGENDRE, A. *Nouvelles Méthodes pour la Détermination des Orbites des Comètes*. Paris: F. Didot, 1805.

[88] LEON, S. J.; BJÖRCK, A.; GANDER, W. Gram-Schmidt Orthogonalization: 100 Years and More. *Numerical Linear Algebra with Applications*, v. 20, n. 3, p. 492–532, 2013.

[89] LIEBMANN, H. Die Angenäherte Ermittelung Harmonischer Funktionen und Konformer Abbildungen. (Nach Ideen von Boltzmann und Jacobi). *S. B. Math. Nat. Kl. Bayerischen Akad. Wiss. München*, p. 85–416, 1918.

[90] LIMA, E. L. *Curso de Análise*. 11. ed. Rio de Janeiro: IMPA, 2014. (Projeto Euclides).

[91] LUENBERGER, D. G.; YE, Y. *Linear and Nonlinear Programming*. 4. ed. New York: Springer, 2016. (International Series in Operations Research & Management Science 228).

[92] MALYSHEV, A. N. A Unified Theory of Conditioning for Linear Least Squares and Tikhonov Regularization Solutions. *SIAM Journal on Matrix Analysis and Applications*, v. 24, n. 4, p. 1186–1196, 2003.

[93] MASON, J.; HANDSCOMB, D. C. *Chebyshev Polynomials*. Boca Raton: Chapman and Hall/CRC, 2003.

[94] MEIJERINK, J. A.; VAN DER VORST, H. A. An Iterative Solution Method for Linear Systems of Which the Coefficient Matrix is a Symmetric M-matrix. *Mathematics of Computation*, v. 31, p. 148–162, 1977.

[95] MEINARDUS, G. Über eine Verallgemeinerung einer Ungleichung von L. V. Kantorowitsch. *Numerische Mathematik*, v. 5, p. 14–23, 1963.

[96] MENG, L.; ZHENG, B. The Optimal Perturbation Bounds of the Moore-Penrose Inverse under the Frobenius Norm. *Linear Algebra and its Applications*, v. 432, n. 4, p. 956 – 963, 2010.

[97] MEURANT, G.; STRAKOŠ, Z. The Lanczos and Conjugate Gradient Algorithms in Finite Precision Arithmetic. *Acta Numerica*, v. 15, p. 471 – 542, 2006.

[98] NACHTIGAL, N. M.; REDDY, S. C.; TREFETHEN, L. N. How Fast are Nonsymmetric Matrix Iterations? *SIAM Journal on Matrix Analysis and Applications*, v. 13, n. 3, p. 778–795, 1992.

[99] NÉKRASSOV, P. Détermination des Inconnues par la Méthode de Moindres Carrés dans le cas où le Nombre D'inconnues est Considérable. *Mat. Sb*, v. 12, p. 189–204, 1885.

[100] NIEVERGELT, Y. A tutorial History of Least Squares with Applications to Astronomy and Geodesy. *Journal of Computational and Applied Mathematics*, v. 121, n. 1, p. 37 – 72, 2000.

[101] OLSHANSKII, M. A.; TYRTYSHNIKOV, E. E. *Iterative Methods for Linear Systems: Theory and Applications*. Philadelphia: Society for Industrial and Applied Mathematics, 2014.

[102] OLVER, P. J.; SHAKIBAN, C. *Applied Linear Algebra*. 2. ed. New York: Springer, 2018. (Undergraduate texts in mathematics).

[103] ORTEGA, J. M. *Numerical Analysis: a Second Course*. Philadelphia: Society for Industrial and Applied Mathematics, 1987.

[104] ORTEGA, J. M.; PLEMMONS, R. J. Extensions of the Ostrowski-Reich Theorem for SOR Iterations. *Linear Algebra and its Applications*, v. 28, p. 177 – 191, 1979.

[105] OSTROWSKI, A. M. On the Linear Iteration Procedures for Symmetric Matrices. *Rend. Mat. Appl.*, v. 14, p. 140–163, 1954.

[106] OVIEDO-LEON, H. F. A delayed weighted gradient method for strictly convex quadratic minimization. *Computational Optimization and Applications*, v. 74, p. 729–746, 2019.

