Física Computacional I
Aula de Revisão 1

Josiel Mendonça Soares de Souza

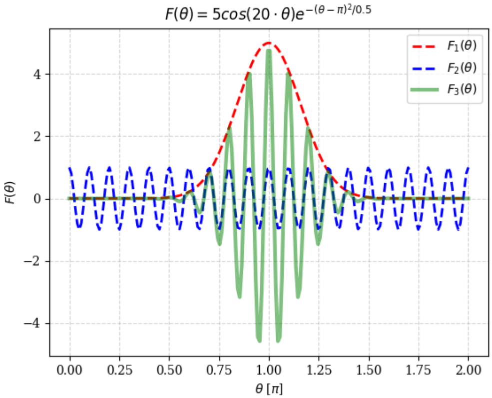
Universidade Federal do Rio Grande do Norte
Departamento de Física Teórica e Experimental

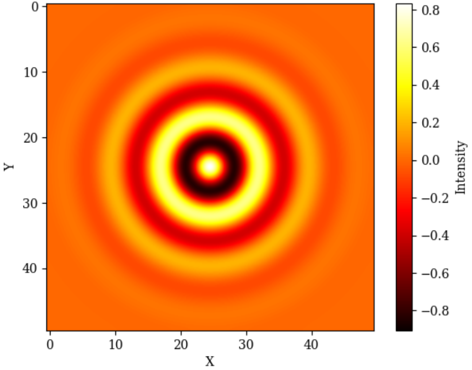
15 de Julho de 2021

Tópicos da Aula de Hoje

1. Gráficos e Visualização
2. Precisão e Velocidade
   - Valores Máximos e Mínimos para *floats*
   - Erros de Arredondamento
3. Integração Numérica
   - Método do Trapezóide
   - Regra do Trapézio Adaptativa
   - Método de Simpson

Gráficos e Visualização





Dúvidas sobre como gerar gráficos como estes acima?

Precisão e Velocidade: Valores Máximos e Mínimos

Quando tratamos como variáveis de ponto flutuante, os *floats*, devemos considerar a precisão no resultado de operações usando esses tipos de variáveis.

Valores Máximos e Mínimos

$$\begin{cases} V_{max}^{(+)} = 10^{308} & \Rightarrow \ \mathit{if}(x > V_{max}^{(+)}): \\ & \qquad x = inf \\ V_{max}^{(-)} = -10^{308} & \Rightarrow \ \mathit{if}(x < V_{max}^{(-)}): \\ & \qquad x = -inf \\ V_{min} = 10^{-308} & \Rightarrow \ \mathit{if}(-V_{min} < x < V_{min}): \\ & \qquad x = 0 \end{cases}$$

Precisão e Velocidade: Valores Máximos e Mínimos

Quando tratamos como variáveis de ponto flutuante, os *floats*, devemos considerar a precisão no resultado de operações usando esses tipos de variáveis.
Primeiramente devemos considerar os valores máximos e mínimos que o Python pode armazenar em variáveis do tipo *float*.

Valores Máximos e Mínimos

$$\begin{cases} V_{max}^{(+)} = 10^{308} & \Rightarrow \; if(x > V_{max}^{(+)}): \\ & \qquad x = inf \\ V_{max}^{(-)} = -10^{308} & \Rightarrow \; if(x < V_{max}^{(-)}): \\ & \qquad x = -inf \\ V_{min} = 10^{-308} & \Rightarrow \; if(-V_{min} < x < V_{min}): \\ & \qquad x = 0 \end{cases}$$

Precisão e Velocidade: Erros de Arredondamento

Um outro erro comum, que ocorre em resultados de operações utilizando *floats*, está associado ao arredondamento até o **16º dígito significativo** da minha variável.

Precisão e Velocidade: Erros de Arredondamento

Um outro erro comum, que ocorre em resultados de operações utilizando *floats*, está associado ao arredondamento até o **16º dígito significativo** da minha variável.

Por exemplo, suponha que o resultado exato de uma operação entre dois números seja 1.2, o resultado dessa operação no python será algo como 1.2000000000000001 ou 1.1999999999999999. Em outras palavras, o resultado obtido será do tipo 1.2 ± 10^{-16}.

Precisão e Velocidade: Erros de Arredondamento

Um outro erro comum, que ocorre em resultados de operações utilizando *floats*, está associado ao arredondamento até o **16º dígito significativo** da minha variável.

Por exemplo, suponha que o resultado exato de uma operação entre dois números seja 1.2, o resultado dessa operação no python será algo como 1.200000000000001 ou 1.1999999999999999. Em outras palavras, o resultado obtido será do tipo 1.2 ± 10^{-16}.

Assim, dizemos que o erro associado ao arredondamento de uma variável x é igual a um número aleatório (uniformemente distribuido) com desvio padrão $\sigma = Cx$, onde $C \approx 10^{-16}$.

$$\sigma_{arr} = Cx, \text{ onde } C \approx 10^{-16}$$

Precisão e Velocidade: Erros de Arredondamento

Um outro erro comum, que ocorre em resultados de operações utilizando *floats*, está associado ao arredondamento até o **16º dígito significativo** da minha variável.

