



Interpolação

Estudo do erro na interpolação

Extrapolação

Cálculo Numérico Computacional



Agenda:

1. Considerações iniciais;
2. Teorema;
3. Exemplo;
4. Extrapolação;
5. Exemplo;
6. Exercícios

Cálculo Numérico Computacional



1. Considerações iniciais

- Como já observamos, ao se aproximar um função $f(x)$ por um polinômio interpolador de grau $\leq n$, comete-se um erro, ou seja:

$$E_n(x) = f(x) - p_n(x)$$

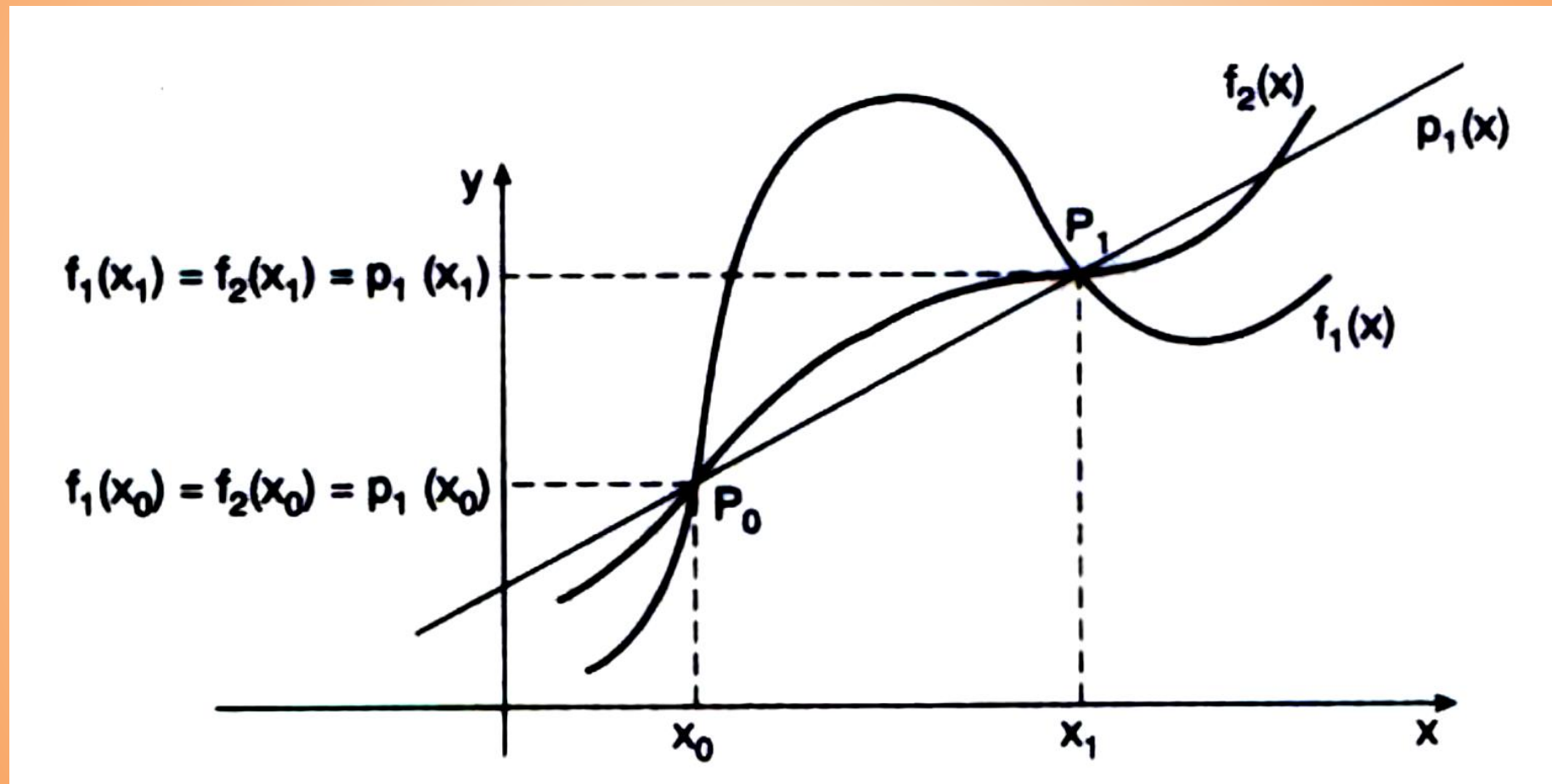
Para todo x no intervalo $[x_0, x_n]$.

