Relatório de Máquina para Derivação e Integração

Análise Matemática II - Atividade 3

Dinis Meireles de Sousa Falcão - 2020130403
David Seco Rodrigues - 2019130152

Índice

1. INTRODUÇÃO ... 3

2. Métodos Numéricos para Derivação ... 3

2.1. Fórmulas ... 3

Fórmulas de diferenças finitas em 2 pontos: ... 3

Fórmulas de diferenças finitas em 3 pontos: ... 3

2.2. Algoritmos e funções ... 3

Diferenças finitas em 2 pontos ... 3

Algoritmo das diferenças progressivas: ... 4

Diferenças regressivas em 2 pontos ... 4

Algoritmo das diferenças regressivas: ... 5

Diferenças progressivas em 3 pontos ... 5

Algoritmo das diferenças progressivas em 3 pontos: ... 6

Diferenças regressivas em 3 pontos ... 6

Algoritmo das diferenças regressivas em 3 pontos: ... 7

Diferenças centradas em 3 pontos ... 7

Algoritmo das diferenças centradas em 3 pontos: ... 8

Diferenças centradas em 3 pontos - 2.ª Derivada ... 8

3. Derivação simbólica no Matlab ... 9

3.1. DIFF ... 9

4. Métodos Numéricos para Integração ... 9

4.1. Fórmulas ... 9

Fórmula dos Trapézios: ... 9

Regra de Simpson: ... 9

4.2. Algoritmos e Funções ... 10

5. Exemplos de aplicação e teste dos métodos ... 11

5.1. Derivação numérica ... 11

5.2. Integração numérica ... 13

6. Conclusão ... 14

1. Introdução

O presente relatório corresponde à “Atividade 03 - Máquina para Derivação e Integração”, cujo objetivos são a aquisição de conhecimentos sobre derivação e integração, nomeadamente a obtenção dos valores da derivada de uma função num conjunto de pontos, conhecendo apenas o valor da função nesses pontos e ainda a sua implementação em Matlab.

Em matemática, existe uma grande família de algoritmos cujo principal objetivo é aproximar o valor de um dado integral definido duma função sem o uso duma expressão analítica para a sua primitiva.

2. Métodos Numéricos para Derivação

2.1. Fórmulas

Fórmulas de diferenças finitas em 2 pontos:

Progressivas: $f'(x_k) = f(x_{k+1}) - f(x_k) h$
Regressivas: $f'(x_k) = f(x_k) - f(x_{k-1}) h$

Fórmulas de diferenças finitas em 3 pontos:

Progressivas: $f'(x_k) = -3 f(x_k) + 4 f(x_{k+1}) - f(x_{k+2}) h$
Regressivas: $f'(x_k) = f(x_k) - 4 f(x_{k-1}) + 3 f(x_k) 2h$
Centradas: $f'(x_k) = f(x_{k+1}) - f(x_{k-1}) h$
2.ª Derivada: $f'(x_k) = f(x_{k+1}) - 2f(x_k) + f(x_{k-1}) h2$

2.2. Algoritmos e funções

Diferenças finitas em 2 pontos

Inputs:

I. f – Função
II. [a,b] – Intervalo de Derivação
III. h - passo de discretização
IV. y - imagens X vs Y

Outputs:

I. [x,y] - malha de pontos
II. dydx - derivada de f

Função:

```
function [x,y,dydx]=DI_DFProgressivas_2(f,a,b,h,y)
x=a:h:b;
n=length(x);
if nargin==4
    y=f(x);
end
dydx=zeros(1,n);
for i=1:n-1
    dydx(i)=(y(i+1)-y(i))/h;
end
dydx(n)=(y(n)-y(n-1))/h;
```

Algoritmo das diferenças progressivas:

- x - vetor de a até b
- n - tamanho de x

Se num de inputs = 4

- y - f(x)

dydx - vetor de zeros de 1 até n

Para i de 1 até n-2

- dydx(i) - (-3*y(i)+4*y(i+1)-y(i+2))/(2*h)
- dydx(n) - (y(n)-y(n-1))/h

Diferenças regressivas em 2 pontos

Inputs:

I. f - Função
II. [a,b] – Intervalo de Derivação
III. h - passo de discretização
IV. y - imagens X vs Y

Outputs:

I. [x,y] - malha de pontos
II. dydx - derivada de f

Função:

```
function [x,y,dydx]=DI_DFRegressivas_2(f,a,b,h,y)
x=a:h:b;
n=length(x);
if nargin==4
    y=f(x);
end
dydx=zeros(1,n);
dydx(1)=(y(2)-y(1))/h;
for i=2:n
    dydx(i)=(y(i)-y(i-1))/h;
end
```

Algoritmo das diferenças regressivas:

- ➢ x - vetor de a até b
- ➢ n - tamanho de x

Se num de inputs = 4

- ➢ y - f(x)

dydx - vetor de zeros de 1 até n
dydx(1) - (y(2)-y(1))/h

Para i de 1 até n

- ➢ dydx(i) - (y(i)-y(i-1))/h
- ➢ dydx(n) - (y(n)-y(n-1))/h

Diferenças progressivas em 3 pontos

Inputs:

I. f - Função
II. [a,b] – Intervalo de Derivação
III. h - passo de discretização
IV. y - imagens X vs Y

Outputs:

I. [x,y] - malha de pontos
II. dydx - derivada de f

Função:

```
function [x,y,dydx]=DI_DFProgressivas_3(f,a,b,h,y)
x=a:h:b;
n=length(x);
if nargin==4
    y=f(x);
end
dydx=zeros(1,n);
for i=1:n-2
    dydx(i)=( (-3)*y(i) + 4*y(i+1) - y(i+2) ) / (2*h);
end
dydx(n-1)=( y(n-3) - 4*y(n-2) + 3*y(n-1) )/(2*h);
dydx(n)=( y(n-2) - 4*y(n-1) + 3*y(n) )/(2*h);
```

Algoritmo das diferenças progressivas em 3 pontos:

- x - vetor de a até b
- n - tamanho de x

Se num de inputs = 4

- y - f(x)

dydx - vetor de zeros de 1 até n

Para i de 1 até n-2

- dydx(i) - (-3*y(i)+4*y(i+1)-y(i+2))/(2*h)
- dydx(n-1) - (-3*y(n-2)+4*y(n-1)-y(n))/(2*h)
- dydx(n) - (-3*y(n-2)+4*y(n-1)-y(n))/(2*h)

Diferenças regressivas em 3 pontos

Inputs:

I. f - Função
II. [a,b] – Intervalo de Derivação
III. h - passo de discretização
IV. y - imagens X vs Y

Outputs:

I. [x,y] - malha de pontos
II. dydx - derivada de f

Função:

```
function [x,y,dydx]=DI_DFRegressivas_3(f,a,b,h,y)
x=a:h:b;
n=length(x);
if nargin==4
    y=f(x);
end
dydx=zeros(1,n);
dydx(1)=( (-3)*y(1) + 4*y(2) - y(3) )/(2*h);
dydx(2)=( (-3)*y(2) + 4*y(3) - y(4) )/(2*h);
for i=3:n
    dydx(i)=( y(i-2) - 4*y(i-1) + 3*y(i) )/(2*h);
end
```

Algoritmo das diferenças regressivas em 3 pontos:

- x - vetor de a até b
- n - tamanho de x

Se num de inputs = 4

- y - f(x)

dydx - vetor de zeros de 1 até n

Para i de 1 até n-2

- dydx(i) - (-3*y(i)+4*y(i+1)-y(i+2))/(2*h)
- dydx(n-1) - (-3*y(n-2)+4*y(n-1)-y(n))/(2*h)
- dydx(n) - (-3*y(n-2)+4*y(n-1)-y(n))/(2*h)

Diferenças centradas em 3 pontos

Inputs:

I. f - Função
II. [a,b] – Intervalo de Derivação
III. h - passo de discretização
IV. y - imagens X vs Y

Outputs:

I. [x,y] - malha de pontos
II. dydx - derivada de f

Função:

```
function [x,y,dydx]=DI_DFCentradas_3(f,a,b,h,y)
x=a:h:b;
n=length(x);
if nargin==4
    y=f(x);
end
dydx=zeros(1,n);
dydx(1)=(y(2)-y(1))/h;
for i=2:n-1
    dydx(i)=(y(i+1)-y(i-1))/(2*h);
end
dydx(n)=(y(n)-y(n-1))/h;
```

Algoritmo das diferenças centradas em 3 pontos:

- x - vetor de a até b
- n - tamanho de x

Se num de inputs = 4

- y - f(x)

dydx - vetor de zeros de 1 até n
dydx(1) - (y(1)-4*y(2)+3*y(3))/(2*h)
dydx(2) - (y(1)-4*y(2)+3*y(3))/(2*h)

Para i de 3 até n-2

- dydx(i) - (y(i-2)-4*y(i-1)+3*y(i))/(2*h)

Diferenças centradas em 3 pontos - 2.ª Derivada

Inputs:

I. f - Função
II. [a,b] – Intervalo de Derivação
III. h - passo de discretização
IV. y - imagens X vs Y

Outputs:

I. [x,y] - malha de pontos
II. dydx - derivada de f

Função:

```
function [x,y,dydx]=DI_DFDerivada2(f,a,b,h,y)
x=a:h:b;
n=length(x);
if nargin==4
    y=f(x);
end
dydx=zeros(1,n);
dydx(1)=(y(3)-2*y(2)+y(1))/(h^2);
for i=2:n-1
    dydx(i)=(y(i+1)-2*y(i)+y(i-1))/(h^2);
end
dydx(n)=(y(n)-2*y(n-1)+y(n-2))/(h^2);
```

3. Derivação simbólica no MatLab

3.1. DIFF

O comando diff(I) em derivação simbólica é utilizado na GUI principal, serve para calcular a derivada de uma função predefinida ou introduzida pelo utilizador, que será mostrada na GUI.

4. Métodos Numéricos para Integração

4.1. Fórmulas

Fórmula dos Trapézios:

$$I_T(f) = \frac{h}{2}[f(x_0) + 2f(x_1) + \cdots + 2f(x_{n-1}) + f(x_n)]$$
$$|E_T| \leq \frac{b-a}{12} h^2 M_2, \quad M_2 = \max_{x\in[a,b]} |f''(x)|$$

Regra de Simpson:

$$I_S(f) = \frac{h}{3}[f(x_0) + 4f(x_1) + 2f(x_2) + \cdots + 2f(x_{n-2}) + 4f(x_{n-1}) + f(x_n)]$$
$$|E_S| \leq \frac{b-a}{180} h^4 M_4, \quad M_4 = \max_{x\in[a,b]} |f^{(4)}(x)|$$