Por exemplo, suponha que o resultado exato de uma operação entre dois números seja 1.2, o resultado dessa operação no python será algo como 1.20000000000001 ou 1.19999999999999. Em outras palavras, o resultado obtido será do tipo 1.2 ± 10^{-16}.

Assim, dizemos que o erro associado ao arredondamento de uma variável x é igual a um número aleatório (uniformemente distribuido) com desvio padrão $\sigma = Cx$, onde $C \approx 10^{-16}$.

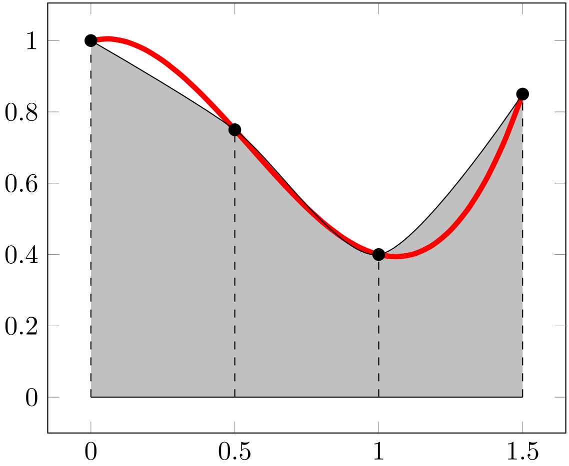
$$\sigma_{arr} = Cx, \text{ onde } C \approx 10^{-16}$$

Comando útil para se verificar o progresso do seu código ao usar o laço *for*:

```
from tqdm import trange
for i in trange(N):
```

Método do Trapezóide

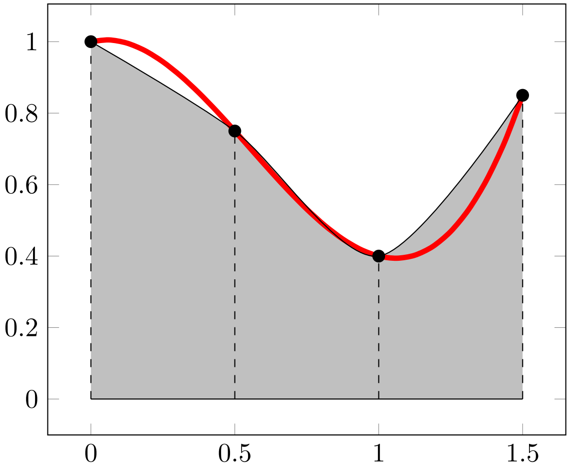
$$A_k = \frac{1}{2}(f_{k+1} + f_k)h$$



Método do Trapezóide

$$A_k = \frac{1}{2}(f_{k+1} + f_k)h$$

$$I \equiv \int_a^b f(x)dx \approx \sum_{k=0}^{N} A_k$$

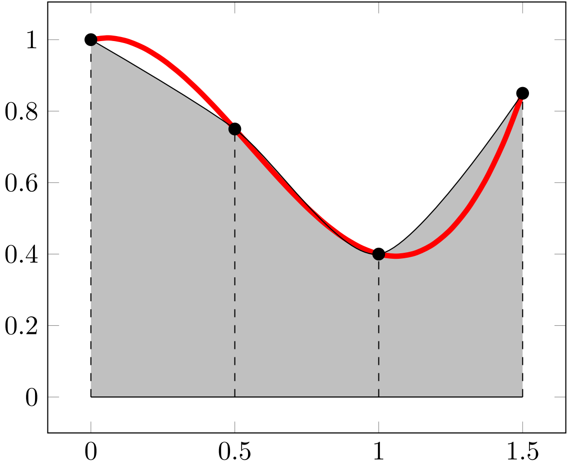


Método do Trapezóide

$$A_k = \frac{1}{2}(f_{k+1} + f_k)h$$

$$I \equiv \int_a^b f(x)dx \approx \sum_{k=0}^{N} A_k$$

$$I \approx \frac{h}{2}[(f_0 + f_1) + (f_1 + f_2) + ... + (f_{N-1} + f_N)]$$



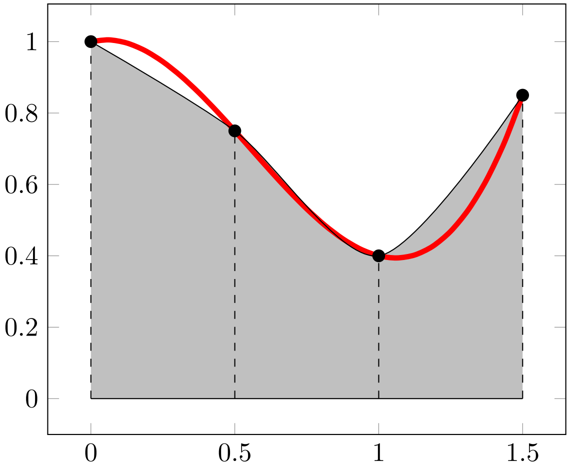
Método do Trapezóide

$$A_k = \frac{1}{2}(f_{k+1} + f_k)h$$

$$I \equiv \int_a^b f(x)dx \approx \sum_{k=0}^{N} A_k$$

$$I \approx \frac{h}{2}[(f_0 + f_1) + (f_1 + f_2) + ... + (f_{N-1} + f_N)]$$

$$I \approx \sum_{k=0}^{N} A_k = h\left[\frac{1}{2}(f_0 + f_N) + \sum_{k=1}^{N-1} f_k\right]$$