[107] PAIGE, C. C. Computational Variants of the Lanczos Method for the Eigenproblem. *IMA Journal of Applied Mathematics*, v. 10, n. 3, p. 373–381, 1972.

[108] PAIGE, C. C.; SAUNDERS, M. A. Solution of Sparse Indefinite Systems of Linear Equations. *SIAM Journal on Numerical Analysis*, v. 12, n. 4, p. 617–629, 1975.

[109] PAIGE, C. C.; SAUNDERS, M. A. Algorithm 583: LSQR: Sparse Linear Equations and Least Squares Problems. *ACM Trans. Math. Softw.*, Association for Computing Machinery, New York, NY, USA, v. 8, n. 2, p. 195–209, 1982.

[110] PAIGE, C. C.; SAUNDERS, M. A. LSQR: An Algorithm for Sparse Linear Equations and Sparse Least Squares. *ACM Trans. Math. Softw.*, v. 8, n. 1, p. 43–71, 1982.

[111] PENROSE, R. A Generalized Inverse for Matrices. *Mathematical Proceedings of the Cambridge Philosophical Society*, Cambridge University Press, v. 51, n. 3, p. 406–413, 1955.

[112] PILKINGTON, M. 3-D Magnetic Imaging Using Conjugate Gradients. *Geophysics*, Society of Exploration Geophysicists, v. 62, n. 4, p. 1132–1142, 1997.

[113] PLACKETT, R. L. Studies in the History of Probability and Statistics. XXIX: The Discovery of the Method of Least Squares. *Biometrika*, [Oxford University Press, Biometrika Trust], v. 59, n. 2, p. 239–251, 1972.

[114] PLEMMONS, R. J. Monotonicity and Iterative Approximations Involving Rectangular Matrices. *Mathematics of Computation*, v. 26, n. 120, p. 853–858, 1972.

[115] POOLE, G.; BOULLION, T. A Survey on M-Matrices. *SIAM Review*, Society for Industrial and Applied Mathematics, v. 16, n. 4, p. 419–427, 1974.

[116] RAINVILLE, E. *Special Functions*. New York: The Macmillan Co., 1960.

[117] REICH, E. On the Convergence of the Classical Iterative Procedures for Symmetric Matrices. *Ann. Math. Statist*, v. 20, p. 448–451, 1949.

[118] REID, J. On the Method of Conjugate Gradients for the Solution of Large Sparse Systems of Linear Equations. In: REID, J. (Ed.). *Proceedings*

of Large Sparse Sets of Linear Equations Conference. New York: Academic Press, 1971. p. 231–254.

[119] RICHARDSON, L. F. The Aapproximate Arithmetical Solution by Finite Differences of Physical Problems Involving Differential Equations, With an Application to the Stresses in a Masonry Dam. *Philosophical Transactions of the Royal Society of London*, v. 210, p. 307–357, 1911.

[120] RIVLIN, T. J. *The Chebyshev Polynomials: From Approximation Theory to Algebra and Number Theory*. New York: John Wiley and Sons Inc, 1974.

[121] RODI, W.; MACKIE, R. L. Nonlinear Conjugate Gradients Algorithm for 2-D Magnetotelluric Inversion. *Geophysics*, Society of Exploration Geophysicists, v. 66, n. 1, p. 174–187, 2001.

[122] RUTISHAUSER, H. *Description of ALGOL 60, Handbook for Automatic Computation*. Berlin: Springer, 1967.

[123] RUTISHAUSER, H. Simultaneous Iteration Method for Symmetric Matrices. In: *Handbook for Automatic Computation*. Berlin: Springer, 1971. II: Linear Algebra.

[124] SAAD, Y. Preconditioning Techniques for Nonsymmetric and Indefinite Linear Systems. *Journal of Computational and Applied Mathematics*, v. 24, n. 1, p. 89–105, 1988.

[125] SAAD, Y. *Iterative Methods for Sparse Linear Systems*. 2. ed. Philadelphia: Society for Industrial and Applied Mathematics, 2003.

