Matemática Computacional
MEBiol, MEBiom e MEFT - Aula 3

Ana Leonor Silvestre

Instituto Superior Técnico, $1^{\underline{o}}$ Semestre, 2020/2021

Sumário da Aula 4

Propagação de erros em funções. Números de condição de uma função.
Propagação de erros em algoritmos.
Condicionamento de problemas. Estabilidade algorítmica.
Exemplo computacional. Exercícios.

Plano

- Propagação de erros nas operações aritméticas elementares
- Propagação de erros em funções univariadas
- Propagação de erros em funções multivariadas
- Propagação de erros em algoritmos: pressupõe execução em sistemas de ponto flutuante $\mathbb{F}$, pelo que os erros de arredondamento em $\mathbb{F}$ serão tidos em consideração

Propagação de erros em funções

Propagação de erros nas operações aritméticas elementares

- Soma: $x + y \approx \tilde{x} + \tilde{y}$ (em $\mathbb{R}$)

$$\boxed{\delta_{\tilde{x}+\tilde{y}} = \frac{x}{x+y}\delta_{\tilde{x}} + \frac{y}{x+y}\delta_{\tilde{y}}}$$

- Subtração: $x - y \approx \tilde{x} - \tilde{y}$

$$\boxed{\delta_{\tilde{x}-\tilde{y}} = \frac{x}{x-y}\delta_{\tilde{x}} - \frac{y}{x-y}\delta_{\tilde{y}}}$$

Pode acontecer que $\delta_{\tilde{x}}$ e $\delta_{\tilde{y}}$ sejam muito pequenos e $\delta_{\tilde{x}-\tilde{y}}$ seja muito grande, concretamente quando x e y são números positivos muito próximos. À perda de precisão daí resultante dá-se o nome de Cancelamento subtrativo.

Exemplo: Cancelamento subtrativo

$$f(x) := \frac{1}{x} - \frac{1}{x+1} \quad (x \in \mathbb{R} \setminus \{-1, 0\})$$

O cálculo de

$$f(10^{20}) \quad (\neq 0)$$

no Matlab R2015b (IEEE-754 com precisão dupla) forneceu:

» format long

» $1/10^{20} - 1/(10^{20}+1)$
ans = 0

Este resultado ans = 0 tem erro de 100% (erro muito grande).

Esta expressão de $f(x)$ envolve a subtração de números muito próximos quando $x \gg 1$.

Propagação de erros nas operações aritméticas elementares

Supondo que $|\delta_{\tilde{x}}| \ll 1$, $|\delta_{\tilde{y}}| \ll 1$, podemos linearizar as expressões dos erros

- Multiplicação: $x \times y \approx \tilde{x} \times \tilde{y}$

$$\boxed{\delta_{\tilde{x}\times\tilde{y}} \approx \delta_{\tilde{x}} + \delta_{\tilde{y}}.}$$

- Divisão: $x/y \approx \tilde{x}/\tilde{y}$

$$\boxed{\delta_{\frac{\tilde{x}}{\tilde{y}}} \approx \delta_{\tilde{x}} - \delta_{\tilde{y}}.}$$

Propagação de erros no cálculo de funções

Seja $f \in C^2(I)$, $x, \tilde{x} \in I$. Para

$$f(x) \approx f(\tilde{x}),$$

supondo que $|\delta_{\tilde{x}}| \ll 1$,

$$\boxed{\delta_{f(\tilde{x})} \approx \frac{x f'(x)}{f(x)} \delta_{\tilde{x}}}$$

onde $p_f(x) := \dfrac{x f'(x)}{f(x)}$ se chama número de condição de f em x.

Propagação de erros no cálculo de funções

Exemplo: $f(x) := \frac{1}{x} - \frac{1}{x+1}$

$$p_f(x) = \frac{x\left(-\frac{1}{x^2} + \frac{1}{(x+1)^2}\right)}{\frac{1}{x} - \frac{1}{x+1}} = \frac{x\left(\frac{1}{x+1} - \frac{1}{x}\right)\left(\frac{1}{x+1} + \frac{1}{x}\right)}{\frac{1}{x} - \frac{1}{x+1}}$$

$$= -x\left(\frac{1}{x+1} + \frac{1}{x}\right) = -\frac{2x+1}{x+1}$$

$\lim_{x\to\infty} p_f(x) = -2 \Rightarrow p_f(x) \approx -2,$ quando x é muito grande

Propagação de erros no cálculo de funções

Seja $f : D \subseteq \mathbb{R}^n \to \mathbb{R}$. Sejam $x, \tilde{x} \in D$, tais que $x = (x_1, ..., x_n) \approx (\tilde{x}_1, ..., \tilde{x}_n)$, o que leva a tomar a aproximação

$$f(x) \approx f(\tilde{x}).$$

Supondo $f \in C^2(D)$,

$$\boxed{\delta_{f(\tilde{x})} \approx \sum_{k=1}^{n} p_{f,k}(x)\delta_{\tilde{x}_k}, \quad p_{f,k}(x) := \frac{x_k \frac{\partial f}{\partial x_k}(x)}{f(x)}}$$

Os coeficientes $p_{f,1}(x)$,..., $p_{f,n}(x)$ de ponderação dos erros relativos de $\tilde{x}$ chamam-se números de condição de f em x.