Método do Trapezóide: Incerteza

Vimos que para se determinar a ordem de imprecisão da nossa integral via método do trapezóide precisavamos verificar a integração de uma função expandida em séries de Taylor em torno de um dado número.
Expandindo $f(x)$ em torno de x_k e de x_{k-1} tinhamos:

$$\begin{aligned} f(x) &\approx f_k + f_k'(x - x_k) + \frac{1}{2} f_k''(x - x_k)^2 + ... \\ f(x) &\approx f_{k-1} + f_{k-1}'(x - x_{k-1}) + \frac{1}{2} f_{k-1}''(x - x_{k-1})^2 + ... \end{aligned}$$

onde, $f_k \equiv f(x_k)$ e $f_{k-1} \equiv f(x_{k-1})$ e suas respectivas derivadas.

Método do Trapezóide: Incerteza

Integrando $f(x)$ entre x_{k-1} e x_k, com $h \equiv x_k - x_{k-1}$, tinhamos:

$$\begin{aligned}
\int_{x_{k-1}}^{x_k} f(x)dx &\approx hf_k - \frac{h^2}{2}f'_k + \frac{h^3}{6}f''_k + O(h^4) \\
\int_{x_{k-1}}^{x_k} f(x)dx &\approx hf_{k-1} + \frac{h^2}{2}f'_{k-1} + \frac{h^3}{6}f''_{k-1} + O(h^4)
\end{aligned}$$

Método do Trapezóide: Incerteza

Integrando $f(x)$ entre x_{k-1} e x_k, com $h \equiv x_k - x_{k-1}$, tinhamos:

$$\begin{aligned} \int_{x_{k-1}}^{x_k} f(x)dx &\approx hf_k - \frac{h^2}{2}f'_k + \frac{h^3}{6}f''_k + O(h^4) \\ \int_{x_{k-1}}^{x_k} f(x)dx &\approx hf_{k-1} + \frac{h^2}{2}f'_{k-1} + \frac{h^3}{6}f''_{k-1} + O(h^4) \end{aligned}$$

Tomando a média das duas integrais:

$$\int_{x_{k-1}}^{x_k} f(x)dx \approx \left[\frac{h}{2}(f_k + f_{k-1})\right] + \frac{h^2}{4}[f'_{k-1} - f'_k] + \frac{h^3}{12}[f''_{k-1} + f''_k] + O(h^4)$$

Método do Trapezóide: Incerteza

Como estamos interessado na integral de $f(x)$ no intervalo de $x_0 = a$ e de $x_N = b$ temos:

$$\int_a^b f(x)dx = \sum_{k=1}^{N} \int_{x_{k-1}}^{x_k} f(x)dx$$

Método do Trapezóide: Incerteza

Como estamos interessado na integral de $f(x)$ no intervalo de $x_0 = a$ e de $x_N = b$ temos:

$$\int_a^b f(x)dx = \sum_{k=1}^{N} \int_{x_{k-1}}^{x_k} f(x)dx$$

Com isso temos:

$$\int_a^b f(x)dx \approx \underbrace{\left[\frac{h}{2}\sum_{k=1}^{N}(f_k + f_{k-1})\right]}_{\textbf{Regra do Trapézio}} + \underbrace{\frac{h^2}{4}\sum_{k=1}^{N}[f'_{k-1} - f'_k] + \frac{h^3}{12}\sum_{k=1}^{N}[f''_{k-1} + f''_k] + O(h^4)}_{\textbf{Termos de Correção}}$$

Método do Trapezóide: Incerteza

Como estamos interessado na integral de $f(x)$ no intervalo de $x_0 = a$ e de $x_N = b$ temos:

$$\int_a^b f(x)dx = \sum_{k=1}^{N} \int_{x_{k-1}}^{x_k} f(x)dx$$

Com isso temos:

$$\int_a^b f(x)dx \approx \underbrace{\left[\frac{h}{2}\sum_{k=1}^{N}(f_k + f_{k-1})\right]}_{\textbf{Regra do Trapézio}} + \underbrace{\frac{h^2}{4}\sum_{k=1}^{N}[f'_{k-1} - f'_k] + \frac{h^3}{12}\sum_{k=1}^{N}[f''_{k-1} + f''_k] + O(h^4)}_{\textbf{Termos de Correção}}$$

Logo nosso método o trapézio é uma **aproximação de primeira ordem em h**, da nossa integral original.

