Matemática Numérica I

Pedro H A Konzen

10 de abril de 2023

Licença

Prefácio

Nestas notas de aula são abordados temas introdutórios de matemática numérica. Como ferramenta computacional de apoio didático, apresentam-se códigos em `Python`.

Agradeço a todos e todas que de modo assíduo ou esporádico contribuem com correções, sugestões e críticas. :)

Pedro H A Konzen

Conteúdo

Capítulo 1

Aritmética de Máquina

```
>>> 0.1 + 0.2 == 0.3
False
```

1.1 Sistema de Numeração Posicional

[YouTube] | [Vídeo] | [Áudio] | [Contatar]

Cotidianamente, usamos o sistema de numeração posicional na base decimal. Por exemplo, temos

$$123{,}5 = 1 \times 10^2 + 2 \times 10^1 + 3 \times 10^0 + 5 \times 10^{-1}, \tag{1.1}$$

onde o algarismo/dígito 1 está na posição 2 (posição das centenas), o dígito 2 está na posição 1 (posição das dezenas) e o dígito 3 está na posição 0 (posição das unidades). Mais geralmente, temos a representação decimal

$$\begin{aligned} &\pm d_n \dots d_2 d_1 d_0{,}d_{-1} d_{-2} d_{-3} \dots \qquad (1.2)\\ &:= \pm\Big(d_n \times 10^n + \cdots + d_2 \times 10^2 + d_1 \times 10^1 + d_0 \times 10^0 \qquad (1.3)\\ &\quad + d_{-1} \times 10^{-1} + d_{-2} \times 10^{-2} + d_{-3} \times 10^{-3} + \cdots\Big), \qquad (1.4) \end{aligned}$$

cujos os dígitos $d_i \in \{0, 1, 2, 3, 4, 5, 6, 7, 8, 9\}$, $i = n, \dots, 2, 1, 0, -1, -2, -3, \dots$. Observamos que esta representação posicional pode ser generalizada para outras bases numéricas.

Definição 1.1.1. (Representação posicional) Dada uma base $b \in \mathbb{N} \setminus \{0\}$, definimos a representação

$$\begin{aligned}
&\pm(d_n \ldots d_2 d_1 d_0{,}d_{-1} d_{-2} d_{-3} \ldots)_b && (1.5)\\
&:= \pm\Big(d_n \times b^n + \cdots + d_2 \times b^2 + d_1 \times b^1 + d_0 \times b^0 && (1.6)\\
&\quad + d_{-1} \times b^{-1} + d_{-2} \times b^{-2} + d_{-3} \times b^{-3} + \cdots\Big), && (1.7)
\end{aligned}$$

onde os dígitos $d_i \in \{0, 1, \ldots, b-1\}$[1], $i = n, \ldots, 2, 1, 0, -1, -2, -3, \ldots$.

Exemplo 1.1.1. (Representação binária) O número $(11010{,}101)_2$ está escrito na representação binária (base $b = 2$). Da Definição 1.1.1, temos

$$\begin{aligned}
&(\overset{4}{1}\,\overset{3}{1}\,\overset{2}{0}\,\overset{1}{1}\,\overset{0}{0}\,,\ \overset{-1}{1}\ \overset{-2}{0}\ \overset{-3}{1})_2 && (1.8)\\
&= 1 \times 2^4 + 1 \times 2^3 + 0 \times 2^2 + 1 \times 2^1 + 0 \times 2^0 && (1.9)\\
&\quad + 1 \times 2^{-1} + 0 \times 2^{-2} + 1 \times 2^{-3} && (1.10)\\
&= 26{,}625. && (1.11)
\end{aligned}$$

```
>>> 1*2**4 + 1*2**3 + 0*2**2 + 1*2**1 + 0*2**0 + \
... 1*2**-1 + 0*2**-2 + 1*2**-3
26.625
```

1.1.1 Mudança de Base

Um mesmo número pode ser representado em diferentes bases. A mudança de base da representação de um dado número pode ser feita de várias formas. De forma geral, se temos um número x representado na base b_1 e queremos obter sua representação na base b_2, fazemos

1. Calculamos a representação do número x na base decimal.
2. Da calculada representação decimal, calculamos a representação de x na base b_2.

Observamos que o passo 1. ($b \to 10$) segue imediatamente da Definição 1.1.1. Agora, o passo 2. ($10 \to b$), podemos usar o seguinte procedimento. Suponhamos que x tenha a seguinte representação decimal

$$d_n d_{n-1} d_{n-2} \ldots d_0{,}d_{-1} d_{-2} d_{-3} \ldots \qquad (1.12)$$

[1]Para bases $b \geq 11$, usamos a representação dos dígitos maiores ou iguais a 10 por letras maiúsculas do alfabeto latino, i.e. $A = 10$, $B = 11$, $C = 12$ e assim por diante.

Então, separamos sua parte inteira $I = d_n d_{n-1} d_{n-2} \dots d_0$ e sua parte fracionária $F = 0{,}d_{-1}d_{-2}d_{-3}\dots$ ($x = I + F$). Então, usando de sucessivas divisões de I pela base b desejada, obtemos sua representação nesta mesma base. Analogamente, usando de sucessivas multiplicações de F pela base b, obtemos sua representação nesta base. Por fim, basta somar as representações calculadas.

Exemplo 1.1.2. Obtenha a representação em base quartenária ($b = 4$) do número $(11010{,}101)_2$.

1. $b = 2 \to 10$. A representação de $(11010{,}101)_2$ segue direto da Definição 1.1.1 (veja, o Exemplo 1.1.1). Ou seja, temos

$$\begin{align}
&(\overset{4}{1}\,\overset{3}{1}\,\overset{2}{0}\,\overset{1}{1}\,\overset{0}{0}\,,\ \overset{-1}{1}\ \overset{-2}{0}\ \overset{-3}{1})_2 \tag{1.13}\\
&= 2^4 + 2^3 + 2^1 + 2^{-1} + 2^{-3} \tag{1.14}\\
&= 26{,}625. \tag{1.15}
\end{align}$$

```
>>> 2**4 + 2**3 + 2 + 2**-1 + 2**-3
26.625
```

2. $b = 10 \to 4$. Primeiramente, decompomos 26,625 em sua parte inteira $I = 26$ e em sua parte fracionária 0,625. Então, ao fazermos sucessivas divisões de I por $b = 4$, obtemos:

$$\begin{align}
I &= 26 \tag{1.16}\\
&= 6 \times 4 + 2 \times 4^0 \tag{1.17}\\
&= (1 \times 4 + 2) \times 4 + 2 \times 4^0 \tag{1.18}\\
&= 1 \times 4^2 + 2 \times 4 + 2 \times 4^0 \tag{1.19}\\
&= (122)_4. \tag{1.20}
\end{align}$$

```
I = int(26.625)
d_int = []
while (I != 0):
  d_int.insert(0, I % 4)
  I //= 4
print('(',*d_int,')_4',sep="")
```

Agora, para a parte fracionária, usamos sucessivas multiplicações de F por $b = 4$, obtendo:

$$
\begin{align}
F &= 0{,}625 \tag{1.21}\\
&= 2{,}5 \times 4^{-1} = 2 \times 4^{-1} + 0{,}5 \times 4^{-1} \tag{1.22}\\
&= 2 \times 4^{-1} + 2 \times 4^{-1} \times 4^{-1} \tag{1.23}\\
&= 2 \times 4^{-1} + 2 \times 4^{-2} \tag{1.24}\\
&= (0{,}22)_4. \tag{1.25}
\end{align}
$$

```
F = 26.625 % 1
d_fra = []
while (F != 0):
  F *= 4
  d_fra.append(int(F))
  F %= 1
print('(0,',*d_fra,')_4',sep="")
```

Por fim, dos passos 1. e 2., temos $(11010{,}101)_2 = (122{,}22)_4$.

```
>>> print('(',*d_int,',',*d_fra,')_4', sep='')
(122,22)_4
```

1.1.2 Exercícios Resolvidos

ER 1.1.1. Forneça a representação decimal dos seguintes números:

a) $(10101)_2$

b) $(0{,}4321)_5$

c) $(23{,}5)_8$

d) $(A2A)_{11}$

e) $(BEBE)_{16}$

Solução.

a) $(\overset{4}{1}\,\overset{3}{0}\,\overset{2}{1}\,\overset{1}{0}\overset{0}{1})_2$

```
>>> 0b10101
21
```

b) $(\overset{0}{0},\overset{-1}{4}\,\overset{-2}{3}\,\overset{-3}{2}\,\overset{-4}{1})_5$

```
>>> 4*5**-1+3*5**-2+2*5**-3+5**-4
0.9376000000000001
```

c) $(\overset{1}{2}\,\overset{0}{3},\overset{-1}{5})_8$

```
>>> 0o235 / 8**1
19.625
```

d) $(\overset{2}{A}\,\overset{1}{2}\,\overset{0}{A})_{11}$

```
>>> int('A2A', 11)
1242
```

e) $(\overset{3}{B}\,\overset{2}{E}\,\overset{1}{B}\,\overset{0}{E})_{16}$

```
>>> 0xBEBE
48830
```

◊

ER 1.1.2. Forneça a representação na base indicada dos seguintes números decimais:

a) 203 → base 2

b) 0,671875 → base 2

c) 17,25 → base 8

d) 3245,875 → base 16

Solução.

a) 203 → base 2 Usando o método Python **bin**, obtemos

```
>>> bin(203)
'0b11001011'
```

ou seja, $203 = (11001011)_2$.

b) 0,671875 → base 2. Executando o código

```
F = 0.671875
digs = []
while (F != 0):
  F *= 2
  digs.append(int(F))
  F %= 1
print('(0,',*digs,')_2',sep="")
```

obtemos que $0{,}671875 = (0{,}101011)_2$.

c) $17{,}25 \to$ base 8 Temos que

$$
\begin{align}
17{,}25 &= 17 + 0{,}25 \tag{1.26}\\
&= 16 + 1 + \frac{2}{8} \tag{1.27}\\
&= 2\cdot 8^1 + 1\cdot 8^0 + 2\cdot 8^{-1} \tag{1.28}\\
&= (21{,}2)_8 \tag{1.29}
\end{align}
$$

d) $3245{,}875 \to$ base 16 Executando o seguinte código

```
# base
b = 16
# dígitos
digs = "0123456789ABCDEF"

# número
x = 3245.875

# parte inteira
I = int(x)
di = []
while (I != 0):
  di.insert(0, I % b)
  I //= b

# parte fracionária
F = x % 1
df = []
while (F != 0):
  F *= b
```

```
    df.append(int(F))
    F %= 1

print('(',*[digs[d] for d in di],\
      ',',*[digs[d] for d in df],f')_{b}',sep="")
```

obtemos $3245{,}875 = (CAD{,}E)_{16}$.

◊

ER 1.1.3. Na base indicada, forneça a representação dos seguintes números:

a) $(1101)_2 \to$ base 8

b) $(1011{,}0101)_2 \to$ base 8

Solução.

a) $(1101)_2 \to$ base 8 »> oct(0b1101) ’0o15’ Ou seja, $(1101)_2 = (15)_8$.

b) $(1011{,}0101)_2 \to$ base 8 Primeiro, convertemos $(1011{,}0101)_2$ para decimal (base 10). »> 0b10110101 / 2**4 11.3125 Logo, convertemos para a base octal (base 8) com o seguinte código:

```
# base
b = 8

# número
x = 11.3125

# parte inteira
I = int(x)
di = []
while (I != 0):
  di.insert(0, I % b)
  I //= b

# parte fracionária
F = x % 1
df = []
while (F != 0):
  F *= b
  df.append(int(F))
```

```
        F %= 1

    print('(',*di,',',*df,f')_{b}',sep="")
```

Com este último, obtemos $(1011{,}0101)_2 = 11{,}3125 = (13{,}24)_8$

◊

1.1.3 Exercícios

Exercício 1.1.1. Obtenha a representação decimal dos seguinte números:

a) $(101101{,}00101)_2$

b) $(23{,}1)_4$

c) $(DAAD)_{16}$

d) $(33{,}11)_8$

e) $(51)_3$

Exercício 1.1.2. Obtenha a representação dos seguintes números decimais na base indicada:

a) 10 na base 2.

b) 45,5 na base 2.

c) 41 na base octal.

d) 66,31640625 na base hexadecimal.

e) $0{,}\overline{3}$ na base 3.

Exercício 1.1.3. Obtenha a representação dos seguintes números na base indicada:

a) $(101101{,}00101)_2$ na base 4.

b) $(23{,}1)_4$ na base 2.

c) $(2001)_{16}$ na base 8.

Exercício 1.1.4. Obtenha a representação dos seguintes números na base indicada:

a) $(0{,}1)_3$ na base decimal.

b) $(0{,}\overline{1})_3$ na base decimal.

c) $0{,}\overline{3}$ na base octal.

Exercício 1.1.5. Obtenha a representação dos seguintes números na base indicada:

a) 0,3 na base 4.

b) 0,3 na base 9.

c) $(A8)_{16}$ na base 5.

1.2 Representação de Números em Máquina

[YouTube] | [Vídeo] | [Áudio] | [Contatar]

Usualmente, números são manipulados em máquina através de suas representações em registros com *n-bits*. Ao longo desta seção, vamos usar a seguinte notação

$$[b_1\ b_2\ b_3\ \cdots\ b_n], \tag{1.30}$$

para representar um registro de *n-bits* $b_i \in \{0, 1\}$, $i = 1, 2, \ldots, n$.

Na sequência, fazemos uma breve discussão sobre as formas comumente usadas para a manipulação de números em computadores.

1.2.1 Números inteiros

O sistema de complemento de 2 é utilizado em computadores para a manipulação de números inteiros. Nesta representação, um registro de n *bits*

$$[d_1\ d_2\ d_3\ \cdots\ d_n], \tag{1.31}$$

representa o número inteiro

$$x = (d_{n-1}\ \ldots\ d_2\ d_1)_2 - d_n 2^{n-1}. \tag{1.32}$$

Exemplo 1.2.1. O registro de 8 *bits*[2]

$$[1\ 1\ 0\ 0\ 0\ 0\ 0\ 0] \tag{1.33}$$

representa o número

$$x = -d_8 \cdot 2^{8-1} + (d_7\ d_6\ \dots\ d_1)_2 \tag{1.34}$$
$$= -0 \cdots 2^7 + (\overset{6}{0}\ \overset{5}{0}\ \overset{4}{0}\ \overset{3}{0}\ \overset{2}{0}\ \overset{1}{1}\ \overset{0}{1})_2 \tag{1.35}$$
$$= 2^1 + 2^0 = 3. \tag{1.36}$$

Podemos implementar um conversor de registro para número inteiro como segue

Código 1.1: packbits8.py

```
def packBitsInt8(dd):
x = -dd[7] * 2**7
for i, d in enumerate(dd[:7]):
    x += d * 2**(i)
return x
```

Esta função, converte uma lista de *bits* (registro) no inteiro corresponde ao sistema de complemento 2.

```
>>> packBitsInt8([1,1,0,0,0,0,0,0])
3
```

Na representação de complemento de 2 com n *bits*, o menor e o maior números inteiros são obtidos com os registros

$$-2^{n-1} \sim [0\ 0\ 0\ 0\ \cdots\ 1], \tag{1.37}$$
$$2^{n-1} - 1 \sim [1\ 1\ 1\ \cdots\ 1\ 0], \tag{1.38}$$

respectivamente. Já o zero é obtido com o registro

$$0 \sim [0\ 0\ 0\ 0\ 0\ 0\ 0\ 0]. \tag{1.39}$$

[2] 8 *bits* = 1 *byte* [B].

Exemplo 1.2.2. Com um registro de 8-*bits*, temos que o menor e o maior números inteiros que podem ser representados são

$$[0\ 0\ 0\ 0\ 0\ 0\ 0\ 1] \tag{1.40}$$
$$\sim -2^7 + (0000000)_2 = -128, \tag{1.41}$$

e

$$[1\ 1\ 1\ 1\ 1\ 1\ 1\ 0] \tag{1.42}$$
$$\sim -0 \cdot 2^7 + (1111111)_2 = 127, \tag{1.43}$$

respectivamente.

Usando o Código 1.1, temos

```
>>> packBitsInt8([0,0,0,0,0,0,0,1])
-128
>>> packBitsInt8([1,1,1,1,1,1,1,0])
127
>>> packBitsInt8([0,0,0,0,0,0,0,0])
0
```

Observação 1.2.1. No NumPy, o `dtype=numpy.int8` corresponde a inteiros de 8 *bits*.

```
>>> import numpy as np
>>> np.array([-127, 0, 3, 128, 129], dtype=np.int8)
array([-127,    0,    3, -128, -127], dtype=int8)
```

Consulte a lista de tipos básicos do NumPy em NumPy:Data types.

A adição de números inteiros na representação de complemento de 2 pode ser feita de maneira simples. Por exemplo, consideremos a soma 3+9 usando registros de 8 *bits*. Temos

$$3 \sim [1\ 1\ 0\ 0\ 0\ 0\ 0\ 0] \tag{1.44}$$
$$9 \sim [1\ 0\ 0\ 1\ 0\ 0\ 0\ 0]\ + \tag{1.45}$$
$$-\ -\ -\ -\ -\ -\ -\ -\ - \tag{1.46}$$
$$12 \sim [0\ 0\ 1\ 1\ 0\ 0\ 0\ 0] \tag{1.47}$$

No sistema de complemento de 2, a representação de um número negativo $-x$ pode ser obtida da representação de x, invertendo seus *bits* e somando 1.

Por exemplo, a representação de -3 pode ser obtida da representação de 3, como segue

$$3 \sim [1\ 1\ 0\ 0\ 0\ 0\ 0\ 0]. \tag{1.48}$$

Invertendo seus *bits* e somando 1, obtemos

$$-3 \sim [1\ 0\ 1\ 1\ 1\ 1\ 1\ 1]. \tag{1.49}$$

A subtração de números inteiros usando a representação de complemento de 2 fica, então, tanto simples quanto a adição. Por exemplo:

$$3 \sim [1\ 1\ 0\ 0\ 0\ 0\ 0\ 0] \tag{1.50}$$

$$-9 \sim [1\ 1\ 1\ 0\ 1\ 1\ 1\ 1]\ + \tag{1.51}$$

$$- - - - - - - - - - - \tag{1.52}$$

$$-6 \sim [0\ 1\ 0\ 1\ 1\ 1\ 1\ 1] \tag{1.53}$$

1.2.2 Ponto flutuante

A manipulação de números decimais em computadores é comumente realizada usando a representação de ponto flutuante de 64 *bits*[3]. Nesta, um dado registro de 64 *bits*

$$[s \mid c_{10}\ c_9\ \dots\ c_0 \mid m_1\ m_2\ \dots\ m_{52}] \tag{1.54}$$

representa o número

$$x = (-1)^s M \cdot 2^{c-1023}, \tag{1.55}$$

onde M é chamada de mantissa e c da característica, as quais são definidas por

$$M := (1{,}m_1 m_2 m_3 \dots m_{52})_2, \tag{1.56}$$

$$c := (c_{10} \dots c_2 c_1 c_0)_2. \tag{1.57}$$

Exemplo 1.2.3. Por exemplo, na representação em ponto flutuante de 64 *bits*, temos que o registro

$$[1 \mid 1\ 0\ \dots\ 0\ 0 \mid 1\ 0\ 1\ 0\ 0\ \dots\ 0] \tag{1.58}$$

[3] Padrão IEEE 754.

representa o número $-3{,}25$.

A seguinte função faz a conversão uma lista de 64 *bits* no número decimal corresponde ao sistema de ponto flutuante de 64 *bits*.

Código 1.2: packBitsDouble.py

```
def packBitsDouble(ld):
s = ld[0]
c = 0
for i, d in enumerate(ld[1:12]):
    c += d * 2**(10-i)
m = 1.
for i, d in enumerate(ld[12:]):
    m += d * 2**(-(i+1))
x = m * 2**(c - 1023)
return -x if s else x
```

Por exemplo, usando-a para o registro acima, obtemos

```
>>> ld[0]=1; ld[1]=1; ld[12]=1; ld[13]=1
>>> packBitsDouble(ld)
-3.5
```

1.2.3 Erro de arredondamento

Dado um número real x, sua representação $fl(x)$ em ponto flutuante é o registro que representa o número mais próximo de x. Este procedimento é chamado de arredondamento por proximidade.

A seguinte função obtém a representação em ponto flutuante de 64 *bits* de um dado número x[4].

Código 1.3: unpackBitsDouble.py

```
import numpy as np

def unpackBitsDouble(x):
    ld = 64*[0]
    if ( x == 0):
```

[4] Esta função não é precisa e pode fornecer registros errados devido a erros de arredondamento. Uma alternativa melhor é apresentada na Observação 1.2.2.

```
        return ld
    elif (x < 0):
        ld[0] = 1
    x = np.fabs(x)
    c = int(np.log2(x) + 1023)
    m = x/2**(c - 1023)
    for i in range(11):
        ld[11 - i] = c % 2
        c //= 2
    m -= 1
    for i in range(52):
        m *= 2
        ld[12+ i] = int(m)
        m %= 1
    return ld
```

Por exemplo, $x = 1{,}1$ é representado pelo registro

```
>>> ld = unpackBitsDouble(1.1)
>>> ld
[0, 0, 1, 1, 1, 1, 1, 1,
 1, 1, 1, 1, 0, 0, 0, 1,
 1, 0, 0, 1, 1, 0, 0, 1,
 1, 0, 0, 1, 1, 0, 0, 1,
 1, 0, 0, 1, 1, 0, 0, 1,
 1, 0, 0, 1, 1, 0, 0, 1,
 1, 0, 0, 1, 1, 0, 0, 1,
 1, 0, 0, 1, 1, 0, 1, 0]
```

que corresponde ao número

```
>>> flx = packBitsDouble(ld)
>>> print(f'{flx:1.51f}')
1.10000000000000008881784197
001252323389053344726562 5
```

O erro de arredondamento é $|x - fl(x)| \approx 8{,}9 \times 10^{-17}$.

Observação 1.2.2. O seguinte código é uma solução mais pythonica para obter-se o registro em ponto flutuante de 64 *bits* de $x = 1{,}1$.

```
>>> ''.join(f'{c:08b}' for c in struct.pack('!d', 1.1))
```

```
  '0011111111110001
  1001100110011001
  1001100110011001
  1001100110011010'
```

Recomendamos consultar [4] para mais informações sobre a conversão eficiente de números decimais em pontos flutuantes.

Observemos que o erro de arredondamento varia conforme o número dado, podendo ser zero no caso de $x = fl(x)$. Comumente, utiliza-se o **épsilon de máquina** como uma aproximação desse erro. O épsilon de máquina é definido como a distância entre o número 1 e seu primeiro sucessor em ponto flutuante. Temos

```
>>> ld = unpackBitsDouble(1)
>>> ld
[0, 0, 1, 1, 1, 1, 1, 1,
1, 1, 1, 1, 0, 0, 0, 0,
0, 0, 0, 0, 0, 0, 0, 0,
0, 0, 0, 0, 0, 0, 0, 0,
0, 0, 0, 0, 0, 0, 0, 0,
0, 0, 0, 0, 0, 0, 0, 0,
0, 0, 0, 0, 0, 0, 0, 0,
0, 0, 0, 0, 0, 0, 0, 0]
>>> ld[63] = 1
>>> x = packBitsDouble(ld)
>>> x-1
2.220446049250313e-16
```

Ou seja, o épsilon de máquina é

$$\text{eps} := 2^{-52} \approx 2{,}22 \times 10^{-16}. \tag{1.59}$$

Observação 1.2.3. O método `numpy.finfo` pode ser usado para obtermos várias informações sobre o sistema de números em ponto flutuante. Por exemplo, temos

```
  >>> import numpy as np
  >>> finfo = np.finfo(np.double)
  >>> finfo.eps
  2.220446049250313e-16
```

```
>>> finfo.min
-1.7976931348623157e+308
>>> finfo.max
1.7976931348623157e+308
```

A aritmética em ponto flutuante requer arredondamentos sucessivos de números. Por exemplo, a computação da soma de dois números dados x e y é feita a partir de suas representações em ponto flutuante $fl(x)$ e $fl(y)$. Então, computa-se $z = fl(x) + fl(y)$ e o resultado é $fl(z)$. Observe, inclusive que $fl(x + y)$ pode ser diferente de $fl(fl(x) + fl(y))$. Por exemplo

```
>>> 0.1 + 0.2 == 0.3
False
```

1.2.4 Exercícios Resolvidos

ER 1.2.1. No sistema de complemento 2 de 8 *bits*, forneça o registro que representa os seguintes números inteiros:

a) 1

b) -1

c) 15

d) -15

Solução. A seguinte função, obtém o registro de complemento 2 de 8-*bits* de um dado número inteiro x.

```
def unpackBitsInt8(x):
ld = 8*[0]
if (x < 0):
    ld[7] = 1
    x += 2**(7)
for i in range(7):
    ld[i] = x % 2
    x //= 2
return ld
```

Usando-a, obtemos os seguintes resultados:

```
>>> # a) 1
>>> unpackBitsInt8(1)
[1, 0, 0, 0, 0, 0, 0, 0]
>>> unpackBitsInt8(-1)
[1, 1, 1, 1, 1, 1, 1, 1]
>>> # c) 15
>>> unpackBitsInt8(15)
[1, 1, 1, 1, 0, 0, 0, 0]
>>> unpackBitsInt8(-15)
[1, 0, 0, 0, 1, 1, 1, 1]
```

$\Diamond$

ER 1.2.2. Qual é o número decimal positivo mais próximo de zero que pode ser representado como um ponto flutuante de 64-*bits*. Também, forneça seu registro.