- O estudo do erro é importante para sabermos quão próximo $f(x)$ está de $p_n(x)$.

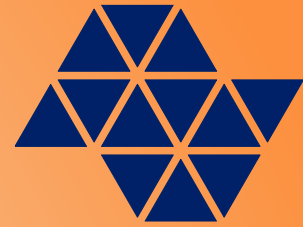
Cálculo Numérico Computacional



Exemplo: Interpolação linear



Cálculo Numérico Computacional



- O mesmo polinômio $p_1(x)$ interpola $f_1(x)$ e $f_2(x)$ em x_0 e x_1 . No entanto, o erro $E_1^1(x) = f_1(x) - p_1(x)$ é maior que $E_1^2(x) = f_2(x) - p_1(x)$, para todo $x \in (x_0, x_1)$;
- Nota-se também que o erro, neste caso, depende da concavidade das curvas, ou seja, de $f_1''(x)$ e $f_2''(x)$;
- O **teorema** a seguir mostra a expressão exata do erro quando aproximamos $f(x)$ por $p_n(x)$, para n qualquer.

Cálculo Numérico Computacional



2. Teorema

- Sejam $x_0 < x_1 < x_2 < \dots < x_n$, $(n + 1)$ pontos.

Seja $f(x)$ com derivadas até ordem $(n + 1)$ para todo x pertencente ao intervalo $[x_0, x_n]$. Seja $p_n(x)$ o polinômio interpolador de $f(x)$ nos pontos x_0, x_1, \dots, x_n .

Então, em qualquer ponto x pertencente ao intervalo $[x_0, x_n]$, o erro é dado por:

$$E_n(x) = f(x) - p_n(x)$$

$$E_n(x) = (x - x_0)(x - x_1)(x - x_2) \dots (x - x_n) \frac{f^{(n+1)}(\xi_x)}{(n + 1)!} \quad \text{onde } \xi_x \in (x_0, x_1)$$



Cálculo Numérico Computacional

- Na prática, o que se faz é uma majoração do erro, tal que:

$$|E_n(x)| = |f(x) - p_n(x)| \leq |(x - x_0)(x - x_1) \dots (x - x_n)| \frac{M_{n+1}}{(n+1)!}$$

Sendo $M_{n+1} = \max |f^{(n+1)}(x)|, x \in [x_0, x_n]$.

3. Exemplo

Seja $f(x) = e^x + x - 1$ tabelada a seguir. Obtenha $f(0,7)$ por interpolação linear e faça uma análise do erro cometido.

Cálculo Numérico Computacional



x	0	0.5	1	1.5	2.0
f(x)	0.0	1.1487	2.7183	4.9811	8.3890

- Neste caso, devemos considerar $n + 1 = 2$. Vamos escolher $x_0 = 0.5$ e $x_1 = 1$. Usando o polinômio interpolador de Newton, temos a tabela de Diferenças Divididas:

Cálculo Numérico Computacional



x	Ordem 0	Ordem 1
0.5	1.1487	
		3.1392
1.0	2.7183	

- O polinômio interpolador fica na forma:

$$p_1(x) = f[x_0] + (x - x_0) \times f[x_0, x_1]$$
$$p_1(x) = 1.1487 + (x - 0.5) \times 3.1392$$

Cálculo Numérico Computacional



- Substituindo $x = 0.7$:

$$p_1(0.7) = 1.1487 + (0.7 - 0.5) \times 3.1392 = 1.7765$$

Portanto: $p_1(0.7) = 1.7765 \approx f(0.7)$.