Método do Trapezóide: Incerteza

Focando agora nos 2 termos seguintes de ordem quadrática e superior em h.
Veja que quase todos os termos do segundo somatório se anulam, exceto pelos valores das derivadas de f nas extremidades:

$$\frac{h^2}{4}\sum_{k=1}^{N}[f'_{k-1} - f'_k] = \frac{h^2}{4}[(f'_0 - f'_1) + (f'_1 - f'_2) + (f'_2 - f'_3) + ... + (f'_{N-1} - f_N)] = \frac{h^2}{4}(f'_0 - f'_N)$$

Método do Trapezóide: Incerteza

Focando agora nos 2 termos seguintes de ordem quadrática e superior em h.
Veja que quase todos os termos do segundo somatório se anulam, exceto pelos valores das derivadas de f nas extremidades:

$$\frac{h^2}{4}\sum_{k=1}^{N}[f'_{k-1} - f'_k] = \frac{h^2}{4}[(f'_0 - f'_1) + (f'_1 - f'_2) + (f'_2 - f'_3) + ... + (f'_{N-1} - f_N)] = \frac{h^2}{4}(f'_0 - f'_N)$$

Já o terceiro somatório pode ser manipulado de modo semelhante ao regra do trapézio:

$$\frac{h^2}{6}\left[\frac{h}{2}\sum_{k=1}^{N}(f''_{k-1} + f''_k)\right] \approx \frac{h^2}{6}\int_a^b f''(x)dx = \frac{h^2}{6}(f'_N - f'_0)$$

Método do Trapezóide: Incerteza

Focando agora nos 2 termos seguintes de ordem quadrática e superior em h.
Veja que quase todos os termos do segundo somatório se anulam, exceto pelos valores das derivadas de f nas extremidades:

$$\frac{h^2}{4}\sum_{k=1}^{N}[f'_{k-1} - f'_k] = \frac{h^2}{4}[(f'_0 - f'_1) + (f'_1 - f'_2) + (f'_2 - f'_3) + ... + (f'_{N-1} - f_N)] = \frac{h^2}{4}(f'_0 - f'_N)$$

Já o terceiro somatório pode ser manipulado de modo semelhante ao regra do trapézio:

$$\frac{h^2}{6}\left[\frac{h}{2}\sum_{k=1}^{N}(f''_{k-1} + f''_k)\right] \approx \frac{h^2}{6}\int_a^b f''(x)dx = \frac{h^2}{6}(f'_N - f'_0)$$

Somando os dois termos em vermelho encontramos o termo de correção quadrática da nossa integral inicial:

$$\epsilon \equiv \frac{h^2}{4}\sum_{k=1}^{N}[f'_{k-1} - f'_k] + \frac{h^3}{12}\sum_{k=1}^{N}[f''_{k-1} + f''_k] \Rightarrow \qquad \underbrace{\epsilon = \frac{h^2}{12}(f'_0 - f'_N)}_{\textbf{Fórmula de Euler-Maclaurin}}$$

Método do Trapezóide: Incerteza

Desprezando os termos seguintes de ordem superior em h, temos que o erro de aproximação da nossa integral escala com h^2, ou seja, com $1/N^2$. Contudo, como já vimos, além do erro de aproximação temos também os erros de arredondamento devido a limitação do python para armazenar variáveis do tipo *float*. Sendo assim, no limite onde ambos os erros se igualam:

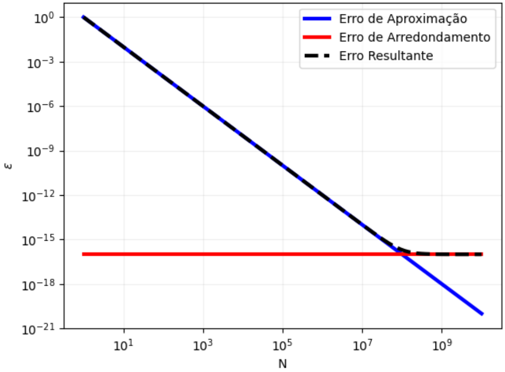
$$\epsilon = \frac{h^2}{12}(f'_0 - f'_N) \approx C \int_a^b f(x)dx$$

Método do Trapezóide: Incerteza

Desprezando os termos seguintes de ordem superior em h, temos que o erro de aproximação da nossa integral escala com h^2, ou seja, com $1/N^2$. Contudo, como já vimos, além do erro de aproximação temos também os erros de arredondamento devido a limitação do python para armazenar variáveis do tipo *float*. Sendo assim, no limite onde ambos os erros se igualam:

$$\epsilon = \frac{h^2}{12}(f_0' - f_N') \approx C \int_a^b f(x)dx$$

$$N \approx (b-a)\sqrt{\frac{f_0' - f_N'}{12C\int_a^b f(x)dx}}$$



Como $C \approx 10^{-16} \Rightarrow N \approx 100.000.000$.

Método do Trapezóide: Incerteza

Vamos relembrar também como se estimar, de modo prático, a incerteza em nossa integral aproximada.

