

## **Cálculo diferencial e integral evoluindo a inteligência computacional mediante a linguagem Python de programação**

### **Differential and integral calculus evolving computational intelligence through the Python programming language**

DOI:10.34117/bjdv8n9-286

Recebimento dos originais: 30/08/2022

Aceitação para publicação: 29/09/2022

#### **Rodrigo Antônio Pereira Junior**

Doutor em Ciências Florestais

Instituição: Instituto Federal de Educação, Ciência e Tecnologia do Pará (IFPA) - Campus Ananindeua

Endereço: Avenida Arterial 5 A, S/N, Icuí-Guajará, Ananindeua – Pará, Brasil

E-mail: rpereira.junior@ifpa.edu.br

#### **Denis Carlos Lima Costa**

Doutor em Sistemas de Energia

Instituição: Instituto Federal de Educação, Ciência e Tecnologia do Pará (IFPA) - Campus Ananindeua

Endereço: Avenida Arterial 5 A, S/N, Icuí-Guajará, Ananindeua – Pará, Brasil

E-mail: denis.costa@ifpa.edu.br

#### **Erick Freitas da Costa**

Graduando em Ciência e Tecnologia

Instituição: Instituto Federal de Educação, Ciência e Tecnologia do Pará (IFPA) - Campus Ananindeua

Endereço: Avenida Arterial 5 A, S/N, Icuí-Guajará, Ananindeua – Pará, Brasil

E-mail: efreitas256@gmail.com

#### **Adriane Cristina Fernandes Reis**

Graduanda em Ciência e Tecnologia

Instituição: Instituto Federal de Educação, Ciência e Tecnologia do Pará (IFPA) - Campus Ananindeua

Endereço: Avenida Arterial 5 A, S/N, Icuí-Guajará, Ananindeua – Pará, Brasil

E-mail: adriane.fr22@gmail.com

#### **Bruno Henrique Parente de Carvalho**

Graduando em Ciência da Computação

Instituição: Instituto Federal de Educação, Ciência e Tecnologia do Pará (IFPA) - Campus Ananindeua

Endereço: Avenida Arterial 5 A, S/N, Icuí-Guajará, Ananindeua – Pará, Brasil

E-mail: brunoparente22@gmail.com

**Pedro Dimas da Cunha Lima**

Graduando em Ciência da Computação

Instituição: Instituto Federal de Educação, Ciência e Tecnologia do Pará  
(IFPA) - Campus AnanindeuaEndereço: Avenida Arterial 5 A, S/N, Icuí-Guajará, Ananindeua – Pará, Brasil  
E-mail: cunhapedro665@gmail.com**Mara Líbia Viana de Lima**

Mestra em Recursos Hídricos e Saneamento Ambiental

Instituição: Instituto Federal de Educação, Ciência e Tecnologia do Pará  
(IFPA) - Campus AnanindeuaEndereço: Avenida Arterial 5 A, S/N, Icuí-Guajará, Ananindeua – Pará, Brasil  
E-mail: mara.lima@ifpa.edu.br**Edson Costa Cruz**

Mestre em Geofísica

Instituição: Instituto Federal de Educação, Ciência e Tecnologia do Pará  
(IFPA) - Campus CametáEndereço: Avenida Dep. Euclides Figueiredo, Marambaia, Cametá - Pará, Brasil  
E-mail: edson.cruz@ifpa.edu.br**RESUMO**

A proeminente relevância do Cálculo Diferencial e Integral, na Ciência e na Tecnologia, foram delineadas neste artigo. Explanados computacionalmente, Limites, de Derivadas e de Integrais, foram desenvolvidos em PYTHON, criando uma interface que favorece o processo de ensino-aprendizagem, com característica multidisciplinar. Três estudos estão dispostos no artigo, comprovando a aplicabilidade da estratégia em resultados, analíticos e gráficos, com altíssimo grau de precisão e acurácia.

**Palavras-chave:** matemática, ciência, tecnologia, cálculo, computação.

**ABSTRACT**

The prominent relevance of Differential and Integral Calculus in Science and Technology were outlined in this article. Computational explanations, Limits, Derivatives and Integrals were developed in PYTHON, creating an interface that favors the teaching-learning process, with a multidisciplinary characteristic. Three studies are available in the article, proving the applicability of the strategy in results, both analytical and graphic, with a very high degree of precision and accuracy.

**Keywords:** mathematics, science, technology, calculation, computing.

**1 INTRODUÇÃO**

Gottfried Wilhelm Leibniz (1646 - 1716) e Isaac Newton (1643 - 1727), desenvolveram, paralelamente, um dos mais belos e importantes métodos destinados à interpretação da Natureza: O Cálculo Diferencial e Integral.

Essa metodologia permite uma ampliação da percepção do comportamento de um dado fenômeno, descrito por funções matemáticas (Fulini, 2016).

Para Eves (2012), O Cálculo Diferencial e Integral adicionado à Geometria Analítica, formam uma das principais descobertas do Século XVII.

Os conceitos e aplicações de Limite, de Derivada e de Integral transcenderam os séculos. Suas aplicações estão cada vez mais presentes em estudos de inúmeras áreas da Ciência e suas Tecnologias. Assim sendo, a resolução algébrica dessa, magnífica, análise infinitesimal, necessita de maior dinâmica. Logo, um mecanismo capaz de resolver cálculos, cada vez mais complexos e com alta precisão e acurácia, como a Computação, é imprescindível.

Batias et al (2021), expõem resultados de um estudo histórico-epistemológico realizado para identificar critérios, a fim de projetar tarefas que promovam a compreensão da noção de Limite em uma função de variável Real. Os critérios pelos autores permitem destacar não só a complexidade matemática subjacente ao estudo do Limite de uma função variável Real, mas também a riqueza de significados que podem ser desenvolvidos para ajudar a compreender esta noção.

Burgos et al (2021), destacam que apreender os conceitos e procedimentos do Cálculo Diferencial, em particular o conceito de Integral, é um desafio para professores e alunos em suas trajetórias acadêmicas. Desenvolver um plano para melhorar os processos instrucionais, requer peculiar atenção à complexidade das características matemáticas da Integral e sua aplicação na interpretação de fenômenos da Natureza.

Verón e Giacomone (2021), discutem diversos significados do conceito de Cálculo Diferencial e aplicam ferramentas e teorias Ontosemióticas, enfatizando os princípios pragmáticos apresentados por Leibniz, Cauchy, Fréchet e Robinson.

Carvalho et al (2021), indagam sobre as aplicações do Cálculo Diferencial e Integral nas Ciências e Engenharias.

Neste artigo, incrementa-se ao Cálculo Diferencial e Integral, a Modelagem Computacional. Para isso, emprega-se a Linguagem PYTHON de Programação, pois, segundo Costa et al (2019), essa linguagem disponibiliza todos os requisitos preconizados pelo Cálculo.

O trabalho compôs uma interface computacional proficiente às operações com Limites, com Derivadas e com Integrais, assim como à resolução e representação gráfica de funções com uma variável independente. São apresentados três estudos de casos que envolvem a multidisciplinaridade.