- **Análise do erro**
- Neste caso, $f(0.7) = e^{0.7} + 0.7 - 1 = 1.713752$
- Erro absoluto: $EA = |1.713752 - 1.7765| = 0.0628$



Cálculo Numérico Computacional

- Uma majoração para o erro é dada por:

$$|E_1(x)| \leq |(x - x_0)(x - x_1)| \frac{M_2}{2}$$

Sendo:

$$M_2 = \max |f''(x)| = e^1 = 2.7183, \quad x \in [0.5, 1]$$

Onde $f(x) = e^x + x - 1$, $f'(x) = e^x + 1$, $f''(x) = e^x$

Assim, o erro em 0.7:

$$|E(0.7)| \leq |(0.7 - 0.5)(0.7 - 1)| \times \frac{e}{2} = 0.0815$$

Cálculo Numérico Computacional



- De fato, $|E_1(0.7)| = 0.0628 < 0.0815$
- Agora, se a função $f(x)$ é dada na forma de tabela, o valor absoluto do erro $|E_n(x)|$ só pode ser estimado. Isto porque, neste caso, não é possível calcular M_{n+1} . Porém, se construirmos a tabela de diferenças divididas até ordem $n + 1$, podemos usar o maior valor (em módulo) destas diferenças como uma aproximação para $\frac{M_{n+1}}{(n+1)!}$ no intervalo $[x_0, x_n]$.

Cálculo Numérico Computacional



- Neste caso, dizemos que:

$$|E_n(x)| \approx (x - x_0)(x - x_1) \dots (x - x_n) \times (\text{máx}|\alpha|)$$

Onde $\alpha = \text{diferenças divididas de ordem } n + 1$.

- Esta aproximação tem suporte no teorema a seguir, que basicamente relaciona o operador diferenças divididas com a derivada da função $f(x)$:

$$f[x_0, x_1, \dots, x_n, x] = \frac{f^{(n+1)}(\xi_x)}{(n+1)!}, \quad x \in [x_0, x_n], \quad \xi_x \in [x_0, x_n]$$



Cálculo Numérico Computacional

- Vamos ilustrar esta aproximação com um exemplo.

Seja $f(x)$ dada na forma:

x	0.2	0.34	0.4	0.52	0.6	0.72
$f(x)$	0.16	0.22	0.27	0.29	0.32	0.37

- (a) Obtenha $f(0.47)$ usando interpolação quadrática;
(b) Forneça uma estimativa para o erro.



Cálculo Numérico Computacional

- Solução:
- Neste caso, $n + 1 = 3$. Vamos escolher os pontos:

x	0.4	0.52	0.6
$f(x)$	0.27	0.29	0.32

Usando a forma de Newton, vem:

$x_0 = 0.4$	0.27		
		0.1667	
$x_1 = 0.52$	0.29		1.0415
		0.375	
$x_2 = 0.6$	0.32		

Cálculo Numérico Computacional



- Assim:

$$p_2(x) = 0.27 + (x - 0.4) \times 0.1667 + (x - 0.4)(x - 0.52) \times 1.0415$$

- Logo:

$$p_2(0.47) = 0.2780 \approx f(0.47)$$

- Aproximação para o erro:

Consideramos aqui o erro $|E_n(x)| \approx (x - x_0)(x - x_1) \dots (x - x_n)| \times (máx|\alpha|)$, onde $\alpha = \textit{diferenças divididas de ordem } n + 1$.

Cálculo Numérico Computacional



x	Ordem 0	Ordem 1	Ordem 2	Ordem 3
0.2	0.16			
		0.4286		
0.34	0.22		2.0235	
		0.8333		-17.8963
$x_0 = 0.4$	0.27		-3.7033	
		0.1667		18.2494
$x_1 = 0.52$	0.29		1.0415	
		0.375		-2.6031
$x_2 = 0.6$	0.32		0.2085	
		0.4167		
0.72	0.37			

Cálculo Numérico Computacional



- Da tabela de Diferenças Divididas, temos que:

$$\alpha = 18.2494$$

Dessa forma:

$$|E_2(0.47)| \approx (0.47 - 0.4)(0.47 - 0.52)(0.47 - 0.6)|18.2494|$$

$$|E_2(0.47)| \approx 8.303 \times 10^{-3}$$

- Vamos estudar agora como estimar o valor da função $f(x)$ quando o x de interesse não está compreendido no intervalo $[x_0, x_n]$. Nesta situação, fazemos uma extrapolação.