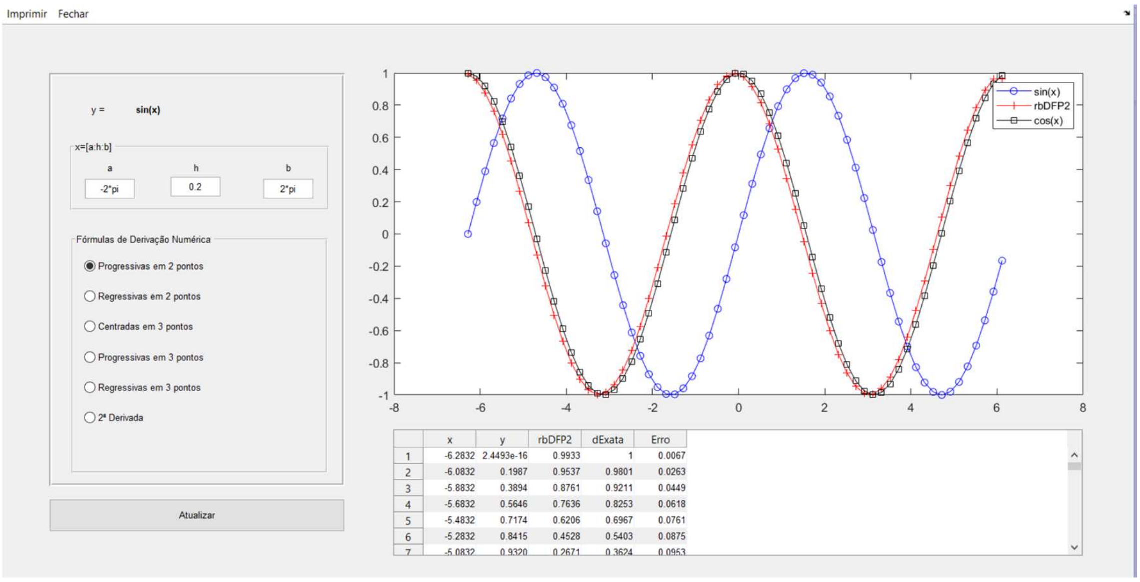
4.2. Algoritmos e Funções

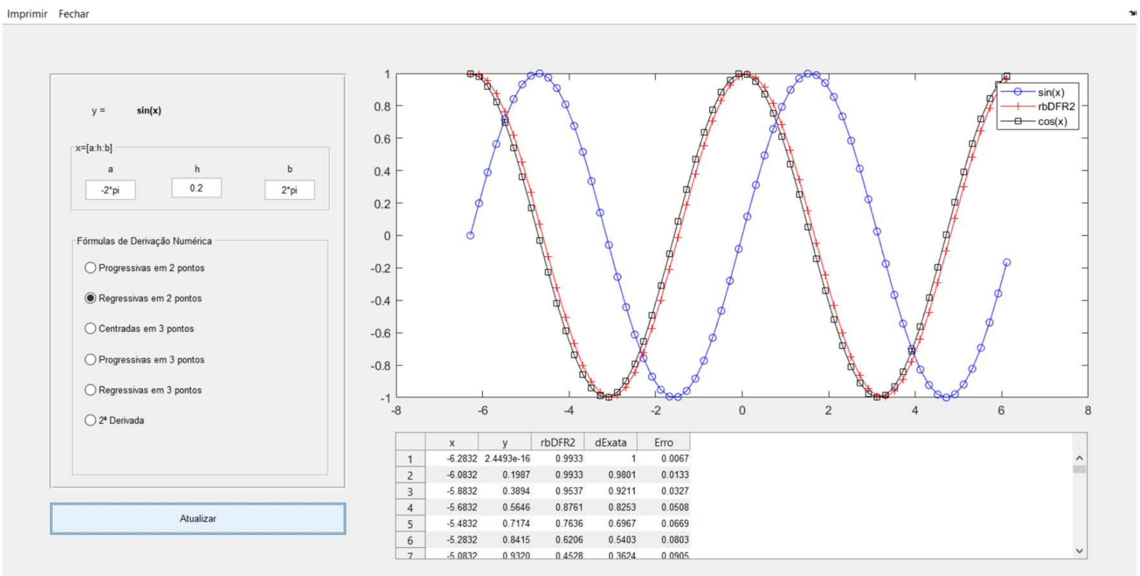
```
function t = DI_RTrapezios(f,a,b,n)

h = (b-a)/n;
t = 0;
x=a;

for i=1:n-1
    x=x+h;
    t = t+2*f(x);
end;

t = h*(f(a)+f(b)+t)/2;
```