Desprezando correções de ordens maiores do que h^2, temos que:

$$\int_a^b f(x)dx \approx I_i + \alpha h_i^2$$

Método do Trapezóide: Incerteza

Vamos relembrar também como se estimar, de modo prático, a incerteza em nossa integral aproximada.
Desprezando correções de ordens maiores do que h^2, temos que:

$$\int_a^b f(x)dx \approx I_i + \alpha h_i^2$$

Calculando de duas integrais aproximadas, o primeiro com o dobro de fatias do segundo:

$$\int_a^b f(x)dx \approx I_{2N} + \alpha h_{2N}^2 = I_N + \alpha h_N^2$$

$$I_{2N} - I_N = \alpha(h_N^2 - h_{2N}^2)$$

Método do Trapezóide: Incerteza

Vamos relembrar também como se estimar, de modo prático, a incerteza em nossa integral aproximada.
Desprezando correções de ordens maiores do que h^2, temos que:

$$\int_a^b f(x)dx \approx I_i + \alpha h_i^2$$

Calculando de duas integrais aproximadas, o primeiro com o dobro de fatias do segundo:

$$\int_a^b f(x)dx \approx I_{2N} + \alpha h_{2N}^2 = I_N + \alpha h_N^2$$

$$I_{2N} - I_N = \alpha(h_N^2 - h_{2N}^2)$$

Como, $h_N = (b-a)/N$ e $h_{2N} = (b-1)/(2N)$:

$$I_{2N} - I_N = \alpha h_{2N}^2(4-1) = 3\epsilon_{2N}$$

$$\boxed{\epsilon_{2N} = \frac{I_{2N} - I_N}{3}}$$

Problema 3 da Lista 2.

Regra do Trapézio Adaptativa

Quando queremos melhorar a precisão da nossa integral aproximada até atingir um nível de incerteza aceitavel, mas não sabemos ao certo quantas fatias será necessário para esse fim, podemos usar o método de integração adaptativa.

Regra do Trapézio Adaptativa

Quando queremos melhorar a precisão da nossa integral aproximada até atingir um nível de incerteza aceitavel, mas não sabemos ao certo quantas fatias será necessário para esse fim, podemos usar o método de integração adaptativa.

Seja I_i a nossa integral aproximada com N_i (par) fatias igualmente espaçada $h_i = (b-a)/N_i$, e I_{i-1} outra integral aproximada com metade das fatias usadas em I_i, de modo que $N_{i-1} = N_i/2$ e $h_{i-1} = 2h_i$.

Regra do Trapézio Adaptativa

Quando queremos melhorar a precisão da nossa integral aproximada até atingir um nível de incerteza aceitavel, mas não sabemos ao certo quantas fatias será necessário para esse fim, podemos usar o método de integração adaptativa.

Seja I_i a nossa integral aproximada com N_i (par) fatias igualmente espaçada $h_i = (b-a)/N_i$, e I_{i-1} outra integral aproximada com metade das fatias usadas em I_i, de modo que $N_{i-1} = N_i/2$ e $h_{i-1} = 2h_i$.

$$I_i = \frac{\delta_i}{2}[f(a) + f(b)] + \delta_i \sum_{k=1}^{N_i-1} f(a + k\delta_i)$$

Regra do Trapézio Adaptativa

Quando queremos melhorar a precisão da nossa integral aproximada até atingir um nível de incerteza aceitavel, mas não sabemos ao certo quantas fatias será necessário para esse fim, podemos usar o método de integração adaptativa.

Seja I_i a nossa integral aproximada com N_i (par) fatias igualmente espaçada $h_i = (b-a)/N_i$, e I_{i-1} outra integral aproximada com metade das fatias usadas em I_i, de modo que $N_{i-1} = N_i/2$ e $h_{i-1} = 2h_i$.

$$I_i = \frac{\delta_i}{2}[f(a) + f(b)] + \delta_i \sum_{k=1}^{N_i-1} f(a + k\delta_i)$$

$$\sum_{k=1}^{N_i-1} f(a + k\delta_i) = \underbrace{\sum_{k=1}^{N_i-1} f(a + k\delta_i)}_{k \text{ ímpar}} + \underbrace{\sum_{k=2}^{N_i-2} f(a + k\delta_i)}_{k \text{ par}}$$

Regra do Trapézio Adaptativa

$$\underbrace{\sum_{k=2}^{N_i-2} f(a+k\delta_i)}_{k\ par} = \sum_{k=1}^{N_{i-1}-1} f(a+2k\delta_i) = \sum_{k=1}^{N_{i-1}-1} f(a+k\delta_{i-1})$$

Regra do Trapézio Adaptativa

$$\underbrace{\sum_{k=2}^{N_i-2} f(a+k\delta_i)}_{k\ par} = \sum_{k=1}^{N_{i-1}-1} f(a+2k\delta_i) = \sum_{k=1}^{N_{i-1}-1} f(a+k\delta_{i-1})$$

$$I_i = \frac{\delta_i}{2}[f(a)+f(b)] + \delta_i \left[\underbrace{\sum_{k=1}^{N_i-1} f(a+k\delta_i)}_{k\ ímpar} + \sum_{k=1}^{N_{i-1}-1} f(a+k\delta_{i-1}) \right]$$