Essa pesquisa ascendeu o desenvolvimento de uma série de componentes curriculares no Bacharelado em Ciência e Tecnologia do Instituto Federal de Educação, Ciência e Tecnologia do Pará – IFPA *Campus* Ananindeua, denominadas Cálculo Diferencial e Computacional I, II, III e IV. Nessas componentes, utiliza-se, além da Linguagem PYTHON, as Linguagem R e MATLAB.

## 2 METODOLOGIA

A metodologia aplicada neste artigo está fundamentada nos seguintes trabalhos: Costa et al (2019), apresentam Métodos Matemáticos desenvolvidos em ambiente computacional, utilizando para isso a linguagem PYTHON de programação.

Costa et al (2020) aplicam a Modelagem Matemática-Computacional para interpretar fenômenos relacionados à gestão ambiental.

Costa et al (2021), divulgam a Matemática Computacional aplicada à Ciência e Tecnologia, desenvolvida em linguagem MATLAB de programação (Mathworks, 2021).

Em todos esses trabalhos é sempre destacado a interdisciplinaridade como instrumento responsável pela evolução das Ciências e suas Tecnologias. E ainda, Costa et al (2019 a 2021) reconhecem que a Modelagem Matemática pode ser, amplamente, implementada por meio da Modelagem Computacional.

Diante disso, esta pesquisa parte da premissa que o ensino de Cálculo Diferencial e Integral pode ser executado com múltiplos objetivos, envolvendo sistemas com várias linguagens, a partir da aplicação das modelagens Matemática e Computacional. Essa combinação de linguagens irá proporcionar uma análise muito mais ampla e uma interpretação bem mais precisa dos fenômenos da Natureza.

A primeira fase da metodologia está vinculada com a ideia de Limites: Se os valores da função  $f(x)$  se aproximarem cada vez mais do número  $L$ , enquanto  $x$  se aproximar cada vez mais do número  $a$ , diz-se que  $L$  é o Limite de  $f(x)$ , quando  $x$  tende a  $a$ , e escreve-se, conforme a Equação (1) (Leithold, 1994).

$$\lim_{x \rightarrow a} f(x) = L \quad (1)$$

Para modelos que retratam Limites comuns e Limites especiais, como por exemplo,  $\frac{0}{0}$ ,  $\frac{k}{0} (k \neq 0)$ ,  $\frac{\infty}{\infty}$ ,  $\frac{k}{\infty} (k \neq 0)$ , entre outros, é possível aplicar o *Script* 1, apresentado nas Figuras 1 e 2, modelado para resolver as seguintes expressões:

$$\lim_{x \rightarrow 4} \frac{x^2 - 2x - 8}{x - 4} = \frac{0}{0} \quad (2)$$

$$\lim_{x \rightarrow \infty} \frac{x^2 - 2x - 8}{x - 4} = \frac{\infty}{\infty} \quad (3)$$

Figura 1. Script 1 - Expressão (2).

```

1  # -*- coding: utf-8 -*-
2  """
3  Created on Sat Jul  2 16:53:50 2022
4
5  @author:
6  GMS - Gradiente de Modelagem Matemática e Simulação Computacional
7  Cálculo de Limites
8
9  """
10 # Importando Bibliotecas
11 import sympy as sy
12 # Declarando a variável simbólica: x
13 x = sy.Symbol('x')
14 # Declarando a Função: f(x)
15 f = (x**2-2*x-8)/(x-4)
16 # Visualizando f
17 print('-----')
18 print('f =',f)
19 # Valor da Tendência: a
20 # Valor do Limite de f para x --> a: L
21 a = 4
22 L = sy.Limit(f,x,a)
23 # Visualizando L
24 print("L =",L)
25 print('-----')

```

Fonte: Autoras(es).

A solução é dada por

$$f = (x^2 - 2x - 8)/(x - 4)$$

$$L = 6$$

Figura 2. Script 1 - Expressão (3).

```

1  # -*- coding: utf-8 -*-
2  """
3  Created on Sat Jul  2 16:53:50 2022
4
5  @author:
6  GMS - Gradiente de Modelagem Matemática e Simulação Computacional
7  Cálculo de Limites
8
9  """
10 # Importando Bibliotecas
11 import sympy as sy
12 import math as mt
13 # Declarando a variável simbólica: x
14 x = sy.Symbol('x')
15 # Declarando a Função: f(x)
16 f = (x**2-2*x-8)/(x-4)
17 # Visualizando f
18 print('-----')
19 print('f =',f)
20 # Valor da Tendência: a
21 # Valor do Limite de f para x --> a: L
22 a = mt.inf
23 L = sy.Limit(f,x,a)
24 # Visualizando L
25 print("L =",L)
26 print('-----')

```

Fonte: Autoras(es).

A solução é dada por

$$f = (x ** 2 - 2 * x - 8)/(x - 4), \text{ sendo } L = \infty$$

Percebe-se que, em ambos os casos, é utilizada a biblioteca *sympy*. Essa biblioteca é utilizada para cálculos com matemática simbólica (Lutz e Ascher, 2007). Ou seja, operações com sistema de álgebra computacional completo, como os executados no Cálculo Diferencial e Integral com uma ou mais variáveis.

Na resolução da Expressão (3) foi necessário a importação de mais uma biblioteca, a *math*. Essa biblioteca disponibiliza elementos e funções matemáticas especiais, como números complexos e as operações com o infinito,  $\infty$  (Ketkar, 2017).

O *script* em PYTHON, destinado a representação gráfica de funções, pode ser visualizado na Figura 3. O código foi desenvolvido baseado na Função (4).

$$f(x) = \frac{x^2 - 2x - 8}{x - 4}, \text{ para } -2 \leq x \leq 10 \quad (4)$$

Figura 3. Script 2 para Função (4).

```

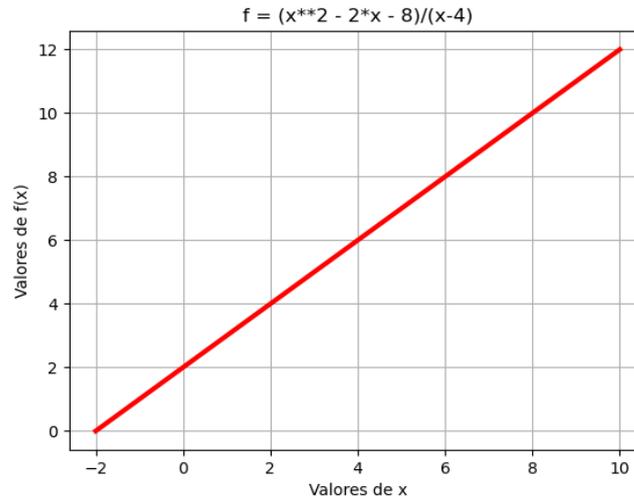
1  # -*- coding: utf-8 -*-
2  """
3  Created on Sun Jul  3 10:41:45 2022
4
5  @author:
6      UFPA - Graduação em Modelagem Matemática e Simulação Computacional
7      Representação Gráfica de funções
8  """
9  # Importando Bibliotecas
10 import numpy as np
11 import matplotlib.pyplot as plt
12 # Declarando o intervalo de análise
13 # x = np.linspace(início, fim, número de pontos)
14 x = np.linspace(-2,10,20)
15 # Declarando a Função: f(x)
16 f = (x**2-2*x-8)/(x-4)
17 # Representação Gráfica
18 plt.plot(x,f,'r',linewidth = 3)
19 plt.xlabel('Valores de x')
20 plt.ylabel('Valores de f(x)')
21 plt.title('f = (x**2 - 2*x - 8)/(x-4)')
22 plt.grid(True)
23

```

Fonte: Autoras(es)

O gráfico de (4), gerado a partir execução do *Script 2*, está exposto na Figura 4.