Solução. Um registro em ponto flutuante de 64-*bits* tem a forma

$$[s \mid c_{10}\ c_9\ \dots\ c_0 \mid m_1\ m_2\ \dots\ m_{52}] \tag{1.60}$$

e representa o número

$$x = (-1)^s M \cdot 2^{c-1023}, \tag{1.61}$$

onde M é chamada de mantissa e c da característica, as quais são definidas por

$$M := (1{,}m_1 m_2 m_3 \dots m_{52})_2, \tag{1.62}$$

$$c := (c_{10} \dots c_2 c_1 c_0)_2. \tag{1.63}$$

Tendo em vista que o registro nulo é reservado para o número decimal zero, temos que o número positivo mais próximo de zero é obtido com sinal $s = 0$, a mantissa $M = 1$ e a característica $c = 1$, no que obtemos o decimal

$$x = 2^{-1022} \tag{1.64}$$

$$\approx 2.2250738585072014e - 308 \tag{1.65}$$

Seu registro é

$$[0 \mid 0\ 0\ \dots\ 1 \mid 0\ 0\ \dots\ 0] \tag{1.66}$$

O resultado pode ser verificado com os seguintes comandos:

```
>>> import numpy as np
>>> import struct
>>> x = np.finfo(np.double).tiny; x
2.2250738585072014e-308
>>> ''.join(f'{c:08b}' \
... for c in struct.pack('!d', x))
'0000000000010000
0000000000000000
0000000000000000
0000000000000000'
```

◊

ER 1.2.3. Em aplicações que não necessitam de muita precisão, a representação de números decimais no sistema de ponto flutuante de 32 *bits* é mais eficiente (no sentido de velocidade de processamento computacional). Neste sistema, um registro de 32-*bits*

$$[s \mid c_7\ c_6\ \ldots\ c_0 \mid m_1\ m_2\ \ldots\ m_{23}] \tag{1.67}$$

representa o número

$$x = (-1)^s \cdot M \cdot 2^{c-127} \tag{1.68}$$

onde,

$$M = (1{,}m_1 m_2 \ldots m_{23})_2 \tag{1.69}$$

$$c = (c_7 c_6 \ldots c_0)_2 \tag{1.70}$$

a) Forneça o registro do ponto flutuante de 32-*bits* que representa o número 42,5.

b) Qual é o sucessor em ponto flutuante de 32-*bits* do número decimal 1. Forneça, também, o épsilon de máquina deste sistema.

Solução.

a) O registro do ponto flutuante de 32-*bits* que representa o número 42,5 pode ser computado com o seguinte código:

```
x = 42.5
ld = 32*[0]
c = int(np.log2(x) + 127)
m = x/2**(c-127)
```

```
for i in range(8):
  ld[8-i] = c % 2
  c //= 2
m -= 1
for i in range(23):
  m *= 2
  ld[9+i] = int(m)
  m %= 1
```

```
>>> ld
[0, 1, 0, 0, 0, 0, 1, 0,
0, 0, 1, 0, 1, 0, 1, 0,
0, 0, 0, 0, 0, 0, 0, 0,
0, 0, 0, 0, 0, 0, 0, 0]
```

Alternativamente, pode-se obter o registro como segue:

```
>>> ''.join(f'{c:08b}' \
... for c in struct.pack('!f', 42.5))
'0100001000101010
0000000000000000'
```

b) No sistema de ponto flutuante de 32-*bits*, o sucessor de 1 tem o registro

$$[0 \mid 0\ 1\ 1\ \dots\ 1 \mid 0\ 0\ \dots\ 0\ 1] \tag{1.71}$$

donde, sua mantissa é $m = 1 + 2^{-23}$, característica $c = 127$ e corresponde ao número decimal

$$x = (-1)^0 \cdot (1 + 2^{-23}) \cdot 2^{127-127} \tag{1.72}$$
$$x = 1 + 2^{-23} \tag{1.73}$$

Portanto, o épsilon de máquina neste sistema é

$$\text{eps} = x - 1 \tag{1.74}$$
$$= 2^{-23} \tag{1.75}$$

```
>>> np.float32(2**-23)
1.1920929e-07
```

◊

Em construção ...

1.2.5 Exercício

Exercício 1.2.1. Considerando a representação de complemento de 2 de números inteiros, obtenha os registros de 8-*bits* dos seguintes números:

a) 17

b) -17

c) 32

d) -32

Exercício 1.2.2. Considerando a representação de complemento de 2 de números inteiros, obtenha os registros de 16-*bits* dos seguintes números:

a) 1024

b) -1024

Exercício 1.2.3. Considerando a representação de complemento de 2 de números inteiros, qual é o maior número que pode ser representado por um registro de 32-*bits* da forma

$$[1\ 0\ b_2\ b_3\ b_4\ \cdots\ b_{30}\ 1], \tag{1.76}$$

onde $b_i \in \{0, 1\}$, $i = 2, 3, 4, \cdots, 30$.

Exercício 1.2.4. Obtenha os registros em ponto flutuante de 64-*bits* dos seguintes números:

a) $-1{,}25$

b) 3

Exercício 1.2.5. Assumindo o sistema de ponto flutuante de 32-*bits*, obtenha o registro e o erro de arredondamento na representação dos seguintes números decimais:

a) 0,1

b) 10,1

c) 100,1

1.3 Notação Científica e Arredondamento

Enquanto que a manipulação de números decimais é comumente feita usando-se da aritmética em ponto flutuante, a interpretação dos parâmetros dos problemas de interesse e seus resultados é normalmente feita com poucos dígitos. Nesta seção, introduziremos algumas notações que serão utilizadas ao longo deste material.

A **notação científica** é a representação de um dado número na forma

$$d_n \ldots d_2 d_1 d_0{,}d_{-1} d_{-2} d_{-3} \ldots \times 10^E, \tag{1.77}$$

onde d_i, $i = n, \ldots, 1, 0, -1, \ldots$, são algarismos da base 10. A parte à esquerda do sinal $\times$ é chamada de mantissa do número e E é chamado de expoente (ou ordem de grandeza).

Exemplo 1.3.1. O número 31,515 pode ser representado em notação científica das seguintes formas

$$\begin{aligned} 31{,}415 \times 10^0 &= 3{,}1415 \times 10^1 && (1.78)\\ &= 314{,}15 \times 10^{-1} && (1.79)\\ &= 0{,}031415 \times 10^3, && (1.80) \end{aligned}$$

entre outras tantas possibilidades.

Em Python, usa-se a letra `e` para separar a mantissa do expoente na notação científica. Por exemplo

```
>>> # 31.415 X 10^0
>>> 31.415e0
31.515
>>> # 3.1415 X 10^1
>>> 3.1415e1
31.515
>>> # 314.15 X 10^-1
>>> 314.15e-1
31.515
>>> # 0.031415 X 10^3
>>> 0.031415e3
31.415
```

No exemplo anterior (Exemplo 1.3.1), podemos observar que a representação em notação científica de um dado número não é única. Para contornar isto, introduzimos a **notação científica normalizada**, a qual tem a forma

$$d_0{,}d_{-1}d_{-2}d_{-3}\ldots \times 10^E, \tag{1.81}$$

com $d_0 \neq 0$[5].

Exemplo 1.3.2. O número 31,415 representado em notação científica normalizada é $3{,}1415 \times 10^1$.

Em Python, podemos usar o método `format` para imprimir um número em notação científica normalizada. Por exemplo, temos

```
>>> x = 31.415
>>> print(f"{x:e}")
3.141500e+01
```

Como vimos na seção anterior, costumeiramente usamos da aritmética de ponto flutuante nas computações, com a qual os números são representados com muito mais dígitos dos quais estamos interessados na interpretação dos resultados. Isto nos leva de volta a questão do arredondamento.

Dizemos que um número está representado com n **dígitos significativos** (na notação científica normalizada) quando está escrito na forma

$$d_0{,}d_1d_2\ldots d_{n-1} \times 10^E, \tag{1.82}$$

com $d_0 \neq 0$.

Exemplo 1.3.3. Estudamos as seguintes representações do número 31,415:

a) com 5 dígitos significativos

```
>>> x = 31.415
>>> print(f"{x:.4e}")
3.1415e+01
```

b) com 6 dígitos significativos

```
>>> print(f"{x:.5e}")
3.14150e+01
```

[5] No caso do número zero, temos $d_0 = 0$.

c) com 4 dígitos significativos

```
>>> print(f"{x:.3e}")
3.142e+01
```

Neste último caso, fez-se necessário arredondar o número.

1.3.1 Arredondamento

Observamos que pode ocorrer a necessidade de se arredondar um número para obter sua representação com um número finito de dígitos significativos. Por exemplo, para representarmos o número $x = 3{,}1415 \times 10^1$ com 3 dígitos significativos, precisamos determinar de que forma vamos considerar a contribuição de seus demais dígitos a direita. Isto, por sua vez, é determinado pelo tipo de arredondamento que iremos utilizar.

O tipo de arredondamento mais comumente utilizado é o chamado **arredondamento por proximidade com desempate par**. Neste, a representação escolhida é aquela mais próxima do número dado. Por exemplo, a representação de

$$x = 3{,}1415 \times 10^1 \tag{1.83}$$

com três dígitos significativos é

$$x = 3{,}14 \times 10^1. \tag{1.84}$$

Agora, sua representação com apenas dois dígitos significativos é

$$x = 3{,}1 \times 10^1. \tag{1.85}$$

No caso de empate, usa-se a seguinte regra: 1) se o último dígito significativo ser par, este é mantido; 2) se o último dígito significativo ser ímpar, este é acrescido de uma unidade. Por exemplo, no caso do número $x = 3{,}1415 \times 10^1$, sua representação com 4 dígitos significativos é

$$x = 3{,}142 \times 10^1. \tag{1.86}$$

Observação 1.3.1. O arredondamento por proximidade com desempate par é o padrão do IEEE 754[6]. No entanto, devemos lembrar que a maioria dos números decimais não tem representação exata no sistema de ponto flutuante. Por exemplo,

[6] Para mais detalhes, consulte IEEE 754: Wikipedia.

```
>>> x = 31.415
>>> print(f'{x:.3e}')
3.141e+01
```

Embora o arrendamento não seja o esperado, o que ocorre é que $x = 31{,}415$ não tem representação exata em ponto flutuante, de fato

```
>>> print(f'{x:.25e}')
3.1414999999999999147348717e+01
```

No restante deste material estaremos assumindo a notação científica normalizada com arredondamento por proximidade com desempate par.

1.3.2 Exercícios Resolvidos

ER 1.3.1. Faça o cálculo exato e a computação de

$$\frac{0{,}33411 \times 10^2 - 271{,}28 \times 10^{-1}}{2000 \times 10^{-3}} \tag{1.87}$$

Forneça os resultados com 4 dígitos significados.

Solução.

- Por cálculo exato.

$$\frac{0{,}33411 \times 10^2 - 271{,}28 \times 10^{-1}}{2000 \times 10^{-3}} \tag{1.88}$$
$$= \frac{334{,}11 \times 10^{-1} - 271{,}28 \times 10^{-1}}{2 \times 10^0} \tag{1.89}$$
$$= \frac{63{,}83 \times 10^{-1}}{2} \tag{1.90}$$
$$= 31{,}415 \times 10^{-1} \tag{1.91}$$

Arredondando o resultado para 4 dígitos significativos, obtemos 3,142.

- Por computação.

```
>>> x = (0.33411e2 - 271.28e-1)/2000e-3
>>> x
3.1415000000000006
>>> print(f'{x:.3e}')
3.142e+00
```

$\Diamond$

ER 1.3.2. Obtenha os arredondamentos dos seguintes números decimais para quantidade de dígitos significativos indicada em cada caso. Então, compare com a computada em ponto flutuante.

a) 2,7128 com 4 dígitos significativos.

b) 2,7128 com 2 dígitos significativos.

c) 1,9910 com 3 dígitos significativos.

d) 1,9910 com 2 dígitos significativos.

e) 5,5555 com 4 dígitos significativos.

f) 5,6555 com 4 dígitos significativos.

Solução.

a) 2,7128 com 4 dígitos significativos = 2,713

```
>>> f'{2.7128:.3e}'
'2.713e+00'
```

b) 2,7128 com 2 dígitos significativos = 2,7

```
>>> f'{2.7128:.1e}'
'2.7e+00'
```

c) 1,9910 com 3 dígitos significativos = 1,99

```
>>> f'{1.9910:.2e}'
'1.99e+00'
```

d) 1,9910 com 2 dígitos significativos = 2,0

```
>>> f'{1.9910:.1e}'
'2.0e+00'
```

e) 5,5555 com 4 dígitos significativos = 5,556

```
>>> f'{5.5555:.3e}'
'5.556e+00'
```

f) 5,6555 com 4 dígitos significativos = 5,556

```
>>> f'{5.6555:.3e}'
'5.655e+00'
```

◊

1.3.3 Exercícios

Exercício 1.3.1. Obtenha a representação dos seguintes números decimais em notação científica normalizada com a quantidade de dígitos indicada em cada caso. Então, compare com o arredondamento computado em ponto flutuante. Caso haja diferença, explique.

a) π com 6 dígitos significativos.

b) $\pi/10$ com 6 dígitos significativos.

c) $\sqrt{2}/\sqrt{3}$ com 7 dígitos significativos.

Exercício 1.3.2. Compute a seguinte expressão

$$\frac{\sqrt{\pi} - \ln(0{,}9)}{75\cos\left(\frac{\pi}{4}\right)}. \tag{1.92}$$

Forneça a resposta com 7 dígitos significativos.

Exercício 1.3.3. Forneça o arredondamento dos seguintes números decimais para 2 dígitos significativos. Então, compare com o arrendamento computado em ponto flutuante. Caso haja diferença, explique.

a) 0,625

b) 0,615

c) 0,635

Exercício 1.3.4. Seja f_s a função que recebe número decimal e retorna sua aproximação por arrendamento com 2 dígitos significativos. Calcule

a) $f_s(2\pi - e)$

b) $2f_s(\pi) - f_s(e)$

c) Por que $f_s(2\pi - e) \neq 2f_s(\pi) - f_s(e)$?

Exercício 1.3.5. Explique o porquê de

```
>>> np.sqrt(3)**2 == 3
False
```

1.4 Tipos e Medidas de Erros

Ao utilizarmos computadores na resolução de problemas matemáticos, acabamos obtendo soluções aproximadas aproximadas. A diferença entre a solução exata e a computada solução aproximada é chamada de erro. O erro é comumente classificado nas seguintes duas categorias:

- **Erro de arredondamento**. Este é o erro que ocorre na representação aproximada de números na máquina.
- **Erro de truncamento**. Este é o erro que ocorre na interrupção (truncamento) de um procedimento com infinitos passos.

Exemplo 1.4.1. O erro de arredondamento em aproximar π por $3{,}1415\times 10^0$ é de aproximadamente $9{,}3 \times 10^{-5}$.

Exemplo 1.4.2. Consideremos a seguinte série numérica $\sum_{k=0}^{\infty} 1/k! = e \approx 2{,}7183 \times 10^0$. Ao computarmos esta série no computador, precisamos truncá-la em algum k suficientemente grande. Por exemplo, trunca a série em seu décimo termo, temos

$$\begin{aligned}\sum_{k=1}^{\infty} \frac{1}{k!} &\approx 1 + 1 + \frac{1}{2} + \frac{1}{3} + \cdots + \frac{1}{9} &(1.93)\\ &\approx 2{,}7182815 \times 10^0 =: \tilde{e}. &(1.94)\end{aligned}$$

A diferença $e - \tilde{e} \approx 3 \times 10^{-7}$ é o erro de truncamento associado.

Suponhamos, agora, que x seja o valor exato de uma quantidade de interesse e $\tilde{x}$ a quantidade computada (i.e., uma aproximação de x). Em matemática numérica, utilizamos frequentemente as seguintes medidas de erro:

- Erro absoluto:

$$e_{abs} := |x - \tilde{x}|. \tag{1.95}$$

- Erro relativo:

$$e_{rel} := \frac{|x - \tilde{x}|}{|x|} \, (\times 100\%) \, . \tag{1.96}$$

A vantagem do erro relativo é em levar em conta a ordem de grandeza das quantidades x e $\tilde{x}$. Vejamos o seguinte exemplo (Exemplo 1.4.3).

Exemplo 1.4.3. Observemos os seguintes casos:

a) $x = 1{,}0$ e $\tilde{x} = 1{,}1$:

$$e_{abs} = |x - \tilde{x}| = 1 \times 10^{-1}. \tag{1.97}$$

$$e_{rel} = \frac{|x - \tilde{x}|}{|x|} = 1 \times 10^{-1} = 10\%. \tag{1.98}$$

b) $x = 10000{,}0$ e $\tilde{x} = 11000{,}0$:

$$e_{abs} = |x - \tilde{x}| = 1 \times 10^{3}. \tag{1.99}$$

$$e_{rel} = \frac{|x - \tilde{x}|}{|x|} = 1 \times 10^{-1} = 10\%. \tag{1.100}$$

Outra medida de erro comumente empregada é o **número de dígitos significativos corretos**. Dizemos que $\tilde{x}$ aproxima x com n dígitos significativos corretos, quando

$$\frac{|x - \tilde{x}|}{|x|} < 5 \times 10^{-n}. \tag{1.101}$$

Exemplo 1.4.4. Vejamos o seguintes casos:

- $x = 2$ e $\tilde{x} = 2{,}5$:

  $$\frac{|x - \tilde{x}|}{|x|} = 0{,}25 < 5 \times 10^{-1}, \tag{1.102}$$

  donde concluímos que $\tilde{x} = 2{,}5$ é uma aproximação com 1 dígito significativo correto de $x = 2$.

- $x = 1$ e $\tilde{x} = 1{,}5$:

  $$\frac{|x - \tilde{x}|}{|x|} = 0{,}5 < 5 \times 10^{-0}, \tag{1.103}$$

  donde concluímos que $\tilde{x} = 1{,}5$ é uma aproximação com zero dígito significativo correto de $x = 1$.

1.4.1 Propagação de Erros

Nesta seção, vamos introduzir uma estimativa para a propagação de erros (de arredondamento) na computação de um problema. Para tando, vamos considerar o caso de se calcular o valor de uma dada função f em um dado ponto x, i.e. queremos calcular y com

$$y = f(x). \tag{1.104}$$

Agora, assumindo que x seja conhecido com um erro $e(x)$, este se propaga no cálculo da f, levando a um erro $e(y)$ no valor calculado de y. Ou seja, temos

$$y + e(y) = f(x + e(x)). \tag{1.105}$$

Notemos que $e_{abs}(x) = |e(x)|$ é o erro absoluto associado a x e $e_{abs}(y) = |e(y)|$ é o erro absoluto associado a y.

Nosso objetivo é de estimar $e_{abs}(y)$ com base em $e_{abs}(x)$. Para tanto, podemos tomar a aproximação de $f(x+e(x))$ dada pelo polinômio de Taylor de grau 1 de f em torno de x, i.e.

$$f(x + e(x)) = f(x) + f'(x)e(x) + O\left(e^2(x)\right). \tag{1.106}$$

Então, de (1.104) e (1.105), temos

$$e(y) = f'(x)e(x) + O(e^2(x)). \tag{1.107}$$

Daí, passando ao valor absoluto e usando a desigualdade triangular, obtemos

$$e_{abs}(y) \leq |f'(x)|e_{abs}(x) + O\left(e^2_{abs}(x)\right). \tag{1.108}$$

Deste resultado, consideraremos a seguinte estimativa de propagação de erro

$$e_{abs}(y) \approx |f'(x)|e_{abs}(x). \tag{1.109}$$

Exemplo 1.4.5. Consideremos o problema em se calcular $y = f(x) = x^2 \operatorname{sen}(x)$ com $x = \pi/3 \pm 0{,}1$. Usando (1.109) para estimarmos o erro absoluto $e_{abs}(y)$ no cálculo de y com base no erro absoluto $e_{abs}(x) = 0{,}1$, calculamos

$$e_{abs}(y) = |f'(x)|e_{abs}(x) \tag{1.110}$$
$$= |2x \operatorname{sen}(x) + x^2 \cos(x)|e_{abs}(x) \tag{1.111}$$

$$= 2{,}3621 \times 10^{-1}. \tag{1.112}$$

Com isso, podemos concluir que um erro em x de tamanho 0,1 é propagado no cálculo de $f(x)$, causando um erro pelo menos duas vezes maior em y. Também, podemos interpretar este resultado do ponto de vista do erro relativo. O erro relativo associado a x é

$$e_{rel}(x) = \frac{e_{abs}(x)}{|x|} = \frac{0{,}1}{\pi/3} = 9{,}5493 \times 10^{-2} \approx 1\%, \tag{1.113}$$

acarretando um erro relativo em y de

$$e_{rel}(y) = \frac{e_{abs}(y)}{|y|} = \frac{e_{abs}(y)}{|f(x)|} = 2{,}4872 \times 10^{-1} \approx 2\%. \tag{1.114}$$

Associada à estimativa (1.4.5), temos

$$\begin{aligned} e_{rel}(y) = \frac{e_{abs}(y)}{|y|} &= \frac{|f'(x)|}{|y|} e_{abs}(x) \\ &= \frac{|x| \cdot |f'(x)|}{|f(x)|} \frac{e_{abs}(x)}{|x|} \\ &= \left| \frac{x f'(x)}{f(x)} \right| e_{rel}(x). \end{aligned}$$

Desta última equação, definimos o **número de condicionamento** de f, denotado por

$$\kappa_f(x) := \left| \frac{x f'(x)}{f(x)} \right|. \tag{1.115}$$

Observamos que $\kappa_f(x)$ é a escala com que erros em x são propagados no cálculo de $y = f(x)$.

Exemplo 1.4.6. O número de condicionamento da função $f(x) = x^2 \operatorname{sen}(x)$ no ponto $x = \pi/3$ pode ser calculado de

$$\begin{aligned} \kappa_f(x) &= \left| \frac{x f'(x)}{f(x)} \right| &(1.116)\\ &= \left| \frac{x(2x \operatorname{sen}(x) + x^2 \cos(x))}{x^2 \operatorname{sen}(x)} \right|. &(1.117) \end{aligned}$$

Substituindo x por $\pi/3$ temos

$$\kappa_f(\pi/3) = 2{,}6046. \tag{1.118}$$

Notamos que este resultado é compatível com as observações feitas no Exemplo 1.4.5.

A estimativa (1.109) pode ser generalizada para uma função de várias variáveis. No caso de uma função $y = f(x_1,x_2,\ldots,x_n)$, temos

$$e_{abs}(y) = \sum_{k=1}^{n} \left|\frac{\partial f}{\partial x_k}\right| e_{abs}(x_k). \tag{1.119}$$

Exemplo 1.4.7. Consideremos o problema em se calcular $z = f(x,y) = x^2 \operatorname{sen}(x)\cos(y)$ com $x = \pi/3 \pm 0{,}1$ e $y = \pi/4 \pm 0{,}02$. Usando (1.119) para estimarmos o erro absoluto $e_{abs}(z)$ no cálculo de z com base nos erros absolutos $e_{abs}(x) = 0{,}1$ e $e_{abs}(y) = 0{,}02$, calculamos

$$\begin{aligned}
e_{abs}(z) &= \left|\frac{\partial f}{\partial x}\right| e_{abs}(x) + \left|\frac{\partial f}{\partial y}\right| e_{abs}(y) && (1.120)\\
&= |(2x\operatorname{sen}(x) + x^2\cos(x))\cos(y)| e_{abs}(x)\\
&+ \left|-x^2\operatorname{sen}(x)\operatorname{sen}(y)\right| e_{abs}(y) && (1.121)\\
&= 1{,}8046 \times 10^{-1}. && (1.122)
\end{aligned}$$

1.4.2 Cancelamento Catastrófico

No computador (com aritmética de ponto flutuante de 64-*bits*), as operações e funções elementares são computadas, usualmente, com um erro próximo do épsilon de máquina (eps $\approx 10^{-16}$). Entretanto, em algumas situações estas operações fundamentais acarretam erros maiores, causando uma perda de precisão.

O chamado cancelamento catastrófico ocorre quando ao computarmos a diferença entre dois números próximos. Para ilustrá-lo, consideremos os seguintes números

$$\begin{aligned}
x &= 314150000001549, && (1.123)\\
y &= 314150000002356. && (1.124)
\end{aligned}$$

Suponhamos, ainda, os arredondamentos de x e y com 12 dígitos significativos

$$\tilde{x} = 314150000002000, \tag{1.125}$$
$$\tilde{y} = 314150000002000. \tag{1.126}$$

Notemos que os erros relativos associados às aproximações de x e y por $\tilde{x}$ e $\tilde{y}$ são

$$e_{rel}(x) = \frac{|x - \tilde{x}|}{|x|} \approx 10^{-10}\%, \tag{1.127}$$
$$e_{rel}(y) = \frac{|y - \tilde{y}|}{|y|} \approx 10^{-10}\%, \tag{1.128}$$

respectivamente. Agora, temos

$$y - x = 807 \quad \text{e} \quad \tilde{y} - \tilde{x} = 0. \tag{1.129}$$

Ou seja, o erro relativa na aproximação de $y - x$ por $\tilde{y} - \tilde{x}$ é

$$e_{rel}(y - x) = \frac{|(y - x) - (\tilde{y} - \tilde{x})|}{(y - x)} = \frac{807}{807} = 100\%! \tag{1.130}$$

Vejamos outros exemplos.

Exemplo 1.4.8. Na Tabela 1.1 temos os erros em se computar

$$\frac{(1 + x^4) - 1}{x^4} \tag{1.131}$$

para diferentes valores de x. Observamos que, para o valor de $x = 0{,}001$ o erro na computação já é da ordem de 10^{-5} e para valores de x menores ou iguais a $0{,}0001$ o erro é catastrófico. Isto ocorre, pois se $x \leq 10^{-4}$, então $x^4 \leq 10^{-16} <$ eps e, portanto, $(1 + x^4) - 1 = 0$.

Exemplo 1.4.9. Uma equação de segundo grau $ax^2 + bx + c = 0$ tem raízes

$$x_1 = \frac{-b + \sqrt{b^2 - 4ac}}{2a}, \tag{1.132}$$
$$x_2 = \frac{-b - \sqrt{b^2 - 4ac}}{2a}. \tag{1.133}$$

x	erro
1	0
10^{-1}	$1{,}1 \times 10^{-13}$
10^{-2}	$6{,}1 \times 10^{-9}$
10^{-3}	$8{,}9 \times 10^{-5}$
10^{-4}	$1{,}0 \times 10^{0}$
10^{-5}	$1{,}0 \times 10^{0}$

Tabela 1.1: Resultados referentes ao Exemplo 1.4.8.

Entretanto, no caso de b e $\sqrt{b^2 - 4ac}$ serem ambos positivos, a fórmula (1.132) não é adequada para a computação da raiz x_1, pois pode ocorrer cancelamento catastrófico. Podemos contornar este problema reescrevendo (1.132) da seguinte forma

$$x_1 = \frac{-b + \sqrt{b^2 - 4ac}}{2a} \frac{-b - \sqrt{b^2 - 4ac}}{-b - \sqrt{b^2 - 4ac}} \tag{1.134}$$

$$= \frac{-b^2 + b^2 - 4ac}{2a(-b - \sqrt{b^2 - 4ac})} \tag{1.135}$$

$$= \frac{2c}{b + \sqrt{b^2 - 4ac}}, \tag{1.136}$$

a qual não sofre mais de cancelamento catastrófico. Observemos que também pode ocorrer cancelamento catastrófico no cálculo de x_2 pela fórmula (1.133), no caso de b e $\sqrt{b^2 - 4ac}$ serem ambos negativos.