Propagação de erros no cálculo de funções

Exemplo: $f(x_1, x_2) := x_1/x_2$

$$p_{f,1}(x) = \frac{x_1 \frac{\partial f}{\partial x_1}(x)}{f(x)} = \frac{x_1 \times 1/x_2}{x_1/x_2} = 1$$

$$p_{f,2}(x) = \frac{x_2 \frac{\partial f}{\partial x_2}(x)}{f(x)} = \frac{x_2 \times (-x_1/x_2^2)}{x_1/x_2} = -1$$

$$\delta_{f(\tilde{x})} \approx p_{f,1}(x)\delta_{\tilde{x}_1} + p_{f,2}(x)\delta_{\tilde{x}_2} = \delta_{\tilde{x}_1} - \delta_{\tilde{x}_2}$$

Recupera-se a fórmula já conhecida:

$$\delta_{\tilde{x}_1/\tilde{x}_2} \approx \delta_{\tilde{x}_1} - \delta_{\tilde{x}_2}$$

Propagação de erros em algoritmos

Cálculo em sistemas de ponto flutuante

Um algoritmo é uma sequência finita de instruções bem definidas e não ambíguas, cada uma das quais pode ser executada mecanicamente num período de tempo finito com uma quantidade de esforço finita.

No contexto do Cálculo Científico,
um algoritmo é uma sequência finita de cálculos elementares (funções elementares)

Um programa corresponde a um algoritmo escrito numa linguagem de programação (linguagem que é entendida pelo computador).

Propagação de erros em algoritmos

Exemplo: $f(x) = \dfrac{1}{x} - \dfrac{1}{x+1} = \dfrac{1}{x+x^2}$

Passos do Algoritmo	Propagação de erros em cada passo
$z_1 = x^2$	$\delta_{\tilde{z}_1} \approx 2\delta_{\tilde{x}} + \delta_{arr_1}$
$z_2 = x + z_1$	$\delta_{\tilde{z}_2} = \dfrac{x}{x+z_1}\delta_{\tilde{x}} + \dfrac{z_1}{x+z_1}\delta_{\tilde{z}_1} + \delta_{arr_2}$
$z_3 = 1/z_2$	$\delta_{\tilde{z}_3} \approx -\delta_{\tilde{z}_2} + \delta_{arr_3}$

$$
\begin{aligned}
\delta_{f_{\mathbb{F}}(\tilde{x})} = \delta_{\tilde{z}_3} &\approx -\delta_{\tilde{z}_2} + \delta_{arr_3} \\
&\approx -\frac{x}{x+z_1}\delta_{\tilde{x}} - \frac{z_1}{x+z_1}\delta_{\tilde{z}_1} - \delta_{arr_2} + \delta_{arr_3} \\
&\approx -\frac{x}{x+x^2}\delta_{\tilde{x}} - \frac{x^2}{x+x^2}\left(2\delta_{\tilde{x}} + \delta_{arr_1}\right) - \delta_{arr_2} + \delta_{arr_3} \\
&\approx -\frac{2x+1}{x+1}\delta_{\tilde{x}} - \frac{x}{x+1}\delta_{arr_1} - \delta_{arr_2} + \delta_{arr_3}
\end{aligned}
$$

Propagação de erros em algoritmos

Conclusão: A implementação da expressão do algoritmo associado à expressão

$$f(x) = \frac{1}{x + x^2}$$

produz a seguinte propagação de erros:

$$\delta_{f_{\mathbb{F}}(\tilde{x})} \approx -\frac{2x+1}{x+1}\delta_{\tilde{x}} - \frac{x}{x+1}\delta_{arr_1} - \delta_{arr_2} + \delta_{arr_3}$$

ou seja,

$$\delta_{f_{\mathbb{F}}(\tilde{x})} \approx p_f(x)\delta_{\tilde{x}} - \frac{x}{x+1}\delta_{arr_1} - \delta_{arr_2} + \delta_{arr_3}, \quad |\delta_{arr_k}| \leq \epsilon_M.$$

Em geral:

$$\boxed{\delta_{f_{\mathbb{F}}(\tilde{x})} \approx \sum_{k=1}^{n} p_{f,k}(x)\delta_{\tilde{x}_k} + \sum_{k=1}^{m} q_k(x)\delta_{\mathrm{arr}_k}}$$

onde os coeficientes $q_k(x)$ dependem do algoritmo.