Figura 4. Representação gráfica da Função (4).



Fonte: Autoras(es).

A segunda fase da metodologia está vinculada com a ideia das Derivadas. Para Villegas (2014), a Derivada de uma função, definida em relação à variável Real  $x$ , da qual depende, se e somente se, o Limite existir no domínio de  $\mathfrak{R}$ , é dada pela Expressão (5).

$$\frac{df}{dx} = \lim_{h \rightarrow 0} \frac{f(x+h) - f(x)}{h} \quad (5)$$

O Script 3, evidenciado na Figura 5, apresenta a Modelagem Computacional, em linguagem PYTHON, designado às Derivadas de ordem 1 e 2, para a Função (6).

$$f(x) = x^2 + e^{4x} + \text{sen}(3x) \quad (6)$$

Figura 5. Script 3 para Função (4).

```

1  # -*- coding: utf-8 -*-
2  """
3  Created on Sat Jul  2 17:07:34 2022
4
5  @author:
6  """
7  """ - Gradiente de Modelagem Matemática e Simulação Computacional
8  Cálculo de Derivadas
9  """
10 # Importando Bibliotecas
11 import sympy as sy
12 # Declarando a variável simbólica: x
13 x = sy.Symbol( 'x' )
14 # Declarando a Função: f(x)
15 f = x**2 + sy.exp(4*x) + sy.sin(3*x)
16 # Visualizando f
17 print( 'f =', f )
18 # Derivada 1ª de f em relação a x: d1f
19 d1f = sy.diff( f, x )
20 # Visualizando d1f
21 print( 'd1f =', d1f )
22 # Derivada 2ª de f em relação a x: d2f
23 d2f = sy.diff( f, x, 2 )
24 # Visualizando d2f
25 print( 'd2f =', d2f )

```

Fonte: Autoras(es).

O conjunto de soluções é dado por

$$f = x ** 2 + \exp(4 * x) + \sin(3 * x)$$

$$d1f = 2 * x + 4 * \exp(4 * x) + 3 * \cos(3 * x)$$

$$d2f = 2 + 16 * \exp(4 * x) - 9 * \sin(3 * x)$$

No *Script 3*, linha 14, percebe-se que as Funções Exponencial e Seno precisam ser importadas da biblioteca *sympy*. Por esse motivo, elas aparecem precedidas do termo *sy*. Processo semelhante ocorre com a operações de Limites, Derivadas e Integrais.

O *Script 2* foi adaptado para representar, graficamente, as resoluções do *Script 3*, conforme mostra a Figura 6.

Figura 6. *Script 2* ajustado à Função (6) e suas derivadas.

```

1  # -*- coding: utf-8 -*-
2  """
3  Created on Sun Jul  3 10:41:45 2022
4
5  @author:
6  """
7  """ Gradiente de Modelagem Matemática e Simulação Computacional
8  Representação Gráfica de funções
9  """
10 # Importando Bibliotecas
11 import numpy as np
12 import matplotlib.pyplot as plt
13 # Declarando o intervalo de análise
14 # x = np.linspace(início, fim, número de pontos)
15 x = np.linspace(9,10,20)
16 # Declarando a Função: f(x)
17 f = x**2 + np.exp(4*x)+np.sin(3*x)
18 d1f = 2*x + 4*np.exp(4*x) + 3*np.cos(3*x)
19 d2f = 2 + 16*np.exp(4*x) - 9*np.sin(3*x)
20 # Representação Gráfica
21 plt.plot(x,f,'r',Label = 'f(x)',linewidth = 3)
22 plt.plot(x,d1f,'b',Label = 'd1f',linewidth = 3)
23 plt.plot(x,d2f,'g',Label = 'd2f',linewidth = 3)
24 plt.xlabel(' x ')
25 plt.ylabel(' f,d1f,d2f ')
26 plt.title('Derivadas de f(x)')
27 plt.legend()
28 plt.grid(True)

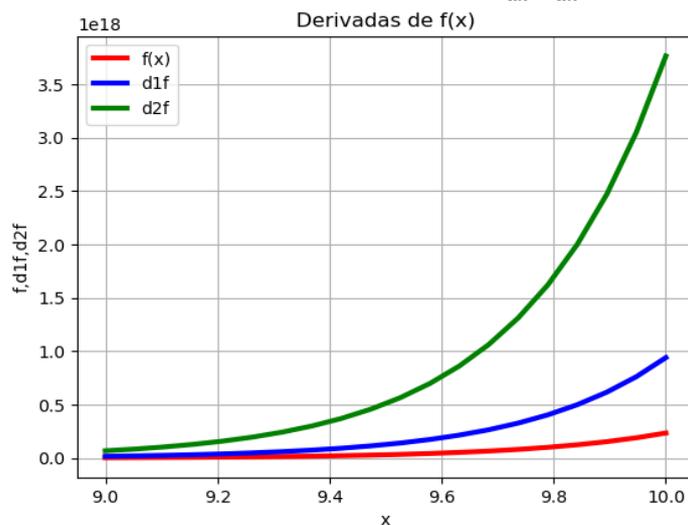
```

Fonte: Autoras(es).

Percebe-se que o *Script 2* sofreu ajustes que possibilitam a plotagem, simultânea, de várias funções. Essa produção de múltiplas curvas facilita a comparação entre a função primitiva  $f(x)$  e suas derivadas  $\frac{df}{dx}$  e  $\frac{d^2f}{dx^2}$ , no mesmo diagrama.

A representação gráfica da resolução do *Script 3*, referente a Função (6), aparece na Figura 7.

Figura 7. Comportamentos de  $f(x)$ ,  $\frac{df}{dx}$  e  $\frac{d^2f}{dx^2}$ .



Fonte: Autoras(es)

A terceira fase da metodologia está vinculada com a ideia das Integrais. De acordo com Villegas (2014), a Integral é uma antiderivada e, portanto, se uma função da forma

$$f(x) = \frac{dF(x)}{dx},$$

decorre que,

$$\int f(x) dx = \int \frac{dF(x)}{dx} dx = F(x) + C \quad (7)$$

Sendo  $F(x)$  a função primitiva de  $f(x)$ .

Generalizando, tendo como premissa o teorema fundamental do Cálculo, pode-se afirmar que, existindo uma função  $f(x)$ , integrável em um intervalo fechado  $[a, b]$ , então

$$\int_a^b f(x) dx = F(b) - F(a) \quad (8)$$

A Figura 8 mostra o *Script* 4, referente à Modelagem Computacional da Integral Indefinida de  $f(x)$ .