[126] SAAD, Y. *Iterative Methods for Linear Systems of Equations: A brief Historical Journey*. 2019.

[127] SAAD, Y.; SCHULTZ, M. H. GMRES: A Generalized Minimal Residual Algorithm for Solving Nonsymmetric Linear Systems. *SIAM Journal on Scientific and Statistical Computing*, v. 7, n. 3, p. 856–869, 1986.

[128] SCHEICK, J. T. *Linear Algebra with Applications*. New York: McGraw-Hill, 1997. (International series in pure and applied mathematics).

[129] SCHMIDT, E. Zur Theorie der Linearen und Nichtlinearen Integralgleichungen. I. Teil: Entwicklung WillkÃ¼rlicher Funktionen nach Systemen Vorgeschriebener. *Mathematische Annalen*, v. 63, p. 433–476, 1907.

[130] SERAFINI, T.; ZANGHIRATI, G.; ZANNI, L. Gradient projection methods for large quadratic programs and applications in training support vector machines. *Optimization Methods and Software*, n. 20, p. 353–378, 2005.

[131] SHELDON, J. W. On the Numerical Solution of Elliptic Difference Equations. *Mathematics of Computation*, v. 9, p. 101–112, 1955.

[132] SHEWCHUK, J. R. *An Introduction to the Conjugate Gradient Method without the Agonizing Pain*. USA, 1994.

[133] SMOKTUNOWICZ, A.; BARLOW, J.; LANGOU, J. A Note on the Error Analysis of Classical Gram-Schmidt. *Numerische Mathematik*, v. 105, n. 2, p. 299–313, 2006.

[134] SONNEVELD, P. CGS, A Fast Lanczos-Type Solver for Nonsymmetric Linear systems. *SIAM Journal on Scientific and Statistical Computing*, v. 10, n. 1, p. 36–52, 1989.

[135] STEWART, G. Research, Development, and LINPACK. In: RICE, J. R. (Ed.). *Mathematical Software*. New York: Academic Press, 1977. p. 1–14.

[136] STEWART, G. W. On the Perturbation of Pseudo-Inverses, Projections and Linear Least Squares Problems. *SIAM Review*, v. 19, n. 4, p. 634–662, 1977.

[137] STEWART, G. W. The QLP Approximation to the Singular Value Decomposition. *SIAM J. Sci. Comput.*, Society for Industrial and Applied Mathematics, USA, v. 20, n. 4, p. 1336–1348, 1999.

[138] STEWART, G. W. *Matrix Algorithms*. Philadelfia: SIAM: Society for Industrial and Applied Mathematics, 2001.

[139] STEWART, G. W. On the Numerical Analysis of Oblique Projectors. *SIAM Journal on Matrix Analysis and Applications*, v. 32, n. 1, p. 309–348, 2011.

[140] STIEFEL, E. Ausgleichung ohne Aufstellung der Gausschen Normalgleichungen. *Wiss. Z. Technische Hochschule Dresden*, v. 2, n. 441-442, 1952/53.

[141] STIEFEL, E. L. Kernel Polynomial in Linear Algebra and Their Numerical Applications, in: Further Contributions to the Determination of Eigenvalues. *U.S. National Bureau of Standards, Applied Mathematics Series*, v. 49, p. 1–22, 1958.

[142] STIELTJES, T. J. Sur les Racines de L'équation $x_n = 0$. *Acta Mathematica*, v. 9, p. 385–400, 1887.

[143] STIGLER, S. M. Gauss and the invention of least squares. *Annals of Statistics*, The Institute of Mathematical Statistics, v. 9, n. 3, p. 465–474, 1981.

[144] STOER, J.; BULIRSCH, R. *Introduction to Numerical Analysis*. 2. ed. New York: Springer, 1996. (Texts in Applied Mathematics, No 12).

[145] SUN, X.; GE, Z.; LI, Z. Conjugate Gradient and Cross-Correlation Based Least-Square Reverse Time Migration and its Application. *Applied Geophysics*, v. 14, p. 381–386, 2017.

[146] TANABE, K. Projection Method for Solving a Singular System of Linear Equations and its Applications. *Numerische Mathematik*, v. 17, p. 203–214, 1971.