1.4.3 Exercícios Resolvidos

Em construção ...

1.4.4 Exercícios

Exercício 1.4.1. Calcule o erro absoluto na aproximação de

a) π por 3,14.

b) $10e$ por 27,18.

Forneça as respostas com 4 dígitos significativos.

Exercício 1.4.2. Calcule o erro relativo na aproximação de

a) π por 3,14.

b) $10e$ por 27,18.

Forneça as respostas em porcentagem.

Exercício 1.4.3. Com quantos dígitos significativos corretos

a) 3,13 aproxima π?

b) 27,21 aproxima $10e$?

Exercício 1.4.4. Obtenha uma estimativa do erro de truncamento em se aproximar o valor de $\operatorname{sen}(1)$ usando-se $p_5(1)$, onde $p_5(x)$ é o polinômio de Taylor de grau 5 da função $\operatorname{sen}(x)$ em torno de $x = 0$.

Exercício 1.4.5. Considerando que $x = 2 \pm 0{,}1$, estime o erro absoluto em se calcular $y = e^{-x^2}\cos(\pi x/3)$. Forneça a estimativa com 7 dígitos significativos por arredondamento.

Exercício 1.4.6. Considerando que $x = 2 \pm 2\%$ e $y = 1{,}5 \pm 0{,}3$, estime o erro absoluto em se calcular $y = e^{-x^2}\cos(\pi y/3)$. Forneça a estimativa com 6 dígitos significativos por arredondamento.

Exercício 1.4.7. Considere a computação de

$$y = \frac{1 - \cos(h)}{h} \tag{1.137}$$

para $h = 10^{-9}$. Compute o valor de y reescrevendo esta expressão de forma a mitigar o cancelamento catastrófico. Forneça o valor computado de y com 2 dígitos significativos por arredondamento.

Capítulo 2

Equação com uma incógnita

Neste capítulo, discutiremos sobre métodos numéricos para resolver equações com uma incógnita real. Para tanto, notemos que toda equação pode ser reescrita na seguinte forma equivalente

$$f(x) = 0, \tag{2.1}$$

onde f é uma função adequada. Isto é, o problema de se encontrar a incógnita de uma dada equação pode ser reescrito como um problema de encontrar os zeros (ou raízes) de uma função de uma variável real.

Os métodos numéricos que abordaremos ao longo deste capítulo são descritos para problemas da forma (2.1).

2.1 Método da bisseção

O método da bisseção explora o fato de que toda função contínua f com $f(a) \cdot f(b) < 0$ (i.e., $f(a)$ e $f(b)$ tem sinais diferentes) tem pelo menos um zero no intervalo (a, b)[1].

Exemplo 2.1.1. Consideremos o problema de resolver

$$\operatorname{sen}^2\left(x + \frac{\pi}{4}\right) = x^3 - \frac{\pi}{4}x^2 - \frac{5\pi^2}{16}x - \frac{3\pi^3}{64}. \tag{2.2}$$

[1]Esta é uma consequência imediata do teorema do valor intermediário.

Este problema é equivalente a encontrar os zeros da seguinte função

$$f(x) = \operatorname{sen}^2\left(x + \frac{\pi}{4}\right) - x^3 + \frac{\pi}{4}x^2 + \frac{5\pi^2}{16}x + \frac{3\pi^3}{64}. \tag{2.3}$$

Os zeros exatos[2] desta função são $x_1 = 3\pi/4 \approx 2{,}3562$ e $x_2 = x_3 = -\pi/4 \approx -0{,}78540$ (veja a Figura 2.1).

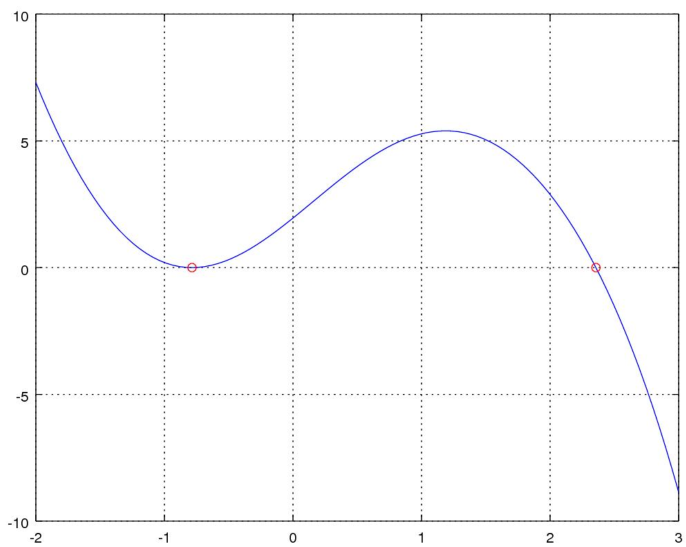

Figura 2.1: Esboço da função f do Exemplo 2.1.1.

Observamos que esta função é contínua e que, por exemplo, $f(-2) > 0$ e $f(3) < 0$, logo $f(-2) \cdot f(3) < 0$ e, de fato, f tem pelo menos um zero[3] no intervalo $(-2, 3)$.

Consideremos, então, uma função f contínua tal que $f(a) \cdot f(b) < 0$. O método da bisseção é iterativo, sendo que a primeira aproximação para uma solução de $f(x) = 0$ tomada como o ponto médio do intervalo (a, b), i.e.

$$x^{(1)} = \frac{a^{(1)} + b^{(1)}}{2}, \tag{2.4}$$

[2]O problema foi construído para que tivesse estas soluções.

[3]De fato, f tem três zeros no intervalo $(-2, 3)$.

onde $a^{(1)} = a$ e $b^{(1)} = b$. Daí, se ocorrer $f(x^{(1)}) = 0$ o problema está resolvido. Caso contrário, f tem pelo menos um zero num dos subintervalos $(a^{(1)}, x^{(1)})$ ou $(x^{(1)}, b^{(1)})$, pois $f(a^{(1)}) \cdot f(x^{(1)}) < 0$ ou $f(x^{(1)}) \cdot f(b^{(1)}) < 0$, respectivamente e exclusivamente. No primeiro caso, escolhemos $(a^{(2)}, b^{(2)}) = (a^{(1)}, x^{(1)})$ ou, no segundo caso, tomamos $(a^{(2)}, b^{(2)}) = (x^{(1)}, b^{(1)})$. Então, a segunda aproximação para uma solução é computada como

$$x^{(2)} = \frac{a^{(2)} + b^{(2)}}{2}. \tag{2.5}$$

Daí, o procedimento se repete até obtermos uma aproximação com a precisão desejada.

Exemplo 2.1.2. Consideremos o problema de encontrar um zero da função

$$f(x) = \operatorname{sen}^2\left(x + \frac{\pi}{4}\right) - x^3 + \frac{\pi}{4}x^2 + \frac{5\pi^2}{16}x + \frac{3\pi^3}{64}. \tag{2.6}$$

Do esboço de seu gráfico (Figura 2.1) vemos que $f(2) \cdot f(3) \neq 0$ sendo que o zero $x = 3\pi/4 \approx 2{,}3562$ de f está no intervalo $(2, 3)$. Então, aplicando o método da bisseção com intervalo inicial $(a^{(1)}, b^{(1)}) = (2, 3)$ e aproximação inicial $x^{(1)} = (a^{(1)} + b^{(1)})/2$, obtemos as aproximações apresentadas na Tabela 2.1.

Tabela 2.1: Resultados referentes ao Exemplo 2.1.2.

k	$a^{(k)}$	$b^{(k)}$	$x^{(k)}$	$f(a^{(k)}) \cdot f(x^{(k)})$
1	2,0000	3,0000	2,5000	-1
2	2,0000	2,5000	2,2500	1
3	2,2500	2,5000	2,3750	-1
4	2,2500	2,3750	2,3125	1
5	2,3125	2,3750	2,3438	1
6	2,3438	2,3750	2,3594	-1
7	2,3438	2,3594	2,3516	1
8	2,3516	2,3594	2,3555	1
9	2,3555	2,3594	2,3574	-1
10	2,3555	2,3574	2,3564	-1

2.1.1 Análise de convergência

Dada uma função estritamente monótona[4] e contínua $f : [a,b] \to \mathbb{R}$ com $f(a) \cdot f(b) < 0$, temos que o método da bisseção converge para o zero de f no intervalo (a,b).

De fato, como consequência imediata do teorema do valor intermediário, temos que f tem pelo menos um zero no intervalo (a,b). Agora, da hipótese de monotonicidade estrita, temos que f tem um único zero neste intervalo, o qual denotaremos por x^*.

Da construção das iteradas do método, temos

$$|x^{(k)} - x^*| \leq \frac{b^{(k)} - a^{(k)}}{2} \tag{2.7}$$
$$\leq \frac{b^{(k-1)} - a^{(k-1)}}{2^2} \tag{2.8}$$
$$\vdots \tag{2.9}$$
$$\leq \frac{b^{(1)} - a^{(1)}}{2^k}, \tag{2.10}$$

donde, temos a seguinte estimativa do erro de truncamento

$$|x^{(k)} - x^*| \leq \frac{b^{(1)} - a^{(1)}}{2^k}. \tag{2.11}$$

E, daí também, segue a convergência do método da bisseção, pois

$$\lim_{k\to\infty} |x^{(k)} - x^*| = \lim_{k\to\infty} \frac{b^{(1)} - a^{(1)}}{2^k} = 0. \tag{2.12}$$

Observação 2.1.1. No caso de f não ser estritamente monótona no intervalo (a,b), ainda podemos garantir a convergência do método da bisseção. Isto segue do fato de que após algumas iteradas, digamos k iteradas, a função f terá apenas um zero no intervalo $(a^{(k)}, b^{(k)})$. A partir daí, as estimativas acima podem ser aplicadas.

Exemplo 2.1.3. No Exemplo 2.1.2 aplicamos o método da bisseção para a função

$$f(x) = \operatorname{sen}^2\left(x + \frac{\pi}{4}\right) - x^3 + \frac{\pi}{4}x^2 + \frac{5\pi^2}{16}x + \frac{3\pi^3}{64}. \tag{2.13}$$

[4] Estritamente crescente ou estritamente decrescente.

no intervalo $(2, 3)$. Observando os resultados mostrados na Tabela 2.1, vemos que

$$|x^{(10)} - x^*| = 2{,}5\text{E}{-4}, \tag{2.14}$$

com $x^* = x_1 = 3\pi/4$. Observamos que este resultado é consistente com a estimativa do erro de truncamento (2.11), da qual temos

$$|x^{(10)} - x^*| \leq \frac{b^{(1)} - a^{(1)}}{2^{10}} \tag{2.15}$$

$$= \frac{1}{2^{10}} = 9{,}8\text{E}{-4}. \tag{2.16}$$

Observação 2.1.2. (Taxa de convergência.) A estimativa de convergência (2.11) também pode ser usada para mostrarmos que, assintoticamente, o método da bisseção tem a seguinte taxa de convergência linear

$$\left|x^{(k+1)} - x^{(k)}\right| \lesssim \frac{1}{2}\left|x^{(k)} - x^{(k-1)}\right|^{\mathbf{1}}. \tag{2.17}$$

2.1.2 Zeros de multiplicidade par

Sejam f uma função suave e x^* um zero de multiplicidade par de f. Observamos que o método da bisseção não é diretamente aplicável para aproximar x^*. Isto ocorre, pois, neste caso, x^* será um ponto de mínimo ou de máximo local de f, não havendo pontos a e b próximos de x^* tal que $f(a) \cdot f(b) < 0$.

Agora, sendo x^* é um zero de multiplicidade $2m$ de f, temos que ela admite a seguinte decomposição

$$f(x) = (x - x^*)^{2m} g(x), \tag{2.18}$$

onde g é uma função suave e $g(x^*) \neq 0$. Daí, a derivada de f

$$f'(x) = 2m(x - x^*)^{2m-1} g(x) + (x - x^*)^{2m} g'(x), \tag{2.19}$$

tem x^* como um zero de multiplicidade $2m - 1$ (ímpar) e, desta forma, podemos aplicar o método da bisseção em f' para aproximar x^*.

Exemplo 2.1.4. A função

$$f(x) = \operatorname{sen}^2\left(x + \frac{\pi}{4}\right) - x^3 + \frac{\pi}{4}x^2 + \frac{5\pi^2}{16}x + \frac{3\pi^3}{64}. \tag{2.20}$$

tem $x = -\pi/4 \approx -0{,}7854$ como um zero de multiplicidade par (veja Figura 2.1). Para aplicarmos o método da bisseção para aproximarmos este zero, primeiramente, derivamos f

$$f'(x) = 2\sin(x + \pi/4)\cos(x + \pi/4) - 3x^2 + \frac{\pi}{2}x + \frac{5\pi^2}{16}. \tag{2.21}$$

O esboço do gráfico de f' (Figura 2.2) mostra que $f'(-1) \cdot f'(0) < 0$ sendo que no intervalo $(-1, 0)$ f' tem um zero de multiplicidade ímpar. Então, aplicando o método da bisseção a f' no intervalo inicial $(a^{(1)}, b^{(1)}) = (-1,\ 0)$, obtemos os resultados apresentados na Tabela 2.2. Nesta tabela são apresentados as iteradas até a convergência da solução com precisão de 10^{-3}.

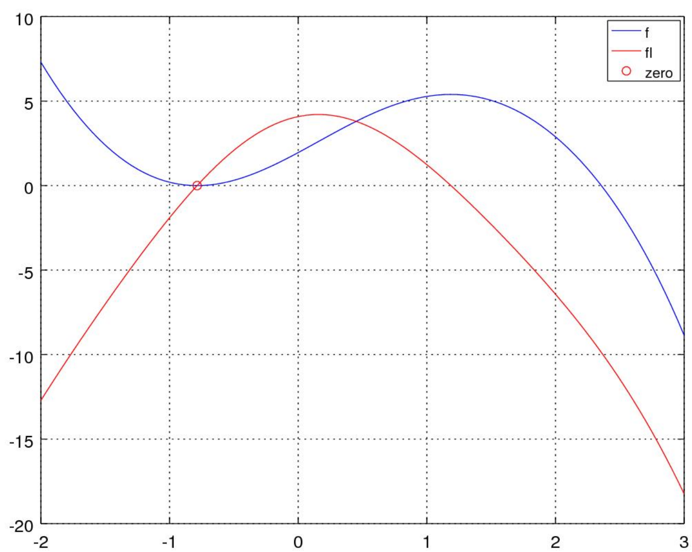

Figura 2.2: Esboço do gráfico da f e de sua derivada f' dada no Exemplo 2.1.4.

Exercícios

Exercício 2.1.1. Use o método da bisseção para aproximar um zero de $f(x) = x^3 \operatorname{sen}(x) - \cos(x)$, aplicando como intervalo inicial $(a^{(1)}, b^{(1)}) = (0{,}5,\ 1)$ e aproximação inicial $x^{(1)} = (a^{(1)} + b^{(1)})/2$. Faça, então, 6 iterações

Tabela 2.2: Resultados referentes ao Exemplo 2.1.2.

k	$a^{(k)}$	$b^{(k)}$	$x^{(k)}$	$f'(a^{(k)}) \cdot f'(x^{(k)})$
1	$-1{,}0000\text{E}{+}0$	$0{,}0000\text{E}{+}0$	$-5{,}0000\text{E}{-}1$	-1
2	$-1{,}0000\text{E}{+}0$	$-5{,}0000\text{E}{-}1$	$-7{,}5000\text{E}{-}1$	-1
3	$-1{,}0000\text{E}{+}0$	$-7{,}5000\text{E}{-}1$	$-8{,}7500\text{E}{-}1$	1
4	$-8{,}7500\text{E}{-}1$	$-7{,}5000\text{E}{-}1$	$-8{,}1250\text{E}{-}1$	1
5	$-8{,}1250\text{E}{-}1$	$-7{,}5000\text{E}{-}1$	$-7{,}8125\text{E}{-}1$	-1
6	$-8{,}1250\text{E}{-}1$	$-7{,}8125\text{E}{-}1$	$-7{,}9688\text{E}{-}1$	1
7	$-7{,}9688\text{E}{-}1$	$-7{,}8125\text{E}{-}1$	$-7{,}8906\text{E}{-}1$	1
8	$-7{,}8906\text{E}{-}1$	$-7{,}8125\text{E}{-}1$	$-7{,}8516\text{E}{-}1$	-1
9	$-7{,}8906\text{E}{-}1$	$-7{,}8516\text{E}{-}1$	$-7{,}8711\text{E}{-}1$	1
10	$-7{,}8711\text{E}{-}1$	$-7{,}8516\text{E}{-}1$	$-7{,}8613\text{E}{-}1$	1

de forma a obter a aproximação $x^{(7)}$ e forneça-a com 7 dígitos significativos por arredondamento.

Exercício 2.1.2. Considere que o método da bisseção para aproximar um zero de $f(x) = x^3 \operatorname{sen}(x) - \cos(x)$, aplicando como intervalo inicial $(a^{(1)}, b^{(1)}) = (0{,}5,\ 1)$ e aproximação inicial $x^{(1)} = (a^{(1)} + b^{(1)})/2$. Use a estimativa de convergência (2.11)

$$\left|x^{(k)} - x^*\right| \leq \frac{b^{(1)} - a^{(1)}}{2^k}, \tag{2.22}$$

para estimar o número mínimo de iterações k_{conv} necessárias para se obter a solução com exatidão de 10^{-4}. Então, compute $x^{(k_{conv})}$ e forneça-o com 6 dígitos significativos por arredondamento.

Exercício 2.1.3. Use o método da bisseção para encontrar uma aproximação com precisão de 10^{-4} do zero de

$$f(x) = (-x^2 + 1{,}154x - 0{,}332929)\cos(x) + x^2 - 1{,}154x + 0{,}332929 \tag{2.23}$$

no intervalo $(0{,}55,\ 0{,}65)$. Forneça a aproximação computada com 7 dígitos significativos por arredondamento.

2.2 Método da falsa posição

O método da falsa posição é uma variação do método da bisseção. Dada uma função f contínua, escolhemos um intervalo inicial (a, b) tal que $f(a) \cdot f(b) < 0$ (i.e. f tem sinais trocados nos pontos a e b). Então, uma aproximação para o zero de f neste intervalo é computada como o ponto de interseção da reta secante a f pelos pontos $(a, f(a))$ e $(b, f(b))$, i.e.

$$x = a - \frac{b - a}{f(b) - f(a)} f(a). \tag{2.24}$$

Veja a Figura 2.3.

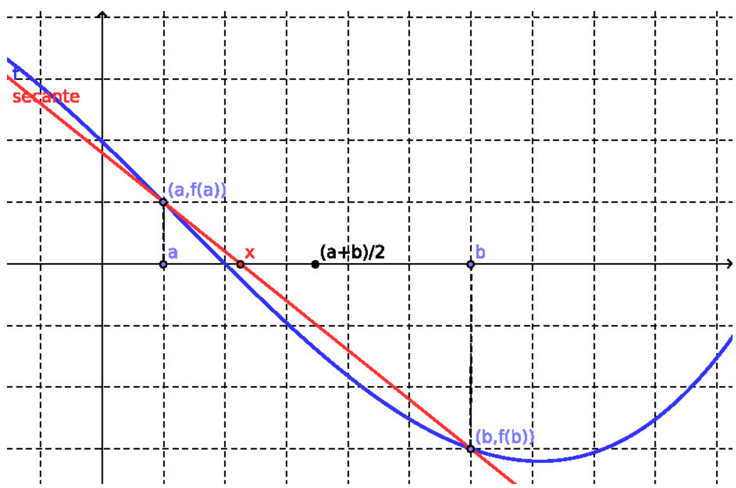

Figura 2.3: Ilustração do método da falsa posição (veja no Geogebra).

Mais explicitamente, o método da falsa posição consiste no seguinte procedimento iterativo:

1. Determinar um intervalo $(a^{(1)}, b^{(1)})$ tal que $f(a^{(1)}) \cdot f(b^{(1)}) < 0$.
2. Para $k = 1, 2, 3, \cdots, N$:

   2.1 $x^{(k)} = a^{(k)} - \dfrac{b^{(k)} - a^{(k)}}{f(b^{(k)}) - f(a^{(k)})} f(a^{(k)})$

   2.2 Verificar critério de parada.

   2.3 Se $f(a^{(k)}) \cdot f(x^{(k)}) < 0$, então $a^{(k+1)} = a^{(k)}$ e $b^{(k+1)} = x^{(k)}$.

2.4 Se $f(x^{(k)}) \cdot f(b^{(k)}) > 0$, então $a^{(k+1)} = x^{(k)}$ e $b^{(k+1)} = b^{(k)}$.

Exemplo 2.2.1. Consideremos o problema de aproximar o zero de

$$f(x) = \operatorname{sen}^2\left(x + \frac{\pi}{4}\right) - x^3 + \frac{\pi}{4}x^2 + \frac{5\pi^2}{16}x + \frac{3\pi^3}{64}. \tag{2.25}$$

no intervalo $(0, 3)$. A Tabela 2.3 mostra os resultados obtidos da aplicação do método da falsa posição com intervalo inicial $(a^{(1)}, b^{(1)}) = (2, 3)$. Aqui, o método foi iterado até a convergência com cinco dígitos significativos.

Tabela 2.3: Resultados referentes ao Exemplo 2.2.1.

k	$a^{(k)}$	$b^{(k)}$	$x^{(k)}$	$f'(a^{(k)}) \cdot f'(x^{(k)})$
1	2,0000	3,0000	2,2455	1
2	2,2455	3,0000	2,3240	1
3	2,3240	3,0000	2,3470	1
4	2,3470	3,0000	2,3536	1
5	2,3536	3,0000	2,3555	1
6	2,3555	3,0000	2,3560	1
7	2,3560	3,0000	2,3561	1
8	2,3561	3,0000	2,3562	1
9	2,3562	3,0000	2,3562	1
10	2,3562	3,0000	2,3562	1

Exercícios

Exercício 2.2.1. Use o método da falsa posição para aproximar um zero de $f(x) = x^3 \operatorname{sen}(x) - \cos(x)$, aplicando como intervalo inicial $(a^{(1)}, b^{(1)}) = (0{,}5,\ 1)$ e aproximação inicial

$$x^{(1)} = a^{(1)} - \frac{b^{(1)} - a^{(1)}}{f(b^{(1)}) - f(a^{(1)})} f(a^{(1)}). \tag{2.26}$$

Faça, então, 4 iterações deste método de forma a obter a aproximação $x^{(5)}$ e forneça-a com 7 dígitos significativos por arredondamento.

Exercício 2.2.2. Use o método da bisseção para encontrar uma aproximação com precisão de 10^{-4} do zero de

$$f(x) = (-x^2 + 1{,}154x - 0{,}332929)\cos(x) + x^2 - 1{,}154x + 0{,}332929 \tag{2.27}$$

no intervalo (0,55, 0,65). Forneça a aproximação computada com 7 dígitos significativos por arredondamento.

2.3 Iteração de ponto fixo

O ponto fixo de uma função dada g é o ponto x tal que

$$g(x) = x. \tag{2.28}$$

Geometricamente, pontos fixos são interseções do gráfico da g com a reta $y = x$, veja a Figura 2.4.

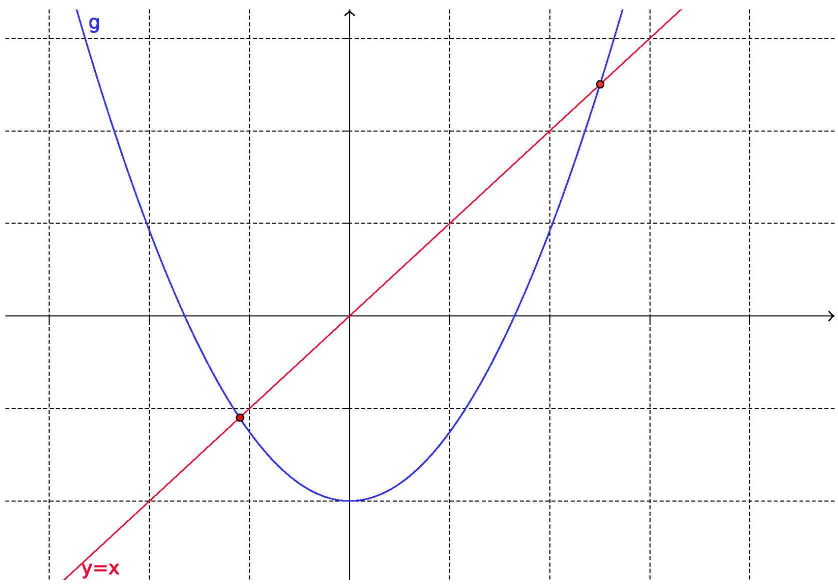

Figura 2.4: Exemplos de pontos fixos (veja no Geogebra).

Observamos que toda equação de uma incógnita pode ser reescrita de forma equivalente como um problema de ponto fixo.

Exemplo 2.3.1. Consideremos o problema de resolver

$$\operatorname{sen}^2\left(x + \frac{\pi}{4}\right) = x^3 - \frac{\pi}{4}x^2 - \frac{5\pi^2}{16}x - \frac{3\pi^3}{64}. \tag{2.29}$$

Podemos reescrevê-la como o problema de se obter os zeros da seguinte função

$$f(x) = \operatorname{sen}^2\left(x + \frac{\pi}{4}\right) - x^3 + \frac{\pi}{4}x^2 + \frac{5\pi^2}{16}x + \frac{3\pi^3}{64}. \tag{2.30}$$

Por sua vez, este problema é equivalente aos seguintes problemas de ponto fixo (entre outros):

$$a)\ g_1(x) = \frac{16}{5\pi^2}\left[-\operatorname{sen}^2\left(x+\frac{\pi}{4}\right) + x^3 - \frac{\pi}{4}x^2 - \frac{3\pi^3}{64}\right] = x. \tag{2.31}$$

$$b)\ g_2(x) = \sqrt[3]{\operatorname{sen}^2\left(x+\frac{\pi}{4}\right) + \frac{\pi}{4}x^2 + \frac{5\pi^2}{16}x + \frac{3\pi^3}{64}} \tag{2.32}$$

Na Figura 2.5 podemos observar que os zeros da f (a saber, $x_1 = 3\pi/4 \approx 2{,}3562$ e $x_2 = x_3 = -\pi/4 \approx -0{,}78540$) coincidem com os pontos fixos das funções g_1 e g_2.