A função aplicada, na simulação, está representada na Expressão (9).

$$f(x) = x + \text{sen}(x) \quad (9)$$

Em seguida, na Figura 9, o *Script 4* experimentará uma adequação que permitirá efetuar a Integral Definida de  $f(x)$ , no intervalo  $[0, 2\pi]$ . Dessa forma, estará realizada a Modelagem Matemática-Computacional para as Expressões (7) e (8).

Figura 8. *Script 4a* – Integral Indefinida de  $f(x)$ .

```

1 # -*- coding: utf-8 -*-
2 """
3 Created on Sat Jul  2 17:12:08 2022
4
5 @author:
6 UNISC - Gradiente de Modelagem Matemática e Simulação Computacional
7 Cálculo de Integrais
8 """
9 # Importando Bibliotecas
10 import sympy as sy
11 # Declarando a variável simbólica: x
12 x = sy.Symbol('x')
13 # Declarando a Função: f(x)
14 f = x + sy.sin(x)
15 # Visualizando f
16 print('f =',f)
17 # Integral de f(x) em relação a x: Ifx
18 Ifx = sy.integrate(f,x)
19 # Visualizando Ifx
20 print('Ifx =',Ifx)
21

```

Fonte: Autoras(es)

A solução é dada por

$$f = x + \sin(x)$$

$$Ifx = x ** 2 / 2 - \cos(x)$$

Sendo  $Ifx$  a Integral da função  $f$ .

Figura 9. *Script 4b* – Cálculo da Integral Definida de uma função  $f(x)$ .

```

1 # -*- coding: utf-8 -*-
2 """
3 Created on Sat Jul  2 17:12:08 2022
4
5 @author:
6 UNISC - Gradiente de Modelagem Matemática e Simulação Computacional
7 Cálculo de Integrais
8 """
9 # Importando Bibliotecas
10 import sympy as sy
11 # Declarando a variável simbólica: x
12 x = sy.Symbol('x')
13 # Declarando a Função: f(x)
14 f = x + sy.sin(x)
15 # Visualizando f
16 print('f =',f)
17 # Integral de f(x) em relação a x: Ifx
18 # Limites de Integração: [a,b]
19 a = 0
20 b = 2*sy.pi
21 Ifx = sy.integrate(f,(x, a, b))
22 # Visualizando Ifx
23 print('Ifx =',Ifx)
24

```

Fonte: Autoras(es)

A solução é dada por

$$f = x + \sin(x)$$

$$Ifx = 2 * \pi ** 2$$

A representação gráfica da Função (9) e sua Integral ( $Ifx$ ), foi limitada pelo intervalo

$$-2\pi \leq x \leq 2\pi,$$

conforme mostra a Figura 10.

Figura 10. *Script 2* – ajustado para o cálculo da Integral de  $f(x)$ .

```

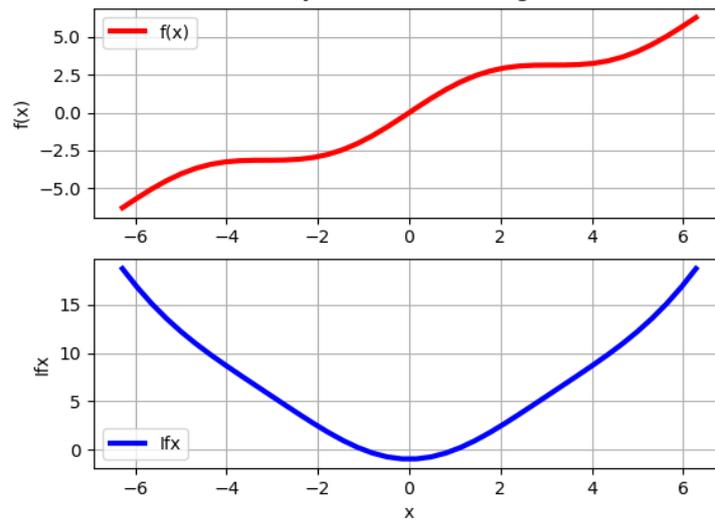
1  # -*- coding: utf-8 -*-
2  """
3  Created on Sun Jul  3 10:41:45 2022
4
5  @author:
6  GABRIEL GRADIENTE DA ROCHA MATEUS E SIMÃO COMPUTACIONAL
7  Representação Gráfica de funções
8  """
9  # Importando Bibliotecas
10 import numpy as np
11 import matplotlib.pyplot as plt
12 # Declarando o intervalo de análise
13 # x = np.linspace(início, fim, número de pontos)
14 x = np.linspace(-2*np.pi, 2*np.pi, 40)
15 # Declarando as Funções: f(x) e Ifx
16 f = x + np.sin(x)
17 Ifx = x**2/2 - np.cos(x)
18 # Representação Gráfica
19 # Gráfico da função primitiva: f
20 plt.subplot(2, 1, 1)
21 plt.plot(x, f, 'r', Label = 'f(x)', linewidth = 3)
22 plt.title('Funções Primitiva e Integral')
23 plt.ylabel('f(x)')
24 plt.legend()
25 plt.grid(True)
26 # Gráfico da Integral de f: Ifx
27 plt.subplot(2, 1, 2)
28 plt.plot(x, Ifx, 'b', Label = 'Ifx', linewidth = 3)
29 plt.xlabel('x')
30 plt.ylabel('Ifx')
31 plt.legend()
32 plt.grid(True)
33

```

Fonte: Autoras(es).

O produto da execução do *Script 2*, ajustado para a representação gráfica de  $f(x)$  e sua Integral Definida, pode ser avaliado na Figura 11.

Figura 11. Comportamentos das Funções  $f(x)$  e  $Ifx$ .  
Funções Primitiva e Integral



Fonte: Autoras(es)

### 3 RESULTADOS E DISCUSSÃO

Esta pesquisa realizou aplicações multidisciplinares, de Cálculo Diferencial e Computacional, segmentadas em três estudos de caso.

*Caso I* – Este estudo relaciona a taxa de crescimento de uma planta em função no nível de precipitação. Para Song et al (2016), as alterações no clima têm ampla interferência nos ciclos de carbono e água do ecossistema, modificando as permutas gasosas das plantas. Em seus experimentos de manipulação de mudanças globais foi revelado respostas ecofisiológicas, indicando que o tamanho de uma planta, em aquecimento climático, varia com a precipitação  $x$  (mm), em concordância com a Equação (10).

$$B(x) = \frac{40x}{\sqrt{1+x^2}} + 20 \quad (10)$$

Sendo  $B$ , o tamanho da planta em metro (m).

A taxa de variação do crescimento, em relação à precipitação, pode ser estimada pela Derivada 1ª da Equação (10), de acordo com a Equação (11).

$$\frac{dB}{dx} = \frac{40}{\sqrt{1+x^2}} - \frac{40x^2}{\sqrt{(1+x^2)^3}} \quad (11)$$

Para uma precipitação tendendo a 62.75 mm, o tamanho da planta tenderá a 59.9949 m, conforme estimado por Song et al (2016). Este resultado pode ser comprovado fazendo

$$\lim_{x \rightarrow 62.75} \frac{40x}{\sqrt{1+x^2}} + 20 = 59.9949 \quad (12)$$

Analisando a Figura 12, percebe-se que o *Script 3* foi regulado para calcular a Derivada 1ª da Equação (10).