[147] TREFETHEN, L. N.; Bau III, D. *Numerical Linear Algebra*. Philadelfia: Society for Industrial and Applied Mathematics, 1997.

[148] VAN DER SLUIS, A.; VAN DER VORST, H. The Rate of Convergence of Conjugate Gradients. *Numerische Mathematik*, v. 48, p. 543–560, 1986.

[149] VAN DER SLUIS, A.; VELTKAMP, G. Restoring Rank and Consistency by Orthogonal Projection. *Linear Algebra and its Applications*, v. 28, p. 257 – 278, 1979.

[150] VAN DER VORST, H. A. Bi-CGSTAB: A Fast and Smoothly Converging Variant of Bi-CG for the Solution of Nonsymmetric Linear Systems. *SIAM Journal on Scientific and Statistical Computing*, v. 13, n. 2, p. 631–644, 1992.

[151] VAN DER VORST, H. A. *Iterative Krylov Methods for Large Linear Systems*. New York: Cambridge University Press, 2003. (Cambridge Monographs on Applied and Computational Mathematics).

[152] VARGA, R. S. *Matrix Iterative Analysis*. 2. ed. New York: Springer-Verlag, 2000. (Springer Series in Computational Mathematics 27).

[153] VAZ JR., J.; OLIVEIRA, E. C. de. *Métodos Matemáticos*. Campinas: Editora da Unicamp, 2016.

[154] WALKER, H. F. Implementation of the GMRES Method Using Householder Transformations. *SIAM Journal on Scientific and Statistical Computing*, v. 9, n. 1, p. 152–163, 1988.

[155] WANG, X.; GALLIVAN, K. A.; BRAMLEY, R. CIMGS: An Incomplete Orthogonal Factorization Preconditioner. *SIAM J. Sci. Comput.*, Society for Industrial and Applied Mathematics, USA, v. 18, n. 2, p. 516–536, 1997.

[156] WATKINS, D. *Fundamentals of Matrix Computations*. New York: John Wiley and Sons, 1991.

[157] WEDIN, P. Perturbation Theory for Pseudo-Inverses. *BIT*, v. 13, p. 217–232, 1973.

[158] WENDLAND, H. *Numerical Linear Algebra: An Introduction*. Cambridge: Cambridge University Press, 2017. (Cambridge Texts in Applied Mathematics).

[159] WONG, Y. K. An Application of Orthogonalization Process to the Theory of Least Squares. *Annals of Mathematical Statistics*, The Institute of Mathematical Statistics, v. 6, n. 2, p. 53–75, 1935.

[160] YOUNG, D. *Iterative Methods for Solving Partial Differential Equations of Elliptic Type*. Tese (Doutorado) — Harvard University, Cambridge, MA, 1950.

[161] YOUNG, D. Iterative Methods for Solving Partial Difference Equations of Elliptic Type. *Transactions of the American Mathematical Society*, v. 76, n. 1, p. 92–111, 1954.

[162] YOUNG, D. M. Convergence Properties of the Symmetric and Unsymmetric Successive Overrelaxation Methods and Related Methods. *Mathematics of Computation*, American Mathematical Society, v. 24, p. 793–807, 1970.

[163] YUAN, J.-Y. The Ostrowski-Reich Theorem for SOR Iterations: Extensions to the Rank Deficient Case. *Linear Algebra and its Applications*, v. 315, n. 1, p. 189 – 196, 2000.

[164] ZHANG, L. New Versions of the Hestenes-Stiefel Nonlinear Conjugate Gradient Method Based on the Secant Condition for Optimization. *Computational & Applied Mathematics*, v. 28, p. 111 – 133, 2009.

[165] ZILL, D. G.; SHANAHAN, P. D. *A First Course in Complex Analysis with Applications*. New York: Jones and Bartlett Publishers, 2003.

[166] ZLATEV, Z.; NIELSEN, H. Solving Large and Sparse Linear Least-Squares Problems by Conjugate Gradient Algorithms. *Computers & Mathematics with Applications*, v. 15, n. 3, p. 185–202, 1988.

Índice