Regra do Trapézio Adaptativa

$$\underbrace{\sum_{k=2}^{N_i-2} f(a+k\delta_i)}_{k\ par} = \sum_{k=1}^{N_{i-1}-1} f(a+2k\delta_i) = \sum_{k=1}^{N_{i-1}-1} f(a+k\delta_{i-1})$$

$$I_i = \frac{\delta_i}{2}[f(a)+f(b)] + \delta_i \left[\underbrace{\sum_{k=1}^{N_i-1} f(a+k\delta_i)}_{k\ ímpar} + \sum_{k=1}^{N_{i-1}-1} f(a+k\delta_{i-1}) \right]$$

$$I_i = \frac{1}{2}\left[\frac{\delta_{i-1}}{2}[f(a)+f(b)] \sum_{k=1}^{N_{i-1}-1} f(a+k\delta_{i-1})\right] + \delta_i \underbrace{\sum_{k=1}^{N_i-1} f(a+k\delta_i)}_{k\ ímpar}$$

Regra do Trapézio Adaptativa

Com isso chegamos a nossa forma adaptativa da regra do trapézio, onde podemos aproveitar os somatórios com um dado número de fatias, para uma aproximaxão melhorarda com o dobro do número de fatias:

$$\boxed{I_i = \frac{1}{2} I_{i-1} + \delta_i \underbrace{\sum_{k=1}^{N_i - 1} f(a + k\delta_i)}_{k \text{ ímpar}}}$$

Regra do Trapézio Adaptativa

Com isso chegamos a nossa forma adaptativa da regra do trapézio, onde podemos aproveitar os somatórios com um dado número de fatias, para uma aproximaxão melhorarda com o dobro do número de fatias:

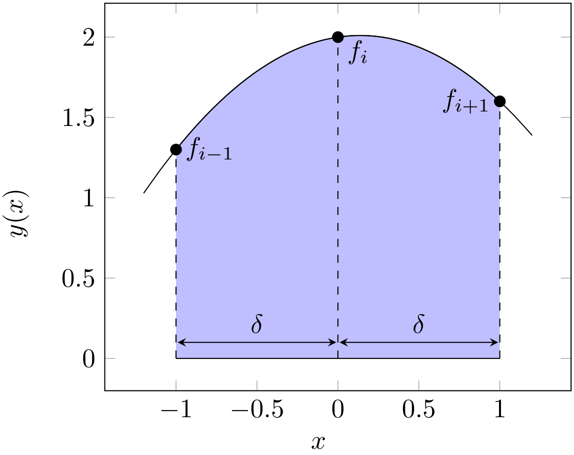
$$\boxed{I_i = \frac{1}{2} I_{i-1} + \delta_i \underbrace{\sum_{k=1}^{N_i - 1} f(a + k\delta_i)}_{k\ \acute{\imath}mpar}}$$

Problema 4 da Lista 2.

Método de Simpson

Utilizando 3 pontos de uma curva para definir um polinômio de segundo grau.

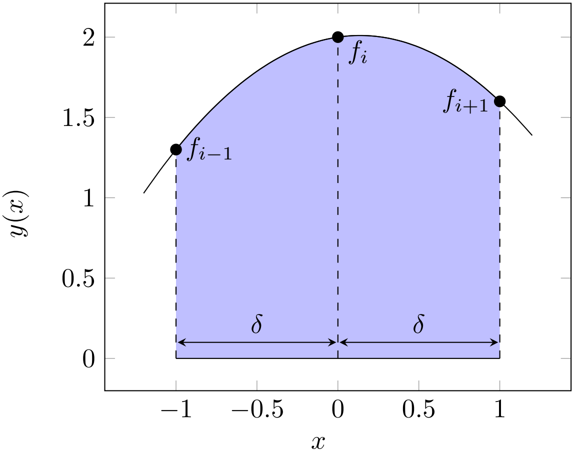
- $y(x) = Ax^2 + Bx + C$



Método de Simpson

Utilizando 3 pontos de uma curva para definir um polinômio de segundo grau.

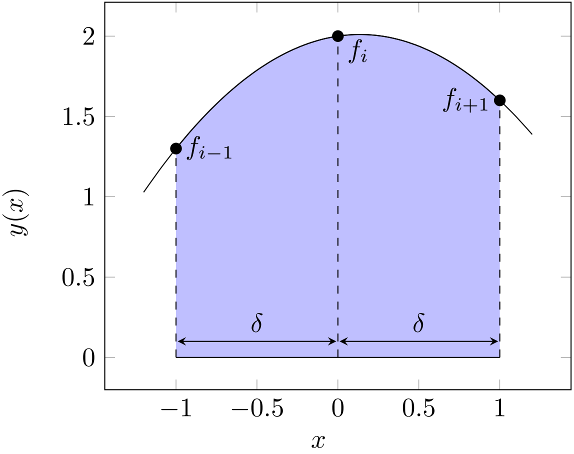
- $y(x) = Ax^2 + Bx + C$
- $y(-\delta) = f_{i-1} = A\delta^2 - B\delta + C$



Método de Simpson

Utilizando 3 pontos de uma curva para definir um polinômio de segundo grau.