Figura 12. *Script 3* – ajustado à Equação (10).

```

1  # -*- coding: utf-8 -*-
2  """
3  Created on Sat Jul  2 17:07:34 2022
4
5  @author:
6  GMSL - Gradiente de Modelagem Matemática e Simulação Computacional
7  Cálculo de Derivadas
8  """
9
10 # Importando Bibliotecas
11 import sympy as sy
12 # Declarando a variável simbólica: x
13 x = sy.Symbol('x')
14 # Declarando a Função: f(x)
15 f = (40*x/sy.sqrt(1+x**2))+20
16 # Visualizando f
17 print('f =',f)
18 # Derivada 1ª de f em relação a x: d1f
19 d1f = sy.diff(f,x)
20 # Visualizando d1f
21 print('d1f =',d1f)

```

Fonte: Autoras(es).

O resultado encontrado está de acordo com a Equação (11).

$$d1f = -40 * x ** 2 / (x ** 2 + 1) ** (3/2) + 40 / \text{sqrt}(x ** 2 + 1)$$

A Figura 13 exibe o *Script 1*, regulado para calcular o Limite da Equação (10), avaliando a sua tendência, em conformidade com a Expressão (12).

Figura 13. *Script 1* – ajustado à Equação (10).

```

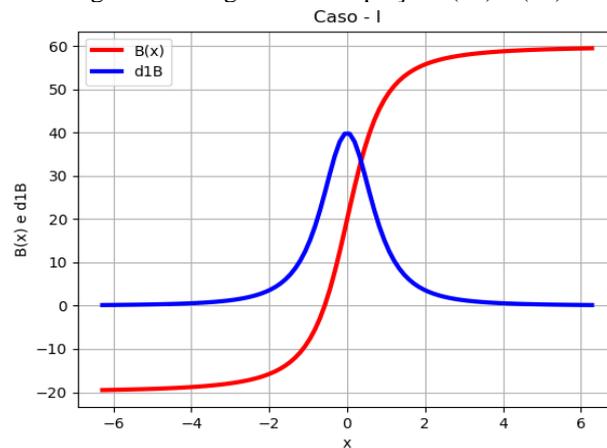
1  # -*- coding: utf-8 -*-
2  """
3  Created on Sat Jul  2 16:53:50 2022
4
5  @author:
6  GMSL - Gradiente de Modelagem Matemática e Simulação Computacional
7  Cálculo de Limites
8  """
9
10 # Importando Bibliotecas
11 import sympy as sy
12 # Declarando a variável simbólica: x
13 x = sy.Symbol('x')
14 # Declarando a Função: f(x)
15 f = (40*x/sy.sqrt(1+x**2))+20
16 # Visualizando f
17 print('-----')
18 print('f =',f)
19 # Valor da Tendência: a
20 # Valor do Limite de f para x --> a: L
21 a = 62.75
22 L = sy.Limit(f,x,a)
23 # Visualizando L
24 print("L =",L)
25 print('-----')
26

```

Fonte: Autoras(es).

As dinâmicas da relação  $B = f(x)$  e da sua taxa de variação, estão na Figura 14.

Figura 14. Diagrama das Equações (10) e (11).



Fonte: Autoras(es).

*Caso II* – Este estudo relaciona o Valor Médio Quadrático da Corrente Elétrica. Fundamentado em Costa et al (2021), entende-se que a transmissão eficiente em um Sistema Elétrico está associada à Corrente Elétrica ( $i$ ) do circuito CA (Corrente Alternada). Essa Corrente, frequentemente, é configurada em um modelo de onda senoidal, cujo Valor Médio Quadrático ( $i_{vmq}$ ) pode ser calculado pela Equação (13).

$$i_{vmq} = \sqrt{\int_0^t [10e^{-t} \text{sen}(2\pi t)]^2 dt} \quad (13)$$

Sendo,  $t$  o instante final de avaliação do Sistema de Energia Elétrica.

A Figura 15 apresenta o *Script* 4, ajustado para resolver a Equação (13), no intervalo  $t = [0, 1]$ s.

Figura 15. Script 4 – ajustado à Equação (13).

```

1 # -*- coding: utf-8 -*-
2 """
3 Created on Sat Jul  2 17:12:08 2022
4
5 @author:
6 GR*SC - Gradiente de Modelagem Matemática e Simulação Computacional
7 Cálculo de Integrais
8 """
9 # Importando Bibliotecas
10 import sympy as sy
11 # Declarando a variável simbólica: x
12 t = sy.Symbol('t')
13 # Declarando a Função: f(x)
14 f = 10*sy.exp(-1*t)*sy.sin(2*sy.pi*t)
15 # Visualizando f
16 print('f =',f)
17 # Integral de f(x) em relação a x: Ifx
18 # Limites de Integração: [a,b]
19 a = 0
20 b = 1
21 Ifx = sy.integrate(f,(t, a, b))
22 i_vmq = sy.sqrt(Ifx)
23 # Visualizando Ifx
24 print('Ifx = ',Ifx)
25 # Visualizando i_vmq
26 print('i_vmq =',i_vmq)
27

```

Fonte: Autoras(es).

Os resultados obtidos são

$$f = 10 * \exp(-t) * \sin(2 * \pi * t)$$

$$Ifx = -20 * \pi / (e + 4 * e * \pi ** 2) + 20 * \pi / (1 + 4 * \pi ** 2)$$

Para a Integral Indefinida tem-se

$$i\_vmq = \sqrt{-10 * \sin(2 * \pi * t) / (\exp(t) + 4 * \pi ** 2 * \exp(t)) - 20 * \pi * \cos(2 * \pi * t) / (\exp(t) + 4 * \pi ** 2 * \exp(t))} \quad (14)$$

Para a Integral Definida tem-se

$$i\_vmq = \sqrt{-20 * \pi / (e + 4 * e * \pi ** 2) + 20 * \pi / (1 + 4 * \pi ** 2)}$$

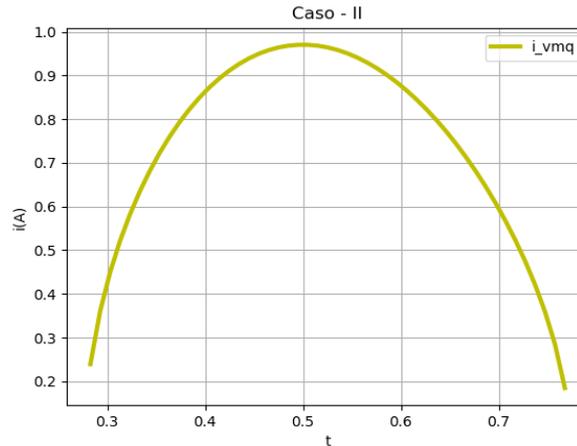
Ou seja,

$$Ifx = 0.981197$$

$$i\_vmq = 0.990554 A.$$

A representação gráfica da Equação (14) pode ser visualizada na Figura 16.

Figura 16. Diagrama da Equação (14).



Fonte: Autoras(es)

Para a construção do gráfico da Equação (14) foi utilizado o *Script 2*, adaptado, para esta expressão, como mostra a Figura 17.