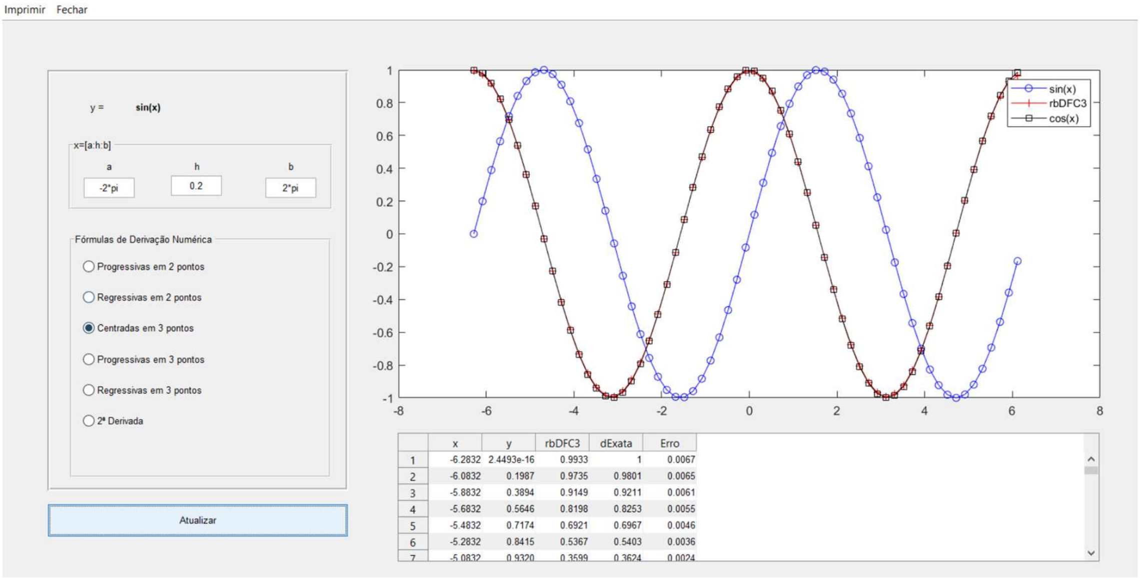
```
function s = DI_RSimpson(f,a,b,n)

h = (b-a)/n;
s = 0;
x=a;

for i=1:n-1
    x=x+h;
    if mod(i,2)==0
        s = s+2*f(x);
    else
        s = s+4*f(x);
    end;
end;

s = h*(f(a)+f(b)+s)/3;
```

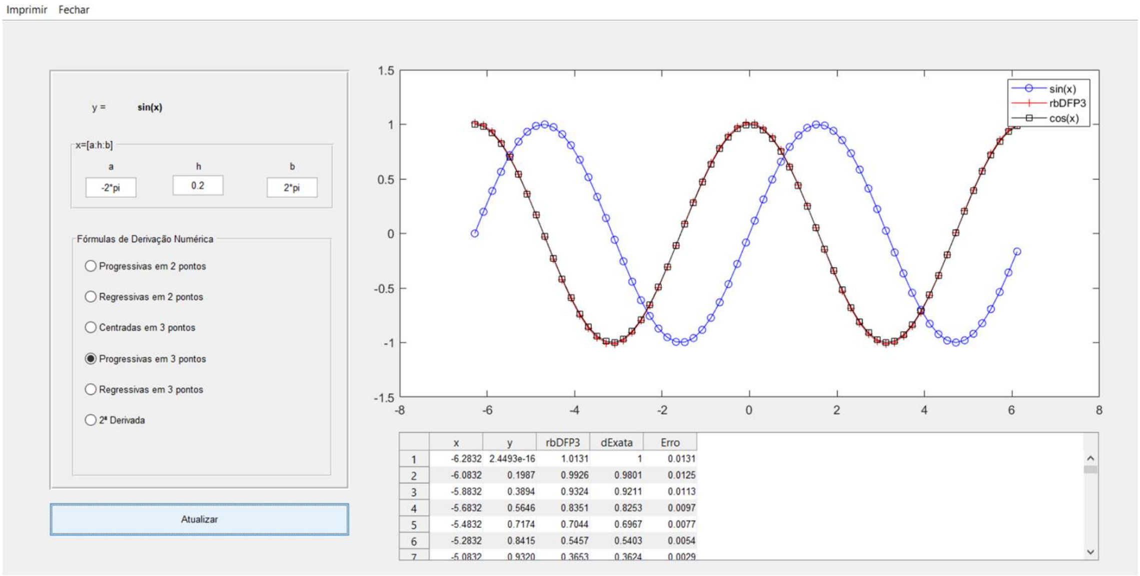
5. Exemplos de aplicação e teste dos métodos