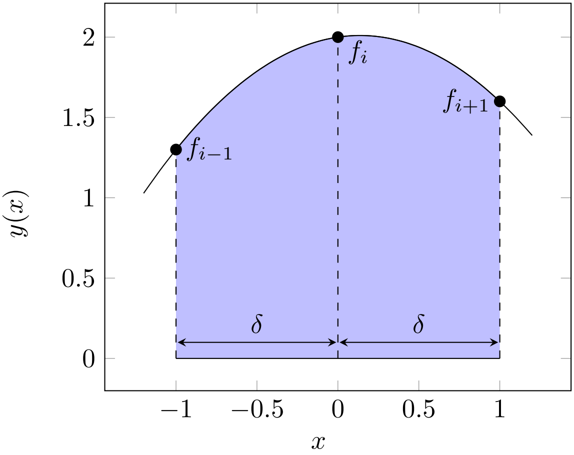
- $y(x) = Ax^2 + Bx + C$
- $y(-\delta) = f_{i-1} = A\delta^2 - B\delta + C$
- $y(0) = f_i = C$



Método de Simpson

Utilizando 3 pontos de uma curva para definir um polinômio de segundo grau.

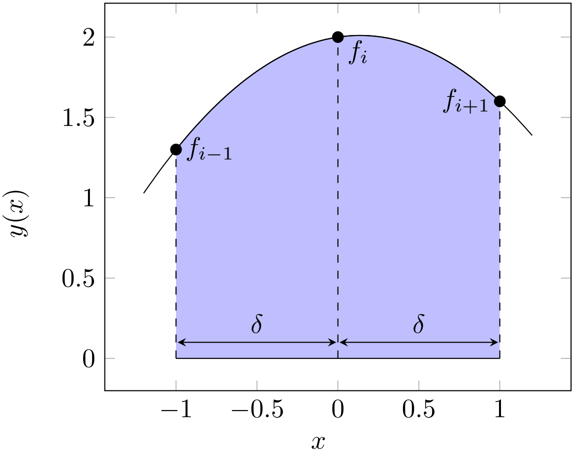
- $y(x) = Ax^2 + Bx + C$
- $y(-\delta) = f_{i-1} = A\delta^2 - B\delta + C$
- $y(0) = f_i = C$
- $y(\delta) = f_{i+1} = A\delta^2 + B\delta + C$



Método de Simpson

Utilizando 3 pontos de uma curva para definir um polinômio de segundo grau.

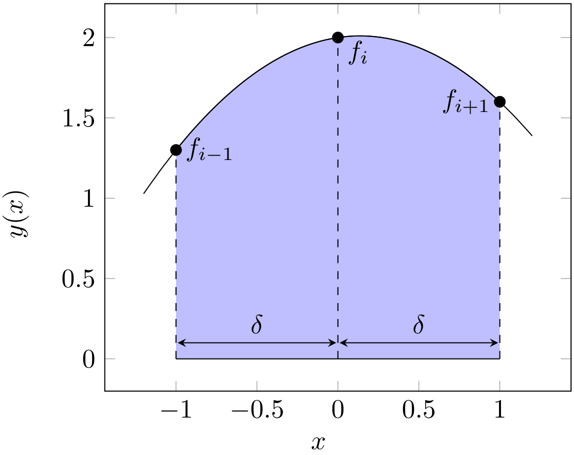
$$
\begin{aligned}
A &= \frac{1}{2\delta^2}(f_{i-1} + f_{i-1} - 2f_i) \\
B &= \frac{1}{2\delta}(f_{i+1} - f_{i-1}) \\
C &= f_i
\end{aligned}
$$



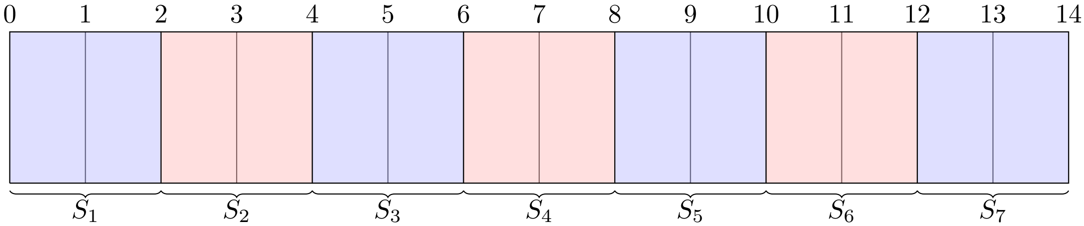
Método de Simpson

Utilizando 3 pontos de uma curva para definir um polinômio de segundo grau.

$$
\begin{aligned}
S_i &\equiv \int_{-\delta}^{\delta} y(x)dx = \frac{2}{3}A\delta^2 + 2C\delta \\
&= \frac{\delta}{3}(f_{i+1} + f_{i-1} + 4f_i)
\end{aligned}
$$