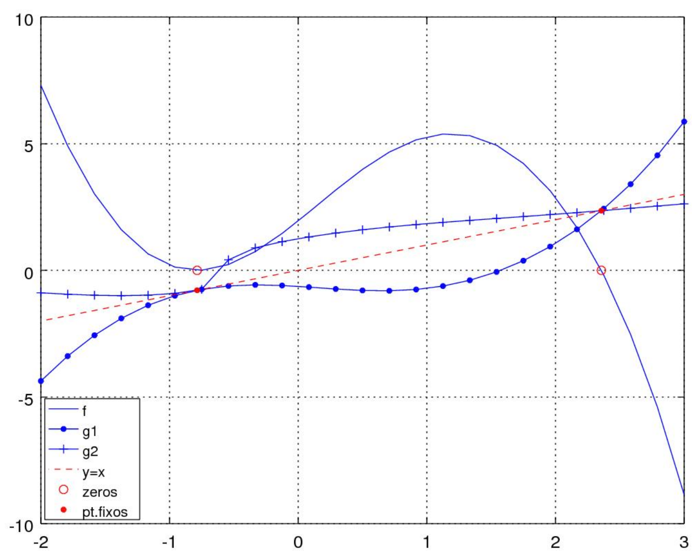

Figura 2.5: Esboço da função f do Exemplo 2.3.1.

Em muitos casos, é possível obter aproximações de um ponto fixo de uma dada função g pela chamada **iteração de ponto fixo**:

$$x^{(1)} = \text{aprox. inicial} \tag{2.33}$$

$$x^{(k+1)} = g(x^{(k)}), \quad k = 1, 2, 3, \ldots \tag{2.34}$$

Exemplo 2.3.2. Observemos que para a função g_1 dada no Exemplo 2.3.1, temos as seguintes iterações:

$$x^{(1)} = -0{,}70000, \tag{2.35}$$
$$x^{(2)} = -0{,}70959, \tag{2.36}$$
$$x^{(3)} = -0{,}71716, \tag{2.37}$$
$$\vdots \tag{2.38}$$
$$x^{(100)} = -0{,}77862, \tag{2.39}$$
$$\vdots \tag{2.40}$$
$$x^{(1000)} = -0{,}78466, \tag{2.41}$$
$$\vdots \tag{2.42}$$
$$x^{(20000)} = -0{,}78536. \tag{2.43}$$

Ou seja, neste caso as iterações de ponto fixo convergem (lentamente) para o ponto fixo $x = -\pi/4 \approx -0{,}78540$.

Agora, se usarmos a iteração de ponto fixo com esta mesma função para aproximar o ponto fixo $x = 3\pi/4 \approx 2{,}3562$, obtemos

$$x^{(1)} = 2{,}50000, \tag{2.44}$$
$$x^{(2)} = 2{,}9966, \tag{2.45}$$
$$x^{(3)} = 5{,}8509, \tag{2.46}$$
$$\vdots \tag{2.47}$$
$$x^{(8)} = 4{,}8921e \times 10^{121}. \tag{2.48}$$

Donde observamos que as iterações divergem rapidamente.

Entretanto, se usarmos a iteração de ponto fixo com a função f_2 dada no Exemplo 2.3.1, obtemos

$$x^{(1)} = 2{,}50000, \tag{2.49}$$
$$x^{(2)} = 2{,}4155, \tag{2.50}$$
$$x^{(3)} = 2{,}3805, \tag{2.51}$$
$$\vdots \tag{2.52}$$

$$x^{(10)} = 2{,}3562. \tag{2.53}$$

A qual, portanto, converge para o ponto fixo esperado.

Este último exemplo (Exemplo 2.3.2) mostra que a iteração do ponto fixo nem sempre é convergente. Antes de vermos condições suficientes para a convergência, vejamos sua interpretação geométrica.

2.3.1 Interpretação geométrica

A Figura 2.6 apresenta o caso de uma iteração de ponto fixo convergente. As iterações iniciam-se no ponto $x^{(1)}$ e seguem para $x^{(2)} = g(x^{(1)})$ e $x^{(3)} = g(x^{(2)})$.

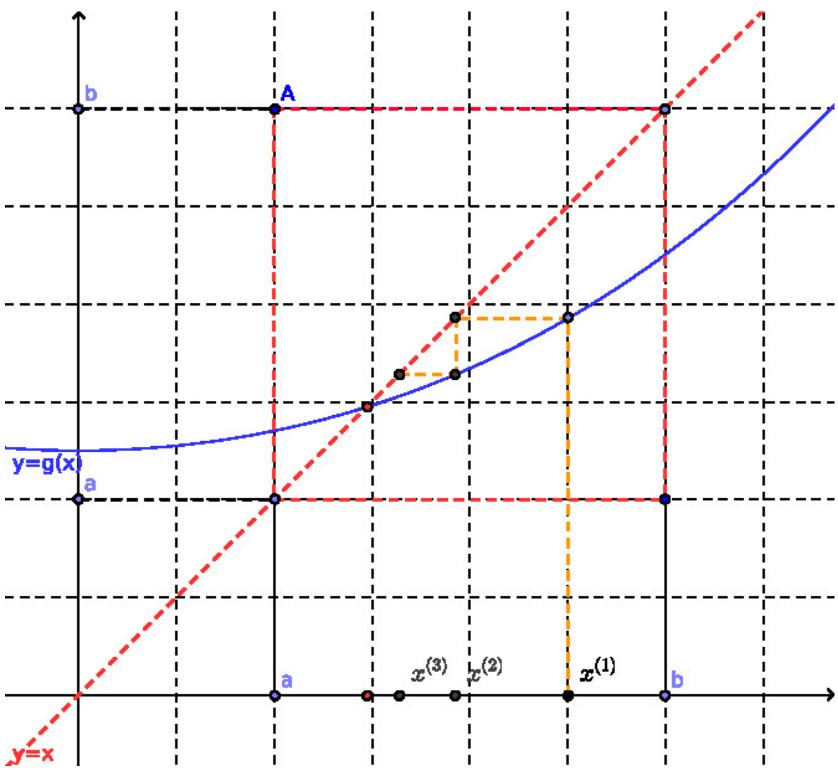

Figura 2.6: Interpretação geométrica da iteração de ponto fixo.

2.3.2 Análise de convergência

O seguinte teorema nos fornece condições suficientes para a convergência das iterações de ponto fixo.

Teorema 2.3.1. *(*Teorema do ponto fixo) Seja uma dada função g continuamente diferenciável satisfazendo

a) $g([a,b]) \subset [a,b]$,

b) $|g'(x)| < K < 1$ para todo $x \in [a,b]$.

Então, g tem um único ponto fixo $x^* \in [a,b]$ e as iterações $x^{(k+1)} = x^{(k)}$, $k = 1, 2, 3, \ldots$, convergem para x^*, para qualquer escolha de $x^{(1)} \in [a,b]$.

Demonstração. Da hipótese b), temos que g é uma contração com

$$|g(x) - g(y)| < K \cdot |x - y|, \, \forall x{,}y \in [a,b]. \tag{2.54}$$

Com isso, da hipótese a) e tomando $x^{(1)} \in [a,b]$, temos

$$\begin{align}
|x^{(k+1)} - x^{(k)}| &= |g(x^{(k)}) - g(x^{(k-1)})| \tag{2.55}\\
&\leq K|x^{(k)} - x^{(k-1)}| \tag{2.56}\\
&\vdots \tag{2.57}\\
&\leq K^{k-1}|x^{(2)} - x^{(1)}|, \tag{2.58}
\end{align}$$

para todo $k = 2, 3, \ldots$. Como $K < 1$, temos $|x^{(k+1)} - x^{(k)}| \to 0$ quando $k \to \infty$ e, portanto, $x^{(k)}$ converge para algum $x^* \in [a,b]$.

De fato, x^* é ponto fixo de g, pois da continuidade da g, temos

$$x^* = \lim_{k\to\infty} x^{(k+1)} = \lim_{k\to\infty} g(x^{(k)}) = g(x^*). \tag{2.59}$$

Por fim, x^* é único, pois assumindo a existência de outro ponto fixo $x^{**} \neq x^*$ teríamos

$$|x^* - x^{**}| = |g(x^*) - g(x^{**})| < K|x^* - x^{**}|. \tag{2.60}$$

□

Agora, dado um problema de encontrar um zero de uma dada função f, i.e. $f(x) = 0$, podemos construir uma função g para a iteração de ponto fixo associada da seguinte forma:

$$f(x) = 0 \Leftrightarrow \underbrace{x - \alpha f(x)}_{g(x)} = x, \tag{2.61}$$

com $\alpha \in \mathbb{R}$ escolhido de forma a satisfazer as hipóteses do teorema do ponto fixo (Teorema 2.3.1).

Exemplo 2.3.3. Retornamos ao problema de encontrar o zero da função

$$f(x) = \operatorname{sen}^2\left(x + \frac{\pi}{4}\right) - x^3 + \frac{\pi}{4}x^2 + \frac{5\pi^2}{16}x + \frac{3\pi^3}{64}. \tag{2.62}$$

no intervalo [2,3]. Para construir uma função g para a iteração de ponto fixo neste intervalo, podemos tomar

$$g(x) = x - \alpha f(x), \tag{2.63}$$

com $\alpha = -0{,}1$. A Figura 2.7 mostra esboços dos gráficos de g e $|g'|$ no intervalos $[2, 3]$ e podemos observar que esta escolha de α faz com que a g satisfaça o teorema do ponto fixo.

Então, fazendo as iterações de ponto fixo com aproximação inicial $x^{(1)} = 2{,}6$, obtemos os resultados apresentados na Tabela 2.4.

Tabela 2.4: Resultados referentes ao Exemplo 2.3.3.

k	$x^{(k)}$	$\|x^{(k)} - x^{(k-1)}\|$
1	2,6000	-x-
2	2,3264	2,7E−1
3	2,3553	2,9E−2
4	2,3562	8,4E−4
5	2,3562	1,1E−5

Observação 2.3.1. (Taxa de convergência) A iteração de ponto fixo tem taxa de convergência linear

$$|x^{(k+1)} - x^{(k)}| < K|x^{(k)} - x^{(k-1)}|^{\mathbf{1}}, \tag{2.64}$$

onde $K > 0$ é a constante dada na hipótese b) do teorema do ponto fixo (Teorema 2.3.1). Além disso, isso mostra que quanto menor o valor da constante K, mais rápida será a convergência das iterações de ponto fixo.

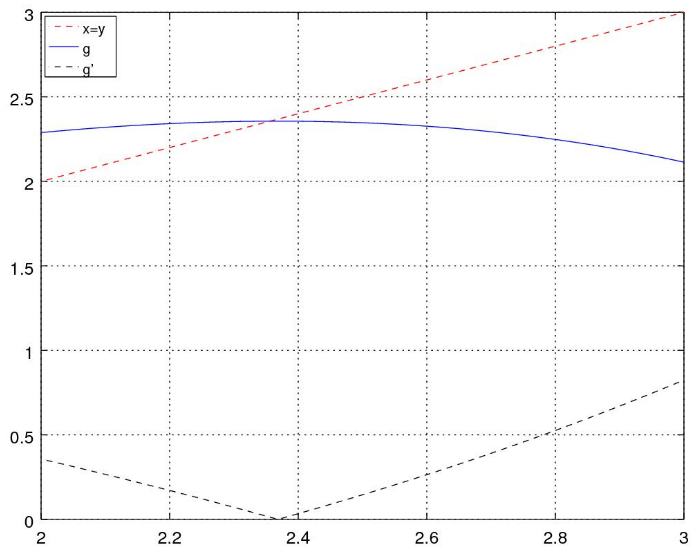

Figura 2.7: Esboço dos gráficos de g e $|g'|$ discutidas no Exemplo 2.3.3.

Exercícios

Exercício 2.3.1. Considere o problema de computar uma aproximação do zero de $f(x) = x - \cos(x)$. Resolva-o aplicando a iteração de ponto fixo para a função auxiliar

$$g(x) = x - \alpha f(x), \tag{2.65}$$

restrita ao intervalo $[a, b] = [0.5, 1]$ com aproximação inicial $x^{(1)} = (a + b)/2$. Escolha o melhor valor de α entre os seguintes:

1. $\alpha = 1$
2. $\alpha = 0{,}5$
3. $\alpha = -0{,}5$
4. $\alpha = 0{,}6$

Então, compute uma aproximação do zero de f com 5 dígitos significativos de precisão.

2.4 Método de Steffensen

O método de Steffensen[5] é uma aplicação do método de aceleração de convergência Δ^2 de Aitken[6] à iteração de ponto fixo.

2.4.1 Acelerador Δ^2 de Aitken

Seja dada uma sequência $(x^{(k)})_{k=1}^{\infty}$ monotonicamente convergente para x^*. Assumamos que k seja suficientemente grande tal que

$$\frac{x^{(k+1)} - x^*}{x^{(k)} - x^*} \approx \frac{x^{(k+2)} - x^*}{x^{(k+1)} - x^*}. \tag{2.66}$$

Então, isolando x^* obtemos

$$x^* \approx \frac{x^{(k)}x^{(k+2)} - (x^{(k+1)})^2}{x^{(k)} - 2x^{(k+1)} + x^{(k+2)}}. \tag{2.67}$$

Ainda, somando e subtraindo $(x^{(k)})^2$ e $2x^{(k)}x^{(k+1)}$ no numerador acima e rearranjando os termos, obtemos

$$x^* \approx x^{(k)} - \frac{(x^{(k+1)} - x^{(k)})^2}{x^{(k+2)} - 2x^{(k+1)} + x^{(k)}}. \tag{2.68}$$

O observado acima, nos motiva a introduzir o acelerador Δ^2 de Aitken

$$\Delta^2\{x^{(k)},x^{(k+1)},x^{(k+2)}\} := x^{(k)} - \frac{(x^{(k+1)} - x^{(k)})^2}{x^{(k+2)} - 2x^{(k+1)} + x^{(k)}}. \tag{2.69}$$

Exemplo 2.4.1. Consideremos o problema de encontrar o zero da função

$$f(x) = \operatorname{sen}^2\left(x + \frac{\pi}{4}\right) - x^3 + \frac{\pi}{4}x^2 + \frac{5\pi^2}{16}x + \frac{3\pi^3}{64}. \tag{2.70}$$

no intervalo $[2,3]$. Para tanto, podemos aplicar a iteração de ponto fixo dada por

$$x^{(k+1)} = g(x^{(k)}) := x^{(k)} - \alpha f(x^{(k)}), \quad k = 1,2,\dots, \tag{2.71}$$

com $\alpha = -0{,}05$ e $x^{(1)} = 2{,}6$. Na Tabela 2.5 temos os valores das iteradas $x^{(k)}$ e das correções $\Delta^2 = \Delta^2\{x^{(k)},x^{(k+1)},x^{(k+2)}\}$ de Aitken. Neste caso, a aceleração de convergência é notável.

[5] Johan Frederik Steffensen, matemático e estatístico dinamarquês, 1873 - 1961. Fonte: Wikipedia.

[6] Alexander Aitken, matemático neozelandês, 1895 - 1967. Fonte: Wikipedia.

Tabela 2.5: Resultados referentes ao Exemplo 2.4.1.

k	$x^{(k)}$	Δ^2
1	2,6000	-x-
2	2,4632	-x-
3	2,4073	2,3687
4	2,3814	2,3590
5	2,3688	2,3569
6	2,3625	2,3564
7	2,3594	2,3562
8	2,3578	2,3562

2.4.2 Algoritmo de Steffensen

O método de Steffensen consiste em aplicar o acelerador Δ^2 de Aitken à iteração de ponto fixo. Mais especificamente, sejam uma aproximação inicial $x^{(1)}$ e uma iteração de ponto fixo

$$x^{(k+1)} = g(x^{(k)}), \quad k = 1,2,\ldots. \tag{2.72}$$

O algoritmo de Steffensen pode ser descrito como segue:

1. Fazemos $x = x^{(1)}$.
2. Para $k = 1,2,3,\ldots,N-1$:
   (a) Computamos $x_1 = g(x^{(k)})$.
   (b) Computamos $x_2 = g(x_1)$.
   (c) Computamos $x^{(k+1)} = \Delta^2\{x^{(k)},x_1,x_2\}$
   (d) Verificamos o critério de parada.

Exemplo 2.4.2. Retornemos ao exemplo anterior (Exemplo 2.4.1. Na Tabela 2.6 temos os valores das iteradas de Steffensen $x^{(k)}$ e do indicador de convergência $|x^{(k)} - x^{(k-1)}|$. Observe que os resultados são compatíveis com aqueles obtidos no último exemplo.

Exercícios

Tabela 2.6: Resultados referentes ao Exemplo 2.4.2.

k	$x^{(k)}$	$\lvert x^{(k)} - x^{(k-1)}\rvert$
1	2,6000	-x-
2	2,3687	2,3E$-$1
3	2.3562	1,2E$-$2
4	2,3562	4,2E$-$5

Exercício 2.4.1. Use o método de Steffensen para obter uma aproximação do zero de $f(x) = x^3 \operatorname{sen}(x) - \cos(x)$ no intervalo $[0{,}5, 1]$ com precisão de 10^{-6}.

2.5 Método de Newton

Seja x^* um zero de uma dada função f, i.e. $f(x^*) = 0$. Usando de expansão em polinômio de Taylor da f em um dado ponto $\tilde{x}$, temos

$$f(x^*) = f(\tilde{x}) + f'(\tilde{x})(x^* - \tilde{x}) + O((x^* - \tilde{x})^2). \tag{2.73}$$

Como $f(x^*) = 0$, temos

$$x^* + O((x^* - \tilde{x})^2) = \tilde{x} - \frac{f(\tilde{x})}{f'(\tilde{x})}. \tag{2.74}$$

Esta última expressão nos indica que dada uma aproximação $\tilde{x}$ do zero de f a expressão

$$\tilde{x} - \frac{f(\tilde{x})}{f'(\tilde{x})}, \tag{2.75}$$

aproxima x^* com um erro da ordem de $(x^* - \tilde{x})^2$.

Estes observações nos levam a **iteração de Newton**[7]

$$x^{(1)} = \text{aprox. inicial}, \tag{2.76}$$

$$x^{(k+1)} = x^{(k)} - \frac{f(x^{(k)})}{f'(x^{(k)})}, \tag{2.77}$$

com $k = 1, 2, \ldots$.

[7]Sir Isaac Newton, matemático e físico inglês, 1642 - 1726/27. Fonte: Wikipedia.

Exemplo 2.5.1. Retornamos ao problema de encontrar o zero da função

$$f(x) = \operatorname{sen}^2\left(x + \frac{\pi}{4}\right) - x^3 + \frac{\pi}{4}x^2 + \frac{5\pi^2}{16}x + \frac{3\pi^3}{64}. \tag{2.78}$$

no intervalo [2,3]. Então, fazendo as iterações de Newton com aproximação inicial $x^{(1)} = 2{,}6$, obtemos os resultados apresentados na Tabela 2.7.

Tabela 2.7: Resultados referentes ao Exemplo 2.5.1.

k	$x^{(k)}$	$\lvert x^{(k)} - x^{(k-1)}\rvert$
1	2,6000	-x-
2	2,3836	2,2E−1
3	2,3566	2,7E−2
4	2,3562	3,9E−4
5	2,3562	8,3E−8

Observação 2.5.1. O método de Newton é uma iteração de ponto fixo ótima. Do Teorema do ponto fixo (Teorema 2.3.1), uma iteração de ponto fixo

$$x^{k+1} = g(x^{(k)}) := x^{(k)} - \alpha f(x) \tag{2.79}$$

tem taxa de convergência[8]

$$|x^{(k+1)} - x^{(k)}| \leq K|x^{(k)} - x^{(k-1)}|, \tag{2.80}$$

com K tal que $|g'(x)| = |1 - \alpha f'(x)| < K < 1$. Isto nos indica que a melhor escolha para α é

$$\alpha = \frac{1}{f'(x)}, \tag{2.81}$$

de forma que (2.79) coincide com a iteração de (2.77).

2.5.1 Interpretação geométrica

Dada uma aproximação $x^{(k)}$ de um zero de uma dada função f, a iteração de Newton fornece uma nova aproximação $x^{(k+1)}$ com

$$x^{(k+1)} = x^{(k)} - \frac{f(x^{(k)})}{f'(x^{(k)})}. \tag{2.82}$$

[8]Supondo as demais hipóteses do Teorema 2.3.1.

Subtraindo $x^{(k+1)}$ e multiplicando por $-f'(x^{(k)})$, obtemos

$$0 = f'(x^{(k)})(x^{(k+1)} - x^{(k)}) + f(x^{(k)}), \tag{2.83}$$

Observemos que o lado direito desta última equação corresponde a expressão da reta tangente ao gráfico de f pelo ponto $(x^{(k)}, f(x^{(k)}))$, avaliada em $x^{(k+1)}$. Mais precisamente, a equação desta reta tangente é

$$y = f'(x^{(k)})(x - x^{(k)}) + f(x^{(k)}) \tag{2.84}$$

e a equação (2.83) nos informa que em $x = x^{(k+1)}$ a reta tangente cruza o eixo x.

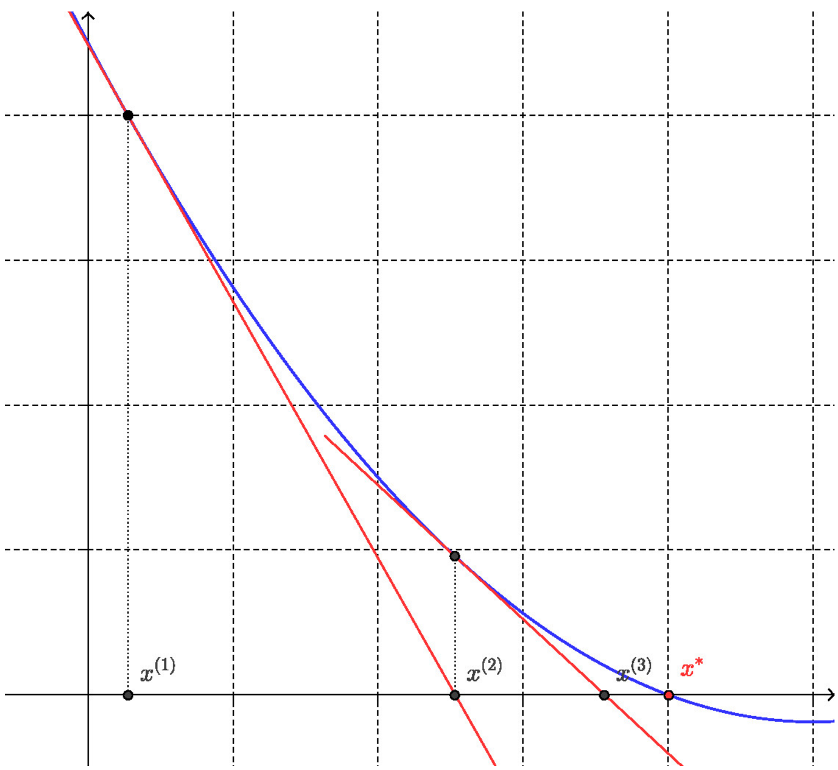

Figura 2.8: Interpretação geométrica das iterações de Newton. Veja no Geogebra.

Destas observações, concluímos que a iterada $x^{(k+1)}$ do método de Newton corresponde ao ponto de interseção da reta tangente ao gráfico da f pelo ponto $(x^k, f(x^k))$. Veja a Figura 2.8.

Exemplo 2.5.2. Consideremos que o método de Newton seja usado para aproximarmos o zero de

$$f(x) = (x - 1)e^{-x^2}. \tag{2.85}$$

Observemos que esta função tem $x = 1$ como seu único zero. Agora, se escolhermos $x^{(1)} = 0{,}5$ as iterações de Newton convergem para este zero, mas, se escolhermos $x^{(1)} = 1{,}5$ não (veja a Tabela 2.8).

Tabela 2.8: Resultados referentes ao Exemplo 2.5.2

k	$x^{(k)}$	$x^{(k+1)}$
1	5,0000E−1	1,5000E+0
2	8,3333E−1	2,5000E+0
3	9,6377E−1	2,7308E+0
4	9,9763E−1	2,9355E+0
5	9,9999E−1	3,1223E+0
6	1,0000E+0	3,2955E+0
7	1,0000E+0	3,4580E+0

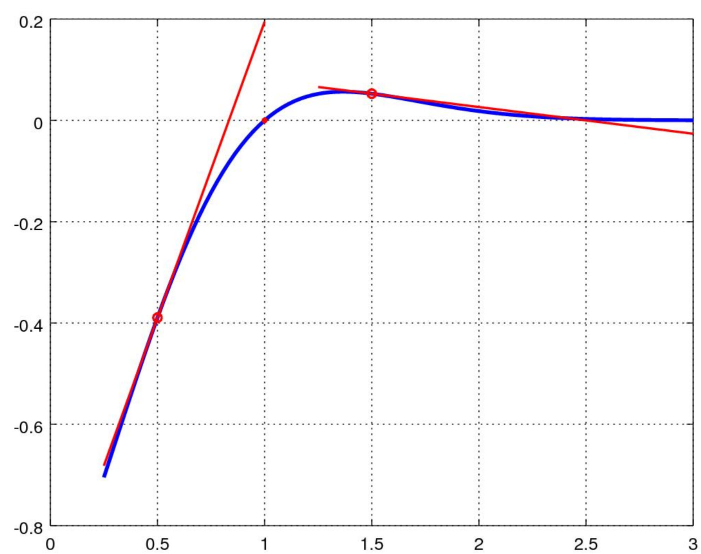

Figura 2.9: Escolha da aproximação inicial para o método de Newton.

Embora ambas aproximações iniciais estão a mesma distância da solução $x = 1$, quando tomamos $x^{(1)} = 1{,}5$ as iterações irão divergir, como podemos observar da interpretação geométrica dada na Figura 2.9.