Figura 17. *Script 2* – adaptado à Equação (14).

```

1  # -*- coding: utf-8 -*-
2  """
3  Created on Sun Jul  3 10:41:45 2022
4
5  @author:
6  BQ*SC - Gradiente de Modelagem Matemática e Simulação Computacional
7  Representação Gráfica de funções
8  """
9  # Importando Bibliotecas
10 import numpy as np
11 import matplotlib.pyplot as plt
12 # Declarando o intervalo de análise
13 # t = np.linspace(início, fim, número de pontos)
14 t = np.linspace(0,1,100)
15 # Declarando as Funções: f(x) e Ifx
16 f = np.sqrt(-10*np.sin(2*np.pi*t)/(np.exp(t)+4*np.pi**2*np.exp(t))-
17           20*np.pi*np.cos(2*np.pi*t)/(np.exp(t)+4*np.pi**2*np.exp(t)))
18 # Representação Gráfica
19 # Gráfico da função primitiva: f
20 plt.plot(t,f,'y',label='i_vmq',linewidth=3)
21 plt.xlabel('t')
22 plt.ylabel('i(A)')
23 plt.title('Caso - II')
24 plt.legend()
25 plt.grid(True)
26

```

Fonte: Autoras(es)

*Caso III* – Este estudo refere-se à análise de modelos que interpretam a eficiência de Custo, Receita e Lucro das unidades, para a tomada de decisão (Ashrafi e Kaleibar, 2017). Para os autores a eficiência de financeira pode ser definida como a relação entre os Custos a Receita.

Suponha que a taxa de variação da Receita de uma empresa possa ser estimada pela Equação (15).

$$\frac{dR}{dt} = -t^2 + 11t \quad (15)$$

Sendo,  $R$  o valor (em \$  $10^6$ ) da Receita Bruta em venda daqui a  $t$  anos.

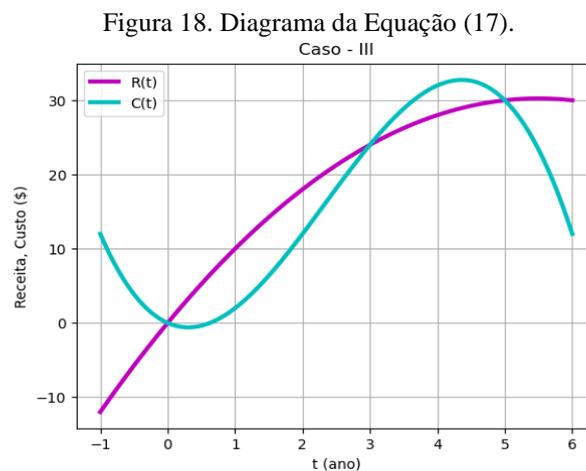
Da mesma forma, a taxa de variação dos Custos (em \$  $10^6$ ) pode ser calculada pela Equação (16).

$$\frac{dC}{dt} = -t^3 + 7t^2 - 4t \quad (16)$$

Em conformidade com Ashrafi e Kaleibar (2017), a eficiência do Lucro Total (em \$  $10^6$ ) pode ser evidenciada pela Equação (17).

$$L(t) = \int_{t_0}^{t_1} [-t^2 + 11t] dt - \int_{t_0}^{t_1} [-t^3 + 7t^2 - 4t] dt \quad (17)$$

Os comportamentos das Equações (15) e (16), podem ser examinados na Figura 18.



Fonte: Autoras(es)

Para a construção do gráfico da Equação (17) foi utilizado o *Script 2*, adaptado, para esta expressão, como mostra a Figura 19.

Figura 19. *Script 2* – adaptado à Equação (17).

```

1  # -*- coding: utf-8 -*-
2  """
3  Created on Sun Jul  3 10:41:45 2022
4
5  @author:
6  # Instituto de Matemática e Física - Universidade Federal do Rio de Janeiro
7  Representação Gráfica de funções
8  """
9  # Importando Bibliotecas
10 import numpy as np
11 import matplotlib.pyplot as plt
12 # Declarando o intervalo de análise
13 # t = np.linspace(início, fim, número de pontos)
14 t = np.linspace(-1,6,100)
15 # Declarando as Funções Receita, Custo e Lucro
16 R = -t**2 + 11*t
17 C = -t**3 + 7*t**2 - 4*t
18 L = R - C
19 # Representação Gráfica
20 # Gráfico das funções R e C
21 plt.figure(1)
22 plt.plot(t,R,'m',Label='R(t)',Linewidth=3)
23 plt.plot(t,C,'c',Label='C(t)',Linewidth=3)
24 plt.xlabel('t (ano)')
25 plt.ylabel('Receita, Custo ($)')
26 plt.title('Caso - III')
27 plt.legend()
28 plt.grid(True)
29 # Gráfico da função L
30 plt.figure(2)
31 plt.plot(t,L,'r',Label='L(t) = R(t) - C(t)',Linewidth=3)
32 plt.xlabel('t(ano)')
33 plt.ylabel('Lucro ($)')
34 plt.title('Caso - III')
35 plt.legend()
36 plt.grid(True)
37

```

Fonte: Autoras(es)

Para determinação do Lucro, em função de tempo, deve-se estabelecer os valores para o intervalo de análise, ou seja,  $[t_0, t]$ . Esses valores serão utilizados como limites para as Integrais, em conformidade com a Equação (17). Percebe-se, visualizando a Figura (18), que há três interseções entre as funções  $R(t)$  e  $C(t)$ . Estes pontos de interseção definem os intervalos de análise e foram determinados da seguinte maneira:

$$R(t) = C(t)$$

Ou ainda,

$$R(t) - C(t) = 0.$$

O *Script 5* foi implementado para resolver equações, ou seja, encontrar o conjunto solução ( $S$ ) da variável independente responsável por zerar uma determinada função.

A Figura 20 apresenta o *Script 5* estruturado resolver a Equação (18).

$$L(t) \rightarrow R(t) - C(t) = 0 \tag{18}$$

Figura 20. Script 2 – adaptado à Equação (18).

```

1  # -*- coding: utf-8 -*-
2  """
3  Created on Tue Jul  5 15:10:08 2022
4
5  @author:
6  GMSC - Gradiente de Modelagem Matemática e Simulação Computacional
7  Resolução de Equações - Encontrar Raízes
8  """
9  # Importando Biblioteca
10 import sympy as sy
11 # Declarando a variável simbólica: t
12 t = sy.symbols('t')
13 print('Equação a ser resolvida:')
14 # Declarando a Equação a ser resolvida: L = R - C
15 R = -t**2 + 11*t
16 C = -t**3 + 7*t**2 - 4*t
17 L = R - C
18 print('L =', L)
19 # Conjunto Solução: S
20 S = sy.solve(L)
21 print('Solução da Equação:')
22 print('S =', S)
23

```

Fonte: Autoras(es)

Os resultados obtidos são

$$L = t^3 - 8t^2 + 15t$$

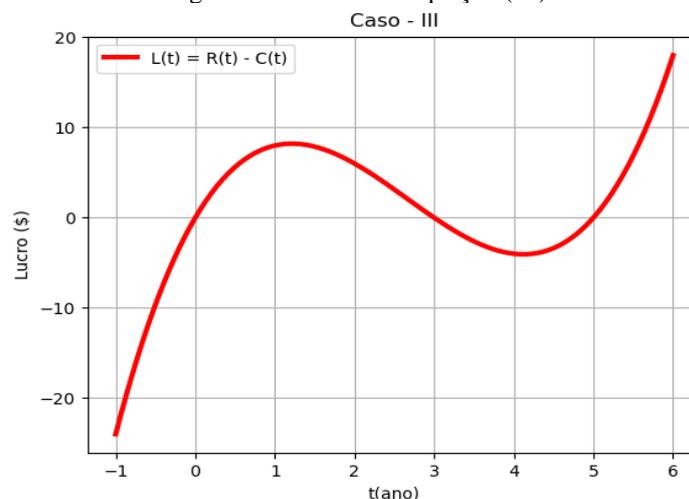
$$S = [0, 3, 5]$$

Ou seja,

$$L(t) = t^3 - 8t^2 + 15t \tag{19}$$

O comportamento da Equação (19), pode ser observado na Figura 21.