Cálculo Numérico Computacional



4. Extrapolação

- Quando desejamos estimar o valor da função para $x = \bar{x}$, onde $x \in [x_0, x_n]$, estamos fazendo uma interpolação. O caso onde $x \notin [x_0, x_n]$ corresponde à extrapolação;
- Podemos usar a extrapolação com a forma de Lagrange ou Newton vistas anteriormente. Entretanto, o erro na extrapolação é maior do que o erro na interpolação. Neste caso, o Método dos Mínimos Quadrados é o mais indicado.

Cálculo Numérico Computacional



5. Exemplo

Seja $f(x) = e^x + x - 1$ tabelada abaixo. Vamos encontrar aproximações para $f(2.5)$ e $f(3)$ usando extrapolação linear.

x	0	0.5	1	1.5	2.0
f(x)	0.0	1.1487	2.7183	4.9811	8.3890

Cálculo Numérico Computacional



- Solução:

- Neste caso, $n + 1 = 2$. Vamos escolher os pontos:

x	1.5	2.0
$f(x)$	4.9811	8.3890

- Usando a forma de Newton, pela tabela de Diferenças Divididas:

x	Ordem 0	Ordem 1
1.5	4.9811	6.8158
2.0	8.3890	

Cálculo Numérico Computacional



- Assim:

$$p_1(x) = 4.9811 + (x - 1.5) \times 6.8158$$

$$p_1(2.5) = 11.7969 \approx f(2.5)$$

$$p_1(3) = 15.2048 \approx f(3)$$

- Se fizermos os cálculos pela função:

$$f(2.5) = e^{2.5} + 2.5 - 1 = 13.6825$$

$$f(3) = e^3 - 3 + 1 = 22.0855$$

- O erro cometido é dado por (absoluto):

$$E_{2.5} = |13.6825 - 11.7969| = 1.8856$$

$$E_3 = |22.0855 - 15.2048| = 6.8807$$



EXERCÍCIOS

Cálculo Numérico Computacional



1. Dada a tabela abaixo,

- a) Calcule $e^{3.1}$ usando um polinômio de interpolação sobre três pontos.
- b) Dê um limitante para o erro cometido.

x	2.4	2.6	2.8	3.0	3.2	3.4	3.6	3.8
e^x	11.02	13.46	16.44	20.08	24.53	29.96	36.59	44.70

1. a) Escolhendo os pontos $x_0 = 2.8$, $x_1 = 3.0$ e $x_2 = 3.2$, obteremos: $f(3.1) \approx 22.20375$.

b) $|E(3.1)| \leq 1.23 \times 10^{-2}$.

Cálculo Numérico Computacional



9. Construa a tabela de diferenças divididas com os dados

x	0.0	0.5	1.0	1.5	2.0	2.5
$f(x)$	-2.78	-2.241	-1.65	-0.594	1.34	4.564

- a) Estime o valor de $f(1.23)$ da melhor maneira possível, de forma que se possa estimar o erro cometido.
- b) Justifique o grau do polinômio que você escolheu para resolver o item (a).

9. Polinômio de grau 3 porque as diferenças divididas de grau 3 são aproximadamente constantes. Escolhendo $x_0 = 0.5$, $x_1 = 1.0$, $x_2 = 1.5$ e $x_3 = 2.0$ obtemos $f(1.23) \approx -1.247$ com $|E(1.23)| \approx 2.327 \times 10^{-5}$.

Cálculo Numérico Computacional



Próxima aula:

Aula 14

- Aproximação de funções pelo Método dos Mínimos Quadrados: caso discreto e caso contínuo.