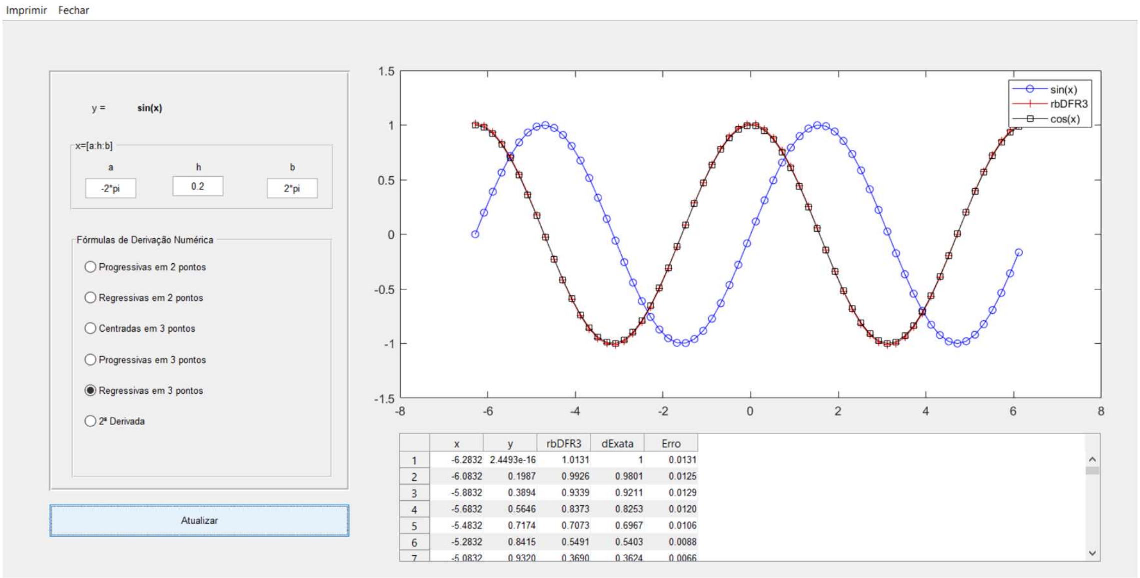
5.1 Derivação numérica



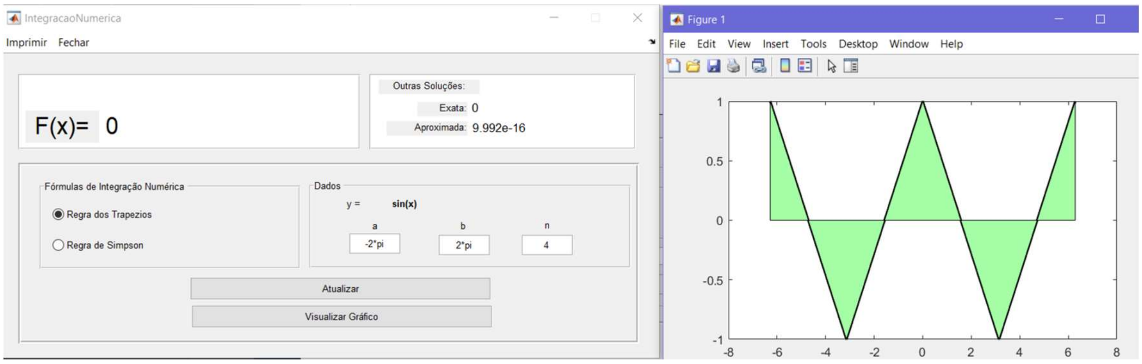


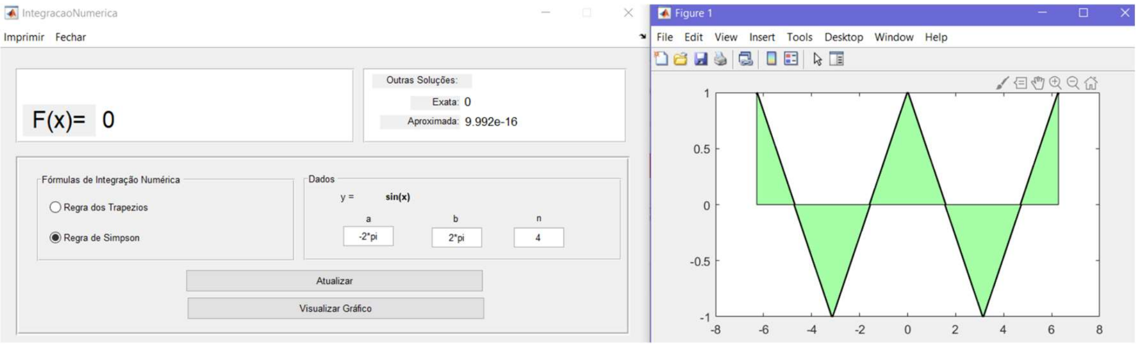

Imprimir Fechar
y = sin(x)
x=[a:h:b]
a h b
-2*pi 0.2 2*pi
Fórmulas de Derivação Numérica
Progressivas em 2 pontos
Regressivas em 2 pontos
Centradas em 3 pontos
Progressivas em 3 pontos
Regressivas em 3 pontos
2ª Derivada
Atualizar
sin(x)
rbDFC3
cos(x)
x y rbDFC3 dExata Erro
1 -6.2832 2.4493e-16 0.9933 1 0.0067
2 -6.0832 0.1987 0.9735 0.9801 0.0065
3 -5.8832 0.3894 0.9149 0.9211 0.0061
4 -5.6832 0.5646 0.8198 0.8253 0.0055
5 -5.4832 0.7174 0.6921 0.6967 0.0046
6 -5.2832 0.8415 0.5367 0.5403 0.0036


Imprimir Fechar
y = sin(x)
x=[a:h:b]
a h b
-2*pi 0.2 2*pi
Fórmulas de Derivação Numérica
Progressivas em 2 pontos
Regressivas em 2 pontos
Centradas em 3 pontos
Progressivas em 3 pontos
Regressivas em 3 pontos
2ª Derivada
Atualizar
sin(x)
rbDFP3
cos(x)
x y rbDFP3 dExata Erro
1 -6.2832 2.4493e-16 1.0131 1 0.0131
2 -6.0832 0.1987 0.9926 0.9801 0.0125
3 -5.8832 0.3894 0.9324 0.9211 0.0113
4 -5.6832 0.5646 0.8351 0.8253 0.0097