Figura 21. Gráfico da Equação (21).



Fonte: Autoras(es).

De acordo com o gráfico da Figura 21, percebe-se que no intervalo,  $t_p = [0, 3]$  o Lucro é positivo e, no intervalo  $t_n = [3, 5]$ , o Lucro é negativo (prejuízo).

Essa informação é corroborada pela análise da Figura 18, em que no intervalo  $t_p = [0, 3]$  a curva da Função Receita está acima da curva da Função Custo e, no intervalo  $t_n = [3, 5]$  a curva da Função Receita está abaixo da curva da Função Custo.

Diante disso, é presumível que, o Lucro Total pode ser contabilizado como o somatório das áreas delimitadas pelas curvas  $R(t)$  e  $C(t)$ , no intervalo  $t = [0, 5]$ , limitantes mínimo e máximo da interseção entre elas, conforme a Equação (19) e ilustrado na Figura 22.

Figura 22. Script 4 – ajustado à Equação (17).

```

1  # -*- coding: utf-8 -*-
2  """
3  Created on Sat Jul  2 17:12:08 2022
4
5  @author:
6  GWSS - Gradiente de Modelagem Matemática e Simulação Computacional
7  Cálculo de Integrais
8  """
9
10 # Importando Bibliotecas
11 import sympy as sy
12 # Declarando a variável simbólica: x
13 t = sy.Symbol('t')
14 # Declarando as Funções: R(t) e C(t)
15 R = -t**2 + 11*t
16 C = -t**3 + 7*t**2 - 4*t
17 # Visualizando R e C
18 print('R =', R)
19 print('C =', C)
20 # Limites de Integração: [a,b]
21 a = 0.0
22 b = 5.0
23 # Integral de R em relação a t: IRt
24 IRt = sy.integrate(R, (t, a, b))
25 # Integral de C em relação a t: ICt
26 ICt = sy.integrate(C, (t, a, b))
27 # Arredondando: round(Variável, nº de casas)
28 # Visualizando IRt
29 print('IRt =', round(IRt, 2))
30 # Visualizando ICt
31 print('ICt =', round(ICt, 2))
32 # Valor do Lucro: L = IRt - ICt
33 L = IRt - ICt
34 # Visualizando o Valor do Lucro
35 print('L =', round(L, 2))

```

Fonte: Autoras(es)

Os resultados obtidos são

$$\begin{aligned}
 R &= -t ** 2 + 11 * t \\
 C &= -t ** 3 + 7 * t ** 2 - 4 * t \\
 IRt &\cong 95.83 \\
 ICt &\cong 85.41 \\
 L &\cong 10.42
 \end{aligned}$$

A Inteligência Computacional permite resolver  $n - vezes$ , Cálculos de Limites, Derivadas e de Integrais, com altíssima precisão e acurácia. Assim sendo, a aferição das áreas, delimitadas pelas curvas  $R(t)$  e  $C(t)$ , será executada em duas formas.

As Figuras 23 e 24, exibem essas resoluções considerando os seguintes limites de Integração:

$$\text{Intervalo I} \rightarrow t_p = [0, 3];$$

$$\text{Intervalo II} \rightarrow t_n = [3, 5].$$

Considerando essas duas circunstâncias, as observações sobre o comportamento do Lucro estarão mais evidenciadas.

Figura 23. Script 4 – Equação (17), intervalo I.

```

1  # -*- coding: utf-8 -*-
2  """
3  Created on Sat Jul  2 17:12:08 2022
4
5  @author:
6  CRISTIANE - Graduada em Modelagem Matemática e Simulação Computacional
7  Cálculo de Integrais
8  """
9  # Importando Bibliotecas
10 import sympy as sy
11 # Declarando a variável simbólica: x
12 t = sy.Symbol('t')
13 # Declarando as Funções: R(t) e C(t)
14 R = -t**2 + 11*t
15 C = -t**3 + 7*t**2 - 4*t
16 # Visualizando R e C
17 print('R =', R)
18 print('C =', C)
19 # Limites de Integração: [a,b]
20 a = 0.0
21 b = 3.0
22 # Integral de R em relação a t: IRt
23 IRt = sy.integrate(R, (t,a,b))
24 # Integral de C em relação a t: ICt
25 ICt = sy.integrate(C, (t,a,b))
26 # Arredondando: round(Variável, nº de casas)
27 # Visualizando IRt
28 print('IRt =', round(IRt,4))
29 # Visualizando ICt
30 print('ICt =', round(ICt,4))
31 # Valor do Lucro: L = IRt - ICt
32 L = IRt - ICt
33 # Visualizando o Valor do Lucro
34 print('L =', round(L,2))
35

```

Fonte: Autoras(es).

Os resultados obtidos são

$$R = -t ** 2 + 11 * t$$

$$C = -t ** 3 + 7 * t ** 2 - 4 * t$$

$$IRt = 40.5000$$

$$ICt = 24.7500$$

$$L = 15.75$$

Figura 24. Script 4 – Equação (17), intervalo II.

```

1  # -*- coding: utf-8 -*-
2  """
3  Created on Sat Jul  2 17:12:08 2022
4
5  @author:
6  GUST - Gradiente de Modelagem Matemática e Simulação Computacional
7  Cálculo de Integrais
8  """
9  # Importando Bibliotecas
10 import sympy as sy
11 # Declarando a variável simbólica: x
12 t = sy.Symbol('t')
13 # Declarando as Funções: R(t) e C(t)
14 R = -t**2 + 11*t
15 C = -t**3 + 7*t**2 - 4*t
16 # Visualizando R e C
17 print('R =', R)
18 print('C =', C)
19 # Limites de Integração: [a,b]
20 a = 3.0
21 b = 4.0
22 # Integral de R em relação a t: IRt
23 IRt = sy.integrate(R, (t, a, b))
24 # Integral de C em relação a t: ICt
25 ICt = sy.integrate(C, (t, a, b))
26 # Arredondando: round(Variável, nº de casas)
27 # Visualizando IRt
28 print('IRt =', round(IRt, 4))
29 # Visualizando ICt
30 print('ICt =', round(ICt, 4))
31 # Valor do Lucro: L = IRt - ICt
32 L = IRt - ICt
33 # Visualizando o Valor do Lucro
34 print('L =', round(L, 2))
35

```

Fonte: Autoras(es).