2.5.2 Análise de convergência

Seja x^* o zero de uma dada função f duas vezes continuamente diferenciável com $f'(x) \neq 0$ para todo $x \in [x^* - \varepsilon_0, x^* + \varepsilon_0]$ para algum $\varepsilon_0 > 0$. Seja, também, $(x^{(k)})_{k=1}^{\infty}$ a sequência das iteradas de Newton

$$x^{(k+1)} = x^{(k)} - \frac{f(x^{(k)})}{f'(x^{(k)})}, \quad k = 1, 2, \dots, \tag{2.86}$$

com aproximação inicial $x^{(1)} \in (x^* - \varepsilon_0, x^* + \varepsilon_0)$. Então, do polinômio de Taylor de grau 1 de f em torno de $x^{(1)}$, temos

$$f(x^*) = f(x^{(1)}) + f'(x^{(1)})(x^* - x^{(1)}) + \frac{f''(\xi^{(1)})}{2!}(x^* - x^{(1)})^2, \tag{2.87}$$

onde $\xi^{(1)}$ está entre $x^{(1)}$ e ξ. Daí, rearranjamos os termos e notamos que $f(x^*) = 0$ para obtermos

$$x^* - \left[x^{(1)} - \frac{f(x^{(1)})}{f'(x^{(1)})}\right] = \frac{f''(\xi^{(1)})}{2f'(x^{(1)})}(x^* - x^{(1)})^2. \tag{2.88}$$

Então, da iteração de Newton (2.86), temos

$$x^* - x^{(2)} = \frac{f''(\xi^{(1)})}{2f'(x^{(1)})}(x^* - x^{(1)})^2 \tag{2.89}$$

Logo,

$$|x^* - x^{(2)}| \leq C|x^* - x^{(1)}|^2, \tag{2.90}$$

com

$$C = \sup_{x,y \in [x^* - \varepsilon, x^* + \varepsilon]} \left|\frac{f''(x)}{2f'(y)}\right|. \tag{2.91}$$

Segue, então, que se $x^{(1)} \in (x^* - \varepsilon, x^* + \varepsilon)$ para algum $\varepsilon > 0$ tal que

$$C|x^* - x|^2 < 1, \quad \forall x \in (x^* - \varepsilon, x^* + \varepsilon), \tag{2.92}$$

então $x^{(2)} \in (x^* - \varepsilon, x^* + \varepsilon)$.

Logo, por indução matemática[9], temos que o método de Newton tem taxa de convergência quadrática

$$|x^{(k+1)} - x^*| \leq C|x^{(k)} - x^*|^{\mathbf{2}}, \tag{2.93}$$

para qualquer escolha de $x^{(1)}$ suficientemente próximo de x^*, i.e. $x^{(1)} \in (x^* - \varepsilon, x^* + \varepsilon)$.

Observação 2.5.2. O intervalo $(x^* - \varepsilon, x^* + \varepsilon)$ é chamado de **bacia de atração** do método de Newton.

Exemplo 2.5.3. Retornamos ao problema de encontrar o zero da função

$$f(x) = \operatorname{sen}^2\left(x + \frac{\pi}{4}\right) - x^3 + \frac{\pi}{4}x^2 + \frac{5\pi^2}{16}x + \frac{3\pi^3}{64}. \tag{2.94}$$

no intervalo [2,3]. Este problema foi construído de forma que $x^* = 3\pi/4$ é um zero de f. Então, fazendo as iterações de Newton com aproximação inicial $x^{(1)} = 2{,}6$, obtemos os resultados apresentados na Tabela 2.9, os quais evidenciam a convergência quadrática das iterações computadas.

Tabela 2.9: Resultados referentes ao Exemplo 2.5.3.

k	$x^{(k)}$	$\lvert x^{(k)} - x^*\rvert$
1	2,6000	2,4E−01
2	2,3836	2,7E−02
3	2,3566	3,9E−04
4	2,3562	8,3E−08
5	2,3562	3,6E−15

2.5.3 Zeros múltiplos

Na análise de convergência acima foi necessário assumir que $f'(x) \neq 0$ para todo x em uma vizinha do zero x^* da função f. Isto não é possível no caso de x^* ser um zero duplo pois, então, $f'(x^*) = 0$. Neste caso, podemos aplicar o método de Newton a $f'(x)$, a qual tem x^* como um zero simples.

Exemplo 2.5.4. Consideremos o problema de aproximar o zero da função

$$f(x) = \operatorname{sen}^2\left(x + \frac{\pi}{4}\right) - x^3 + \frac{\pi}{4}x^2 + \frac{5\pi^2}{16}x + \frac{3\pi^3}{64}. \tag{2.95}$$

[9] Veja o exercício 2.5.3.

no intervalo $[-1{,}0]$. Este problema foi construído de forma que $x^* = -\pi/4$ é um zero duplo de f. Então, aplicamos o método de Newton a

$$f'(x) = 2\,\text{sen}\left(x + \frac{\pi}{4}\right)\cos\left(x + \frac{\pi}{4}\right) - 3x\check{s} + \frac{\pi}{2}x + \frac{5\pi^2}{16}. \tag{2.96}$$

Ou seja, as iterações de Newton são

$$x^{(k+1)} = x^{(k)} - \frac{f'(x^{(k)})}{f''(x^{(k)})}, \quad k = 1{,}2,\dots, \tag{2.97}$$

sendo $x^{(1)}$ uma aproximação inicial. Na Tabela 2.10, temos os resultados obtidos da computação destas iterações com $x^{(1)} = -0{,}5$.

Tabela 2.10: Resultados referentes ao Exemplo 2.5.4.

k	$x^{(k)}$	$\|x^{(k)} - x^*\|$
1	$-5{,}0000\text{E}{-1}$	$2{,}9\text{E}{-01}$
2	$-8{,}3407\text{E}{-1}$	$4{,}9\text{E}{-02}$
3	$-7{,}8619\text{E}{-1}$	$7{,}9\text{E}{-04}$
4	$-7{,}8540\text{E}{-1}$	$2{,}3\text{E}{-07}$
5	$-7{,}8540\text{E}{-1}$	$1{,}9\text{E}{-14}$

Observação 2.5.3. No caso de zeros de multiplicidade n de uma dada função f, podemos aplicar o método de Newton a derivada $n-1$ de f.

Exercícios

Exercício 2.5.1. Use o método de Newton para obter uma aproximação do zero de $f(x) = x^3\,\text{sen}(x) - \cos(x)$ no intervalo $[0{,}5, 1]$ com precisão de 10^{-6}.

Exercício 2.5.2. Use o método de Newton para obter uma aproximação do zero de

$$f(x) = (-x^2 + 1{,}154x - 0{,}332929)\cos(x) + x^2 - 1{,}154x + 0{,}332929 \tag{2.98}$$

no intervalo $(0{,}55,\ 0{,}65)$ com precisão de 10^-5.

Exercício 2.5.3. Complete a demonstração por indução matemática de que o método de Newton tem taxa de convergência quadrática.

2.6 Método da secante

O método da secante é uma variação do método de Newton. Observemos que para duas aproximações $x^{(k)}$ e $x^{(k-1)}$ suficientemente próximas, temos[10]

$$f'(x^{(k)}) \approx \frac{f(x^{(k)}) - f(x^{(k-1)})}{x^{(k)} - x^{(k-1)}}. \tag{2.99}$$

Assim sendo, substituindo esta aproximação em (2.77), obtemos as iterações do método da secante:

$$x^{(1)}, x^{(2)} = \text{aprox. iniciais}, \tag{2.100}$$

$$x^{(k+1)} = x^{(k)} - f(x^{(k)})\frac{x^{(k)} - x^{(k-1)}}{f(x^{(k)}) - f(x^{(k-1)})}, \tag{2.101}$$

para $k = 2,3,\ldots$.

Exemplo 2.6.1. Consideremos o problema de encontrar o zero da função

$$f(x) = \operatorname{sen}^2\left(x + \frac{\pi}{4}\right) - x^3 + \frac{\pi}{4}x^2 + \frac{5\pi^2}{16}x + \frac{3\pi^3}{64}. \tag{2.102}$$

no intervalo [2,3]. Então, fazendo as iterações do método da secante com aproximações iniciais $x^{(1)} = 2{,}6$ e $x^{(2)} - 2{,}5$, obtemos os resultados apresentados na Tabela 2.11.

Tabela 2.11: Resultados referentes ao Exemplo 2.6.1.

k	$x^{(k-1)}$	$x^{(k)}$	$\lvert x^{(k)} - x^{(k-1)}\rvert$
2	2,6000	2,5000	-x-
3	2,5000	2.3728	1,3E−1
4	2,3728	2,3574	1,5E−2
5	2,3574	2,3562	1,2E−3
6	2,3562	2,3562	1,1E−5
7	2,3562	2,3562	7,0E−9

[10] Razão fundamental do Cálculo.

2.6.1 Interpretação geométrica

A iteração do método da secante é

$$x^{(k+1)} = x^{(k)} - f(x^{(k)})\frac{x^{(k)} - x^{(k-1)}}{f(x^{(k)}) - f(x^{(k-1)})}, \tag{2.103}$$

donde segue que

$$0 = x^{(k+1)} - x^{(k)} + f(x^{(k)})\frac{x^{(k)} - x^{(k-1)}}{f(x^{(k)}) - f(x^{(k-1)})}, \tag{2.104}$$

bem como que

$$0 = \frac{f(x^{(k)}) - f(x^{(k-1)})}{x^{(k)} - x^{(k-1)}}(x^{(k+1)} - x^{(k)}) + f(x^{(k)}). \tag{2.105}$$

Agora, observemos que

$$y = \frac{f(x^{(k)}) - f(x^{(k-1)})}{x - x^{(k-1)}}(x^{(k+1)} - x^{(k)}) + f(x^{(k)}). \tag{2.106}$$

é a equação da reta secante ao gráfico de f pelos pontos $(x^{(k)}, f(x^{(k)}))$ e $(x^{(k-1)}, f(x^{(k-1)}))$.

Do observado acima, temos que (2.105) nos mostra que a iterada $x^{(k+1)}$ é a a interseção do eixo x com a reta secante ao gráfico de f pelos pontos $(x^{(k)}, f(x^{(k)}))$ e $(x^{(k-1)}, f(x^{(k-1)}))$. Veja a Figura 2.10.

Observação 2.6.1. A interpretação geométrica do método da secante pode nos ajudar a escolher as aproximações iniciais $x^{(1)}$ e $x^{(2)}$. Como uma boa prática, escolhemo-las próximas do zero (por inspeção gráfica), tomando $x^{(2)}$ como uma aproximação melhor que $x^{(1)}$.

2.6.2 Análise de convergência

Pode-se mostrar[11] que a taxa de convergência do método da secante é super-linear com

$$|x^{(k+1)} - x^*| \leq C|x^{(k)} - x^*|^{\varphi}, \tag{2.107}$$

onde $\varphi = (1 + \sqrt{5})/2 \approx 1{,}618$ (razão áurea) e x^* é o zero de f.

[11] Veja, por exemplo, Seção 3.5.2 do livro “Cálculo Numérico” do projeto REAMAT.

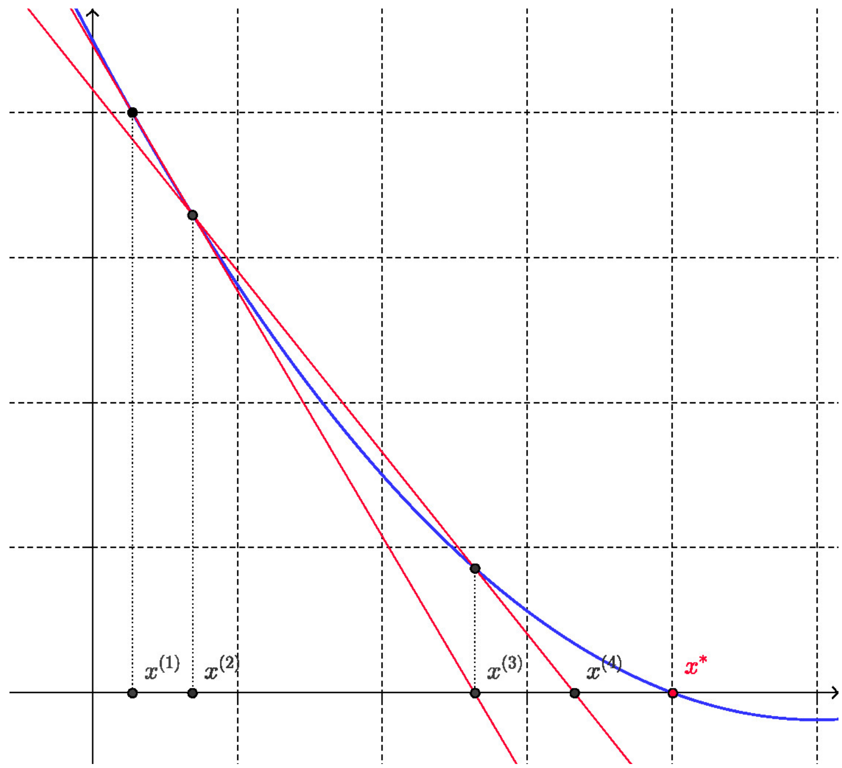

Figura 2.10: Interpretação geométrica das iterações do método da secante. Veja no Geogebra.

Exemplo 2.6.2. Consideremos o problema de encontrar o zero da função

$$f(x) = \operatorname{sen}^2\left(x + \frac{\pi}{4}\right) - x^3 + \frac{\pi}{4}x^2 + \frac{5\pi^2}{16}x + \frac{3\pi^3}{64}. \tag{2.108}$$

no intervalo [2,3]. Este problema foi construído de forma que $x^* = 3\pi/4$ é um zero de f. Agora, fazendo as iterações do método da secante com aproximações iniciais $x^{(1)} = 2{,}6$ e $x^{(2)} = 2{,}5$, obtemos os resultados apresentados na Tabela 2.12.

Observação 2.6.2. (Zeros de multiplicidade par.) A taxa de convergência super-linear do método da secante não se mantém para o caso de x^* ser um zero múltiplo. Para contornar este problema, pode-se aplicar o método à derivada $n-1$ de f, a fim de se aproximar um zero de multiplicidade n.

Observação 2.6.3. (Cancelamento catastrófico.) Conforme convergem as

Tabela 2.12: Resultados referentes ao Exemplo 2.6.2.

k	$x^{(k)}$	$\|x^{(k)} - x^*\|$
3	2,3728	1,7E−02
4	2,3574	1,2E−03
5	2,3562	1,1E−05
6	2,3562	7,0E−09

iterações do método da secante, o denominador $f(x^{(k)}) - f(x^{(k-1)})$ pode convergir rapidamente para zero, ocasionando uma divisão por zero.

Exercícios

Exercício 2.6.1. Use o método da secante para obter uma aproximação do zero de $f(x) = x^3 \operatorname{sen}(x) - \cos(x)$ no intervalo $[0{,}5, 1]$ com precisão de 10^{-5}.

Exercício 2.6.2. Use o método da secante para obter uma aproximação do zero de

$$f(x) = (-x^2 + 1{,}154x - 0{,}332929)\cos(x) + x^2 - 1{,}154x + 0{,}332929 \quad (2.109)$$

no intervalo $(0{,}55,\ 0{,}65)$ com precisão de 10^-5.

2.7 Raízes de polinômios

Nesta seção, veremos como o método de Newton pode ser aplicado de forma robusta a polinômios com o auxílio do método de Horner[12]. A questão central é que dado um polinômio de grau n escrito na forma

$$p(x) = a_1x^n + a_2x^{n-1} + \cdots + a_nx + a_{n+1}, \quad (2.110)$$

o procedimento de avaliá-lo em um ponto requer $n!n$ multiplicações e n adições. Desta forma, a aplicação do método de Newton para obter as raízes de p torna-se computacionalmente custosa, por requer a avaliação de p e de sua derivada a cada iteração. Agora, o método de Horner nos permite avaliar p em um ponto qualquer usando apenas n multiplicações e n adições, reduzindo enormemente o custo computacional.

[12]William George Horner, matemático britânico, 1786 - 1837. Fonte: Wikipedia

2.7.1 Método de Horner

Sejam dados um número x_0 e um polinômio de grau n

$$p(x) = p_1x^n + p_2x^{n-1} + \cdots + p_nx + p_{n+1}s. \quad (2.111)$$

O método de Horner consiste em computar $p(x_0)$ pela iteração

$$q_1 = p_1, \quad (2.112)$$
$$q_k = p_k + q_{k-1}x_0, \quad k = 2,3,\ldots,n+1, \quad (2.113)$$

sendo que, então, $p(x_0) = q_{n+1}$ e, além disso,

$$p(x) = (x - x_0)q(x) + q_{n+1}, \quad (2.114)$$

com

$$q(x) = q_1x^{n-1} + q_2x^{n-2} + \cdots + q_{n-1}x + q_n. \quad (2.115)$$

De fato, a verificação de (2.114) é direta, uma vez que

$$\begin{aligned}(x - x_0)q(x) + q_{n+1} &= (x - x_0)(q_1x^{n-1} + q_2x^{n-2} + \cdots + q_{n-1}x + q_n)\\ &\quad + q_{n+1} &(2.116)\\ &= q_1x^n + (q_2 - q_1x_0)^{n-1} + \cdots + (q_{n+1} - q_nx_0). &(2.117)\end{aligned}$$

E, então, igualando a $p(x)$ na forma (2.111), temos as equações (2.112)-(2.113).

Exemplo 2.7.1. Consideremos o polinômio

$$p(x) = x^3 - 3x^2 + 4. \quad (2.118)$$

Para computarmos $p(1)$ pelo método de Horner, tomamos $x_0 = 1$, $q_1 = p_1 = 1$ e

$$q_2 = p_2 + q_1x_0 = -3 + 1 \cdot 1 = -2 \quad (2.119)$$
$$q_3 = p_3 + q_2x_0 = 0 + (-2) \cdot 1 = -2 \quad (2.120)$$
$$q_4 = p_4 + q_3x_0 = 4 + (-2) \cdot 1 = 2. \quad (2.121)$$

Com isso, temos $p(3) = q_4 = 4$ (verifique!).

Observação 2.7.1. Ao computarmos $p(x_0)$ pelo método de Horner, obtemos a decomposição

$$p(x) = (x - x_0)q(x) + b_{n+1}. \tag{2.122}$$

Desta forma, temos

$$p'(x) = q(x) + (x - x_0)q'(x), \tag{2.123}$$

donde temos que $p'(x_0) = q(x_0)$. Com isso, para computarmos $p'(x_0)$ podemos aplicar o método de Horner a $q(x)$.

2.7.2 Método de Newton-Horner

A implementação do método de Newton a polinômios pode ser feita de forma robusta com o auxílio do método de Horner. Dado um polinômio p e uma aproximação inicial $x^{(1)}$ para uma de suas raízes reais, a iteração de Newton consiste em

$$x^{(k+1)} = x^{(k)} - \frac{p(x^{(k)})}{p'(x^{(k)})}, \tag{2.124}$$

na qual podemos utilizar o método de Horner para computar $p(x^{(k)})$ e $p'(x^{(k)})$.

Exemplo 2.7.2. Consideremos o caso de aplicar o método de Newton para obter uma aproximação da raiz $x = -1$ de

$$p(x) = x^3 - 3x^2 + 4, \tag{2.125}$$

com aproximação inicial $x^{(1)} = -2$. Na Tabela 2.13 temos os resultados obtidos.

Tabela 2.13: Resultados referentes ao Exemplo 2.7.2.

k	$x^{(k)}$	$\vert x^{(k)} - x^{(k-1)}\vert$
1	$-2{,}0000$	-x-
2	$-1{,}3333$	6,7E$-$1
3	$-1{,}0556$	2,8E$-$1
4	$-1{,}0019$	5,4E$-$2
5	$-1{,}0000$	1,9E$-$3

Exercícios

Exercício 2.7.1. Use o método de Newton-Horner para computar a aproximação da raiz de $p(x) = x^3 - 3x^2 + 4$ no intervalo [1,3]. Observe que p tem um zero duplo deste intervalo.

Capítulo 3

Métodos diretos para sistemas lineares

Neste capítulo, discutiremos sobre métodos diretos para a resolução de sistemas lineares de n-equações com n-incógnitas. Isto é, sistemas que podem ser escritos na seguinte forma algébrica

$$a_{11}x_1 + a_{12}x_2 + \cdots + a_{1n}x_n = b_1 \tag{3.1}$$
$$a_{21}x_1 + a_{22}x_2 + \cdots + a_{2n}x_n = b_2 \tag{3.2}$$
$$\vdots \tag{3.3}$$
$$a_{n1}x_1 + a_{n2}x_2 + \cdots + a_{nn}x_n = b_n. \tag{3.4}$$

3.1 Eliminação gaussiana

Um sistema linear

$$a_{11}x_1 + a_{12}x_2 + \cdots + a_{1n}x_n = b_1 \tag{3.5}$$
$$a_{21}x_1 + a_{22}x_2 + \cdots + a_{2n}x_n = b_2 \tag{3.6}$$
$$\vdots \tag{3.7}$$
$$a_{n1}x_1 + a_{n2}x_2 + \cdots + a_{nn}x_n = b_n. \tag{3.8}$$

pode ser escrito na forma matricial

$$A\boldsymbol{x} = \boldsymbol{b}, \tag{3.9}$$

onde $A = [a_{ij}]_{i,j=1}^{n,n}$ é chamada de matriz dos coeficientes, $\boldsymbol{x} = (x_1, x_2, \ldots, x_n)$ é o vetor (coluna) das incógnitas e $\boldsymbol{b} = (b_1, b_2, \ldots, b_n)$ é o vetor (coluna) dos termos constantes.

Outra forma matricial de representar o sistema (3.5)-(3.8) é pela chamada matriz estendida

$$E = [A\ \boldsymbol{b}]. \tag{3.10}$$

No caso, E é a seguinte matriz $n \times (n+1)$

$$E = \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} & b_1 \\ a_{21} & a_{22} & \cdots & a_{2n} & b_2 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ a_{n1} & a_{n2} & \cdots & a_{nn} & b_n \end{bmatrix} \tag{3.11}$$

O método de eliminação gaussiana consistem em realizar operações sobre as equações (sobre as linhas) do sistema (3.5)-(3.8) (da matriz estendida E) de forma a reescrevê-lo como um sistema triangular, ou diagonal. Para tanto, podemos utilizar as seguintes operações:

1. permutação entre as equações (linhas) ($E_i \leftrightarrow E_j$).
2. multiplicação de uma equação (linha) por um escalar não nulo ($E_i \leftarrow \lambda E_i$).
3. substituição de uma equação (linha) por ela somada com a multiplicação de uma outra por um escalar não nulo ($E_i \leftarrow E_i + \lambda E_j$).

Exemplo 3.1.1. O sistema linear

$$-2x_1 - 3x_2 + 2x_3 + 3x_4 = 10 \tag{3.12}$$
$$-4x_1 - 6x_2 + 6x_3 + 6x_4 = 20 \tag{3.13}$$
$$-2x_1 + 4x_3 + 6x_4 = 10 \tag{3.14}$$
$$4x_1 + 3x_2 - 8x_3 - 6x_4 = -17 \tag{3.15}$$

pode ser escrito na forma matricial $A\boldsymbol{x} = \boldsymbol{b}$, onde

$$A = \begin{bmatrix} -2 & -3 & 2 & 3 \\ -4 & -6 & 6 & 6 \\ -2 & 0 & 4 & 6 \\ 4 & 3 & -8 & -6 \end{bmatrix}, \tag{3.16}$$

$\boldsymbol{x} = (x_1, x_2, x_3, x_4)$ e $\boldsymbol{b} = (10, 20, 10, -17)$. Sua matriz estendida é

$$E = \begin{bmatrix} -2 & -3 & 2 & 3 & 10 \\ -4 & -6 & 6 & 6 & 20 \\ -2 & 0 & 4 & 6 & 10 \\ 4 & 3 & -8 & -6 & -17 \end{bmatrix} \tag{3.17}$$

Então, usando o método de eliminação gaussiana, temos

$$E = \begin{bmatrix} -2 & -3 & 2 & 3 & 10 \\ -4 & -6 & 6 & 6 & 20 \\ -2 & 0 & 4 & 6 & 10 \\ 4 & 3 & -8 & -6 & -17 \end{bmatrix} E_2 \leftarrow E_2 - (e_{21}/\boldsymbol{e_{11}})E_1 \tag{3.18}$$

$$\sim \begin{bmatrix} \mathbf{-2} & -3 & 2 & 3 & 10 \\ 0 & 0 & 2 & 0 & 0 \\ -2 & 0 & 4 & 6 & 10 \\ 4 & 3 & -8 & -6 & -17 \end{bmatrix} E_3 \leftarrow E_3 - (e_{31}/\boldsymbol{e_{11}})E_1 \tag{3.19}$$

$$\sim \begin{bmatrix} \mathbf{-2} & -3 & 2 & 3 & 10 \\ 0 & 0 & 2 & 0 & 0 \\ 0 & 3 & 2 & 3 & 0 \\ 4 & 3 & -8 & -6 & -17 \end{bmatrix} E_4 \leftarrow E_4 - (e_{41}/\boldsymbol{e_{11}})E_1 \tag{3.20}$$

$$\sim \begin{bmatrix} \mathbf{-2} & -3 & 2 & 3 & 10 \\ 0 & 0 & 2 & 0 & 0 \\ 0 & \mathbf{3} & 2 & 3 & 0 \\ 0 & -3 & -4 & 0 & 3 \end{bmatrix} E_2 \leftrightarrow E_3 \tag{3.21}$$

$$\sim \begin{bmatrix} \mathbf{-2} & -3 & 2 & 3 & 10 \\ 0 & \mathbf{3} & 2 & 3 & 0 \\ 0 & 0 & 2 & 0 & 0 \\ 0 & -3 & -4 & 0 & 3 \end{bmatrix} E_4 \leftarrow E_4 - (e_{42}/\boldsymbol{e_{22}})E_2 \tag{3.22}$$

$$\sim \begin{bmatrix} \mathbf{-2} & -3 & 2 & 3 & 10 \\ 0 & 3 & 2 & 3 & 0 \\ 0 & 0 & \mathbf{2} & 0 & 0 \\ 0 & 0 & -2 & 3 & 3 \end{bmatrix} E_4 \leftarrow E_4 - (e_{43}/\boldsymbol{e_{33}})E_3 \tag{3.23}$$

$$\sim \begin{bmatrix} \mathbf{-2} & -3 & 2 & 3 & 10 \\ 0 & \mathbf{3} & 2 & 3 & 0 \\ 0 & 0 & \mathbf{2} & 0 & 0 \\ 0 & 0 & 0 & \mathbf{3} & 3 \end{bmatrix} \tag{3.24}$$

Esta última matriz estendida é chamada de **matriz escalonada** do sistema. Desta, temos que (3.12)-(3.15) é equivalente ao seguinte sistema triangular

$$\begin{aligned} -2x_1 - 3x_2 + 2x_3 + 3x_4 &= 10 && (3.25)\\ 3x_2 + 2x_3 + 3x_4 &= 0 && (3.26)\\ 2x_3 &= 0 && (3.27)\\ 3x_4 &= 3. && (3.28)\end{aligned}$$

Resolvendo da última equação para a primeira, temos

$$\begin{aligned} x_4 &= 1, && (3.29)\\ x_3 &= 0, && (3.30)\\ x_2 &= \frac{-2x_3 - 3x_4}{3} = -1, && (3.31)\\ x_1 &= \frac{10 + 3x_2 - 3x_3 - 3x_4}{-2} = -2. && (3.32)\end{aligned}$$

Observação 3.1.1. Para a resolução de um sistema linear $n \times n$, o método de eliminação gaussiana demanda

$$\frac{n^3}{3} + n^2 - \frac{n}{3} \tag{3.33}$$

multiplicações/divisões e

$$\frac{n^3}{3} + \frac{n^2}{2} - \frac{5n}{6} \tag{3.34}$$

adições/subtrações [2].