5 -5.4832 0.7174 0.7044 0.6967 0.0077
6 -5.2832 0.8415 0.5457 0.5403 0.0054


Imprimir Fechar
y = sin(x)
x=[a:h:b]
a h b
-2*pi 0.2 2*pi
Fórmulas de Derivação Numérica
Progressivas em 2 pontos
Regressivas em 2 pontos
Centradas em 3 pontos
Progressivas em 3 pontos
Regressivas em 3 pontos
2ª Derivada
Atualizar
sin(x)
rbDFR3
cos(x)
x y rbDFR3 dExata Erro
1 -6.2832 2.4493e-16 1.0131 1 0.0131
2 -6.0832 0.1987 0.9926 0.9801 0.0125
3 -5.8832 0.3894 0.9339 0.9211 0.0129
4 -5.6832 0.5646 0.8373 0.8253 0.0120
5 -5.4832 0.7174 0.7073 0.6967 0.0106
6 -5.2832 0.8415 0.5491 0.5403 0.0088

5.2 Integração numérica





6. Conclusão

Após a realização desta atividade, conclui-se que o método mais preciso é o das diferenças centradas utilizando três pontos.

Relativamente à derivação numérica, é possível notar que, os métodos que utilizam três pontos são mais precisos do que os de dois pontos.

Quanto aos métodos de integração, tendo em conta os resultados obtidos, conclui-se que os métodos de Simpson e dos Retângulos são os mais precisos, ou seja, apresentam um erro menor quando comparados com o método dos Trapézios.