Os resultados obtidos são

$$R = -t ** 2 + 11 * t$$

$$C = -t ** 3 + 7 * t ** 2 - 4 * t$$

$$IRt = 26.1667$$

$$ICt = 28.5833$$

$$L = -2.42$$

#### 4 CONCLUSÃO

A crescente evolução tecnológica exige um constante aperfeiçoamento científico. A consciência matemática contribui para que essa evolução acrescente uma concepção crítica e reflexiva sobre a mecânica da Natureza, em todos os seus aspectos, sociais, ambientais, físicos, químicos, biológicos, entre outros.

Esse artigo transfere uma estratégia que permite otimizar a execução do Cálculo Diferencial e Integral, sem reduzir a relevância algébrica dessa esplêndida criação da humanidade.

Elaborada, inicialmente por Gottfried Wilhelm Leibniz e Isaac Newton, o Cálculo Diferencial e Integral foi incrementado e obteve uma interface computacional que estimula suas aplicações em modelos interdisciplinares.

Fundamentado pela Linguagem PYTHON de programação, foram apresentados *scripts* que operam Limites, Derivadas e Integrais. Essa Modelagem Matemática-Computacional diminui a possibilidade de erros algébricos e maximiza a compreensão do fenômeno estudado.

Três estudos foram retratados. Todos apresentaram elevados níveis de precisão e acurácia em seus resultados analíticos e gráfico.

Para trabalhos futuros, estima-se a implementação de modelos baseados em funções com múltiplas variáveis independentes, para análises de multicritérios.

### **AGRADECIMENTOS**

Ao Instituto Federal de Educação, Ciência e Tecnologia do Pará – IFPA *Campus* Ananindeua, pela administração do Bacharelado em Ciência e Tecnologia desse *campus*. E ao grupo de pesquisa Gradiente de Modelagem Matemática e Simulação Computacional – GM<sup>2</sup>SC, pela gestão e motivação científica de Discentes e Docentes.

**REFERÊNCIAS**

ASHRAFI, A.; KALEIBAR, M. Mansouri. Cost, Revenue and Profit Efficiency Models in Generalized Fuzzy Data Envelopment Analysis. *Fuzzy Information and Engineering* - Elsevier. <https://doi.org/10.1016/j.fiae.2017.06.007>. 2017.

BASTIAS, Daniela A.; PINO-FAN, Luis R.; MEDRANO, Iván G.; CASTRO, Walter F. Epistemic Criteria for Designing Limit Tasks on a Real Variable Function. *Scientific Electronic Library Online*. Bolema, Rio Claro (SP), v. 35, n. 69, p. 179-205. ISSN 1980-4415 DOI: <http://dx.doi.org/10.1590/1980-4415v35n69a0>. 2021.

BURGOS, María; BUENO, Seydel; PÉREZ, Olga; Godino, Juan D. Onto-semiotic complexity of the Definite Integral. Implications for teaching and learning Calculus REDIMAT – Journal of Research in Mathematics Education, 10(1), 4-40. doi:10.17583/redimat.2021.6778. 2021.

CARVALHO, João P. A.; MACÊDO, Josué A. de; LOPES, Lailson dos R. P. Some Applications of Differential and Integral Calculus. *Research, Society and Development*, [S.l.], v. 10, n. 8, p. e22410817220. DOI: 10.33448/rsd-v10i8.17220. Disponível em: <https://rsdjournal.org/index.php/rsd/article/view/17220>. 2021.

COSTA, Denis C. L., COSTA, Heictor A. de O., & NEVES, Lucas P. Métodos Matemáticos Aplicados nas Engenharias via Sistemas Computacionais. SINEPEM-IFPA. 2019.

COSTA, Denis C. L.; COSTA, Heictor A. de O.; CASTRO, Ana P. S.; CRUZ, Edson C.; AZANCORT NETO, Julio L.; CRUZ, Brennus C. C. da. The dimensions of Mathematical and Computational Modeling prescribed to Environmental Management. *Research, Society and Development*, [S. l.], v. 9, n. 10, p. e6939109013. DOI: 10.33448/rsd-v9i10.9013. Disponível em: <https://rsdjournal.org/index.php/rsd/article/view/9013> 2020.

COSTA, Denis C. L., COSTA, Heictor A. de O., Silva, Hugo C. M. da, & Silva, Silvio T. T. da. Matemática Computacional Aplicada à Ciência e Tecnologia. Belém, PA, SINEPEM-IFPA. 2021.

COSTA, Heictor A. de O.; COSTA, Denis C. L.; MENESES, Lair A. de. Interdisciplinarity Applied to the Optimized Dispatch of Integrated Electricity and Natural Gas Networks using the Genetic Algorithm. *Research, Society and Development*, [S. l.], v. 10, n. 2, p. e42110212641. DOI: 10.33448/rsd-v10i2.12641. Disponível em: <https://rsdjournal.org/index.php/rsd/article/view/12641>. 2021.

FULINI, Márcio A. História do Cálculo Diferencial e Integral. Universidade Federal de São João Del-Rei – UFSJ Núcleo de Educação à Distância Departamento de Matemática e Estatística – DEMAT. <http://hdl.handle.net/123456789/86>. 2016.

EVES, Howard. Introdução à História da Matemática. Tradução HYGINO H. Domingues. 5ª ed. Campinas: Editora da Unicamp, 2012.

KETKAR, Nikhil. Deep Learning with PYTHON – A Hands-on Introduction. Apress. ISBN-13 (pbk): 978-1-4842-2765-7. ISBN-13 (electronic): 978-1-4842-2766-4. DOI 10.1007/978-1-4842-2766-4. Springer Science+Business Media New York. 2017.

LEITHOLD, Louis. *Cálculo com Geometria Analítica*. Tradução: Cyro de Carvalho Patarra. São Paulo: Harbra, 1994.

LUTZ, Mark e ASCHER, David. *Aprendendo PYTHON*. Tradução - João Tortello. Porto Alegre: Bookman. ISBN: 978-85-7780-013-1. 2<sup>a</sup> ed. 2007.

SONG, Bing; NIU, Shuli; WAN, Shiqiang. Precipitation regulates plant gas exchange and its long-term response to climate change in a temperate grassland. *Journal of Plant Ecology*. Volume 9, Issue 5, Pages 531–541, <https://doi.org/10.1093/jpe/rtw010>. 2016.

MATHWORKS. MathWorks leading developer of Mathematical Computing software for Engineers and Scientists. <https://www.mathworks.com/>. 2021.

VERÓN, Manuel A.; GIACOMONE, Belén. Análise dos significados do conceito de Diferencial de uma perspectiva Ontosemiótica. *Revemop - Revista científica da Universidade Federal de Ouro Preto – UFOP/PPGEDMAT*. v. 3, p. e202109. DOI: <https://doi.org/10.33532/revemop.e202109>. 2021.

VILLEGAS, Mauricio V. *Física Integrada con Cálculo*. Ibagué: Universidad de Ibagué. 492 p. ISBN Digital: 978-958-754-112-0. 2014.