Com o mesmo custo computacional, podemos utilizar o método de eliminação gaussiana para transformar o sistema dado em um sistema diagonal.

Exemplo 3.1.2. Voltando ao exemplo anterior (Exemplo 3.1.1, vimos que a matriz estendida do sistema (3.12)-(3.15) é equivalente a

$$E \sim \begin{bmatrix} -2 & -3 & 2 & 3 & 10 \\ 0 & 3 & 2 & 3 & 0 \\ 0 & 0 & 2 & 0 & 0 \\ 0 & 0 & 0 & 3 & 3 \end{bmatrix}. \tag{3.35}$$

Então, podemos continuar aplicando o método de eliminação gaussiana, agora de baixo para cima, até obtermos um sistema diagonal equivalente. Vejamos

$$E \sim \begin{bmatrix} -2 & -3 & 2 & 3 & 10 \\ 0 & 3 & 2 & 3 & 0 \\ 0 & 0 & 2 & 0 & 0 \\ 0 & 0 & 0 & \mathbf{3} & 3 \end{bmatrix} \begin{matrix} E_1 \leftarrow E_1 - (e_{14}/e_{44})E_4 \\ E_2 \leftarrow E_2 - (e_{24}/e_{44})E_4 \\ \\ \\ \end{matrix} \tag{3.36}$$

$$\sim \begin{bmatrix} -2 & -3 & 2 & 0 & 7 \\ 0 & 3 & 2 & 0 & -3 \\ 0 & 0 & \mathbf{2} & 0 & 0 \\ 0 & 0 & 0 & 3 & 3 \end{bmatrix} \begin{matrix} E_1 \leftarrow E_1 - (e_{13}/e_{33})E_3 \\ E_2 \leftarrow E_2 - (e_{23}/e_{33})E_3 \\ \\ \\ \end{matrix} \tag{3.37}$$

$$\sim \begin{bmatrix} -2 & -3 & 0 & 0 & 4 \\ 0 & \mathbf{3} & 0 & 0 & -3 \\ 0 & 0 & 2 & 0 & 0 \\ 0 & 0 & 0 & 3 & 3 \end{bmatrix} \begin{matrix} E_1 \leftarrow E_1 - (e_{12}/e_{22})E_2 \\ \\ \\ \\ \end{matrix} \tag{3.38}$$

$$\sim \begin{bmatrix} \mathbf{-2} & 0 & 0 & 0 & 4 \\ 0 & \mathbf{3} & 0 & 0 & -3 \\ 0 & 0 & \mathbf{2} & 0 & 0 \\ 0 & 0 & 0 & \mathbf{3} & 3 \end{bmatrix} \begin{matrix} E_1 \leftarrow E_1/e_{11} \\ E_2 \leftarrow E_2/e_{22} \\ E_3 \leftarrow E_3/e_{33} \\ E_4 \leftarrow E_4/e_{44} \end{matrix} \tag{3.39}$$

$$\sim \begin{bmatrix} 1 & 0 & 0 & 0 & -2 \\ 0 & 1 & 0 & 0 & -1 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 \end{bmatrix} \tag{3.40}$$

Esta última matriz é chamada de matriz escalonada reduzida (por linhas) e a solução do sistema encontra-se em sua última coluna, i.e. $\boldsymbol{x} = (-2, -1, 0, 1)$.

Exercícios

Exercício 3.1.1. Use o método de eliminação gaussiana para obter a matriz escalonada reduzida do seguinte sistema

$$-3x_1 + 2x_2 - 5x_4 + x_5 = -23 \tag{3.41}$$

$$-x_2 - 3x_3 = 9 \tag{3.42}$$

$$-2x_1 - x_2 + x_3 = -1 \tag{3.43}$$

$$2x_2 - 4x_3 + 3x_4 = 8 \tag{3.44}$$

$$x_1 - 3x_3 - x_5 = 11 \tag{3.45}$$

Exercício 3.1.2. Use o método de eliminação gaussiana para obter a matriz escalonada reduzida do seguinte sistema

$$\begin{aligned}
-10^{-12}x_1 + 20x_2 - 3x_3 &= -1 &(3.46)\\
2{,}001x_1 + 10^{-5}x_2 - x_3 &= -2 &(3.47)\\
4x_1 - 2x_2 + x_3 &= 0{,}1 &(3.48)
\end{aligned}$$

3.2 Norma e número de condicionamento

Nesta seção, fazemos uma rápida discussão sobre normas de vetores e matrizes, bem como do número de condicionamento de uma matriz.

3.2.1 Norma L^2

A norma L^2 de um dado vetor $v = (v_1, v_2, \dots, v_n) \in \mathbb{R}^{\ltimes}$ é definida por

$$\|v\| := \sqrt{v_1^2 + v_2^2 + \cdots + v_n^2}. \tag{3.49}$$

Proposição 3.2.1. *Dados os vetores $u,v \in \mathbb{R}^{\ltimes}$ e um escalar $\lambda \in \mathbb{R}$, temos:*

a) $\|v\| \geq 0$.

b) $\|v\| = 0 \Leftrightarrow v = 0$.

c) $\|\lambda v\| = |\lambda| \cdot \|v\|$.

d) $\|u + v\| \leq \|u\| + \|v\|$ *(desigualdade triangular).*

e) $u \cdot v \leq \|u\| \cdot \|v\|$ *(desigualdade de Cauchy-Schwarz).*

Exemplo 3.2.1. A norma L^2 do vetor $v = (1, -2, 3, -4)$ é

$$\begin{aligned}
\|v\| &= \sqrt{v_1^2 + v_2^2 + v_3^2 + v_4^2} &(3.50)\\
&= \sqrt{1^2 + (-2)^2 + 3^2 + (-4)^2} &(3.51)\\
&= 5{,}4772. &(3.52)
\end{aligned}$$

A norma L^2 induzida de uma dada matriz real $A = [a_{ij}]_{i,j=1}^{n}$ é definida por

$$\|A\| := \sup_{x\in\mathbb{R}^n, \|x\|=1} \|Ax\|. \tag{3.53}$$

Pode-se mostrar que

$$\|A\| = \sqrt{\lambda_{max}(A^TA)}, \tag{3.54}$$

onde $\lambda_{max}(A^TA) := \max\{|\lambda|;\ \lambda \text{ é autovalor de } A^TA\}$.

Proposição 3.2.2. *Dadas as matrizes reais A, B $n \times n$, um vetor $v \in \mathbb{R}^2$ e um escalar λ, temos*

a) $\|A\| \geq 0$.

b) $\|A\| = 0 \Leftrightarrow A = 0$.

c) $\|\lambda A\| = |\lambda| \cdot \|A\|$.

d) $\|A + B\| \leq \|A\| + \|B\|$ *(desigualdade triangular).*

e) $\|AB\| \leq \|A\|\|B\|$.

f) $\|Av\| \leq \|A\|\|v\|$.

Exemplo 3.2.2. A matriz

$$A = \begin{bmatrix} 1 & -1 & 2 \\ -2 & \pi & 4 \\ 7 & -5 & \sqrt{2} \end{bmatrix} \tag{3.55}$$

tem norma L^2

$$\|A\| = 9{,}3909. \tag{3.56}$$

3.2.2 Número de condicionamento

O número de condicionamento de uma matriz é uma medida referente a propagação de erros de ocorre da sua aplicação. Mais especificamente, assumamos que seja dada uma matriz invertível $A = [a_{ij}]_{i,j=1}^{n,n}$, um vetor $x \in \mathbb{R}^n$ e uma perturbação $\delta_x \in \mathbb{R}^n$. Além disso, sejam

$$y = Ax \tag{3.57}$$

$$y + \delta_y = A(x + \delta_x). \tag{3.58}$$

Ou seja, δ_y é a perturbação em y propagada da aplicação de A em x com perturbação δ_x.

Agora, podemos estimar a razão entre os erros relativos $e_{rel}(y) := \|\delta_y\|/\|y\|$ e $e_{rel}(x) := \|\delta_x\|/\|x\|$ da seguinte forma

$$\frac{\frac{\|y\|}{\|\delta_y\|}}{\frac{\|x\|}{\|\delta_x\|}} = \frac{\|y\|}{\|\delta_y\|}\frac{\|\delta_x\|}{\|x\|} \tag{3.59}$$

$$= \frac{\|Ax\|}{\|\delta_y\|}\frac{\|A^{-1}\delta_y\|}{\|x\|} \tag{3.60}$$

$$\leq \frac{\|A\|\|x\|\|A^{-1}\|\|\delta_y\|}{\|\delta_y\|\|x\|} \tag{3.61}$$

$$\leq \|A\|\|A^{-1}\|. \tag{3.62}$$

Logo, temos a seguinte estimativa de propagação de erro

$$e_{rel}(y) \leq \|A\|\|A^{-1}\|e_{rel}(x). \tag{3.63}$$

Isto nos motiva a definir o **número de condicionamento** da matriz A por

$$\kappa(A) := \|A\|\|A^{-1}\|. \tag{3.64}$$

Observação 3.2.1. A matriz identidade tem o menor número de condicionamento, o qual é

$$\kappa(I) = 1. \tag{3.65}$$

Exemplo 3.2.3. Um exemplo de uma matriz bem condicionada é

$$A = \begin{bmatrix} 1 & -1 & 2 \\ -2 & \pi & 4 \\ 7 & -5 & \sqrt{2} \end{bmatrix}, \tag{3.66}$$

cujo número de condicionamento é $\kappa(A) = 13{,}997$.

Já, a matriz

$$B = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & -2 \\ 10^5 & 10^{-4} & 10^5 \end{bmatrix}, \tag{3.67}$$

tem número de condicionamento

$$\kappa(B) = 1{,}5811 \times 10^{14}, \tag{3.68}$$

o que indica que B é uma matriz mal condicionada.

Exercícios

Exercício 3.2.1. Considere o seguinte sistema linear

$$10^{-12}x_1 + 20x_2 + 3x_3 = -1, \tag{3.69}$$
$$2{,}001x_1 + 10^{-5}x_2 + -x_3 = -2, \tag{3.70}$$
$$x_1 - 2x_2 - 0{,}1x_3 = 0{,}1. \tag{3.71}$$

a) Compute a norma L^2 do vetor dos termos constantes deste sistema.

b) Compute a norma L^2 da matriz dos coeficientes deste sistema.

c) Compute o número de condicionamento da matriz dos coeficientes deste sistema.

3.3 Método de eliminação gaussiana com pivotamento parcial com escala

O método de eliminação gaussiana é suscetível a propagação dos erros de arredondamento, em particular, quando os pivôs são números próximos de zero. Isto pode ser mitigado com o chamado pivotamento parcial com escala. Nesta variação do método de eliminação gaussiana, o pivô é escolhido como sendo o candidato que é o maior em relação aos elementos em sua linha.

Dado um sistema $Ax = b$ com n-equações e n-incógnitas, o método pode ser descrito pelo seguinte pseudo-código:

1. $E \leftarrow [A\ b]$.
2. Para $i = 1, 2, \ldots, n$, faça $s_i \leftarrow \max_{1\leq j\leq n} |e_{i,j}|$.
3. Para $i = 1, 2, \ldots, n-1$:

3.1 Compute j tal que

$$\frac{e_{j,i}}{s_j} \geq \frac{e_{k,i}}{s_k}, \quad \forall k = i, i+1, \dots, n. \tag{3.72}$$

3.2 Permute as linhas i e j, i.e. $E_i \leftrightarrow E_j$.

3.3 Para $j = i+1, i+2, \dots, n$:

3.3.1 $E_j \leftarrow E_j - \frac{e_{ji}}{e_{ii}} E_i$.

4. Para $i = n, n-1, \dots, 2$:

4.1 Para $j = i-1, i-2, \dots, 1$:

4.1.1 $E_j \leftarrow E_j - \frac{e_{ji}}{e_{ii}} E_i$.

Exemplo 3.3.1. Vamos empregar o método de eliminação gaussiana com pivotamento parcial com escala para resolvermos o seguinte sistema linear

$$10^{-12}x_1 + 20x_2 + 3x_3 = -1, \tag{3.73}$$
$$2{,}001x_1 + 10^{-5}x_2 - x_3 = -2, \tag{3.74}$$
$$x_1 - 2x_2 - 0{,}1x_3 = 0{,}1. \tag{3.75}$$

Para tanto, tomamos a matriz estendida

$$E = \begin{bmatrix} 10^{-12} & 20 & 3 & -1 \\ 2{,}001 & 10^{-5} & -1 & -2 \\ 1 & -2 & -0{,}1 & 0{,}1 \end{bmatrix} \tag{3.76}$$

e computamos os valores máximos em módulo dos elementos de cada linha da matriz A, i.e.

$$s = (20,\ 2{,}001,\ 2). \tag{3.77}$$

Agora, observamos que $e_{2,1}$ é o maior pivô em escala, pois

$$\frac{e_{11}}{s_1} = 5 \times 10^{-14}, \frac{e_{21}}{s_2} = 1, \frac{e_{31}}{s_3} = 0{,}5. \tag{3.78}$$

Então, fazemos a permutação entre as linhas 1 e 2, de forma a obtermos

$$E = \begin{bmatrix} 2{,}001 & 10^{-5} & -1 & -2 \\ 10^{-12} & 20 & 3 & -1 \\ 1 & -2 & -0{,}1 & 0{,}1 \end{bmatrix} \tag{3.79}$$

Em seguida, eliminamos abaixo do pivô, e temos

$$E = \begin{bmatrix} 2{,}001 & 10^{-5} & -1 & -2 \\ 0 & 20 & 3 & -1 \\ 0 & -2 & 3{,}9975 \times 10^{-1} & 1{,}0995 \end{bmatrix} \tag{3.80}$$

Daí, o novo pivô é escolhido como e_{22}, pois ambos candidatos tem o mesmo valor em escala

$$\frac{e_{2,2}}{s_2} = 1, \frac{e_{3,2}}{s_3} = 1. \tag{3.81}$$

Logo, eliminamos abaixo do pivô para obtermos

$$E = \begin{bmatrix} 2{,}001 & 10^{-5} & -1 & -2 \\ 0 & 20 & 3 & -1 \\ 0 & 0 & 6{,}9975 \times 10^{-1} & 9{,}9950 \end{bmatrix} \tag{3.82}$$

Procedendo a eliminação para cima, obtemos a matriz escalonada reduzida

$$E = \begin{bmatrix} 1 & 0 & 0 & -2.8567\text{E}-1 \\ 0 & 1 & 0 & -2.6425\text{E}-1 \\ 0 & 0 & 1 & 1{,}4284\text{E}+0 \end{bmatrix} \tag{3.83}$$

Exercícios

Exercício 3.3.1. Use o método de eliminação gaussiana com pivotamento parcial com escala para obter a matriz escalonada reduzida do seguinte sistema

$$-2 \times 10^{-12} x_1 + 10 x_2 - 3 \times 10^{-4} x_3 = 2 \tag{3.84}$$

$$10^5 x_1 + 10^{-13} x_2 - x_3 = -2 \tag{3.85}$$

$$x_1 - 2x_2 + 3 \times 10^{-7} x_3 = 4 \tag{3.86}$$

3.4 Fatoração LU

A fatoração LU é uma forma eficiente de se resolver sistemas lineares. Dado um sistema $Ax = b$, a ideia é de fatorar a matriz A como o produto de uma matriz triangular inferior[1] L com uma matriz triangular superior[2] U, i.e.

$$A = LU. \tag{3.87}$$

Com isso, o sistema $Ax = b$ pode ser reescrito na forma

$$(LU)x = b \Leftrightarrow L(Ux) = b. \tag{3.88}$$

Denotando, $Ux = y$, podemos resolver o seguinte sistema triangular

$$Ly = b. \tag{3.89}$$

Tendo resolvido este sistema, a solução do sistema $Ax = b$ pode, então, ser computada como a solução do seguinte sistema triangular

$$Ux = y. \tag{3.90}$$

Ou seja, a decomposição LU nos permite resolver uma sistema pela resolução de dois sistemas triangulares.

O procedimento de decomposição LU é equivalente ao método de eliminação gaussiana. Consideremos uma matriz $A = [a_{ij}]_{i,j=1}^{n,n}$, com $a_{1,1} \neq 0$. Denotando esta matriz por $U^{(1)} = [u_{i,j}^{(1)}]_{i,j=1}^{n,n} = A$ e tomando $L^{(1)} = I_{n\times n}$, observamos que a eliminação abaixo do pivô $u_{1,1}^{(1)}$, pode ser computada com as seguintes operações equivalentes por linha

$$U^{(1)} = \begin{bmatrix} u_{1,1}^{(1)} & u_{1,2}^{(1)} & u_{1,3}^{(1)} & \dots & u_{1,n}^{(1)} \\ u_{2,1}^{(1)} & u_{2,2}^{(1)} & u_{2,3}^{(1)} & \dots & u_{2,n}^{(1)} \\ u_{3,1}^{(1)} & u_{3,2}^{(1)} & u_{3,3}^{(1)} & \dots & u_{3,n}^{(1)} \\ \vdots & \vdots & \vdots & \dots & \vdots \\ u_{n,1}^{(1)} & u_{n,2}^{(1)} & a_{n,3}^{(1)} & \dots & u_{n,n}^{(1)} \end{bmatrix} \begin{array}{l} U_1^{(2)} \leftarrow U_1^{(1)} \\ U_2^{(2)} \leftarrow U_2^{(1)} - m_{2,1}U_1^{(1)} \\ U_3^{(2)} \leftarrow U_3^{(1)} - m_{3,1}U_1^{(1)} \\ \vdots \\ U_n^{(2)} \leftarrow U_n^{(1)} - m_{n,1}U_1^{(1)} \end{array} \tag{3.91}$$

onde, $m_{i,1} = u_{i,1}^{(1)}/u_{1,1}^{(1)}$, $i = 2, 3, \dots, n$.

Destas computações, obtemos uma nova matriz da forma

$$U^{(2)} = \begin{bmatrix} u_{1,1}^{(2)} & u_{1,2}^{(2)} & u_{1,3}^{(2)} & \dots & u_{1,n}^{(2)} \\ 0 & u_{2,2}^{(2)} & u_{2,3}^{(2)} & \dots & u_{2,n}^{(2)} \\ 0 & u_{3,2}^{(2)} & u_{3,3}^{(2)} & \dots & u_{3,n}^{(2)} \\ \vdots & \vdots & \vdots & \dots & \vdots \\ 0 & u_{n,2}^{(2)} & u_{n,3}^{(2)} & \dots & u_{n,n}^{(2)} \end{bmatrix} \tag{3.92}$$

[1]Do inglês, *low triangular matrix*.

[2]Do inglês, *upper triangular matrix*.

Observemos, também, que denotando

$$L^{(2)} = \begin{bmatrix} 1 & 0 & 0 & \dots & 0 \\ m_{2,1} & 1 & 0 & \dots & 0 \\ m_{3,1} & 0 & 1 & \dots & 0 \\ \vdots & \vdots & \vdots & \dots & \vdots \\ m_{n,1} & 0 & 0 & \dots & 1 \end{bmatrix} \tag{3.93}$$

temos

$$A = L^{(2)}U^{(2)}. \tag{3.94}$$

No caso de $u_{2,2}^{(2)} \neq 0$, podemos continuar com o procedimento de eliminação gaussiana com as seguintes operações equivalentes por linha

$$U^{(2)} = \begin{bmatrix} u_{1,1}^{(2)} & u_{1,2}^{(2)} & u_{1,3}^{(2)} & \dots & u_{1,n}^{(2)} \\ 0 & u_{2,2}^{(2)} & u_{2,3}^{(2)} & \dots & u_{2,n}^{(2)} \\ 0 & u_{3,2}^{(2)} & u_{3,3}^{(2)} & \dots & u_{3,n}^{(2)} \\ \vdots & \vdots & \vdots & \dots & \vdots \\ 0 & u_{n,2}^{(2)} & u_{n,3}^{(2)} & \dots & u_{n,n}^{(2)} \end{bmatrix} \begin{array}{l} U_1^{(3)} \leftarrow U_1^{(2)} \\ U_2^{(3)} \leftarrow U_2^{(2)} \\ U_3^{(3)} \leftarrow U_3^{(2)} - m_{3,2}U_2^{(2)} \\ \vdots \\ U_n^{(3)} \leftarrow U_n^{(2)} - m_{n,2}U_2^{(2)} \end{array} \tag{3.95}$$

onde, $m_{i,2} = u_{i,2}^{(2)}/u_{2,2}^{(2)}$, $i = 3, 4, \dots, n$. Isto nos fornece o que nos fornece

$$U^{(3)} = \begin{bmatrix} u_{1,1}^{(3)} & u_{1,2}^{(3)} & u_{1,3}^{(3)} & \dots & u_{1,n}^{(3)} \\ 0 & u_{2,2}^{(3)} & u_{2,3}^{(3)} & \dots & u_{2,n}^{(3)} \\ 0 & 0 & u_{3,3}^{(3)} & \dots & u_{3,n}^{(3)} \\ \vdots & \vdots & \vdots & \dots & \vdots \\ 0 & 0 & u_{n,3}^{(3)} & \dots & u_{n,n}^{(3)} \end{bmatrix}. \tag{3.96}$$

Além disso, denotando

$$L^{(3)} = \begin{bmatrix} 1 & 0 & 0 & \dots & 0 \\ m_{2,1} & 1 & 0 & \dots & 0 \\ m_{3,1} & m_{3,2} & 1 & \dots & 0 \\ \vdots & \vdots & \vdots & \dots & \vdots \\ m_{n,1} & m_{n,2} & 0 & \dots & 1 \end{bmatrix} \tag{3.97}$$

temos

$$A = L^{(3)}U^{(3)}. \tag{3.98}$$

Continuando com este procedimento, ao final de $n-1$ passos teremos obtido a decomposição

$$A = LU, \tag{3.99}$$

onde L é a matriz triangular inferior

$$L = L^{(n)} = \begin{bmatrix} 1 & 0 & 0 & \dots & 0 \\ m_{2,1} & 1 & 0 & \dots & 0 \\ m_{3,1} & m_{3,2} & 1 & \dots & 0 \\ \vdots & \vdots & \vdots & \dots & \vdots \\ m_{n,1} & m_{n,2} & m_{n,3} & \dots & 1 \end{bmatrix} \tag{3.100}$$

e U é a matriz triangular superior

$$U = U^{(n)} = \begin{bmatrix} u_{1,1}^{(n)} & u_{1,2}^{(n)} & u_{1,3}^{(n)} & \dots & u_{1,n}^{(n)} \\ 0 & u_{2,2}^{(n)} & u_{2,3}^{(n)} & \dots & u_{2,n}^{(n)} \\ 0 & 0 & u_{3,3}^{(n)} & \dots & u_{3,n}^{(n)} \\ \vdots & \vdots & \vdots & \dots & \vdots \\ 0 & 0 & 0 & \dots & u_{n,n}^{(n)} \end{bmatrix}. \tag{3.101}$$

Exemplo 3.4.1. Consideremos a seguinte matriz

$$A = \begin{bmatrix} -1 & 2 & -2 \\ 3 & -4 & 1 \\ 1 & -5 & 3 \end{bmatrix}. \tag{3.102}$$

Então, para obtermos sua decomposição LU começamos com

$$L^{(1)} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix},\ U^{(1)} = \begin{bmatrix} -1 & 2 & -2 \\ 3 & -4 & 1 \\ 1 & -5 & 3 \end{bmatrix} \tag{3.103}$$

Então, observando que a eliminação abaixo do pivô $u_{1,1} = -1$ pode ser feita com as seguintes operações equivalentes por linha $U_2^{(2)} \leftarrow U_2^{(1)} - (-3)U_1^{(1)}$ e $U_3^{(2)} \leftarrow U_3^{(1)} - (-1)U_3^{(1)}$, obtemos

$$L^{(2)} = \begin{bmatrix} 1 & 0 & 0 \\ -3 & 1 & 0 \\ -1 & 0 & 1 \end{bmatrix},\ U^{(1)} = \begin{bmatrix} -1 & 2 & -2 \\ 0 & 2 & -5 \\ 0 & -3 & 1 \end{bmatrix} \tag{3.104}$$

Agora, para eliminarmos abaixo do pivô $u_{2,2} = 2$, usamos a operação $U_3^{(3)} \leftarrow U_3^{(2)} - (-3/2)U_3^{(2)}$, donde temos

$$L^{(2)} = \begin{bmatrix} 1 & 0 & 0 \\ -3 & 1 & 0 \\ -1 & -1{,}5 & 1 \end{bmatrix}, \; U^{(1)} = \begin{bmatrix} -1 & 2 & -2 \\ 0 & 2 & -5 \\ 0 & 0 & -6{,}5 \end{bmatrix} \tag{3.105}$$

o que completa a decomposição tomando $L = L^{(3)}$ e $U = U^{(3)}$.

Exemplo 3.4.2. Vamos resolver o seguinte sistema linear

$$-x_1 + 2x_2 - 2x_3 = 6 \tag{3.106}$$
$$3x_1 - 4x_2 + x_3 = -11 \tag{3.107}$$
$$x_1 - 5x2 + 3x_3 = -10. \tag{3.108}$$

No exemplo anterior (Exemplo 3.4.2), vimos que a matriz de coeficientes A deste sistema admite a seguinte decomposição LU

$$\underbrace{\begin{bmatrix} -1 & 2 & -2 \\ 3 & -4 & 1 \\ 1 & -5 & 3 \end{bmatrix}}_{A} = \underbrace{\begin{bmatrix} 1 & 0 & 0 \\ -3 & 1 & 0 \\ -1 & -1{,}5 & 1 \end{bmatrix}}_{L} \underbrace{\begin{bmatrix} -1 & 2 & -2 \\ 0 & 2 & -5 \\ 0 & 0 & -6{,}5 \end{bmatrix}}_{U} \tag{3.109}$$

Daí, iniciamos resolvendo o seguinte sistema triangular inferior $Ly = b$, i.e.

$$y_1 = -10 \Rightarrow y_1 = 6 \tag{3.110}$$
$$-3y_1 + y_2 = -11 \Rightarrow y_2 = 7 \tag{3.111}$$
$$-y_1 - 1{,}5y_2 + y_3 = -10 \Rightarrow y_3 = 6{,}5. \tag{3.112}$$

Por fim, computamos a solução x resolvendo o sistema triangular superior $Ux = y$, i.e.

$$-6{,}5x_3 = 6{,}5 \Rightarrow x_3 = -1, \tag{3.113}$$
$$2x_2 - 5x_3 = 7 \Rightarrow x_2 = 1 \tag{3.114}$$
$$-x_1 + 2x_2 - 2x_3 = 6 \Rightarrow x_1 = -2. \tag{3.115}$$

3.4.1 Fatoração LU com pivotamento parcial

O algoritmo discutido acima não prevê a necessidade de permutações de linhas no processo de eliminação gaussiana. Isto pode ser corrigido com a utilização de matrizes de permutação.