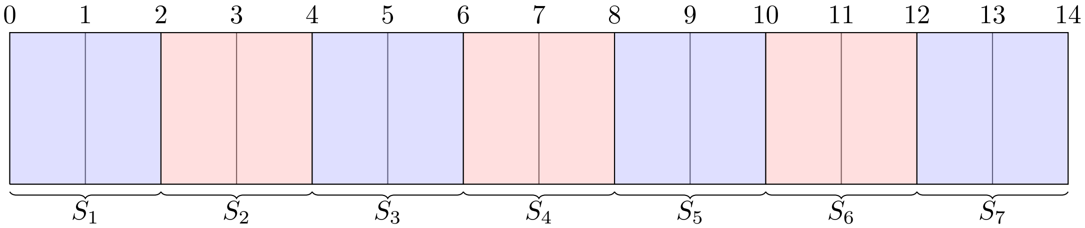


Método de Simpson



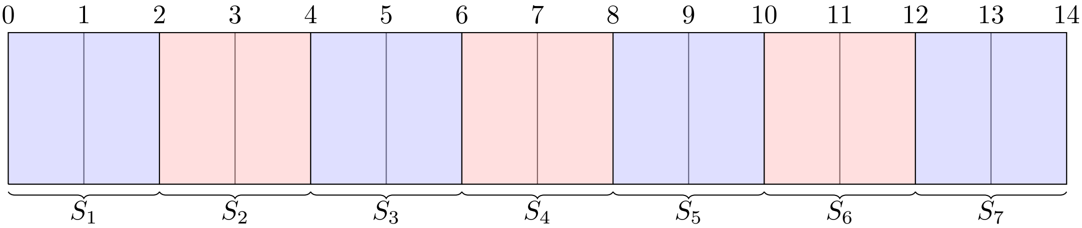
$$\int_{x_0}^{x_{14}} y(x)dx \approx \sum_{i=1}^{7} S_i$$

Método de Simpson



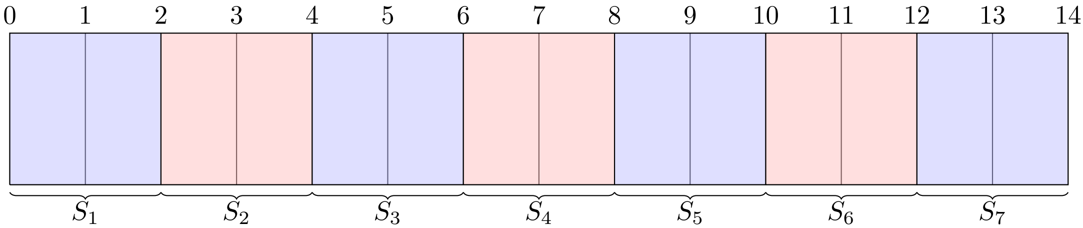
$$\begin{aligned}\int_{x_0}^{x_{14}} y(x)dx \quad &\approx \quad \frac{\delta}{3}[(f_2+f_0+4f_1)+(f_4+f_2+4f_3)+(f_6+f_4+4f_5)\\ &+(f_8+f_6+4f_5)+(f_{10}+f_8)+4f_9)+(f_{12}+f_{10}+4f_{11})+(f_{14}+f_{12}+4f_{13})]\end{aligned}$$

Método de Simpson



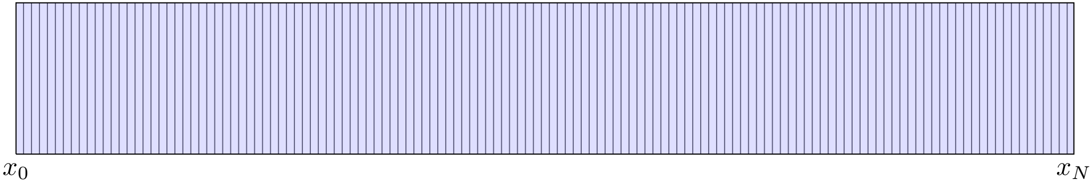
$$\int_{x_0}^{x_{14}} y(x)dx \approx \frac{\delta}{3}[(f_0 + f_{14}) + 2(f_2 + f_4 + f_6 + f_8 + f_{10} + f_{12}) + 4(f_1 + f_3 + f_5 + f_7 + f_9 + f_{11} + f_{13})]$$

Método de Simpson



$$\int_{x_0}^{x_{14}} y(x)dx \approx \frac{\delta}{3}\left\{(f_0+f_{14}) + 2\left[\sum_{i=1}^{6} f_{2i}\right] + 4\left[\sum_{i=0}^{5} f_{2i+1}\right]\right\}$$

Método de Simpson



$$\int_{x_0}^{x_N} y(x)dx \approx \frac{\delta}{3}\left\{(f_0 + f_N) + 2\left[\sum_{i=1}^{N/2-1} f_{2i}\right] + 4\left[\sum_{i=0}^{N/2-2} f_{2i+1}\right]\right\}$$