Condicionamento e Estabilidade

Condicionamento

A noção de condicionamento refere-se à sensibilidade de um problema matemático a pequenas variações nos seus dados.

Definição

Um problema diz-se bem condicionado se a pequenos erros relativos nos dados corresponde um pequeno erro relativo no resultado. Caso contrário, o problema diz-se mal condicionado.

Exemplo: A subtração $f(x) := x_1 - x_2$ de dois números muito próximos é um problema mal condicionado. Tem-se

$$p_{f,1}(x) = \frac{x_1}{x_1 - x_2}, \quad p_{f,2}(x) = \frac{x_2}{x_2 - x_1}$$

e

$$\lim_{x_1 - x_2 \to 0} |p_{f,i}(x)| = +\infty, \quad i = 1, 2.$$

Neste caso, pode ocorrer cancelamento subtrativo.

Condicionamento

Problema: Cálculo de

$$f(x) = \frac{1}{x} - \frac{1}{x+1}$$

quando x é muito grande.

- O problema é <u>bem condicionado</u>:

$$\delta_{f(\tilde{x})} \approx p_f(x)\delta_{\tilde{x}}$$

$$p_f(x) = \frac{xf'(x)}{f(x)} = -\frac{2x+1}{x+1}$$

$$\lim_{x\to\infty} p_f(x) = -2 \Rightarrow p_f(x) \approx -2, \text{ quando } x \text{ é muito grande}$$

Estabilidade de algoritmos

A noção de estabilidade diz respeito à sensibilidade de um algoritmo utilizado para resolver um problema no computador.

Definição

Um algoritmo diz-se numericamente estável ou computacionalmente estável se a pequenos erros relativos nos dados (input) e a pequenos valores da unidade de arredondamento do sistema de ponto flutuante corresponde um pequeno erro relativo no resultado calculado (output). Caso contrário, o algoritmo diz-se computacionalmente instável.

Da fórmula

$$\delta_{f_{\mathbb{F}}(\tilde{x})} \approx \sum_{k=1}^{n} p_{f,k}(x)\delta_{\tilde{x}_k} + \sum_{k=1}^{m} q_{a,k}(x)\delta_{\mathrm{arr}_k}$$

conclui-se que a estabilidade de um algoritmo pode ser decidida em termos dos números de condição de f em x e dos coeficientes $q_{a,k}(x)$.

Exemplo

$$f(x) := \frac{1}{x} - \frac{1}{x+1} = \frac{1}{x(x+1)} \quad (x \in \mathbb{R} \backslash \{-1, 0\})$$

No Matlab R2015b (IEEE-754 com precisão dupla):

» format long

» $1/10^{20} - 1/(10^{20} + 1)$
ans = 0

» $1/((10^{20} + 1) * 10^{20})$
ans = 9.999999999999999e-41 $\quad (\approx 10^{-40})$

O resultado ans = 0 tem erro de 100%.

Como se explica este resultado?

Análise da estabilidade dos algoritmos

- O **algoritmo 1** associado à expressão $f(x) = \frac{1}{x} - \frac{1}{x+1}$

$$z_1 = 1/x$$
$$z_2 = x + 1$$
$$z_3 = 1/z_3$$
$$z_4 = z_1 - z_3$$

produz a seguinte propagação de erros quando executado num sistema de ponto flutuante $\mathbb{F}$:

$$\delta_{f_{\mathbb{F}}(\tilde{x})} \approx p_f(x)\delta_{\tilde{x}} + (x+1)\delta_{arr_1} + x\delta_{arr_2} - x\delta_{arr_3} + \delta_{arr_4}$$

sendo $|\delta_{arr_k}| \leq \epsilon_M$

Este algoritmo é instável para x muito grande.
Ocorre cancelamento subtrativo no último passo do algoritmo.

Análise da estabilidade dos algoritmos

$$f(x) = \frac{1}{x} - \frac{1}{x+1} = \frac{1}{x(x+1)}$$

- O **algoritmo 2** associado à expressão $f(x) = \frac{1}{x(x+1)}$

$$w_1 = x + 1$$

$$w_2 = x \times w_1$$

$$w_3 = 1/w_2$$

produz a seguinte propagação de erros quando executado num sistema de ponto flutuante $\mathbb{F}$:

$$\delta_{f_{\mathbb{F}}(\tilde{x})} \approx p_f(x)\delta_{\tilde{x}} - \delta_{arr_1} - \delta_{arr_2} + \delta_{arr_3}$$

sendo $|\delta_{arr_k}| \leq \epsilon_M$

Este algoritmo é estável para x muito grande.

Em resumo:

Podem surgir maus resultados com a execução de um programa/algoritmo no computador porque

- o problema é mal condicionado para certos valores de input;
- o algoritmo implementado é numericamente instável.

O cancelamento subtrativo pode causar instabilidade numérica.