Assim como na eliminação gaussiana com pivotamento parcial, na fatoração LU com pivotamento parcial o pivô fazemos permutações de linha na matriz de forma que o pivô seja sempre aquele de maior valor em módulo. Por exemplo, suponha que o elemento $a_{3,1}$ seja o maior valor em módulo na primeira coluna da matriz $A = U^{(1)}$ com

$$U^{(1)} = \begin{bmatrix} u_{1,1}^{(1)} & u_{1,2}^{(1)} & u_{1,3}^{(1)} & \dots & u_{1,n}^{(1)} \\ u_{2,1}^{(1)} & u_{2,2}^{(1)} & u_{2,3}^{(1)} & \dots & u_{2,n}^{(1)} \\ \boldsymbol{u_{3,1}^{(1)}} & u_{3,2}^{(1)} & u_{3,3}^{(1)} & \dots & u_{3,n}^{(1)} \\ \vdots & \vdots & \vdots & \dots & \vdots \\ u_{n,1}^{(1)} & u_{n,2}^{(1)} & a_{n,3}^{(1)} & \dots & u_{n,n}^{(1)} \end{bmatrix}. \tag{3.116}$$

Neste caso, o procedimento de eliminação na primeira coluna deve usar $u_{3,1}^{(1)}$ como pivô, o que requer a permutação entre as linhas 1 e 3 ($U_1^{(1)} \leftrightarrow U_3^{(1)}$). Isto pode ser feito utilizando-se da seguinte matriz de permutação

$$P = \begin{bmatrix} 0 & 0 & 1 & \dots & 0 \\ 0 & 1 & 0 & \dots & 0 \\ 1 & 0 & 0 & \dots & 0 \\ \vdots & \vdots & \vdots & \dots & \vdots \\ 0 & 0 & 0 & \dots & 1 \end{bmatrix}. \tag{3.117}$$

Com essa, iniciamos o procedimento de decomposição LU com $PA = L^{(1)}U^{(1)}$, onde $L^{(1)} = I_{n\times n}$ e $U^{(1)} = PA$. Caso sejam necessárias outras mudanças de linhas no decorrer do procedimento de decomposição, a matriz de permutação P deve ser atualizada apropriadamente.

Exemplo 3.4.3. Vamos fazer a decomposição LU com pivotamento parcial da seguinte matriz

$$A = \begin{bmatrix} -1 & 2 & -2 \\ 3 & -4 & 1 \\ 1 & -5 & 3 \end{bmatrix} \tag{3.118}$$

Começamos, tomando

$$P^{(1)} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}, \quad L^{(1)} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}, \quad U^{(1)} = \begin{bmatrix} -1 & 2 & -2 \\ 3 & -4 & 1 \\ 1 & -5 & 3 \end{bmatrix} \tag{3.119}$$

O candidato a pivô é o elemento $u_{2,1}$. Então, fazemos as permutações de linhas $P_1 \leftrightarrow P_2$ e $U_1 \leftrightarrow U_2$ e, na sequência, as operações elementares por linhas $U_{2:3} \leftarrow U_{2:3} - m_{2:3,1}U_1$, donde obtemos

$$P^{(2)} = \begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}, \tag{3.120}$$

$$L^{(2)} = \begin{bmatrix} 1 & 0 & 0 \\ -0{,}\overline{3} & 1 & 0 \\ 0{,}\overline{3} & 0 & 1 \end{bmatrix}, U^{(2)} = \begin{bmatrix} 3 & -4 & 1 \\ 0 & 0{,}\overline{6} & -1{,}\overline{6} \\ 0 & -3{,}\overline{6} & 2{,}\overline{6} \end{bmatrix} \tag{3.121}$$

Agora, o candidato a pivô é o elemento $u_{3,2}$. Assim, fazemos as permutações de linhas $P_2 \leftrightarrow P_3$, $U_2 \leftrightarrow U_3$ (análogo para os elementos da coluna 1 de L) e, então, a operação elementar por linha $U_3 \leftarrow U_3 - m_{3,2}U_2$. Com isso, obtemos

$$P^{(2)} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \end{bmatrix}, \tag{3.122}$$

$$L^{(2)} = \begin{bmatrix} 1 & 0 & 0 \\ 0{,}\overline{3} & 1 & 0 \\ -0{,}\overline{3} & -0{,}\overline{18} & 1 \end{bmatrix}, U^{(2)} = \begin{bmatrix} 3 & -4 & 1 \\ 0 & -3{,}\overline{6} & 2{,}\overline{6} \\ 0 & 0 & -1{,}\overline{18} \end{bmatrix} \tag{3.123}$$

Com isso, temos obtido a decomposição LU de A na forma

$$PA = LU, \tag{3.124}$$

com $P = P^{(3)}$, $L = L^{(3)}$ e $U = U^{(3)}$.

Exercícios

Exercício 3.4.1. Resolva o sistema

$$-x_1 + 2x_2 - 2x_3 = -1 \tag{3.125}$$
$$3x_1 - 4x_2 + x_3 = -4 \tag{3.126}$$
$$-4x_1 - 5x_2 + 3x_3 = 20 \tag{3.127}$$

usando fatoração LU com pivotamento parcial.

Capítulo 4

Métodos iterativos para sistemas lineares

4.1 Métodos de Jacobi e de Gauss-Seidel

Nesta seção, discutiremos os métodos de Jacobi[1] e de Gauss-Seidel[2] para a aproximação da solução de sistemas lineares.

4.1.1 Método de Jacobi

Dado um sistema $A\boldsymbol{x} = \boldsymbol{b}$ com n equações e n incógnitas, consideramos a seguinte decomposição da matriz $A = L + D + U$:

$$A = \begin{bmatrix} a_{11} & a_{12} & a_{13} & \dots & a_{1n} \\ a_{21} & a_{22} & a_{23} & \dots & a_{2n} \\ a_{31} & a_{32} & a_{33} & \dots & a_{3n} \\ \vdots & \vdots & \vdots & \dots & \vdots \\ a_{n1} & a_{n2} & a_{n3} & \dots & a_{nn} \end{bmatrix} \tag{4.1}$$

[1]Carl Gustav Jacob Jacobi, 1804 - 1851, matemático alemão. Fonte: Wikipedia.

[2]Johann Carl Friedrich Gauss, 1777 - 1855, matemático alemão. Philipp Ludwig von Seidel, 1821 - 1896, matemático alemão. Fonte: Wikipedia.

$$= \underbrace{\begin{bmatrix} 0 & 0 & 0 & \dots & 0 \\ a_{21} & 0 & 0 & \dots & 0 \\ a_{31} & a_{32} & 0 & \dots & 0 \\ \vdots & \vdots & \vdots & \dots & \vdots \\ a_{n1} & a_{n2} & a_{n3} & \dots & 0 \end{bmatrix}}_{L} \tag{4.2}$$

$$+ \underbrace{\begin{bmatrix} a_{11} & 0 & 0 & \dots & 0 \\ 0 & a_{22} & 0 & \dots & 0 \\ 0 & 0 & a_{33} & \dots & 0 \\ \vdots & \vdots & \vdots & \dots & \vdots \\ 0 & 0 & 0 & \dots & a_{nn} \end{bmatrix}}_{D} \tag{4.3}$$

$$+ \underbrace{\begin{bmatrix} 0 & a_{12} & a_{13} & \dots & a_{1n} \\ 0 & 0 & a_{23} & \dots & a_{2n} \\ 0 & 0 & a_{33} & \dots & a_{3n} \\ \vdots & \vdots & \vdots & \dots & \vdots \\ 0 & 0 & 0 & \dots & a_{nn} \end{bmatrix}}_{U}. \tag{4.4}$$

Isto é, a matriz A decomposta como a soma de sua parte triangular inferior L, de sua diagonal D e de sua parte triangular superior U.

Desta forma, podemos reescrever o sistema $A\boldsymbol{x} = b$ da seguinte forma:

$$\begin{align} A\boldsymbol{x} = \boldsymbol{b} &\Leftrightarrow (L + D + U)\boldsymbol{x} = \boldsymbol{b} \tag{4.5}\\ &\Leftrightarrow D\boldsymbol{x} = -(L + U)\boldsymbol{x} + \boldsymbol{b} \tag{4.6}\\ &\Leftrightarrow \boldsymbol{x} = -D^{-1}(L + U)\boldsymbol{x} + D^{-1}\boldsymbol{b}. \tag{4.7} \end{align}$$

Ou seja, resolver o sistema $A\boldsymbol{x} = \boldsymbol{b}$ é equivalente a resolver o problema de ponto fixo

$$\boldsymbol{x} = T_J\boldsymbol{x} + \boldsymbol{c}_J, \tag{4.8}$$

onde $T_J = -D^{-1}(L + U)$ é chamada de **matriz de Jacobi** e $\boldsymbol{c}_J = D^{-1}\boldsymbol{b}$ é chamado de **vetor de Jacobi**.

Exemplo 4.1.1. Consideremos o sistema linear $A\boldsymbol{x} = \boldsymbol{b}$ com

$$A = \begin{bmatrix} -4 & 2 & -1 \\ -2 & 5 & 2 \\ 1 & -1 & -3 \end{bmatrix}, \quad \boldsymbol{b} = \begin{bmatrix} -11 \\ -7 \\ 0 \end{bmatrix}. \tag{4.9}$$

Este sistema tem solução $\boldsymbol{x} = (2, -1, 1)$. Neste caso, temos a decomposição $A = L + D + U$ com

$$L = \begin{bmatrix} 0 & 0 & 0 \\ -2 & 0 & 0 \\ 1 & -1 & 0 \end{bmatrix}, \quad D = \begin{bmatrix} -4 & 0 & 0 \\ 0 & 5 & 0 \\ 0 & 0 & -3 \end{bmatrix} \tag{4.10}$$

e

$$U = \begin{bmatrix} 0 & 2 & -1 \\ 0 & 0 & 2 \\ 0 & 0 & 0 \end{bmatrix}. \tag{4.11}$$

Ainda, observamos que

$$T_J\boldsymbol{x} + \boldsymbol{c}_J = -D^{-1}(L+U)\boldsymbol{x} + D^{-1}\boldsymbol{b} \tag{4.12}$$

$$= \underbrace{\begin{bmatrix} 0 & 1/2 & 1/4 \\ 2/5 & 0 & -2/5 \\ 1/3 & -1/3 & 0 \end{bmatrix}}_{T_J} \underbrace{\begin{bmatrix} 2 \\ -1 \\ 1 \end{bmatrix}}_{\boldsymbol{x}} + \underbrace{\begin{bmatrix} 11/4 \\ -7/5 \\ 0 \end{bmatrix}}_{\boldsymbol{c}_J} \tag{4.13}$$

$$= \underbrace{\begin{bmatrix} 2 \\ -1 \\ 1 \end{bmatrix}}_{\boldsymbol{x}}. \tag{4.14}$$

Com o exposto acima, o **método de Jacobi** consiste na seguinte iteração de ponto fixo

$$\boldsymbol{x}^{(1)} = \text{aprox. inicial}, \tag{4.15}$$

$$\boldsymbol{x}^{(k+1)} = T_J\boldsymbol{x}^{(k)} + \boldsymbol{c}_J, \tag{4.16}$$

onde $\boldsymbol{x}^{(k)} = (x_1^{(k)}, x_2^{(k)}, \ldots, x_n^{(k)})$ é a k-ésima aproximação (ou iterada) de Jacobi.

A iteração (4.16) pode ser equivalentemente escrita na seguinte forma algébrica

$$x_i^{(k+1)} = \frac{b_i - \displaystyle\sum_{\substack{j=1 \\ j\neq i}}^{n} a_{ij}x^{(k)}}{a_{ii}}, \; i = 1, 2, \ldots, n, \tag{4.17}$$

a qual não requer a computação da matriz T_J e $\boldsymbol{c}_J$.

Exemplo 4.1.2. Consideremos o sistema $A\boldsymbol{x} = \boldsymbol{b}$ com

$$A = \begin{bmatrix} -4 & 2 & -1 \\ -2 & 5 & 2 \\ 1 & -1 & -3 \end{bmatrix}, \quad \boldsymbol{b} = \begin{bmatrix} -11 \\ -7 \\ 0 \end{bmatrix}. \tag{4.18}$$

Aplicando o método de Jacobi com aproximação inicial $\boldsymbol{x}^{(1)} = (0, 0, 0)$ obtemos os resultados da Tabela 4.1.

k	$\boldsymbol{x}^{(k)}$	$\\|A\boldsymbol{x}^{(k)} - \boldsymbol{b}\\|$
1	(0,0, 0,0, 0,0)	1,3E+1
2	(2,8, − 1,4, 0,0)	7,4E+0
3	(2,0, − 0,3, 1,4)	4,6E+0
4	(2,3, − 1,1, 0,8)	2,2E+0
5	(2,0, − 0,8, 1,1)	1,4E+0
6	(2,1, − 1,1, 0,9)	6,9E−1
7	(2,0, − 0,9, 1,0)	4,2E−1
8	(2,0, − 1,0, 1,0)	2,2E−1
9	(2,0, − 1,0, 1,0)	1,3E−1
10	(2,0, − 1,0, 1,0)	6,9E−2

Tabela 4.1: Resultados referentes ao Exemplo 4.1.4.

4.1.2 Método de Gauss-Seidel

Como acima, começamos considerando um sistema linear $A\boldsymbol{x} = \boldsymbol{b}$ e a decomposição $A = L + D + U$, onde L é a parte triangular inferior de A, D é sua parte diagonal e U sua parte triangular superior. Então, observamos que

$$\begin{aligned} A\boldsymbol{x} = \boldsymbol{b} &\Leftrightarrow (L + D + U)\boldsymbol{x} = \boldsymbol{b} && (4.19) \\ &\Leftrightarrow (L + D)\boldsymbol{x} = -U\boldsymbol{x} + \boldsymbol{b} && (4.20) \\ &\Leftrightarrow \boldsymbol{x} = -(L + D)^{-1}U\boldsymbol{x} + (L + D)^{-1}\boldsymbol{b}. && (4.21) \end{aligned}$$

Isto nos leva a iteração de Gauss-Seidel

$$\begin{aligned} \boldsymbol{x}^{(1)} &= \text{aprox. inicial}, && (4.22) \\ \boldsymbol{x}^{(k+1)} &= T_G\boldsymbol{x}^{(k)} + \boldsymbol{c}_G, && (4.23) \end{aligned}$$

onde $T_G = -(L+D)^{-1}U$ é a chamada **matriz de Gauss-Seidel** e $\boldsymbol{c}_G = (L+D)^{-1}\boldsymbol{b}$ é o chamado **vetor de Gauss-Seidel**.

Observamos, também, que a iteração (4.23) pode ser reescrita na seguinte forma algébrica

$$x_i^{(k+1)} = \frac{b_i - \sum_{j=1}^{i-1} a_{ij}x^{(k+1)} - \sum_{j=i+1}^{n} a_{ij}x^{(k)}}{a_{ii}}, \; i = 1, 2, \ldots, n. \tag{4.24}$$

Exemplo 4.1.3. Consideremos o sistema $A\boldsymbol{x} = \boldsymbol{b}$ com

$$A = \begin{bmatrix} -4 & 2 & -1 \\ -2 & 5 & 2 \\ 1 & -1 & -3 \end{bmatrix}, \quad \boldsymbol{b} = \begin{bmatrix} -11 \\ -7 \\ 0 \end{bmatrix}. \tag{4.25}$$

Aplicando o método de Gauss-Seidel com aproximação inicial $\boldsymbol{x}^{(1)} = (0, 0, 0)$ obtemos os resultados da Tabela 4.2.

k	$\boldsymbol{x}^{(k)}$	$\|A\boldsymbol{x}^{(k)} - \boldsymbol{b}\|$
1	(0,0, 0,0, 0,0)	1,3E+1
2	(2,8, − 0,3, 1,0)	2,6E+0
3	(2,3, − 0,9, 1,1)	1,2E+0
4	(2,0, − 1,0, 1,0)	2,5E−1
5	(2,0, − 1,0, 1,0)	4,0E−2

Tabela 4.2: Resultados referentes ao Exemplo 4.1.3.

4.1.3 Análise de convergência

Observamos que ambos os métodos de Jacobi e de Gauss-Seidel consistem de iterações da forma

$$\boldsymbol{x}^{(k+1)} = T\boldsymbol{x}^{(k)} + \boldsymbol{c}, \; k = 1, 2, \ldots, \tag{4.26}$$

com $x^{(1)}$ uma aproximação inicial dada, T e c a matriz e o vetor de iteração, respectivamente. O seguinte teorema nos fornece uma condição suficiente e necessária para a convergência de tais métodos.

Teorema 4.1.1. *Para qualquer* $\boldsymbol{x}^{(1)} \in \mathbb{R}^n$*, temos que a sequência* $\{\boldsymbol{x}^{(k+1)}\}_{k=1}^{\infty}$ *dada por*

$$\boldsymbol{x}^{(k+1)} = T\boldsymbol{x}^{(k)} + \boldsymbol{c}, \tag{4.27}$$

converge para a solução única de $\boldsymbol{x} = T\boldsymbol{x} + \boldsymbol{c}$ *se, e somente se,* $\rho(T) < 1$[3].

Demonstração. Veja [2, Cap. 7, Sec. 7.3]. □

Observação 4.1.1. (Taxa de convergência) Para uma iteração da forma (4.26), vale

$$\|\boldsymbol{x}^{(k)} - \boldsymbol{x}\| \approx \rho(T)^{k-1}\|\boldsymbol{x}^{(1)} - \boldsymbol{x}\|, \tag{4.28}$$

onde $\boldsymbol{x}$ é a solução de $\boldsymbol{x} = T\boldsymbol{x} + \boldsymbol{c}$.

Exemplo 4.1.4. Consideremos o sistema $A\boldsymbol{x} = \boldsymbol{b}$ com

$$A = \begin{bmatrix} -4 & 2 & -1 \\ -2 & 5 & 2 \\ 1 & -1 & -3 \end{bmatrix}, \quad \boldsymbol{b} = \begin{bmatrix} -11 \\ -7 \\ 0 \end{bmatrix}. \tag{4.29}$$

Nos Exemplos 4.1.4 e 4.1.3 vimos que ambos os métodos de Jacobi e de Gauss-Seidel eram convergentes, sendo que este convergiu aproximadamente duas vezes mais rápido que esse. Isto é confirmado pelos raios espectrais das respectivas matrizes de iteração

$$\rho(T_J) \approx 0{,}56, \quad \rho(T_G) \approx 0{,}26. \tag{4.30}$$

Observação 4.1.2. Matriz estritamente diagonal dominante Pode-se mostrar que se A é uma matriz estritamente diagonal dominante, i.e. se

$$|a_{ii}| > \sum_{\substack{j=1 \\ j\neq i}}^{n} |a_{ij}|, \; \forall i = 1, 2, \dots, n, \tag{4.31}$$

então ambos os métodos de Jacobi e de Gauss-Seidel são convergentes.

[3] $\rho(T)$ é o raio espectral da matriz T, i.e. o máximo dos módulos dos autovalores de T.

Exercícios

Exercício 4.1.1. Considere o seguinte sistema linear

$$-4x_1 + x_2 + x_3 - x_4 = -1 \tag{4.32}$$
$$5x_2 - x_3 + 2x_4 = 3 \tag{4.33}$$
$$-x_1 + 4x_3 - 2x_4 = -2 \tag{4.34}$$
$$x_1 - x_2 - 5x_4 = 1 \tag{4.35}$$

Compute a quinta iterada $x^{(5)}$ do método de Jacobi aplicado a este sistema com aproximação inicial $x^{(1)} = (1, 1, -1, -1)$. Também, compute $\|Ax^{(5)} - b\|$.

Exercício 4.1.2. Considere o seguinte sistema linear

$$-4x_1 + x_2 + x_3 - x_4 = -1 \tag{4.36}$$
$$5x_2 - x_3 + 2x_4 = 3 \tag{4.37}$$
$$-x_1 + 4x_3 - 2x_4 = -2 \tag{4.38}$$
$$x_1 - x_2 - 5x_4 = 1 \tag{4.39}$$

Compute a quinta iterada $x^{(5)}$ do método de Gauss-Seidel aplicado a este sistema com aproximação inicial $x^{(1)} = (1, 1, -1, -1)$. Também, compute $\|Ax^{(5)} - b\|$.

4.2 Método do gradiente

Começamos observando que se A é uma matriz $n \times n$ positiva definida[4], temos que $\boldsymbol{x} \in \mathbb{R}^n$ é solução de

$$A\boldsymbol{x} = \boldsymbol{b} \tag{4.40}$$

se, e somente se, $\boldsymbol{x}$ é solução do seguinte problema de minimização

$$\min_{\boldsymbol{x}\in\mathbb{R}^n} f(\boldsymbol{x}) := \frac{1}{2}\boldsymbol{x}^T A\boldsymbol{x} - \boldsymbol{x}^T\boldsymbol{b}. \tag{4.41}$$

[4] A é simétrica e $x^T Ax > 0$ para todo $x \neq 0$.

O método do gradiente é um algoritmo da forma: dada uma aproximação inicial $\boldsymbol{x}^{(1)}$ da solução de (4.41) (ou, equivalentemente, de (4.40)), computamos novas aproximações da forma iterativa

$$\boldsymbol{x}^{(k+1)} = \boldsymbol{x}^{(k)} + \alpha^{(k)}\boldsymbol{d}^{(k)}, \quad k = 1, 2, \dots, \tag{4.42}$$

onde $\alpha^{(k)}$ é o tamanho do passo (um escalar) e $\boldsymbol{d}^{(k)} \in \mathbb{R}^n$ é a direção de busca.

Para escolhermos a direção $\boldsymbol{d}^{(k)}$, tomamos a fórmula de Taylor de f em torno da aproximação $\boldsymbol{x}^{(k)}$

$$f(\boldsymbol{x}^{(k+1)}) = f(\boldsymbol{x}^{(k)}) + \alpha^{(k)}\nabla f(\boldsymbol{x}^{(k)}) \cdot \boldsymbol{d}^{(k)} + O\left((\alpha^{(k)})^2\right), \tag{4.43}$$

com $\alpha^{(k)} \to 0$, onde ∇f denota o gradiente de f, i.e.

$$\begin{align}\nabla f(\boldsymbol{x}^{(k)}) &= \left(\frac{\partial f}{\partial x_1}(\boldsymbol{x}^{(k)}), \frac{\partial f}{\partial x_2}(\boldsymbol{x}^{(k)}), \dots, \frac{\partial f}{\partial x_n}(\boldsymbol{x}^{(k)})\right) \tag{4.44}\\ &= A\boldsymbol{x}^{(k)} - \boldsymbol{b}. \tag{4.45}\end{align}$$

De (4.43), segue que se

$$\nabla f(\boldsymbol{x}^{(k)}) \cdot \boldsymbol{d}^{(k)} < 0, \tag{4.46}$$

então $f(\boldsymbol{x}^{(k+1)}) < f(\boldsymbol{x}^{(k)})$ se $\alpha^{(k)}$ é suficientemente pequeno. Em particular, podemos escolher

$$\boldsymbol{d}^{(k)} = -\nabla f(\boldsymbol{x}^{(k)}), \tag{4.47}$$

se $\nabla f(\boldsymbol{x}^{(k)}) \neq 0$.

Do exposto acima, temos a **iteração do método do gradiente**

$$\begin{align}\boldsymbol{x}^{(1)} &= \text{aprox. inicial} \tag{4.48}\\ \boldsymbol{x}^{(k+1)} &= \boldsymbol{x}^{(k)} - \alpha^{(k)}\boldsymbol{r}^{(k)},\ k = 1, 2, \dots, \tag{4.49}\end{align}$$

onde $\boldsymbol{r}^{(k)}$ é o resíduo da iterada k dado por

$$\boldsymbol{r}^{(k)} = A\boldsymbol{x}^{(\boldsymbol{k})} - \boldsymbol{b}. \tag{4.50}$$

Tabela 4.3: Resultados referentes ao Exemplo 4.2.1.

k	$\boldsymbol{x}^{(k)}$	$\|A\boldsymbol{x}^{(k)} - \boldsymbol{b}\|$
1	(0,0, 0,0, 0,0, 0,0)	5,1E+0
2	(−1,5, 1,0, 1,0, −1,5)	1,6E+0
3	(−1,0, 0,8, 0,8, −1,0)	5,0E−1
4	(−1,1, 0,9, 0,9, −1,1)	1,8E−1
5	(−1,1, 0,9, 0,9, −1,1)	8,8E−2
6	(−1,1, 0,9, 0,9, −1,1)	6,2E−2
7	(−1,0, 0,9, 0,9, −1,0)	4,9E−2
8	(−1,0, 0,9, 0,9, −1,0)	4,0E−2
9	(−1,0, 0,9, 0,9, −1,0)	3,2E−2
10	(−1,0, 1,0, 1,0, −1,0)	2,6E−2
11	(−1,0, 1,0, 1,0, −1,0)	2,1E−2

Exemplo 4.2.1. Consideremos o sistema $Ax = b$ com

$$A = \begin{bmatrix} 2 & -1 & 0 & 0 \\ -1 & 2 & -1 & 0 \\ 0 & -1 & 2 & -1 \\ 0 & 0 & -1 & 2 \end{bmatrix}, \quad b = \begin{bmatrix} -3 \\ 2 \\ 2 \\ -3 \end{bmatrix}. \tag{4.51}$$

Na Tabela 4.3 temos os resultados do emprego do método do gradiente com $\boldsymbol{x}^{(1)} = (0, 0, 0, 0)$ e com passo constante $\alpha^{(k)} \equiv 0{,}5$.

4.2.1 Escolha do passo

Da iteração do método do gradiente, temos que a melhor escolha do passo $\alpha^{(k)}$ é tal que

$$f(\boldsymbol{x}^{(k)} + \alpha^{(k)}\boldsymbol{r}^{(k)}) = \min_{\alpha>0} f(\boldsymbol{x}^{(k)} + \alpha\boldsymbol{r}^{(k)}). \tag{4.52}$$

Desta forma,

$$\frac{\mathrm{d}}{\mathrm{d}\alpha} f(\boldsymbol{x}^{(k)} + \alpha\boldsymbol{r}^{(k)}) = 0 \Rightarrow \nabla f(\boldsymbol{x}^{(k+1)}) \cdot \boldsymbol{r}^{(k)} = 0, \tag{4.53}$$

$$\Rightarrow \left(A(\boldsymbol{x}^{(k)} + \alpha^{(k)}\boldsymbol{r}^{(k)}) - b\right) \cdot \boldsymbol{r}^{(k)} = 0, \tag{4.54}$$

$$\Rightarrow (A\boldsymbol{x}^{(k)} - \boldsymbol{b}) \cdot \boldsymbol{r}^{(k)} + \alpha^{(k)}\boldsymbol{r}^{(k)} \cdot A\boldsymbol{r}^{(k)} = 0, \tag{4.55}$$

donde

$$\alpha^{(k)} = -\frac{\boldsymbol{r}^{(k)} \cdot \boldsymbol{r}^{(k)}}{\boldsymbol{r}^{(k)} \cdot A\boldsymbol{r}^{(k)}}. \tag{4.56}$$

Exemplo 4.2.2. Consideremos o sistema $Ax = b$ com

$$A = \begin{bmatrix} 2 & -1 & 0 & 0 \\ -1 & 2 & -1 & 0 \\ 0 & -1 & 2 & -1 \\ 0 & 0 & -1 & 2 \end{bmatrix}, \quad b = \begin{bmatrix} -3 \\ 2 \\ 2 \\ -3 \end{bmatrix}. \tag{4.57}$$

Na Tabela 4.4 temos os resultados do emprego do método do gradiente com $\boldsymbol{x}^{(1)} = (0, 0, 0, 0)$ e com passo

$$\alpha^{(k)} = -\frac{\boldsymbol{r}^{(k)} \cdot \boldsymbol{r}^{(k)}}{\boldsymbol{r}^{(k)} \cdot A\boldsymbol{r}^{(k)}}. \tag{4.58}$$

Tabela 4.4: Resultados referentes ao Exemplo 4.2.2.

k	$\boldsymbol{x}^{(k)}$	$\|A\boldsymbol{x}^{(k)} - \boldsymbol{b}\|$
1	(0,0, 0,0, 0,0, 0,0)	5,1E+0
2	(−1,1, 0,8, 0,8, − 1,1)	1,5E−1
3	(−1,0, 1,0, 1,0, − 1,0)	3,0E−2
4	(−1,0, 1,0, 1,0, − 1,0)	8,8E−4
5	(−1,0, 1,0, 1,0, − 1,0)	1,8E−4

Exercícios

Em construção ...

4.3 Método do gradiente conjugado

O método do gradiente conjugado é uma variação do método do gradiente (veja Seção 4.2). Aqui, a solução de um dado sistema $A\boldsymbol{x} = \boldsymbol{b}$, com A uma matriz positiva definida, é computada de forma iterativa por

$$\boldsymbol{x}^{(1)} = \text{aprox. inicial}, \tag{4.59}$$

$$\boldsymbol{d}^{(1)} = \boldsymbol{r}^{(1)}, \tag{4.60}$$

$$\tag{4.61}$$

$$\boldsymbol{x}^{(k+1)} = \boldsymbol{x}^{(k)} + \alpha_k \boldsymbol{d}^{(k)}, \tag{4.62}$$

$$\alpha^{(k)} = -\frac{\boldsymbol{r}^{(k)} \cdot \boldsymbol{d}^{(k)}}{\boldsymbol{d}^{(k)} \cdot A\boldsymbol{d}^{(k)}}, \tag{4.63}$$

$$\boldsymbol{d}^{(k+1)} = -\boldsymbol{r}^{(k+1)} + \beta_k \boldsymbol{d}^{(k)}, \tag{4.64}$$

$$\beta^{(k)} = \frac{\boldsymbol{r}^{(k+1)} \cdot A\boldsymbol{d}^{(k)}}{\boldsymbol{d}^{(k)} \cdot A\boldsymbol{d}^{(k)}}, \tag{4.65}$$

para $k = 1, 2, \ldots,$ e $\boldsymbol{r}^{(k)} = A\boldsymbol{x}^{(k)} - \boldsymbol{b}$.

Exemplo 4.3.1. Consideremos o sistema $Ax = b$ com

$$A = \begin{bmatrix} 2 & -1 & 0 & 0 \\ -1 & 2 & -1 & 0 \\ 0 & -1 & 2 & -1 \\ 0 & 0 & -1 & 2 \end{bmatrix}, \quad b = \begin{bmatrix} -3 \\ 2 \\ 2 \\ -3 \end{bmatrix}. \tag{4.66}$$

Na Tabela 4.5 temos os resultados do emprego do método do gradiente conjugado com $\boldsymbol{x}^{(1)} = (0, 0, 0, 0)$.

Tabela 4.5: Resultados referentes ao Exemplo 4.3.1.

k	$\boldsymbol{x}^{(k)}$	$\|A\boldsymbol{x}^{(k)} - \boldsymbol{b}\|$
1	(0, 0, 0, 0)	5.1E+0
2	$(-1{,}1,\ 0{,}8,\ 0{,}8,\ -1{,}1)$	1,5E−1
3	$(-1{,}0,\ 1{,}0,\ 1{,}0,\ -1{,}0)$	0,0E+0

Exercícios

Em construção ...

Capítulo 5

Sistema de equações não lineares

Neste capítulo, discutiremos sobre métodos para a resolução de sistema de equações não lineares. Vamos tratar o caso de problemas da forma: encontrar $\boldsymbol{x} \in \mathbb{R}^n$ tal que

$$F(\boldsymbol{x}) = \boldsymbol{0}, \tag{5.1}$$

onde $F : \mathbb{R}^n \to \mathbb{R}^n$ é uma função vetorial dada.

5.1 Método de Newton

Consideremos o problema de encontrar $\boldsymbol{x} = (x_1, x_2, \ldots, x_n) \in \mathbb{R}^n$ tal que

$$F(\boldsymbol{x}) = \boldsymbol{0}, \tag{5.2}$$

onde $F : \mathbb{R}^n \to \mathbb{R}^n$ é uma função vetorial dada com $F(\boldsymbol{x}) = (f_1(\boldsymbol{x}), f_2(\boldsymbol{x}), \ldots, f_n(\boldsymbol{x}))$. Suponhamos, então, que $\boldsymbol{x}^*$ seja a solução exata deste problema e que $\boldsymbol{x}^{(1)}$ seja uma aproximação de $\boldsymbol{x}^*$. Assim sendo, tomemos a seguinte expansão de F em polinômio de Taylor:

$$F(\boldsymbol{x}^*) = F(\boldsymbol{x}^{(1)}) + J_F(\boldsymbol{x}^{(1)})(\boldsymbol{x}^* - \boldsymbol{x}^{(1)}) + R, \tag{5.3}$$

onde J_F é a matriz jacobiana[1] de F

$$J_F := \begin{bmatrix} \frac{\partial f_1}{\partial x_1} & \frac{\partial f_1}{\partial x_2} & \cdots & \frac{\partial f_1}{\partial x_n} \\ \frac{\partial f_2}{\partial x_1} & \frac{\partial f_2}{\partial x_2} & \cdots & \frac{\partial f_2}{\partial x_n} \\ \vdots & \vdots & \vdots & \vdots \\ \frac{\partial f_n}{\partial x_1} & \frac{\partial f_n}{\partial x_2} & \cdots & \frac{\partial f_n}{\partial x_n} \end{bmatrix} \tag{5.4}$$

e $\|R\|^2 \to 0$ com $\|\boldsymbol{x}^{(1)} - \boldsymbol{x}^*\| \to 0$.

Daí, como $F(\boldsymbol{x}^*) = \mathbf{0}$, temos

$$J_F(\boldsymbol{x}^{(1)})(\boldsymbol{x}^* - \boldsymbol{x}^{(1)}) \approx -F(\boldsymbol{x}^{(1)}). \tag{5.5}$$

Então, multiplicando a inversa da jacobiana à esquerda, obtemos

$$\boldsymbol{x}^* - \boldsymbol{x}^{(1)} \approx -J_F^{-1}(\boldsymbol{x}^{(1)})F(\boldsymbol{x}^{(1)}) \tag{5.6}$$

e também

$$\boldsymbol{x}^* \approx \boldsymbol{x}^{(1)} - J_F^{-1}(\boldsymbol{x}^{(1)})F(\boldsymbol{x}^{(1)}). \tag{5.7}$$

O exposto acima nos motiva a **iteração de Newton**[2]:

$$\boldsymbol{x}^{(1)} = \text{aprox. inicial}, \tag{5.8}$$

$$\boldsymbol{x}^{(k+1)} = \boldsymbol{x}^{(k)} - J_F^{-1}(\boldsymbol{x}^{(k)})F(\boldsymbol{x}^{(k)}), \tag{5.9}$$

com $k = 1, 2, \ldots$.

Exemplo 5.1.1. Consideremos o seguinte sistema de equações não lineares

$$x_1x_2^2 = x_1^2x_2 - 6, \tag{5.10}$$

$$x_1^2x_2^3 - 7 = -x_1. \tag{5.11}$$

Para usarmos o método de Newton, reescrevemos o sistema na seguinte forma

$$x_1x_2^2 - x_1^2x_2 + 6 = 0, \tag{5.12}$$

$$x_1 + x_1^2x_2^3 - 7 = 0. \tag{5.13}$$

[1] Carl Gustav Jacob Jacobi, 1804 - 1851, matemático alemão. Fonte: Wikipedia.

[2] Sir Isaac Newton, 1642 - 1726/27, matemático e físico inglês. Fonte: Wikipedia.

Com isso, identificamos a função objetivo

$$F(\boldsymbol{x}) = \begin{bmatrix} f_1(\boldsymbol{x}) \\ f_2(\boldsymbol{x}) \end{bmatrix} = \begin{bmatrix} x_1x_2^2 - x_1^2x_2 + 6 \\ x_1 + x_1^2x_2^3 - 7 \end{bmatrix} \tag{5.14}$$

e sua matriz jacobiana

$$J_F(\boldsymbol{x}) = \frac{\partial(f_1, f_2)}{\partial(x_1, x_2)} = \begin{bmatrix} \frac{\partial f_1}{\partial x_1} & \frac{\partial f_1}{\partial x_2} \\ \frac{\partial f_2}{\partial x_1} & \frac{\partial f_2}{\partial x_2} \end{bmatrix} \tag{5.15}$$

$$= \begin{bmatrix} x_2^2 - 2x_1x_2 & 2x_1x_2 - x_1^2 \\ 1 + 2x_1x_2^3 & 3x_1^2x_2^2 \end{bmatrix} \tag{5.16}$$

Definidas F e J_F e tomando $\boldsymbol{x}^{(1)} = (1{,}5,\ 1{,}5)$ como aproximação inicial, computamos as iterações de Newton de forma a obtermos os resultados apresentados na Tabela 5.1.

k	$\boldsymbol{x}^{(k)}$	$\|F(\boldsymbol{x}^{(k)})\|$
1	$(-1{,}50,\ 1{,}50)$	1,2E+0
2	$(-1{,}07,\ 1{,}82)$	1,2E+0
3	$(-9{,}95\mathrm{E}{-1},\ 2{,}00)$	7,6E−2
4	$(-1{,}00,\ 2{,}00)$	1,2E−4
5	$(-1{,}00,\ 2{,}00)$	2,1E−9

Tabela 5.1: Resultados referentes ao Exemplo 5.1.1.

5.1.1 Considerações sobre convergência

Para uma função F suficientemente suave e com uma escolha apropriada da aproximação inicial $\boldsymbol{x}^{(1)}$, temos que as iterações de Newton

$$\boldsymbol{x}^{(k+1)} = \boldsymbol{x}^{(k)} - J_F^{-1}(\boldsymbol{x}^{(1)})F(\boldsymbol{x}^{(1)}), \tag{5.17}$$

são quadraticamente convergentes, i.e.

$$\|\boldsymbol{x}^{(k+1)} - \boldsymbol{x}^*\| \leq C\|\boldsymbol{x}^{(k)} - \boldsymbol{x}^*\|^2, \tag{5.18}$$

onde $\boldsymbol{x}^*$ é a solução exata, i.e. $F(\boldsymbol{x}^*) = \boldsymbol{0}$.

Exemplo 5.1.2. Consideremos o seguinte sistema de equações não lineares

$$x_1x_2^2 - x_1^2x_2 + 6 = 0, \tag{5.19}$$

$$x_1 + x_1^2x_2^3 - 7 = 0. \tag{5.20}$$

A Figura 5.1 é um esboço do gráfico da $\|F(\cdot)\|$. Este problema foi confeccionado de forma que $\boldsymbol{x}^* = (-1,\ 2)$. Então, tomando $\boldsymbol{x}^{(1)} = (1{,}5,\ 1{,}5)$ como aproximação inicial, computamos as iterações de Newton para este problema, donde obtemos os resultados reportados na Tabela 5.2.

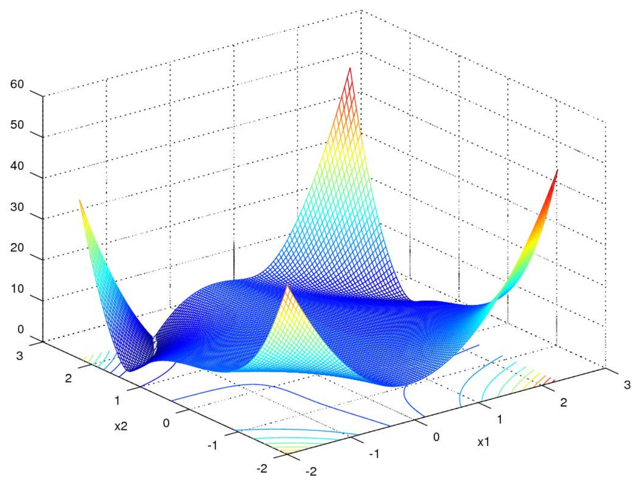

Figura 5.1: Esboço do gráfico de $\|F(\cdot)\|$ referente ao Exemplo 5.1.2.

k	$\boldsymbol{x}^{(k)}$	$\|\boldsymbol{x}^{(k)} - \boldsymbol{x}^*\|$
1	$(-1{,}50,\ 1{,}50)$	$7{,}1\text{E}{-}01$
2	$(-1{,}07,\ 1{,}82)$	$2{,}0\text{E}{-}01$
3	$(-9{,}95\text{E}{-}1,\ 2{,}00)$	$5{,}1\text{E}{-}03$
4	$(-1{,}00,\ 2{,}00)$	$2{,}6\text{E}{-}05$
5	$(-1{,}00,\ 2{,}00)$	$2{,}0\text{E}{-}10$

Tabela 5.2: Resultados referentes ao Exemplo 5.1.2.

Exercícios

Exercício 5.1.1. Use o método de Newton para obter uma aproximação de uma solução de

$$x_2 \operatorname{sen}(x_3) + x_1 - 2 = 0, \tag{5.21}$$
$$x_1 x_2 - \operatorname{sen}(x_2) + 0{,}2 = 0, \tag{5.22}$$
$$x_3^2 + \cos(x_1 x_2) - 4{,}5 = 0. \tag{5.23}$$

Para tanto, use $\boldsymbol{x}^{(1)} = (1,\ -1,\ -1)$.

5.2 Métodos *quasi*-Newton

5.2.1 Método do acorde

O método do acorde consiste na seguinte iteração

$$\boldsymbol{x}^{(1)} = \text{aprox. inicial}, \tag{5.24}$$
$$\boldsymbol{x}^{(k+1)} = \boldsymbol{x}^{(k)} - J_F^{-1}(\boldsymbol{x}^{(1)}) F(\boldsymbol{x}^{(k)}). \tag{5.25}$$

Ou seja, é a iteração de Newton com jacobina constante.

Exemplo 5.2.1. Consideremos o seguinte sistema de equações não lineares

$$x_1 x_2^2 - x_1^2 x_2 + 6 = 0, \tag{5.26}$$
$$x_1 + x_1^2 x_2^3 - 7 = 0. \tag{5.27}$$

Definidas F e J_F e tomando $\boldsymbol{x}^{(1)} = (1{,}5,\ 1{,}5)$ como aproximação inicial, computamos as iterações do método do acorde de forma a obtermos os resultados apresentados na Tabela 5.3.

5.2.2 Jacobiana aproximada

A jacobiana $J_F(\boldsymbol{x})$ de uma dada função $F(\boldsymbol{x}) = (f_1(\boldsymbol{x}), f_2(\boldsymbol{x}), \ldots, f_n(\boldsymbol{x}))$ é a matriz cujo elemento da i-ésima linha e j-ésima coluna é

$$\frac{\partial f_i}{\partial x_j} = \lim_{h \to 0} \frac{f_i(\boldsymbol{x} + \boldsymbol{e}_j h) - f_i(\boldsymbol{x})}{h}, \tag{5.28}$$

onde $\boldsymbol{e}_j$ é o j-ésimo vetor da base canônica de $\mathbb{R}^n$, i.e. $\boldsymbol{e}_j = (0, \ldots, 0, 1, 0, \ldots, 0)$ com 1 na j-ésima posição.

k	$\boldsymbol{x}^{(k)}$	$\|\boldsymbol{x}^{(k)} - \boldsymbol{x}^*\|$
1	$(-1{,}50,\ 1{,}50)$	-x-
2	$(-1{,}07,\ 1.82)$	5,3E$-$1
3	$(-1{,}02,\ 1{,}93)$	1,2E$-$1
4	$(-1{,}00,\ 1{,}98)$	5,2E$-$2
5	$(-9{,}98\text{E}-1,\ 2{,}00)$	1,8E$-$2
6	$(-9{,}98\text{E}-1,\ 2{,}00)$	4,7E$-$3
7	$(-9{,}99\text{E}-1,\ 2{,}00)$	9,0E$-$4
8	$(-1{,}00,\ 2{,}00)$	7,4E$-$4
9	$(-1{,}00,\ 2{,}00)$	4,3E$-$4

Tabela 5.3: Resultados referentes ao Exemplo 5.2.1.

Com isso, podemos computar uma jacobiana aproximada tomando

$$\frac{\partial f_i}{\partial x_j} \approx \frac{f_i(\boldsymbol{x} + \boldsymbol{e}_j h) - f_i(\boldsymbol{x})}{h}, \tag{5.29}$$

com h suficientemente pequeno.

Exemplo 5.2.2. Consideremos o seguinte sistema de equações não lineares

$$x_1 x_2^2 - x_1^2 x_2 + 6 = 0, \tag{5.30}$$
$$x_1 + x_1^2 x_2^3 - 7 = 0. \tag{5.31}$$

Definida F, sua jacobina aproximada $\tilde{J}_F$ e tomando $\boldsymbol{x}^{(1)} = (1{,}5,\ 1{,}5)$ como aproximação inicial, computamos as iterações do *quasi*-método de forma a obtermos os resultados apresentados na Tabela 5.4.

k	$\boldsymbol{x}^{(k)}$	$\|\boldsymbol{x}^{(k)} - \boldsymbol{x}^*\|$
1	$(-1{,}50,\ 1{,}50)$	-x-
2	$(-1{,}07,\ 1{,}82)$	5,3E$-$1
3	$(-9{,}95\text{E}-1,\ 2{,}00)$	2,0E$-$1
4	$(-1{,}00,\ 2{,}00)$	5,1E$-$3
5	$(-1{,}00,\ 2{,}00)$	2,6E$-$5

Tabela 5.4: Resultados referentes ao Exemplo 5.2.2.

Exercícios

Em construção ...

Resposta dos Exercícios

Exercício 1.1.1. a) 45,15625; b) 11,25; c) 55981; d) 27,140625; e) 1220

Exercício 1.1.2. a) $(1010)_2$; b) $(101101{,}1)_2$; c) $(51)_8$; d) $(42{,}51)_{16}$; e) $(0{,}1)_3$

Exercício 1.1.3. a) $(231{,}022)_4$; b) $(1011{,}01)_2$; c) $(20001)_8$

Exercício 1.1.4. a) $0{,}\overline{3}$; b) 1.5; c) $(0{,}\overline{25})_8$;

Exercício 1.1.5. a) $(0{,}1\overline{03})_4$; b) $(0{,}\overline{27})$; c) $(2{,}2)_5$

Exercício 1.2.1. a) [10001000]; b) [11110111]
c) [00000100]; d) [00000111]

Exercício 1.2.2. a) [0000000000100000];
b) [0000000000111111];

Exercício 1.2.3. $[10111\ \ldots\ 11] \sim -3$

Exercício 1.2.4. a) $[1 \mid 0\ 1\ 1\ \ldots\ 1 \mid 1\ 0\ 1\ 0\ 0\ \ldots\ 0]$;
b) $[0 \mid 1\ 0\ 0\ \ldots\ 0 \mid 1\ 0\ 0\ \ldots\ 0]$

Exercício 1.2.5.

a)
```
[0, 0, 1, 1, 1, 1, 0, 1,
```

```
         1, 1, 0, 0, 1, 1, 0, 0,
         1, 1, 0, 0, 1, 1, 0, 0,
         1, 1, 0, 0, 1, 1, 0, 1]
```

$|0{,}1 - fl(0{,}1)| \approx 1{,}5e^{-9}$

b)

```
[0, 1, 0, 0, 0, 0, 0, 1,
 0, 0, 1, 0, 0, 0, 0, 1,
 1, 0, 0, 1, 1, 0, 0, 1,
 1, 0, 0, 1, 1, 0, 1, 0]
```

$|10{,}1 - fl(10{,}1)| \approx 3{,}8e^{-7}$

c)

```
[0, 1, 0, 0, 0, 0, 1, 0,
 1, 1, 0, 0, 1, 0, 0, 0,
 0, 0, 1, 1, 0, 0, 1, 1,
 0, 0, 1, 1, 0, 0, 1, 1]
```

$|100{,}1 - fl(100{,}1)| \approx 1{,}5e^{-6}$

Exercício 1.3.1. a) $3{,}14159\times10^{0}$, `3.14159e00`+; b) $3{,}14159\times10^{-1}$, `3.14159e-01`; c) $8{,}164922 \times 10^{-1}$, `8.164966e-01`

Exercício 1.3.2. $3{,}540841 \times 10^{-1}$

Exercício 1.3.3. a) $6{,}2 \times 10^{-1}$, `6.2e-01`; b) $6{,}2 \times 10^{-1}$, `6.1e-01`; c) $6{,}4 \times 10^{-1}$; `6.4e-01`

Exercício 1.3.4. a) 3,5; b) 3,6; c) Operar sobre números arredondados acarreta perda de exatidão.

Exercício 1.3.5. Dica: $\sqrt{3}$ não tem representação exata em ponto flutuante.

Exercício 1.4.1.

a) $1{,}593 \times 10^{-3}$; b) $2{,}818 \times 10^{-1}$;

Exercício 1.4.2.

a) 0,051%; b) 0,01%;

Exercício 1.4.3. a) 3; b) 3

Exercício 1.4.4. $1/6! \approx 1{,}4 \times 10^{-3}$.

Exercício 1.4.5. $2{,}002083 \times 10^{-3}$

Exercício 1.4.6. $5{,}75403 \times 10^{-3}$

Exercício 1.4.7. $5{,}0 \times 10^{-10}$

Exercício 2.1.1. $9{,}179688 \times 10^{-1}$

Exercício 2.1.2. 9,15833E−1

Exercício 2.1.3. $5{,}770508 \times 10^{-1}$

Exercício 2.2.1. $9{,}158079 \times 10^{-1}$

Exercício 2.2.2. $5{,}76984 \times 10^{-1}$

Exercício 2.3.1. $\alpha = 0{,}6$; $7{,}3909 \times 10^{-1}$

Exercício 2.4.1. $9{,}15811 \times 10^{-1}$

Exercício 2.5.1. $9{,}15811 \times 10^{-1}$

Exercício 2.5.2. $5{,}7700 \times 10^{-1}$

Exercício 2.6.1. $9{,}1581 \times 10^{-1}$

Exercício 2.6.2. $5{,}7700 \times 10^{-1}$

Exercício 2.7.1. 2

Exercício 3.1.1.

$$\begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 & 0 & -3 \\ 0 & 0 & 1 & 0 & 0 & -2 \\ 0 & 0 & 0 & 1 & 0 & 2 \\ 0 & 0 & 0 & 0 & 1 & -4 \end{bmatrix}$$

Exercício 3.1.2.

$$\begin{bmatrix} 1.0000 & 0.0000 & 0.0000 & -3.9435\text{E}-1 \\ -0.0000 & 1.0000 & -0.0000 & -2.3179\text{E}-1 \\ 0.0000 & 0.0000 & 1.0000 & 1.2120\text{E}+0 \end{bmatrix}$$

Exercício 3.2.1. a) 2,2383; b) 2,0323E+1; c) 3,5128E+1

Exercício 3.3.1.

$$\begin{bmatrix} 1{,}0000 & 0{,}0000 & 0{,}0000 & 6{,}2588\text{E}-1 \\ 0{,}0000 & 1{,}0000 & 0{,}0000 & -1{,}6777\text{E}+0 \\ 0{,}0000 & 0{,}0000 & 1{,}0000 & 6{,}2589\text{E}+4 \end{bmatrix}$$

Exercício 3.4.1. $x_1 = -3$, $x_2 = -1$, $x_3 = 1$

Exercício 4.1.1. $x^{(5)} = (-1{,}00256,\ 2{,}95365,\ -1{,}95347,\ 0{,}97913)$; $\|Ax^{(5)} - b\| = 0{,}42244$

Exercício 4.1.2. $x^{(5)} = (-1{,}00423,\ 3{,}00316,\ -2{,}00401,\ 0{,}99852)$; $\|Ax^{(5)} - b\| = 0{,}025883$

Exercício 5.1.1. $x_1 = 1{,}7519\text{E}+0$, $x_2 = -2{,}6202\text{E}-1$, $x_3 = -1{,}8983\text{E}+0$

Bibliografia

[1] A. Björk. *Numerical methods for least squares problems.* SIAM, 1996.

[2] R.L. Burden, J.D. Faires, and A.M. Burden. *Análise Numérica.* CENGAGE Learning, 10. edition, 2015.

[3] E. Isaacson and H.B. Keller. *Analysis of numerical methods.* Dover, 1994.

[4] Daniel Lemire. Number parsing at a gigabyte per second. *Software: Practice and Experience*, 51(8):1700–1727, may 2021.

[5] J. Nocedal and S.J. Wright. *Numerical optimization.* Springer, 2006.

[6] W.H. Press, S.A. Teukolsky, W.T. Vetterling, and B.P. Flannery. *Numerical recipes.* Cambridge University Press, 3. edition, 2007.

[7] J. Stoer and R. Bulirsch. *Introduction to numerical analysis.* Springer-Verlag, 2. edition, 1993.

Índice